






## Article

# Trajectory Planning of Aerial Robotic Manipulator Using Hybrid Particle Swarm Optimization

Suping Zhao <sup>1,2</sup>, Chaobo Chen <sup>1,2,\*</sup>, Jichao Li <sup>1,2</sup>, Song Gao <sup>1,2,\*</sup> and Xinxin Guo <sup>3</sup><sup>1</sup> Autonomous Unmanned Systems Research Laboratory, Xi'an Technological University, Xi'an 710021, China<sup>2</sup> School of Electronic Information Engineering, Xi'an Technological University, Xi'an 710021, China<sup>3</sup> Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen 518005, China

\* Correspondence: chenchabo@xatu.edu.cn (C.C.); gaos@xatu.edu.cn (S.G.)

**Abstract:** The trajectory planning of an aerial robotic manipulator system is studied using Hybrid Particle Swarm Optimization (HPSO). The aerial robotic manipulator is composed of an unmanned aerial vehicle (UAV) base and a robotic manipulator. The robotic manipulator is dynamically singular. In addition, strong coupling exists between the UAV base and the robotic manipulator. To overcome the problems, the trajectory planning is studied in the joint space using HPSO. HPSO combines superiorities of PSO and GA (Genetic Algorithm), prohibiting particles from becoming trapped in a local minimum. In addition, the control parameters are self-adaptive and contribute to fast searching for the global optimum. The trajectory planning problem is converted into a parameter optimization problem. Each joint trajectory is parameterized with a Bézier curve. The HPSO is implemented to optimize joint trajectories, satisfying specific objectives and imposed constraints. Numerical simulations are also carried out to validate the effectiveness of the proposed method.

**Keywords:** aerial robotic manipulator; unmanned aerial vehicle; Bézier curve; particle swarm optimization



**Citation:** Zhao, S.; Chen, C.; Li, J.; Gao, S.; Guo, X. Trajectory Planning of Aerial Robotic Manipulator Using Hybrid Particle Swarm Optimization. *Appl. Sci.* **2022**, *12*, 10892. <https://doi.org/10.3390/app122110892>

Academic Editors: Wei Huang and Dong Zhang

Received: 20 September 2022

Accepted: 21 October 2022

Published: 27 October 2022

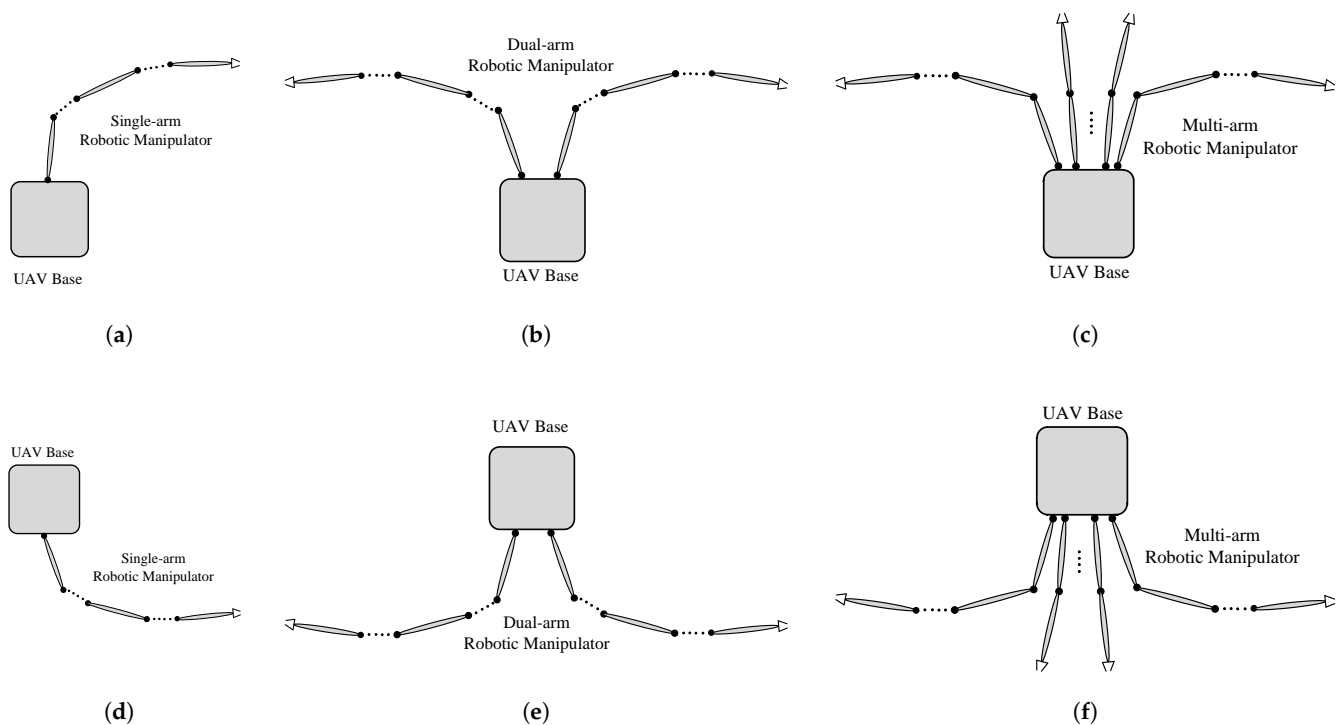
**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**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

Aerial Robotic Manipulators (AEROMs) [1,2] are combined systems, composed of mobile platforms and robotic manipulators. The mobile platform system is usually a UAV, while the robotic manipulator system includes one or several arms. The increasing demands of aerial pipeline maintenance, bridge inspection, fruit picking, etc. call for application of aerial robotic manipulators to perform tasks in aerial environment. Examples include “Aerial Cognitive Integrated Multi-task Robotic System with Extended Operation Range and Safety” [3], “Aerial Robotic Training for the Next Generation of European Infrastructure and Asset Maintenance Technologies” [4], “Hybrid Flying-rolling with Snake-arm Robot for Contact Inspection” [5] and “Aerial Robotic System Integrating Multiple Arms” [6]. In these programs, aerial missions are performed with various categories of aerial robotic manipulators, as shown in Figure 1. In Figure 1a–c, the robotic manipulator locates on the top of UAV base, opposite to Figure 1d–f. A category of combined system is of great interest to both scientists and practitioners. As shown in Figure 1a,d, the system consists of a UAV and a single-arm robotic manipulator. This is because the modelling is with lower difficulty and the system is easily controlled. However, the combined system still exhibits some special characteristics. Strong coupling exists between the robotic manipulator and the UAV. Moreover, the manipulator is dynamically singular. Accordingly, specific trajectory planning strategies have to be investigated to cope with the aforementioned problems.



**Figure 1.** Two-dimensional (2-D) sketch of aerial robotic manipulator composed of a UAV base and a robotic manipulator, where the manipulator denotes (a,d) one single arm, (b,e) two arms or (c,f) multiple arms.

The developments of an AEROM system includes structure design, mathematical modelling, trajectory planning, control, etc. Actually, an AEROM model includes two parts, the kinematic model and the dynamic model. The trajectory planning is studied on the basis of AEROM kinematics [7,8], while the control development is based on both kinematics and dynamics [9]. The trajectory planning aims at optimizing joint trajectories, end-effector trajectories or UAV trajectories. As a matter of fact, the trajectory planning has been well developed for robots in outer space [7,8], where the satellite base is usually under free-floating mode. Two common trajectory planning techniques are respectively based on inverse kinematics and forward kinematics. For the technique using inverse kinematics, various concepts have been proposed, including Generalized Jacobian Matrix [9], Enhanced Disturbance Map [10], Path Dependent/Independent workspace [11] and Reaction Null Space [12]. However, this approach can result in singularities. This shortage is well avoided in the technique using forward kinematics, where the trajectory planning issue is transformed into an optimization issue. Intelligent Optimization Algorithms (IOAs) are utilized to solve the optimization issue. The procedure based on forward kinematics is followed. In order to perform optimization, joint trajectories are firstly parametrized with mathematical functions such as sinusoidal functions [13,14], polynomial functions [15,16], spline functions [17,18], etc. Second, a few function coefficients are evaluated according to initial conditions, including joint angles, joint velocities, joint accelerations or EE pose at initial time or final time. Third, values of unknown function coefficients are searched via IOAs, in conjunction with fitness functions and constraint conditions. Common IOAs include evolutionary algorithms like GA [19,20] and Differential Evolution (DE) algorithm [21,22], swarm intelligence algorithms like PSO [23,24] and Ant Colony Optimization [25,26], Neural Network Algorithm [27,28], etc. Fitness functions are usually minimum path length, time spent or energy cost. Constraint conditions denote limits on joint angles, joint velocities and joint accelerations.

This paper proposes a novel approach for trajectory planning of an aerial robotic manipulator, coping with multiple constraints and objectives. The Bézier curves are utilized

to represent joint trajectories, with the superiority of simply coping with constraints. HPSO is implemented to search for optimal solution of the optimization problem. HPSO is an improved PSO combining the superiorities of PSO and GA. Furthermore, the control parameters of HPSO have self-adaptive characteristics.

The paper is organized as follows. Section 2 presents a kinematic model and dynamic model of the aerial robotic manipulator system. Afterwards, the trajectory planning problem is discussed in Section 3. The delineation of the problem as an optimization issue is also shown in this section. Section 4 develops the HPSO, the selection of cost functions and the handling of constraints. Section 5 tests the performance of the proposed method, including discussion about the simulation results. Conclusions are listed in Section 6.

