ASME Paper Title: Constrained Quasi-Spectral MPSP With Application to High-Precision Missile Guidance With Path Constraints

Authors: Sabyasachi Mondal, Radhakant Padhi

ASME Journal Title: Journal of Dynamic Systems, Measurement, and Control

Volume/Issue: 143/3

Date of Publication (VOR* Online): 15/10/2020

ASME Digital Collection URL: https://asmedigitalcollection.asme.org/dynamicsystems/article-abstract/143/3/031001/1087168/Constrained-Quasi-Spectral-MPSP-With-Application

DOI: https://doi.org/10.1115/1.4048488

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019; Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488
Copyright (c) 2020 by ASME

Journal of Dynamic Systems, Measurement, and Control
Volume 143, Issue 3, March 2021, Article Number 031001
DOI: 10.1115/1.4048488

Constrained Quasi-Spectral MPSP with Application to High-Precision Missile Guidance with Path Constraints

Sabyasachi Mondal
School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, UK *

Radhakant Padhi
Department of Aerospace Engineering, Indian Institute of Science, Bangalore,India †

ABSTRACT

This paper extends the recently-developed Quasi-Spectral Model Predictive Static Programming (QS-MPSP) to include state and control path-constraints and yet retain its computational efficiency. This is achieved by (i) formulating the entire problem in the control variables alone by (a) converting the system dynamics to an equivalent algebraic constraint and (b) converting the state constraints to equivalent control constraints, both of which is done by manipulating the system dynamics, (ii) representing the control variables in Quasi-spectral form, which makes the number of free-variables independent of time-grids and (iii) using a computationally-efficient optimization algorithm to solve this low-dimensional problem. This generic computationally-efficient technique is utilized next as an effective lead angle, and lateral acceleration constrained optimal missile guidance to intercept incoming high-speed ballistic targets with high precision successfully. Both of these constraints, as well as near-zero miss-distance, are of high practical significance for this challenging problem. Extensive three-dimensional simulation studies show the effectiveness of the newly-proposed constrained QS-MPSP guidance algorithm. Six degree-of-freedom simulation studies have also been carried out using autopilot in the loop to validate the results more realistically.

1 Introduction

Optimal control offers a very powerful framework to solve a host of challenging control synthesis problems in practice as it has the capability to handle various types of constraints on states and control variables. In addition to the state equation constraint, additional constraints may appear during the control operation time (path constraints) and initial and/or final points of time (terminal constraints). They can be imposed on state or control or both. Optimal control theory addresses every possible combination of constraints successfully. Usually, these constraints are augmented to different kinds of problems such as regulator problems, tracking problems and so on. Quite naturally it finds its applications in many different engineering fields such as aerospace, electrical, mechanical, robotics, process control and biomedical, to name a few. Many classic books have been written on optimal control and its capability to solve various classes of problems in the past (see, e.g. [1–3]). Some recent books have also appeared containing a few recently-developed algorithms and several challenging applications [4–7].

Solutions methods of optimal control problems are broadly divided into two categories, e.g., a) indirect method and b) direct method. The indirect method follows the calculus of variations approach and results in the state equation, adjoint equation, control equation, and the transversality condition [3], thereby leading to a two-point boundary value problem [1]. However, this approach suffers from the well-known ‘curse of complexity’ [8]. Generally, the analytic closed-form solution to these equations is possible only if the plant dynamics and constraints are simple, and the dimension of state and control variables is limited (e.g., the standard linear quadratic regulator theory [9]). However, most of the real-life problems not only include a complex nonlinear system but also have inequality path constraints. In this case, the numerical approach is a better candidate. It can be mentioned that, the shooting method [10] and gradient method [1] are mostly used among all the existing

[*] Postdoctoral Researcher; Email: sabyasachi.mondal@cranfield.ac.uk. This work was done while being a Ph.D. student at the Indian Institute of Science, Bangalore, India.

[†] Professor (Fellow, INAE and Associate Fellow, AIAA), Tel. +91-80-2293-2756; Email: padhi@iisc.ac.in

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

numerical methods for the indirect approach. The direct method is the dynamic programming which uses the principle of optimality to calculate the control. It leads to the Hamilton-Jacobi-Bellman (HJB) equation [2], a partial differential equation (PDE) in state-space and there are techniques to numerically compute approximate solutions [11]. However, this approach suffers from Bellman's 'curse of dimensionality' and is restricted to the small state dimensions.

One of the popular approaches for solving optimal control problem is to transcribe it into an equivalent NLP [12] which can be solved by several numerical optimization techniques. The approach is often called as 'first discretize, then optimize'. The primary advantage of this approach is that the inequality constraints are handled easily because the well developed NLP methods are capable of dealing with inequality constraints and active set changes. Unfortunately, this technique leads to a large-dimensional optimization problem and needs huge computational burden making it impossible to apply in its basic form for many practical problems. This disadvantage paves the way for model predictive control (MPC) [13, 14], which is well-established predictive control theory, including convergence and optimality guarantees for both linear and nonlinear problems under relevant assumptions [15, 16]. However, MPC suffers from the associated computational burden, and hence, applicable to slow-varying systems (such as chemical and biomedical process control applications). The larger control update time window helps in computing the control in real-time. Even though of late literature has appeared for Economic MPC [17] relaxing it from regulator problems and Fast MPC to address the computational time issue [18], these are still restricted mainly to relatively slow-varying linear systems, since the computations involved are typically not sufficiently fast.

Problems in the aerospace domain are usually governed by complex fast-changing system dynamics. Therefore, such problems need a rapid update of the control variables, so that the objectives and constraints should be satisfied (e.g. ensuring zero miss distance in missile guidance in addition to satisfying the path constraints). For such challenging problems, the existing 'Fast MPC' algorithms cannot provide the desired computation speed. Keeping such applications in mind, however, innovative optimal control solution approaches were proposed over the last three decades, such as pseudo-spectral optimal control [19, 20], adaptive critic technique [21] and so on. One such powerful and innovative approach is *Model Predictive Static Programming (MPSP)* [22, 23]. MPSP provides a solution to optimization problems with nonlinear plant dynamics and satisfies the terminal constraints in a fixed final time setting. The key idea is to convert the nonlinear optimal control problem to a static optimization problem with control variables being the only optimizing variables, resulting in a reduction of problem size. In addition, recursive computation of the sensitivity matrices that form the core of this algorithm, the MPSP technique has been found to be computationally very efficient. Over the last decade, the original version of MPSP has been modified to include variability in the final time [24], tracking problems [25], problems with uncertainty in time-invariant system parameters and/or initial conditions [26], and impulsive nature of the control action [27]. The MPSP technique has also been applied to a host of challenging problems such as missile guidance [28, 29], re-entry guidance [23], mobile robotics [25], lunar soft-landing [30] and so on. It is important to note that in the basic MPSP and most of its variants, the number of optimizing variables depends on the number of grid points. Hence, it can increase the size of the optimization problem depending on the time-horizon and grid size, which may reduce its computational efficiency. To address this issue, inspired by the pseudo-spectral philosophy, the Quasi-Spectral MPSP (QS-MPSP) [31] philosophy has also been proposed recently, which enhances the computational efficiency of MPSP even further. A major advantage of QS-MPSP is the reduced number of optimizing variables which are nothing but the coefficients of the basis function that are used to represent the control variable, and thus the optimizing variables are independent of the grid points. Therefore the computation burden is reduced substantially. Note that the number of optimizing variables being independent of the number of grid points is a well-known property in all spectral approaches. In fact, the pseudo-spectral optimal control thrives on that, and the QS-MPSP philosophy is inspired by that as well. It is because of the fundamental basis-function based representation approach. Another advantage is the smoothness of control, which helps the inner-loop autopilot to operate in reduced bandwidth.

QS-MPSP approach presented in [31] deals with terminal constraints but did not include the inequality path constraints on state and control variables. In the context of missile guidance problems, the constraint on look angle (a function of the states of missile model) needs to be satisfied during the entire terminal phase since the seeker cannot afford to lose the target. Moreover, because of the high-altitude engagement, where the dynamic pressure is low owing to substantially reduced atmosphere density, one must explicitly ensure that the lateral acceleration demand is restricted to an achievable limit throughout the trajectory.

In this paper, the QS-MPSP algorithm is modified to include the path constraints on states and control. It is named as 'Constrained Quasi-Spectral MPSP' (Constrained QS-MPSP) for obvious reasons. An inspiring innovation here is the fact that the path constraints on states have been converted into equivalent constraints on the control variables. Hence, it retains a significant advantage of the QS-MPSP philosophy, i.e., the number of free variables to optimize is still restricted only to the number control variables multiplied by the number of coefficients of the basis functions used to represent those. This is a critical factor for retaining its computational efficiency. Next, all applicable constraints are imposed as a set of "linear constraints" in terms of control deviations, which in turn leads to linear constraints in terms of the coefficients to be optimized. Therefore, these constraints, along with a quadratic cost function in terms of the coefficients, leads to a standard linear-quadratic static optimization problem, which has many efficient solutions with well-established theory. In summary, transforming the entire optimal control problem in terms of control variables alone and posing all applicable constraints as a set of linear constraints contributes to the computational efficiency of the algorithm significantly.

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Next, attention is focused on applying the proposed technique to solve the challenging problem reported in [31], but by imposing necessary constraints on the lead angle and lateral acceleration throughout the path, which are quite relevant for the problem, thereby making the problem substantially more challenging. This requirement comes from the fact that it is required to continuously track the target within the available field-of-view of the onboard seeker. Moreover, the commanded lateral acceleration must be within the achievable limits of the autopilot. Note that the concept of 'lead angle' (which is the angle between the line-of-sight and velocity vector) has been accounted here as the true look angle (which is the angle between the line-of-sight and the seeker x-axis) is not available in a point-mass engagement dynamics. However, as long as the seeker is mounted along the body x-axis (which is typically the case) and both angles of attack and side-slip angles are small (which is usually also true), the two angles are not significantly different. Hence, imposing constraints on the lead angle generally suffices the need. To make this point clear, one can refer to Fig. 1. The seeker antenna is fixed at the nose of the interceptor along its body 'x' axis (X_b in Fig. 1) on a gimbal platform. One of the objectives of the guidance system is to maneuver the interceptor in such a way that the seeker can always maintain a line-of-sight (LOS) ('R' in Fig. 1) to the target, i.e., to keep the target within the field-of-view (FOV). The seeker tracking module tries to point the antenna bore-sight axis (X_A) towards the target. However, as the seeker's field of view is limited, if the velocity vector (V_m) departs away from the LOS beyond a specific angle (due to maneuver), the seeker may lose sight of the target. Hence the angle between the velocity vector and LOS must be confined within a specific limit. This angle is known as the 'Lead angle' ('σ' in Fig. 1). If the angle between velocity and body axis (angle-of-attack 'σ_T') is assumed to be small, the lead angle is approximately equal to look angle, i.e., $\sigma \approx \sigma_T$. Therefore, constraints on the lead angle are very important and must be considered while designing a guidance law. It can also be pointed out here that the guidance commands are the lateral acceleration that helps the interceptor to maneuver. However, the commanded lateral acceleration is achieved by the deflection of the fins, which can be deflected only to a maximum angle. Hence, the constraints on lateral acceleration should also be considered in the guidance formulation.

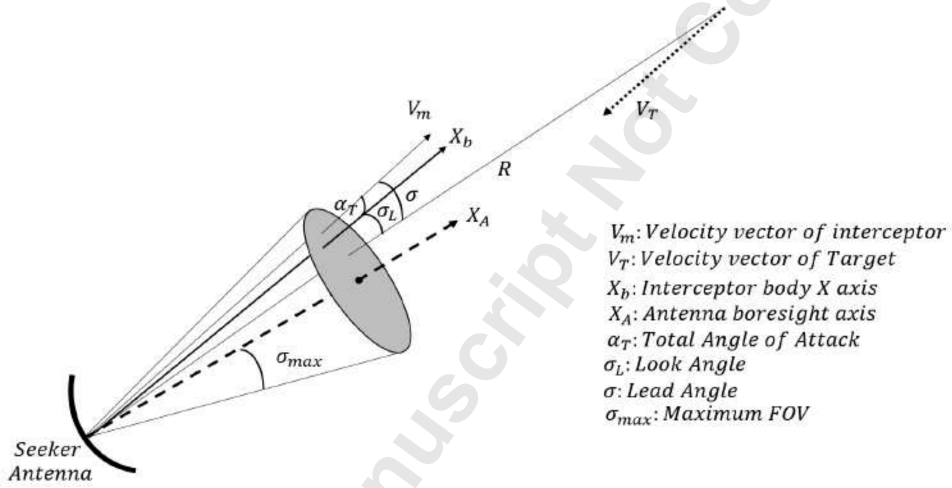
V_m
X_b
V_T
R
α_T
σ_L
σ
X_A
σ_{max}
Seeker
Antenna
V_m: Velocity vector of interceptor
V_T: Velocity vector of Target
X_b: Interceptor body X axis
X_A: Antenna boresight axis
α_T: Total Angle of Attack
σ_L: Look Angle
σ: Lead Angle
σ_{max}: Maximum FOV

Fig. 1 Seeker Antenna and Lead angle

It is interesting to note that some guidance laws with applicable constraints on the lead angle and lateral acceleration have been reported in the recent literature. One of them happens to be the circular navigation guidance [32] with a hybrid guidance logic. It is designed to handle the lead angle saturation. However, this paper does not impose the lead angle constraint directly. Instead, it considers the lead angle saturation and attempts to address the concern indirectly. In [33], Proportional Navigation (PN) guidance law is used to design an impact angle constrained guidance law, which also considers constraints on the lead angle and lateral acceleration. Different navigation ratios are used to design different phase of the proposed guidance scheme. For the initial guidance phase, the navigation ratio is selected as 1, while the final guidance phase is designed with navigation ratio greater than 2. In [34], a bias shaping method is proposed for biased-PN guidance. This guidance law keeps the target within the field-of-view of the seeker during the terminal phase of engagement. An alternate two-phased PN guidance scheme is presented in [35], where the lead angle and lateral acceleration are considered to be constraints. The navigation gains are obtained by the numerical solution. In [36], the lead angle constraint is handled by biased proportional navigation scheme. These papers, however, present variants of the conventional proportional navigation (PN) guidance, which can only be used under restrictive assumptions.

It can be mentioned that the lead angle constrained guidance problem has also been formulated in the Optimal control framework. In [37], optimal control theory has been applied to handle the lead angle constraint. The lead angle is considered to be a function of state variables. The maximum limit is imposed on the lead angle as inequality constraints. The problem is solved by applying optimal control theory with state constraints. The proposed guidance scheme provides three types of acceleration commands. The first command is for attaining the maximum lead angle. The second one is to maintain the lead angle at the constraint boundary. The third one is designed to intercept the target with the desired terminal impact angle. In [38], the lead angle constraint problem is formulated in an optimal control platform, and a closed-form guidance expression is obtained considering linearized engagement kinematics.

In this paper, on the other hand, the applicable nonlinear engagement dynamics is considered without any approximation. The relevant path constraints are also imposed. Even though the proposed Constrained QS-MPSP guidance compromises on the closed-form nature of the solution, it leads to a very powerful guidance logic indeed. This iterative logic is computationally fast (owing to the technical reasons explained before) and hence the authors sincerely believe that it can be implemented in on-board processors. The simulation results are quite satisfactory not only in terms of the performance measures such as near-zero miss-distance and constraint handling capability but it also leads to satisfactory results in terms of significantly less number of iterations for convergence, very less time for each iteration cycle and so on.

The specific contributions in this work are given as follows:

i. The inequality state and control path constraints were not included in our earlier formulation in [30]. Once these constraints are introduced, the complexity of the problem grows substantially, and, unlike [30], iterations can no longer be done in closed form. Significant new contributions of in this paper include (i) transforming the entire optimal control problem in terms of control variables alone (thereby reducing the number of free variables to be optimized), (ii) including transforming the state constraints to control constraints, and (iii) posing all applicable constraints as a set of linear constraints. These three factors enhance the computational efficiency of the algorithm significantly.
ii. The look-angle constraint (or lead angle constraint to be precise in the context of point mass dynamics) throughout the trajectory is being given prominence in the recent literature because of the significant reason that at no point of the missile trajectory in the homing phase it should result in loss of tracking of the target. In [30], this important practical issue was not taken into consideration. Hence, the problem discussed here, even though sounds like the one in [30] at first glance, it, in fact, is a "new application problem" having a lot more practical significance. The proposed Constrained QS-MPSP does give a platform to solve such challenging application problems.
iii. An important analysis of the computational complexity of the proposed Constrained QS-MPSP is included in this paper. This analysis reflects the insight of the computation burden. It was not included in [30] and [38].
iv. Mathematical details of the method to find the initial guess for coefficients of basis functions is included. Also, a guideline (useful but not strict) has been provided to select the number of basis function. It was not included in [30] and [38].
v. In earlier work [38], the guess history was collected using PN. In this paper, the guess history is collected using Biased PN. All the engagement scenario and results have been presented in a more realistic way with detailed explanation.
vi. A comparison with Impact Angle Control Guidance (IACG) is presented in this paper. It was not included in [38].
vii. The generated guidance commands are applied to Six-DOF dynamics to get the feel of practical implementation. It was not included in [30] and [38].

