

Article

Allocation Strategy Optimization Using Repulsion-Enhanced Quantum Particle Swarm Optimization for Multi-AUV Systems

Changjian Lin ¹ , Dan Yu ^{2,*} and Shibo Lin ³

¹ School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China; lin_changjian@cumt.edu.cn

² College of Electronics and Internet of Things Engineering, Chongqing Industry Polytechnic College, Chongqing 401120, China

³ School of Mechatronic Engineering, Changchun University of Technology, Changchun 130012, China; 20222630@stu.ccut.edu.cn

* Correspondence: cocomomo@hrbeu.edu.cn

Abstract: In the context of multi-autonomous underwater vehicle (multi-AUV) operations, the target assignment is addressed as a multi-objective allocation (MOA) problem. The selection of strategy for multi-AUV target allocation is dependent on the current non-cooperative environment. This paper establishes a multi-AUV allocation situation advantage evaluation system to assess and quantify the non-cooperative environment. Based on this framework, a multi-AUV target allocation model using a bi-matrix game theory is developed, where multi-AUV target allocation strategies are considered as part of the strategic framework within the game. The payoff matrix is constructed based on factors including the situational context of multi-AUV operations, effectiveness, and AUV operational integrity. The Nash equilibrium derived from the game analysis serves as the optimal solution for resource distribution in multi-AUV non-cooperative scenarios. To address the challenge of finding the Nash equilibrium in bi-matrix games, this paper introduces a repulsion process quantum particle swarm optimization (RPQPSO) algorithm. This method not only resolves the complexities of Nash equilibrium computation but also overcomes the limitations of traditional optimization methods that often converge to local optima. A simulation experiment of multi-AUV operations is designed to validate the multi-AUV target allocation model, demonstrating that the RPQPSO algorithm performs effectively and is applicable to multi-AUV task scenarios.



Citation: Lin, C.; Yu, D.; Lin, S. Allocation Strategy Optimization Using Repulsion-Enhanced Quantum Particle Swarm Optimization for Multi-AUV Systems. *J. Mar. Sci. Eng.* **2024**, *12*, 2270. <https://doi.org/10.3390/jmse12122270>

Academic Editor: Rafael Morales

Received: 7 November 2024

Revised: 1 December 2024

Accepted: 9 December 2024

Published: 10 December 2024



Copyright: © 2024 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/>).

Keywords: game theory; multi-AUV; target assignment; Nash equilibrium; repulsion processes quantum particle swarm optimization (RPQPSO)

1. Introduction

The target allocation problem in multi-AUV (autonomous underwater vehicle) operations represents a significant challenge in constrained combinatorial optimization, closely analogous to resource allocation problems in other domains. This problem, first introduced by [1] and established as NP-complete by [2] in 1986, is characterized by exponentially increasing solution times as the number of targets grows. Traditional enumeration-based algorithms [3] are therefore limited to addressing only small-scale problems. The central objective of target allocation is to minimize resource utilization while maximizing operational benefits, requiring an effective allocation of targets to tasks under various scenarios. Two critical aspects define the problem: the probability of AUV effectiveness, which determines the success of the allocation strategy, and a refined analytical description of the allocation process, which is essential for developing computational models and algorithms.

The target allocation problem can be categorized into static and dynamic allocation problems [4]. In static allocation, all resources are assigned at one stage, with complete knowledge of problem parameters, making the goal to optimize temporary operational tasks. In contrast, dynamic allocation involves multiple stages, where decisions are made

iteratively based on the outcomes of previous stages. In multi-AUV non-cooperative scenarios, the target allocation problem is inherently dynamic [5], and it can be modeled as a finite-state Markov decision process [6]. Unlike static problems, dynamic allocation considers not only the matching of resources to objectives but also the interrelations between decision phases, objectives, and resources, making it a complex triple-optimization problem [7]. Effective algorithms for this problem must meet three requirements: rapidity, rationality, and effectiveness of allocation.

Existing research on target assignment has primarily focused on two types of models: asset-based optimization models [6–11] and goal-based optimization models [12–14]. Ref. [15] proposed an asset-based dynamic target assignment optimization model, which was later extended by [16] to incorporate four key constraints: capability, strategy, resource, and feasibility. Dynamic target allocation has also been formulated as a Markovian decision process, with [6] applying approximate dynamic programming (ADP) to solve representative examples. Other works, such as [9,10], have developed integer linear programming models or linearization algorithms to address constraints and resource interdependencies. Additionally, ref. [17] introduced a hierarchical approach within the observe–orient–decide–act (OODA) framework for non-cooperative scenarios. In non-cooperative scenarios, asset-based models often rely on intelligence gathering and analysis, which can be challenging in real-world applications.

In non-cooperative environments, where asset-based models depend heavily on intelligence gathering, some researchers have shifted toward goal-based optimization models that focus on minimizing target threats or maximizing operational gains without relying on asset values. For instance, ref. [18] proposed a dual-target allocation optimization model balancing expected threat neutralization and resource expenditure, while ref. [19] developed a two-target multi-stage tasking model emphasizing sensor-platform synergy. Building on these approaches, ref. [12] defined an objective function that maximizes system security and engagement sequence stability. To address uncertainty, ref. [20] introduced a bi-objective optimization model using a conditional value-at-risk measure, and ref. [21] proposed a multi-objective optimization model with probability thresholds and time window constraints. Chandra et al. [22] explored the distributed decision-making problem in multi-agent systems and proposed a game-theoretic control barrier function-based distributed multi-robot navigation method. Other works, such as [23,24], explored nonlinear multi-objective planning and zero-sum game approaches, with [14] synthesizing these methods into a two-party zero-sum dual-matrix game model for multi-AUV scenarios. These models are particularly useful in environments where intelligence is limited or unreliable, allowing decision-makers to focus on high-level objectives rather than specific resource capabilities.

To solve dynamic target assignment problems, numerous algorithms have emerged, particularly with the advent of intelligent optimization techniques. Simulated annealing [25], genetic algorithms [9,26,27], ant colony optimization [28,29], and particle swarm optimization [30,31] have all been applied to this domain. Romano et al. [32] studied the social behavior of robots, providing a foundation for researching collective intelligence. These studies [33–35] reveal the influence of phenotypic traits, group structure, and dynamic principles on collective movement and behavioral decision-making. Hybrid algorithms have also gained traction: [36] combined ant colony optimization (ACO) with simulated annealing (SA), while [29] integrated ACO with greedy algorithms for large-scale problems. Novel approaches, such as an improved membrane swarm algorithm [37], an (ABC) algorithm using rule-based heuristic factors [38], and elite collaborative genetic algorithms [39], have been proposed to address challenges like slow convergence and low search efficiency. Additionally, ref. [40] developed an improved genetic algorithm for cross-regional collaborative task allocation, and ref. [21] introduced a hybrid multi-objective discrete particle swarm optimization algorithm for cooperative environments. Recently, ref. [41] proposed an adaptive simulated annealing–particle swarm optimization (SA–PSO) algorithm, integrating simulated annealing to enhance convergence speed and real-time performance in multi-objective target assignment. However, most existing algorithms focus on reducing the number of

iterations or improving convergence efficiency, often at the expense of robustness. This can lead to suboptimal solutions, especially in dynamic, non-cooperative environments where avoiding local optima is critical.

Despite these advancements, several gaps remain. Most existing models and algorithms focus on either defensive strategies or offensive surprise tactics, with limited attention to dynamic, non-cooperative target allocation models that simultaneously consider the strategies of both sides in multi-AUV tasks. Additionally, while many studies aim to improve computational efficiency, fewer efforts have been directed toward designing algorithms that effectively escape local optima and achieve globally optimal solutions in complex, real-world scenarios. To address these challenges, this paper makes the following contributions:

- We established a game model for the non-cooperative target allocation problem in multi-AUV tasks, creating an AUV dominance evaluation system and quantifying and analyzing the benefits of target allocation schemes.
- We proposed a Nash equilibrium solution algorithm for the non-cooperative target allocation game to address the computational challenges of the NP-hard problem.
- We improved the quantum particle swarm algorithm, which can be applied not only to the solution of Nash equilibrium problems but also to hard NP problems with “steep” non-global optimal solutions.

The remaining organizational parts of the article are as follows: Section 2 develops a dual matrix game model for multi-AUV target assignment. In Section 3, we design a repulsion-based quantum particle swarm Nash equilibrium solution algorithm. Finally, in Sections 4 and 5, simulation experiments are designed to verify the effectiveness and superiority of the proposed algorithm.

2. Game Modeling of Multi-AUVs Target Assignment Problem

To address the complex demands of target assignment in multi-AUV operations, particularly in non-cooperative scenarios, this paper adopts game theory principles by modeling the red and blue AUVs as participants in a strategic game. In multi-AUV operations, target assignment refers to the process of allocating specific tasks or objectives (targets) to each autonomous underwater vehicle (AUV) within a system to ensure the effective completion of the overall mission. This task becomes especially critical in non-cooperative scenarios, where opposing parties, such as red and blue AUV groups, act independently and potentially in conflict with one another. The importance of solving the target assignment problem lies in maximizing operational effectiveness while minimizing resource waste and conflict. In non-cooperative settings, understanding and predicting the behavior of the opposing side is crucial for achieving a strategic advantage. For example, a poor assignment of targets could lead to redundant efforts, missed mission objectives, or even jeopardizing the success of an operation due to adversarial interference. In this paper, we establish a situation advantage evaluation model using the operational information of both parties. The target assignment strategies of the red and blue AUVs are treated as game strategies, and the interactions are formalized as a bi-matrix game. A payoff function is constructed to evaluate the total gains and losses of both sides under various target assignment combinations, capturing the competitive dynamics of resource allocation and mission success. By solving the Nash equilibrium of the game, the optimal target assignment strategy for each side is determined, ensuring that no participant can unilaterally improve its outcome without disrupting the balance.

