Вісник Тернопільського національного технічного університету
https://doi.org/10.33108/visnyk_tntu
Scientific Journal of the Ternopil National Technical University
2024, № 1 (117) https://doi.org/10.33108/visnyk_tntu2025.01
ISSN 2522-4433. Web: ***visnyk.tntu.edu.ua***

UDC 621.396.96

INTEGRATING SWARM INTELLIGENCE AND EDGE COMPUTING FOR AUTONOMOUS MULTI-DRONE OPERATIONS

Leonid Romaniuk[1]; Ihor Chykhira[1]; Halyna Tulaidan[2]; Andrii Holovko[1]

[1]Ternopil Ivan Puluj National Technical University, Ternopil, Ukraine
[2]Ternopil Volodymyr Hnatyuk National Pedagogical University, Ternopil, Ukraine

Abstract. *This study presents an adaptive PSO (Particle Swarm Optimization) algorithm as the foundation for a swarm intelligence approach in multi-UAV operations. The traditional PSO formula for particle velocity and position updates was modified to incorporate a variation strategy from Differential Evolution (DE), enabling UAVs to dynamically adjust their trajectories. The integration of deep reinforcement learning (DRL) further enhances the model's ability to optimize task offloading and computational distribution, ensuring that UAVs function as efficient edge nodes. An experimental evaluation was conducted to assess the proposed PSO-Edge method compared to other machine learning techniques, specifically Random Forest and Support Vector Machine (SVM). The experimental setup involved a simulation environment where UAVs were tasked to monitor data and execute missions over a defined area. The hardware included an Intel Xeon Gold 6248R CPU, 128 GB RAM, and an NVIDIA Tesla V100 GPU, with the simulation executed using Python 3.8. The proposed PSO-Edge algorithm demonstrated superior performance across multiple metrics: reducing task completion time by 42.1 minutes compared to Random Forest and SVM; achieving the lowest energy consumption per task at 28.9 Wh; demonstrating efficient communication with the least latency at 0.15 seconds; and achieving the highest task accuracy at 96%. The results confirm that the PSO-Edge method outperforms traditional machine learning approaches in task efficiency, energy consumption, communication latency, and accuracy. This highlights the benefits of integrating edge computing with the PSO algorithm, establishing it as a robust solution for multi-UAV operations. The findings have significant implications for optimizing UAV-based applications, particularly in environments requiring dynamic adaptation and efficient resource management.*

Key words: *autonomous trajectory optimization, deep reinforcement learning, multi-agent edge computing, collision avoidance metrics, real-time data processing, adaptive energy modulation.*

https://doi.org/10.33108/visnyk_tntu2025.01.067 *Received 20.01.2025*

1. INTRODUCTION

Operations involving multiple drones are becoming increasingly significant across various fields, from agricultural monitoring and disaster response to logistics and infrastructure inspection. However, as the scale and complexity of these operations grow, they face significant challenges in coordination, communication, and efficient data management. Traditional manual control is insufficient to address the dynamic nature of tasks involving multiple drones, particularly in unpredictable environments or over large areas requiring real-time decision-making. This makes automation not just a beneficial addition but a critical component for the effective operation of multiple unmanned aerial vehicles (UAVs).

To solve these problems, a promising solution lies in combining swarm intelligence and the concept of edge computing. Swarm intelligence enables drones to collaborate by mimicking the natural efficiency observed in animal groups, while real-time data processing ensures that drones respond swiftly to changing conditions. On the other hand, edge computing provides the capability for decentralized data analysis, allowing drones to process information locally without relying solely on external networks.

Individual UAV systems often encounter limitations such as restricted range, payload capacity, and sensing capabilities, making it difficult to efficiently perform large-scale operations. To overcome these constraints, the cooperative use of multiple UAVs, guided by the principles of swarm intelligence, ensures greater adaptability, reliability, and efficiency in task execution. Integrating the Particle Swarm Optimization (PSO) algorithm, Differential Evolution (DE), and intelligent sensing networks can significantly enhance the coordination and adaptability of multi-drone swarms, offering solutions for real-time data processing, dynamic trajectory planning, and adaptive control.

