

Research Article

A Virtual Motion Camouflage Approach for Cooperative Trajectory Planning of Multiple UCAVs

Xueqiang Gu, Lincheng Shen, Jing Chen, Yu Zhang, and Wanpeng Zhang

College of Mechatronic Engineering and Automation, National University of Defense Technology, Changsha, Hunan 410073, China

Correspondence should be addressed to Jing Chen; chenjing00100@sina.com

Received 22 August 2013; Revised 28 November 2013; Accepted 1 December 2013; Published 22 January 2014

Academic Editor: Xinjie Zhang

Copyright © 2014 Xueqiang Gu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper investigates cooperative trajectory planning of multiple unmanned combat aerial vehicles (multi-UCAV) in performing autonomous cooperative air-to-ground target attack missions. By integrating an approximate allowable attack region model, several constraint models, and a multicriterion objective function, the problem is formulated as a cooperative trajectory optimal control problem (CTOCP). Then, a virtual motion camouflage (VMC) for cooperative trajectory planning of multi-UCAV, combining with the differential flatness theory, Gauss pseudospectral method (GPM), and nonlinear programming, is designed to solve the CTOCP. In particular, the notion of the virtual time is introduced to the VMC problem formulation to handle the temporal cooperative constraints. The simulation experiments validate that the CTOCP can be effectively solved by the cooperative trajectory planning algorithm based on VMC which integrates the spatial and temporal constraints on the trajectory level, and the comparative experiments illustrate that VMC based algorithm is more efficient than GPM based direct collocation method in tackling the CTOCP.

1. Introduction

Nowadays, it is an active research area to perform autonomous cooperative air-to-ground target attack (CA/GTA) missions using multiple unmanned combat aerial vehicles (multi-UCAV) [1]. However, compared with single UCAV planning and coordinated formation control problems [2], new technical challenges in the CA/GTA missions are emerging. The cooperative trajectory planning is one of the key challenging technologies, due to its high dimensionality, severe equality and inequality constraints involved, and the requirement of spatial-temporal cooperation of multi-UCAV.

Recently, various algorithms have been developed to solve this cooperative trajectory planning problem [3, 4], including artificial neural network methods [5], sample-based planning methods [6], maneuver automation (MA) [7], and optimal control methods. There is no doubt that the optimal control theory is the most natural framework for this type of problem with dynamic constraints [8]. However, the rapid solution to optimal control problems (OCPs) for complicated nonlinear systems, such as UCAVs, is a challenging task [9].

Analytical solutions are seldom available or even possible. As a result, one usually resorts to numerical techniques [3]. The techniques can be classified into two general types, namely, indirect and direct methods. Indirect methods [10] solve the OCPs by formulating the first-order optimality conditions, applying Pontryagin's Maximum Principle and nonlinear programming (NLP) to tackle the resulting two-point boundary value problem numerically. Direct methods [11], on the other hand, are devoted to reduce the OCPs to finite-dimensional NLP problems by discretization and parameterization of subsets of the state and control vectors and then are solved by developed optimizers. As one of direct methods, Gauss pseudospectral method (GPM) is applied widely and efficiently in the trajectory optimization field [12–14], due to its advantages of fewer parameters and higher precision in the calculation procedure. However, the method to achieve the trajectory planning for a single vehicle or cooperative multivehicle required a high computational load. To reduce the computational complexity, the inverse dynamics methods [15, 16] and differential flatness theory (DFT) based methods [17–20] are introduced. Compared to pseudospectral methods, these methods can use any models

and any performance indexes and transform the OCP into a low dimensional NLP problem. However, they are still time-consuming. Particularly, when the number of the vehicles is too large, the computational cost will be unacceptable. For this purpose, the virtual motion camouflage (VMC) algorithm [21–23] is introduced into this work. Inspired by the biological motion known as the motion camouflage, the VMC algorithm can dramatically reduce the problem dimension and the computational cost. Some numerical simulations [22–24] suggest that its computational speed could be much faster than the pseudospectral methods. Motivated by these advantages, this paper employs virtual motion camouflage approach to develop cooperative trajectory planning algorithms.

For the cooperative trajectory planning of multi-UCAV, the objective is to generate dynamically feasible trajectories for UCAVs, which can guide them to the goals in the shortest time or distance, without collisions with obstacles or each other. To date, a number of theories and techniques have been developed to accomplish the cooperative tasks, including several collision avoidance techniques and time adjustment strategies [17, 25–30]. Bollino and Lewis [25] addressed the optimal path planning of UAVs in obstacle-rich environments and proposed a collision-free path planning for multi-UAV using optimal control techniques and pseudospectral methods. Kuwata and How [26] presented a cooperative distributed robust trajectory optimization approach, using RH-MILP with independent dynamics but coupled objectives and hard constraints. To improve the convergence rate, the virtual motion camouflage method was applied to the cooperative electronic combat air vehicles (ECAVs) [27] and unmanned aerial vehicle (UAV) formation control [28]. Reference [27] described how an interesting bioinspired motion strategy can be used to design real-time trajectories for cooperative electronic combat air vehicles. The constant speed motion camouflage law was developed to derive the feasible condition of a constant speed ECAVs and PT coherent mission, and its feasibility conditions were found. Reference [28] proposed a divide and conquer hierarchical approach in three levels to solve the UAV formation flight trajectory plan problem considering dynamics, state, and control variable inequality and equality constraints. These approaches mentioned above could generate collision-free trajectories, but the temporal cooperation was ignored. In contrast, Lian [17] introduced a differential flatness theory (DFT) based method to optimally formulate the cooperative path planning for multiagent dynamical systems considering spatial and temporal constraints. McLain and Beard [29] proposed the coordination variables (CV) and coordination function (CF) based strategy to achieve cooperative timing among teams of vehicles by coordinating the velocity and path length of each vehicle. But these approaches failed to deal with the dynamics constraints of vehicles, and the generated paths were not always flyable and smooth. Kaminer et al. [30] presented a general framework for coordinated control problem of multiple autonomous agents. On the basis of decoupling space and time in the problem formulation, it reduced the number of optimization parameters and made it easy to implement optimization in real time. However, the

time coordination lays in the path following and the design of control laws in path following algorithms was much more difficult. Although the previous investigations have described several valuable strategies in the cooperative path planning, these methods cannot tackle the point-to-region cooperative trajectory planning for CA/GTA missions directly, which needs to integrate both the spatial and temporal constraints on the trajectory level.

To address the problems mentioned previously, a novel cooperative trajectory planning method for multi-UCAV in performing the CA/GTA missions is presented in this paper. Firstly, some constraints including individual and cooperative constraints are modeled. Particularly, an approximate allowable attack region is built for the critical terminal constraints. Then, after the multicriterion objective function is constructed, the cooperative trajectory planning problem is formulated as a cooperative trajectory optimal control problem (CTOCP). Owing to the temporal constraints, a notion of the virtual time is introduced and the VMC problem is reformulated in the virtual time domain. Inspired by VMC, DFT, and GPM algorithms, a new cooperative trajectory planning algorithm in an optimal control framework is proposed. The proposed approach is demonstrated by two typical CA/GTA examples, and the comparative experiments between GPM and VMC are carried out. The results show that the proposed approach is feasible, effective, and efficient.

The rest of the paper is organized as follows. In Section 2, the integrated model is set up. In Section 3, the problem is formulated as a CTOCP followed by a multicriteria objective function. Section 4 develops a novel cooperative trajectory planning algorithm based on virtual motion camouflage in virtual time domain, which can solve the problem efficiently. Section 5 presents several numerical examples, and finally the conclusions and future works are outlined in Section 6.

2. Modeling

2.1. Aircraft Model. The kinematic and dynamic model of vehicles is needed in the cooperative trajectory planning of multi-UCAV. In this work, a team of N_v homogeneous UCAVs is considered, each of which is described by the kinematic and dynamic model according to a full-blown three-degree of freedom (3-DOF) model as follows [31]:

$$\begin{aligned}\dot{x} &= V \cos \gamma \cos \psi, & \dot{y} &= V \cos \gamma \sin \psi, & \dot{z} &= V \sin \gamma, \\ \dot{V} &= g(n_x - \sin \gamma), & \dot{\gamma} &= \frac{g}{V}(n_z \cos \mu - \cos \gamma), \\ \dot{\psi} &= \frac{g}{V \cos \gamma} n_z \sin \mu,\end{aligned}\tag{1}$$

where x , y , and z are the aircraft coordinates, that is, longitude, latitude, and height, V is the aircraft velocity, γ is the flight-path angle, ψ is the heading angle, μ is the roll angle, g is the gravity acceleration, and n_x and n_z are the longitudinal and normal components of the load factor, respectively.

For the optimal control problem, the state vector \mathbf{x} and control vector \mathbf{u} can be defined as

$$\mathbf{x} = \left\{ (x, y, z, V, \gamma, \psi) \mid \begin{array}{l} z_{\min} \leq z \leq z_{\max} \\ V_{\min} \leq V \leq V_{\max} \\ \gamma_{\min} \leq \gamma \leq \gamma_{\max} \\ \psi_{\min} \leq \psi \leq \psi_{\max} \end{array} \right\} \in \mathbf{X} \subset \mathbb{R}^6, \quad (2)$$

$$\mathbf{u} = \left\{ (\mu, n_x, n_z) \mid \begin{array}{l} \mu_{\min} \leq \mu \leq \mu_{\max} \\ n_{x,\min} \leq n_x \leq n_{x,\max} \\ n_{z,\min} \leq n_z \leq n_{z,\max} \end{array} \right\} \in \mathbf{U} \subset \mathbb{R}^3.$$

2.2. Allowable Attack Region Model. For the point-to-region trajectory planning problem, the allowable attack regions (AARs) of targets are defined as the areas where weapon delivery operations can be effectively performed by UCAVs. Therefore, in order to plan accurate and optimal attack trajectories, the AARs and the delivery parameters need to

be obtained. The AARs of the i th target TAR_i , denoted as $R(\text{TAR}_i)$, are such sets of all feasible release states that TAR_i can be effectively attacked whenever the aircraft belongs to those state sets. When n_v ($n_v \geq 2$) UCAVs are assigned to attack the same target TAR_i cooperatively, there will be n_v AARs in $R(\text{TAR}_i)$; that is, $R(\text{TAR}_i) = \bigcup_j^{n_v} \text{AAR}_j$. Each AAR_j can be formulated as an abstract 6-dimensional space [32]:

$$\text{AAR}_j = \{x, y, z, V, \gamma, \psi\} \subset \mathbb{R}^6. \quad (3)$$

Obviously, AAR_j is a high-dimensional nonlinear space, which is difficult to handle. By presetting an appropriate weapon release speed \widehat{V}_r and the flight-path angle $\widehat{\gamma}_r$ based on estimating the weapon impact effects and destruction requirements to the target and predetermining release heading $\widehat{\psi}_r$, AAR_j can be reduced to a 3-dimensional space as

$$\begin{aligned} & \text{LAR}_{\text{sp}}^j(\text{TAR}_i, [\widehat{V}_r, \widehat{\gamma}_r]) \\ &= \left\{ (x, y, z, \widehat{V}_r, \widehat{\gamma}_r, \widehat{\psi}_r) \mid \begin{array}{l} (x, y, z) \in \text{LAR}_{\text{sp}}^j(\text{TAR}_i, [\widehat{V}_r, \widehat{\gamma}_r, \widehat{\psi}_r]) \\ \widehat{\psi}_r = \text{azimuth of } \text{TAR}_i \text{ relative to position } (x, y, z) \end{array} \right\}. \end{aligned} \quad (4)$$