2.1. AUV Capability Model

The prerequisite for matching is the quantification of capability. In many existing studies, whether addressing AUVs or other unmanned systems in target allocation problems, there is a lack of detailed explanation regarding threat capabilities. Typically, a simple probability (often a fixed value) is used to represent it. While this approach has minimal impact on algorithms in static allocation problems, in dynamic allocation scenarios, the

distribution at different stages relies on feedback from the previous moment’s results. As the task duration increases, using fixed values to represent allocation effects can lead to cumulative errors that grow larger over time, resulting in significant deviations between experimental outcomes and actual situations. Therefore, before modeling the target allocation problem, it is important to provide a clear description of the AUV’s threat capability.

In actual gameplay, there are scenarios where multiple AUVs select the same target. This situation can be simulated as a simultaneous volley, ignoring the time interval of the shooting process, and the threat capability can be described using coverage domains. Assuming that all AUVs operate independently during this process, at time t , n AUVs select a target, and their hit probabilities, depending on their respective states, are denoted as q_1, q_2, \dots, q_n . The capability function of each AUV follows a zero-bias circular normal distribution, meaning that the capability of a single AUV can be expressed as follows:

$$P_K = 1 - e^{(-X/2)}, \tag{1}$$

where $X = \sum_{i=1}^n \frac{R_i^2}{\sigma_i^2}$, the standard error of the error variable in each direction is σ , and the threat radius of the AUV is R . The definition $X = \sum_{i=1}^n \frac{R_i^2}{\sigma_i^2}$ represents the capability of this group to simultaneously select targets.

In real-world scenarios, targets are often not singular, necessitating consideration of multiple targets when modeling the threat capability of AUVs. To simplify this study, the concept of a simultaneous “volley” scenario is introduced into the modeling process. Although achieving truly simultaneous strikes on multiple targets is challenging in practice, the underwater environment involves significant time costs for feedback on detection information as well as other unique characteristics. For instance, underwater communication is constrained by low efficiency and high latency due to the physical properties of acoustic communication. Additionally, AUVs in underwater environments face slower movement speeds and limited navigation and positioning accuracy, which add complexity to task planning and allocation. Therefore, the process can be simplified by assuming no immediate feedback after deploying the AUV, effectively treating the launches as “simultaneous”. In this context, “simultaneous” does not emphasize precise timing consistency but rather focuses on the absence of feedback after each AUV completes its mission, allowing for modeling as a single, unified engagement phase.

2.2. Bi-Matrix Game Model of Multi-AUVs Target Assignment

The two-person zero-sum bi-matrix game $\Gamma = (N, S_R^*, S_B^*; A, B;)$ can be viewed as an extension of the n -person finite game $\Gamma = (N, \{S_i\}, \{u_i\})$, where the game individual set is $N = \{R, B\}$, and R, B are red and blue parties composed of multi-AUVs. S_R^* and S_B^* are extensions of the strategy set $\{S_i\}$ to represent the mixed strategy set of game individuals, respectively, which are the target assignment schemes of red and blue. A, B are the extensions of the payoff function and the payoff matrix corresponding to the strategy, respectively. Under the mixed strategy (x, y) , the expected returns of individual R and individual B are, respectively, $E_R(x, y) = xAy^T$ and $E_B(x, y) = xBy^T$. (x^*, y^*) is a mixed Nash equilibrium of game Γ .

$$\begin{cases} xAy^{*T} \leq x^*Ay^{*T}, \forall x \in S_1^* \\ x^*By^T \leq x^*By^{*T}, \forall y \in S_2^* \end{cases} \tag{2}$$

In the calculation of multi-AUV target assignment, the computational difficulty increases significantly with the number of AUVs. Therefore, this paper introduces the concept of strategic superior super [32], firstly slimming down the payoff matrix.

Definition 1. In game $\Gamma = (N, \{S_i\}, \{u_i\})$, the strategies $s_g^{(i)}$ and $s_h^{(i)}$ of game i make

$$u_i(s_g^{(i)}, s_{-i}) > u_i(s_h^{(i)}, s_{-i}), \forall s_{-i} \in \prod_{k \neq i} S_k. \tag{3}$$

This indicated that strategy $s_g^{(i)}$ of game i is superior to strategy $s_h^{(i)}$.

The relevant corollaries for the solution of the Nash equilibrium mixed strategy are given below. $A_{.j}$ is the j th ($i = 1, 2, \dots, m$) column of A , and A_i ($j = 1, 2, \dots, n$) is the i th row of A .

Inference 1. Let $\Gamma = (S_1^*, S_2^*; A, B;)x^* \in S_1^* y^* \in S_2^*$ be a bi-matrix game; then, (x^*, y^*) is a mixed Nash equilibrium of Γ , IFF:

$$\begin{cases} A_i \cdot y^{*T} \leq x^* A y^{*T}, i = 1, 2, \dots, m \\ x^* B_{.j} \leq x^* B y^{*T}, j = 1, 2, \dots, n \end{cases} \tag{4}$$

$$\begin{cases} x^* A y^{*T} = \max_{1 \leq i \leq m} A_i \cdot y^{*T} \\ x^* B y^{*T} = \max_{1 \leq j \leq n} x^* B_{.j} \end{cases} \tag{5}$$

Theorem 1. Let (x^*, y^*) be a mixed Nash equilibrium of $\Gamma = (S_1^*, S_2^*; A, B;)$; then, there are complementary relaxation conditions:

$$\begin{cases} x_i^* (x^* A y^{*T} - A_i \cdot y^{*T}) = 0, i = 1, 2, \dots, m \\ y_j^* (x^* B y^{*T} - x^* B_{.j}) = 0, j = 1, 2, \dots, n \end{cases} \tag{6}$$

According to Definition 1, (x^*, y^*) is the mixed Nash equilibrium of bi-matrix game $\Gamma = (S_1^*, S_2^*; A, B;)$ only when (x^*, y^*) is the solution of the inequality set as follows:

$$\begin{cases} A_i \cdot y^{*T} \leq x^* A y^{*T}, i = 1, 2, \dots, m, \\ x^* B_{.j} \leq x^* B y^{*T}, j = 1, 2, \dots, n, \\ \sum_{i=1}^m x_i \leq 1, \\ \sum_{j=1}^n y_j \leq 1, \\ x_i \geq 0, i = 1, 2, \dots, m, \\ y_j \geq 0, j = 1, 2, \dots, n. \end{cases} \tag{7}$$

Definition 2. By solving the above equation, the Nash equilibrium mixed strategy (x^*, y^*) of the individual game can be obtained.

2.3. Multi-AUVs Confrontation Superiority Evaluation System

Based on the analysis of the multi-AUVs countermeasures problem, the situation advantage function based on the heading, velocity, and distance of both sides and the attack and defense efficiency advantage function based on the attack performance of both sides' AUVs is established to form the multi-AUVs attack and defense countermeasures dominance evaluation system, which provides the basis for the establishment of the payment matrix.

2.3.1. Situation Advantage Function

The red AUV cluster is $R = \{R_1, R_2, \dots, R_i\}$, where R_i represents the i^{th} red AUV. The blue AUV cluster is $B = \{B_1, B_2, \dots, B_j\}$, where B_j represents the j^{th} blue AUV. The offensive

and defensive situations of R_i and B_j are shown in Figure 1. In Figure 1, D_{ij} is the linear distance between two AUVs, and V_{R_i} and V_{B_j} are the velocity vectors of the red AUV and blue AUV, respectively. q_{ij} is the off-axis angle of R_i relative to B_j , and is defined as the entry angle of the target.

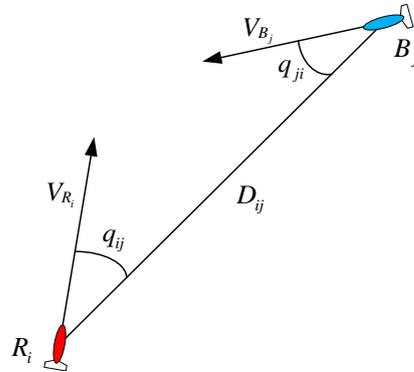


Figure 1. Antagonistic situation diagram.

The right deviation of the target bearing and target entry angle is positive, and the left deviation is negative. According to the confrontation situation between red and blue, the heading advantage is defined as follows:

$$W_a = \frac{1}{2} \left[1 + \frac{|q_{ij}| - |q_{ji}|}{\pi} \right]. \tag{8}$$

When $q_{ij} = 0$ and $q_{ji} = \pi$, the red AUVs have the advantage of angle attack. When $q_{ij} = \pi$ and $q_{ji} = \pi$, the two sides are in a state of equilibrium. When $q_{ij} = 0$ and $q_{ji} = 0$, both sides are at a disadvantage. When $q_{ij} = \pi$ and $q_{ji} = 0$, blue AUVs are in the dominant state.

