Stag hunt game-based approach for cooperative UAVs

Lanh V. Nguyen[1], Ignacio T. Herrera[1], Duong M. Phung[1,2], Ricardo P. Aguilera[1] and Quang P. Ha[1]

[1]School of Electrical and Data Engineering, University of Technology Sydney (UTS), Australia
[2]VNU University of Engineering and Technology (VNU-UET), Vietnam National University, Hanoi (VNU), Vietnam

{vanlanh.nguyen; ignacio.j.torresherrera; manhduong.phung; ricardo.aguilera; quangha}@uts.edu.au

Abstract -

Unmanned aerial vehicles (UAVs) are being employed in construction at an unprecedented rate, making construction one of the ever-growing industrial sectors. Over the last decade, UAV technology has been widely employed in numerous construction project phases, ranging from site mapping, progress monitoring, building inspection, damage assessments, and material delivery. While extensive studies have been conducted on the advantages of UAVs for various construction-related processes, studies on UAV collaboration and how to improve their overall efficiency are still scarce. This paper proposes a new cooperative path planning algorithm for multiple UAVs based on the stag hunt game and particle swarm optimization (PSO). First, a cost function for each UAV is defined, incorporating multiple objectives and constraints. The UAV game framework is then developed to formulate the multi-UAV path planning into the problem of finding payoff-dominant equilibrium. Next, a PSO-based algorithm is proposed to obtain the payoff-dominant equilibrium, representing optimal paths for the UAVs. Simulation results indicate the effectiveness of the proposed algorithm in generating feasible and efficient flight paths for UAV formation control.

Keywords -

Unmanned aerial vehicle; Cooperative path planning; Stag hunt game; Payoff-dominant equilibrium; Particle swarm optimization.

1 Introduction

The immense development of unmanned aerial vehicles (UAVs) technologies has been drawing significant attention in civilian sectors. In the construction domain, researchers and enterprises tend to seek safer and more efficient solutions for carrying out construction-related tasks. As modern technologies have reduced the cost of UAVs while increasing their dependability, operating time, and maneuverability, smart drone-powered solutions have been introduced as a platform to assist construction activities. They are well-established in numerous construction project stages such as construction site monitoring and 3D mapping [1], building and damage inspection [2], and package delivery logistics [3], demonstrating prospects for wide usage of drones.

Due to the increased quantity and complexity of construction jobs, such as large-site 3D mapping or multiple-package delivery, a single drone with restricted size and capability can sometimes not fulfill the requirements. Consequently, multi-UAVs are encouraged to collaborate as a team for the applications mentioned above in order to optimize processing time and operating possibilities [4].

A hierarchical structure formation system includes three layers: task management, path planning, and task execution, as shown in Figure 1. The task management layer holds and keeps track of the mission's objectives. Based on these objectives, this layer allocates resources and tasks to UAVs and acts as a decision-maker. From mission requirements, the path planning layer generates feasible trajectories for the formation. This layer aims to plan optimal paths for a group of UAVs moving in a known environment from the start to their target locations. This layer comprises three blocks: real-time trajectory modification, data acquisition, and cooperative path planning, wherein the collaborative path planning block is the primary function of the system and determines the overall optimized path for each quadcopter. Nonetheless, due to uncertainties that may be included along the route in practical applications, the real-time trajectory modification block is combined with the system. The formation can deal with emergencies such as a suddenly appearing obstacle. Generated paths will then be passed down to the task execution layer. This layer directly connects with the propulsion system of the quadcopter and generates the control laws. To enhance system performance, real-time data, i.e., UAVs' position and velocity, is fed back to the path planning layer to adjust the path, providing a closed control loop.

This paper focuses on cooperative path planning, where the path-planner produces trajectories to fulfill the mission objectives. The mission objectives include formation shape maintenance, minimum path lengths, and threat avoidance.

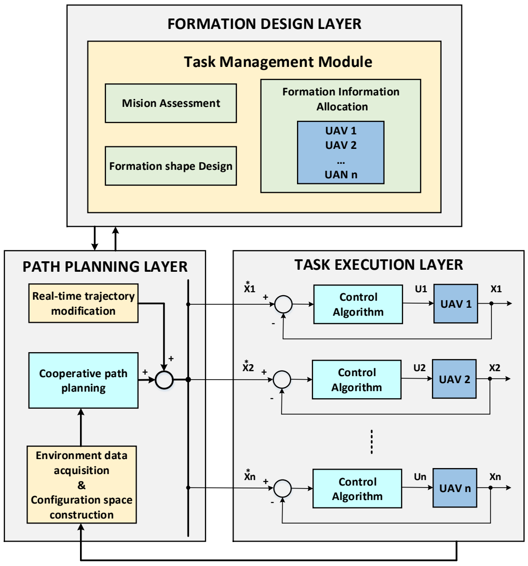

Figure 1. Hierarchical UAV control structure.

