

Article

A PI Control Method with HGSO Parameter Regulator for Trajectory Planning of 9-DOF Redundant Manipulator

Meijiao Liu ^{1,2,*}, Tianyu Liu ^{1,2}, Mingchao Zhu ¹ and Liheng Chen ¹

¹ CAS Key Laboratory of On-Orbit Manufacturing and Integration for Space Optics System, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China

² School of Optoelectronics, University of Chinese Academy of Sciences, Beijing 100049, China

* Correspondence: liumeijiao17@mails.ucas.ac.cn

Abstract: In order to solve the tracking accuracy problem of the redundant manipulator, a PI control method with Henry gas solubility optimization parameter regulator (PI-HGSO) is proposed in this paper. This method consists of the controller and the parameter regulator. The characteristic is that the position deviation of a manipulator is equivalent to a specific function; namely, the proportional-integral (PI) controller is used to adjust the deviation input. The error can be better corrected by the processing of the PI controller so that the inverse kinematics solution of the minimum error can be realized. At the same time, the parameter selection of PI controllers has always been a difficulty in controller design. To address the problem, Henry gas solubility optimization (HGSO) is selected as a parameter regulator to optimize the parameters and obtain the optimal controller, thereby achieving high-precision trajectory tracking. Experiments on 9-DOF redundant manipulator show that our method achieves competitive tracking accuracy in contrast with others. Meanwhile, the efficiency and accuracy of the PI controller are greatly guaranteed by using HGSO to automatically optimize controller parameters instead of making approximate adjustments through infinite manual trial and error. Therefore, the feasibility and competitive superiority of PI-HGSO is fully proved in trajectory planning of redundant manipulators.

Keywords: 9-DOF redundant manipulator; trajectory planning; HGSO; PI control; parameter adjustment



Citation: Liu, M.; Liu, T.; Zhu, M.; Chen, L. A PI Control Method with HGSO Parameter Regulator for Trajectory Planning of 9-DOF Redundant Manipulator. *Sensors* **2022**, *22*, 6860. <https://doi.org/10.3390/s22186860>

Academic Editor: Hossam A. Gabbar

Received: 4 August 2022

Accepted: 6 September 2022

Published: 10 September 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

With the development of technology and industry, redundant manipulator that can achieve high flexibility and high precision is gradually developed to meet the needs of modern intelligent industry. Although the redundant manipulator can complete more complex tasks, it is more difficult to solve the problems of related kinematics planning and control. Trajectory planning of redundant manipulators includes planning a set of end trajectories and completing the movement of the end trajectory under the manipulator's limited conditions of the tracking accuracy, joint velocity, and joint angle during the movement.

Trajectory planning methods of the redundant manipulator are mainly divided into the geometric method, iterative method, and numerical method [1–9]. The geometric method is to analyze each joint angle according to the geometric relationship between the end posture and the specific angles in sequence. However, it is only applicable to manipulators with simple or special structures. The iterative method is to solve the inverse kinematics solution iteratively by an intelligent algorithm. Nevertheless, the iterative method is used to solve the joint angles corresponding to each pose, the computation is heavy, and the real-time performance is affected. At the same time, the traditional intelligent optimization algorithms, including a genetic algorithm (GA) [10–12], ant colony optimization (ACO) [13,14], particle swarm optimization (PSO) [15,16], grey wolf optimizer (GWO) [17–20], etc., have certain shortcomings, such as consistent initial value setting, slow initial optimization speed, complex algorithm structure, slow convergence speed, and easy

to fall into the local optimal. The algebraic numerical method is to calculate joint angles by the Jacobian matrix, such as the least squares method, gradient projection method [21–23], etc. The algebraic method is a universal method, but the parameter value has a great impact on the accuracy. Usually, it is not easy to find the appropriate parameter value.

In this paper, a PI control method with an HGSO parameter regulator (PI-HGSO) is proposed to perform trajectory planning of a 9-DOF redundant manipulator. Proportional-integral (PI) controller is designed to set the optimum joint angles from the current to the desired end pose. Meanwhile, Henry gas solubility optimization (HGSO) is used to solve the appropriate controller parameters to achieve the trajectory planning of the redundant manipulator. PI controller [5,24–26] has strong model identification ability and can more accurately equivalent various functions. In addition, the PI controller is simple in structure and easy to use to give a value close to the deviation that needs to be assigned to the end-effector. Therefore, the PI controller is selected to deal with the deviation of the end to the expected pose to achieve motion control. Traditional parameter tuning relies on empirical trial and error with low efficiency and accuracy. At the same time, the introduction of intelligent algorithms for parameter adjustment can automatically achieve efficient and accurate parameter selection. HGSO is based on the principle of gas solubility function to design the optimization algorithm [27–31]. The algorithm has not only a strong optimization ability but also groups the swarm particles to achieve multi-group parallel optimization. This characteristic improves the speed of optimization and effectively overcomes the problem of local optimum. Hence, HGSO is selected to optimize the PI controller parameters to further improve the accuracy of the motion planning.

The specific contributions of the PI control method with HGSO parameter regulator (PI-HGSO) for trajectory planning of 9-DOF redundant manipulator can be summarized as follows:

- The proposed PI-HGSO method enables the designed PI controller to obtain accurate gains during the trajectory planning process for redundant manipulators, achieving a highly precise position tracking of the end-effector.
- The design can obtain PI controller parameters more quickly and efficiently by using the HGSO parameter regulator instead of the traditional empirical trial and error, improving the controller parameter tuning cycle.
- The PI controller is used to replace the traditional gain compensation system to achieve the end-effector position compensation of the deviation part efficiently. The design structure is simple, and the error is effectively suppressed. Besides, the feasibility of the design has been guaranteed since this paper has proved that PI-HGSO is asymptotically stable in the Cartesian deviation trajectory of the endpoint.

The structure of the rest is organized as follows: the second part describes the PI controller with joint limits for trajectory planning; the third part describes the design of parameter regulator by HGSO; the fourth part completes the simulation and results of analysis under the 9-DOF redundant manipulator; the fifth part is conclusion and summary.

2. PI Controller with Joint Limits for Trajectory Planning of Redundant Manipulator

In order to solve the inverse kinematics of the redundant manipulator, the algebraic method needs to calculate the joint angles through the pose deviation. The deviation calculation of the end-effector needs to satisfy a certain functional relationship, and this characteristic has a strong fit with the PI characteristic. Therefore, in this paper, the expected path planning of the end-effector is equivalent to a PI controller [5,24,26]. The schematic diagram is shown in Figure 1, and the equivalent formula is described as:

$$\delta = k_p(x_d - x_t) + k_i \int_0^t (x_d - x_t) dt \quad (1)$$

where δ is defined to be the PI controller as the deviation expected to be given at the end-effector; $(x_d - x_t)$ is the difference between the expected position and the current position; k_p and k_i are the PI controller parameters.

