JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 5, November 2008

Path Planning for Global-Positioning-System-Guided Indirect Fire Weapons

Richard Kenefic*
Raytheon, 1010 Production Rd, Fort Wayne, IN 46808

DOI: 10.2514/1.36888

Global-positioning-system-guided indirect fire weapons can precisely engage high-value targets at long range, and the use of these weapons has been contemplated in complex terrain with obstacles and restricted airspace. Path planning for loitering weapons, such as unmanned aerial vehicles, has been well studied in the recent literature, but no path planning methods for global-positioning-system-guided indirect fire weapons are currently available. This paper presents a path planning method for this case by modifying a framework commonly used for loitering weapons.

Nomenclature

a_x	coefficient of t^3 in the parametric representation of x
a_z	coefficient of t^3 in the parametric representation of z
b_x	coefficient of t^2 in the parametric representation of x
b_z	coefficient of t^2 in the parametric representation of z
c	speed of sound
c_x	coefficient of t in the parametric representation of x
c_z	coefficient of t in the parametric representation of z
$\boldsymbol{D}_b$	axial force from the projectile in the body reference frame
$\boldsymbol{D}_e$	axial force from the elevator in the body reference frame
DTED	digital terrain elevation data
$D_{\dot{\theta}}(m)$	pitch damping moment coefficient
d_x	constant in the parametric representation of x
d_z	constant in the parametric representation of z
F_x	x component of the resultant force acting on the projectile
F_z	z component of the resultant force acting on the projectile
$G(V, E)$	search graph with vertex set V and edge set E
g	acceleration of gravity
I	projectile moment of inertia
K	fixed pure proportional navigation gain constant
L	characteristic length
$\boldsymbol{L}_b$	normal force from the projectile in the body reference frame
$\boldsymbol{L}_e$	normal force from the elevator in the body reference frame
M	projectile mass
m	Mach number

Received 28 January 2008; revision received 11 July 2008; accepted for publication 12 July 2008.
* Principal Systems Engineer, Richard_J_Kenefic@raytheon.com

q_{dyn}	dynamic pressure
S	characteristic area
t_i	time associated with a configuration for a parent vertex
t_f	time associated with a configuration for a child vertex
t_f'	next iterate for t_f in the local planner
$\boldsymbol{v}$	projectile velocity in the inertial frame
w_m	mth downrange waypoint
x	downrange position in the inertial frame
x_{Aim}	downrange aimpoint
$\bar{x}_I$	state of the initial configuration
x_i	downrange target position for the parent vertex
x_f	downrange target position for a child vertex
x_t	downrange target position
z	altitude in the inertial frame
z_i	altitude for a parent vertex
z_f	altitude for a child vertex
α	angle of attack
α_d	desired angle of attack
α_f	angle of attack for a child vertex
α_i	angle of attack for the parent vertex
δ	elevator chord to center of gravity
$\bar{\delta}$	velocity error tolerance
ε	elevation angle
$\bar{\varepsilon}$	position error tolerance
θ	pitch angle
$\lambda(\alpha, \varepsilon)$	PPN gain constant
ρ	air density
τ	autopilot time constant
ϕ	line of sight angle

I. Introduction

NET-enabled weapons promise to bring near real-time command and control to battlefield commanders. In dynamic scenarios, where weapons are directed in near real time, it is necessary to optimize weapon–target pairing from a set of feasible assignments. In a feasible assignment every weapon can prosecute its assigned target, satisfying the kinematics of the weapon and the constraints in the working space. Commanders will have a variety of loitering and indirect fire weapons available. (Indirect fire weapons include artillery, mortars, and rockets that glide into targets. These weapons do not rely on direct viewing of the target through a sight and require command and control systems for fire direction.) It should, however, be noted that while path planning for loitering weapons has been well studied in the recent literature [1–3], this is not true for indirect fire weapons.

Global-Positioning-System (GPS) guided indirect fire weapons are canard controlled [4] and can either use proportional navigation for guidance [5,6] or a trajectory shaping guidance law [7,8] to control the flight path impact angle. Trajectory shaping guidance for indirect fire weapons has traditionally been used near impact [9], not for obstacle avoidance, and there is no guarantee that a near-vertical impact will avoid terrain or restricted airspace.

The path planning models for loitering weapons often use a turn rate limited vehicle in a horizontal plane with obstacles present, a model commonly used for the Dubins car [10]. One approach is sampling based, where a sampling scheme is used to probe the configuration space along with a collision detection module. These approaches input the initial and goal configurations and search until a feasible path is found or report a failure to find such a path. Rapidly exploring dense trees (RDTs) have been used to develop efficient path planners of this type [10]. When differential constraints must be satisfied, a means for connecting two configurations, the local planner, must be determined. The

local planner accepts the start and end configurations and returns the control that will connect them. For a Dubins car, any two configurations in the plane can be connected.

In this paper a path planner for GPS-guided indirect fire weapons is obtained from motion planning under differential constraints (formulation 14.1) in La Valle [10], the same formulation used for the Dubins car. This path planner requires an efficient local planner, which requires the solution of a two-point boundary value problem, and the development of a suitable search tree. These components are developed below, along with some examples of their use in path planning.