2 Related works

Path planning for multiple UAVs has been becoming an active research topic recently. The objectives and approaches differ depending on the application domain. There are currently numerous methods accessible in the academic literature, and each technique has its own merits.

The artificial potential field is one of the most widely used techniques for UAV cooperative path planning [5]. It considers the operating space as a potential field, with attractive fields surrounding the target and repulsive fields around obstacles. For cooperation, new fields are added, including the internal attractive potential fields to retain the formation configuration and the internal repulsive fields to prevent the UAVs from colliding with each other. The paths are then generated when the total force acts on the UAVs at each position. This technique can produce smooth and continuous paths. It, however, faces local minima problems when the total force is equal to zero.

In another direction, optimal control methods have been used for cooperative path planning by considering it as a numerical optimization problem subject to multiple constraints [6]. It first finds paths for single vehicles and then attains their cooperation by constraints. To generate paths, optimal control signals for individual quadcopters are computed using the mixed-integer linear programming (MILP) and then applied to the system. Although the MILP can solve the optimization problem with different constraints, it involves high computational complexity.

Recently, evolutionary algorithms (EA) have been used to solve multi-UAV cooperative path planning with the capability to find optimal solutions in complex scenarios [7]. They often include two layers, one for individual UAV path planning and the other for path cooperation. Initially, each vehicle is associated with an EA process to generate a feasible path to fulfill its task. Those paths are then adjusted via a global cost function to achieve the required cooperation. This approach can generate smooth cooperative paths as it does not discretize the workspace. It however may converge to sub-optimal solutions if cooperative constraints and maneuver properties of UAVs are not properly addressed.

On the other hand, as a theoretical framework for strategic interactions among competing players, game theory has gained its applications in a variety of fields such as construction bidding [8] and environmental management in the mining industry [9]. In the game, a player tries to maximize his profit, which depends on not only his actions but also others. Therefore, the best strategy relies on what the player expects others to do. Most games in the literature can be classified into cooperative and non-cooperative games[10]. In cooperative games (CGs), several players share a common objective to better achieving it than those working alone. However, a major issue with CGs is a trade-off between the stability of the player groups and the system efficiency [11]. In contrast, each player in noncooperative games (NCGs) has their own properties such as the payoff function, procedural details of the game, intention, and possible strategies which make him more inclusive than in CGs. Among NCGs, Nash Game is widely applied to the situation that all players have to simultaneously make their decisions in symmetric competitions, such as in exploring the public-private partnership investment incentives [12] or seeking strategies for clusters in a distributed system [13]. In these games, each player only has partial information about the choices of others.

The vast majority of Nash games revolve around the prisoner's dilemma (PD), therein the appearance and stability of cooperation are inhibited by its most restrictive conditions: defecting is a dominating option as it always offers a greater payout regardless of how the other player performs. As a result, collaboration is risky, and there is a tendency to defect [14]. Meanwhile, numerous more types of dilemmas are available, the most famous of which is the stag hunt game [15], therein players desire to coordinate, i.e., the preferred choice is to always do as the rival acts. In this game, not only mutual defection but also mutual cooperation are evolutionary stable strategies, but only the latter one results in a pay-off dominant equilibrium [16].

From the UAVs' cooperation perspective, where profit from cooperation is much more than profit from individual effort, the stag hunt game captures the conflict between individual UAVs, which naturally arises in cooperative path planning, better representing the situation than the PD.

This is especially the case when examining the influence of group selection, in which social interactions aim to maximize the group's performance. Indeed, the stag hunt game idea corresponds well with the cooperative path planning issue and may be utilized to solve it theoretically.

This paper proposes a stag hunt game based algorithm for UAV cooperative path planning. A cost function is first defined including all requirements on formation, path feasibility, safety and optimality. Unlike existing EA approaches, our method considers cooperative constraints in every individual to maximize the overall profit. The path planning problem then can be model as a game where UAVs are the players. Based on the stag hunt game, the strategy for each UAV is then formulated and enhanced PSO is introduced to obtain the payoff-dominant equilibrium. As a result, optimal paths can be achieved with the formation being maintained.

This paper is structured as follows. Section 3 formulates the cooperative path planning problem. Section 4 presents the proposed stag hunt based game and enhanced PSO. Numerical simulation results are provided in Section 5. Finally, conclusions are given in Section 6.

3 Problem formulation

3.1 Multi-vehicle path planning

Consider a team of N drones operating in a given flying area, including numerous obstacles, as shown in Figure 2. The position of the UAV group is determined in an earth frame, xyz, by $P = [P_1{}^T, P_2{}^T, ..., P_N{}^T]^T$, where N is the total number of drone members and $P_n = (x_n, y_n, z_n)^T$ is the location of the n-th vehicle.