The study [1] addresses the limitations of using UAVs for bridge inspection, such as high noise levels and insufficient resolution in 3D bridge models created via photogrammetry. A novel 3D trajectory planning method based on Building Information Modeling (BIM) has been proposed to enhance UAV flight plans and improve the quality of photogrammetric models. This method uses a simplified BIM model of the bridge as input data, generating efficient UAV viewpoints while considering photogrammetry requirements and flight safety regulations. It subsequently adjusts inaccessible viewpoints and creates obstacle-free flight trajectories. The research under consideration validates the proposed method by testing it on a real beam bridge and comparing it with conventional UAV flight plans. The obtained results have demonstrated that the proposed method significantly reduces noise, improves model resolution, and enhances efficiency, resulting in higher-quality 3D bridge models suitable for damage detection with minimal human intervention. This method has potential for broader applications in reconstructing other types of infrastructure.

In study [2], an advanced trajectory planning method for UAVs is presented using an optimized Artificial Potential Field (APF) approach. By integrating a pre-planned trajectory generated by the Rapidly Expanding Random Tree (RRT) algorithm, the proposed method addresses common issues in traditional APF approaches, such as local minimum problems. The method under consideration has introduced some continuous particles and intermediate path points to create attractive forces that assist UAVs in avoiding local minima. Additionally, dynamic adjustments of gravity and repulsion coefficients, as well as non-gravity zones around obstacles, have been implemented to enhance obstacle avoidance. The improved APF method has been evaluated through simulations, demonstrating enhanced trajectory optimization and obstacle avoidance for both single and multiple UAVs.

Furthermore, noteworthy contributions by other researchers include Atamanchuk A. V. [3], Fomin I. I. [4], Tupytsya I. M., Kryvonos V. M., Kibitkin S. O., Ivashchuk L. A., Belivtsov A. O. [5], Oleksenko O. O., Avramenko O. V., Fedorov A. V., Snitsarenko V. V., Chernavina O. Ye. [6], Kartashov V. M. [7], Nekhin M., Kanevskyi L., Myronchuk Yu. [9], Wong S. Y., Choe C. W. C., Goh H. H., Low Y. W., Cheah D. Y. S., Pang C. [10], You H. E. [11], Lin C., Han G., Qi X., du J., Xu T., Martinez-Garcia M. [12], Elghitani F. [13], Simo A., Dzitac S., Dzitac I., Frigura-Iliasa M., Frigura-Iliasa F. M. [14], Deng Y., Zhang H., Chen X., Fang Y. [15] and many others.

2. CALCULATION METHODS

The objective of this work is to develop a method for automating the control and trajectory planning of multiple UAVs using swarm intelligence and edge computing.

The adaptive PSO algorithm serves as the foundation for the swarm intelligence approach to multi-drone operations. The original PSO formula for updating particle velocity and position is modified to incorporate a variation strategy from Differential Evolution (DE), enabling drones to dynamically adjust their trajectories:

$$v_i^{k+1} = \omega v_i^k + c_1 r_1 (p_i - x_i^k) + c_2 r_2 (g - x_i^k) \tag{1}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{2}$$

where v_i^k is the velocity of the particle i on iteration k, ω is inertia weight which is dynamically adapted by means of $\omega(k) = \omega_{max} - (\omega_{max} - \omega_{min})\frac{k}{K}$, c_1, c_2 – are learning factors that are adjusted on the basis of iteration ,r_1, $к_2$ – are random values from 0 to 1, p_i – is the best known position of the particle i, g– is the best global position.

To introduce the mutation strategy mediated by Differential Evolution (DE), the velocity is updated using the formula:

$$v_i^{k+1} = v_i^k + F \times \left(p_i - x_i^k\right) + F(x_{r_1} - x_{r_2}) \tag{3}$$

To optimize the trajectory and energy efficiency of a multi-drone swarm, the fitness function incorporates multiple objectives, expressed as:

$$f(x) = w_1 f_t + w_2 f_e + w_3 f_{col} \tag{4}$$

f_t minimizes the total travel time for all UAVs, f_e accounts for energy consumption calculated using the kinetic energy equation $E = \frac{1}{2} m v_2$,f_{col} – is a penalty function that prevents collisions between drones w_1, w_2, w_3– are weights that determine the priority of each objective.

