UKACC International Conference on Control 2012
Cardiff, UK, 3-5 September 2012

Resource Allocation with Cooperative Path Planning for Multiple UAVs

Hyo-Sang Shin,Cédric Leboucher, and Antonios Tsourdos
Department of Informatics and Systems Engineering
Cranfield University, Defence Academy of the UK
Swindon SN6 8LA, UK
Email: h.shin; c.leboucher; a.tsourdos@cranfield.ac.uk

***Abstract*—This study proposes an optimal resource allocation algorithm of multiple UAVs with cooperative path planning using a geometric approach. The focus of the resource allocation is on mission and task completion, also known as feasibility whilst coping with operational and physical constraints of UAVs. Therefore, this study first introduces a geometric path planning algorithm based on Pythagorean Hodgraphs (PH). Using Bernstein Bzier polynomials, the path planning algorithm can generate feasible and safe (obstacle and inter-collision free) paths which can also meet position and orientation constraints of UAVs. We then optimise the resource allocation based on Evolutionary Game Particle Swarm Optimisation (EGPSO) and paths generated by the geometric planning. The input parameter of the optimal allocation problem is the allocation policy and the performance index is chosen to be the total flight time of the UAVs. Here the flight time is computed from the path produced by the path planning algorithm. The optimal allocation algorithm changes the allocation policy and finds the best allocation policy which minimise the performance index. The performance of the proposed algorithm is investigated by numerical examples simulated under realistic scenarios.**

I. Introduction

Inexpensive unmanned aerial vehicles (UAVs) have considerable potential for use in remote sensing operations. They are cheaper and more versatile than manned vehicles, and are ideally suited for dangerous, long and/or monotonous missions that would be inadvisable or impossible for a human pilot. Groups of UAVs are of special interest due to their ability to coordinate simultaneous coverage of large areas, or cooperate to achieve common goals. Specific applications under consideration for groups of co-operating UAVs include, but not limited, border patrol, search and rescue, surveillance, mapping and environmental monitoring. In these applications, the group of UAVs becomes a mobile resource/sensor and consequently routes and tasks for each UAV need to be properly and optimally assigned in order to cooperatively achieve their mission. Therefore, this study addresses the vehicle routing problem of of multiple UAVs.

The vehicle routing problem, has been mainly handled in the operational research area ([1], [2], [3], [4]) and can be generally classified by two categories: one is the Traveling Salesman Problem (TSP) which finds a shortest circular trip through a given number of cities, and the other is the Chinese Postman Problem (CPP) finding the shortest path with considering path constraints on an entire network of road. The TSP using multiple UAVs can be considered as a task assignment problem to minimise the cost of time or energy for a certain mission by assigning each target to an UAV, for which binary linear programming ([5]), iterative network flow ([6]), tabu search algorithm ([7]) and receding horizon control ([8]) have been proposed. Recently, [9] proposed a route optimisation algorithm for multiple searchers to detect one or more probabilistically moving targets incorporating other factor such as environmental and platform-specific effects. Meanwhile, the CPP is normally used for ground vehicle applications such as road maintenance, snow disposal ([10]), boundary coverage ([11]), and graph searching and sweeping ([12], [13]). Since the general vehicle routing algorithms approximate their path to a straight line shape to reduce computational load, the physical constraints imposed on the vehicle are not to be addressed.

In order to mitigate this issue, this study divides the routing problem into two parts: the first part is to design cooperative path planning and the second one is to find the optimal resource allocation policy based on the paths obtained in the first part. Cooperative path planning algorithm is designed using the differential geometry concepts, especially Pythagorean Hodographs (PH) curves, which was proposed in our previous study [14]. Path planning algorithms based on differential geometry examine the evolution of guidance geometry over time to derive curvature satisfying the guidance goals. Guidance command such as a manoeuvre profile can be then computed using the derived curvature of the guidance geometry. One of main advantages of this approach is that the number of design parameters can be significantly reduced whilst maintaining the guidance performance. Therefore, this approach will enable us not only to design fast and more lightweight algorithms, but also to generate safe and feasible paths for multiple UAVs. This would be preferred for integration of path planning with the optimal resource allocation. Since reaching targets at the same instant with specific orientations could improve the overall effectiveness and survivability of the UAVs, simultaneous arrivals with predefined orientations are considered as constraints with the physical ones such as obstacle avoidance and the maximum turning rate of of the UAVs.