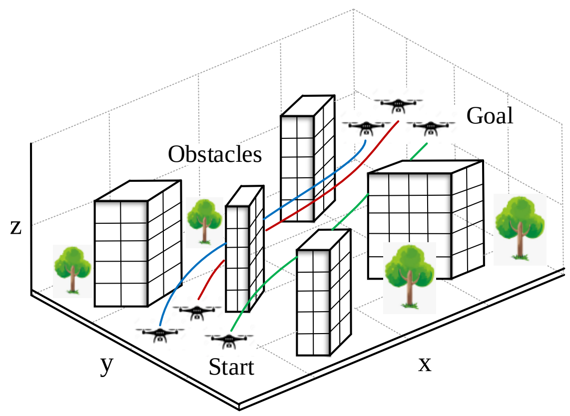

Figure 2. Definition of the path planning problem.

The problem of path planning is to establish a feasible route connecting the start and target positions in a collision-free environment while fulfilling a number of constraints. The problem can be expressed as an optimization process that is subjected to several costs. For a single UAV planning problem, it can be formulated as

$$P(0) \xrightarrow[s.t.\ J_s(X(k))]{X(k)} P(end), \tag{1}$$

where $P(0)$ and $P(end)$ are corresponding to the start and the target poses, k stands for the waypoint instant, and $X(k)$ denotes the path of UAV including K waypoint subjected to the single-UAV cost $J_s(X(k))$.

In single-vehicle path planning, to attain the most effective and efficient path, the cost $J_s(X(k))$ should be optimized, fulfilling constraints on path length, threat avoidance, and turning angle limit. It is defined as

$$\begin{aligned} J_s(X(k)) = {} & \omega_1 \sum_{k=1}^{K-1} L(k) + \omega_2 \sum_{k=1}^{K-1} \sum_{\tau=1}^{\mathcal{T}} D_\tau(k) \\ & + \omega_3 \sum_{k=1}^{K} H(k) \quad + \omega_4 \sum_{k=1}^{K-2} \theta(k) \\ & + \omega_5 \sum_{k=1}^{K-1} |\varphi(k) - \varphi(k+1)|, \end{aligned} \tag{2}$$

where $L(k)$ is the path length, $D(k)$ is the safety cost concerning $\mathcal{T}$ threats, $H(k)$ stands for the altitude payoff, $\theta_n(k)$ and $\varphi_n(k)$ correspond to the turning angle and climbing angle, ω_i, for $i = \{1, 2, ..., 5\}$, are the weight coefficients. More details about the single cost function are presented in our previous work [17].

Extending the problem into the multi-vehicle formation path planning, it is written as

$$P_n(0) \xrightarrow[s.t.\ J(X_n(k), X_n^-(k))]{X_n(k)} P_n(end),\ n = 1, 2, ..., N, \tag{3}$$

where $X_n(k)$ and $X_n^-(k)$ are corresponding to the path of UAV$_n$ and a set of its neighbours' paths. The multi-vehicle cost function, $J(X_n(k), X_n^-(k))$, consists of a single cost and a formation cost, computed as

$$J(X_n(k), X_n^-) = J_s(X_n) + \beta J_f(X_n, X_n^-), \tag{4}$$

where $J_s(X_n)$ is computed as (2), $J_f(X_n, X_n^-)$ is the formation cost, and β is a weighting factor. The cost function for the formation constraint is determined as follows.

3.2 Formation cost function

To represent the structure of the formation and interaction among UAVs, the graph theory is used. A graph is defined by $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, in which $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$ and $\epsilon = (v_n, v_{n'}) \in \mathcal{E}$ represent the drones in the group and their interconnections, respectively. To form a formation, there must exists an interconnection in $\mathcal{E}$ between any two vertices $(v_n, v_{n'}) \in \mathcal{V}$. Consider our graph is egde-weighted, i.e., each interconnection in the graph is weighted by $\mu_{nn'}$. The graph incidence matrix $\mathcal{D}$ has the dimension of $N \times M$, where its uv-th entry is equal to 1 or -1 if the UAV$_n$ is the head or tail of the v-th edge, and 0 otherwise.

The formation error between UAV$_n$ and UAV$'_n$ is calculated as $P_n - P_{n'} - P_{nn'_r}$. Let $\hat{\mathcal{D}} = \mathcal{D} \otimes I_3$, where operator $\otimes$ is the Kronecker product. From the definition of the incidence matrix, the formation error for UAV$_n$ can be expressed as

$$\begin{aligned} E_n &= \sum_{n' \in \mathcal{E}} \mu_{nn'} \|P_n - P_{n'} - P_{nn'_r}\|^2 \\ &= (P - P_r)^T \hat{\mathcal{D}} \hat{W}_n \hat{\mathcal{D}}^T (P - P_r) = \|P - P_r\|^2_{\hat{\mathcal{D}} \hat{W}_n \hat{\mathcal{D}}^T}, \end{aligned} \tag{5}$$

where $\hat{W}_n = W_n \otimes I_3$ and $W_n = \text{diag}[\mu_{nn'}]$ is a diagonal weight matrix of dimension M.

