

Method for Optical-Flow-Based Precision Missile Guidance

IAN R. MANCHESTER

Umeå University
Sweden

ANDREY V. SAVKIN

University of New South Wales
Australia

FARHAN A. FARUQI

Defence Science and Technology Organization
Australia

A new precision guidance law is presented for three-dimensional intercepts against a moving target. In contrast to previously published guidance laws, it does not require knowledge of the range to the target. This makes it appropriate for use on platforms which have an imaging device, such as a video camera, as a primary sensor.

We prove that with idealized dynamic model, the guidance law results in zero miss distance, and a formula is given for impact angle error which tends to zero as does target speed, making this method particularly suitable against slow moving targets. Computer simulations are used to test the law with a more realistic model, with a video camera and optical-flow algorithm providing target information. It is shown to perform well compared with another law from the literature, despite requiring less information.

Manuscript received May 28, 2004; revised April 12 2006; released for publication July 2, 2007.

IEEE Log No. T-AES/44/3/929704.

Refereeing of this contribution was handled by L. M. Kaplan.

This work was supported by the Australian Research Council and the Centre of Excellence in Guidance and Control.

Authors' addresses: I. R. Manchester, Dept. of Applied Physics and Electronics, Umeå University, Umeå, SE 907 31, Sweden, E-mail: (ian.manchester@tfe.umu.se); A. Savkin, School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, NSW, 2052, Australia; F. A. Faruqi, Weapon Systems Division, Defence Science and Technology Organization, Salisbury, SA, Australia.

0018-9251/08/\$25.00 © 2008 IEEE

I. INTRODUCTION

There are many practical problems of missile guidance in which simply minimizing miss distance is not sufficient, in which the direction from which the target hits the target is also important. These include situations where a heavily armored target is best hit from a specific angle, or when it is desired to disable a plane without hitting either a dangerous payload or the pilot.

Despite the many applications and recent interest in such guidance laws, papers concerning them are still relatively sparse in the literature. For this problem we use the term precision missile guidance, although there does not seem to be a standard terminology in the literature. Further to this problem, there is interest in the use of cheap sensors such as video cameras to reduce overall system cost.

To our knowledge, the first published work on this problem was by Kim and Grider in [1]. The authors of [2]–[7] study variations of the problem. All of these, however, are formulated as two-dimensional intercepts, and most assume that the position of the target is directly measureable.

Circular navigation guidance (CNG), introduced in [2], is a novel guidance strategy specifically designed for the case of impact with angular constraint. It has been shown to give perfect intercepts under idealized conditions when full state information is available, and the target is nonmanoeuvring. Originally derived for a two-dimensional planar intercept, we extend it here to the three-dimensional case, and simulate an implementation using a video camera as the primary sensor, coupled with an optical-flow algorithm. CNG has been shown to give good results in a variety of situations, including those investigated in [2] and also in other studies [3, 8] when coupled with robust control and estimation techniques from [9]–[11].

The optical-flow algorithm we use was introduced by Srinivasan in [12], [13]. It is a simple and effective algorithm which does not involve any iterative calculations, and is therefore well suited to use in real-time navigation systems. The location of the target in the image must also be calculated, however we do not discuss this task here. It is assumed that the target has been found in the image.

After notational conventions are introduced in Section II, the main part of the paper can be divided into two parts. In the first part, Sections III–VI, we introduce a simple idealized model of the missile-target intercept dynamics. Some preliminary results are provided for two-dimensional intercepts, which motivate the definition of the guidance law. The main theoretical result of the paper is then proved for this simple model.

In the second part, Sections VII–VIII, we introduce more a more complex model incorporating

a realistic video camera model, noises on the target-location measurement, an optical-flow algorithm and uncertainties in the autopilot/airframe dynamics. Computer simulations are used to assess the performance of the guidance law in these more realistic situations, and it is compared with the biased proportional navigation (PN) law introduced in [4].

Some brief conclusions are provided in Section IX, and MATLAB code for the control algorithm is given as an appendix.

II. NOTATIONAL PRELIMINARIES

Throughout the paper, matrices are denoted by capital letters, vectors by boldface lower-case letters, and scalars by standard lower-case letters.

The transpose of a matrix A is denoted A' , and $\mathbf{v}'$ refers to the row-vector form of a column vector $\mathbf{v}$. I_n denotes the n -dimensional identity matrix, and $0_{n,m}$ is an n by m matrix of zeros. The symbol $\|\cdot\|$ denotes the standard Euclidean norm of a vector. The symbol $\times$ means the cross product of vectors.

We use the standard projection operator:

DEFINITION 1 The Proj function is defined as

$$\text{Proj}_{\mathbf{y}}\mathbf{x} := (\mathbf{x}'\mathbf{z})\mathbf{z}$$

$$\text{where } \mathbf{z} := \frac{\mathbf{y}}{\|\mathbf{y}\|}.$$

We also use the formula for the angle between two vectors:

DEFINITION 2 The Angle function is defined as

$$\text{Angle}(\mathbf{x}, \mathbf{y}) := \cos^{-1} \left(\frac{\mathbf{x}'\mathbf{y}}{\|\mathbf{x}\|\|\mathbf{y}\|} \right).$$

Note that this function maps into the interval $[0, \pi]$.

At several stages, we will only be interested in a vector's orientation, and we use the notation $\vec{\mathbf{a}}$ for the orientation of $\mathbf{a}$:

DEFINITION 3 $\vec{\mathbf{a}} = \vec{\mathbf{b}}$ means that $\mathbf{a} = \alpha\mathbf{b}$ for some $\alpha > 0$.

When discussing vector orientations the angle subtended between two vectors is the metric we use, and associated with this metric we have the following open balls:

DEFINITION 4 The notation $B_\delta(\vec{\mathbf{b}})$ denotes the set $\{\mathbf{a} : \text{Angle}(\mathbf{a}, \mathbf{b}) < \delta\}$.

We will also often want to split a vector into components relative to another vector:

DEFINITION 5 The symbols $\mathbf{a}|_{\mathbf{b}}$ and $\mathbf{a}|_{\mathbf{b}^\perp}$ refer to the components of $\mathbf{a}$ colinear with, and orthogonal to $\mathbf{b}$, respectively.

That is:

$$\mathbf{a}|_{\mathbf{b}} := \text{Proj}_{\mathbf{b}}\mathbf{a}$$

$$\mathbf{a}|_{\mathbf{b}^\perp} := \mathbf{a} - \text{Proj}_{\mathbf{b}}\mathbf{a}.$$

Where confusion is unlikely to occur, we will often omit the time dependence of variables to improve the clarity and compactness of equations.

III. PROBLEM STATEMENT

We wish to guide the missile through three-dimensional space to some point in that space, approaching in some desired direction.

We consider a simple model of Newtonian physics, subject to the kinematic constraint that the missile may accelerate only in a direction orthogonal to its current velocity. That is, the missile may only perform turning motions.

We now define a constrained linear state-space system which describes this model. The state consists of three three-dimensional vectors:

- 1) relative position— $\mathbf{x}_R(t) := \text{target position} - \text{missile position}$,
- 2) missile velocity— $\mathbf{v}_M(t)$,
- 3) target velocity— $\mathbf{v}_T(t)$.

Now let a state vector $\mathbf{x}(t)$ represent these combined:

$$\mathbf{x}(t) := \begin{bmatrix} \mathbf{x}_R(t) \\ \mathbf{v}_M(t) \\ \mathbf{v}_T(t) \end{bmatrix} \in \mathbb{R}^9.$$

Let $\mathbf{u}_c$ be the missile's acceleration, regarded as a control input. Then the system dynamics are governed by the following differential equation:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}_1\mathbf{u}_c(t) \quad (1)$$

where the matrices in (1) are defined as follows:

$$\mathbf{A} = \begin{bmatrix} 0_{3,3} & -I_3 & I_3 \\ 0_{6,9} \end{bmatrix}, \quad \mathbf{B}_1 = \begin{bmatrix} 0_{3,3} \\ I_3 \\ 0_{3,3} \end{bmatrix} \quad (2)$$

and the inputs are subject to the constraints:

$$\mathbf{u}_c(t)' \mathbf{v}_M(t) = 0 \quad \forall t. \quad (3)$$

There are two main measurements available to the guidance system. Firstly, the angular position of the target, i.e., the orientation of the relative position vector. Secondly, the results of an optical-flow calculation are available.

Ideally, an optical-flow field would be a projection of the velocity field of objects in the imaging sensor's field of view (FOV) onto the imaging plane. From this, assuming a calibrated sensor, the angular velocity of the relative position vector can be derived. We denote this angular velocity vector $\mathbf{f}_T$.

The calculation of optical-flow fields is a difficult and on-going area of research, and we discuss the problem further in Section VIIC. For the moment, we just assume the measurement is available.

So the complete measurement signal, denoted $\mathbf{y}(t)$, is defined as follows:

$$\mathbf{y}(t) := \begin{bmatrix} \overrightarrow{\mathbf{x}_R(t)} \\ \mathbf{f}_T(t) \end{bmatrix} \quad (4)$$

where

$$\mathbf{f}_T := \frac{(\mathbf{v}_T(t) - \mathbf{v}_M(t))|_{\mathbf{x}_R^\perp(t)}}{\|\mathbf{x}_R(t)\|}. \quad (5)$$

It is also assumed that the missile knows its own current velocity $\mathbf{v}_M$.

We define a vector $\mathbf{v}_F \in \mathbb{R}^3$ as the “desired final velocity vector.” Then the problem is to find some control signal:

$$\mathbf{u}_c(t) = f(\mathbf{y}(t), \mathbf{v}_M(t), \mathbf{v}_F(t)) \in \mathbb{R}^3$$

such that there exists some unspecified final time T by which the constrained linear system (1), (3) is driven to a state satisfying:

$$\begin{aligned} \|\mathbf{x}_R(T)\| &\rightarrow \min \\ \text{Angle}(\mathbf{v}_M(T), \mathbf{v}_F(T)) &\rightarrow \min. \end{aligned}$$

Corresponding to this, we make the following definition:

DEFINITION 6 A target intercept is considered perfect if there exists some finite time T such that

$$\begin{aligned} \|\mathbf{x}_R(T)\| &= 0 \\ \text{Angle}(\mathbf{v}_M(T), \mathbf{v}_F(T)) &= 0. \end{aligned}$$

For most of the rest of the paper, in order to keep the equations simple, we drop the (t) arguments from the above-defined signals whenever they are unnecessary.

