IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received April 22, 2018, accepted June 5, 2018, date of publication June 13, 2018, date of current version July 6, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2846769

A New Method for the Optimal Control Problem of Path Planning for Unmanned Ground Systems

JIE LIU[1], WEI HAN[1], CHUN LIU[2], AND HAIJUN PENG[3]
[1]Naval Aeronautical and Astronautical University, Yantai 264001, China
[2]650 Aircraft Design Institute of AVIC Hongdu, Nanchang 330024, China
[3]Dalian University of Technology, Dalian 116024, China

Corresponding author: Jie Liu (liuyexiaobao@163.com)

ABSTRACT To model the optimal control problem of path planning for unmanned ground systems (UGSs), the motion and boundary constraints are described first by using the mathematical model proposed in this paper, and the time-energy performance indicators are described by the Bolza cost function. Since the traditional symplectic algorithm hardly solves the problem with uncertain time, the pseudospectral method, almost the only way to solve the optimal control problem of path planning while the rapid path planning is difficult to achieve, is prone to the phenomenon named "Curse of Dimensionality" with increasing the number of discrete points. The symplectic pseudospectral method for improving the efficiency and the precision of the calculation, based on the symplectic theory, third kind of generation function and pseudospectral method is first proposed in this paper. Furthermore, the one-sided approximation is designed, and the one-sided symplectic pseudospectral (OSSP) algorithm is established to solve the model introduced in this paper. Finally, the experiments are conducted using the OSSP method and the pseudospectral method, respectively, to verify the feasibility and the efficiency of the method. The results show that the OSSP is the method with the highest accuracy, efficiency, and good stability to solve the optimal control problem of path planning for UGS, and it has great maneuverability and feasibility for practical application.

INDEX TERMS Optimization, motion control, nonlinear control systems, autonomous vehicles.

I. INTRODUCTION

Recently, the Unmanned Ground Systems (UGS), including unmanned vehicles and robots, have been widely used in the aerospace, military, civil and other fields, and more and more research on UGS is being performed. The path planning and control technologies are the key questions of the UGS, and it mainly comprises scene modeling, obstacle avoidance, optimal path and control problems. Generally, there are four main types of algorithms that are widely used for the UGS path planning problem, namely the search algorithm based on nodes, the potential field method, the intelligent algorithm and the optimal control method.

The main idea of the search algorithm, based on nodes, is simple: first, the environment map is divided into several small areas, setting a rule to find some key nodes within the feasible region and then connect the nodes to generate the feasible path. It mainly includes Visible Graph, Voronoi Diagram [1]–[3], A$*$ algorithm [4]–[7], algorithm based on tree theory [8], [9], bionic algorithm [10], [11], etc. The algorithms are highly efficient, but it's easy to ignore the nonholonomic motion constraints, and even ignore the posture information of the initial and final position.

The environment map is viewed as physical potential field in the Potential field method, the final position provides the gravitational field and the obstacles provide the repulsive force field, and the path can be generated using the mechanics theory in [12]–[14]. The method can achieve on-line planning with high efficiency, but the design of the appropriate gravitational and repulsion field is a hard work, and it's easy to fall into local optimal solution, and even ignores the nonholonomic motion constraints too.

With the development of artificial intelligence, the intelligent algorithm has been applied to path planning [15]. A good path planning model requires a lot of training with large path and obstacles data, such as reinforcement learning [16] and Support Vector Machine (SVM) [17], etc. This kind of

algorithm is more self-learning and intelligent, but there is a high demand for samples.

The above three methods, with different advantages and disadvantages, are mainly focused on how to solve avoiding collision problems and shorten the total distance as far as possible, and the path planning is considered separately from the corresponding control issues. However, in many cases, the shortest path is not optimal without considering nonholonomic motion and control constraints. Accordingly, we adopted the optimal control technique to solve the above problems in this paper.

The optimal control technique uses a mathematical model to describe the path planning problem and constraints, and gets the solution to minimize the objective function subjected to the constraints. The method takes the control variables into consideration, thus the optimal path and corresponding control law can be obtained. And it satisfies the nonholonomic constraints, the shortest time, as well as the minimum energy requirements. Although the method is highly precise, the main problem is hard to solve, and the computational efficiency is poor.

The methods for solving the optimal control problem can be divided into two types: the indirect and direct methods. The indirect method transforms the original problem into a Hamilton boundary value problem by using the variational method, the Pontryagin's maximum principle and the Lagrangian function in [18]–[22]. This method can achieve the local optimal solution with high precision, but the costates are so sensitive to the initial value that it is difficult to be applied in engineering practice. The direct method, based on the discretion of state and control variables, transforms the continuous optimization problem into a nonlinear programming problem. With the development of computational science, the direct method has gradually become the mainstream method of solving optimal control problems, such as the Pseudospectral method. However, the computational efficiency is still hard to meet the needs of path planning, and the increase of distribution points can easily cause the ''Curse of Dimensionality''. Gong *et al.* [23] and Lewis *et al.* [24] described a path planning problem with constrained nonlinear optimal control problem, and obtained the optimal path using the Pseudospectral method based on the DIDO toolbox. Zeng *et al.* [25] divided the path planning process into two stages: the shortest distance and the shortest time stage, and solved the path planning problem for a multi-robot using the harmony search algorithm.

Based on the characteristics of the direct and indirect methods, a high computationally efficient algorithm is considered by combining the direct and indirect method. The four kinds of generating functions proposed by Peng *et al.* [26]–[28], based on the Symplectic theory and the variational method, laid the foundation for the Symplectic algorithm. The Symplectic algorithm, combined the First and the Second Kind of Generation Function with the Pseudospectral method in [29]–[31], has obvious advantages of accuracy, convergence rate, computational efficiency for the energy optimal problem. Additionally, based on the Symplectic theory introduced, the characteristics of the original system will not be destroyed after discretization while other methods are hard to guarantee it. However, the complex constraints optimization problem with uncertain time has not been studied. And the time-energy optimal problem, very important for path planning, has not been taken into consideration too. Accordingly, a new method, based on the Third Kind of Generation Function, Pseudospectral method and one-sided approximation, is proposed in this paper for the first time.

II. METHODS

A. THE OPTIMAL CONTROL PROBLEM FOR UGS PATH PLANNING

The UGS path planning and optimal control problem described in this paper is solving a time-energy optimal control problem with the given initial and final posture conditions, the nonholonomic motion and boundary constraints. Thus, we need to describe them separately.

1) THE DESCRIPTION OF MOTION CONSTRAINTS

The acceleration of motion control for UGS is provided by the rear wheels, and the orientation is controlled by the front wheels. Assuming that sliding rolling will not occur, the motion can be analyzed without considering the horizontal thrust, friction or inertia characteristics, and a nonholonomic constraint can be obtained in [32]. The motion relationships for the UGS are illustrated in the Fig. 1,

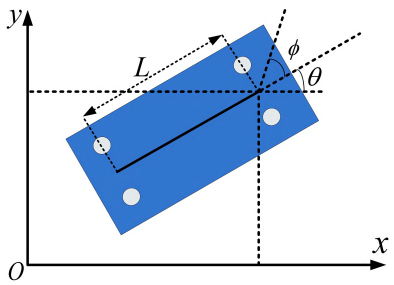


FIGURE 1. The motion relationships for the UGS.

where, L denotes the longitudinal distance between the front wheel and the rear wheel. The direction variable θ is the angle between the x axis and the longitudinal axis of the UGS, ϕ is the steering angle of the front wheel with respect to the heading of the UGS. The kinematics equations [33] of motion is

$$\frac{d\mathbf{X}(t)}{dt} = \frac{d}{dt}\begin{pmatrix} x \\ y \\ \theta \end{pmatrix} = \begin{pmatrix} V\cos(\theta) \\ V\sin(\theta) \\ \dfrac{V\tan(\phi)}{L} \end{pmatrix} \tag{1}$$

where, the state variables $\mathbf{X}$ include two coordinates $\begin{pmatrix} x & y \end{pmatrix}$ and direction variable θ, the control variable $\mathbf{U}$ consists of translational velocity V and steering angle ϕ. In addition, some system may have strict requirements on translational

velocity of the beginning and end moments. In order to give a more representative model, translational velocity will be seen as state variables, and acceleration a will be introduced in the control variable.

