

An Improved Optimization Algorithm Combined Model Parameterization and Sequential Quadratic Programming

Weiwei Bian^a, Zhenfang Xin^b

Beijing Machine and Equipment Institute, Beijing 100854, China

^akarovie@163.com, ^bxzf340825@163.com

Keywords: Model Parameterization; Sequential Quadratic Programming; Trajectory Optimization

Abstract. According to the nonlinear problem of the trajectory optimization problem of rocket projectile, an improved optimization algorithm combined the direct model parameterization method and the sequential quadratic programming method was proposed. The direct numerical method was presented to convert the infinite dimensional optimization problem into a finite dimensional nonlinear programming problem. The improved sequential quadratic programming algorithm was derived by making a special treatment on the constraint conditions. Taking the range as the independent variable instead of the time, and taking the maximum range trajectory and the shortest time trajectory as the function of the performance index respectively, the trajectory optimization model was presented, and the numerical simulations were carried out. The results show that the proposed algorithm is effective to solve the trajectory optimization problem.

Introduction

Generally speaking, it is difficult to solve the trajectory optimization problem of rocket projectile for its nonlinear characteristics by the analytical method, and the numerical method has become the main algorithm to solve it. The numerical method includes indirect numerical method and direct numerical method [1] [2]. The indirect numerical method takes the variation method to search the first order necessary conditions, which can obtain the extremum for the optimization problem. Those first order necessary conditions are usually in the form of two point boundary value problems that described as a set of differential and algebraic equations. The system state, the control input and the parameters satisfy the two point boundary value problem by the numerical calculation, and then, the optimal solution of the optimization problem can be obtained. But it needs to introduce the co-state variables and the co-state equations in the process of the derivation of the necessary optimal conditions. It is difficult to obtain the initial estimates of the co-state variables, which has huge impact on the two point boundary value problem solving. Avoiding the tedious derivation of the formula and the co-state variables initial value estimation problem of the indirect numerical method, the direct numerical method can convert the infinite dimensional optimization problem into a finite dimensional nonlinear programming problem. It will make the control optimization problem solving more suitable for the characteristics of digital computers, and easier to achieve. In this paper, an improved optimization strategy combined the direct parameterization method and the SQP (Sequential Quadratic Programming) method [3] is presented to solve the trajectory optimization problem of rocket projectile.

Optimization Algorithm

Mathematical description

Essentially, the trajectory optimization problem of rocket projectile can be abstracted as an optimal control problem, and its purpose is to get the control law of a given system, which makes the system have optimal solution under given conditions of some specified performance indexes. In a general way, the optimal control problem can be described as equation (1).

$$\left\{ \begin{array}{l} \min_{\mathbf{u}, \mathbf{p}} J = \Phi(\mathbf{x}(t_f), \mathbf{p}) + \int_{t_0}^{t_f} L(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) dt \\ \text{s.t.} \\ \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \quad t \in [t_0, t_f] \\ 0 = \mathbf{c}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \quad t \in [t_0, t_f] \\ 0 \geq \mathbf{d}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \quad t \in [t_0, t_f] \\ \mathbf{x}(t_0) = \mathbf{x}_0 \\ 0 = \mathbf{Y}(\mathbf{x}(t_f), \mathbf{p}) \\ 0 \geq \mathbf{G}(\mathbf{x}(t_f), \mathbf{p}) \end{array} \right. \quad (1)$$