The integration of an intelligent sensing network using wireless sensor networks (WSN) enhances the autonomy and adaptability of UAVs. This ensures accurate localization and real-time communication between drones.

$$f_m = R \cdot MN \cdot \sum_{i=1}^{n} (V_i - V_a)^2 \tag{5}$$

R – is the rotation matrix representing the drone's orientation, MN corresponds to the number of anchor nodes, and,V_i – are the position vectors of the unknown nodes.

The approach based on DRL (Deep Reinforcement Learning) has been integrated with the PSO algorithm to optimize the offloading of computational tasks to UAVs acting as edge nodes. The proposed method employs a multi-agent DRL structure, where each UAV operates as an independent agent, learning optimal strategies for data offloading and task allocation through a combination of local observation and communication with neighboring drones. The reward function R_t for each UAV agent has been designed to maximize the overall system utility while minimizing the risks of data interception and is calculated as follows:

$$R_t = \gamma \cdot (U_{sys} - P_{eav}) \tag{6}$$

where U_{sys} represents the system utility derived from efficient task offloading and computation. P_{eav} indicates the probability of data interception. γ – is the discount factor that balances immediate and future rewards. The DRL algorithm adjusts the UAV's position and transmission power to ensure secure communication, even in environments with significant interference or potential eavesdropping threats.

The implementation of autonomous control based on swarm intelligence is carried out using Python, integrating the PSO algorithm with adaptive learning and perception capabilities, as shown in Figure 1.

Edge computing in operations with multiple drones requires a multi-level architecture where UAVs equipped with edge computing capabilities form a hierarchical communication network with ground IoT devices and central servers. This architecture enhances data processing efficiency, reduces latency, and improves decision-making accuracy. The communication model is presented in the form of:

$$S_c = \alpha_1 T_{loc} + \alpha_2 T_{off} + \alpha_3 E_u \quad (7)$$

S_c – is the total efficiency of the system. T_{loc} is the local computation time. T_{off} represents the offloading time to the UAV. E_u denotes the energy consumed by the UAV during computations. The weights $\alpha_1, \alpha_2, \alpha_3$ are adaptively adjusted based on real-time performance metrics, enabling the swarm to optimize energy consumption, task execution time, and communication efficiency.

By utilizing edge computing nodes, task delegation, sensor data processing, and real-time decision-making are achieved. The onboard edge node of each UAV processes environmental data, updates its trajectory, and communicates with neighboring drones to adapt to changing task requirements.

```
import numpy as np
from sklearn.feature_selection import SelectKBest, f_classif

def adaptive_pso_feature_selection (X, y, num_particles=30, iterations=100):
    num_features = X.shape[1]
    particles= np.random.randint(2, size=(num_particles, num_features))
    velocities = np.zeros((num_particles, num_features))
    personal_best_positions= particles.copy()
    personal_best_scores= np.zeros(num_particles)

    for i in range(num_particles):
        selected_features= particles[i]
        selected_X = X[:, selected_features== 1]
        if selected_X.shape[1]== 0:
        continue

        k_best= SelectKBest (score_func=f_classif, k='all').fit(selected_X, y)
        score= k_best.scores_.sum()
        personal_best_scores[i] = score

    for iteration in range(iterations):
        for i in range(num_particles):
            velocities[i] = 0.5* (personal_best_positions[i] - particles[i]) + 0.3
*np.random.rand(num_features)
            particles [i] = np.where (np.random.rand(num_features) < velocities[i], 1, 0)

            selected_features= particles[i]
            selected_X= X[:, selected_features ==1]
            if selected_X.shape[1]== 0:
            continue

            score = SelectKBest (score_func=f_classif, k='all').fit(selected_X, y).scores_.sum()
            if score > personal_best_scores[i]:
              personal_best_scores[i] = score
              personal_best_positions[i]= particles [i]

best_particle_index = np.argmax(personal_best_scores)
 return np.where (personal_best_positions [best_particle_index] ==1) [0]
```