So, $\mathbf{X}(t) = \begin{pmatrix} x & y & \theta & V \end{pmatrix}^T$, $\mathbf{U}(t) = \begin{pmatrix} \phi & a \end{pmatrix}^T$, and the kinematics equations of motion can be obtained as follows,

$$\frac{d\mathbf{X}(t)}{dt} = \frac{d}{dt}\begin{pmatrix} x \\ y \\ \theta \\ V \end{pmatrix} = \begin{pmatrix} V\cos(\theta) \\ V\sin(\theta) \\ \dfrac{V\tan(u_1)}{L} \\ u_2 \end{pmatrix} \tag{2}$$

To facilitate computation, the $\tan(u_1)$ will be replaced by u_1, and the motion constraints are

$$\frac{d\mathbf{X}(t)}{dt} = \frac{d}{dt}\begin{pmatrix} x \\ y \\ \theta \\ V \end{pmatrix} = \begin{pmatrix} V\cos(\theta) \\ V\sin(\theta) \\ \dfrac{Vu_1}{L} \\ u_2 \end{pmatrix} \tag{3}$$

where, the boundary value condition is $\mathbf{X}(t_0)$ and $\mathbf{X}(t_f)$. The velocity and control variables should also satisfy the following relation,

$$\begin{cases} V_{\min} \le V \le V_{\max} \\ \phi_{\min} \le \phi \le \phi_{\max} \\ a_{\min} \le a \le a_{\max} \end{cases} \tag{4}$$

where, $V_{\min}$, $V_{\max}$, $a_{\min}$, $a_{\max}$ depends on the UGS, $\phi_{\min}$, $\phi_{\max}$ can be calculated by the turning radius rf, the specific way of calculation is

$$rf = \frac{L}{\tan\phi} \tag{5}$$

2) THE DESCRIPTION OF BOUNDARY CONSTRAINT

In the path planning, the most critical problem is the obstacle avoidance, and how to find a feasible path connecting the start and end position without any obstacle. Therefore, it is a crucial step to use a suitable mathematical model to describe the obstacles as accurately as possible. The circle and polygon are widely used to describe the obstacles, and the circle facilitates subsequent calculations too.

In Fig. 2, the red area indicates the UGS, $O_1\begin{pmatrix} x & y \end{pmatrix}$ denotes the geometric center and r is the radius. The green area represents the obstacles, $O_2\begin{pmatrix} x_o & y_o \end{pmatrix}$ is the geometric center, r_o is the radius, grey area represents the safe area and $dist = r_{so} - r_o$ is the safe distance. According to the relationship of the geometric center, it's safe (no collision) when the UGS satisfies the following relationship,

$$1 - \left(\frac{x - x_0}{r + r_{so}}\right)^2 - \left(\frac{y - y_0}{r + r_{so}}\right)^2 \le 0 \tag{6}$$

However, the polygon can accurately describe the obstacles of any shape, and it may be used for describing the obstacles too. In order to avoid the sharp corners of polygon, the following formula is used:

$$\left(\frac{x - x_o}{a}\right)^p + \left(\frac{y - y_o}{b}\right)^p = 1 \tag{7}$$

The graphical representation is shown in Fig. 3,

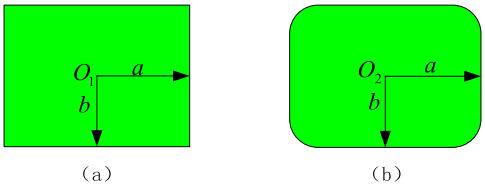


FIGURE 3. The original rectangle and smooth quadrilateral.

The quadrilateral in Fig. 3 (a) is the original rectangle with $p = 200$, and the corner becomes a smooth curve in Fig. 3 (b) resulting from Eq. (8) with $p = 3$. The graph will become the rectangle with $p \to \infty$, and the corner of the rectangle become more and more smooth with $p \to 2$, and the circle or ellipse will be obtained with $p = 2$. For calculating efficiency and precision, p is 3 or 4 when a rectangle is selected to describe the obstacle in this paper.

And the rhombus will be obtained with $p \to 1$, it's shown in Fig. 4 with $1 \le p < 2$,

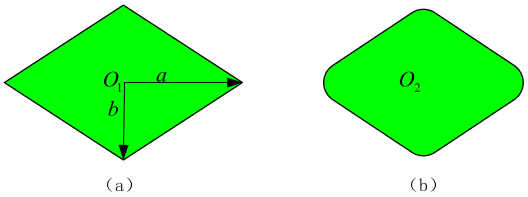


FIGURE 4. The original rhombus and smooth quadrilateral.

The corner becomes a smooth curve in Fig. 4 (b) with $1 < p < 2$, and the original rhombus in Fig. 4 (a) with $p = 1$. It is safe (no collision) when the UGS obeys the following relationships,

$$1 - \left(\frac{x - x_o}{a + dist}\right)^p - \left(\frac{y - y_o}{b + dist}\right)^p \le 0 \tag{8}$$

where, $p \in [1\ \infty)$, and the value can be determined based on the geometric shape of the actual obstacle. And the boundary constraints of all obstacles can be expressed as:

$$1 - \left(\frac{x - x_{oi}}{a_i + dist}\right)^{pi} - \left(\frac{y - y_{oi}}{b_i + dist}\right)^{pi} \le 0 \tag{9}$$

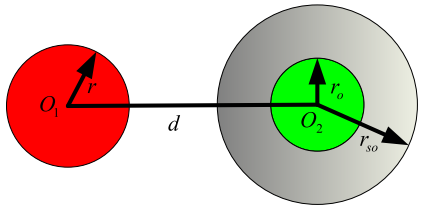


FIGURE 2. The collision detection for UGS.

where, $1 \leq i \leq n$. The obstacles constraints, velocity and control variables constraints are inequalities, then the Eq. (4) and Eq. (9) can be considered together, and the boundary constraints for UGS can be denoted as $\mathbf{h} \leq \mathbf{0}$.

3) THE OBJECTIVE FUNCTION

The shortest time from the starting point to the goal is usually the main index for path planning. However, the velocity needs to be as stable as possible in terms of controlling the problem. As the high incidence stage is the turning process, it's necessary to make the path as straight as possible. And the time, velocity and steering angle should be taken into consideration simultaneously. Thus, the objective function can be considered as a time-energy optimal Bolza function,

$$J\left(\mathbf{X}(*)\ \mathbf{U}(*)\ t\right) = wk\left(t_f - t_0\right) + \frac{1}{2}\int_{t_0}^{t_f} \mathbf{U}^T\mathbf{U}dt \quad (10)$$

where, t_f is the time to reach the goal, t_0 is the departure time, wk is a weight factor and J is the objective function.

The system of differential equations is a nonlinear coupled system, and it's hard to solve it directly. Thus, we can adopt the quasilinearization method to solve the equation. So, the nonlinear optimal control model can be converted into a series of linear-quadratic optimal control problems (LQOCPS) by the linearization of nonholonomic motion and boundary constraints. Thus, the objective function can be expressed as,

$$J^{[k+1]} = wk\left(t_f - t_0\right) + \frac{1}{2}\int_{t_0}^{t_f} \left(\mathbf{U}^{[k+1]}\right)^T \mathbf{U}^{[k+1]}dt \quad (11)$$

where, $(*)^{[k+1]}$ denotes the value within the $(k+1)-th$ iterations.

The nonholonomic motion constraints can be expressed as

$$\dot{\mathbf{X}}^{[k+1]} = \mathbf{A}^{[k]}\mathbf{X}^{[k+1]} + \mathbf{B}^{[k]}\mathbf{U}^{[k+1]} + \mathbf{W}^{[k]}, \mathbf{X}(t_0) = \mathbf{X}_0 \quad (12)$$

where,

$$\mathbf{A}^{[k]} = \frac{\partial \mathbf{f}\left(\mathbf{X}\ \mathbf{U}\ t\right)}{\partial \mathbf{X}}\Big|_{\mathbf{X}^{[k]},\mathbf{U}^{[k]}}$$
$$\mathbf{B}^{[k]} = \frac{\partial \mathbf{f}\left(\mathbf{X}\ \mathbf{U}\ t\right)}{\partial \mathbf{U}}\Big|_{\mathbf{X}^{[k]},\mathbf{U}^{[k]}}$$
$$\mathbf{W}^{[k]} = \mathbf{f}^{[k]} - \mathbf{A}^{[k]}\mathbf{X}^{[k]} - \mathbf{B}^{[k]}\mathbf{U}^{[k]}$$

The boundary constraints can be expressed as follows,

$$\mathbf{C}^{[k]}\mathbf{X}^{[k+1]} + \mathbf{D}^{[k]}\mathbf{U}^{[k+1]} + \mathbf{V}^{[k]} \leq \mathbf{0} \quad (13)$$

where,

$$\mathbf{C}^{[k]} = \frac{\partial \mathbf{h}\left(\mathbf{X}\ \mathbf{U}\ t\right)}{\partial \mathbf{X}}\Big|_{\mathbf{X}^{[k]},\mathbf{U}^{[k]}}$$
$$\mathbf{D}^{[k]} = \frac{\partial \mathbf{h}\left(\mathbf{X}\ \mathbf{U}\ t\right)}{\partial \mathbf{U}}\Big|_{\mathbf{X}^{[k]},\mathbf{U}^{[k]}}$$
$$\mathbf{V}^{[k]} = \mathbf{h}^{[k]} - \mathbf{C}^{[k]}\mathbf{X}^{[k]} - \mathbf{D}^{[k]}\mathbf{U}^{[k]}$$