The performance index for the optimal resource allocation problem is the total flight time of the multiple UAVs since this is necessary to have a manageable task in the available time. The total flight time for each candidate allocation is computed by using the velocity profile and paths generated by cooperative path planning and this is used to find the optimal allocation policy. The optimisation method implemented in this paper is Evolutionary Game Particle Swarm Optimisation (EGPSO) which is proposed in our previous study [15]. The proposed EGPSO algorithm integrates the Evolutionary Game Theory (EGT) concepts with those of Particle Swarm Optimisation to find the optimal weight of the coefficients considering the entire swarm fitness. Moreover, it is shown that this algorithm efficiently works in the general allocation problem [15].

The overall structure of this paper is given as: Section II briefly introduces a target tracking filter design, trajectory classification to model the behaviour of ground vehicles, and behaviour recognition algorithm using string matching theory. Section III introduces rule-based decision making algorithm to find suspicious or anomalous behaviour based on a fuzzy logic. Section IV presents numerical simulation results of behaviour monitoring for both military and civilian traffic scenario using realistic ground vehicle trajectory data. Lastly, conclusions and future works are addressed in Section V.

II. Problem Formulation

A. Scenario

The scenario considered in this study is similar to that in our previous paper [14]. In the scenario, it is assumed that a group of N UAVs leaves from a base and they have to reach the target area at the same time with predetermined orientations. The individual start and finish points for each UAV are represented by position coordinates (x, y) and orientation by angle θ. These are assumed to be known a priori. The UAVs are assumed to be of same type and are flying at same the speed at constant altitude. Each UAV has the same maximum bound on its curvature and the environment has static obstacles. The UAVs are required to avoid collision with other UAVs and with other objects in the air-space, as well as avoiding the static obstacles. The allocation of the UAVs needs to be optimised.

B. Optimal Allocation Problem

Let us first consider a path between a single UAV from the base to the target position with no constraints. The starting point P_s is at the base position and the finishing point P_f is at the target position. The path connecting the poses is represented by the label r. The path planner produces a path connecting the start pose $P_s(x_s, y_s, \theta_s)$ and the finish pose $P_f(x_f, y_f, \theta_f)$.

$$P_s(x_s, y_s, \theta_s) \xrightarrow{r(t)} P_f(x_f, y_f, \theta_f) \tag{1}$$

where t is a path length parameter.

Extending equation (1) to account for a group of N UAVs gives:

$$P_{si}(x_{si}, y_{si}, \theta_{si}) \xrightarrow{r_i(t)} P_{fi}(x_{fi}, y_{fi}, \theta_{fi}), \tag{2}$$

$$\max|\kappa_i| < \kappa_{max}, \coprod_{safe}, \coprod_{length}, \ i = 1 \dots N, \tag{3}$$

where κ is the path curvature, κ_{max} is the maximum curvature bound obtained from the maximum turning rate, and $\coprod_{\text{safe}}$ and $\coprod_{\text{length}}$ are the constraints on safety and path length respectively.

The safety constraints are described as:

$$\begin{aligned} \coprod_{length} \quad : \quad & d(P_i(t), P_j(t)) > d_{sep}, \\ & d(P_i(t), O_k(t)) > d_{sep} \\ & i \neq j = 1 \dots N, k = 1, \dots, n_o \end{aligned} \tag{4}$$

where $P_i(t)$ and $O_k(t)$ are the positions of the ith UAV and the kth obstacle, n_o the number of obstacles, and d_{sep} the minimum separation distance. To enable simultaneous arrivals, the constraint on length is given as:

$$\begin{aligned} \coprod_{length} : s_i(t_{fi}) &= s_{cm}, i = 1 \dots N \\ s_i(t_{fi}) &= \int_{t_{si}}^{t_{fi}} \sqrt{\dot{x}_i(t)^2 + \dot{y}_i(t)^2} dt, \end{aligned} \tag{5}$$

where s_{cm} is a common path length which is automatically obtained regarding to the allocation.