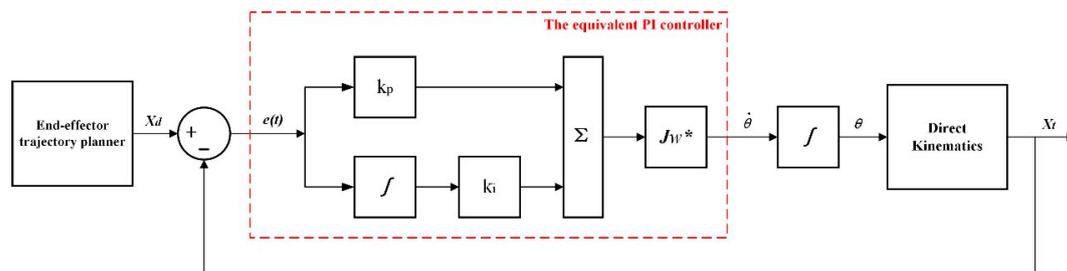


Figure 1. The block diagram of the path planning method that is equivalent to a PI controller.

Theorem 1. *The stability of the equivalent PI controller method can be proved by the Lyapunov stability theorem [5,25,32].*

We denote the end-Cartesian space deviation by $e(t) = x_d - x_t$, where $\dot{e}(t)$ represents its time derivative. The speed of position deviation can be compensated by designing a PI linear system. Thus, it is written as $\dot{e}(t) = -\delta$, we have:

$$\dot{e}(t) = -\delta = -k_p e(t) - k_i \int_0^t e(t) dt \quad (2)$$

We define the Lyapunov function as:

$$v(t) = e^2(t) + k_i \left(\int_0^t e(t) dt \right)^2 \quad (3)$$

Which is obtain $v(t) \geq 0$, and the positive definiteness of $v(t)$ is guaranteed. For the defined Lyapunov function $v(t)$, we can obtain the time derivative $\dot{v}(t)$, which is derived as:

$$\begin{aligned} \dot{v}(t) &= 2e(t)\dot{e}(t) + 2k_i e(t) \int_0^t e(t) dt \\ &= 2e(t) \left(-k_p e(t) - k_i \int_0^t e(t) dt \right) + 2k_i e(t) \int_0^t e(t) dt \\ &= -2k_p e^2(t) \leq 0 \end{aligned} \quad (4)$$

where $\dot{v}(t) \leq 0$, it can be guaranteed the negative definiteness of $\dot{v}(t)$. According to the Lyapunov theory, the trajectory of the end-Cartesian space deviation $e(t)$ is asymptotically stable in the equivalent PI controller method.

In the inverse kinematics solution process of the redundant robot, the value that needs to be acted on the end-effector is mapped to the joint space through the Jacobian matrix. Whereupon, the velocity of joint space is obtained, which is expressed as:

$$\dot{\theta} = J^{-1} \delta \quad (5)$$

Minimum norm method with weighting coefficient [9,33–35] was proposed to solve the redundant system problem. Its basic idea is to use the minimum weighted norm by adding the weighted matrix to realize the task limitation. In order to satisfy the joint angle constraint and reduce the joint angular velocity, the Jacobian matrix optimized by minimum norm with weighting coefficient is usually used. When the joint angular velocity is too large, the weighted coefficient is increased as a penalty factor to reduce the joint velocity and achieve the restriction of joint velocity.

The Jacobian matrix with weighting coefficient is expressed as:

$$J_w = JW \quad (6)$$

$$W = \text{diag}(w_1, w_2, \dots, w_i, \dots, w_9) \quad (7)$$

$$w_i = \begin{cases} 1 + \left| \frac{\partial H(\theta)}{\partial \theta_i} \right| & \text{if } \Delta \left| \frac{\partial H(\theta)}{\partial \theta_i} \right| \geq 0 \\ 1 & \text{if } \Delta \left| \frac{\partial H(\theta)}{\partial \theta_i} \right| < 0 \end{cases} \quad (8)$$

where J is the Jacobian matrix, J_w is the weighted Jacobian, W is the weighted matrix, and $H(\theta)$ is the constructed objective function.

To avoid approaching the joint limit, the objective optimization function $H(\theta)$ is defined as:

$$H(\theta) = \frac{1}{9} \sum_{i=1}^9 \frac{(\theta_{i,\max} - \theta_{i,\min})^2}{(\theta_{i,\max} - \theta_i)(\theta_i - \theta_{i,\min})} \quad (9)$$

$$\frac{\partial H(\theta)}{\partial \theta_i} = \frac{1}{9} \sum_{i=1}^9 \frac{(\theta_{i,\max} - \theta_{i,\min})^2 (2\theta_i - \theta_{i,\max} - \theta_{i,\min})}{(\theta_{i,\max} - \theta_i)^2 (\theta_i - \theta_{i,\min})^2} \quad (10)$$

Figure 2 is the graph of the objective function. It can be seen from Figure 2 that when the joint angle approaches the joint limit, the designed objective optimization function $H(\theta)$ approaches infinity, and the value of $\frac{\partial H(\theta)}{\partial \theta_i}$ also approaches infinity. Moreover, $H(\theta)$ and $\frac{\partial H(\theta)}{\partial \theta_i}$ are almost 0 within the joint limit range. Therefore, in order to avoid the joint angle exceeding the limit, the smaller the value of the optimization function the better. When the joint angle changes to the limit value, the joint velocity will be reduced by increasing the weight coefficient. Therefore, the restriction on the joint angle will be implemented.

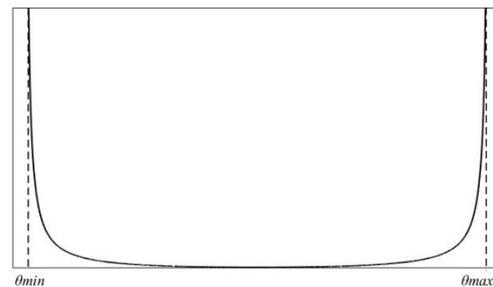


Figure 2. The graph of the objective function.

For the redundant manipulator, its Jacobian matrix is a long square matrix, and the inverse cannot be calculated. So, the pseudo-inverse can be used. However, the pseudo-inverse of the Jacobian matrix [9,36,37] will lead to the problem of matrix singularity. In order to avoid the singularity, the Jacobian inverse with singularity robustness is introduced, and its formula is described as:

$$J^* = J^T (J J^T + \lambda I)^{-1} \quad (11)$$

The Jacobian inverse with singularity robustness and weighting coefficient is expressed as:

$$J_w^* = W^{-1} J^T (J W^{-1} J^T + \lambda I)^{-1} \quad (12)$$

Thus, the inverse kinematics solution of using the weighted minimum norm method is expressed as:

$$\dot{\theta} = J_w^* \delta = W^{-1} J^T (J W^{-1} J^T + \lambda I)^{-1} \delta \quad (13)$$

By combining the weighted Jacobian minimum norm and singularity robustness inverse, the joint velocity norm can be reduced by J_w^* . Therefore, this will play a role in planning the joint velocity to some extent. And I am going to give λ a small enough value to avoid the singularities.

3. A PI Control Method with HGSO Parameter Regulator

For the joint angle constraint problem and joint angle singularity problem, the Jacobian pseudoinverse with weight coefficient is applied to complete the transformation of joint angular velocity and joint angle. At the same time, in order to solve the problem of manual selection of PI controller parameters by experience, HGSO is selected to optimize

PI parameters, ensuring the requirement of a small error value in the process of redundant manipulator trajectory planning. Thereupon, A PI control method with an HGSO parameter regulator is designed to perform trajectory planning of the 9-DOF redundant manipulator, and the specific scheme is presented in Figure 3.