And the quasilinearization will not lead to the loss of precision for introducing the $\mathbf{W}^{[k]}$ and $\mathbf{V}^{[k]}$. In order to make it more concise, the iteration designation of constraints will be ignored. The equality can be obtained by introducing a non-negative compensation vector $\boldsymbol{\alpha}$ for the inequalityč

$$\mathbf{CX} + \mathbf{DU} + \mathbf{V} + \boldsymbol{\alpha} = \mathbf{0} \quad (14)$$

Then, the optimal control model for path planning is described as

$$\text{Minimize } J = wk\left(t_f - t_0\right) + \frac{1}{2}\int_{t_0}^{t_f} \left((\mathbf{U})^{\mathbf{T}}\mathbf{U}\right)dt \quad (15)$$

$$\text{subject to } \begin{cases} \dot{\mathbf{X}} = \mathbf{AX} + \mathbf{BU} + \mathbf{W}, \\ \mathbf{X}(t_0) = \mathbf{X}_0, \mathbf{X}\left(t_f\right) = \mathbf{X}_f \\ \mathbf{CX} + \mathbf{DU} + \mathbf{V} + \boldsymbol{\alpha} = \mathbf{0} \end{cases} \quad (16)$$

And Eq. (15∼16) can be transformed into an unconstrained problem by introducing the Lagrange multiplier vector $\boldsymbol{\lambda}$ and the multiplier vector $\boldsymbol{\beta}$. According to the Pontryagin's maximum principle, the multiplier vector should satisfies $\boldsymbol{\alpha}^{\mathbf{T}}\boldsymbol{\beta} = 0$, and $\boldsymbol{\beta} \geq \mathbf{0}$. So, the objective function can be described as

$$J = \int_{t_0}^{t_f} \left(wk + H - \boldsymbol{\lambda}^{\mathbf{T}}\dot{\mathbf{X}}\right)dt \quad (17)$$

where, the Hamiltonian function [34-35] is

$$H = \frac{1}{2}\mathbf{U}^{\mathbf{T}}\mathbf{U} + \boldsymbol{\lambda}^{\mathbf{T}}(\mathbf{AX} + \mathbf{BU} + \mathbf{W}) + \boldsymbol{\beta}^{\mathbf{T}}(\mathbf{CX} + \mathbf{DU} + \mathbf{V} + \boldsymbol{\alpha}) \quad (18)$$

According to the classical variational method, if J is minimal, the Hamiltonian system needs to satisfy the control equation, the Hamiltonian canonical equation and the transversality conditions for the terminal state. And the control equation is

$$\frac{\partial H}{\partial \mathbf{U}} = \mathbf{U} + \mathbf{B}^{\mathbf{T}}\boldsymbol{\lambda} + \mathbf{D}^{\mathbf{T}}\boldsymbol{\beta} = \mathbf{0} \quad (19)$$

The Hamiltonian canonical equation is

$$\begin{cases} \dot{\mathbf{X}} = \dfrac{\partial H}{\partial \boldsymbol{\lambda}} \\ \dot{\boldsymbol{\lambda}} = -\dfrac{\partial H}{\partial \mathbf{X}} \end{cases} \quad (20)$$

The transversality conditions of the terminal state is

$$H\left(t_f\right) = -\frac{\partial\left(wk\left(t_f - t_0\right)\right)}{\partial t_f} \quad (21)$$

The Hamiltonian function is the function of $\mathbf{X}$, $\mathbf{U}$, $\boldsymbol{\lambda}$, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$, and $\mathbf{U}$ can be expressed by $\mathbf{X}$, $\boldsymbol{\lambda}$, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ according to Eq. (19). Thus, H can be viewed as the function of the four independent variables $\mathbf{X}$, $\boldsymbol{\lambda}$, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$.

And the Third Kind of Generation Function within the time interval $[a\ b]$ is

$$S = (\boldsymbol{\lambda}_b)^{\mathbf{T}}\mathbf{X}_b + \int_a^b \left(\boldsymbol{\lambda}^{\mathbf{T}}\dot{\mathbf{X}} - H\right)dt \quad (22)$$

According to the Hamiltonian canonical equation, the variation of S is

$$\delta S = (\delta \boldsymbol{\lambda}_a)^{\mathbf{T}} \mathbf{X}_a + (\delta \mathbf{X}_b)^{\mathbf{T}} \boldsymbol{\lambda}_b \tag{23}$$

So, the Third Kind of Generation Function is just a function of the state variables at the right end of the interval and costate variables at the left end of the interval, and the OSSP method will be developed in the next section based on it.

B. THE OSSP METHOD

1) THE APPROXIMATION OF THE FOUR INDEPENDENT VARIABLES BASED ON THE LEGENDRE PSEUDOSPECTRAL METHOD

First, initialize $t_f = t_f^0 (t_f^0 > t_0)$, $t_f^0 \in \left[t_F^{low},\ t_F^{up} \right]$, where the $\left[t_F^{low},\ t_F^{up} \right]$ is the scope of initial guess.

The time domain $T = \left[t_0\ t_f \right]$ can be discretized into P intervals, and the jth interval is $T^j = \left[t_{j-1}\ t_j \right]$, $j = 1, 2, \cdots, P$. $\forall t \in T^j$, there is a mapping relations Ξ to realize the transformation from t to $\tau \in \left[-1\ 1 \right]$,

$$\Xi : \tau = \frac{2t - (t_j + t_{j-1})}{t_j - t_{j-1}} \tag{24}$$

Then, the following relationship can be obtained,

$$\frac{d\tau}{dt} = \frac{2}{t_j - t_{j-1}} \tag{25}$$

In the next step, the $\mathbf{X}^j$, $\boldsymbol{\lambda}^j$, $\boldsymbol{\beta}^j$ and $\boldsymbol{\alpha}^j$ in the $j-th$ interval should be approximated by the N^j order Lagrange interpolating polynomial based on the LGL quadrature nodes, and the LGL nodes τ_l^j, $l = 1, 2, \cdots, N^j - 1$, are the roots of the derivative of the Legendre polynomial $\dot{L}^j(\tau) = 0$. The nodes are located in $\left[-1\ 1 \right]$, and the LGL nodes are $\left(-1,\ \tau_1^j,\ \cdots,\ \tau_{N^j-2}^j,\ 1 \right)$ combined with the first nodes $\tau_0^j = -1$ and the last nodes $\tau_{N^j}^j = 1$. The expression of $\mathbf{H}\Gamma^j$ is as follows:

$$\mathbf{H}\Gamma^j(\tau) = \sum_{l=0}^{N^j} \mathbf{H}\Gamma_l^j \frac{\left(\tau^2 - 1\right)\dot{L}^j(\tau)}{N^j\left(N^j+1\right)\left(\tau - \tau_l^j\right)L_l^j} \tag{26}$$

where, $\mathbf{H}\Gamma^j$ can be $\mathbf{X}^j$, $\boldsymbol{\lambda}^j$, $\boldsymbol{\beta}^j$and $\boldsymbol{\alpha}^j$, and $(*)_0^j = (*)_{N^{j-1}}^{j-1}$, $j = 2, 3, \cdots P$.