The velocity advantage function is shown in Equation (9):

$$W_v = \begin{cases} 0.1 & V_{R_i} \leq 0.5V_{B_j} \\ 0.2 \left(\frac{V_{R_i}}{V_{B_j}} \right)^2 + 0.5 \left(\frac{V_{R_i}}{V_{B_j}} \right) - 0.2 & 0.5V_{B_j} < V_{R_i} < 1.5V_{B_j} \\ 1 & V_{R_i} \geq 1.5V_{B_j} \end{cases} \tag{9}$$

where V_{R_i} and V_{B_j} are the velocity vectors of the red AUV and blue AUV, respectively. The higher the velocity of the AUV, the greater the attack advantage will be.

The distance advantage function is shown in Equation (10):

$$W_r = \begin{cases} 0 & D_{ij} < R_{\min} \text{ and } D_{ij} > R_{\max} \\ -\frac{\sigma^2}{16}(D_{ij} - R_0)^2 + 1 & R_{\min} \leq D_{ij} \leq R_{\max} \end{cases}, \tag{10}$$

where D_{ij} is the linear distance between two red AUVs and blue AUVs, $\sigma = 2(R_{\max} - R_{\min})$, $R_0 = (R_{\max} + R_{\min})/2$; R_{\max} is the maximum effective range of the AUV, and R_{\min} is the minimum delivery distance of the AUV. When $D_{ij} > R_{\max}$, the distance advantage is considered to be zero, and the distance advantage gradually increases with the decrease in distance; when $D_{ij} = R_0$, the distance advantage reaches the maximum, and the distance advantage gradually decreases with the further decrease in distance.

According to the above frontal, velocity, and distance advantage functions, the multi-AUV offensive and defense situation advantage function is shown in Equation (11), where k_1, k_2, k_3 is the weighting coefficient, and $k_1 + k_2 + k_3 = 1$.

$$f(W_a, W_v, W_r) = k_1 e^{-\alpha_1 \cdot W_a} + k_2 \ln(1 + \beta_2 \cdot W_v) + k_3 \frac{1}{1 + \gamma_3 \cdot W_r}. \tag{11}$$

2.3.2. AUV Capability Efficiency Evaluation Function

The confrontation capability of the AUV is mainly measured by six factors: maneuver capability, detection capability, communication capability, load capability, stealth capability, and AUV performance. The calculation method is as follows:

$$C = [\ln B + \ln(\sum A_1 + 1) + \ln(\sum A_2)] \varepsilon_1 \varepsilon_2 \varepsilon_3, \tag{12}$$

where B is the maneuver capability parameter, A_1 is the performance parameter of the AUV threat capacity, A_2 is the detection performance parameter, ε_1 is the communication capability coefficient, ε_2 is the stealth performance coefficient, and ε_3 is the load capacity coefficient. After the standardized operation of Equation (12), the advantage function of red and blue AUVs can be obtained W_c .

$$W_c = \begin{cases} 0 & C_{R_i}/C_{B_j} < 0.3 \\ 0.25 & 0.3 \leq C_{R_i}/C_{B_j} < 1 \\ 0.5 & C_{R_i}/C_{B_j} = 1 \\ 0.75 & 1 \leq C_{R_i}/C_{B_j} < 1.5 \\ 1 & C_{R_i}/C_{B_j} \geq 1.5 \end{cases}, \tag{13}$$

where C_{R_i} and C_{B_j} are the target assignment game advantages of red and blue, respectively.

In summary, the advantage function of the attack and defense game of multi-AUVs is as follows:

$$W = k_s W_s + k_c W_c, \tag{14}$$

where W_s is the situational advantage function of both sides, and W_c is the advantage function of offensive and defensive effectiveness. k_s and k_c are the weighted coefficients, and $k_s + k_c = 1$.

2.4. Multi-AUVs Target Assignment Game Payment

Through the establishment of a multi-AUV target assignment dominance evaluation system, the basis of the multi-AUVs target assignment game payment matrix is provided. The overall revenue function of red and blue is as follows:

$$U^R = \sum_{i=1}^{N_R} \tilde{v}_i^{RR} L_i^R(K) - \sum_{j=1}^{N_B} \tilde{v}_j^{RB} L_j^B(K), \tag{15}$$

$$U^B = \sum_{j=1}^{N_B} \tilde{v}_j^{BB} L_j^B(K) - \sum_{i=1}^{N_R} \tilde{v}_i^{BR} L_i^R(K), \tag{16}$$

where U^R and U^B are the total offensive and defensive gains of the red and blue sides, respectively; N_R and N_B are the AUV numbers of red and blue, respectively. \tilde{v}_i^{RR} , \tilde{v}_j^{RB} (\tilde{v}_j^{BB} , and \tilde{v}_i^{BR}), respectively, represent the value return of UUV_{R_i} and UUV_{B_j} relative to red AUVs (blue AUVs); $L_i^R(K)$ are the survival probabilities of UUV_{R_i} and UUV_{B_j} after K steps, which are determined by the following:

$$L_i^R(k) = L_i^R(k-1) \cdot \prod_{j=1}^{N_B} [1 - p_{ji}^B(k)]^{u_{ji}^B(k)}, \tag{17}$$

$$L_j^B(k) = L_j^B(k-1) \cdot \prod_{i=1}^{N_R} [1 - p_{ij}^R(k)]^{u_{ij}^R(k)}, \tag{18}$$

$$p_{ij}^R(k) = W_{ij}(k) \cdot p_{ij}^R \cdot L_i^R(k-1), \tag{19}$$

$$p_{ji}^B(k) = W_{ji}(k) \cdot p_{ji}^B \cdot L_j^B(k - 1), \tag{20}$$

where $u_{ji}^B(k)$ and $p_{ji}^B(k)$, respectively, represent $k(k = 1, 2, \dots, K)$. The number of AUVs is used by the j th AUV of blue AUVs to attack the i th AUV of red AUVs and the damage hit probability of a single one; $u_{ij}^R(k)$ and $p_{ij}^R(k)$, respectively, represent the number of AUVs used by the i th red AUV to attack the j th blue AUV in step k and the damage probability of a single one. p_{ji}^B and p_{ij}^R represent the damage hit probability under ideal experimental conditions; $W_{ij}(k)$ and $W_{ji}(k)$ are the dominance coefficients of the i th AUV of red AUVs to the j th AUV of blue AUVs at the k th step and the dominance coefficients of the j th AUV of blue AUVs to the i th AUV of red AUVs at the k th step. Define $L_i^R(0) = 1$ and $L_j^B(0) = 1$, and the initial survival probability of both parties is 1. According to the control of both sides on the use of AUVs, the model constraint condition is determined as follows:

$$\sum_{k=1}^K \sum_{j=1}^{N_B} u_{ji}^B(k) \leq M_i^B, \forall i = 1, 2, \dots, N_R, \tag{21}$$

$$\sum_{k=1}^K \sum_{i=1}^{N_R} u_{ij}^R(k) \leq M_j^R, \forall j = 1, 2, \dots, N_B, \tag{22}$$

$$\sum_{i=1}^{N_R} u_{ji}^B(k) \leq m_j^B(k), \forall j = 1, 2, \dots, N_B, \tag{23}$$

$$\sum_{j=1}^{N_B} u_{ij}^R(k) \leq m_i^R(k), \forall i = 1, 2, \dots, N_R, \tag{24}$$

where M_i^B and M_j^R represent the maximum number of targets assigned by the red and blue parties to the red i^{th} boat and the blue j^{th} boat in the $K - round$ game. $m_i^R(k)$ and $m_j^B(k)$ represent the maximum number of targets sent by red i /blue j AUV in the $k - round$ game.

2.5. Nash Equilibrium of the Multi-AUVs Target Assignment Game

To address the Nash equilibrium problem in the bi-matrix game for target allocation, the process begins with optimizing the payoff matrix through strategic refinement to eliminate non-optimal strategies, thereby forming an optimal strategy set. Next, the feasibility of a pure strategy solution is assessed. If feasible, both parties select the pure strategy scenario as the optimal solution for target assignment at this stage. The mixed Nash equilibrium solution is then calculated using quantum particle swarm optimization based on a repulsion process. Targets are assigned according to the mixed Nash equilibrium, with assignments made probabilistically. Given that the Nash equilibrium solution for target assignment in multi-AUV non-cooperative scenarios is an NP-hard problem, this paper will introduce the repulsion process quantum particle swarm optimization (RPQPSO) algorithm to solve the Nash equilibrium in the subsequent chapter.

3. Repulsion Process Quantum Particle Swarm (RPQPSO) Hybrid Nash Equilibrium Algorithm

3.1. Principles of Quantum Particle Swarm Optimization (QPSO)

QPSO is a new particle swarm optimization model based on quantum mechanics theory. This algorithm is optimized for the shortcomings of PSO, such as the difference in local search ability, weak robustness, and inaccurate selection of initial parameters, and it enhances the ability of global search, fast parameter selection, and fast convergence. In the QPSO algorithm, each particle represents a possible solution, and the fitness of each particle is obtained by calculating the fitness payment function. Particles with quantum behavior can appear at any position in the whole feasible solution space, the position of occurrence is determined by the particle updating probability, and this position may have better fitness than the population optimal particle in the current population, so the algorithm can

achieve the optimal optimization in the feasible solution space. The algorithm principle is as follows:

$$m_{best}^k = \frac{1}{N_P} \sum_{t=1}^{N_P} p_{best_t}^k, \tag{25}$$

$$m_{best}^k = \frac{1}{N_P} \sum_{t=1}^{N_P} p_{best_t}^k, \tag{26}$$

$$X_t^{k+1} = P_t \pm \alpha |m_{best}^k - X_t^k| \ln\left(\frac{1}{u}\right), \tag{27}$$

where X_t^k is the position of the t particle at the k^{th} iteration, m_{best}^k is the middle position of the particle swarm at the k^{th} iteration, and r and u are random numbers between $[0, 1]$. The optimal value of the t^{th} particle is $p_{best_t}^k$, and the optimal solution of the whole population is g_{best} .