2 Constraints in Constrained QS-MPSP

Constrained optimal control problem is formulated by considering two categories of constraints viz., path, and terminal. Path constraints appear as inequality constraints on states and control or both. Terminal constraints appear at final time instant, and they are primarily equality constraints on states or function of states. A general form of constrained optimal control problem can be given as follows.

$$\min\ \mathbf{J} = \mathbf{E}(\mathbf{X}(\mathbf{t_f})) + \int_{\mathbf{t_0}}^{\mathbf{t_f}} \mathbf{L}(\mathbf{X}(\mathbf{t}), \mathbf{U}(\mathbf{t}))\mathbf{dt} \tag{1}$$

subject to state dynamics

$$\dot{\mathbf{X}}(\mathbf{t}) = \mathbf{f}(\mathbf{X}(\mathbf{t}), \mathbf{U}(\mathbf{t})) \tag{2}$$

with path constraints

$$\mathbf{g}(\mathbf{X}(\mathbf{t}), \mathbf{U}(\mathbf{t})) \leq \mathbf{0},\ \ \forall \mathbf{t} \in [\mathbf{t_0}, \mathbf{t_f}] \tag{3}$$

and terminal constraints

$$\phi(\mathbf{X}(\mathbf{t_0}),\mathbf{X}(\mathbf{t_f})) = \mathbf{0} \tag{4}$$

where, $\mathbf{X}(\mathbf{t}) \in \Re^{\mathbf{n}},\ \mathbf{U}(\mathbf{t})\Re^{\mathbf{m}}$. This problem is transformed in NLP using MPSP platform where the key idea is to express the path and terminal constraints in terms of constraints on the control variable only. The control variables are represented in spectral form, i.e., the weighted sum of basis functions. It is important to note that these weights are known as coefficients whose values are unknown. Therefore the constraints on states and control are represented as constraints on these coefficients only. The mathematical details are given in the following sections.

2.1 Derivation for State Inequality Constraints

In guidance design, nonlinear model of the plant is considered. The model is given by

$$\dot{\mathbf{X}} = f(\mathbf{X},\mathbf{U}) \tag{5}$$
$$\mathbf{Y} = h(\mathbf{X}) \tag{6}$$

where, $\mathbf{X} \in \Re^n$, $\mathbf{Y} \in \Re^p$, and $\mathbf{U} \in \Re^m$ denote state, output, and input or control of the system. MPSP deals with the discretized model which is obtained (Eqns. 7 and 8) using Euler discretization method

$$\mathbf{X}_{k+1} = \mathbf{X}_k + \triangle t f(\mathbf{X}_k,\mathbf{U}_k) = F(\mathbf{X}_k,\mathbf{U}_k) \tag{7}$$
$$\mathbf{Y}_k = h(\mathbf{X}_{\ k}) \tag{8}$$

where $k = 1,\ldots,N-1$ are the discrete time instants. Usually, in a state constrained problem, the value of a variable needs to be constrained. This variable might be a state or derived from states of the model. The variable can be represented by $Z(\mathbf{X}) \in R^q$. The value of this variable at k^{th} time instant and i^{th} iteration is denoted as $Z_k^i = Z(\mathbf{X}_k^i)$ (shown in Fig. 2). In i^{th} iteration, Z_k^i does not satisfy the constraints. It is required to restrict the variable in the next iteration, i.e., Z_k^{i+1} within the constraints (Z_{max} and Z_{min}). The requirement can be fulfilled by correcting Z_k^i by an unknown amount of dZ_k^i at i^{th} iteration.

$$dZ_k{}^i = Z_k^{i+1} - Z_k^i = Z\left(\mathbf{X}_k^{i+1}\right) - Z\left(\mathbf{X}_k{}^i\right) \tag{9}$$

The correction can be achieved by updating the control at i^{th} iteration i.e., by $\mathbf{U}_k^i$, $k = 1,2,\ldots,N-1$, by an unknown quantity

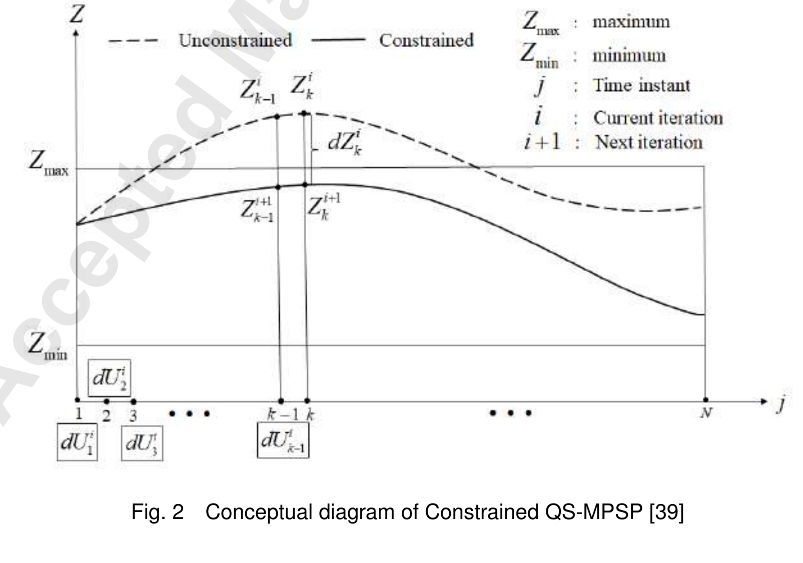

Fig. 2 Conceptual diagram of Constrained QS-MPSP [39]

$d\mathbf{U}_k^i$, to find an updated control $\mathbf{U}_k^{i+1} = \mathbf{U}_k^i + d\mathbf{U}_k^i$ (Fig. 3) in the present iteration and apply it to the system to obtain bounded Z_k^{i+1}.

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

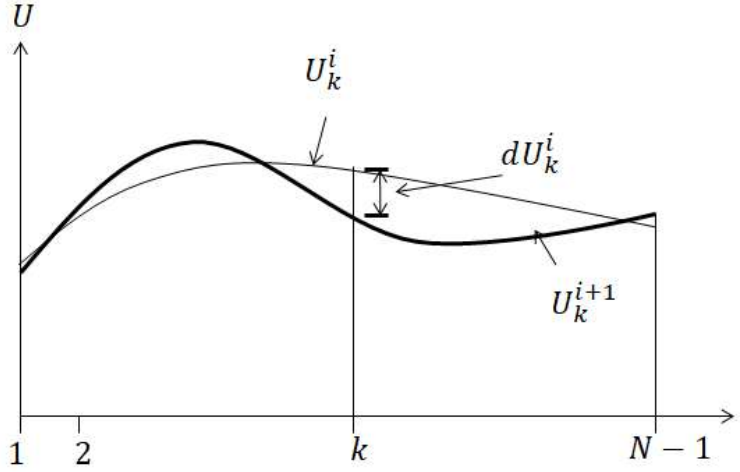

Fig. 3 Definition of error in control ($d\mathbf{U}_k^i$) in consecutive iteration [39]

The optimal changes in U_k^i at $k = 1,\ldots,k-1$, i.e., $d\mathbf{U}_1^i, d\mathbf{U}_2^i, \ldots, d\mathbf{U}_{k-1}^i$ affect the function Z_k^i to obtain the required correction dZ_k^i and constraints are satisfied. Therefore the optimizing variables are $d\mathbf{U}_1^i, d\mathbf{U}_2^i, \ldots, d\mathbf{U}_{k-1}^i, \ldots, d\mathbf{U}_{N-1}^i$.

The change in control, i.e., $d\mathbf{U}_k^i$ will change the states in the next iteration. The states at '$i+1^{th}$' iteration can be represented as $\mathbf{X}_k^{i+1} = \mathbf{X}_k^i + d\mathbf{X}_k^i$. Therefore the function Z_k can be represented as $Z(\mathbf{X}_k^{i+1})$. The effect of change in states on the function can be obtained by expanding it using Taylor Series as follows

$$Z(\mathbf{X}_k^{i+1}) = Z(\mathbf{X}_k^i) + \left[\frac{\partial Z_k^i}{\partial \mathbf{X}_k^i}\right] d\mathbf{X}_k^i + H.O.T \tag{10}$$

Neglecting the higher order terms (H.O.T) in Eqn.(10), the expression of $Z(\mathbf{X}_k^{i+1})$ becomes

$$Z(\mathbf{X}_k^{i+1}) = Z(\mathbf{X}_k^i) + \left[\frac{\partial Z_k^i}{\partial \mathbf{X}_k^i}\right] d\mathbf{X}_k^i \tag{11}$$

Eqn.(9) can be written as

$$d{Z_k}^i = \left[\frac{\partial {Z_k}^i}{\partial {\mathbf{X}_k}^i}\right] d{\mathbf{X}_k}^i \tag{12}$$

where $d{Z_k}^i$ represents the change in Z_k between the subsequent iterations and $d\mathbf{X}_k^i$ is the first variation of $\mathbf{X}_k^i$ which is obtained from Eqn.(8) as

$$\mathbf{X}_k^i = \mathbf{X}_{k-1}^i + \triangle t f(\mathbf{X}_{k-1}^i, \mathbf{U}_{k-1}^i) = F(\mathbf{X}_{k-1}^i, \mathbf{U}_{k-1}^i) \tag{13}$$

Therefore $d\mathbf{X}_k^i$ is given as

$$d\mathbf{X}_k^i = \left[\frac{\partial F}{\partial \mathbf{X}_{k-1}^i}\right] d\mathbf{X}_{k-1}^i + \left[\frac{\partial F}{\partial \mathbf{U}_{k-1}^i}\right] d\mathbf{U}_{k-1}^i \tag{14}$$

It can be observed that Eqn.(14) contains $d\mathbf{X}_{k-1}^i$ which can be written in similar fashion in terms of $\mathbf{X}_{k-2}^i$ and $\mathbf{U}_{k-2}^i$ as follows

$$d\mathbf{X}_{k-1}^i = \left[\frac{\partial F}{\partial \mathbf{X}_{k-2}^i}\right] d\mathbf{X}_{k-2}^i + \left[\frac{\partial F}{\partial \mathbf{U}_{k-2}^i}\right] d\mathbf{U}_{k-2}^i \tag{15}$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Similar expressions can be obtained for $d\mathbf{X}_{k-2}^i, \ldots, d\mathbf{X}_1^i$. These expressions are substituted in Eqn.(12) to obtain the expression of dZ_k^i as

$$\begin{aligned} d{Z_k}^i &= P_1^i d\mathbf{U}_1^i + {P_2}^i d{\mathbf{U}_2}^i + \cdots + P_{k-1}^i d\mathbf{U}_{k-1}^i \\ &= \sum_{j=1}^{k-1} {P_j}^i d\mathbf{U}_j^i \end{aligned} \tag{16}$$

where,

$$P_j^i = \left[\frac{\partial Z_k^i}{\partial \mathbf{X}_k^i}\right] \left[\frac{\partial F_{k-1}^i}{\partial \mathbf{X}_{k-1}^i}\right] \cdots \left[\frac{\partial F_{j+1}^i}{\partial \mathbf{X}_{j+1}^i}\right] \left[\frac{\partial {F_j}^i}{\partial \mathbf{U}_j}\right] \tag{17}$$

and $P_j^i \in \Re^{q \times m}$. It can be observed from Eqn.(9) that expression of Z_k^{i+1} becomes

$$Z_k^{i+1} = Z_k^i + d{Z_k}^i \tag{18}$$

Z_k^{i+1} needs to satisfy the constraints (lower boundary ($Z_{k_{\min}}$) and upper boundary ($Z_{k_{\max}}$)), i.e.,

$$Z_{k_{\min}} \le Z_k^{i+1} \le Z_{k_{\max}} \tag{19}$$

Substituting the expression of Z_k^{i+1} from Eqn.(18) in Eqn.(19) gives

$$Z_{k_{\min}} \le Z_k^i + dZ_k^i \le Z_{k_{\max}} \tag{20}$$

Simplification of Eqn.(20) leads to

$$Z_{k_{\min}} - Z_k^i \le dZ_k^i \le Z_{k_{\max}} - Z_k^i \tag{21}$$

Substituting dZ_k^i by $\left(\sum_{j=1}^{k-1} P_j^i d\mathbf{U}_j^i\right)$ (Eqn.(16)) in Eqn.(21) gives

$$Z_{k_{\min}} - Z_k^i \le \sum_{j=1}^{k-1} {P_j}^i d{\mathbf{U}_j}^i \le Z_{k_{\max}} - Z_k^i \tag{22}$$

It can be observed that Eqn.(22) imposes inequality constraint for each $k = 2, \ldots, N$. These constraints can be written in a compact form as follows

$$Z_l \le P d\mathbf{U}^i \le Z_u \tag{23}$$

where,

$$Z_l = \begin{bmatrix} Z_{1_{min}} - Z_1^i \\ Z_{2_{min}} - Z_2^i \\ \vdots \\ Z_{N-1_{min}} - Z_{N-1}^i \end{bmatrix} \quad Z_u = \begin{bmatrix} Z_{1_{max}} - Z_1^i \\ Z_{2_{max}} - Z_2^i \\ \vdots \\ Z_{N-1_{max}} - Z_{N-1}^i \end{bmatrix} \quad P = \begin{bmatrix} P_1^i & 0 & 0 & \cdots & 0 \\ P_1^i & P_2^i & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ P_1^i & P_2^i & P_3^i & \cdots & P_{N-1}^i \end{bmatrix}$$

$$d\mathbf{U}^i = \left[d\mathbf{U}_1^i \; d\mathbf{U}_2^i \; \ldots \; d\mathbf{U}_{N-1}^i\right]^T$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Eqn.(23) can be represented as

$$\begin{bmatrix} P^i \\ -P^i \end{bmatrix} d\mathbf{U}^i \leq \begin{bmatrix} Z_l \\ -Z_u \end{bmatrix} \tag{24}$$

Terminal constraints appears as equality constraint equation which are same as shown in [40]

$$\begin{aligned} d\mathbf{Y}^i_N &= B^i_1 d\mathbf{U}^i_1 + B^i_2 d\mathbf{U}^i_2 + \cdots + B_{N-1}{}^i d\mathbf{U}^i_{k-1} \\ &= \sum_{j=1}^{N-1} B^i_j d\mathbf{U}^i_j \end{aligned} \tag{25}$$

Eqn.(25) can be written in matrix form as

$$B^i d\mathbf{U}^i = d\mathbf{Y}^i_N \tag{26}$$

where,

$$B^i = \begin{bmatrix} B^i_1 & B^i_2 & \cdots & B^i_{N-1} \end{bmatrix} \tag{27}$$

At this stage, it is clear that the path and terminal constraints on states are represented by constraints only on the control variable (Eqns.(24) and (26)). In this process, the number of optimizing variables is reduced. Finally, the number of optimizing variables $d\mathbf{U}^i$ is $m \times N - 1$ (since, $N-1$ steps and $\mathbf{U} \in \Re^m$). The minimizing function is the updated control energy $\mathbf{U}^{i+1^T}\mathbf{U}^{i+1}$. The optimization problem can be written as follows

$$\min_{d\mathbf{U}^i} J = \frac{1}{2}(\mathbf{U}^i + d\mathbf{U}^i)^T R (\mathbf{U}^i + d\mathbf{U}^i) \tag{28}$$

subject to

$$\begin{aligned} B^i d\mathbf{U}^i &= d\mathbf{Y}^i_N \\ \begin{bmatrix} P^i \\ -P^i \end{bmatrix} d\mathbf{U}^i &\leq \begin{bmatrix} Z_l \\ -Z_u \end{bmatrix} \end{aligned} \tag{29}$$

The dimension of the optimization problem given in Eqns.(28) and (29) varies with N (grid points or optimizing horizon) and m (number of control variables). Obviously, the dimension increases with an increase in either or both of them. Therefore the conventional efficient NLP techniques like Interior-point, Active-set, Sequential Quadratic Programming requires a larger time to converge.