IV. SOME PRELIMINARY RESULTS IN TWO DIMENSIONS

In this section we recall a result from [2] which will motivate the derivation of the control law in the next section.

Suppose for a moment that the missile is moving in a two-dimensional plane, and the target is a stationary point in that plane. We introduce two angles of interest: λ , defined as the angle between the missile’s current velocity vector and the line-of-sight between missile and target, and ε , defined as that between the line-of-sight and the desired final velocity vector.

This is visualized in Fig. 1, where the point A is the target position, and B the missile’s initial position. BA, then, is the line-of-sight, and let AZ (equivalently BY) be the desired final missile velocity vector.

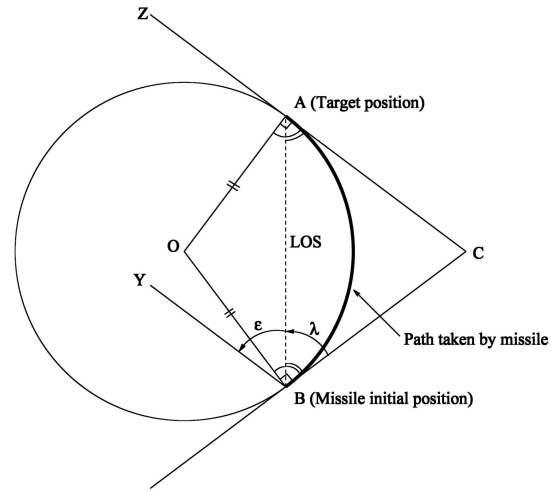


Fig. 1. Geometry for Theorem 1.

The following theorem was proved in [2], and is the basis of CNG.

THEOREM 1 Consider the case of a stationary target. Introduce the circle uniquely defined by the following properties:

The initial position of the missile lies on the circle.

The position of the target lies on the circle.

The desired final velocity vector at the target’s position is a tangent to the circle.

Suppose that a controller is designed such that the angles $\lambda(t)$ and $\varepsilon(t)$ are kept exactly equal over the full flight time, then the missile’s trajectory will be an arc on this circle. Furthermore, this will result in a perfect intercept, as defined in Definition 6.¹

For the following corollary, observe that

$$\dot{\lambda} = \dot{\lambda}_0 - \frac{u_c}{\|\mathbf{v}_M\|}$$

where

$$\dot{\lambda}_0 = \frac{\|\mathbf{v}_M\|}{\|\mathbf{x}_R\|} \sin(\lambda).$$

Now, suppose that

$$u_c = 2 \frac{\|\mathbf{v}_M\|^2}{\|\mathbf{x}_R\|} \sin(\lambda) \quad (6)$$

then clearly

$$\dot{\lambda} = -\dot{\lambda}_0.$$

Furthermore:

$$\begin{aligned} \dot{\varepsilon} &= -\frac{\|\mathbf{v}_M\|}{\|\mathbf{x}_R\|} \sin(\lambda) \\ &= -\dot{\lambda}_0 \\ &= \dot{\lambda} \end{aligned}$$

¹In [2] the definition of perfect intercept was slightly different. However, in the case we consider here it is equivalent to Definition 6.

and hence $\dot{\lambda}(t) = \dot{\varepsilon}(t) \forall t$. Therefore, we can state the following corollary to Theorem 1.

COROLLARY 1 *Suppose the control law (6) is used against a stationary target. Then the missile will follow a circular path to the target, achieving zero miss distance, and the impact angle will be equal to $\lambda(0) - \varepsilon(0)$.*

So this control law would achieve the guidance aims for a stationary target if the initial velocity of the missile was correct, i.e., $\lambda(0) = \varepsilon(0)$.

V. THREE-DIMENSIONAL GUIDANCE LAW

In this section we show how the two-dimensional geometric principles discussed above can be applied to three-dimensional guidance problems using information from a video camera, or a similar sensor. Our aim in this section is to develop the reasoning behind the control law. The complete algorithm is given in MATLAB code in the appendix.

As above, we consider two points, one with a vector attached and use the fact that these three statistics define a unique circle. Of course, if defined in three-dimensional space, they also define a unique plane on which the above-mentioned circle exists, assuming that this vector is not colinear with the line joining the points.

The most straightforward way to extend the “keep λ and ε equal” strategy to three dimensions is to introduce the unique “desired velocity” vector, for which this condition is satisfied. If the missile maintains the desired velocity, then the above-mentioned circle will be the path the missile takes through space.

This vector, denoted $\mathbf{v}_D$, is the reflection of the desired final velocity vector, $\mathbf{v}_F$, with respect to the current relative position vector $\mathbf{x}_R$

$$\mathbf{v}_D := \mathbf{v}_F|_{\mathbf{x}_R} - \mathbf{v}_F|_{\mathbf{x}_R^\perp}.$$

We say that the CNG condition is satisfied whenever $\mathbf{v}_M$ equals $\mathbf{v}_D$.

The problem then is to design a control law which will bring $\mathbf{v}_M$ close to $\mathbf{v}_D$. The control strategy is made up of two parts:

- 1) a control signal $\mathbf{u}_p$ which acts to rotate $\mathbf{v}_M$ toward $\mathbf{v}_D$ using the optical position of the target,
- 2) a control signal $\mathbf{u}_f$ which uses optical-flow information to rotate the missile with the same angular velocity as $\mathbf{v}_D$, i.e., to keep their relative orientation constant.

A. Optical Position-Based Control Component

We then introduce an error signal e_λ , which is the angle between the current velocity and this desired

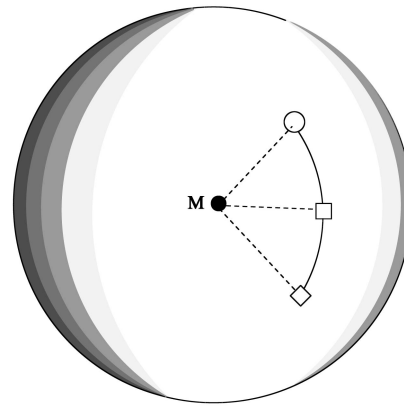


Fig. 2. FOV representation.

velocity:

$$e_\lambda := \text{Angle}(\mathbf{v}_D, \mathbf{v}_M).$$

And, remembering that we can only apply a control signal in a direction orthogonal to $\mathbf{v}_M$, we calculate a unit vector, $\mathbf{u}_{D1}$ representing the direction in which control should be applied to bring $\mathbf{v}_M$ closer to $\mathbf{v}_D$. This is the normalized projection of $\mathbf{v}_D$ onto the subspace orthogonal to $\mathbf{v}_M$:

$$\mathbf{u}_{D1} := \frac{\mathbf{v}_D|_{\mathbf{v}_M^\perp}}{\|\mathbf{v}_D|_{\mathbf{v}_M^\perp}\|}.$$

Then a simple proportional control law can be calculated as some gain k_p times the error, issued in this direction:

$$\mathbf{u}_p = (k_p e_\lambda) \mathbf{u}_{D1}.$$

We now discuss how this control law can be derived from optical information.

Fig. 2 is a depiction of the FOV of the missile, represented as a unit sphere. The missile is the point M at the center of the sphere. Points in the FOV are vector orientations, such as $\bar{\mathbf{x}}_R$. In different terms, they are equivalence classes of vectors, unique up to positive scalar multiplication, from M out into space in some direction. They can be uniquely represented as a point on the surface of a unit sphere centred at M (e.g. the circle, square and diamond depicted, which is referred to later). Through every 2 distinct points there is a unique “great circle” in three-dimensional space on the surface of the sphere. The “distance” between two points is defined as the angle subtended by vectors they represent in three-dimensional space in which the sphere is embedded (this obviously does not depend on the particular vectors chosen from the equivalence classes).

In Fig. 3, we “flatten out” the FOV into a plane for simplicity of visualization. Note that this cannot be done isometrically, so “lines” are great circles, and “distances” refer to angles subtended, not Euclidean distances.

It is assumed that the desired final velocity vector $\mathbf{v}_F$ is known (or at least estimated) and therefore its orientation corresponds to a unique point in the FOV.

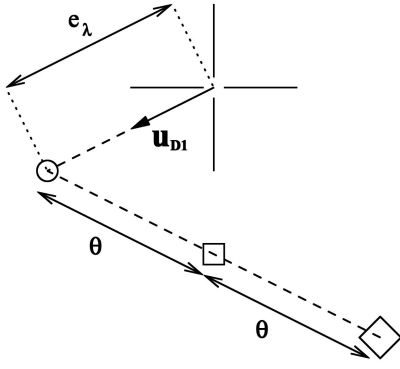


Fig. 3. Proportional control.

This is represented as a diamond ($\diamond$). The target is assumed to be visible, and thus also corresponds to a point in the FOV, which we depict as a square ($\square$).

Consider a line drawn between the $\mathbf{v}_F$ point ($\diamond$) and the target point ($\square$). Now, extrapolate this line such that it is exactly twice as long as it was. The end point of this line, represented by a circle ($\circ$), is the point in the FOV that corresponds to the desired velocity vector $\mathbf{v}_D$. In summary:

$$\begin{aligned}\diamond &\Leftrightarrow \bar{\mathbf{v}}_F \\ \square &\Leftrightarrow \bar{\mathbf{x}}_R \\ \circ &\Leftrightarrow \bar{\mathbf{v}}_D.\end{aligned}$$

$\mathbf{u}_{D1}$ is a unit vector giving the direction between the current heading $\mathbf{v}_M$ (cross-hairs) and this point ($\circ$), and e_λ is the angular separation of the two. So our control signal

$$\mathbf{u}_p = (k_p e_\lambda) \mathbf{u}_{D1}$$

is a simple first-order proportional controller, which drives the missile heading toward this desired heading ($\circ$).

B. Optical-Flow-Based Control Component

If it is possible to measure or estimate the movement of the target across the missile's FOV, we may also construct a second control signal based on this. Later in the paper we use an optical-flow algorithm which approximates this quantity.

Consider Fig. 4. The desired final velocity ($\diamond$) is static, so it can be considered a fixed point on the sphere. The target's optical position ($\square$) is then a point moving across the surface of this sphere. As above, the desired velocity ($\circ$) is the extrapolation of a great circle C defined by these two, extended to twice the subtended angle.

Infinitesimal movements of the points $\circ$ and $\square$ are vectors on the plane tangent to the sphere at each point. We can break each of these vectors into two components: one orthogonal to C and another tangential. These are denoted du_T and dv_T , respectively, for the target point ($\square$) and du_D and dv_D , respectively, for the desired velocity ($\circ$); see Fig. 4.

