Optimal Trajectory Regulation for Radar Imaging Guidance

Asif Farooq[*]

MBDA UK, Ltd., Stevenage, England SG1 2DA, United Kingdom

and

David J. N. Limebeer[†]

Imperial College, London, England SW7 2BT, United Kingdom

This paper investigates optimal trajectory tracking for air-to-surface missiles using bank-to-turn steering and equipped with Doppler-beam-sharpening seekers. The seeker generates a radar image of the terrain that is continually updated in flight. The guidance problem for this type of sensor is complex, because constraints must be imposed on the flight path of the missile to ensure that the seeker can obtain adequate crossrange resolution for target recognition and aim-point selection. Terminal position, incidence, and angle constraints are also imposed in the guidance problem. An offline optimal trajectory can be obtained through trajectory optimization techniques that have been presented in previous papers. The focus of this work is on following the precomputed optimal trajectories. The method presented uses a linear quadratic regulator to calculate closed-loop in-flight controls. We also investigate the effectiveness of a preview term based on the prediction of the future trajectory from the current state. Preview control is an optimal tracking method that uses future trajectory samples within a preview window to improve the tracking accuracy. A set of preview gains is computed and is used in addition to the regulator gains. Over the last 2 km of flight, an impact-angle guidance law that minimizes the miss distance and impact-angle error is used to achieve the terminal constraints. The guidance scheme is evaluated for tracking offline Doppler-beam-sharpening trajectories in the presence of modeling errors, sensor noise, and disturbances.

Nomenclature

A, B, C, D	=	discrete-time state-space matrices
A_c, B_c, C_c, D_c	=	continuous-time state-space matrices
A'_{my}, A'_{mz}	=	yaw and pitch accelerations measured at the accelerometer, $\mathrm{m\,s^{-2}}$
A_p	=	preview shift-register matrix
A_{yd}, A_{zd}	=	bank-to-turn acceleration demands, $\mathrm{m\,s^{-2}}$
A'_{yd}, A'_{zd}	=	skid-to-turn acceleration demands, $\mathrm{m\,s^{-2}}$
a_m	=	true acceleration vector in body axes, $\mathrm{m\,s^{-2}}$
a_s	=	true acceleration vector in space axes, $\mathrm{m\,s^{-2}}$
C_l, C_m, C_n	=	aerodynamic moment coefficients
$C_{lp}, C_{l\beta},$ $C_{l\delta}, C_{l\alpha\beta1}$	=	roll stability derivatives, $\mathrm{rad^{-1}}$
$C_{m0}, C_{mq},$ $C_{m\alpha}, C_{m\delta}$	=	pitch stability derivatives, $\mathrm{rad^{-1}}$
$C_{nr}, C_{n\beta}, C_{n\delta}$	=	yaw stability derivatives, $\mathrm{rad^{-1}}$
C_x, C_y, C_z	=	aerodynamic force coefficients
$C_{x0}, C_{x\alpha}, C_{x\delta p},$ $C_{x\delta q}, C_{x\delta r}$	=	body axial stability derivatives, $\mathrm{rad^{-1}}$
$C_{y\beta}, C_{y\delta}$	=	body side-force stability derivatives, $\mathrm{rad^{-1}}$
$C_{z0}, C_{z\alpha}, C_{z\delta}$	=	body lift-force stability derivatives, $\mathrm{rad^{-1}}$
c_1	=	decay rate constant for terminal incidence control, $\mathrm{s^{-1}}$
d	=	distance from the accelerometer to the center of gravity in the body x axis, m
E_{pk}	=	predicted future tracking-error matrix
e_k	=	current-time tracking error
e_{pk}	=	predicted future tracking-error vector, $\mathrm{vec}(E_{\mathrm{pk}})$
$f(.)$	=	nonlinear prediction model
g	=	acceleration due to gravity, $\mathrm{m\,s^{-2}}$
$g_{\mathrm{compy}}, g_{\mathrm{compz}}$	=	yaw and pitch gravity-compensation components, $\mathrm{m\,s^{-2}}$
g_s	=	gravity vector, $\mathrm{m\,s^{-2}}$
I_r	=	identity matrix of dimension r
I_{xx}, I_{yy}, I_{zz}	=	roll, pitch, and yaw moments of inertia, $\mathrm{kg\,m^2}$
I_{xz}	=	product of inertia, $\mathrm{kg\,m^2}$
J	=	performance index
j	=	counter for terminal incidence logic
K_a, K_p	=	linear quadratic regulator and preview gains
$K_{1p}, K_{2p},$ K_{3p}, K_{4p}	=	pitch autopilot gains
$K_{1y}, K_{2y},$ K_{3y}, K_{4y}	=	yaw autopilot gains
$K_{1\phi}, K_{1r}$	=	roll autopilot gains
k	=	sample index
k_d	=	parabolic drag coefficient
k_{p1}, k_{p2}	=	proportional and integral gains used in terminal guidance law
l_r	=	reference length, m
m	=	missile mass, kg
N	=	finite horizon index
N_p	=	preview length
p, q, r	=	missile body rates, $\mathrm{rad\,s^{-1}}$
$\mathrm{proj}(.)$	=	projection operator
Q, R	=	linear quadratic regulator weighting matrices
Q_d	=	dynamic pressure, $\mathrm{kg\,m^{-1}\,s^{-2}}$
q_0, q_1, q_2, q_3	=	components of quaternion vector
R	=	range-to-go, m
$\mathbb{R}$	=	field of real numbers
R_{sx}, R_{sy}, R_{sz}	=	true position states (downrange, crossrange, and height), m
R_t	=	range at which to start terminal incidence logic, m
S	=	Riccati equation solution for inflated (plant and preview) system

S_r	=	reference area, m^2
S_{11}, S_{12}	=	Riccati equation solutions for plant and preview parts of the system
SFC	=	specific fuel consumption, kg/Ns
T	=	applied thrust, N
T_{BS}	=	body-to-space transformation matrix
T_{dem}	=	demanded thrust, N
U, V, W	=	linear inertial velocities in body axes, m s^{-1}
U^*_{pk}	=	optimal control matrix
U', V', W'	=	linear velocities relative to air mass in body axes, m s^{-1}
$\boldsymbol{u}_c$	=	acceleration-demand vector calculated by guidance law (velocity axes), m s^{-2}
$\boldsymbol{u}_{d1}, \boldsymbol{u}_{d2}$	=	unit vectors in velocity axes
u_{gw}, v_{gw}, w_{gw}	=	gust velocities in Earth axes, m s^{-1}
$\boldsymbol{u}_p, \boldsymbol{u}_f$	=	control components in the guidance law, m s^{-2}
$\boldsymbol{u}_k^{\text{prev}}$	=	preview-control signal
u_{ss}, v_{ss}, w_{ss}	=	steady-state wind velocities in Earth axes, m s^{-1}
u_1, u_2, u_3	=	control signals (pitch, roll, and thrust demands)
$\hat{u}_1$	=	pitch acceleration demand at critical range R_t, m s^{-2}
V_t	=	total missile speed, m s^{-1}
$\boldsymbol{v}_a$	=	accelerometer noise vector, m s^{-2}
$\boldsymbol{v}_d$	=	reflection of desired velocity about the relative position vector
$\boldsymbol{v}_f$	=	desired final velocity vector
$\boldsymbol{v}_g$	=	gyro noise vector, rad s^{-1}
$\boldsymbol{v}_m$	=	missile velocity vector (space axes), m s^{-1}
$\boldsymbol{v}_o$	=	sightline rate (sightline axes), rad s^{-1}
$\bar{\boldsymbol{v}}_s$	=	estimated velocity vector (space axes), m s^{-1}
vec(.)	=	converts a matrix to a vector
$\bar{X}_{\text{pk}}$	=	future predicted state matrix
X^*_{pk}	=	future optimal state matrix
$x_{\text{ap}}, y_{\text{ap}}$	=	internal autopilot states, rad
$\bar{x}_k, x_k$	=	measured and true state vectors
x_r	=	relative position vector, m
x^*, u^*	=	optimal state and control
x_k^*, u_k^*	=	resampled optimal state and control
$\bar{Y}_{\text{pk}}$	=	predicted output-trajectory matrix
Y^*_{pk}	=	optimal output-trajectory matrix
α, β	=	angles of incidence and sideslip, rad
$\alpha_{\text{imp}}, \beta_{\text{imp}}$	=	desired elevation and azimuth impact angles, rad
γ	=	flight path angle, rad
Δ_i	=	perturbation for uncertainty in stability derivatives
$\delta_p, \delta_q, \delta_r$	=	fin-angle demands, rad
$\delta_p', \delta_q', \delta_r'$	=	fin angles achieved, rad
δt	=	control sample rate, s
$\delta \boldsymbol{B}_a, \delta \boldsymbol{B}_g$	=	bias vector for accelerometers and gyros
$\delta \boldsymbol{S}_a, \delta \boldsymbol{S}_g$	=	scale-factor-error matrix for accelerometers and gyros
$\delta \boldsymbol{u}_k, \delta \boldsymbol{x}_k, \delta \boldsymbol{y}_k$	=	control deviation, state deviation, and output deviation
ϵ	=	angle between the sightline vector and the desired velocity vector in elevation, rad
η	=	angle between the velocity vector and relative position vector, rad
θ, ψ, ϕ	=	pitch, yaw, and roll angles, rad
θ_d	=	angle between the missile velocity vector and the desired final velocity vector, rad
λ	=	angle between the sightline vector and the missile velocity vector in elevation, rad
ρ	=	air density, kg m^{-3}
σ	=	elevation sightline angle, rad
$\boldsymbol{\Phi}_c$	=	discrete-time closed-loop transition matrix
ϕ_d	=	roll-angle demand, deg
$\bar{\boldsymbol{\omega}}_b, \boldsymbol{\omega}_b$	=	measured and true body-rate vectors, rad s^{-1}
$(.)^T$	=	matrix transpose
$\| \cdot \|$	=	vector 2-norm
$\bar{(.)}$	=	measured or estimated quantity

I. Introduction

THIS paper investigates optimal trajectory tracking for missiles with radar imaging seekers. The radar imaging seeker requires the ground-plane trajectory to be shaped to ensure that adequate crossrange resolution is obtained during the terminal guidance phase. This flight path requires an angular offset between the velocity vector and the sightline in azimuth. In practice, the angular offset is constrained by the seeker's gimbal-angle limits, as well as the need to home in on the target during the terminal guidance phase. In addition, the missile must fly at low altitudes to evade detection by air-defense units. For our applications, an elevation-impact-angle constraint and terminal incidence constraints are also imposed for enhanced lethality.

An offline optimal trajectory can be designed using trajectory optimization methods (e.g., see [1], which presents results for bank-to-turn airframes). The current study augments the previous work and studies closed-loop guidance methods that follow the precomputed optimal trajectory. The model is linearized around the optimal state and control pair $\boldsymbol{x}^*$ and $\boldsymbol{u}^*$. A digital controller design based on discrete linear quadratic regulator (LQR) theory with preview is presented. The optimal trajectories of [1] are used as test cases for the feedback controller, which is used up to a range-to-go of 2 km. The terminal phase of the trajectory is controlled by an impact-angle control law that is used to achieve the requisite terminal constraints. The missile model is based on the model presented in [1], though gusts, instrumentation errors, steady-state wind, and modeling errors have now been included. Closed-loop guidance performance is evaluated using Monte Carlo methods.

The paper is structured as follows. A brief summary of previous research is contained in Sec. II. The vehicle dynamics model for a bank-to-turn missile is reviewed in Sec. III. Because linear quadratic methods require knowledge of the full state, state estimation is required; this is considered in Sec. IV. Discrete linear quadratic regulator theory with preview is briefly reviewed in Sec. V. Performance in Monte Carlo simulations is then presented for two test cases in Sec. VI. Conclusions and areas for further work are then discussed in Sec. VII.

II. Previous Research

Trajectory optimization with Doppler beam-sharpening (DBS) constraints has been considered in [2] for skid-to-turn airframes and in [1] for bank-to-turn airframes. These papers combine the DBS requirements with a bunt trajectory to obtain steep impact angles. A trajectory optimization method based on direct shooting is used. The controls are parameterized using piecewise constant functions and are combined with a variable step integrator to obtain numerical solutions. In [3], a skid-to-turn terminal guidance problem is solved using a similar trajectory optimization method. It is possible to adapt guidance laws to the path following problem [4], and one possible approach has been presented in [3]. The approach in this paper adopts an alternative method involving regulation around a precomputed optimal trajectory.

One of the early methods proposed for closed-loop optimal guidance is neighboring optimal control [5]. In this method, the original cost function is expanded to second order and the system is linearized around the optimal trajectory. The optimal trajectory must satisfy a number of technical conditions [5]. The method requires second derivatives of the Hamiltonian evaluated on the optimal trajectory to calculate the feedback gains. The resulting guidance law is composed of an open-loop feedforward control and a linear time-varying state feedback. A neighboring optimal controller for aircraft guidance in winds is presented in [6].

A linear state-feedback law can also be obtained using a linear quadratic regulation technique [7]. A linearization approach is again applied, but a new quadratic cost function is minimized (based on a

quadratic function of the state deviation and control deviations from the optimal trajectory). The method can be applied to any feasible trajectory, but not necessarily an optimal one [7]. In many applications, a suboptimal gain-scheduling approach is adopted using the steady-state LQR gains [7]. The LQR approach has proved to be popular in reentry guidance [7,8]. A disadvantage of the gain-scheduled approach is that stability is not theoretically guaranteed over the full horizon and so extensive simulation testing is required. Receding horizon methods can circumvent this difficulty by solving a finite horizon optimal control problem over a shorter horizon [9]. Another possibility is to use linear-parameter-varying methods. Linear quadratic regulators and neighboring optimal controllers use similar control laws (open-loop feedforward and linear state feedback), although the weighting matrices used are different [6]. Linear quadratic regulators are easier to apply to state-constrained optimal control problems, and some margin for the constraints has to be assumed. Active state constraints and other hard constraints can be problematic for neighboring optimal controllers [10], and more stringent technical conditions for the nominal trajectory are required [5]. With LQR, the gain profiles computed for a single reference trajectory can sometimes be used to track different reference trajectories [7,8]. A comparison between neighboring optimal controllers and LQR for commercial airliners is presented in [11].