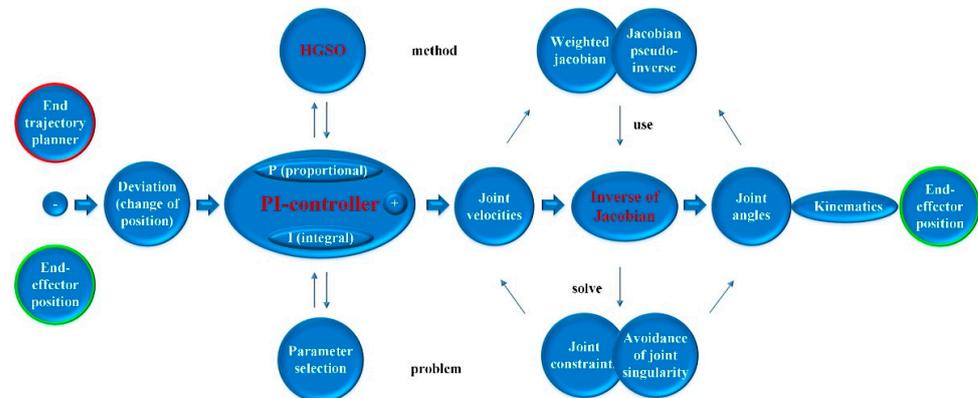


Figure 3. The specific scheme structure diagram of PI-HGSO.

3.1. Introduction and Description of HGSO

A novel physics-based algorithm named Henry gas solubility optimization (HGSO) was proposed by Fatma et al. in 2019 [27]. Furthermore, the optimization performance is better than other algorithms and successfully applied to the related optimization problems. HGSO was inspired by Henry's Gas Law, formulated by William Henry in 1803 [38]. The basic idea of Henry's law of gas solubility is described as [27–29]: at a constant temperature and pressure, the solubility of a gas in a given volume of liquid is proportional to the partial pressure of the gas in equilibrium. So, temperature and pressure are going to affect the gas solubility. The schematic diagram is shown in Figure 4. The gas law is expressed by the following equation:

$$S_g = H \times P_g \quad (14)$$

where S_g is the gas solubility, H is Henry's constant, P_g is the gas partial pressure.

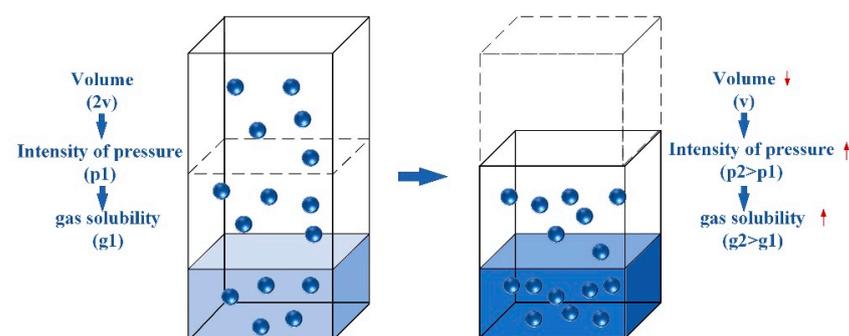


Figure 4. The relationship between the amount of dissolved gas particles and pressure.

The Henry's constant H is expressed as:

$$H(T) = H^\theta \times \exp\left(\frac{-\nabla_{sol}E}{R}\left(\frac{1}{T} - \frac{1}{T^\theta}\right)\right) \quad (15)$$

where, T and T^θ are, respectively, the current temperature and reference temperature, H^θ is the reference Henry constant, R is the gas constant, and the enthalpy of dissolution is denoted by $\nabla_{sol}E$.

$\frac{\nabla_{sol}E}{R}$ is the van't Hoff formula, and is defined as a constant C . Therefore, the Henry's constant H is reformulated as:

$$H(T) = H^\theta \times \exp\left(-C \times \left(1/T - 1/T^\theta\right)\right) \quad (16)$$

The HGSO is established on the basis of Henry's law of gas solubility, and related mathematical modeling formulas and steps are presented as follows:

Step 1: Initialization

At the beginning of the HGSO algorithm, N gas particles are randomly initialized to their gas positions, the initialization is based on the formula:

$$X_i(t+1) = X_{\min} + r \times (X_{\max} - X_{\min}) \quad (17)$$

where X_i is described as the position of the i th gas in population N , X_{\max} and X_{\min} are, respectively, the upper and lower limits of position, r is a random constant that can generate randomly the number between 0 and 1, t is the iteration time.

Furthermore, gas i is segmented into type cluster j and the other initial parameter values related to Henry's law are expressed as:

$$\begin{aligned} H_j(t) &= l_1 \times \text{rand}(0,1) \\ P_{i,j} &= l_2 \times \text{rand}(0,1) \\ C_j &= l_3 \times \text{rand}(0,1) \end{aligned} \quad (18)$$

where $H_j(t)$ is the Henry's constant of cluster j , $P_{i,j}$ is the particle pressure of gas i in cluster j , C_j is the van't Hoff formula value of cluster j . Among them, $l_1 = 5 \times 10^{-2}$, $l_2 = 100$, $l_3 = 1 \times 10^{-2}$.

Step 2: Clustering

There are M type clusters in population N , gas population needs to be divided into different clusters. So, the different Henry's constants $H_1, \dots, H_j, \dots, H_M$ of M clusters are obtained.

Step 3: Fitness evaluation

According to the evaluation index function, the gas particles in each cluster j are evaluated, and the best gas particles are identified. At the same time, the best gas particles in all kinds of clusters are compared to select as the best gas particle in the whole population.

Step 4: Update Henry's coefficient

The updating formula of Henry's constant can be described as:

$$\begin{aligned} H_j(t+1) &= H_j(t) \times \exp(-C_j \times (1/T(t) - 1/T^\theta)) \\ T(t) &= \exp(-t/t_{iter}), T^\theta = 298.15 \end{aligned} \quad (19)$$

where t_{iter} is whole iterations, $T(t)$ is the temperature of iteration t .

Step 5: Update gas solubility.

The gas solubility $S_{i,j}(t)$ needed to be updated and the formula is as:

$$S_{i,j}(t) = K \times H_j(t+1) \times P_{i,j}(t) \quad (20)$$

where $S_{i,j}(t)$ is the current solubility of gas i in cluster j , $P_{i,j}(t)$ is the current pressure of gas i in cluster j , K is the constant and usually defined as 1.

