# AFRL-VA-WP-TP-2003-303

# LOW-OBSERVABLE NONLINEAR TRAJECTORY GENERATION FOR UNMANNED AIR VEHICLES

K. Misovec
T. Inanc
J. Wohletz
R. Murray

**MARCH 2003**

**Approved for public release; distribution is unlimited.**



**This work, resulting from Department of Air Force contract number F33615-01-C-3149, has been submitted for publication in the Proceedings of the 2003 IEEE Conference on Decision and Control. If published, IEEE may assert copyright. If so, the United States has for itself and others acting on its behalf an unlimited, nonexclusive, irrevocable, paid-up royalty-free worldwide license to use for its purposes.**

**AIR VEHICLES DIRECTORATE**
**AIR FORCE RESEARCH LABORATORY**
**AIR FORCE MATERIEL COMMAND**
**WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7542**

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*



| 1. REPORT DATE *(DD-MM-YY)* | 2. REPORT TYPE | 3. DATES COVERED *(From - To)* |
|---|---|---|
| March 2003 | Journal Article Preprint | |

**4. TITLE AND SUBTITLE**
LOW-OBSERVABLE NONLINEAR TRAJECTORY GENERATION FOR UNMANNED AIR VEHICLES

**5a. CONTRACT NUMBER**
F33615-01-C-3149

**5b. GRANT NUMBER**

**5c. PROGRAM ELEMENT NUMBER**
69199F

**6. AUTHOR(S)**
K. Misovec
T. Inanc
J. Wohletz
R. Murray

**5d. PROJECT NUMBER**
ARPF

**5e. TASK NUMBER**
04

**5f. WORK UNIT NUMBER**
T1

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
ALPHATECH, Inc.
50 Mall Road
Burlington, MA 01803

**8. PERFORMING ORGANIZATION REPORT NUMBER**

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Air Vehicles Directorate
Air Force Research Laboratory
Air Force Materiel Command
Wright-Patterson Air Force Base, OH 45433-7542

**10. SPONSORING/MONITORING AGENCY ACRONYM(S)**
AFRL/VACA

**11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)**
AFRL-VA-WP-TP-2003-303

**12. DISTRIBUTION/AVAILABILITY STATEMENT**
Approved for public release; distribution is unlimited.

**13. SUPPLEMENTARY NOTES**
To be presented at the Conference on Decision and Control, Maui, HI, 9-12 Dec 03.



**14. ABSTRACT** ***(Maximum 200 Words)***
This paper explores low observablity flight path planning of unmanned air vehicles in the presence of radar detection systems. The probability of detection model of an aircraft near enemy radar depends on aircraft attitude, range, and configuration. A detection model is coupled with a simplified aircraft dynamics model. The Nonlinear Trajectory Generation (NTG) software package developed at Caltech is used. NTG algorithm is a gradient descent optimization method that combines three technologies: B-splines, output space collocation and nonlinear optimization tools. Implementations are formulated with temporal constraints that allow periods of high observablity interspersed with periods of low observablity. Illustrative examples of optimized routes for low observablity are presented.

**15. SUBJECT TERMS**
Trajectory Generation, Path Planning, Low-Observable, UAV, RADAR, Probability of Detection

| 16. SECURITY CLASSIFICATION OF: | | | 17. LIMITATION OF ABSTRACT: | 18. NUMBER OF PAGES | 19a. NAME OF RESPONSIBLE PERSON (Monitor) |
|---|---|---|---|---|---|
| a. REPORT | b. ABSTRACT | c. THIS PAGE | | | Mark Mears |
| Unclassified | Unclassified | Unclassified | SAR | 28 | 19b. TELEPHONE NUMBER *(Include Area Code)* (937) 255-8685 |

**Standard Form 298 (Rev. 8-98)**
Prescribed by ANSI Std. Z39-18

# Low-Observable Nonlinear Trajectory Generation for Unmanned Air Vehicles

K. Misovec, T. Inanc, J. Wohletz, R. Murray

Extended Draft Version
to the
2003 Conference on Control and Decision

**Abstract**

This paper explores low observability flight path planning of unmanned air vehicles in the presence of radar detection systems. The probability of detection model of an aircraft near an enemy radar depends on aircraft attitude, range, and configuration. A detection model is coupled with a simplified aircraft dynamics model. The Nonlinear Trajectory Generation (NTG) software package developed at Caltech is used. NTG algorithm is a gradient descent optimization method that combines three technologies: B–splines, output space collocation and nonlinear optimization tools. Implementations are formulated with temporal constraints that allow periods of high observability interspersed with periods of low observability. Illustrative examples of optimized routes for low observability are presented.

**Keywords:** Trajectory Generation, Path Planning, Low-Observable, UAV, RADAR, Probability of Detection

## 1 Introduction

As the development of new unmanned vehicles progresses, research focus on coordinating large teams of these vehicles has highlighted many new challenges for control. Due to complexity, the approach taken in the DARPA MICA program is to decompose the control design in a hierarchical manner. The cooperative path planning layer routes the vehicles. At this level of the controller, the design goals are to command the vehicles to pursue dynamically feasible routes, such as speed and turning rate limits, while avoiding threats and collisions with other vehicles. Driven by battlespace management needs, real-world threats can be much more complex than those modeled in prior trajectory generation work. The detectability of an aircraft traveling near an enemy radar depends on more than just the distance to the radar; it depends on the aircraft attitude and configuration as well. This feature in the threat model introduces path dependencies as well as sharp gradients into the underlying optimization problems, and it presents new challenges for trajectory generation techniques. In this paper, we investigate the use of the nonlinear trajectory generation (NTG) method as a solution to the low-observability path planning challenge.

For the low-observability problem, the use of NTG is motivated by recent extensions in its ability to deal with temporal constraints. In [5], NTG is used for a missile intercept problem. In [4], NTG is extended to a multi-vehicle problem with precedence constraints such as *look after strike* and *simultaneous strike*. The low-observability routing problem could be considered to be temporal in nature, by allowing periods of high observability interspersed with periods of low observability. This is desirable because of the way the enemy fire-control systems work. Although it might not be possible to get close to the enemy targets while maintaining low-observability at all times; by strategically flying low-observable paths for part of the time, it may be possible to drive the enemy systems into a condition called *lock-loss*. This condition aborts the enemy fire control plans after a specified time of no detection.

