UDC 629.735.2:681.324(045)
DOI:10.18372/1990-5548.84.20197

[1]V. M. Sineglazov,
[2]D. V. Taranov

PREDICTION OF MOVING TARGETS AND ADAPTIVE AVOIDANCE IN HYBRID PSO-MPC FOR A SWARM OF UAV`S

[1,2]Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation Electronics and Telecommunications, State University "Kyiv Aviation Institute", Kyiv, Ukraine
E-mails: [1]svm@nau.edu.ua ORCID 0000-0002-3297-9060, [2]4637199@stud.kai.edu.ua

Abstract**—***The paper proposes a hybrid approach to the safe control of a multicopter swarm in the presence of two moving obstacles based on a combination of the particle swarm algorithm and model predictive control. The first stage of the algorithm is a global search for new target positions of subgroup centers using particle swarm algorithm based on predicted data, which allows the front subgroup to smoothly climb and avoid the danger zone. The second stage is the local adjustment of the movement of each vehicle within the model predictive control, taking into account dynamic constraints, which ensures accurate adherence to the calculated targets and prevents formation disruption. Simulation experiments demonstrate that the developed algorithm ensures coordinated maneuvers of all subgroups, timely avoidance of both moving threats, and return to the original formation without sudden jumps in altitude or chaotic behavior.*

Keywords—UAV swarms; moving obstacle avoidance; particle swarm algorithm; model predictive control; trajectory prediction; formation formation; repulsive forces; adaptive control; cohesive flight; formation and avoidance; flight safety; swarming algorithms; intelligent systems; remote forecasting; cooperative control.

I. INTRODUCTION

Unmanned aerial vehicles (UAVs) are becoming ever more prominent in both military and civilian contexts, such as in the tasks of territorial surveillance, cargo transport, and synchronized search and rescue missions. Swarms are particularly beneficial in a situation where a swarm of small drones collaborate on a task by redistributing tasks and synchronizing their movement to accomplish a shared objective. In such systems, one of the main benefits is scalability; as drones increase in number, reliability along with coverage improves although each drone on its own lacks the power necessary to accomplish complex tasks on its own. Meanwhile, managing the swarm is also challenging since following a specific formation has to be maintained, keeping the drones safe from each other, and getting around sudden obstacles in their path swiftly.

Swarm motion can be categorized into a number of organizational structures: a centralized approach, where a single "leader" or ground controller manages all the devices, and a decentralized approach, where each drone independently makes conclusions based on local information from surrounding units. These methods tend to merge the principles of potential fields, established formation geometries such as linear, rhomboidal, and city-grade shapes, and optimization algorithms, including various iterations of particle swarm optimization (PSO). These methods enable flexible adaptation of positions in the collective. Through model predictive control (MPC), each drone can carry out precise local control maneuvers considering acceleration, velocity, and size constraints, while ensuring coherence in the swarm.

Under normal circumstances, most of the obstacles are stationary (e.g., buildings and trees), but it is not rare for an unexpected obstacle to pop up suddenly; examples include a flying helicopter, ground vehicle, other unmanned vehicle, or even a bird flying across the intended path. If the formation cannot forecast the trajectory of such an object and change its own trajectory in time, there is a high likelihood of collision. Thus, contemporary algorithms deployed for swarm intelligence must incorporate elements of forecasting the motion trajectories of dynamic obstacles, examining potential points of collision, and reconfiguring the formation collectively prior to the threat approaching its critical distance.

The main challenge here is to combine three aspects: prediction of the movement of the obstacle on the horizon several steps ahead, quick global recalculating of new centers of subgroups based on the foreseen position of the threat (through PSO), and accurate tracking of these new targets locally, taking into account the dynamics constraints of each drone (through MPC). The goal in this paper is to illustrate that a hybrid PSO-MPC, coupled with a simple Kalman filter for prediction, provides smooth and safe evasions of multiple moving obstacles simultaneously without disrupting the formation or causing excessive computational overhead. The primary problem to be researched is whether prediction methods integration, global optimization technique, and local control scheme can be employed to good effect in guiding the swarm along distinct routes of mobile agents, with formation maintained as required and minimal deviations and safe inter-object intervals.