In the point-to-region trajectory planning, weapon delivery points (WDPT) as terminals of trajectories need to be included in the AARs, that is, meeting terminal constraints [33]. The formula can be denoted as $\text{WDPT}_j \in \text{AAR}_j$; that is,

$$\begin{aligned} |x_A^j - x_f^j| &\leq \Delta x, & |y_A^j - y_f^j| &\leq \Delta y, \\ |z_A^j - z_f^j| &\leq \Delta z, \end{aligned} \quad (5)$$

where (x_A^j, y_A^j, z_A^j) are the coordinates of the center of the AAR_j , (x_f^j, y_f^j, z_f^j) are the coordinates of WDPT_j , and $(\Delta x, \Delta y, \Delta z)$ are the thresholds of errors.

2.3. No-Fly Zone Model. In the battlefield environment, the no-fly zone (NFZ) is an area where vehicles are not permitted to fly over, due to the presence of military restrictions (e.g., armed enemies and the missile killing range), the physical obstacles (e.g., mountains and buildings within the natural and urban environment, resp.), and civil restrictions mainly due to safety reasons (e.g., densely populated areas and severe weather condition zones).

It is complex and unnecessary to describe a NFZ's exact shape and size. In this work, the p -norm method [34] is used to mathematically model the shapes of the NFZs, given by

$$h(x, y, z) = \left\| \left(\frac{x - x_c}{k_1 + b_s}, \frac{y - y_c}{k_2 + b_s}, \frac{z - z_c}{k_3 + b_s} \right) \right\|_p^p - k_4^p, \quad (6)$$

$$\|(x, y, z)\|_p = (|x|^p + |y|^p + |z|^p)^{1/p}, \quad p \in \mathbf{N}, \quad (7)$$

where (x_c, y_c, z_c) indicates the location of the geometric center of the NFZ, $h(x, y, z)$ is the distance between a point (x, y, z) and the boundary of the NFZ, while $k_1 \sim k_4$ are the constant parameters chosen to define the size of the NFZ, and b_s represents the width of a safe buffer which can effectively expand the NFZ boundary by an amount that accounts for the size of the vehicle and improve the robustness of plans. By varying parameters $(p, k_1 \sim k_4)$, one can easily model a number of generic shapes (see Figure 1). And any NFZ can be approximately modeled by fitting one of those shapes around it.

Accordingly, the above NFZs can be expressed by the following path constraints in the OCP formulation as

$$h_n(x, y, z) \geq 0, \quad n = 1, 2, \dots, N_{\text{NFZ}}, \quad (8)$$

where h_n represents the n th NFZ and N_{NFZ} is the total of NFZs. However, (8) is not well scaled. If a large p is chosen or the vehicle is far from the NFZ, $h_n(x, y, z)$ can produce very large numbers; hence, for computational efficiency, the path constraint is scaled by the natural logarithm function as follows:

$$\begin{aligned} h_n(x, y, z) = \ln \left(\frac{1}{(k_4^n)^{p_n}} \left(\left(\frac{x - x_c^n}{k_1^n + b_s^n} \right)^{p_n} + \left(\frac{y - y_c^n}{k_2^n + b_s^n} \right)^{p_n} \right. \right. \\ \left. \left. + \left(\frac{z - z_c^n}{k_3^n + b_s^n} \right)^{p_n} \right) \right) \geq 0. \end{aligned} \quad (9)$$

2.4. Cooperative Constraint Model. In order to make the UCAVs arrive in the AARs of targets safely in an expected

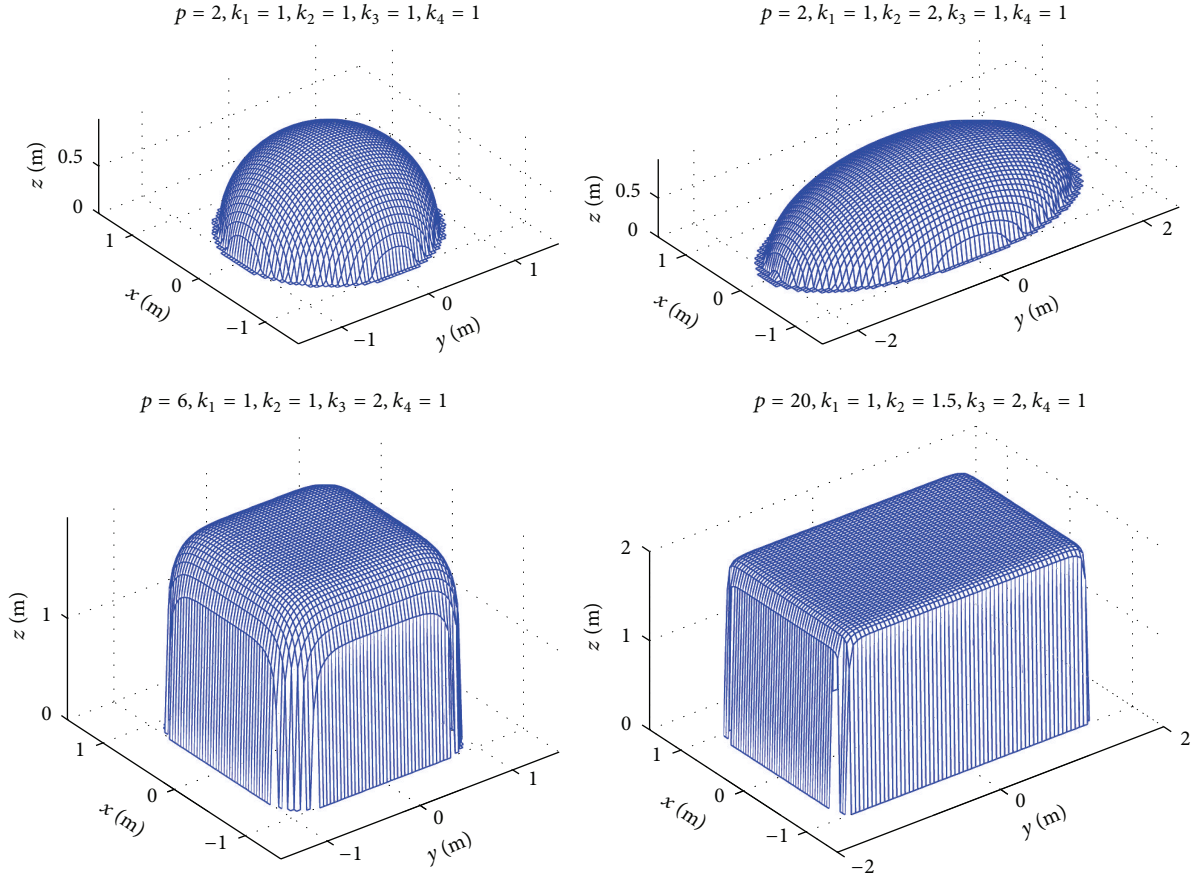


FIGURE 1: Several shapes generated from (6).

time sequence, cooperative trajectories should satisfy the following constraints.

2.4.1. Spatial Constraints. During the mission, multi-UCAV should maintain a safe distance to guarantee them not to collide with each other. The model can be denoted as

$$\|\rho^j(t_i) - \rho^k(t_i)\|_2 \geq \max(d_{\text{safe}}^j, d_{\text{safe}}^k), \quad (10)$$

$$\forall j \neq k, \quad j, k = 1, 2, \dots, N_v,$$

where $\rho^j(t_i) = \{x^j(t_i), y^j(t_i), z^j(t_i)\}$ is the spatial position of each UCAV_j at the time t_i , d_{safe}^j is the predetermined minimum safety radius of UCAV_j (based on the vehicle size, maneuver agility, etc.), and N_v is the total of UCAVs.

2.4.2. Temporal Constraints. The temporal constraints include the simultaneous arrival constraint and sequencing constraint, which are described as

$$t_f^j + \Delta_{jk} - t_f^k \leq 0, \quad \forall j \neq k, \quad j, k = 1, 2, \dots, N_v, \quad (11)$$

where t_f^j is the terminal time of each UCAV_j and Δ_{jk} is the arrival interval between UCAV_j and UCAV_k. For the simultaneous arrival constraint, $\Delta = 0$, and for the

sequencing constraint, Δ is the plus fixed quantity. Equation (11) is of the general form $f_T(t_f^1, \dots, t_f^{N_v}) \leq 0$.

3. Problem Formulation

3.1. Objective Function. The objective function of the entire team is a sum of each individual cost, which is a multicriterion objective function, and each criterion could compete with each other as follows:

$$J = \min \sum_{j=1}^{N_v} J^j = \min \sum_{j=1}^{N_v} (w_t^j J_t^j + w_p^j J_{\text{prd}}^j + w_r^j J_{\text{rob}}^j), \quad (12)$$

and the criterions can be, respectively, defined as

$$J_t^j = \int_t dt = t_f^j - t_0^j, \quad (13)$$

$$J_{\text{prd}}^j = \int_t \text{PRD}^j(t) = \int_t \left(1 - \prod_{r=1}^{n_r} (1 - P_d^j(t, r)) \right), \quad (14)$$

$$J_{\text{rob}}^j = \int_t R^j(t) = \int_t \sum_{n=1}^{N_{\text{NEZ}}} w_r^{n,j} (e^{e^{-h_n}} - 1), \quad (15)$$

where t_0^j and t_f^j denote the initial and terminal time of UCAV_j. The first term, J_t^j , denotes the total fight time of UCAV_j. The second term, J_{prd}^j , denotes the detection-probability of the n_r -radar system to UCAV_j at the time t , which describes the threat risk criterion of UCAV_j as follows:

$$\text{PRD}^j(t) = 1 - \prod_{r=1}^{n_r} (1 - P_d^j(t, r)), \quad (16)$$

where $P_d^j(t, r)$ is the model of the radar detection probability between the trajectory point of UCAV_j at the time t and the r th radar [35]. The final term, J_{rob}^j , guarantees the robustness of the vehicle collision avoidance against the model imprecision, disturbing in the battlefield and operating errors that could otherwise cause vehicles to collide with obstacles or other vehicles. The robustness cost is defined as

$$R^j(t) = \sum_{n=1}^{N_{\text{NFZ}}} w_r^{n,j} (e^{e^{-h_n}} - 1), \quad (17)$$

where $w_r^{n,j}$ are weights that are incremented to increase the maneuver robustness when the trajectory of a vehicle passes in close proximity to an obstacle or another vehicle, which can be chosen by the method presented by Hurni et al. [36].

