Simulating Aerial Targets in 3D Accounting for the Earth's Curvature

DAVID FREDERIC CROUSE

Most dynamic models for target tracking are derived for a flat Earth. When simulating the trajectories of targets over long ranges, the curvature of the Earth becomes important. This paper demonstrates how most nonlinear, 3D, continuous-time, flat-Earth dynamic models can be easily adapted for use on a curved Earth without incorrectly spiraling into the poles as is typical of previous techniques. The algorithm is demonstrated with flat-Earth non-maneuvering, turning, weaving, and spiraling dynamic models. The underlying technique can also compute geodesic curves on or above the Earth in a singularity-free manner when used with a non-maneuvering dynamic model.

THE QUICK AND DIRTY IMPLEMENTATION

The mapping of a flat-Earth, continuous-time dynamic model to a curved-Earth is embodied in the rewritten drift function of (33), which rewrites the function in terms of new, rotating bases. Readers who want to quickly map continuous-time flat-Earth dynamic models quickly to an ellipsoidal Earth approximation without going through the details of the mathematics can skip to the algorithmic summary in Fig. 6. The flat-Earth dynamic model should consist of, at a minimum, position and velocity components as in (6).

I. INTRODUCTION

The modeling of aerospace vehicle dynamics with varying degrees of freedom and accounting for the Earth's gravity has been extensively studied, for example in [84], where a number of models are presented. However, such curved-Earth target models often require knowledge about basic aircraft aerodynamic parameters, such as the lift, thrust, and drag. On the other hand, target tracking algorithms in radar systems must be able to maintain track on targets with known aerodynamic models as well as on targets that cannot be identified. Thus, a number of lower-fidelity dynamic models, such as those surveyed in [40] and [41], are commonly used in tracker simulations. These low-fidelity simulations can often be defined using a number of simple parameters, like the limits on the speed, turn rate, range and ceiling of three well-known fighter aircraft shown in Table I. The simulations in Section V use these bounds to justify the realism of the simulated scenarios. Unfortunately, most of the non-ballistic models in [40] as well as the ballistic models in [41] and [83, Ch. 8, 9] are made ignoring the curvature of the Earth. When tracking targets using networked radars, the curvature and undulation of the Earth can become significant. For example, if a flat-Earth dynamic model is propagated over a long distance, a target in level flight under a flat-Earth dynamic model will tangentially depart a curved Earth. This paper derives a technique for using arbitrary, nonlinear, continuous-time, deterministic, flat-Earth dynamic models on a curved Earth that mitigates this error.

TABLE I
A number of basic parameters for Russian MiG-29, the Chinese and Pakistani FC-1, and the American F-16 fighter aircraft are taken from [30] and [75] (No specific model of F-16 is used). The parameters listed above are not sufficient for high-fidelity modeling of the aerodynamics of the aircraft, but can set bounds for simple models when testing tracking algorithms.

	MiG-29	FC-1	F-16
Positive G-Force Limit	9 G	8.5 G	9 G
Maximum Speed (At Altitude)	679 m/s	347 m/s	670 m/s
Service Ceiling	17 km	16 km	$>$ 15 km
Maximum Range	1,500 km	2,037 km	$>$ 3,222 km

Manuscript received July 16, 2014; revised October 29, 2014; released for publication October 30, 2014.

Refereeing of this contribution was handled by Huimin Chen.

This paper is an extended version of the conference paper [11].

Author's address: Naval Research Laboratory, Radar Division, Code 5344, 4555 Overlook Ave., SW, Washington, DC 20375-5320, USA, e-mail: (david.crouse@nrl.navy.mil).