Where, $\mathbf{x}(t) \in \mathbf{R}^m$ is the state variable; t is the time variable, t_0 and t_f are the bounds; $J(\mathbf{x}, \mathbf{u}, \mathbf{p}, t)$ is the object function, which consists of terminal penalty term $\Phi(\mathbf{x}(t_f), \mathbf{p})$ and integral term $L(\mathbf{x}, \mathbf{u}, \mathbf{p}, t)$; $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t)$ is the state equation; $\mathbf{c}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \in \mathbf{R}^{n_c}$ is the equality constraint function; $\mathbf{d}(\mathbf{x}, \mathbf{u}, \mathbf{p}, t) \in \mathbf{R}^{n_d}$ is the inequality constraint function; $\mathbf{x}(t_0) = \mathbf{x}_0$ is the initial state; $\mathbf{Y}(\mathbf{x}(t_f), \mathbf{p}) \in \mathbf{R}^{n_Y}$ is the terminal equality constraint function; $\mathbf{G}(\mathbf{x}(t_f), \mathbf{p}) \in \mathbf{R}^{n_G}$ is the terminal inequality constraint function.

Equation (1) describes an optimal control problem with confirmed initial and terminal states, which can take the unknown terminal state as the parameters to be optimized by the time variable standardized. The trajectory optimization problem of rocket projectile has the above characteristics, so it can be described as equation (1) that abbreviated as OCP (Optimal Control Problem).

Direct parameterization

Generally, the OCP is infinite dimensional optimization problem. In order to solve it by the direct numerical method, it needs to be parameterized firstly. The parameterized process can be described as follows:

- (1) Divide the fixed time interval into N sub intervals, and $\tau_i \in [t_0, t_i]$, $i = 0, 1, 2, \dots, N$, $t_N = t_f$;
- (2) Take control input $\mathbf{u}(t)$ as $\mathbf{u}_i (i = 0, 1, \dots, N)$ corresponded to $N+1$ time nodes. $\mathbf{u}(\tau_i) = IU(\mathbf{u}_0, \mathbf{u}_1, \dots, \mathbf{u}_N, \tau_i)$ is the interpolation function about the parameter \mathbf{u}_i and the time τ_i .
- (3) Take the $N+1$ state estimates as $\mathbf{s}_i \in \mathbf{R}^n$, $i = 0, 1, 2, \dots, N$.

The state of every sub interval can be transformed into initial value solving problem. Equation (1) can be converted into a finite dimensional optimization problem described as equation (2).

$$\left\{ \begin{array}{l} \min_{\mathbf{u}_i, \mathbf{p}} J = \Phi(\mathbf{x}(t_N), \mathbf{p}) + \sum_{i=0}^{N-1} \int_{t_i}^{t_{i+1}} L(\mathbf{x}, \mathbf{u}_i, \mathbf{p}, t) dt \\ \text{s.t.} \\ \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}_i, \mathbf{p}, t) \quad t \in [t_i, t_{i+1}], i = 0, 1, \dots, N-1 \\ \mathbf{x}(t_i) = \mathbf{s}_i \\ 0 = \mathbf{c}(\mathbf{x}, \mathbf{u}_i, \mathbf{p}, t) \quad t \in [t_i, t_{i+1}], i = 0, 1, \dots, N-1 \\ 0 \geq \mathbf{d}(\mathbf{x}, \mathbf{u}_i, \mathbf{p}, t) \quad t \in [t_i, t_{i+1}], i = 0, 1, \dots, N-1 \\ \mathbf{x}(t_0) = \mathbf{x}_0 \\ 0 = \mathbf{Y}(\mathbf{x}(t_N), \mathbf{p}) \\ 0 \geq \mathbf{G}(\mathbf{x}(t_N), \mathbf{p}) \end{array} \right. \quad (2)$$

Introduce a new variable $\mathbf{y}(t)$: $\dot{\mathbf{y}} = L(\mathbf{x}, \mathbf{u}, \mathbf{p}, t)$, and $\mathbf{y}(t_0) = 0$. Make the control input $\mathbf{x}(t) = \tilde{\mathbf{x}}(\mathbf{u}_i, t)$, the equality constraints $\mathbf{g} \in \mathbf{R}^{n_c(N+1)+n_Y}$, and the inequality constraints $\mathbf{h} \in \mathbf{R}^{n_d(N+1)+n_G}$ discrete respectively. And then, the OCP can be converted into finite dimensional parameters nonlinear programming problem that abbreviated as NLP (Nonlinear Programming), which described as equation (3).