II. PROBLEM STATEMENT

To understand why the problem of automated swarm management needs to be improved and why it is becoming increasingly important, let's look at several key aspects. First, each drone should strive to maintain distance from its neighbors and maintain a collective formation without violating the conditions of sensor zone overlap or collisions. If the forward subgroup in the diamond shape sees an obstacle and the middle drones fail to change their course in time, the entire formation will fall apart, negating the benefits of the swarm architecture. Secondly, moving objects – other drones, helicopters, birds, and sometimes ground vehicles - are not guaranteed to follow fixed trajectories, and therefore no pre-written script will provide a timely evasion maneuver. That is, the system must not only react, but also predict the future movement of the obstacle and adjust formation plans on the fly. An automatic scheme is needed that combines threat trajectory prediction, a global decision to rebuild the formation, and local control laws for each drone.

The problem can be summarized in the following flowchart, which illustrates the main stages of the solution (Fig. 1).

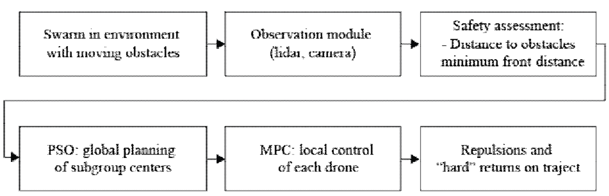

Fig. 1. Flowchart of the main stages of the solution

Before describing the solution in detail, let us formalize the mathematical model of the swarm and the algorithm for agent interaction. Let $i = 1, \ldots, N$ denote the index of a drone, each of which has a state

$$x_i(k) = \begin{pmatrix} p_i(k) \\ v_i(k) \end{pmatrix}, p_i(k) \in \mathbb{R}^3, v_i(k) \in \mathbb{R}^3, \quad (1)$$

which is updated at each step k according to a linear law

$$x_i(k+1) = Ax_i(k) + Bu_i(k), \quad (2)$$

where A is the matrix of motion for the inertial component; B contains the effect of control accelerations; $u_i(k) \in \mathbb{R}^3$ is the vector. Agent interaction is realized in two components. Firstly, when two drones are within closer than threshold $d_{\min}$, there is a repulsive force generated along the gradient of the Euclidean norm to avoid collision. Second, maintaining the chosen formation (diamond) is achieved by each drone trying to remain in a specific relative position relative to the center of its subgroup – nominally, this is achieved by fixed formation center targets, which are adjusted globally in response to a threat. The fact that there is no single "leader" implies that decisions on formation changes are made from a global consideration of the status of all agents and anticipated obstacles. In order to be capable of detecting a target (i.e., moving obstacle), every drone employs local sensors: it calculates the distance $\| p_i(k) - p_{obs}(k) \|$ and direction to the object. However, because the same measurements can only determine the instantaneous location, a prediction module (simple Kalman filter) has to be included. It progressively predicts the obstacle's speed and direction, building a prediction several steps ahead $\hat{p}_{obs}(k+l)$. This is utilized by the whole PSO algorithm, which, when triggers are activated (when the minimum front distance d_{front} or any $\| p_i - p_{obs} \|$ is less than the threshold d_{trig}) computes new displacements $\{\Delta c_g\}$ of the centers of the three subgroups so that the front subgroup moves smoothly above the danger zone throughout the maneuver.

Thus, the problem is formulated as follows: it is necessary to build a swarm control system that simultaneously takes into account the current state of all drones and their formation;

- forecasting the trajectory of a moving obstacle;
- global search for new subgroup centers that minimizes formation deviations and prevents collisions;
- local control of each drone, taking into account the dynamics constraints;
- repulsive mechanisms to ensure that none of the drones is too close to another drone or an obstacle;

- global algorithm to construct a new formation is a necessity to guarantee not just the integrity of the swarm a mechanism for returning to the initial height and formation after the round is completed.

Unless you heed the forecast and subjective evaluation of the situation, the swarm will act chaotically: the front subgroup will notice the danger too late, the middle agents will scatter in various directions, and the rear ones will simply not have time to change the trajectory. The result is collisions, formation disruption, and drone loss. Therefore, having an automatic target prediction unit and a, but also reduce algorithm restarts, response time, and total traversal time. This issue is the subject of the following discussion of our solution.