3.2. Design of the QPSO Algorithm

The mixed strategy of all game individuals is represented by each particle in the algorithm; that is, $X = \{x_1, x_2, \dots, x_n\}$. Since the generation of particles is random and may exceed the range of feasible solutions, the algorithm also designs the particle updating mechanism. By controlling the step size of algorithm iteration, the particles are guaranteed to always be in the feasible solution space during algorithm iteration.

Theorem 2. *If every particle is initialized in the space of possible solutions to the mixed strategy combination of the game, $\sum_{j=1}^{m_i} x_j^i = 1, x_j^i \geq 0, i = 1, 2, \dots, n$, and m_i is the number of pure strategies of game i . As long as the parameters of each particle in the algorithm are updated according to Equation (28), where $a_t \in [0, 1]$ is the largest step that keeps X_t^{k+1} within $\Omega = \{X | x^i \geq 0, \dots, n\}$, all particles can be guaranteed to remain in the feasible solution space of the mixed strategy combination of the game during the iteration process.*

$$X_t^{k+1} = P_t \pm a_t \cdot \alpha |m_{best}^k - X_t^k| \ln\left(\frac{1}{u}\right). \tag{28}$$

Proof. If the i th vector x_i in each initialized particle satisfies $\sum_{j=1}^{m_i} x_j^i = 1, x_j^i \geq 0, i = 1, 2, 3, \dots, n$, x_i is in a simplex, iterated according to Equations (25) and (26), where each particle is a linear combination of the positions of the previous generation of particles; that is, the i th vector in each particle is a linear combination of vectors in the hyperplane $\sum_{j=1}^{m_i} x_j^i = 1, i = 1, 2, 3, \dots, n$. Therefore, in the iteration of the algorithm, the i th vector of the particle will always remain in the hyperplane $\sum_{j=1}^{m_i} x_j^i = 1, i = 1, 2, 3, \dots, n$.

According to Equation (27), if X_t^{k+1} jumps out of region $\Omega = \{X | x^i \geq 0, i = 1, 2, 3, \dots, n\}$ in the iterative process, the following is true:

$$\alpha |m_{best}^k - X_t^k| \ln\left(\frac{1}{u}\right); \tag{29}$$

then, X_t^{k+1} can be kept within Ω by controlling step size $\beta_t \in [0, 1]$ in Equation (29):

$$X_t^{k+1} = P_t \pm \beta_t \cdot \alpha |m_{best}^k - X_t^k| \ln\left(\frac{1}{u}\right), \tag{30}$$

where $\beta_t \in [0, 1]$ is the maximum step length, which makes $X_t^{k+1} \geq 0$. $X_t^k \geq 0$ can always take β_t and make $X_t^{k+1} \geq 0$.

$$\left(X_t^k \right)_j^i + \beta_j^i \cdot \left(\alpha |m_{best}^k - \left(X_t^k \right)_j^i| \ln \left(\frac{1}{u} \right) \right) = 0, \quad (31)$$

$$i = 1, 2, \dots, n; j = 1, 2, \dots, m_i$$

Although $\beta_{i,(X_t^k)_j^i}$ is the corresponding element of X_t^k , and $\beta_t = \min \{ \beta_i^j \mid \beta_i^j \geq 0, i = 1, 2, \dots, n; j = 1, 2, \dots, m_i \}$, β_t is the biggest step that makes $X_t^{k+1} \geq 0$. \square

3.3. Local Extreme Repulsion Process

In the process of solving an extreme problem with a particle swarm or quantum particle swarm algorithm, the algorithm may converge to a local extreme that has been detected. In this paper, a local extreme repulsion process is designed to alleviate the multi-local extremum problem of the function.

Suppose $X^* = \{X_j^*; j = 1, 2, \dots, m\}$ is a series of extreme values detected by the algorithm, if some extreme values $X = \{X_i; i = 1, 2, \dots\} \subseteq X^*$ have been detected for the i^{th} particle X_i , calculate $\|X_i - X_j^*\|$ and check whether X_i is in the repulsion region, where r_{ij} represents the range of the repulsion region. If the particle is in the repulsion region, the particle will be repelled from the center of the local pole by the joint action of vector and particle X_i , where p_{ij} represents a pre-specified constant and Z_{ij} is the unary vector from X_j^* to X_i representing the force of repulsion, as shown in the Equation (32):

$$Z_{ij} = \frac{X_i - X_j^*}{\|X_i - X_j^*\|}. \quad (32)$$

The value of input is $X^*, S, r_{ij}, p_{ij} (i = 1, 2, \dots, N_p, j = 1, 2, \dots, m)$. The pseudo code of the Algorithm 1 is as follows:

Algorithm 1 RPQPSO algorithm

Input: Population size N_p
Output: Updated positions X_i
for $i = 1$ **To** N_p **do**
 if $X^* \neq 0$ **then**
 for $j = 1$ **To** m **do**
 $d_{ij} \leftarrow \|X_i - X_j\|$
 if $d_{ij} < r_{ij}$ **then**
 $Z_{ij} \leftarrow \frac{X_i - X_j^*}{\|X_i - X_j^*\|}$
 $X_i \leftarrow X_i + p_{ij} Z_{ij}$
 end if
 end for
 end if
end for

3.4. Flow of Algorithm

The algorithm flow for solving the mixed Nash equilibrium of the bi-matrix game with the RPQPSO is as follows:

- (1) Initialize the position information of particles and determine the particle population size N_p and particle dimension $\sum m_i$;
- (2) Calculate the m_{best} value of the middle position of the particle swarm;
- (3) Calculate the fitness of each particle, and select the particle with the optimal fitness as the optimal particle $p_{best,i}$;

- (4) The fitness of all p_{best} are compared, and the particle with the best fitness is selected as the global optimal particle g_{best} ;
- (5) For each dimension with particles, a random point P_i is obtained between g_{best} and p_{best_i} according to Equation (27);
- (6) Obtain a new position according to Equation (28);
- (7) Check whether the t^{th} particle meets the limit condition $X_t^{k+1} > 0$; otherwise, solve the control step β_t so that

$$X_t^{k+1} = P_t \pm \beta_t \cdot \alpha |mbest^k - X_t^k| \ln\left(\frac{1}{u}\right), \tag{33}$$

and make X_t^{k+1} return to the feasible mixed strategy space;

- (8) The repulsion process ejects particles entering the local extremum field;
- (9) Repeat steps 2–8 until the algorithm reaches the accuracy standard or the maximum number of iterations, and output the global optimal particle position and its fitness.

4. Simulation Experiment of the Target Assignment for the Multi-AUVs Task

To verify the RPQPSO algorithm within the non-cooperative game process of multi-AUV target assignment, this paper describes a multi-AUV simulation experiment. The game payoff matrix for multi-AUV target assignment in non-cooperative scenarios was established. To solve the mixed Nash equilibrium, a traditional particle swarm optimization algorithm was implemented for comparison with the RPQPSO algorithm proposed in Section 5. This comparison verified the algorithm’s real-time performance and accuracy. A multi-round game was constructed based on the multi-AUV target assignment model, verifying the feasibility of the solution method for the multi-AUV task allocation in non-cooperative scenarios.

4.1. Multi-AUVs Non-Cooperative Scenario Parameter Setting

In the unbounded calm (without the influence of waves and currents) seabed, there were 12 isomorphic AUVs in the red team, denoted as $AUV_{R_1}, AUV_{R_2}, \dots, AUV_{R_{12}}$, which constructed three AUV groups G_1^R, G_2^R, G_3^R . There were six isomorphic AUVs in the blue group, denoted as $AUV_{B_1}, AUV_{B_2}, \dots, AUV_{B_6}$, which constructed two AUV groups G_1^B, G_2^B . The experiments were conducted on a Windows system for algorithm computations, using VS2013 for numerical simulations and AnyLogic for visualization simulations.

The red AUVs were informed of the general position of the blue AUVs and moved towards them. Blue AUVs stayed on course. All the team members had good detection and communication performance, and the group maintained communication and shared information during the task. Tables 1 and 2 show the comparison of the parameters of the basic AUV performance and the performance carried out by the two sides. The initial velocity and pose assignment of the red and blue AUV groups are shown in Figure 2, where (x, y, ω, u) represents the coordinate (x, y) of the current northeast coordinate system of the AUV, the heading angle is ω , and the longitudinal velocity is u .

Table 1. AUV performance parameters of blue and red AUVs [42].

Parameter	Red AUV	Blue AUV
Max Thrust τ_u	150 N	200 N
Max Torque τ_r	20 N · m	30 N · m
Max Velocity u_{max}	13.5 kn	15.5 kn
Max Detection Radius D_R	10 nmile	12 nmile
Max Communication Range C_R	15 nmile	15 nmile
Max Load Number L_{max}	3	3

Table 2. Target parameters [42].

