Quad-Segment Polynomial Trajectory Guidance for Impact-Time Control of Precision-Munition Strike

MARK SNYDER, Member, IEEE
ZHIHUA QU, Fellow, IEEE
University of Central Florida
Orlando, FL, USA

RICHARD HULL, Member, IEEE
UTC Aerospace Systems
Kissimmee, FL, USA

RICHARD PRAZENICA, Member, IEEE
Embry Riddle Aeronautical University
Daytona Beach, FL, USA

Accurate calculation of time to go until impact is an essential component in coordinating multiple precision-guided munitions for a simultaneous strike. Conventional means for calculating time to go contain potential errors, due in part to the trajectory curvatures, and may fall short on the accuracy required to achieve a desired impact time, especially if velocity changes rapidly due to atmospheric drag and gravity. A closed-loop guidance law will be presented in which an optimized, piecewise continuous, and smooth family of polynomial trajectories is parameterized in terms of downrange distance and has a closed-form solution of arc length for improved calculation of time to go.

Manuscript received June 3, 2014; revised March 21, 2015, September 26, 2015, March 2, 2016; released for publication May 20, 2016.

DOI. No. 10.1109/TAES.2016.140420.

Refereeing of this contribution was handled by M.-J. Tahk.

This work was supported in part by L-3 Communications Link Simulation and Training.

Authors' addresses: M. Snyder, Z. Qu, Department of Electrical and Computer Engineering, University of Central Florida, EECS, 4000 Central Florida Blvd., Orlando, FL 32816; R. Hull, 2805 Scenic Lane, Kissimmee, FL 34744; R. Prazenica, Embry Riddle Aeronautical University, 600 S. Clyde Morris Blvd., Daytona Beach, FL 32114. Corresponding author is M. Snyder, E-mail: (marksnyder93@gmail.com).

I. INTRODUCTION

Missiles and precision-guided munitions form an integral part of modern offensive warfare, and the need for weaponry to act in a coordinated or cooperative manner will be a necessary component of future advanced warfare. In this context, the term *cooperative* implies that multiple airborne weapons have the ability to formulate a consensus, in either a distributed or a centralized manner, on certain guidance-related parameters in order to neutralize a target more effectively. A cooperative missile strike becomes necessary if simultaneous arrival of warheads is required to overwhelm defensive countermeasures such as a close-in weapons system, or if arrival on target in spaced time intervals is necessary as in the case of bunker neutralization. For interceptor applications, multiple simultaneous engagement poses a better probability of target kill.

The cooperation of weapons boils down to a primary core capability: the ability to control each weapon's impact time. Calculation of impact time results directly from the time-to-go parameter and is thus considered to be a large component of the feedback source in control design. In general, however, controlling impact time is quite difficult, since most missiles and projectiles possess reduced degrees of control freedom due to the fact that velocity is uncontrollable. Such systems are considered to be underactuated. In addition, calculating time to go using existing methods, such as range-over-range-rate and range-over-missile-velocity, is subject to considerable error during periods of changing velocity (high drag acceleration) and/or excessive trajectory curvature. The issues of underactuation and the potential for error in the time-to-go calculation pose significant control-design challenges.

Searching the open literature, we can segregate existing work into six groups. In Group 1 [1–3], we find works which formulate optimal guidance laws with constraints on impact angle. In these works, missile velocity is considered constant and linearized models of either the missile kinematics or engagement kinematics are used. While numerous papers concern optimal guidance with terminal-angle constraints, very few works pertaining to impact-time constraints can be found. These comprise Group 2. In this group, [4] formulates an optimal guidance law with constraints on impact angle as well as impact time, and [5] considers a time-optimal formulation with impact-angle constraints. In [6], a Lyapunov-based guidance design for impact-time control is chosen. Ref. [7] considers an optimal formulation for impact-time control alone. An interesting note about this work is that the independent variable in the state model is in terms of distance instead of time. This is done to keep the terminal boundary condition from becoming explicit in impact time. In each of these works, missile velocity is constant and standard formulations for time to go are used. In most of the cases, linearized models are also used. Ref. [8] considers time-of-flight control as well as range extension

for precision munitions. Atmosphere and disturbances are considered in this work, which makes it suitable for some general performance comparisons in the simulation study.

Attempts to improve estimates of time to go for both classical and modern guidance laws comprise Group 3 [9–15]. In most of these works, linear models are used in the formulation, and missile velocity and acceleration profiles are assumed to be fully known or modeled with linear or linear piecewise equations. No in-flight correction due to disturbance is proposed. Ref. [9] attempts to improve on this by deriving a recursive algorithm to correct the time-to-go estimate when either a heading-angle error is present or changes in path length occur. It does not take into account changes in time to go due to drag or atmospheric disturbances. The change in path length is estimated from the deviation from a straight-line path. The works contained in Group 4 [16–18] considers the development of optimal guidance laws when the missile velocity is not constant. In [17], the derived guidance law relies on predicted velocity profiles which can be updated to compensate for error. However, the law is derived using linearized models and is explicit in a terminal time-boundary constraint. Searching the literature further, we find scarce results for Group 5: cooperative guidance laws for missiles. In [19] and [20], optimal formulations are derived to address the issue of cooperative salvo attack on a target; however, both works assume constant velocity, and in [20], the time-to-go estimation methods of [9] are utilized. In [19], attempts to estimate time to go are done using an arc-length approximation of the expected engagement. The resulting equation is approximated, since few arc-length algorithms result in closed-form solutions. Ref. [21] discusses work in cooperative-missile research and the problems associated with.

In Group 6, a path-planning approach to guidance synthesis is taken. Ref. [22] takes a somewhat similar approach to quad-segment polynomial trajectory (QSPT) guidance in that distance is segmented and a spline-based trajectory is developed. However, that work uses the standard range-over-missile-velocity for time-to-go calculation, and the terminal boundary constraints are a function of the impact time. Each of the six groups discussed contains one or more potential sources of error due to the linearization of the equations of motion, the time-to-go calculation used, or the constant-velocity assumption.

In contrast to each of the groups discussed, QSPT, introduced in [23], is an optimized guidance law that is nonexplicit in time to go, has an exact closed-form solution for arc length for an improved time-to-go calculation, and does not rely on small-angle approximation or the constant-velocity assumption in order to gain a closed-form solution. The novelty of QSPT exists in the availability of a closed-form solution for arc length. This provides an advantage in that the arc-length-based time-to-go calculation is unaffected by trajectory curvature, thus removing a significant component of error found in current methods. A second advantage to this approach is that the optimization (as well as the entire guidance law) is nonexplicit in time to go. A third advantage is that trajectory synthesis is independent of the time parameter. This provides the ability to conduct accurate preflight analysis along prescribed trajectories using high-fidelity drag and atmospheric models to produce tabular reference models of impact time for use in the corresponding impact-time control design. The individual elements of the tabular models are referenced with respect to downrange distance and not with respect to time. This avoids problematic timing issues between actual in-flight data and the reference model.

This paper is organized as follows: Section II formulates the general problem, presents the equations of motion, defines the configuration space, and specifies the arc-length equations. The design and implementation of the closed-loop guidance law are also presented. Section III derives the QSPT coefficients required for trajectory design. Section IV details the trajectory optimization. Section V gives a step-by-step process of designing a QSPT trajectory. Section VI details the derivation of the time-to-go equations as well as the procedure for conducting preflight analysis. Section VII derives the impact-time control law, and Section VIII provides extensive simulation analysis of the proposed design.

II. PROBLEM FORMULATION

The problem of achieving a prescribed impact time for a guided munition at a stationary target located at some terminal position in the downrange/cross-range plane is considered. The trajectory is analytically designed in closed form, and the length of the trajectory is analytically calculated. Although projectile velocity changes due to gravity, drag, and atmosphere, the time to go can be estimated online based upon the current velocity and the remainder of the trajectory (not merely the downrange to the target). For this work, initial conditions for generation of the guidance solution are considered to be trajectory apogee after boost phase or the deployment point, such as the bay door of an aircraft.

A. Equations of Motion

The configuration space is shown in Fig. 1; the equations of motion used for synthesis of the guidance law are

$$\dot{V} = -a_{\mathrm{d}} - g\sin(\gamma) \tag{1}$$

$$\dot{\chi} = \frac{1}{V\cos(\gamma)}u_{\chi} \tag{2}$$

$$\dot{\gamma} = \frac{1}{V}\left(u_{\gamma} + g\cos(\gamma)\right) \tag{3}$$

$$\dot{x} = V\cos(\gamma)\cos(\chi) \tag{4}$$

$$\dot{y} = V\cos(\gamma)\sin(\chi) \tag{5}$$

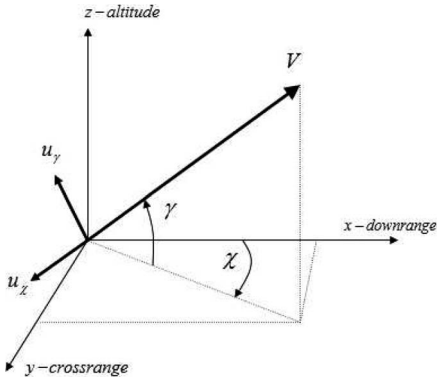

Fig. 1. Configuration space.

$$\dot{z} = V \sin(\gamma), \tag{6}$$

where the set of initial (subscript 0) and terminal (subscript T) conditions is given by

$$\begin{cases} (x_0, y_0), (x_0, z_0), (\chi_0, \gamma_0) \\ (x_\mathrm{T}, y_\mathrm{T}), (x_\mathrm{T}, z_\mathrm{T}), (\chi_\mathrm{T}, \gamma_\mathrm{T}). \end{cases} \tag{7}$$

The velocity vector V is contained within the configuration space consisting of downrange x, cross-range y, and altitude z. Divert controls u_χ and u_γ in (2) and (3) are normal to V; acceleration due to drag a_d is in the negative direction of V; and χ and γ are, respectively, the heading and flight-path angles. The value of gravity g used in this work is 9.81 m/s^2. Equation (1) contains unknown drag a_d to be compensated for by trajectory planning and corresponding impact-time control. A family of trajectories satisfying (4)–(6) is prescribed next.