Step 6: Update position

The relevant formula of gas position update strategy is as follows:

$$\begin{aligned} X_{i,j}(t+1) &= X_{i,j}(t) + F \times r \times \gamma \times (X_{j,best}(t) - X_{i,j}(t)) + F \times r \times \alpha \times (S_{i,j}(t) \times X_{best}(t) - X_{i,j}(t)) \\ \gamma &= \beta \times \exp\left(-\frac{F_{best}(t)+\varepsilon}{F_{i,j}(t)+\varepsilon}\right), \varepsilon = 0.05 \end{aligned} \quad (21)$$

where $X_{i,j}$ is the position of gas i in cluster j , $X_{j,best}$ is the best gas in cluster j , X_{best} is the current best gas in the whole population, F is the function that can change the searching

direction to achieve diversity and it is denoted by 1 or -1 , r is a random generator that can obtain a random number between 0 and 1, α represents the influencing factor of the other gases to gas i in cluster j and defined as 1, γ is the interaction ability between gas i and other gases in cluster j , β is a constant, $F_{i,j}$ is the fitness of gas i in cluster j , and $F_{best}(t)$ is the fitness of the best gas in the whole population.

Step 7: Escape from local optimum and reset the position of the worst agents

In order to avoid the algorithm falling into the local optimum, the total particles are sorted according to the fitness value and the number of the worst particles is defined. Then these worst particles are selected, and their positions are reset. The position reset strategy is shown in Formula (21), and the relevant formula of the worst particle number is denoted as:

$$N_w = N \times (\text{rand}(c_2 - c_1) + c_1), \quad c_1 = 0.1 \text{ and } c_2 = 0.2 \quad (22)$$

where c_1 and c_2 are constants about rough worst particle proportions, N_w is the number of the worst particles in the whole population.

3.2. HGSO Parameter Regulator for PI Controller

Traditionally, the parameter selection of the PI controller is mainly based on the experience and the trend of the result to adjust and choose the approximate value [24–26]. The method not only fails to find the appropriate value but also wastes a lot of time in adjusting parameters. Due to the optimization ability of HGSO being better than other intelligent algorithms, HGSO is chosen as a parameter adjuster to optimize the parameters of the PI controller to achieve high-precision trajectory tracking and planning [24,27–31]. The specific method flow is shown in Figure 5.

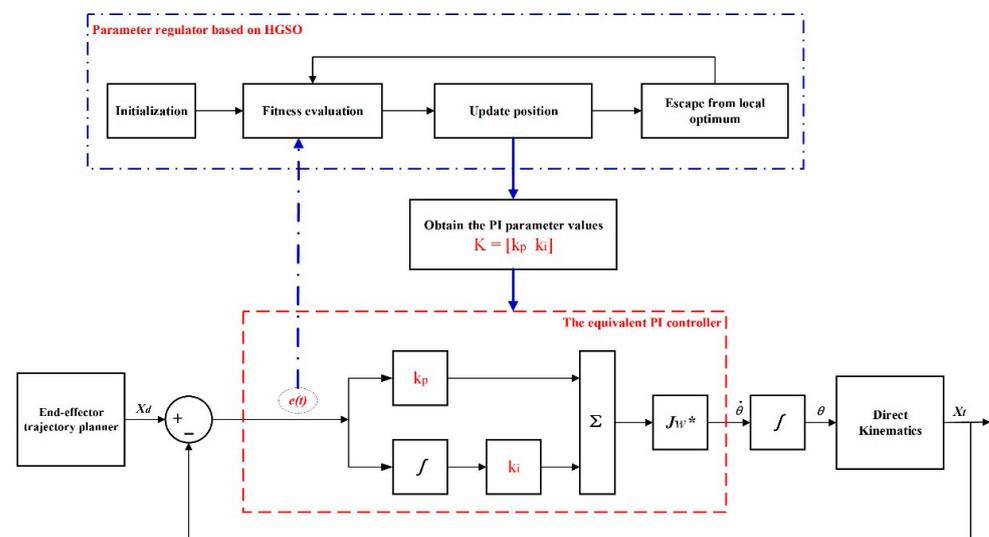


Figure 5. The schematic diagram of PI-HGSO for trajectory planning.

K is defined as the PI control parameter to be optimized, including the proportional parameter k_p and the integral parameter k_i . The parameter K is described as:

$$K = [k_p \ k_i] \quad (23)$$

The sum of the deviations between the actual and the expected trajectory points, is designed as the evaluation index, and the formula of the evaluation function is expressed as:

$$F_{fitness} = \frac{1}{n} \sum_{i=1}^n \left(x_{e(i)}^2 + y_{e(i)}^2 + z_{e(i)}^2 \right)^{1/2} \quad (24)$$

The main principles of PI control method based on HGSO parameter adjuster for trajectory planning are as follows: Firstly, PI parameter K was set and optimized by HGSO. Next, the calculation of the joint angle and position is completed, and the fitness $F_{fitness}$ is obtained by the evaluation function. Therefore, a better K is selected according to the evaluation index. Then, the new K is obtained through the updating mechanism of HGSO, and the parameter regulator will continue to evaluate and select the better K by HGSO. Finally, when the iteration ends, the optimized K is obtained to solve the trajectory tracking. The specific steps are as follows:

Step 1: Initialization

Initialize population particle size N , the number of type clusters M , the particle position information X_i that is the variable K of PI controller to be optimized, and other parameters, $H_j, P_{i,j}, C_j$, etc. ($i = 1, \dots, N; j = 1, \dots, M$).

Step 2: The setting of PI controller parameters:

The position information X_i obtained by HGSO, namely PI parameter K , is assigned to the PI controller for solving joint angles in trajectory planning.

Step 3: Evaluation and update:

Complete the evaluation of the current position X_i of variable K by the fitness function Equation (23). Afterward, according to Equation (20), the position information X_i of the variable K is updated through HGSO's update mechanism.

Step 4: Update the worst particles:

Select the poor particles according to Equation (21) by fitness $F_{fitness}$, meanwhile, reset the position information of the worst particles to avoid local optimality.

Step 5: Repetition and iteration:

Return to Step 2 and continue to complete the next loop until the iteration termination condition is met. The optimized parameters K are obtained, and the trajectory planning of the redundant manipulator is carried out by the PI control method.

The Algorithm 1 implementation pseudocode of the PI control method with HGSO parameter regulator for trajectory planning is as follows:

Algorithm 1 PI control method with HGSO parameter regulator

Import: trajectory points

Output: PI parameters and tracking trajectory

Initialization: Initialize HGSO parameters, N, M, X_i (about PI controller parameters $4K$), $H_j, P_{i,j}, C_j$, etc. using Equation (17) and Equation (18) ($i = 1, \dots, N; j = 1, \dots, M$)

while $t <$ the maximum number of iterations **do**

 Set the X_i obtained by HGSO to PI parameters K for solving joint angles according to the part of PI controller.

 Complete the evaluation of the current X_i by Equation (24).

for each search gas particles **do**

 Update the position information X_i using Equation (21).

for end

 Update Henry's coefficient of each gas type clusters using Equation (19).

 Update solubility of each gas using Equation (20)

 Select the poor particles according to Equation (22) by $F_{fitness}$.

 Reset the position of these worst particles by Equation (19) to Equation (21).