III. RELATED WORKS

In modern research on multichannel control of UAV swarms, tracking algorithms of moving targets based on global optimization along with local safety methods are widely used. One of the earliest population-based methods of searching for moving objects was the realization of Motion-Encoded PSO, which allowed using knowledge of the target's speed and motion direction as implicit input signals to optimize the search trajectory [1], [3]. Yet, this approach did not take into consideration the stringent limitations on the dynamics of every vehicle, resulting in the presence of delayed reactions to abrupt target motion changes.

To eliminate this disadvantage, model predictive control (MPC) methods have been actively applied, which provide formal adherence to the constraints on velocity, acceleration, and safe distance to obstacles. Specifically, the potential for applying nonlinear MPC to prevent collisions with moving obstacles was shown, where the input constraints involved current estimates of moving obstacle state, and the optimization problem was solved online [2, 8]. Such systems have demonstrated control stability with a modest number of computational resources, yet scalability to larger swarms has been hindered by the substantial complexity of solving the quadratic programming problem at every step.

This combination of PSO and MPC has attracted researcher attention as a way of combining the power of global search with strong security assurances. Step-by-step hybrid algorithms first generated a set of target points through PSO based on possible movements of obstacles, then the MPC part refined local trajectories for each vehicle such that the deviation from the global optimal line was not more than some threshold [9], [12]. But the lack of a forecasting system increased the likelihood of a collision every time there was a sudden change in the motion of a moving body.

A number of recent efforts have tried to incorporate modules of trajectory prediction of moving objects into hybrid architectures. For instance, the PANTHER system used perception-driven planning on anticipated obstacle positions computed from local sensor readings and predictor algorithms, enabling real-time trajectory change; however, it struggled to scale up to large swarms because of its centralized computational structure [4]. Other writers have been interested in distributed MPC: in such architectures, each agent only exchanges principal statistics regarding the likely movement of obstacles, which reduces the communications but at the same time keeps the prediction pertinent to local controllers [5].

One more direction of research is the exploration of adaptive control of PSO coefficients depending on the environment. Dynamic modification of inertial and cognitive coefficients during information gathering on obstacle density results in more profound differentiation of the global search, preventing premature convergence and providing fast adaptation to new information about target movement [3]. Also relevant are the works on building modular hybrid architectures, in which a separate clustering unit analyzes the history of the movement of dynamic objects, forming short-term forecasts that are subsequently fed to the input of the global PSO [6], [7], [11].

Thus, the current state of the art identifies the following key areas: (1) synthesis of algorithms for predicting the trajectory of dynamic obstacles based on clustering and statistical evaluation, (2) development of hybrid PSO-MPC structures with formal security guarantees and the ability to scale to large swarms, and (3) implementation of adaptive mechanisms for adjusting global search parameters to increase the system's resilience to unstable operating conditions. The research carried out in these areas is a sound basis for the creation of the proposed hybrid algorithm with built-in prediction of mobile targets.

IV. PROBLEM SOLUTION

In the present research, all drones are envisioned as linear copters, so their state can be described by a six-dimensional vector-column.

$$x_i(k) = \left(\frac{p_i(k)}{v_i(k)} \right) = \left(x_i(k), y_i(k), z_i(k), v_{x,i}(k), v_{y,i}(k), v_{z,i}(k) \right)^{\mathrm{T}}, \quad (3)$$

where $p_i(k) = (x_i, y_i, z_i)^{\mathrm{T}}$ is the geographical location of the device at step k, and $v_i(k) = (v_{x,i}, v_{y,i}, v_{z,i})^{\mathrm{T}}$ is its linear velocity. This model accurately describes horizontal motion in the *XY* plane and vertical displacement in terms of climb or descent along the *Z* axis. Assuming a fixed sampling step T_s, we obtain a linear discrete model of the motion of every drone, represented in matrix form

$$x_i(k+1) = \mathbf{A}x_i(k) + \mathbf{B}u_i(k), \tag{3}$$

where matrix

$$\mathbf{A} = \begin{pmatrix} I_3 & T_s I_3 \\ 0_3 & I_3 \end{pmatrix}, \tag{4}$$

designates the inertial-free motion part (inertial component velocity transfer to the positioning unit plus velocity constancy between steps), and