B. Trajectory Parameterization

In the proposed development, segments of cross-range and altitude trajectories are chosen analytically within the family of second-order polynomials of the form

$$\begin{aligned} y_n(x) &= a_n^y + c_n^y x + \kappa_n^y x^2 \\ z_n(x) &= a_n^z + c_n^z x + \kappa_n^z x^2, \end{aligned} \tag{8}$$

where $n \in \{1, \cdots, N\}$ denotes the index of trajectory segments. Parameterization of the trajectories should be chosen to satisfy dynamic equations (4)–(6) or, equivalently,

$$\frac{\partial y}{\partial x} = \tan(\chi), \quad \frac{\partial z}{\partial x} = \frac{\tan(\gamma)}{\cos(\chi)}, \tag{9}$$

which are obtained by dividing (4) into (5) and (6), respectively. In fact, the equations in (9) are satisfied for all times if the trajectories in (8) satisfy (9) at the initial and terminal conditions given by (7). Therefore, the six boundary conditions provided by (7) map into the path-planning boundary conditions required by (8) according to

$$\begin{cases} (x_0, y_0), \ \frac{\partial}{\partial x} y_1(x_0) = \tan(\chi_0) \\ (x_\mathrm{T}, y_\mathrm{T}), \ \frac{\partial}{\partial x} y_4(x_\mathrm{T}) = \tan(\chi_\mathrm{T}) \end{cases} \tag{10}$$

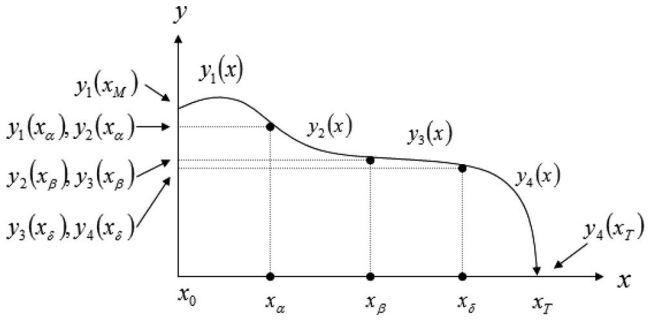

Fig. 2. Cross-range profile—segmentation of downrange.

and

$$\begin{cases} (x_0, z_0), \ \frac{\partial}{\partial x} z_1(x_0) = \frac{\tan(\gamma_0)}{\cos(\chi_0)} \\ (x_\mathrm{T}, z_\mathrm{T}), \ \frac{\partial}{\partial x} z_4(x_\mathrm{T}) = \frac{\tan(\gamma_\mathrm{T})}{\cos(\chi_\mathrm{T})}. \end{cases} \tag{11}$$

An advantage of using the second-order polynomials in the form of (8) is that arc length can be calculated in closed form for ease of time-to-go calculation, and such a computation is free from the typical error due to curvature. Specifically, the arc length of the trajectories in (8) is given by

$$\begin{aligned} S &= \int ds \\ &= \int_{x_0}^{x_\mathrm{f}} \sqrt{1 + \left(\frac{\partial}{\partial x} y_n(x)\right)^2 + \left(\frac{\partial}{\partial x} z_n(x)\right)^2}\, dx, \end{aligned} \tag{12}$$

where $ds \triangleq \sqrt{(dx)^2 + (dy)^2 + (dz)^2}$ is the incremental length. The following section illustrates the basic structure of a QSPT trajectory.

C. QSPT Guidance

The trajectories prescribed by (8) are *each* composed of $N = 4$ individual segments. The choice of four segments provides enough design coefficients for the satisfaction of initial and terminal boundary conditions as well as enough remaining coefficients to satisfy the boundary conditions required to impose smooth and continuous segmentation of the trajectory. Under the $N = 4$ design choice, one free design variable remains after satisfaction of all boundary conditions, and is used for trajectory optimization as well as the trajectory planning required for impact-time control purposes. We now define the free variables to be κ_4^y for the cross-range trajectory and κ_4^z for the altitude trajectory. From this point forward, development of QSPT coefficient equations will be carried out for the cross-range only, unless otherwise specified. In every case, the derived equations apply directly to the altitude trajectory, given the appropriate boundary conditions.

Fig. 2 illustrates the general structure of a QSPT trajectory, where x_0 and x_T denote the initial and terminal downrange positions and x_α, x_β, and x_δ are internal downrange segmentation points.

A specific choice is that the downrange x is divided into four equal sections:

$$\begin{aligned} x_\alpha &= x_0 + \frac{1}{4}(x_\mathrm{T} - x_0) \\ x_\beta &= x_0 + \frac{1}{2}(x_\mathrm{T} - x_0) \\ x_\delta &= x_0 + \frac{3}{4}(x_\mathrm{T} - x_0). \end{aligned} \tag{13}$$

Initial and terminal boundary conditions for $y_n(x)$ are enforced according to (10). The altitude trajectory, of course, would be subject to (11). Internal boundary conditions are used to impose the smooth and continuous segmentation of the trajectory and are defined at the downrange locations of (13). The sets of internal boundary conditions for $y_n(x)$ are given as

$$\begin{cases} y_1(x_\alpha) = y_2(x_\alpha) \\ \dfrac{\partial}{\partial x} y_1(x_\alpha) = \dfrac{\partial}{\partial x} y_2(x_\alpha) \end{cases} \tag{14}$$

$$\begin{cases} y_2(x_\beta) = y_3(x_\beta) \\ \dfrac{\partial}{\partial x} y_2(x_\beta) = \dfrac{\partial}{\partial x} y_3(x_\beta) \\ \dfrac{\partial^2}{\partial x^2} y_2(x_\beta) = \dfrac{\partial^2}{\partial x^2} y_3(x_\beta) \end{cases} \tag{15}$$

$$\begin{cases} y_3(x_\delta) = y_4(x_\delta) \\ \dfrac{\partial}{\partial x} y_3(x_\delta) = \dfrac{\partial}{\partial x} y_4(x_\delta). \end{cases} \tag{16}$$

In (14), only position and the first partial are enforced as boundary conditions, which allows this location to be a point of inflection of the trajectory, as shown in Fig. 2. This is also true for (16). On the other hand, (15) enforces the second-order partial, which is required in order to join the first two segments with the last two in a smooth, continuous manner. The coefficient equations that result from enforcing the internal boundary conditions are given in Section III. In that section, coefficient time-derivative equations are also developed which are required for the closed-loop guidance design presented next.

D. Guidance Design and Implementation

The open-loop guidance law is given from (2)–(4) as

$$\begin{aligned} u_\chi &= \frac{\partial \chi}{\partial x} V^2 \cos^2(\gamma)\cos(\chi), \\ u_\gamma &= \frac{\partial \gamma}{\partial x} V^2 \cos(\gamma)\cos(\chi) - g\cos(\gamma), \end{aligned} \tag{17}$$

where, according to (8) and (9), the open-loop angular profiles derived in terms of QSPT are given as

$$\begin{aligned} \chi &= \tan^{-1}\left(c_n^y + 2\kappa_n^y x\right) \\ \gamma &= \tan^{-1}\left(\left(c_n^z + 2\kappa_n^z x\right)\cos\left(\tan^{-1}\left(c_n^y + 2\kappa_n^y x\right)\right)\right) \\ &= \tan^{-1}\left(\frac{c_n^z + 2\kappa_n^z x}{\sqrt{1 + \left(c_n^y + 2\kappa_n^y x\right)^2}}\right) \end{aligned} \tag{18}$$

and the partial derivatives can be easily calculated.

Alternatively, we can implement the corresponding closed-loop guidance design. To this end, define

$$e_y = y - y_n, \quad e_z = z - z_n, \tag{19}$$

where y_n and z_n are given by (8) for the period of time when the nth segment of the trajectories is being implemented. Second-order time derivatives of (19) are explicit in the controls u_χ and u_γ through the equations

$$\begin{aligned} \ddot{y} &= -a_\mathrm{d}\cos(\gamma)\sin(\chi) - 2g\sin(\gamma)\cos(\gamma)\sin(\chi) \\ &\quad - \sin(\gamma)\sin(\chi)u_\gamma + \cos(\chi)u_\chi \\ \ddot{z} &= -a_\mathrm{d}\sin(\gamma) + g\left(\cos^2(\gamma) - \sin^2(\gamma)\right) + \cos(\gamma)u_\gamma \\ \ddot{y}_n &= 2\dot{c}_n^y\dot{x} + 2\kappa_n^y(\dot{x})^2 + 4\dot{\kappa}_n^y x\dot{x} + c_n^y\ddot{x} + 2\kappa_n^y x\ddot{x} \\ \ddot{z}_n &= c_n^z\ddot{x} + 2\kappa_n^z(\dot{x})^2 + 2\kappa_n^z x\ddot{x} \end{aligned} \tag{20}$$

where $\dot{x}$ is given from (4) and $\ddot{x}$ is given by

$$\begin{aligned} \ddot{x} &= -a_\mathrm{d}\cos(\gamma)\cos(\chi) - 2g\sin(\gamma)\cos(\gamma)\cos(\chi) \\ &\quad - \sin(\gamma)\cos(\chi)u_\gamma - \sin(\chi)u_\chi. \end{aligned} \tag{21}$$

It should be noted that the distinct differences between $\ddot{y}_n$ and $\ddot{z}_n$ in (20) relating to the coefficient time derivatives are due to the design choices imposed on the free variables κ_4^y and κ_n^z. This will be addressed in detail in Section III.B.

It follows from the second derivative of (19), and the substitution of (21) into (20), that the error system is given by

$$\begin{bmatrix} \ddot{e}_y \\ \ddot{e}_z \end{bmatrix} = A + B\begin{bmatrix} u_\chi \\ u_\gamma \end{bmatrix}, \tag{22}$$

where

$$A = \begin{bmatrix} -a_\mathrm{d}\cos(\gamma)\sin(\chi) - 2g\sin(\gamma)\cos(\gamma)\sin(\chi) \\ -\left(2\dot{c}_n^y\dot{x} + 4\dot{\kappa}_n^y x\dot{x} + 2\kappa_n^y(\dot{x})^2\right) \\ +\dfrac{\partial}{\partial x} y_n\left(a_\mathrm{d}\cos(\gamma)\cos(\chi) + 2g\sin(\gamma)\cos(\gamma)\cos(\chi)\right) \\ -a_\mathrm{d}\sin(\gamma) + g\left(\cos^2(\gamma) - \sin^2(\gamma)\right) - 2\kappa_n^z(\dot{x})^2 \\ +\dfrac{\partial}{\partial x} z_n\left(a_\mathrm{d}\cos(\gamma)\cos(\chi) + 2g\sin(\gamma)\cos(\gamma)\cos(\chi)\right) \end{bmatrix} \tag{23}$$

$$B = \begin{bmatrix} \cos(\chi) + \frac{\partial}{\partial x} y_n\sin(\chi) & -\sin(\gamma)\sin(\chi) + \frac{\partial}{\partial x} y_n\sin(\gamma)\cos(\chi) \\ \frac{\partial}{\partial x} z_n\sin(\chi) & \cos(\gamma) + \frac{\partial}{\partial x} z_n\sin(\gamma)\cos(\chi) \end{bmatrix}. \tag{24}$$

Hence, the closed-loop guidance law is

$$\begin{bmatrix} u_\chi \\ u_\gamma \end{bmatrix} = B^{-1}\left\{\begin{bmatrix} k_y(y - y_n) + k'_y(\dot{y} - \dot{y}_n) \\ k_z(z - z_n) + k'_z(\dot{z} - \dot{z}_n) \end{bmatrix} - A\right\}, \tag{25}$$

under which the actual trajectories converge asymptotically and exponentially to the guidance trajectories of y_n and z_n. The closed-loop gains are k_y, k_z, k'_y, and k'_z, and drag acceleration a_d is estimated in real time using the data from the onboard accelerometer.