To use the optimization techniques in real-time processes, the dimension of optimizing variables should be reduced. Relevant work has been published by the authors [40], where the number of optimizing variables is made independent of grid points by implementing the idea of spectral representation of control variable as follows.

$$\mathbf{U}(t_k)^{i+1} = \sum_{l=1}^{N_p} C^{i+1}_l \varphi_l(t_k) \tag{30}$$

where, $C^{i+1}_l \in \Re^m$. The control in Eqn.(30) is the weighted sum of finite number (N_p) of basis functions (φ). The spectral representation of control reduces the number of optimization variables from N to N_p (number of basis functions) and $N_p <<$ N. The impact of this idea can be realized in terms of computational complexity. Generally, a QP problem with n states variables, m input variables, N prediction horizon requires $O(N^3(n+m)^3)$ operations per iteration in the Interior-point method [41]. Here the QP is made more efficient by reformulating the problem in terms of control variables alone. Hence, the overall complexity per iteration is reduced to $O(N_p^3 m^3) << O(N^3(n+m)^3)$.

The expression of $\mathbf{U}_k^{i+1}$ in $d\mathbf{U}_k^i = \mathbf{U}_k^i - \mathbf{U}_k^{i+1}$ is replaced by the expression in Eqn.(30) to obtain $\sum_{j=1}^{k-1} P_j^i d\mathbf{U}_j^i$ as follows

$$\sum_{j=1}^{k-1} P_j^i d\mathbf{U}_j^i = \sum_{j=1}^{k-1} P_{0_j}^i - \left[\sum_{j=1}^{k-1}\sum_{l=l}^{N_p} P_j^i {C_l}^{i+1} \varphi_l\left(t_j\right)\right] \tag{31}$$

where, $P_{0_j}^i \in \Re^q$. Substitution of expressions of $\sum_{j=1}^{k-1} P_j^i d\mathbf{U}_j^i$ in Eqn.(22) gives

$$Z_{k_{\min}} - Z_k^i - \sum_{j=1}^{k-1} P_{0_j}^i \leq -\sum_{j=1}^{k-1}\sum_{l=1}^{N_p} P_j^i C_l^{i+1} \varphi_l\left(t_j\right) \leq Z_{k_{\max}} - Z_k^i - \sum_{j=1}^{k-1} P_{0_j}^i \tag{32}$$

Simplification of the term $\left(\sum_{j=1}^{k-1}\sum_{l=1}^{N_p} P_j^i {C_l}^{i+1} \varphi_j\left(t_j\right)\right)$ in Eqn.(32) is shown below

$\sum_{j=1}^{k-1}\sum_{l=1}^{N_p} P_j^i {C_l}^{i+1} \varphi_l\left(t_j\right)$

$$\begin{aligned} &= \left[P_1^i \varphi_1\left(t_1\right) + \cdots + P_{k-1}^i \varphi_1\left(t_{k-1}\right)\right] {C_1}^{i+1} + \cdots + \left[P_1^i \varphi_{Np}\left(t_1\right) + \cdots + P_{k-1}^i \varphi_{Np}\left(t_{k-1}\right)\right] {C_{N_p}}^{i+1} \\ &= \left[\sum_{j=1}^{k-1} P_j^i \varphi_1\left(t_j\right) \;\; \sum_{j=1}^{k-1} P_j^i \varphi_2\left(t_j\right) \cdots \sum_{j=1}^{k-1} P_j^i \varphi_{N_p}\left(t_j\right)\right] \begin{bmatrix} {C_1}^{i+1} \\ \vdots \\ {C_{N_p}}^{i+1} \end{bmatrix} \end{aligned} \tag{33}$$

Eqn.(33) is substituted in Eqn. (32) to obtain

$$-\left[\sum_{j=1}^{k-1} P_j^i \varphi_1\left(t_j\right) \;\; \sum_{j=1}^{k-1} P_j^i \varphi_2\left(t_j\right) \cdots \sum_{j=1}^{k-1} P_j^i \varphi_{N_p}\left(t_j\right)\right] \begin{bmatrix} {C_1}^{i+1} \\ \vdots \\ {C_{N_p}}^{i+1} \end{bmatrix} \leq g_k \tag{34}$$

where,

$$g_k = Z_{k_{\max}} - Z_k^i - \sum_{j=1}^{k-1} P_{0_j}^i$$

The inequalities for each $k = 1, 2, \ldots, N-1$ can be written in a matrix form as shown below

$$DC^{i+1} \leq G \tag{35}$$

where,

$$D = -\begin{bmatrix} P_1^i \varphi_1\left(t_1\right) & P_1^i \varphi_2\left(t_1\right) & \cdots & P_1^i \varphi_{N_p}\left(t_1\right) \\ \vdots & \vdots & \vdots & \vdots \\ \sum_{j=1}^{k-1} P_j^i \varphi_1\left(t_j\right) & \sum_{j=1}^{k-1} P_j^i \varphi_2\left(t_j\right) & \cdots & \sum_{j=1}^{k-1} P_j^i \varphi_{N_p}\left(t_j\right) \\ \vdots & \vdots & \ddots & \vdots \\ \sum_{j=1}^{N-1} P_j^i \varphi_1\left(t_j\right) & \sum_{j=1}^{N-1} P_j^i \varphi_2\left(t_j\right) & \cdots & \sum_{j=1}^{N-1} P_j^i \varphi_{N_p}\left(t_j\right) \end{bmatrix}$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

$$G = \left[g_1 \ g_2 \ \ldots \ g_{N-1}\right]^T$$

and

$$C^{i+1} = \left[{C_1}^{i+1} \ {C_2}^{i+1} \ \ldots \ {C_{N_p}}^{i+1}\right]^T$$

Eqn.(32) is rewritten as

$$l_k \leq -\left[\sum_{j=1}^{k-1} P_j^i \varphi_1\left(t_j\right) \ \sum_{j=1}^{k-1} P_j^i \varphi_2\left(t_j\right) \cdots \sum_{j=1}^{k-1} P_j^i \varphi_{N_p}\left(t_j\right)\right] \begin{bmatrix} {C_1}^{i+1} \\ \vdots \\ {C_{N_p}}^{i+1} \end{bmatrix} \tag{36}$$

where,

$$l_k = Z_{k_{\min}} - Z_k^i - \sum_{j=1}^{k-1} P_{0_j}^i$$

Eqn.(36) is equivalent to

$$\left[\sum_{j=1}^{k-1} P_j^i \varphi_1\left(t_j\right) \ \sum_{j=1}^{k-1} P_j^i \varphi_2\left(t_j\right) \cdots \sum_{j=1}^{k-1} P_j^i \varphi_{N_p}\left(t_j\right)\right] \begin{bmatrix} {C_1}^{i+1} \\ \vdots \\ {C_{N_p}}^{i+1} \end{bmatrix} \leq -l_k \tag{37}$$

These inequalities for $k = 1, 2, \ldots, N-1$ are written in matrix form as follows

$$-DC^{i+1} \leq L \tag{38}$$

where,

$$L = \left[-l_1 \ -l_2 \ \ldots \ -l_{N-1}\right]^T$$

Eqns.(35) and (38) are combined to represent all the state inequality constraints as

$$\begin{bmatrix} -D \\ D \end{bmatrix} C^{i+1} \leq \begin{bmatrix} L \\ G \end{bmatrix} \tag{39}$$

For simplicity, Eqn.(39) is written as

$$WC^{i+1} \leq V \tag{40}$$

where,

$$W = \begin{bmatrix} -D \\ D \end{bmatrix}$$

and

$$V = \begin{bmatrix} L \\ G \end{bmatrix}$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Spectral control expression in Eqn.(30) is substituted in Eqn.(25) to obtain

$$\begin{aligned} d\mathbf{Y}_N^i &= B_1 dU_1 + B_2 dU_2 + \ldots + B_{N-1} dU_{N-1} \\ &= B_0 - \sum_{k=1}^{N-1} \sum_{l=1}^{N_p} B_k^i {C_l}^{i+1} \varphi_l\left(t_k\right) \end{aligned} \tag{41}$$

where, $B_0 = \sum\limits_{k=1}^{N-1} B_k^i \mathbf{U}_k^i$. Expression of $\left(\sum\limits_{k=1}^{N-1} \sum\limits_{l=1}^{N_p} B_k^i {C_l}^{i+1} \varphi_l\left(t_k\right) \right)$ in Eqn.(41) can be written as

$$\sum_{k=1}^{N-1} \sum_{l=1}^{N_p} B_k^i {C_l}^{i+1} \varphi_l\left(t_k\right) = \left[\sum_{k=1}^{N-1} B_k^i \varphi_1\left(t_k\right) \; \sum_{k=1}^{N-1} B_k^i \varphi_2\left(t_k\right) \cdots \sum_{k=1}^{N-1} B_k^i \varphi_{N_p}\left(t_k\right) \right] \begin{bmatrix} C_1^{i+1} \\ C_2^{i+1} \\ \vdots \\ C_{N_p}^{i+1} \end{bmatrix} \tag{42}$$

The expression of $\left(\sum\limits_{k=1}^{N-1} \sum\limits_{l=1}^{Np} B_k^i {C_l}^{i+1} \varphi_l\left(t_k\right) \right)$ (shown above) is substituted in Eqn.(41) to obtain

$$BC^{i+1} = B_0 - d\mathbf{Y}_N^i \tag{43}$$

where,

$$B = \left[\sum_{k=1}^{N-1} B_k^i \varphi_1\left(t_k\right) \; \sum_{k=1}^{N-1} B_k^i \varphi_2\left(t_k\right) \ldots \sum_{k=1}^{N-1} B_k^i \varphi_{N_p}\left(t_k\right) \right]$$

For convenience, Eqn.(43) is written as

$$BC^{i+1} = E \tag{44}$$

where,

$$E = B_0 - d\mathbf{Y}_N^i$$

Maximum value of control expression in Eqn.(30) is obtained as

$$\begin{aligned} |\mathbf{U}_k^{i+1}| &= |C_1^{i+1}\varphi_1(t_k) + C_2^{i+1}\varphi_2(t_k) + \ldots + C_{N_p}^{i+1}\varphi_{N_p}(t_k)| \\ &\leq |C_1^{i+1}\varphi_1(t_k)| + |C_2^{i+1}\varphi_2(t_k)| + \ldots + |C_{N_p}^{l+1}\varphi_{N_p}(t_k)| \\ &\leq |C_1^{i+1}||\varphi_1(t_k)| + |C_2^{i+1}||\varphi_2(t_k)| + \ldots + |C_{N_p}^{i+1}||\varphi_{N_p}(t_k)| \end{aligned} \tag{45}$$

The basis functions are selected by the user and their values $(|\varphi_l(t_k)|; l = 1, \ldots, N_p)$ at every time instants $k = 1, \ldots, N-1$, are known. Therefore minimization of unknown coefficients C_l^{i+1}, $l = 1, 2, \ldots, N_p$ will result in minimization of $\mathbf{U}_k^{l+1}$. Hence the number of optimizing variables is independent of the grid points N. The minimization problem is given as follows

$$\min_{C^{i+1}} J = \frac{1}{2} {C^{i+1}}^T R C^{i+1} \tag{46}$$

subject to

$$BC^{i+1} = E \tag{47}$$

$$WC^{i+1} \leq V \tag{48}$$

The state constraints are discussed so far. Next, the inequality constraints on control variables need to be written in terms of spectral coefficients as well. They are discussed in the following section.

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

2.2 Derivation for Control Constraint

Constraints on the control variable are an obvious event. The physical limitations of systems are the cause of these constraints. In the context of a missile, the control constraint means that the control surfaces of a missile can be deflected by limited angle. Therefore the guidance algorithm must produce the control command, which satisfies the constraints. The control input should be constrained within $\mathbf{U}_{\max}$ and $\mathbf{U}_{\min}$, as shown in Fig. 4. Mathematical representation of this constraint is given by

$$\mathbf{U}_{k_{\min}} \leq \mathbf{U}_k^{i+1} \leq \mathbf{U}_{k_{\max}} \tag{49}$$

The control in spectral form is given by

$$\begin{aligned} \mathbf{U}_k^{i+1} &= C_1^{i+1}\varphi_1(t_k) + C_2^{i+1}\varphi_2(t_k) + \cdots + C_{N_p}^{i+1}\varphi_{N_p}(t_k) \\ &= \begin{bmatrix}\varphi_1(t_k) \ \varphi_2(t_k) \ \ldots \ \varphi_{N_p}(t_k)\end{bmatrix} \begin{bmatrix} C_1^{i+1} \\ C_2^{i+1} \\ \vdots \\ C_{N_p}^{i+1} \end{bmatrix} \end{aligned} \tag{50}$$

The expression in Eqn.(50) is substituted in Eqn. (49) to obtain

$$\mathbf{U}_{k_{\min}} \leq \begin{bmatrix}\varphi_1(t_k) \ \varphi_2(t_k) \ \ldots \ \varphi_{N_p}(t_k)\end{bmatrix} \begin{bmatrix} C_1^{i+1} \\ C_2^{i+1} \\ \vdots \\ C_{N_p}^{i+1} \end{bmatrix} \leq \mathbf{U}_{k_{\max}} \tag{51}$$

Similar inequalities can be obtained for $k = 1, 2, \ldots, N-1$. These inequalities can be written in a matrix form as

$$\mathbf{U}_{\min} \leq \Phi C^{i+1} \leq \mathbf{U}_{\max} \tag{52}$$

where

$$\Phi = \begin{bmatrix} \varphi_1(t_1) & \varphi_2(t_1) & \ldots & \varphi_{N_p}(t_1) \\ \varphi_1(t_2) & \varphi_2(t_2) & \ldots & \varphi_{N_p}(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_1(t_{N-1}) & \varphi_2(t_{N-1}) & \ldots & \varphi_{N_p}(t_{N-1}) \end{bmatrix}$$

$\mathbf{U}_{\min} = [\mathbf{U}_{1_{\min}} \ \ldots \ \mathbf{U}_{N-1_{\min}}]^T$ and $\mathbf{U}_{\max} = [\mathbf{U}_{1_{\max}} \ \ldots \ \mathbf{U}_{N-1_{\max}}]^T$. It can be seen that Eqn.(52) has both type of inequalities. These inequalities are converted in ‘lesser than inequality’ only. Therefore, Eqn.(52) is written as

$$QC^{i+1} \leq \mathbf{U}_b \tag{53}$$

where,

$$Q = \begin{bmatrix} \Phi \\ -\Phi \end{bmatrix}$$

and

$$\mathbf{U}_b = \begin{bmatrix} \mathbf{U}_{\max} \\ -\mathbf{U}_{\min} \end{bmatrix}$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

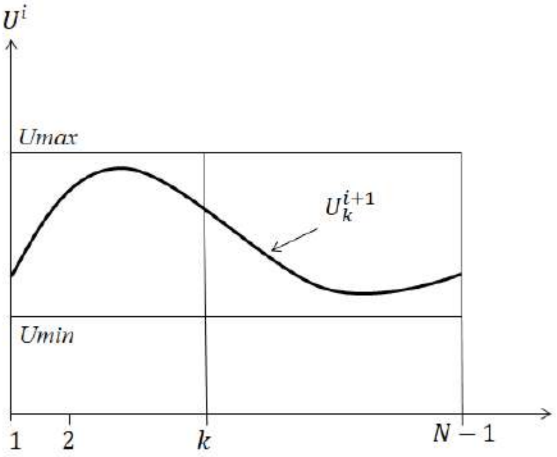

Fig. 4 Control Constraints [39]

It can be observed that the control constraints (Eqns.(53) and (48)) have the similar form as the state constraints with same unknown coefficients $C_1^{i+1}, C_2^{i+1}, \ldots, C_{N_p}^{i+1}$. Therefore the constraints are combined together to obtain the complete set of constraints on state and control.

$$\begin{bmatrix} W \\ Q \end{bmatrix} C^{i+1} \leq \begin{bmatrix} V \\ \mathbf{U}_b \end{bmatrix} \tag{54}$$

Hence, the optimization problem becomes

$$\min_{C^{i+1}} J = \frac{1}{2} {C^{i+1}}^T R C^{i+1} \tag{55}$$

subject to

$$BC^{i+1} = E \tag{56}$$

$$\begin{bmatrix} W \\ Q \end{bmatrix} C^{i+1} \leq \begin{bmatrix} V \\ \mathbf{U}_b \end{bmatrix} \tag{57}$$

The constrained optimal control problem given in Eqns. (1)-(4) is transformed in Eqns.(55)-(57). It is a convex optimization problem with quadratic cost, linear equality, and inequality constraints. There are various nonlinear optimization techniques that can solve this problem. Some of them are Sequential Quadratic Programming, Active-set, Interior Point (IP) etc. In this work, IP is used.

