mathematics

Article

Cubature Kalman Filters Model Predictive Static Programming Guidance Method with Impact Time and Angle Constraints Considering Modeling Errors

Zihan Xie, Jialun Pu *, Changzhu Wei and Yingzi Guan

School of Astronautics, Harbin Institute of Technology, Harbin 150000, China
* Correspondence: nosay@hit.edu.cn

Abstract: This paper proposes a CKF-MPSP guidance method for hitting stationary targets with impact time and angle constraints for missiles in the presence of modeling errors. This innovative guidance scheme is composed of three parts: First, the model predictive static programming (MPSP) algorithm is used to design a nominal guidance method that simultaneously satisfies impact time and angle constraints. Second, the cubature Kalman filter (CKF) is introduced to estimate values of the influence of the inevitable modeling errors. Finally, a one-step compensation scheme is proposed to eliminate the modeling errors' influence. The proposed method uses a real missile dynamics model, instead of a simplified one with a constant-velocity assumption, and eliminates the effects of modeling errors with the compensation scheme; thus, it is more practical. Simulations in the presence of modeling errors are conducted, and the results illustrate that the CKF-MPSP guidance method can reach the target with a high accuracy of impact time and angles, which demonstrates the high precision and strong robustness of the method.

Keywords: terminal guidance; impact time constraint; impact angle constraint; model predictive static programming; cubature Kalman filter; modeling errors

MSC: 37M10

check for updates

Citation: Xie, Z.; Pu, J.; Wei, C.; Guan, Y. Cubature Kalman Filters Model Predictive Static Programming Guidance Method with Impact Time and Angle Constraints Considering Modeling Errors. *Mathematics* **2023**, *11*, 2990. https://doi.org/10.3390/math11132990

Academic Editors: Haizhao Liang, Jianying Wang and Chuang Liu

Received: 9 May 2023
Revised: 18 June 2023
Accepted: 23 June 2023
Published: 4 July 2023

1. Introduction

Guidance methods have always been a hot research topic in the field of missiles. The early guidance-law design considers only the minimized miss distance requirement, such as proportional navigation (PN). With the development of military science and technology, classical guidance methods no longer satisfy combat requirements [1]. There is an urge to research advanced guidance methods with multiple constraints. Impact angle and impact time constraints are essential for advanced terminal guidance methods. Impact angles are divided into path angle and azimuth angle. Attacking the target with a missile with proper impact angles may improve the destructive effect and hit weak parts of the target. The impact time is vital for attacking time-sensitive targets. Moreover, the guidance method with impact angle and time constraints gives the multi-missile cooperative guidance capability. Thus, investigating the impact time and angle-constrained guidance method (ITACG) is very important.

In this paper, a CKF-MPSP guidance method with impact time and angle constraints for a stationary target is proposed considering modeling errors. A baseline guidance method with impact time and angle constraints is designed based on the MPSP algorithm. The modeling errors are estimated by the CKF and compensate the baseline guidance method to eliminate their effects. The main contributions of this paper are shown below.

(1) An ITACG is designed for a stationary target based on the MPSP algorithm, which can simultaneously achieve impact time and angle constraints. This guidance method

considers the missile's dynamic model instead of a constant-velocity model. Therefore, the proposed method is more suitable for practical missiles.

(2) The proposed guidance method takes the desired time as a terminal condition for static planning. Time-to-go information is not required during the guidance process, which avoids the influence of time-to-go estimation errors on time control accuracy.

(3) A CKF-based modeling error compensation scheme is proposed to solve the problem of the MPSP algorithm being unable to be used for error conditions. This improvement enhances the feasibility of the guidance method in practical applications since the modeling error is inevitable.

It is worth noting that, although the CKF-MPSP guidance method is proposed in the scenario of a missile attacking a stationary target on the ground, in this paper, it can also be used for position control, such as aircraft landing. Furthermore, by predicting and introducing the target's motion model, the CKF-MPSP method can be used for moving-target interception. However, these applications are not the focus of this paper and require further research in the future.

This article is organized as follows. Section 2 provides a literature review of existing achievements in related research fields. Section 3 formulates the problem researched in this paper. Section 4 proposes the CKF-MPSP guidance method to implement time- and angle-constrained guidance. Simulation results are given in Section 5. Section 6 gives the conclusion.

2. Literature Review

Many scholars have conducted much research on ITACG problems. The mainstream methods can be divided into non-predictive guidance and predictive guidance methods.

The guidance laws that utilize the current relative motion information between missiles and targets are called non-predictive guidance. These kinds of methods only adopt the relative motion model instead of the actual model of the missile, making the design of such methods relatively simple. The existing non-predictive guidance methods basically follow two design paradigms.

The first design paradigm is to design guidance laws both in the line-of-sight (LOS) and normal LOS direction for time control and angle constraints, respectively. These kinds of methods are widely used in cooperative guidance scenarios. Zhang [2] proposed an ITACG with finite time convergence. Yu [1], Chen [3], and Lin [4] proposed fixed-time ITACGs based on the sliding mode theory. Ma [5] designed a disturbance-observer-based ITACG to enable the interception of maneuvering targets. Wang [6] proposed a decoupled three-dimensional sliding mode guidance law achieving simultaneous arrival at the target for multiple missiles with angle constraints. Jing [7] proposed a predefined-time convergence ITACG method for a multi-missile cooperative guidance scenario. Because of the design paradigm, the methods presented in Refs. [1–7] all need a control force both in the LOS and normal LOS direction for time control and angle constraints, respectively. However, most existing missiles are thrust-free and controlled by aerodynamic force in the terminal guidance period. In reality, missiles cannot provide the guidance command in the LOS direction, limiting the practical application of such guidance methods.

The second design paradigm is to design guidance laws only in the normal direction of LOS/velocity, which is more practical but more challenging compared to the above methods. Chen [8] simplified missile dynamics under a small heading error approximation and derived an optimal guidance law with impact time and angle constraints against a stationary target. Kim [9] introduced a polynomial guidance method considering impact time and angle constraints. Zhao [10] designed the trajectory as a function with two undetermined parameters and adjusted them to control the impact time and angle. Kang [11] derived a look-angle shaping scheme for ITACG. Hou [12,13] proposed a time-to-go estimation scheme for terminal sliding mode guidance with an impact angle constraint and further designed a nonsingular terminal sliding mode guidance law considering impact time and angle simultaneously. Chen [14] designed a two-stage guidance that satisfies time and angle constraints through a proper guidance-switching strategy. Zhang [15] proposed an ITACG by introducing an impact time feedback control term based on biased PN. Yan [16] proposed a computational geometry guidance against stationary targets, satisfying the constraints by iterating parameters of the geometry curve. Majumder [17] proposed a sliding-mode-control-based nonlinear guidance scheme for controlling both impact angle and impact time simultaneously. Liu [18] designed an adaptive sliding mode ITACG method, increasing its adaptability and robustness. Wang [19] designed a two-stage guidance method, achieving ITACG through reasonable switching between two guidance rules. In Refs. [8–19], the guidance methods rely on time-to-go or range-to-go estimation. However, the estimation is difficult because of the uncontrollable varying velocity. To simplify the design process, the missile's velocity is assumed as constant, which may cause significant time estimation errors, especially when the trajectory is winding due to impact angle constraints. Thus, the guidance effect is not satisfactory in reality. Overall, there are difficulties in the practical application of the non-predictive ITACG methods in missiles because of the model mismatch.

Unlike non-predictive guidance, the predictive guidance methods predict the terminal states using a real model, which can avoid the model mismatch and thus can derive a better guidance performance. As one of the predictive guidance methods, the MPSP-based guidance methods have received widespread attention in recent years. The MPSP algorithm was first introduced in Ref [20]. Combining the philosophy of approximate dynamic programming and model predictive control, the MPSP algorithm obtains the terminal estimation of the output vector by integral prediction. Then, it efficiently solves the optimization problem by turning the dynamic programming problem into a static one. It has been applied in guidance problems due to its ability to deal with varying velocities. The existing MPSP-based guidance methods are mainly focused on the angle constraint only. Oza [21] designed an angle-constrained guidance method for an air-to-ground missile and verified the feasibility of MPSP guidance. Maity [22] further introduced the static Lagrange multiplier in MPSP guidance, improving the computational efficiency. Refs. [23–27] improved the computational efficiency of the MPSP algorithm through further improvements and designed angle-constrained guidance methods, respectively. Refs. [21–27] verified that the MPSP algorithm is feasible for solving guidance problems online. However, during integral prediction, modeling errors will cause the accumulation of estimation errors, affecting the algorithm's performance and stability. Although the receding horizon strategy can correct some previous errors, it cannot eliminate the influence of modeling errors. Thus, the MPSP algorithm is highly dependent on the model's accuracy. To the best of the authors' knowledge, no paper has studied MPSP guidance in the presence of modeling errors so far.