Preview control for state-space models was first proposed in [12] for discrete-time tracking of a reference trajectory over a fixed horizon. In the aerospace community, preview control has been considered for helicopter and missile terrain, following [13,14]. The use of preview controllers for approximate stable inversion of nonminimum phase plants has been studied in [15,16]. Further afield, preview controllers are prevalent in vehicle suspension [17] and automated road tracking [18]. A good overview of preview control and its relation to other digital tracking controllers can be found in [19].

Some aerospace researchers have recently investigated the potential of online trajectory optimization using two-degree-of-freedom controllers (a trajectory generator combined with a feedback controller) [20]. Trajectory optimization using differential flatness [20] or pseudospectral methods [21] have been considered for the design of the feedforward component. The feedback controller can be designed using regulation methods or neighboring optimal control. In our application, the optimal trajectory is generated before the mission and is not updated in flight.

It is well known that standard proportional navigation (PN) laws do not allow impact angles to be prescribed [22]. There are, however, impact-angle guidance laws that allow both the miss distance and impact-angle errors to be minimized. The approaches can be categorized as those based on geometrical principles [23], biased PN [24], or optimal control theory [25].

Most model-based approaches make simplifying assumptions to derive impact-angle laws (e.g., linear models and zero lags), and so performance has to be judged in nonlinear simulation models. Possibly for these reasons, the extension to 3-D is less well covered in the literature. Biased PN for a bank-to-turn airframe has been used in [26]. The acceleration demands are calculated in Cartesian axes with bias terms included, and polar converting logic is used to transform these demands into bank-to-turn controls. Geometrical principles in conjunction with vector mechanics are used in [27,28] to derive a guidance law that can be used in 3-D. Similarly, a vector formulation for an impact-angle guidance law using optimal control theory is presented in [25]. Online navigation gain adjustment is tested using a 3-D point-mass model in [29]. Within the industry, impact-angle guidance laws based on optimal control theory have been successfully implemented on a number of missile systems [30].

III.   Model

The six-degree-of-freedom (6-DOF) model used in this study is based on the bank-to-turn airframe model presented in [1]. The vehicle equations are summarized as follows:

$$\dot{R}_{sx} = U[\cos(\theta)\cos(\psi)] + V[\sin(\phi)\sin(\theta)\cos(\psi) \\ - \cos(\phi)\sin(\psi)] + W[\sin(\phi)\sin(\psi) + \cos(\phi)\sin(\theta)\cos(\psi)] \tag{1}$$

$$\dot{R}_{sy} = U[\cos(\theta)\sin(\psi)] + V[\cos(\phi)\cos(\psi) \\ + \sin(\phi)\sin(\theta)\sin(\psi)] + W[\cos(\phi)\sin(\theta)\sin(\psi) \\ - \sin(\phi)\cos(\psi)] \tag{2}$$

$$\dot{R}_{sz} = U[\sin(\theta)] - V[\sin(\phi)\cos(\theta)] - W[\cos(\phi)\cos(\theta)] \tag{3}$$

$$\dot{U} = \frac{Q_d S_r C_x - \dot{m}U}{m} + \frac{T}{m} - g\sin(\theta) - rV + qW \tag{4}$$

$$\dot{V} = \frac{Q_d S_r C_y - \dot{m}V}{m} + g\sin(\phi)\cos(\theta) + pW - rU \tag{5}$$

$$\dot{W} = \frac{Q_d S_r C_z - \dot{m}W}{m} + g\cos(\phi)\cos(\theta) + qU - pV \tag{6}$$

$$\dot{\phi} = p + q\tan(\theta)\sin(\phi) + r\tan(\theta)\cos(\phi) \tag{7}$$

$$\dot{\theta} = q\cos(\phi) - r\sin(\phi) \tag{8}$$

$$\dot{\psi} = q\left(\frac{\sin(\phi)}{\cos(\theta)}\right) + r\left(\frac{\cos(\phi)}{\cos(\theta)}\right) \tag{9}$$

$$\dot{p} = \frac{Q_d S_r l_r C_l}{I_{xx}} - \frac{(I_{zz} - I_{yy})}{I_{xx}}qr \tag{10}$$

$$\dot{q} = \frac{Q_d S_r l_r C_m}{I_{yy}} - \frac{(I_{xx} - I_{zz})}{I_{yy}}rp \tag{11}$$

$$\dot{r} = \frac{Q_d S_r l_r C_n}{I_{zz}} - \frac{(I_{yy} - I_{xx})}{I_{zz}}pq \tag{12}$$

$$\dot{T} = -T + T_{\text{dem}} \tag{13}$$

$$\dot{m} = -(T)\text{SFC} \tag{14}$$

$$Q_d = 0.5\rho(U^2 + V^2 + W^2) \tag{15}$$

The inertial linear (U, V, and W) and angular (p, q, and r) velocities are specified in body-fixed axes. The linear velocity components are then expressed in space axes and integrated to obtain the position coordinates (R_{sx}, R_{sy}, and R_{sz}).

A standard atmosphere model is used, and so the air density ρ and sonic speed are polynomial functions of altitude. The aerodynamic forces and moments are calculated as

$$C_x = (C_{x0} + k_d C_{x\alpha}\alpha^2 + C_{x\delta p}|\delta_p'| + C_{x\delta q}|\delta_q'| + C_{x\delta r}|\delta_r'|) \tag{16}$$

$$C_y = (C_{y\beta}\beta + (C_{y\delta})\delta_r') \tag{17}$$

$$C_z = (C_{z0} + C_{z\alpha}\alpha + (C_{z\delta})\delta'_q) \tag{18}$$

$$C_l = \left(C_{l\beta}\beta + \frac{C_{lp}l_r p}{V_t} + (C_{l\delta})\delta'_p + (C_{l\alpha\beta1})\alpha\beta\right) \tag{19}$$

$$C_m = \left(C_{m0} + C_{m\alpha}\alpha + \frac{C_{mq}l_r q}{V_t} + (C_{m\delta})\delta'_q\right) \tag{20}$$

$$C_n = \left(C_{n\beta}\beta + \frac{C_{nr}l_r r}{V_t} + (C_{n\delta})\delta'_r\right) \tag{21}$$

Note that the expressions for the drag coefficient in Eq. (16) and the roll coefficient in Eq. (19) have extra terms, which are neglected in the open-loop optimal control calculations (the drag due to fin deflections is now included). The additional data for these derivatives are $C_{x\delta p} = 0.42$, $C_{x\delta q} = 0.21$, $C_{x\delta r} = 0.18$, and $C_{l\alpha\beta1} = -16.41$. Higher-order terms or tabular lookups may be required for sustained pitching maneuvers.

Finally, a set of autopilot equations to calculate the fin-angle demands is required:

$$\dot{x}_{ap} = K_{1p}(A_{zd} - A'_{mz}) - K_{2p}(g\cos(\theta)\cos(\phi) + A'_{mz})$$
$$- K_{3p}(q) + K_{1\phi}(\phi_d - \phi)\beta \tag{22}$$

$$\dot{y}_{ap} = K_{1y}(A_{yd} - A'_{my}) + K_{2y}(g\cos(\theta)\sin(\phi) + A'_{my})$$
$$- K_{3y}(r) + K_{1\phi}(\phi_d - \phi)\alpha \tag{23}$$

$$\delta_p = K_{1\phi}(\phi_d - \phi) - K_{1r}(p) \tag{24}$$

$$\delta_q = x_{ap} - K_{4p}(q) \tag{25}$$

$$\delta_r = y_{ap} - K_{4y}(r) \tag{26}$$

where K_{ip}, K_{iy} ($i = 1, \ldots, 4$), $K_{1\phi}$, and K_{1r} are autopilot gains. The pitch and yaw autopilots assume a proportional–integral–derivative structure for improved steady-state tracking and are of third order. The roll autopilot is a simple second-order proportional-derivative controller, which ensures a fast response to roll-angle demands. The controls are the pitch autopilot input $u_1 = A_{zd}$, the roll autopilot input $u_2 = \phi_d$, and the thrust demand $u_3 = T_{dem}$ in Eq. (13). Note that for a bank-to-turn system, the yaw autopilot demand A_{yd} is set to zero. The accelerations measured at the accelerometers A'_{my} and A'_{mz} are

$$A'_{my} = \frac{Q_d S_r C_y}{m} + d(\dot{r}) \tag{27}$$

$$A'_{mz} = \frac{Q_d S_r C_z}{m} - d(\dot{q}) \tag{28}$$

where d is the distance between the accelerometers and the center of gravity, and $Q_d S_r C_y/m$ and $Q_d S_r C_z/m$ are the accelerations at the center of gravity. The fin angles are mixed to obtain control surface commands that are subject to actuator dynamics. The actuator dynamics are modeled as second-order lags [1]. The actuator outputs are then converted into achieved fin angles δ'_p, δ'_q, and δ'_r. Additional equations for the seeker gimbal angles and the DBS angles can be found in [1,2]. Similarly, the nominal missile data can be found in [1].

In the closed-loop simulations, the model is upgraded to include steady-state wind, gusts, and modeling errors (discussed in this section) and instrumentation/navigation errors (Sec. IV). The main modeling assumptions are as follows:

1) A flat-Earth model is used.

2) The target coordinates are fixed.

3) The seeker is staring at the target throughout flight and is space-stabilized against the missile motion.

4) The missile's main body product of inertia I_{xz} is neglected.

5) The ground terrain is flat and level.

The use of a terminal guidance law would allow assumption 2 to be relaxed. This is because the switch to the terminal guidance law means that the missile is no longer confined to following the optimal trajectory. Hence, the guidance scheme should be able to accommodate a moderate level of uncertainty in the target position (which depends on the reachability of the terminal constraints as well as the effectiveness of the terminal guidance law and the available homing time). Assumption 3 is imposed as the current study is focused on the guidance performance, and so a simplified seeker representation is used. Assumptions 4 and 5 are inherited from the open-loop studies, but are not critical to the design methodology presented here.

To compensate for gusts and modeling errors, the closed-loop simulations require some control reserve, and approximately 1-g reserve in the pitch control (u_1) is assumed, along with 5 deg in the roll-angle demand u_2 and 1500 N thrust reserve u_3. Hence, the control limits are as follows in the closed-loop simulations:

$$-50 \text{ m s}^{-2} \leq u_1 \leq 50 \text{ m s}^{-2} \tag{29}$$

$$-85 \text{ deg} \leq u_2 \leq 85 \text{ deg} \tag{30}$$

$$1000 \text{ N} \leq u_3 \leq 6000 \text{ N} \tag{31}$$

Note that for the open-loop simulations, $|u_1| \leq 40$ m s^{-2}, $|u_2| \leq 80$ deg, and the thrust upper limit is a function of altitude (an upper limit of 4500 N is used at sea level [1]). By adjusting the thrust, the controller will attempt to follow a time-parameterized trajectory (as opposed to a path following approach in which the temporal requirements are not specified).

To test the closed-loop guidance methods in more realistic conditions, model uncertainties are included for each of the linear stability derivatives. The model uncertainties are represented as fixed additive perturbations. A maximum level of uncertainty of 30% is chosen and a uniform random number is chosen for the modeling uncertainty. This gives the perturbed model that is used in the closed-loop testing. For instance, the zero-incidence drag derivative C_{x0} is varied according to

$$C'_{x0} = C_{x0}(1 + \Delta_1) \tag{32}$$

$$-0.3 \leq \Delta_1 \leq 0.3 \tag{33}$$

where Δ_1 has a uniform distribution and is fixed for each Monte Carlo replication. The other stability derivatives are perturbed in a similar fashion (using different random values for the perturbations $\Delta_2, \Delta_3, \ldots, \Delta_{21}$).

The wind is first modeled in Earth axes as the sum of a steady-state component (u_{ss}, v_{ss}, and w_{ss}) and a gust component (u_{gw}, v_{gw}, and w_{gw}), which is based on the Dryden turbulence model [31]. The steady-state wind components are drawn from a uniform distribution with bidirectional limits of 5 m s^{-1}. The wind description is then rotated into body axes and subtracted from the vehicle inertial velocities in body axes to obtain the velocities of the vehicle relative to the air (U', V', and W') [32]. The velocities relative to the air are then used to compute the airspeed, dynamic pressure, and incidence angles.

IV. State Construction

The control law requires an estimated state vector. The estimated state vector will be used to compute the guidance signals that steer the missile onto the optimal trajectory. The state vector is composed of the positions (R_{sx}, R_{sy}, and R_{sz}), the linear velocities (U, V, and W),

the angular velocities (p, q, and r), the Euler angles (θ, ψ, ϕ), and the internal autopilot states (x_{ap} and y_{ap}). This is a reduced-order state vector, because the states corresponding to engine lags, fuel, and actuator dynamics are not included in the low-order control law; there are 14 states that have to be estimated to implement the control law. Additionally, the autopilots will use measured data that will be subject to noise. The vehicle inertial measurement unit (IMU) provides periodic measurements of accelerations and body rates in body-fixed axes. A strapdown inertial navigation system (INS) can use these measurements to estimate the attitude of the vehicle and to obtain the position and velocities in space axes.

Additional sensors could be onboard the vehicle that could be used to improve the navigation accuracy. Global positioning systems provide measurements of position and velocities in space axes (albeit at a slower rate than the INS) and can also be used to derive attitude information. In addition, the DBS measurements of range and closing speed can be combined with the INS data to improve the navigation estimates. These possibilities have not been pursued in this study, but could improve the trajectory-tracking performance further. An alternative method is to use a Kalman filter to estimate the state deviation from the optimal trajectory (rather than estimating the absolute value of the state).