Player	Effectiveness Range	Velocity
Red AUV	2–3.5 nmile	80 kn
Blue AUV	5–4.5 nmile	80 kn

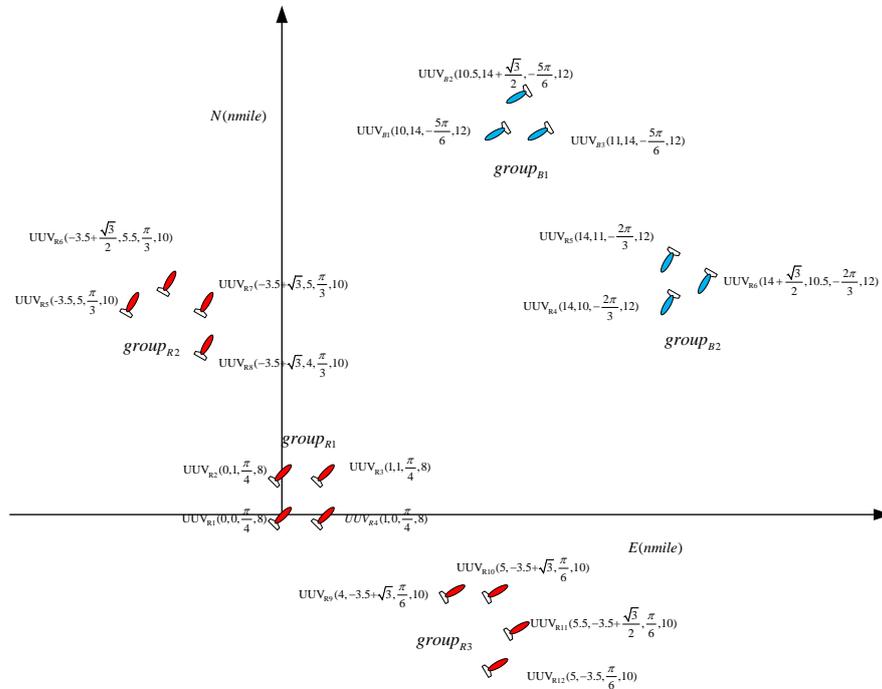


Figure 2. Multi-AUV confrontation initial situation diagram.

The AUV value parameter matrix of both sides is shown in Equations (33) and (34), where \tilde{v}_i^{RR} and \tilde{v}_j^{RB} (\tilde{v}_j^{BB} and \tilde{v}_i^{BR}), respectively, represent the value benefits of UUV_{Ri} and UUV_{Bj} relative to red AUVs (blue AUVs), which are obtained from intelligence information.

$$\tilde{v}^{BB} \begin{bmatrix} 0.8, 1.0, 0.8, 0.7, 0.9, 0.7 \end{bmatrix}, \quad \tilde{v}^{RB} \begin{bmatrix} 0.9, 1.1, 0.9, 0.9, 1.2, 0.9 \end{bmatrix}, \quad (34)$$

$$\tilde{v}^{RR} \begin{bmatrix} 0.6, 0.5, 0.7, 0.5, 0.5, 0.6, 0.6, 0.5, 0.5, 0.6, 0.6, 0.5 \end{bmatrix}, \quad \tilde{v}^{BR} \begin{bmatrix} 0.7, 0.6, 0.8, 0.6, 0.5, 0.7, 0.7, 0.5, 0.5, 0.7, 0.7, 0.5 \end{bmatrix}. \quad (35)$$

Constraints on the maximum number of targets sent by UUV_{Ri}/UUV_{Bj} in each step:

$$m_i^R(k) = m_j^B(k) = 1, i = 1, 2, \dots, 12, j = 1, 2, \dots, 6. \quad (36)$$

Blue AUVs/red AUVs maximum number of targets assigned to UUV_{Ri} and UUV_{Bj} constraints:

$$M_i^B = M_j^R = 2i = 1, 2, \dots, 12, j = 1, 2, \dots, 6. \quad (37)$$

4.2. First-Round Simulation of the Target Assignment

According to the proximity rule, the red group chooses the closest blue group to attack. In the situation shown in Figure 2, red group G_2^R attacks blue group G_1^B , red group G_3^R attacks blue group G_2^B , and red group G_1^R does not participate in the confrontation. The following takes the target assignment game in which red group G_2^R chooses to attack blue group G_1^B in the first round of attack and defense as an example.

According to the situation assessment system, the game advantage matrix of both sides can be obtained:

$$G_2^R(w) = \begin{matrix} & B_1 & B_2 & B_3 \\ R_5 & 0.8874 & 0.8890 & 0.8854 \\ R_6 & 0.8894 & 0.8894 & 0.8855 \\ R_7 & 0.8916 & 0.8931 & 0.8891 \\ R_8 & 0.8948 & 0.8961 & 0.8923 \end{matrix}, \tag{38}$$

$$G_1^B(w) = \begin{matrix} & R_5 & R_6 & R_7 & R_8 \\ B_1 & 0.9125 & 0.9127 & 0.9084 & 0.9051 \\ B_2 & 0.9109 & 0.9105 & 0.9069 & 0.9038 \\ B_3 & 0.9146 & 0.9145 & 0.9109 & 0.9077 \end{matrix}. \tag{39}$$

Red group G_2^R has $3^4 - C_3^1 C_2^1 C_4^3 - 3 = 54$ target assignment schemes, and blue group G_1^B has $4^3 - 4 = 60$ target assignment schemes. The two sides' task assignment schemes are expressed as Equations (40) and (41).

$$s_t^R = \begin{matrix} R_4 \\ R_5 \\ R_6 \\ R_7 \end{matrix} \begin{bmatrix} R_{4,1} & R_{4,2} & R_{4,3} \\ R_{5,1} & R_{5,2} & R_{5,3} \\ R_{6,1} & R_{6,2} & R_{6,3} \\ R_{7,1} & R_{7,2} & R_{7,3} \end{bmatrix}, t = 1, 2, \dots, 54, \tag{40}$$

$$s_t^B = \begin{matrix} B_1 \\ B_2 \\ B_3 \end{matrix} \begin{bmatrix} B_{1,4} & B_{1,5} & B_{1,6} & B_{1,7} \\ B_{2,4} & B_{2,5} & B_{2,6} & B_{2,7} \\ B_{3,4} & B_{3,5} & B_{3,6} & B_{3,7} \end{bmatrix}, t = 1, 2, \dots, 60. \tag{41}$$

In the equation, $R_{i,j}$ represents the number of targets assigned by $UUVR_i$ to $UUVB_j$, and $B_{i,j}$ represents the number of targets assigned by $UUVB_j$ to $UUVR_i$.

All the load assignment schemes of Equations (40) and (41) of red group G_2^R and blue group G_1^B and the advantage matrix formula refer to Equations (38) and (39), and we put them into Equations (19) and (20) to obtain the actual hit probability of the AUV of both parties. Then, we put Equation (19) into Equations (17) and (18) to obtain the predicted survival probability of blue and red AUVs, and finally, the payment matrix is obtained. Because the target assignment scheme of the two sides has 54 and 60 options, respectively, the payment matrix of both sides is a complex matrix of 54*60, which is difficult to calculate directly. Therefore, the strategy optimization is adopted to eliminate the strategy with low payment value, simplify the payment matrix of both sides, and finally, simplify as follows:

$$G_2^R(v) = \begin{matrix} & B_{18} & B_{28} & B_{44} \\ R_8 & -0.06677 & 0.11490 & -0.20258 \\ R_{28} & -0.06612 & 0.11425 & 0.11547 \\ R_{41} & -0.38482 & -0.20379 & 0.11612 \end{matrix}, \tag{42}$$

$$G_1^B(v) = \begin{matrix} & B_{18} & B_{28} & B_{44} \\ R_8 & 0.132515 & -0.37642 & -0.37784 \\ R_{28} & -0.16614 & -0.37698 & -0.37840 \\ R_{41} & -0.16558 & -0.07833 & -0.07975 \end{matrix}, \tag{43}$$

where R_i represents the i th strategy of the red group, and B_j represents the j th strategy of the blue group.

The scribing method is adopted to solve the Nash equilibrium. The maximum values of each column of $G_2^R(v)$ are (R_{28}, B_{18}) , (R_8, B_{28}) , (R_{41}, B_{44}) . The maximum values of each row of $G_1^B(v)$ are (R_8, B_{18}) , (R_{28}, B_{18}) , and (R_{41}, B_{28}) , respectively. It can be seen that (R_{28}, B_{18}) is the overlapping point, so (R_{28}, B_{18}) is the Nash equilibrium. The op-

timal mixed Nash equilibrium solution of the bi-matrix game was obtained by PSO as (0.0433, 0.9475, 0.0092; 0.9763, 0, 0.237), and the average time was 1.47 s. The optimal mixed Nash equilibrium strategy of the bi-matrix game was obtained by the RPQPSO as (0.0003, 0.9997, 0, 0.9999, 0, 0.0001), and the average time was 2.47 s. The calculation result of QPSO based on the repulsion process was selected as the optimal result. That is, the red side chose the 28th strategy with a probability of 0.9997, and the blue side chose the 18th strategy with a probability of 0.9999. The expected returns of red and blue AUVs implementing this game strategy were, respectively, (−0.06612, −0.16614).

If the G_1^B unilaterally chooses to execute the 44th strategy instead of the 18th strategy, the G_2^R will continue to adopt the 18th strategy and obtain an expected return of 0.11547, while the G_1^B can only obtain an expected return of −0.37840, which proves that the individual players who unilaterally do not abide by the Nash equilibrium can only lead to a reduction in benefits.