Let $\bar{d}_n(k)$ be the Euclidean distance from UAV$_n$ to its nearest neighbor at the waypoint k, r_n be the radius of UAV$_n$, and d_s be the safe distance. To avoid collision between vehicles, the distance between a UAV and its nearest neighbor needs to be smaller than the sum of a safe distance, d_s, and twice the UAV radius, r_n. Therefore, we introduce the so-called death penalty into the formation error as

$$E_n(k) = \begin{cases} \|P(k) - P_r\|^2_{\hat{\mathcal{D}} \hat{W}_n \hat{\mathcal{D}}^T}, & \text{if } \bar{d}_n(k) > d_s + 2r_n \\ \infty, & \text{if } \bar{d}_n(k) \leq d_s + 2r_n. \end{cases} \tag{6}$$

The formation cost function is then defined as below:

$$J_f(X_n, X_n^-) = \sum_{k=1}^{K} E_n(k). \tag{7}$$

4 Stag hunt game-based algorithm for UAV cooperative path planning

Given the cost function $J(X_n, X_n^-)$ defined for each UAV, the cooperative path planning becomes finding paths $X_n, n = 1, 2, \ldots, N$ to simultaneously minimize $J(X_n, X_n^-)$. Since this cost depends on not only path X_n generated for UAV$_n$ but also its rivals' paths X_n^-, finding optimal solutions is a challenging problem that requires a new method.

4.1 The game of stag hunt

As an essential branch of mathematics, game theory is the study of conflicts and interactions among rational decision-makers [18]. The game players can pursue their individual objectives by considering possible goals, behaviors, and countermeasures of other decision-makers to achieve a win-win situation.

The stag hunt game, which originated in [19], also called the coordination game or trust dilemma in game theory literature, illustrates a conflict between safety and social cooperation. In the game, two hunters independently decide whether to hunt a stag or a hare without knowing the other's decision. One hunter can catch a hare individually with a high guarantee of success. Meanwhile, the value of a shared stag is far greater than that of a hare, but cooperation between hunters is required to hunt a stag successfully. Therefore, it would be much better for each hunter to choose a more ambitious and far more rewarding goal instead of deciding on a total autonomy and minimal risk strategy. The payoff matrix in Figure 3 illustrates a generic stag hunt with two players.

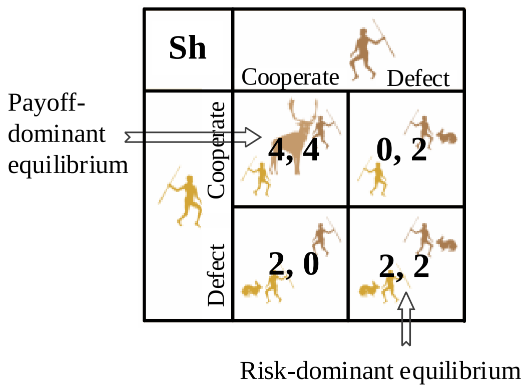

Figure 3. Payoff matrix of stag hunt game

In a game, "Nash equilibrium" is a situation where the optimal outcome of a game has no incentive to deviate from the initial strategy. In other words, no player can obtain more profits if others do not change their strategies. Formally, a stag hunt is a game with two pure Nash equilibria: risk dominant and payoff dominant. A Nash equilibrium is called "risk dominant" if it has the largest basin of attraction, implying that it is less risky. This means that the more ambiguity players have about the other player's intentions, the more likely they are to pick the plan that best suits them. Meanwhile, the payoff dominant equilibrium is defined as being Pareto superior to all other Nash equilibria in the game.

Pareto optimality is a fundamental concept representing efficiency in a multi-objective optimization problem consisting of several conflicting objectives. A set of alternatives is considered a Pareto optimal solution if no reallocation can further improve any one of the objectives without degrading at least one other. In the stag hunt game, when confronted with a decision among multiple equilibria, all players would vote on the payoff dominant one since it provides each member with at least as much profit as the other Nash equilibria.

