Energy-Efficient Routing in UAVs Supported Perimeter Security Networks

Alexander K. Alexandrov [1,*], Anastass N. Madzharov [1]

[1]Institute of Robotics, Bulgarian Academy of Sciences, Acad. G. Bonchev str., 1113 Sofia, Bulgaria

Abstract

UAV-assisted ground and underwater perimeter security sensor networks represent a sophisticated integration of aerial, ground, and underwater technologies for surveillance and security purposes. This system combines Unmanned Aerial Vehicles (UAVs) with underwater sensors to monitor and protect strategic areas like harbors, offshore installations, and coastal facilities. Unmanned Aerial Vehicles (UAVs) have become pivotal in modern surveillance and security operations. Their versatility, mobility, and technological adaptability make them ideal for perimeter security systems. This study examines the integration of group of UAVs into perimeter security, evaluating their effectiveness, operational frameworks, technological advancements, and potential future developments. We analyze and implement a PSO (Particle Swarm Optimization) algorithm, related to group of UAVs trajectory optimization, review case studies, and identify key considerations for effective development.

Keywords

UAV, PSO, sensor network, perimeter security

1. Introduction

UAV-assisted underwater perimeter security sensor networks represent a cutting-edge blend of aerial and maritime technologies, designed to enhance the security of critical aquatic areas. This integration of Unmanned Aerial Vehicles (UAVs) and underwater sensors provides a robust solution for monitoring and safeguarding sensitive zones like naval bases, coastal areas, ports, and offshore installations.

The key components in the UAV-assisted underwater perimeter security sensor networks are the sensors, UAVs, and the control center.

Underwater sensors typically include acoustic sensors (such as sonars), geophones, hydrophones for detecting sound under water, and magnetic anomaly detectors for identifying metallic objects. These sensors continuously scan underwater environments to detect and track potential threats, like submarines, divers, or unmanned underwater vehicles (UUVs).

UAVs provide real-time aerial surveillance, significantly extending the range of observation beyond the immediate perimeter. They act as a vital link between the underwater sensors and the control center, especially important in deep-water areas where direct communication is difficult.

Control center provides data processing and decision making. Here the data from both UAVs and underwater sensors is analyzed, processed, and fused to form a comprehensive operational picture. Control center assesses potential threats based on the gathered information and coordinates appropriate responses.

One of the challenges in the UAVs assisted underwater perimeter security sensor networks is the energy management. Both the UAVs and underwater sensors must efficiently manage their power to ensure prolonged operational capabilities. The present study focuses on energy management, especially in energy-efficient and reliable routing of groups of UAVs. The UAVs energy-efficient routing is a multifaceted challenge that involves optimizing the flight paths and operational strategies of UAVs.

The objective is to maintain vigilant monitoring and rapid response capabilities while minimizing energy consumption, which is critical for the longevity and effectiveness of the UAVs in defense operations. The aim is to create routes and operational patterns that minimize energy usage while ensuring comprehensive security coverage.

Altitude and speed optimization in UAV-supported underwater perimeter security sensor networks is a critical aspect of ensuring energy-efficient routing and effective operation. The right balance of altitude and speed directly impacts the UAVs' energy consumption, coverage area, sensor effectiveness, and response times.

1.1. Altitude optimization

Higher altitudes can offer less air resistance, but the benefit must be balanced against increased energy requirements for climbing and maintaining altitude. Higher altitudes may increase the coverage area but could reduce the detail or accuracy of sensor data. The right altitude affects UAV performance in different weather

BISEC'23: 14th International Conference on Business Information Security, November 24, 2023, Niš, Serbia

*Corresponding author.
akalexandrov@ir.bas.bg (A. K. A.); a.madzharov@ir.bas.bg (A. N. M.)

conditions. For example, flying above or below certain weather layers (like fog or clouds) can be crucial.

The optimized altitude can ensure communication with both the underwater sensor network and the control station [1].

1.2. Speed optimization

Generally, faster speeds increase energy consumption. The optimization algorithm should identify the most energy-efficient cruising speed for each UAV model. Faster speeds allow for quicker coverage of an area but might reduce the effectiveness of sensors due to motion blur or reduced processing time.

Speed must be optimized to balance routine surveillance with the need for rapid response in case of detected threats [2, 3]. Tailwind can be exploited to reduce energy consumption, whereas flying into headwinds will require more energy, affecting optimal speed decisions.