$t = t + 1$

while end

return X_{best} (The final optimized parameters K)

4. Simulation and Results

4.1. The Introduction of 9-DOF Hyper-Redundant Manipulator

In this paper, based on the model of the 9-DOF hyper-redundant manipulator, the feasibility and advantages of the proposed PI-HGSO are verified. The structure of the 9-DOF hyper-redundant manipulator is shown in Figure 6a, and its rotation axis is assembled alternately at 90 degrees. At the same time, the screw method is adopted for kinematic modeling, and the centroid of each joint is established as the origin of the coordinates. The establishment of a coordinate system is shown in Figure 6b, and the related parameters are shown in Table 1.

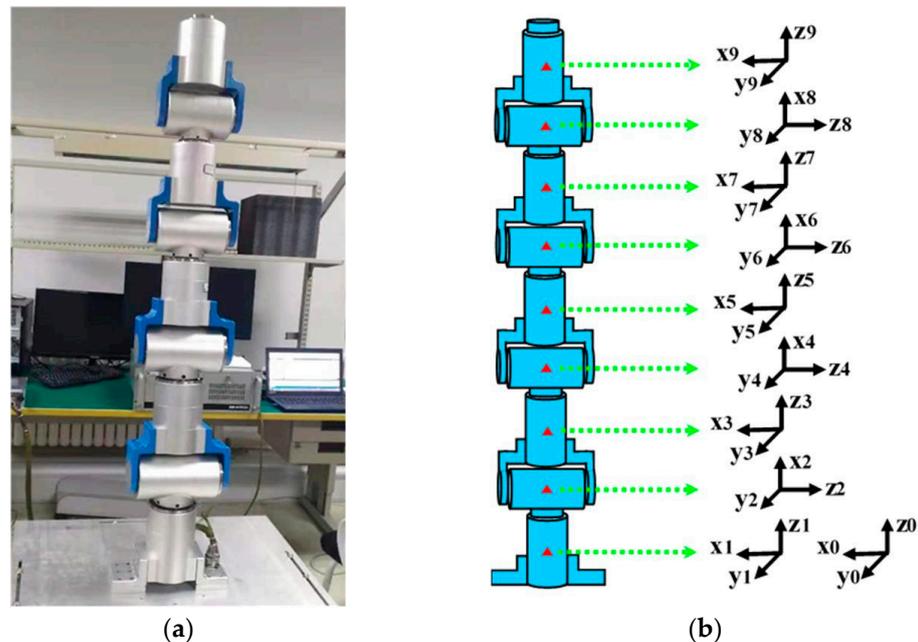


Figure 6. The correlogram of 9-DOF hyper-redundant manipulator. (a) Structure of the manipulator; (b) Establishment of coordinate system.

Table 1. The parameter table of screw modeling.

Joint i	Rotating Vector w	Translation Vector v (Unit: mm)
1	[0 0 1]	[0 0 0]
2	[1 0 0]	[-136.80 -0.10 5.33]
3	[0 0 1]	[13.66 -0.10 -286.11]
4	[1 0 0]	[-483.75 -0.10 4.85]
5	[0 0 1]	[16.26 -0.53 -621.91]
6	[1 0 0]	[-815.07 0.23 7.38]
7	[0 0 1]	[14.17 -0.50 -940.63]
8	[1 0 0]	[-1114.45 -0.31 6.24]
9	[0 0 1]	[10.92 -0.57 -1241.08]

4.2. The Introduction of 9-DOF Hyper-Redundant Manipulator

In the experimental part, the linear trajectory and the curve trajectory are taken as examples to prove the effectiveness of PI-HGSO by simulation. The related parameters of algorithms are listed in Table 2.

Table 2. The Parameter settings of algorithms.

Algorithm	Parameters
GA	Population size $N = 50$ Crossover probability $\alpha = 0.8$ Mutation probability $\beta = 0.5$ Iterations $Nc = 50$
PSO	Population size $N = 50$ Inertia weight decreases linearly from 0.9 to 0.4 Individual-best acceleration factor $C1 = 2$ Global-best acceleration factor $C2 = 2$ Iterations $Nc = 50$
ACO	Population size $N = 50$ Pheromone trail factor $\alpha = 1$ Heuristic information factor $\beta = 5$ Evaporation rate $\rho = 0.5$ Iterations $Nc = 50$
GWO	Population size $N = 50$ Variable a decreases linearly from 2 to 0 Iterations $Nc = 50$
HGSO	Population size $N = 50$ Cluster number $M = 5$ Influencing factor $\alpha = 1$ Component parameter of interaction ability $\beta = 1$ Iterations $Nc = 50$

The gain results of parameter optimization obtained by PI-HGSO are as follows: linear trajectory gain is $k_p = [0.2784 \times 10^{-4} \quad 30.6587]$, $k_i = [0.3399]$; curve trajectory gain is $k_p = [174.7392 \quad 143.3465]$, $k_i = [2.9170 \times 10^3]$. Figure 7 shows the trajectory planning results of the 9-DOF hyper-redundant manipulator through the linear trajectory and curve trajectory planning under PI-HGSO. At the same time, in order to further prove the advantageous performance superiorities of the designed PI-HGSO than others, the PI algorithm without parameter regulator is compared with the PI-HGSO. Furthermore, the PI-PSO, PI-GA, PI-ACO, and PI-GWO use particle swarm optimization algorithm, genetic algorithm, ant colony algorithm, and grey wolf optimizer to adjust the PI controller, are separately compared with the PI-HGSO. The simulation results are shown in Table 3. The effectiveness and superiority of PI-HGSO can be illustrated in Figure 7 and Table 3.

The trajectory planning results of the 9-DOF manipulator through the linear trajectory and curve trajectory planning are presented by PI-HGSO in Figure 7. What is more, Table 3 shows the comparison of PI-HGSO and other algorithms. It can be seen from Figure 7(a1,a2) that joint angles are all within joint limits, regardless of lines or curves. Meanwhile, the joint velocities are less than 2.5×10^{-2} rad/s and 3×10^{-2} rad/s, respectively, in Figure 7(b1,b2), it fully meets the joint velocity constraint of 0.1 rad/s. Figure 7(c1,c2) show the errors of the end effector on the x, y, and z axes, which are within 3×10^{-4} m and 2.5×10^{-4} m, respectively. At the same time, the vector two norm of the whole end effector errors serve as the overall end effector errors. It can be known that the error values are all under 4×10^{-4} m from Figure 7(d1,d2), and the errors are much smaller than the PI method, which adjusts parameters by experience in Table 3. Figure 7(e1,e2) are the motion trajectory simulation diagrams of the 9-DOF manipulator. In summary, it can be concluded that the trajectory planning errors are small, and the joint angles and joint velocities are within the limits by using PI-HGSO.