As can be seen, the individual objective function J^j is defined by the weighted sum of the three separate running cost terms with appropriate weighting factors w_t^j , w_p^j , and w_r^j . The three criteria (13)–(15) represent different physical meanings, and they are difficult to be compared directly. Hence the selection of these weighting factors is a skilled technique. In the experimentations, the multi-objective fuzzy optimization method proposed by Wang et al. [37] is employed. The method includes two main steps: (a) normalizing each single objective function and (b) solving the membership function about each criterion by using fuzzy distribution function.

3.2. Cooperative Trajectory Planning Problem Formulation. After establishing the above models, the cooperative trajectory planning problem is formulated in this section. The problem under consideration is a cooperative scheme, consisting of N_v homogeneous UCAVs. According to the system equations (1), the general form of the system for UCAV_j is given by

$$\dot{\mathbf{x}}^j(t) = f^j[\mathbf{x}^j(t), \mathbf{u}^j(t), t], \quad j = 1, \dots, N_v. \quad (18)$$

As stated above, to obtain optimal or suboptimal cooperative trajectories, the cooperative trajectory planning for the CA/GTA missions can be formulated as a cooperative trajectory optimal control problem (CTOCP).

Problem 1 (CTOCP). Find the trajectories, which drive the system from given initial conditions to desired final conditions over time horizons $[t_0, t_f]$, while the cooperative objective function is minimized as

$$\begin{aligned} \min \quad J &= \sum_{j=1}^{N_v} J^j(\mathbf{x}, \mathbf{u}) \\ &= \sum_{j=1}^{N_v} \left[w_t^j (t_f^j - t_0^j) + w_p^j \int_{t_0^j}^{t_f^j} (1 - P_d^j(t, r)) dt \right. \\ &\quad \left. + w_r^j \int_{t_0^j}^{t_f^j} \sum_{n=1}^{N_{\text{NFZ}}} w_r^{n,j} (e^{e^{-h_n}} - 1) dt \right], \end{aligned} \quad (19)$$

subject to the system equation (18) and the boundary constraints (i.e., the initial and terminal states (5))

$$\Phi[\mathbf{x}^j(t_0^j), \mathbf{u}^j(t_0^j), t_0^j, \mathbf{x}^j(t_f^j), \mathbf{u}^j(t_f^j), t_f^j] = 0, \quad (20)$$

$\forall j = 1, 2, \dots, N_v,$

and several inequality and equality constraints, the individual and cooperative constraints, including the state and control vectors (2) and (9)–(11), are denoted as

$$\begin{aligned} \delta^j[\mathbf{x}^j(t), \mathbf{u}^j(t), t] &\leq 0, \\ \varphi^j[\mathbf{x}^j(t), \mathbf{u}^j(t), \mathbf{x}^k(t), \mathbf{u}^k(t), t] &\leq 0, \quad (21) \\ f_T^j(t_f^j, t_f^k) &\leq 0, \quad \forall j \neq k, \quad j, k = 1, 2, \dots, N_v. \end{aligned}$$

4. Virtual Motion Camouflage Based Cooperative Trajectory Planning

To tackle the cooperative trajectory planning for CA/GTA missions, an efficient virtual motion camouflage based planning method is proposed for numerically solving the CTOCP, which combines with the benefits of several other classical trajectory generation techniques, including DFT, GPM, and NLP. Correspondingly, this task contains three primary steps. The first is to determine outputs by VMC and DFT in the virtual time domain, such that the dynamics system can be mapped to a lower-dimensional output space. Then, GPM is used to discretize the outputs. Finally, the system is transformed into a NLP problem and the sequential quadratic programming (SQP) is used to solve the NLP to minimize the cost and subject to constraints in output space.

4.1. Virtual Motion Camouflage and Problem Formulation in the Output Space

4.1.1. Virtual Motion Camouflage. Inspired by mating hoverflies, Srinivasan and Davey [38] described a new form of the stealth strategy which is the so-called motion camouflage (MC). MC involves two moving objects, a prey and an aggressor, and occurs when the aggressor conceals its motion

(i.e., maintains a constant bearing in the prey's coordinates), except the inevitable size change, as viewed by the moving prey. Here the MC concept is introduced into the optimal trajectory planning and is called virtual motion camouflage (VMC) [21–23] because the “prey” is a virtual one. In the VMC framework, the aggressor trajectory (i.e., the aggressor position vector) $\mathbf{p}_a(t) \in \mathbb{R}^{n_a}$ is confined by the virtual prey trajectory (VPT) $\mathbf{p}_p(t) \in \mathbb{R}^{n_a}$, the selected reference point $\mathbf{p}_{\text{ref}}(t) \in \mathbb{R}^{n_a}$, and the path control parameter (PCP) $v(t) \in \mathbb{R}$ as follows:

$$\mathbf{p}_a(t) = \mathbf{p}_{\text{ref}} + v(t) (\mathbf{p}_p(t) - \mathbf{p}_{\text{ref}}). \quad (22)$$

The reference point is considered fixed over time, so the derivatives of the trajectory can be described as

$$\begin{aligned} \dot{\mathbf{p}}_a(t) &= \dot{v}(t) (\mathbf{p}_p(t) - \mathbf{p}_{\text{ref}}) + v(t) \dot{\mathbf{p}}_p(t), \\ \ddot{\mathbf{p}}_a(t) &= \ddot{v}(t) (\mathbf{p}_p(t) - \mathbf{p}_{\text{ref}}) + v(t) \ddot{\mathbf{p}}_p(t) + 2\dot{v}(t) \dot{\mathbf{p}}_p(t). \end{aligned} \quad (23)$$

From [23], it can be known that when the proposed VMC suboptimal trajectory design method is used, the complexity of the problem can be reduced by $(n_{sv} + m_{cv})$ times from the collocation method or n_{sv} times from the differential inclusion, where n_{sv} and m_{cv} are the numbers of the state variables and the control variables. In VMC framework, if the dimension of the position vector $\mathbf{x}_a(t)$ is one (i.e., $n_a = 1$), the dimension of the problem is the same as that of the differential inclusion method. Otherwise, when $n_a > 1$, the reference point can be fixed or regarded as a parameter to be optimized, whereas VPT can be predefined or predetermined by the initial and terminal conditions of the aggressor trajectory. Thus, the dimension of the problem is lowered from n_a to one, such that the solving speed can increase.

4.1.2. VMC Problem Formulation in Virtual Time Domain.

Obviously, the time factor of trajectories is an argument of the state and control. To deal with temporal constraints and achieve the time cooperation of multi-UCAV, the time along the trajectories should be considered separately. Therefore, the independent intermediate variable (called virtual time here, $\bar{t} \in [0, 1]$) is introduced and described as

$$\bar{t} = \frac{(t - t_0)}{(t_f - t_0)}, \quad (24)$$

such that the trajectories can be generated in the virtual time domain from $\bar{t}_0 = 0$ to $\bar{t}_f = 1$.

For UCAVs, which are required to take off at the same time, it can be assumed that the initial time of all UCAVs is zero ($t_0 = 0$). Thus, the terminal time t_f can be written as $t_f = t/\bar{t}$. That is, t_f denotes the ratio between the true time variable and the newly defined virtual time variable. To coordinate the arrival time of all UCAVs, the terminal time can be defined as an argument to be optimized in the dynamical system, designated as T_f . Then, the following relationship

between the virtual time domain and true time domain can be obtained for an arbitrary variable χ as follows:

$$\begin{aligned} \chi'(\bar{t}) &= \frac{d\chi(t)}{d\bar{t}} = \frac{T_f d\chi(t)}{dt} = T_f \dot{\chi}(t), \\ \chi''(\bar{t}) &= \frac{d\chi'(\bar{t})}{d\bar{t}} = \frac{T_f d\chi'(\bar{t})}{dt} = \frac{T_f d(T_f \dot{\chi}(t))}{dt} = T_f^2 \ddot{\chi}(t). \end{aligned} \quad (25)$$

Particularly, the derivative of the speed variable can be denoted as

$$\begin{aligned} V(\bar{t}) &= \sqrt{(x')^2 + (y')^2 + (z')^2} \\ &= T_f \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2} = T_f V(t), \end{aligned} \quad (26)$$

where the superscript “ $'$ ” represents the derivative with respect to the virtual time.

According to (18) and (24), the system equations can be rearranged as

$$\begin{aligned} \mathbf{x}'^j(\bar{t}) &= T_f \dot{\mathbf{x}}^j(t) = T_f f^j[\mathbf{x}^j(t), \mathbf{u}^j(t), t] \\ &= g^j[\mathbf{x}^j(\bar{t}), \mathbf{u}^j(\bar{t}), T_f^j], \\ T_f'^j &= 0, \quad j = 1, \dots, N_v. \end{aligned} \quad (27)$$

Hence, the Problem 1 can be reformulated in virtual time domain (VTD) as follows.