2. Related works

There are some existing solutions related to the UAVs assisted underwater perimeter security sensor networks as:

- **DJI Enterprise Drones** - the solution is used for inspection and surveillance of commercial and military complexes. The drone is equipped with thermal imaging sensors, high-resolution cameras, and programmable flight paths and is programmed for routine patrols or dispatched upon alerts from ground and underwater sensors.
- **AeroVironment Raven RQ-11B** - the solution is used for battlefield reconnaissance and surveillance. The UAV is equipped with GA (Genetic Algorithms), based trajectory optimization system and interfaces with ground and underwater control systems and sensor networks.
- **Elbit Systems Skylark I-LEX** – this is electrically propelled UAV equipped with MPC (Model Predictive Control) trajectory optimization system, designed to collect data and interface with ground and underwater sensors for a comprehensive security net and is utilized by military and homeland security for national borders and sensitive areas.
- **Anduril Industries' Lattice** – this is a complete system that integrates drones, ground and underwater sensors, and AI-powered analysis to detect, classify, and respond to threats.
- **Asylon DroneCore** - automated drone deployment system that works with perimeter sensors to conduct autonomous patrols and respond to intrusions. The system is integrated with existing security infrastructure, providing a bird's-eye view when a ground sensor is triggered.
- **General Atomics Predator B** - used for national border surveillance, can be used in conjunction with ground sensor arrays for detecting and tracking movements and is equipped with high-resolution cameras and advanced signal intelligence equipment that can integrate with sensor network data.

All the mentioned UAVs have a custom design navigation systems with included energy-efficient software algorithms for routing and altitude/speed optimization, using various algorithms such as RL (Reinforcement Learning, Dynamic Programming, Dijkstra, GA (Genetic algorithms) in different combinations.

3. Proposed solution

The current research is focused on the development and implementation of altitude (elevation) and speed optimization algorithm in custom designed UAVs.

The proposed algorithm is based on PSO (Particle Swarm Optimization) [4, 5, 6]. This is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality.

It solves a problem by having a population of candidate solutions, here dubbed particles, and moving these particles around in the search-space according to simple mathematical formulae over the particle's position and velocity.

Each particle's movement is influenced by its local best known position but is also guided toward the best known positions in the search-space, which are updated as better positions are found by other particles. When applying PSO for altitude and speed optimization in UAVs supporting underwater perimeter security sensor networks, the goal is to determine the optimal flight paths, altitudes, and speeds for the UAVs to maximize coverage, efficiency, and responsiveness while minimizing energy consumption.

3.1. Challenges in the speed and elevation optimization

The following challenges related to the speed/elevation optimization problem were defined during the research:

- High Dimensionality: The speed/elevation optimization problem can be high-dimensional, especially when considering 3D space and time, making it computationally intensive [7].

- Dynamic Constraints: UAVs must respond to dynamic changes in the environment, which requires the PSO to be adaptable and responsive in real-time.
- Local Minima: The PSO algorithm may get trapped in local minima. This issue can be mitigated by tuning the parameters (ω, c_1, c_2) or by hybridizing PSO with other optimization techniques.
- Safety and Collision Avoidance: Ensuring safety is paramount. The algorithm must incorporate collision avoidance with other UAVs, terrain, and obstacles [8].

3.2. Implementation

mplementing a Particle Swarm Optimization (PSO) algorithm for altitude and speed optimization in UAV-supported underwater perimeter security sensor networks involves several mathematical concepts. Here's an mathematical overview of the proposed algorithm [9, 10, 11] :

Objective Function

Let's denote the objective function as $f(x)$, where x represents a vector of the decision variables (altitude and speed in this case) for UAVs. The function might aim to minimize energy consumption while maximizing area coverage, response time, or signal quality.

This could be a weighted sum or a more complex function based on the mission requirements.

Constraints

Include constraints like battery life (B), maximum and minimum altitude $(A_{\{\max\}}, A_{\{\min\}})$, and speed limits $(S_{\{\max\}}, S_{\{\min\}})$.

PSO Algorithm Structure

Particle Representation - each particle i in the swarm represents a potential solution, with its position pi indicating a particular set of altitudes and speeds for a UAV.

Initialization: randomly initialize the position pi and velocity vi of each particle within the feasible space defined by the constraints.

Velocity and Position Update Rules

Velocity update:

$$\begin{aligned} v_i^{t+1} = \omega v_i^{(t)} + c_1 r_1 \left(p_{best,i} - p_i^{(t)}\right) \\ + c_2 r_2 \left(p_{global\,best} - p_i^{(t)}\right), \end{aligned} \tag{1}$$

where ω is the inertia weight, c_1 and c_2 are cognitive and social coefficients, respectively, r_1, r_2 are random numbers between 0 and 1.

Position update:

$$p_i^{t+1} = p_i^{(t)} + v_i^{t+1}. \tag{2}$$

Ensure that the updated position adheres to the constraints.

Evaluation:

Evaluate the fitness of each particle using the objective function $f(x)$.