$$\mathbf{B} = \left(\frac{0_3}{T_s I_3} \right), \tag{5}$$

incorporates the effect of control accelerations $u_i(k) = (a_{x,i}(k), a_{y,i}(k), a_{z,i}(k))^{\mathrm{T}}$ on a velocity change. Discretization of the system ensures that we will be able to calculate accurately the new position and velocity of the agent at a constant rate, which is very useful for future MPC applications. In parallel, we introduced a control value constraint to take into consideration the drone hardware limitations: the horizontal component of the acceleration satisfies the condition $\left|u_{i,xy}(k)\right| \le \max\_acc$, while the vertical component is strictly limited $\left|u_{i,z}\right| \le u_{z,\max}$. The use of a linear model with appropriately defined constraints enables realistic modeling of the actual dynamics of the copter. Furthermore, such a structure sets the foundation for the generation of optimal trajectories and the enabling of smooth transitions between maneuvers in a dynamic obstacle environment.

We use a discretized system with step T_s according to which the evolution of the state of each device is provided by the matrix **A** and **B** that is

$$x_i(k+1) = \mathbf{A}x_i(k) + \mathbf{B}u_i(k), \tag{6}$$

where optimal control in the form of accelerations. There are certain limits imposed for the value of control so that the horizontal norm $| u_{i,xi} \| \le \max\_acc$, and the vertical component did not exceed $\left|u_{i,z}\right| \le u_{z,\max}$.

Our strategy starts with the deployment of *N* agents in a diamond formation, centers partitioned into three subgroups of four drones each. These subgroups travel along the *X*-axis at 15 m/s constant speed. The main and secondary barriers both present a converging threat simultaneously: both specify linear trajectories in the spatial space that are certain to intersect the path of the swarm. To conduct the simulation, the parameters of the obstacle trajectories were deliberately adjusted to ensure that the swarm trajectories intersect with the obstacle trajectories. In each cycle of the simulation, we calculate the horizontal distances of all agents to the current position of each obstacle.

If any one or more of these values is less than the selected threshold of 22m or if any one of the drones comes closer to the obstacle, the global recalculation process of the target centers of the subgroups using the PSO algorithm is triggered. It is required that PSO is re-run independently for the first and second hurdles, i.e. if the frontline soldiers are within 22 meters of the first, the prediction of the first is prioritized, and otherwise.

The fitness function, which we minimize using PSO, is usually the sum of the squares of the displacements of the new centers of each cluster from the original centers plus a penalty for approaching the expected obstacle points. In formal notation, let Δc_g be the vector of displacements of the centers g subgroups. All components of the penalty system are denoted as:

$$\left\|\Delta c_g\right\|^2$$
$$+\sum_{l=1}^{N_h} \sum_{i \in front} \max\left(d_{\min} - \left\| p_i + \frac{l}{N_h}\Delta c_g - \hat{p}_{obs}(l) \right\| 0 \right)^2, \tag{7}$$

where $d_{\min} \approx 2\,\mathrm{m}$ is the minimum required distance. Thus, whenever the variations denoted as Δc create a scenario in which the estimated trajectory of the frontline drone and the trajectory of the obstacle intersect at a distance less than $d_{\min}$, a quadratic penalty will be applied. The penalty drastically adds to the overall value of the function in a way that this arrangement becomes economically infeasible. Simultaneously, PSO tries to keep the clusters "close" to the original centers, which makes the deviations from the formation minimal. After collecting all the estimates, the individual and global best positions of PSO particles get updated, followed by their velocities and spatial positions. Ultimately, we get the optimal $\left\{\Delta c_g^*\right\}_{g=1}^{G}$ which determine the new horizontal coordinates of centers of the three subgroups in the current simulation step.