The IMU measures the acceleration vector $\boldsymbol{a}_m = [A_{mx}, A_{my}, A_{mz}]^T$ and the body-rate vector $\boldsymbol{\omega}_b = [p, q, r]^T$.[‡] The approach adopted in this study uses a simplified representation of an INS with scale factor errors, bias, and noise. It is assumed that the IMU measurements of acceleration and body rates are available throughout the flight. These measurements are corrupted by additive noise, bias, and scale factor errors that contaminate the true values:

$$\bar{\boldsymbol{a}}_m = (\boldsymbol{I} + \delta\boldsymbol{S}_a)(\boldsymbol{a}_m + \delta\boldsymbol{B}_a + \boldsymbol{v}_a) \qquad (34)$$

$$\bar{\boldsymbol{\omega}}_b = (\boldsymbol{I} + \delta\boldsymbol{S}_g)(\boldsymbol{\omega}_b + \delta\boldsymbol{B}_g + \boldsymbol{v}_g) \qquad (35)$$

An overbar is used to denote a measured or estimated quantity. The measurement noise ($\boldsymbol{v}_a$ and $\boldsymbol{v}_g$) is represented by zero-mean Gaussian noise that is uncorrelated for each axis.[§] The matrix $\boldsymbol{I}$ is a 3×3 identity matrix, $\delta\boldsymbol{S}_a$ and $\delta\boldsymbol{S}_g$ are instrument scale factor errors, and $\delta\boldsymbol{B}_a$ and $\delta\boldsymbol{B}_g$ are the accelerometer and gyro biases. The values assumed for the instruments are listed in Table 1. The measurements are assumed to be available at an update rate of 1 KHz. Quaternions are used to obtain the attitude of the vehicle and the body-to-space transformation matrix [31]; the quaternion elements are q_0, q_1, q_2, and q_3.[¶] The quaternion differential equations are

$$\begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & -\bar{p} & -\bar{q} & -\bar{r} \\ \bar{p} & 0 & \bar{r} & -\bar{q} \\ \bar{q} & -\bar{r} & 0 & \bar{p} \\ \bar{r} & \bar{q} & -\bar{p} & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \qquad (36)$$

with $q_0(0) = 1.0$, $q_1(0) = 0.0$, $q_2(0) = 0.0$, and $q_3(0) = 0.0$. This initial condition corresponds to straight and level flight with zero roll angle (the body rates are also assumed to be zero at the initial time). The estimated attitude of the vehicle is calculated using

$$\bar{\phi} = \tan^{-1}\left[\frac{2(q_2q_3 + q_0q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2}\right] \qquad (37)$$

$$\bar{\theta} = \sin^{-1}[-2(q_1q_3 - q_0q_2)] \qquad (38)$$

$$\bar{\psi} = \tan^{-1}\left[\frac{2(q_1q_2 + q_0q_3)}{q_0^2 + q_1^2 - q_2^2 + q_3^2}\right] \qquad (39)$$

[‡]The accelerometer-moment-arm correction is assumed in the following analysis.

[§]Correlated noise has also been investigated and the results are not significantly altered.

[¶]Quaternions are preferable in the implementation of an inertial navigation system, due to their good numerical properties [31].

Table 1 IMU error characteristics

Instrument	Scale factor error	Bias	Noise standard deviation
Gyro	1.5×10^{-4}	5×10^{-6} rad/s	1.75×10^{-4} rad/s
Accelerometer	3×10^{-4}	9.81×10^{-3} m s^{-2}	2×10^{-2} m s^{-2}

The Euler angles need to be derived from the quaternions, because the original trajectory optimization used an Euler-angle parameterization. In addition, the roll autopilot uses the estimated roll angle and, similarly, the gravitational compensation terms in the pitch and yaw autopilots are calculated using the estimated Euler angles. The body-to-space transformation matrix is

$$\begin{aligned} \boldsymbol{T}_{\mathrm{BS}} &= \begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ T_{31} & T_{32} & T_{33} \end{bmatrix} \\ &= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \end{aligned}$$

$$(40)$$

The gravity-compensation terms in the autopilot are

$$g_{\mathrm{compy}} = T_{32}g \qquad (41)$$

$$g_{\mathrm{compz}} = T_{33}g \qquad (42)$$

which are sent to the pitch and yaw autopilots. Now attention is turned to the position and velocity measurements. First, the inertial body axes accelerations are transformed into inertial space axes, with compensation for gravity in the z direction:

$$\bar{\boldsymbol{a}}_s = \boldsymbol{T}_{\mathrm{BS}}\bar{\boldsymbol{a}}_m + \boldsymbol{g}_s \qquad (43)$$

where $\bar{\boldsymbol{a}}_s$ is the acceleration vector in space axes, and $\boldsymbol{g}_s = [0, 0, g]^T$ is the gravity vector in space axes. Next, the velocity and position estimates in space axes are obtained through integration:

$$\dot{\bar{\boldsymbol{v}}}_s = \bar{\boldsymbol{a}}_s \qquad (44)$$

$$\dot{\bar{\boldsymbol{r}}}_s = \bar{\boldsymbol{v}}_s \qquad (45)$$

with initial conditions $\bar{\boldsymbol{v}}_s(0) = [272 \text{ m s}^{-1}, 0, 0]^T$ and $\bar{\boldsymbol{r}}_s(0) = [0, 0, 60 \text{ m}]^T$. The body-axis components of velocity can be obtained by rotating the velocity components in space axes into body axes (to obtain $\bar{U}$, $\bar{V}$, and $\bar{W}$). An estimate of the incidence and sideslip angles can be obtained from $\bar{U}$, $\bar{V}$, and $\bar{W}$, and the estimated position vector is $\bar{\boldsymbol{r}}_s = [\bar{R}_{sx}, \bar{R}_{sy}, \bar{R}_{sz}]^T$. In this paper, the incidence and sideslip terms in the autopilot are estimated from the INS data and are subject to errors when wind and gust are present. Although the performance is not very sensitive to the estimation errors in incidence and sideslip, a more complex estimator could be used to accommodate strong gusts [33].

Attention is now turned to the internal autopilot states x_{ap} and y_{ap} that are used in the control law. These would be used by the autopilot software to construct the actuator commands. Performance is only weakly dependent on these states. Because the autopilot calculates the fin-angle demands via these states, they are accessible with a one-sample delay (because the fin angle can only be calculated once the acceleration demand is specified, and the acceleration demand cannot be specified until the state estimate is constructed). This is not a problem, because the update rate for the controls in the current simulation is quite high (0.01 s). Denoting k as the sample time, the state estimate at time k is

$$\bar{\boldsymbol{x}}_k = [\bar{R}_{sx}, \bar{R}_{sy}, \bar{R}_{sz}, \bar{U}, \bar{V}, \bar{W}, \bar{\phi}, \bar{\theta}, \bar{\psi}, \bar{p}, \bar{q}, \bar{r}, \bar{x}_{\mathrm{ap}}, \bar{y}_{\mathrm{ap}}]^T \qquad (46)$$

Previously, we assumed that the initial state for the INS corresponds to the true vehicle state in Eqs. (36), (44), and (45). In reality, the true initial state and the initial state used for navigation may be mismatched. To account for this difference, initial position errors are assumed (this may correspond to a buildup of position errors from the midcourse to terminal guidance phases). A random position error with a uniform distribution of 5 m in each axis is imposed on the true state; that is, $R_{sx}(0)$, $R_{sy}(0)$, and $R_{sz}(0)$ are randomized.[**]

Because the noise on the IMU measurements is relatively small, using these measurements directly proves to be satisfactory.

Finally, it should be mentioned that in the closed-loop simulations, the autopilot equations (22) and (23) are also modified to use estimated quantities (from the INS) in lieu of true quantities.

V. Controller Design

The trajectory optimization stage is treated in detail in [1] and is therefore only briefly described here. Broadly, the guidance problem involves low-altitude flight to minimize exposure, while satisfying the constraints associated with DBS radars. To achieve adequate crossrange resolution (with DBS radars), the azimuth angle between the sightline and velocity vector has to be maintained; fine downrange resolution can be obtained using pulse compression. The optimal control problem uses a bank-to-turn airframe, and the cost function to be minimized is a linear combination of flight time and time-integrated height deviation from a reference level. Terminal constraints associated with the miss distance, incidence at impact, and the impact angle are also mandatory and are thus included. The impact-angle constraints usually require a pop-up-and-dive (bunt) maneuver. Control bounds and state constraints are also imposed [1]. There are many ways of numerically solving optimal control problems [5,34]. Direct methods discretize the optimal control problem and use nonlinear programming software to solve the resulting problem [34], whereas indirect methods invoke the necessary conditions of optimal control and require estimates of the costates. The direct methods are more widely used, because they are easier to apply to complex problems. In [1], a direct method was employed, the controls were parameterized using piecewise constant functions, and some characteristic DBS trajectories were presented. These trajectories provide a good tradeoff between the various competing constraints. Solving the trajectory optimization problem yields the optimal state trajectory x^* and the open-loop optimal controls u^* (pitch-acceleration, roll-angle, and thrust demands).

The task here is to design an online controller that preserves the optimal trajectory x^* in flight in spite of the modeling errors, disturbances, and physical constraints. In this paper, a reduced-order set of equations is used for the controller design purposes; the states relating to thrust lag [Eq. (13)], vehicle mass [Eq. (14)], and actuator dynamics are neglected.[††]

Assuming the optimal trajectory data of [1], the optimal controls are resampled with a time step of 0.01 s. The optimal trajectories are then regenerated by integrating the state equations using a fixed-step integrator (the optimal trajectories in [1] were computed using a variable-time-step integrator). This step gives a sampled version of the optimal states and controls x_k^* and u_k^*. There is a minor mismatch between x^* and x_k^* due to integration errors and the simplifications used in the open-loop trajectory optimization. Because the controller will switch to a terminal guidance law, it is possible to follow x_k^* and still achieve good overall performance.

The closed-loop controller design is based on linear quadratic output regulation. The closed-loop controller is based on (linear) models that were evaluated for small perturbations around the nominal trajectory state pair x_k^* and u_k^* at 2-s (or 200-sample) intervals. In one of the study cases (case B), some additional linearization points were used (at 43.0 and 45.0 s) to improve performance. Symbolic tools are used to obtain the linearized system matrices that are based on the reduced-order 6-DOF equations with

the nominal missile data. These calculations produce a set of continuous-time matrices A_c, B_c, C_c, and D_c. The matrices have the following dimensions: $A_c \in \mathbb{R}^{n \times n}$, $B_c \in \mathbb{R}^{n \times m}$, $C_c \in \mathbb{R}^{r \times n}$, and $D_c \in \mathbb{R}^{n \times m}$, where n is the number of states, m is the number of controls, and r is the number of regulated outputs. These matrices are then discretized using Euler's method with a time step of 0.01 s (giving A, B, C, D). The objective is to track the position states R_{sx}, R_{sy}, and R_{sz} and the forward body-axis velocity U so that C is

$$C = [\, I_4 \quad 0 \,] \tag{47}$$

in which I_4 is a 4×4 identity matrix with the null matrix $0 \in \mathbb{R}^{4 \times 10}$. The optimal output states at the current time are given by $y_k^* = C x_k^*$, and the difference between the optimal output states and the achieved states at the current time is $e_k = y_k^* - y_k$. The perturbed control is $\delta u_k = u_k^* - u_k$. The linear system representing deviations from the open-loop trajectory is

$$\delta x_{k+1} = A \delta x_k + B \delta u_k \tag{48}$$

$$\delta y_k = C \delta x_k \tag{49}$$

where δx_k, δu_k and δy_k represent deviations from the precomputed open-loop trajectories at sample index k. The following cost function is minimized:

$$J = \lim_{N \to \infty} \left[\frac{1}{2} \sum_{k=0}^{N-1} \delta y_k^T Q \delta y_k + \delta u_k^T R \delta u_k \right] \tag{50}$$

The optimal control δu_k is a linear state feedback derived from deviations δx_k from the optimal trajectory. The optimal (operating-point-dependent) gains can be found by solving a discrete algebraic Riccati equation,

$$S_{11} = A^T S_{11} A - (A^T S_{11} B)(B^T S_{11} B + R)^{-1}(A^T S_{11} B) + Q \tag{51}$$

to give the optimal gain sequence:

$$K_a = (B^T S_{11} B + R)^{-1} B^T S_{11} A \tag{52}$$

Linear interpolation that is based on time indexing is used to schedule the gains between operating points. Time-to-go indexing would be more suitable if the thrust cannot be controlled or if the feedback regulation approach is used over the full horizon [35]. The subscripts in S_{11} are used to distinguish the standard state-feedback part of the Riccati equation from the full preview solution that will be introduced later. The Riccati equation associated with the inflated (plant and preview) system is

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{12}^T & S_{22} \end{bmatrix} \tag{53}$$

By linearizing around the nominal trajectory, the control problem is reduced to a regulation problem (i.e., the desired reference trajectory is zero). The open-loop optimal control is a feedforward term that contains future trajectory information (based on the nominal model). The idea explored in this paper is a regulation scheme that makes use of linear quadratic preview and linear quadratic state-feedback control, which is directed to the minimization of trajectory deviations. The control δu_k is computed using the state deviations (from open-loop optimal) and the predicted trajectory errors. These ideas are illustrated in Fig. 1 for tracking an optimal height profile. The nonlinear model is used to calculate the future predicted trajectory $\bar{X}_{pk} \in \mathbb{R}^{14 \times N_P}$ and the predicted future outputs $\bar{Y}_{pk} \in \mathbb{R}^{4 \times N_P}$. The current-time optimal trajectory matrix $X_{pk}^* \in \mathbb{R}^{14 \times N_P}$ is distinct from the current-time optimal state vector $x_k^* \in \mathbb{R}^{14 \times 1}$; similarly, the optimal control matrix $U_{pk}^* \in \mathbb{R}^{3 \times N_P}$ should not be confused with the current-time optimal control vector $u_k^* \in \mathbb{R}^{3 \times 1}$. The current-time optimal state and optimal controls are related to the optimal control and optimal state matrices by

---

[**]It seems arbitrary whether the initial condition errors are imposed on the true initial state or on the initial navigation state.