Problem 2 (CTOCP-VTD). Minimize the cooperative objective function (19) of all UCAVs represented with respect to the new independent variable \bar{t} as

$$\begin{aligned} \min \quad J &= \sum_{j=1}^{N_v} J^j(\mathbf{x}, \mathbf{u}, T_f^j) \\ &= \sum_{j=1}^{N_v} \left[w_t^j T_f^j + w_p^j \int_0^1 \text{PRD}^j(\bar{t}) d\bar{t} \right. \\ &\quad \left. + w_r^j \int_0^1 R^j(\bar{t}) d\bar{t} \right], \end{aligned} \quad (28)$$

subject to the system equations (27), and the boundary constraints (20), written as

$$\phi[\mathbf{x}^j(0), \mathbf{u}^j(0), \mathbf{x}^j(1), \mathbf{u}^j(1), T_f^j] = 0, \quad (29)$$

Meanwhile satisfying the inequality and equality constraints (21) and additional temporal constraints as follows:

$$\begin{aligned} \delta^j[\mathbf{x}^j(\bar{t}), \mathbf{u}^j(\bar{t}), T_f^j] &\leq 0, \\ \varphi^j[\mathbf{x}^j(\bar{t}), \mathbf{u}^j(\bar{t}), \mathbf{x}^k(\bar{t}), \mathbf{u}^k(\bar{t}), T_f^j, T_f^k] &\leq 0, \\ f_T^j(t_f^j, t_f^k) &\leq 0, \quad \forall j \neq k, \quad j, k = 1, 2, \dots, N_v. \end{aligned} \quad (30)$$

From (1), it can be clearly known that the dynamics system under consideration is a higher-dimensional space system. Due to the complexity in solving this type of the problem, the VMC based method is used to lower the system dimension and the state and control vectors can be related by the differential flatness theory based method [39]. In this system, the state vector $\mathbf{x} \in \mathbb{R}^6$ can be separated into two parts: the trajectory position (e.g., spatial trajectory) $\boldsymbol{\rho}_a = [x, y, z] \in \mathbb{R}^3$ and the corresponding state rate $\boldsymbol{\rho}_r = [V, \psi, \gamma] \in \mathbb{R}^3$, by which the trajectory position in virtual time domain and the terminal time T_f can be defined as the flat output vector

$$\boldsymbol{\vartheta} = [\boldsymbol{\rho}_a(\bar{t}), T_f] = [x(\bar{t}), y(\bar{t}), z(\bar{t}), T_f]^T. \quad (31)$$

Thus, the original state vector \mathbf{x} and control vector \mathbf{u} can be recovered from the flat outputs and their derivatives as

$$\begin{aligned} \mathbf{x} &= \alpha(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}', \boldsymbol{\vartheta}'') = \alpha(\boldsymbol{\rho}_a, \boldsymbol{\rho}_a', \boldsymbol{\rho}_a'', T_f), \\ \mathbf{u} &= \beta(\boldsymbol{\vartheta}, \boldsymbol{\vartheta}', \boldsymbol{\vartheta}'') = \beta(\boldsymbol{\rho}_a, \boldsymbol{\rho}_a', \boldsymbol{\rho}_a'', T_f). \end{aligned} \quad (32)$$

According to (1), the relation of the state vector and control vector in the VTD is described as follows:

$$\begin{aligned} V &= \sqrt{(x')^2 + (y')^2 + (z')^2}, \\ \psi &= \arctan\left(\frac{y'}{x'}\right), \\ \gamma &= \arctan\left(\frac{z'}{\sqrt{(x')^2 + (y')^2}}\right), \\ n_x &= \frac{V'}{(T_f g)} + \sin \gamma, \\ n_z &= \frac{\sqrt{(V\gamma' + T_f^2 g \cos \gamma)^2 + (V\psi' \cos \gamma)^2}}{T_f g}, \\ \mu &= \arctan\left(\frac{V\psi' \cos \gamma}{(V\gamma' + T_f^2 g \cos \gamma)}\right). \end{aligned} \quad (33)$$

In the VMC framework, the trajectory position $\boldsymbol{\rho}_a$ can be defined as the aggressor position vector. As described above, the state and the control variables of the dynamics system can be recovered from the flat outputs. Hence, according to (22), these variables are also functions of the PCP, the predefined virtual prey motion, the reference point, and their corresponding derivatives. Once the predefined virtual prey motion and the reference point are selected, the system of cooperative trajectory planning of UCAVs, including the objective function and the constraints, is mapped to a lower-dimensional output space (here, the dimension is one). Obviously, the dynamic constraints of this system (27) can be automatically satisfied. Therefore, the Problem 2 can be redefined as follows.

Problem 3 (CTOCP-VTD in the output space). Consider

$$\begin{aligned} \min \quad J(\mathbf{v}) &= \sum_{j=1}^{N_v} J^j(\mathbf{v}^j) \\ &= \sum_{j=1}^{N_v} \left[\left(v^j, v'^j, \dots, T_f^j \right) \right. \\ &\quad \left. + \int_0^1 f^j(v^j, v'^j, \dots, \bar{t}) d\bar{t} \right] \\ \text{s.t.} \quad \phi^j[v^j(0), v^j(1), T_f^j] &= 0, \\ \delta^j(v^j, v'^j, \dots) &\leq 0, \\ \varphi^j(v^j, v'^j, \dots, v^k, v'^k, \dots) &\leq 0, \\ f_T^j(t_f^j, t_f^k) &\leq 0, \quad \forall j \neq k, \quad j, k = 1, 2, \dots, N_v. \end{aligned} \quad (34)$$

4.2. Collocation Using Gauss Pseudospectral Method. In order to solve the Problem 3 through a NLP algorithm, the PCP history $v(\bar{t})$ should be discretized into $c = 0, 1, \dots, N_c$, $N_c + 1$ nodes, with $v_0 = v(\bar{t}_0)$ and $v_{N_c+1} = v(\bar{t}_{N_c+1})$. In this paper, GPM is selected as the discretization method [12], which is an orthogonal collection method where the collocation points are the Legendre-Gauss (LG) points. Similar to other pseudospectral methods, a finite basis of global interpolating polynomials is used to approximate the state and control space at a set of discretization nodes in the GPM and then the optimal control equations can be transformed into nonlinear algebra equations, such that the OCP can be solved by a NLP algorithm.

The standard interval considered here is denoted as $\tau \in [-1, 1]$. By using a linear transformation, the virtual time \bar{t} can be expressed as a function of τ via

$$\bar{t} = \frac{[(\bar{t}_f - \bar{t}_0)\tau + (\bar{t}_f + \bar{t}_0)]}{2} = \frac{(\tau + 1)}{2}. \quad (35)$$

Let $\tau_1 < \dots < \tau_{N_c}$ be collocation points in $(-1, 1)$ and $\tau_0 = -1, \tau_f = 1$. The time history of the approximated PCP at the N_c LG points is given by

$$v(\tau) \approx \bar{v}(\tau) = \sum_{c=0}^{N_c} v(\tau_c) L_c(\tau), \quad (36)$$

where $v(\tau_c)$ is an approximant at the node τ_c and the base function $L_c(\tau)$ is the Lagrange interpolating polynomial of degree N_c , expressed as

$$L_c(\tau) = \prod_{l=0, l \neq c}^{N_c} \frac{\tau - \tau_l}{\tau_c - \tau_l}, \quad c = 0, 1, \dots, N_c, \quad (37)$$

satisfying the isolation property

$$L_c(\tau_l) = \delta_{cl} = \begin{cases} 1, & c = l, \\ 0, & c \neq l. \end{cases} \quad (38)$$

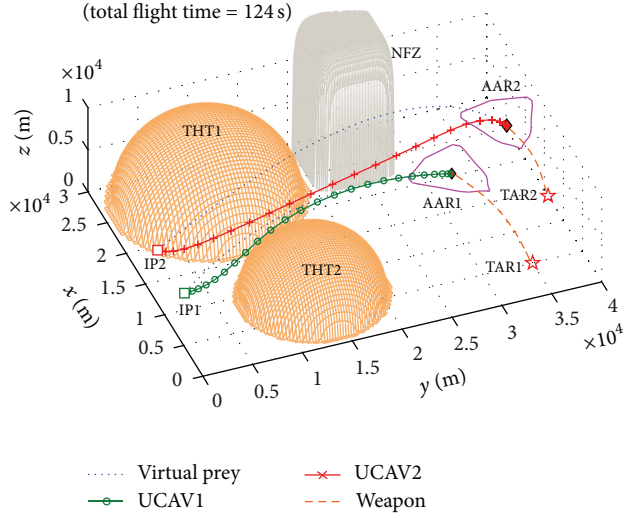


FIGURE 2: Two collision-free UCAV trajectories with arriving simultaneously.

And the endpoint can be approximated according to the following formula:

$$v(\tau_f) \approx \bar{v}(\tau_f) = \bar{v}(\tau_0) + \frac{\bar{t}_f - \bar{t}_0}{2} \times \sum_{c=1}^{N_c} w_c f[\bar{v}(\tau_c), \tau_c], \quad (39)$$

where τ_c ($c = 1, \dots, N_c$) are the LG points and the variables ω_c are the LG weights given by

$$w_c = \int_{-1}^1 L_c(\tau) d\tau. \quad (40)$$

The derivative of the state can be approximated as the exact derivative of the interpolating polynomial. Evaluating the derivative at the LG points results in

$$\begin{aligned} v'(\tau_l) &\approx \bar{v}'(\tau_l) = \sum_{c=0}^{N_c} v(\tau_c) L'_c(\tau_l) \\ &= \sum_{c=0}^{N_c} v(\tau_c) D_{lc}, \quad l = 1, \dots, N_c, \end{aligned} \quad (41)$$

where $\mathbf{D} = [D_{lc}] \in \mathbb{R}^{N_c \times (N_c+1)}$ is a differential matrix and can be offline determined as

$$\begin{aligned} D_{lc} &= \sum_{k_1=0}^{N_c} \frac{\prod_{k_2=0, k_2 \neq c, k_1}^{N_c} (\tau_l - \tau_{k_2})}{\prod_{k_2=0, k_2 \neq c}^{N_c} (\tau_c - \tau_{k_2})}, \\ l &= 1, \dots, N_c, \quad c = 0, \dots, N_c. \end{aligned} \quad (42)$$

The discretized states and controls satisfying the vehicle dynamics can be computed by ensuring that equation

$$\sum_{c=0}^{N_c} v(\tau_c) D_{lc} - \frac{\bar{t}_f - \bar{t}_0}{2} f[v(\tau_l), \tau_l] = 0, \quad l = 1, \dots, N_c. \quad (43)$$

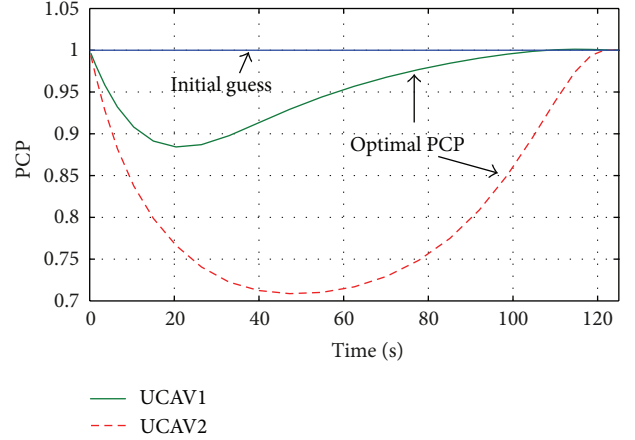


FIGURE 3: PCP histories of two UCAVs.