Figure 1. Implementation of PSO in the context of multi-drone operations

To further optimize multi-drone operations, UAVs equipped with edge computing leverage adaptive Particle Swarm Optimization (PSO) with embedded intelligence. The velocity update formula (1) integrates real-time edge processing feedback, ensuring that each UAV rapidly adapts to environmental changes:

$$v_i^{k+1} = \omega v_i^k + c_1 r_1 (p_i - x_i^k) + c_2 r_2 (g - x_i^k) + E_f \tag{9}$$

where E_f represents the feedback from edge nodes, indicating real-time adjustments required for task execution, obstacle avoidance, or energy optimization. Figures 2–3 demonstrate the integration of the edge computing concept into the multi-drone PSO algorithm.

```
class Drone EdgeNode:
    def __init__(self, position, energy, processing_capacity):
        self.position = position
        self.energy = energy
        self.processing_capacity = processing_capacity
        self.task_queue = []

    def process_task(self, task):
        if self.energy > task.energy_requirement and len(self.task_queue)
< self.processing_capacity:
            self.task_queue.append(task)
            self.energy  - = task.energy_requirement
            return True # Task assigned successfully
        return False # Insufficient energy or capacity

    def update_position(self, swarm_position, edge_feedback):
        inertia_weight = 0.5 # Adaptive value based on edge feedback
        new_velocity = inertia_weight * self.velocity + edge_feedback
        self.position += new_velocity
```

Figure 2. DroneEdgeNode class

The DroneEdgeNode class represents a UAV equipped with edge computing capabilities. This class includes methods for task processing (process_task) by checking energy levels and processing power availability to ensure efficient execution and methods for a drone position updating (update_position) based on feedback received from the swarm and the edge computing network, facilitating adaptive adjustments in dynamic environments.

This class encapsulates both computational and positional functionalities essential for coordinated multi-drone operations within the PSO-Edge framework.

```
def offload_task_to_edge(swarm, task):
    best_node = None
    max_reward = -float('inf')
    for drone in swarm:
        expected_reward = drone.predict_reward(task) # DRL-based prediction
        if expected_reward > max_reward and drone.process_task(task):
            max_reward = expected_reward
            best_node= drone
    return best_node
```

Figure 3. Offload_task_to_edge function

The offload_task_to_edge function demonstrates how tasks can be offloaded to the most suitable drone using Deep Reinforcement Learning (DRL). The function iterates through the swarm of drones, predicting the expected reward for offloading a task to each drone. It selects the drone offering the highest reward while ensuring the task can be processed, thereby optimizing task allocation within the swarm.

3. RESULTS AND DISCUSSION

To evaluate the proposed PSO-Edge method compared to other machine learning methods (Random Forest and SVM), an experiment was conducted in a controlled simulation environment. The objective was to assess task completion efficiency, energy consumption, communication latency, and task completion accuracy.

The experiment involved a simulated environment where UAVs were required to monitor data and perform tasks over a 200 x 200 m area. Metrics evaluated included task execution time, energy consumption, communication latency, and task execution accuracy. The hardware setup included an Intel Xeon Gold 6248R processor, 128 GB of RAM, and an NVIDIA Tesla V100 GPU. The simulation was conducted using Python 3.8 with appropriate machine learning libraries.

Figure 4 shows that the PSO-Edge method significantly outperformed other methods, reducing task execution time to 42.1 minutes, compared to 62.3 minutes with Random Forest and 58.7 minutes with SVM.

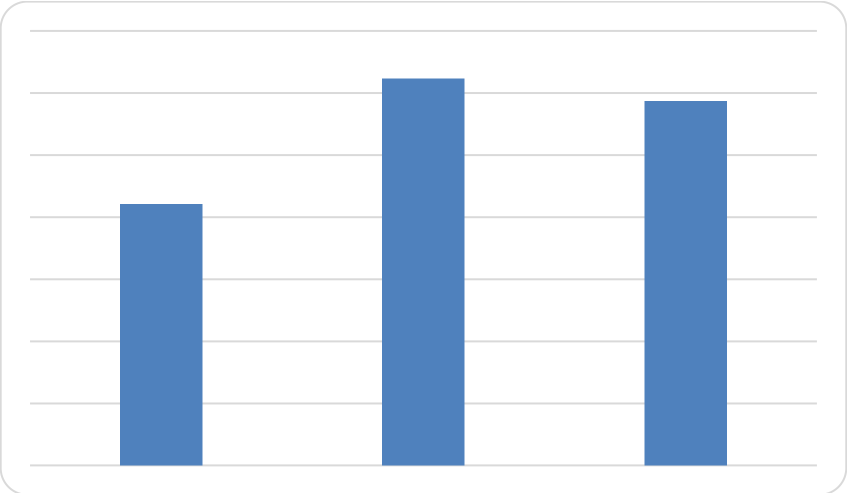

Figure 4. Task Completion Efficiency

Figure 5 demonstrates that the PSO-Edge method achieved the lowest energy consumption–28.9 Wh per task–compared to Random Forest (37,8 Wh) and SVM (34,5 Wh).