Through the analysis of the ITACG method literature, we can draw the following conclusions:

(1) Compared to non-predictive guidance, predictive guidance may present better performance for unpowered missile reality applications.
(2) As one of the predictive guidance methods, MPSP-based guidance can avoid the model mismatch and thus can derive a better guidance performance, which has been verified in Refs. [21–27].
(3) The existing MPSP-based guidance methods are mainly focused on the angle constraint only. The MPSP-based ITACG methods still need further research.
(4) As an inevitable but significant factor affecting the MPSP algorithm, the modeling errors have not been considered so far.

Based on the above analysis, this paper focuses on the ITACG problems for unpowered missiles in the presence of modeling errors, trying to fill existing research gaps.

3. Problem Description

A 3-D terminal guidance scenario of a missile attacking a stationary ground target is considered in this paper, as shown in Figure 1. M is the missile whose position is (x_0, y_0, z_0). T is a stationary ground target whose position is (x_t, y_t, z_t). The missile is supposed to arrive at the target point with the desired impact path angle θ_f, impact azimuth angle ψ_{vf}, and impact time t_f.

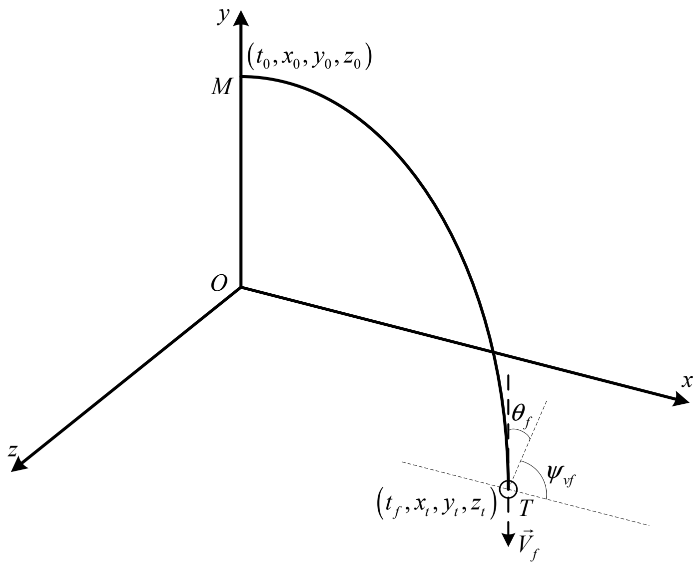


Figure 1. Terminal guidance scenario.

The dynamic model of the missile with an unknown modeling error is written as

$$\dot{\boldsymbol{X}} = \boldsymbol{f}(\boldsymbol{X}, \boldsymbol{U}, \boldsymbol{d}) = \begin{bmatrix} V\cos\theta\cos\psi_v \\ V\sin\theta \\ -V\cos\theta\sin\psi_v \\ -D(\alpha,\beta) - g\sin\theta + d_x \\ \frac{L(\alpha)-g\cos\theta}{V} + d_y \\ -\frac{Z(\beta)}{V\cos\theta} + d_z \end{bmatrix}, \tag{1}$$

where $\boldsymbol{X} = [x, y, z, V, \theta, \psi_v]^{\mathrm{T}}$ is the state vector. x, y, and z are 3-D position coordinates of the missile. V, θ, and ψ_v are the missile's velocity, path angle, and azimuth angle, respectively. The angle of attack (AOA) α and the side slip angle (SSA) β compose the control vector $\boldsymbol{U} = [\alpha, \beta]^{\mathrm{T}}$. The drag acceleration D, lift acceleration L, and lateral acceleration Z are written as

$$\begin{cases} D = C_x(\alpha,\beta) q S_{ref}/m \\ L = C_y(\alpha) q S_{ref}/m \\ Z = C_z(\beta) q S_{ref}/m \end{cases}. \tag{2}$$

In Equation (2), aerodynamic coefficients C_x, C_y, and C_z are related to AOA and SSA. q is dynamic pressure, S_{ref} is reference area, m is the missile's mass. $\boldsymbol{d} = \left[d_x, d_y, d_z\right]^{\mathrm{T}}$ is the unknown modeling error, which may be caused by unmodeled dynamics, uncertain parameters, external disturbances, etc.

Selecting the output vector as $\boldsymbol{Y} = \begin{bmatrix} x & y & z & \theta & \psi_v \end{bmatrix}^{\mathrm{T}}$, the purpose of our guidance method is to determine proper control commands $\boldsymbol{U}$, making sure $\boldsymbol{Y} \to \boldsymbol{Y}_d$ when $t \to t_f$, where $\boldsymbol{Y}_d = \begin{bmatrix} x_t & y_t & z_t & \theta_f & \psi_{vf} \end{bmatrix}^{\mathrm{T}}$.

4. CKF-MPSP Terminal Guidance Method

A CKF-MPSP guidance method is proposed and applied in the terminal guidance scenario to achieve offset-free control in the presence of modeling errors. The CKF-MPSP method comprises three parts: nominal MPSP guidance, CKF modeling error estimation, and one-step modeling-error compensation. The nominal MPSP guidance generates a baseline guidance command ignoring modeling errors. The CKF modeling error estimation generates the guessed modeling error, and the one-step modeling-error compensation introduces an error compensation term in the baseline guidance command to maintain precision and stability.

The schematic of the CKF-MPSP guidance method is shown in Figure 2. The nominal MPSP guidance method is firstly used to obtain the nominal control command $\boldsymbol{U}_n$ according to initial states $\boldsymbol{X}_0$ and a guess value of control command $\boldsymbol{U}_0$. Then, regarding the disturbance as an initial guess value $\hat{\boldsymbol{d}}_0$, the one-step modeling-error compensation method is utilized to eliminate the modeling error's influence and generate the control command $\boldsymbol{U}$, which is substituted into the dynamic model to update the states $\boldsymbol{X}$. The CKF algorithm is utilized to generate the estimation of the states and the modeling error, which are denoted as $\hat{\boldsymbol{X}}$ and $\hat{\boldsymbol{d}}$, according to the measurement $\hat{\boldsymbol{y}}$ of the global navigation satellite system (GNSS). $\hat{\boldsymbol{X}}$, $\hat{\boldsymbol{d}}$, and $\boldsymbol{U}$ are used as initial values to calculate the control vector in the next guidance period. Repeating the process until hit, the ITACG is achieved.

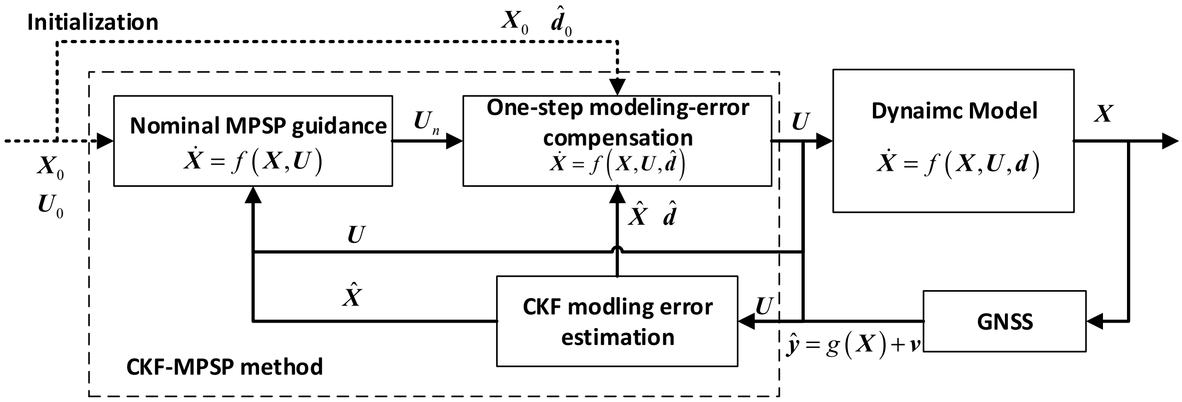


Figure 2. CKF-MPSP guidance method schematic.