The optimal allocation problem is then formulated as: minimising the following performance index

$$J = \Sigma_1^N \int_{t_{si}}^{t_{fi}} dt \tag{6}$$

subject to equation (2) and (3).

III. Cooperative Path Planning Using Pythagorean Hodographs Curves

One of well known path planning approaches based on the differential geometry concepts is Dubins path planning [14], [16]. The Dubins trajectory ([17]) is the shortest path connecting two configurations represented by position and pose under the constraints of a bound on curvature or turning radius. The Dubins path is a composite curve of both lines and circles and is easy to produce. However, it lacks a smooth variation of curvature. Mathematically, the Dubins path provides only tangent continuity, C^1. The curvature continuity is important as the curvature is proportional to the lateral acceleration of the UAV. Therefore, curvature discontinuity results in an abrupt maneouvre of the UAV. A smooth motion needs curvature continuity C^2. Therefore, it is necessary to seek for an alternate path with curvature continuity. The equation of curvature is:

$$\kappa(t) = \frac{\dot{\mathbf{r}} \times \ddot{\mathbf{r}}}{|\dot{\mathbf{r}}|^3}, \quad \dot{\mathbf{r}} = \frac{d\mathbf{r}}{dt}, \ddot{\mathbf{r}} = \frac{d^2\mathbf{r}}{dt^2} \tag{7}$$

From the equation (7), the curvature is a function of first two derivatives of a curve, $\mathbf{r}(t)$, so the path needs to be at least twice continuously differentiable, that is C^2 continuity. There are many polynomial curves which can provide C^2 continuity. However, we choose Pythagorean Hodograph (PH) curve known for its rational properties.

A smoother curve can be produced by using techniques such as PH, where basis curves are used to piece together using Bernstein Bézier polynomials. For a planar parametric curve, $\mathbf{r}(t) = \{x(t), y(t)\}$, the hodographs are $\dot{x}(t)$ and $\dot{y}(t)$ so the velocity vectors of a curve are its hodograph. The path-length of the curve $\mathbf{r}(t)$ is:

$$s = \int_{t_1}^{t_2} \sqrt{\dot{x}^2 + \dot{y}^2} dt = \int_{t_1}^{t_2} |\dot{\mathbf{r}}(t)| dt \tag{8}$$

where s is the path-length, t is a parameter such that $t \in [t_1, t_2]$ and $\dot{x} = \frac{dx}{dt}$, $\dot{y} = \frac{dy}{dt}$, and $\dot{r} = \frac{dr}{dt}$. The term inside the square root in equation (8) is the sum of the square of the hodographs of the curve, $\mathbf{r}(t)$. For the PH path the path-length is an integral of a polynomial $\sigma(t)$ such that:

$$\begin{aligned} \sigma(t) &= \sqrt{\dot{x}^2 + \dot{y}^2} \\ s &= \int_{t_1}^{t_2} |\sigma(t)| dt \end{aligned} \tag{9}$$

The PH curve can be produced by selecting two Bernstein polynomials $u(t)$ and $v(t)$ such that:

$$\dot{x}(t) = u^2(t) - v^2(t) \tag{10}$$

$$\dot{y}(t) = 2u(t)v(t) \tag{11}$$

$$\tag{12}$$

This gives:

$$|\sigma(t)| = u^2(t) + v^2(t) \tag{13}$$

Note that the PH path thus provides exact calculation of path length and it's curvature, as well as the orders of the two polynomials, $u(t)$ and $v(t)$, determine the order of PH curves. In this research, we use a fifth order PH curve as this is the lowest order curve which has inflexion points which can provide sufficient flexibility [**?**]. For a fifth order PH curve, $u(t)$ and $v(t)$ can be approximated as second order polynomials:

$$u(t) = \sum_{k_u=0}^{2} u_k \binom{2}{k_u} (1-t)^{(2-k_u)} t^{k_u}; \tag{14}$$

$$v(t) = \sum_{k_v=0}^{2} b_k \binom{2}{k_v} (1-t)^{(2-k_v)} t^{k_v}; \tag{15}$$

Hence the curve, $\mathbf{r}(t)$, is given by Berstein form:

$$\mathbf{r}(t) = \sum_{k=0}^{5} P_k \binom{5}{k} (1-t)^{(5-k)} t^k; \tag{16}$$

where $P_k(x_k, y_k),\ k = 0, 1, 2, 3, 4, 5$ are control points. Note that these control points determine the curve $\mathbf{r}(t)$ and can be derived as:

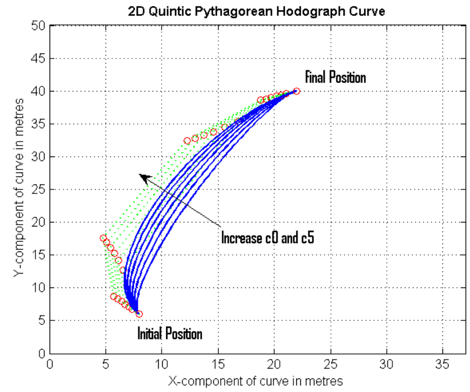

Fig. 1. PH paths

$$P_1 = P_0 + \frac{1}{5}\begin{pmatrix} u_0^2 - v_0^2 \\ 2u_0v_0 \end{pmatrix} \tag{17a}$$

$$P_2 = P_1 + \frac{1}{5}\begin{pmatrix} u_0u_1 - v_0v_1 \\ u_0v_1 + u_1v_0 \end{pmatrix} \tag{17b}$$

$$P_3 = P_2 + \frac{1}{5}\begin{pmatrix} u_1^2 - v_1^2 \\ 2u_1v_1 \end{pmatrix} + \frac{1}{15}\begin{pmatrix} u_0u_2 - v_0v_2 \\ u_0v_2 + u_2v_0 \end{pmatrix} \tag{17c}$$

$$P_4 = P_3 + \frac{1}{5}\begin{pmatrix} u_1u_2 - v_1v_2 \\ u_1v_2 + u_2v_1 \end{pmatrix} \tag{17d}$$

$$P_5 = P_4 + \frac{1}{5}\begin{pmatrix} u_2^2 - v_2^2 \\ 2u_2v_2 \end{pmatrix} \tag{17e}$$

When the initial and final configurations (pose and heading) for each vehicles are known, $P_k(x_k, y_k),\ k = 0, 1, 2, 3, 4, 5$ can be specified. From the configurations, P_0, P_1, P_4, P_5 are directly obtained as:

$$P_0 = (x_s, y_s) \tag{18a}$$

$$P_5 = (x_f, y_f) \tag{18b}$$

$$P_1 = P_0 + (1/5) * d_0 \tag{18c}$$

$$P_4 = P_5 - (1/5) * d_5 \tag{18d}$$

where

$$d_0 = c_0(\cos(\theta_s), \sin(\theta_s)) \tag{19}$$

$$d_5 = c_5(\cos(\theta_f), \sin(\theta_f)) \tag{20}$$

where $c_0 \in (0, \infty]$ and $c_5 \in (0, \infty]$. Note that when c_0 and c_5 are specified, the control points (P_0, P_1, P_4, P_5) in (18) are fixed by configuration. Moreover, u_i and v_i for $i = 1, 2, 3$ are also uniquely derived from equation (17), which implies that P_2 and P_3 are also fixed. Therefore, the number of control parameters reduces to two of c_0 and c_5 whilst maintaining the continuity of the curve. Increasing the values of c_0 and c_5 will increase the length of tangent vectors $P_0 = |\boldsymbol{P_0P_1}|$ and $P_5 = |\boldsymbol{P_5P_4}|$ and in turn P_2 and P_3 get shifted to meet the PH condition (17). This is shown in figure 1. As shown in this figure, varying the two control parameters controls the curvature, which in turn will determine the space curve.