1557-6418/15/$17.00 © 2015 JAIF

The best-known published attempts to adapt flat-Earth dynamic models for use on a curved Earth for target tracking are [43] and [81], where flat-Earth discrete-time dynamic models are applied in local East-North-Up (ENU) coordinate systems that rotate as the aircraft moves above the surface of the Earth.[1] However, such a constant-bearing approximation is only valid when going East-West or North-South, but not at any other bearings, and the approximation becomes extremely bad near and at the poles. Considering that the sea ice in the Northwest Passage sufficiently melted for the first time in recorded history in 2007 to allow travel through the passage [8], it is increasingly likely that ships (possibly carrying radars) will regularly approach the North Pole in the future.

On the other hand, the curvature of an ellipsoidal Earth has been properly handled in the design of inertial navigation systems, where a wander coordinate system, also known as a wander frame, the wander azimuth coordinate frame, and the wander azimuth navigation frame, is often maintained in place of a local East-North-Up coordinate system. Such a coordinate system keeps one axis aligned with the gravitational (or ellipsoidal) vertical, and the other axes are kept in the local tangent plane, without any particular relation to true North. The wander coordinate system is used in [27], [46] for strapdown inertial navigation systems, where gyroscopes are fixed to the body of an aircraft rather than remaining on a platform that tries to physically rotate to maintain a constant orientation to true North which is increasingly difficult as one approaches the poles.[2] The wander coordinate system is used in [50] for simulating aircraft trajectories under continuous-time dynamic models on a curved Earth, where solutions for four dynamic models are provided. More recently, the method of [50] was rediscovered and applied to curved-Earth target simulation in [62], [63], where the mapping of a flat-Earth model to a curved-Earth is referred to as a "geodetic projection." In this paper, a method of mapping flat-Earth dynamic models to a curved Earth is derived in vector notation, generalized to describe motion on an arbitrary terrain or above an ellipsoidal Earth. Under an ellipsoidal Earth model, when considering similar dynamic models, the algorithm of this paper reduces to the solutions of [50], [62].

In [73, Sec. 5], similar physical principles to those used in the models of [84, Ch. 8, 9, 10] (which require a detailed description of the flight dynamics of the aircraft) are used to obtain simplified dynamic expressions for straight, level flight, for steady turns, and for pitching flight (weaving up and down) above a spherical Earth. However, no expressions for spiraling (corkscrewing) targets are given, with the justification being that the models would have to be overly simplified for a solution to be obtained. In the following sections, it is shown that most flat-Earth dynamic models can be plugged into a simple algorithm to adapt them for use above a curved Earth, including a spiraling model derived in Appendix E.

A constant-bearing course means that a target maintains the same direction with respect to a local ENU coordinate system. As described in [1], a constant-bearing course is called a rhumb, whereas the curve on the surface of the Earth traced out by such a course is a loxodrome. An example of a constant-bearing coarse is a ship traveling 20° North of East. A constant-bearing course is a straight line on a Mercator projection map. However, as clearly illustrated in Fig. 7(c) in Section V and as discussed in detail in [1], constant-bearing courses that do not follow East-West or North-South trajectories spiral infinitely as an object approaches the poles. The algorithms of [43] and [81] cause straight-line targets to follow rhumb lines. A rhumb is seldom the shortest trajectory between any two points and a navigator must constantly update a vehicle's bearing to travel the shortest path.[3]

Before taking the time to implement a complicated curved-Earth simulation routine, one should first check whether the scale of the bias accumulated as a result of ignoring the Earth's curvature matters. As derived in [10], a simple closed-form approximation for the bias, Δ, accumulated using a local flat-Earth approximation can be derived using a spherical Earth approximation,

$$\Delta \approx \sqrt{a^2\left(1-\cos\left[\frac{L}{a}\right]\right)^2+\left(L-a\sin\left[\frac{L}{a}\right]\right)^2}, \quad (1)$$

where L is the distance traveled and a is the radius of the spherical Earth. For example, using the semi-major axis $a = 6{,}378{,}137.0$ m from the reference ellipsoid of the U.S. Department of Defense's (DoD) World Geodetic System 1984 (WGS-84) standard [13], a target traveling five kilometers in a straight line would accumulate a bias of only 1.96 m. On the other hand, a target traveling 500 km would accumulate a bias of 19.6 km, which is probably significant in most instances.