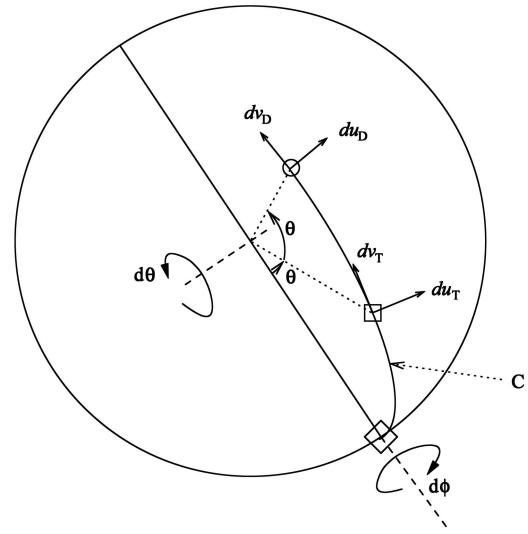


Fig. 4. Optic flow control.

Considering $\diamond$ as a fixed pole, we notice that for the $\diamond$ – $\square$ – $\circ$ relation to be maintained the “latitude”² (θ rotation in Fig. 4) of $\circ$ must be twice that of $\square$, and their “longitude” (ϕ rotation in Fig. 4) must be equal. Their derivatives must obviously obey the same rule, so we can deduce the following relations between their magnitudes:

$$dv_D = 2d\theta = 2dv_T \quad (7)$$

$$\frac{du_D}{\sin(2\theta)} = d\phi = \frac{du_T}{\sin(\theta)}$$

$$\begin{aligned}du_D &= \frac{\sin(2\theta)}{\sin(\theta)} du_T \\ &= 2 \cos(\theta) du_T.\end{aligned} \quad (8)$$

In order to calculate a control law which will achieve this relation, use the second measurement output $\mathbf{f}_T$. This vector is essentially the angular velocity of the relative orientation $\bar{\mathbf{x}}_R$.

We can see $\mathbf{f}_T$ as inhabiting the tangent space to the sphere at $\square$, and represent it in this space as components $(\dot{u}_T, \dot{v}_T)$ normal and tangential, respectively, to C (see Fig. 4). This and the next step require simple linear mappings between coordinate systems which are not of principal importance, and are relegated to the appendix.

Then, in accordance with (7) and (8):

$$\dot{\mathbf{v}}_D = \begin{bmatrix} \dot{u}_D \\ \dot{v}_D \end{bmatrix} = \begin{bmatrix} 2 \cos \theta \dot{u}_T \\ 2 \dot{v}_T \end{bmatrix}$$

where $\dot{\mathbf{v}}_D$ is a vector in the tangent space at $\circ$, coordinated with a first component orthogonal to C , second tangential, as above.

²This is not quite like geographical latitude, which is measured from the equator, since θ is measured from the pole. It is actually a zenith angle.

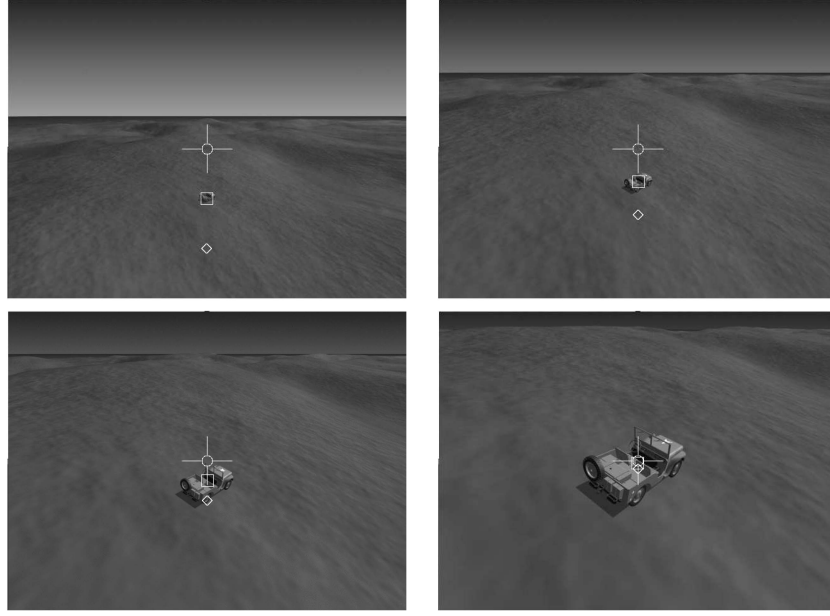


Fig. 5. Simulated optical sequence.

From this, we can derive an angular velocity vector

$$\mathbf{w} = \mathbf{v}_D \times \dot{\mathbf{v}}_D$$

which carries an implicit assumption that $\mathbf{v}_D$ follows a great circle, i.e., its angular velocity is orthogonal to its position. This is, in a sense, analogous to assuming nonaccelerating motion in Euclidean geometry, and seems the simplest assumption for this underconstrained problem.

Then, our purpose being for the missile to rotate in accordance with $\mathbf{v}_D$, we generate the control signal $\mathbf{u}_f$ by applying this angular velocity to the missile's current velocity:

$$\mathbf{u}_f = \mathbf{w} \times \mathbf{v}_M.$$

Now, the relative orientation of $\mathbf{v}_M$ and $\mathbf{v}_D$ is unchanged by relative motion of the missile and target.

C. Illustration

To illustrate this section, we set out in Fig. 5 a sequence of four computer generated images depicting the missile's FOV as it closes in on a target, with the symbols (□, ◇, ○) superimposed on the images, indicating how they change with time. Fig. 6 shows the trajectory taken by the missile from another perspective.

VI. IDEAL CASE SOLUTION

In the previous section we presented a control law which drives $\mathbf{v}_M$ towards $\mathbf{v}_D$, and if they are equal, keeps them equal. In this section, we prove results guaranteeing intercept performance supposing

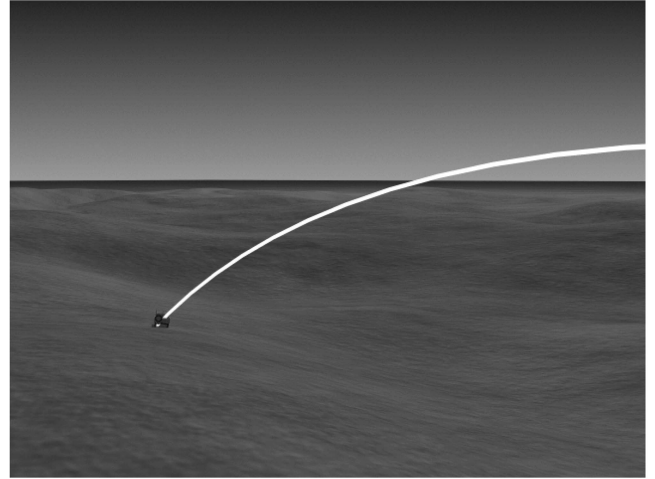


Fig. 6. Trajectory taken by missile.

that $\mathbf{v}_M = \mathbf{v}_D$. In the idealized model presented above, this can be achieved within any desired level of accuracy by increasing the gain k_p . However practical considerations such as noise and unmodelled dynamics will limit the size of k_p .

DEFINITION 7 In the following we refer to a set of initial conditions as being thin. This set consists of a single point on the two-dimensional compact manifold, the sphere $\mathbb{S}^2$, and thus has zero measure.

The following theorem is the main theoretical result of this paper:

THEOREM 2 Suppose the target is moving with constant velocity $\mathbf{v}_T$, but the missile is faster than the target, i.e., $\|\mathbf{v}_M\| > \|\mathbf{v}_T\|$. Let β represent the angle between $\mathbf{v}_T$ and $\mathbf{v}_F$. If a controller of the form (6) is designed such that $\mathbf{v}_M(t) = \mathbf{v}_D(t)$ over the full

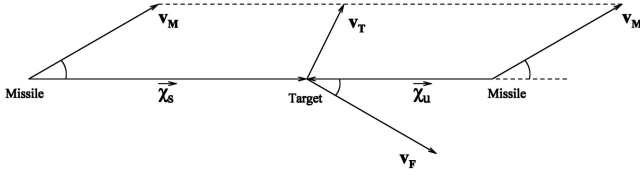


Fig. 7. Fixed points of orientation of $\mathbf{x}_R$.

flight time, then for all initial conditions, except on a thin set, the missile will impact the target with zero miss distance, and angle error given by the following formula:

$$\text{Angle}(\mathbf{v}_M(T), \mathbf{v}_F(T)) = 2 \tan^{-1} \left[\frac{\|\mathbf{v}_T\| \sin \beta}{\|\mathbf{v}_M\| + \|\mathbf{v}_T\| \cos \beta} \right]. \quad (9)$$

We are primarily interested in the orientation of the relative position: $\mathbf{x}_R$. Since, by definition,

$$\dot{\mathbf{x}}_R = \mathbf{v}_T - \mathbf{v}_M$$

$\mathbf{x}_R$ will be fixed when the components of $\mathbf{v}_T$ and $\mathbf{v}_M$ orthogonal to $\mathbf{x}_R$ are equal. If the CNG condition holds, then for any given $\mathbf{v}_T$ and $\mathbf{v}_F$, there will be two fixed points of $\mathbf{x}_R$, one for which $\|\mathbf{x}_R\|$ is decreasing, and one for which it is increasing (see Fig. 7). That is, one which results in a collision, one which does not. We denote these $\vec{\chi}_s$ and $\vec{\chi}_u$, respectively, the subscripts implying “stable” and “unstable.”

This theorem essentially proves that, for all possible initial conditions except $\vec{\chi}_u$, the missile will hit the target, and will do so with a known angle error. The proof of the theorem follows from three lemmas which we now prove.

LEMMA 1 *From any initial conditions except where $\mathbf{x}_R = \vec{\chi}_u$, for all $\delta \in (0, \pi)$, $\mathbf{x}_R$ eventually enters the set $B_\delta(\vec{\chi}_s)$ and never leaves it.*

PROOF OF LEMMA 1 From the fact that χ_s is a fixed point, and from the CNG principle, the following facts hold:

$$\mathbf{v}_M|_{\chi_s^\perp} = \mathbf{v}_T|_{\chi_s^\perp} \quad (10)$$

$$\mathbf{v}_F|_{\chi_s} = \mathbf{v}_M|_{\chi_s} \quad (11)$$

$$\mathbf{v}_F|_{\chi_s^\perp} = -\mathbf{v}_M|_{\chi_s^\perp}. \quad (12)$$

Suppose $\mathbf{x}_R = \vec{\chi}_s$. Then the missile trajectory is a straight line and, with the target's trajectory, forms a collision triangle such as the one shown on the left hand side of Fig. 8, and we have

$$\vec{\chi}_s = \overrightarrow{\mathbf{v}_M - \mathbf{v}_T}.$$

Now we show that

$$\vec{\chi}_s = \overrightarrow{\mathbf{v}_F + \mathbf{v}_T} \quad (13)$$

as illustrated in the right hand side of Fig. 8.