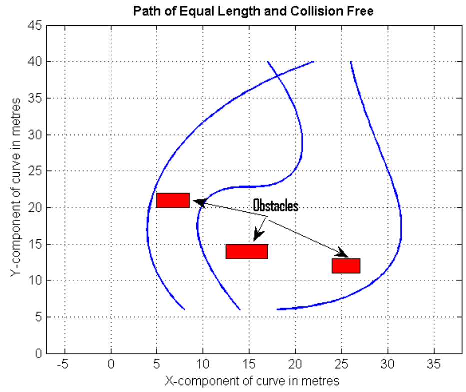

Fig. 2. PH paths, all equal in length and avoiding obstacles

For simultaneous arrival, the paths are required to be made equal in path if UAV speeds are constant and same. The variable speed UAVs can have difference in path lengths. This is achieved by increasing the shorter paths to that of the longest one. The path lengths of the flyable and safe paths are calculated using (8). For N number of UAVs, with the length of each path s_i, the set of path lengths Σ is:

$$\Sigma = \{s_i\}, \quad i = 1, \ldots, N \tag{21}$$

The longest of the safe flyable path is the reference path. That is the maximum of Σ. The path lengths of $(N-1)$ UAVs are increased to that of the reference path. Lengths of the PH path is increased by changing the control parameters of c_0 and c_5:

$$\text{Find } c_0 \text{ and } c_5\text{, such that} \quad s_i - \max s_i = 0, \quad i = 1, \ldots N-1 \tag{22}$$

Obstacles as well as collisions can be also avoided by using the parametric freedom of c_0 and c_5. Such a set of paths (equal length and collision free) is shown in figure 2

IV. Resource Allocation using EGPSO

The approach, which is used to solve the allocation of the tasks to the UAVs in this paper, is a Discrete Particle Swarm Optimisation (DPSO) combined to Evolutionary Game Theory (EGT). One of the main implementation issues of DPSO is the choice of inertial, individual and social coefficients. In order to resolve this problem, those coefficients are optimised by using a dynamical approach based on EGT. The strategies are either to keep going with only inertia, or only with individual, or only with social coefficients. Since the optimal strategy is usually a mixture of the three coefficients, the fitness of the swarm can be maximized when an optimal rate for each coefficient is obtained. This algorithm is described in our previous study [15]. In this method, all the particles $X = (t_i)_{i \in [1 \ldots T]}$ are considered as a vector of feasible solution, where the t_i denote the tasks $i \in [1 \ldots T]$ and T is the number of tasks to achieve. The index of the vector represents the id of the UAV. Moreover, permuting the elements of X in this representation gives all the possible solutions so it could enable to deal with high dimensional problems. The single constraint of assigning one UAV to one task does not enable us to deal with the common DPSO algorithm described by Eberhart and Kennedy in [18]. Due to the context and the application, it was required to adapt the form of the particles according to the problem.

A. Swarm organisation

In the PSO algorithm, the establishment of the networks is a key point to maximise the exploration and the global efficiency of the algorithm to solve a problem. In this paper, in order to assigned the tasks to the UAVs, three different swarms are used. Each of them has its own features. One will adopt only inertial behaviour ($c_1 \neq 0$ and $c_2 = c_3 = 0$). One will adopt only selfish behaviour ($c_2 \neq 0$ and $c_1 = c_3 = 0$). One will adopt only social behaviour ($c_3 \neq 0$ and $c_1 = c_2 = 0$). Then the last one is following the common behaviour of the PSO with the coefficients determined by the result of EGT and the four previous swarms. The coefficients are chosen dynamically according to the current state of all the other swarms.