Considerable research over the years has focused on following the shortest and straightest path along a curved Earth or other surface. It is reasonable to assume that a target obeying a flat-Earth dynamic model on a curved Earth experiences no accelerations in the local tangent plane to the surface beyond those present in the flat-Earth model. That is, the target is not pulled left or right by any accelerations that are not part of

[1] When considering only straight-line trajectories, algorithms such as [19] are also relevant. However, this paper considers more general dynamic models.

[2] Gyroscopic systems can infer the direction of true North based on the precession of the gyroscope as the Earth rotates, as observed by Foucault in 1852 [22].

[3] For example, a rhumb-line-based navigation technique was implemented for operational use in the U.S. Navy's Navigation Sensor System Interface (for ships) when handling optical celestial observations in the 1990s [31], [32]. However, in the presence of other navigation sources (for example, Global Positioning System [GPS] satellites) and near the poles, one would assume that navigation software would take the shortest path rather than following a rhumb line.

the flat-Earth model, but it must be tipped as it moves so that the vertical axis of its local coordinate system remains aligned with the direction of acceleration due to gravity.[4]

A geodesic curve[5] between two points on a surface is the locally shortest path between those points constrained to the surface. A geodesic curve also represents a curve of zero tangential acceleration as discussed, for example in [60]. Geodesics are studied in the mathematical field of differential geometry, more information on which is available in [15, Sec. 4-4], among many other sources. As discussed in Section II, a target lacking acceleration in the local tangent plane under a constant-velocity flat-Earth model mapped to a curved Earth will follow a geodesic curve.

A common method of describing the curvature or shape of the Earth is in terms of variations in the gravitational field of the Earth. For example, altitudes on maps are usually taken with respect to mean sea level (MSL), which is a hypothetical surface of constant gravitational potential. Geodesy is the study of the shape of the Earth, especially in terms of its gravity potential. A brief overview of geodetic concepts is given in [42], with a more thorough introduction provided in the text [29]. Section IV describes the gravitational shape of the Earth, focusing on a simple ellipsoidal approximation that is used in the development of the flat-Earth-to-curved-Earth mapping algorithm of this paper.

An ellipse of revolution is the three-dimensional surface formed by rotating an ellipse around its semi-major axis . The shape of the Earth is approximately an ellipse of revolution formed by rotating an ellipse about its minor axis. This type of an ellipsoidal is often called an oblate spheroid.[6] On the other hand, an ellipse of rotation formed by rotating an ellipse about its major axis is known as a prolate spheroid. The literature often refers to an approximation of the Earth's shape using an ellipse of revolution as an "ellipsoidal approximation." For the purposes of navigation, a lot of work has been done studying geodesic curves using an ellipsoidal approximation of the Earth. Though navigation literature often uses circular or elliptical curves to approximate geodesic paths on an ellipsoidal Earth, geodesic curves are generally neither elliptical nor circular as is discussed in [79], and this paper does not use circular or elliptical approximations when mapping flat-Earth trajectories to a curved Earth.

Navigators often study the direct and indirect geodetic problems. In the direct geodetic problem, one must determine the ending point of a trajectory when given a starting point, an initial bearing, and a distance. Though the primary focus of this paper is the adaptation of flat-Earth motion models to a curved Earth, the algorithm in this paper can also be used with a non-maneuvering motion model to *solve the direct geodetic problem*, providing Cartesian coordinates of points along the trajectory. Unlike existing algorithms for solving the direct geodetic problem, which are constrained to the surface of the Earth, the algorithm of this paper can solve the problem for arbitrary constant altitudes above the Earth. The algorithm of this paper *does not solve the indirect geodetic problem*. However, algorithms that solve the indirect geodetic problem can be used as a basis for confirming the validity of the direct geodetic solutions obtained by the algorithm in this paper.