4.1. Nominal MPSP Guidance

Ignoring the unknown modeling error, the MPSP method [20] is used to generate the baseline control commands. The nominal dynamic model is represented as

$$\dot{\boldsymbol{X}} = \boldsymbol{f}(\boldsymbol{X}, \boldsymbol{U}) = \begin{bmatrix} V\cos\theta\cos\psi_v \\ V\sin\theta \\ -V\cos\theta\sin\psi_v \\ -D - g\sin\theta \\ \frac{L-g\cos\theta}{V} \\ -\frac{Z}{V\cos\theta} \end{bmatrix}. \tag{3}$$

Discretizing Equation (3), the discretized dynamic model can be described as

$$\begin{cases} \boldsymbol{X}_{k+1} = \boldsymbol{F}_k(\boldsymbol{X}_k, \boldsymbol{U}_k) = \boldsymbol{X}_k + h\boldsymbol{f}_k(\boldsymbol{X}_k, \boldsymbol{U}_k) \\ \boldsymbol{Y}_k = \boldsymbol{C}\boldsymbol{X}_k \end{cases}, \tag{4}$$

where $k = 1, 2, 3, \cdots, N$ represents the time grids and h is the simulation step. The output matrix $\boldsymbol{C}$ is shown as

$$\boldsymbol{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5}$$

After settling the desired terminal time N and the desired terminal output $\boldsymbol{Y}_d = \begin{bmatrix} x_t & y_t & z_t & \theta_f & \psi_{vf} \end{bmatrix}^{\mathrm{T}}$, the prediction output vector $\boldsymbol{Y}_N$ can be obtained through Runge–

Kutta integration using initial states and previous control commands. Then, the updated control commands can be obtained according to the deviation between $\boldsymbol{Y}_N$ and $\boldsymbol{Y}_d$.

Denote the terminal output deviation as $\Delta \boldsymbol{Y}_N = \boldsymbol{Y}_N - \boldsymbol{Y}_d$. Expanding $\Delta \boldsymbol{Y}_N$ at $\boldsymbol{Y}_d$ and ignoring the high-order terms, we can obtain

$$\Delta \boldsymbol{Y}_N \cong \mathrm{d}\boldsymbol{Y}_N = \left(\frac{\partial \boldsymbol{Y}_N}{\partial \boldsymbol{X}_N}\right)\mathrm{d}\boldsymbol{X}_N. \tag{6}$$

According to Equation (4), we can determine $\partial \boldsymbol{Y}_N / \partial \boldsymbol{X}_N = \boldsymbol{C}$, and the following formula holds.

$$\mathrm{d}\boldsymbol{X}_{k+1} = \left(\frac{\partial \boldsymbol{F}_k}{\partial \boldsymbol{X}_k}\right)\mathrm{d}\boldsymbol{X}_k + \left(\frac{\partial \boldsymbol{F}_k}{\partial \boldsymbol{U}_k}\right)\mathrm{d}\boldsymbol{U}_k, \tag{7}$$

where $\mathrm{d}\boldsymbol{X}_k$ and $\mathrm{d}\boldsymbol{U}_k$ are the change in state and control at the k-th step, respectively.

The partial derivative of $\boldsymbol{F}_k$ with respect to $\boldsymbol{X}_k$ and $\boldsymbol{U}_k$ are shown as below:

$$\begin{aligned}\frac{\partial \boldsymbol{F}_k}{\partial \boldsymbol{X}_k} &= \boldsymbol{I}_{6\times6} + h\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{X}_k} \\ &= \boldsymbol{I}_{6\times6} + h\begin{bmatrix} 0 & 0 & 0 & \cos\theta\cos\psi_v & -V\sin\theta\cos\psi_v & -V\cos\theta\sin\psi_v \\ 0 & 0 & 0 & \sin\theta & V\cos\theta & 0 \\ 0 & 0 & 0 & -\cos\theta\sin\psi_v & V\sin\theta\sin\psi_v & -V\cos\theta\cos\psi_v \\ 0 & 0 & 0 & 0 & -g\cos\theta & 0 \\ 0 & 0 & 0 & -\frac{L-g\cos\theta}{V^2} & \frac{g\sin\theta}{V} & 0 \\ 0 & 0 & 0 & \frac{Z}{V^2\cos\theta} & -\frac{Z\sin\theta}{V\cos^2\theta} & 0 \end{bmatrix},\end{aligned} \tag{8}$$

$$\frac{\partial \boldsymbol{F}_k}{\partial \boldsymbol{U}_k} = h\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{U}_k} = h\begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{-C_x^{\alpha} q S_{ref}}{m} & \frac{-C_x^{\beta} q S_{ref}}{m} \\ \frac{C_y^{\alpha} q S_{ref}}{mV} & 0 \\ 0 & -\frac{C_z^{\beta} q S_{ref}}{mV\cos\theta} \end{bmatrix}, \tag{9}$$

where C_x^{α} is the partial derivative of the drag coefficient with respect to AOA, C_y^{α} is the derivative of the lift coefficient with respect to AOA, C_x^{β} is the partial derivative of the drag coefficient with respect to SSA, and C_z^{β} is the derivative of the lateral coefficient with respect to SSA. They can be obtained from the aerodynamic data.

Substituting Equation (7) into Equation (6), we can obtain

$$\mathrm{d}Y_N = \left(\frac{\partial \boldsymbol{Y}_N}{\partial \boldsymbol{X}_N}\right)\left[\left(\frac{\partial \boldsymbol{F}_{N-1}}{\partial \boldsymbol{X}_{N-1}}\right)\mathrm{d}\boldsymbol{X}_{N-1} + \left(\frac{\partial \boldsymbol{F}_{N-1}}{\partial \boldsymbol{U}_{N-1}}\right)\mathrm{d}\boldsymbol{U}_{N-1}\right]. \tag{10}$$

In Equation (10), $\mathrm{d}\boldsymbol{X}_{N-1}$ can be expressed as

$$\mathrm{d}\boldsymbol{X}_{N-1} = \left(\frac{\partial \boldsymbol{F}_{N-2}}{\partial \boldsymbol{X}_{N-2}}\right)\mathrm{d}\boldsymbol{X}_{N-2} + \left(\frac{\partial \boldsymbol{F}_{N-2}}{\partial \boldsymbol{U}_{N-2}}\right)\mathrm{d}\boldsymbol{U}_{N-2} \tag{11}$$

And $\mathrm{d}\boldsymbol{X}_{N-2}$ can be further expressed by $\mathrm{d}\boldsymbol{X}_{N-3}$ and $\mathrm{d}\boldsymbol{U}_{N-3}$. Repeating the above process until $\mathrm{d}\boldsymbol{X}_1$ and $\mathrm{d}\boldsymbol{U}_1$, it is clear that Equation (10) can be rewritten as

$$\mathrm{d}\boldsymbol{Y}_N = \boldsymbol{A}\mathrm{d}\boldsymbol{X}_1 + \boldsymbol{B}_1\mathrm{d}\boldsymbol{U}_1 + \boldsymbol{B}_2\mathrm{d}\boldsymbol{U}_2 + \cdots + \boldsymbol{B}_{N-1}\mathrm{d}\boldsymbol{U}_{N-1}, \tag{12}$$

where

$$\begin{aligned} &A \triangleq \frac{\partial \boldsymbol{Y}_N}{\partial \boldsymbol{X}_N}\frac{\partial \boldsymbol{F}_{N-1}}{\partial \boldsymbol{X}_{N-1}}\frac{\partial \boldsymbol{F}_{N-2}}{\partial \boldsymbol{X}_{N-2}}\cdots\frac{\partial \boldsymbol{F}_1}{\partial \boldsymbol{X}_1} \\ &\begin{cases} B_k \triangleq \frac{\partial \boldsymbol{Y}_N}{\partial \boldsymbol{X}_N}\frac{\partial \boldsymbol{F}_{N-1}}{\partial \boldsymbol{X}_{N-1}}\frac{\partial \boldsymbol{F}_{N-2}}{\partial \boldsymbol{X}_{N-2}}\cdots\frac{\partial \boldsymbol{F}_{k+1}}{\partial \boldsymbol{X}_{k+1}}\frac{\partial \boldsymbol{F}_k}{\partial \boldsymbol{U}_k}, & k = 1, 2, \cdots, N-2 \\ B_k \triangleq \frac{\partial \boldsymbol{Y}_N}{\partial \boldsymbol{X}_N}\frac{\partial \boldsymbol{F}_{N-1}}{\partial \boldsymbol{U}_{N-1}}, & k = N-1 \end{cases}\end{aligned}. \tag{13}$$

Since the initial states are with no errors ($\mathrm{d}\boldsymbol{X}_1 = 0$), the final output error is only decided by control commands as

$$\mathrm{d}\boldsymbol{Y}_N = \sum_{k=1}^{N-1} \boldsymbol{B}_k \mathrm{d}\boldsymbol{U}_k. \tag{14}$$

The purpose of guidance is to find a series of control commands $\boldsymbol{U}_k = \boldsymbol{U}_k^0 - \mathrm{d}\boldsymbol{U}_k$ $(k = 1, 2, \cdots, N)$ to make $\mathrm{d}\boldsymbol{Y}_N \to 0$, where $\boldsymbol{U}_k^0$ is the previous control history solution. It is worth noting that Equation (14) has $2 \times (N-1)$ unknowns and 5 equations. Usually, $2 \times (N-1) > 5$; thus, the solutions are not unique. To maximize guidance performance, set the following energy-optimal performance index and aim to minimize it.

$$J = \frac{1}{2} \sum_{k=1}^{N-1} \left(\boldsymbol{U}_k^0 - \mathrm{d}\boldsymbol{U}_k \right)^{\mathrm{T}} \boldsymbol{R}_k \left(\boldsymbol{U}_k^0 - \mathrm{d}\boldsymbol{U}_k \right), \tag{15}$$

where $\boldsymbol{R}_k$ is a positive definite weight coefficient matrix.