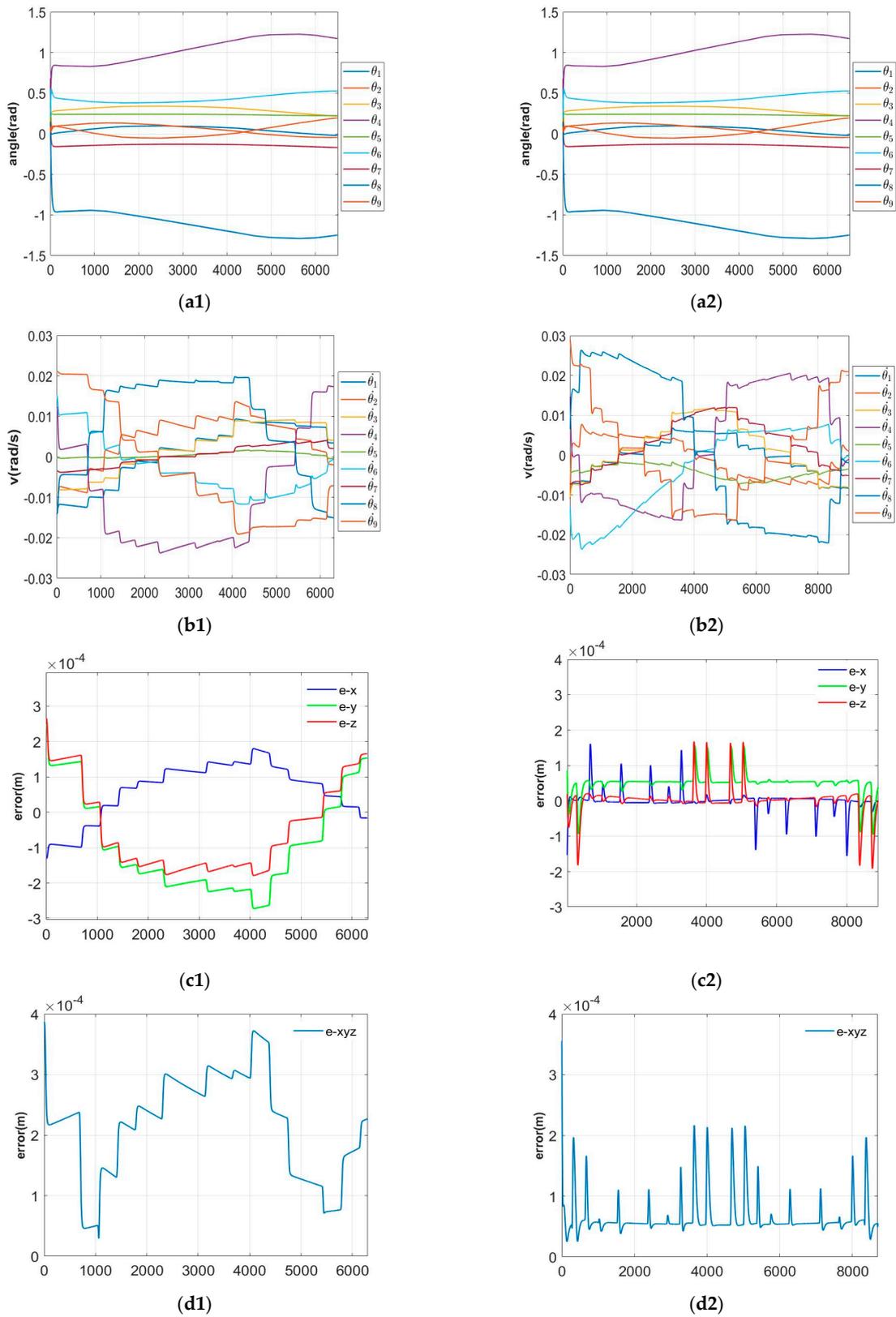


Figure 7. Cont.

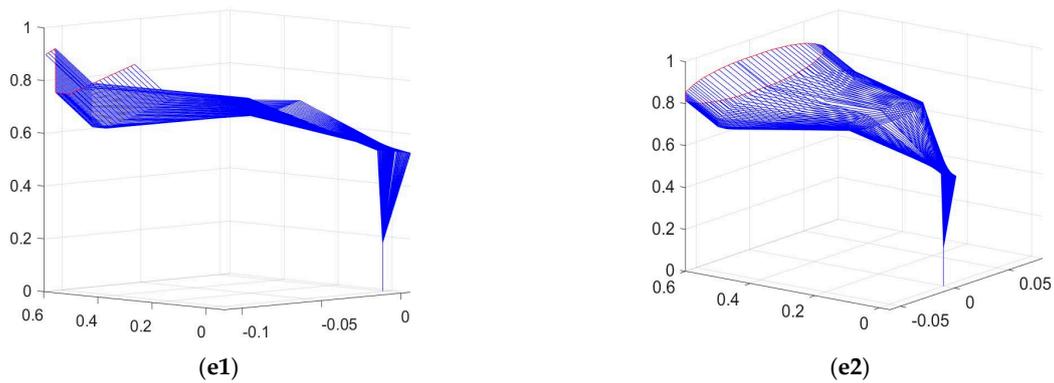


Figure 7. Simulations of the trajectory planning by PI-HGSO under lines and curves. (a1) Joint angles under line; (a2) Joint angles under curve; (b1) Joint velocities under line; (b2) Joint velocities under curve; (c1) Error in x, y, and z under line; (c2) Error in x, y, and z under curve; (d1) Error of end-effector under line; (d2) Error of end-effector under curve; (e1) Motion trajectories under line; (e2) Motion trajectories under curve.

Table 3. The comparative results of PI-HGSO and other algorithms.

Type	Algorithm	The Value and Percent of Deviation					Wilcoxon Rank Sum Test
		Max		Ave		Std	
		Value	Variation	Value	Percent	Value	
Line	No PI	1.42×10^{-3}	/	1.11×10^{-3}	/	1.56×10^{-4}	/
	PI	1.23×10^{-3}	/	1.07×10^{-3}	/	1.80×10^{-4}	/
	PI-PSO	8.42×10^{-3}	-7.19×10^{-3}	2.75×10^{-3}	61.09% ↘	7.32×10^{-4}	+
	PI-GA	1.44×10^{-3}	-0.21×10^{-3}	4.67×10^{-4}	56.35% ↗	5.83×10^{-4}	+
	PI-ACO	5.60×10^{-3}	-4.37×10^{-3}	1.00×10^{-3}	6.54% ↗	3.67×10^{-4}	-
	PI-GWO	4.06×10^{-4}	0.82×10^{-3}	2.44×10^{-4}	77.19% ↗	8.30×10^{-5}	+
	PI-HGSO	3.38×10^{-4}	0.89×10^{-3}	2.15×10^{-4}	79.91% ↗	8.47×10^{-5}	+
Curve	No PI	1.61×10^{-3}	/	1.12×10^{-3}	/	1.87×10^{-4}	/
	PI	1.60×10^{-3}	/	1.16×10^{-3}	/	2.11×10^{-4}	/
	PI-PSO	4.86×10^{-3}	-3.26×10^{-3}	1.27×10^{-4}	89.05% ↗	7.07×10^{-4}	+
	PI-GA	5.53×10^{-3}	-3.93×10^{-3}	4.76×10^{-4}	58.96% ↗	9.32×10^{-4}	+
	PI-ACO	5.33×10^{-3}	-3.73×10^{-3}	1.40×10^{-4}	87.93% ↗	7.72×10^{-4}	+
	PI-GWO	1.55×10^{-3}	0.05×10^{-3}	8.99×10^{-5}	92.22% ↗	1.61×10^{-4}	+
	PI-HGSO	3.55×10^{-4}	1.25×10^{-3}	8.32×10^{-5}	92.83% ↗	3.35×10^{-5}	+

Furthermore, for each algorithm, 30 iterations of experiments were carried out, and corresponding data analysis values were obtained. As shown in Table 3 and Figure 8, the PI controller parameters optimized by the algorithms will be better than the result of manually adjusting the PI controller in average trajectory errors. The results of Wilson’s rank-sum test also show that the designed PI-HGSO has different average error values and is much better than PI. The experimental results are depicted in Table 3, the average errors of PI-HGSO are improved by 79.91% and 92.83%, respectively, under the test of line and curve trajectory than PI. This suggests that the errors are far less than that of manually adjusting PI and the trajectory planning effect is very good.