E. Inversion-Matrix Singularities

The inversion matrix of (24) becomes singular if the flight-path angle reaches $\pm\pi/2$; however, the slope of second-order polynomials cannot reach vertical angles, as

that would require the partial derivative $(\partial/\partial x)y_n$ to be undefined. Therefore, the inversion of (24) can never reach a singularity under normal operation. On the other hand, it is a requirement in some cases that a precision munition reach a vertical angle in order to drop in on a target from directly above. An easy solution to this problem is to reparameterize (8) in terms of a new independent variable, such as z, and switch the guidance law accordingly. This can be done at a point when the projectile is close to the target. Under the reparameterization, (8) can operate at vertical angles and the switched-form guidance law is free of singularity at $\pm\pi/2$. In this case, we restrict the heading angle to $\chi < \pm\pi/2$, which is a reasonable constraint to impose. The process of reparameterization and the derivation of the switched guidance law are outlined in Appendix A and will be demonstrated in the simulation.

III. DESIGN OF QSPT COEFFICIENTS

Solution sets for the coefficients in (8) must be found according to initial, terminal, and internal boundary conditions. In the steps that follow, the four segments are joined together to form a single continuous trajectory. The first two steps, A1 and A2, join Segments 1 and 2, resulting in coefficient equations for $a_{1,2}^y$ and $c_{1,2}^y$. The next two steps, A3 and A4, join Segments 3 and 4, which results in equations for $a_{3,4}^y$ and $c_{3,4}^y$. The final step, A5, then joins Segments 2 and 3 and results in a vector matrix expression which gives equations for $\kappa_{1,2,3}^y$. The design of free variable κ_4^y is addressed in Sections III.B and IV.

A. Solving the Coefficients

Step A1) Solve the coefficients of Segment 1 with respect to initial boundary conditions—i.e.,

$$y_1(x) = a_1^y + c_1^y x + \kappa_1^y x^2, \tag{26}$$

where a_1^y and c_1^y are determined from the initial boundary conditions given in (10) as

$$\begin{aligned} a_1^y &= y_1(x_0) - c_1^y x_0 - \kappa_1^y x_0^2 \\ c_1^y &= \frac{\partial}{\partial x} y_1(x_0) - 2\kappa_1^y x_0. \end{aligned} \tag{27}$$

Step A2) Join Segment 1 to Segment 2 by enforcing the internal boundary conditions of (14). This requires solving the equations

$$\begin{aligned} y_1(x_\alpha) &= y_2(x_\alpha) \\ \frac{\partial}{\partial x} y_1(x_\alpha) &= \frac{\partial}{\partial x} y_2(x_\alpha) \end{aligned} \tag{28}$$

for a_2^y and c_2^y, which results in

$$\begin{aligned} a_2^y &= a_1^y + c_1^y x_\alpha + \kappa_1^y x_\alpha^2 - c_2^y x_\alpha - \kappa_2^y x_\alpha^2 \\ c_2^y &= c_1^y + 2\kappa_1^y x_\alpha - 2\kappa_2^y x_\alpha. \end{aligned} \tag{29}$$

Step A3) Solve Segment 4 subject to the terminal boundary conditions given in (10). This results in

$$\begin{aligned} a_4^y &= y_\mathrm{T} - c_4^y x_\mathrm{T} - \kappa_4^y x_\mathrm{T}^2 \\ c_4^y &= \frac{\partial}{\partial x} y_4(x_\mathrm{T}) - 2\kappa_4^y x_\mathrm{T}. \end{aligned} \tag{30}$$

Step A4) Join Segment 4 to Segment 3 by enforcing the internal boundary conditions of (16). This requires solving the boundary equations

$$\begin{aligned} y_3(x_\delta) &= y_4(x_\delta) \\ \frac{\partial}{\partial x} y_3(x_\delta) &= \frac{\partial}{\partial x} y_4(x_\delta) \end{aligned} \tag{31}$$

for a_3^y and c_3^y, which results in

$$\begin{aligned} a_3^y &= a_4^y + c_4^y x_\delta + \kappa_4^y x_\delta^2 - c_3^y x_\delta - \kappa_3^y x_\delta^2 \\ c_3^y &= c_4^y + 2\kappa_4^y x_\delta - 2\kappa_3^y x_\delta. \end{aligned} \tag{32}$$

Step A5) Join Segment 3 to Segment 2 by enforcing the boundary conditions in (15). This requires solving the equations

$$\begin{aligned} y_2(x_\beta) &= y_3(x_\beta) \\ \frac{\partial}{\partial x} y_2(x_\beta) &= \frac{\partial}{\partial x} y_3(x_\beta) \\ \frac{\partial^2}{\partial x^2} y_2(x_\beta) &= \frac{\partial^2}{\partial x^2} y_3(x_\beta), \end{aligned} \tag{33}$$

which results in a vector matrix expression. Simultaneously solving these equations for κ_1^y, κ_2^y, and κ_3^y gives

$$\begin{bmatrix} \kappa_1^y \\ \kappa_2^y \\ \kappa_3^y \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}^{-1} \begin{bmatrix} D_{11} \\ D_{21} \\ D_{31} \end{bmatrix} = C^{-1} D, \tag{34}$$

where

$$\begin{aligned} C_{11} &= -x_0^2 + x_\alpha^2 + 2x_0 x_\beta - 2x_\alpha x_\beta \\ C_{12} &= -x_\beta^2 + 2x_\alpha x_\beta - x_\beta^2 \\ C_{13} &= x_\delta^2 - 2x_\beta x_\delta + x_\beta^2 \\ C_{21} &= 2(x_0 - x_\alpha) \\ C_{22} &= 2(x_\alpha - x_\beta) \\ C_{23} &= 2(x_\beta - x_\delta) \\ C_{31} &= 0 \\ C_{32} &= -2 \\ C_{33} &= 2 \end{aligned} \tag{35}$$

and

$$\begin{aligned} D_{11} &= y_1(x_0) + \frac{\partial}{\partial x} y_1(x_0)(x_\beta - x_0) - y_4(x_\mathrm{T}) \\ &\quad + \frac{\partial}{\partial x} y_4(x_\mathrm{T})(x_\mathrm{T} - x_\beta) \\ &\quad - \kappa_4^y (x_\mathrm{T}^2 - x_\delta^2 - 2x_\mathrm{T} x_\beta + 2x_\beta x_\delta) \\ D_{21} &= \frac{\partial}{\partial x} y_1(x_0) - \frac{\partial}{\partial x} y_4(x_\mathrm{T}) - 2\kappa_4^y (x_\delta - x_\mathrm{T}) \\ D_{31} &= 0. \end{aligned} \tag{36}$$

In the next section, time derivatives of the coefficient equations are derived.

B. Solutions of Time Derivatives

The C-matrix inversion elements of (35) are strictly in terms of downrange segmentation points and, for a

stationary target, remain constant and nonsingular throughout the engagement. The equations in (36), on the other hand, are a function of free variable κ_4^y, and the elements of this 3×1 matrix can be subject to change during the engagement depending on the design choices made for κ_4^y.

As stated in Section II.C, two possible functionalities are imposed on the free variable κ_4^y. The first is an initializing requirement—that is, find κ_4^y which optimizes the trajectory in some manner. In fact, once found, the optimizing value defined as ${}^*\kappa_4^y$ remains constant throughout the engagement.

The second functionality for κ_4^y involves adjustment of the trajectory in order to change the on-target arrival time, or impact time. In this case, κ_4^y moves from the optimizing value ${}^*\kappa_4^y$ in a direction which attempts to reduce the error between a currently computed impact time and a desired impact time. Therefore, the coefficient time derivatives $\dot{a}_{1,2,3,4}^y$, $\dot{c}_{1,2,3,4}^y$, and $\dot{\kappa}_{1,2,3}^y$ are driven strictly by $\dot{\kappa}_4^y$.

In this work, we choose to involve only the cross-range in trajectory planning for the control of impact time. The altitude trajectory remains fixed at the value set by ${}^*\kappa_4^z$. A particular choice for κ_4^y can be prescribed as the state solution to the first-order system

$$\dot{\kappa}_4^y = \mu, \tag{37}$$

where the initial condition is ${}^*\kappa_4^y$ and μ is the impact-time control. The single-integrator system of (37) was chosen to ensure a smooth continuous transition of the QSPT coefficients. A recursive approach can be chosen, but care should be taken to prevent step disturbances in the trajectory.

Because the altitude trajectory remains fixed, coefficient derivatives $\dot{a}_n^z$, $\dot{c}_n^z$, and $\dot{\kappa}_n^z$ equal zero. Furthermore, the choice of bang-bang control μ given in Section VII results in impulse functions for the second-order time derivatives of (37) which are thus ignored. This is the reason for the differences in the coefficient derivatives between $\ddot{y}_n$ and $\ddot{z}_n$ in (20).