From the mathematical point of view, a formal presentation of the payoff-dominant equilibrium is as follows. Consider a stag hunt game decribed as

$$G = (N, S, U), \tag{8}$$

where N is a set of players, $S = (S_1, S_2, ..., S_N)$ denotes strategy sets, where $S_n = (x_{n_1}, x_{n_2}, ..., x_{n_\Sigma}), n = \{1, ..., N\}$, represents all Σ strategies made by the n-th player, and a $U = (U(x_{n_1}), U(x_{n_2}), ..., U(x_N)$ stands for a set of players's utility. The allocation $\overset{*}{X} = \{\overset{*}{x}_1, \overset{*}{x}_2, ..., \overset{*}{x}_N\}$, where $\overset{*}{X}_n \in S_n$, is defined as Pareto optimality if it dominates all other reallocation $X = \{x_1, x_2 ..., x_N\}$, i.e., both of the following requirements are met:

$$\forall n \in \{1, ..., N\} | U(\overset{*}{x}_n) \leq U(x_n), \tag{9a}$$

$$\nexists n' \in \{1, ..., N\} | U(\overset{*}{x}_{n'}) < U(x_{n'}). \tag{9b}$$

4.2 The game of UAV cooperation

Each vehicle in the cooperative path planning problem has been assigned a cost function $J(X_n, X_n^-)$ defined in (4). These cost functions interact among vehicles. Therefore, it is challenging to solve multiple optimal problems simultaneously. As can be seen, the stag hunt game concept aligns well with the cooperative path planning problem and thus can be used as a theoretical framework to solve it. Motivated by those observations, this paper proposes a stag hunt game-based approach for cooperative UAVs consisting of two steps. The two-step procedure to implement the proposed game-based scheme is as follows.

In the first step, a UAV game framework is formulated to model interactions among the drones, including three key elements: players, strategies, and utility. Each vehicle in the formation is considered as a player, also called a decision-maker. During the game, all UAV players have to simultaneously provide a route, X_n, defined as the player's strategy, without knowledge of the other player's decision. Each player will get his own utility, corresponding to the multi-vehicle cost $J(X_n, X_n^-)$, which is a function of strategies made by himself X_n and his rivals, X_n^-.

Indeed, one UAV can reach its target position alone by solving its single UAV path planning problem to obtain the minimum single cost, $J_s(X_n)$. In a cooperative task, however, this individual optimal solution could result in a significant formation error, $E_n(k)$, leading to a high formation cost $J_f(X_n, X_n^-)$ and hence much less reward. To successfully perform a cooperative UAV mission with a higher profit, it would be much better for each player to choose the more ambitious goal of achieving a far greater reward by providing a formation preserving path in exchange for the other vehicle's cooperation. Therefore, the multi-vehicle cost, $J(X_n, X_n^-)$, combining both single cost $J_s(X_n)$ and formation cost $J_f(X_n, X_n^-)$, should be optimized for all players simultaneously. Accordingly, the UAV game aims to find a payoff-dominant equilibrium.

In the second step, an enhanced PSO-based algorithm is introduced to solve the Pareto optimality, resulting in a payoff-dominant equilibrium as a desired outcome of the game. This step will be presented in the following section.

4.3 Enhanced PSO-based approach for finding payoff-dominant equilibrium

Particle swarm optimization (PSO) is a stochastic optimization algorithm for optimizing a problem by iteratively improving a candidate solution concerning a particular quality measure. It solves a problem by generating a population of possible solutions, known as particles, and relocating them in the search space using a few simple formulae based on the particle's position and velocity. Each particle's movement is guided by its local best-known position and the global best-known pose in the search space, updated when other particles discover better places. This is anticipated to direct the swarm toward the best options.

Formally, consider a d-dimension search space and a swarm consisting of N_{pop} particles, each partilce i has a position $X_i \in \mathbb{R}^d$ and a velocity $V_i \in \mathbb{R}^d$. Let Q_i be the best known position of particle i and Q_g be the best known position of the entire swarm. The movement algorithm of the swarm is defined as below:

$$V_i(t+1) = c_0 V_i(t) + c_1 r_1 [Q_i(t) - X_i(t)] + c_2 r_2 [Q_i(t) - X_i(t)], \tag{10}$$

$$X_i(t+1) = X_i(t) + V_i(t+1), \tag{11}$$

where c_0 is the inertia weight, c_1 and c_2 are corresponding to self confidence and swarm confidence parameters, and r_1 and r_2 are random values uniformly distributed in the interval $[0, 1]$.

In the UAV path planning problem, the position of a particle is encoded by the flight path X_n. Accordingly, the entire swarm consists of N_{pop} path particles, which are updated to search for the optimal solution. To speed up the search process, we employ in this study a variant of PSO named spherical vector-based particle swarm optimization (SPSO) developed in our previous work [17]. In the SPSO, waypoints of a flight path are represented in the spherical coordinate system to exploit the corresponding magnitude, elevation, and azimuth components of the variables with speed, turning angle, and climbing slope of the UAV.

To further develop the SPSO for cooperative path planning involving multiple UAVs, we introduce an enhanced SPSO to find the payoff-dominant equilibrium. The pseudo-code for the optimization process is described in Algorithm 1. The detail of the algorithm is as follows.