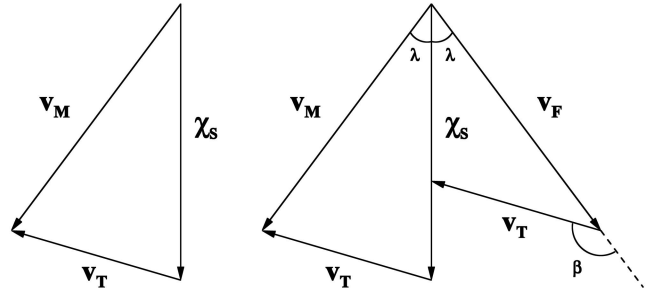


Fig. 8. Proof that orientation of χ_s equals that of $\mathbf{v}_F + \mathbf{v}_T$.

From (10):

$$\mathbf{v}_M - \mathbf{v}_T = \mathbf{v}_M|_{\chi_s} - \mathbf{v}_T|_{\chi_s} \quad (14)$$

and from (10), (11), and (12):

$$\mathbf{v}_F + \mathbf{v}_T = \mathbf{v}_M|_{\chi_s} + \mathbf{v}_T|_{\chi_s}. \quad (15)$$

So χ_s and $\mathbf{v}_F + \mathbf{v}_T$ must be colinear.

Now, from (10) and the assumption $\|\mathbf{v}_M\| > \|\mathbf{v}_T\|$, it follows that

$$\|\mathbf{v}_M|_{\chi_s}\| > \|\mathbf{v}_T|_{\chi_s}\| \quad (16)$$

from which it is clear that the vectors expressed by (14) and (15) have the same orientation, so the assertion of (13) holds.

Now, let us take the particular vector $\chi_s := \mathbf{v}_F + \mathbf{v}_T$, which obviously has orientation $\vec{\chi}_s$. Now, by definition,

$$\dot{\mathbf{x}}_R := \mathbf{v}_T - \mathbf{v}_M$$

$$\mathbf{v}_M := \mathbf{v}_F|_{\mathbf{x}_R} - \mathbf{v}_F|_{\mathbf{x}_R^\perp}$$

so we have

$$\begin{aligned} \chi_s &= \dot{\mathbf{x}}_R + \mathbf{v}_M + \mathbf{v}_F \\ &= \dot{\mathbf{x}}_R + 2\mathbf{v}_F|_{\mathbf{x}_R} \end{aligned}$$

and so the vectors χ_s , $\mathbf{x}_R$, $\dot{\mathbf{x}}_R$ must all lie in one plane, and so must any other vector with orientation $\vec{\chi}_s$.

This implies that, except at fixed points, $\mathbf{x}_R$ is either rotating directly towards, or directly away from $\vec{\chi}_s$, as its derivative isn't taking it out of this plane. Now, by inspection of

$$\dot{\mathbf{x}}_R = \chi_s - 2\mathbf{v}_F|_{\mathbf{x}_R}$$

we see that the derivative of $\mathbf{x}_R$ is from a point in the subspace spanned by $\mathbf{x}_R$ to the vector χ_s , and hence whatever $\mathbf{x}_R$ is, its orientation must be rotating towards χ_s . The only exception is if $\mathbf{x}_R = \vec{\chi}_u$, which is the opposite of $\vec{\chi}_s$, so $\mathbf{x}_R$ does not rotate. In this case, $\text{Angle}(\mathbf{x}_R, \chi_s) = \pi$, its maximum.

So, on the set $(0, \pi)$, $\text{Angle}(\mathbf{x}_R, \chi_s)$ is monotonically decreasing, and it is bounded below by zero, therefore it converges to zero. The statement of the Lemma follows.

LEMMA 2 *There exists a δ such that, if $\mathbf{x}_R \in B_\delta(\vec{\chi}_s)$, range is strictly decreasing, its derivative bounded away*

from zero by a finite amount: i.e., $d\|\mathbf{x}_R\|/dt < -\epsilon$ for some $\epsilon > 0$.

PROOF OF LEMMA 2 In this lemma we consider the range rate dynamics:

$$\frac{d\|\mathbf{x}_R\|}{dt} = \|\mathbf{v}_T\| \cos[\text{Angle}(\mathbf{v}_T, \mathbf{x}_R)] - \|\mathbf{v}_M\| \cos[\text{Angle}(\mathbf{v}_M, \mathbf{x}_R)]. \quad (17)$$

By the CNG condition, $\text{Angle}(\mathbf{v}_M, \mathbf{x}_R) = \text{Angle}(\mathbf{v}_F, \mathbf{x}_R)$. Also, we note that $\mathbf{v}_T$, $\mathbf{v}_F$ and $\|\mathbf{v}_M\|$ are constant, so the range rate is a static, continuous function of $\mathbf{x}_R$ only.

By (16), for $\mathbf{x}_R = \vec{\chi}_s$, it is strictly decreasing, so by continuity it is strictly decreasing in some ball $\mathbf{x}_R \in B_\delta(\vec{\chi}_s)$.

LEMMA 3 *If collision occurs, then the angle error is as stated in the theorem.*

PROOF OF LEMMA 3 Suppose a collision has occurred, so $\mathbf{x}_R(T) = 0$ for some time T . Then, since all trajectories are smooth, for a sufficiently short time before collision, the trajectories form a collision triangle like the one on the left in Fig. 8.

We make the following definitions, as indicated on the right hand side of that figure:

$$\lambda = \text{Angle}(\mathbf{v}_M, \mathbf{x}_R) = \text{Angle}(\mathbf{v}_F, \mathbf{x}_R)$$

$$\beta = \text{Angle}(\mathbf{v}_F, \mathbf{v}_T).$$

The internal angle of the collision triangle opposite $\mathbf{v}_M$ is $\pi + \lambda - \beta$, so by the sine rule,

$$\frac{\sin(\pi + \lambda - \beta)}{\|\mathbf{v}_M\|} = \frac{\sin(\lambda)}{\|\mathbf{v}_T\|}$$

$$\|\mathbf{v}_T\|(\sin \beta \cos \lambda - \cos \beta \sin \lambda) = \|\mathbf{v}_M\| \sin \lambda$$

$$\tan \lambda (\|\mathbf{v}_M\| + \|\mathbf{v}_T\| \cos \beta) = \|\mathbf{v}_T\| \sin \beta$$

giving

$$\lambda = \tan^{-1} \left[\frac{\|\mathbf{v}_T\| \sin \beta}{\|\mathbf{v}_M\| + \|\mathbf{v}_T\| \cos \beta} \right].$$

Now, the angle error is $\text{Angle}(\mathbf{v}_M, \mathbf{v}_F) = 2\lambda$, and the statement of the lemma follows.

PROOF OF THEOREM 2 From Lemmas 1 and 2 it follows that, unless $\overrightarrow{\mathbf{x}_R(0)} = \vec{\chi}_u$ (a thin set as per Definition 7), there exists a time after which range is strictly decreasing, and its derivative is bounded away from zero. Therefore, range reaches zero in finite time, i.e., collision occurs. Then, by Lemma 3 the angle error is as stated.

This completes the proof of the theorem.

VII. MODEL FOR COMPUTER SIMULATIONS

The preceding section described the performance of CNG in an ideal case where acceleration can be

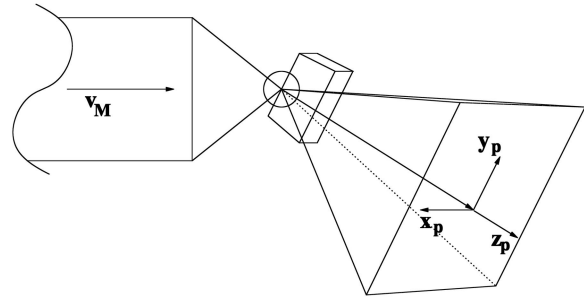


Fig. 9. Gimbal-mounted camera model.

commanded directly, and angle information of the target can be measured without errors. The ability of the guidance law to match this level of performance will depend heavily on how close the missile system it is guiding comes to meeting this ideal.

An optical sensor is prone to a number of errors, primarily due to vibration and rotation of the missile body, difficulty finding and segmenting the target from the background in the image, and errors inherent in the vision algorithms used.

In this section we show how all the information required for the guidance law can be derived from position and optical flow of the target in the viewplane of a video camera. For our sensor model, the target must remain within the camera's FOV for the entire intercept. The camera is mounted on a rotating gimbal, which is controlled so as to keep the image of the target as close to the centre as possible. We have assumed that the missile body is small enough relative to the intercept scale that its position can be considered a single point, and the camera has its focal point at this point.

A. Camera Model

We assume a simple camera with a pointing axis $\mathbf{z}_p$, focal length f , and horizontal and vertical coordinates in the image plane that correspond to the vectors $\mathbf{x}_p$, $\mathbf{y}_p$, respectively. This is visualized in Fig. 9.

Suppose the target has coordinates (x, y, z) with respect to the reference frame with the vectors $(\mathbf{x}_p, \mathbf{y}_p, \mathbf{z}_p)$ as a basis. Then the image of the target on the screen has coordinates (u, v) , where $u = fx/z$, $v = fy/z$.

B. Gimbal Model

A rotating gimbal is controlled so as to keep the camera pointing at the target. We do not study this issue in detail, but model the system as a first-order lag.

The camera pointing axis always rotates towards the current estimate of target's direction—derived

in Sections VIID and VIIE, with an angular velocity of k_g times the angle separating them:

$$\mathbf{w}_g = k_g \text{Angle}(\mathbf{z}_p, \hat{\mathbf{x}}_R) \frac{\mathbf{z}_p \times \hat{\mathbf{x}}_R}{\|\mathbf{z}_p \times \hat{\mathbf{x}}_R\|}$$

$$\dot{\mathbf{z}}_c = \mathbf{w}_g \times \mathbf{z}_p$$

in the case $\|\mathbf{z}_p \times \hat{\mathbf{x}}_R\| = 0$, i.e., the two vectors are parallel, then let $\mathbf{w}_g = 0$.