$$\left\{ \begin{array}{l} \min_{\tilde{\mathbf{u}}} J(\tilde{\mathbf{u}}) \\ \text{s.t.} \\ \mathbf{g}(\tilde{\mathbf{u}}) = 0 \\ \mathbf{h}(\tilde{\mathbf{u}}) = 0 \end{array} \right. \quad (3)$$

Improved Sequential Quadratic Programming

In spite of all the equality and inequality constraints are considered by SQP, the K-T conditional form of the SQP sub problem is complex and difficult to solve without taking special treatment on the inequality constraints. The improved SQP combined the direct model parameterization with sequential quadratic programming can be used to solve the inequality constraints of the NLP: at each iteration point, set the known feasible point as the starting point, take the working inequality constraints as equality constraints, don't take care of the rest of the inequality constraints, and after a better feasible point obtained, repeat the procedure.

Firstly, considering all the not satisfied inequality constraints are working, add them to the equality constraints, form a new sets, and then equation (3) can be represented as equation(4).

$$\begin{cases} \min J(\tilde{\mathbf{u}}) \\ \text{s.t.} \\ \bar{\mathbf{g}}(\tilde{\mathbf{u}}) = 0 \end{cases} \quad (4)$$

Where, $\bar{\mathbf{g}}(\tilde{\mathbf{u}}) \in R^{n_{\bar{g}}}$, $n_{\bar{g}}$ consists of $n_c(N+1)$, n_y and the working inequality constraints.

Secondly, the SQP sub problem corresponding to equation (4) can be described as equation (5).

$$\begin{cases} \min_{d_{\tilde{\mathbf{u}}}} \nabla J(\tilde{\mathbf{u}}^k)^T d_{\tilde{\mathbf{u}}} + \frac{1}{2} d_{\tilde{\mathbf{u}}}^T \mathbf{H}_k d_{\tilde{\mathbf{u}}} \\ \text{s.t.} \\ \bar{\mathbf{g}}(\tilde{\mathbf{u}}^k) + \nabla \bar{\mathbf{g}}(\tilde{\mathbf{u}}^k)^T d_{\tilde{\mathbf{u}}} = 0 \end{cases} \quad (5)$$

The Lagrange function $\bar{L}(\tilde{\mathbf{u}}, \mathbf{l})$ can be described as equation (6).

$$\bar{L}(\tilde{\mathbf{u}}, \mathbf{l}) = J(\tilde{\mathbf{u}}) + \mathbf{l}^T \bar{\mathbf{g}}(\tilde{\mathbf{u}}) \quad (6)$$

The K-T constraint $\bar{L}(\tilde{\mathbf{u}}, \mathbf{l})$ can be described as equation (7).

$$\begin{cases} \nabla J(\tilde{\mathbf{u}}^k) + \mathbf{H}_k d_{\tilde{\mathbf{u}}} + \nabla \bar{\mathbf{g}}(\tilde{\mathbf{u}}) \mathbf{l} = 0 \\ \bar{\mathbf{g}}(\tilde{\mathbf{u}}^k) + \nabla \bar{\mathbf{g}}(\tilde{\mathbf{u}}^k)^T d_{\tilde{\mathbf{u}}} = 0 \end{cases} \quad (7)$$

Mark the solution of equation (7) as $(d_{\tilde{\mathbf{u}}^k}, \mathbf{l}^k)$. If $d_{\tilde{\mathbf{u}}^k} = 0$, $\tilde{\mathbf{u}}^k$ is the approximate value of the optimal solution of the NLP problem, else, take one dimensional linear search along the direction of $d_{\tilde{\mathbf{u}}^k}$ as $\tilde{\mathbf{u}}^{k+1} = \tilde{\mathbf{u}}^k + \sigma d_{\tilde{\mathbf{u}}^k}$, where, the step factor $\sigma \in (0, 1]$ should satisfy the constraint that $\tilde{\mathbf{u}}^{k+1}$ is a better solution compared to $\tilde{\mathbf{u}}^k$. Repeat the procedure until the convergence conditions are satisfied. Penalty function as equation (8) is used for the one dimensional linear search to determine the step factor.