Computational efficiency as well as the capability to enforce more realistic constraints, are two additional motivations that have prompted the choice for NTG in past problems. The method

combines three technologies- differential flatness, splines and nonlinear optimization and has been extensively investigated [6, 7, 3] . Differentially flat systems have the property that the input variables and states can be written in terms of the output variables and their derivatives. This can aid in computationally efficiency by eliminating the need to integrate as in shooting methods [1]. The use of splines also addresses computational issues by allowing complicated functions to be written with low order polynomials that are active over distinctive intervals.The NPSOL [8] package, which uses sequential quadratic programming, is used in this work as a nonlinear optimization tool.

In this paper, we investigate the use of NTG for low-observability trajectory generation for unmanned air vehicles. The aircraft model and detection models are described in the second section. The third section presents an NTG approach formulated for low observability. The fourth section contains examples. The examples demonstrate the ability of NTG to converge to solutions which constrain observability to acceptable levels; however, they also highlight some difficulties with the approach.

## 2 Model

This section presents the model. These models are based on those developed as part of the DARPA MICA program by [2]. In this program, as the problems become progressively more challenging, the models become more refined to capture more realistic features. For the purposes of this report, the models used here are simplified.

The two main components are the aircraft and the detection models, as shown in figure 2. The aircraft model shown is a simple point mass model. Throughout this paper, we assume that the aircraft maintains a fixed altitude and the radar is on the ground. The inputs to the model are a series of waypoints as well as the average speeds required to travel between waypoints. Changes in speed and heading occur instantaneously at the waypoints. The output of the aircraft model is the aircraft position and attitude in inertial coordinates (East, North, Up). Figure 1 illustrates an aircraft traveling between waypoints. The equations below represent the state equations for the vehicle traveling from waypoint $i$ to waypoint $j$.

$$\dot{x}_{ac} = U_{ij} sin(\psi_{ij}) \tag{1}$$

$$\dot{y}_{ac} = U_{ij} cos(\psi_{ij}) \tag{2}$$

$$\tag{3}$$

where $U_{ij}$ is the speed and $\psi$ is the heading, the angle between the nose and north. Equivalently, assuming that the velocity vector is aligned with the nose,

$$\psi = tan^{-1} \frac{\dot{x}_{ac}}{\dot{y}_{ac}} \tag{4}$$

The aircraft position $(x_{ac}, y_{ac}, z_{ac})$ is combined with the radar position $(x_R, y_R, z_R)$ to form a vector from the aircraft to the radar.

$$R = (x_R - x_{ac}) x_{east} + (y_R - y_{ac}) y_{north} + (z_R - z_{ac}) z_{up} \tag{5}$$

This vector is then transformed to body axes. Assuming zero bank and pitch angle, for this model, the transformation is

$$R_{body} = \begin{bmatrix} x_{Rac} \\ y_{Rac} \\ z_{Rac} \end{bmatrix} = \begin{bmatrix} cos(\psi) & sin(\psi) & 0 \\ -sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} (x_R - x_{ac}) \\ (y_R - y_{ac}) \\ (z_R - z_{ac}) \end{bmatrix} \tag{6}$$

The inputs to the detection model are the azimuth, elevation and slant range $(az, el, R_s)$. To obtain these values the vector $R_{body}$ is transformed to spherical coordinates as follows

Figure 1: The aircraft model is a point mass model.

$$az = tan^{-1}\left(\frac{y_{Rac}}{x_{Rac}}\right) \tag{7}$$

$$el = tan^{-1}\left(\frac{z_{Rac}}{R_g}\right) \tag{8}$$

$$R_s = \sqrt{x_{Rac}^2 + y_{Rac}^2 + z_{Rac}^2} \tag{9}$$

where $R_g = \sqrt{x_{Rac}^2 + y_{Rac}^2}$ is the ground range. Note that since heading is a function of velocities as in equation 4, then these equations can be rewritten as functions of the position and velocity $x, \dot{x}, y, \dot{y}$

For the detection model, two tables are used. these tables are based on the small uav and Long SAM radar model parameters in [2]The first table computes the radar signature given the azimuth and elevation in degrees. The signature is a unitless, intermediary variable that is related to radar cross section. Note for azimuths within +/- 30 degrees ("nose-in" flight), the signature values are lower than for azimuths outside this range ("nose-out" flight). The second table relates the probability of detection with the signature and the slant range.

Table 1. Signature Values Respect to Azimuth and Elevation

| el /az | 0.0 | +/-30 | +/-31 | +/-180 |
|---|---|---|---|---|
| 0.0 | 1.5 | 1.5 | 5.5 | 5.5 |
| +/-20 | 2.5 | 2.5 | 5.5 | 5.5 |
| +/-45 | 3.5 | 3.5 | 6.0 | 6.0 |
| +/-90 | 6.5 | 6.5 | 6.5 | 6.5 |

Table 2. Probability of Detection (Pd) Values Respect to Range and Signature

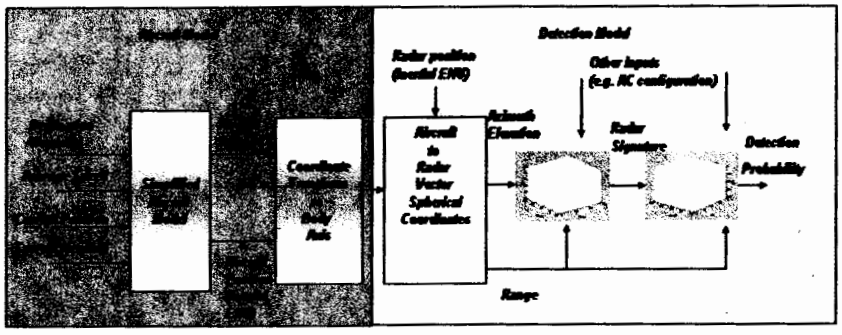

Figure 2: The aircraft dynamics and the detection model comprise two main components of the model.

| Signature | Pd=.99 | Pd=.5 | Pd=.1 | Pd=.01 |
|---|---|---|---|---|
| 7 | 380.1 | 481.2 | 555.6 | 656.6 |
| 6 | 213.7 | 270.6 | 312.5 | 369.2 |
| 5 | 120.2 | 152.2 | 175.7 | 207.6 |
| 4 | 67.6 | 85.6 | 98.8 | 116.8 |
| 3 | 38.0 | 48.1 | 55.6 | 65.7 |
| 2 | 21.4 | 27.1 | 31.2 | 36.9 |
| 1 | 12.0 | 15.2 | 17.6 | 20.8 |

At fixed altitude, it is interesting to consider ~~the~~ how the probability of detection varies with ground range as in figure {3}. The figure shows that "nose in" flight is signficantly less detectable than "nose out flight". Near 50 km, the approximate range of a homing missile, the detection probabilities vary approximatlely from 0.4 to 0.65. Outside the 80 km radar range, the detection probabilities are zero.