### 2. Kinematics and Dynamics of Aerial Robotic Manipulator

Figure 2 shows the 2-D sketch of an AEROM system composed of  $n + 1$  bodies. The UAV mobile platform denotes the base and is remarked as body  $B_0$ . The robotic manipulator is a single arm with  $n$  bodies and an EE. Adjacent bodies are connected with a revolute joint with one degree of freedom (DOF). Table 1 lists meanings of symbols and variables in Figure 2. Axes of  $\Sigma_0$  are respectively parallel to axes of  $\Sigma_I$  at the initial time of the system maneuver. The axes of  $\Sigma_e$  are also parallel to the axes of  $\Sigma_n$  at the initial time. The frames  $\Sigma_i$  ( $i = 1, \dots, n$ ) are built using the Denavit–Hartenberg (D-H) approach [29,30]. The detailed procedure is summarized as follows:

1. Align the axis  $Z_i$  and the origin  $O_i$ .  $Z_i$  is located at the  $(i + 1)$ th joint, and  $O_i$  denotes the crosspoint of  $Z_i$  and common perpendicular of  $Z_i$  and  $Z_{i-1}$ .
2. Align the axis  $X_i$ . The  $X_i$ -axis is along the common perpendicular of  $Z_i$  and  $Z_{i-1}$ , and the direction is from  $J_i$  to  $J_{i+1}$ .
3. Align the axis  $Y_i$  according to the right-hand rule.

Along with establishment of frames  $\Sigma_i$ , four categories of D-H parameters are constructed as follows.

- $a_i$ — The distance between  $Z_i$  and  $Z_{i+1}$  along  $X_i$ .
- $\alpha_i$ — The angle from  $Z_i$  to  $Z_{i+1}$  around  $X_i$ .
- $d_i$ — The distance between  $X_{i-1}$  and  $X_i$  along  $Z_i$ .
- $\theta_i$ — The angle from  $X_{i-1}$  to  $X_i$  around  $Z_i$ .

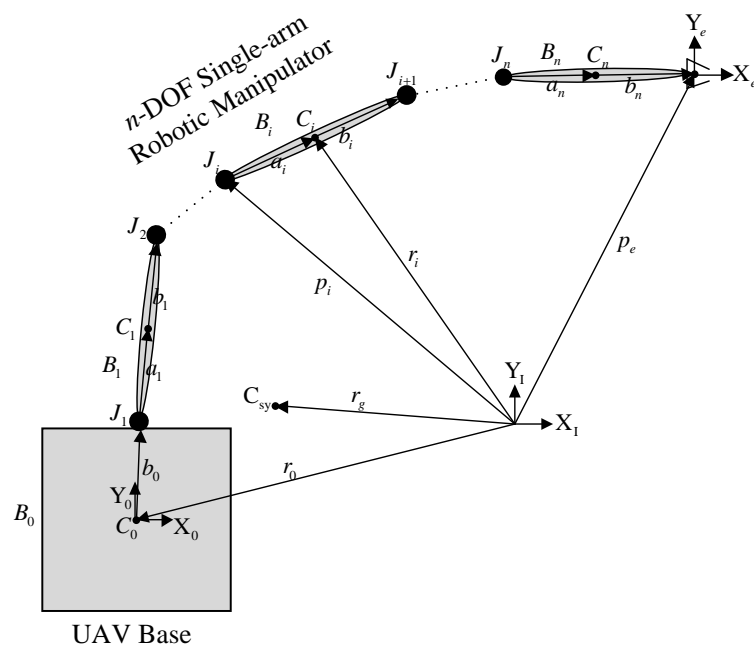


Figure 2. 2-D sketch of an AEROM system, with symbols employed for the derivation of AEROM mathematical models.

**Table 1.** Representation of symbols in Figure 2.

Symbol	Representation
$C_{sy}$	Center of mass of AEROM system
$B_i (i= 0, \dots, n)$	Body $i$ , where $B_0$ denotes UAV base and $B_i (i= 1, \dots, n)$ denotes $i$ th link of robotic manipulator
$C_i (i= 0, \dots, n)$	Center of mass of $B_i$
$J_i (i= 0, \dots, n)$	Joint $i$ , connecting $B_{i-1}$ and $B_i$
$\mathbf{a}_i \in \mathbb{R}^3 (i= 1, \dots, n)$	Vector from $J_i$ to $C_i$
$\mathbf{b}_i \in \mathbb{R}^3 (i= 0, \dots, n)$	Vector from $C_i$ to $J_{i+1}$
$\mathbf{r}_i \in \mathbb{R}^3 (i= 0, \dots, n)$	Position of $C_i$ in inertial coordinate system
$\mathbf{p}_i \in \mathbb{R}^3 (i= 1, \dots, n)$	Position of $J_i$ in inertial coordinate system
$\mathbf{r}_g \in \mathbb{R}^3$	Position of $C_{sy}$ in inertial coordinate system
$\mathbf{p}_e \in \mathbb{R}^3$	Position of end effector in inertial coordinate system
$\Sigma_i (i= 0, \dots, n)$	Coordinate system established in $B_i$ with origin $O_i$ , axes $X_i, Y_i$ and $Z_i$
$\Sigma_I$	Inertial coordinate system with origin $O_I$ , axes $X_I, Y_I$ and $Z_I$
$\Sigma_e$	Coordinate system established in end effector, with origin $O_e$ , axes $X_e, Y_e$ and $Z_e$

In this paper, the AEROM system is considered as a unique entity. Each component  $B_i (i = 0, \dots, n)$  of AEROM is assumed to be a rigid body with uniform mass distribution. The center of mass of  $B_i$  is located at the geometric center  $C_i$ . On the basis of Figure 2, positions of  $i$ th body centre and EE are depicted as:

$$\mathbf{r}_i = \mathbf{r}_0 + \mathbf{b}_0 + \sum_{k=1}^{i-1} (\mathbf{a}_k + \mathbf{b}_k) + \mathbf{a}_i \tag{1}$$

$$\mathbf{p}_e = \mathbf{r}_0 + \mathbf{b}_0 + \sum_{k=1}^n (\mathbf{a}_k + \mathbf{b}_k) \tag{2}$$

Taking the derivatives of  $\mathbf{r}_i$  and  $\mathbf{p}_e$ , the linear velocities of  $i$ th body centre and EE are then formulated as:

$$\mathbf{v}_i = \dot{\mathbf{r}}_i = \mathbf{v}_b + \boldsymbol{\omega}_b \times (\mathbf{r}_i - \mathbf{r}_0) + \sum_{k=1}^i [\mathbf{k}_k \times (\mathbf{r}_i - \mathbf{p}_k)] \dot{q}_k \tag{3}$$

$$\mathbf{v}_e = \dot{\mathbf{p}}_e = \mathbf{v}_b + \boldsymbol{\omega}_b \times (\mathbf{p}_e - \mathbf{r}_0) + \sum_{k=1}^n [\mathbf{k}_k \times (\mathbf{p}_e - \mathbf{p}_k)] \dot{q}_k \tag{4}$$

where  $\mathbf{v}_i \in \mathbb{R}^3, \mathbf{v}_e \in \mathbb{R}^3, \mathbf{v}_b \in \mathbb{R}^3$  and  $\boldsymbol{\omega}_b \in \mathbb{R}^3$ .  $\mathbf{k}_i \in \mathbb{R}^3$  denotes the unit rotational velocity of  $i$ th joint.  $\mathbf{v}_b$  and  $\boldsymbol{\omega}_b$  are, respectively, the linear velocity and angular velocity of the UAV base. Angular velocities of  $i$ th body centre and EE are calculated as:

$$\boldsymbol{\omega}_i = \boldsymbol{\omega}_b + \sum_{k=1}^i \mathbf{k}_k \dot{q}_k \tag{5}$$

$$\boldsymbol{\omega}_e = \boldsymbol{\omega}_b + \sum_{k=1}^n \mathbf{k}_k \dot{q}_k \tag{6}$$

where  $\boldsymbol{\omega}_i \in \mathbb{R}^3$  and  $\boldsymbol{\omega}_e \in \mathbb{R}^3$ . Combining Equations (4) and (6), EE velocity is given as follows:

$$\dot{\boldsymbol{\chi}}_e = \begin{bmatrix} \mathbf{v}_e \\ \boldsymbol{\omega}_e \end{bmatrix} = \mathbf{J}_b \begin{bmatrix} \mathbf{v}_b \\ \boldsymbol{\omega}_b \end{bmatrix} + \mathbf{J}_m \dot{\mathbf{q}} \tag{7}$$

where  $\mathbf{J}_b \in \mathbb{R}^{6 \times 6}$  and  $\mathbf{J}_m \in \mathbb{R}^{6 \times n}$  are, respectively, Jacobian matrices of the UAV base and the robotic manipulator.  $\mathbf{q} = [q_1, \dots, q_n] \in \mathbb{R}^n$  denotes joint configuration, where  $q_i (i = 1, \dots, n)$  denotes the  $i$ th joint angle.  $\dot{\mathbf{q}}$  denotes the velocity vector of  $n$  joints.