2) THE APPLICATION OF THE SYMPLECTIC METHOD BASED ON THE THIRD KIND OF GENERATION FUNCTION TO EACH INTERVAL

The derivative of state variables at the LGL node is given by

$$\frac{d\mathbf{X}_k^j(\tau)}{d\tau} = \sum_{l=0}^{N^j} \mathbf{X}_l^j \mathbf{D}_{kl}^j \tag{27}$$

where, the differentiation matrix $\mathbf{D}_{kl}^j$ is defined as the Pseudospectral differential matrix. And the Third Kind of Generation Function can be expressed as

$$\begin{aligned} S\left(\boldsymbol{\lambda}_0^j\ \mathbf{X}_{N^j}^j\right) &= \left(\boldsymbol{\lambda}_0^j\right)^T \mathbf{X}_0^j \\ &+ \sum_{k=0}^{N^j} w_k^j \left[\left(\lambda_k^j\right)^T \sum_{l=0}^{N^j} \mathbf{D}_{kl}^j \mathbf{X}_l^j - \frac{t^j - t^{j-1}}{2} H \right] \end{aligned} \tag{28}$$

where, w_k^j is the weight coefficient of the $j-th$ interval, and the expression is

$$w_k^j = \frac{2}{N^j\left(N^j+1\right)\left(L_k^j\right)^2} \tag{29}$$

Since the Third Kind of Generation Function is just a function of $\boldsymbol{\lambda}_0^j$ and $\mathbf{X}_{N^j}^j$, it can be considered as independent variables, while the others are stationary points of S^j in the $j-th$ interval, thus the stationary condition can be applied to them as follows,

$$\begin{cases} \dfrac{\partial S^j}{\partial \bar{\boldsymbol{X}}^j} = \mathbf{0} \\ \dfrac{\partial S^j}{\partial \bar{\boldsymbol{\lambda}}^j} = \mathbf{0} \end{cases} \tag{30}$$

where,

$$\begin{aligned} \bar{\boldsymbol{X}}^j &= \left\{ \left(\mathbf{X}_0^j\right)^T, \left(\mathbf{X}_1^j\right)^T, \cdots, \left(\mathbf{X}_{N^j-1}^j\right)^T \right\}^T \\ \bar{\boldsymbol{\lambda}}^j &= \left\{ \left(\boldsymbol{\lambda}_1^j\right)^T, \left(\boldsymbol{\lambda}_2^j\right)^T, \cdots, \left(\boldsymbol{\lambda}_{N^j}^j\right)^T \right\}^T \end{aligned}$$

And the costate and state variable at $\boldsymbol{\lambda}_0^j$, $j = 1, 2, \cdots, P$, and $\mathbf{X}_{N^j}^j$ can be obtained as follows

$$\begin{cases} \dfrac{\partial S^j}{\partial \boldsymbol{\lambda}_0^j} = \mathbf{X}_0^j \\ \dfrac{\partial S^j}{\partial \mathbf{X}_{N^j}^j} = \boldsymbol{\lambda}_{N^j}^j \end{cases} \tag{31}$$

Describing the Eq. (30 $\sim$ 31) in a uniform format as

$$\begin{cases} \dfrac{\partial S_0^j}{\partial \boldsymbol{\lambda}_0^j} = \mathbf{X}_0^j \\ \dfrac{\partial S_m^j}{\partial \boldsymbol{\lambda}_m^j} = \mathbf{0}, \quad m = 1, 2, \cdots, N^j \\ \dfrac{\partial S_m^j}{\partial \mathbf{X}_m^j} = \mathbf{0}, \quad m = 0, 1, \cdots, N^j - 1 \\ \dfrac{\partial S_{N^j}^j}{\partial \mathbf{X}_{N^j}^j} = \boldsymbol{\lambda}_{N^j}^j \end{cases} \tag{32}$$

And $\frac{\partial S_m^j}{\partial \mathbf{X}_m^j}$, $\frac{\partial S_m^j}{\partial \boldsymbol{\lambda}_m^j}$ can be expressed as

$$\begin{cases} \dfrac{\partial S_m^j}{\partial \boldsymbol{\lambda}_m^j} = \sum\limits_{n=0}^{N^j} \left(\mathbf{K}_{mn}^{\lambda\lambda}\right)^j \boldsymbol{\lambda}_n^j + \sum\limits_{n=0}^{N^j} \left(\mathbf{K}_{mn}^{\lambda X}\right)^j \mathbf{X}_n^j + \left(\boldsymbol{\xi}_m^{\boldsymbol{\lambda}}\right)^j \boldsymbol{\beta}_m^j + \left(\boldsymbol{\gamma}_m^{\boldsymbol{\lambda}}\right)^j \\ \dfrac{\partial S_m^j}{\partial \mathbf{X}_m^j} = \sum\limits_{n=0}^{N^j} \left(\mathbf{K}_{mn}^{\mathbf{X}\lambda}\right)^j \boldsymbol{\lambda}_n^j + \sum\limits_{n=0}^{N^j} \left(\mathbf{K}_{mn}^{\mathbf{XX}}\right)^j \mathbf{X}_n^j + \left(\boldsymbol{\xi}_m^{\mathbf{X}}\right)^j \boldsymbol{\beta}_m^j + \left(\boldsymbol{\gamma}_m^{\mathbf{X}}\right)^j \end{cases} \tag{33}$$

where,

$$\begin{aligned} \left(\mathbf{K}_{mn}^{\lambda\lambda}\right)^j &= \frac{t^j - t^{j-1}}{2} w_m^j \mathbf{B}_m^j \left(\mathbf{B}_m^j\right)^T \delta_m^n \\ \left(\mathbf{K}_{mn}^{\lambda X}\right)^j &= \left[\left(K_{mn}^{\mathbf{X}\lambda}\right)^j\right]^T \\ &= -w_m^j \mathbf{D}_{mn}^j - \delta_m^0 \delta_m^0 \mathbf{I} - \frac{t^j - t^{j-1}}{2} w_m^j \left(\mathbf{A}_m^j\right)^T \delta_m^n \\ \left(\mathbf{K}_{mn}^{\mathbf{XX}}\right)^j &= (\mathbf{0})_{4(N^j+1)\times 4(N^j+1)} \\ \left(\boldsymbol{\xi}_m^{\boldsymbol{\lambda}}\right)^j &= \frac{t^j - t^{j-1}}{2} w_m^j \mathbf{B}_m^j \left(\mathbf{D}_m^j\right)^T \\ \left(\boldsymbol{\xi}_m^{\mathbf{X}}\right)^j &= -\frac{t^j - t^{j-1}}{2} w_m^j \left(\mathbf{C}_m^j\right)^T \\ \left(\boldsymbol{\gamma}_m^{\boldsymbol{\lambda}}\right)^j &= -\frac{t^j - t^{j-1}}{2} \mathbf{W}_m^j \\ \left(\boldsymbol{\gamma}_m^{\boldsymbol{\lambda}}\right)^j &= (\mathbf{0})_{4(N^j+1)\times 1} \end{aligned}$$

Then,

$$\begin{bmatrix} \mathbf{K}_{11}^j & \mathbf{K}_{12}^j & \mathbf{K}_{13}^j & \mathbf{K}_{14}^j \\ \mathbf{K}_{21}^j & \mathbf{K}_{22}^j & \mathbf{K}_{23}^j & \mathbf{K}_{24}^j \\ \mathbf{K}_{31}^j & \mathbf{K}_{32}^j & \mathbf{K}_{33}^j & \mathbf{K}_{34}^j \\ \mathbf{K}_{41}^j & \mathbf{K}_{42}^j & \mathbf{K}_{43}^j & \mathbf{K}_{44}^j \end{bmatrix} \boldsymbol{\sigma}^j + \begin{bmatrix} \boldsymbol{\xi}_{\boldsymbol{\lambda}}^j \\ \boldsymbol{\xi}_{\mathbf{X}}^j \end{bmatrix} \boldsymbol{\beta}^j + \begin{bmatrix} \boldsymbol{\gamma}_{\boldsymbol{\lambda}}^j \\ \boldsymbol{\gamma}_{\mathbf{X}}^j \end{bmatrix} = \begin{bmatrix} \mathbf{X}^{j-1} \\ (\mathbf{0})_{N^j\times 1} \\ (\mathbf{0})_{N^j\times 1} \\ \boldsymbol{\lambda}^j \end{bmatrix}$$

where, the expression of $\boldsymbol{\sigma}^j$ and $\boldsymbol{\beta}^j$ is

$$\begin{cases} \boldsymbol{\sigma}^j = \left\{ \left(\boldsymbol{\lambda}^{j-1}\right)^T, \left(\bar{\boldsymbol{\lambda}}^j\right)^T, \left(\bar{\boldsymbol{X}}^j\right)^T, \left(\mathbf{X}^j\right)^T \right\}^T \\ \boldsymbol{\beta}^j = \left\{ \left(\boldsymbol{\beta}_0^j\right)^T, \left(\boldsymbol{\beta}_1^j\right)^T, \cdots, \left(\boldsymbol{\beta}_{N^j}^j\right)^T \right\}^T \end{cases} \tag{34}$$

The detailed coefficient expressions of $\boldsymbol{\sigma}^j$ is $\mathbf{K}_{11}^j = \left(\mathbf{K}^{\lambda\lambda}\right)_{(1:4)\times(1:4)}^j$, $\mathbf{K}_{12}^j = \left(\mathbf{K}^{\lambda\lambda}\right)_{(1:4)\times(5:end)}^j$, $\mathbf{K}_{13}^j = \left(\mathbf{K}^{\lambda X}\right)_{(1:4)\times(1:4)}^j$, $\mathbf{K}_{21}^j = \left(\mathbf{K}_{12}^j\right)^T$, $\mathbf{K}_{23}^j = \left(\mathbf{K}^{\lambda X}\right)_{(5:end)\times(1:4)}^j$, $\mathbf{K}_{24}^j = \left(\mathbf{K}^{\lambda X}\right)_{(5:end)\times(5:end)}^j$, $\mathbf{K}_{31}^j = \left(\mathbf{K}_{13}^j\right)^T$, $\mathbf{K}_{32}^j = \left(\mathbf{K}_{23}^j\right)^T$, $\mathbf{K}_{33}^j = \left(\mathbf{K}^{\lambda\lambda}\right)_{(1:end-4)\times(1:4)}^j$, $\mathbf{K}_{34}^j = \left(\mathbf{K}^{\lambda\lambda}\right)_{(1:end-4)\times(5:end)}^j$, $\mathbf{K}_{41}^j = \left(\mathbf{K}_{14}^j\right)^T$, $\mathbf{K}_{42}^j = \left(\mathbf{K}_{24}^j\right)^T$, $\mathbf{K}_{43}^j = \left(\mathbf{K}_{34}^j\right)^T$.