To aid in understanding how the probability of detection varies along various trajectories, we focus on an example where the radar is located at the origin of an inertial coordinate frame with coordinates (x (east),y (north),z (up)). An aircraft starts at position (-100 km, -100 km, 12 km) and may travel along different flight paths. Two flight paths are shown in figure 4. Positions are shown in the (x (east),y (north)) plane. Times are not indicated on the plot. The trajectory marked with circles is a straight line approach to the origin. The heading is initially 45 degrees, and the azimuth is always near zero. As an airplane following the "circle" trajectory flies closer to the origin, the elevation angle has a larger magnitude and the probability of detection rises. The 'plus' trajectory consists of several segments with portions in easterly, northerly and straight line approaches towards the radar. Note that the occasional sharp changes in the "plus" trajectory occur because the azimuth changes in and out of the +/- 30 degree bound. In the first table, azimuths outside this range yielded higher signatures and consequently higher probability of detections.

From the two tables, an analytical observability model, generically shown in equation (10) can

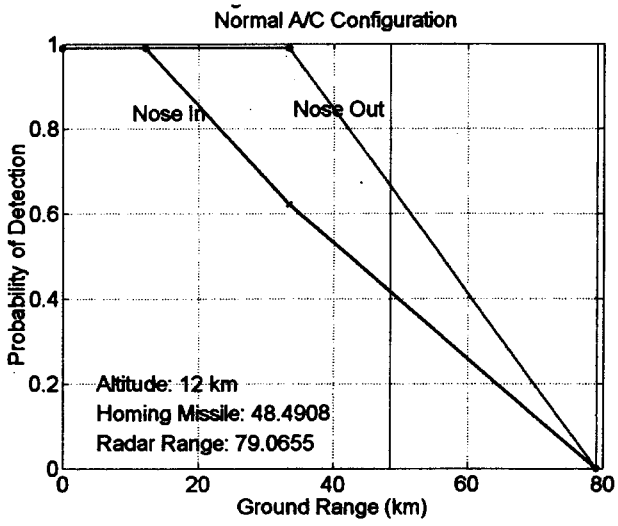


Figure 3: At fixed altitude the probability of detection is smaller for "nose in" flight with azimuths within 30 deg and decreases with increasing ground range

be developed any number of function approximation techniques.

$$sig = sig(az, el)$$
$$p_d = p_d(R, sig)$$

We have used the B–spline curve fits because of their flexibility and ease of computing their derivatives to find analytical models for the signature and probability of detection data given in tables (1) and (2), respectively. Figure (5) illustrates the signature values given in table (1) in 3–d. Figure (6) and (7) show the result of the fit function model of the signature values by B–spline tensor product functions and the error between the fit function and actual data values, respectively. Similarly, Figure (8) illustrates the probability of detection values respect to range and signature values given in table (1) in 3–d. Figure (??) and (??) show the result of the fit function model of the probability of detection values by B–spline tensor product functions and the error between the fit function and actual data values, respectively.

Then the problem is to find trajectories for the dynamic system in equation 3 such that the trajectories are optimized with respect to the probability of detection models described in this section. This section illustrates that even for highly simplified models, the optimization of trajectories for low observability flight can be quite challenging.

# 3 Approach

In this section, we first briefly outline the NTG algorithm with temporal constraints and then describe our approach to the low observability trajectory optimization problem formulation. Illustrative examples will be given in the next section.

## 3.1 Brief Summary of NTG Algorithm

The baseline NTG algorithm has been described extensively in the literature [5, 7, 6, 3], therefore in this section we outline it briefly.

The NTG software package is based on a combination of nonlinear control theory, spline theory and sequential quadratic programming. There are three steps in the NTG algorithm. The first step

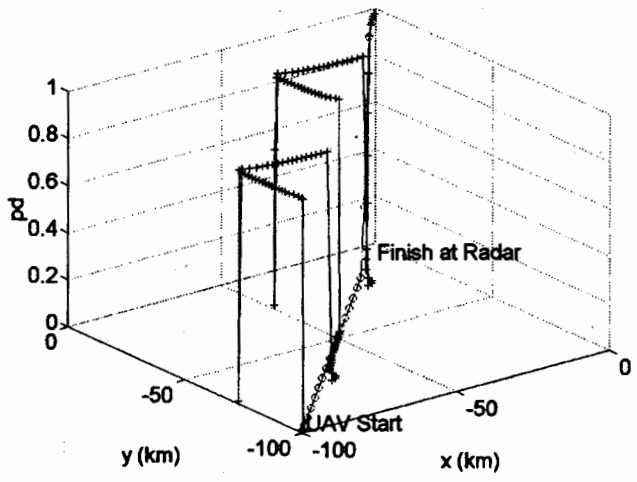


Figure 4: Probability of Detection exhibits sharp gradients and path dependencies.

is to use the differential flatness property [?] to find a new set of outputs of a system so that the system dynamics can be mapped down to a lower dimensional space. All the states and controls of the system must be recovered from the new lower dimensional representation of the system. Let's consider a nonlinear system given by:

$$\dot{x} = f(x, u) \tag{10}$$

where $x \in R^n$ and $u \in R^m$ represents the states and the inputs, respectively. It is assumed that all vector fields and functions are real–analytic. The goal is to find a minimizing trajectory of the system given in Equation (10) for the following cost function ($J$)which consists of initial ($\phi_0$), final ($\phi_f$) and trajectory ($L$) cost functions;

$$J(x, u) = \phi_0(x(t_0), u(t_0)) + \phi_f(x(t_f), u(t_f)) + \int_{t_0}^{t_f} L(x(t), u(t))dt \tag{11}$$

The system given in Equation (10) is subject to the following constraints;

$$\begin{aligned} lb_0 \leq \psi_0(x(t_0), u(t_0)) \leq ub_0 \qquad & N_0 \text{ initial constraints} \\ lb_f \leq \psi_f(x(t_f), u(t_f)) \leq ub_f \qquad & N_f \text{ final constraints} \\ lb_t \leq \psi_t(x(t), u(t)) \leq ub_t \qquad & N_t \text{ trajectory constraints} \end{aligned} \tag{12}$$

The system in Equation (10) can be mapped to a lower dimensional space, in which it will be easier and computationally more efficient to solve the optimization problem, by finding an output $z = z_1, \cdots, z_q$ of the form

$$z = A\left(x, u, u^{(1)}, \cdots, u^{(r)}\right) \tag{13}$$

where $u^{(i)}$ denotes $i$th derivative of $u$ respect to time.