To find expressions for the cross-range coefficient derivatives, we begin with (34) and group the D matrix according to terms involving κ_4^y. From direct inspection of (36) we find

$$\begin{bmatrix} D_{11} \\ D_{21} \\ D_{31} \end{bmatrix} = \begin{bmatrix} f_{11} + g_{11}\kappa_4^y \\ f_{22} + g_{22}\kappa_4^y \\ 0 \end{bmatrix}, \tag{38}$$

where

$$\begin{aligned} f_{11} &= y_1(x_0) - y_4(x_\mathrm{T}) + \frac{\partial}{\partial x} y_1(x_0)\left(x_\beta - x_0\right) \\ &\quad + \frac{\partial}{\partial x} y_4(x_\mathrm{T})\left(x_\mathrm{T} - x_\beta\right) \\ g_{11} &= -x_\mathrm{T}^2 + x_\delta^2 + 2x_\mathrm{T}x_\beta - 2x_\beta x_\delta \\ f_{22} &= \frac{\partial}{\partial x} y_1(x_0) - \frac{\partial}{\partial x} y_4(x_\mathrm{T}) \\ g_{22} &= 2\left(x_\mathrm{T} - x_\delta\right), \end{aligned} \tag{39}$$

which then results in

$$\begin{bmatrix} \kappa_1^y \\ \kappa_2^y \\ \kappa_3^y \end{bmatrix} = C^{-1}\left[\begin{bmatrix} f_{11} \\ f_{22} \\ 0 \end{bmatrix} + \begin{bmatrix} g_{11} \\ g_{22} \\ 0 \end{bmatrix}\kappa_4^y\right]. \tag{40}$$

Matrix C^{-1} and the equations in (39) are constant. Given (37), the derivative of (40) is therefore

$$\begin{bmatrix} \dot{\kappa}_1^y \\ \dot{\kappa}_2^y \\ \dot{\kappa}_3^y \end{bmatrix} = C^{-1}\begin{bmatrix} g_{11} \\ g_{22} \\ 0 \end{bmatrix}\mu. \tag{41}$$

Then, finding the coefficient derivatives of (27), (29), (30), and (32) gives

$$\begin{aligned} \dot{a}_1^y &= -\dot{c}_1^y x_0 - \dot{\kappa}_1^y x_0^2 \\ \dot{c}_1^y &= -2\dot{\kappa}_1^y x_0 \\ \dot{a}_2^y &= \dot{a}_1^y + \dot{c}_1^y x_1 + \dot{\kappa}_1^y x_\alpha^2 - \dot{c}_2^y x_0 - \dot{\kappa}_2^y x_\alpha^2 \\ \dot{c}_2^y &= \dot{c}_1^y + 2\dot{\kappa}_1^y x_1 - 2\dot{\kappa}_2^y x_1 \\ \dot{a}_3^y &= \dot{a}_4^y + \dot{c}_4^y x_\delta + x_\delta^2\mu - \dot{c}_3^y x_\delta - \dot{\kappa}_3^y x_\delta^2 \\ \dot{c}_3^y &= \dot{c}_4^y + 2x_\delta\mu - 2\dot{\kappa}_3^y x_3 \\ \dot{a}_4^y &= -\dot{c}_4^y x_\mathrm{T} - x_\mathrm{T}^2\mu \\ \dot{c}_4^y &= -2x_\mathrm{T}\mu. \end{aligned} \tag{42}$$

The coefficient time-derivative equations of (41) and (42) are initialized to values set by (27), (29), (30), (32), and (34) with the proper optimizing values of ${}^*\kappa_4^y$ and ${}^*\kappa_4^z$ chosen.

IV. MINIMIZATION OF PERFORMANCE INDEX

A minimization of control energy can be found using a one-time numerical line search of κ_4^y and κ_4^z values which minimize a given performance index and result in ${}^*\kappa_4^y$ and ${}^*\kappa_4^z$. A basic performance index for the minimization of control energy is

$$J = \frac{1}{2}\int_{t_0}^{t_\mathrm{f}} \mathbf{u}^\mathrm{T}\mathbf{R}\mathbf{u}dt, \tag{43}$$

where the control

$$\mathbf{u} = \begin{bmatrix} u_\chi & u_\gamma \end{bmatrix}^\mathrm{T} \tag{44}$$

is a vector of the open-loop controls given by (17) and superscript T denotes the transpose. For convenience, the 2×2 weighting matrix $\mathbf{R}$ is chosen as unity. Two options can be chosen for the minimization of (43). First, the open-loop controls of (17) can be written in terms of QSPT using the angular profiles given by (18). Then J can be numerically integrated and values for κ_n^y and κ_n^z can be found that minimize J. A less involved method results in a second option where each trajectory is minimized separately by assuming either $\chi = 0$ or $\gamma = 0$. To that end, if $\gamma = 0$ is chosen, the open-loop controls of (17) become

$$\begin{aligned} u_\chi &= V^2\frac{\partial\chi}{\partial x}\cos(\chi) \\ u_\gamma &= 0, \end{aligned} \tag{45}$$

with the corresponding equation for χ given for the nth segment as

$$\chi_n = \tan^{-1}\left(c_n^y + 2\kappa_n^y x\right), \tag{46}$$

from which the partial derivative required by u_χ is easily obtained. In light of (45), the performance index reduces to

$$J = \frac{1}{2}\int_{t_0}^{t_f} u_\chi^2 dt, \tag{47}$$

which is a minimization of control energy for the cross-range trajectory. Substituting (46) into (45) and then (45) with the corresponding partial derivative into (47) gives the performance index for the nth segment as

$$J_n = \frac{1}{2}\int_{t_0}^{t_f}\left[\frac{2\kappa_n^y V^2}{1+\left(c_n^y + 2\kappa_n^y x\right)^2}\cos\left(\tan^{-1}\left(c_n^y + 2\kappa_n^y x\right)\right)\right]^2 dt, \tag{48}$$

which simplifies to

$$J_n = \int_{t_0}^{t_f}\frac{2\left(\kappa_n^y\right)^2 V^4}{\left(1+\left(c_n^y + 2\kappa_n^y x\right)^2\right)^3} dt. \tag{49}$$

For convenience, a change in the variable of integration can be done with the transformation (see Appendix B)

$$dt = \frac{\sqrt{1+\left(c_n^y + 2\kappa_n^y x\right)^2}}{V} dx, \tag{50}$$

which when substituted for dt in (49) gives

$$J_n = \int_{x_0}^{x_f}\frac{2\left(\kappa_n^y\right)^2 V^3}{\left(1+\left(c_n^y + 2\kappa_n^y x\right)^2\right)^{\frac{5}{2}}} dx. \tag{51}$$

A closed-form solution for the integral of (51) is then given as

$$J_n = \frac{\kappa_n^y V^3\left(2\left(c_n^y\right)^3 + 12\left(c_n^y\right)^2\kappa_n^y x + 24c_n^y\left(\kappa_n^y\right)^2 x^2 + 3c_n^y + 16\left(\kappa_n^y\right)^3 x^3 + 6\kappa_n^y x\right)}{3\left[1+\left(c_n^y + 2\kappa_n^y x\right)^2\right]^{3/2}}, \tag{52}$$

which when evaluated at the boundary conditions for the corresponding nth segment can be minimized with respect to κ_4^y as

$$J = \min_{\kappa_4^y}\left\{J_1|_{x_0}^{x_\alpha} + J_2|_{x_\alpha}^{x_\beta} + J_3|_{x_\beta}^{x_\delta} + J_4|_{x_\delta}^{x_T}\right\}, \tag{53}$$

which results in the optimizing value ${}^*\kappa_4^y$. The minimization of (53) involves a simple numerical line search of κ_4^y values at the initialization stage before launch. This same process is followed for the choice of $\chi = 0$, which results in an altitude-trajectory minimization of control energy. If gravity is ignored, the same general design equations result.

V. SUMMARY OF QSPT ALGORITHM

The following steps are required in order to implement (8) and create an optimized QSPT trajectory.

Step B1) Choose initial and terminal boundary conditions for (8) given by (10) and (11). Initial conditions will be the initial position and attitude of the missile at the beginning of the engagement. Terminal conditions will be the target's current position and the desired impact angle.

Step B2) Choose the independent variable and implement (13) to determine numerical values for the segmentation locations.

Step B3) Find the optimizing values ${}^*\kappa_4^y$ and ${}^*\kappa_4^z$.

Step B4) Solve the coefficients of (27), (29), (30), (32), and (34) each for y_n and z_n. For a basic trajectory requiring no impact-time control, all coefficients are computed once and remain constant. If impact-time control is to be utilized, these coefficient values become initializing values for the coefficient time derivatives of Section III.

Step B5) Implement the closed-loop guidance law of (25).

VI. TIME TO GO

With all of the necessary parameters required to create a trajectory now defined, a solution for time to go can be formulated in terms of arc length and the velocity along the arc as

$$T_{go} = \frac{S}{V}, \tag{54}$$

where S is given from (12) and is continually updated as the missile's current downrange position moves from the initial position to the terminal position.

To compute the initial arc length, the integrations required by (12) must span all four segments, subject to the following bounds of integration, as

$$S = \int_{x_0}^{x_\alpha}\sqrt{1+\left(\frac{\partial}{\partial x}y_1(x)\right)^2 + \left(\frac{\partial}{\partial x}z_1(x)\right)^2} dx + \int_{x_\alpha}^{x_\beta}\sqrt{1+\left(\frac{\partial}{\partial x}y_2(x)\right)^2 + \left(\frac{\partial}{\partial x}z_2(x)\right)^2} dx$$

$$+\int_{x_\beta}^{x_\delta}\sqrt{1+\left(\frac{\partial}{\partial x}y_3(x)\right)^2+\left(\frac{\partial}{\partial x}z_3(x)\right)^2}\,dx$$
$$+\int_{x_\delta}^{x_T}\sqrt{1+\left(\frac{\partial}{\partial x}y_4(x)\right)^2+\left(\frac{\partial}{\partial x}z_4(x)\right)^2}\,dx, \quad (55)$$

and a general solution of the integrals for the nth segment is given as

$$S_n=\frac{1}{4}\left[\frac{a+b\ln(c)+d\ln(2)}{\left(\left(\kappa_n^y\right)^2+\left(\kappa_n^z\right)^2\right)^{3/2}}\right], \quad (56)$$

where

$$a=\left[\kappa_n^y\frac{\partial}{\partial x}y_n(x)+\kappa_n^z\frac{\partial}{\partial x}z_n(x)\right]\sqrt{\left(\kappa_n^y\right)^2+\left(\kappa_n^z\right)^2}$$
$$\times\sqrt{1+\left(\frac{\partial}{\partial x}y_n(x)\right)^2+\left(\frac{\partial}{\partial x}z_n(x)\right)^2}$$
$$b=\left(\kappa_n^y\right)^2+\left(\kappa_n^z\right)^2+\left(c_n^y\right)^2\left(\kappa_n^z\right)^2+\left(c_n^z\right)^2\left(\kappa_n^y\right)^2-2c_n^y\kappa_n^yc_n^z\kappa_n^z$$
$$c=\kappa_n^y\frac{\partial}{\partial x}y_n(x)+\kappa_n^z\frac{\partial}{\partial x}z_n(x)+\sqrt{\left(\kappa_n^y\right)^2+\left(\kappa_n^z\right)^2}$$
$$\times\sqrt{1+\left(\frac{\partial}{\partial x}y_n(x)\right)^2+\left(\frac{\partial}{\partial x}z_n(x)\right)^2}$$
$$d=2\left(\kappa_n^y\right)^2+2\left(\kappa_n^z\right)^2+2\left(c_n^y\right)^2\left(\kappa_n^z\right)^2+2\left(c_n^z\right)^2\left(\kappa_n^y\right)^2$$
$$-4c_n^y\kappa_n^yc_n^z\kappa_n^z. \quad (57)$$

The total initial arc length is then given as

$$S=S_1|_{x_0}^{x_\alpha}+S_2|_{x_\alpha}^{x_\beta}+S_3|_{x_\beta}^{x_\delta}+S_4|_{x_\delta}^{x_T}. \quad (58)$$

In real-time applications, (54) can be continuously updated with the current downrange value. It will be shown in Section VIII that under constant velocity, (54) produces a linear response over the engagement and is therefore not subject to error due to trajectory curvature. In this case, the time to go initially computed at the beginning of the engagement is, in fact, the impact time.