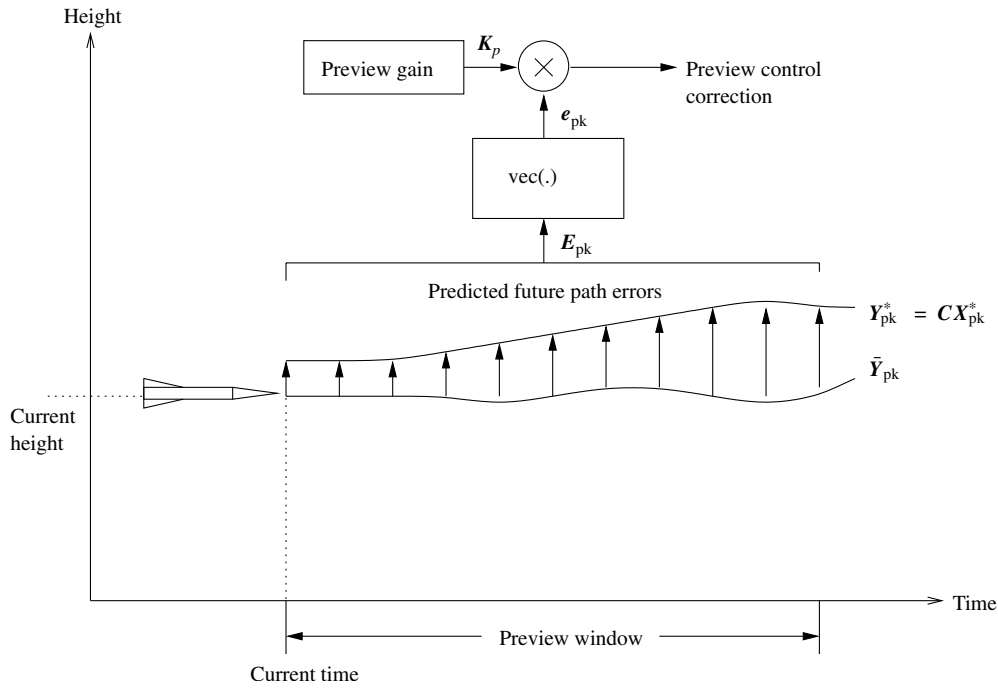[††]These states are restored in the closed-loop simulations.

Fig. 1 The use of preview. The predicted trajectory errors are the difference between the precomputed optimal trajectory Y_{pk}^* and the estimated future trajectory $\bar{Y}_{\mathrm{pk}}$.

$$x_k^* = X_{\mathrm{pk}}^* \overbrace{[1 \quad 0 \quad \cdots \quad 0]^T}^{N_p}, \qquad u_k^* = U_{\mathrm{pk}}^* \overbrace{[1 \quad 0 \quad \cdots \quad 0]^T}^{N_p} \quad (54)$$

By exploiting preview control, the missile is able to anticipate if it is being driven off the optimal trajectory (rather than purely estimating its current position relative to the optimal trajectory). Future controls are essentially deviations from the open-loop optimal control u_k^*.

The integrated control scheme makes use of the precomputed open-loop optimal control, linear quadratic regulator feedback, and preview. This scheme is illustrated in Fig. 2, in which the plant represents the missile system that has the autopilot demands and thrust demand as the input u_k. The preview and LQR gain matrices are denoted by K_a and K_p. Both the plant and sensors are subject to additive noise (v and w). The trajectory generator is an offline process, and so x_k^* and u_k^* are not regenerated online. In the case without preview, $u_k^{\mathrm{prev}} = 0$ and the controller reduces to a standard discrete-time regulator. If $\bar{X}_{\mathrm{pk}} = X_{\mathrm{pk}}^*$ (and so $\bar{Y}_{\mathrm{pk}} = Y_{\mathrm{pk}}^*$), the preview-control sequence u_k^{prev} will be identically zero. In the same way, if the measured state $\bar{x}_k$ is equal to x_k^*, the feedback control δu_k will also be identically zero. In the event of perfect control $u_k = u_k^*$, the missile is flown as if it were the simulation model used to solve the open-loop optimal control problem. The preview control is combined with the LQR feedback term and the open-loop optimal

control feedforward term:

$$u_k = u_k^* + K_a(\bar{x}_k - x_k^*) + u_k^{\mathrm{prev}} \qquad (55)$$

In the perfect-information case, the true current-time state x_k is used to calculate the controls (i.e., $\bar{x}_k = x_k$). The benefits of using the preview term are investigated in the sequel. A standard LQR method is used (with the preview term set to zero) as a baseline. The preview-control correction u_k^{prev} is calculated using the following steps:

1) Set $k = 0$.

2) Set $\bar{x}_{\mathrm{pk}1} = \bar{x}_k$, where $\bar{x}_{\mathrm{pk}1}$ denotes the first column of the $\bar{X}_{\mathrm{pk}}$ matrix and the current-time measured state is $\bar{x}_k$.

3) Predict the future trajectory using

$$\bar{x}_{\mathrm{pk}(i+1)} = f(\bar{x}_{\mathrm{pk}i}, u_{k+i-1}^*)$$

for $i = 1, 2, \ldots, N_p - 1$ and form the matrix

$$\bar{X}_{\mathrm{pk}} = [\bar{x}_{\mathrm{pk}1} \quad \bar{x}_{\mathrm{pk}2} \quad \cdots \quad \bar{x}_{pkN_p}]$$

4) Calculate

$$E_{\mathrm{pk}} = \bar{Y}_{\mathrm{pk}} - Y_{\mathrm{pk}}^* = C(\bar{X}_{\mathrm{pk}} - X_{\mathrm{pk}}^*)$$
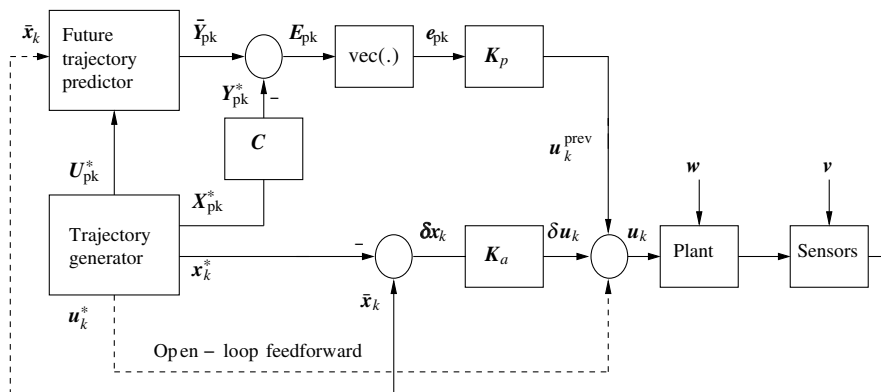

Fig. 2 Trajectory-tracking controller structure.

The matrix $E_{pk} \in \mathbb{R}^{4 \times N_p}$ contains the predicted trajectory errors within the preview window.

5) Calculate $e_{pk} = \text{vec}(E_{pk})$. The vec(.) operator converts a matrix to a vector; that is, for matrix $X \in \mathbb{R}^{m \times n}$,

$$\text{vec}(X) = [x_{11}, x_{21}, \ldots, x_{m1}, x_{12}, \ldots, x_{mn}]^T$$

Hence, $e_{pk} \in \mathbb{R}^{4N_p \times 1}$ is formed by stacking the predicted trajectory errors at each point in the preview window.

6) Calculate the preview correction $u_k^{\text{prev}} = K_p e_{pk}$;

7) Set $k = k + 1$ and return to step 2.

Multiplying e_{pk} by the preview-gain matrix $K_p \in \mathbb{R}^{3 \times 4N_p}$ gives $u_k^{\text{prev}} \in \mathbb{R}^{3 \times 1}$. The prediction

$$\bar{x}_{pk(i+1)} = f(\bar{x}_{pki}, u_{k+i-1}^*)$$

uses the nominal nonlinear model in the open-loop optimal control calculations, and so it includes the thrust lag and vehicle mass variations. Step 3 is accomplished using Euler integration with a time step of 0.01 s. When the preview horizon exceeds the nominal horizon, the preview horizon and gain matrices are truncated. In the next section, an efficient method to compute the matrix K_p is described.

A. Computing the Preview-Gain Matrix

The preview gains can be derived from the LQR gains via a simple recursion [12]. The preview-gain matrix K_p can be shown to be [12]

$$K_p = (B^T S_{11} B + R)^{-1} B^T S_{12} A_p \tag{56}$$

where $A_p \in \mathbb{R}^{4N_p \times 4N_p}$ is a shift-register matrix, and $S_{12} \in \mathbb{R}^{14 \times 4N_p}$ is the matrix induced by the partitioning of the Riccati equation into nonpreview and preview parts. There is a well-known and efficient way to compute S_{12} (see, for example, [12]). The columns of the S_{12} matrix can be calculated using the following recursion:

$$S_{121} = -C^T Q \tag{57}$$

$$S_{12(i+1)} = \Phi_c^T S_{12i} \qquad i = 1, 2, \ldots, N_p - 1 \tag{58}$$

where $\Phi_c = A - BK_a$ is the closed-loop transition matrix. Because this matrix is stable, the amount of preview required will be dependent on the closed-loop eigenvalues. Once S_{12} is computed, then K_p follows from Eq. (56). As with the LQR gains, the preview-gain matrix K_p is stored at each operating point and linearly interpolated.

B. Tuning Parameters for LQR Design

The weighting matrices and preview length have to be tuned to achieve good performance. The following choice of Q and R is used:

$$Q = \text{diag}(50, 50, 50, 100) \tag{59}$$

$$R = \text{diag}(500, 2 \times 10^5, 5 \times 10^{-4}) \tag{60}$$

along with a preview length N_p of 1400 samples (corresponding to a look-ahead time of 14.0 s). Lower-control weightings will lead to a higher-bandwidth controller (with less preview needed). The matrices Q and R are not adjusted at each operating point, because adequate performance is obtained without any retuning.

C. Impact-Angle Control

The regulator illustrated in Fig. 2 operates up until a fixed range-to-go (currently set at 2 km); after this, a terminal guidance law is used to home in on the target. If very good navigation accuracy is available, the regulator can be used up until the intercept point (simulations with perfect state feedback showed that small position errors and impact-angle errors could be achieved using the steady-state regulator over a finite horizon). For extended flight times, the navigation errors will build up and the seeker's measurements must be used for homing guidance.

In our application, the miss distance, the impact-angle error, and the incidence at impact have to be minimized. The standard proportional navigation laws do not address the requirement for a desired approach angle [22], and so alternative guidance laws must be sought. Most air-to-surface missiles approach the target in a straight line in the ground plane. As a result, the impact-angle requirements effectively reduce to a single (vertical) plane problem. For DBS guidance, this simplified approach is not possible due to the sensor constraints (i.e., the missile has to fly a curved approach to the target). In short, a desired approach direction in 3-D has to be specified.

We tested a number of guidance laws available in the open literature that are applicable to 3-D scenarios. A scheme based on circular navigation guidance was selected for further study. The guidance law attempts to drive the missile onto a circular arc impacting the target with the required impact angle. This scheme was presented in 2-D in [23] using state feedback; the extension to 3-D is presented in [27]. These papers use a simplified kinematic model and so it was not immediately clear how the guidance law would perform in a nonlinear simulation model. In our application, the target is static, but the guidance law is extensible to the moving target case [23]. A brief description of the guidance law is presented for completeness (for more detail, the reader is referred to [23,27,28]). It is more natural to explain the principle of the guidance law in 2-D and then to extend the law to the 3-D case.

The geometry of a 2-D engagement is shown in Fig. 3 at a particular time instant in the engagement. The missile and target positions are joined by a circular arc; the desired final velocity vector v_f is a tangent to the circle at the target position. The angle between the sightline vector and the horizontal reference (the sightline angle) is denoted as σ, α_{imp} is the desired impact angle (with respect to a horizontal reference), λ is the angle between the missile velocity vector and the sightline vector, and γ is the missile flight path angle. In Fig. 3, a vertical impact ($\alpha_{\text{imp}} = -90$ deg) is assumed. The guidance law attempts to keep $\epsilon = \lambda$; this strategy ensures that the missile follows the circular arc, hitting the target at the correct impact angle (a proof can be found in [23]). From the geometry, the following relationships can be deduced:

$$\lambda = \sigma - \gamma \tag{61}$$

$$\epsilon = \alpha_{\text{imp}} - \sigma \tag{62}$$

The rate of change of the sightline vector is
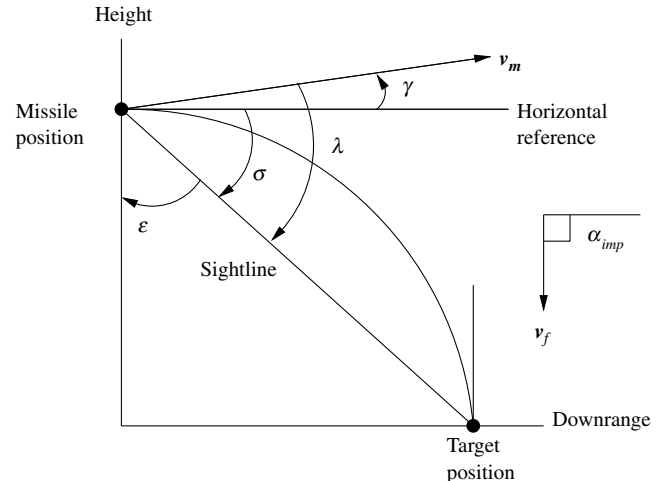
$$\dot{\sigma} = \frac{V_t \sin(\lambda)}{R} \tag{63}$$

Fig. 3   Circular navigation guidance principle (2-D).

where V_t is the magnitude of the velocity vector $\boldsymbol{v}_m$. If Eq. (61) is differentiated, then

$$\dot{\lambda} = \dot{\sigma} - \frac{u_{c1}}{V_t} \tag{64}$$

where u_{c1} is the acceleration demand normal to the velocity vector; dividing this acceleration by the missile speed gives the flight path rate $\dot{\gamma}$. If the control is chosen such that

$$u_{c1} = 2V_t\dot{\sigma} \tag{65}$$

then substitution shows that $\dot{\lambda} = \dot{\epsilon}$. Hence, the control signal u_{c1} will keep the missile on the circular arc, provided that $\lambda(0) = \epsilon(0)$. If $\lambda(0) \neq \epsilon(0)$ (due to a misalignment in the missile velocity vector), there will be an impact-angle error (this is shown in Fig. 3, in which $\lambda \neq \epsilon$). A feedback term proportional to the heading error $k_{p1}(\lambda - \epsilon)$ is necessary to continually correct initial condition errors in $\lambda(0)$. The missile will then eventually converge to the correct circular path if k_{p1} is chosen appropriately. To summarize, in 2-D, the normal acceleration command (in velocity axes) is

$$u_c = \frac{2V_t^2 \sin(\lambda)}{R} + k_{p1}(\lambda - \epsilon) \tag{66}$$

As outlined in [23], this can be viewed as a proportional–integral controller that attempts to regulate the error $\lambda - \epsilon$ to zero.