If Equation (10) is differentially flat then the states and inputs of the system, $(x, u)$, can be completely established from Equation (14). If there is no flat output exists or one cannot find a flat output, then $(x, u)$ can still be completely determined from the lowest dimensional space possible given in Equation (15). A necessary condition for the existance of such outputs is given in [?].

$$(x,u) = B\left(z, z^{(1)}, \cdots, z^{(s)}\right) \tag{14}$$

$$(x,u) = B_1\left(z, z^{(1)}, \cdots, z^{(s_1)}\right) \tag{15}$$
$$(x,u) = B_2\left(z, z^{(1)}, \cdots, z^{(s_2)}\right)$$

where $z^{(i)}$ denotes $i$th derivative of $z$ respect to time.

The second step in NTG is to further represent these outputs in terms of the B–spline functions as

$$z_j(t) = \sum_{i=1}^{p_j} B_{i,k_j}(t) C_i^j \qquad \text{for the knot sequence} \quad t_j, \;\; j = 1, \cdots, q$$

where $B_{i,k_j}(t)$ represents the B–spline basis function for the output $z_j$ with the degree of spline polynomial $k_j$. $C_i^j$ represents the coefficients of the B–splines, $l_j$ is the number of knot intervals, and $m_j$ is the number of smoothness conditions at the knot points. $p_j$ is the number of coefficients of the each output given by

$$p_j = l_j(k_j - m_j) + m_j$$

Finally to solve the coefficients of the B–spline functions by sequential quadratic programming package NPSOL, the cost function and constraints given in equations (11) and (12), respectively, are re–formulated in terms of the B–spline coefficients. Therefore, the problem now can be stated as the following nonlinear programming form:

$$\min_{y \in R^M} \; F(y) \qquad \text{subject to} \qquad lb \leq c(y) \leq ub$$

where $J(x,u) \rightarrow F(y)$ and $\{\phi_0(x(t_0), u(t_0)), \;\; \phi_f(x(t_f), u(t_f)), \;\; \phi_t(x(t), u(t))\} \rightarrow c(y)$

$y = \left(C_1^1, \cdots, C_{p_1}^1, \cdots, C_1^q, \cdots, C_{p_q}^q\right)$ and $M = \sum_{i=1}^{q} p_i$

### 3.2 Using Temporal Constraints with NTG

While the NTG formulation allows any spatial constraint to be easily coded into the constraint set, including temporal constraints requires more care [5, 4]. A key idea in the temporal formulation of NTG is to let event times become state variables in the optimization. For example suppose that $n$ events occur at $T_i$ for $i = 0, 1, \cdots, n$. This allows precedence constraints to be included in the optimization by equations relating the event times. In equation 16, we define $\tau$ as a scaled time variable. It is equal zero when time, $t$, is equal to zero, but it is equal to one when all events have occurred at $T = \sum_0^n T_i$. In the setup of the optimization problem, which will be detailed below, scaled time $\tau$ rather than time $t$ is used. In the optimization details below, the use of new time variable, $\tau$, has implications for the way integrals and derivatives are written. For example, derivatives with respect to $t$ become derivatives with respect to $\tau$ with the chain rule as in 17.

$$\tau = \frac{t}{T} \tag{16}$$

$$\frac{d}{dt} \rightarrow \frac{d}{d\tau}\frac{d\tau}{dt} = \frac{1}{T}\frac{d}{d\tau} \tag{17}$$

As a result, after introducing new state variable $T$, cost and constraint functions given in equations (11) and (12) become

$$J(x,u,T) = \phi_0(x(0),u(0),T) + \phi_f(x(1),u(1),T) + \int_0^1 L(x(\tau),u(\tau),T)d\tau \tag{18}$$

$$\begin{aligned} lb_0 \leq \psi_0(x(0),u(0),T) \leq ub_0 \qquad & N_0 \text{ initial constraints} \\ lb_f \leq \psi_f(x(1),u(1),T) \leq ub_f \qquad & N_f \text{ final constraints} \\ lb_t \leq \psi_t(x(\tau),u(\tau),T) \leq ub_t \qquad & N_t \text{ trajectory constraints} \end{aligned} \tag{19}$$

There will be also additional temporal constraints which can be expressed as a set of inequalities given below

$$lb_T \leq \psi_T(T) \leq ub_T \qquad N_T \text{ temporal constraints}$$

### 3.3 Formulation of the Low Observability Problem

Now that we have outlined the general methods, we focus on how to apply these methods to the low observability problem. Define a set of event times $T_{2i}$ and $T_{2i+1}$ for $i = 0, 1, \cdots, n$ and the number of events $2n+1$ is specified. We will set up events such that periods of low observability $T_{2i-1} \leq t \leq T_{2i}$ are interspersed with times of high observability, $T_{2i} \leq t \leq T_{2i+1}$.

The system dynamics for this problem consist of the vehicle dynamics in equation (3) together with the following dynamics on the new state variables

$$\frac{dT_{2i}}{d\tau} = 0$$

$$\frac{dT_{2i+1}}{d\tau} = 0 \qquad i = 0, \ldots n$$

Note that the system is differentially flat. All variables of interest can be written as a function of the output variables, $x, y, T_{2i}, T_{2i+1}$ and their derivatives. The observability model in equation (10) is a function of the flat variables, and is used in the optimization below.