C. Optical-Flow Calculation

To implement the guidance law we require the movement of the target object across the missile's FOV. An optical-flow algorithm calculates the movement of the image "brightness pattern" across the FOV. Technically, these values are not the same, however if the target illumination is sufficiently consistent, and the target image has sufficiently rich detail, they will be close. A good survey of the issues and techniques of optical-flow calculations can be found in [14].

The intended purpose of many of these algorithms is a dense vector field describing the optical flow of each point in the image. For our purpose, measuring the motion of the target, such detail is not necessary. A single vector approximating the average flow for the region of the image covered by the target is sufficient.

We use an algorithm of Srinivasan's [12], which is broadly similar to local differential methods, such as the Lucas Kanade algorithm, except that differences between shifted images are used as approximate differentials, instead of a derivative filter. This algorithm provides such an average for a region of any size, and computationally only requires a few summations and the solution of a two-by-two system of linear equations. In contrast, global algorithms giving a dense flow field usually require iterative solutions of a partial differential equation. We now describe the algorithm we use, following the notation of [12].

A monochrome image is recorded at discrete times $t_0, t_1, t_2, \dots$. Let $f(t_k, u, v)$ be the image at time t_k , having horizontal and vertical coordinates u and v . Two successive images are compared to find the flow that has occurred to take one to the other. We denote them $f_0(u, v) := f(t_{k-1}, u, v)$ and $f(t_k, u, v)$. From now on we drop the u and v arguments for the sake of brevity.

In the next subsection we detail a state estimator which provides estimates of the current optical flow at any time.

We choose two reference shifts $\Delta u_{\text{ref}}, \Delta v_{\text{ref}}$; then four new images, f_1, f_2, f_3, f_4 , are created by shifting f_0 by $-\Delta u_{\text{ref}}, \Delta u_{\text{ref}}, -\Delta v_{\text{ref}}, \Delta v_{\text{ref}}$, respectively.

The motivation for this is that the change from one frame to the next is approximately equal to a weighted sum of the reference shifts, so:

$$f - f_0 \approx \left(\frac{\widehat{\Delta u}}{\Delta v_{\text{ref}}} \right) (f_2 - f_1) + \left(\frac{\widehat{\Delta v}}{\Delta v_{\text{ref}}} \right) (f_4 - f_3). \quad (18)$$

The coefficients $\widehat{\Delta u}$ and $\widehat{\Delta v}$ are found which come closest to equality, where closest means smallest square-integral.

In order to focus the attention of the algorithm on the target, we weight the integrations with a Gaussian kernel function Ψ centred on the current estimate of the target location.

Then the coefficients are found by solving the following linear equations for $\widehat{\Delta u}, \widehat{\Delta v}$ [12]:

$$\begin{aligned} \left(\frac{\widehat{\Delta u}}{\Delta u_{\text{ref}}} \right) & \int \int \Psi (f_2 - f_1)^2 du dv + \\ \left(\frac{\widehat{\Delta v}}{\Delta v_{\text{ref}}} \right) & \int \int \Psi (f_4 - f_3)(f_2 - f_1) du dv \\ & = 2 \int \int \Psi (f - f_0)(f_2 - f_1) du dv, \\ \left(\frac{\widehat{\Delta u}}{\Delta u_{\text{ref}}} \right) & \int \int \Psi (f_2 - f_1)(f_4 - f_3) du dv + \\ \left(\frac{\widehat{\Delta v}}{\Delta v_{\text{ref}}} \right) & \int \int \Psi (f_4 - f_3)^2 du dv \\ & = 2 \int \int \Psi (f - f_0)(f_4 - f_3) du dv. \end{aligned}$$

We use finite shifts in image coordinates and time to approximate infinitesimal flow, so there is an analogue of the aliasing problem in signal sampling. High spatial frequency components in the image can cause errors in the calculation, so the image is first convolved with a Gaussian function, with a half-width approximately equal to the largest optical flow expected between successive images. An equivalent effect could be achieved optically by deliberately using out-of-focus images for the optical-flow calculation.

In addition, the size of the reference shifts can have considerable effect on the performance of the algorithm. The choice of shift size which gives good performance will vary with practical circumstances.

The success of the algorithm can be judged following the calculation of $\widehat{\Delta u}, \widehat{\Delta v}$. Let

$$\hat{f} := f_0 + \left(\frac{\widehat{\Delta u}}{\Delta v_{\text{ref}}} \right) (f_2 - f_1) + \left(\frac{\widehat{\Delta v}}{\Delta v_{\text{ref}}} \right) (f_4 - f_3). \quad (19)$$

This is the prediction of f based on f_0 and the calculated flow.

And consider the quantity

$$\phi(f, \hat{f}) := \frac{\int \int \Psi f \hat{f} du dv}{\sqrt{\int \int \Psi f^2 du dv} \sqrt{\int \int \Psi \hat{f}^2 du dv}}. \quad (20)$$

This is the Ψ -weighted correlation between f and $\hat{f}$. $\phi(f, \hat{f})$ is a number between 0 and 1, and we can set a threshold, below which the values of $\hat{\Delta}u$, $\hat{\Delta}v$ will not be used in the state estimator.

D. State Estimator

In this section we construct a state estimator to combine the information from the optical position measurements and the optical-flow algorithm.

We use a state estimator with a continuous state, and which uses both discrete and continuous measurements, a type referred to as a “hybrid” estimator in [10]. The reasons for this are:

- 1) It is of the nature of the optical-flow algorithm we use that measurements come at discrete instants, which may not have a fixed period of arrival,
- 2) the guidance law as defined requires continuous feedback data.

It would be straightforward to consider either continuous or discrete measurements of optical position. Depending on the target recognition algorithm used, discrete-time measurements of position, perhaps at a different rate to those of optical flow, may be most appropriate. Since we don't consider any particular algorithm, we assume position measurements are continuous for simplicity.

The system we estimate has two states: 1) the position of the target in screen coordinates $\mathbf{p} = [u \ v]'$, and 2) the component of optical flow due to relative velocity $\mathcal{O}_v$.

The total optical flow of the target on the screen can be decomposed into three components:

$$\mathcal{O} = \mathcal{O}_v + \mathcal{O}_m + w_p$$

$\mathcal{O}_v$ is the flow due to the relative velocity of missile and target, which is the quantity we need for the guidance law. The component $\mathcal{O}_m$ is due to rotation of the imaging sensor, both from missile body acceleration and from rotation of the gimbal which carries the sensor. Measurements are available of these quantities from an inertial measurement unit and from the gimbal system. If we let $\mathbf{w}$ be the angular velocity of the imaging sensor in the reference frame with $\mathbf{x}_p$, $\mathbf{y}_p$, $\mathbf{z}_p$ as a basis, then

$$\mathcal{O}_m := V\mathbf{w}$$

where V is a projection defined as

$$V := \begin{bmatrix} \frac{uv}{f} & -\left(f + \frac{u^2}{f}\right) & v \\ \left(f + \frac{v^2}{f}\right) & \frac{-uv}{f} & -u \end{bmatrix}.$$

The last term w_p is a disturbance term which represents errors in the measurements of $\mathcal{O}$ and $\mathcal{O}_m$, due to airframe vibrations and calculation errors from the algorithms used.

Since we lack a dynamical model with which to estimate what the rate of change of the optical flow is, this is denoted by another disturbance term w_d . So the dynamics of this system are

$$\dot{\mathbf{p}} = \mathcal{O} = \mathcal{O}_v + \mathcal{O}_m + w_p$$

$$\dot{\mathcal{O}}_v = w_v$$

where w_p and w_v are unknown disturbance signals.

We now give the equations for a hybrid state estimator of this system. It is of the type presented in [10].

Let s be some finite time, and let $\mathcal{J}$ be a set of discrete times, $0 \leq t_0 < t_1 < \dots < t_J \leq s$, at which optical-flow measurements occur, and $\phi(f, \hat{f})$ is above some chosen threshold. Let v_c and w be continuous time signals defined in $[0, s]$, and v_d a discrete time signal, defined at times in $\mathcal{J}$.

The state estimator equations, with a state $x_o = [\mathbf{p}' \ \mathcal{O}_v']'$ and known control input $u = \mathcal{O}_a$ are defined thus:

$$\dot{x}_o = Ax_o + B_1 w + B_2 u$$

$$y_c = C_c x_o + n_c$$

$$y_d = C_d x_o + n_d$$

$$A = \begin{bmatrix} 0_{2,2} & I_2 \\ 0_{2,2} & 0_{2,2} \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0_{2,2} \\ I_2 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -I_2 \\ 0_{2,2} \end{bmatrix}$$

$$C_c = [I_2 \ 0_{2,2}], \quad C_d = [0_{2,2} \ I_2].$$

The signals n_c and w represent errors in optical position measurement, and the derivative of $\mathcal{O}_v$, respectively. v_d represents errors in optical-flow calculation. We can represent the sizes of these signals with a constraint of the following form:

$$\int_0^s [v_c(t)' R_c v_c(t) + w(t)' Q w(t)] dt + (x(0) - x_0)' N (x(0) - x_0) + \sum_{j=0}^J v_d(t_j)' R_d n_d(t_j) \leq 1.$$

If statistical information is known about the measurement errors and dynamics of $\dot{\mathcal{O}}_v$ then R_c , R_d ,

Q , N can be defined as the inverses of the appropriate covariance matrices, but the above description also allows for deterministic forms of uncertainty.

The state estimate is then given by the following equations, using the notation that $g(t^-)$ is the value of g just before t , i.e., $g(t^-) := \lim_{\epsilon \rightarrow 0, \epsilon > 0} g(t - \epsilon)$.

$$\begin{aligned}\dot{\hat{x}}_o &= A\hat{x}_o + B_2u + PC_c'R_c(y_c - C_c\hat{x}_o) & \text{for } t \notin \mathcal{J} \\ \hat{x}_o(t) &= \hat{x}_o(t^-) + P(t^-)C_d'R_d(y_d - C_d\hat{x}_o(t^-)) & \text{for } t \in \mathcal{J} \\ \hat{x}_o(0) &= x_0\end{aligned}\quad (21)$$

where P is the solution of the following ‘‘jump-Riccati equation’’:

$$\begin{aligned}\dot{P} &= AP + PA' + B_1Q^{-1}B_1' - PC_c'R_cC_cP & \text{for } t \notin \mathcal{J} \\ P(t) &= [P(t^-)^{-1} + C_d'R_dC_d]^{-1} & \text{for } t \in \mathcal{J} \\ P(0) &= N^{-1}.\end{aligned}\quad (22)$$

E. Extraction of Information Required by the Guidance Law

From the information in the state estimator above, we can calculate the control law in any reference from in which the vectors $(\mathbf{z}_p, \mathbf{x}_p, \mathbf{y}_p)$ are known, for example, the missile’s body frame.