The guidance law is extended to 3-D in [27,28]. The 3-D guidance law uses similar geometrical principles, though a vector formulation is adopted for convenience. In 3-D, it is more convenient to specify a desired final velocity vector $\boldsymbol{v}_f \in \mathbb{R}^{3 \times 1}$. This vector is related to the impact angles in azimuth and elevation (called β_{imp} and α_{imp}, respectively):

$$\boldsymbol{v}_f = \begin{bmatrix} \cos(\beta_{\mathrm{imp}})\cos(\alpha_{\mathrm{imp}}) \\ \sin(\beta_{\mathrm{imp}})\cos(\alpha_{\mathrm{imp}}) \\ \sin(\alpha_{\mathrm{imp}}) \end{bmatrix} \tag{67}$$

Note that both the impact angles in azimuth and elevation have to be specified (even if there is not a requirement for an azimuth impact angle). If there is no azimuth-impact-angle requirement, one approach is to assume a straight-line approach between the missile and target from the current missile position. Testing has shown that it is better to regard β_{imp} as an additional tuning parameter.

The 3-D guidance law requires the following estimates: the relative position vector $\boldsymbol{x}_r$ and the missile velocity vector $\boldsymbol{v}_m$ (in space axes). The control law uses two vectors: $\boldsymbol{u}_p \in \mathbb{R}^{3 \times 1}$ and $\boldsymbol{u}_f \in \mathbb{R}^{3 \times 1}$; adding these two vectors together gives the desired acceleration demands in velocity axes (including the along-axis component). The calculation of the signal $\boldsymbol{u}_f$ will be described first. The desired velocity vector is reflected about the relative position vector to give $\boldsymbol{v}_d$:

$$\boldsymbol{v}_d = 2\,\mathrm{proj}_{\boldsymbol{x}_r}\boldsymbol{v}_f - \boldsymbol{v}_f \tag{68}$$

where the projection operator performs the calculation:

$$\mathrm{proj}_{\boldsymbol{x}_r}\boldsymbol{v}_f = \left(\frac{\boldsymbol{v}_f^T\boldsymbol{x}_r}{\|\boldsymbol{x}_r\|}\right)\frac{\boldsymbol{x}_r}{\|\boldsymbol{x}_r\|} \tag{69}$$

where $\|\,.\,\|$ is the vector 2-norm. A consideration of the geometry should convince the reader that $\boldsymbol{v}_d$ is the reflection of $\boldsymbol{v}_f$ about $\boldsymbol{x}_r$. The principle of the guidance law in 3-D is to align $\boldsymbol{v}_m$ with $\boldsymbol{v}_d$. This ensures that the missile will approach the target from the correct direction. As before, the controller uses two terms (a term proportional to the angle error and an integral term). The calculation of the proportional term is first described.

The angle between $\boldsymbol{v}_d$ and $\boldsymbol{v}_m$ can be calculated via simple vector geometry:

$$\theta_d = \cos^{-1}\left(\frac{\boldsymbol{v}_d^T\boldsymbol{v}_m}{\|\boldsymbol{v}_d\|\|\boldsymbol{v}_m\|}\right) \tag{70}$$

A control signal proportional to the angle error θ_d is constructed. A unit vector $\boldsymbol{u}_{d1}$ is used to specify the direction of the applied signal, which should be orthogonal to $\boldsymbol{v}_m$. The unit vector $\boldsymbol{u}_{d1}$ is the projection of $\boldsymbol{v}_d$ onto the subspace orthogonal to $\boldsymbol{v}_m$:

$$\boldsymbol{u}_{d1} = \frac{\boldsymbol{v}_d - \mathrm{proj}_{\boldsymbol{v}_m}\boldsymbol{v}_d}{\|\boldsymbol{v}_d - \mathrm{proj}_{\boldsymbol{v}_m}\boldsymbol{v}_d\|} \tag{71}$$

The control signal is then:

$$\boldsymbol{u}_p = k_{p1}\theta_d\boldsymbol{u}_{d1} \tag{72}$$

where k_{p1} is a scalar gain. This proportional term acts to reduce the angular error between $\boldsymbol{v}_m$ and $\boldsymbol{v}_d$, provided that the gain k_{p1} is chosen appropriately.

As with the 2-D case, an additional term in the control law ($\boldsymbol{u}_f$) is calculated that is related to the sightline rate. First, a signal $\boldsymbol{v}_o$ is constructed; the components of $\boldsymbol{v}_o$ are the missile velocity components orthogonal to the sightline vector:

$$\boldsymbol{v}_o = \frac{(-\boldsymbol{v}_m + \mathrm{proj}_{\boldsymbol{x}_r}\boldsymbol{v}_m)}{R} \tag{73}$$

This is the angular velocity of the sightline vector (it is also possible to substitute the seeker's sightline rate vector for $\boldsymbol{v}_o$). Note that the magnitude of the relative position vector $\boldsymbol{x}_r$ is denoted by the range R. The angle between the velocity vector and relative position vector is

$$\eta = \cos^{-1}\left(\frac{\boldsymbol{x}_r^T\boldsymbol{v}_m}{\|\boldsymbol{x}_r\|\|\boldsymbol{v}_m\|}\right) \tag{74}$$

The vector $\boldsymbol{v}_o$ is now rotated into velocity axes using this angle according to

$$\boldsymbol{u}_{d2} = \frac{\boldsymbol{v}_o\cos(\eta)}{\|\boldsymbol{v}_o\|} + \frac{\boldsymbol{x}_r\sin(\eta)}{R} \tag{75}$$

and $\boldsymbol{u}_f$ is defined as

$$\boldsymbol{u}_f = k_{p2}(\|\boldsymbol{v}_o\|\|\boldsymbol{v}_m\|)\boldsymbol{u}_{d2} \tag{76}$$

The total control signal is then calculated as

$$\boldsymbol{u}_c = \boldsymbol{u}_f + \boldsymbol{u}_p \tag{77}$$

After compensating for gravity, the second and third components of $\boldsymbol{u}_c$ are sent to the pitch and yaw autopilots for implementation. For bank-to-turn systems, polar converting logic is needed to convert the acceleration demands in body axes into bank-to-turn controls (see Sec. V.E). The acceleration along the velocity vector is discarded. During the terminal homing phase, the nominal thrust (4500 N) is used. Values of $k_{p1} = 950$ and $k_{p2} = 2$ are used in this study. The azimuth approach angle β_{imp} is also adjusted according to the case. When there are no impact-angle requirements, a standard PN law can be used instead.

One characteristic feature of the impact-angle guidance law is that it is not optimal with respect to control energy [23]. Paradoxically, this can be an advantage because more control action is used early in the trajectory, leaving more control reserve near intercept. Impact-angle performance can be degraded if the controls saturate near intercept. If the guidance law is activated at long ranges, it can suffer from an excessively large turn radius (a potential remedy using saturation functions is suggested in [23]). For our application, the guidance law is only active over the last 2 km, and so problems with excessive height are not relevant. Because the missile is following the optimal trajectory, there should be favorable conditions for the impact-angle guidance law, because less heading-error correction is required (the missile is usually near the apogee when the guidance law is activated). In practice, not all impact angles will be achievable, due to heading and navigation errors, finite homing time, disturbances, and maneuverability constraints.

D. Guidance Law Implementation

The guidance law requires estimates of the relative position vector $\boldsymbol{x}_r$ and the missile velocity vector $\boldsymbol{v}_m$. The former can be estimated by the seeker (or obtained from the INS if seeker estimates are not available); the latter can be approximated using the INS outputs.

The seeker measures the range, range rate (along the sightline), and the angles to the target. The angles are measured with respect to the seeker's antenna and are then filtered to track the target. Seeker noise with a standard deviation of 1×10^{-3} rad is added to the true sightline angles, and noise with a standard deviation of 0.25 m is added to the true range. At the 2-km range, the seeker should have a high signal-to-noise ratio so that the range error will be determined by the seeker's range resolution (this will also depend on the DBS trajectory flown). The angle noise can be computed using estimates of the signal-to-noise ratio and the seeker's angle-tracking slopes. The noisy measurements are filtered using fading-memory filters [22], and the filtered estimates are used to reconstruct the relative position vector $\boldsymbol{x}_r$. The guidance law can then be implemented using the estimated values of $\boldsymbol{x}_r$ and $\boldsymbol{v}_m$.

E. Bank-to-Turn Logic

The guidance law supplies the acceleration demands in velocity axes. These are rotated into body axes (and compensated for gravity), yielding pitch and yaw acceleration demands A'_{zd} and A'_{yd}. These signals are converted to bank-to-turn controls using

$$A_{zd} = \text{sign}(A'_{zd})\sqrt{(A'_{yd})^2 + (A'_{zd})^2} \qquad (78)$$

$$\phi_d = \text{atan}\left(\frac{-A'_{yd}}{A'_{zd}}\right) \qquad (79)$$

The logic ensures that the direction of the maneuver is preserved. A singularity occurs when A'_{zd} approaches zero, which may induce roll oscillations if the pitch demand is noisy. These signals are then sent to the pitch and roll autopilots for implementation.

F. Terminal Incidence Control

In addition to the impact-angle constraint, there is a requirement for small incidence angles at impact. There is a tradeoff between incidence at impact, impact-angle error, and miss distance. A simple scheme is used to minimize the incidence at impact. Suppose that a critical range R_t is reached. Let the pitch acceleration demand at this range be $\hat{u}_1$ and set a counter j to zero at this range. The controls are then reduced to zero using

$$u_{1j+1} = u_{1j} - \delta t(c_1 u_{1j}) \qquad (80)$$

with $u_{10} = \hat{u}_1$. This logic reduces the controls to zero at some finite range-to-go, the decay rate being governed by c_1. The values used in this study are $R_t = 100$ m and $c_1 = 8.0$. These were obtained by manual tuning rather than numerical optimization.

VI.   Results

The aim of this section is to analyze the performance of the controller for two test cases (see Secs. VI.A and VI.B) that are taken from [1]. The method will then be evaluated using Monte Carlo analysis in Sec. VI.C. There are two interrelated objectives: accurately following the optimal trajectory up until the terminal guidance range of 2 km and achieving impact-angle, incidence, and miss-distance specifications using the impact-angle controller. It is important for these objectives to be met in the presence of the modeling errors, gusts, and sensor noise.

Two of the optimal trajectories presented in [1] are used to test the controller. Other optimal trajectories were also tested (including the other case in [1]), but the results are not presented here, due to space limitations. The model is integrated using a fixed-step fourth-order Runge–Kutta integrator with a time step of 0.001 s. The preview calculations and control updates are processed every 0.01 s. The DBS

Table 2   Summary of cases

Case	A	B
Terminal crossrange offset, km	0.0	1.0
Terminal pitch angle, deg	−75	−45
Terminal yaw angle, deg	Free	−86

imaging will only be required up to a finite range-to-go, after this, seeker updates every 0.01 s (over the terminal homing phase) are assumed. The differences between the cases are summarized in Table 2. Case B is the more challenging of the two test cases, because aggressive maneuvering is required; this case requires DBS guidance up until (500-m) range-to-go. It was not possible to achieve the steep azimuth impact (−86 deg) for case B, and so this angle was reduced to −63 deg in the closed-loop simulations. The available control margins are quite small in case B and it may not be feasible to achieve a steep azimuth-impact-angle constraint (which is potentially less important than the elevation impact constraint). The reduction in the azimuth impact angle in case B means that DBS cannot be used down to the 500-m range; simulations show that a reasonable azimuth offset angle for DBS (greater than 10 deg) can still be achieved up until the 1-km range with the relaxed azimuth impact angle.

A number of performance metrics are included for each case, enabling a comparison to be undertaken: the missile exposure time is calculated as the time spent above 100 m height, and the bunt range is defined as the range-to-go when the missile crosses the 100 m exposure height for the first time [1]. The crossrange resolution is computed at the 2-km range-to-go (corresponding to a DBS integration period of 0.3 s) [1]. The miss distance is computed as the point of closest approach to the target. The other performance indices should be self-explanatory.

To evaluate the tracking performance, the tracking errors for downrange, crossrange, and height are computed. Both the mean square error (MSE) and maximum tracking errors are computed for each of these cases. Miss-distance and impact error statistics are computed for the successful cases (cases in which the miss is less than 10 m). The number of unsuccessful cases is also noted for each example. The failures are principally due to the combination of errors and result in controller saturation. In these cases, it may not be possible to achieve both a small miss and impact angle, and so some relaxation of the impact angle might be necessary.

A.   Case A: 0-km Offset

An example from case A (which has a 0-km crossrange offset) is presented for illustrative purposes. Perfect state information is assumed for both the regulator and the terminal guidance law, although modeling errors and gusting are present. The corresponding 3-D trajectory is shown in Fig. 4. The crossrange and roll-angle demand profiles, as functions of downrange, are shown in Fig. 5, and the height and pitch acceleration-demand profiles appear in Fig. 6. The speed and thrust demand, as functions of downrange, appear in Fig. 7, and the corresponding position-tracking errors (for the preview and nonpreview cases) are shown in Fig. 8. In Figs. 5 and 6, the open-loop optimal trajectories and controls are also displayed; these correspond to the outputs of the optimal control software $\boldsymbol{x}^*$ and $\boldsymbol{u}^*$, rather than the resampled trajectories and controls $\boldsymbol{x}_k^*$ and $\boldsymbol{u}_k^*$ used in the feedback simulations. The regulator will attempt to minimize the output tracking error $\boldsymbol{C}(\boldsymbol{x}_k^* - \bar{\boldsymbol{x}}_k)$. Note that in Fig. 8, the tracking errors are only computed over the period when the LQR controller is active (up until 49.0 s or approximately 14 km downrange). After this point, the impact-angle controller is free to choose any path to intercept the target; this path will usually deviate from the optimal trajectory, because the impact-angle controller will use a different control strategy (although the two paths will eventually converge at the impact point). The switchover point is denoted by an asterisk on the plots. The position and speed plots are for the LQR controller without preview; the trajectories with the preview term are very similar, and so they are not shown.
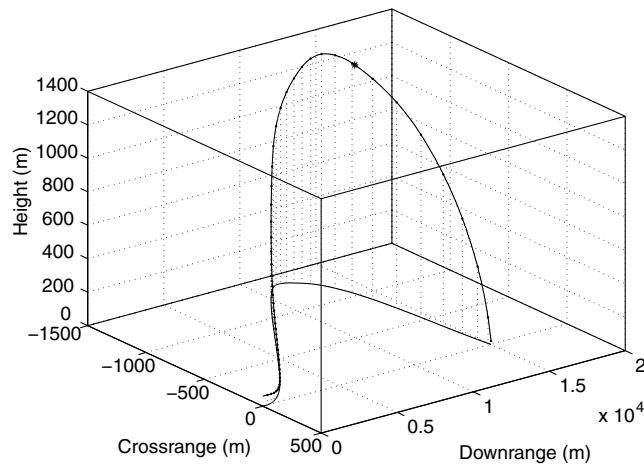
Fig. 4   Three-dimensional trajectory (case A); the asterisk marks the point at which the terminal impact controller replaces the LQR controller.