B. Particle movement

Using the common process of the PSO, the probability of movement toward another solution is introduced. To guarantee that the particles are moving on the feasible solution space and won't need to be repaired, we use an a priori method. In fact, the particles are built in such a way that the solution space is obtained by permuting feasible solutions. One of the key issue when it is required to convert the PSO into DPSO, is the computation of the velocity. Indeed, in discrete space, the velocity does not really makes sense and it was essential to adapt it to discrete case. The main approach used in [18], [19], [20], [21] is the sigmoid function which enables to convert a velocity into a probability. The proposed method is based on that principle, and set the sigmoid function as $s(v_{id}^t) = 1 - \frac{2}{1+e^{v_{id}^t}}$, $(\forall v_{id}^t \in \mathbb{R}_+)$ $(s_{id}^t \in [0,1])$. Once all the probabilities of change for a particle are obtained, the final step is to draw a random number and compare it to each coefficient of probability. All the coefficients greater than the random number are selected as potential candidate for a permutation. Then we draw randomly two particle coefficients and permute them. (In case we obtain a random number greater than only one coefficient, we consider the particle won't move).

C. Principle of EGT in the determination of the DPSO coefficients

In order to improve the convergence speed of the DPSO, it is proposed to combine it with EGT. If this way has already been investigated by Miranda and Fonseca in [22] to improve the local exploration of the particles, then by Di Chio in [23] and Liu and Wang [24]. The proposed approach is considering the global swarm's welfare instead of the particle's welfare. Thus, like described in IV-A, there are 3 available strategies to play: inertial, individual, or social. Each swarm will play one pure strategy and will provide its welfare to the others. Then, the EGT process, which is based on the replicator dynamic [15], find the equilibrium strategy which enables to the main swarm to improve or keep its mean welfare.

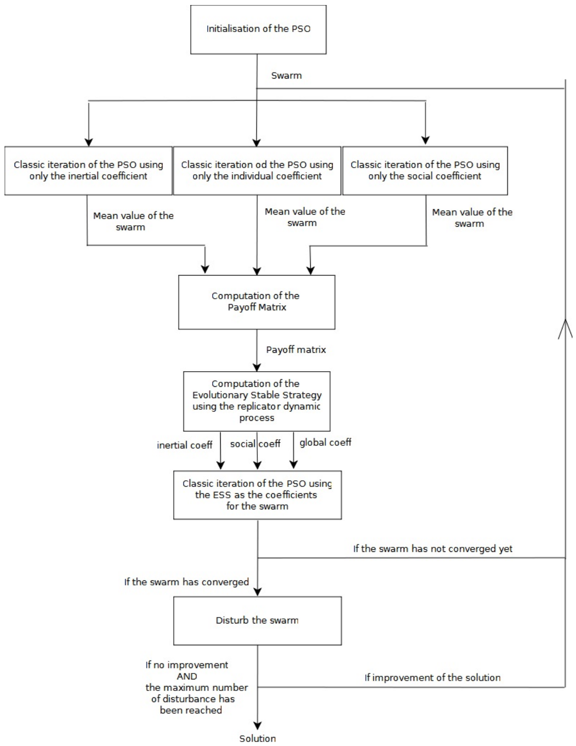

Fig. 3. Overview of the proposed method

D. Scheme of the process

The figure 3 shows how is designed the algorithm.

V. Numerical Simulations

In numerical simulations, a swarm of four UAVs is considered for mission deployment. All the UAVs will start from certain starting point at the same time and will reach the goal point at the same time. The goal points of each UAV will be determined from the optimal allocation. During their flights from starting position to finishing position all the UAVs will avoid inter-collisions and collisions with known obstacles. The proposed path planning algorithm will plan safe and flyable flight paths for all the UAVs in the group. For this simulation, the initial poses, the curvature constraints and the safety radii of all the UAVs are summarised in table I. The final poses,

TABLE I
Numerical simulation conditions

UAVs	*Position (m)*	*Heading (deg)*	*κ (m)*	*Safety Radius*
UAV1	(8, 6)	130	-1/13, +1/3	1
UAV2	(14, 6)	124	-1/13, +1/3	1
UAV3	(18, 6)	3	-1/13, +1/3	1
UAV4	(25, 6)	20	-1/13, +1/3	1

which represent targets, are given in table II. All the known static obstacles are given in the terrain database. For simplicity, all the obstacles are assumed to be of rectangular shape in this simulation. The locations of the obstacle are described by the coordinates of their vertices. The vertices of the obstacle1, obstacle2 and obstacle 3 are given table III. The optimal