4.3. Simulation of the Multi-AUV Target Assignment Process

Because the game involves a delay for the targets to reach the designated positions, and the multi-AUV task execution is not accomplished instantly, it generally requires multiple rounds of interaction. When multi-AUVs engage in tasks based on target allocation, the expected benefits and target assignment for both sides in the first round of task execution can be determined. The expected rewards for the blue team and the AUVs are as follows:

$$G_2^R(v, k_1) = -0.0661197, G_3^R(v, k_1) = -0.145939, \tag{44}$$

$$G_1^B(v, k_1) = -0.166139, G_1^B(v, k_1) = -0.284513, \tag{45}$$

and the assignment is shown as follows:

$$G_2^R(s, k_1) = \begin{matrix} R_5 \\ R_6 \\ R_7 \\ R_8 \end{matrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \tag{46}$$

$$G_1^B(s, k_1) = \begin{matrix} B_1 \\ B_2 \\ B_3 \end{matrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{47}$$

$$G_3^R(s, k_1) = \begin{matrix} R_9 \\ R_{10} \\ R_{11} \\ R_{12} \end{matrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \tag{48}$$

$$G_2^B(s, k_1) = \begin{matrix} B_4 \\ B_5 \\ B_6 \end{matrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{49}$$

After each round assignment for a mission, both sides' AUVs maneuver. Players in the second round were as follows: G_2^R vs. G_1^B ; G_3^R vs. G_2^B . Since the distributive effect in the first round did not hit the target, neither side could determine the benefits of the first round of assignment. Therefore, the expected rewards were calculated by combining the distributive effect in the first round and the current maneuver situation.

$$G_2^R(v, k_2) = -0.097709, G_1^B(v, k_2) = -0.307278, \tag{50}$$

$$G_3^R(v, k_2) = 0.065799, G_2^B(v, k_2) = -0.253934. \tag{51}$$

The opposing parties choose the target assignment scheme according to probability, and the final target assignment is as follows:

$$G_2^R(s, k_2) = \begin{matrix} R_5 \\ R_6 \\ R_7 \\ R_8 \end{matrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \tag{52}$$

$$G_1^B(s, k_2) = \begin{matrix} B_1 \\ B_2 \\ B_3 \end{matrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{53}$$

$$G_3^R(s, k_2) = \begin{matrix} R_9 \\ R_{10} \\ R_{11} \\ R_{12} \end{matrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \tag{54}$$

$$G_2^B(s, k_2) = \begin{matrix} B_4 \\ B_5 \\ B_6 \end{matrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{55}$$

The first round of assignment effected by the red and blue parties and the positions of each AUV are shown in Figure 3. If the probability of survival is less than 0.4, the victim will be wounded and will not participate in the confrontation.

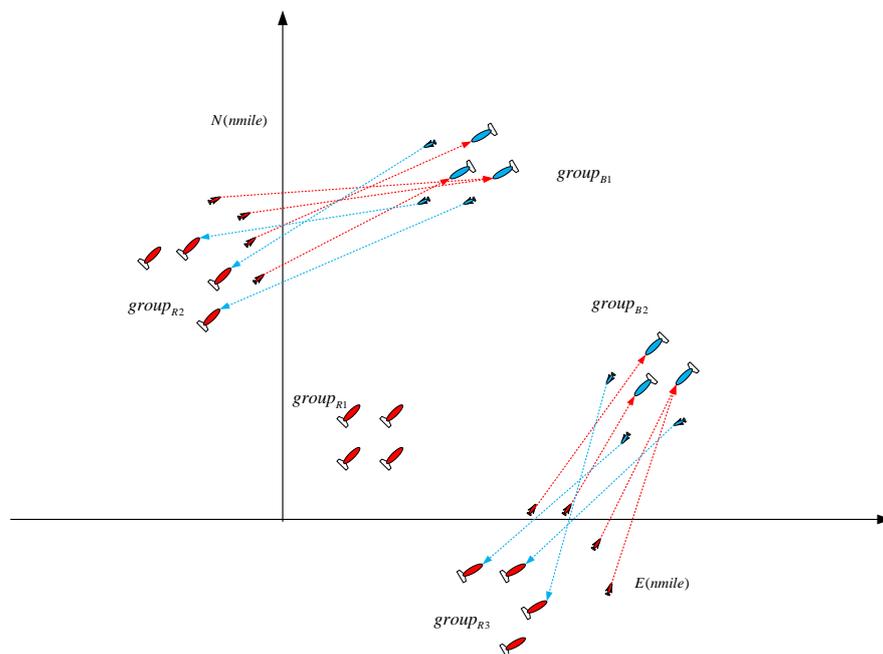


Figure 3. Situation of red and blue AUVs before the second load assignment.

When the second round of target assignment is completed, both sides begin to prepare for the third round of assignment. Before the third round of the game, the first round of engagement had already occurred, and its effects/consequences were known. Based on the probability of target neutralization and the probability of failure, the survival probability for both parties can be calculated as follows:

$$G_1^R(p, k_1) = [1 \quad 1 \quad 1 \quad 1], \tag{56}$$

$$G_2^R(p, k_1) = [0.45346 \quad 0.45264 \quad 0.45347 \quad 1], \tag{57}$$

$$G_3^R(p, k_1) = [1 \quad 0.45347 \quad 0.45264 \quad 0.45346], \tag{58}$$

$$G_1^B(p, k_1) = [0.55258 \quad 0.30844 \quad 0.55546], \tag{59}$$

$$G_2^B(p, k_1) = [0.55258 \quad 0.31057 \quad 0.45346]. \tag{60}$$

The status before the third round of assignment and the situation from the second round of target allocation can be assessed. It can be observed from the game results that the loss for the red group G_3^R is significant, leading G_3^R to withdraw from the game in the third round. Participation in the third round of the game involves G_2^R vs. G_1^B and G_1^R vs. G_2^B . Since the targets from the first round have been engaged, and the second round of targets have been deployed but have not yet reached the target area, the outcomes cannot yet be determined. Therefore, by combining the second round of target assignments with the results from the first round engagements, the expected returns for the second and third rounds can be calculated as follows:

$$G_1^R(v, k_3) = 1.048889, G_2^R(v, k_2) = 0.065799, \tag{61}$$

$$G_1^B(v, k_2) = -0.253934, G_2^B(v, k_3) = -1.482267. \tag{62}$$

The two sides choose the target assignment scheme according to the probability results obtained by the matrix game, and the final target assignment is as follows:

$$G_1^R(s, k_2) = \begin{matrix} R_1 \\ R_2 \\ R_3 \\ R_4 \end{matrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \tag{63}$$

$$G_2^B(s, k_2) = \begin{matrix} B_4 \\ B_5 \\ B_6 \end{matrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{64}$$

$$G_2^R(s, k_2) = \begin{matrix} R_5 \\ R_6 \\ R_7 \\ R_8 \end{matrix} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \tag{65}$$

$$G_1^B(s, k_2) = \begin{matrix} B_1 \\ B_2 \\ B_3 \end{matrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{66}$$

The final survival probability of each AUV group member on the red and blue sides is shown as follows:

$$G_1^R(p, k_3) = [0.45647 \quad 1 \quad 0.52135 \quad 1], \tag{67}$$

$$G_2^R(p, k_3) = [0 \quad 0.45483 \quad 0.23204 \quad 0.46782], \tag{68}$$

$$G_1^B(p, k_3) = [0 \quad 0.15332 \quad 0.26692], \tag{69}$$

$$G_2^B(p, k_3) = [0.31845 \quad 0 \quad 0.33421]. \tag{70}$$

At this point, the blue AUVs have lost the ability to continue the task, marking the end of the game. An integrated analysis is conducted on the target distribution over three rounds of tasks. For intuitive expression, the coding scheme for multi-round integration is designed according to group, sub-target, and multi-round integration, as illustrated in Figures 4 and 5. The first row in this diagram represents the AUV r_{21} in the red group G_R^2 in the first distribution and the AUV b_{21} in the blue group G_B^1 that are selected. Similarly, the blue target allocation in the figure is also represented according to these coding rules, and the 0 in the figure indicates that the allocation did not participate in this round or that the ability to continue allocation was lost in the previous round.

	Red_1				Red_2				Red_3			
	r_{11}	r_{12}	r_{13}	r_{14}	r_{21}	r_{22}	r_{23}	r_{24}	r_{31}	r_{32}	r_{33}	r_{34}
Round_1	0	0	0	0	12	12	13	11	21	23	22	22
Round_2	0	0	0	0	13	13	12	11	21	22	23	23
Round_3	21	23	23	21	0	12	13	12	0	0	0	0

Figure 4. Red target distribution.

	Blue_1			Blue_2		
	b_{11}	b_{12}	b_{13}	b_{21}	b_{22}	b_{23}
Round_1	22	21	23	33	32	34
Round_2	22	23	24	33	31	32
Round_3	21	0	23	0	12	13

Figure 5. Blue target distribution.