Algorithm 1 Enhanced PSO implementation

1. Initialize PSO parameters: $c_0, c_1, c_2, maxIt, nPop$;
2. Set $It = 0$, generate random player's strategies;
3. Obtain the initial optimal strategies $\overset{*}{X}_n$, $\overset{*}{X}_n^-$;

for $It = 1 : maxIt$ **do**
  4. Calculate $J(X_n(It), X_n^-(It))$, for $n = 1, 2, ..., N$;
  if $J(X_n(It), X_n^-(It)) \leq J(\overset{*}{X}_n, \overset{*}{X}_n^-)$, $\forall n = 1, 2, ..., N$ **then**
    5. Update $\overset{*}{X}_n = X_n(It)$; $\overset{*}{X}_n^- = X_n^-(It)$;
  end if
  6. Record $\overset{*}{X}_n$, $\overset{*}{X}_n^-$;
  7. Update X_n, X_n^-;
end for
8. Obtain $\overset{*}{X}_n$, $\forall n = 1, 2, ..., N$.

(i) *Initialization:*

Initially, parameters of the PSO including c_0, c_1, c_2, number of iterations $maxIt$, and number of particles $nPop$ are first initialized. At this stage, corresponding to $It = 0$, random strategies of the players are also generated and assigned as initial optimal strategies $\overset{*}{X}_n$, $\overset{*}{X}_n^-$.

(ii) *Evaluation:*

At each iteration, from $It = 1$ to $maxIt$, cost values representing the players' profit, $J_i(X_n(It), X_n^-(It))$, are computed as (4), where i denotes a particle in the n-th swarm.

(iii) *Optimize the player's strategy:*

The best strategies of all players associated to the particle i at the iteration It are updated if there is more benefit for at least one player without decreasing the other players' profit, i.e., the condition (9) is met. Based on them, the strategy of each player is adjusted for the subsequent iteration according to equations (10) and (11) of the PSO.

(iv) *Terminate the optimal strategies:*

Check if the multi-profit optimization criteria are met to terminate the process. At the end of this stage, payoff-dominant equilibrium, or the best allocation, $\overset{*}{X} = \{\overset{*}{X}_1, \overset{*}{X}_2, ..., \overset{*}{X}_N\}$, is obtained.

5 Simulation result

This section presents simulation results of the proposed path planning algorithm for a fleet of three drones. It aims to generate paths for three UAVs flying in an equilateral triangle formation. However, it should be noted that there are no specific constraints on the configuration of formation shapes. The incidence matrix $\mathcal{D}$ is defined as

$$\mathcal{D} = \begin{bmatrix} 1 & 1 & 0 \\ -1 & 0 & 1 \\ 0 & -1 & -1 \end{bmatrix}. \tag{12}$$

The interconnection weights are set as $W_1 = [1\ 1\ 0]$, $W_2 = [1\ 0\ 1]$, and $W_3 = [0\ 1\ 1]$. Noting that the weights for interconnections are set equal among players since all players play a similar role in the team.

In the simulation, we consider two scenarios with different sizes of threats, illustrating different levels of complexity, to validate the efficiency of the proposed algorithm.

5.1 Evaluation in scenario with big-size obstacles

Scenario 1 considers a construction ~~land~~ with dimensions of $100m \times 100m \times 35m$. The drones are required to travel from the start location to the goal to perform a site monitoring task. The start locations of UAVs are set at: $P_1^{start} = (15; 18.66; 20)$, $P_2^{start} = (10; 10; 20)$, and $P_3^{start} = (20; 10; 20)$. The goal poses are confirmed at $P_1^{goal} = (85; 88.66; 20)$, $P_2^{goal} = (80; 80; 20)$, and $P_3^{goal} = (90; 80; 20)$. The formation reference is obtained at the same as the target position, i.e., $P_{1_r} = P_1^{goal}$, $P_{2_r} = P_2^{goal}$, and $P_{3_r} = P_3^{goal}$. In the construction site, there two threat areas modeled as yellow cylinders located at $(40, 40)$ and $(60, 60)$ with a radius of 9 m. Parameters of the PSO are set as $c_0 = 0.999$, and $c_1 = c_2 = 1.5$. The PSO run with 2000 particles for 1500 iterations. Aside from the start and goal nodes, each path is established by $K = 10$ waypoints.

In the simulation, we consider two scenarios with different sizes of threats, illustrating different complexity levels to validate the efficiency of the proposed algorithm.