In the first case, the pose of UAV base is invariable (i.e., position and attitude of UAV are controlled). The first term of right-hand side in Equation (7) is  $\mathbf{0}$ , shown as:

$$\dot{\chi}_e = J_m \dot{q} \tag{8}$$

In the second case, pose of UAV base is free (i.e., position and attitude of UAV are not controlled). Assume gravity of each component of AEROM system is exactly compensated by corresponding rotors or servo-motors, the resultant of external force acting on the AEROM system is  $\mathbf{0}$ . Then, the AEROM system satisfies the conservation of momentum, shown as follows:

$$H_b \dot{\chi}_b + H_{bm} \dot{q} = \mathbf{C} \tag{9}$$

where  $\dot{\chi}_b = [v_b; \omega_b]$ .  $H_b \in \mathbb{R}^{6 \times 6}$  is inertia matrix of UAV base and has the ortho-symmetric property.  $H_{bm} \in \mathbb{R}^{6 \times n}$  is coupling inertia matrix of UAV base and robotic manipulator, and  $\mathbf{C}$  is a constant vector. The constant  $\mathbf{C}$  is usually set as  $\mathbf{0}$  [7,8,31]; the same goes for this case. Then, the UAV velocity is formulated as:  $\dot{\chi}_b = H_b^{-1} H_{bm} \dot{q}$ . According to the Equation (7) and the Conservation of Momentum (9), the EE velocity is formulated as:

$$\dot{\chi}_e = \begin{bmatrix} v_e \\ \omega_e \end{bmatrix} = (J_m - J_b H_b^{-1} H_{bm}) \dot{q} = J_g \dot{q} \tag{10}$$

The Equations (8) and (10) are, namely, differential kinematic equations of AEROM system in the above two cases. The AEROM dynamics are necessary to control the EE of robotic manipulator. The Lagrange–Euler approach [30,32,33] directly calculates the variations of the moment of inertia, vital for analysing the stability of controlling AEROM. The kinetic energy  $E_k$  and the potential energy  $E_p$  can be firstly obtained [32], formulated as:

$$E_k = \frac{1}{2} \begin{bmatrix} \dot{\chi}_b \\ \dot{q} \end{bmatrix}^T \begin{bmatrix} H_{b'} & H_{bm'} \\ H_{bm'}^T & H_m \end{bmatrix} \begin{bmatrix} \dot{\chi}_b \\ \dot{q} \end{bmatrix} \tag{11}$$

$$E_p = - \sum_{i=1}^n m_i g^T \left[ b_0 + \sum_{k=1}^i (AC_{j-1} - AC_j) \right] \tag{12}$$

where  $\dot{\chi}_b = [v_b; \omega_b]$ .  $H_{b'} \in \mathbb{R}^{6 \times 6}$  and  $H_m \in \mathbb{R}^{n \times n}$  are respectively inertia matrices of UAV base and robotic manipulator.  $H_{bm'} \in \mathbb{R}^{6 \times n}$  is the coupling inertia matrix of UAV and manipulator.  $AC_j$  are matrices related to  $b_i, p_i, r_i, k_i$ , etc.  $m_i$  denotes the mass of the  $i$ th body. Then, the dynamic equation of AEROM system can be derived through the Lagrange–Euler approach [33], shown as:

$$\begin{bmatrix} H_b & H_{bm} \\ H_{bm}^T & H_m \end{bmatrix} \begin{bmatrix} \ddot{\chi}_b \\ \ddot{q} \end{bmatrix} + \begin{bmatrix} C_b \\ C_m \end{bmatrix} + \begin{bmatrix} G_b \\ G_m \end{bmatrix} = \begin{bmatrix} F_p \\ \tau_m \end{bmatrix} \tag{13}$$

where  $H_b \in \mathbb{R}^{6 \times 6}$  and  $H_m \in \mathbb{R}^{n \times n}$  are, respectively, inertia matrices of UAV base and robotic manipulator.  $H_{bm} \in \mathbb{R}^{6 \times n}$  is the coupling inertia matrix of UAV and manipulator. The work [32] describes the difference between  $H_{b'}$  and  $H_b$ , and the difference between  $H_{bm'}$  and  $H_m$ .  $C_b \in \mathbb{R}^{6 \times 1}$  and  $C_m \in \mathbb{R}^{n \times 1}$  are respectively the Coriolis force and the centrifugal force.  $G_b \in \mathbb{R}^{6 \times 1}$  and  $G_m \in \mathbb{R}^{n \times 1}$  are gravity items related to UAV base and robotic manipulator.  $F_p \in \mathbb{R}^{6 \times 1}$  and  $\tau_m \in \mathbb{R}^{n \times 1}$  denote force/moment vectors of UAV and  $n$  joints, respectively.  $F_p$  and  $\tau_m$  both consist of two parts, the actuator force/moment and the external force/moment. The external force/moment denotes the atmosphere influence, air flow, etc. Since this paper focuses on AEROM trajectory planning through kinematics, the AEROM dynamics are not further analyzed.

### 3. Trajectory Planning of Aerial Robotic Manipulator

#### 3.1. Description of Trajectory Planning Problem

The trajectory planning problem for AEROM is studied here. The pose of UAV base is free, i.e., the second case of the aforementioned kinematics. The AEROM system has the non-holonomic characteristic, implying that the pose of AEROM relies on both inverse kinematics and coupling between UAV base and robotic manipulator. The initial, desired and actually planned poses of EE are respectively denoted as  $\chi_e^{in}$ ,  $\chi_e^{de}$  and  $\chi_e^{ac}$ . The initial, desired and actually planned poses of UAV base are respectively denoted as  $\chi_b^{in}$ ,  $\chi_b^{de}$  and  $\chi_b^{ac}$ . The initial, desired and actually planned joint configurations are respectively denoted as  $q^{in}$ ,  $q^{de}$  and  $q^{ac}$ . On the basis of aforementioned AEROM kinematics, the actually planned poses of EE and UAV base are respectively calculated as:

$$\chi_e^{ac} = \chi_e^{in} + \int_{t_0}^{t_f} J_g \dot{q} dt \tag{14}$$

$$\chi_b^{ac} = \chi_b^{in} + \int_{t_0}^{t_f} H_b^{-1} H_{bm} \dot{q} dt \tag{15}$$

The differences between the actually planned pose and desired pose of EE and UAV base are respectively given by Equations (16) and (17). The differences between actually planned joint configuration and desired joint configuration is given by Equation (18).

$$\delta\chi_e = \chi_e^{ac} - \chi_e^{de} \tag{16}$$

$$\delta\chi_b = \chi_b^{ac} - \chi_b^{de} \tag{17}$$

$$\delta q = q^{ac} - q^{de} \tag{18}$$

The trajectory planning of AEROM aims to obtain appropriate joint trajectories yielding initial values and constraints:

$$\left\{ \begin{array}{l} \chi_b(t_0) = \chi_b^0, \chi_e(t_0) = \chi_e^0, \chi_e(t_f) = \chi_e^f \\ q(t_0) = q^0, \dot{q}(t_0) = \dot{q}(t_0) = \mathbf{0} \\ q(t_f) = q^f, \dot{q}(t_f) = \dot{q}(t_f) = \mathbf{0} \\ q_{min} \leq q \leq q_{max} \\ \dot{q}_{min} \leq \dot{q} \leq \dot{q}_{max} \\ \ddot{q}_{min} \leq \ddot{q} \leq \ddot{q}_{max} \end{array} \right. \tag{19}$$

In Equation (19),  $\chi_b^0$  is the initial pose of UAV base.  $\chi_e^0$  and  $\chi_e^f$  are respectively the initial pose and the final pose of EE.  $q^0$  and  $q^f$  are the initial and the final joint configurations, respectively. In this paper,  $q_{min} = -q_{max}$ ,  $\dot{q}_{min} = -\dot{q}_{max}$ , and  $\ddot{q}_{min} = -\ddot{q}_{max}$ . According to assignments of  $\dot{q}_{max}$  and  $\ddot{q}_{max}$ , the AEROM maneuver time  $T$  can be determined as [23,24]:

$$T \geq \max \left( \frac{|\dot{q}_{max}|}{\dot{q}_{max}}, \sqrt{\frac{|\ddot{q}_{max}|}{\ddot{q}_{max}}} \right) \tag{20}$$

where  $\dot{q}_{max}$  and  $\ddot{q}_{max}$  respectively denote the maximum components of vectors  $\frac{dq}{du}$  and  $\frac{d^2q}{du^2}$ ,  $\dot{q}_{max}$  and  $\ddot{q}_{max}$  respectively denote the maximum components of vectors  $\frac{dq}{dt}$  and  $\frac{d^2q}{dt^2}$ . The symbols  $t$  and  $u$  respectively denote the time and the normalization of time. Detailed explanations about  $t$  and  $u$  are provided in Section 3.3.

Actually, the relationship between  $\chi_e$  and  $q$  can be derived through the D-H approach or integration of Equation (10), remarked as  $\chi_e = f(q)$ . For an assignment of  $q$  denoted as  $q_s$ , the equation  $\chi_{es} = f(q_s)$  works. However,  $q_s \neq f^{-1}(\chi_{es})$  in the case that the robotic manipulator is redundant. This is because  $f^{-1}(\chi_{es})$  has infinite solutions. The trajectory planning requires the UAV pose, the EE pose and the joint configuration approach to their desired values, namely  $\delta\chi_b \Rightarrow \mathbf{0}$ ,  $\delta\chi_e \Rightarrow \mathbf{0}$  and  $\delta q \Rightarrow \mathbf{0}$ .