A. Nonconstant Velocity

Unlike standard methods, arc length is used in the calculation of time to go instead of line of sight between the missile and target. However, similar to existing methods, (54) is updated continuously and the velocity is considered constant over the small sampling-time interval. If the velocity changes over time, the time-to-go response over the engagement is no longer perfectly linear and the initially calculated time to go will not be the final impact time. Exact changes in the velocity can never be known, but one way to deal with this source of error is to conduct a preflight analysis.

Typically, as part of the mission-planning phase for a munition strike, preflight analysis is conducted to determine a feasible launch or firing window for the group of weapons in order to reach a target as close to simultaneously as possible. Current methods for preflight analysis integrate the standard guidance law into the future to find the point of closest approach. However, this is only a general approximation, and the actual flight trajectory may deviate considerably. Employing the use of prescribed QSPT trajectories with high-fidelity drag profiles and atmospheric models can produce accurate reference models predicting how the impact-time value changes over the engagement due to changes in velocity. Any disturbances occurring during the engagement can be corrected by the corresponding impact-time control.

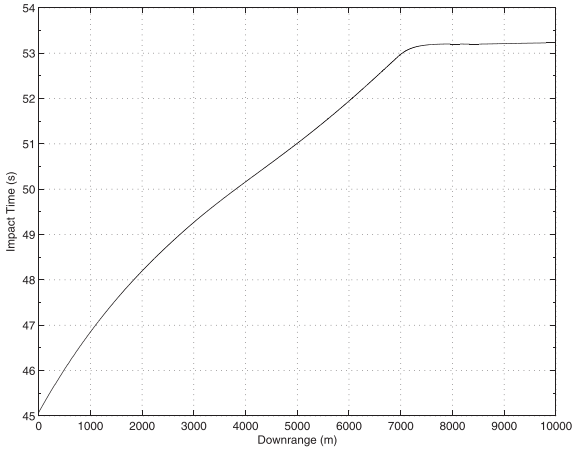

Fig. 3. Preflight reference model for impact time.

B. Preflight Reference Data

Through numerical simulation of (1)–(6), subject to the trajectory design steps in Section V, estimated reference data can be generated that describe how changes in the estimated velocity $\hat{V}$ alter the anticipated impact time due to drag and nonlinear atmospheric effects. The estimated time to go is given as

$$\hat{T}_{\text{go}}=\frac{S}{\hat{V}}, \quad (59)$$

which results in the estimated impact time of

$$\hat{T}_{\text{I}}=\hat{T}_{\text{go}}+t, \quad (60)$$

where t is the current preflight simulation time. Fig. 3, generated by (60), shows the resulting impact-time data over the simulated engagement, with values for impact time referenced with respect to the downrange axis. These data demonstrate that with drag and gravity, the velocity can change rapidly and alter the impact time considerably over the engagement. For this particular scenario, the velocity becomes more constant as the munition reaches a terminal velocity and the final estimated impact time of 53.3 s is reached. Additionally, wind and other disturbances can be modeled to estimate the effects of current atmospheric conditions.

The data shown in Fig. 3 were obtained using a simulation step size of 0.01 s. This produced a table of 5303 impact-time values. These estimated reference data are compared with real-time data during the actual

engagement and used to correct any disturbances which arise.

VII. IMPACT-TIME CONTROL

The impact-time control design presented in this section was chosen as a bang-bang control for simplicity. Certainly more advanced designs are possible. The real-time calculated impact time is given as

$$T_{\mathrm{I}} = T_{\mathrm{go}} + t, \tag{61}$$

where T_{go} is given by (54). Equation (61) is expected to contain some error with respect to the impact-time reference model $\hat{T}_{\mathrm{I}}$:

$$e = T_{\mathrm{I}} - \hat{T}_{\mathrm{I}}. \tag{62}$$

The bang-bang control for the prescribed linear system of (37) is then given as

$$\mu = \begin{cases} \delta & \text{if } e > 0 \\ 0 & \text{if } e = 0 \\ -\delta & \text{if } e < 0, \end{cases} \tag{63}$$

where δ is found empirically. The control of (63) is designed to command the state trajectory of κ_4^y to a location which drives the error in (62) to zero, or as close as possible. A decision logic is employed to ensure that the state trajectory of κ_4^y travels in a direction which minimizes (62). It should also monitor the rate of change of error versus the amount of applied control μ, since a reduction in error may not always be possible due to wind disturbance late in flight, when little trajectory planning can be applied to correct the error.

VIII. SIMULATION RESULTS

The following simulation results demonstrate the effectiveness of the optimization algorithm and robustness of the guidance law to measurement error. Improved performance of the time-to-go algorithm is also shown. Two impact-time control cases are also presented. Case 1 analyzes the ability of the proposed guidance law to reduce the impact-time error under a range of unknown disturbances. Case 2 is similar but employs a different engagement scenario.

A. Trajectory Optimization

Table I shows control energies with corresponding arc lengths computed for the cross-range trajectory. The initial conditions of $x_0 = 0$, $y_0 = 0$, and $\chi_0 = -\pi/4$, and the terminal conditions of $x_{\mathrm{T}} = 10\,000$, $y_{\mathrm{T}} = 10\,000$, and $\chi_{\mathrm{T}} = -\pi/4$ were chosen. The velocity was 400 m/s.

The numeric minimization of (53) for the given boundary conditions results in ${}^*\kappa_n^y = -4.5929 \times 10^{-4}$, the boldface value in the table. Multiple simulation runs were executed for κ_4^y on either side of the computed minimum with the corresponding control energy. Results show (53) to be an effective optimization with respect to control energy.

TABLE I
Cross-Range Control-Energy Minimization

κ_4^y	Total Control Energy (J)	Arc Length (m)
-8.0000×10^{-4}	8.5555×10^{-4}	1.7001×10^{-4}
-7.0000×10^{-4}	8.4480×10^{-4}	1.6437×10^{-4}
-5.0000×10^{-4}	8.4395×10^{-4}	1.6123×10^{-4}
$\mathbf{-4.5929 \times 10^{-4}}$	8.4295×10^{-4}	1.6173×10^{-4}
-3.0000×10^{-4}	8.4465×10^{-4}	1.6777×10^{-4}
-2.0000×10^{-4}	9.1285×10^{-4}	1.7571×10^{-4}
-1.0000×10^{-4}	10.8775×10^{-4}	1.8737×10^{-4}

Bold face value indicates computed minimum.

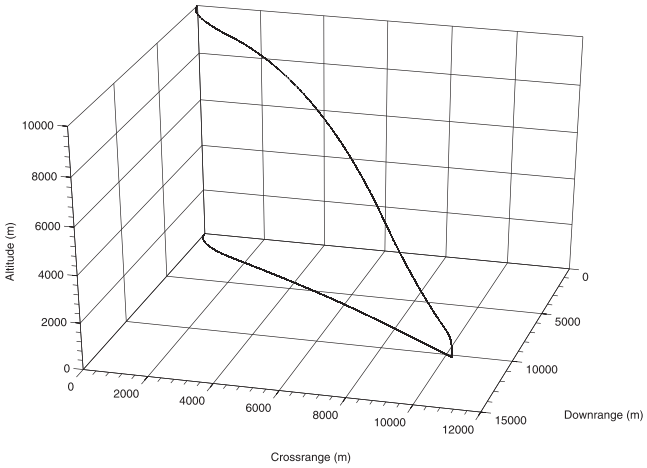

Fig. 4. Trajectory with curvature.

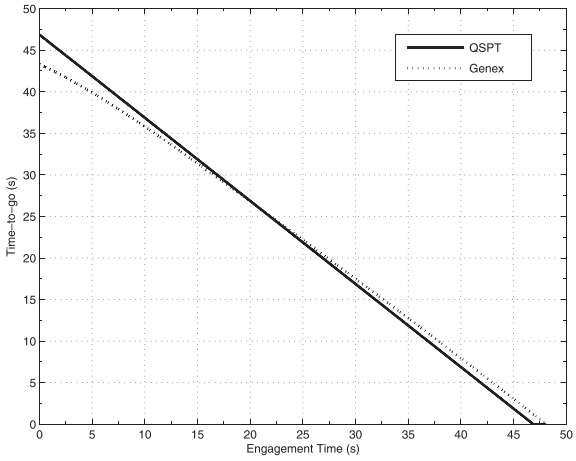

Fig. 5. Comparison of QSPT time-to-go algorithm versus GENEX using range-over-missile-velocity.

B. Performance of Time-to-Go Algorithm

A time-to-go comparison was generated for constant velocity along the trajectory of Fig. 4, and the results are given in Fig. 5. For clarity, the cross-range/downrange projection of the trajectory has been included in all 3-D plots. Fig. 5 demonstrates that the time-to-go response generated by (54) over the engagement is linear and is thus not effected by curvature of the trajectory. In other types of applications where the velocity along the trajectory could actually be constant, the initially calculated time to

TABLE II
Sensor and Gyro Error

Error Type	Unit	Measure	Value
Drag acceleration	m/s^2	%	3.0
Gyro bias	°/h	1σ	3.0
Gyro scale factor	ppm	1σ	300
Gyro random-walk noise	$^\circ/\sqrt{h}$	Nom	0.02

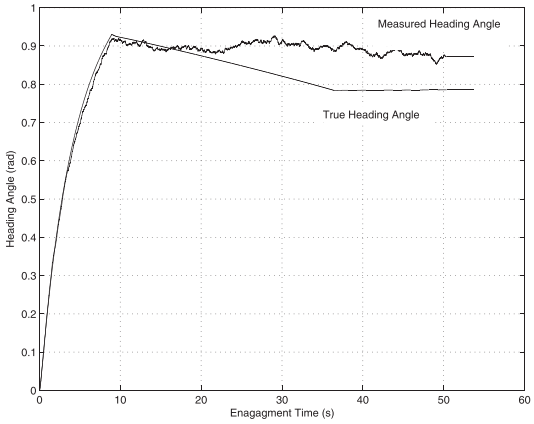

Fig. 6. True heading angle versus measured.

go is in fact the final impact time, as seen in Fig. 5. The performance of the switched-form guidance law and reparameterized trajectory given in appendix A can also be observed in Fig. 4. The switched-form guidance law takes control at the downrange, cross-range, and altitude positions of 10 000, 10 000, and 1000 m, respectively.