If the target has estimated view-plane coordinates $\mathbf{p} = (u, v)$, then the relative position vector can be reconstructed with the following equation:

$$\hat{\mathbf{x}}_R = f\mathbf{z}_p + u\mathbf{x}_p + v\mathbf{y}_p.$$

If u and v are correct, then $\hat{\mathbf{x}}_R$ is correct up to multiplication by a positive scalar (i.e., it has the correct direction) which is all that is needed by the guidance law.

Let

$$K := [\mathbf{x}_p \ \mathbf{y}_p]$$

then the component of optical flow due to relative velocity can be represented in three dimensions with

$$\mathcal{O}_{v3} := K\mathcal{O}_v$$

and the angular flow used in Section V can be reconstructed as

$$\mathbf{f}_T = \frac{\mathcal{O}_{v3} - \text{Proj}_{\hat{\mathbf{x}}_R} \mathcal{O}_{v3}}{\|\hat{\mathbf{x}}_R\|}.$$

The control signal can then be calculated as per Section V, the computer code for which is given in the appendix.

F. Autopilot Model

It is common when deriving guidance laws to assume that the controller can directly affect the acceleration of the missile. In reality, there is always some delay introduced by the dynamics of the actuators, airframe, and seeker system.

In the simulations we test the effects of unmodelled autopilot dynamics. We model such effects as a simple first-order lag. Let $\mathbf{u}_C$ be the acceleration commanded by the control law, and let $\mathbf{a}_M$ be the actual acceleration that results. Then:

$$\dot{\mathbf{a}}_M = -p_A\mathbf{a}_M + p_A\mathbf{u}_C. \quad (23)$$

VIII. SIMULATION RESULTS

The theoretical results of Section VI depend on the missile system’s ability to keep the CNG condition true, that is, make $\mathbf{v}_M$ track $\mathbf{v}_D$. Measurement errors and autopilot lag make this task more difficult. In this section we use computer simulations to analyze the performance of the guidance law when such difficulties are introduced, and compare it with the performance of another law from the literature under the same conditions.

The law we compare CNG to is biased PN guidance (BPNG), which was presented in [4] as a two-dimensional guidance problem. In order to compare it with CNG, we have adapted it to three dimensions, in keeping with the stated design philosophy behind BPNG. The next subsection describes the conversion of BPNG from 2D to 3D, and after that we give the results of the simulations.

A. Biased Proportional Navigation Guidance

The biased PN law consists of two terms. One term is the standard PN guidance, except with a time-varying navigation constant N , which nulls the line-of-sight rate of the target in the missile’s FOV. The second term is a time-varying bias which pulls the missile around the target so that it approaches from the correct angle.

These two terms are, in a sense, competing for control of the missile, as the PN term will always drive the missile towards the target, whereas the bias term pulls it away so as to change the approach angle. This philosophy is in contrast to that of CNG, in which a particular velocity and acceleration are desired which will result in the desired collision, and the missile is controlled towards this heading and velocity. Despite this, the authors of [4] prove that BPNG will result in a perfect collision, albeit under idealized circumstances and a restricted set of initial conditions with small initial heading errors.

The most significant difference between BPNG and CNG, however, is that BPNG requires knowledge of the range to the target, whereas CNG uses only target bearing information. For the missile systems we consider, which have a video camera as their primary sensor, this makes a big difference to the practicability of the guidance laws.

The BPNG guidance law in two dimensions, using the notation from Section IV, is

$$u = N\|\mathbf{v}_M\|\dot{\sigma} - \eta \frac{\|\mathbf{v}_M\|^2 \varepsilon}{\|\mathbf{x}_R\| \cos(\lambda)} \quad (24)$$

where η is a constant gain chosen by the designer, and N is a time-varying gain defined by

$$N = N' + \frac{\rho + 0.5\alpha\|\mathbf{x}_R\|}{\|\mathbf{v}_M\| \cos \lambda} \quad (25)$$

where ρ is the ratio of target speed to missile speed, and α and N' are designer chosen constants.

The motivation for the bias term is to pull the missile around the target so as to correct the approach angle. We adapt this into three dimensions by taking $\dot{\sigma}$ to be its equivalent $\mathbf{f}_T$ and the direction for it to be applied as orthogonal to the missile's heading, since all missile acceleration commands must be in this subspace, and it is normalized:

$$\mathbf{u}_{D2} = \frac{\mathbf{f}_T - \text{Proj}_{\mathbf{v}_M} \mathbf{f}_T}{\|\mathbf{f}_T - \text{Proj}_{\mathbf{v}_M} \mathbf{f}_T\|}. \quad (26)$$

The bias term is calculated in the following way: The desired final impact vector $\mathbf{v}_F$ is projected onto the subspace orthogonal to $\mathbf{x}_R$, and then reversed:

$$\mathbf{v}_F^* = -(\mathbf{v}_F - \text{Proj}_{\mathbf{x}_R} \mathbf{v}_F). \quad (27)$$

This vector $\mathbf{v}_F^*$ is the direction in which the missile should move, in order to rotate the relative position vector $\mathbf{x}_R$ towards that from which the missile should approach the target. And the direction for it to be applied is, similar to above:

$$\mathbf{u}_{D3} = \frac{\mathbf{v}_F^* - \text{Proj}_{\mathbf{v}_M} \mathbf{v}_F^*}{\|\mathbf{v}_F^* - \text{Proj}_{\mathbf{v}_M} \mathbf{v}_F^*\|}. \quad (28)$$

Now the complete three-dimensional guidance law can be given:

$$\mathbf{u}_c = N\|\mathbf{v}_M\|\|\mathbf{f}_T\|\mathbf{u}_{D2} + \eta \frac{\|\mathbf{v}_M\|^2 \text{Angle}(\mathbf{v}_F, \mathbf{x}_R)}{\|\mathbf{x}_R\| \cos[\text{Angle}(\mathbf{v}_M, \mathbf{x}_R)]} \mathbf{u}_{D3} \quad (29)$$

where

$$N = N' + \frac{\rho + 0.5\alpha\|\mathbf{x}_R\|}{\|\mathbf{v}_M\| \cos[\text{Angle}(\mathbf{v}_M, \mathbf{x}_R)]}. \quad (30)$$

B. Comparative Simulations

In this section we compare the performance of CNG and BPNG. We assume that the signals $\mathbf{x}_R$ and $\mathbf{f}_T$ used in the control laws are corrupted by noise. In the following section we give results of simulations using a realistic image sequence and Srinivasan's optical-flow algorithm.

We now describe the noise on the measurement of $\mathbf{x}_R$. Let $\mathbf{w}_p(t)$ be a random vector signal, with each component having zero mean and covariance σ_p^2 and

TABLE I
Numerical Values for Simulation

Variable	Value
$\mathbf{x}_R(0)$	$[1000 \ 0 \ -200]'$
$\mathbf{v}_T(0)$	$[0 \ 0 \ 0]'$
$\mathbf{v}_F(0)$	$[\sin \theta \ -0.1 \ -\cos \theta]'$
$\ \mathbf{v}_M\ $	300
k_p	1
k_g	20
T_A	0.2
σ_p^2	0.005
σ_f^2	0.05

a bandwidth of 10 Hz, and let w_i represent the i th element of $\mathbf{w}_p$. We form the following matrices:

$$W_p := \begin{bmatrix} 0 & -w_3 & w_2 \\ w_3 & 0 & -w_1 \\ -w_2 & w_1 & 0 \end{bmatrix} \quad (31)$$

$$R_p := \exp(W_p) = I + \frac{W_p}{\|\mathbf{w}_p\|} \sin(\|\mathbf{w}_p\|) + \frac{W_p^2}{\|\mathbf{w}_p\|^2} (1 - \cos(\|\mathbf{w}_p\|)). \quad (32)$$

W_p is a matrix form of the linear operation of taking a cross product with $\mathbf{w}_p$, i.e. $W_p \mathbf{x} = \mathbf{w}_p \times \mathbf{x}$. R_p is a rotation matrix, representing integrated rotation with an angular velocity $\mathbf{w}_p$. If $\mathbf{w}_p$ is small, R_p is close to the identity (see [15, p. 27]).

The noisy measurement of optical position $\mathbf{x}_R$ is then simulated as

$$\hat{\mathbf{x}}_R = R_p \mathbf{x}_R. \quad (33)$$

The optical-flow measurements are perturbed by $\mathbf{n}_f(t)$, a random vector signal with zero mean and covariance σ_f^2 and bandwidth 10 Hz, projected on to the subspace orthogonal to $\mathbf{x}_R$. That is, the measurement of $\mathbf{f}_T$ is

$$\hat{\mathbf{f}}_T = \mathbf{f}_T + (\mathbf{n}_f - \text{Proj}_{\mathbf{x}_R} \mathbf{n}_f). \quad (34)$$

The parameters listed in Table I were used in all simulations, except where otherwise noted. The angle θ is a random angle with mean value $\pi/4$ and covariance $\pi/12$. The gains for BPNG were $N' = 2.9$, $\alpha = 0.28$, $\eta = 1.3$, identical to those chosen for the simulations in [4].

The first set of simulations we performed was to establish the effect of unmodelled dynamics in the guidance loop. All the simulation parameters are as in Table I except the time constant of the autopilot $T_A = 1/p_A$, which is varied over the range from 0 (the control law instantaneously controls the missile acceleration) to 1 s.

The results for miss distances and angle errors for CNG and BPNG are displayed in Fig. 10. As can be

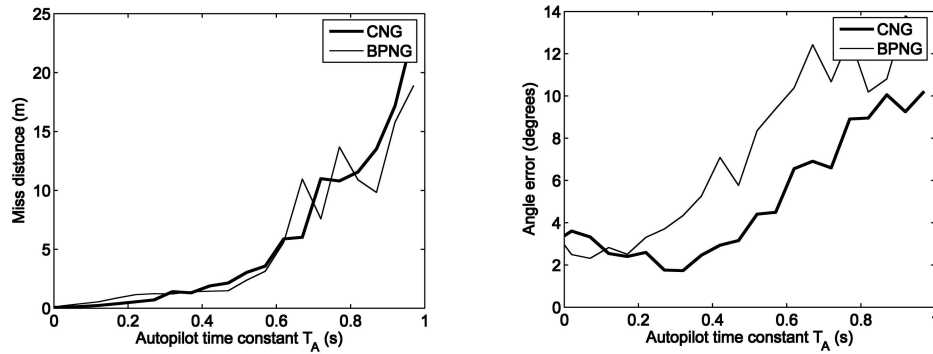


Fig. 10. Performance comparison of CNG and BPNG with unmodelled autopilot dynamics.