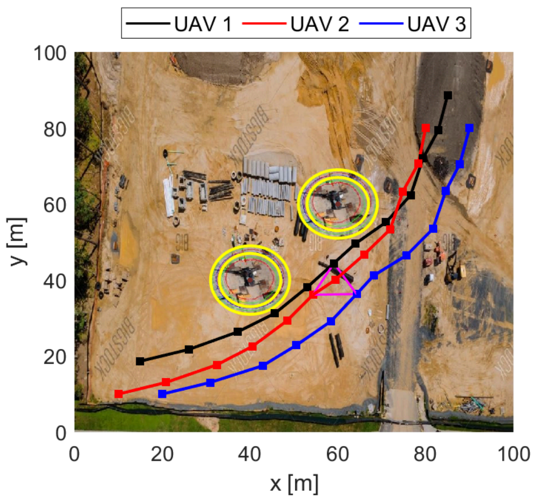

Figure 4. Generated paths for Sencario 1

5.2 Comparison in scenario with small-size obstacles

Scenario 2 examines the UAV team that has to fly ~~at~~ a construction site to ~~deliver multiple packages, such as building materials~~. The map dimensions, UAV's start and goal locations in Scenario 2 are similar to those in Scenario

1. In Scenario 2, however, two construction cranes as obstacles are located at $(40,48)$ and $(74,43)$ with a radius of 4 m. To enhance the level of complexity, two more virtual obstacles with the same radius are added at $(20,70)$ and $(70,60)$. In this simulation, the performance of the proposed stag hunt game-based algorithm is compared with other available techniques that treat the entire UAV fleet as a rigid body, and path planning is achieved for a virtual drone placed at the centre of the formation [20].

Figure 5 and Figure 6 depict the planned stag hunt game-based path and the planned rigid formation path, respectively. It can be seen that both techniques achieve collision-free and formation-preserving routes. However, the whole group of UAVs, using the rigid formation method, travels around obstacles, resulting in long distances. Meanwhile, the proposed game-based approach presents a capability to split and merge the UAV fleet to avoid small-sized threats and reduce the cost. This further illustrates the benefit of the proposed algorithm.

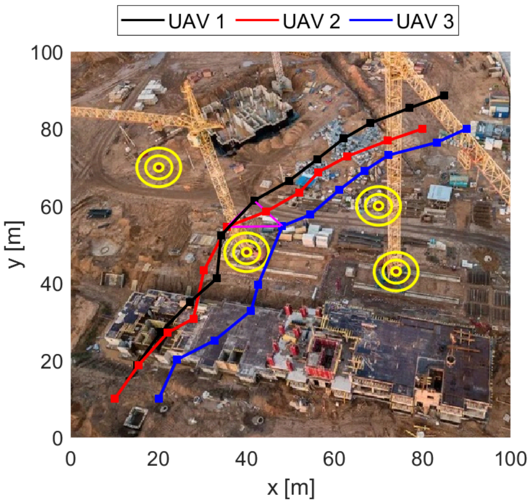

Figure 5. Stag hunt gamed based path

Figure 7 depicts the convergence of all stag hunt game-based profit values after 1000 iterations for all UAVs, implying that the payoff-dominant equilibrium in the UAV game is achieved. Compared to the best cost value of the rigid formation cost, as illustrated in Figure 8, all three UAVs achieved better utility. This further confirms that the obtained game-based strategy dominates the rigid formation strategy.

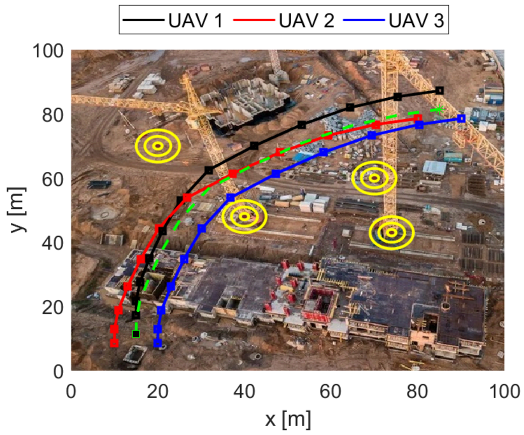

Figure 6. Rigid formation path

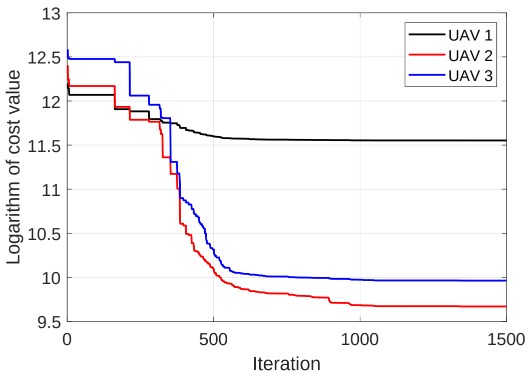

Figure 7. Stag hunt gamed-based profits

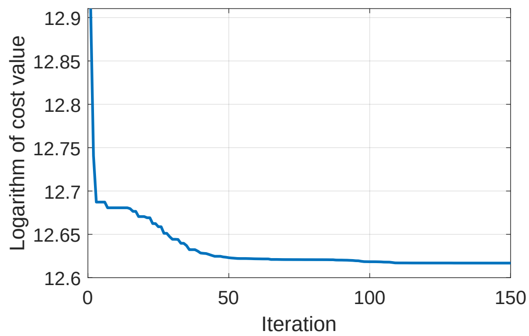

Figure 8. Rigid formation profit

6 Conclusion

This study has introduced a novel method based on the stag hunt game theory and enhanced particle swarm optimization for cooperative UAVs navigating ~~a desired~~ formation configuration. The UAV collaborative path planning problem is solved by finding the payoff dominant equilibrium of a stag hunt-based UAV game. An optimization framework using PSO was integrated to find the Pareto optimality by minimizing all cost functions simultaneously. Simulations have been conducted to evaluate the performance of the proposed method. Our future work will develop receding horizon game theory-based platforms for cooperative UAVs in a dynamic construction environment.