$$P(\tilde{\mathbf{u}}, \mathbf{r}) = J(\tilde{\mathbf{u}}) + \sum_{i=1}^{n_c(N+1)+n_y} r_i |g_i(\tilde{\mathbf{u}})| + \sum_{i=1}^{n_d(N+1)+n_G} r_{n_c(N+1)+n_y+i} \max(0, h_i(\tilde{\mathbf{u}})) \quad (8)$$

Where, $P(\tilde{\mathbf{u}}, \mathbf{r}) > P(\tilde{\mathbf{u}}^{k+1}, \mathbf{r})$.

The penalty factor that should be positive can be described as equation (9).

$$r_i = \max(|\lambda_i|, 0.5(|\lambda_i| + r_i^*)) \quad (9)$$

Where, λ_i is the corresponding factor of the Lagrange factor; r_i^* is the penalty factor of the previous iterative computation.

A positive definite matrix \mathbf{B}_k that can be solved by MNFGS (Modified Broyden Fletcher Goldfarb Shanno) algorithm [4] as equation(10) is normally selected as substitute for Hessian \mathbf{H}_k , which is cannot guaranteed as positive definite matrix and difficult to solve.

$$\begin{cases} \mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\boldsymbol{\gamma}\boldsymbol{\gamma}^T}{\boldsymbol{\gamma}^T\mathbf{v}} - \frac{\mathbf{B}_k\mathbf{v}\mathbf{v}^T\mathbf{B}_k}{\mathbf{v}^T\mathbf{B}_k\mathbf{v}} \\ \mathbf{v} = \tilde{\mathbf{u}}^{k+1} - \tilde{\mathbf{u}}^k \\ \boldsymbol{\gamma}' = \mathbf{L}_{\tilde{\mathbf{u}}}(\tilde{\mathbf{u}}^{k+1}, \lambda) - \mathbf{L}_{\tilde{\mathbf{u}}}(\tilde{\mathbf{u}}^k, \lambda) \\ \xi = \begin{cases} 1 & \mathbf{v}^T\boldsymbol{\gamma}' \geq 0.2\mathbf{v}^T\mathbf{B}_k\mathbf{v} \\ 0.8\mathbf{v}^T\mathbf{B}_k\mathbf{v}/(\mathbf{v}^T\mathbf{B}_k\mathbf{v} - \mathbf{v}^T\boldsymbol{\gamma}') & \mathbf{v}^T\boldsymbol{\gamma}' < 0.2\mathbf{v}^T\mathbf{B}_k\mathbf{v} \end{cases} \end{cases} \quad (10)$$

Where, $\mathbf{L}_{\tilde{\mathbf{u}}}$ is the derivative form of Lagrange function to $\tilde{\mathbf{u}}$.

Trajectory Optimization Model

Considering the general gliding flight process of the rocket projectile, in order to grasp the main aspects of the trajectory optimization problem, set the detection and control systems are ideal for work, the atmosphere is the Artillery Standard Atmospheric, the surface of the earth is plane, based on the Instantaneous Equilibrium Assumption [], take range x as the independent variable to replace time t , and the trajectory optimization model of gliding extended rocket projectile can be described as follows:

- (1) The performance index of the maximum range trajectory is $J = -(x(t_f) - x(t_0))$.
- (2) The state equations adopt the centroid trajectory model in the longitudinal plane.
- (3) The control variables are the elevation angle θ_0 and gliding attack angle α .
- (4) The initial state is the elevation state of the rocket projectile, and $t_0 = 0$.
- (5) The constraints are $\theta_{0\min} \leq \theta_0 \leq \theta_{0\max}$ and $|\alpha| \leq \alpha_{\max}$.
- (6) The terminal state are $x(t_f) = x_f$ and $y(t_f) = y_f$.