TABLE II
Final poses

No.	*Position (m)*	*Heading (deg)*
1	(22, 40)	24
2	(17, 40)	120
3	(26, 40)	113
4	(10, 40)	40

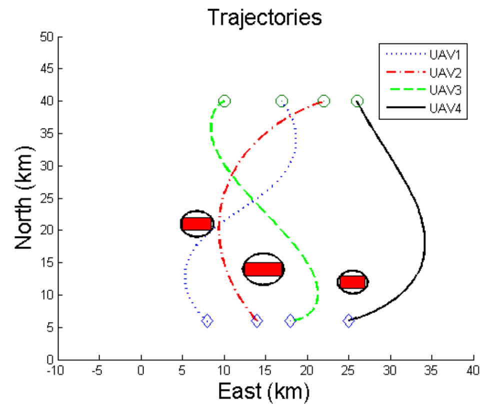

Fig. 4. Optimal PH paths, all equal in length and avoiding obstacles

TABLE III
Location of the known obstacles

Obstacles	*Vertex1*	*Vertex2*	*Vertex3*	*Vertex4*
Obstacle1	(5, 20)	(5,22)	(8.5, 22)	(8.5, 20)
Obstacle2	(12.5, 13)	(12.5, 15)	(17, 15)	(17, 13)
Obstacle3	(24, 11)	(24, 13)	(27, 13)	(27, 11)

allocation obtained is $[2, 1, 4, 3]$ which represent the first UAV is allocated to the $2nd$ target, the $2nd$ UAV is assigned to the $1st$ target, and so on. The optimal paths of the UAVs are shown in figure 4. As shown in the figure, there is neither inter-collision between UAVs, nor collision with obstacles. The curvature constraint is also satisfied. The length of all the paths are made equal to the longest path by further change of curvature of each path. In this simulation the lengths of each path comes out to be $[36.8, 37.3, 39.7, 36.2]$ meters.

In order to examine the performance of the optimal resource allocation, the optimal result is compared to an arbitrary allocation, $[3, 4, 2, 1]$. The paths for the allocation are shown figure 5. The performance improvement of the optimal allocation compare to the arbitrary allocation is 12.12%.

VI. Conclusions

In this paper, optimal resource allocation with 2D path planning algorithm (2D Path Planner) is proposed. In order to consider operational and physical constraints of the UAVs in the resource allocation design procedure, the proposed algorithm consists of two parts: cooperative path planning based on Pythagorean Hodograph quintic and optimal resource allocation using Evolutionary Game Particle Swarm Optimisation (EGPSO). The algorithm successfully calculated safe

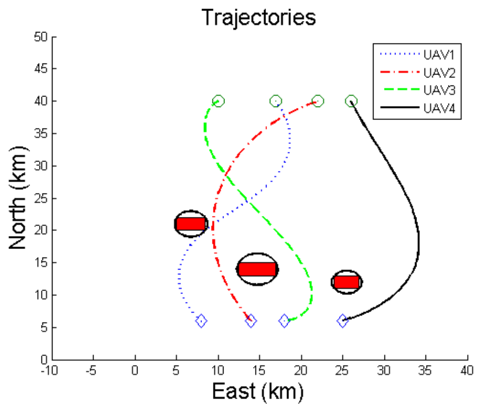

Fig. 5. PH paths for an arbitrary allocation, all equal in length and avoiding obstacles

and flyable paths (feasible paths) for all the UAVs in the group and optimally allocated the UAVs to a group of targets. The performance of the proposed algorithm is evaluated using numerical simulations.

REFERENCES

[1] A. Gibbons. *Algorithmic Graph Theory*. Cambridge University Press, 1999.

[2] D. Ahr. *Contributions to Multiple Postmen Problems*. PhD thesis, Heidelberg University, 2004.

[3] J.L. Gross and J. Yellen. *Handbook of Graph Theory*. CRC Press, 2003.