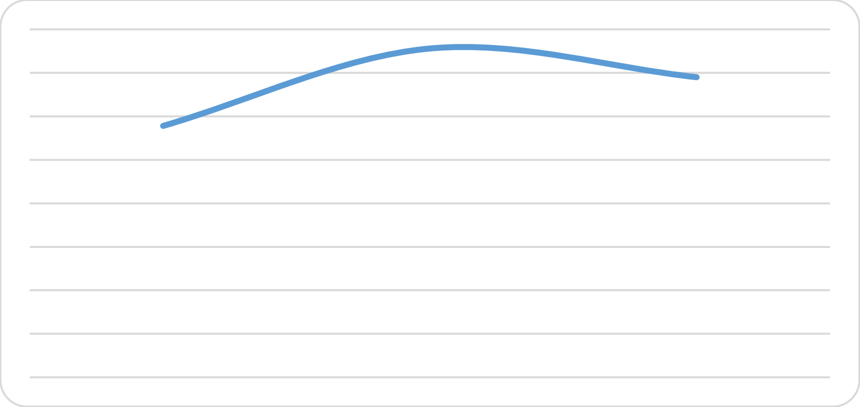

Figure 5. Energy Consumption per Task

As shown in Figure 6, the PSO-Edge method demonstrated the lowest communication latency (0.15 seconds), exceeding Random Forest (0.35 seconds) and SVM (0.28 seconds), indicating more efficient data transmission.

Figure 6. Communication Latency

Figure 7 demonstrates that the PSO-Edge method achieved the highest task execution accuracy (96%) compared to Random Forest (83%) and SVM (87%).

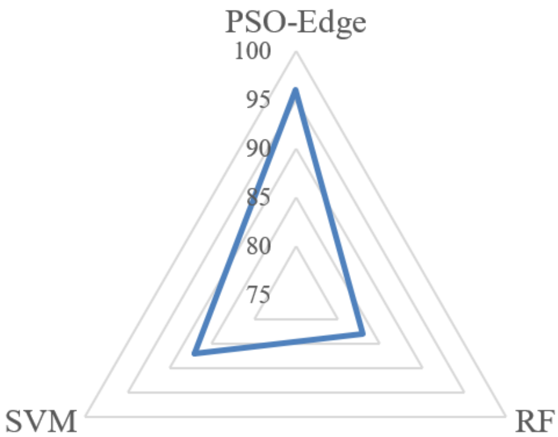

Figure 7. Task Completion Accuracy

The results confirm that the proposed PSO-Edge method outperforms traditional machine learning methods (Random Forest and SVM) in terms of task competion efficiency, energy consumption, communication latency, and task accuracy. These improvements highlight the advantages of integrating edge computing with the PSO algorithm, making it a reliable solution for multi-UAV operations.

4. CONCLUSIONS

In summary, the adaptive PSO-Edge algorithm demonstrates significant advantages over traditional machine learning methods, such as Random Forest and SVM, in optimizing multi-UAV operations. By integrating Differential Evolution strategies and deep reinforcement learning, the proposed method achieves superior performance in task execution time, energy efficiency, communication latency, and accuracy. These findings validate the effectiveness of combining swarm intelligence with edge computing, establishing the PSO-Edge approach as a highly efficient solution for dynamic and resource-intensive UAV-based applications.