After the PSO has accepted new centers, each agent i for the MPC portion is assigned its own set of references

$$(p_{ref,i}(l), v_{ref,i}(l)) \tag{8}$$

on the horizon $l = 1,\ldots,N_h$. In particular, the point lth step $p_{ref,i}(l)$ is calculated as a linear interpolation between the old center $c_{g(i)}$ and new $c_{g(i)} + \Delta c_{g(i)}$ in proportion to l / N_h. If the first subgroup is in the "threat" zone (i.e. at least one

$$d_{front} \le front\_d\_trig \tag{9}$$

this height is set $z_0 + H_{clear}$, otherwise it is equal to z_0. The procedure for forming a speed sequence $v_{ref,i}(l)$ is trivial: since the goal is to move to new positions evenly over N_h for cycles with zero initial and final velocities, it is enough to zero all velocity components in the reference. In doing this, each agent is assigned target points received by it through the MPC controller, designed with respect to the global PSO distribution. Having these N_h references, the MPC part of each ith drone solves a quadratic program:

$$\min_{u_i(0),\ldots,u_i(N_h-1)} \sum_{l=1}^{N_h} \left\| x_i(k+l) - x_{ref,i}(l) \right\|_Q^2 + \sum_{l=0}^{N_h-1} \left\| u_i(l) \right\|_R^2 \tag{10}$$

by dynamics

$$x_i(k+l+1) = Ax_i(k+l) + Bu_i(l) \tag{11}$$

and conditions

$$\left\| u_{i,xy(l)} \right\| \le \max\_acc, \left| u_{i,z}(l) \right| \le u_{z,\max}. \tag{12}$$

The Q weight in the position variables is more than the acceleration weight in R, and this ensures that the MPC favors following the new trajectory. We solve and obtain the first optimal acceleration $u_i^*(k)$ and implement it into the simulation, i.e., the agent behaves according to the first element of the optimal profile, and the rest of the calculations are held for the next iteration. If the MPC is unable to compute a feasible solution – i.e., there exists no trajectory that can fulfill the combined demands of safety and dynamics – the controller will output a zero vector. This, in effect, lets the repulsive terms take the lead in evasion.

In order not to collide inside the swarm and with other objects, a local model of repulsive forces has been incorporated. The repulsion interaction among agents is calculated based on a gradient function defined by the current distance: if any two drones are nearer than the parameter defined then a similar repulsive force pushes them towards increasing the distance. Likewise, every agent employs the Euclidean norm to determine the repulsive contribution from both of the two obstacles with their current positions (p_{obs1}, p_{obs2}) and the safety radius $r = 1$ m, as well as an extra gap of a maximum of 2 m. In a case where the MPC prescribes getting too close, the combination of the MPC and repulsive vectors steers the drones away from the obstacle path without breaking the acceleration constraint. Finally, whenever the nominal distance between any agent and both obstacles is always greater than $return\_threshold \approx 15$ consecutive cycles, then the swarm has completed the avoidance maneuver successfully. The front subgroup, which was earlier on its way up, glides down to the height of z_0. If MPC does not require evasion maneuvers again, all end agents perform a "hard" return to their initial center in the rhombus: a vector

$$u_{rigid} = \beta(c_{g(i)} - p_i), \tag{13}$$

with a small coefficient β is added which smooths the return. Because of this, all drones will return to the initial formation after overcoming the obstacle, and the algorithm continues waiting for the next threat.

Therefore, the most critical steps of calculations in each iteration are: prediction update and obstacle position, trigger detection, start PSO if necessary and calculate new targets, send these targets to MPC, blend MPC output with hard returns and repulsions, and lastly update physical state of each drone. It is this recursive entwining of world PSO searching and local formal MPC, supplemented with obstacle prediction and repulsion, that enables smooth and safe avoidance of even two moving menaces simultaneously.

IV. RESULTS

The integrated PSO-MPC approach facilitated smooth and productive evasive maneuvers against two approaching obstacles with guaranteed satisfaction of both drones' dynamic constraints. The integration of global exploration for new formation center coordinates by PSO with local formal control through MPC had the following major advantages. First, the algorithm allowed the front subgroup to make the maneuver at the prognosis step, well before the point blank area of the obstacles: as it can be seen in Fig. 2.