In the indirect geodetic problem, one is tasked with finding the shortest path between two points on the surface of the Earth. Algorithms that solve the indirect geodetic problem, which is to determine the initial heading and distance traveled to go from one point (a) to another point (b) along the shortest geodesic path, are commonly used with an ellipsoidal Earth approximation in navigation applications. Since non-maneuvering, level, flat-Earth motion models follow geodesic curves (solving the direct geodetic problem), the validity of the algorithm developed in this paper can be verified for level flight using the solution to the indirect geodetic problem. That is, an initial target state can have position components corresponding to point (a) and an initial velocity vector that points in the direction of the solution to the indirect geodetic problem from (a) to (b). If the model is propagated forward in time such that it covers a distance equal to the distance between (a) and (b) found when solving the indirect geodetic problem, then the target should be located at point (b). Any deviation from point (b) would indicate an error in the target propagation algorithm or numerical precision limitations.

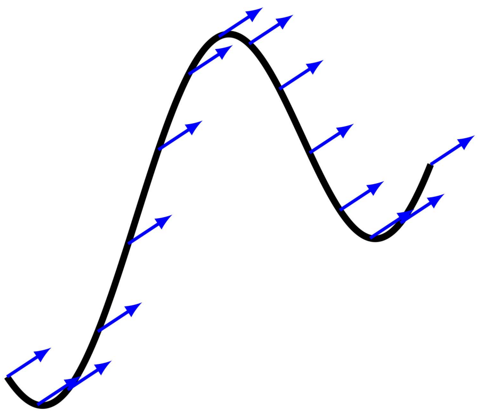
A detailed look at a number of traditional methods for solving the direct and indirect problems is given in [61], including a derivation of Vincenty's equation in [12], which is one of the most commonly used techniques, but for which a derivation is lacking in Vincenty's paper [77]. With modern computers, differential equation solvers can be used to solve both the direct and indirect problems of geodesics [38], [57], [64], [74], and accurate iterative techniques are also available [34], [35].

The majority of the algorithms in the literature for solving the indirect geodetic problem are not suitable for use in any mission-critical navigation system, because they will fail if two points lie nearly on the same meridian (have the same longitude or longitudes approximately 180° apart). However, one robust algorithm

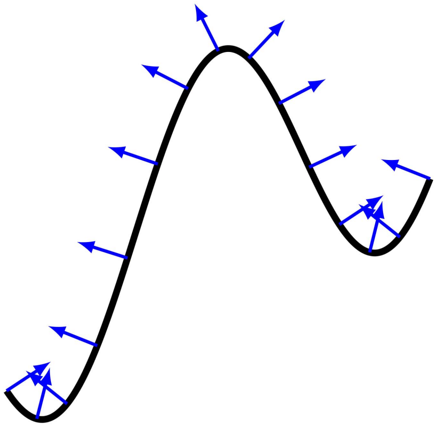
[4] In some instances, not considered in this paper, one might also wish to account for the Coriolis force. The Coriolis force is a small, velocity-dependent force to which a moving object on the surface of the Earth appears to be subjected when viewed from an Earth-fixed coordinate system (a derivation is given in [69, Ch. 7]).

[5] A geodesic curve on a sphere or an ellipsoid is sometimes referred to as an orthodrome (in contrast to a loxodrome). However, the term orthodrome is more frequently applied to geodesic curves on the surface of spheres rather than general ellipsoids.

[6] A more generally shaped ellipsoid is known as a triaxial ellipsoid.