Next, we develop a set of cost and constraint functions. Note that we will develop a general set, with design options to eliminate some components of the cost and constraint functions. A cost function, $J$, is shown in equation (20). The first term in the cost function is standard in the temporal approach. Since $T$ is the mission time as shown in equation (21), this term acts to minimize the total mission time. The second term is a speed penalty. It has also been used in the past as a way to penalize control action. The third and fourth terms are cumulative penalities on observability. Since probability of detection and signature are always non-negative, these terms are appropriate and may be useful. $W_u$, $W_p$ and $W_s$ represent the weight functions on the speed, probability and signature penalties, respectively. The integral in equation (20) is respect to the scaled time, $\tau$, and has bounds from zero to one. The cost function presented here is a classic tradeoff between various perfomance measures and control action measures. In the NTG code setup, the cost function here would use two subroutines: $ucf$, the "unintegrated cost function" subroutine is used for the costs that are integrated and $icf$, the "initial cost function" may be used to add constant terms to the cost.

$$J = T^2 + \int_0^1 \left( W_u \frac{1}{T^2} \left( \left(\frac{dx}{d\tau}\right)^2 + \left(\frac{dx}{d\tau}\right)^2 \right) + W_p p_d + W_s sig \right) T d\tau \tag{20}$$

where

$$T = \sum_{i=0}^{i=n} (T_{2i} + T_{2i+1}) \tag{21}$$

Next we present a set of constraints, and we divide the constraints into subsets based on the type of constraint used. First there are constraints on the initial conditions for the variables. There are linear constraints as in equation (22) on the initial position and event times. The initial condition constraints on velocities become nonlinear because of the differentiation with respect to scaled time shown above in equation (17). The constraints on velocities are presented in equation (23). In the NTG code, the linear initial conditions are set up with a variable, while the nonlinear initial conditions are written with a subroutine, ***nlicf***.

$$\begin{aligned} x(0) \leq x(\tau)|_{\tau=0} \leq x(0) \\ y(0) \leq y(\tau)|_{\tau=0} \leq y(0) \\ minT_{2i} \leq T_{2i}|_{\tau=0} \leq maxT_{2i} \\ minT_{2i+1} \leq T_{2i+1}|_{\tau=0} \leq maxT_{2i+1} \end{aligned} \tag{22}$$

$$\begin{aligned} T\dot{x}(0) \leq \frac{dx}{d\tau}|_{\tau=0} \leq T\dot{x}(0) \\ T\dot{y}(0) \leq \frac{dx}{d\tau}|_{\tau=0} \leq T\dot{y}(0) \end{aligned} \tag{23}$$

where $\dot{(\cdot)} = \frac{d(\cdot)}{dt}$

Speed and radius of curvature limits are nonlinear functions of scaled time. Constraints for these are shown in equations (24) and (25). Note that the equations are scaled to aid in convergence of the nonlinear optimization codes. These constraints are written with the subroutine ***nltcf*** in the NTG code.

$$\frac{v_{min}^2}{v_{max}^2} \leq \frac{1}{T^2}\left(\left(\frac{dx}{d\tau}\right)^2 + \left(\frac{dx}{d\tau}\right)^2\right) \leq 1 \tag{24}$$

$$\frac{\rho_{min}}{\rho_{max}} \leq \frac{\frac{dx}{d\tau}\frac{d^2y}{d\tau^2} - \frac{dy}{d\tau}\frac{d^2x}{d\tau^2}}{\left(\left(\frac{dx}{d\tau}\right)^2 + \left(\frac{dx}{d\tau}\right)^2\right)^{1.5}} \leq 1 \tag{25}$$

Finally, constraints on observability can be included. Consider a constraint on signature as in equations (26). In order to circumvent difficulties with discontinuous constraints, which are not permitted with the NTG method, a single constraint on signature is represented with two continuous constraints. During the different low or high observability events, only one of these two constraints is active. During low-observable times the signature is constrained to be low as in the first equation in the set of equations in (26). The second constraint in the second line of (26) is present, but is automatically satisfied if the first constraint is satisfied. During the times of high-observability, the fourth equation in (26) allows the signature to reach its high value. The constraint in the third line of (26) is present, but is automatically satisfied.

$$\begin{aligned} T_{2i} \leq t \leq T_{2i+1} \quad & 0 \leq sig \leq sig_L \\ & 0 \leq sig \leq sig_H \\ T_{2i+1} \leq t \leq T_{2i+2} \quad & 0 \leq sig(T_{2i}) \leq sig_L \\ & 0 \leq sig \leq sig_H \end{aligned} \tag{26}$$

Constraints on probability of detection can be included in a similar fashion shown in equation (27).

$$\begin{aligned} T_{2i} \le t \le T_{2i+1} \quad & 0 \le p_d \le p_{d_L} \\ & 0 \le p_d \le p_{d_H} \\ T_{2i+1} \le t \le T_{2i+2} \quad & 0 \le p_d(T_{2i}) \le p_{d_L} \\ & 0 \le p_d \le p_{d_H} \end{aligned} \tag{27}$$

Note that the constraints on signature and probability of detection are nonlinear function of scaled time and are represented in the NTG code similarly to speed and curvature limits.

The last constraint in equation (28) ensures that the vehicle reaches the destination waypoint within a radius bound $e_{min}$ and $e_{max}$. This constraint is nonlinear, and it is a function of the final scaled time, $\tau = 1$, which corresponds to unknown actual final time of $t_f$.

$$\frac{e_{min}^2}{e_{max}^2} \le (x(1) - x_{destinationwaypoint}(1))^2 + (y(1) - y_{destinationwaypoint}(1))^2 \le 1 \tag{28}$$

# 4 Examples

In this secton, illustrative examples to our approach for the real time low observability nonlinear trajectory generation problem are given. We assume that altitude is fixed at 12 km. The initial location of the aircraft is at (-100 km, -100 km), and there is a radar located at the origin. Note that the difficulties with flying trajectories in these examples were highlighted earlier in figure ( 4). In the examples below, the aircraft travels to a target located at (5 km, -15 km). The examples highlight the ability of the temporal methods to design for distinct periods of high and low observability. However, they also demonstrate sensitivity to the initial coarse route.
Figures (9),(10), and (11) result from a simulation in which the initial coarse route input to the NTG algorithm is at (-100km, -15 km). Figure (9) shows that the optimized path first circles through the left side of the radar, thereby constraining the signature value of the aircraft during the low-observability event as shown in Figure (10). In this example the low signature bound is at 4.2. After ?*sec*, the signature is allowed to increase to the maximum value at 6.5. The azimuth angle of the corresponding flight path is given in Figure (11).