II. Equations of Motion

The dynamics of projectiles in flight has been well studied [11,12]. To simplify the equations of motion and facilitate the presentation of the main ideas of the paper, it is assumed that the projectile is spin stabilized and has three degrees of freedom (3DOF).

Consider the body reference frame shown in Fig. 1. The variables δ and ε represent the distance between the elevator chord and center of gravity and the elevator angle respectively. The body normal and axial forces are $\boldsymbol{L}_b$ and $\boldsymbol{D}_b$, respectively where normal is in the positive Z_b direction and axial in the negative X_b direction (lift and drag are typically in the wind reference frame with axial and normal reserved for the body reference frame.). Both $\boldsymbol{L}_b$ and $\boldsymbol{D}_b$ are functions of Mach number m and angle of attack α. The incremental elevator normal and axial forces are $\boldsymbol{L}_e$ and $\boldsymbol{D}_e$ respectively with identical directions and dependencies vis-à-vis the body forces, but with an additional dependence on elevator angle ε.

The earth (inertial) reference frame is shown in Fig. 2. The munition is located at $\boldsymbol{r}$ and the target is at $\boldsymbol{r}_t$ from the munition CG. The munition velocity is $\boldsymbol{v}$, angle of attack is α, and pitch is θ. This is a 3DOF system (x, z, θ). The second-order nonlinear differential equations of motion for this system can be integrated to determine the munition trajectory given the initial conditions and the autopilot and guidance control laws used to stabilize and steer the munition into the target.

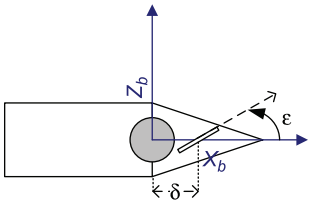

Fig. 1 Munition body reference frame showing CG and elevator.

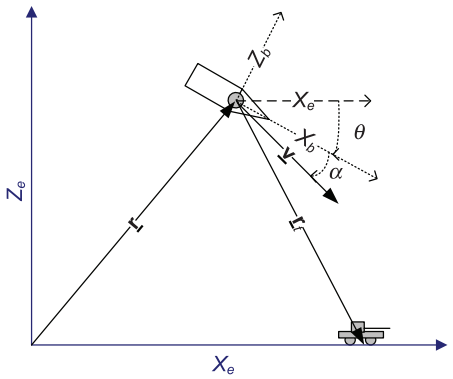

Fig. 2 Inertial reference frame showing the angle of attack and target.

Summing forces and moments in the inertial frame it follows that

$$\begin{aligned} M\ddot{x} &= -D_t(m,\alpha,\varepsilon)\cos(\theta) - L_t(m,\alpha,\varepsilon)\sin(\theta) \\ M\ddot{z} &= -D_t(m,\alpha,\varepsilon)\sin(\theta) + L_t(m,\alpha,\varepsilon)\cos(\theta) - Mg \end{aligned} \tag{1a}$$

$$I\ddot{\theta} = \delta L_e(m,\alpha,\varepsilon) + D_{\dot{\theta}}(m)\dot{\theta} \tag{1b}$$

where M is mass, I is moment of inertia, and $D_t = D_b + D_e$ and $L_t = L_b + L_e$ are the total normal and axial forces, g is acceleration of gravity, and $D_{\dot{\theta}}(m)$ is the pitch damping moment coefficient. Expressing the velocity in the body reference frame leads to

$$\alpha = \text{atan2}(-\dot{x}\sin(\theta) + \dot{z}\cos(\theta), \dot{x}\cos(\theta) + \dot{z}\sin(\theta)) \tag{2}$$

where atan2 is the four quadrant $\tan^{-1}$. The Mach number is

$$m = \sqrt{\dot{x}^2 + \dot{z}^2}/c(z) \tag{3}$$

where c is the speed of sound at altitude z. Normal force, axial force, and pitch moment, $\delta L_e(m,\alpha,\varepsilon)$ are obtained from the corresponding coefficients in the aero database multiplied by the dynamic pressure, $q_{\text{dyn}} = \frac{1}{2}\rho(z)(\dot{x}^2 + \dot{z}^2)$ and S, where ρ is the air density and S is a projected or characteristic area. The pitch damping moment coefficient is $D_{\dot{\theta}}(m) = \frac{1}{2}C_{\text{MQ}}(m)q_{\text{dyn}}SL^2/\sqrt{\dot{x}^2 + \dot{z}^2}$ where L is a characteristic length and $C_{\text{MQ}}(m)$ is obtained from the aero database. The equations of motion (1) to (3) will be complete when the elevator angle control law has been specified.