(a) Parallel Transport



(b) Darboux Frame

Fig. 1. In (a), a vector is parallel transported along a path in a plane. At all points along the path, the vector remains parallel to the original vector. On the other hand, a vector in the Darboux frame in a plane rotates with the changing direction of the curve as shown in (b). Thus, a vector that is initially perpendicular to the curve remains perpendicular to the curve.

is given in [34], [35]. This algorithm is convenient to use both because it is well documented and because code implementing it is available for Matlab at [36] and for C, Fortran, Java, and a number of other programming languages at [37]. The algorithm provides an initial heading to reach the destination as well as the geodesic distance between the points. However, the geodesic curves used in the simulation section (for validating the solution to the direct geodetic problem provided by the algorithm of this paper) are calculated using the algorithm of [57]. Although the algorithm of [57] is not suitable for mission-critical navigation, it provides the trajectory in a format that can be easily plotted.

The examples in this paper focus on the use of an ellipsoidal Earth approximation, because it is simple and because higher-order gravitational approximations would not be expected to provide more accurate predictions, as air pressure variations begin to dominate at higher precisions. A commercial aircraft that flies at a constant instrument "altitude" reading varies its true height above the ground and above mean sea level, because most commercial aircraft operate in regions where their altitudes are computed based on a reference air pressure and not on current meteorological conditions.[7] As a result, a plane that is flying at a constant altitude according to its own instruments does not fly a precise geodesic curve.

Using the data offered by [52], the WGS-84 geoid, a higher-order approximation than the reference ellipsoid for gravity at sea level, differs from the reference ellipsoid up to $+84$ m and -107 m, which is less than one flight level used by air traffic control in the United States [21].[8] The geoid undulations are less than altitude variations one would expect to see in an aircraft due to changing meteorological conditions, which are quantified in Appendix A.

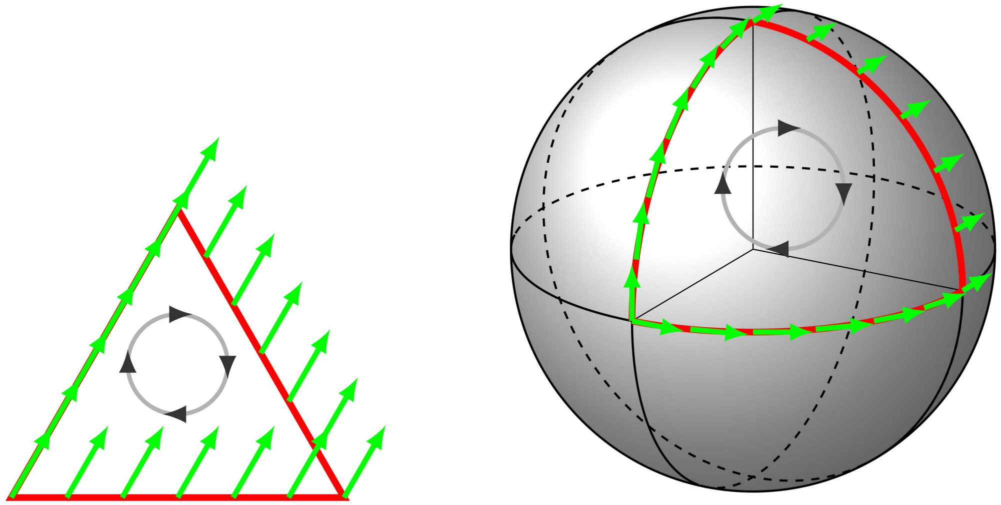
Section II describes the concept of parallel transport from differential geometry as well as its realization through a naturally evolving coordinate system. The naturally evolving coordinate system allows one to make as few changes as possible to a flat-Earth dynamic model to adapt it to a curved Earth. Section III then discusses the adaptation of flat-Earth dynamic models to a curved Earth. Section IV describes the ellipsoidal approximation to the shape of the Earth, and a full algorithm for the ellipsoidal approximation is given in Subsection IV-D. The validity of the algorithm is demonstrated through simulation in Section V, and the results are summarized in Section VI.