The coefficient matrix of $\boldsymbol{\beta}^j$ is $\boldsymbol{\xi}^j$, the constant matrix term is $\boldsymbol{\gamma}^j$, the right side term is $\mathbf{r}^j$, then,

$$\mathbf{K}^j \boldsymbol{\sigma}^j + \boldsymbol{\xi}^j \boldsymbol{\beta}^j + \boldsymbol{\gamma}^j = \mathbf{r}^j \tag{35}$$

As $\mathbf{U} = \mathbf{g}\left(\mathbf{X},\ \boldsymbol{\lambda},\ \boldsymbol{\beta}\right)$, the equation in the $j-th$ interval can be organized as

$$\mathbf{C}^j \mathbf{X}^j - \mathbf{H}^j \boldsymbol{\lambda}^j - \mathbf{M}^j \boldsymbol{\beta}^j + \mathbf{V}^j + \boldsymbol{\alpha}^j = 0 \tag{36}$$

where, the detailed coefficient expressions of $\mathbf{X}^j$, $\boldsymbol{\lambda}^j$, $\boldsymbol{\beta}^j$ are

$$\begin{cases} \mathbf{C}^j = diag\left(\mathbf{C}_0^j,\ \mathbf{C}_1^j,\ \cdots,\ \mathbf{C}_{N^j}^j\right) \\ \mathbf{H}^j = diag\left\{\mathbf{D}_0^j \left(\mathbf{B}_0^j\right)^T,\ \mathbf{D}_1^j \left(\mathbf{B}_1^j\right)^T,\ \cdots,\ \mathbf{D}_{N^j}^j \left(\mathbf{B}_{N^j}^j\right)^T\right\} \\ \mathbf{M}^j = diag\left\{\mathbf{D}_0^j \left(\mathbf{D}_0^j\right)^T,\ \mathbf{D}_1^j \left(\mathbf{D}_1^j\right)^T,\ \cdots,\ \mathbf{D}_{N^j}^j \left(\mathbf{D}_{N^j}^j\right)^T\right\} \end{cases}$$

and

$$\begin{aligned} \mathbf{V}^j &= \left\{ \left(\mathbf{V}_0^j\right)^T,\ \left(\mathbf{V}_1^j\right)^T,\ \cdots,\ \left(\mathbf{V}_{N^j}^j\right)^T \right\}^T \\ \boldsymbol{\alpha}^j &= \left\{ \left(\boldsymbol{\alpha}_0^j\right)^T,\ \left(\boldsymbol{\alpha}_1^j\right)^T,\ \cdots,\ \left(\boldsymbol{\alpha}_{N^j}^j\right)^T \right\}^T \end{aligned}$$

Accordingly, the compact formula of the $j-th$ interval is

$$\begin{cases} \mathbf{K}^j \boldsymbol{\sigma}^j + \boldsymbol{\xi}^j \boldsymbol{\beta}^j + \boldsymbol{\gamma}^j = \mathbf{r}^j \\ \boldsymbol{\Gamma}^j \boldsymbol{\sigma}^j - \mathbf{M}^j \boldsymbol{\beta}^j + \mathbf{V}^j + \boldsymbol{\alpha}^j = \mathbf{0} \\ \left(\boldsymbol{\alpha}^j\right)^T \boldsymbol{\beta}^j = 0, \quad \boldsymbol{\alpha}^j \geq \mathbf{0}, \ \ \boldsymbol{\beta}^j \geq \mathbf{0} \end{cases} \tag{37}$$

where, $\boldsymbol{\Gamma}^j = [\,-\mathbf{H}^j,\ \mathbf{C}^j\,]$.

3) THE RESULT OF THE WHOLE-TIME DOMAIN

The compact form can be obtained by assembling the result of each interval in the whole-time domain as follows,

$$\begin{cases} \mathbf{K}\sigma + \boldsymbol{\xi}\boldsymbol{\beta} + \boldsymbol{\gamma} = \mathbf{r} \\ \boldsymbol{\Gamma}\boldsymbol{\sigma} - \mathbf{M}\beta + \mathbf{V} + \boldsymbol{\alpha} = \mathbf{0} \\ (\boldsymbol{\alpha})^T \boldsymbol{\beta} = 0, \quad \boldsymbol{\alpha} \geq \mathbf{0}, \ \ \boldsymbol{\beta} \geq \mathbf{0} \end{cases} \tag{38}$$

where, the coefficient $\mathbf{K}$ is a sparse and symmetric matrix, and the coefficient matrixes of Eq. (38) are

$$\mathbf{K} = \begin{bmatrix} \mathbf{K}^1 & \mathbf{Z}^1 & & \\ \left(\mathbf{Z}^1\right)^T & \mathbf{K}^2 & \mathbf{Z}^2 & \\ & \left(\mathbf{Z}^2\right)^T & \ddots & \mathbf{Z}^{P-1} \\ & & \left(\mathbf{Z}^{P-1}\right)^T & \mathbf{K}^P \end{bmatrix}$$

$$\mathbf{Z}^j = \begin{bmatrix} \mathbf{0}_{(8\times N^j+4)\times 4} & \mathbf{0}_{(8\times N^j+4)\times(8\times N^j+4)} \\ -\mathbf{I}_{4\times 4} & \mathbf{0}_{4\times(8\times N^j+4)} \end{bmatrix}$$

$$\begin{cases} \boldsymbol{\xi} = diag\left(\boldsymbol{\xi}^1,\ \boldsymbol{\xi}^2,\ \cdots,\ \boldsymbol{\xi}^P\right) \\ \boldsymbol{\Gamma} = diag\left(\boldsymbol{\Gamma}^1,\ \boldsymbol{\Gamma}^2,\ \cdots,\ \boldsymbol{\Gamma}^P\right) \\ \mathbf{M} = diag\left(\mathbf{M}^1,\ \mathbf{M}^2,\ \cdots,\ \mathbf{M}^P\right) \end{cases}$$

The constant matrix terms are

$$\begin{cases} \boldsymbol{\gamma} = \left\{ \left(\boldsymbol{\gamma}^1\right)^T, \left(\boldsymbol{\gamma}^2\right)^T, \cdots, \left(\boldsymbol{\gamma}^P\right)^T \right\}^T \\ \mathbf{r} = \left(\mathbf{X}_0,\ \mathbf{0}_{1\times sd},\ \boldsymbol{\lambda}_f\right)^T \\ \mathbf{V} = \left\{ \left(\mathbf{V}^1\right)^T, \left(\mathbf{V}^2\right)^T, \cdots, \left(\mathbf{V}^P\right)^T \right\}^T \\ \boldsymbol{\alpha} = \left\{ \left(\boldsymbol{\alpha}^1\right)^T, \left(\boldsymbol{\alpha}^2\right)^T, \cdots, \left(\boldsymbol{\alpha}^P\right)^T \right\}^T \end{cases}$$

where, $sd = \sum_{k=1}^{P} 8\left(N^j + 1\right) - 8$, and

$$\begin{cases} \mathbf{X} = \left\{ \left(\mathbf{X}^1\right)^T, \left(\mathbf{X}^2\right)^T, \cdots, \left(\mathbf{X}^P\right)^T \right\}^T \\ \boldsymbol{\lambda} = \left\{ \left(\boldsymbol{\lambda}^1\right)^T, \left(\boldsymbol{\lambda}^2\right)^T, \cdots, \left(\boldsymbol{\lambda}^P\right)^T \right\}^T \\ \boldsymbol{\beta} = \left\{ \left(\boldsymbol{\beta}^1\right)^T, \left(\boldsymbol{\beta}^2\right)^T, \cdots, \left(\boldsymbol{\beta}^P\right)^T \right\}^T \\ \boldsymbol{\sigma} = \left\{ \left(\boldsymbol{\sigma}^1\right)^T, \left(\boldsymbol{\sigma}^2\right)^T, \cdots, \left(\boldsymbol{\sigma}^P\right)^T \right\}^T \end{cases}$$