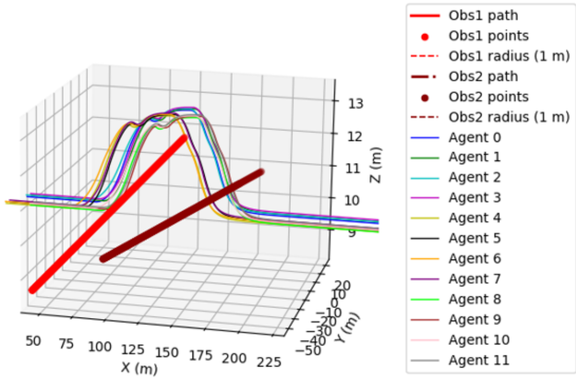

Fig. 2. Result of swarm behavior trajectory simulation

The first and second barriers were below the trigger level, and the PSO immediately recalculated which avoided violent "jumps" or uncontrollable scattering of drones. Secondly, by producing a smooth front subgroup rise and involving its rise in the MPC references, the drones did not roll out to too great a height, with the formation integrity.

The primary result of the simulation was guaranteed compliance with the safe distance to both obstacles. From the minimum distance graph (Fig. 3), minimum value $d_{\min}$ was 1.49 m, which gives a safety margin.

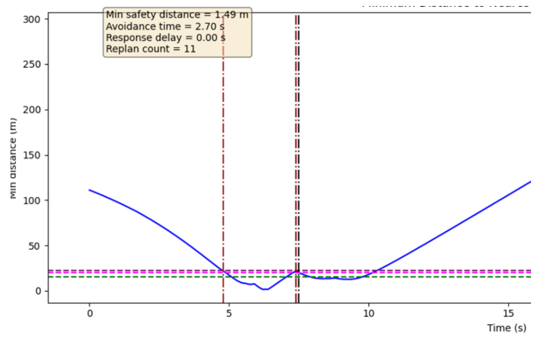

Fig. 3. Schedule of minimum distance, maneuver time, and number of PSO restarts

The average Avoidance time was 2.70 seconds, and the Response delay was close to zero because the front subgroup began to turn as soon as the threat was triggered. On average, the algorithm called the PSO 11 times, a sign of moderate computational complexity and the ability to react to two unrelated threats without too much restart. Some other technical advantages are the absence of the need to specify complicated evasion heuristic rules: PSO inherently identifies the optimal subgroup center shifts in respect to anticipated paths, and MPC maximizes the distances from these centers according to dynamic constraints. For this reason, even when the second obstacle came at the same time, the swarm was able to correct the path without colliding with the first object easily. Separate subgraphs for each subgroup (Figs 4 – 6) confirm that all four drones in each cluster rose and fell in synchrony without further disturbing the diamond formation.