3 Lead Angle and Lateral Acceleration Constrained Guidance

In this section Constrained QS-MPSP algorithm is used for the guidance design of the interceptor to engage an incoming ballistic target. In general, the incoming ballistic missiles come with very high speed. The gravity turning effect is usually very small, but it has been accounted for in the target model. Punishing re-entry conditions coupled with very small time-to-go offers a very limited window in all such manoeuvres. Sometimes, however, due to the asymmetry in the mass distribution, targets go through unintentional "spiral manoeuvres" during the re-entry. Realizing its benefit to confuse an interceptor, some modern targets intend to introduce limited intentional spiral manoeuvres to enhance its probability of survival. Very few targets also intentionally introduce lateral acceleration to aim for a different ground target, which adds to its deception factor. Such a study falls under "target state estimation", which is beyond the scope of this paper. However, note that as long as the target state is accurately estimated from a good estimation logic, the same will automatically be exploited in the proposed predictive guidance in this paper. Hence, for the demonstration of the concept proposed in this paper, a ballistic target with no manoeuvring capability has been assumed.

The interceptor has limited maneuvering capability, i.e., it can produce limited lateral acceleration. It carries an active seeker which tracks the target in its terminal phase of guidance. It is required to keep the target inside the Field of View (FOV)

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

of the seeker. Since the FOV of a seeker is limited, the look angle is also limited. Considering a point mass dynamic model and assuming the Angle-of-attack (AoA) and Side-slip angle to be small, the lead angle becomes an important parameter to deal with. Therefore, it is necessary to engage the target with lead angle constraint and limited lateral acceleration capability.

3.1 Simulation of $3D$ Engagement

The simulation study is performed in a 3D engagement scenario. The lateral accelerations cause the interceptor to maneuver in pitch and yaw direction. In this work, the point mass dynamics of interceptor and target is considered, which are given in the following subsections. The simulation study is performed using MATLAB R2019a on a Laptop with processor $i5$ 6^{th} generation 2.30 Ghz and 4 GB RAM. Even though the onboard processors are typically less capable in computational speed, onboard programs are typically written in the assembly language, which is substantially faster than MATLAB. Moreover, quite capable space-grade processors are also available in the market [42], which can be used to harness the advantage of the proposed technique.

3.1.1 Dynamic Model of the Interceptor and Target

The point-mass dynamic model of the interceptor in 3D is given as follows

$$\begin{bmatrix} \dot{V}_m \\ \dot{\gamma}_m \\ \dot{\psi}_m \\ \dot{x}_m \\ \dot{y}_m \\ \dot{z}_m \end{bmatrix} = \begin{bmatrix} -\frac{D}{m} - g\sin(\gamma_m) \\ \frac{a_z - g\cos(\gamma_m)}{V_m} \\ \frac{a_y}{V_m\cos(\gamma_m)} \\ V_m\cos(\gamma_m)\cos(\psi_m) \\ V_m\cos(\gamma_m)\sin(\psi_m) \\ V_m\sin(\gamma_m) \end{bmatrix} \tag{58}$$

where, $V_m, \gamma_m,\ \psi_m, x_m, y_m, z_m$ denote the velocity, flight-path angle, heading angle, x, y and z positions in inertial reference frame of the interceptor, respectively.

The point-mass dynamic model of the target in 3D is given as follows

$$\begin{bmatrix} \dot{V}_T \\ \dot{\gamma}_T \\ \dot{\psi}_T \\ \dot{x}_T \\ \dot{y}_T \\ \dot{z}_T \end{bmatrix} = \begin{bmatrix} \frac{-0.5\rho V_T^2}{\beta} - g\sin(\gamma_T) \\ -\frac{g\cos\gamma_T}{V_T} \\ 0 \\ V_T\cos(\gamma_T)\cos(\psi_T) \\ V_T\cos(\gamma_T)\sin(\psi_T) \\ V_T\sin(\gamma_T) \end{bmatrix} \tag{59}$$

where, $V_T, \gamma_T, \psi_T x_T, y_T, z_T$ denote the velocity, flight path angle, heading angle, x, y and z positions of target in inertial reference frame respectively.

3.1.2 Calculation of Lead Angle in $3D$

The lead angle for $3D$ engagement scenario can be computed as the angle between the velocity ($\vec{V}_m$) and LOS vector ($\vec{R}$) in space. The dot product of velocity vector and LOS vector can be given as

$$\vec{V}_m.\vec{R} = |\vec{V}_m|.|\vec{R}|\cos(\sigma) \tag{60}$$

$$\cos(\sigma) = \frac{\vec{V}_m.\vec{R}}{|\vec{V}_m|.|\vec{R}|} \tag{61}$$

where, $\vec{V}_m = [\dot{x}_m\ \dot{y}_m\ \dot{z}_m]^T$ and $\vec{R} = [(x_T - x_m)\ \ (y_T - y_m)\ \ (z_T - z_m)]^T$. It can be observed that the lead angle is a function of states. $Z = \cos(\sigma)$ is considered as the function to be constrained. Since the lead angle is calculated in $3D$, the value will be bounded within a certain upper limit of σ_{max}.

3.2 Initial Guess for Coefficients C

It has been described that the control is expressed as a weighted sum of basis function $\varphi(t)$ and the weights or coefficients are determined by a static optimization process. Hence, the initial guess control history is approximated by using N_p

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

coefficients denoted by $C_l, l = 1, 2, \ldots, N_p$. These values of C serve as the initial value in the optimization process. It can be mentioned that there are N grid points and N_p unknown coefficients. At each grid points or time instant t_k, the control value is expressed as

$$\mathbf{U}(t_k) = C_1\varphi_1(t_k) + C_2\varphi_2(t_k) + \ldots + C_{N_p}\varphi_{N_p}(t_k) \tag{62}$$

Therefore, it is obvious that the control values at N_p time instants are required for evaluating $C_1, C_2, \ldots, C_{N_p}$. However, there are approximation errors in this process. Hence, the values of the coefficients are obtained considering N time instants. For example, consider the following expressions

$$C_1\varphi_1(t_1) + C_2\varphi_2(t_1) + \ldots + C_{N_p}\varphi_{N_p}(t_1) = \mathbf{U}(t_1) \tag{63}$$

$$C_1\varphi_1(t_2) + C_2\varphi_2(t_2) + \ldots + C_{N_p}\varphi_{N_p}(t_2) = \mathbf{U}(t_2) \tag{64}$$

$$\vdots \tag{65}$$

$$C_1\varphi_1(t_{N-1}) + C_2\varphi_2(t_{N-1}) + \ldots + C_{N_p}\varphi_{N_p}(t_{N-1}) = \mathbf{U}(t_{N-1}) \tag{66}$$

Eqns.(63)-(66) can be written as

$$Ay = b \tag{67}$$

where

$$A = \begin{bmatrix} \varphi_1(t_1) & \varphi_2(t_1) & \ldots & \varphi_{N_p}(t_1) \\ \varphi_1(t_2) & \varphi_2(t_2) & \ldots & \varphi_{N_p}(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_1(t_{N-1}) & \varphi_2(t_{N-1}) & \ldots & \varphi_{N_p}(t_{N-1}) \end{bmatrix}$$

$$y = \left[C_1\ C_2\ \ldots\ C_{N_p}\right]^T$$

$$b = [\mathbf{U}(t_1)\ \mathbf{U}(t_2)\ \ldots\ \mathbf{U}(t_{N-1})]^T$$

The solution of y is obtained as

$$y = (A^TA)^{-1}A^Tb \tag{68}$$

In this work, the guess control history is collected by simulating the engagement scenario (in Tab. 1) using Biased PN guidance [43]. The lateral accelerations are shown in Figs. 5 and 6.

Table 1 Initial Conditions for the Engagement

Vehicle	γ, deg	ψ, deg	V, ms^{-1}	x, m	y, m	z, m
Interceptor	45	36	1200	15500	15000	16000
Target missile	39	50	1600	28000	28000	28000

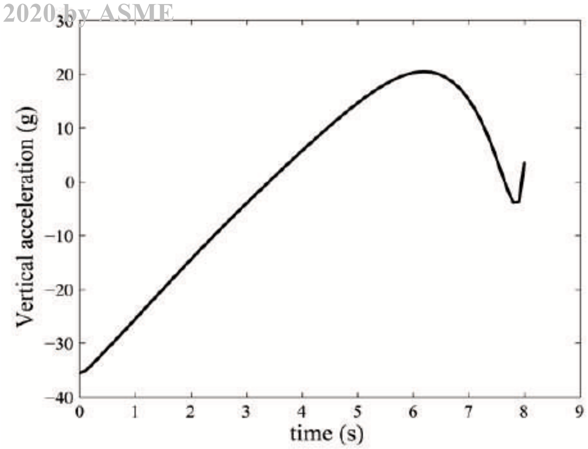

Fig. 5 Vertical Acceleration BPN

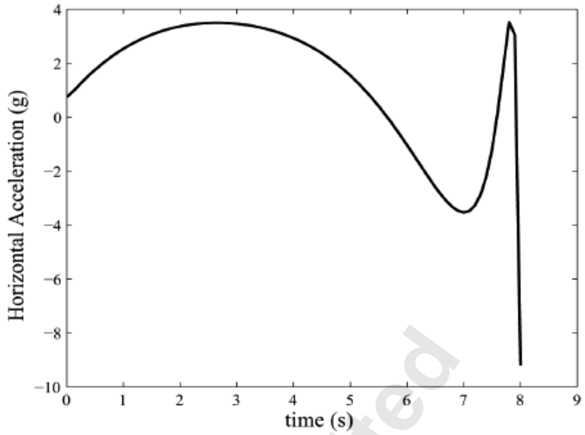

Fig. 6 Horizontal Acceleration BPN

After obtaining the lateral acceleration, the initial values of the coefficients are extracted by the method described above. It can be mentioned that in a 3D scenario, there is vertical and horizontal lateral acceleration, which causes the missile to turn in space. Hence the dimension of the control **U** is two, i.e., $\mathbf{U} \in \Re^2$ and consequently the dimension of each coefficient is two, i.e., $C_l \in \Re^2$. In this work, the control is represented as a weighted sum of five basis functions (i.e., $N_p = 5$) which are polynomials of t_{go} (i.e., $\varphi_1(t_{go_k}) = 1, \varphi_2(t_{go_k}) = t_{go_k}, \ldots, \varphi_5(t_{go_k}) = t_{go_k}^4$). The expression of the control can be written as

$$\mathbf{U}(t_{go_k}) = \sum_{l=1}^{5} C_l \varphi_l(t_{go_k}) \tag{69}$$

It can be mentioned that, there is no strict rule for selecting the value of N_p. As a heuristic, the minimum number of basis functions that can approximate the guess control with an acceptable error margin can be selected as the value of N_p. The control guess history is shown in Figs. 5 and 6. Corresponding initial values of the coefficients are given as follows

$$C_1 = \begin{bmatrix} -10.01 \\ -20.73 \end{bmatrix}, C_2 = \begin{bmatrix} 14.63 \\ 15.55 \end{bmatrix}, C_3 = \begin{bmatrix} -2.41 \\ -1.85 \end{bmatrix}, C_4 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, C_5 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

These values of the coefficients are the inputs to the Interior-Point algorithm to start with. It can be mentioned that this way of initialising an optimization algorithm is regarded as warm-starting which helps the algorithm to converge to the optimal value faster.

3.3 Results and Discussion

The objective of the mission is to intercept the target with minimum miss distance which is a terminal constraint. It can be mentioned that to satisfy the terminal constraint, the instantaneous position of the interceptor is chosen as output vector, i.e., $\mathbf{Y} = [x_m \ y_m \ z_m]^T$. Ideally, at the time of interception, the interceptor should reach Predicted Interception Point (PIP), i.e., $\mathbf{Y}^* = [x_{PIP} \ y_{PIP} \ z_{PIP}]^T$. The interception point for BPN is considered as PIP. Therefore the engagement time for Constrained QS-MPSP guidance is the same as that for BPN. It is important to note that the limitation on vertical acceleration is considered as 60% of the maximum value of vertical acceleration generated by BPN in the positive and negative direction. Due to the symmetrical structure, it is assumed that the missile can produce the same lift in both pitch and yaw directions. Therefore, the bound on horizontal acceleration is considered the same as that on vertical acceleration. The trajectories generated by Constrained QS-MPSP is different from Biased PN (Fig. 7) for the same initial conditions given in Tab. 1. This difference in trajectories is resulted due to different lateral accelerations generated by two guidance algorithms. Vertical and horizontal accelerations are shown in figs. 9 and 10, respectively. It can be observed from Fig. 9 that vertical acceleration generated by BPN (guess control) violets the bound for $0 - 1.4$ sec and $4.7 - 7.2$ sec (around 47% of total flight time). With this control as initial guess, that Constrained QS-MPSP has produced optimal lateral acceleration within the bound in such a way that the lead angle constraint (18 deg) is satisfied. The constraint on the lead angle is considered to be around 5 deg less than the maximum value of lead angle generated by BPN. The lead angle generated by BPN and Constrained QS-MPSP is shown in Fig. 8. It can be observed from the figure that the lead angle generated by BPN violets the constraint for almost 3 sec, which is around 35% of total flight time. Performance of Constrained QS-MPSP is shown in Tab. 2. It can be observed

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

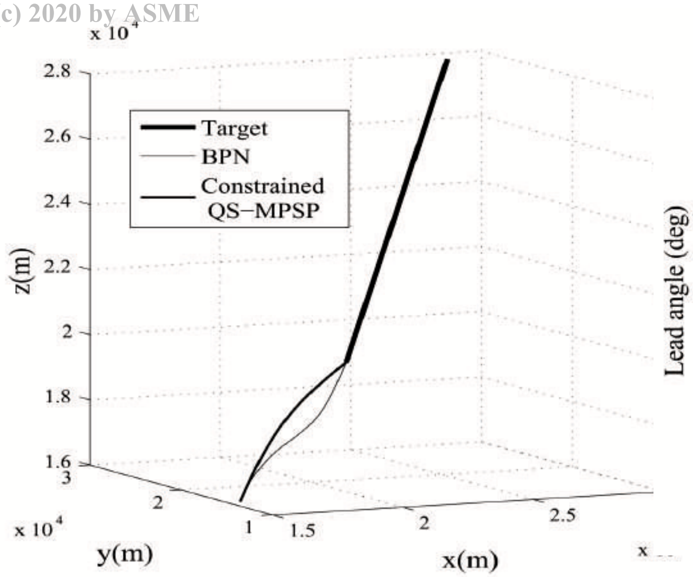

Fig. 7 Engagement

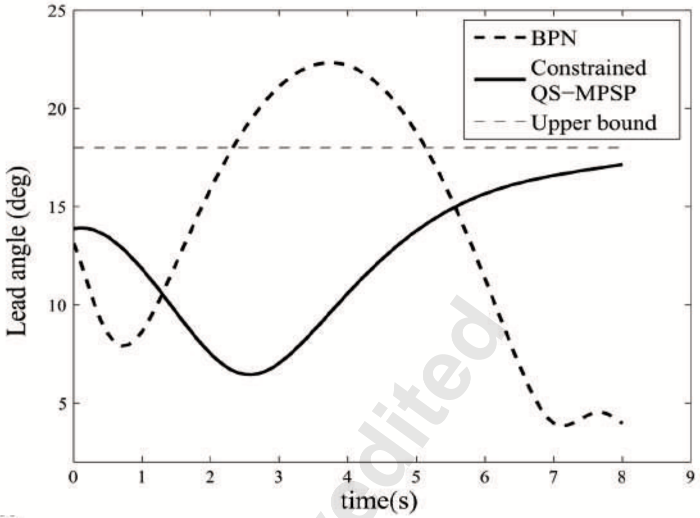

Fig. 8 Lead Angle

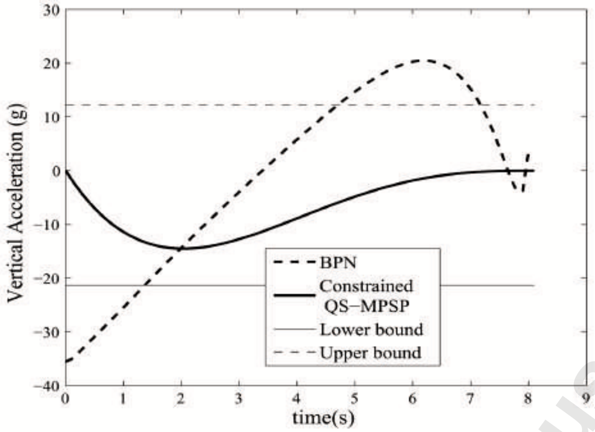

Fig. 9 Vertical Acceleration

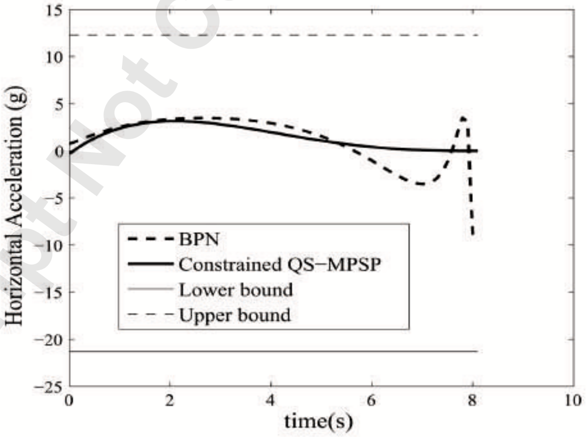

Fig. 10 Horizontal Acceleration

Table 2 Performance of Constrained QS-MPSP

iteration	Convergence time, sec	Miss dist., m
7	0.94	0.41

that Constrained QS-MPSP has converged in 7 iterations by spending 0.94 sec, and the miss-distance is $0.41m$, which is fairly acceptable. The time taken by the algorithm will be reduced significantly (more than 100 times) if the simulation is performed in 'C' language with a dedicated embedded processor.