Considering the boundary conditions $\mathbf{X}_0$ and $\mathbf{X}_f$, the relevant matrixes need to be modified as follows,

(1). The elements in the row $4\left(N^1+1\right)+1$:$4\left(N^1+2\right)$ of $\mathbf{K}$, $\boldsymbol{\xi}$ and $\boldsymbol{\gamma}$ are replaced with 0, the sections of columns $4\left(N^1+1\right)+1$:$4\left(N^1+2\right)$ and rows $4\left(N^1+1\right)+1$:$4\left(N^1+2\right)$ are replaced with the unit matrix, the rows $4\left(N^1+1\right)+1$:$4\left(N^1+2\right)$ of $\mathbf{r}$ are replaced with $\mathbf{X}_0$;

(2). The elements in the last 4 rows of $\mathbf{K}$, $\boldsymbol{\xi}$ and $\boldsymbol{\gamma}$ are replaced with 0, the sections of the last 4 columns and the last 4 rows of $\mathbf{K}$ are replaced with the unit matrix, and the last 4 rows of $\mathbf{r}$ are replaced with $\mathbf{X}_f$.

The state variables and covariates can be obtained as follows,

$$\boldsymbol{\sigma} = -\mathbf{K}^{-1}\boldsymbol{\xi}\boldsymbol{\beta} - \mathbf{K}^{-1}\left(\boldsymbol{\gamma} - \mathbf{r}\right) \tag{39}$$

Thus, the optimal control solution can be obtained by the following relationship.

$$\begin{cases} \mathbf{Y}\beta + \mathbf{q} = \boldsymbol{\alpha} \\ (\boldsymbol{\alpha})^T \boldsymbol{\beta} = 0, \quad \boldsymbol{\alpha} \geq \mathbf{0}, \ \ \boldsymbol{\beta} \geq \mathbf{0} \end{cases} \tag{40}$$

where, $\mathbf{Y} = \boldsymbol{\Gamma}\boldsymbol{K}^{-1}\boldsymbol{\xi} + \mathbf{M}$, $\mathbf{q} = \boldsymbol{\Gamma}\boldsymbol{K}^{-1}\left(\boldsymbol{\gamma} - \mathbf{r}\right) - \mathbf{V}$.

Since the $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ satisfy the orthogonality relationship, the Lemke method is adopted to obtain the $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ in this paper, and the optimal control $\mathbf{U} = \mathbf{g}\left(\mathbf{X}, \boldsymbol{\lambda}, \boldsymbol{\beta}\right)$ can also be obtained. Thus, the optimal control question at the condition of $t = t_f$ is completely solved.

4) THE DETERMINATION OF THE OPTIMAL TIME FOR THE TIME-ENERGY QUESTION BASED ON THE ONE-SIDED APPROXIMATION

It's necessary to meet the transversality conditions of the terminal state for the optimal control problem with uncertain time.

$$\Lambda\left(t_f\right) = H\left(t_f\right) + wk = 0 \tag{41}$$

The above relationship can be regarded as a question to solve the nonlinear equation about t_f. The equation of transversality conditions is difficult to solve directly, it's necessary to adopt the approach of approximation or iteration. As the velocity and steering angle are both constrained, there is a the shortest theoretical motion time from the initial position to destination along the shortest path with maximum velocity, and it's certainly no solution for the trajectory-planning if motion time is shorter than theoretical minimum. Considering the stability of the control, the optimal solution is bigger than the minimum, but the corresponding value is likely to get a smaller value than the theoretical minimum in iterative process, and most of the relevant methods can't avoid it. In order to solve this problem, one-sided approximation from the right side of theoretical minimum (movement time is more abundant) will adopted to get the optimal solution. It can make sure that the corresponding value for each iteration is greater than the theoretical minimum.

To determine the shortest time $t_F = \min t_f$, the iteration relationship can be designed as follows,

$$t_f^{u+1} = t_f^u - \Im\left(t_f^u - t_F^{low}\right)\Lambda\left(t_f^u\right) \tag{42}$$

where, $\Im$ is an adjusting parameter. According to the Eq. (42), the result of the $u+1$ iterations is determined by the u iterations, it will continue iterating until meeting the following relationship,

$$\left|\Lambda\left(t_f^u\right)\right| \leq \rho \tag{43}$$

where, ρ is the precision of the convergence. Then, $\mathbf{X}^u_{t=t_f}$, $\mathbf{U}^u_{t=t_f}$, $\boldsymbol{\lambda}^u_{t=t_f}$, $\boldsymbol{\beta}^u_{t=t_f}$ and $\boldsymbol{\alpha}^u_{t=t_f}$ can be obtained under the assumptions of $t_f = t_f^u$ by using the method. Thus, the corresponding t_F can be obtained with given initial value $t_f^0 \in \left[t_F^{low}, t_F^{up}\right]$ by using the OSSP method.

5) THE DESCRIPTION OF THE OSSP METHOD

In this paper, the OSSP method is proposed for solving the optimal control problem, the specific calculation steps are as follows:

(1) Determination of the number of the LGL nodes in each interval;

(2) Initialization of t_f and the initial condition $\mathbf{X}^{[0]}$, $\mathbf{U}^{[0]}$, $\boldsymbol{\lambda}^{[0]}$, and discrete time domain;

(3) Determination of the approximation of the four independent variables;

(4) Application of the Symplectic method based on the Third Kind of Generation Function to each interval;

(5) Assembling the matrixes in the whole-time domain, and the relevant matrixes need to be modified considering the boundary conditions $\mathbf{X}_0$ and $\mathbf{X}_f$;

(6) $\mathbf{X}^u_{t=t_f}$, $\mathbf{U}^u_{t=t_f}$, $\boldsymbol{\lambda}^u_{t=t_f}$, $\boldsymbol{\beta}^u_{t=t_f}$ and $\boldsymbol{\alpha}^u_{t=t_f}$ can be obtained under the assumptions of $t_f = t_f^u$ by iterating until $\left\|\mathbf{X}^{[k]} - \mathbf{X}^{[k-1]}\right\| / \left\|\mathbf{X}^{[k]}\right\| \leq \varepsilon$.

(7) If $\left|H\left(t_f\right) + wk\right| \leq \rho$, stop and exit; Or, continue iterating, update t_f based on Eq. (42), and returning to step (2), until satisfying Eq. (43).

According to the above steps, the optimal path and t_f can be obtained.

III. RESULTS AND DISCUSSION

A. THE DESCRIPTION OF THE ENVIRONMENT

In order to verify the efficiency and accuracy of the OSSP method, the environment with static and dynamic obstacles are selected in this paper. The computing environment is: Win7 64bit, RAM 4.00 GB, MATLAB R2010b. The state and control variables constraints for an UGS are: $V_{\min} = 0m/s$,

$V_{\max} = 1m/s$, $u_{1\min} = -2$, $u_{1\max} = 2$, $a_{\min} = -1m/s^2$, $a_{\max} = 1m/s^2$. The whole-time domain can be divided into 8 intervals, and $P = 8$, the 8-order Legendre polynomial interpolation is applied to each interval, and $N^j = 8$. Thus, the whole domain can be divided into 64 segments with 65 discrete points. The index of convergence ε is 10^{-4}, ρ is 10^{-3}.

The corresponding time will be greater than the energy by analyzing the objective function. And several experiments also show that the time index is at least 20 times larger than the energy index. So, in order to balance the relationship between time and energy term, and show the importance of both, the value of wk cannot be too large here, $wk = 0.075$ in this case (the value is determined by the different understanding of the importance for time and energy index within the right range). In this case, there are 6 obstacles, the detailed parameters are listed in Table 1.

TABLE 1. The parameters of obstacles.

O	p	a	b	x_0 (m)	y_0 (m)
1	2	8	8	$-30+0.003t^2+0.03\cos t$	$20+0.5t$
2	2	3	8	110	70
3	2	8	8	65	85
4	4	38	18	55	20
5	2	8	8	90	50
6	1.2	8	8	40	72

The radius of the UGS is $r = 10m$, the safe distance is $dist = 2m$, the boundary conditions are $\mathbf{X}_0 = \begin{pmatrix} 0 & 0 & \pi & 0 \end{pmatrix}^T$, $\mathbf{X}_f = \begin{pmatrix} 110 & 110 & 0 & 0 \end{pmatrix}^T$ and $L = 7m$, $t_0 = 0s$. The $\mathbf{X}^{[0]}$ is the state variable corresponding to any approximately feasible path, and $\mathbf{U}^{[0]}$, $\boldsymbol{\lambda}^{[0]}$ could be any value (0.6 is given for this case).