Meanwhile, it is obtained that the average errors of PI-HGSO are 2.15×10^{-4} m and 8.32×10^{-5} m, which are better than other comparison algorithms in Table 3 and Figure 8. The maximum errors of all trajectory points are 3.38×10^{-4} m and 3.55×10^{-4} m, which are the minimum in the comparison algorithms. In addition, the standard deviations are 8.47×10^{-5} m and 3.35×10^{-5} m, and the error changes of tracking points are small and stable in the whole trajectory tracking process. In the case of Table 3 and Figure 8, to sum up, PI-HGSO has a smaller average trajectory error, maximum error, and standard deviation

of tracking points than PI, PI-PSO, PI-GA, PI-ACO, and PI-GWO. That is, the trajectory planning error and error fluctuation, which are obtained by PI-HGSO, are small, and the best trajectory planning of redundant manipulator can be achieved by PI-HGSO than others.

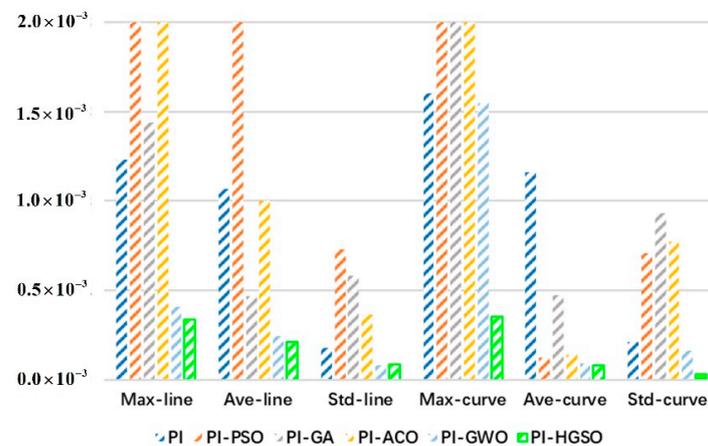


Figure 8. The comparative results of PI-HGSO and other algorithms about errors.

5. Experiment

In order to complete the practical test of the proposed algorithm in this paper, the experimental verification of the PI-HGSO was carried out with the 9-DOF hyper-redundant manipulator entity of our laboratory. The above simulation data of the elliptic curve was used for the experiment, and the experimental trajectory result is shown in Figure 9. According to the experimental process and result, the joint angles are within the constraint range. Simultaneously, the joint velocities and the position tracking deviations are close to the simulation results. The results of each index meet the expected requirements, and the feasibility and superiority of the designed PI-HGSO are fully verified.



Figure 9. The experimental results of the PI-HGSO.

6. Conclusions

In this paper, a PI control method with HGSO parameter regulator for trajectory planning of 9-DOF redundant manipulator is proposed. In this method, the PI controller was selected to optimize the motion deviation of the manipulator end-effector. Then, the Jacobian matrix with weight coefficient is used to limit the joint angles. Meanwhile, HGSO was used to adjust PI controller parameters, and appropriate parameters were obtained to minimize the trajectory planning error. Experimental results demonstrate that the proposed method can not only obtain a small error value but also, the fluctuation of the error of each point is smaller than other algorithms. Besides, PI-HGSO can greatly improve the adjustment time and parameter accuracy compared with the original manual trial with experience.

However, there are some areas where this approach needs to be improved. For example, the iteration time of superior intelligent algorithms is limited, and the real-time performance after different trajectory changes cannot be realized. That could be the future research direction to realize the fast response of parameter adjustment.

Author Contributions: Conceptualization, M.L.; methodology, M.L.; validation, T.L.; formal analysis, M.L. and T.L.; data curation, M.L. and T.L.; writing—original draft preparation, M.L.; writing—review and editing, M.L. and T.L.; supervision, M.Z. and L.C.; project administration, M.Z. and L.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China (no. 11672290).

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

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

References

1. Zaplana, I.; Basanez, L. A novel closed-form solution for the inverse kinematics of redundant manipulators through workspace analysis. *Mech. Mach. Theory* **2018**, *121*, 829–843. [[CrossRef](#)]
2. Faria, C.; Ferreira, F.; Erhagen, W.; Monteiro, S.; Bicho, E. Position-based kinematics for 7-DoF serial manipulators with global configuration control, joint limit and singularity avoidance. *Mech. Mach. Theory* **2018**, *121*, 317–334. [[CrossRef](#)]
3. Zhang, Y.; Wang, J.; Xia, Y. A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits. *IEEE Trans. Neural Netw.* **2003**, *14*, 658–667. [[CrossRef](#)]
4. Parasuraman, S.; Hou, C.M.; Ganapathy, V. Motion Planning for a Redundant Manipulator Using Genetic Algorithm. *Key Eng. Mater.* **2011**, *467–469*, 782–787. [[CrossRef](#)]
5. Guo, D.; Xu, F.; Yan, L. New Pseudoinverse-Based Path-Planning Scheme with PID Characteristic for Redundant Robot Manipulators in the Presence of Noise. *IEEE Trans. Control. Syst. Technol.* **2018**, *26*, 2008–2019. [[CrossRef](#)]
6. Yazdani, M.; Novin, R.S.; Masouleh, M.T.; Menhaj, M.B.; Abdi, H. An experimental study on the failure tolerant control of a redundant planar serial manipulator via pseudo-inverse approach. In Proceedings of the 2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM), Tehran, Iran, 7–9 October 2015; pp. 365–370.
7. Wan, J.; Wu, H.T.; Rui, M.; Zhang, L. A study on avoiding joint limits for inverse kinematics of redundant manipulators using improved clamping weighted least-norm method. *J. Mech. Sci. Technol.* **2018**, *32*, 1367–1378. [[CrossRef](#)]
8. Fan, Y.; Li, P.; Zhen, X.; Shen, Z. A Multiple Solutions Approach to the Inverse Kinematics Problem of a General Serial Manipulator. *J. Phys. Conf. Ser.* **2021**, *2002*, 012023. [[CrossRef](#)]
9. Buss, S.R. Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Trans. Robot. Autom.* **2004**, *17*, 16.
10. Tabandeh, S.; Melek, W.W.; Clark, C.M. An adaptive niching genetic algorithm approach for generating multiple solutions of serial manipulator inverse kinematics with applications to modular robots. *Robotica* **2010**, *28*, 493–507. [[CrossRef](#)]
11. Marcos, M.; Machado, J.; Azevedo-Perdicoulis, T.P. Trajectory planning of redundant manipulators using genetic algorithms. *Commun. Nonlinear Sci. Numer. Simul.* **2009**, *14*, 2858–2869. [[CrossRef](#)]
12. Abo-Hammour, Z.S.; Mirza, N.M.; Mirza, S.M.; Arif, M. Cartesian path planning of robot manipulators using continuous genetic algorithms. *Robot. Auton. Syst.* **2002**, *41*, 179–223. [[CrossRef](#)]
13. Singh, R.; Bhushan, B. Improved ant colony optimization for achieving self-balancing and position control for balancer systems. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 8339–8356. [[CrossRef](#)]
14. Chen, C.; Li, J.; Luo, J.; Xie, S.; Li, H. Seeker optimization algorithm for optimal control of manipulator. *Ind. Robot* **2016**, *43*, 677–686. [[CrossRef](#)]
15. Deng, H.; Xie, C. An improved particle swarm optimization algorithm for inverse kinematics solution of multi-DOF serial robotic manipulators. *Soft Comput.* **2021**, *25*, 13695–13708. [[CrossRef](#)]