5. Analysis of Algorithm Superiority

In order to prove the effectiveness of the proposed algorithm, a comparison experiment was designed to compare the computational effects of RPQPSO, RPSO, QPSO, and PSO on dynamic target allocation. The PSO algorithm takes the particle population number as 20, the maximum iteration times as 1000, and $\omega_{max} = 0.9$, $\omega_{min} = 0.4$, $c_1 = 2$, $c_2 = 2$, and the updating formula ω adopts Equation (71).

$$\omega = \omega_{max} - (\omega_{max} - \omega_{min}) * \left(\frac{k}{K}\right)^2 \tag{71}$$

The QPSO algorithm takes the particle population as 20 and the maximum iteration times as 1000, and $a_{max} = 0.8$, $a_{min} = 0.6$, a uses a linear value reduction strategy. The process parameters of repulsion are, respectively, $r_{ij} = 0.2$, $p_{ij} = 1$.

$$\omega = \omega_{max} + (\omega_{max} - \omega_{min}) * \frac{K - k}{K} \tag{72}$$

The parameters in the formula refer to Section 2.4. In order to reduce the influence of random factors, the offline calculation of each algorithm was repeated 50 times under the same conditions. Figure 6 shows the comparison results of algorithm convergence. Figures 7–9 are the statistical tables of optimal solution and optimization time of the algorithm, respectively, where t_{min} represents the minimum operation time of the algorithm; t_{avg} represents the average running time of the algorithm; g_{min} represents the minimum number of iterations the algorithm runs; g_{avg} represents the average number of iterations the algorithm runs; c is the number of times to find the optimal solution.

Compared with the experimental results, it can be seen that the particle swarm optimization algorithm exhibits good convergence performance but is prone to falling into local optima, achieving the optimal solution only 5 times out of 50 experiments. In contrast, the quantum particle swarm optimization algorithm based on the repulsion process achieves the optimal solution 42 times out of 50 experiments. Its convergence curve is smoother, less prone to abrupt changes, and demonstrates stronger global convergence capability and higher accuracy. However, the convergence speed is slower, resulting in a correspondingly longer solving time.

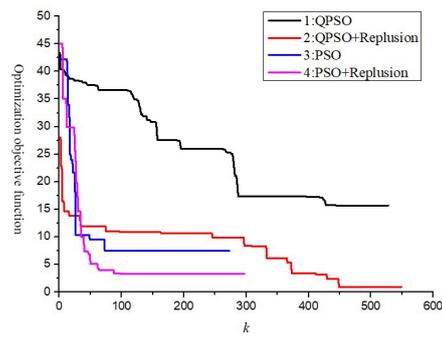


Figure 6. Algorithm performance comparison.

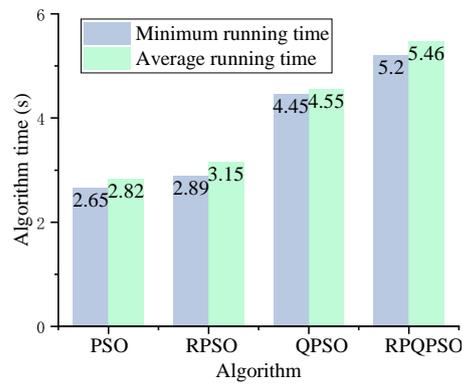


Figure 7. Algorithm time comparison.

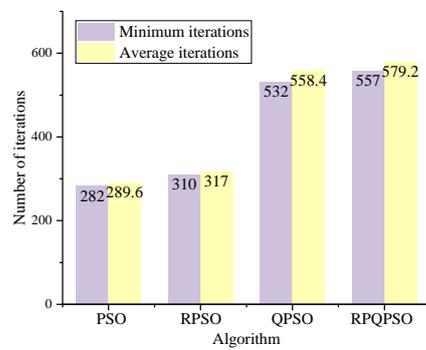


Figure 8. Algorithm iteration times comparison.

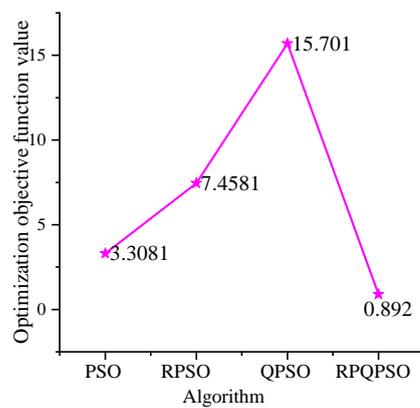


Figure 9. Algorithm optimization effect comparison.

In order to verify whether the error of early termination of the algorithm is small, the quantum particle swarm optimization algorithm based on the repulsion process is run 100 times under the above conditions. The optimal fitness f_s when the algorithm reaches stability and the optimal fitness f_m when the algorithm reaches the maximum number of iterations, as well as the running time t_s and the number of iterations g_s when the algorithm reaches stability, are obtained. Table 3 shows the statistical results of the effectiveness of the particle swarm optimization algorithm based on the repulsion process when it reaches stability, where f_{\max} is the maximum difference between f_s and f_m in 100 experiments; f_{avg} is the average of the difference between f_s and f_m .

Table 3. Target parameters.

Max Difference f_{\max}	Ave Difference f_{ave}	Running Time t_s (s)	Iterations g_s (gen.)
0.17	0.11	4.17	450

As can be seen from Table 3, the stable state detection of the quanta particle swarm algorithm based on the repulsion process can effectively reduce the running time of the algorithm and has an error of 0.17 at most, which has little influence on the error of solving accuracy and improves the real-time noise of the algorithm. In conclusion, the quantum particle swarm optimization algorithm based on the repulsion process proposed in this paper can solve mixed Nash equilibrium well, both concerning real-time and accuracy.

6. Conclusions

The target assignment problem has received extensive attention and research, but research on non-cooperative target assignment problems is still relatively limited. This paper studies target assignments in the context of multi-AUV non-cooperative tasks. Firstly, a bi-matrix game model of target assignment based on a multi-AUV system was proposed. Secondly, in order to solve the Nash equilibrium of the bi-matrix game effectively and avoid local optima, a quantum particle swarm optimization algorithm with local exclusion was designed to solve the Nash equilibrium. Finally, the advantages of the algorithm were analyzed in terms of the number of iterations and the fitness value of the optimal solution.

Experiments show that the multi-AUV non-cooperative target assignment model designed in this paper meets the requirements of non-cooperative scenarios and can objectively reflect real task conditions. The established AUV situation assessment system provides a foundation for constructing the payoff matrix and subsequent target assignment. Additionally, the dual matrix game-based non-cooperative target assignment model designed in this paper can adapt to dynamic target assignment tasks. Experiments verify that any deviation from the resolved Nash equilibrium solution by either party leads to a decline in returns, confirming the model’s effectiveness.

Since this paper starts from the perspective of both parties and uses target assignment as a means of managing non-cooperative scenarios, with Nash equilibrium as the final assignment strategy, solving the Nash equilibrium becomes the focus. However, it is challenging to solve the Nash equilibrium in practical situations. By designing a quantum particle swarm optimization algorithm based on a repulsion process, the Nash equilibrium solution meets the requirements of non-cooperative scenarios. A typical matrix game experiment is designed to verify the effectiveness and superiority of the proposed algorithm for solving Nash equilibrium in bi-matrix games.

Future research can further expand and enhance the outcomes of this study in the following aspects: first, more complex non-cooperative scenarios can be explored, such as multi-party games or mixed target assignment problems involving multiple types of tasks, to improve the model’s applicability and real-world relevance. Second, incomplete information game models could be introduced in dynamic environments, enabling AUVs to make smarter decisions when faced with dynamically changing task conditions or uncertain information. Additionally, algorithm performance can be further optimized

by incorporating methods such as deep learning or multi-agent reinforcement learning, thereby improving the efficiency and adaptability of Nash equilibrium solutions. Finally, in practical applications, simulation tests or the actual deployment of multi-AUV task systems can be conducted to validate the robustness and feasibility of the proposed approach. This would provide stronger technical support for fields such as ocean resource exploration, military confrontation, and environmental monitoring.

Author Contributions: D.Y. and C.L. designed the study, performed the research, analyzed the data, and wrote the paper. S.L. contributed to refining the ideas, carrying out additional analyses, and finalizing this paper. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China under Grant 62303467 and the Natural Science Foundation of Jiangsu Province under Grant BK20231061.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article.