Acknowledgements

This work is supported by the Vingroup Science and Technology Scholarship (VSTS) Program for Overseas Study for Master's and Doctoral Degrees. The VSTS Program is managed by VinUniversity and sponsored by Vingroup.

References

[1] Alexey Bulgakov, Daher Sayfeddine, Thomas Bock, and Awny Fares. Generation of orthomosaic model for construction site using unmanned aerial vehicle. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 37, pages 900–904. IAARC Publications, 2020.

[2] N Bolourian, MM Soltani, AH Albahria, and A Hammad. High level framework for bridge inspection using lidar-equipped uav. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 34. IAARC Publications, 2017.

[3] Jacek Grzybowski, Karol Latos, and Roman Czyba. Low-cost autonomous uav-based solutions to package delivery logistics. In *Advanced, Contemporary Control*, pages 500–507. Springer, 2020.

[4] Fran Real, Ángel R Castaño, Arturo Torres-González, Jesús Capitán, Pedro J Sánchez-Cuevas, Manuel J Fernández, Manuel Villar, and Aníbal Ollero. Experimental evaluation of a team of multiple unmanned aerial vehicles for cooperative construction. *IEEE Access*, 9:6817–6835, 2021.

[5] Thi Thoa Mac, Cosmin Copot, Robin De Keyser, and Clara M Ionescu. The development of an autonomous navigation system with optimal control of an uav in partly unknown indoor environment. *Mechatronics*, 49:187–196, 2018.

[6] Farid Kendoul. Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *Journal of Field Robotics*, 29(2):315–378, 2012.

[7] Rahul Kala. Multi-robot path planning using co-evolutionary genetic programming. *Expert Systems with Applications*, 39(3):3817–3831, 2012.

[8] Muaz O Ahmed, Islam H El-Adaway, Kalyn T Coatney, and Mohamed S Eid. Construction bidding and the winner's curse: Game theory approach. *Journal of Construction engineering and Management*, 142 (2):04015076, 2016.

[9] Benjamin C Collins and Mustafa Kumral. Game theory for analyzing and improving environmental management in the mining industry. *Resources Policy*, 69:101860, 2020.

[10] José Moura and David Hutchison. Game theory for multi-access edge computing: Survey, use cases, and future trends. *IEEE Communications Surveys & Tutorials*, 21(1):260–288, 2018.

[11] Zhu Han, Dusit Niyato, Walid Saad, Tamer Başar, and Are Hjørungnes. *Game theory in wireless and communication networks: theory, models, and applications*. Cambridge university press, 2012.

[12] Shiau-Jing Ho, Sheng-Lung Lin, and Hui-Ping Tserng. An analysis using game theory on the investment incentive of ppp projects. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction*, volume 34. IAARC Publications, 2017.

[13] Xianlin Zeng, Jie Chen, Shu Liang, and Yiguang Hong. Generalized nash equilibrium seeking strategy for distributed nonsmooth multi-cluster game. *Automatica*, 103:20–26, 2019.

[14] Michael W Macy and Andreas Flache. Learning dynamics in social dilemmas. *Proceedings of the National Academy of Sciences*, 99(suppl 3):7229–7236, 2002.

[15] Brian Skyrms. *The stag hunt and the evolution of social structure*. Cambridge University Press, 2004.

[16] Herbert Gintis. *Game theory evolving*. Princeton university press, 2009.

[17] Manh Duong Phung and Quang Phuc Ha. Safety-enhanced uav path planning with spherical vector-based particle swarm optimization. *Applied Soft Computing*, 107:107376, 2021.

[18] Ankang Ji, Xiaolong Xue, QP Ha, Xiaowei Luo, and Minggong Zhang. Game theory–based bilevel model for multiplayer pavement maintenance management. *Automation in Construction*, 129:103763, 2021.

[19] Jean-Jacques Rousseau. Discourse on the origin and basis of inequality among men. *Rousseau J.-J. Traktaty*, pages 31–109, 1754.

[20] YongBo Chen, JianQiao Yu, XiaoLong Su, and GuanChen Luo. Path planning for multi-uav formation. *Journal of Intelligent & Robotic Systems*, 77 (1):229–246, 2015.