Figure 8 shows that the preview term reduces the maximum tracking errors along all three axes, and lower mean square errors are also achieved. Although this should help the missile achieve good precision with respect to the terminal constraints, this may not always be the case (e.g., if the missile is slightly higher than the optimal trajectory this may give more favorable conditions for the impact-angle controller). Nonetheless, it is encouraging that the preview term reduces the trajectory-tracking errors.

The 3-D trajectory in Fig. 4 shows the characteristic features of a DBS trajectory. The missile moves away from the target in crossrange, achieving an offset of approximately 1381 m before homing in on the target. The exposure time is approximately 34.0 s. The open-loop and closed-loop controls are well matched up until 14 km downrange (see Figs. 5 and 6). There is some slight noise on the closed-loop controls, due to gusts. At 2-km range-to-go (approximately 13.8 km downrange), the controller switches to the impact-angle guidance law. The pitch acceleration hits its upper saturation limit of 50 m s^{-2} between approximately 14 and 14.5 km downrange. This is a characteristic feature of the guidance law;

essentially, the strategy is to use maximum control early, leaving more control available near intercept. Over the last 500 m in downrange, the pitch demand comes off its saturation limit, reducing to approximately 2.2 g, but then begins to increase gradually. There is gust noise on the pitch acceleration demand during this period. The roll-angle demand is reducing to zero during this period. At 100 m range-to-go, the pitch demand is gracefully reduced to zero, which is effective in reducing the incidence at impact. This, however, tends to increase the miss distance and impact-angle error, and so some tradeoff is necessary. In this example, the tradeoff is very well-handled. Both the preview and nonpreview controllers achieve similar results (a miss distance of 2 m) with a small impact-angle error (1.7 deg). The incidence at impact is approximately 0.8 deg. Hence, the impact-angle controller is able to achieve good accuracy with respect to the terminal constraints.

The thrust demand attempts to adjust the speed to match the open-loop speed profile; a maximum speed of approximately 302 m s^{-1} is reached. The thrust is kept constant up until 11 km downrange, there then follows a slight increase in thrust followed by a decrease to approximately 1800 N during the climb phase of the bunt. The thrust is then reset to the nominal value (4500 N) during the final 2 km. The differences between the open-loop and closed-loop thrust profiles are attributable to the modeling errors and the inclusion of gusts. The closed-loop speed profile is well matched to the open-loop speed. There is a slight difference over the last 2 km. Because the closed-loop controller has 5-g limits, more incidence is used, with an attendant drag penalty. The closed-loop controller recovers the speed loss using slightly more thrust over the last 2 km; both the open-loop and closed-loop impact speeds are quite similar (within 5 m s^{-1} of one another). The impact speed is approximately 273 m s^{-1} in the closed-loop simulation (an impact speed above 250 m s^{-1} is preferred, and so this value is more than acceptable).

Because the open-loop trajectories are followed closely, the state constraints (minimum height and look-angle constraints) are easily satisfied in the closed-loop simulations and do not need any additional margin. The impact-angle controller is also effective if the control margins are reduced (e.g., reducing the pitch control to 45 m s^{-2} did not affect performance; similarly, the 5-deg margin in roll angle does not appear to be critical). The thrust margin is important, because the speed can vary significantly due to
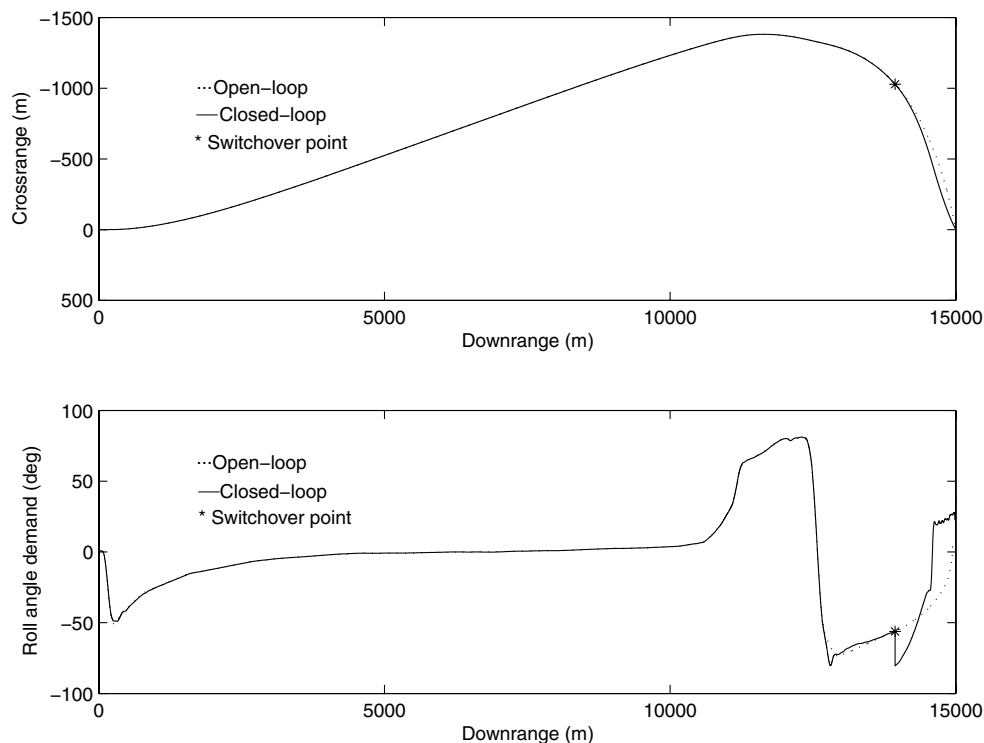

Fig. 5   Open- and closed-loop trajectories and controls in crossrange for case A.
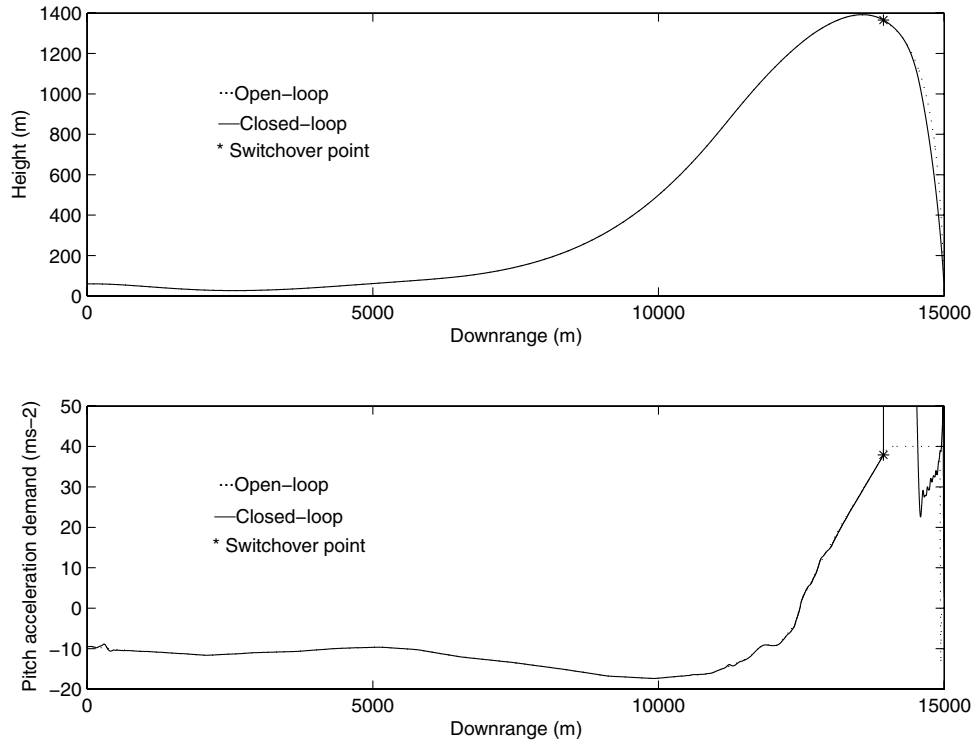
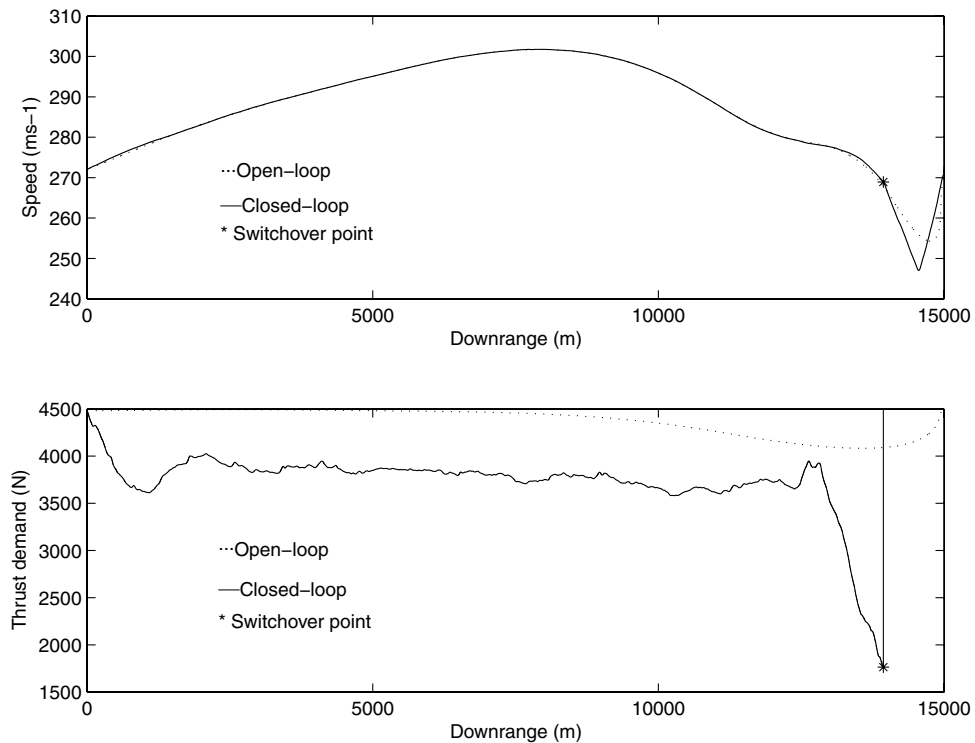Fig. 6 Open- and closed-loop height and control trajectories for case A.

Fig. 7 Open- and closed-loop speed and thrust demands for case A.

aerodynamic uncertainties, and the upper saturation limit of 6000 N is sometimes necessary for the missile to satisfy the temporal requirements.

B. Case B: 1-km Offset

The more difficult example is now presented. In this case, there is a 1-km crossrange offset and a requirement to approach the target from behind (i.e., a negative azimuth impact angle is specified). The

elevation impact angle is 45 deg. The azimuth-impact-angle requirement is −63 deg; it is noted that higher angles could not be achieved with good performance. In this example, the navigation and seeker errors are activated (i.e., perfect state information is not assumed) and the tracking errors will be larger. There are also random initial position errors in all three axes.

The 3-D trajectory is shown in Fig. 9, the crossrange and roll-angle demands are shown in Fig. 10, the height and pitch acceleration demands are shown in Fig. 11, the speed profile and thrust demands
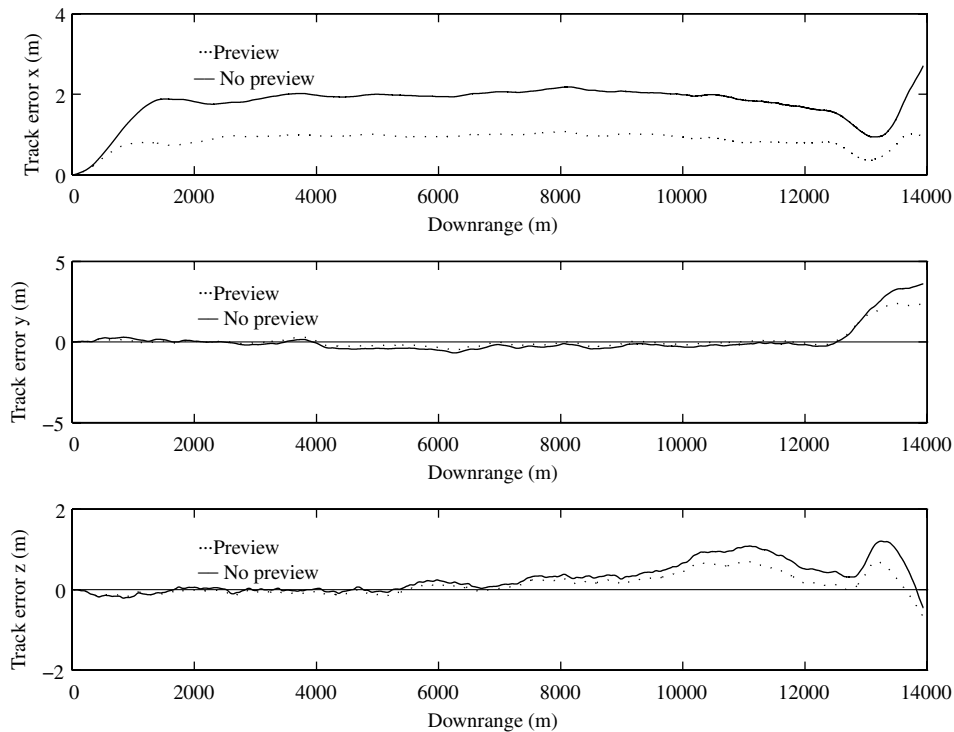
Fig. 8   Position-tracking errors, with and without preview, for case A.
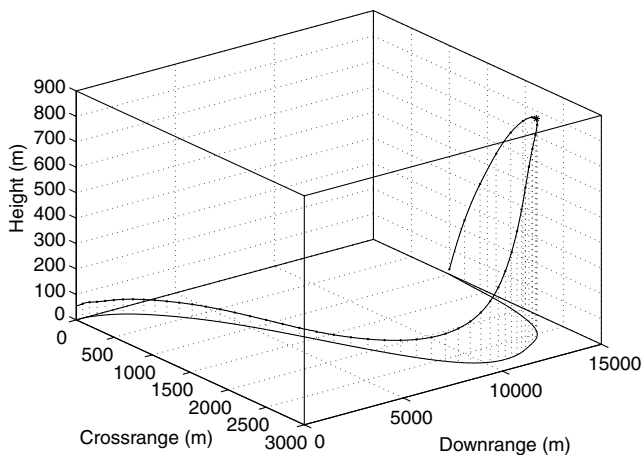

Fig. 9   Three-dimensional trajectory (case B).

are shown in Fig. 12, and the tracking errors with and without preview are shown in Fig. 13.