Update the personal best pbest,i if the current position of the particle yields a better value of the objective function. Update the global best $p_{global\,best}$ if any particle achieves a better value than the current global best.

Termination:

Continue iterating until a maximum number of iterations is reached or convergence criteria are met (e.g., minimal improvement in the global best).

Example Objective Function

Consider a simplified example where the objective is to minimize energy consumption E while ensuring good area coverage C. The objective function might look like this:

$$f\left(x\right) = \alpha E\left(x\right) - \beta C\left(x\right). \tag{3}$$

Here, α and β are weights reflecting the importance of energy consumption versus coverage.

The functions $E(x)$ and $C(x)$ compute the energy consumption and coverage based on the altitude and speed parameters in x.

The mathematical overview provided here is a simplified version of what could be a complex real-world implementation. In practice, the functions and parameters would need to be tailored to specific UAV capabilities, sensor characteristics, environmental factors, and mission goals.

Additionally, various enhancements to the basic PSO, such as constriction factors or varying inertia weight, might be employed to improve convergence and solution quality.

To implement the PSO algorithm for altitude and speed optimization in UAV-supported underwater perimeter security sensor networks, we will develop a structured pseudocode.

This pseudocode will help visualization the flow of the algorithm and serve as a guide for actual programming.

Remember that PSO is inherently iterative and works with a population of solutions, adjusting them over time based on a defined objective function.

The related PSO algorithm written in pseudocode is shown below:

```
PSO algorithm for UAVs elevation/speed
optimization
Inputs:
- num_particles: Number of particles in the
  swarm
- max_iterations: Maximum number of
  iterations
- objective_function: Function to optimize
  (minimize or maximize)
- A_max, A_min: Maximum and minimum
  allowable altitudes
- S_max, S_min: Maximum and minimum
  allowable speeds
- omega: Inertia weight
- c1, c2: Cognitive and social coefficients
Initialize:
- Create num_particles particles with random
  positions and velocities
- for each particle i:
      - position[i] = Random within
       [A_min, A_max] and [S_min, S_max]
      - velocity[i] = Random initial
      velocity
      - pbest[i] = position[i]
- gbest = position of the best particle based
  on objective_function
Main Loop:
- for iter = 1 to max_iterations:
   - for each particle i:
     - Update velocity:
       - r1, r2 = Random numbers
       between 0 and 1
       - velocity[i] = omega * velocity[i]
        + c1 * r1 * (pbest[i] - position[i])
         + c2 * r2 * (gbest - position[i])
     - Update position:
         - position[i] = position[i]
        + velocity[i]
         - Ensure position[i] adheres to
       [A_min, A_max] and [S_min, S_max]
     - Evaluate:
     - If objective_function(position[i]) is
      better than objective_function(pbest[i]):
         - pbest[i] = position[i]
     - If objective_function(position[i]) is
      better than objective_function(gbest):
         - gbest = position[i]
- Return gbest as the optimal solution
End Algorithm
```

Related to the pseudocode above please note:

Initialization: The initial positions and velocities are randomly assigned within the permissible ranges for altitude and speed. Each particle's initial position is considered its personal best (*pbest*).

Updating Velocities and Positions: The velocities are updated considering both the particle's own best position and the global best (*gbest*). The updated velocity influences the new position. It's important to ensure that the updated positions are within the allowed ranges.

Evaluating and Updating Best Positions: After updating positions, evaluate them using the objective_function. If a particle's new position is better than its *pbest*, update *pbest.* If it's better than the current *gbest*, update *gbest.*

Termination: The algorithm iterates through this process, gradually moving the swarm towards the best solution. The process repeats either until the maximum number of iterations is reached or some other stopping criterion (like a convergence threshold) is met.

Returning the Optimal Solution: Finally, the gbest after the last iteration is returned as the optimal set of altitude and speed parameters.

Customization for the specific use-case: The objective function should be designed specifically for the UAV's operational requirements, taking into account factors like energy consumption, area coverage, sensor effectiveness, and other mission-specific metrics.

Parameters such as ω, c_1, and c_2 may need tuning for optimal performance in specific scenarios.

Additional constraints or enhancements can be integrated into the algorithm based on specific requirements and operational environments.

4. Key Takeaways

Enhanced Efficiency: The PSO algorithm effectively optimizes UAV flight parameters (altitude and speed), leading to improved energy efficiency. This results in longer mission durations and reduced operational costs.

Adaptive Flight Paths: The algorithm's ability to dynamically adapt flight paths in response to changing environmental conditions and mission requirements is a significant advantage, ensuring optimal coverage and data collection.

Collaborative Functionality: PSO inherently supports multi-UAV coordination, allowing for effective swarm operations. This results in comprehensive area surveillance and redundant systems for critical defense missions.