### 3.2. Cost Functions

For an AEROM system, a large disturbance to UAV base is infeasible during the manipulator maneuver. This is because of the accuracy requirements of EE performing tasks such as aerial refuelling and maintenance of the aerial pipe. Although the controllers can be designed to compensate the disturbance, energy is limited in an aerial environment. Efficient trajectory planning contributes to optimal joint trajectories with minimum disturbance, and contributes to saving energies indirectly. This cost function is formulated as:

$$\Gamma_1 = \int_{t_0}^{t_f} \|\chi_b(t)\|_{M_1} dt \tag{21}$$

where the symbol  $\|\cdot\|$  depicts the vector norm, the symbol  $M$  depicts a weight matrix with positive definite properties. Another cost function represents angular variations of joints, formulated as:

$$\Gamma_2 = \int_{t_0}^{t_f} \|\mathbf{q}(t) - \mathbf{q}^0\|_{M_2} dt \tag{22}$$

The trajectory planning problem of AEROM is formulated as:

$$\begin{aligned} \min_{\mathbf{q}(t)} \quad & \Gamma(\mathbf{q}(t)) \\ \text{s.t.} \quad & \mathbf{g}(\mathbf{q}(t)) < \mathbf{0} \\ & \mathbf{h}(\mathbf{q}(t)) = \mathbf{0} \end{aligned} \tag{23}$$

where  $\mathbf{g}(\cdot)$  and  $\mathbf{h}(\cdot)$  represents inequality constraints and equality constraints, respectively.

### 3.3. Parametrization of Joint Trajectories

The trajectory planning problem in Equation (23) is studied as a nonlinear programming (NP) problem. The NP problem considers functional requirements of UAV disturbance and EE manipulability, equality constraint of AEROM forward kinematics, and inequality constraints of joint motions. Each joint trajectory is depicted with a Bézier curve [34,35]. The Bézier curve contributes to modelling smooth curves in computer graphics, formulated as:

$$q_i(u) = \sum_{j=0}^m b_{j,m}(u) P_{ij} \tag{24}$$

where  $P_{ij}$  is predefined points for the construction of a Bézier curve,  $b_{j,m}(u)$  is the Bernstein basis polynomial function as followed:

$$b_{j,m}(u) = C_m^j (1-u)^{m-j} u^j, \quad u \in [0, 1] \tag{25}$$

where  $C_m^j$  denotes a binomial coefficient,  $C_m^j = \frac{m!}{j!(m-j)!}$ . The symbol  $u$  denotes the normalization of time. The AEROM maneuver time is depicted as  $T = t_f - t_0$ . Combined with the form  $u = \frac{t}{T}$ , the joint trajectory is formulated as:

$$q_i(t) = \sum_{j=0}^m C_m^j \left(1 - \frac{t}{T}\right)^{m-j} \left(\frac{t}{T}\right)^j P_{ij} \tag{26}$$

The angular velocity is:

$$\dot{q}_i(t) = \frac{dq_i}{du} \frac{du}{dt} = \frac{m}{T} \sum_{j=0}^{m-1} b_{j,m-1} \left(\frac{t}{T}\right) (P_{i,j+1} - P_{ij}) \tag{27}$$

In this paper, the quintic Bézier curve is employed to parametrize joint trajectories, i.e.,  $m = 5$ .  $\mathbf{p}_i = [P_{i1}, \dots, P_{i5}] (i = 1, \dots, n)$  are designed as variables of  $i$ th joint trajectory.

According to aforementioned equations from (19) to (27), the elements  $P_{ij}(i = 1, \dots, n; j = 0, 1, \dots, 5)$  are calculated as:

$$\begin{cases} P_{i,0} = P_{i,1} = P_{i,2} = q_i^0 \\ P_{i,3} = P_{i,4} = P_{i,5} \end{cases} \quad (28)$$

In Equation (28), the value  $q_i^0$  is predefined. Once the parameter  $P_{i,5}$  is derived,  $p_i$  is determined. Namely, the  $i$ th joint trajectory is obtained.

#### 4. Hybrid Particle Swarm Optimization

PSO, proposed by Kennedy and Eberhart [36], was inspired by motions of birds or swimming fishes. PSO is a heuristic random search method with a simple principle. It easily gets into a local optimum [37] due to smaller step size, or misses the global optimum due to larger step size. For a global optimum with high efficiency, each particle should balance the exploration ability and the exploitation ability. The exploration ability contributes to global search at initial stage, while the exploitation ability focuses on local search at final stage. In order to balance the two abilities, various particle updating mechanisms are proposed such as the dynamic adaptation, the switch mechanism and the exponential form. In this paper, HPSO with self-adaptive ability is proposed. The velocity and the position of each particle in HPSO is updated as followed:

$$\begin{cases} v_{i,k} = w_{i,k}v_{i,k} + c_{1i,k}r_1(x_{b_{i,k}} - x_{i,k}) + c_{2i,k}r_2(x_{g_k} - x_{i,k}) \\ x_{i,k} = x_{i,k} + v_{i,k} \end{cases} \quad (29)$$

In Equation (29)  $i = 1, \dots, m$  and  $k = 1, \dots, N$ .  $m$  and  $N$  respectively depict the population size and the maximum iterations. The symbols  $v_{i,k}$  and  $x_{i,k}$  represent the velocity and the position of  $i$ th particle at  $k$ th iteration, respectively. In conjunction with the Section 3.3, the position  $x_{i,k}$  represents a group assignment of  $(P_{1,5}, \dots, P_{n,5})$ . The constants  $r_1$  and  $r_2$  are uniformly distributed in the region of  $[0, 1]$ .  $x_{b_{i,k}}$  represents the best position of  $i$ th particle so far, while  $x_{g_k}$  represents the best position of the population so far.  $w_{i,k}$ ,  $c_{1i,k}$  and  $c_{2i,k}$  are three control parameters with self-adaptive abilities. Inspired by the increasing abilities of the exponential function,  $w_{i,k}$ ,  $c_{1i,k}$  and  $c_{2i,k}$  are formulated as:

$$w_{i,k} = (w_0 - w_f) \cdot e^{\left(-\frac{w_0 - w_f}{g_{x_b}} \cdot \frac{i}{N}\right)} + w_f \quad (30)$$

$$c_{1i,k} = (c_{10} - c_{1f}) \cdot e^{\left(-\frac{c_{10} - c_{1f}}{g_{x_b}} \cdot \frac{i}{N}\right)} + c_{1f} \quad (31)$$

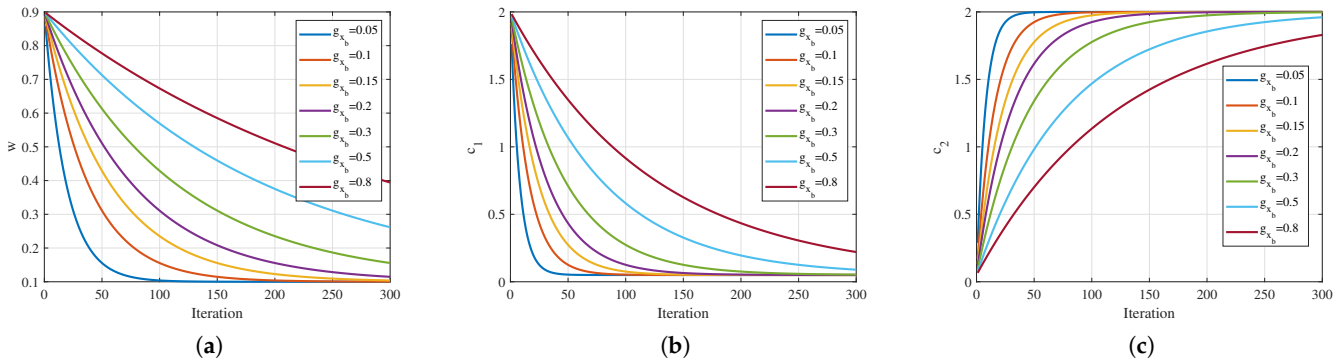
$$c_{2i,k} = (c_{20} - c_{2f}) \cdot e^{\left(-\frac{c_{20} - c_{2f}}{g_{x_b}} \cdot \frac{i}{N}\right)} + c_{2f} \quad (32)$$

where  $w_0$ ,  $c_{10}$  and  $c_{20}$  are initial values,  $w_f$ ,  $c_{1f}$  and  $c_{2f}$  are final values. The control parameters are defined in an exponential way, in order to balance the exploration ability and the exploitation ability. Furthermore, an self-adaptive strategy is introduced to adjust each control parameter.  $g_{x_b} = d(x_{g_k}, x_{b_{i,k}}) = \|x_{g_k} - x_{b_{i,k}}\|$  represents the distance between the best position of the  $i$ th particle and the global best position.

As the iterations increase,  $c_{1i,k}$  and  $c_{2i,k}$  decrease while  $w_{i,k}$  increases. Therefore, each particle has high exploration ability at early stage and high exploitation ability at final stage. Furthermore,  $g_{x_b}$  influences the variation velocities of control parameters shown in Figure 3. Large  $g_{x_b}$  makes a slow variation of  $w$ ,  $c_1$  and  $c_2$ , contributing to superior exploration ability of particles. However, small  $g_{x_b}$  contributes to fast variation of control parameters and makes particles exhibit superior exploitation ability. Actually, if the distance between a particle position and the global best position is large, searching through the whole space is



required. Then, the particle moves towards the global best position as fast as possible. If the distance is small, searching through the local space and around the global best position is required.



**Figure 3.** Variation curves of the three control parameters with different  $g_{x_b}$ , where (a)  $w$ , (b)  $c_1$ , and (c)  $c_2$ .

Utilizing the dynamic system theory, a stable system meets the sufficient condition that the magnitude of solution is less than 1. Namely, eigenvalues of coefficient matrix in Equation (29) is less smaller than 1. Then, the convergence region of system (29) is calculated as:

$$\Delta = \{(\omega, c_1, c_2) \mid \omega < 1, c_1r_1 + c_2r_2 > 0, 4\omega - 4(c_1r_1 + c_2r_2) + 4 > 0\} \quad (33)$$

Furthermore, the eigenvalues influence the system convergent behaviour as follows.