In this example, the controller achieves impact-angle errors of 0.05 deg in elevation and 2.97 deg in azimuth, with a miss distance of 0.8 m. Terminal incidence is 0.08 deg and the impact speed is approximately 272 m s^{-1}. Because navigation errors are included, the tracking errors exhibit drift in all three axes (the error in crossrange is approximately 10 m at 2-km range-to-go, with errors of approximately 5 m in downrange and height). The controller is still able to achieve good terminal performance. One interesting feature of the trajectory is that the missile has a very shallow heading angle between 10 and 13 km downrange; this is due to the short-range DBS requirement that requires that the angle between the velocity and sightline vectors in azimuth does not become too small (below 10 deg) until very short (less than 1 km) ranges. The impact-angle controller does not directly impose this constraint, but simulations show that the offset angle can be kept within a 10-deg limit via the terminal heading-angle constraint. When switching to the impact-angle controller, the roll angle stays near its upper saturation boundary and matches the open-loop roll demand fairly well. There

is a bank reversal at approximately 14.6 km downrange, which can be attributed to the reduction in the pitch control at the same point. The pitch demand uses maximum control to try to reduce the heading error and deviates significantly from the open-loop control. There is some noise on the pitch demand near intercept as the controller tries to reduce the impact-angle error and the miss distance. The navigation errors are generally much larger in this case, due to the larger accelerations that are characteristic of the trajectory.

C.   Monte Carlo Performance for Test Cases

An initial set of results is computed assuming perfect information (no state estimation or seeker errors); this is principally used for validation purposes and also provides a performance benchmark. The preview term is effective in reducing the mean square and maximum tracking errors averaged over a number of Monte Carlo replications. The results are not shown here, due to space limitations. Instead, we focus on the Monte Carlo results with instrumentation, navigation, and seeker errors included. The navigation errors are important: the trajectory-tracking errors will increase, because the missile will not know its absolute position exactly. Table 3 shows the performance for case A, and Table 4 shows the performance for case B. The statistics are only averaged for successful cases with a miss distance below 10 m. For each performance measure, the mean value is stated first, followed by the standard deviation, which is displayed in parentheses. The trajectory-tracking performance can be quantified by the MSE and the maximum errors in downrange, crossrange, and height; these are only computed during the period that the LQR controller is used. Twenty Monte Carlo simulations are used to evaluate the performance.

The results show that the LQR controller is able to achieve good performance in both cases. In case A, there is one failure that is due to the impact-angle controller failing to achieve a good intercept, due to the modeling uncertainty (rather than the LQR controller performing poorly). The preview controller performs similarly to the LQR controller in case A, with only one failure case. On average, the preview term reduces the tracking errors slightly, and the worst case errors are reduced along with the mean square errors in all three axes. Further simulations with larger navigation errors suggest that the preview can be less effective if the missile is far from the optimal trajectory. The terminal guidance performance is quite similar with
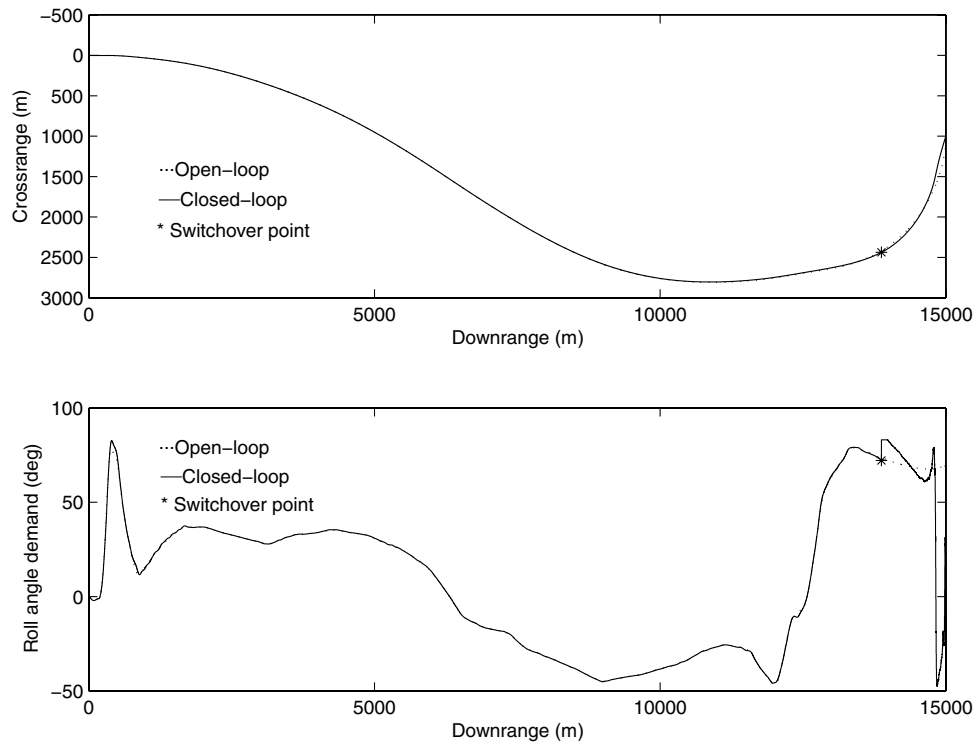
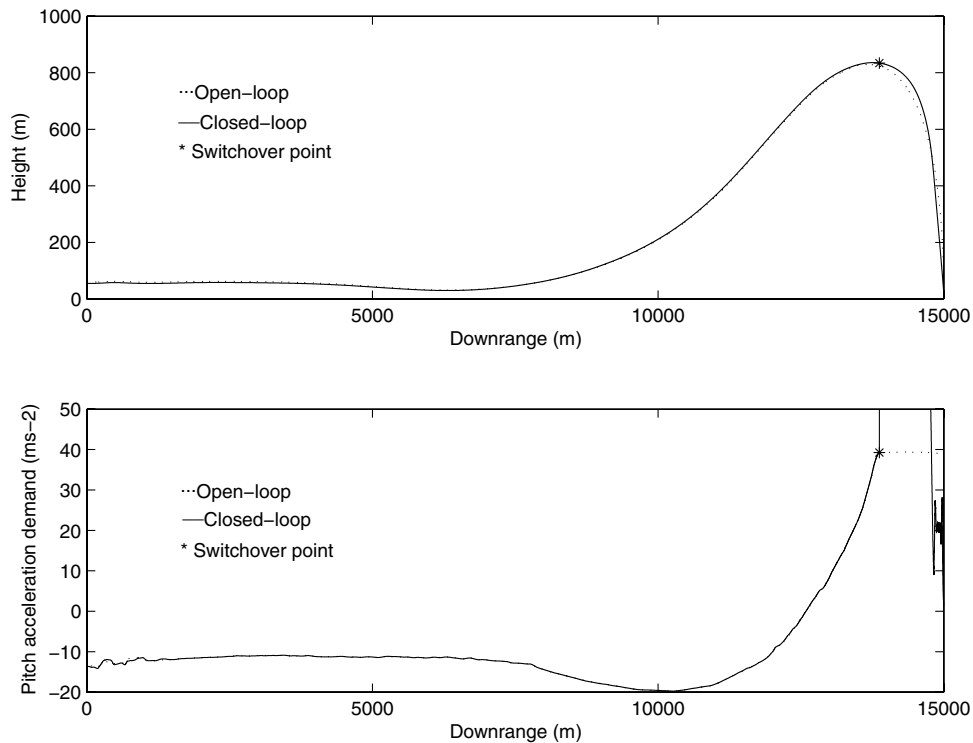Fig. 10   Open- and closed-loop crossrange and roll-angle demands for case B.

Fig. 11   Open- and closed-loop height and pitch acceleration demands for case B.

and without preview; this is because the states are very similar when switching over to the impact-angle guidance law. Low miss distance, impact-angle error, and incidence at impact are achieved for the successful intercepts.

The preview has a negative effect in case B with one more failure. The mean square tracking errors and maximum tracking errors are again reduced using the preview term (e.g., the mean value of the maximum tracking error is reduced by between 0.3 to 0.5 m in

crossrange and height). The terminal guidance performance is excellent in case B, with a mean miss distance below 1 m. Good precision is also achieved with respect to the elevation impact angle and incidence at impact (both have a mean value below 2 deg). The mean impact speed is approximately 260 m s^{-1}, and so the lethality performance should be acceptable.

Further Monte Carlo runs with the modeling errors turned off demonstrate near-perfect performance (there are no failures and all
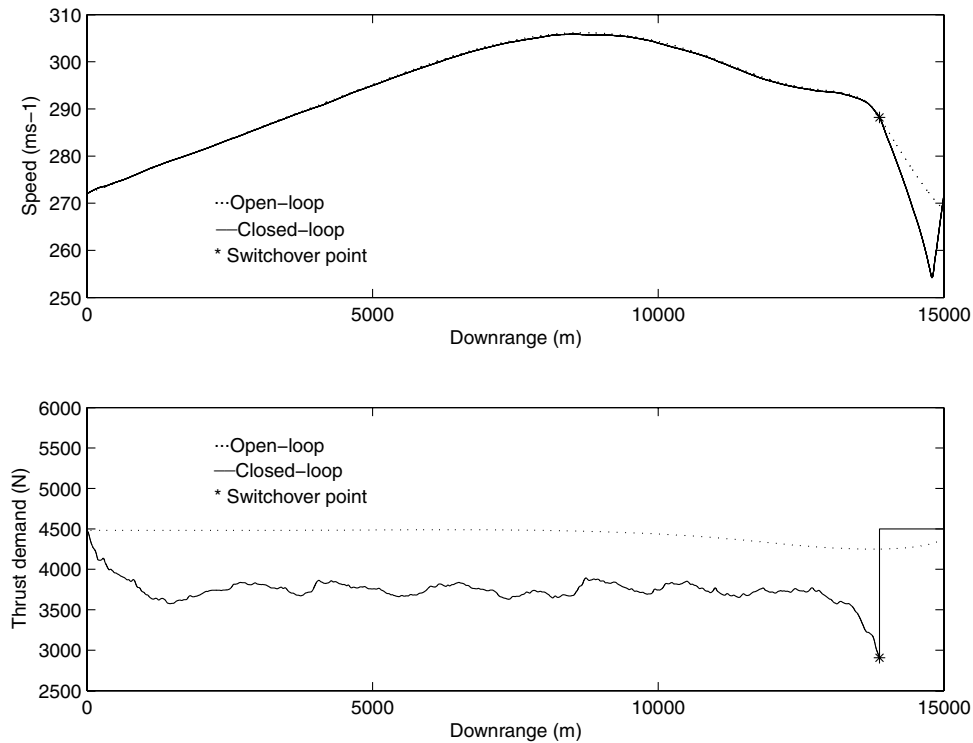
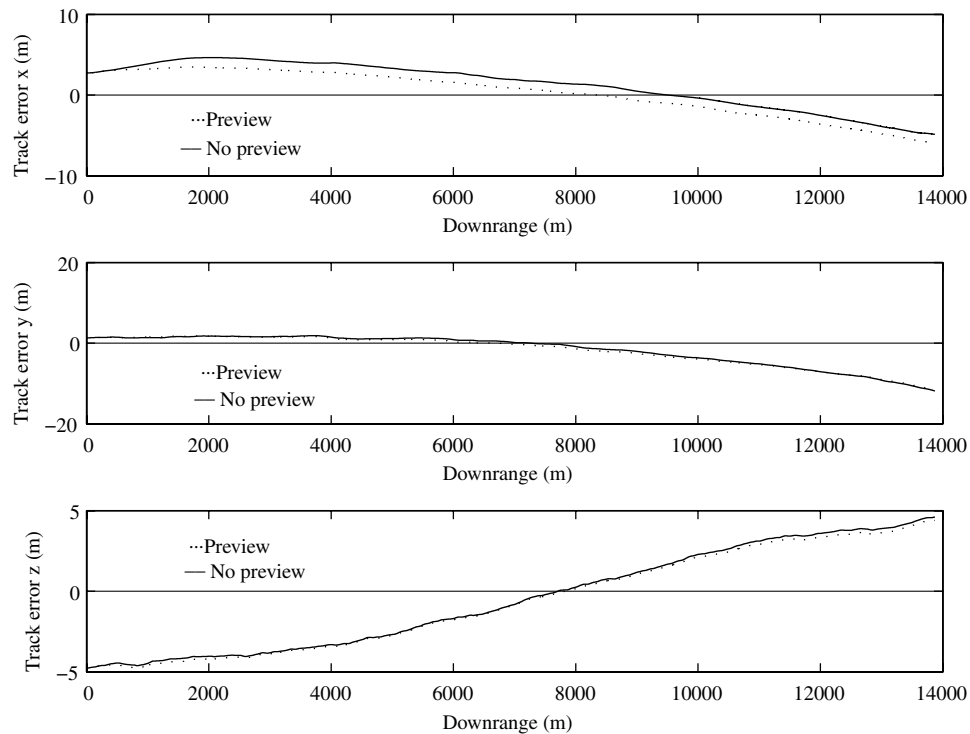Fig. 12   Open- and closed-loop speed and thrust demands for case B.