References

1. Wang F., Zou Y., del Rey Castillo E., Ding Y., Xu Z., Zhao H.-W., Lim J. (2022). Automated UAV Path-Planning for High-Quality Photogrammetric 3D Bridge Reconstruction. Structure and Infrastructure Engineering, pp. 1–20. https://doi.org/10.1080/15732479.2022.2152840
2. Lu X. (2023). Improved Path Planning Method for Unmanned Aerial Vehicles Based on Artificial Potential Field. Applied and Computational Engineering, pp. 64–71. https://doi.org/10.54254/2755-2721/10/20230142
3. Atamanchuk A. V. (2022). Metod vyiavlennia ta identyfikatsii BPLA z zastosuvanniam neironnoi merezhi: kvalifikatsiina robota mahistra za spetsialnistiu “172 – telekomunikatsii ta radiotekhnika”. Ternopil: TNTU, pp. 89.
5. Fomin I. I. (2021). Zakhyst kanalu upravlinnia bezpilotnykh litalnykh aparativ vid nesanktsionovanoho dostupu : kvalifikatsiina robota mahistra za spetsialnistiu “125 – kiberbezpeka”. Ternopil: TNTU, pp. 66.
6. Tupytsia I. M., Kryvonos V. M., Kibitkin S. O., Ivashchuk L. A., Bielivtsov A. O. (2023) Kontseptualna model avtomatyzatsii protsesu deshyfruvannia danykh povitrianoi rozvidky z vykorystanniam tekhnolohii systemy shtuchnoho intelektu. Systems of Arms and Military Equipment, no. 1 (73), pp. 75–81. https://doi.org/10.30748/soivt.2023.73.09
7. Oleksenko O. O., Avramenko O. V., Fedorov A. V., Snitsarenko V. V., Chernavina O. Ie. (2023) Zastosuvannia bezpilotnykh litalnykh aparativ zbroinymy sylamy Rosiiskoi Federatsii u viini proty Ukrainy. Science and Technology of the Air Force of Ukraine, no. 4 (49), pp. 37–42. https://doi.org/10.30748/nitps.2022.49.05
8. Kartashov V. M., Oleinykov V. N., Sheiko S. A., Babkyn S. Y., Koryttsev Y. V., Zubkov O. V. (2018) Osobennosty obnaruzhenyia y raspoznavanyia malыkh bespylotnыkh letatelnыkh apparatov. Radyotekhnyka, no. 195, pp. 235–243. https://doi.org/10.30837/rt.2018.4.195.24
9. Kartashov V., Oleynikov V., Koryttsev I. Processing and Recognition of Small Unmanned Vehicles’ Sound Signals. Department of Media Engineering and Information Radio Electronic Systems Kharkiv National University of Radio Electronics. Available at: http://openarchive.nure.ua/handle/document/ (accessed: 01.10.2024).
10. Nekhin M., Kanevskyi L., Myronchuk Yu. (2023). Formuvannia sukupnosti parametriv boiovykh mozhlyvostei udarnykh bezpilotnykh litalnykh aparativ na osnovi fasetnoi systemy klasyfikatsii. Zhytomyrskyi viiskovyi instytut imeni S. P. Korolova, Ukraina, pp. 87–99. https://doi.org/10.33577/2312-4458.28.2023.87-99
11. Wong S. Y., Choe C. W. C., Goh H. H., Low Y. W., Cheah D. Y. S., Pang C. (2021) Power Transmission Line Fault Detection and Diagnosis Based on Artificial Intelligence Approach and Its Development in UAV: A Review. Arabian Journal for Science and Engineering, no. 46 (10), pp. 9305–9331. https://doi.org/10.1007/s13369-021-05522-w
12. You H. E. (2020) Mission-Driven Autonomous Perception and Fusion Based on UAV Swarm. Chinese Journal of Aeronautics, no. 33 (11), pp. 2831–2834. https://doi.org/10.1016/j.cja.2020.02.027
13. Lin C., Han G., Qi X., du J., Xu T., Martinez-Garcia M. (2021) Energy-Optimal Data Collection for Unmanned Aerial Vehicle-Aided Industrial Wireless Sensor Network-Based Agricultural Monitoring System: A Clustering Compressed Sampling Approach. IEEE Transactions on Industrial Informatics, no. 17 (6), pp. 4411–4420. https://doi.org/10.1109/TII.2020.3027840
14. Elghitani F. (2024) Dynamic UAV Routing for Multi-Access Edge Computing. IEEE Transactions on Vehicular Technology, pp. 1–11. https://doi.org/10.1109/TVT.2024.3360253
15. Simo A., Dzitac S., Dzitac I., Frigura-Iliasa M., Frigura-Iliasa F. M. (2021) Air Quality Assessment System Based on Self-Driven Drone and LoRaWAN Network. Computer Communications, no. 175, pp. 13–24. https://doi.org/10.1016/j.comcom.2021.04.032
16. Deng Y., Zhang H., Chen X., Fang Y. (2024) UAV-Assisted Multi-Access Edge Computing With Altitude-Dependent Computing Power. IEEE Transactions on Wireless Communications, no. 23, pp. 9404–9418. https://doi.org/10.1109/TWC.2024.3362375
17. Romaniuk L., Chykhira I. (2020) Aerodynamic model of a group of uavs in aircraft space. Computer-integrated technologies: education, science, production. Lutsk, no. 38, pp. 59–66. https://doi.org/10.36910/6775-2524-0560-2020-38-10
18. Romaniuk L., Chykhira I., Tulaidan H., Mykytyshyn A. Model of motion route of unmanned aerial vehicles operations with obstacles avoidance. ICAAEIT 2021, 15–17 December 2021. Tern.: TNTU, Zhytomyr “Publishing house “Book-Druk“” LLC, 2021. P. 193–199. (Mathematical modeling in power engineering and information technologies).
19. Romaniuk L., Chykhira I., Kartashov V., Dombrovskyi I. (2023) UAV movement planning in mountainous terrain. Scientific Journal of TNTU, vol. 110, no. 2, pp. 15–22. https://doi.org/10.33108/visnyk_tntu2023.02.015
20. Romaniuk L., Bernas M., Kartashov V., Chykhira I., Tulaidan H. Aircraft automation principles as a basis for the use of information technologies CEUR Workshop Proceedings, 2024, 3742, pp. 270–282