If NTG is initially fed with a better course route (0, -100 km), then NTG produces a different solution as illustrated in figures (12),(13), and (14). In this example the UAV flies "close in", with low azimuths, directly toward the target. The signature values are lower than in the first example.

Similar example set also included in Figures (15) through (22). The examples in figures (21) through (22) highlight that the feature of the temporal method where the signature is allowed to be high for a period of time.

# 5 Conclusions

**We will work on this section**

Because NTG is a gradient descent method, solutions that are local minima are possible. Future work is investigating the use of NTG combined with global approaches to address these issues.

Research supported as part of the MICA program for DARPA and AFRL, under contract *insertcontractnumber*. ?????

## References


[1] John T. Betts. *Practical Methods for Optimal Control Using Nonlinear Programming.* Society of Industrial and Applied Mathematics, 2001.

[2] Corman D. and Knutti J. Mica oep 1.1 user guide. *The Boeing Company Internal Report*, 2003.

[3] Lian F.-L. and Murray R.M. Real-time trajectory generation for the cooperative path planning of multi-vehicle systems. *Conference on Decision and Control*, 2002.

[4] Lian F.-L. and Murray R.M. Cooperative task planning of multi-robot systems with temporal constraints. *International Conference on Robotics and Automation*, 2003.

[5] Milam M.B. Missile interception research report. *California Institute of Technology Internal Report*, 2002check.

[6] Milam M.B., Mushambi K., and Murray R.M. A computational approach to real-time trajectory generation for constrained mechanical systems. *Conference on Decision and Control*, 2000.

[7] Milam M.B., Franz R., and Murray R.M. Real-time constrained trajectory generation applied to a flight control experiment. *IFAC 2002*, 2002.

[8] Gill P.E., Murray W., Saunders M.A., and Wright M. H. User's guide for npsol 5.0 a fortran package for nonlinear programming. *Systems Optimization Laboratory, Stanford University, Stanford CA*, 1998.

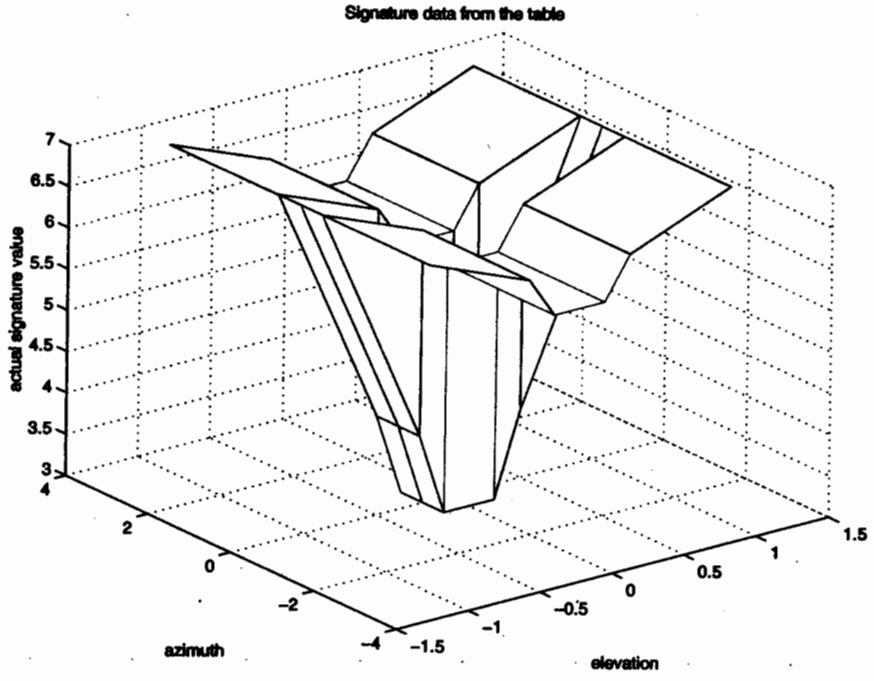


Figure 5: The Signature Data Versus Elevation and Azimuth Angles.

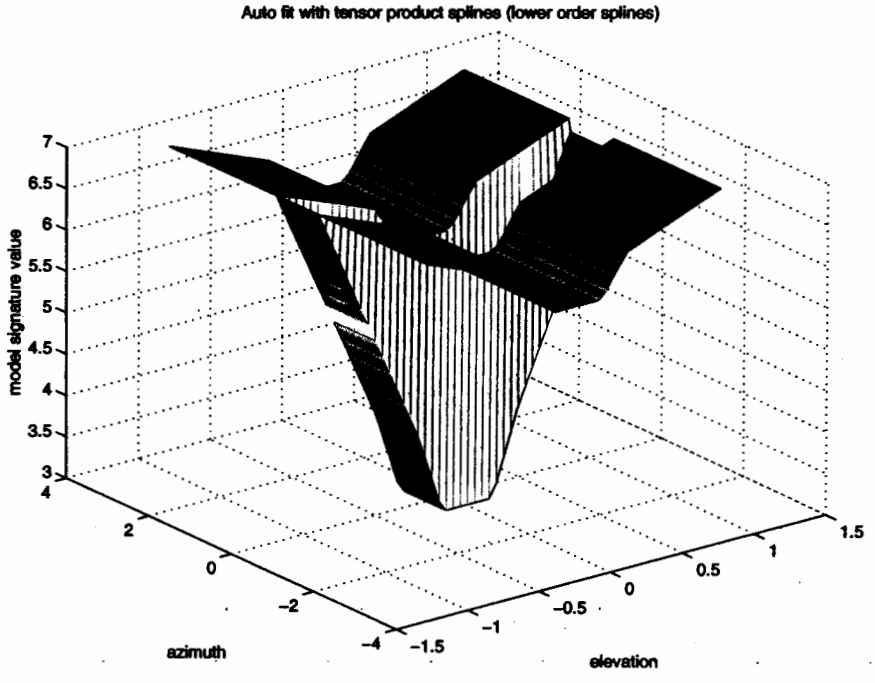


Figure 6: The Signature Data Fit Function by B–Spline Tensor Product Functions.