Therefore, the trajectory optimization model can be represented as equation (10).

$$\begin{cases} \min_{m_p, \theta_0, t_{p0}, t_p, \alpha} J = -(x(t_f) - x(t_0)) \\ \text{s.t.} \\ \begin{cases} \dot{x} = (P \cos \alpha_B - 0.5\rho S v^2 (c_{x0} + c_{x2}\alpha_B^2) - mg \sin \theta) / m \\ \dot{y} = (P \sin \alpha_B + 0.5\rho S v^2 (c_y^\alpha \alpha_B + c_y^{\delta_z} \delta_{zB}) - mg \cos \theta) / (mv) \\ \dot{\theta} = v \cos \theta \\ \dot{\alpha} = v \sin \theta \\ \dot{m} = -m_c \\ m_z^\alpha \alpha_B + m_z^{\delta_z} \delta_{zB} = 0 \end{cases} \\ \theta_{0\min} \leq \theta_0 \leq \theta_{0\max}, \theta(t_0) = \theta_0, \theta(t_s) = 0 \\ m_{p\min} \leq m_p \leq m_{p\max}, t_{p0\min} \leq t_{p0} \leq t_{p0\max}, t_{w\min} \leq t_w \leq t_{w\max} \\ (\alpha + \alpha_{\max})(\alpha - \alpha_{\max}) \leq 0, t \in [t_s, t_f] \\ v(t_0) = v_0 = \sqrt{2E_0 / (f_p m_p + m_k)} \\ x(t_0) = x_0, y(t_0) = y_0 \\ v(t_f) - v_f \geq 0, \theta(t_f) - \theta_f \geq 0, y(t_f) - y_f = 0 \end{cases} \quad (10)$$

Where, the symbolic meaning of the equation (10) is detailed in the literature [5].

Examples and Analysis

Taking a certain type rocket projectile as an example, the initial simulation conditions are as follows: initial velocity $v_0 = 50\text{m/s}$, initial weight $m_0 = 105\text{kg}$, engine impulse of solid rocket motor $I_{sp} = 2600\text{Ns/kg}$, propellant weight $m_p = 35\text{kg}$, engine working time $t_{pw} = 7\text{s}$, trajectory tilt angle $\theta_0 \in (30^\circ, 90^\circ)$, attack angle constraints $\alpha_{\max} = 3^\circ, 5^\circ, 8^\circ, 10^\circ$, height of point of fall $y_f = 0\text{m}$.

Based on those assumptions, the results of the trajectory optimization are: the initial elevation angle $\theta_0^* = 60^\circ$, the trajectory summit time $t_s^* = 56.5\text{s}$, the trajectory maximum height $y_s^* = 20537.8\text{m}$. Corresponded to different attack angle constraints, the comparison results are shown in Table 1.

Table 1: The comparison results of different attack angle constraints.

$\alpha_{\max} / ^\circ$	t_f^*/s	x_f^*/m	$v_f^*/m \cdot s^{-1}$	$\theta_f^*/^\circ$	Extended efficiency/%
No gliding	143.7	32780.5	245.74	-83.30	—
3	223.8	54444.7	174.50	-33.17	66.1
5	296.9	66083.7	143.31	-28.35	101.6
8	334.6	68781.0	113.78	-28.50	109.8
10	338.7	68982.1	99.37	-28.74	110.4

Compared with the conventional no gliding maximum range trajectory, the range of the gliding trajectory increased significantly. When $\alpha_{\max} = 5^\circ$, the range increases about 12.5km than $\alpha_{\max} = 5^\circ$, and when $\alpha_{\max} = 8^\circ$, the range increase about 2.7km than $\alpha_{\max} = 5^\circ$, but when $\alpha_{\max} = 10^\circ$, the range just increases about 200m than $\alpha_{\max} = 8^\circ$. It illustrates that when the attack angle constraint reaches a certain value, it cannot increase the range significantly continuously. The reason is the relative control ability is limited by the rudder deflection angle.