УДК 621.396.96

ІНТЕГРАЦІЯ РОЙОВОГО ІНТЕЛЕКТУ ТА КОРДОННИХ ОБЧИСЛЕНЬ ДЛЯ АВТОНОМНОЇ РОБОТИ КІЛЬКОХ ДРОНІВ

Леонід Романюк[1]; Ігор Чихіра[1]; Галина Тулайдан[2]; Андрій Головко[1]

Тернопільський національний технічний університет імені Івана Пулюя, Тернопіль, Україна
Тернопільський національний педагогічний університет імені Володимира Гнатюка, Тернопіль, Україна

Резюме. *Представлено адаптивний алгоритм PSO (Particle Swarm Optimization – оптимізація роїв частинок) як основу для підходу до ройового інтелекту в операціях з кількома БПЛА. Традиційну формулу PSO для оновлення швидкості та положення частинок було змінено, щоб включити стратегію варіації від Differential Evolution (DE), що дозволяє БПЛА динамічно коригувати свої траєкторії. Інтеграція глибокого навчання з підкріпленням (DRL) додатково підвищує здатність моделі оптимізувати розвантаження завдань і розподіл обчислень, гарантуючи, що БПЛА функціонують як ефективні периферійні вузли. Проведено експериментальне оцінювання запропонованого методу PSO-Edge порівняно з іншими методами машинного навчання, зокрема випадковим лісом (RF) і методом опорних векторів (SVM). Експериментальна установка включала симуляційне середовище, де БПЛА було поставлено завдання контролювати дані та виконувати місії над визначеною територією. Апаратне забезпечення включало процесор IntelXeonGold 6248R, 128 ГБ оперативної пам'яті та графічний процесор NVIDIATeslaV100 із моделюванням, виконаним за допомогою Python 3.8. Запропонований алгоритм PSO-Edge продемонстрував високу продуктивність за кількома показниками: скорочення часу виконання завдання на 42,1 хвилини порівняно з RandomForest і SVM; досягнення найнижчого енергоспоживання на завдання – 28,9 Wh; демонстрація ефективного зв'язку з найменшою затримкою в 0,15 секунди; досягнення найвищої точності виконання завдання в 96%. Результати підтверджують, що метод PSO-Edge перевершує традиційні підходи машинного навчання за ефективністю виконання завдань, енергоспоживанням, затриманням зв'язку та точністю. Це підкреслює переваги інтеграції периферійних обчислень з алгоритмом PSO, створюючи його як надійне вирішення для операцій з кількома БПЛА. Отримані результати мають значні наслідки для оптимізації додатків на основі БПЛА, особливо в середовищах, що вимагають динамічної адаптації та ефективного управління ресурсами.*

Ключові слова: *автономна оптимізація траєкторії, глибоке навчання з підкріпленням, багатоагентні кордонні обчислення, показники уникнення зіткнень, опрацювання даних у реальному часі, адаптивна модуляція енергії.*

https://doi.org/10.33108/visnyk_tntu2025.01.067 *Отримано 20.01.2025*