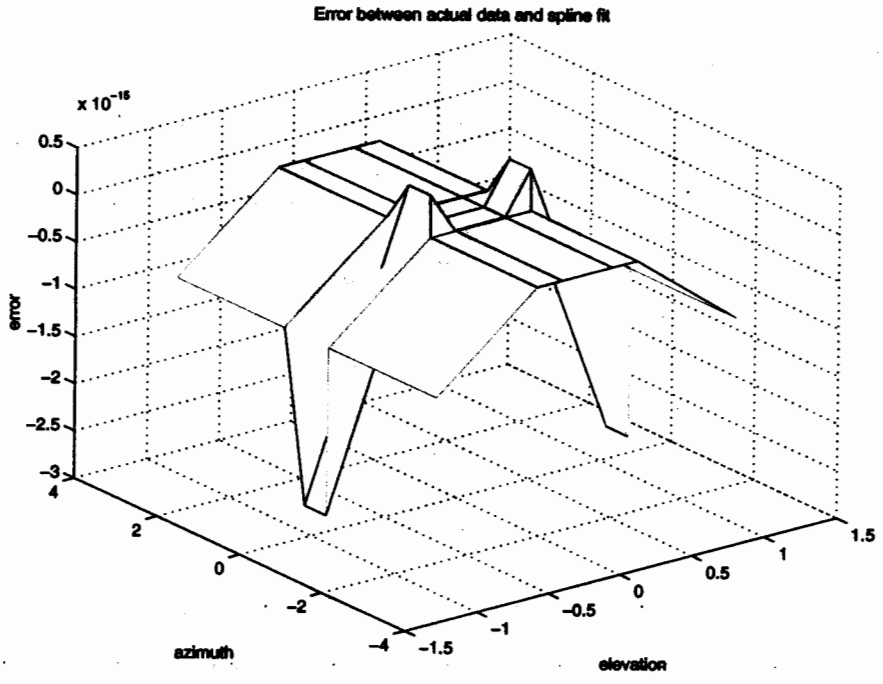


Figure 7: The Error Between Signature Data and Fit Model.

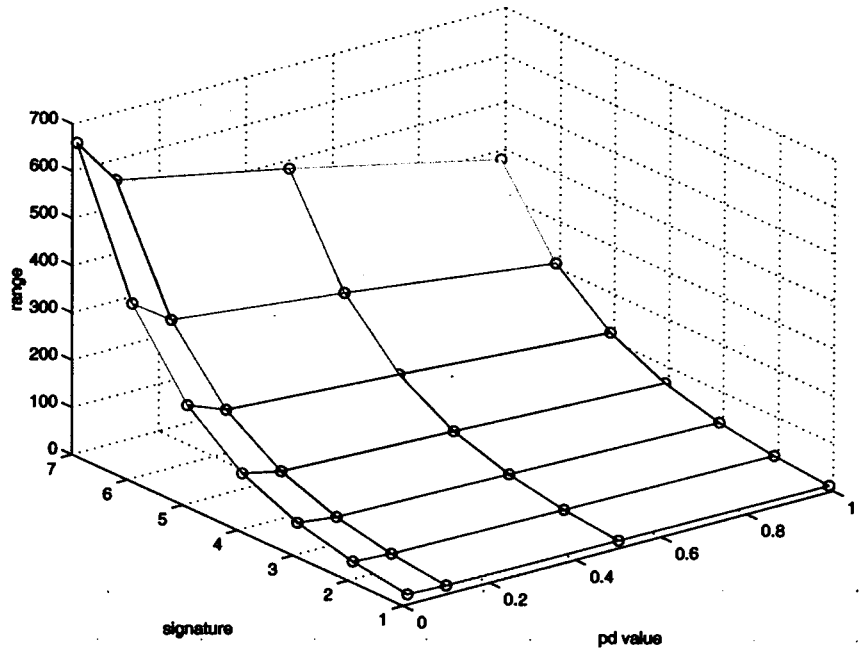


Figure 8: The Probability of Detection Data Versus Range and Signature Values.

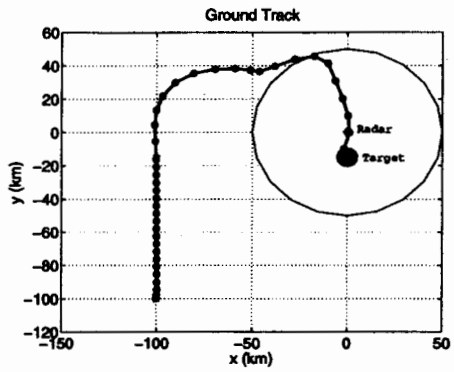


Figure 9: Ground Track of the Optimized Path for Low Observability

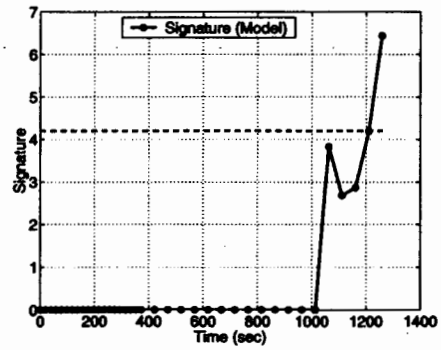


Figure 10: Signature Value Across the Optimized Path for Low Observability

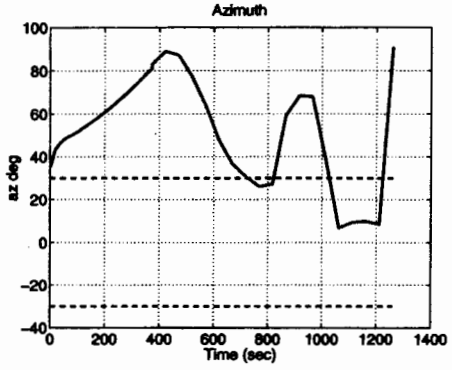


Figure 11: Azimuth Angle Across the Optimized Path for Low Observability

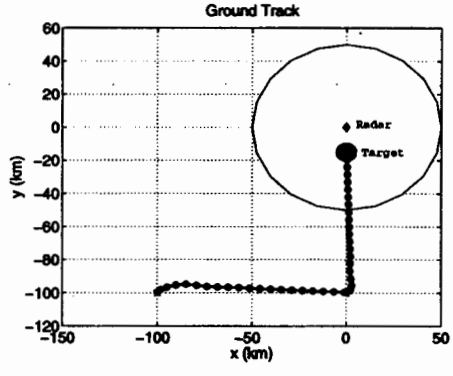


Figure 12: Ground Track of the Optimized Path for Low Observability with Better Initial Course Route