II. PARALLEL TRANSPORT ON A SURFACE IN 3D AND NATURALLY EVOLVING COORDINATES

On a flat surface, an individual walking along a curved path generally perceives oneself as moving and turning, not that one is still while the rest of the world moves and rotates. The perception of self-motion means that, as illustrated in Fig. 1(a), a direction vector in the local coordinate system that one carries will not rotate as one moves. That is, while the vector might move with the observer, for example representing an axis in a moving local coordinate system, the vector at one time is always parallel to the vector at a future time. It is "parallel transported" along the line of motion. Such an evolution of vectors in the plane can be contrasted, for example, with the coordinate system given by the Frenet formulas [15, Ch. 1–5] (the Frenet frame), whereby the direction of the local coordinate system evolves such

[7] In the United States, aircraft below 18,000 ft (5,486.4 m) determine their altitudes based on their height determined from the local pressure and the current pressure reading on the ground, translating into a "true" altitude; whereas aircraft operating above 18,000 ft assume the presence of a standard atmosphere, so their true altitude varies with local pressure variations [21]. Similar rules apply in other countries. For example, in Britain the switch to a standard atmospheric assumption occurs at 3,000 ft.

[8] Aircraft in controlled airspace are allocated flight levels that are 500 ft (≈ 152 m) apart.



(a) Parallel Transport Around a Flat Triangle

(b) Parallel Transport Around a Triangle on a Sphere

Fig. 2. In (a) and (b), a vector is parallel transported around a triangle in the direction of the arrows on the circle (clockwise) starting from the lower left corner of the triangle. In (a), the triangle is in a plane and the vector has the same orientation at the beginning of the loop as at the end. In (b), the vector is pointed in a different direction (in the local tangent plane) after having completed the loop.

that one axis always points in the direction along a curve of motion. In the Frenet frame, the evolution of the coordinate system along a curve is determined by the curvature of a curve, not of the surface of which the curve might be located. The Darboux frame, on the other hand, can be viewed as a modified Frenet frame for a curve on a surface, where one axis of the local coordinate system follows a curve along the surface and another axis remains normal to the surface [15, Ch. 4-4]. Figure 1(b) shows an example of how a vector would evolve under the Darboux frame on a flat surface.

The notion of parallel transporting a coordinate system along a curve has been considered in [5], [16], [28], where in [16] tips for numerically implementing such a system are provided. The parallel transport frame along a curve has even been used to describe the local coordinate system of an unmanned aerial vehicle (UAV) [26]. On the other hand, differential geometry for describing motion on a curved surface has been applied to geodetic problems [7]. However, the work of [7] only considers the problem of simulating targets constrained to a surface, and does not specifically consider the use of arbitrary dynamic models.

The notion of parallel transport can be extended to curved surfaces, just as the Darboux frame is a curved-surface equivalent to the Frenet frame. Parallel transport on a curved surface arises naturally as one would expect that the local coordinate system associated with a person walking along an arbitrary path on the surface of the Earth would experience no acceleration in a plane tangential to the surface of the Earth; the world does not "rotate" around the person, but the direction of the "up" vector always stays the same. Figure 2 illustrates how parallel transport on a surface (in this case, a sphere) depends both on the direction traveled as well as on the curvature of the surface. In Fig. 2(a), a vector is shown parallel transported clockwise around a triangle, remaining parallel at all times; whereas in Fig. 2(b), a vector is parallel transported counterclockwise around a triangle on a sphere, returning to the starting point at a different orientation. The topic of parallel transport is discussed in the following subsection.

A geodesic curve is the locally shortest distance between two points on a surface.[9] When the vector being parallel transported with a local observer is the observer's own velocity vector and the observer follows a trajectory of constant velocity, the observer follows a geodesic curve. The concepts of parallel transport and geodesics are covered formally in detail in [15, Sec. 4-4] and [25, Ch. 8].