As described above, the continuous OCP can be transformed to the discretized NLP problem. The parameters to be optimized are the PCP nodes $\mathbf{v} = [v_c]_{c=0,1,\dots,N_c+1}$. When the prey motion $\mathbf{p}_{p,c}$ is equal to the aggressor motion $\mathbf{p}_{a,c}$ for $c = 0, N_c + 1$, one can assume that $v_0 = v_{N_c} = 1$, so the boundary conditions are no longer considered. And the discretized NLP problem can be denoted as follows.

Problem 4 (CTP-NLP). Its standard form is denoted as

$$\begin{aligned} \min J(\mathbf{v}) &= \sum_{j=1}^{N_v} J^j(\mathbf{v}^j) \\ &= \sum_{j=1}^{N_v} \left[\Phi^j[\mathbf{v}^j, T_f^j] \right. \\ &\quad \left. + \frac{\bar{t}_f - \bar{t}_0}{2} \sum_{c=1}^{N_c} f^j(\mathbf{v}^j, T_f^j) \omega_c \right] \\ \text{s.t. } \delta_l^j[\mathbf{v}^j] &\leq 0, \quad l = 1, \dots, N_c, \\ \varphi^j[\mathbf{v}^j, \mathbf{v}^k] &\leq 0, \\ f_T^j(t_f^j, t_f^k) &\leq 0, \quad \forall j \neq k, \quad j, k = 1, 2, \dots, N_v, \end{aligned} \quad (44)$$

where $\mathbf{v}^j = [v_c^j]_{c=1,\dots,N_c}$ is the vector form of the discretized PCP.

Then the resulting Problem 4 can be solved through well-developed algorithms, such as the SQP algorithm. In this paper, TOMLAB/SNOPT software toolbox is chosen due to its advantages on solution effectiveness for the large-scale NLP problems [40].

4.3. Convergence and Initial Guess. As analyzed in [41], the GPM exhibits global convergence properties in many

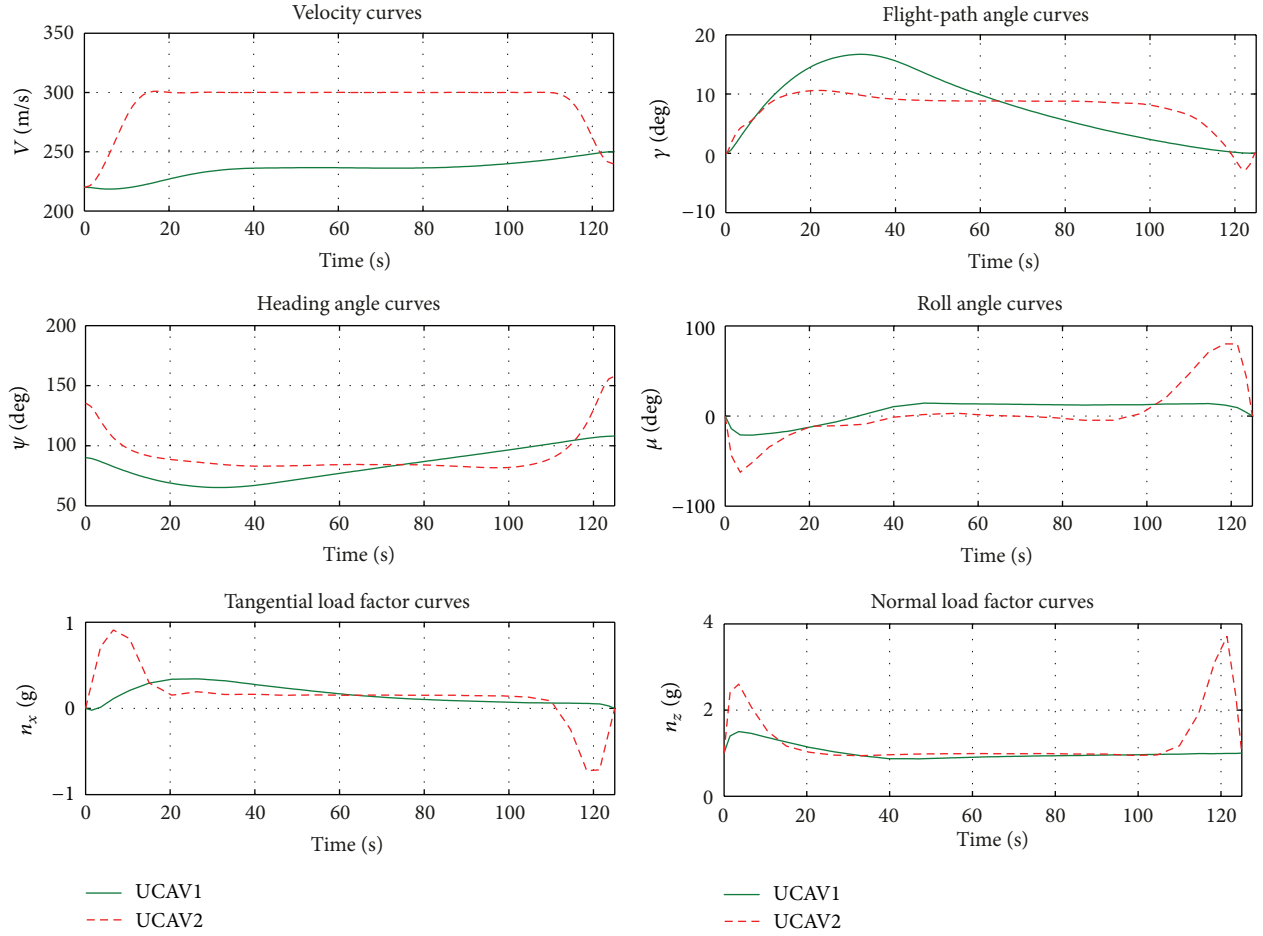


FIGURE 4: Example 1—state and control time histories of two UCAVs.

applications. In addition, the convergence of the virtual motion camouflage has been proved by Xu et al. [21, 23], in which [23] showed that if the problem was only one-dimensional, the VMC based trajectory design method could converge to the optimal solution, but when the problem was multi-dimensional and highly nonlinear, the result might be suboptimal in the full solution space, and [21] showed that the VMC framework together with the pseudospectral discretization method could converge to the optimal solution.

By introducing elastic programming concept and some advances in both mathematical programming techniques and pseudospectral methods, Ross and Gong [42] designed a guess-free trajectory optimization algorithm. These advances have the effect of eliminating the guessing problem in the trajectory optimization, which has been used in the current GPM algorithm. In the proposed VMC method, the motion of the virtual prey and a reference point are guessed. However, both of them have direct physical relation to the problem. Typically the virtual prey motion is selected according to the boundary conditions and a proper guess is not a difficult task [23]. In this paper, the trivial guess for the reference point can be the initial point (IP), and an initial guess of $v_c = 1$ is chosen for $c = 1, \dots, N_c$. The virtual prey motion is defined

as a 2-order curve, determined by the initial and terminal conditions.

5. Numerical Examples

The basic ideas presented in this paper are illustrated in the following three examples. The specific vehicle platform used in simulations is the Storm shadow UCAV. Its relevant parameters are all taken from [43], summarized in Table 1.

The experimental test environment is a rectangle area of $30 \times 40 \text{ km}^2$, as shown in Figure 2, where the NFZ1 caused by the severe weather condition is modeled as a cuboid with infinite altitude, and the NFZs caused by enemy threats are modeled as two hemispheres, denoted as THT1 and THT2. According to Figure 1, the shape parameters can be set as follows: for the NFZ1, $k_1 = k_2 = k_3 = k_4 = 1$, $p = 2$, and the length of a side is set as 8 km; for THT1 and THT2, $k_1 = k_2 = k_4 = 1$, $k_3 = 2$, $p = 6$, and the radius of the threats can be set as 7 km and 9 km, respectively. In order to avoid contact, a safe buffer b_s ($b_s = 200 \text{ m}$) is added around the outside edge of each obstacle/threat boundary. All the results presented below are generated using TOMLAB/SNOPT software toolbox on a

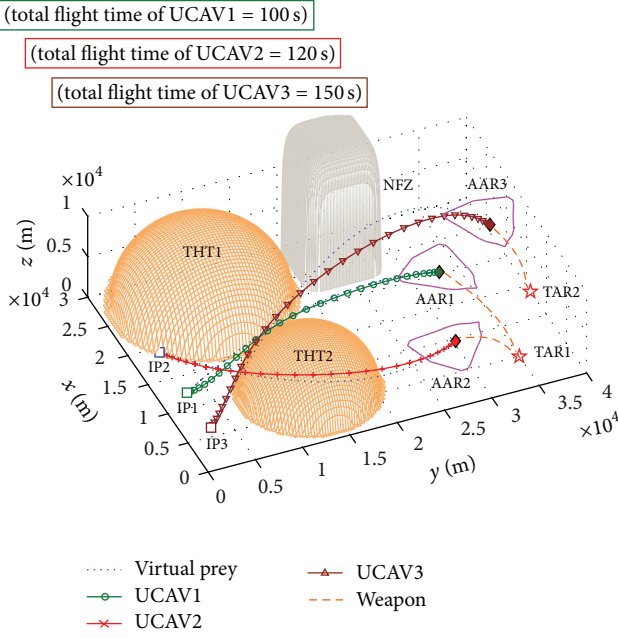


FIGURE 5: Three collision-free UCAV trajectories with arriving in sequence.

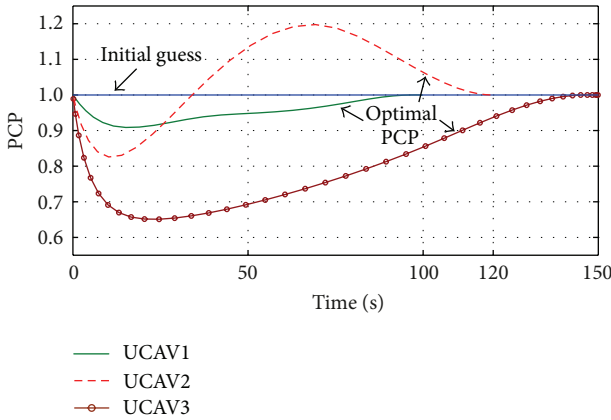


FIGURE 6: Path control parameter histories of three UCAVs.

2.4 GHz CPU and 2 G RAM computer running Windows XP and MATLAB R2009b. Table 2 summarizes the parameters used in the algorithm. To simplify the problem, the same weight coefficients for the objective function of each UCAV are set, and it is assumed that the target assignment is already completed before.