Equations (14) and (15) constitute a static optimization problem, whose solution at every time step $k = 1, 2, \cdots, N$, according to static optimization theory, is

$$\boldsymbol{U}_k^* = \boldsymbol{U}_k^0 - \mathrm{d}\boldsymbol{U}_k = \boldsymbol{R}_k^{-1} \boldsymbol{B}_k^{\mathrm{T}} \boldsymbol{A}_\lambda^{-1} (\mathrm{d}\boldsymbol{Y}_N - \boldsymbol{b}_\lambda), \tag{16}$$

where $\boldsymbol{A}_\lambda \triangleq -\sum_{k=1}^{N-1} \boldsymbol{B}_k \boldsymbol{R}_k^{-1} \boldsymbol{B}_k^T, \boldsymbol{b}_\lambda \triangleq \sum_{k=1}^{N-1} \boldsymbol{B}_k \boldsymbol{U}_k^0$.

4.2. Modeling Error Estimation Based on CKF

The MPSP guidance method highly relies on modeling accuracy because of the integral prediction. However, the realistic model inevitably has unknown modeling errors or external disturbances. It has been pointed out in the literature [28] that, in the presence of model mismatch, the MPSP method cannot realize the desired terminal states. Estimating and compensating for modeling errors are common ways to achieve offset-free terminal state control. This section uses the CKF algorithm to estimate states and modeling errors simultaneously for subsequent compensation.

To estimate the modeling errors, consider them as constants and extend them to states. The dynamic model (1) can be rewritten as

$$\dot{\boldsymbol{X}}^E = \boldsymbol{f}_E\left(\boldsymbol{X}^E\right) = \begin{bmatrix} V\cos\theta\cos\psi_v \\ V\sin\theta \\ -V\cos\theta\sin\psi_v \\ -D - g\sin\theta + d_x \\ \frac{L - g\cos\theta}{V} + d_y \\ -\frac{Z}{V\cos\theta} + d_z \\ 0 \\ 0 \\ 0 \end{bmatrix} + \boldsymbol{w}, \tag{17}$$

where $\boldsymbol{X}^E = \left[x, y, z, V, \theta, \psi_v, d_x, d_y, d_z\right]^{\mathrm{T}}$ is the expansion state vector, $\boldsymbol{w}$ is Gaussian-distributed process noise, and $E\left[\boldsymbol{w}\boldsymbol{w}^T\right] = \boldsymbol{Q}$.

During flight, GNSS measures the missile's motion in real time. So, the measurement equations can be denoted as

$$\hat{\boldsymbol{y}}^E = \boldsymbol{X}^E + \boldsymbol{v}, \tag{18}$$

where $\boldsymbol{v}$ is Gaussian-distributed measurement noise and $E\left[\boldsymbol{v}\boldsymbol{v}^T\right] = \boldsymbol{R}$. $\hat{\boldsymbol{y}}^E = \left[\hat{x}, \hat{y}, \hat{z}, \hat{V}, \hat{\theta}, \hat{\psi}_v, \hat{d}_x, \hat{d}_y, \hat{d}_z\right]^{\mathrm{T}}$ is the expansion output vector.

Discretizing Equations (17) and (18), we can obtain the nonlinear filter model:

$$\begin{cases} \boldsymbol{X}_k^E = \boldsymbol{F}_E\left(\boldsymbol{X}_{k-1}^E\right) + \boldsymbol{w}_{k-1} \\ \boldsymbol{Y}_k^E = \boldsymbol{X}_k^E + \boldsymbol{v}_{k-1} \end{cases}, \tag{19}$$

where $\boldsymbol{F}_E\left(\boldsymbol{X}_{k-1}^E\right) = \boldsymbol{X}_{k-1}^E + h\boldsymbol{f}_E\left(\boldsymbol{X}_{k-1}^E\right)$.

The CKF algorithm consists of two procedures: Time Update and Measurement Update [29]. Combined with the filter model, the CKF algorithm process is shown below.

4.2.1. Time Update

Assume at time k that the posterior probability density function $p\left(\boldsymbol{X}_{k-1}^E | \boldsymbol{y}_{k-1}\right) = N\left(\hat{\boldsymbol{X}}_{k-1}^E, \boldsymbol{P}_{k-1}\right)$ is known. Denote the Cholesky factorization of the error covariance $\boldsymbol{P}_{k-1}$ as $\boldsymbol{S}_{k-1}$.

$$\boldsymbol{P}_{k-1} = \boldsymbol{S}_{k-1}\boldsymbol{S}_{k-1}^{\mathrm{T}}. \tag{20}$$

Calculate the cubature points $\chi_{k-1}^{(i)}$ based on the third-degree cubature rule:

$$\chi_{k-1}^{(i)} = \hat{\boldsymbol{X}}_{k-1}^E + \boldsymbol{S}_{k-1}\boldsymbol{\xi}_i, \quad i = 1, 2, \cdots, 2n, \tag{21}$$

where n is the dimension of states. $\boldsymbol{\xi}_i = \sqrt{n}[\boldsymbol{1}]_i$ is the basic cubature point set. The point set $[\boldsymbol{1}]$ is defined as

$$[\boldsymbol{1}] = \underbrace{\left\{ \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}_{n\times 1}, \begin{bmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{bmatrix}, \cdots, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{bmatrix}, \begin{bmatrix} -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ -1 \\ \vdots \\ 0 \end{bmatrix}, \cdots, \begin{bmatrix} 0 \\ 0 \\ \vdots \\ -1 \end{bmatrix} \right\}}_{2n}, \tag{22}$$

and $[\boldsymbol{1}]_i$ represents the i-th column vector in $[\boldsymbol{1}]$.

Calculate the one-step prediction at time k and its error covariance:

$$\chi_{k|k-1}^{*(i)} = \boldsymbol{F}_E\left(\chi_{k-1}^{(i)}\right), \tag{23}$$

$$\hat{\boldsymbol{X}}_{k|k-1}^E = \frac{1}{2n}\sum_{i=1}^{2n} \chi_{k|k-1}^{*(i)}, \tag{24}$$

$$P_{k|k-1} = \frac{1}{2n}\sum_{i=1}^{2n} \left[\chi_{k|k-1}^{*(i)} - \hat{\boldsymbol{X}}_{k|k-1}^E\right]\left[\chi_{k|k-1}^{*(i)} - \hat{\boldsymbol{X}}_{k|k-1}^E\right]^{\mathrm{T}} + \boldsymbol{Q}_{k-1}. \tag{25}$$

4.2.2. Measurement Update

Calculate the cubature points for Measurement Update, and then calculate measurement prediction:

$$\boldsymbol{P}_{k|k-1} = \boldsymbol{S}_{k|k-1}\boldsymbol{S}_{k|k-1}^{\mathrm{T}}, \tag{26}$$

$$\chi_{k|k-1}^{(i)} = \hat{\boldsymbol{X}}_{k|k-1}^E + \boldsymbol{S}_{k|k-1}\boldsymbol{\xi}_i, \quad i = 1, 2, \cdots, 2n, \tag{27}$$

$$\hat{\boldsymbol{y}}_{k|k-1}^E = \frac{1}{2n}\sum_{i=1}^{2n} \chi_{k|k-1}^{(i)}. \tag{28}$$