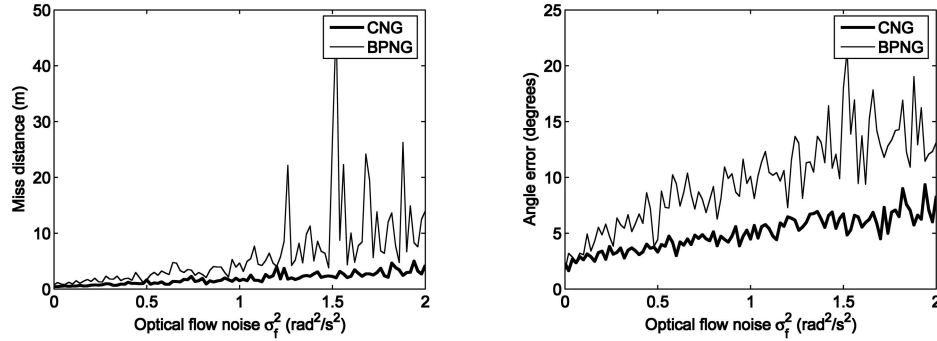


Fig. 11. Performance comparison of CNG and BPNG with noise in optical-flow calculations.

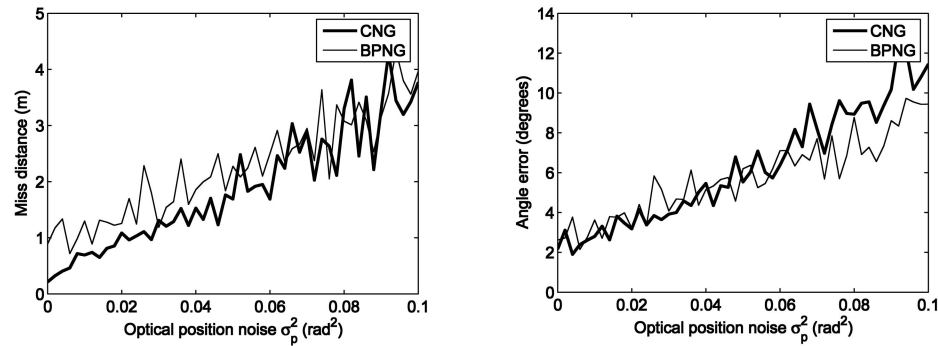


Fig. 12. Performance comparison of CNG and BPNG with noise in optical position calculations.

seen, the two guidance laws performed comparably. In terms of final-angle error, CNG appears slightly more robust to large autopilot time constants.

In the second set of simulations, the variance of the error in optical-flow measurements, σ_f^2 is varied over a range of values. The results are presented in Fig. 11. It is clear that CNG is significantly more robust to this type of noise than BPNG.

In the third set of simulations, the variance of the error in f , the target's position vector measurements σ_p^2 , is varied over a range of values. The results are plotted in Fig. 12; it is clear that both laws have comparable robustness to this sort of noise.

These simulations indicate that the two guidance laws have similar robustness to measurement errors and autopilot dynamics, with CNG giving slightly better performance. In particular CNG appears more

robust to errors in the measurement of optical flow. This is significant given the difficulties inherent in calculating optical flow from a sequence of video images.

C. Simulations with Optical-Flow Algorithm

To test the use of the optical-flow algorithm, we implemented a flight simulator, projecting a texture-mapped flat ground plane onto the missile camera plane. The image used for the ground map is similar to those in Fig. 5. Srinivasan's algorithm was run on successive images from this model, and its output incorporated in the jump-Kalman filter. The position of the target in the camera plane was assumed available, but corrupted by noise, as in the previous section.

All parameters were the same as in Table I except with $\sigma_p^2 = 0.01$ and $\mathbf{v}_F = [1 \ -0.1 \ -1]'$ and $T_A = 0.1$. The threshold for use of optical-flow estimates was $\phi(f, \hat{f}) = 0.85$. The reference shifts used in the optical-flow algorithm, $(\Delta u_{\text{ref}}, \Delta v_{\text{ref}})$, were chosen at each point in time to be twice the current estimate of the optical flow, multiplied by the time interval between the images. This was found to give good performance in our simulations, but performance in practice may vary. The parameters of the estimator were as follows:

$$\begin{aligned} P(0) &= 0_{4,4}, & Q &= \begin{bmatrix} 10I_2 & 0_{2,2} \\ 0_{2,2} & 0.1I_2 \end{bmatrix} \\ R_c &= 50I_2, & R_d &= 50I_2. \end{aligned} \quad (35)$$

BPNG requires range information. In order to test CNG against the most stringent standards, BPNG was provided with perfect knowledge of range in our simulations, although in practice this would be very difficult to attain with a video camera.

Fig. 13 shows trajectories that missiles take using CNG and BPNG. The end point of each is very similar, as both achieved the goal, but the paths they take are somewhat different. BPNG takes a more direct path towards the target early in the intercept, and then quickly turns around to the right approach angle. CNG, on the other hand, deviates more sharply early on, and then takes a smoother path to the target in the final stages.

Fig. 14 shows the magnitude of the missile acceleration for the two guidance laws. As might be expected from the analysis of the trajectories, BPNG has a lower acceleration in the early stages, but requires higher acceleration towards the end of the intercept, very sharply increasing immediately before impact. In contrast, CNG has larger acceleration early in the intercept, as $\mathbf{v}_M$ is driven towards $\mathbf{v}_D$, and lower acceleration later in the intercept. If a perfect circular path were followed, the acceleration in the late stages would be constant.

In Fig. 15 we see the angle between the missile velocity $\mathbf{v}_M$ and the desired final velocity $\mathbf{v}_F$ for each guidance law. Both go towards zero quite smoothly, however inspection of this figure and the previous figure indicate that the larger acceleration required by BPNG towards the end of the intercept, combined with the unmodelled autopilot dynamics, result in an overshoot of zero, although it may be possible to reduce this effect by tuning the gains in BPNG.

We now examine the effectiveness of the image-processing and estimation systems for a typical trajectory of CNG. Figs. 16 and 17 show the horizontal and vertical positions of the target in the video camera's imaging plane. Also plotted are the estimates of this value using the estimator from Section VIID. Fig. 18 plots the correlation between the estimated and actual images in the optical-flow

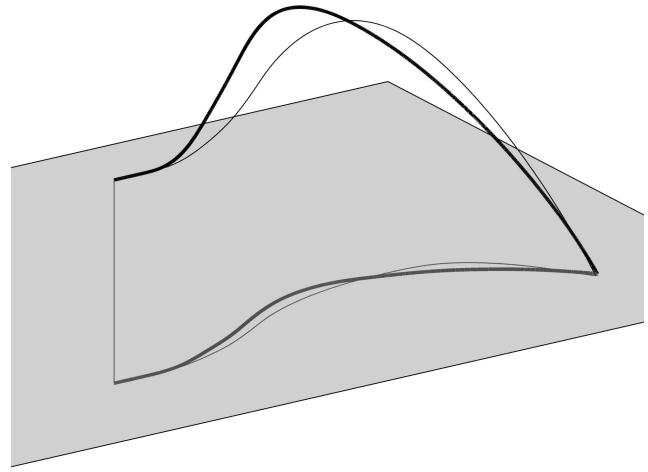


Fig. 13. Trajectory of missiles using Srinivasan's optical-flow algorithm. Thick line is CNG, thin line BPNG.

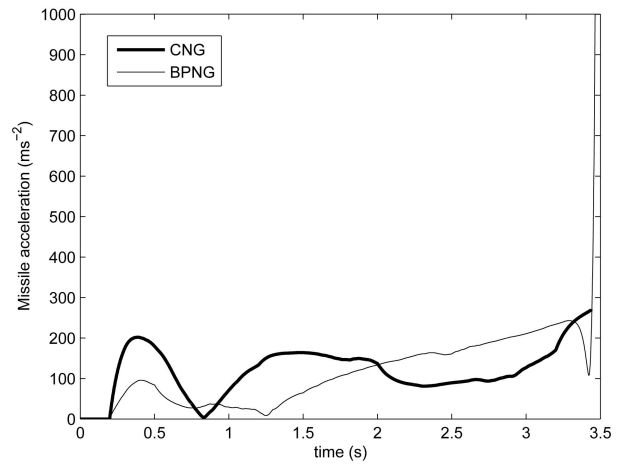


Fig. 14. Missile acceleration resulting from each guidance law versus time.

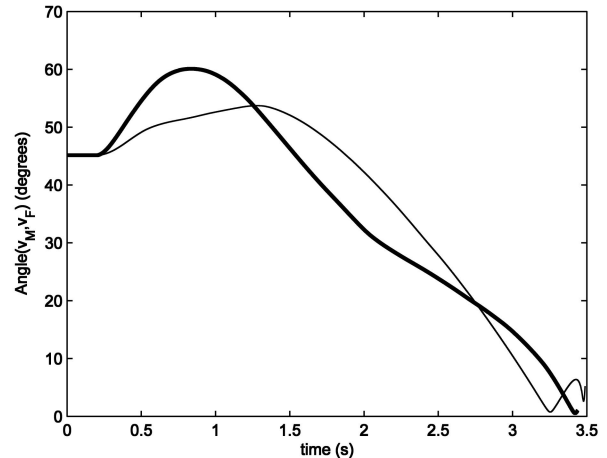


Fig. 15. Angle between missile velocity and desired final missile velocity versus time.

algorithm, $\phi(f, \hat{f})$ from (20). A threshold of 0.85 was set, above which the optical-flow estimate is used in the observer, below which it is not. This threshold has been plotted in Fig. 18 as a dotted line.