5.1. Example 1: Cooperative Trajectories of Two UCAVs Arriving Simultaneously. In this example, two UCAVs cooperatively attack two stationary ground targets while avoiding a series of static obstacles/threats detected and collision with each other en route and meeting aircraft dynamical constraints, especially simultaneous arrival constraint. UCAV1 and UCAV2 start at each initial point, that is, IP1 (10, 2, 2) km and IP2 (17, 2, 2) km. Then they fly into the AARs of

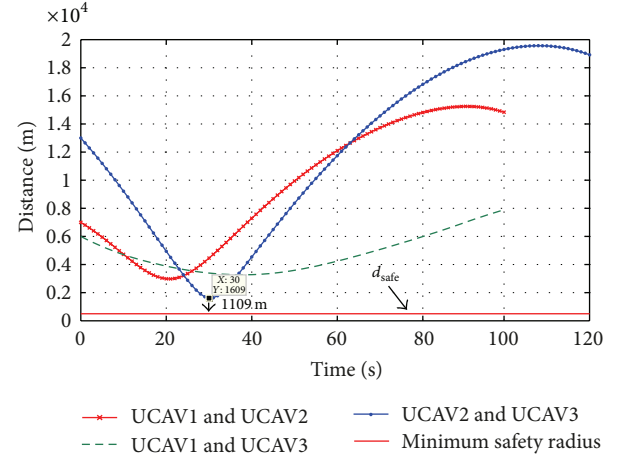


FIGURE 7: Distance between each pair of UCAVs.

two targets, TAR1 (4, 34, 0) km and TAR2 (13, 40, 0) km, respectively. The initial remaining three states and control inputs of each UCAV are preset as $\rho_{r,0}^1 = (V_0^1, \gamma_0^1, \psi_0^1) = (220 \text{ m/s}, 0^\circ, 90^\circ)$, $\rho_{r,0}^2 = (V_0^2, \gamma_0^2, \psi_0^2) = (220 \text{ m/s}, 0^\circ, 140^\circ)$, and $\mathbf{u}_0^1 = \mathbf{u}_0^2 = (\mu_0, n_{x,0}, n_{z,0}) = (0^\circ, 0 \text{ g}, 1 \text{ g})$, and the terminal remaining three states of each UCAV are predetermined as $\rho_{r,f}^1 = (\hat{V}_f^1, \hat{\gamma}_f^1, \hat{\psi}_f^1) = (250 \text{ m/s}, 0^\circ, 110^\circ)$ and $\rho_{r,f}^2 = (\hat{V}_f^2, \hat{\gamma}_f^2, \hat{\psi}_f^2) = (250 \text{ m/s}, 0^\circ, 160^\circ)$. In VMC framework, the reference point ρ_{ref} for each UCAV is presumed as its IP, and the virtual prey path is predefined as a 2-order curve, determined by the initial and terminal conditions.

Figure 2 shows the overall collision-free attack trajectories of two UCAVs, whose total flight time is equal, equaling to 124 s. It means that two UCAVs arrive simultaneously. As can be seen from Figure 2, along the resulting smooth trajectories, the UCAVs can avoid collision with all obstacles, threats, and the other en route and then successfully fly into the AARs (fuchsia areas) to perform weapon delivery missions. In addition, the approximate weapon trajectories are drawn to illustrate the attack process. The PCP histories of two UCAVs are shown in Figure 3, which are smooth and lie in the range from 0 to 1. And the time histories of the UCAVs' states (V, γ, ψ) and control inputs (μ, n_x, n_z) are shown in Figure 4. It is clear that the constraints on these variables, especially the cooperative constraints, are all satisfied (see Tables 1 and 2), which means that the resulting trajectories are feasible and safe.

5.2. Example 2: Cooperative Trajectories of Multi-UCAV Arriving in Sequence. In this example, three UCAVs attack two stationary ground targets cooperatively. The only additional requirement is that the UCAVs arrive at their AARs in sequence rather than simultaneously, and the intervals between UCAVs are set to 20 s and 30 s, respectively, denoted as $\Delta_{12} = 20 \text{ s}$ and $\Delta_{23} = 30 \text{ s}$. The coordinates of the initial points IP1, IP2 and two targets TAR1, TAR2 are the same with those in Example 1, and the third initial point is

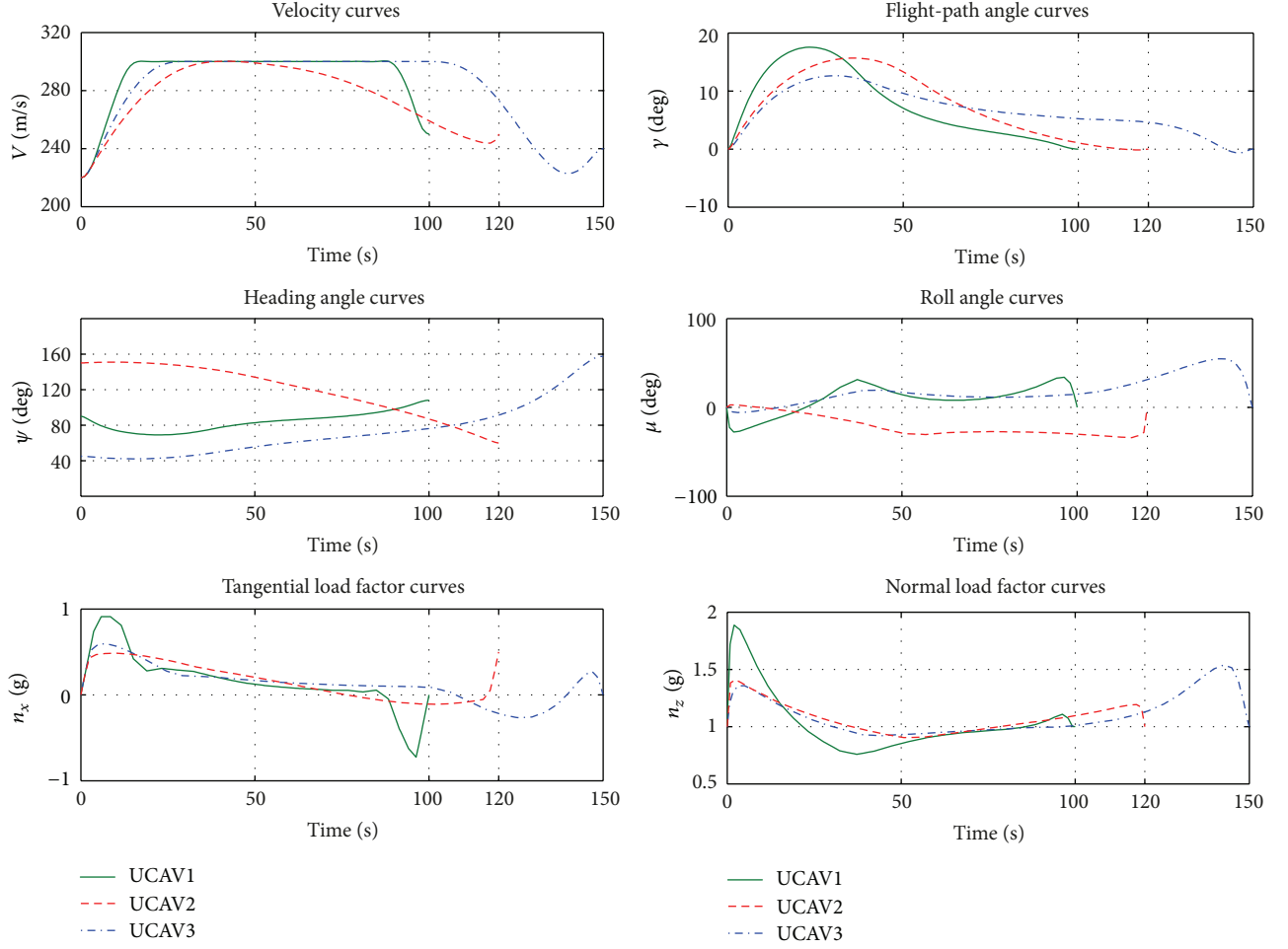


FIGURE 8: Example 2—state and control time histories of three UCAVs.

IP3 (4, 2, 2) km. The result of the previously finished target assignment is that the UCAV1 and UCAV2 attack TARI and UCAV3 attacks the other one. The initial remaining three states and control inputs of the UCAV3 are preset as $\mathbf{p}_{r,0}^3 = (V_0^3, \gamma_0^3, \psi_0^3) = (220 \text{ m/s}, 0^\circ, 45^\circ)$ and $\mathbf{u}_0^3 = (\mu_0, n_{x,0}, n_{z,0}) = (0^\circ, 0 \text{ g}, 1 \text{ g})$, and the terminal remaining three states of the UCAV3 are predetermined as $\mathbf{p}_{r,f}^3 = (\hat{V}_f^3, \hat{\gamma}_f^3, \hat{\psi}_f^3) = (220 \text{ m/s}, 0^\circ, 160^\circ)$. In VMC framework, the reference point and the virtual prey path are predetermined, using the same strategy as the one in Example 1.

The overall collision-free attack trajectories of multi-UCAV are shown in Figure 5, and the total flight time of three UCAVs is 100 s, 120 s, and 150 s, respectively. It can be clearly demonstrated that UCAVs can avoid all obstacles or threats and successfully fly into the AARs in sequence to perform the weapon delivery. The PCP histories of three UCAVs are shown in Figure 6. And Figure 7 shows the distance between each pair of UCAVs. From it, one can find that the minimum distance between each UCAV is more than the minimal safety radius of d_{safe} ($d_{\text{safe}} = 500 \text{ m}$); that is, the UCAVs can avoid collision with each other en route. Figure 8 shows the time

TABLE 1: State and control constraints of UCAVs.

Item	Minimum value	Maximum value
Flight altitude (m)	$z_{\min} = 200$	$z_{\max} = 8\,000$
Airspeed (m/s)	$V_{\min} = 60$	$V_{\max} = 300$
Flight-path angle (deg)	$\gamma_{\min} = -89$	$\gamma_{\max} = 89$
Roll angle (deg)	$\mu_{\min} = -80$	$\mu_{\max} = 80$
Tangential load factor (g)	$n_{x,\min} = -0.725$	$n_{x,\max} = 0.91$
Normal load factor (g)	$n_{z,\min} = -3.2$	$n_{z,\max} = 8$
Roll angle rate (deg/s)	$\dot{\mu}_{\min} = -30$	$\dot{\mu}_{\max} = 30$
Rate of change of tangential load factor (g/s)	$\dot{n}_{x,\min} = -0.2$	$\dot{n}_{x,\max} = 0.2$
Rate of change of normal load factor (g/s)	$\dot{n}_{z,\min} = -2$	$\dot{n}_{z,\max} = 2$

histories of the UCAVs' states (V, γ, ψ) and control inputs (μ, n_x, n_z). Obviously, the resulting trajectories are feasible and safe, since all the constraints listed in Tables 1 and 2 are satisfied.