[4] T. Bektas. The multiple traveling salesman problem: An overview of formulations and solution procedures. *The International Journal of Management Science*, 34(3):209–219, 2006.

[5] J. Bellingham, M. Tillerson, A. Richards, and J. How. *Multi-Task Allocation and Path Planning for Cooperating UAVs, Cooperative Control: Models, Applications and Algorithms*. Kluwer Academic Publishers, 2003.

[6] P.R. Chandler, M. Pachter, D. Swaroop, J.M. Hwlett, S. Rasmussen, C. Schumacher, and K. Nygard. Compexity in uav cooperation control. In *American Control Conference*, American Control Conference, Anchorage, AK, 2002.

[7] J. Ryan, T. Bailey, J. Moore, and W. Carlton. Reactive tabu search in unmanned aerial reconnaissance simulations. In *30th Conference on Winter Simulation*, Washington, DC, 1998.

[8] A. Ahmadzadeh, G. Buchman, P. Cheng, A. Jadbabaie, J. Keller, V. Kumar, and G. Pappas. Cooperative control of UAVs for search and coverage. In *Conference on Unmanned Systems*, 2006.

[9] J.O. Royset and H. Sato. Route optimization for multiple searchers. *Naval Research Logistics*, 57:701717, 2010.

[10] N. Perrier, A. Langevin, and J.F. Campbell. A survey of models and algorithms for winter road maintenance. *Computers and Operational Research, Part IV: Vehicle Routing and Fleet Sizing for Plowing and Snow Disposal*, 34:258–294, 2007.

[11] K. Easton and J. Burdick. A coverage algorithm for multi-robot boundary inspection. In *IEEE International Conference on Robotics and Automation*, 2005.

[12] B. Alspach. Searching and sweeping graphs: a brief survey. *Matematiche (Catania)*, 59:5–37, 2006.

[13] T.D. PARSONS. *Pursuit-evasion in a graph*. Theory and Applications of Graphs. Springer-verlagf, Berlin, 1976.

[14] M. Shanmugavel, A. Tsourdos, B. A. White, and R. Zbikowski. Cooperative path planning of multiple uavs using dubins paths with clothoid arcs. *Control Engineering Practice*, 18(9):1084–1092, 2010.

[15] C. Leboucher, R. Chelouah, P. Siarry, and S. Le Ménec. A swarm intelligence method combined to evolutionary game theory applied to ressources allocation problem. In *International Conference on Swarm Intelligence*, 2010.

[16] M. Shanmugavel. *Path planning of multiple autonomous vehicles*. PhD thesis, Cranfield University, 2007.

[17] L.E. Dubins. On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3):497–516, 1957.

[18] J. Kennedy and R.C. Eberhart. A discrete binary version of the particle swarm algorithm. In *The 1997 IEEE International Conference on Systems, Man, and Cybernetics*, volume 5, pages 4104–4108, Orlando (USA), October 1997.

[19] B. Al-kazemy and C. K. Mohan. Multi-phase discrete particle swarm optimization. In *Fourth International Workshop on Frontiers in Evolutionary*, 2000.

[20] M. Clerc. Discrete particle swarm optimization, illustred by the travelling salesman problem. *New Optimization Techniques in Engineering*, 1:219–239, 2004.

[21] C.J. Liaoa, C.T. Tsengb, and P. Luarnb. A discrete version of particle swarm optimization for flowshop scheduling problems. *Computers & Operations Research*, 34:3099–3111, 2007.

[22] V. Miranda and N. Fonseca. Epso - evolutionary particle swarm optimization, a new algorithm with application in power systems. In *Transmission and Distribution Conference and Exhibition*, volume 2, pages 745–750, October 2002.

[23] C. Di Chio, P. Di Chio, and M. Giacobini. An evolutionary game-theoretical approach to particle swarm optimisation. In *Conference on Applications of evolutionary computing*, pages 575–584, Naples, 2008.

[24] W-B. Liu and X-J. Wang. An evolutionary game based particle swarm optimization algorithm. *Journal of Computational and Applied Mathematics*, 214(1):30–35, 2008.