B. RESULTS OF EXPERIMENTS

Based on the OSSP method, the minimum objective function $J_{\min} = 26.6398$, the optimal time $t_F = 191.6242s$, the time index term is 14.3718, the energy index term is 12.2680. And the optimal path can be obtained as shown in Fig. 5.

The hollow red circles in Fig. 5 represent the UGS, and the hollow purple circles represent the dynamic obstacle, the black line is the trajectory of the center for UGS and O1 to O6 are obstacles corresponding to those listed in Table 1 (The safe area is included in obstacles in Fig. 5). It's concluded that the path is very smooth and meets the boundary state.

In order to clearly show the relative position between the UGS and various obstacles, the distance index function, combined with the Eq. (8), is introduced to represent the relative distance between the UGS and each obstacle, the details are shown below.

$$d = \left(\frac{x - x_o}{a + dist}\right)^p + \left(\frac{y - y_o}{b + dist}\right)^p - 1 \tag{44}$$

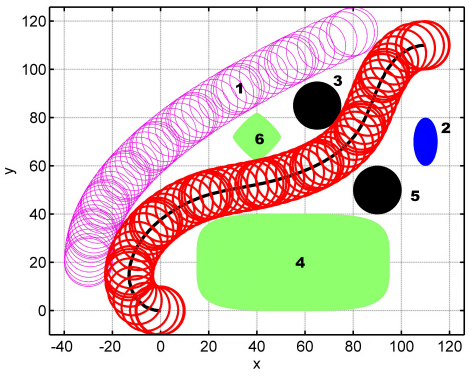


FIGURE 5. The optimal path in the environment with dynamic obstacles.

where, there is no collision for UGS with $d \geq 0$. And the 6 distance indexes can be obtained by Eq. (44), and it's shown as follows.

As shown in Fig. 5 and Fig. 6, the optimal path from the starting point to the goal is along the edge of O1, O3, O4, O6. The corresponding minimum value is: $d_1 = 1.6263 \times 10^{-12}$, $d_2 = 1.8716$, $d_3 = 2.4219 \times 10^{-10}$, $d_4 = 3.6708 \times 10^{-8}$, $d_5 = 0.2627$, $d_6 = 1.5125 \times 10^{-8}$, and all of them are positive. So, there is no collision for UGS along the path obtained by using the proposed method. Additionally, the control variables are

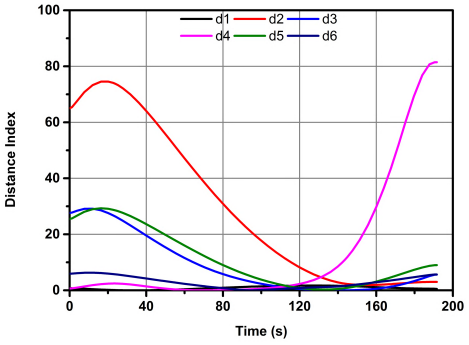


FIGURE 6. The figure of 6 distance indexes.

The control variables in Fig. 7 are relatively stable, $u1$ and $u2$ keep a small value, and the acceleration is $0m/s^2$ unless the starting and ending phase. With the increase of the curve

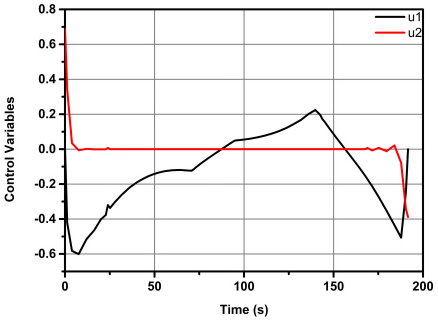


FIGURE 7. The figure of costates and control variable.

of UGS at $t \in [20\ 30]$ and $t \in [175\ 190]$, the $u1$ is sharply changed, and the number of discrete points is so few that it hardly fully expressed changing laws. So, the control variables of this segment fluctuate within a very small range, and it can be solved by adding the number of discrete points. In order to further verify that the high quality path can be got by this method, V and θ are analyzed as follows,

Comparative analysis of Fig. 7 and Fig. 8 indicates that the change of θ and V are smooth during the whole movement process. The UGS accelerates from $0m/s$ to the maximum speed $1m/s$ as soon as possible, and maintains uniform motion until the speed is reduced to $0m/s$ at the end. The result shows that both the path and the corresponding state variables are continuous and relatively smooth, and a good result will be obtained by using the OSSP method.

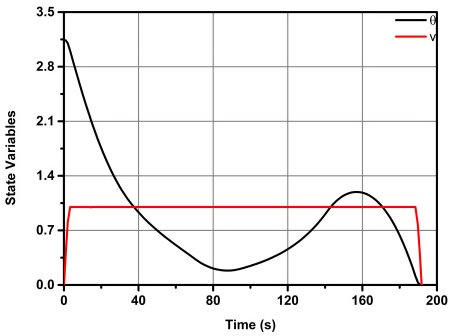


FIGURE 8. The figure of V and θ.

The specific results obtained by the OSSP method compared with the Pseudospectral method, almost the only way used to solve the optimal control problem for path planning, are shown in Table 2.

TABLE 2. The comparison results.

	OSSP	Pseudospectral
t_F	191.6242s	191.7083s
Number of iteration	7	15
Values of energy	12.2680	12.3471
Computation time	6.9500s	89.7281s

According to the analysis results presented in Table 2, the ratio of the computation time of the OSSP to the Pseudospectral is 7.75%, and the computational efficiency is significantly better than that of the Pseudospectral method under the condition of static obstacles environment. To verify the calculation efficiency of the proposed algorithm with high dimension, experiments were conducted with different number of discrete points, and the results are as follows.

The analysis results presented in Table 3 reveal that the computational efficiency is lower with the increase of the number of discrete points and the dimension. Additionally, the computation time increases dramatically with the increase of discrete points, and the phenomenon of ''Curse of Dimensionality'' will occur by using the Pseudospectral method

TABLE 3. The comparison results with different number of discrete points.

Number of discrete points	OSSP	Pseudospectral
37	2.1805s	20.1392s
50	4.5162s	29.4023s
91	17.0358s	----
170	52.5212s	----

with high dimension. Although the computation time will also slight increase using the OSSP method proposed in this paper, the efficiency of optimization the calculation will be higher compared with Pseudospectral method, and the phenomenon of ''Curse of Dimensionality'' can be avoided too.

Accordingly, the OSSP method, with good accuracy and computation efficiency, can solve optimal control problem of the path planning for UGS under the complex conditions with static and dynamic obstacles. The initial state variables are the corresponding value of the feasible path, and there is no requirement for the initial value of costates and control variables, which means the algorithm is of good applicability and operability, and it's suitable for solving path planning for the optimal control problem of UGS.

IV. CONCLUSION

The optimal control problem of the path planning for UGS was modeled. To solve the problem, the OSSP method is proposed based on the one-sided approximation, the Third Kind of Generation Function and Pseudospectral method. To verify the feasibility and efficiency of the method, experiments were conducted under a static and dynamic obstacles environment, and the results were compared with those obtained by the Pseudospectral method. The results revealed that the OSSP method can solve the optimal control problem of the path planning for UGS with high accuracy and efficiency, and the phenomenon of ''Curse of Dimensionality'' can be avoided. In addition, there is almost no requirement for the initial assignment of costates and control variables, which results in very strong maneuverability and feasibility in practical application. However, although it's easy to find the initial assignment of state variables by the corresponding feasible path, the different initial assignment will affect the convergence speed. It is an interesting and meaningful work to solve this problem so that it can be applied to the engineering problem solving by the OSSP method. This study also laid a good foundation for the direction of the follow-up work. Indeed, to further improve the calculation efficiency, the algorithm can be further optimized too.

REFERENCES

[1] H. Tong, ''Path planning of UAV based on Voronoi diagram and DPSO,'' *Procedia Eng.*, vol. 29, pp. 4198–4203, Jan. 2012.

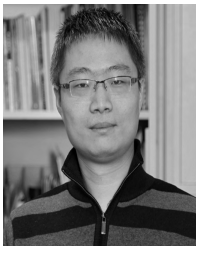
[2] M. Candeloro, A. M. Lekkas, J. Hegde, and A. J. Sørensen, ''A 3D dynamic Voronoi diagram-based path-planning system for UUVs,'' in *Proc. OCEANS MTS/IEEE Monterey*, Monterey, CA, USA, Sep. 2016, pp. 1–8.