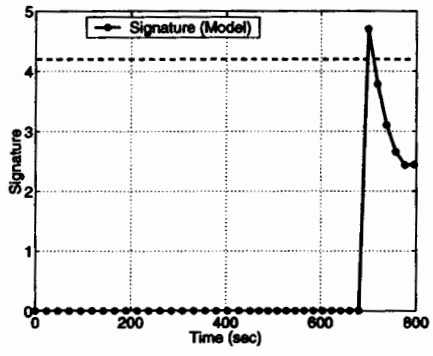


Figure 13: Signature Value Across the Optimized Path for Low Observability with Better Initial Course Route

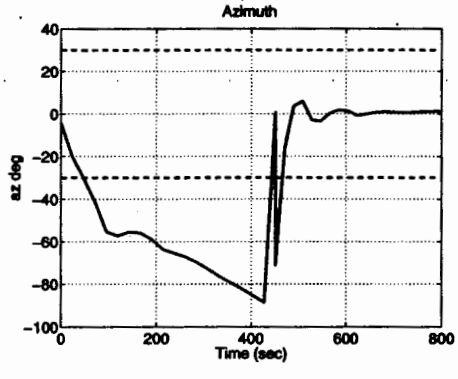


Figure 14: Azimuth Angle Across the Optimized Path for Low Observability with Better Initial Course Route

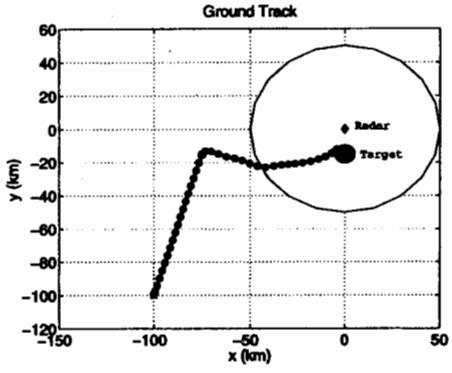


Figure 15: Ground Track of the Optimized Path for Low Observability

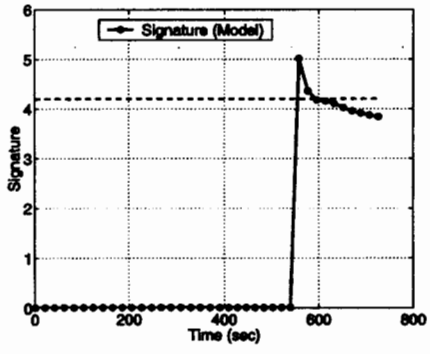


Figure 16: Signature Value Across the Optimized Path for Low Observability

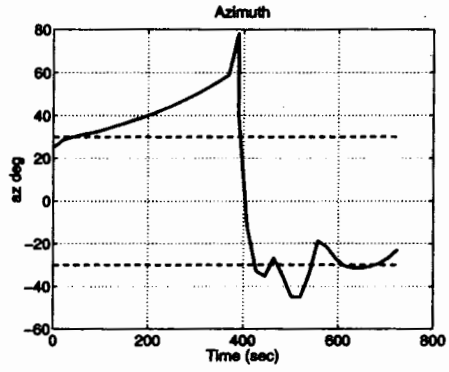


Figure 17: Azimuth Angle Across the Optimized Path for Low Observability

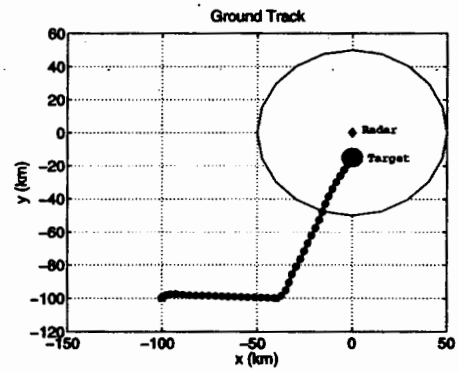


Figure 18: Ground Track of the Optimized Path for Low Observability

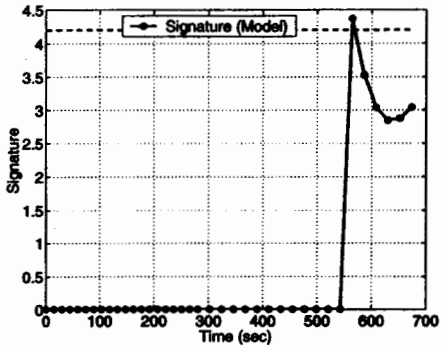


Figure 19: Signature Value Across the Optimized Path for Low Observability

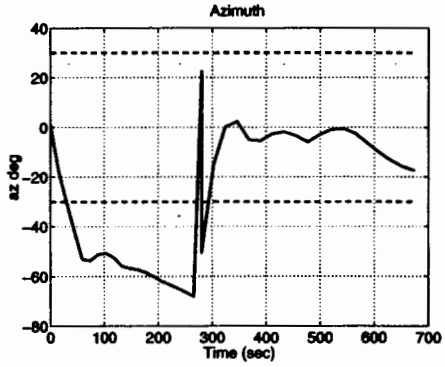


Figure 20: Azimuth Angle Across the Optimized Path for Low Observability

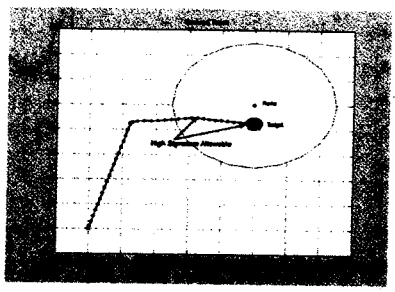

Figure 21: Ground track example 5

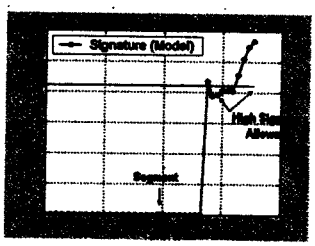


Figure 22: Signature example 5