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

References

1. Manne, A.S. A Target-Assignment Problem. *Oper. Res.* **1958**, *6*, 346–351. [\[CrossRef\]](#)
2. Lloyd, S.P.; Witsenhausen, H.S. Weapons Allocation is NP-Complete. In Proceedings of the 1986 Summer Computer Simulation Conference, Reno, NV, USA, 28–30 July 1986; pp. 1054–1058.
3. Dirik, N.; Hall, S.N.; Moore, J.T. Maximizing Strike Aircraft Planning Efficiency for a Given Class of Ground Targets. *Optim. Lett.* **2015**, *9*, 1729–1748. [\[CrossRef\]](#)
4. Kline, A.; Ahner, D.; Hill, R. The Weapon-Target Assignment Problem. *Comput. Oper. Res.* **2019**, *105*, 226–236. [\[CrossRef\]](#)
5. Xin, B.; Wang, Y.; Chen, J. An Efficient Marginal-Return-Based Constructive Heuristic to Solve the Sensor-Weapon-Target Assignment Problem. *IEEE Trans. Syst. Man Cybern. Syst.* **2019**, *49*, 2536–2547. [\[CrossRef\]](#)
6. Summers, D.S.; Robbins, M.J.; Lunday, B.J. An Approximate Dynamic Programming Approach for Comparing Firing Policies in a Networked Air Defense Environment. *Comput. Oper. Res.* **2020**, *117*, 104890. [\[CrossRef\]](#)
7. Davis, M.T.; Robbins, M.J.; Lunday, B.J. Approximate Dynamic Programming for Missile Defense Interceptor Fire Control. *Eur. J. Oper. Res.* **2017**, *259*, 873–886. [\[CrossRef\]](#)
8. Bogdanowicz, Z.R.; Patel, K. Quick Collateral Damage Estimation Based on Weapons Assigned to Targets. *IEEE Trans. Syst. Man Cybern. Syst.* **2014**, *45*, 762–769. [\[CrossRef\]](#)
9. Xin, B.; Chen, J.; Zhang, J.; Dou, L.; Peng, Z. Efficient Decision Makings for Dynamic Weapon-Target Assignment by Virtual Permutation and Tabu Search Heuristics. *IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.)* **2010**, *40*, 649–662. [\[CrossRef\]](#)
10. Lu, Y.; Chen, D.Z. A New Exact Algorithm for the Weapon-Target Assignment Problem. *Omega* **2021**, *98*, 102138. [\[CrossRef\]](#)
11. Andersen, A.C.; Pavlikov, K.; Toffolo, T.A.M. Weapon-Target Assignment Problem: Exact and Approximate Solution Algorithms. *Ann. Oper. Res.* **2022**, *312*, 581–606. [\[CrossRef\]](#)
12. Silav, A.; Karasakal, E.; Karasakal, O. Bi-Objective Dynamic Weapon-Target Assignment Problem with Stability Measure. *Ann. Oper. Res.* **2022**, *311*, 1229–1247. [\[CrossRef\]](#)
13. Kong, L.; Wang, J.; Zhao, P. Solving the Dynamic Weapon Target Assignment Problem by an Improved Multiobjective Particle Swarm Optimization Algorithm. *Appl. Sci.* **2021**, *11*, 9254. [\[CrossRef\]](#)
14. Ma, Y.; Wang, G.; Hu, X.; Luo, H. Two-Stage Hybrid Heuristic Search Algorithm for Novel Weapon Target Assignment Problems. *Comput. Ind. Eng.* **2021**, *162*, 107717. [\[CrossRef\]](#)
15. Lee, Z.-J.; Su, S.-F.; Lee, C.-Y. Efficiently Solving General Weapon-Target Assignment Problem by Genetic Algorithms with Greedy Eugenics. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2003**, *33*, 113–121.
16. Chen, J.; Xin, B.; Peng, Z.; Dou, L.; Zhang, J. Evolutionary Decision-Makings for the Dynamic Weapon-Target Assignment Problem. *Sci. China Ser. F Inf. Sci.* **2009**, *52*, 2006–2018. [\[CrossRef\]](#)
17. Zhang, K.; Zhou, D.; Yang, Z.; Zhao, Y.; Kong, W. Efficient Decision Approaches for Asset-Based Dynamic Weapon Target Assignment by a Receding Horizon and Marginal Return Heuristic. *Electronics* **2020**, *9*, 1511. [\[CrossRef\]](#)
18. Li, Y.; Kou, Y.; Li, Z.; Xu, A.; Chang, Y. A Modified Pareto Ant Colony Optimization Approach to Solve Biobjective Weapon-Target Assignment Problem. *Int. J. Aerosp. Eng.* **2017**, *2017*, 1746124. [\[CrossRef\]](#)
19. Xu, W.; Chen, C.; Ding, S.; Pardalos, P.M. A Bi-Objective Dynamic Collaborative Task Assignment under Uncertainty Using Modified MOEA/D with Heuristic Initialization. *Expert Syst. Appl.* **2020**, *140*, 112844. [\[CrossRef\]](#)
20. Li, J.; Xin, B.; Pardalos, P.M.; Chen, J. Solving Bi-Objective Uncertain Stochastic Resource Allocation Problems by the CVaR-Based Risk Measure and Decomposition-Based Multi-Objective Evolutionary Algorithms. *Ann. Oper. Res.* **2021**, *296*, 639–666. [\[CrossRef\]](#)

21. Peng, G.; Fang, Y.; Chen, S.; Peng, W.; Yang, D. A Hybrid Multi-Objective Discrete Particle Swarm Optimization Algorithm for Cooperative Air Combat DWTA. In Proceedings of the Bio-Inspired Computing–Theories and Applications: 11th International Conference, BIC-TA 2016, Xi’an, China, 28–30 October 2016; Revised Selected Papers, Part II 11; pp. 114–119.
22. Chandra, R.; Zinage, V.; Bakolas, E.; Biswas, J.; Stone, P. Decentralized multi-robot social navigation in constrained environments via game-theoretic control barrier functions 2023. *arXiv* **2023**, arXiv:2308.10966.
23. Hocaoglu, M.F. Weapon Target Assignment Optimization for Land Based Multi-Air Defense Systems: A Goal Programming Approach. *Comput. Ind. Eng.* **2019**, *128*, 681–689. [[CrossRef](#)]
24. Golany, B.; Goldberg, N.; Rothblum, U.G. A Two-Resource Allocation Algorithm with an Application to Large-Scale Zero-Sum Defensive Games. *Comput. Oper. Res.* **2017**, *78*, 218–229. [[CrossRef](#)]
25. Tosun, U. A New Tool for Automated Transformation of Quadratic Assignment Problem Instances to Quadratic Unconstrained Binary Optimisation Models. *Expert Syst. Appl.* **2022**, *201*, 116953. [[CrossRef](#)]
26. Erdem, E.; Ozdemirel, N.E. An Evolutionary Approach for the Target Allocation Problem. *J. Oper. Res. Soc.* **2003**, *54*, 958–969. [[CrossRef](#)]
27. Yang, S.; Yang, M.; Wang, S.; Huang, K. Adaptive Immune Genetic Algorithm for Weapon System Portfolio Optimization in Military Big Data Environment. *Clust. Comput.* **2016**, *19*, 1359–1372. [[CrossRef](#)]
28. Wang, T.; Fu, L.; Wei, Z.; Zhou, Y.; Gao, S. Unmanned Ground Weapon Target Assignment Based on Deep Q-Learning Network with an Improved Multi-Objective Artificial Bee Colony Algorithm. *Eng. Appl. Artif. Intell.* **2023**, *117*, 105612. [[CrossRef](#)]
29. Rezende, M.D.; De Lima, B.S.L.P.; Guimarães, S. A Greedy Ant Colony System for Defensive Resource Assignment Problems. *Appl. Artif. Intell.* **2018**, *32*, 138–152. [[CrossRef](#)]
30. Lai, C.-M.; Wu, T.-H. Simplified Swarm Optimization with Initialization Scheme for Dynamic Weapon-Target Assignment Problem. *Appl. Soft Comput.* **2019**, *82*, 105542. [[CrossRef](#)]
31. Zhang, L.; Zhang, Y.; Li, Y. Mobile robot path planning based on improved localized particle swarm optimization. *IEEE Sens. J.* **2020**, *21*, 6962–6972. [[CrossRef](#)]
32. Romano, D.; Stefanini, C. Robot-locust social information transfer occurs in predator avoidance contexts. *Int. J. Soc. Robot.* **2024**, *16*, 489–500. [[CrossRef](#)]
33. Romano, D.; Stefanini, C. Any Colour You Like: Fish Interacting with Bioinspired Robots Unravel Mechanisms Promoting Mixed Phenotype Aggregations. *Bioinspir. Biomim.* **2022**, *17*, 045004. [[CrossRef](#)] [[PubMed](#)]
34. Papageorgiou, D.; Farine, D.R. Group Size and Composition Influence Collective Movement in a Highly Social Terrestrial Bird. *eLife* **2020**, *9*, e59902. [[CrossRef](#)] [[PubMed](#)]
35. Ouellette, N.T. A Physics Perspective on Collective Animal Behavior. *Phys. Biol.* **2022**, *19*, 021004. [[CrossRef](#)]
36. Wang, J.; Gao, X.; Zhu, Y. Solving Algorithm for TA Optimization Model Based on ACO-SA. *J. Syst. Eng. Electron.* **2011**, *22*, 628–639. [[CrossRef](#)]
37. Yadav, K.; Alshudukhi, J.S.; Dhiman, G.; Viriyasitavat, W. iTSA: An Improved Tunicate Swarm Algorithm for Defensive Resource Assignment Problem. *Soft Comput.* **2022**, *26*, 4929–4937. [[CrossRef](#)]
38. Chang, T.; Kong, D.; Hao, N.; Xu, K.; Yang, G. Solving the Dynamic Weapon Target Assignment Problem by an Improved Artificial Bee Colony Algorithm with Heuristic Factor Initialization. *Appl. Soft Comput.* **2018**, *70*, 845–863. [[CrossRef](#)]
39. Huang, J.; Li, X.; Yang, Z.; Kong, W.; Zhao, Y.; Zhou, D. A Novel Elitism Co-Evolutionary Algorithm for Antagonistic Weapon-Target Assignment. *IEEE Access* **2021**, *9*, 139668–139684. [[CrossRef](#)]
40. Yu, X.; Gao, X.; Wang, L.; Wang, X.; Ding, Y.; Lu, C.; Zhang, S. Cooperative Multi-UAV Task Assignment in Cross-Regional Joint Operations Considering Ammunition Inventory. *Drones* **2022**, *6*, 77. [[CrossRef](#)]
41. Liu, S.; Liu, W.; Huang, F.; Yin, Y.; Yan, B.; Zhang, T. Multitarget Allocation Strategy Based on Adaptive SA-PSO Algorithm. *Aeronaut. J.* **2022**, *126*, 1069–1081. [[CrossRef](#)]
42. Wang, H.; Yu, D.; Xu, X.; Chen, T.; Zhang, H. Cooperative Defense and Confrontation Strategy for Multiple UUV Bases under Asymmetric Game. *Cogn. Syst. Res.* **2022**, *17*, 348–359.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.