3.4 Comparison with Impact-Angle Control Guidance

In this section the comparison study is performed between Constrained QS-MPSP and Impact Angle Control Guidance (IACG) given in [44]. First the engagement is performed by using Biased PN and the interception point is considered as PIP for IACG. The engagements are shown in Fig. 11. The trajectories generated by the guidance algorithms are different. The lead-angles for the guidance algorithms are shown in Fig. 12. As expected, the IACG performs better that BPNG, which helps in the convergence process of QS-MPSP as well. However, the lead-angle constraint is getting violated at the beginning, which can have the grave implication of no/delayed starting of the terminal homing phase. Moreover, the vertical acceleration (Fig. 14) obtained by IACG violets the acceleration limit by a large amount. The horizontal acceleration is within

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

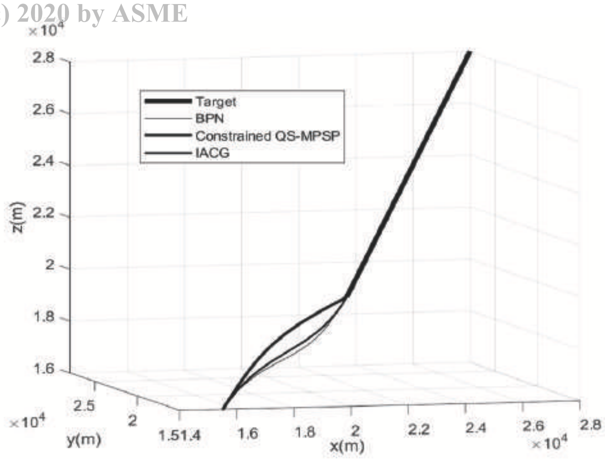

Fig. 11 Comparison of Engagement

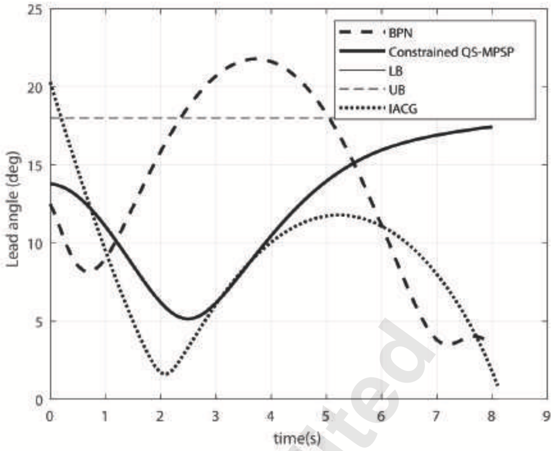

Fig. 12 Comparison of Lead Angles

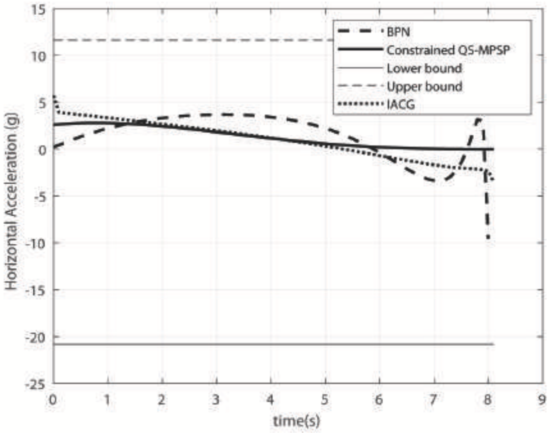

Fig. 13 Comparison of Horizontal Accelerations

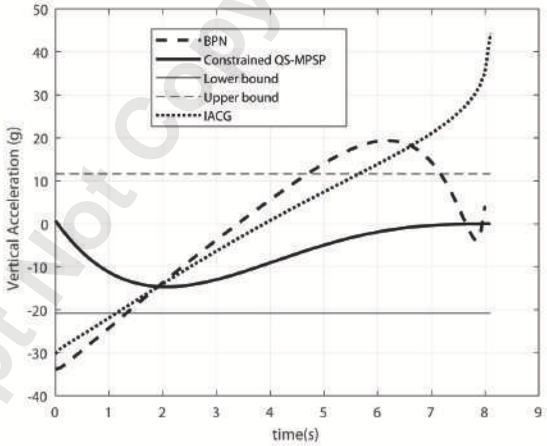

Fig. 14 Comparison of Vertical Accelerations

the limit but pulls little more 'g' than Constrained QS-MPSP, especially towards the end. Hence, the overall performance of Constrained QS-MPSP is much better as compared to the IACG.

3.5 Variation of Flight path angle

In this case, only the flight path angle of the interceptor (γ_m) is varied, keeping other initial conditions of the interceptor fixed as shown in Tab. 3. Target initial conditions are the same as shown in Tab. 1. The performance of Constrained QS-MPSP with variation of flight path angle is shown in Tab. 4. Four cases are considered where flight path angle is varied from 40 deg to 52 deg. It can be observed that the algorithm is converged within $6-8$ iterations in short time, and miss distances are acceptable (less than $1m$). Engagement plots with Constrained QS-MPSP for these cases are shown in Fig. 15. Magnified view of the engagements in $X-Z$ plane, and around interception is also shown in this figure. Trajectories generated for these cases are shown with different line formats. The curvature of the trajectories is resulted due to the difference in lateral acceleration generated by Constrained QS-MPSP. Vertical and horizontal accelerations are shown in figs. 16 and 17, respectively. For each case, the guess control is different and therefore the bound are also different (since

Table 3 Initial Conditions of the engagement for variation in γ_m

Vehicle	ψ_m, deg	V_m, ms^{-1}	x_m, m	y_m, m	z_m, m
Interceptor	39	1200	15500	15000	16000

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Table 4 Performance of Constrained QS-MPSP for variation in γ_m

Case	γ_m, deg	iteration	Convergence time, sec	Miss dist., m
1	40	8	1.06	0.39
2	44	6	0.79	0.66
3	48	6	0.77	0.32
4	52	6	0.89	0.71

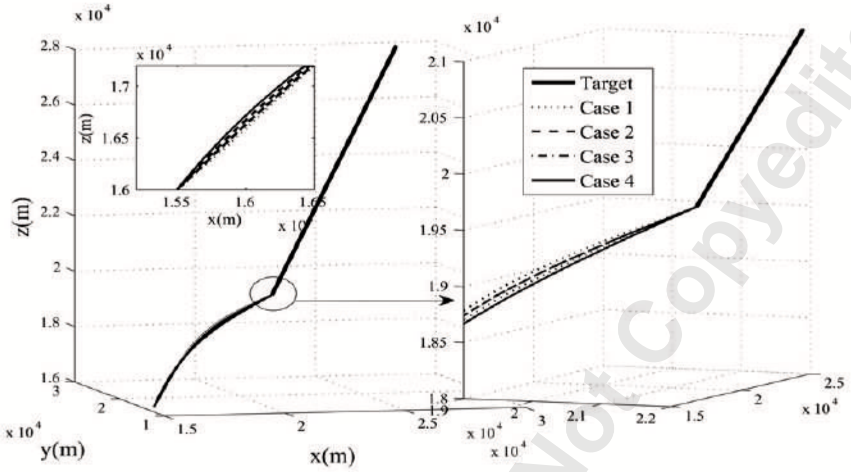

Fig. 15 Engagement with varying γ_m

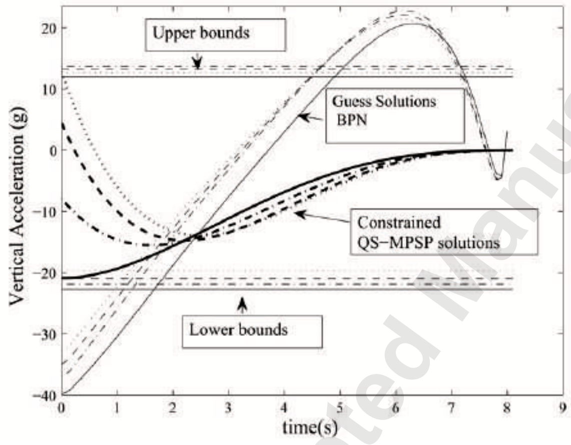

Fig. 16 Vertical Acceleration with varying γ_m

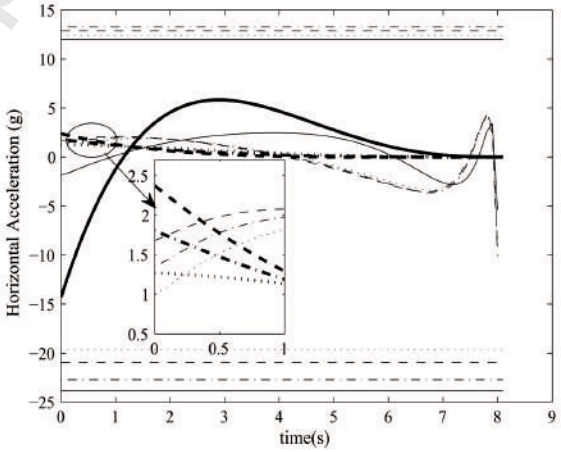

Fig. 17 Horizontal Acceleration with varying γ_m

the bound is considered as 60% of lateral acceleration generated by BPN). Similar line formats are used to represent the lateral accelerations (generated by both BPN and Constrained QS-MPSP) and the bounds for each case. Since the flight path angles are varied, a significant change in vertical acceleration for all the cases can be observed. Lead angles generated by BPN and Constrained QS-MPSP are shown in Fig. 18. Lead angle generated by BPN in each case violets the bound (18 deg) for a significant amount of time (about $30-35\%$ of flight time), and Constrained QS-MPSP makes the lead angle to confine within the bound for all the time of flight.

3.6 Variation of Heading path angle

In this section, heading angle is varied keeping other initial conditions fixed. Target initial conditions are considered same as given in Tab. 1. Four cases are considered where heading angle (ψ_m) is varied from 30 deg to 50 deg keeping the

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

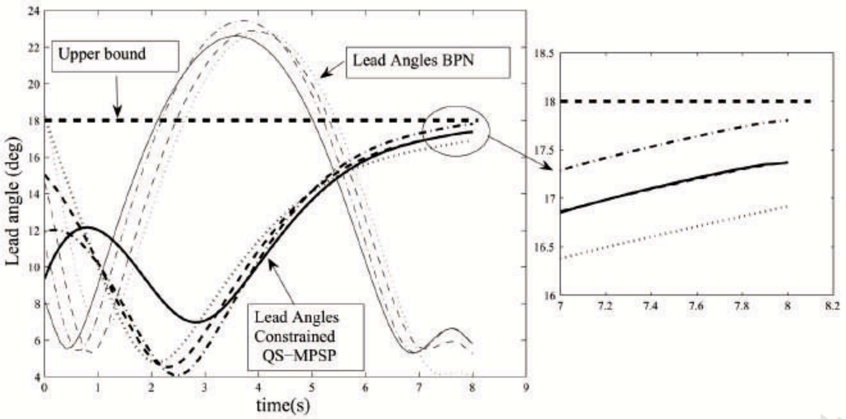

Fig. 18 Lead angle with varying γ_m

Table 5 Initial Conditions of the engagement for variation in ψ_m

Vehicle	γ_m, deg	V_m, ms^{-1}	x_m, m	y_m, m	z_m, m
Interceptor	44	1200	15500	15000	16000

other initial conditions fixed as shown in Tab. 5. The performance of Constrained QS-MPSP is shown in Table 6.

Table 6 Performance of Constrained QS-MPSP for variation in ψ_m

[h!]

Case	ψ_m, deg	iteration	Convergence time, sec	Miss dist., m
1	30	6	0.83	0.41
2	36	6	0.85	0.67
3	42	6	0.86	0.34
4	50	5	0.77	0.64

Table 6 shows that Constrained QS-MPSP is converged within $5-6$ iterations in a short time and miss distances are in acceptable range (less than 1 m). Engagements for all cases in Tab. 6 with Constrained QS-MPSP are shown in Fig. 19. This figure also shows the magnified view of engagement in $X-Y$ plane and around interception. Trajectory generated for each case is shown with different line format. Lateral acceleration generated for each case is shown in figs. 20 and 21, respectively. Lateral accelerations and bounds for each case are shown in a similar line format. Vertical accelerations in Fig. 20 change by a small amount while the horizontal accelerations change significantly. It happens due to the fact that the heading angles are changed, keeping the flight path angle fixed. The guess vertical control for all cases is very close to each other. For this reason, the bounds are very closely placed. A magnified view of these bounds is shown to make them visible. A portion of the accelerations is also magnified for better clarity. The lead angle and its magnified view are shown in Fig. 22. Lead angle generated by BPN violets the bound for $30-35\%$ of flight time, but Constrained QS-MPSP has restricted it within the bound for all the flight time.

3.7 Variation of Constraint bounds

This section describes the performance of Constrained QS-MPSP with variation in constraint bound. The initial conditions considered for this purpose are shown in Tab. 7. In previous cases the bound on lead angle is considered as 18 deg. In this study, the bound is varied by 1 deg. Therefore, the bounds on the lead angle have become $17, 18$, and 19 deg. The performance of Constrained QS-MPSP corresponding to each case are shown in Tab. 8. It can be observed from the table that Constrained QS-MPSP requires $7-9$ iterations to converge. It converges in 9 iterations for case 1 because it is a tighter bound compared to other bounds. Engagements and their magnified views around interception are shown in Fig.