- Non-oscillatory form. If both eigenvalues are real and at least one of them is positive, the dynamic system converges in the non-oscillatory behaviour.
- Harmonic form. If both eigenvalues are complex, the dynamic system converges in the harmonic behaviour.
- Zigzagging form. If at least one of the eigenvalues, complex or real, has negative real part, the dynamic system converges in the zigzagging behaviour.

The assignment of  $g_{x_b}$  affects control parameters in (30), (31) and (32), and thus influences the convergent behaviour of the system (29). In addition, initial assignments of control parameters, i.e.,  $\omega_0, \omega_f, c_{10}, c_{1f}, c_{20}$ , and  $c_{2f}$ , should follow the convergent region.

Figure 4 shows the flow chart of HPSO for AEROM trajectory planning. Assignments are firstly provided, including particle number  $n_p$ , precision threshold  $\varepsilon$ , maximum iterations  $N$ , maximum iterations with the same fitness value  $k_{max}$ , etc. During the iteration process, positions of particles are updated if the terminal condition is not satisfied. It means that the optimum value is within the predefined region, or the iterations with the same optimum value are larger than predefined value  $k_{max}$ . The position updating mechanism combines superiorities of PSO and GA, shown in Figure 5. Firstly, the velocity and the position of each particle are updated utilizing PSO with self-adaptive control parameters. Then, particles are randomly selected for mutation or crossover with predefined probabilities. Therefore, part particles are updated twice, and HPSO is prohibited from becoming trapped in a local optimum.

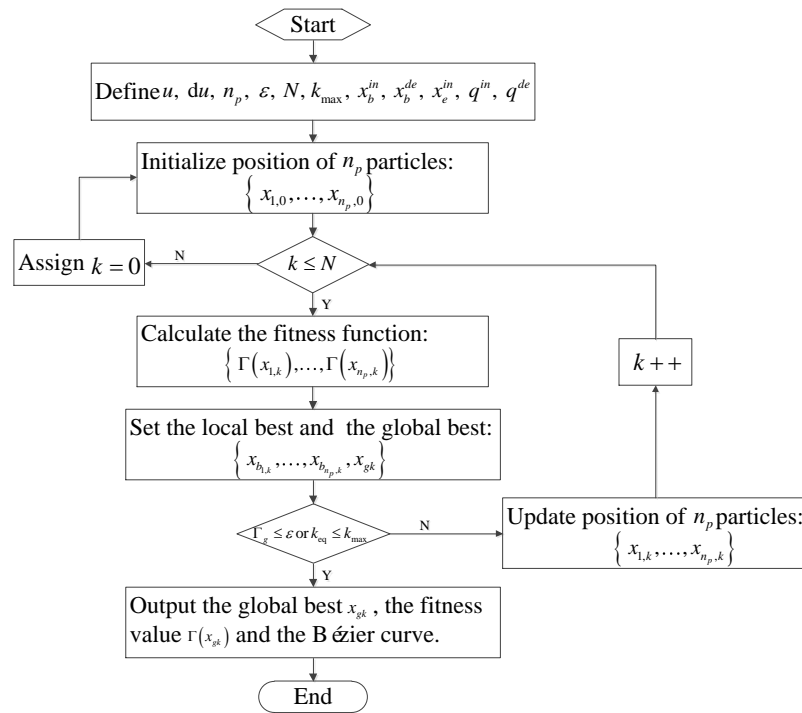


Figure 4. Flow chart of HPSO with self-adaptive ability for AEROM trajectory planning.

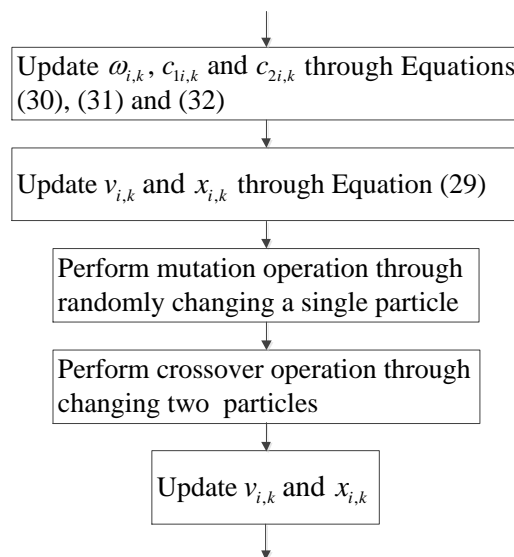


Figure 5. Velocity and position updating mechanisms of particles in HPSO.

### 5. Numerical Simulations

#### 5.1. Parameters of the AEROM System

In this section, three case studies are carried out to validate the proposed HPSO. The AEROM system consists of a 6-DOF UAV and a 6-DOF robotic manipulator with single arm, namely,  $n = 6$  in Figure 2. Table 2 lists mass and rotational inertia parameters of the AEROM system. The length of the links are respectively 0.2 m, 0.38 m, 0.25 m, 0.32 m, 0.25 m and 0.40 m. During the search process shown in Figures 4 and 5, the global best solution is derived for the construction of Bézier curves. Combining the AEROM maneuver time  $T$  through Equation (20), the joint trajectories are then derived. In this paper  $T = 20$  s. The parameters of HPSO are followed:  $\omega_0 = 0.9, \omega_f = 0.1, c_{10} = c_{2f} = 2.0, c_{20} = c_{1f} = 0.1,$

the particle number  $m = 30$ , the iteration number  $N = 300$ , the crossover probability  $p_c = 0.75$ , the mutation probability  $p_m = 0.15$ , the threshold value  $k_{max} = 100$ .

**Table 2.** Physical parameters of the AEROM system.

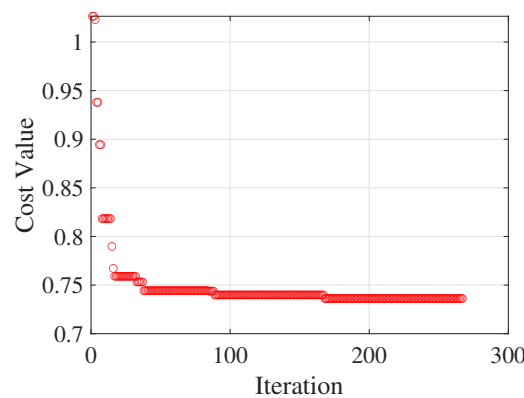
	Mass (kg)	$I_{xx}$ (kg·m <sup>2</sup> )	$I_{xy}$ (kg·m <sup>2</sup> )	$I_{xz}$ (kg·m <sup>2</sup> )	$I_{yy}$ (kg·m <sup>2</sup> )	$I_{yz}$ (kg·m <sup>2</sup> )	$I_{zz}$ (kg·m <sup>2</sup> )
UAV base	80	44	0.3	0.4	43	0.3	26
Link 1	1.5	$5.2 \times 10^{-3}$	$1.9 \times 10^{-5}$	$2.3 \times 10^{-4}$	$7.6 \times 10^{-3}$	$2.8 \times 10^{-5}$	$3.5 \times 10^{-3}$
Link 2	1.1	$3.6 \times 10^{-3}$	$1.1 \times 10^{-3}$	$1.2 \times 10^{-2}$	$1.2 \times 10^{-1}$	$3.0 \times 10^{-4}$	$1.1 \times 10^{-1}$
Link 3	1.5	$9.1 \times 10^{-3}$	$3.1 \times 10^{-5}$	$1.2 \times 10^{-3}$	$1.1 \times 10^{-2}$	$8.1 \times 10^{-5}$	$2.5 \times 10^{-3}$
Link 4	1.0	$5.4 \times 10^{-2}$	$5.6 \times 10^{-4}$	$7.4 \times 10^{-4}$	$4.5 \times 10^{-2}$	$0.1 \times 10^{-2}$	$0.1 \times 10^{-2}$
Link 5	1.5	$0.1 \times 10^{-2}$	$3.1 \times 10^{-5}$	$8.1 \times 10^{-5}$	$9.1 \times 10^{-3}$	$1.2 \times 10^{-4}$	$2.5 \times 10^{-3}$
Link 6	4.0	$2.0 \times 10^{-2}$	$2.0 \times 10^{-3}$	$1.4 \times 10^{-3}$	$2.3 \times 10^{-2}$	$3.2 \times 10^{-3}$	$1.0 \times 10^{-2}$

5.2. Simulation Results and Analysis

Numerical simulations are carried out on a stand PC under the Matlab/Simulink environment. Referring the work [20], the initial state and the desired state of AEROM are detailed joint configurations. The simulations ignoring EE trajectories refers to the following work. Actually, the end-effector trajectory can be directly derived according to the integration of kinetic equation of AEROM, or according to the D-H method.

5.2.1. Case 1

In the first case, the position and the attitude of UAV base are controlled. Therefore, the initial and the final poses of UAV base are not considered. The kinematic equation of the AEROM system is shown in Equation (8). The cost function in (24) considers angular variations of joints (22), i.e.  $\Gamma = \Gamma_1$ . The weight matrix  $M_2$  is assigned as  $M = \frac{1}{\sin(\frac{\pi}{360})} E_6$ , where  $E_6$  is a sixth-order unit matrix. The AEROM system is commanded to move from initial state  $q^0 = [0, 30, 30, 20, 50, 30]$ (deg) to desired state  $q^f = [10, 0, 60, 10, 30, 20]$ (deg). Figure 6 shows the variations of cost function. The proposed HPSO algorithm stops when  $k_{eq} = 101 > 100 = k_{max}$ , where  $N = 267$  and the final cost value is 0.7362. Figure 7 shows the time histories of joint angles, meeting aforementioned conditions. Figure 8 shows variations of joint velocities, where the components are all 0 at initial time and final time. According to the simulation results, the joints move smoothly and the robotic manipulator successfully arrives at the desired state. The error vector  $\delta q$  in Equation (18) is  $\delta q = [-6.07, 1.01, 1.07, 0.311, 2.49, 0.1] \times 10^{-8}$ (deg). Actually, multiple solutions exist due to the geometric construction of robotic manipulator. The selected optimal solution is just one of the multiple solutions. In addition, the represented solution can be employed for further purpose of subsequent optimization.