16. Dereli, S.; Kker, R. A meta-heuristic proposal for inverse kinematics solution of 7-DOF serial robotic manipulator: Quantum behaved particle swarm algorithm. *Artif. Intell. Rev.* **2020**, *53*, 949–964. [[CrossRef](#)]
17. Mirjalili, S.; Mirjalili, S.M.; Lewis, A. Grey Wolf Optimizer. *Adv. Eng. Softw.* **2014**, *69*, 46–61. [[CrossRef](#)]
18. Nadimi-Shahraki, M.H.; Taghian, S.; Mirjalili, S. An improved grey wolf optimizer for solving engineering problems. *Expert Syst. Appl.* **2020**, *166*, 113917. [[CrossRef](#)]
19. Roy, R.G.; Ghoshal, D. Grey Wolf Optimization-Based Second Order Sliding Mode Control for Inchworm Robot. *Robotica* **2019**, *38*, 1539–1557. [[CrossRef](#)]
20. Dhargupta, S.; Ghosh, M.; Mirjalili, S.; Sarkar, R. Selective Opposition based Grey Wolf Optimization. *Expert Syst. Appl.* **2020**, *151*, 113389. [[CrossRef](#)]
21. Liu, Y.; Zhao, J.; Xie, B. Obstacle avoidance for redundant manipulators based on a Novel Gradient Projection Method with a functional scalar. In Proceedings of the 2010 IEEE International Conference on Robotics and Biomimetics, Tianjin, China, 14–18 December 2010; pp. 1704–1709.
22. Zhang, H.; Jin, H.; Liu, Y.; Yuan, H. An improved real-time tracking scheme based on gradient projection for kinematically redundant manipulators. In Proceedings of the 2017 2nd International Conference on Robotics and Automation Engineering (ICRAE), Shanghai, China, 29–31 December 2017; pp. 136–140.
23. Kang, J.I.; Choi, H.S.; Mai, T.V.; Duc, N.N.; Ji, D.-H.; Kim, J.-H. Experimental study of dynamic stability of underwater vehicle-manipulator system using zero moment point. *J. Mar. Sci. Technol.* **2017**, *25*, 767–774.
24. Ekinci, S.; Hekimolu, B.; Izci, D. Opposition based Henry gas solubility optimization as a novel algorithm for PID control of DC motor. *Eng. Sci. Technol. Int. J.* **2021**, *24*, 331–342. [[CrossRef](#)]
25. Zhong, G.; Wang, C.; Dou, W. Fuzzy adaptive PID fast terminal sliding mode controller for a redundant manipulator. *Mech. Syst. Signal Process.* **2021**, *159*, 107577. [[CrossRef](#)]
26. Parra-Vega, V.; Arimoto, S.; Liu, Y.-H.; Hirzinger, G.; Akella, P. Dynamic sliding PID control for tracking of robot manipulators: Theory and experiments. *IEEE Trans. Robot. Autom.* **2003**, *19*, 967–976. [[CrossRef](#)]
27. Hashim, F.A.; Houssein, E.H.; Mai, S.M.; Al-Atabany, W.; Mirjalili, S. Henry gas solubility optimization: A novel physics-based algorithm. *Future Gener. Comput. Syst.* **2019**, *101*, 646–667. [[CrossRef](#)]
28. Hashim, F.A.; Houssein, E.H.; Hussain, K.; Mabrouk, M.S.; Al-Atabany, W. A modified Henry gas solubility optimization for solving motif discovery problem. *Neural Comput. Appl.* **2020**, *32*, 10759–10771. [[CrossRef](#)]
29. Xie, W.; Xing, C.; Wang, J.S.; Guo, S.; Guo, M.-W.; Zhu, L.-F. Hybrid Henry Gas Solubility Optimization Algorithm Based on the Harris Hawk Optimization. *IEEE Access* **2020**, *8*, 144665–144692. [[CrossRef](#)]
30. Ravikumar, S.; Kavitha, D. CNN-oppositional-based Henry gas solubility optimization model for autonomous vehicle control system. *J. Field Robot.* **2021**, *38*, 967–979. [[CrossRef](#)]
31. Cao, W.; Liu, X.; Ni, J. Parameter Optimization of Support Vector Regression Using Henry Gas Solubility Optimization Algorithm. *IEEE Access* **2020**, *8*, 88633–88642. [[CrossRef](#)]
32. Falco, P.; Natale, C. On the Stability of Closed-Loop Inverse Kinematics Algorithms for Redundant Robots. *IEEE Trans. Robot.* **2011**, *27*, 780–784. [[CrossRef](#)]
33. Cao, H.R.; Gu, X.Q.; Yu, C.H. Obstacle Avoidance Algorithm for Redundant Manipulators Based on Weighted Generalized Inverse. *Appl. Mech. Mater.* **2017**, *872*, 303–309. [[CrossRef](#)]
34. Tian, X.; Xu, Q.; Zhan, Q. An analytical inverse kinematics solution with joint limits avoidance of 7-DOF anthropomorphic manipulators without offset. *J. Frankl. Inst.* **2021**, *358*, 1252–1272. [[CrossRef](#)]
35. Chan, T.F.; Dubey, R.V. A weighted least-norm solution based scheme for avoiding joint limits for redundant manipulators. *IEEE Trans. Robot. Autom.* **1995**, *11*, 286–292. [[CrossRef](#)]
36. Kiréanski, M.V.; Petrović, T.M. Combined Analytical-Pseudoinverse Inverse Kinematic Solution for Simple Redundant Manipulators and Singularity Avoidance. *Int. J. Robot. Res.* **2016**, *12*, 188–196. [[CrossRef](#)]
37. Khalil, W.; Dombre, E. Inverse kinematic model of serial robots. In *Modeling Identification and Control of Robots*; CRC Press: Boca Raton, FL, USA, 2002; pp. 85–115.
38. Li, Y.; Zhang, W.; Wang, L.; Zhao, F.; Han, W.; Chen, G. Henry's law and accumulation of weak source for crust-derived helium: A case study of weihe basin, China. *J. Nat. Gas Geosci.* **2017**, *2*, 333–339. [[CrossRef](#)]