A seeker head angle can be synthesized from the projectile and target positions to control the acceleration normal to the velocity vector [6]. This is the pure proportional navigation guidance law (PPN) used here. When $\varepsilon = \alpha$ the elevator is aligned with the wind direction in the body frame and no pitching moment is generated. If the line-of-sight angle, ϕ, is increasing, then acceleration normal to $\boldsymbol{v}$ should be generated to decrease it. Such acceleration is obtained after a positive pitching moment has changed the angle of attack to increase or decrease the total lift. From Fig. 2

$$\begin{aligned} \phi &= \text{atan2}(x_t - x, z) \\ \dot{\phi} &= \frac{z\dot{x} + (x_t - x)\dot{z}}{z^2 + (x_t - x)^2} \end{aligned} \tag{4}$$

where x_t is the target position (which could change with time, but is considered stationary here). The PPN control law is then [6]

$$a_N = \lambda(\alpha,\varepsilon)(\dot{\phi})\sqrt{\dot{x}^2 + \dot{z}^2} \tag{5}$$

where a_N is the acceleration to be applied normal to the velocity vector and $\lambda(\alpha,\varepsilon)$ is the PPN gain. As a_N is proportional to the force normal to $\boldsymbol{v}$ the guidance law used is

$$L(m,\alpha,\varepsilon)\cos(\alpha) - D(m,\alpha,\varepsilon)\sin(\alpha) = MK\dot{\phi}\sqrt{\dot{x}^2 + \dot{z}^2} \tag{6}$$

where K is a fixed gain to be determined. The force on the left of Eq. (6) depends strongly on angle of attack and Mach number, and weakly on elevator angle. The control is achieved by adjusting the elevator angle to create a pitching moment to change the angle of attack. Some of the aero data used here for pitching moment are shown in Fig. 3.

Note that the moment is zero for $\alpha = \varepsilon = -10$, $\alpha = \varepsilon = 0$, and $\alpha = \varepsilon = 10$ in Fig. 3. Hence for a first approximation it is assumed that

$$\varepsilon = \min(|\alpha|, 10)\text{sgn}(\alpha) \tag{7}$$

in Eq. (6), so that α can be obtained by interpolation using Eqs. (6) and (7). (The variation is nonlinear, and an improved approximation would use a piecewise linear fit to the aero data in Fig. 3.) The result is the desired angle of attack, which is obtained by adjusting the elevator angle, so once α_d is obtained by interpolation, the desired ε_d is computed from (7). To avoid programming the autopilot loop, the dynamics in Eq. (1b) are omitted by assuming that

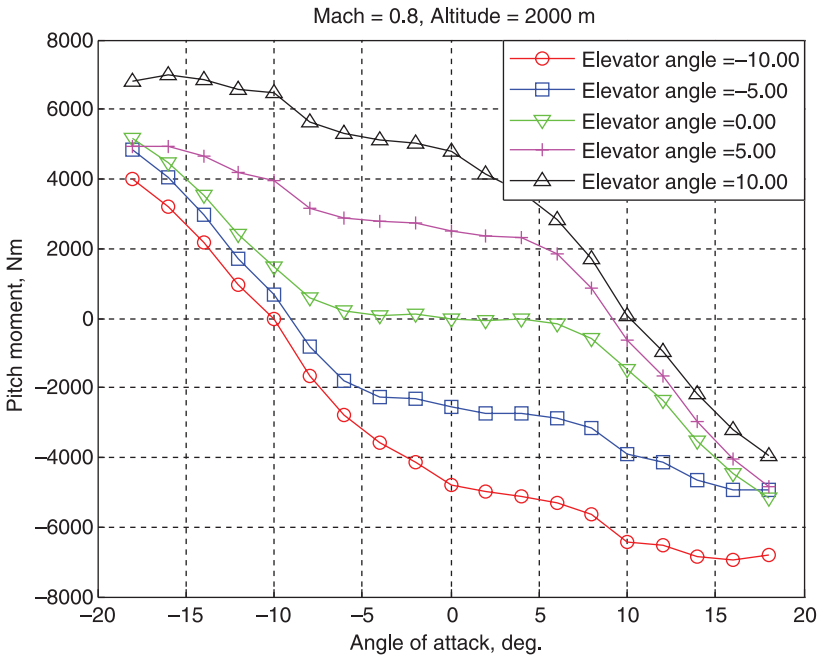

Fig. 3 An example of pitching moment vs angle of attack for various elevator angles.

the autopilot adjusts the angle of attack with a time constant τ. The angle of attack is obtained by low-pass filtering the desired angle to account for the lag in response to an input signal

$$\dot{\alpha} = (\alpha_d - \alpha)/\tau \tag{8}$$

where τ is the time constant. The integration of angle of attack takes place by adding it to the state vector. Note that the equations of motion can be put in the form $\dot{\bar{x}} = f(\bar{x}, x_{\text{Aim}})$ where x_{Aim} is the target position x_t and $\bar{x} = [x \quad z \quad \alpha \quad \dot{x} \quad \dot{z}]'$ is the state. Equations (1) to (8) are used to express $\dot{\bar{x}}$ in terms of the state variables. Note that this computation requires interpolation in the aero database to determine the lift and drag coefficients. In the examples presented below this form is integrated as an initial value problem for comparison with the results obtained using the path planning algorithm.

III. Path Planning

The path planning method used here [13] follows a slight modification of the general framework for single query sampling-based planning algorithms presented by La Valle [10] (The GPS guided munitions considered here are not powered and cannot revisit higher altitudes in the vertical plane; the formulation in [10] does not contain such a restriction).