**Figure 6.** The best values of cost function in case 1.

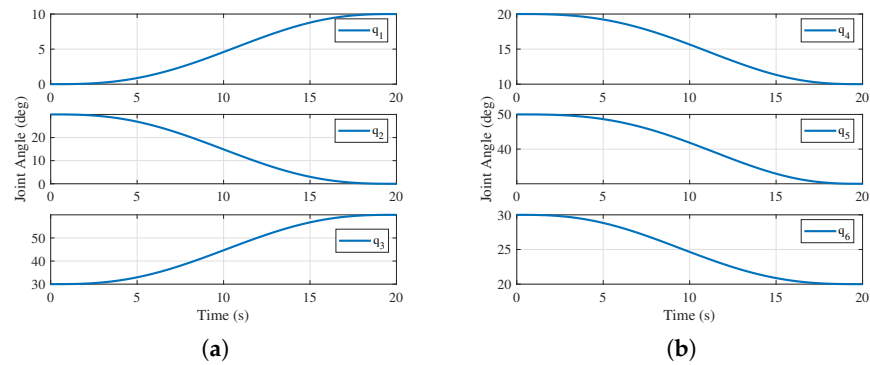


Figure 7. Angular variation curves of *i*th joint, where (a) *i* = 1, 2, 3, and (b) *i* = 4, 5, 6.

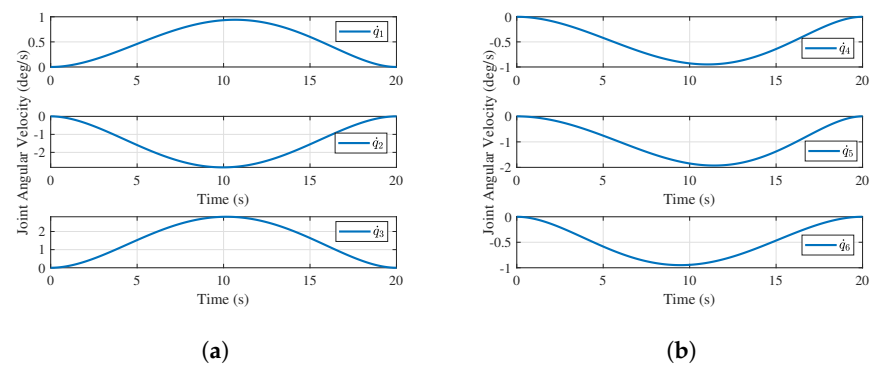


Figure 8. Velocity variation curves of *i*th joint, where (a) *i* = 1, 2, 3, and (b) *i* = 4, 5, 6.

5.2.2. Case 2

In the second case, the position and the attitude of UAV base are not controlled, similar to the free-floating space robot [13] in outer space environment. The kinematic equation of the AEROM system is shown in Equation (10). The pose disturbance acting on UAV base is considered. The cost function in (24) includes the UAV disturbance (21) and the angular variations of joints (22), shown as  $\Gamma = \Gamma_1 + \Gamma_2$ . The weight matrix  $M_1$  is assigned as  $M_1 = \begin{bmatrix} E_3 & O_3 & O_3 & \frac{1}{\sin(\frac{\pi}{360})} E_3 \end{bmatrix}$ . The weight matrix  $M_2$  is assigned as  $M_2 = \frac{1}{\sin(\frac{\pi}{360})} E_6$ .  $E_3$  and  $E_6$  are respectively 3-order and 6-order unit matrices. The AEROM system is commanded to move from initial state  $q^0 = [5, -10, 10, 20, -10, -20]$  (deg) to desired state  $q^f = [-10, -20, 0, 0, 0, -40]$  (deg). Figure 9 shows the variations of the cost function. The proposed HPSO algorithm stops when  $k_{eq} = 101 > 100 = k_{max}$ , where  $N = 278$  and the final value is 1.0789. Figure 10 shows the time histories of joint angles, meeting aforementioned conditions. Figure 11 shows the variations of joint velocities with the final value of  $(5.0 \times 10^{-15}, 5.0 \times 10^{-15}, -1.0 \times 10^{-17}, 1.0 \times 10^{-19}, -7.5 \times 10^{-15}, -1.0 \times 10^{-19})$  (deg). Figure 12 shows the attitude and the position variations of UAV base. The actually planned UAV attitude and UAV position are respectively  $(-5.3 \times 10^{-3}, 5.5 \times 10^{-3}, -6.4 \times 10^{-3})$  (deg) and  $(4.1 \times 10^{-4}, 3.3 \times 10^{-3}, 2.7 \times 10^{-4})$  (m). It means the disturbance order magnitude acting on UAV base is  $-3$ . Therefore, the planned error is within the predefined threshold range, where  $\|[-5.3 \times 10^{-3}, 5.5 \times 10^{-3}, -6.4 \times 10^{-3}]\| = 9.9649 \times 10^{-3} < \varepsilon(1) = 0.1$  with the unit of degree and  $\|[4.1 \times 10^{-4}, 3.3 \times 10^{-3}, 2.7 \times 10^{-4}]\| = 3.3363 \times 10^{-3} < \varepsilon(2) = 0.1$  with the unit of meter. The error vector  $\delta q$  in Equation (18) is  $\delta q = [0.01, 0.11, 0.51, 2.09, -3.73, 4.1] \times 10^{-7}$  (deg). According to the simulation results, the joints move smoothly and the robotic manipulator successfully arrives at the desired state. Furthermore, the objective of minimizing UAV pose disturbance is realized.

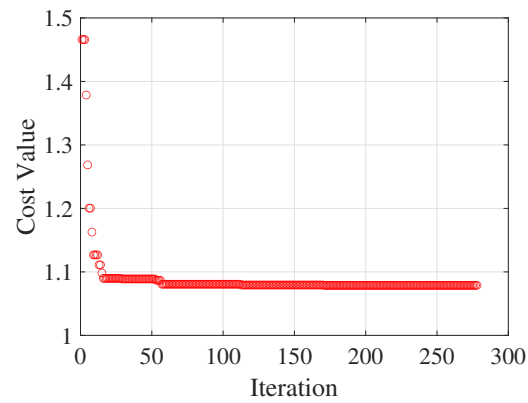


Figure 9. The best values of cost function in case 2.

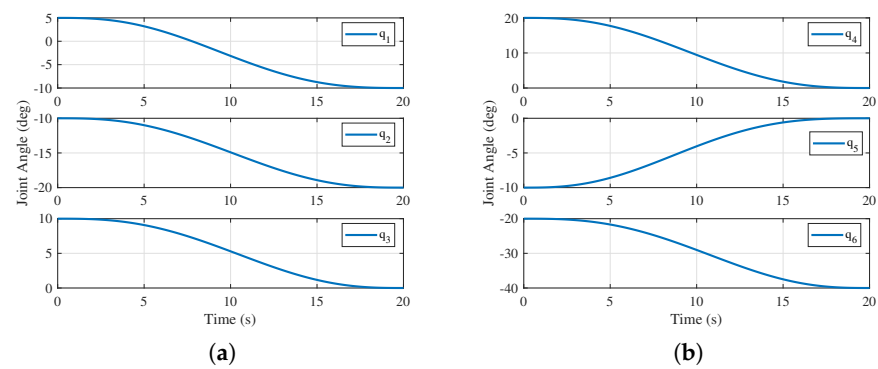


Figure 10. Angular variation curves of  $i$ th joint, where (a)  $i = 1, 2, 3$ , and (b)  $i = 4, 5, 6$ .

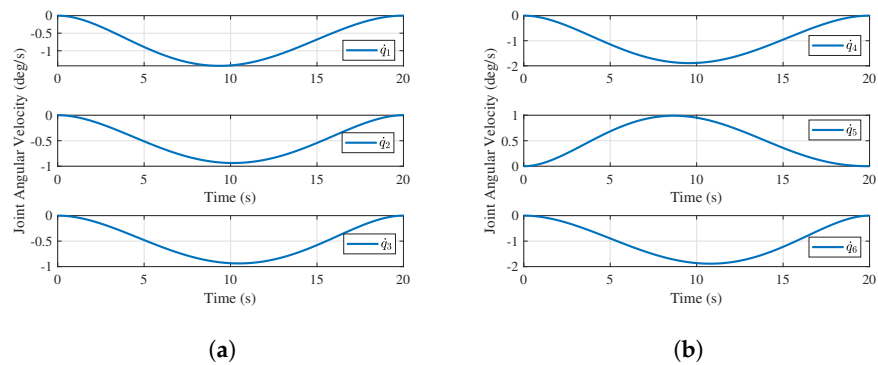


Figure 11. Velocity variation curves of  $i$ th joint, where (a)  $i = 1, 2, 3$ , and (b)  $i = 4, 5, 6$ .