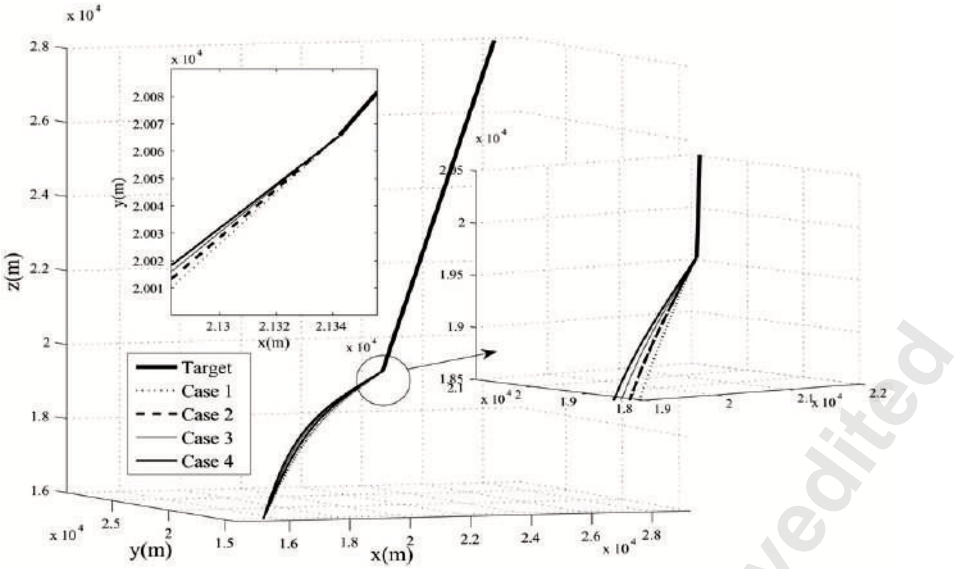

Fig. 19 Engagement with varying ψ_m

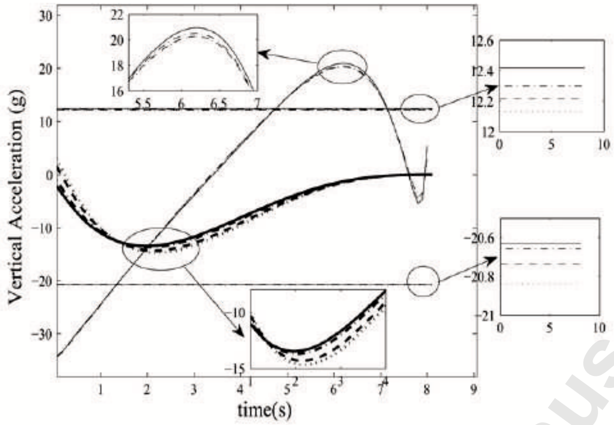

Fig. 20 Vertical Acceleration with varying ψ_m

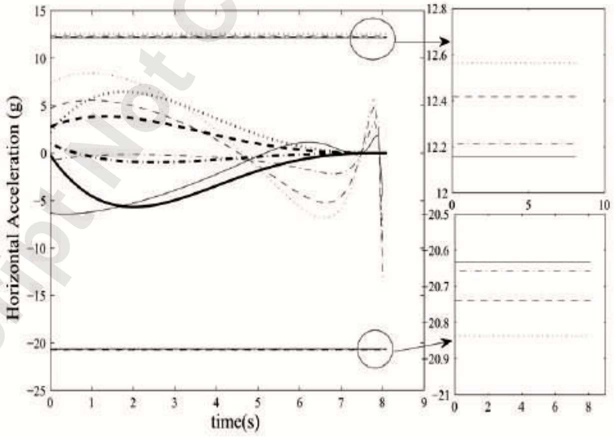

Fig. 21 Horizontal Acceleration with varying ψ_m

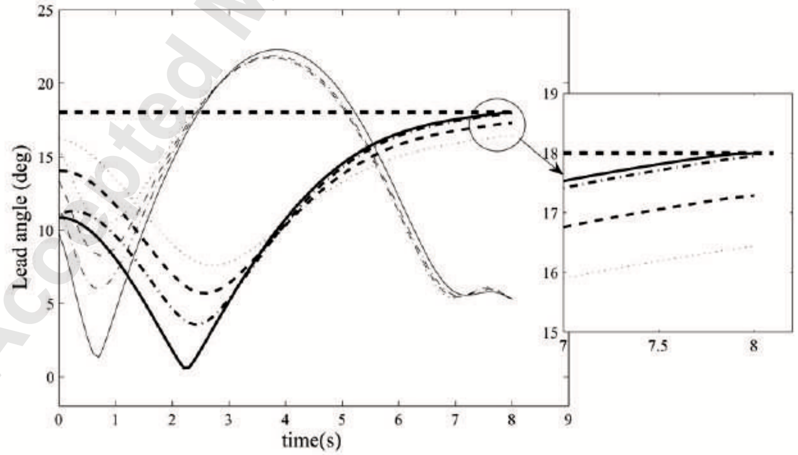

Fig. 22 Lead angle with varying ψ_m

23. Trajectories corresponding to each case are shown in different line formats. Lateral accelerations generated by BPN

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Table 7 Initial Conditions of the engagement for variation in Constraint bounds

Vehicle	γ_m, deg	ψ_m, deg	V_m, ms^{-1}	x_m, m	y_m, m	z_m, m
Interceptor	41	39	1200	15500	15000	16000

Table 8 Variation in Constraint bounds

Case	Lead Angle bound, deg	iteration	Convergence time, sec	Miss dist., m
1	17	9	1.36	0.81
2	18	7	1.02	0.62
3	19	7	0.97	0.66

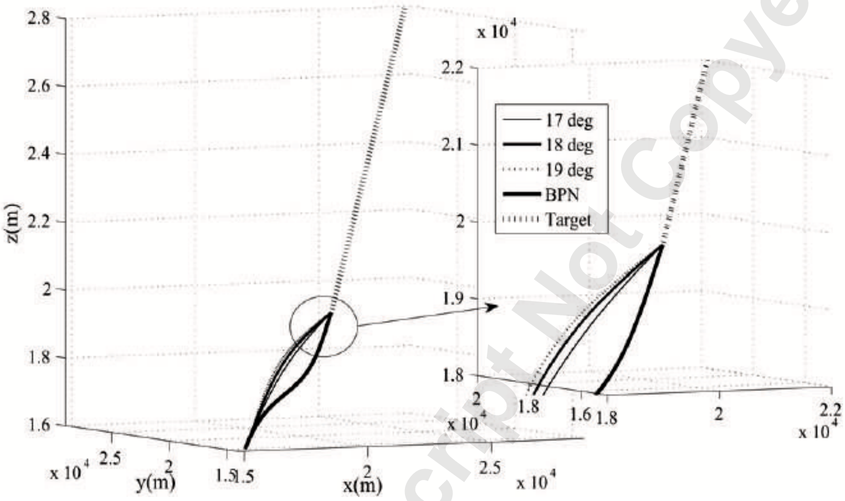

Fig. 23 Engagement for varying Constraint bounds

and Constrained QS-MPSP are shown in figs. 24 and 25 respectively. Horizontal accelerations in Fig. 25 varies more than vertical acceleration (Fig. 24). The lead angle for BPN and Constrained QS-MPSP are shown in Fig. 26. It can be observed

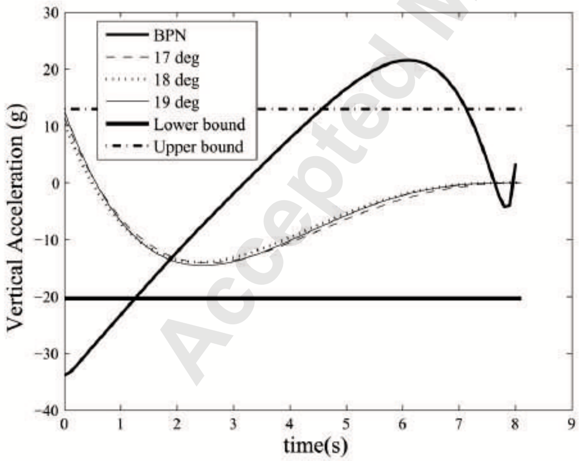

Fig. 24 Vertical Acceleration for varying bounds

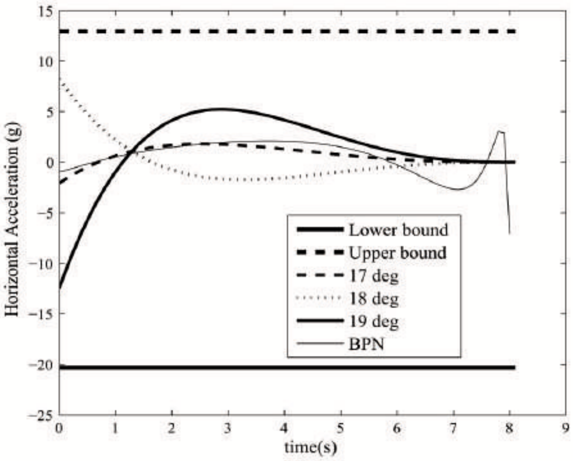

Fig. 25 Horizontal Acceleration for varying bounds

that the lead angle history generated by Constrained QS-MPSP varies for each case. Magnified view of these lead angle histories is also shown in this figure. It is clear that Constrained QS-MPSP generates the lateral accelerations in such a way that lead angle constraints are satisfied for each case.

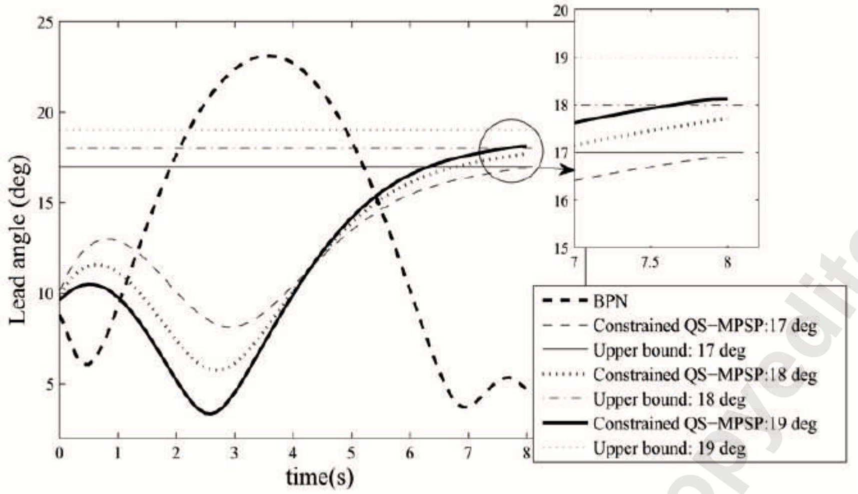

Fig. 26 Lead Angle for varying Constraint bounds

3.8 Variation of Barrier Parameter μ

In this section, the barrier parameter μ of the Interior-Point method is varied, keeping all the initial conditions fixed as shown in Tab. 7. The values of μ are varied from $0.09-0.0005$, and the lead angle corresponding to these values is shown in Fig. 27. It can be observed that the lead angle closes the boundary with a decrease in value of μ. Therefore, the user can select a value for 'μ' to push it closer to the boundary.

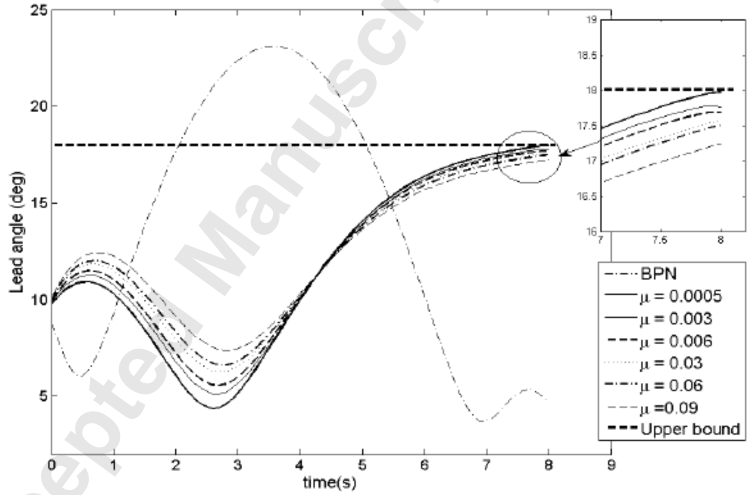

Fig. 27 Lead angle with varying Barrier Parameter μ

4 Validation of Constrained QS-MPSP Guidance with Six-DOF Simulation

The guidance commands generated in the guidance loop are validated using Six-DOF simulation studies. Guidance commands are essentially the lateral accelerations that are required for maneuvering of the interceptor in vertical and horizontal planes. In this study, Constrained QS-MPSP is used as a guidance algorithm to generate the lateral acceleration commands

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

for a 3D engagement with point mass dynamics. Then these guidance commands are provided as input to autopilot and six-DOF dynamics to verify the lead angle satisfies the constraint.

4.1 Six-DOF Plant Dynamics

The six-DOF model [45] of the interceptor is considered in inertial and fin frame. The inertial frame is considered as a launcher fixed VEN (Vertical-East -North) frame. The fins are the control surfaces which ate attached to the tail of the missile. The fin frame has its x axis aligned to X_b, but Y_f and Z_f axes are rotated about X_b by an angle of 45^o, as shown in Figs. 28 and 29.

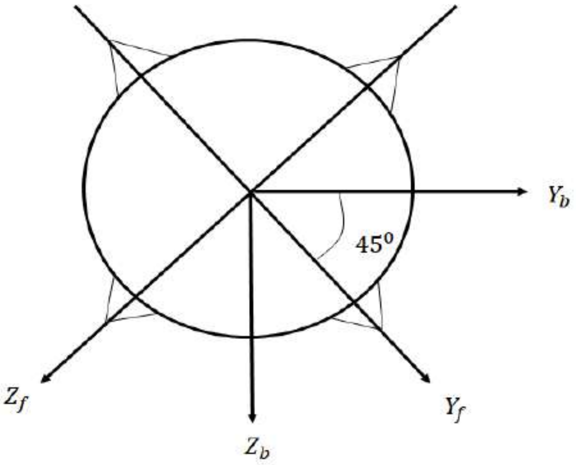

Fig. 28 Vertical Acceleration for varying bounds

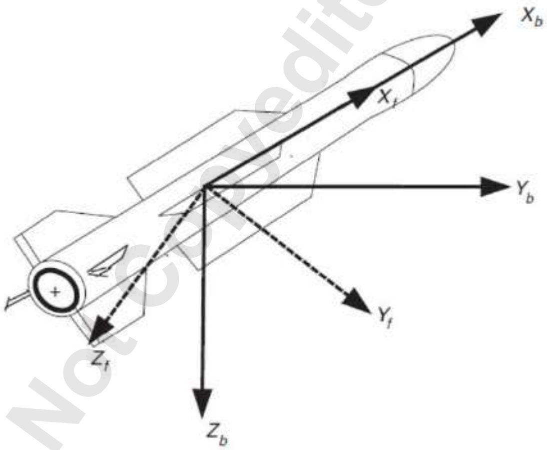

Fig. 29 Horizontal Acceleration for varying bounds

Six-DOF dynamics is given as follows

$$\dot{u} = vr - wq - \frac{QSC_D}{m} - Q_{11}g \tag{70}$$

$$\dot{v} = wp - ur + \frac{QSC_{NBN}}{m} - \frac{QSC_{N_\delta}}{m}\delta_y - \frac{(Q_{11}+Q_{31})}{\sqrt{2}}g \tag{71}$$

$$\dot{w} = uq - vp + \frac{QSC_{NAN}}{m} - \frac{QSC_{N_\delta}}{m}\delta_p - \frac{(Q_{31}-Q_{21})}{\sqrt{2}}g \tag{72}$$

$$\dot{p} = \frac{1}{I_{XX}}\left(-QSdC_{R_m} - QSdC_{l_\zeta}\delta_r + \frac{QSd^2C_{L_p}p}{2V_m}\right) \tag{73}$$

$$\dot{q} = \frac{1}{I_{YY}}\left(-QSdC_{M_{XCGA}} - QSC_{N_\delta}(X_{CP_\delta} - X_{CG})\delta_p - (I_{XX} - I_{ZZ})pr + \frac{QSd^2C_{M_q}q}{2V_m}\right) \tag{74}$$

$$\dot{r} = \frac{1}{I_{ZZ}}\left(-QSdC_{M_{XCGA}} - QSC_{N_\delta}(X_{CP_\delta} - X_{CG})\delta_y - (I_{YY} - I_{XX})pq + \frac{QSd^2C_{N_R}r}{2V_m}\right) \tag{75}$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

$$\dot{q}_1 = \frac{1}{2}(q_4p - q_3q + q_2r) \tag{76}$$

$$\dot{q}_2 = \frac{1}{2}(q_3p - q_4q + q_1r) \tag{77}$$

$$\dot{q}_3 = \frac{1}{2}(-q_2p + q_1q + q_4r) \tag{78}$$

$$\dot{q}_4 = \frac{1}{2}(-q_1p - q_2q - q_3r) \tag{79}$$

$$\begin{bmatrix}\dot{x}_m\\ \dot{y}_m\\ \dot{z}_m\end{bmatrix} = T_F^I \begin{bmatrix}u\\ v\\ w\end{bmatrix} \tag{80}$$

$$\dot{\zeta} = p \tag{81}$$

The velocity components along the fin frame axes are u, v, and w. The body rates along these axes are p, q, and r. Fin frame attitude is expressed in terms of quaternions q_1, q_2, q_3, and q_4. The interceptor position in the inertial frame is given by x_m, y_m, and z_m. It can be mentioned that the roll attitude of the interceptor need not change. The primary reason for the setting is rolling creates difficulties in seeker gimbal stabilization and difficulties in implementing the required fin deflections. Therefore the roll is stabilized at some preferred roll attitude, which is achieved by introducing a new roll state (ζ) in the dynamics (Eqn. (81)). The parameters used in the six-DOF model are described in Tab. 9. Aerodynamic coefficients in the six-DOF model have been given in Tab. 10. The body to inertial frame transformation is achieved by a transformation matrix T_I^B.