Real-Time Decision Making: The implementation enables UAVs to make real-time adjustments, crucial for responding to emergent underwater threats or anomalies detected by the sensor network.

Operational Flexibility: The algorithm's flexibility allows it to be tailored to various mission scenarios, UAV types, and sensor network configurations, making it broadly applicable in underwater perimeter defense.

4.1. Challenges and Considerations

Complex Environmental Dynamics: The underwater and aerial environments present unique challenges, including variable weather conditions and underwater currents, which can affect the algorithm's performance.

Communication Limitations: Ensuring reliable communication between UAVs and underwater sensors remains a challenge, impacting the coordination and effectiveness of the network.

Computational Demands: PSO, especially in real-time applications, can be computationally intensive, necessitating robust onboard processing capabilities. Security and Robustness: The system must be secured against potential cyber threats and robust enough to handle operational uncertainties and potential system failures.

5. Conclusion

The implementation of a Particle Swarm Optimization (PSO) algorithm for altitude and speed optimization in UAVs supporting underwater perimeter security sensor networks is a sophisticated approach that leverages the strengths of swarm intelligence for operational efficiency. The conclusion drawn from this implementation can highlight its significance, potential benefits, and areas for future enhancement. Future steps:

- incorporating advanced variants of PSO or hybrid algorithms could further optimize performance, especially in highly dynamic or unpredictable environments.
- leveraging AI for predictive analytics and machine learning for continuous improvement of flight path algorithms based on historical data can enhance operational efficiency.
- incorporating sustainable technologies, such as solar-powered UAVs, can extend mission durations and reduce environmental impact.

The implementation of a PSO algorithm for optimizing a group of UAVs' altitude and speed in underwater perimeter security sensor networks demonstrates significant potential in improving maritime security operations [12]. While challenges remain, the continuous advancements in technology and algorithmic strategies hold promise for developing more sophisticated, efficient, and robust defense networks in the future . This approach exemplifies the innovative integration of aerial and maritime technologies, paving the way for enhanced security solutions in coastal and offshore environments.

In conclusion, PSO is a robust and versatile algorithm widely used for solving complex optimization problems. Its ongoing developments and applications across diverse fields highlight its relevance in the current technological landscape.

References

[1] H. He, S. Zhang, Y. Zeng, R. Zhang, Joint altitude and beamwidth optimization for uav-enabled multiuser communications, IEEE Communications Letters 22 (2017) 344–347.

[2] N. H. Chu, D. T. Hoang, D. N. Nguyen, N. Van Huynh, E. Dutkiewicz, Joint speed control and energy replenishment optimization for uav-assisted iot data collection with deep reinforcement transfer learning, IEEE Internet of Things Journal 10 (2022) 5778–5793.

[3] W. Ye, J. Luo, F. Shan, W. Wu, M. Yang, Offspeeding: Optimal energy-efficient flight speed scheduling for uav-assisted edge computing, Computer Networks 183 (2020) 107577.

[4] D. Wang, D. Tan, L. Liu, Particle swarm optimization algorithm: an overview, Soft computing 22 (2018) 387–408.

[5] O. F. Aje, A. A. Josephat, The particle swarm optimization (pso) algorithm application–a review, Global Journal of Engineering and Technology Advances 3 (2020) 001–006.

[6] M. Ehteram, A. Seifi, F. B. Banadkooki, Structure of particle swarm optimization (pso), in: Hellenic Conference on Artificial Intelligence, Springer, 2022, pp. 23–32.

[7] S. Schräder, D. Schleich, S. Behnke, Two-step planning of dynamic uav trajectories using iterative-spaces, in: International Conference on Intelligent Autonomous Systems, Springer, 2022, pp. 257–271.

[8] J. Hong, D. Chen, W. Li, Z. Fan, Trajectory planner for uavs based on potential field obtained by a kinodynamic gene regulation network, Sensors 23 (2023) 7982.

[9] R. Poli, J. Kennedy, T. Blackwell, Particle swarm intelligence. an overview, Swarm Intelligence 1 (2007) 33–57.

[10] Y. Yang, X. Xiong, Y. Yan, Uav formation trajectory planning algorithms: A review, Drones 7 (2023) 62.

[11] Q. Geng, Z. Zhao, A kind of route planning method for uav based on improved pso algorithm, in: 2013 25th Chinese control and decision conference (CCDC), IEEE, 2013, pp. 2328–2331.

[12] H. Zhang, B. Xin, L.-h. Dou, J. Chen, K. Hirota, A review of cooperative path planning of an unmanned

aerial vehicle group, Frontiers of Information Technology & Electronic Engineering 21 (2020) 1671–1694.