In contrast, Fig. 5 also shows the time to go generated by the GENEX guidance law of [2], which uses the range-over-missile-velocity method. The velocity was constant and the initial and terminal angles were the same ones used to generate Fig. 4. The effects of trajectory curvature on the standard time-to-go calculation can clearly be seen. The initial calculation of time to go is 43.3 s, but the actual final impact time is 48 s—nearly a 5 s disparity, even with constant velocity and a stationary target.

C. Robustness of the Guidance Law

The remaining simulation results in this section were generated using a 1962 Standard Atmosphere along with tabular drag profiles for a generic projectile. It is assumed that the positional measurements provided by the inertial measurement unit are perfect. However, since the inversion matrix of the guidance law is explicit in angles γ and χ, gyro errors consistent with a tactical-grade inertial measurement unit are considered. In addition, a percent error in sensed drag acceleration a_d is considered as well. Table II lists the error values used to obtain the following results.

Gyro bias, scale factor, and noise are considered in addition to 3% error in the sensed drag acceleration. Figs. 6 and 7 contrast the measured values against the true values for χ and γ, respectively, produced along the trajectory of Fig. 4.

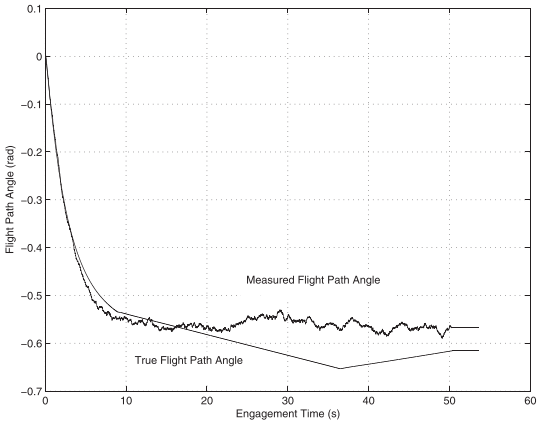

Fig. 7. True flight-path angle versus measured.

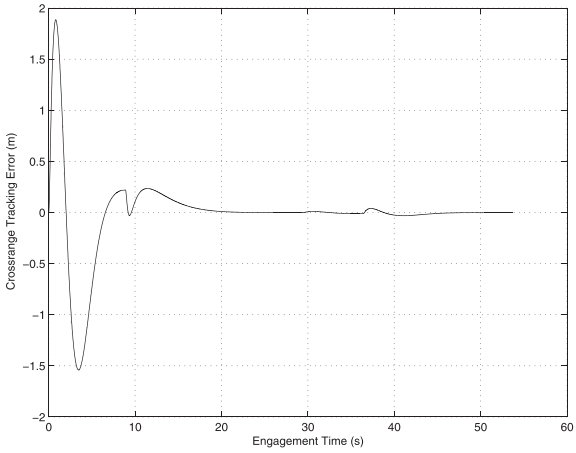

Fig. 8. Cross-range tracking performance under no disturbance.

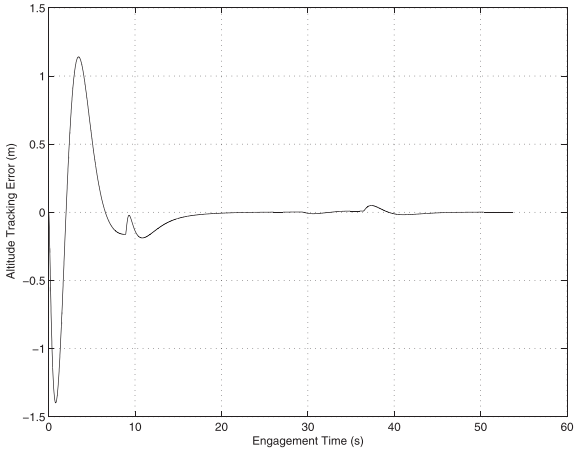

Fig. 9. Altitude tracking performance under no disturbance.

Figs. 8 and 9 show the tracking-error performance of the guidance law using perfect measurements of χ and γ. In contrast, Figs. 10 and 11 show the cross-range and altitude-tracking performance of (25) in response to the measurement errors listed in Table II. The proposed

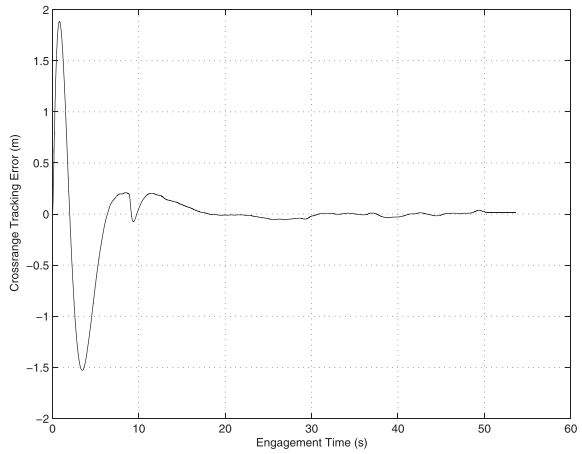

Fig. 10. Cross-range tracking performance subject to the disturbances of Table I.

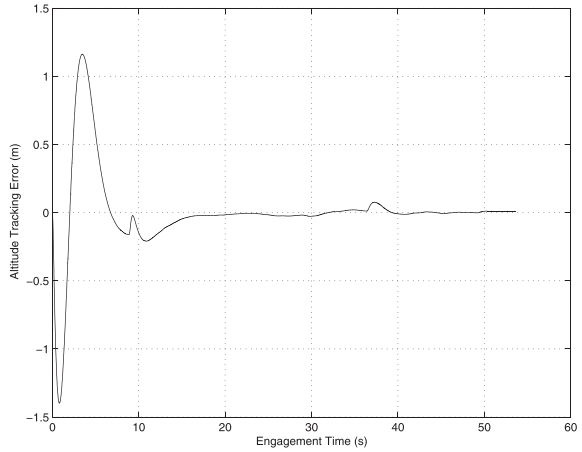

Fig. 11. Altitude tracking performance subject to the disturbances of Table I.

guidance law demonstrates good tracking performance with negligible deterioration.

D. Impact-Time Control: Case 1

For the final portion of this study, an unknown wind disturbance of 15 m/s in the positive downrange direction is considered and, in addition to the measurement errors of Table II, an unknown lumped nonlinear disturbance is employed. The wind disturbance spans the full altitude of the engagement, from 10 000 m to the ground, and remains constant over that range. The combined effect of the wind and disturbances without any control over impact time produces a final impact time of 48.825 s. The preflight analysis which considered drag and atmosphere but excluded disturbances determined a desired impact time of 53.7 s. This requires the corresponding impact-time control to correct for an error of 4.875 s over the engagement. The desire is to reduce the error in impact time to fractions of a second under these given conditions. Table III provides the initial and terminal conditions for the engagement.

Fig. 12 contrasts the difference between the impact-time reference model and the resulting in-flight

TABLE III
Initial and Terminal Conditions, Case 1

Variable	Unit	Initial	Terminal
Velocity	m/s	400	
Downrange	m	0	10 100.0
Cross-range	m	0	10 100.0
Altitude	m	10 000.0	0
χ	rad	0	$\pi/4$
γ	rad	0	$-\pi/2$
Required impact time	s		53.70

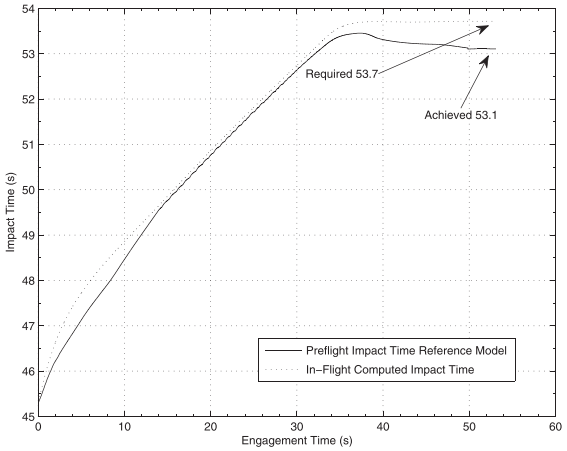

Fig. 12. Comparison between desired and achieved impact-time profiles.

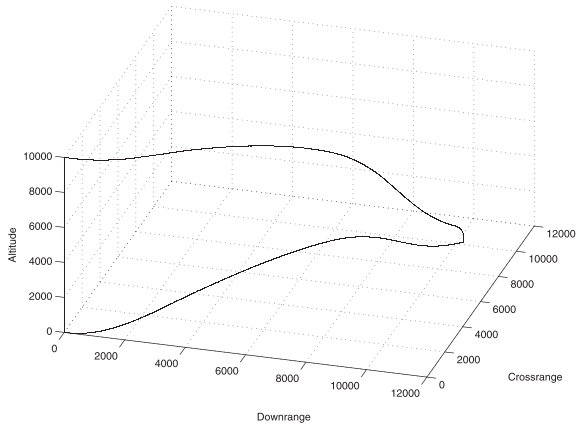

Fig. 13. Resulting trajectory under impact-time control.

impact time. The resulting trajectory is given in Fig. 13. Under the bang-bang control signals of Fig. 14, the corresponding impact-time error was reduced to 0.577 s. Fig. 15 compares the effect of the control on the impact-time error. In that figure, the uncontrolled impact-time response is shown against the controlled impact time. Fig. 16 shows the state trajectory for K_4^y generated by the control. The proposed guidance law demonstrates a good ability to reduce the error of (62) under heavy disturbances.