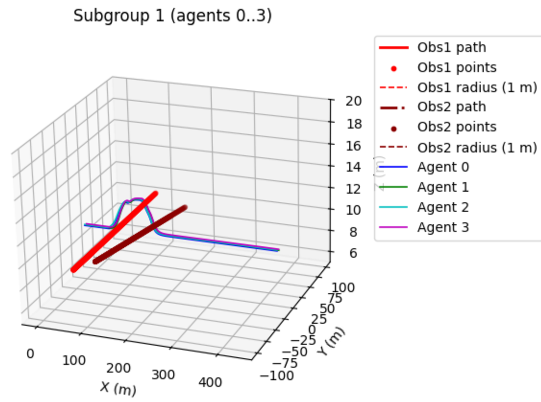

Fig. 4. The result of modeling subgroup 1 of the swarm behavior trajectory

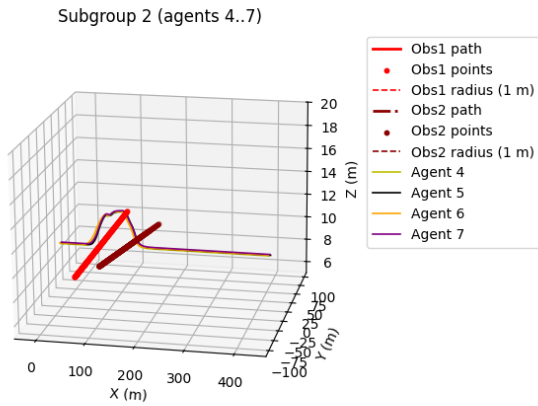

Fig. 5. The result of modeling subgroup 2 of the swarm behavior trajectory

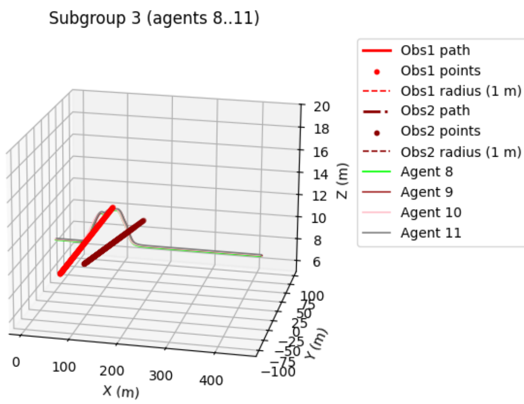

Fig. 6. The result of modeling subgroup 3 of the swarm behavior trajectory

V. CONCLUSION

The proposed algorithm demonstrates resilience to changes and to shift the collision point in space, the swarm realigns its trajectory automatically without compromising safety. Certain metrics such as minimum distance, travel time, reaction delay, and PSO restarts may be used to compare objectively with other swarm management methods against mobile threats. The union of PSO and MPC provides:

1) proactive response due to obstacle movement forecasting;

2) smooth high-altitude maneuver;

3) guarantee of a safe distance with simultaneous obstacles;

4) minimum number of PSO restarts (11 times) for the entire simulation cycle.

This makes the described algorithm suitable for real-world conditions where the number of moving objects can vary and computing resources are limited. The nearest prospect for development is to test the approach in field tests, taking into account real sensor noise and unforeseen trajectories, which will allow us to assess the resistance to measurement errors.

REFERENCES

[1] M. D. Phung, & Q. P. Ha, *Motion-Encoded Particle Swarm Optimization for Moving Target Search Using UAVs*, 2020. arXiv preprint arXiv:2010.02039. https://doi.org/10.1016/j.asoc.2020.106705

[2] B. Lindqvist, S. S. Mansouri, A. Agha-mohammadi, & G. Nikolakopoulos, *Nonlinear MPC for Collision Avoidance and Control of UAVs with Dynamic Obstacles*, 2020. arXiv preprint arXiv:2008.00792. https://doi.org/10.1109/LRA.2020.3010730

[3] Y. Li, W. Chen, B. Fu, Z. Wu, L. Hao, & G. Yang, "Research on Dynamic Target Search for Multi-UAV Based on Cooperative Coevolution Motion-Encoded Particle Swarm Optimization. Applied Sciences," 14(4), 1326, 2024. https://doi.org/10.3390/app14041326

[4] J. Tordesillas, & J. P. How, *PANTHER: Perception-Aware Trajectory Planner in Dynamic Environments*. 2021. arXiv preprint arXiv:2103.06372. https://doi.org/10.1109/ACCESS.2022.3154037

[5] B. Lindqvist, P. Sopasakis, & G. Nikolakopoulos, *A Scalable Distributed Collision Avoidance Scheme for Multi-agent UAV Systems*. 2021. arXiv preprint arXiv:2104.03783. https://doi.org/10.1109/IROS51168.2021.9636293

[6] Y. Zhou, B. Rao, & W. Wang, "UAV Swarm Intelligence: Recent Advances and Future Trends," *IEEE Access*, 8, 2020, 176229–176256. https://doi.org/10.1109/ACCESS.2020.3028865

[7] X. Li, & Y. Wang, "A review of current studies on the unmanned aerial vehicle-based target tracking," *Sensors*, 25(2), 439, 2025. https://doi.org/10.3390/s18020439

[8] J. Wu, & S. Mei, "A Novel MPC Formulation for Dynamic Target Tracking With Evacuation Model, *International Journal of Control, Automation and Systems*, 22, 1234–1247, 2024.

[9] L. Wang,, & Q. Zhang, "UAV Swarm Formation Reconfiguration Control Based on Variable Topology," *Frontiers of Information Technology & Electronic Engineering*, 23(6), 1908–1923, 2022.

[10] K. Panjavarnam, Z. H. Ismail, C. H. H. Tang, & K. Sekiguchi, "Model Predictive Control for Autonomous UAV Landings: A Comprehensive Review of Strategies, Applications and Challenges," *Journal of Engineering*, vol. 2025, Issue 1, 2025. https://doi.org/10.1049/tje2.70085

[11] D. Chen, X. Liu, & H. Li, "Multi-UAV Collaborative Trajectory Planning for Non-Cooperative Target Tracking," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 238(10), 980–996, 2024. https://doi.org/10.1177/09544100251335357

[12] A. Smith, B. Johnson, & C. Lee, "Decentralized Mesh-Based Model Predictive Control for Swarms of UAVs," *Sensors*, 20(14), 3892, 2022.