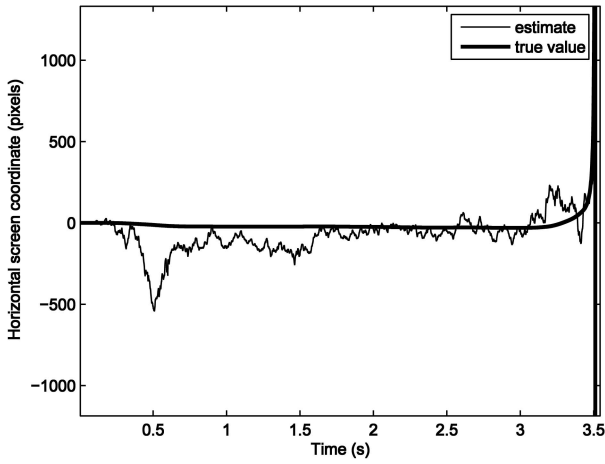


Fig. 16. Optical position of target, and estimate of same, horizontal coordinate.

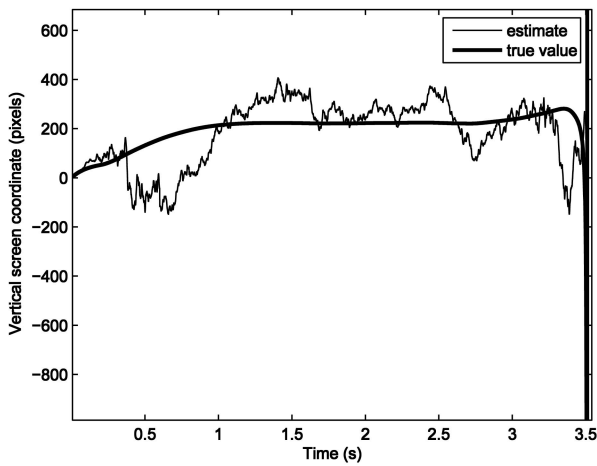


Fig. 17. Optical position of target, and estimate of same, vertical coordinate.

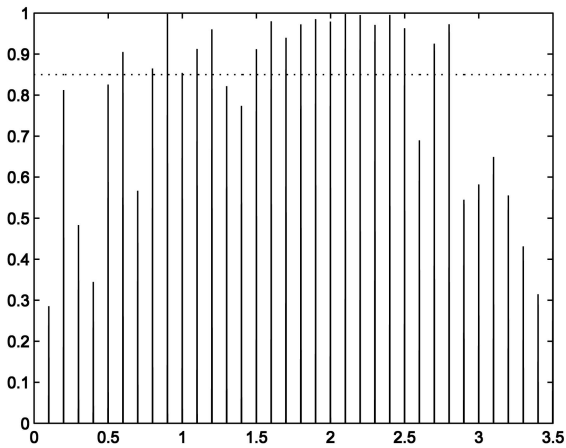


Fig. 18. Correlation of optical-flow estimate $\phi(f, \hat{f})$ and threshold of 0.85.

We can observe in Fig. 17 at just after 1.5 s, and just before 2.5 s, that the estimate of the optical position of the target has drifted away from the true value, but jumps due to the optical-flow algorithm bringing it back. It can be seen from comparing

Figs. 16, 17, and 18 that at times when $\phi(f, \hat{f})$ is close to 1, the estimates of optical position are very good.

IX. CONCLUSIONS

In this paper, the method of CNG has been developed for three-dimensional intercepts. We have proven that, in the ideal case, it will give perfect intercepts against stationary targets, and achieves zero miss distance and a known angle error against constant-velocity targets.

Furthermore, it was shown how the law may be implemented on a platform with a video camera as a primary sensor. In comparative simulations, it was shown to perform as well as, or better than, a similar law from the literature, despite requiring less information. That CNG does not require knowledge of the range to the target is an important benefit.

APPENDIX. MATLAB CODE FOR CONTROL LAW

In this appendix we give MATLAB code for the control algorithm. It uses the measurement outputs: $\mathbf{x}_R$, of which only the orientation is used (so any positive scalar multiple of $\mathbf{x}_R$ will work), and $\mathbf{f}_T$. It also uses the desired final velocity $\mathbf{v}_F$ and the missile's current velocity $\mathbf{v}_M$. The gain k_p is a parameter to be chosen by the designer. Refer to Section V for discussion of the reasoning behind the algorithm.

The functions `proj` and `vecAng` have been implemented as per `Proj` and `Angle`, respectively, from Section II.

```
function uc = calc_u(xR,vF,vM,fT,kp)

% Calculate desired velocity -l up
vD = 2*proj(vF,xR)-vF;
e_lambda = vecAng(vM,vD);
uD1 = (vD-proj(vD,vM))/norm(vD-proj(vD,vM));
up = kp*norm(vM)*e_lambda*uD1;

% Values needed for calculation of optical flow
theta = vecAng(vF,xR);
xRn = xR/norm(xR);

% Calculate normalized coordinate
% vectors at xR: duTn,dvTn
dvT = -(vF-proj(vF,xR));
dvTn = dvT/norm(dvT);
duTn = cross(xRn,dvTn);

% Calculate normalized coordinate
% vectors at vD: duDn, dvDn
dvD = -(xR-proj(xR,vD));
dvDn = dvD/norm(dvD);
vDn = vD/norm(vD);
duDn = cross(vDn,dvDn);
```

```

%% Get components of fT in terms of the
%% basis duTn, dvTn
KT = [duTn dvTn];
VcT = inv(KT'*KT)*KT';
flowT = VcT*fT;

%% Calculate flow of vD in terms of the
%% basis duDn, dvDn
flowD = [flowT(1)*2*cos(theta);flowT(2)*2];

%% Transform this desired flow back to 3d space
KD = [duDn dvDn];
vDdot = KD*flowD;

%% Express it as an angular-velocity cross
%% product and calculate uf
w = cross(vD,vDdot);
uf = cross(w,vM);

%% add uf and up to get control signal uc
uc = up+uf;
%% End of code.

```

REFERENCES

- [1] Kim, M., and Grider, K. V.
Terminal guidance for impact attitude angle constrained flight trajectories.
IEEE Transactions on Aerospace and Electronic Systems, **9**, 6 (Dec. 1973), 852–859.
- [2] Manchester, I. R., and Savkin, A. V.
Circular navigation guidance law for precision missile target engagements.
Journal of Guidance, Control, and Dynamics, **29**, 2 (Mar.–Apr. 2006), 314–320.
- [3] Manchester, I. R., and Savkin, A. V.
Circular navigation guidance law with incomplete information and uncertain autopilot model.
Journal of Guidance, Control, and Dynamics, **27**, 8 (Nov.–Dec. 2004), 1078–1083.
- [4] Kim, B. S., Lee, G. L., and Han, H. S.
Biased PNG law for impact with angular constraint.
IEEE Transactions on Aerospace and Electronic Systems, **34**, 1 (Jan. 1998), 277–288.
- [5] Song, T. L., Shin, J. S., and Han, H. S.
Impact angle control for planar engagements.
IEEE Transactions on Aerospace and Electronic Systems, **35**, 4 (Oct. 1999), 1439–1444.
- [6] Song, T. L., and Shin, J. S.
Time optimal impact angle control for vertical plane engagements.
IEEE Transactions on Aerospace and Electronic Systems, **35**, 2 (Apr. 1999), 738–742.
- [7] Savkin, A. V., Pathirana, P. N., and Faruqi, F. A.
The problem of precision missile guidance: LQR and $\mathcal{H}_\infty$ frameworks.
IEEE Transactions on Aerospace and Electronic Systems, **39**, 3 (2003), 901–910.
- [8] Malyavej, V., Manchester, I. R., and Savkin, A. V.
Precision missile guidance using radar/multiple-video sensor fusion via communication channels with bit-rate constraints.
Automatica, **42**, 5 (May 2006), 763–769.
- [9] Petersen, I. R., Ugrinovskii, V. A., and Savkin, A. V.
Robust Control Design using H^∞ Methods.
London: Springer-Verlag, 2000.
- [10] Petersen, I. R., and Savkin, A. V.
Robust Kalman Filtering for Signals and Systems with Large Uncertainties.
Boston: Birkhauser, 1999.
- [11] Savkin, A. V., and Petersen, I. R.
Recursive state estimation for uncertain systems with an integral quadratic constraint.
IEEE Transactions on Automatic Control, **40**, 6 (June 1995), 1080–1083.
- [12] Srinivasan, M. V.
An image-interpolation technique for the computation of optic flow and egomotion.
Biological Cybernetics, **71** (1994), 401–415.
- [13] Nagle, M. G., and Srinivasan, M. V.
Structure from motion: Determining the range and orientation of surfaces by image interpolation.
Journal of the Optical Society of America A, **13**, 1 (1996), 25–34.
- [14] Beauchemin, S. S., and Barron, J. L.
The computation of optical flow.
ACM Computing Surveys, **27**, 3 (1995), 433–467.
- [15] Ma, Y., Soatto, S., Košecák, J., and Sastry, S. S.
An Invitation to 3-D Vision.
New York: Springer-Verlag, 2004.



Ian R. Manchester was born in Sydney, Australia, in 1979. He completed the B.E. (Hons 1) degree in 2001 and the Ph.D. degree in 2005, both in electrical engineering from the University of New South Wales, Sydney, Australia.

Since 2005 he has held a post-doctoral position in the control systems group at the Department of Applied Physics and Electronics, Umeå University, Sweden. In Oct./Nov. 2006 he was a visiting researcher at the Intelligent Systems Research Center, Sungkyunkwan University, Suwon, South Korea. His current research interests include vision-based guidance and robotics, control of underactuated and non-holonomic systems, and modelling and identification of the human cerebrospinal fluid system.

Dr. Manchester has published several journal and conference articles in the areas of his research interests.



Andrey V. Svakin was born in 1965 in Norilsk, USSR. He received the m.S. degree (1987) and the Ph.D. degree (1991) from the Leningrad University, USSR.

From 1987 to 1992, he worked in the All-Union Television Research Institute, Leningrad. From 1992 to 1994, he held a postdoctoral position in the Department of Electrical Engineering, Australian Defence Force Academy, Canberra. From 1994 to 1996, he was a research fellow with the Department of Electrical and Electronic Engineering and the Cooperative Research Center for Sensor Signal and Information Processing at the University of Melbourne, Australia. From 1996 to 2000, he was a senior lecturer, and then an associate professor with the Department of Electrical and Electronic Engineering at the University of Western Australia, Perth. Since 2000, he has been a professor with the School of Electrical Engineering and Telecommunications, The University of New South Wales, Sydney. His current research interests include robust control and filtering, hybrid dynamical systems, missile guidance, networked control systems, computer-integrated manufacturing, control of mobile robots, computer vision, and application of control and signal processing to biomedical engineering and medicine.

Dr. Svakin has published four books and numerous journal and conference papers and served as associate editor for several international journals and conferences.