Calculate the innovation covariance matrix $\boldsymbol{P}_{xy}$ and the cross-covariance matrix $\boldsymbol{P}_{yy}$:

$$\boldsymbol{P}_{xy} = \frac{1}{2n}\sum_{i=1}^{2n}\left[\chi_{k|k-1}^{(i)} - \hat{\boldsymbol{X}}_{k|k-1}^{E}\right]\left[\chi_{k|k-1}^{(i)} - \hat{\boldsymbol{y}}_{k|k-1}^{E}\right]^{\mathrm{T}}, \tag{29}$$

$$\boldsymbol{P}_{yy} = \frac{1}{2n}\sum_{i=1}^{2n}\left[\chi_{k|k-1}^{(i)} - \hat{\boldsymbol{y}}_{k|k-1}^{E}\right]\left[\chi_{k|k-1}^{(i)} - \hat{\boldsymbol{y}}_{k|k-1}^{E}\right]^{\mathrm{T}} + \boldsymbol{R}_k. \tag{30}$$

Calculate the Kalman gain:

$$\boldsymbol{K}_k = \boldsymbol{P}_{xy}\boldsymbol{P}_{yy}^{-1}. \tag{31}$$

Estimate the updated state and the corresponding error covariance:

$$\hat{\boldsymbol{X}}_k^E = \hat{\boldsymbol{X}}_{k|k-1}^E + \boldsymbol{K}_k\left(\boldsymbol{y}_k^E - \hat{\boldsymbol{y}}_{k|k-1}^E\right), \tag{32}$$

$$\boldsymbol{P}_k = \boldsymbol{P}_{k|k-1} - \boldsymbol{K}_k\boldsymbol{P}_{yy}\boldsymbol{K}_k^{\mathrm{T}}. \tag{33}$$

4.3. One-Step Modeling-Error Compensation

After estimating the modeling errors, compensate for the effect of modeling errors on the system by attaching an additional control term $\Delta\boldsymbol{U}_k$ to the MPSP optimal command $\boldsymbol{U}_k^*$ at time k. Denoting the estimation modeling error vector as $\hat{\boldsymbol{d}}_k = \left[\hat{d}_{xk}, \hat{d}_{yk}, \hat{d}_{zk}\right]^{\mathrm{T}}$, the disturbed system model can be described as

$$\begin{cases} \widetilde{\boldsymbol{X}}_{k+1} = \widetilde{\boldsymbol{X}}_k + h\boldsymbol{f}_k\left(\widetilde{\boldsymbol{X}}_k, \widetilde{\boldsymbol{U}}_k, \hat{\boldsymbol{d}}_k\right) \\ \widetilde{\boldsymbol{Y}}_{k+1} = \boldsymbol{C}\widetilde{\boldsymbol{X}}_{k+1} \end{cases}. \tag{34}$$

At time k, denote $\boldsymbol{Y}_{k+1}$ as the output vector obtained by a one-step calculation with current state $\boldsymbol{X}_k$ and the MPSP control command $\boldsymbol{U}_k^*$. The objective of modeling-error compensation is to generate a modified control command $\widetilde{\boldsymbol{U}}_k = \boldsymbol{U}_k^* + \Delta\boldsymbol{U}_k$ making $\widetilde{\boldsymbol{Y}}_{k+1} \to \boldsymbol{Y}_{k+1}$.

Denote the output error as

$$\Delta\boldsymbol{Y}_{k+1} = \widetilde{\boldsymbol{Y}}_{k+1} - \boldsymbol{Y}_{k+1}. \tag{35}$$

Substituting Equations (4) and (34) into (35) and because of $\widetilde{\boldsymbol{X}}_k = \boldsymbol{X}_k$, we can obtain

$$\Delta\boldsymbol{Y}_{k+1} = h\boldsymbol{C}\left[\boldsymbol{f}_k(\boldsymbol{X}_k, \boldsymbol{U}_k^* + \Delta\boldsymbol{U}_k, \hat{\boldsymbol{d}}_k) - \boldsymbol{f}_k(\boldsymbol{X}_k, \boldsymbol{U}_k^*)\right]. \tag{36}$$

Expanding $\boldsymbol{f}_k\left(\boldsymbol{X}_k, \boldsymbol{U}_k^* + \Delta\boldsymbol{U}_k, \hat{d}_k\right)$ and ignoring high-order terms, we can obtain

$$\boldsymbol{f}_k(\boldsymbol{X}_k, \boldsymbol{U}_k^* + \Delta\boldsymbol{U}_k, \hat{\boldsymbol{d}}_k) \cong \boldsymbol{f}_k(\boldsymbol{X}_k, \boldsymbol{U}_k^*) + \frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{U}_k}\Delta\boldsymbol{U}_k + \frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{d}_k}\hat{\boldsymbol{d}}_k. \tag{37}$$

Substituting Equation (37) into (36), we can obtain

$$\Delta\boldsymbol{Y}_{k+1} = h\boldsymbol{C}\left[\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{U}_k}\Delta\boldsymbol{U}_k + \frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{d}_k}\hat{\boldsymbol{d}}_k\right]. \tag{38}$$

The compensation term is desired to make the output error zero. According to Equation (38), we can obtain the desired additional control term as

$$\Delta\boldsymbol{U}_k = -\left(\boldsymbol{C}\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{U}_k}\right)^{-1}\left(\boldsymbol{C}\frac{\partial \boldsymbol{f}_k}{\partial \boldsymbol{d}_k}\hat{\boldsymbol{d}}_k\right). \tag{39}$$

The modified control command at time k is

$$\widetilde{\boldsymbol{U}}_k = \boldsymbol{U}_k^* + \Delta \boldsymbol{U}_k. \quad (40)$$

4.4. CKF-MPSP Terminal Guidance Process

Considering the above, the CKF-MPSP guidance process is summarized below (Algorithm 1).

Algorithm 1: CKF-MPSP Terminal Guidance Scheme.

INPUT: current time k, desired terminal time N, desired terminal output $\boldsymbol{Y}_d$, current states $\boldsymbol{X}_k$, guessed control command $\left[\boldsymbol{U}_k^0, \boldsymbol{U}_{k+1}^0, \cdots, \boldsymbol{U}_{N-1}^0\right]$, estimated modeling error $\hat{\boldsymbol{d}}_k$, CKF initial values $\hat{\boldsymbol{X}}_{k-1}^E$, $\hat{\boldsymbol{P}}_{k-1}$.
1: **while** current time k is no larger than N, **do**
2: **while** terminal output deviation $\Delta \boldsymbol{Y}_N$ is larger than tolerance value $\boldsymbol{\varepsilon}$, **do**
3: predict the terminal output vector $\boldsymbol{Y}_N$ through Runge-Kutta integration with the nominal dynamic model (4), the current states $\boldsymbol{X}_k$ and the guessed control command $\left[\boldsymbol{U}_k^0, \boldsymbol{U}_{k+1}^0, \cdots, \boldsymbol{U}_{N-1}^0\right]$.
4: calculate terminal output deviation $\Delta \boldsymbol{Y}_N = \boldsymbol{Y}_N - \boldsymbol{Y}_d$.
5: calculate matrices $\left[\boldsymbol{B}_k, \boldsymbol{B}_{k+1}, \cdots, \boldsymbol{B}_{N-1}\right]$ according to Equation (13).
6: calculate the optimal control command $\left[\boldsymbol{U}_k^*, \boldsymbol{U}_{k+1}^*, \cdots, \boldsymbol{U}_{N-1}^*\right]$ according to Equation (16).
7: take $\left[\boldsymbol{U}_{k+1}^*, \cdots, \boldsymbol{U}_{N-1}^*\right]$ as the new guessed control command.
8: **end while**.
9: calculate the one-step output $\widetilde{\boldsymbol{Y}}_{k+1}$ with the disturbed system model (34) in presence of the estimated modeling error $\hat{\boldsymbol{d}}_k$
10: calculate the modified control command $\widetilde{\boldsymbol{U}}_k$ at time k, according to Equations (39) and (40).
11: substitute $\widetilde{\boldsymbol{U}}_k$ into the realistic dynamic model (1) and obtain the updated state $\boldsymbol{X}_{k+1}$.
12: estimate the filter state $\hat{\boldsymbol{X}}_k^E$, error covariance $\hat{\boldsymbol{P}}_k$, and modeling error $\hat{\boldsymbol{d}}_{k+1}$, using CKF algorithm (20)~(33).
13: time update, $k = k + 1$.
14: **end while**