TABLE 2: Parameter values for the algorithm.

Item	Parameter	Value
Minimum safety radius (m)	d_{safe}	500
Weighting factors of the objective function	w_t	0.3
	w_p	0.4
	w_r	0.3
	w_r^1	0.5
Weighting factors of the robust subobjective	w_r^2	0.3
	w_r^3	0.2
Safe buffer of NFZs (m)	b_s	200

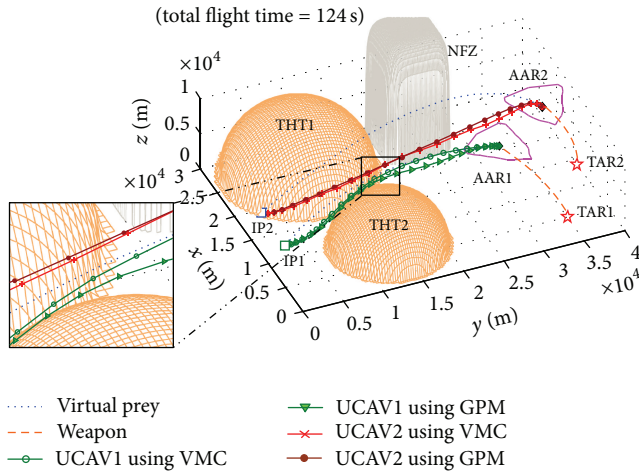


FIGURE 9: Comparison of two collision-free UCAV trajectories with arriving simultaneously.

In order to validate the convergence rate of VMC based algorithm in cooperative trajectory planning for multi-UCAV further, the comparative experiments are carried out with different numbers of UCAVs (N_v is set as 1, 2, 3, 5, 8, and 10, resp.) and the same number of nodes ($N_c = 20$). The performance comparison of multi-UCAV is summarized in Table 3. As can be seen clearly, along with the increase of the number of UCAVs, the average planning time is rising at a slow rate, which is acceptable for the preplanning applications. Particularly, for one UCAV, the average planning time is 2.21 s. When the number of UCAVs increases to 10, the average planning time increases by only more than quadruple of that for one UCAV, which is 9.62 s.

5.3. Comparison between VMC Based Method and GPM Based Direct Collocation Method. To further evaluate the performance of the proposed algorithm, a recently developed optimal trajectory generation method, GPM based direct collocation method, is used for comparison, which is basically converting the original optimal control problem to a NLP problem via the GPM directly. In the following comparative experiments, Example 1 is simulated again using the two algorithms, respectively, performed on the same desktop with different numbers of nodes (N_c is set as 10, 15, 20, and 30, resp.). All the results are presented below.

TABLE 3: Performance comparison with the increase of the number of UCAVs.

Number of UCAVs (N_v)	Number of optimization variables	Average planning time (s)	Ratio of the planning time (%)
1	$(20 \times 1 + 1) \times 1 = 21$	2.21	—
2	$(20 \times 1 + 1) \times 2 = 42$	2.83	128.05
3	$(20 \times 1 + 1) \times 3 = 63$	3.32	150.23
5	$(20 \times 1 + 1) \times 5 = 105$	4.93	223.08
8	$(20 \times 1 + 1) \times 8 = 168$	7.70	348.42
10	$(20 \times 1 + 1) \times 10 = 210$	9.62	435.29

Figures 9 and 10 show the comparative solutions attained by using the GPM and VMC based methods, respectively. It can be seen that the trajectories and the state and control time histories of these two algorithms are similar. The detailed performance comparison is summarized in Table 4. With comparison, when the number of nodes increases, the number of optimization variables in GPM is much more than VMC. As a result, the computational speed of VMC is more than an order of magnitude faster than GPM, while the optimization performance has only small loss.

In order to compare the convergence rate and analyze the impacts of the number of UCAVs on the computation time between VMC and GPM further, the comparative experiment is run multiple times with different numbers of UCAVs (N_v is set as an integer from 1 to 10, resp.) and the same number of nodes ($N_c = 20$). Figure 11 shows the change of the average planning time required to solve the optimization problem as the number of UCAVs increases. It can be seen that the increase of the number of UCAVs results in an exponential increase of the required computation time for GPM. In contrast, the computation time for VMC based algorithm increases slowly. When the number of UCAVs increases to 10, the number of the optimization variables for GPM is up to 1810. However, for VMC, the number is only 210. Since the required computation time strongly depends on the number of optimization variables, the computational time required by VMC is more than an order of magnitude less than that required by GPM. Maybe it can be concluded that the larger the number of UCAVs is, the more obvious the advantage of VMC to GPM is.

6. Conclusions

This paper is devoted to explore the cooperative collision-free trajectory planning for multiple UCAVs in performing the CA/GTA missions. The main contributions of the paper are as follows. Firstly, the cooperative trajectory planning problem under consideration is mathematically formulated as a cooperative trajectory optimal control problem (CTOCP). In order to consider the weapon delivery constraints, an

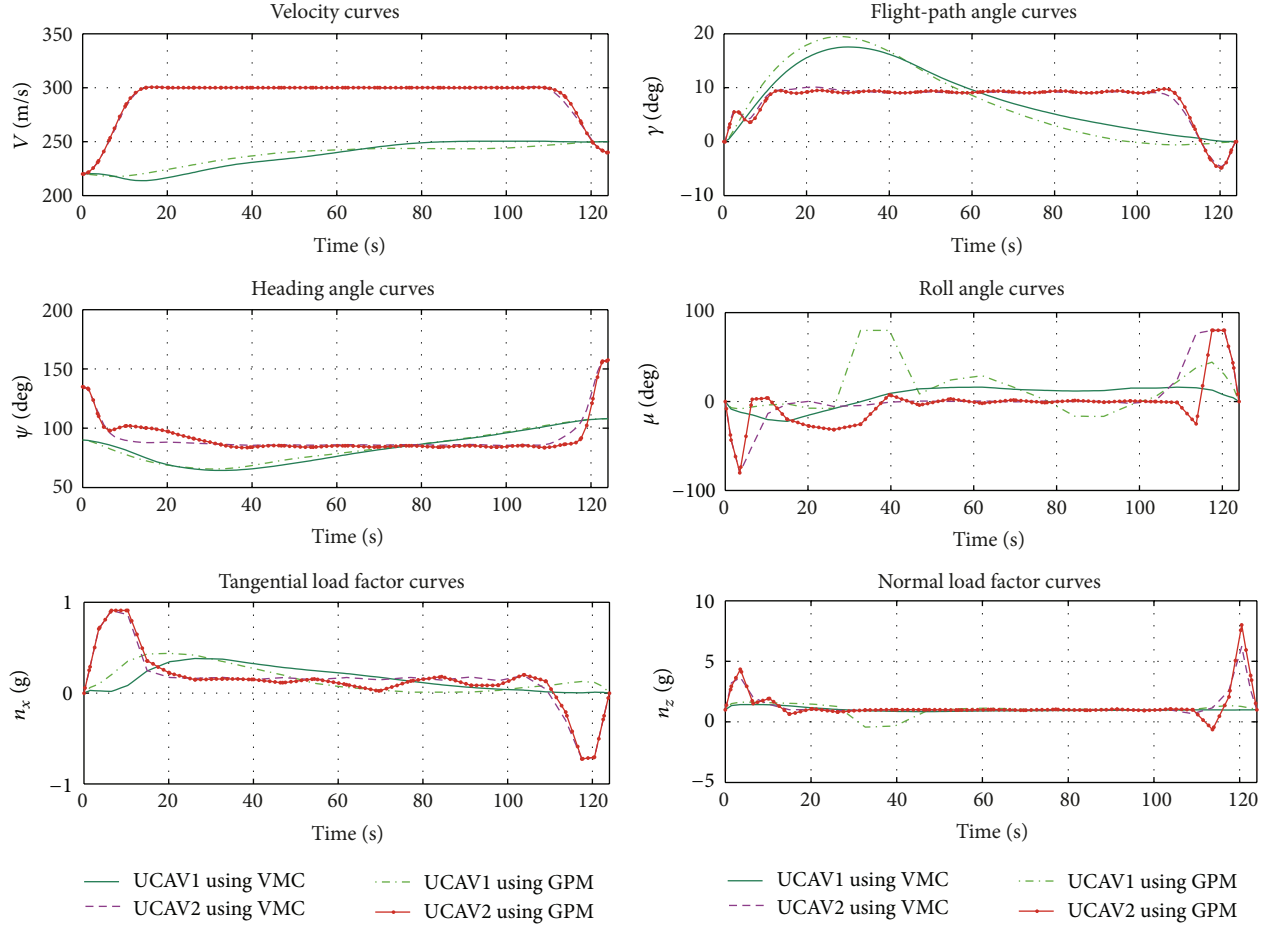


FIGURE 10: Comparison of state and control time histories.

TABLE 4: Comparison between GPM and VMC for Example 1.

Method	N_c	Number of optimization variables	Objective function value	Average planning time (s)	Ratio of the planning time (%)
GPM	10	$(10 \times 9 + 1) \times 2 = 182$	1.776	12.31	—
	15	$(15 \times 9 + 1) \times 2 = 272$	1.731	16.28	—
	20	$(20 \times 9 + 1) \times 2 = 362$	1.698	20.31	—
	30	$(30 \times 9 + 1) \times 2 = 542$	1.705	28.67	—
VMC	10	$(10 \times 1 + 1) \times 2 = 22$	1.851	2.23	18.12
	15	$(15 \times 1 + 1) \times 2 = 32$	1.840	2.46	15.11
	20	$(20 \times 1 + 1) \times 2 = 42$	1.769	2.83	13.93
	30	$(30 \times 1 + 1) \times 2 = 62$	1.731	3.18	11.09

approximate allowable attack region model is proposed and integrated into the problem formulation. Secondly, a particular virtual motion camouflage approach combining with differential flatness theory, Gauss pseudospectral method, and nonlinear programming is used to develop the trajectory planner for solving CTOCP. The advantages of this method are that it can automatically satisfy all boundary conditions and use any models and any performance indexes, does not require numerical integration of differentiation of the state

equations, and transforms the OCP into a NLP problem of very low dimension. Finally, in order to handle the temporal constraints, the notion of the virtual time is introduced into the virtual motion camouflage approach to realize the spatial-temporal cooperation.

The proposed approach is validated by numerical examples. The results show that the approach is able to generate both feasible and near-optimal attack trajectories with meeting the spatial and temporal constraints very efficiently.