1) Initialization: Let $G(V, E)$ represent a directed search graph where V contains the initial configuration $\bar{x}_I$ at the initial altitude z_I and E is an initially empty set of edges connecting configurations in V.
2) Vertex selection method (VSM): create a set of vertices at the next lowest altitude corresponding to aimpoints, x_{Aim}, selected from the set $[x_t, w_1, \ldots, w_M]$ where w_m, $m = 1, \ldots, M$ are waypoints along the x-axis and x_t is the location of the target.
3) Local planning method (LPM): solve a two-point boundary value problem to determine the unknown components of the configuration associated with each vertex at the next lowest altitude.
4) Collision detection: determine if the line segment connecting the vertex at the previous altitude with the vertex at the next lowest altitude intersects terrain or restricted airspace.
5) Insert an edge: add the vertex to V and the directed edge connecting this vertex to the one at the previous altitude to E if no collision is detected.

6) Return to step 2: if the next lowest altitude was not zero
7) Check for a solution: return all directed paths that lead from the initial configuration to any point within δ of x_t. Report failure if no such paths exist.

A. Vertex Selection

The initialization vertex creates $M+1$ child vertices at the next lowest altitude, one for each waypoint and one for the target position. Subsequently, vertices are created at the next lowest altitude by considering each vertex at the previous altitude and creating child vertices at the next lowest altitude. The rule used here is to create only one child using $x_{\text{Aim}} = x_t$ if the parent used $x_{\text{Aim}} = x_t$, and create two children, one using $x_{\text{Aim}} = x_t$ and another using $x_{\text{Aim}} = w_m$, if the parent used $x_{\text{Aim}} = w_m$. This is a single waypoint strategy to determine the altitude at which the aimpoint should switch to the target, and it creates a dense search tree (shown below) where the number of vertices is linear in the depth of the tree.

B. Local Planning

The equations of motion for the three degree of freedom model can be written as

$$\begin{aligned} \ddot{x} &= F_x(z, \dot{x}, \dot{z}, \alpha)/M \\ \ddot{z} &= F_z(z, \dot{x}, \dot{z}, \alpha)/M - g \end{aligned} \tag{9}$$

where the pitch, autopilot, and guidance loop dynamics are approximated by

$$\dot{\alpha} = (\alpha_d(x, z, \dot{x}, \dot{z}, x_{\text{Aim}}) - \alpha)/\tau \tag{10}$$

and where τ is the guidance loop time constant, and x_{Aim} the aimpoint. The forces F_x and F_z depend upon angle of attack, elevator angle, dynamic pressure, lift and drag coefficients, and pitch angle. The pitch angle is computed from the velocity vector and angle of attack. The elevator angle is set to the angle that yields zero pitching moment, assuming a fast autopilot loop, and then the lift and drag coefficients are obtained by a table look up on angle of attack, elevator angle, and Mach number. The desired angle of attack, α_d, is obtained by using PPN to compute the desired force normal to the velocity vector from the rate of the line of sight angle and then interpolating to determine the desired angle of attack.

Given the final altitude z_f, Eqs. (9) and (10) can be solved as a two point boundary value problem (TPBVP) by using a shooting method to find the final time that yields the desired final altitude. To efficiently solve this TPBVP, the downrange and altitude coordinates are given parametrically in the form

$$\begin{aligned} x(t) &= a_x t^3 + b_x t^2 + c_x t + d_x \\ z(t) &= a_z t^3 + b_z t^2 + c_z t + d_z \end{aligned} \tag{11}$$

Note that there are five initial conditions in the initial configuration, four of which apply to x and z in Eq. (11). From Eqs. (9) and (11) it follows that

$$\begin{aligned} 6a_x t_i + 2b_x &= F_x(z_i, \dot{x}_i, \dot{z}_i, \alpha_i)/M \\ 6a_z t_i + 2b_z &= F_z(z_i, \dot{x}_i, \dot{z}_i, \alpha_i)/M - g \end{aligned} \tag{12}$$

which yields two more relations from the initial configuration. Two more relations can be obtained from Eqs. (9), (10), and (11) as

$$\begin{aligned} 6a_x t_f + 2b_x &= F_x(z_f, \dot{x}_f, \dot{z}_f, \alpha_f) \\ 6a_z t_f + 2b_z &= F_z(z_f, \dot{x}_f, \dot{z}_f, \alpha_f)/M - g \\ \alpha_f &= \alpha_i e^{-\frac{t_f - t_i}{\tau}} + \alpha_d(x_i, z_i, \dot{x}_i, \dot{z}_i, x_{\text{Aim}})\{1 - e^{-(t_f - t_i)/\tau}\} \end{aligned} \tag{13}$$

by assuming that α_d is constant over the interval (t_i, t_f). Testing revealed that it was necessary to use the arithmetic average of the desired force computed at the start and end of each altitude step in the computation of α_d. In Eq. (13),

z_f is known and t_f, $\dot{x}_f$, and $\dot{z}_f$ are not known. If those quantities were known, it would be possible to solve a linear two by two system for a_x and b_x and also for a_z and b_z using Eqs. (12) and (13). The remaining unknowns, c_x, d_x, c_z, and d_z, can then be obtained easily from the initial configuration using back substitution. The TPBVP is solved using the following iterative procedure:

1) Given $(x_i, z_i, \dot{x}_i, \dot{z}_i, \alpha_i)$ at t_i, set: $\dot{x}_f \leftarrow \dot{x}_i$, $\dot{z}_f \leftarrow \dot{z}_i$, $t_f \leftarrow t_i + \frac{z_f - z_i}{\dot{z}_f}$, $x_f \leftarrow x_i + (t_f - t_i)\dot{x}_f$.
2) Compute α_f, F_x and F_z at t_f for the linear system coefficients.
3) Solve the linear system for a_x, b_x, c_x, d_x, a_z, b_z, c_z, d_z.
4) Select t'_f from the roots of $a_z t^3 + b_z t^2 + c_z t + d_z = z_f$.
5) $\dot{x}'_f \leftarrow 3a_x t'^2_f + 2b_x t'_f + c_x$, $\dot{z}'_f \leftarrow 3a_z t'^2_f + 2b_z t'_f + c_z$, $x'_f \leftarrow a_x t'^3_f + b_x t'^2_f + c_x t'_f + d_x$.
6) If $(|t_f - t'_f| > \bar{\eta})$ OR $(|x_f - x'_f| > \bar{\varepsilon})$ OR $(|\dot{x}_f - \dot{x}'_f| > \bar{\delta})$ OR $(|\dot{z}_f - \dot{z}'_f| > \bar{\delta})$.
   a) $\dot{x}_f \leftarrow \dot{x}'_f$, $\dot{z}_f \leftarrow \dot{z}'_f$, $t_f \leftarrow t'_f$, $x_f \leftarrow x'_f$.
   b) go to 2.
7) Return the final configuration, $(x_f, z_f, \dot{x}_f, \dot{z}_f, \alpha_f)$, at t_f.

The proposed procedure makes use of the method of iteration and tabulated aerodynamic data. Convergence is not guaranteed. It was found that monitoring the 2-norm of the error between the coefficient vectors at each iterative step would indicate when the iteration had entered a limit cycle (norm is constant) or was diverging (norm increases) to facilitate an early exit. In testing, the TPBVP solver has been fast and accurate enough to perform path planning for indirect fire GPS guided weapons.

C. Collision Detection

To detect collisions with terrain it is assumed that level 1 digital terrain elevation data (DTED) is available to the planning algorithm. By triangulating the terrain, collisions can be detected by looking for intersections between the line connecting a parent and child vertex and the set of adjacent triangles that lie along the ground trace of this line segment. Methods for quickly finding these triangles, and the method for detecting an intersection, have both been well studied in the literature [14,15]. These methods also apply to representations of restricted airspace. Collision detection is a straightforward addition to the algorithm presented here.

IV. Test Cases

The examples treated here make use of aero data for a typical GPS guided projectile. The data used were 3DOF, and so the path planning algorithm treated here is restricted to 2D. Also, because the projectile is not very maneuverable, the path planner was restricted to a single waypoint strategy.

For all cases considered here the initial configuration is at an altitude of 2000 m, Mach 0.8, pitch 45 degrees, and angle of attack zero. The guidance loop uses PPN with $K = 6$ and $\tau = 0.3$ s. The search tree was generated for 16 levels, from 2-km to the ground in 125 m steps using 11 waypoints uniformly spaced on [300, 5300]. The target is located 3000 or 1000 m downrange from the initial position.

A. Case One: 3000 m Downrange

Consider the search tree shown in Fig. 4. In practice those branches that impact terrain or violate airspace will be terminated at the altitude where the violation occurs by adding the collision detection algorithm discussed above. To simplify the illustration of the method, all such obstacles have been omitted. The example shown below demonstrates that the proposed algorithm can find a trajectory that clears by 50 m an obstacle at 2500 m downrange with a height of 300 m.

The aimpoints vs altitude are shown in Fig. 5. For the upper bounding trajectory the waypoint is at 5300 m downrange to an altitude of 875 m, then the aim-point is 3000 m to impact. For the lower bounding trajectory the waypoint is at 300 m downrange to an altitude of 1750 m, then the aim-point is 3000 m to impact.

The waypoints are validated in Fig. 6 by integrating the governing differential equations (using the waypoints) up to impact. Note that the bounding trajectories indicate that only a limited amount of trajectory shaping can be accomplished. Nevertheless, an obstacle at 2500 m downrange with a height of 300 m can be cleared by 50 m using the upper bounding trajectory shown in Fig. 6.