Subsection II-A covers the topic of 2D motion on a 3D surface in a differential geometric framework. The evolution of the local coordinates of an observer on a surface is shown to depend on the parameterization of the surface. The concept of a naturally evolving coordinate system is introduced in Subsection II-B as a means of describing the evolution of a local coordinate system without the need for surface-dependent acceleration terms in the local observer's coordinate system.

A. Parallel Transport on a Surface in 3D

Let **r** be a vector in a global orthogonal (Cartesian) coordinate system (such as Earth-centered, Earth-fixed) from the origin to a point on the surface under consideration. In this paper, the point is on the surface of

[9] The distance is locally shortest, because no minor variations to the path will make it shorter. However, it is not necessarily the globally shortest path. For example, two possible geodesic curves can go from New York to Paris. One goes East and the other goes West. However, the eastward path is the shorter geodesic. When using an optimization algorithm to minimize a distance function to find the shortest path, a geodesic curve is a locally optimal solution.

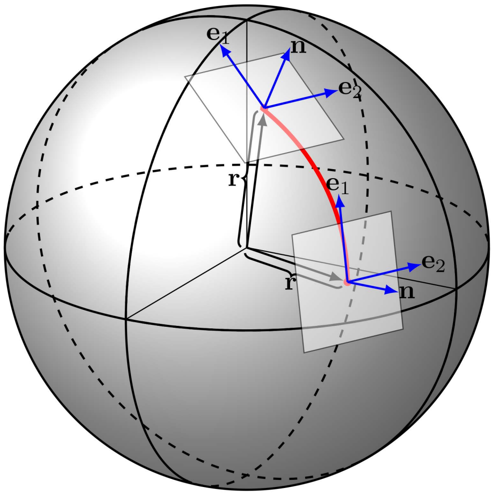


Fig. 3. At a particular point on a surface (specified by a position vector $\mathbf{r}$ at two possible points along a curve), one can have a local coordinate system defined by vectors in the tangent plane, $\mathbf{e}_1$ and $\mathbf{e}_2$, and a normal vector $\mathbf{n}$. When moving along a path on a surface, all of the vectors must change their direction so that the normal to the surface remains pointing "up." Whereas the direction of the normal vector is uniquely defined everywhere, any two orthogonal vectors in the local tangent plane can be used for $\mathbf{e}_1$ and $\mathbf{e}_2$. Parallel transport of the basis vectors, which is the principle behind the naturally evolving coordinate system, means that there is no acceleration in the tangent plane as one moves along the curve; that is, $\mathbf{e}_1$ and $\mathbf{e}_2$ must rotate to remain in the local tangent plane, but none of the rotation is about the normal vector (there is no acceleration in the local tangent plane).

the Earth. Assume that the surface can be parameterized using N_d parameters denoted by $x_1,\ldots,x_{N_d}$. For example, if the shape is the reference ellipsoid for the Earth, N_d is two, even though $\mathbf{r}$ is a three-dimensional position vector, because any point on an ellipsoid can be expressed in terms of geodetic latitude ϕ and longitude λ. However, such a choice of parameterization is not unique.

A vector $\mathbf{v}$ in the tangent plane to the surface of a shape can be expressed as a weighted sum of basis vectors in the tangent plane to the surface. Tangent vectors are given by derivatives of $\mathbf{r}$ with respect to the parameterization of the surface. Specifically,

$$\mathbf{v} = \sum_{i=1}^{N_d} v_i^s \overbrace{\frac{\partial \mathbf{r}}{\partial x_i}}^{\mathbf{e}_i}, \tag{2}$$

where the v_i^s are not the elements of $\mathbf{v}$, (which represents a 3D Cartesian vector), but instead represent weightings in the tangent space of the surface at a particular point. They can be considered the "local" coordinates of $\mathbf{v}$ as seen by an observer on the surface. The ith tangential basis vector is $\mathbf{e}_i$. Figure 3 illustrates the position vector and local basis vectors when considering a trajectory on the surface of a sphere. The concern of this subsection is the mathematical description of how vectors in the local tangent plane evolve over time as one moves on a surface.