Table 9 Description of the Parameters

Notation	Description
Q	Dynamic pressure in $Pa(N/m^2)$
δ	Fin (control surfaces) deflections in deg
S	Reference area in m^2
m	Mass of the interceptor in kg
d	Diameter in m
g	Acceleration due to gravity m/s^2

$$T_I^B = \begin{bmatrix}Q_{11} & Q_{12} & Q_{13}\\ Q_{21} & Q_{22} & Q_{23}\\ Q_{31} & Q_{32} & Q_{33}\end{bmatrix} \tag{82}$$

The matrix components of T_I^B in Eqn.(82), i.e., $Q_{11}, ..., Q_{33}$ are the functions of the quaternion q_1, q_2, q_3, q_4. As discussed earlier, the transformation from the body to fin frame is achieved by a rotating matrix

$$T_B^F = \begin{bmatrix}1 & 0 & 0\\ 0 & \cos\pi/4 & \sin\pi/4\\ 0 & -\sin\pi/4 & \cos\pi/4\end{bmatrix} \tag{83}$$

Consequently, the transformation matrix from inertial to fin frame can be written as $T_I^F = T_B^F T_I^B$. The total velocity magnitude is calculated as $V_m = \sqrt{u_b^2 + v_b^2 + w_b^2}$, where, u_b, v_b, and w_b are the velocity components along the body axes X_b, Y_b, and Z_b, respectively.

$$\begin{bmatrix}u\\ v\\ w\end{bmatrix} = T_B^F \begin{bmatrix}u_b\\ v_b\\ w_b\end{bmatrix} \tag{84}$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Table 10 Aerodynamic Coefficients and their units

Parameters	Description
I_{XX}, I_{YY}, I_{ZZ}	Moment of inertia about body axes $kg-m^2$
C_D	Drag coefficient
C_{NAN}	Pitch force coefficient
C_{NBN}	Yaw force coefficient
C_{N_δ}	Tail normal force coefficient per unit δ
C_{R_M}	Rolling moment coefficient
C_{l_ζ}	Roll moment control coefficient per unit roll deflection
$C_{L_p}, C_{M_Q}, C_{N_R}$	Damping derivatives
C_{MXCGA}	Pitching and yawing moment coefficients
$X_{CP\delta}$	Tail moment arm (with respect to nose) m
X_{CG}	Axial position of center of gravity from nose m

The guidance commands are realized by the autopilot designed using Dynamic Inversion. It can be noted that the rotational dynamics is faster than translational, which is known as the time-scale separation property. Nonlinear Dynamic Inversion philosophy [46] exploits this property to remove the problem, which appears due to the non-minimum phase behaviour of the airframe transfer function. Nonlinear Dynamic Inversion controller designed for the interceptor has two loops, as shown in Fig. 30. The slow dynamic variables (α and β) are used in the outer loop, and the fast dynamic variables (p,q,r) are used in the inner loop. The outer loop variables are controlled by the fast dynamics. The inner loop controller uses the fin deflection to control the fast dynamics.

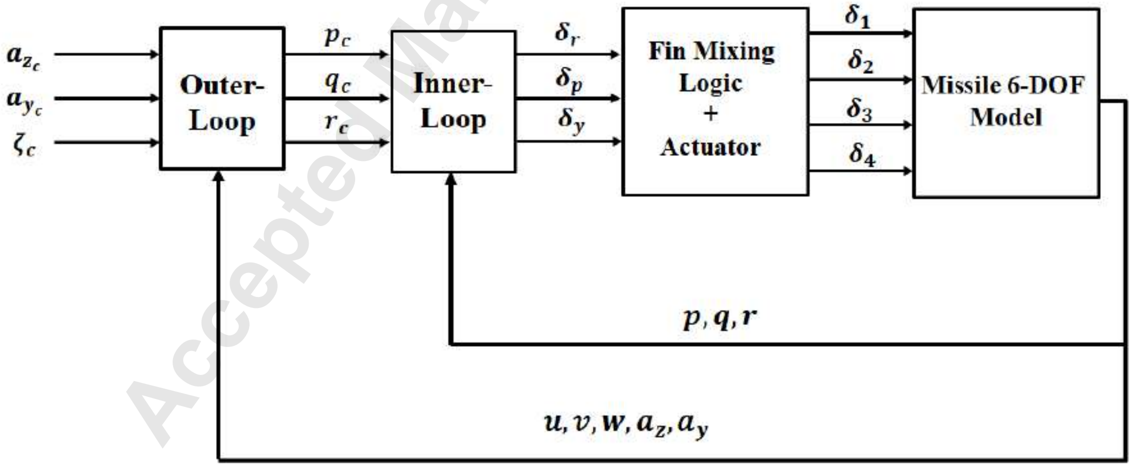

Fig. 30 Block Diagram of Autopilot

4.2 Outer Loop of DI

The objective of the outer loop is to produce required body rates (roll (p_c), pitch (q_c), and yaw (r_c)). After obtaining the guidance command ($a_{z_c},\ a_{y_c}$), commanded side-slip angle β_c, and angle-of-attack α_c are calculated as

$$\beta_c = \frac{ma_{y_c}}{QSC_{y_\beta}} \tag{85}$$

$$\alpha_c = \frac{ma_{z_c}}{QSC_{N_\alpha}} \tag{86}$$

where, $Q = \frac{\rho V_m^2}{2}$. It can be observed that $\dot{\alpha}_c$ and $\dot{\beta}_c$ can be obtained by using the following equation

$$\begin{bmatrix} \dot{\alpha}_c \\ \dot{\beta}_c \end{bmatrix} = \begin{bmatrix} \dot{a}_{z_c}\frac{m}{QSC_{N_\alpha}} \\ \dot{a}_{y_c}\frac{m}{QSC_{Y_\beta}} \end{bmatrix} \tag{87}$$

Eqn.(87) needs $\dot{a}_{z_c}$ and $\dot{a}_{y_c}$ which are calculated by enforcing a first order dynamics written as [47]

$$\begin{bmatrix} \dot{a}_{y_c} \\ \dot{a}_{z_c} \end{bmatrix} = \begin{bmatrix} \omega_\alpha(a_{y_c} - a_y) \\ \omega_\beta(a_{z_c} - a_z) \end{bmatrix} \tag{88}$$

The commanded body rates are generated using the dynamics $(\zeta_c, \dot{\alpha}, \dot{\beta})$ which are given as follows [47]

$$\begin{bmatrix} p_c \\ q_c \\ r_c \end{bmatrix} = \begin{bmatrix} \omega_\zeta(\zeta_c - \zeta) \\ \dot{\alpha}_c + \left[(p\cos\alpha + r\sin\alpha)\sin\beta + \frac{1}{mV_m}(-0.5\rho V^2 SC_L - mg_{vel})\right] \\ \frac{1}{-\cos\alpha}\left[\dot{\beta}_c - p\sin\alpha - \frac{1}{mV_m}(0.5\rho V^2 SC_Y + mg_{vel})\right] \end{bmatrix} \tag{89}$$

The aerodynamic coefficients are given in Eqn.(90).

$$\begin{bmatrix} C_D \\ C_L \\ C_Y \\ C_N \\ C_y \end{bmatrix} = \begin{bmatrix} C_x\cos\alpha\cos\beta + C_N\sin\alpha + C_y\sin\beta \\ C_N\cos\alpha - C_x\sin\alpha \\ C_y\cos\beta - C_x\sin\beta \\ C_{N_\alpha}\alpha + C_{N_{\delta_q}}\delta_q \\ C_{y_\beta}\beta + C_{y_{\delta_r}}\delta_r \end{bmatrix} \tag{90}$$

C_x, C_N, and C_y denote the drag force, normal force, and side force coefficient measured in body axes, respectively. C_D, C_L, and C_Y are coefficient of drag, lift, and side force calculated in the velocity frame. It can be mentioned that the velocity frame is obtained by rotating the body axes frame by an angle of α and β about Y_b and Z_b, respectively.

4.3 Inner Loop of DI

The Inner loop is designed to calculate the required control surface deflection (δ_p, δ_q, and δ_r) from the commanded body rates, which is the input to the inner loop. The desired dynamics of commanded body rates are required to obtain proper fin deflection. In this work, a second-order dynamics is used to find the desired dynamics for commanded body rates.

$$\frac{p}{p_c} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{91}$$

Therefore the desired dynamics $(\dot{p}_d, \dot{q}_d, \dot{r}_d)$ as obtained using Eqn.(91) is given as follows

$$\begin{bmatrix} \dot{p}_d \\ \dot{q}_d \\ \dot{r}_d \end{bmatrix} = \begin{bmatrix} -2\xi\omega_p p + \omega_p^2\int(p - p_c) \\ -2\xi\omega_q q + \omega_q^2\int(q - q_c) \\ -2\xi\omega_r r + \omega_r^2\int(r - r_c) \end{bmatrix} \tag{92}$$

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

After obtaining desired dynamics $\dot{p}_d, \dot{q}_d, \dot{r}_d$ in Eqn.(92), fin deflections $(\delta_r, \delta_p, \delta_y)$ are required to be calculated. For this purpose, the desired aerodynamic coefficients are to be calculated. It can be observed that Eqns.(73,74,75) can be written as

$$\dot{p} = \frac{QSd}{I_{XX}} C_l \tag{93}$$

$$\dot{q} = \frac{-(I_{XX} - I_{ZZ})pr}{I_{XX}} + \frac{QSd}{I_{YY}} C_m \tag{94}$$

$$\dot{r} = \frac{-(I_{YY} - I_{XX})pq}{I_{ZZ}} + \frac{QSd}{I_{ZZ}} C_n \tag{95}$$

where,

$$C_l = \left(-C_{R_m} - C_{l_\zeta} \delta_r + \frac{C_{L_p} pd}{2V_m} \right) \tag{96}$$

$$C_m = \left(-C_{M_{XCGA}} - C_{N_\delta} (X_{CP_\delta} - X_{CG}) \delta_p / d + \frac{C_{M_q} qd}{2V_m} \right) \tag{97}$$

$$C_n = \left(-C_{M_{XCGA}} - C_{N_\delta} (X_{CP_\delta} - X_{CG}) \delta_y / d + \frac{C_{N_R} rd}{2V_m} \right) \tag{98}$$

The value of the desired dynamics $\dot{p}_d, \dot{q}_d, \dot{r}_d$ from Eqn.(92) are substituted in Eqns.(93-95) to obtain desired moment coefficients as follows

$$\begin{bmatrix} C_{l_d} \\ C_{m_d} \\ C_{n_d} \end{bmatrix} = \begin{bmatrix} \frac{\dot{p}_d I_{XX}}{QSd} \\ \frac{(\dot{q}_d + (I_{XX} - I_{ZZ})pr/I_{YY})I_{YY}}{QSd} \\ \frac{(\dot{r}_d + (I_{YY} - I_{XX})pq/I_{ZZ})I_{ZZ}}{QSd} \end{bmatrix} \tag{99}$$

After obtaining the desired moment coefficients in Eqn.(99), the required deflections are obtained using Eqns.(96)-(98) as follows [47]

$$\begin{bmatrix} \delta_r \\ \delta_p \\ \delta_y \end{bmatrix} = \begin{bmatrix} \frac{C_{l_d} + C_{R_m} - C_{L_p} pd/2V_m}{C_{l_\zeta}} \\ -\frac{C_{m_d} + C_{M_{XCGA}} - C_{M_q} qd/2V_m}{(C_{N_\delta}(X_{CP_\delta} - X_{CG})/d)} \\ -\frac{C_{m_d} + C_{M_{XCGA}} - C_{N_R} rd/2V_m}{(C_{N_\delta}(X_{CP_\delta} - X_{CG})/d)} \end{bmatrix} \tag{100}$$

The fin deflection generated by the inner loop controller is used to find the required control surface deflection. The required control surfaces deflections are obtained by a Fin-Mixing logic. Then the control surfaces are deflected to achieve the maneuver.

4.4 Simulation Study

The initial conditions for the simulation study are given in Tab. 11. The maximum value of the lead angle is considered as 28^o. Constrained QS-MPSP took four iterations and $0.83s$ to converge, with a miss distance of $0.15m$. Commanded lateral acceleration is generated in the velocity frame using Constrained QS-MPSP. For realizing the commanded lateral acceleration (a_{y_c}, a_{z_c}) they are required to transfer to the fin frame. Demanded lateral acceleration in the velocity frame and fin frame are shown in Figs. 31 and 32 respectively. The Lead angle history during the fight time for Six-DOF simulation is shown in Fig. 33. It can be observed that the lead angle obtained for the six-DOF model satisfies the constraints. Therefore, the target always remains within the FOV of the seeker, which is required for a successful interception. The velocity components of the missile in fin frame is given in Fig. 34. Commanded and achieved pitch and yaw rates are shown in Figs. 35 and 36, respectively. In Fig. 39 all aerodynamic angles are plotted together. Commanded roll, pitch, and yaw deflections are shown in Fig. 37. These deflections are realized by the fin deflections. Therefore the commanded deflections are given as inputs to the fin mixing logic to obtain deflection for the four fins. The deflections of the four fins are shown in Fig. 38.

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

Table 11 Initial Conditions and performance of Constrained QS-MPSP

Cases	x_{m_0}, m	y_{m_0}, m	z_{m_0}, m	γ_{m_0}, deg	ψ_{m_0}, deg	V_{m_0}, m/s	iter	time, s	miss dist., m
1	13000	9800	9000	48	48	1000	4	0.93	0.17

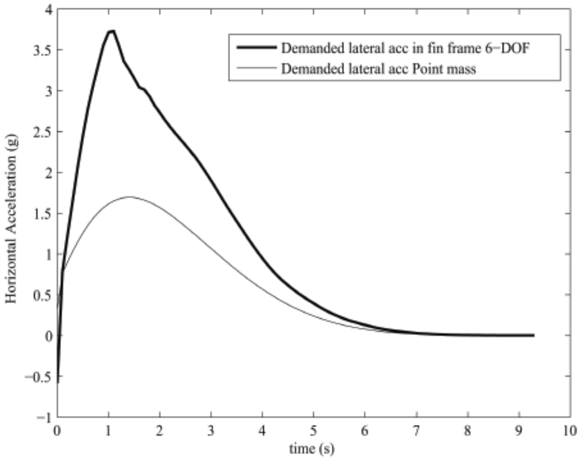

Fig. 31 Commanded Horizontal Acceleration

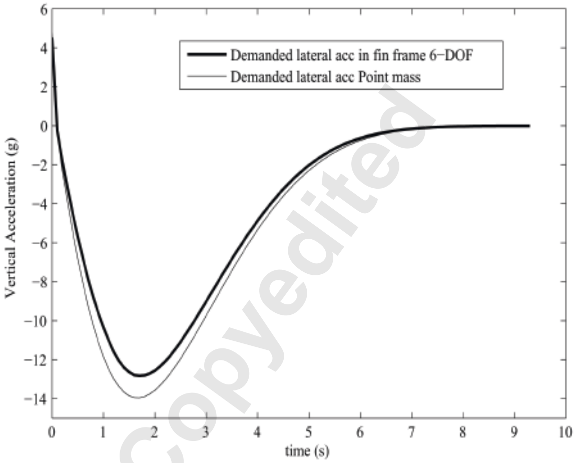

Fig. 32 Commanded Vertical Acceleration

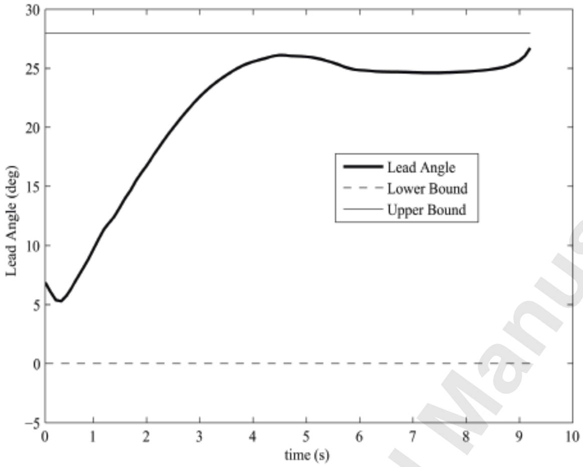

Fig. 33 Lead angle for Six-DOF model

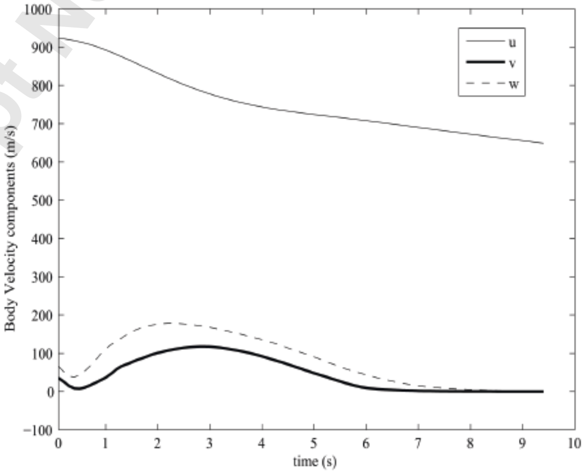

Fig. 34 Velocity components of Missile in fin frame

5 Conclusion

The State and control constrained optimal control problem is successfully converted into a low-dimensional nonlinear programming problem using the 'Constrained Quasi-Spectral MPSP' philosophy. The spectral representation of the control variable, as well as conversion of all applicable path constraints to equivalent control constraints, results in reducing the number of optimizing variables significantly. This reduction of the size of the optimization problem is explained through quantitative analysis of computational complexity as well. The Constrained QS-MPSP is found to be capable of handling the constraints on both the lead angle and lateral acceleration during engagement of an incoming ballistic target. The performance of the algorithm is analyzed in terms of iteration, convergence time, and miss distance. In the opinion of the authors, it is sufficiently efficient to be applied in real-time.