Remark 1. *The MPSP guidance method takes the desired impact time as the terminal time N of static planning. The target's position and impact angles are regarded as desired terminal output $\boldsymbol{Y}_d$. Making the terminal output deviation $\Delta \boldsymbol{Y}_N$ no larger than tolerance value $\boldsymbol{\varepsilon}$ by iterating $\boldsymbol{U}^*$, the impact time and angle constraints can be satisfied simultaneously.*

Remark 2. *The MPSP guidance method relies on initial guessed control commands $\left[\boldsymbol{U}_k^0, \boldsymbol{U}_{k+1}^0, \cdots, \boldsymbol{U}_{N-1}^0\right]$. The guessed control commands are quickly generated through some simple guidance laws in common. In this paper, the traditional PN guidance law is used to obtain the initial guessed control commands. Usually, the impact time of PN, represented by the symbol N_P, is different from the desired time N. For the MPSP algorithm, N_P must be no less than N, so the outputs at time N can be predicted. In this paper, a protection mechanism is introduced to make sure the MPSP algorithm normally runs even if N_P is less than N: Let $\boldsymbol{U}_k^0 = \boldsymbol{U}_{N_p-1}^0$, for $N_P - 1 < k \leq N - 1$.*

5. Simulations and Results

In this section, several numerical simulations are carried out to evaluate the performance of the proposed terminal guidance method in the presence of modeling errors. A three-dimensional guidance scenario of a missile attacking a stationary target is constructed. The initial simulation conditions are listed below (Table 1):

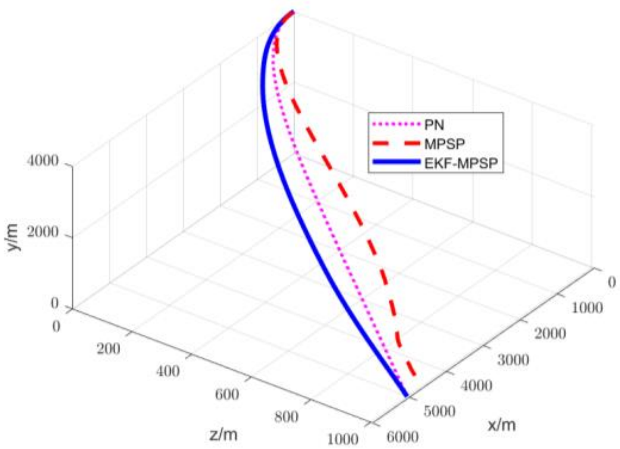
Table 1. Simulation initial conditions.

Parameters	Values
Missile's initial velocity V	200 m/s
Missile's initial path angle θ	0°
Missile's initial azimuth angle ψ_v	0°
Missile's initial position (x, y, z)	(0 m, 4000 m, 0 m)
Target's position (x_t, y_t, z_t)	(5000 m, 0 m, 1000 m)

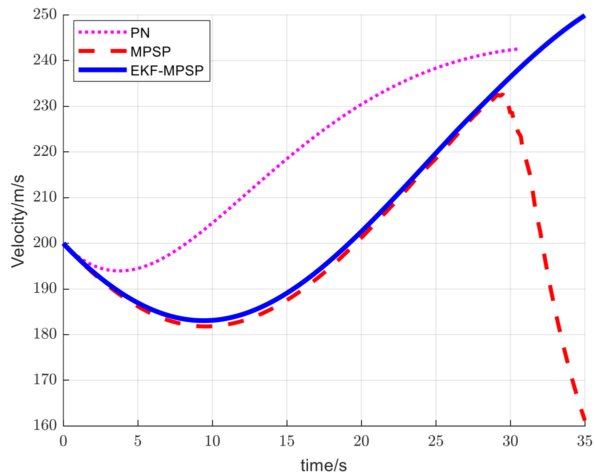
Besides achieving precision arrival, the impact time and impact angles are also required. The desired impact time is settled as 35 s, and the desired terminal path angle and azimuth angle are settled as $-80°$ and $-50°$, respectively. The dynamic modeling errors in Equation (1) are settled as $d_x = -0.4\sin(t/400)\cos(t/400)$, $d_y = 1$, and $d_z = 2.5\cos(\pi t/200)$. The measurement errors from the GNSS system are assumed to be normally distributed. The position error, velocity error, and acceleration error are settled to be 10 m (3σ), 1 m/s (3σ), and 0.1 m/s^2 (3σ), respectively.

For the MPSP algorithm, the guessed control commands $[\boldsymbol{U}_k^0, \boldsymbol{U}_{k+1}^0, \cdots, \boldsymbol{U}_{N-1}^0]$ are needed. The traditional PN [30] is used to produce the initial values of $[\boldsymbol{U}_k^0, \boldsymbol{U}_{k+1}^0, \cdots, \boldsymbol{U}_{N-1}^0]$ in this paper, and the navigation ratio is settled as 6. In addition, a comparison with the MPSP guidance method presented in [21] is provided to validate the superiority of the method. The end condition for PN simulation is that the missile reaches the target. And the end conditions for the other two simulations are that the simulation times reach the desired impact time, which is 35 s on this occasion. For MPSP and CKF-MPSP methods, the tolerance value vector is settled as $\boldsymbol{\varepsilon} = [1\text{m}, 1\text{m}, 1\text{m}, 0.1°, 0.1°]^{\text{T}}$. The simulation results are shown below.

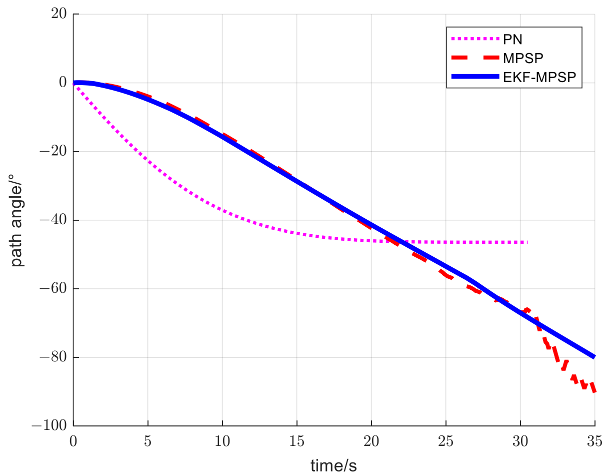
To evaluate the guidance accuracy, some crucial parameters of the three methods shown in Figure 3 are provided in Tables 2 and 3, which include the terminal miss distance, terminal velocity, terminal path angle, terminal azimuth angle, and impact time.



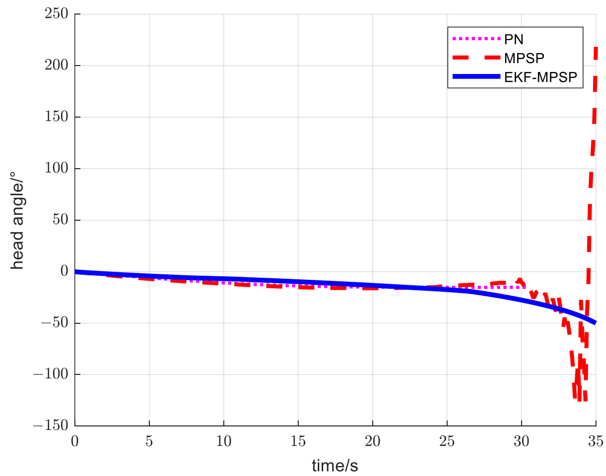
(a)



(b)

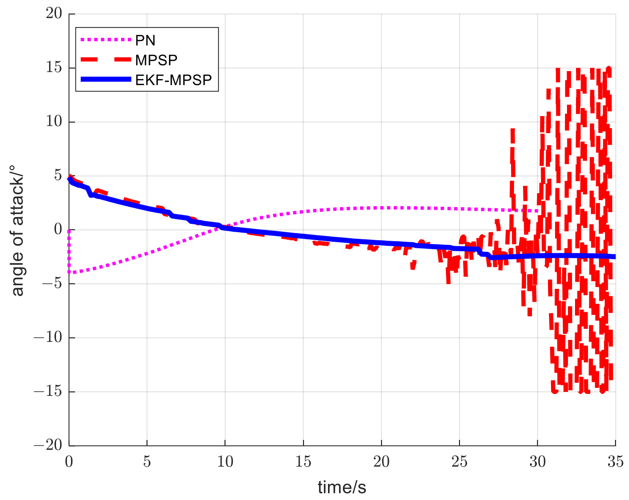


(c)

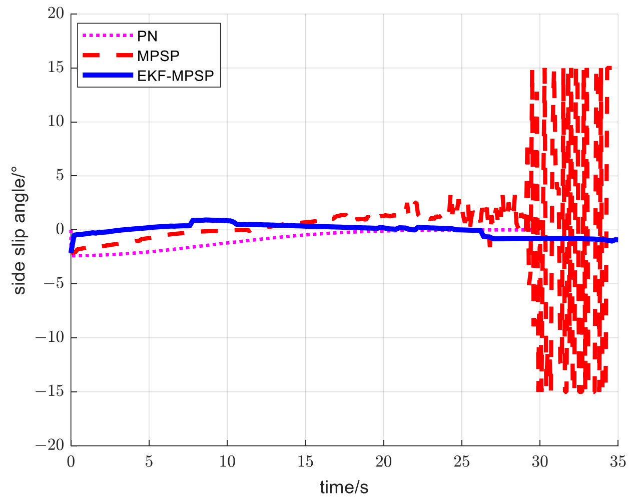


(d)

Figure 3. *Cont.*



(e)



(f)

Figure 3. Comparative simulation results of three methods: (**a**) three-dimensional trajectory; (**b**) missile velocity profiles; (**c**) path angle profiles; (**d**) azimuth angle profiles; (**e**) angle-of-attack profiles; and (**f**) side slip angle profiles.