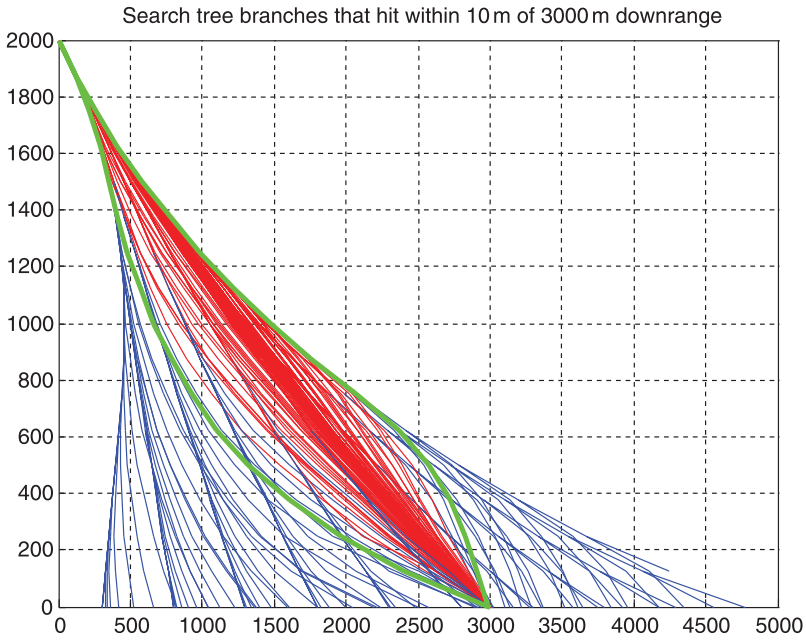

Fig. 4 One waypoint search tree for a target at 3000 m downrange. Valid paths in red. Bounding paths (extremes of arrival angle) in green.

B. Case Two: 1000 m Downrange

Consider the search tree shown in Fig. 7. In practice those branches that impact terrain or violate airspace will be terminated at the altitude where the violation occurs by adding the collision detection algorithm discussed above. To simplify the illustration of the method, all such obstacles have been omitted. The example shown below demonstrates that the proposed algorithm can find a trajectory that clears by 350 m an obstacle at 600 m downrange with a height of 1000 m.

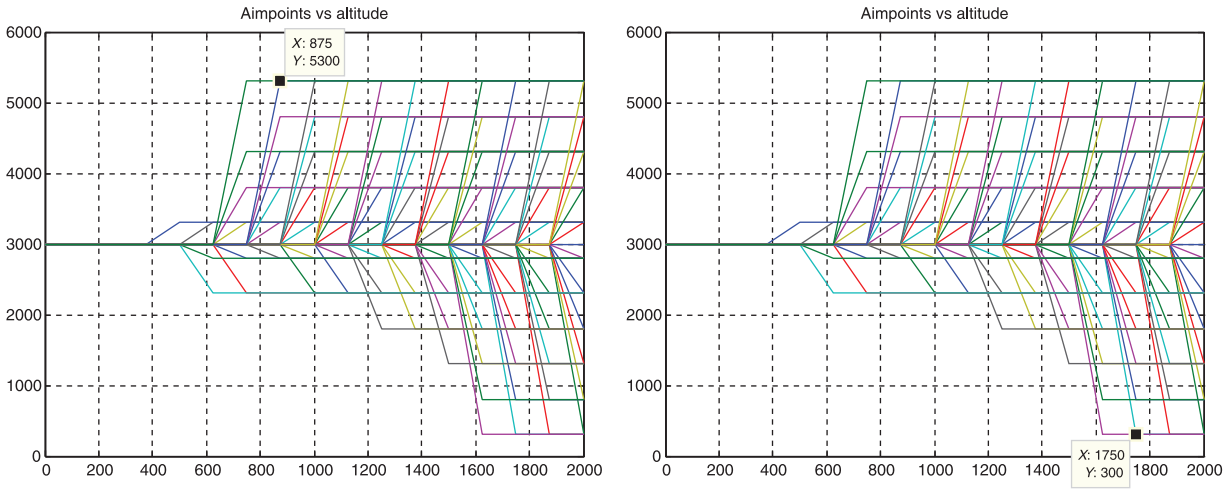

Fig. 5 Aimpoints for the bounding trajectories that satisfy the miss criteria in Fig. 4.

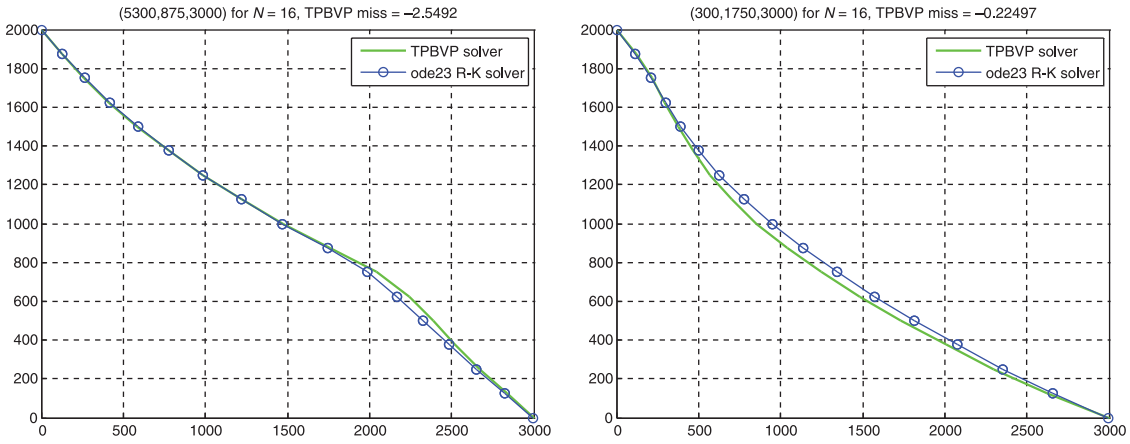

Fig. 6 Validation of the waypoints shown in Fig. 5. Trajectory errors are small enough for obstacle avoidance.