Journal of Dynamic Systems, Measurement and Control. Received October 20, 2019;
Accepted manuscript posted September 22, 2020. doi:10.1115/1.4048488

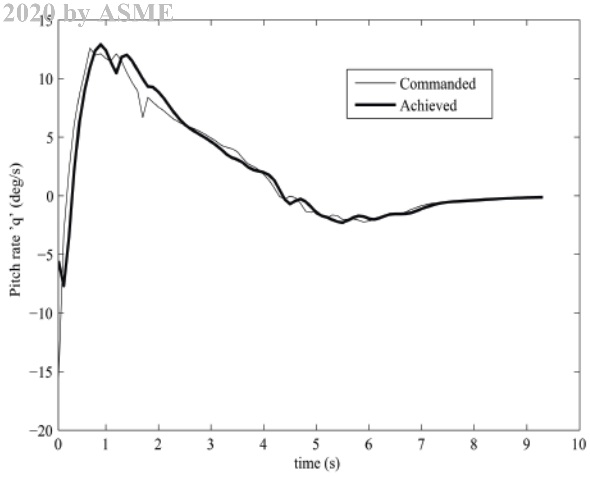

Fig. 35 Pitch rate

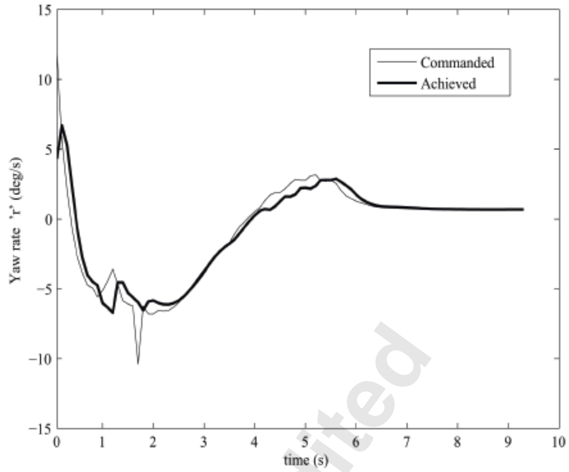

Fig. 36 Yaw rate

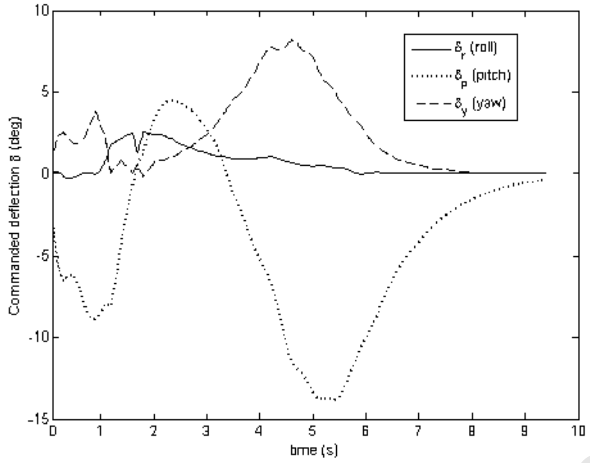

Fig. 37 Fin deflection

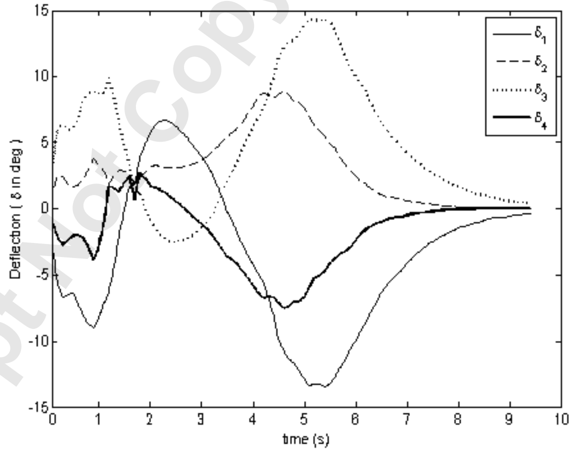

Fig. 38 Roll, pitch and yaw deflection

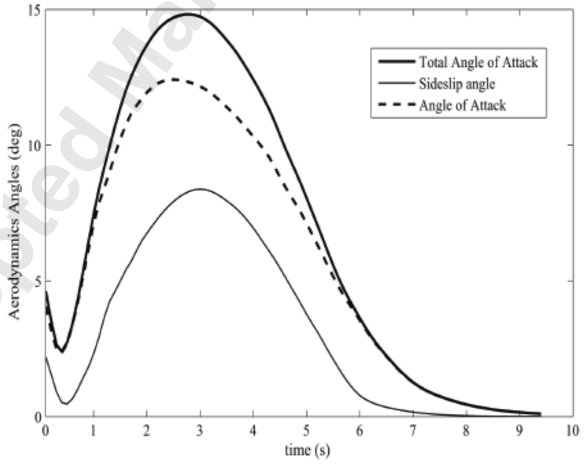

Fig. 39 α, β, and α_T

References

[1] Kirk, D. E., 1970. *Optimal Control Theory: An Introduction*. Prentice Hall.
[2] Sage, A., 1968. *Optimum Systems Control*. Networks series. Prentice-Hall.

[3] Bryson, J. A. E., and Ho, Y.-C., 1975. *Applied Optimal Control: Optimization, Estimation and Control*. Hemisphere Publishing Corporation, January.

[4] Hager, W. W., and Pardalos, P. M., 2013. *Optimal control: theory, algorithms, and applications*, Vol. 15. Springer Science & Business Media.

[5] Hull, D. G., 2013. *Optimal control theory for applications*. Springer Science & Business Media.

[6] Longuski, J. M., Guzmán, J. J., and Prussing, J. E., 2014. *Optimal control with aerospace applications*. Springer.

[7] Ben-Asher, J. Z., 2010. *Optimal control theory with aerospace applications*. American institute of aeronautics and astronautics.

[8] Ross, I., 2015. *A Primer on Pontryagin's Principle in Optimal Control: Second Edition*. Collegiate Publishers.

[9] Naidu, D. S., 2003. *Optimal Control Systems*. CRC Press, Florida, USA.

[10] Morrison, D. D., Riley, J. D., and Zancanaro, J. F., 1962. "Multiple shooting method for two-point boundary value problems". *Commun. ACM,* **5**(12), Dec., pp. 613–614.

[11] Larson, R., and Casti, J., 1982. *Principles of Dynamic Programming: Advanced theory and applications*. Control and Systems Theory. M. Dekker.

[12] Betts, J. T., 2001. *Practical Methods for Optimal Control Using Nonlinear Programming*. Advances in design and control. Society for Industrial and Applied Mathematics.

[13] Wang, L., 2009. *Model predictive control system design and implementation using MATLAB®*. Springer Science & Business Media.

[14] Allgöwer, F., and Zheng, A., 2012. *Nonlinear Model Predictive Control*. Progress in Systems and Control Theory. Birkhäuser Basel.

[15] Chen, H., and Allgöwer, F., 1998. "A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability". *Automatica,* **34**(10), pp. 1205 – 1217.

[16] Mayne, D., Rawlings, J., Rao, C., and Scokaert, P., 2000. "Constrained model predictive control: Stability and optimality". *Automatica,* **36**(6), pp. 789 – 814.

[17] Rawlings, J. B., Angeli, D., and Bates, C. N., 2012. "Fundamentals of economic model predictive control". In 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), pp. 3851–3861.

[18] Wang, Y., and Boyd, S., 2010. "Fast model predictive control using online optimization". *IEEE Transactions on Control Systems Technology,* **18**(2), pp. 267–278.

[19] Fahroo, F., and Ross, I. M., 2002. "Direct trajectory optimization by a chebyshev pseudospectral method". *Journal of Guidance, Control, and Dynamics,* **25**(1), Jan., pp. 160–166.

[20] Gong, Q., Fahroo, F., and Ross, I. M., 2008. "Spectral algorithm for pseudospectral methods in optimal control". *Journal of Guidance, Control, and Dynamics,* **31**(3), May, pp. 460–471.

[21] Balakrishnan, S. N., and Biega, V., 1996. "Adaptive-critic based neural networks for aircraft optimal control". *Journal of Guidance, Control and Dynamics,* **19**(4), July-August, pp. 893–898.

[22] Padhi, R., and Kothari, M., 2009. "Model predictive static programming: a computationally efficient technique for suboptimal control design". *International journal of innovative computing, information and control,* **5**(2), pp. 399–411.

[23] Halbe, O., Raja, R. G., and Padhi, R., 2014. "Robust reentry guidance of a reusable launch vehicle using model predictive static programming". *Journal of Guidance, Control, and Dynamics,* **37**(1), pp. 134–148.

[24] Maity, A., Padhi, R., Mallaram, S., Rao, G. M., and Manickavasagam, M., 2016. "A robust and high precision optimal explicit guidance scheme for solid motor propelled launch vehicles with thrust and drag uncertainty". *International Journal of Systems Science,* **47**(13), pp. 3078–3097.

[25] Kumar, P., Anoohya, B. B., and Padhi, R., 2018. "Model predictive static programming for optimal command tracking: A fast model predictive control paradigm". *ASME Journal of Dynamic Systems, Measurement, and Control,* **141**(2), Oct., pp. 021014–021014–12.

[26] Mathavaraj, S., and Padhi, R., 2019. "Unscented mpsp for optimal control of a class of uncertain nonlinear dynamic systems". *Journal of Dynamic Systems, Measurement, and Control,* **141**(6).

[27] Sakode, C. M., and Padhi, R., 2014. "Computationally efficient suboptimal control design for impulsive systems based on model predictive static programming". *IFAC Proceedings Volumes,* **47**(1), pp. 41 – 46. 3rd International Conference on Advances in Control and Optimization of Dynamical Systems (2014).

[28] Dwivedi, P. N., Bhattacharya, A., and Padhi, R., 2011. "Suboptimal midcourse guidance of interceptors for high-speed targets with alignment angle constraint". *Journal of Guidance, Control, and Dynamics,* **34**(3), pp. 860–877.

[29] Oza, H. B., and Padhi, R., 2012. "Impact-angle-constrained suboptimal model predictive static programming guidance of air-to-ground missiles". *Journal of Guidance, Control, and Dynamics,* **35**(1), pp. 153–164.

[30] Sachan, K., and Padhi, R., 2019. "Waypoint constrained multi-phase optimal guidance of spacecraft for soft lunar landing". *Unmanned Systems,* **07**(02), pp. 83–104.

[31] Mondal, S., and Padhi, R., 2018. "Angle-constrained terminal guidance using quasi-spectral model predictive static programming". *Journal of Guidance, Control, and Dynamics,* **41**(3), pp. 783–791.

[32] Manchester, I. R., and Savkin, A. V., 2006. "Circular-navigation-guidance law for precision missile/target engagements". *Journal of guidance, control, and dynamics,* **29**(2), pp. 314–320.

[33] Park, B., Jeon, B., Kim, T., Tahk, M., and Kim, Y., 2012. "Composite guidance law for impact angle control of tactical missiles with passive seekers". In Asia-Pacific International Symposium on Aerospace Technology, pp. 13–15.

[34] Kim, T.-H., Park, B.-G., and Tahk, M.-J., 2013. "Bias-shaping method for biased proportional navigation with terminal-angle constraint". *Journal of Guidance, Control, and Dynamics,* **36**(6), pp. 1810–1816.

[35] Tekin, R., and Erer, K. S., 2015. "Switched-gain guidance for impact angle control under physical constraints". *Journal of Guidance, Control, and Dynamics*.

[36] Erer, K. S., Tekin, R., and Özgören, M. K., 2015. "Look angle constrained impact angle control based on proportional navigation". In _ AIAA Guidance, Navigation, and Control Conference _, pp. 6–2015.

[37] Park, B.-G., Kim, T.-H., and Tahk, M.-J., 2016. "Range-to-go weighted optimal guidance with impact angle constraint and seeker's look angle limits". *IEEE Transactions on Aerospace and Electronic Systems,* **52**(3), pp. 1241–1256.

[38] Shaferman, V., 2017. "Optimal guidance with an in route look-angle constraint". In AIAA Guidance, Navigation, and Control Conference, p. 1507.

[39] Mondal, S., and Padhi, R., 2018. "State and input constrained missile guidance using spectral model predictive static programming". In 2018 AIAA Guidance, Navigation, and Control Conference, p. 1584.

[40] Mondal, S., and Padhi, R., 2018. "Angle-constrained terminal guidance using quasi-spectral model predictive static programming". *Journal of Guidance, Control, and Dynamics,* **41**(3), pp. 783–791.

[41] Wang, Y., and Boyd, S., 2009. "Fast model predictive control using online optimization". *IEEE Transactions on control systems technology,* **18**(2), pp. 267–278.

[42] Lovelly, T. M., and George, A. D., 2017. "Comparative analysis of present and future space-grade processors with device metrics". *Journal of Aerospace Information Systems,* **14**(3), pp. 184–197.

[43] Akhil, G., and Ghose, D., 2012. "Biased pn based impact angle constrained guidance using a nonlinear engagement model". In American Control Conference (ACC), 2012, IEEE, pp. 950–955.

[44] Kim, J., Cho, N., and Kim, Y., 2019. "Field-of-view constrained impact angle control guidance guaranteeing error convergence before interception". In AIAA Scitech 2019 Forum, p. 1927.

[45] Padhi, R., Chawla, C., and Das, P. G., 2014. "Partial integrated guidance and control of interceptors for high-speed ballistic targets". *Journal of Guidance, Control, and Dynamics*.

[46] Enns, D., Bugajski, D., Hendrick, R., and Stein, G., 1994. "Dynamic inversion: an evolving methodology for flight control design". *International Journal of control,* **59**(1), pp. 71–91.

[47] Dwivedi, P. N., Bhattacharya, A., and Padhi, R., 2011. "Suboptimal midcourse guidance of interceptors for high-speed targets with alignment angle constraint". *Journal of Guidance, Control, and Dynamics,* **34**(3), pp. 860–877.

2020-10-15

Constrained quasi-spectral MPSP with application to high-precision missile guidance with path constraints

Mondal, Sabyasachi

American Society of Mechanical Engineers (ASME)

Mondal S, Padhi R. (2021) Constrained quasi-spectral MPSP with application to high-precision missile guidance with path constraints, Journal of Dynamic Systems, Measurement, and Control, Volume 143, Issue 3, March 2021, Article Number 031001

https://doi.org/10.1115/1.4048488