In the previous simulation scenario, the constant wind disturbance extended from apogee to the ground. It is difficult to exactly achieve a prescribed impact time under

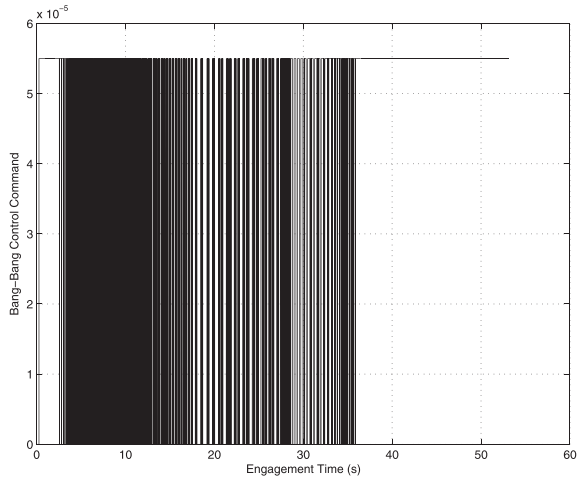

Fig. 14. Bang-bang control commands.

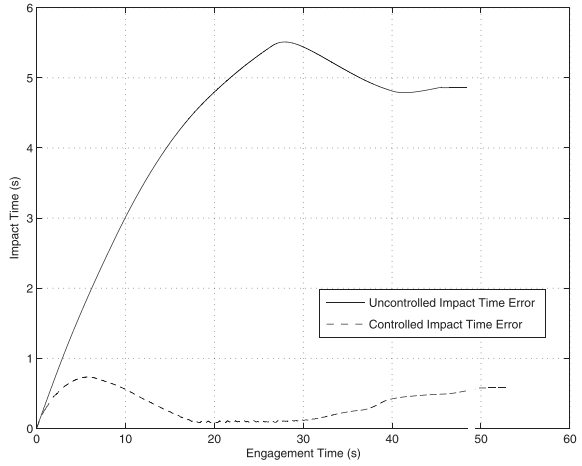

Fig. 15. Error comparison between controlled and uncontrolled impact time.

these conditions, since the correction of impact-time error relies on trajectory planning. Therefore, the ability to correct for disturbances late in flight diminishes. In addition, the trajectory must satisfy impact-angle requirements that take priority over satisfying impact time. In the next simulation run, we consider the wind to taper off with lower altitude and reduce to zero near the ground, in order to show an improvement in impact-time control.

Fig. 17 shows a considerable improvement in the impact-time error when the wind diminishes in the lower altitudes. In this scenario, the impact-time control is capable of reducing the error to 0.05 s, a vast improvement if the disturbances are minimal in the final few seconds of the engagement. Fig. 18 shows perfect satisfaction of the desired impact time with negligible or no wind.

Few works exist in the open literature to conduct a fair performance comparison against. However, [8] considers a time-of-flight control problem for a guided projectile with several error sources. While the overall scope of that work also considers range maximization, some brief comparisons can be made. In [8], a prescribed impact time is achieved as long as perfect knowledge of both muzzle

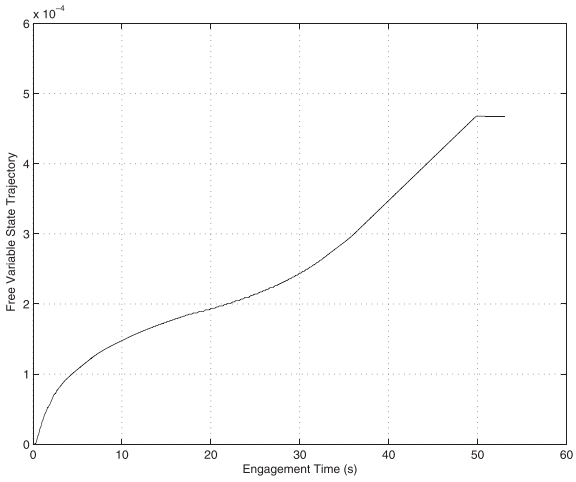

Fig. 16. State trajectory of κ_4^y.

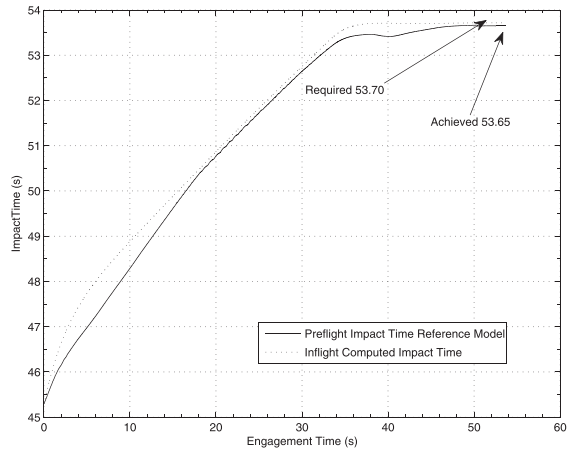

Fig. 17. Impact time with wind tapering off.

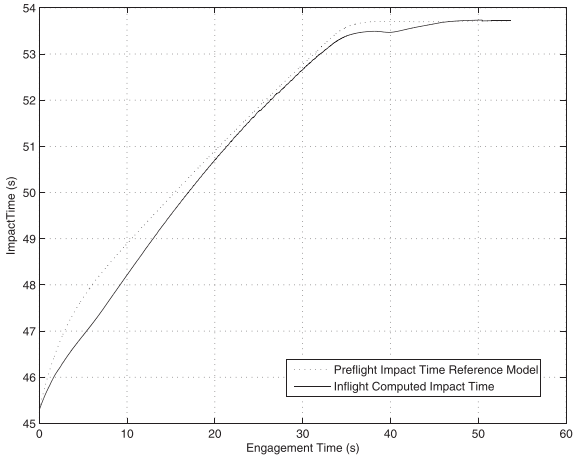

Fig. 18. Impact time with no wind.

exit velocity and wind are obtained. The performance of the algorithm is expected to deteriorate under measurement error, and the requirement is then to bring the impact-time dispersal between weapons to within approximately 2.0 s. As demonstrated in the previous three impact-time control scenarios, the proposed QSPT

TABLE IV
Initial and Terminal Conditions, Case 2

Variable	Unit	Initial	Terminal
Velocity	m/s	400	
Downrange	m	0	10 000.0
Cross-range	m	0	10 000.0
Altitude	m	10 000.0	0
χ	rad	$\pi/4$	$-\pi/18$
γ	rad	0	$-\pi/4$
Required impact time	s		56.16

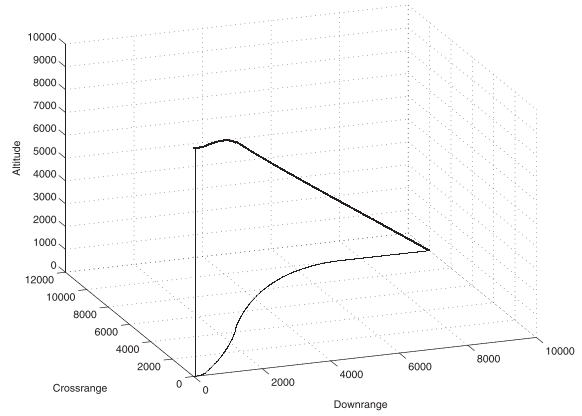

Fig. 19. Minimum-curvature trajectory with no impact-time control.

guidance algorithm can reduce the impact-time error to well below 1 s under a series of unknown disturbances.

E. Impact-Time Control: Case 2

In Case 2, all of the previous measurement errors are considered and an unknown wind disturbance of 7 m/s in the positive downrange direction is present. Table IV details the engagement parameters and Fig. 19 shows the optimized trajectory set by ${}^*K_4^y$ and ${}^*K_4^z$ with no impact time adjustment made by the control.

From the preflight-analysis stage, a desired impact time was determined to be 56.16 s. The unknown wind disturbance causes the projectile to reach the target earlier, at 54.17 s. The corresponding impact-time control reduced the impact-time error to 0.298 s, and the resulting trajectory is given in Fig. 20. The impact time error is given in Fig. 21.

IX. CONCLUSION

Results demonstrate that the proposed guidance law is robust with respect to measurement noise and error. The arc-length-based time-to-go calculation has also been shown to be free of error resulting from trajectory curvature, thus improving the in-flight calculation of impact time. Impact-time control Cases 1 and 2 show that when subjected to unknown wind disturbances, the corresponding impact-time control is capable of reducing the impact-time error to much less than 1 s. It can also be seen from these results that for disturbances which occur late in flight, when trajectory planning is not possible, the

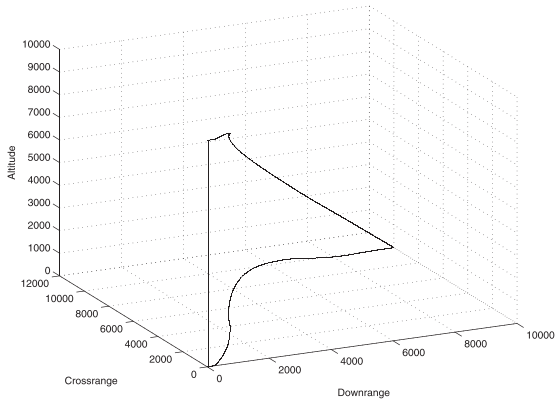

Fig. 20. Trajectory resulting from controlled impact time.

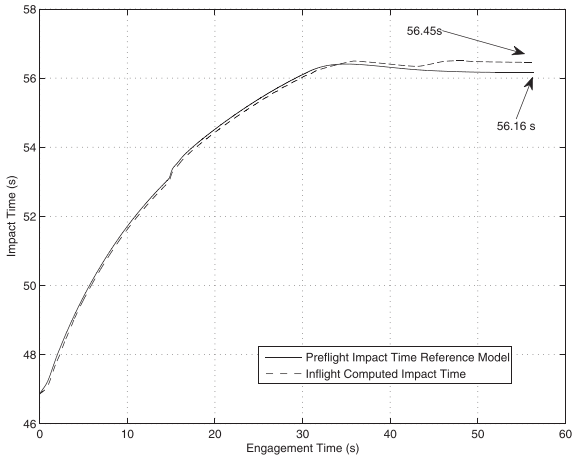

Fig. 21. Comparison between preflight reference model and in-flight computed impact time.

impact-time error can grow slightly, as seen in both Cases 1 and 2. When very little disturbance is encountered in the final seconds of flight, the impact-time error is reduced to zero, as shown in Case 1.