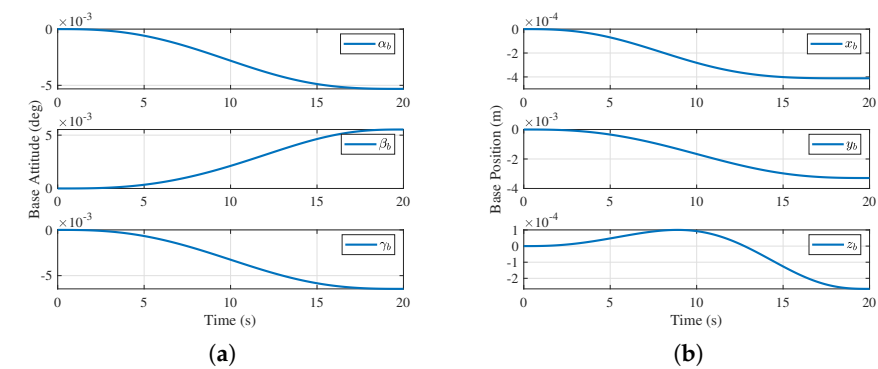


Figure 12. Pose variation of UAV base, where (a) UAV attitude, and (b) UAV position.

5.2.3. Case 3

In the third case, the position and the attitude of UAV base are not controlled. Similar to the second case, the pose disturbance acting on UAV base is considered. However, the robotic manipulator is constrained to move from initial state  $q^0 = [-10, 30, 40, 60, 30, 75]$  to desired state  $q^f = [10, 20, 50, 75, 45, 90]$  with unit of degree. Furthermore, the UAV base is required to move from initial state  $x_b^0 = [-5(\text{deg}), 5(\text{deg}), -5(\text{deg}), 0(\text{m}), 0(\text{m}), 0(\text{m})]$  to desired state  $x_b^{de} = [0(\text{deg}), 0(\text{deg}), 0(\text{deg}), 0(\text{m}), 0(\text{m}), 0(\text{m})]$ . The cost function in (24) includes the UAV disturbance (21) and the angular variations of joints (22), shown as  $\Gamma = \Gamma_1 + \Gamma_1$ . The weight matrix  $M_1$  is assigned as  $M_1 = \begin{bmatrix} E_3 & O_3; O_3 & \frac{1}{\sin(\frac{\pi}{360})} E_3 \end{bmatrix}$ . The weight matrix  $M_2$  is assigned as  $M_2 = \frac{1}{\sin(\frac{\pi}{360})} E_6$ .  $E_3$  and  $E_6$  are respectively 3-order and 6-order unit matrices. Figure 13 shows the variations of cost function. The proposed HPSO algorithm stops when  $k_{eq} = 101 > 100 = k_{max}$ , where  $N = 285$  and the final cost value is 1.1895. Figure 14 shows the time histories of joint angles, meeting aforementioned conditions. Figure 15 shows variations of joint velocities with the value of  $(3.97 \times 10^{-13}, 7.79 \times 10^{-14}, 1.54 \times 10^{-13}, 1.01 \times 10^{-13}, -4.73 \times 10^{-13}, -2.26 \times 10^{-13})$  (deg/s) at final time. Figure 16 shows the attitude and the position variations of UAV base. The actually planned UAV attitude and position are  $(-5.77 \times 10^{-3}, -6.21 \times 10^{-3}, -3.56 \times 10^{-6})$  (deg) and  $(-1.34 \times 10^{-5}, 4.49 \times 10^{-6}, -8.92 \times 10^{-6})$  (m), respectively. It means the disturbance order magnitude acting on UAV base is  $-3$ . Therefore, the planned error is within the predefined threshold range, where  $\|[-5.77 \times 10^{-3}, -6.21 \times 10^{-3}, -3.56 \times 10^{-6}]\| = 8.4679 \times 10^{-3} < \varepsilon(1) = 0.1$  with the unit of degree and  $\|[-1.34 \times 10^{-5}, 4.49 \times 10^{-6}, -8.92 \times 10^{-6}]\| = 1.6712 \times 10^{-5} < \varepsilon(2) = 0.1$  with the unit of meter. The error vector  $\delta q$  in Equation (18) is  $\delta q = [1.27, 1.07, -0.11, -0.47, -1.51, -2.21] \times 10^{-10}$  (deg). According to the simulation results, the joints move smoothly and the AEROM system successfully arrives at the desired state. Conclusively, the joint trajectories in aforementioned three cases are all smooth. Furthermore, the generated results are applicable to controlling the AEROM system and meeting the requirements in Section 3.

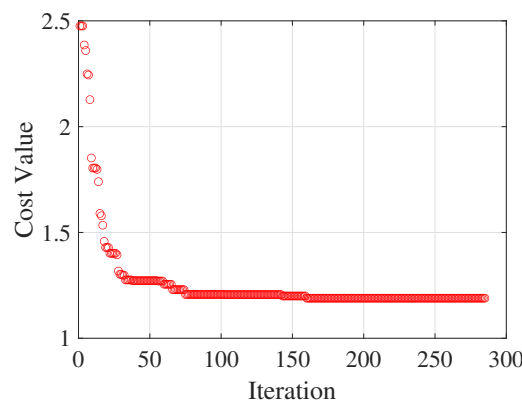


Figure 13. The best values of cost function in case 3.

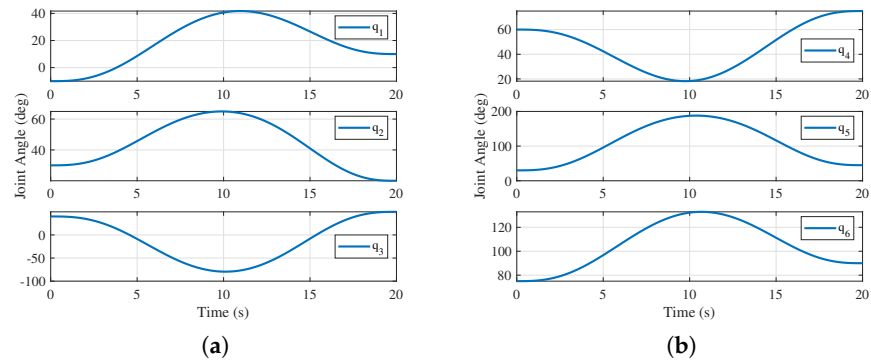


Figure 14. Angular variation curves of  $i$ th joint, where (a)  $i = 1, 2, 3$ , and (b)  $i = 4, 5, 6$ .

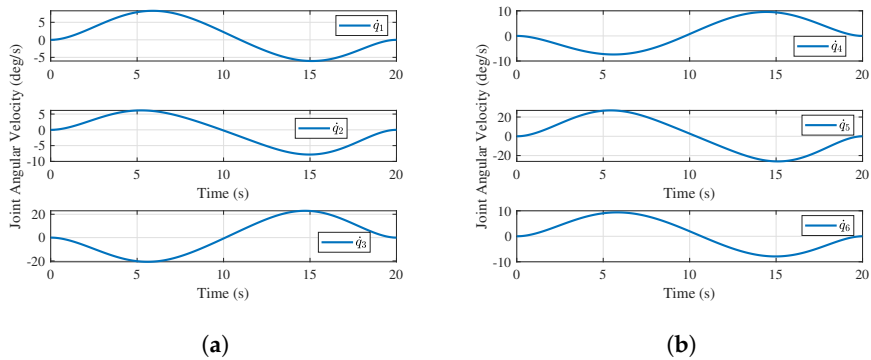


Figure 15. Velocity variation curves of  $i$ th joint, where (a)  $i = 1, 2, 3$ , and (b)  $i = 4, 5, 6$ .

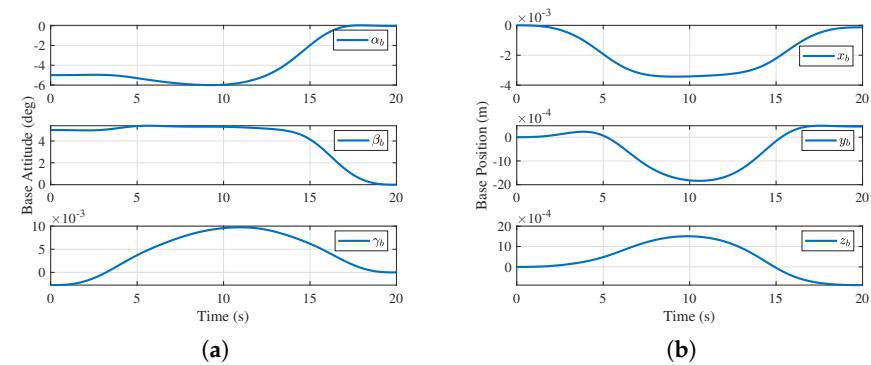


Figure 16. Pose variation of UAV base, where (a) UAV attitude, and (b) UAV position.

### 6. Conclusions

Due to the dynamically singular factor of robotic manipulator and the strong coupling between the UAV base and the robotic manipulator, the inverse kinematics of AEROM system cannot be directly employed for trajectory planning. A novel trajectory planning method for an AEROM system is proposed in this paper, where the AEROM system consists of a UAV base and a robotic manipulator with single arm. The following factors are considered: (i) a specific case, where the AEROM is non-holonomic, (ii) each joint trajectory is depicted with a Bézier curve due to its characteristics of simplicity, smoothness and normalization, (iii) the AEROM maneuver time is derived through boundaries of joint velocities and accelerations, (iv) HPSO with self-adaptive ability is proposed to search through the global space for the construction of each Bézier curve. (v) multiple objectives can be simultaneously optimized utilizing the proposed method. Numerical simulations show that the minimum disturbance acting on the UAV base is realized. The optimization of multiple objectives, i.e., various cost functions, is also well performed.

Variations of physical parameters such as rotational inertia are out of consideration, which will be studied in future work. In addition, the collision avoidance issue is also one of the subjects of future work. Since the trajectory planning is studied through AEROM kinematics, AEROM dynamics are not further considered and analyzed. The AEROM control, based on both kinematics and dynamics, will be studied in a follow-up study.