Table 2. Simulation results of PN method.

Method	Miss Distance (m)	Velocity (m/s)	Path Angle (°)	Azimuth Angle (°)	Impact Time (s)
PN	0.26	242.56	−46.42	−15.23	30.5

Table 3. Simulation results of MPSP and CKF-MPSP methods at the desired impact time (35 s).

Method	Miss Distance (m)	Velocity (m/s)	Path Angle (°)	Azimuth Angle (°)
MPSP	289.73	161.18	−90.24	218.31
CKF-MPSP	0.29	249.90	−79.99	−50.00

It is obvious that the CKF-MPSP method has good accuracy for miss distance while strictly constraining the terminal path angle, azimuth angle, and impact time. The PN method leads to the minimum miss distance. However, it cannot consider impact time and angle constraints. Because of the dynamic modeling errors, the MPSP method cannot find a feasible solution, which leads to a significant guidance error. Affected by modeling errors, the MPSP method's velocity is lower than the CKF-MPSP's at every identical moment, which is also the main reason for the MPSP method not reaching the destination. The CKF-MPSP method estimates the modeling errors with high accuracy. Referring to Figure 4, the estimation error of modeling errors is no larger than 0.04 m/s^2. The influence of modeling errors can be reduced by compensating for the nominal command acceleration. The simulation results illustrate the effectiveness and superiority of the CKF-MPSP method in the presence of modeling errors.

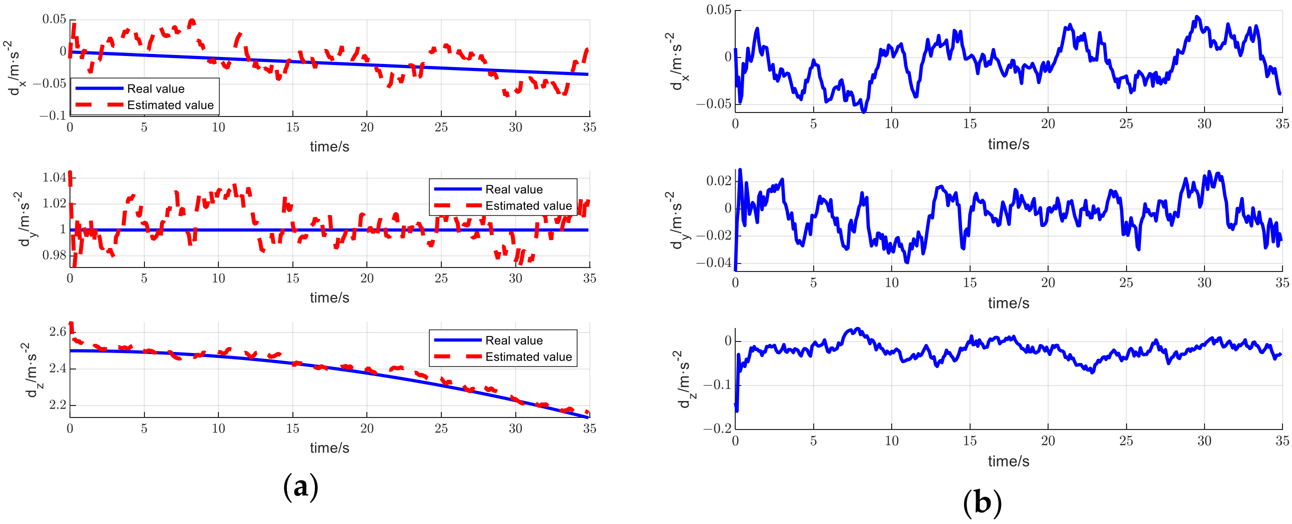


Figure 4. CKF estimation results: (**a**) CKF estimation of modeling errors and (**b**) estimation error of modeling errors.

Based on the above simulation, the influence of atmospheric density deviation, aerodynamic parameter deviation, and the random variation in the dynamic modeling errors are further considered. The deviations are assumed to be normally distributed, and their values are 10% (3σ). The results of 200 Monte Carlo simulations are as follows.

The key indexes in Figure 5 is summarized in Table 4. According to Table 4, the miss distances are no larger than 5.56 m and the average impact angle errors are 0.055° and 0.077°, respectively. The missile maintains high guidance accuracy in the presence of disturbances. The simulation results illustrate that the proposed guidance method has strong robustness.

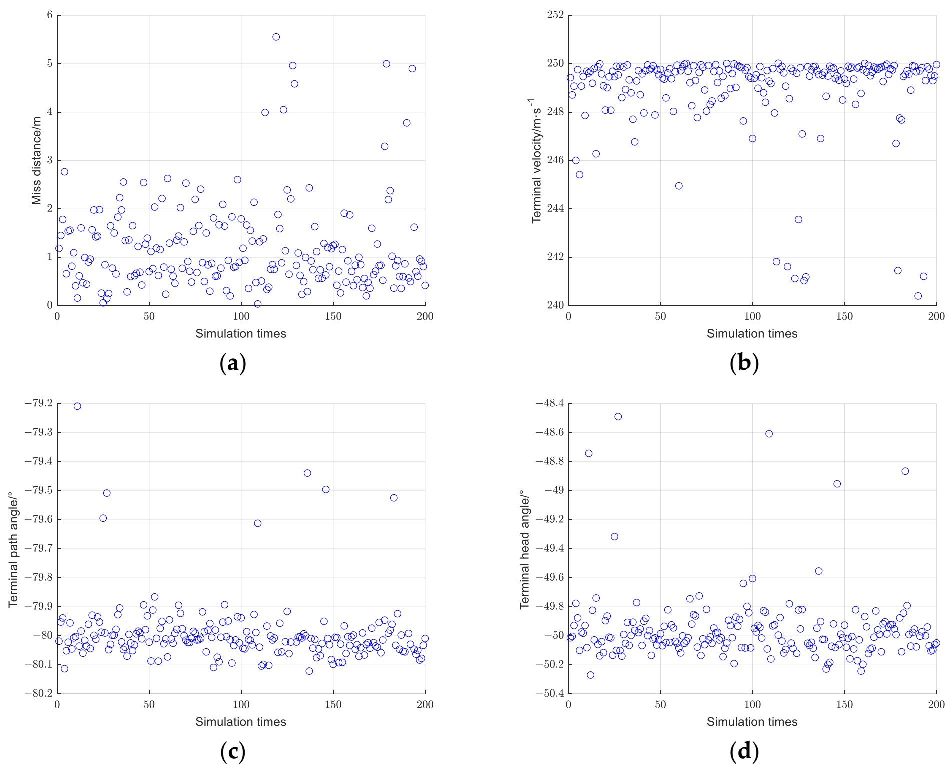


Figure 5. Monte Carlo simulation results: (**a**) terminal miss distance; (**b**) terminal velocity; (**c**) path angle profiles; and (**d**) azimuth angle profiles.

Table 4. Monte Carlo simulation results.

Method	Miss Distance (m)	Velocity (m/s)	Path Angle (°)	Azimuth Angle (°)
Average	1.21	248.92	−79.95	−49.93
Maximum value	5.56	250.11	−79.21	−48.49
Minimum value	0.03	240.40	−80.12	−50.26
Standard error	0.95	1.84	0.16	0.34

6. Conclusions

ITACG is a vital field for missiles because it can improve destructive effects and hit weak parts of time-sensitive targets, and make it possible for multiple missiles to attack a target simultaneously. Studying ITACG can effectively enhance the combat effectiveness of missiles.