Take the performance index $J = -(x(t_f) - x(t_0))$ replaced as $J = t_f - t_0$ when solve the shortest flight time trajectory optimization problem. The results of three preset ranges are shown in Table 2.

Table 2: Trajectory optimization results of different preset ranges.

x_f / km	$\theta_0^*/^\circ$	t_p^*/s	$y_{t_p}^*/km$	t_f^*/s	$v_{t_f}^*/ms^{-1}$	$\theta_{t_f}^*/^\circ$
40	48.2	45.2	14.395	140.8	219.4	-38.2
50	55.9	55.5	19.845	179.1	225.5	-41.6
60	60.0	61.3	23.129	217.3	217.5	-38.1

Table 2 illustrates that the elevation angle, the trajectory summit time, the trajectory maximum height, and the flight time increase as the preset range increases, but the velocity and trajectory tilt angle of the point of the fall are almost the same.

Figure 1 and Figure 2 show the results of the trajectory optimization more directly.

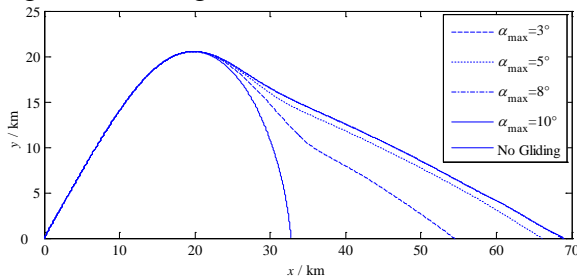


Figure 1: The optimization results of α_{\max} .

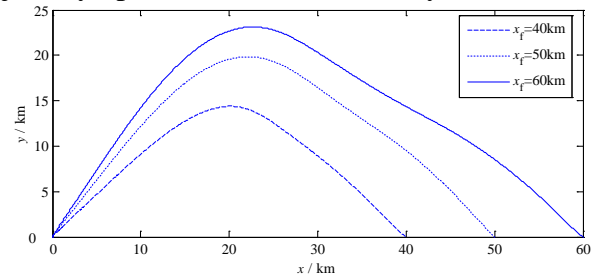


Figure 2: The optimization results of x_f .

Summary

An improved optimization algorithm is proposed to solve the trajectory optimization problem of gliding range-extended rocket projectile with nonlinear characteristics. The results show that the algorithm is suitable for the specific trajectory conditions of rocket projectile and effective for the computation of the trajectory optimization and the analysis of the trajectory characteristics

Acknowledgement

In this paper, the research was sponsored by the Natural Science Foundation of China (No.11402117).

References

- [1] I. Michael Ross, Fariba Fahroo. A Perspective on Methods for Trajectory Optimization, AIAA 2002-4727[R]. 2002.
- [2] M.L. Braakenburg, S. Hartjes, H.G. Visser. Development of a Multi-event Trajectory Optimization Tool for Noise-Optimized Approach Route Design, AIAA 2011-6929[C]. 2011.

- [3] Ross, Steven M. Stochastic Real-Time Optimal Control: A Pseudospectral Approach for Bearing-Only Trajectory Optimization, ADA549111[R]. 2011.
- [4] Broyden C G. A Class of Methods for Solving Nonlinear Simultaneous Equations[J]. Mathematics of Computation, 1965, 19(92): 577-293.
- [5] QIAN Xing-Fang, LIN Rui-Xiong, ZHAO Ya-Nan. Missile Flight Aerodynamics[M]. Beijing: Beijing Institute of Technology Press. 2006. (in Chinese)