[13] H. Zhu et al., "Distributed Multi-Robot Formation Splitting and Merging in Dynamic Environments." *IEEE ICRA*, 2019. https://doi.org/10.1109/ICRA.2019.8793765

[14] A. Chowdhury et al., "Multi-Robot Virtual Structure Switching and Formation Changing Strategy." *IEEE/RSJ IROS*, 2018.

Received March 18, 2025

Sineglazov Victor. ORCID 0000-0002-3297-9060. Doctor of Engineering Science. Professor.
Head of the Department Aviation Computer-Integrated Complexes, Faculty of Air Navigation Electronics and Telecommunications, State University "Kyiv Aviation Institute", Kyiv, Ukraine.
Education: Kyiv Polytechnic Institute, Kyiv, Ukraine, (1973).
Research area: Air Navigation, Air Traffic Control, Identification of Complex Systems, Wind/Solar power plant, artificial intelligence.
Publications: more than 750 papers.
E-mail: svm@kai.edu.ua

Taranov Denys. Postgraduate Student.
Aviation Computer-Integrated Complexes Department, Faculty of Air Navigation, Electronics and Telecommunications, State University "Kyiv Aviation Institute", Kyiv, Ukraine.
Education: National Aviation University, Kyiv, Ukraine, (2021).
Research interests: artificial neural networks, artificial intelligence, programming.
Publications: 5.
E-mail: 4637199@stud.kai.edu.ua

В. М. Синєглазов, Д. В. Таранов. Гібридна методологія перебудови рою дронів на основі локальних потенціалів та глобальної координації

У статті запропоновано гібридний підхід до безпечного керування роєм мультикоптерів за наявності двох рухомих перешкод, заснований на комбінації алгоритму рою частинок та моделі прогнозного керування. Перший етап алгоритму – це глобальний пошук нових цільових позицій центрів підгруп за допомогою алгоритму рою часток на основі прогнозованих даних, що дозволяє передній підгрупі плавно підніматися та уникати небезпечної зони. Другий етап – це локальне коригування руху кожного транспортного засобу в межах моделі прогнозного керування з урахуванням динамічних обмежень, що забезпечує точне дотримання розрахованих цілей та запобігає порушенню формації. Моделювальні експерименти демонструють, що розроблений алгоритм забезпечує скоординовані маневри всіх підгруп, своєчасне уникнення обох рухомих загроз та повернення до початкової формації без різких стрибків висоти або хаотичної поведінки.

Ключові слова: рої БПЛА; уникнення рухомих перешкод; алгоритм рою часток; модель прогнозного керування; прогнозування траєкторії; формування формації; сили відштовхування; адаптивне керування; згуртований політ; формування та уникнення; безпека польотів; алгоритми рою; інтелектуальні системи; дистанційне прогнозування; кооперативне керування.

Синєглазов Віктор Михайлович. ORCID 0000-0002-3297-9060. Доктор технічних наук. Професор.
Завідувач кафедрою авіаційних комп'ютерно-інтегрованих комплексів. Факультет аеронавігації, електроніки і телекомунікацій, Державний університет «Київський авіаційний інститут», Київ, Україна.
Освіта: Київський політехнічний інститут, Київ, Україна, (1973).
Напрям наукової діяльності: аеронавігація, управління повітряним рухом, ідентифікація складних систем, вітроенергетичні установки, штучний інтелект.
Кількість публікацій: більше 750 наукових робіт.
E-mail: svm@kai.edu.ua

Таранов Денис Володимирович. Аспірант.
Кафедра авіаційних комп'ютерно-інтегрованих комплексів, Факультет аеронавігації, електроніки та телекомунікацій, Державний університет «Київський авіаційний інститут», Київ, Україна.
Освіта: Національний авіаційний університет, Київ, Україна, (2021).
Напрям наукової діяльності: штучні нейронні мережі, штучний інтелект, програмування.
Публікації: 5.
E-mail: 4637199@stud.kai.edu.ua