Fig. 13   Position-tracking errors, with and without preview, for case B.

miss distances are below 1 m). The guidance performance is also affected by the control margins and navigation errors. The latter limits the trajectory-tracking accuracy and could be improved by using additional sensor information.

VII.   Conclusions

A methodology for tracking precomputed optimal terminal guidance trajectories for air-to-surface missiles with radar imaging seekers is presented. The closed-loop guidance problem is demanding, because the optimal trajectories require maneuvering in both planes to satisfy the DBS imaging requirements and the vertical impact-angle constraints. The approach is composed of two stages:

1) Trajectory optimization is performed offline using a detailed model of the missile dynamics; this stage has been described in a previous paper.

2) The optimal trajectory is then tracked online using a linear quadratic regulator.

Table 3 Monte Carlo performance with and without preview term (case A)

Performance output	No preview	Preview
No. of failures	1	1
Miss, m	2.59 (1.04)	2.59 (1.04)
Elevation-impact-angle error, deg	1.69 (0.1)	1.70 (0.09)
Incidence at impact, deg	1.94 (1.25)	1.95 (1.33)
Azimuth impact angle, deg	35.5 (0.93)	35.5 (0.68)
Crossrange resolution (2 km), m	0.44 (0.0027)	0.44 (0.0016)
Max height, m	1397 (1.56)	1397 (1.05)
Min speed, m s^{-1}	234.4 (13.0)	234.3 (12.3)
Impact speed, m s^{-1}	257.1 (14.8)	257.0 (14.3)
Flight time, s	58.0 (0.43)	58.0 (0.41)
Exposure time, s	34.6 (0.43)	34.6 (0.40)
Bunt range, m	8432 (18.4)	8432 (18.6)
Max pitch look angle, deg	43.1 (0.68)	43.0 (0.66)
MSE in x, m	5.92 (0.99)	5.86 (0.54)
MSE in y, m	5.31 (0.53)	5.23 (0.50)
MSE in z, m	2.71 (0.60)	2.64 (0.50)
Max tracking error in x, m	13.76 (2.82)	13.5 (1.7)
Max tracking error in y, m	12.51 (3.42)	11.84 (2.59)
Max tracking error in z, m	6.52 (1.79)	6.03 (1.37)

Table 4 Monte Carlo performance with and without preview term (case B)

Performance output	No preview	Preview
No. of failures	2	3
Miss, m	0.81 (0.19)	0.91 (0.33)
Elevation-impact-angle error, deg	0.14 (0.099)	0.14 (0.096)
Incidence at impact, deg	1.34 (0.83)	1.43 (0.80)
Azimuth impact angle, deg	−66.2 (0.4)	−66.1 (0.73)
Crossrange resolution (2 km), m	0.21 (0.0006)	0.21 (0.0005)
Max height, m	840.7 (3.11)	840.3 (2.50)
Min speed, m s^{-1}	242.8 (17.8)	241.3 (17.33)
Impact speed, m s^{-1}	260.2 (17.0)	258.7 (16.99)
Flight time, s	57.4 (0.42)	57.4 (0.41)
Exposure time, s	25.7 (0.45)	25.7 (0.45)
Bunt range, m	6512 (38.6)	6509 (37.2)
Max pitch look angle, deg	33.7 (0.9)	33.7 (0.9)
MSE in x, m	5.30 (2.24)	5.10 (2.21)
MSE in y, m	5.99 (2.16)	5.99 (2.21)
MSE in z, m	5.25 (1.74)	5.10 (1.62)
Max tracking error in x, m	10.59 (3.47)	10.34 (3.26)
Max tracking error in y, m	12.94 (4.06)	12.64 (3.80)
Max tracking error in z, m	9.7 (3.16)	9.27 (2.42)

The effectiveness of a preview term is investigated. The preview term is based on computing the predicted future trajectory from the current state. The closed-loop controls are composed of the open-loop feedforward optimal control with feedback-correction terms. The methodology is applicable to other optimal control problems in the aerospace field, in which it is necessary to decouple trajectory optimization and trajectory tracking.

The regulator operates for a fixed range-to-go, after which a terminal impact-angle guidance law is activated to intercept the target. The impact-angle guidance law studied is the circular navigation guidance scheme. The combined scheme works well and can cater to uncertainties in target location. In some cases, the impact-angle controller is unable to intercept the target, due to controller saturation. The saturation can be due to the combination of factors, though simulation results suggest that aerodynamic uncertainty is the most important factor. Navigation errors will also reduce the performance of the control scheme (results here suggest that a moderate level of navigation inaccuracy can be tolerated, and the guidance scheme has also performed well with larger navigation errors).

The computational requirements of the standard LQR feedback controller are modest: lookup tables of gains have to be stored onboard the missile, along with the precomputed optimal trajectories and controls. The preview control increases the computational requirements, because the preview gains need to be stored in the flight computer and a model is needed to predict the future path. Over the last 2 km, the regulator is not required, and the terminal guidance law can be implemented using seeker and inertial navigation system data.

The trajectory-tracking controller is tested in the presence of modeling and sensor errors and disturbances. Initial results without navigation errors suggested that the preview term could provide some benefit in reducing the trajectory-tracking errors. In these simulations, the missile stays close to the precomputed optimal trajectory. When instrumentation errors are included, the performance with the preview-control term is mixed. There is a slight increase in the number of failures; however, results showed that the tracking errors can be reduced slightly (averaged over the successful intercept cases) using the preview term. Given these initial results, along with the extra computation required for preview, it seems less encouraging to use a preview term in this context. This is because the future trajectory or preview information is also contained in the open-loop optimal control. As might be anticipated, the inclusion of the preview term in the control law has a marginal effect on the tracking performance. For these reasons, a standard linear quadratic regulator that uses the current-time state information is effective in tracking the optimal trajectories.

Further extensions to the current work include 1) enhancing the simulation models to include more detailed aerodynamics representations (this may require the open-loop optimal controls to be recomputed), 2) integrating into the guidance loop detailed representations of the seeker signal processing including target recognition and aim-point algorithms, and 3) state estimation using the INS and DBS data to improve the navigation. It would also be interesting to investigate if alternative guidance laws are able to achieve the same level of precision with respect to the terminal constraints using a reduced amount of control energy.

Acknowledgment

References

[1] Farooq, A., and Limebeer, D. J. N., "Bank-to-Turn Missile Guidance with Radar Imaging Constraints," *Journal of Guidance, Control, and Dynamics*, Nov.–Dec. 2005, pp. 1157–1170.

[2] Farooq, A., and Limebeer, D. J. N., "Trajectory Optimization for Air-to-Surface Missiles with Imaging Radars," *Journal of Guidance, Control, and Dynamics*, Sept.–Oct. 2002, pp. 876–888.

[3] Hodgson, J. A., and Lee, D. W., "Terminal Guidance Using a Doppler Beam Sharpening Radar," AIAA Guidance, Navigation, and Control Conference, Austin, Texas, AIAA Paper 2003-5796, Aug. 2003.

[4] Breivik, M., and Fossen, T. I., "Principles of Guidance-Based Path Following in 2D-3D," *Proceedings of the 44th IEEE Control and Decision Conference*, Inst. of Electronic Engineers, Piscataway, NJ, 2005, pp. 627–634.

[5] Bryson, A. E., and Ho, Y., *Applied Optimal Control*, Hemisphere, New York, 1975, Chaps. 5–6.

[6] Bryson, A. E., and Jardin, M. R., "Neighboring Optimal Aircraft Guidance in Winds," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 4, 2001, pp. 710–715.

[7] Dukeman, G. A., "Profile-Following Entry Guidance Using Linear Quadratic Regulator Theory," AIAA Guidance, Navigation, and Control Conference, Monterey, CA, AIAA Paper 2002-4457, 2002.

[8] Roenneke, A. J., and Well, K. H., "Linear Optimal Control for Re-Entry Flight," *Computational Optimal Control, International Series of Numerical Mathematics*, edited by R. Bulirsch, and D. Kraft, Birkhaeuser Verlag, Berlin, 1994, pp. 339–348.

[9] Lu, P., "Regulation About Time-Varying Trajectories: Precision Entry Guidance Illustrated," *Journal of Guidance, Control, and Dynamics*, Vol. 22, No. 6, 1999, pp. 784–790.

[10] Ross, I. M., Fahroo, F., and Yan, H., "Real-Time Computation of Neighboring Optimal Control Laws," AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA Paper 2002-4657, 2002.

[11] Jardin, M. R., "Ideal Free Flight Through Multiple Aircraft Neighboring Optimal Control," *Proceedings of the American Control Conference*, American Automatic Control Council, Evanston, IL, 2000, pp. 2879–2885.

[12] Tomizuka, M., and Whitney, D. E., "Optimal Discrete Finite Preview Problems (Why and How is Future Information Important?)," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 97, No. 4, Dec. 1975, pp. 319–325.

[13] Li, D., Zhou, D., Hu, Z., and Hu, H., "Optimal Preview Control Applied to Terrain Following Flight," *Proceedings of the 40th IEEE Conference on Decision and Control*, Inst. of Electronic Engineers, Piscataway, NJ, Dec. 2000, pp. 211–216.

[14] Paulino, N., Silvestre, C., and Cunha, R., "Affine Parameter-Dependent Preview Control for Rotorcraft Terrain Following Flight," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, 2006, pp. 1350–1359.
doi:10.2514/1.19341

[15] Middleton, R. H., Chen, J., and Freudenberg, J. S., "Tracking Sensitivity and Achievable H_∞ Performance in Preview Control," *Automatica*, Vol. 40, No. 8, 2004, pp. 1297–1306.
doi:10.1016/j.automatica.2004.02.019

[16] Devasia, S., and Zou, Q., "Preview-Based Stable-Inversion for Nonlinear Nonminimum-Phase Aircraft Tracking: VTOL Example," AIAA Guidance, Navigation, and Control Conference, Keystone, CO, AIAA Paper 2006-6241.

[17] Roh, S. H., and Park, Y., "Stochastic Optimal Preview Control of an Active Vehicle Suspension," *Journal of Sound and Vibration*, Vol. 220, No. 2, 1999, pp. 313–330.
doi:10.1006/jsvi.1998.1935

[18] Sharp, R. S., and Valtetsiotis, V., "Optimal Preview Car Steering Control," *ICTAM: Selected Papers from the 20th International Congress of Theoretical and Applied Mechanics*, edited by P. Lugner and K. Hedrick, Supplement to Vehicle System Dynamics, Vol. 35, Swets and Zeitlinger, Lisse, The Netherlands, May 2001, pp. 101–117.

[19] Tomizuka, M., "On the Design of Digital Tracking Controllers," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 115, June 1993, pp. 412–418.

[20] Milam, M. B., Mushambi, K., and Murray, R. M., "A New Computation Approach for Real-Time Trajectory Generation for Constrained Mechanical Systems," *Proceedings of the 39th IEEE Conference on Decision and Control*, Vol. 1, Inst. of Electronic Engineers, Piscataway, NJ, 12–15 Nov. 2000, pp. 845–851.

[21] Strizzi, J., Fahroo, F., and Ross, I., "Towards Real-Time Computation of Optimal Controls for Nonlinear Systems," AIAA Guidance, Navigation, and Control Conference, Monterey, CA, AIAA Paper 2002-4945.

[22] Zarchan, P., *Tactical and Strategic Missile Guidance*, AIAA Education Series, AIAA, Reston, VA, 2002, pp. 119–141, 541–569.

[23] Manchester, I. R., and Savkin, A. V., "Circular Navigation Guidance Law for Precision Missile/Target Engagements," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 2, 2006, pp. 314–320.

[24] Kim, B. S., Lee, J. G., and Han, H. S., "Biased PNG Law for Impact with Angular Constraint," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 34, No. 1, 1998, pp. 277–287.
doi:10.1109/7.640285

[25] Ohlmeyer, E. J., and Phillips, C. A., "Generalized Vector Explicit Guidance," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 2, 2006, pp. 261–268.

[26] Lee, J. G., Han, H. S., and Kim, Y. J., "Guidance Performance Analysis of Bank-To-Turn Missiles," *Proceedings of the IEEE International Conference on Control Applications*, Inst. of Electronic Engineers, Piscataway, NJ, 1999, pp. 991–996.

[27] Manchester, I. R., Savkin, A. V., and Faruqi, F. A., "Optical-Flow-Based Precision Missile Guidance Inspired by Honeybee Navigation," *Proceedings of the 42nd IEEE Conference on Decision and Control*, Vol. 5, Inst. of Electronic Engineers, Piscataway, NJ, 2003, pp. 5444–5449.

[28] Manchester, I. R., Savkin, A. V., and Faruqi, F. A., "An Optical-Flow Based Precision Missile Guidance Law," *IEEE Transactions on Aerospace and Electronic Systems* (to be published).

[29] Lu, P., Doman, D. B., and Schierman, J. D., "Adaptive Terminal Guidance for Hypervelocity Impact in Specified Direction," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 2, 2006, pp. 269–278.

[30] Vorley, D., "An Investigation of Optimal Trajectory Shapes for a Vertically Launched Point Defence Missile," MBDA Rept. BT13009, Stevenage, England, U.K., Mar. 1982.

[31] Zipfel, P. H., *Modeling and Simulation of Aerospace Vehicle Dynamics*, AIAA Education Series, AIAA, Reston, VA, 2000, pp. 121–127, 467–470.

[32] Stevens, B. L., and Lewis, F. L., *Aircraft Control and Simulation*, Wiley, New York, 1992, pp. 46–47.

[33] Devaud, E., Harcaut, J., and Siguerdidjane, H., "Three Axes Missile Autopilot Design: From Linear to Nonlinear Control Strategies," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 1, 2001, pp. 64–71.

[34] Betts, J. T., "Survey of Numerical Methods for Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 21, No. 2, 1998, pp. 193–207.

[35] Bauer, T. P., Wood, L. J., and Caughey, T. K., "Gain Indexing Schemes for Low-Thrust Perturbation Guidance," *Journal of Guidance, Control, and Dynamics*, Vol. 6, No. 6, 1983, pp. 518–525.