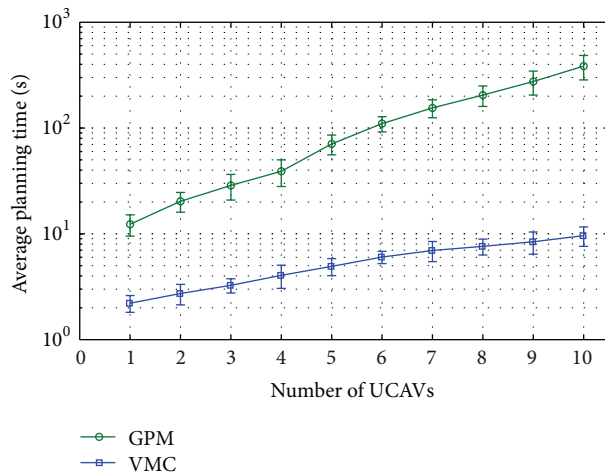


FIGURE 11: Comparison between GPM and VMC with the increase of the number of UCAVs.

Moreover, the convergence rate and average planning time of the method and optimality of the generated trajectories are evaluated via a detailed comparison with GPM based direct collocation method. The results show that the computational speed of virtual motion camouflage approach is more than an order of magnitude faster than GPM, at small loss of optimality.

For the future work, we will analyze some uncertain factors in the true battlefield environment and carry out the research on the real-time cooperative trajectory planning. For the presence of a larger number of UCAVs, we will make further efforts to exploit more efficient trajectory planning algorithm and improve the time cooperative strategy. Moreover, we will try to study another important aspect about how to plan in the opposability battlefield environment.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

Great thanks are due to the editor for helpful comments and the reviewers for their valuable suggestions and comments on the revision of the paper. This work was supported by the Science and Technology on Electro-Optic Control Laboratory of China (no. 20125188006).

References

- [1] R. M. Murray, "Recent research in cooperative control of multi-vehicle systems," *Journal of Dynamic Systems, Measurement and Control*, vol. 129, no. 5, pp. 571–583, 2007.
- [2] M. Fu and J. Jiao, "A hybrid approach for coordinated formation control of multiple surface vessels," *Mathematical Problems in Engineering*, vol. 2013, Article ID 794284, 8 pages, 2013.
- [3] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of Guidance, Control, and Dynamics*, vol. 21, no. 2, pp. 193–207, 1998.
- [4] C. Goerzen, Z. Kong, and B. Mettler, "A survey of motion planning algorithms from the perspective of autonomous UAV guidance," *Journal of Intelligent and Robotic Systems*, vol. 57, no. 1–4, pp. 65–100, 2010.
- [5] J. F. Horn, E. M. Schmidt, B. R. Geiger, and M. P. DeAngelo, "Neural network-based trajectory optimization for unmanned aerial vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 2, pp. 548–562, 2012.
- [6] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.
- [7] P. Narayan, P. Meyer, and D. Campbell, "Embedding human expert cognition into autonomous UAS trajectory planning," *IEEE Transaction on Cybernetics*, vol. 43, no. 2, pp. 530–543, 2013.
- [8] Q. Gong, L. R. Lewis, and I. M. Ross, "Pseudospectral motion planning for autonomous vehicles," *Journal of Guidance, Control, and Dynamics*, vol. 32, no. 3, pp. 1039–1045, 2009.
- [9] H. Liu, S. Chen, L. Shen et al., "An integrated multi-criterion hp-Adaptive pseudospectral method for direct optimal control problems solving," *Mathematical Problems in Engineering*, vol. 2012, Article ID 760890, 22 pages, 2012.
- [10] K. Yang, S. Sukkarieh, and Y. Kang, "Adaptive nonlinear model predictive path tracking control for a fixed-wing unmanned aerial vehicle," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Chicago, Ill, USA, August 2009.
- [11] O. A. Yakimenko, Y. Xu, and G. Basset, "Computing short-time aircraft maneuvers using direct methods," in *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Honolulu, Hawaii, USA, August 2008.
- [12] D. Benson, *A Gauss pseudospectral transcription for optimal control [Dissertation for the doctoral degree]*, Massachusetts Institute of Technology, Cambridge, Mass, USA, 2005.
- [13] M. Soler, A. Olivares, and E. Staffetti, "Hybrid optimal control approach to commercial aircraft trajectory planning," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 3, pp. 985–991, 2010.
- [14] J. Li, H. An, H. Zhu et al., "Geometric pseudospectral method on SE(3) for rigid-body dynamics with application to aircraft," *Mathematical Problems in Engineering*, vol. 2013, Article ID 595086, 16 pages, 2013.
- [15] A. Verma and J. L. Junkins, "Inverse dynamics approach for real-time determination of feasible aircraft reference trajectories," in *Proceedings of the 24th Atmospheric Flight Mechanics Conference*, 1999.
- [16] R. G. Drury, "Performance of NLP algorithms with inverse dynamics for near-real time trajectory generation," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Portland, Ore, USA, 2011.
- [17] F.-L. Lian, "Cooperative path planning of dynamical multi-agent systems using differential flatness approach," *International Journal of Control, Automation and Systems*, vol. 6, no. 3, pp. 401–412, 2008.
- [18] A. Chamseddine, Y. Zhang, C. A. Rabbath et al., "Trajectory planning and peplanning strategies applied to a quadrotor unmanned aerial vehicle," *Journal of Guidance, Control, and Dynamics*, vol. 35, no. 5, pp. 1667–1671, 2012.

- [19] X. Gu, Y. Zhang, S. Chen et al., "Cooperative trajectory planning for multi-UCAV performing air-to-ground target attack missions," *Advanced Materials Research*, vol. 718-720, pp. 1329–1334, 2013.
- [20] X. Gu, Y. Zhang, J. Chen et al., "Real-time cooperative trajectory planning using differential flatness approach and B-splines," *Applied Mechanics and Materials*, vol. 333-335, pp. 1338–1343, 2013.
- [21] Y. Xu, "Analytical solutions to spacecraft formation-flying guidance using virtual motion camouflage," *Journal of Guidance, Control, and Dynamics*, vol. 33, no. 5, pp. 1376–1386, 2010.
- [22] Y. Xu and G. Basset, "Sequential virtual motion camouflage method for nonlinear constrained optimal trajectory control," *Automatica*, vol. 48, no. 7, pp. 1273–1285, 2012.
- [23] Y. Xu, "Virtual motion camouflage and suboptimal trajectory design," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, pp. 555–571, August 2007.
- [24] Y. Xu and G. Basset, "Pre and post optimality checking of the virtual motion camouflage based nonlinear constrained subspace optimal control," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Chicago, Ill, USA, August 2009.
- [25] K. P. Bollino and L. R. Lewis, "Collision-free multi-UAV optimal path planning and cooperative control for tactical applications," in *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit*, Honolulu, Hawaii, USA, August 2008.
- [26] Y. Kuwata and J. P. How Sr., "Cooperative distributed robust trajectory optimization using receding horizon MILP," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 2, pp. 423–431, 2011.
- [27] Y. Xu and G. Basset, "Virtual motion camouflage based phantom track generation through cooperative electronic combat air vehicles," *Automatica*, vol. 46, no. 9, pp. 1454–1461, 2010.
- [28] Y. Xu, M. Xin, and J. Wang, "Unmanned aerial vehicle formation flight via a hierarchical cooperative control approach," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Portland, Ore, USA, 2011.
- [29] T. W. McLain and R. W. Beard, "Coordination variables, coordination functions, and cooperative-timing missions," *Journal of Guidance, Control, and Dynamics*, vol. 28, no. 1, pp. 150–161, 2005.
- [30] I. Kamlner, O. Yakimenko, V. Dobrokhodov et al., "Coordinated path following for time-critical missions of multiple UAVs via \mathcal{L}^1 adaptive output feedback controllers," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, pp. 915–948, August 2007.
- [31] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation*, John Wiley & Sons, New York, NY, USA, 2003.
- [32] Y. Zhang, J. Chen, and L. Shen, "Hybrid hierarchical trajectory planning for a fixed-wing UCAV performing air-to-surface multi-target attack," *Journal of Systems Engineering and Electronics*, vol. 23, no. 4, pp. 256–264, 2012.
- [33] X. Gu, Y. Zhang, N. Wang et al., "Robust multi-objective optimization based on NSGA-II for UCAV weapon delivery," in *Proceedings of the 4th International Conference on Intelligent Human-Machine Systems and Cybernetics*, 2012.
- [34] L. P. R. Lewis, *Rapid motion planning and autonomous obstacle avoidance for unmanned vehicles [M.S. thesis]*, Naval Postgraduate School, Monterey, Calif, USA, 2006.
- [35] E. Besada-Portas, L. de la Torre, J. M. de la Cruz, and B. de Andrés-Toro, "Evolutionary trajectory planner for multiple UAVs in realistic scenarios," *IEEE Transactions on Robotics*, vol. 26, no. 4, pp. 619–634, 2010.
- [36] M. A. Hurni, P. Sekhavat, M. Karpenko, and I. M. Ross, "A pseudospectral optimal motion planner for autonomous unmanned vehicles," in *Proceedings of the American Control Conference (ACC '10)*, pp. 1591–1598, Baltimore, Md, USA, July 2010.
- [37] Y. Wang, T. Wei, and X. Qu, "Study of multi-objective fuzzy optimization for path planning," *Chinese Journal of Aeronautics*, vol. 25, no. 1, pp. 51–56, 2012.
- [38] M. V. Srinivasan and M. Davey, "Strategies for active camouflage of motion," *Proceedings of the Royal Society B*, vol. 259, no. 1354, pp. 19–25, 1995.
- [39] M. Fliess, J. Levine, P. Martin, and P. Rouchon, "Flatness and defect of non-linear systems: introductory theory and examples," *International Journal of Control*, vol. 61, no. 6, pp. 1327–1361, 1995.
- [40] K. Holmström, A. O. Göran, and M. M. Edvall, *User's Guide For TOMLAB /SNOPT*, Department of Mathematics and Physics, Mälardalen University, Västerås, Sweden, 2008.
- [41] S. Chen, H. Liu, J. Chen et al., "Penetration trajectory planning based on radar tracking features for UAV," *Aircraft Engineering and Aerospace Technology*, vol. 85, no. 1, pp. 62–71, 2013.
- [42] I. M. Ross and Q. Gong, "Guess-free trajectory optimization," in *Proceedings of the AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, Honolulu, Hawaii, USA, August 2008.
- [43] "Storm shadow UCAV performance," 2001, <http://www.aerospacweb.org/design/ucav/main.shtml>.