The aimpoints vs altitude are shown in Fig. 8. For the upper bounding trajectory the waypoint is at 2300 m downrange to an altitude of 1625 m, then the aimpoint is 1000 m to impact. For the lower bounding trajectory the waypoint is at 300 m downrange to an altitude of 1375 m, then the aimpoint is 1000 m to impact.

The waypoints are validated in Fig. 9 by integrating the governing differential equations (using the waypoints) up to impact. Note that the bounding trajectories indicate that more trajectory shaping can be accomplished than for the previous case. An obstacle at 600 m downrange with a height of 1000 m can be cleared by 350 m using the upper bounding trajectory shown in Fig. 9.

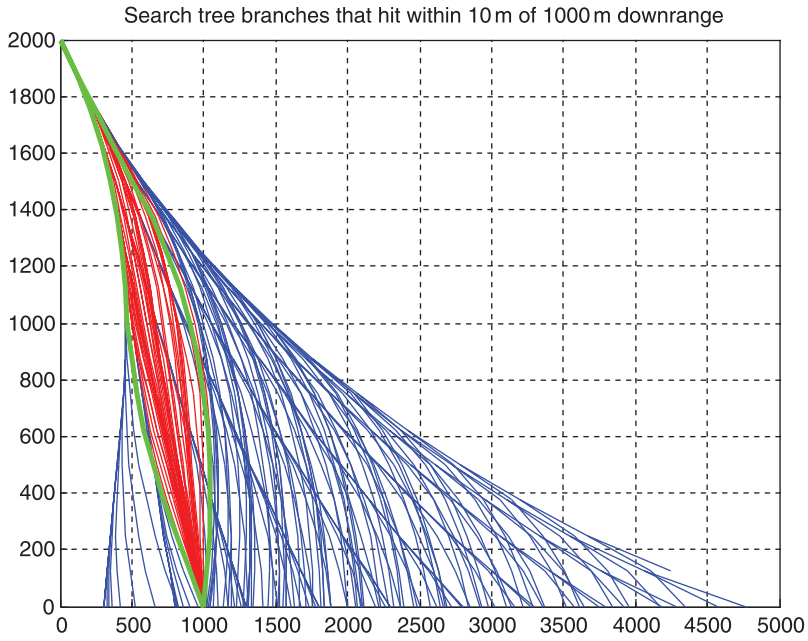

Fig. 7 One waypoint search tree for a target at 1000 m downrange. Valid paths in red. Bounding paths (extremes of arrival angle) in green.

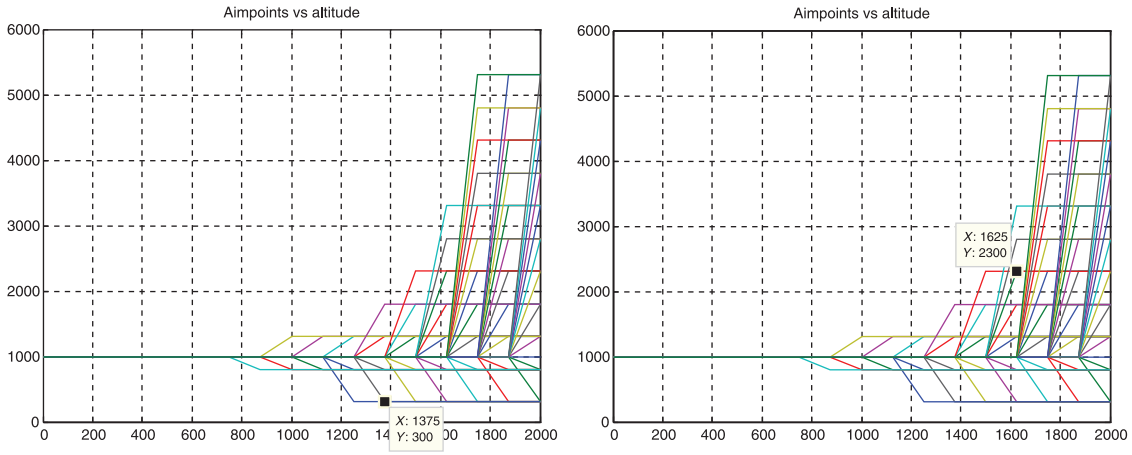

Fig. 8 Aimpoints for the bounding trajectories that satisfy the miss criteria in Fig. 7.