APPENDIX A. REPARAMETERIZATION OF GUIDANCE LAW

The derivation for the reparameterized trajectories and the switched guidance law follows in an identical manner from Section II.D. Reparameterize (8) in terms of z as

$$\begin{aligned} y_n(z) &= a_n^y + c_n^y z + \kappa_n^y z^2 \\ x_n(z) &= a_n^x + c_n^x z + \kappa_n^x z^2. \end{aligned} \tag{64}$$

Parameterization of the trajectories is chosen to satisfy the dynamic equations

$$\frac{\partial y}{\partial z} = \cot(\gamma)\sin(\chi), \quad \frac{\partial x}{\partial z} = \cot(\gamma)\cos(\chi). \tag{65}$$

Then the error is given as

$$e_y = y - y_n, \quad e_x = x - x_n, \tag{66}$$

where

$$\begin{bmatrix} \ddot{e}_y \\ \ddot{e}_x \end{bmatrix} = A + B \begin{bmatrix} u_\chi \\ u_\gamma \end{bmatrix} \tag{67}$$

and

$$A = \begin{bmatrix} -a_{\mathrm{d}}\cos(\gamma)\sin(\chi) - 2g\sin(\gamma)\cos(\gamma)\sin(\chi) - 2\kappa_n^y(\dot{z})^2 \\ \quad + \frac{\partial}{\partial z}y_n\left(a_{\mathrm{d}}\sin(\gamma) + g\left(\sin^2(\gamma) - \cos^2(\gamma)\right)\right) \\ -a_{\mathrm{d}}\cos(\gamma)\cos(\chi) - 2g\sin(\gamma)\cos(\gamma)\cos(\chi) - 2\kappa_n^x(\dot{z})^2 \\ \quad + \frac{\partial}{\partial z}x_n\left(a_{\mathrm{d}}\sin(\gamma) + g\left(\sin^2(\gamma) - \cos^2(\gamma)\right)\right) \end{bmatrix} \tag{68}$$

$$B = \begin{bmatrix} \cos(\chi) & -\sin(\gamma)\sin(\chi) - \frac{\partial}{\partial z}y_n\cos(\gamma) \\ -\sin(\chi) & -\sin(\gamma)\cos(\chi) - \frac{\partial}{\partial z}x_n\cos(\gamma) \end{bmatrix}. \tag{69}$$

The closed-loop guidance law for the reparameterized system of (63) is thus given as

$$\begin{bmatrix} u_\chi \\ u_\gamma \end{bmatrix} = B^{-1}\left[\begin{bmatrix} k_y\left(y - y_n\right) + k'_y\left(\dot{y} - \dot{y}_n\right) \\ k_x\left(x - x_n\right) + k'_x\left(\dot{x} - \dot{x}_n\right) \end{bmatrix} - A\right]. \tag{70}$$

It should be noted that the A matrix in this switched-form guidance law contains no coefficient time derivatives, as compared with the regular form of A in (23). This is because of the fact that during the terminal phase of flight, the munition is approaching the target from above, and trajectory planning is not possible to correct any error in impact time. Therefore, the QSPT coefficients remain constant.

APPENDIX B. INTEGRAL CHANGE OF VARIABLE

Velocity is expressed as

$$V = \sqrt{\dot{x}^2 + \dot{y}^2}. \tag{71}$$

Factoring out $\dot{x}$ gives

$$V = \dot{x}\sqrt{1 + \left(\frac{\partial y}{\partial x}\right)^2}. \tag{72}$$

Writing in terms of QSPT and separating the differential gives

$$V\,dt = \sqrt{1 + \left(c_n^y + 2\kappa_n^y x\right)^2}\,dx, \tag{73}$$

which gives the final transformation

$$dt = \frac{\sqrt{1 + \left(c_n^y + 2\kappa_n^y x\right)^2}}{V}dx. \tag{74}$$

REFERENCES

[1] Ryoo, C.-K., Cho, H., and Tahk, M.-J.
Closed-form solutions of optimal guidance with terminal impact angle constraint.
In *Proceedings of 2003 IEEE Conference on Control Applications*, Istanbul, Turkey, June 23–25, 2003, **1**, 504–509.

[2] Ohlmeyer, E. J.
Control of terminal engagement geometry using generalized vector explicit guidance.
In *Proceedings of the 2003 American Control Conference*, Denver, CO, June 2003.

[3] Ryoo, C.-K., Cho, H., and Tahk, M.-J.
Optimal guidance laws with terminal impact angle constraint.
Journal of Guidance, Control, and Dynamics, **28**, 4 (2005), 724–732.

[4] Lee, J.-I., Jeon, I.-S., and Tahk, M.-J.
Guidance law to control impact time and angle.
IEEE Transactions on Aerospace and Electronic Systems, **43**, 1 (Jan. 2007), 310–310.

[5] Song, T. L., and Shin, S. J.
Time-optimal impact angle control for vertical plane engagements.
IEEE Transactions on Aerospace and Electronic Systems, **35**, 2 (Apr. 1999), 738–742.

[6] Kim, M., Jung, B. Han, B., Lee, S., and Kim, Y.
Lyapunov-based impact time control guidance laws against stationary targets.
IEEE Transactions on Aerospace and Electronic Systems, **51**, 2 (Apr. 2015), 1111–1122.

[7] Jeon, I.-S., Lee, J.-I., and Tahk, M.-J.
Impact-time-control guidance law for anti-ship missiles.
IEEE Transactions on Control Systems Technology, **14**, 2 (Mar. 2006), 260–266.

[8] Phillips, C. A.
Guidance algorithm for range maximization and time-of-flight control of a guided projectile.
Journal of Guidance, Control, and Dynamics, **31**, 5 (2008), 1447–1455. doi:10.2514/1.31327

[9] Ryoo, C.-K., Tahk, M.-J., and Cho, H.
Practical time-to-go estimation methods for optimal guidance.
In *Guidance, Navigation, and Control Conference and Exhibit*, Portland, OR, Aug. 1999, AIAA-99-4143.

[10] Lee, C.-H., Kim, T.-H., and Tahk, M.-J.
Effects of time-to-go errors on performance of optimal guidance laws.
IEEE Transactions on Aerospace and Electronic Systems, **51**, 4 (Oct. 2015), 3270–3281.

[11] Lee, C.-H., Kim, T.-H., Tahk, M.-J., and Whang, I.-H.
Polynomial guidance laws considering terminal impact angle and acceleration constraints.
IEEE Transactions on Aerospace and Electronic Systems, **49**, 1 (Jan. 2013), 74–92.

[12] Weitz, L. A., and Hurtado, J. E.
A time-to-go control law for spacing vehicles at a point.
In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Portland, OR, Aug. 2011.

[13] Lam, V. C.
Time-to-go estimate for missile guidance.
In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, San Francisco, CA, Aug. 2005.

[14] Kim, T.-H., Lee, C.-H., and Tahk, M.-J.
Time-to-go polynomial guidance laws with terminal impact angle/acceleration constraints.
In *Proceedings of the 18th World Congress, The International Federation of Automatic Control*, Milano, Italy, Aug.–Sept. 2011.

[15] Hull, D. G., and Radke, J. J.
Time-to-go prediction for a homing missile based on minimum-time trajectories.
In *Guidance, Navigation and Control Conference*, Minneapolis, MN, Aug. 1988, 88-4064-CP.

[16] Cho, H., Ryoo, C. K., and Tahk, M.-J.
Closed-form optimal guidance law for missiles of time-varying velocity.
Journal of Guidance, Control, and Dynamics, **19**, 5 (1996), 1017–1022.

[17] Cho, H., Ryoo, C. K., and Tahk, M.-J.
Implementation of optimal guidance laws using predicted missile velocity profiles.

Journal of Guidance, Control, and Dynamics, **22**, 4 (1999), 579–588.

[18] Baba, Y., Takehira, T., and Takano, H.
New guidance law for a missile with varying velocity.
In *Guidance, Navigation, and Control Conference*, Scottsdale, AZ, Aug. 1994, AIAA-94-3565-CP.

[19] Lee, J.-I., Jeon, I.-S., and Tahk, M.-J.
Guidance law using augmented trajectory-reshaping command for salvo attack of multiple missiles.
In *International Control Conference*, Glasgow, United Kingdom, Aug.–Sept. 2006.

[20] Arita, S., and Ueno, S.
Improvement of guidance law for simultaneous attack.
In *Proceedings of SICE Annual Conference*, Tokyo, Japan, Sept. 2011.

[21] Wu, J., Ma, P., and Ji, J.
Research on cooperative control method of saturation attack.
In *2007 IEEE International Conference on Automation and Logistics*, Jinan, China, Aug. 2007.

[22] Lin, C.-L., Lin, Y.-P., and Chen, K.-M.
On the design of fuzzified trajectory shaping guidance law.
ISA Transactions, **48**, 2 (Apr. 2009), 148–155.

[23] Snyder, M. G.
A new impact time control guidance law for precise time-on-target missile strike.
In *Aerospace Sciences Conference*, Kissimmee, FL, Jan. 2015.

Mark G. Snyder received his B.S. and M.S. degrees from the University of Central Florida in 2006 and 2009, respectively. He is currently a Ph.D. degree candidate at the University of Central Florida as well as an adjunct faculty member in the Department of Computer and Electrical Engineering at Embry-Riddle Aeronautical University in Daytona Beach, Florida.

Zhihua Qu (M'90—SM'93—F'09) is the SAIC Endowed Professor in the College of Engineering and Computer Science and a professor and the chair of Electrical and Computer Engineering at the University of Central Florida, as well as the director of the FEEDER Center (one of the Department of Energy–funded centers on distributed technologies and a smart grid). His recent research focuses on networked systems and cooperative control, distributed optimization, and their applications to energy/power systems and autonomous vehicles. He is a Fellow of the IEEE and AAAS. He served on the board of governors of the IEEE Control Systems Society and as an associate editor for *IEEE Transactions on Automatic Control* and the *International Journal of Robotics and Automation* (since 1995). Currently, he serves as an associate editor for *Automatica* and *IEEE Access*.

Richard A. Hull received his B.S. degree in engineering science from the University of Florida and his M.S. and Ph.D. degrees in electrical engineering from the University of Central Florida. He has served as a guidance and control system engineer in the aerospace industry for over 40 years, working for United Technologies, Goodrich, SAIC, Lockheed Martin, L3 Aerospace, McDonnell Douglas, and Boeing. He is currently a technical fellow with UTC Aerospace Systems.

Richard J. Prazenica is an assistant professor of aerospace engineering at Embry-Riddle Aeronautical University in Daytona Beach, Florida. He received his B.S. degree in mechanical engineering from the University of Pennsylvania, and his M.S. and Ph.D. degrees in aerospace engineering from the University of Florida.