**Author Contributions:** Conceptualization, S.Z. and C.C.; methodology, S.Z. and J.L.; writing—original-draft preparation, S.Z. and J.L.; writing—review and editing, C.C., S.G. and X.G.; supervision, C.C. and S.G.; funding acquisition, S.Z., S.G. and X.G. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by the technological innovation Guidance Fund of Shaanxi Province (Grant No. 2022QFY01-16), by the Science and Technology foundation of Shaanxi Province (Grant No. 2022GY-236), by National Natural Science Foundation of China (Grant No. 62103182) and by the China Postdoctoral Science Foundation (Grant No. 2021M701548).

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

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors sincerely acknowledge the technological innovation Guidance Fund of Shaanxi Province (Grant No. 2022QFY01-16), the Science and Technology foundation of Shaanxi Province (Grant No. 2022GY-236), the National Natural Science Foundation of China (Grant No. 62103182) and the China Postdoctoral Science Foundation (Grant No. 2021M701548). The authors would also thank the members of the Autonomous Unmanned Systems Research Laboratory for their useful and valuable suggestions.

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

## Abbreviations

The following abbreviations are used in this manuscript:

2-D	Two-Dimensional
AEROM	AERial RObotic Manipulator
D-H	Denavit-Hartenberg
DOF	Degree Of Freedom
EE	End Effector
GA	Genetic Algorithm
HPSO	Hybrid Particle Swarm Optimization
IOA	Intelligent Optimization Algorithm
NP	Nonlinear Programming
PSO	Particle Swarm Optimization
UAV	Unmanned Aerial Vehicle

## References

- Ollero, A.; Tognon, M.; Suarez, A.; Lee, D.; Franchi, A. Past, present, and future of aerial robotic manipulators. *IEEE T. Robot.* **2022**, *38*, 626–645. [CrossRef]
- Welde, J.; Paulos, J.; Kumar, V. Dynamically feasible task space planning for underactuated aerial manipulators. *IEEE Robot. Autom. Let.* **2021**, *6*, 3232–3239. [CrossRef]
- AERIAL-CORE. Available online: <https://aerial-core.eu/objectives/> (accessed on 22 April 2022).
- AERO-TRAIN. Available online: <https://www.aerotrains-etn.eu/> (accessed on 19 June 2022).
- HYFLIERS PROJECT. Available online: <https://www oulu.fi/hyfliders/> (accessed on 28 January 2022).
- AEROARMS. Available online: <https://aeroarms-project.eu/> (accessed on 22 June 2022).
- Xu, W.; Meng, D.; Liu, H.; Wang, X.; Liang, B. Singularity-free trajectory planning of free-floating multiarm space robots for keeping the base inertially stabilized. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2464–2477. [CrossRef]
- Wang, S.; Cao, Y.; Zheng, X.; Zhang, T. Collision-free trajectory planning for a 6-DoF free-floating space robot via hierarchical decoupling optimization. *IEEE Robot. Autom. Let.* **2022**, *7*, 4953–4960. [CrossRef]
- Umetani, Y.; Yoshida, K. Resolved motion rate control of space manipulators with generalized Jacobian matrix. *IEEE Trans. Robot. Autom.* **1989**, *5*, 303–314. [CrossRef]



10. Dubowsky, S.; Torres, M. Path planning for space manipulators to minimize spacecraft attitude disturbances. In Proceedings of the IEEE International Conference on Robotics and Automation, Sacramento, CA, USA, 9–11 April 1991; pp. 2522–2528.
11. Papadopoulos, E.; Dubowsky, S. Dynamic singularities in free-floating space manipulators. In *Space Robotics: Dynamics and Control*; Xu, Y., Kanade, T., Eds.; Springer: Boston, MA, USA, 1993; pp. 77–100.
12. Yoshida, K.; Hashizume, K.; Abiko, S. Zero reaction maneuver: Flight validation with ETS-VII space robot and extension to kinematically redundant arm. In Proceedings of the IEEE International Conference on Robotics and Automation, Seoul, Korea, 21–26 May 2001; pp. 441–446.
13. Zhao S.; Zhu Z.; Luo J. Multitask-based trajectory planning for redundant space robotics using improved Genetic Algorithm. *Appl. Sci.* **2019**, *9*, 2226. [[CrossRef](#)]
14. Zhao, S.; Siciliano, B.; Zhu, Z.; Gutiérrez-Giles, A.; Luo, J. Multi-waypoint-based path planning for free-floating space robots. *Int. J. Robot. Autom.* **2019**, *34*, 461–467. [[CrossRef](#)]
15. Xu, W.; Li, C.; Liang, B.; Liu, Y.; Xu, Y. The cartesian path planning of free-floating space robot using particle swarm optimization. *Int. J. Adv. Robot. Systm.* **2008**, *5*, 301–310. [[CrossRef](#)]
16. Xu, W.; Li, C.; Wang, X.; Liu, Y.; Liang, B.; Xu, Y. Study on non-holonomic cartesian path planning of a free-floating space robotic system. *Adv. Robot.* **2009**, *23*, 113–143. [[CrossRef](#)]
17. Huang, P.; Xu, Y.; Liang, B. Minimum-torque path planning of space robots using genetic algorithms. *Int. J. Robot. Autom.* **2006**, *21*, 229–236. [[CrossRef](#)]
18. Lampariello, R.; Tuong, D.; Castellini, C.; Hirzinger, G.; Peters, J. Trajectory planning for optimal robot catching in real-time. In Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 3719–3726.
19. Seddaoui A.; Saaj C. Collision-free optimal trajectory generation for a space robot using genetic algorithm. *Acta Astronaut.* **2021**, *179*, 311–321. [[CrossRef](#)]
20. Xu, W.; Liu, Y.; Liang, B.; Xu Y.; Li, C.; Qiang, W. Non-holonomic path planning of a free-floating space robotic system using genetic algorithms. *Adv. Robot.* **2008**, *22*, 451–476. [[CrossRef](#)]
21. Wang, M.; Luo, J.; Fang, J.; Yuan, J. Optimal trajectory planning of free-floating space manipulator using differential evolution algorithm. *Adv. Space Res.* **2018**, *61*, 1525–1536. [[CrossRef](#)]
22. Luo, G.; Zou, L.; Wang, Z.; Lv, C.; Ou, J.; Huang, Y. A novel kinematic parameters calibration method for industrial robot based on Levenberg-Marquardt and Differential Evolution hybrid algorithm. *Robot. Comput.-Integr. Manuf.* **2021**, *71*, 102165. [[CrossRef](#)]
23. Wang, M.; Luo, J.; Walter, U. Trajectory planning of free-floating space robot using Particle Swarm Optimization (PSO). *Acta Astronaut.* **2015**, *112*, 77–88. [[CrossRef](#)]
24. Wang, M.; Luo, J.; Yuan, J.; Walter, U. Coordinated trajectory planning of dual-arm space robot using constrained particle swarm optimization. *Acta Astronaut.* **2018**, *146*, 259–272. [[CrossRef](#)]
25. Miao, C.; Chen, G.; Yan, C.; Wu Y. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Comput. Ind. Eng.* **2021**, *156*, 107230. [[CrossRef](#)]
26. Cna, B.; Dvl, A. A comparative study on ant colony optimization algorithm approaches for solving multi-objective path planning problems in case of unmanned surface vehicles. *Ocean Eng.* **2022**, *255*, 111418.
27. Mirza, N. Application of fuzzy neural networks in robotic path planning. In Proceedings of the International Arab Conference on Information Technology (ACIT), Al Ain, United Arab Emirates, 3–5 December 2019; pp. 58–62.
28. Nair, R.; Supriya, P. Robotic path planning using recurrent neural networks. In Proceedings of the International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 1–3 July 2020; pp. 1–5.
29. Heredia, G.; Cano, G.; Jimenez-Cano, A.; Ollero, A. Modeling and design of multirotors with multi-joint arms. In *Aerial Robotic Manipulation: Research, Development and Applications*; Ollero, A., Siciliano, B., Eds.; Springer: Cham, Switzerland, 2019; pp. 15–34.
30. Orsag, M.; Korpela, C.; Oh, P.; Bogdan, S. *Aerial Manipulation*, 1st ed.; Springer: Cham, Switzerland, 2018; pp. 87–122.
31. Wu, Y.; Yu, Z.; Li, C.; He, M.; Chen, Z. Reinforcement learning in dual-arm trajectory planning for a free-floating space robot. *Aerosp. Sci. Technol.* **2020**, *98*, 105657. [[CrossRef](#)]
32. Yang, B.; He, Y.; Han, J.; Liu, G. Modeling and control of rotor-flying multi-joint manipulator. In Proceedings of the IFAC World Congress, Cape Town, South Africa, 24–29 August 2014; pp. 11024–11029.
33. Meng, X.; He, Y.; Han, J. Survey on aerial manipulator: System, modeling, and control-corrigendum. *Robotica* **2020**, *38*, 1288–1317. [[CrossRef](#)]
34. Farin, G. *Curves and Surfaces for Computer-Aided Geometric Design: A Practical Guide*, 4th ed.; Academic Press: San Diego, CA, USA, 1996; pp. 41–63.
35. Faraway, J.; Reed, M.; Wang, J. Modelling three-dimensional trajectories by using bezier curves with application to hand motion. *Appl. Stat.* **2007**, *56*, 571–585. [[CrossRef](#)]
36. Kennedy, J.; Eberhart, R. Particle swarm optimization (PSO), In Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia, 27 November–1 December 1995; pp. 1942–1948.
37. Liu, H.; Cai Z.; Wang, Y. Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. *Appl. Soft Comput.* **2010**, *10*, 629–640. [[CrossRef](#)]