In this paper, an ITACG is proposed based on the MPSP algorithm. By taking the desired impact time and angles as terminal conditions, the guidance method can satisfy these constraints simultaneously. Furthermore, to eliminate the influence of modeling errors on prediction, the CKF algorithm is used for error estimation, and a compensation scheme is designed. The proposed guidance method considers the missile's dynamic model instead of a constant-velocity model. Meanwhile, the modeling errors are estimated and compensated. Thus, this method is more practically significant. A terminal guidance scenario is settled, and the PN method, MPSP method, and CKF-MPSP method are used for simulation in the presence of modeling errors. According to the simulation results, the CKF-MPSP method can achieve impact time and angle constraint guidance, and maintain high accuracy within the influence of modeling errors. Furthermore, the Monte Carlo simulation is conducted, considering the influence of atmospheric density deviations, aerodynamic parameter deviations, and random variations in the dynamic modeling errors. According to the simulation results, the miss distances are no larger than 5.56 m, and the average impact angle errors are 0.055° and 0.077°, respectively. The missile maintains high guidance accuracy in the presence of disturbances. Comprehensively, the simulation results illustrate that the proposed CKF-MPSP guidance method has high precision and strong robustness.

It should be acknowledged that this article still has some limitations. The research of this paper is mainly focused on the ITACG against stationary targets. For moving targets, it is also necessary to introduce their motion models into the guidance method. However, the accurate estimation of the moving targets' motion is still a difficult problem, because of insufficient target information and potential maneuvering. Motion model estimation and MPSP-based guidance for moving targets remain to be researched in the future.

Author Contributions: Conceptualization, Z.X. and J.P.; methodology, Z.X.; software, Z.X.; validation, J.P. and C.W.; writing—original draft, Z.X.; writing—review and editing, J.P., C.W. and Y.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China, grant number U2241215.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Yu, H.; Dai, K.; Li, H.; Zou, Y.; Ma, X.; Ma, S.; Zhang, H. Three-dimensional adaptive fixed-time cooperative guidance law with impact time and angle constraints. *Aerosp. Sci. Technol.* **2022**, *123*, 107450. [CrossRef]
2. Zhang, S.; Guo, Y.; Liu, Z.; Wang, S.; Hu, X. Finite-time cooperative guidance strategy for impact angle and time control. *IEEE Trans. Aerosp. Electron. Syst.* **2020**, *57*, 806–819. [CrossRef]
3. Chen, Z.; Chen, W.; Liu, X.; Cheng, J. Three-dimensional fixed-time robust cooperative guidance law for simultaneous attack with impact angle constraint. *Aerosp. Sci. Technol.* **2021**, *110*, 106523. [CrossRef]
4. Ma, F.; Wu, Y.; Wang, S.; Yang, X.; Hua, Y. Three-dimensional adaptive fixed-time guidance law against maneuvering targets with impact angle constraints and control input saturation. *Trans. Inst. Meas. Control* **2022**, *44*, 1579–1598. [CrossRef]

5. Lin, M.; Ding, X.; Wang, C.; Liang, L.; Wang, J. Three-dimensional fixed-time cooperative guidance law with impact angle constraint and prespecified impact time. *IEEE Access* **2021**, *9*, 29755–29763. [CrossRef]
6. Wang, Z.; Fang, Y.; Fu, W.; Wu, Z.; Wang, M. Cooperative guidance laws against highly maneuvering target with impact time and angle. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2022**, *236*, 1006–1016. [CrossRef]
7. Jing, L.; Wei, C.; Zhang, L.; Cui, N. Cooperative guidance law with predefined-time convergence for multimissile systems. *Math. Probl. Eng.* **2021**, *2021*, 9940240. [CrossRef]
8. Chen, X.; Wang, J. Optimal control based guidance law to control both impact time and impact angle. *Aerosp. Sci. Technol.* **2019**, *84*, 454–463. [CrossRef]
9. Kim, T.H.; Lee, C.H.; Jeon, I.S.; Tahk, M.J. Augmented polynomial guidance with impact time and angle constraints. *IEEE Trans. Aerosp. Electron. Syst.* **2013**, *49*, 2806–2817. [CrossRef]
10. Zhao, Y.; Sheng, Y.; Liu, X. Trajectory reshaping based guidance with impact time and angle constraints. *Chin. J. Aeronaut.* **2016**, *29*, 984–994. [CrossRef]
11. Kang, S.; Tekin, R.; Holzapfel, F. Generalized impact time and angle control via look-angle shaping. *J. Guid. Control Dyn.* **2019**, *42*, 695–702. [CrossRef]
12. Hou, Z.; Liu, L.; Wang, Y. Time-to-go estimation for terminal sliding mode based impact angle constrained guidance. *Aerosp. Sci. Technol.* **2017**, *71*, 685–694. [CrossRef]
13. Hou, Z.; Yang, Y.; Liu, L.; Wang, Y. Terminal sliding mode control based impact time and angle constrained guidance. *Aerosp. Sci. Technol.* **2019**, *93*, 105142. [CrossRef]
14. Chen, X.; Wang, J. Two-stage guidance law with impact time and angle constraints. *Nonlinear Dyn.* **2019**, *95*, 2575–2590. [CrossRef]
15. Zhang, Y.; Ma, G.; Liu, A. Guidance law with impact time and impact angle constraints. *Chin. J. Aeronaut.* **2013**, *26*, 960–966. [CrossRef]
16. Yan, X.; Zhu, J.; Kuang, M.; Yuan, X. A computational-geometry-based 3-dimensional guidance law to control impact time and angle. *Aerosp. Sci. Technol.* **2020**, *98*, 105672. [CrossRef]
17. Majumder, K.; Kumar, S.R. Sliding mode–based simultaneous control of impact angle and impact time. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2022**, *236*, 1269–1281. [CrossRef]
18. Liu, X.; Li, G. Adaptive Sliding Mode Guidance with Impact Time and Angle Constraints. *IEEE Access* **2020**, *8*, 26926–26932. [CrossRef]
19. Wang, Z.; Hu, Q.; Han, T.; Xin, M. Two-Stage Guidance Law with Constrained Impact via Circle Involute. *IEEE Trans. Aerosp. Electron. Syst.* **2021**, *57*, 1301–1316. [CrossRef]
20. Padhi, R.; Kothari, M. Model predictive static programming: A computationally efficient technique for suboptimal control design. *Int. J. Innov. Comput. Inf. Control* **2009**, *5*, 399–411.
21. Oza, H.B.; Padhi, R. Impact-Angle-Constrained Suboptimal Model Predictive Static Programming Guidance of Air-to-Ground Missiles. *J. Guid. Control Dyn.* **2012**, *35*, 153–164. [CrossRef]
22. Maity, A.; Oza, H.B.; Padhi, R. Generalized model predictive static programming and angle-constrained guidance of air-to-ground missiles. *J. Guid. Control Dyn.* **2014**, *37*, 1897–1913. [CrossRef]
23. Mondal, S.; Padhi, R. Angle-constrained terminal guidance using quasi-spectral model predictive static programming. *J. Guid. Control Dyn.* **2018**, *41*, 783–791. [CrossRef]
24. He, X.; Chen, W.; Yang, L. Suboptimal impact-angle-constrained guidance law using linear pseudospectral model predictive spread control. *IEEE Access* **2020**, *8*, 102040–102050. [CrossRef]
25. Zhou, C.; Yan, X.; Tang, S. Generalized quasi-spectral model predictive static programming method using Gaussian quadrature collocation. *Aerosp. Sci. Technol.* **2020**, *106*, 106134. [CrossRef]
26. Mondal, S.; Padhi, R. Constrained Quasi-Spectral MPSP With Application to High-Precision Missile Guidance with Path Constraints. *J. Dyn. Syst. Meas. Control* **2021**, *143*, 031001. [CrossRef]
27. Liu, X.; Li, S.; Xin, M. Pseudospectral Convex Optimization Based Model Predictive Static Programming for Constrained Guidance. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 2232–2244. [CrossRef]
28. Mathavaraj, S.; Padhi, R. Unscented MPSP for optimal control of a class of uncertain nonlinear dynamic systems. *J. Dyn. Syst. Meas. Control* **2019**, *141*, 65001. [CrossRef]
29. Arasaratnam, I.; Haykin, S. Cubature Kalman filters. *IEEE Trans. Autom. Control* **2009**, *54*, 1254–1269. [CrossRef]
30. Becker, K. Closed-form solution of pure proportional navigation. *IEEE Trans. Aerosp. Electron. Syst.* **1990**, *26*, 526–533. [CrossRef]