[3] H. Niu, Y. Lu, A. Savvaris, and A. Tsourdos, "Efficient path planning algorithms for unmanned surface vehicle," *IFAC-PapersOnLine*, vol. 49, no. 23, pp. 121–126, 2016.
[4] A. K. Guruji, H. Agarwal, and D. K. Parsediya, "Time-efficient A* algorithm for robot path planning," *Procedia Technol.*, vol. 23, pp. 144–149, Jan. 2016.
[5] W. Yu and X. Qu, "Obstacle avoidance and path planning for carrier aircraft launching," *Chin. J. Aeronaut.*, vol. 28, no. 3, pp. 695–703, 2015.
[6] L. De Filippis, G. Guglieri, and F. Quagliotti, "A minimum risk approach for path planning of UAVs," *J. Intell. Robot. Syst.*, vol. 61, nos. 1–4, pp. 203–219, 2011.
[7] K. Zhang, P. Liu, W. Kong, J. Zou, and M. Liu, "An improved heuristic algorithm for UCAV path planning," *J. Optim.*, vol. 2017, Apr. 2017, Art. no. 8936164.
[8] A. Viseras, R. O. Losada, and L. Merino, "Planning with ants: Efficient path planning with rapidly exploring random trees and ant colony optimization," *Int. J. Adv. Robot. Syst.*, vol. 13, no. 5, 2016.
[9] Z. Fang, C. Luan, and Z. Sun, "A 2D voronoi-based random tree for path planning in complicated 3D environments," in *Intelligent Autonomous Systems*, vol. 531. 2016, pp. 433–445.
[10] Y. Zhang, L. Wu, and S. Wang, "UCAV path planning by fitness-scaling adaptive chaotic particle swarm optimization," *Math. Problems Eng.*, vol. 2013, Jun. 2013, Art. no. 705238.
[11] P. K. Das, H. S. Behera, P. K. Jena, and B. K. Panigrahi, "Multi-robot path planning in a dynamic environment using improved gravitational search algorithm," *J. Elect. Syst. Inf. Technol.*, vol. 3, no. 2, pp. 295–313, 2016.
[12] K. Groh and S. Röck, "A contribution to collision-free trajectory planning for handling systems in varying environments," *Prod. Eng..*, vol. 4, no. 1, pp. 101–106, 2010.
[13] M. Huptych, K. Groh, and S. Röck, "Online path planning for industrial robots in varying environments using the curve shortening flow method," in *Intelligent Robotics and Applications*, vol. 7101. 2011, pp. 73–82.
[14] M. Huptych and S. Röck, "Online path planning in dynamic environments using the curve shortening flow method," *Prod. Eng.*, vol. 9, nos. 5–6, pp. 613–621, 2015.
[15] J. Wu, H. Wang, N. Li, P. Yao, Y. Huang, and H. Yang, "Path planning for solar-powered UAV in urban environment," *Neurocomputing.*, vol. 275, no. 31, pp. 2055–2065, Jan. 2018.
[16] M. Mancini, G. Costante, P. Valigi, T. A. Ciarfuglia, J. Delmerico, and D. Scaramuzza, "Toward domain independence for learning-based monocular depth estimation," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1778–1785, Jul. 2017.
[17] N. Morales, J. Toledo, and L. Acosta, "Path planning using a multiclass support vector machine," *Appl. Soft Comput.*, vol. 43, pp. 498–509, Jun. 2016.
[18] L. Blackmore, B. Açıkmeşe, and J. M. Carson, III, "Lossless convexification of control constraints for a class of nonlinear optimal control problems," *Syst. Control Lett.*, vol. 61, no. 8, pp. 863–870, 2012.
[19] Y. Mao, M. Szmuk, and B. Açıkmeşe, "Successive convexification of nonconvex optimal control problems and its convergence properties," in *Proc. IEEE 55th CDC*, Las Vegas, NV, USA, Dec. 2016, pp. 3636–3641.
[20] B. Açıkmeşe, J. M. Carson, and L. Blackmore, "Lossless convexification of nonconvex control bound and pointing constraints of the soft landing optimal control problem," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 6, pp. 2104–2113, Nov. 2013.
[21] X. Liu, P. Lu, and B. Pan, "Survey of convex optimization for aerospace applications," *Astrodynamics*, vol. 1, no. 1, pp. 23–40, 2017.
[22] X. Liu and P. Lu, "Solving nonconvex optimal control problems by convex optimization," *J. Guid., Control, Dyn.*, vol. 37, no. 3, pp. 750–765, 2014.
[23] Q. Gong, W. Kang, and I. M. Ross, "A pseudospectral method for the optimal control of constrained feedback linearizable systems," *IEEE Trans. Autom. Control*, vol. 51, no. 7, pp. 1115–1129, Jul. 2006.
[24] L. R. Lewis, I. M. Ross, and Q. Gong, "Pseudospectral motion planning techniques for autonomous obstacle avoidance," in *Proc. 46th IEEE Conf. Decis. Control*, New Orleans, LA, USA, Dec. 2007, pp. 5997–6002.
[25] W. Zeng, J. Yi, X. Rao, and Y. Zheng, "A two-stage path planning approach for multiple car-like robots based on PH curves and a modified harmony search algorithm," *Eng. Optim.*, vol. 49, no. 11, pp. 1995–2012, 2017, doi: 10.1080/0305215X.2017.1281610.
[26] H. J. Peng, Q. Gao, Z. G. Wu, and W. X. Zhong, "Symplectic adaptive algorithm for solving nonlinear two-point boundary value problems in Astrodynamics," *Celestial Mech. Dyn. Astron.*, vol. 110, no. 4, pp. 319–342, 2011.
[27] H. Peng, Q. Gao, Z. Wu, and W. Zhong, "Symplectic approaches for solving two-point boundary-value problems," *J. Guid., Control, Dyn.*, vol. 35, no. 2, pp. 653–659, 2012.
[28] H. Peng, Q. Gao, Z. Wu, and W. Zhong, "Efficient sparse approach for solving receding-horizon control problems," *J. Guid., Control, Dyn.*, vol. 36, no. 6, pp. 1864–1872, 2013.
[29] X. Wang, H. Peng, S. Zhang, B. Chen, and W. Zhong, "A symplectic pseudospectral method for nonlinear optimal control problems with inequality constraints," *ISA Trans.*, vol. 68, pp. 335–352, May 2017.
[30] H. Peng, X. Wang, M. Li, and B. Chen, "An hp symplectic pseudospectral method for nonlinear optimal control," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 42, pp. 623–644, Jan. 2017.
[31] M. Li, H. Peng, and W. Zhong, "A symplectic sequence iteration approach for nonlinear optimal control problems with state-control constraints," *J. Franklin Inst.*, vol. 352, no. 6, pp. 2381–2406, 2015.
[32] J. S. Johnston and E. D. Swenson, "Feasibility study of global-positioning-system-based aircraft-carrier flight-deck persistent monitoring system," *J. Aircr.*, vol. 47, no. 5, pp. 1624–1635, 2010.
[33] A. K. Khalaji and S. A. A. Moosavian, "Stabilization of a tractor-trailer wheeled robot," *J. Mech. Sci. Technol.*, vol. 30, no. 1, pp. 421–428, 2016.
[34] J. de Jesús Rubio, "Stable and optimal controls of a proton exchange membrane fuel cell," *Int. J. Control.*, vol. 87, no. 11, pp. 2338–2347, 2014.
[35] J. de Jesús Rubio and A. G. Bravo, "Optimal control of a PEM fuel cell for the inputs minimization," *Math. Problems Eng.*, vol. 2014, Mar. 2014, Art. no. 698250.

JIE LIU received the M.S. degree from AUAF, China, in 2015. He is currently working and pursuing the Ph.D. degree with Naval Aeronautical and Astronautical University. His research interests focus on the control of unmanned ground systems, path planning for robots, optimal control, and related fields such as the process industry control and automation.

WEI HAN received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics, China, in 2003. He is currently a Professor with Naval Aeronautical and Astronautical University, China. His research interests cover the general area of the process industry control and automation, and related fields such as aerodynamics.

CHUN LIU received the M.S. and Ph.D. degrees from Xiamen University, China, in 2010 and 2013, respectively. He is currently an Engineer with the 650 Aircraft Design Institute of AVIC Hongdu, China. His research interests cover the general area of the control of robots and related fields such as simulation.



HAIJUN PENG received the Ph.D. degree from the Dalian University of Technology, China, in 2012. He is currently an Associate Professor with the State Key Laboratory of Structural Analysis for Industrial Equipment, Department of Engineering Mechanics, Dalian University of Technology. His research interests cover the optimal control, multibody dynamics, and robot system.