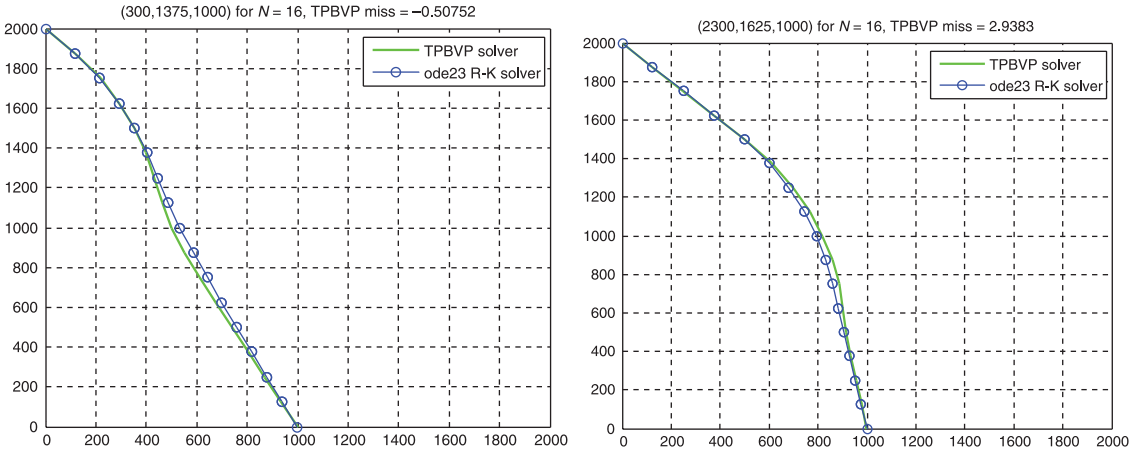

Fig. 9 Validation of the waypoints shown in Fig. 8. Trajectory errors are small enough for obstacle avoidance.

V. Conclusions

A path planning method applicable to GPS-guided indirect fire weapons was presented that is a modification of an existing framework used for single-query sampling based planning. The method uses a dense search tree where the number of vertices is linear in the depth of the tree, and a fast local search method. The example presented here is restricted to PPN, 2D, and a single waypoint vertex selection, but is readily extended to any guidance method, 3D, and a multi-waypoint vertex selection. In testing, the path planner has been accurate enough to find paths that will avoid terrain or restricted airspace. Once the feasible set of paths has been identified, a secondary criterion can be used, such as impact angle or arrival time.

References

[1] Pongpunwattana, A., and Rysdyk, R., "Real-Time Planning for Multiple Autonomous Vehicles in Dynamic Uncertain Environments," *AIAA Journal of Aerospace Computing, Information, and Communication*, Vol. 1, Dec. 2004, pp. 580–604. doi: 10.2514/1.12919

[2] Adams, M. B., Lepanto, J. A., and Hanson, M. L., "Mixed Initiative Command and Control of Autonomous Air Vehicles," *AIAA Journal of Aerospace Computing, Information, and Communication*, Vol. 2, Feb. 2005, pp. 125–153. doi: 10.2514/1.12963

[3] Chang, W.-Y., Hsiao, F.-B., and Sheu, D., "Two-Point Flight Path Planning Using a Fast Graph-Search Algorithm," *AIAA Journal of Aerospace Computing, Information, and Communication*, Vol. 3, Sept. 2006, pp. 453–470.
doi: 10.2514/1.25400

[4] Su, W.-J., Wilson, C., Farina, T., and Trohanowsky, R., "Aerodynamic Characterization of a Canard Guided Artillery Projectile," *45th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, NV, 2007, AIAA-2007-672.

[5] Axelband, E. I., and Hardy, F. W., "Quasi-Optimum Proportional Navigation," *IEEE Transactions on Automatic Control*, Vol. AC-15, Dec. 1970, pp. 620–626.

[6] Yang, C.-D., and Yang, C.-C., "A Unified Approach to Proportional Navigation," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 33, Apr. 1997, pp. 557–567.
doi: 10.1109/7.575895

[7] Pamadi, K. B., and Ohlmeyer, E. J., "Evaluation of Two Guidance Laws for Controlling the Impact Flight Path Angle of a Naval Gun Launched Spinning Projectile," *AIAA Guidance Navigation and Control Conference*, AIAA, Reston, VA, 2006.

[8] Jeong, S. K., Cho, S. J., and Kim, E. G., "Angle Constraint Biased PNG," *5th Asian Control Conference*, Melbourne, Australia, IEEE, 2004, pp. 1849–1854.

[9] Zarchan, P., *Tactical and Strategic Missile Guidance, AIAA Tactical Missile Series*, AIAA, Reston, VA, 2002.

[10] LaValle, S. M., *Planning Algorithms*, Cambridge Univ. Press, New York/London/Cambridge, England, UK, 2006 (available online http://msl.cs.uiuc.edu/∼lavalle/)

[11] McCoy, R. L., *Modern Exterior Ballistics*, Schiffer Publishing, Atglen, PA, 1999.

[12] Hainz III, L. C., and Costello, M., "Modified Projectile Linear Theory for Rapid Trajectory Prediction," *AIAA Journal of Guidance Control and Dynamics*, Vol. 28, Sept.–Oct. 2005, pp. 1006–1014.
doi: 10.2514/1.8027

[13] Kenefic, R., "Method and Apparatus for Path Planning for Guided Munitions," US Patent Application, Serial Number 11/758,340.

[14] Goodman, J. E., and O'Rourke, J., (eds), *Handbook of Discrete and Computational Geometry*, Chapman & Hall/CRC Press, Boca Raton, FL, 2004.

[15] O'Rourke, J., *Computational Geometry in C*, Cambridge Univ. Press, New York/London/Cambridge, England, UK, 1998.

Kelly Cohen
Associate Editor