Since $\mathbf{v}$ is constrained to a tangent plane to the Earth's surface (for example, it could be a heading), the 3D Cartesian direction of $\mathbf{v}$ must depend on position. The total derivative of $\mathbf{v}$ with respect to a coordinate x_j parameterizing the location on the surface is

$$\frac{d\mathbf{v}}{dx_j} = \underbrace{\sum_{i=1}^{N_d} \frac{\partial v_i^s}{\partial x_j}\mathbf{e}_i}_{\substack{\text{Change of}\\ \text{Components}}} + \underbrace{\sum_{i=1}^{N_d} v_i^s \overbrace{\frac{\partial^2 \mathbf{r}}{\partial x_j \partial x_i}}^{\partial \mathbf{e}_i/\partial x_j}}_{\substack{\text{Change of}\\ \text{Basis Vectors}}}. \tag{3}$$

In general, the second term in (3) does not need to be limited to the tangent plane and consequently can be written as

$$\frac{\partial \mathbf{e}_i}{\partial x_j} = \sum_{k=1}^{N_d} \Gamma_{i,j}^k \mathbf{e}_k + \mathbf{n}_{ij}, \tag{4}$$

where $\Gamma_{i,j}^k$ is a Christoffel symbol of the second kind, a projection (inner product) of $\partial^2\mathbf{r}/\partial x_j \partial x_i$ onto the basis vector $\partial \mathbf{r}/\partial x_k$,

$$\Gamma_{i,j}^k = \frac{\partial \mathbf{e}_i}{\partial x_j} \cdot \left(\frac{\mathbf{e}_k}{\|\mathbf{e}_k\|}\right), \tag{5}$$

where the dot indicates an inner product and $\mathbf{n}_{ij}$ consists of any second derivative components that do not lie in the tangent plane. The two vertical lines denote the magnitude of the enclosed vector (the l_2-norm). Note that $\Gamma_{ij}^k = \Gamma_{ji}^k$.

As long as $\mathbf{v}$ remains on the surface, it is always confined to the tangent plane, suggesting that derivative components that cannot be expressed in terms of tangential basis vectors are not necessarily essential to defining the evolution of the coefficients on the surface. The covariant derivative is simply the total derivative with the components $\mathbf{n}_{ij}$, which are orthogonal to the tangent plane, discarded. The covariant derivative is thus

$$\nabla_{x_j}\mathbf{v} = \sum_{i=1}^{N_d} \frac{\partial v_i^s}{\partial x_j}\mathbf{e}_i + \sum_{i=1}^{N_d}\sum_{k=1}^{N_d} v_i^s \Gamma_{i,j}^k \mathbf{e}_k. \tag{6}$$

A zero covariant derivative implies zero acceleration in the local tangent plane. To use a flat-Earth dynamic model on a curved Earth, zero tangential acceleration means that no extra terms need to be added to the flat-Earth model in adapting it to the curved Earth; only the basis vectors change.

The notion of parallel transport arises when considering how a vector changes direction as an object moves along a curve on the surface. If the instantaneous velocity vector $\mathbf{v}$ of a non-maneuvering object is parallel transported along the surface in the direction of $\mathbf{v}$, then the object will follow a geodesic curve. In general, however, $\mathbf{v}$ can be any vector (such as acceleration) in the local coordinate system of the target moving on an arbitrary path on the surface. Consequently, the concept of parallel transport shall be described for how $\mathbf{v}$ evolves as an object traverses an arbitrary curve.

Let the curve on the surface that the target follows be parameterized by t, which could be time. Consequently, the x-coordinates parameterizing $\mathbf{r}$ on the surface are functions of t. The covariant derivative of $\mathbf{v}$ with respect to t can be evaluated using the chain rule:

$$\nabla_t \mathbf{v} = \sum_{j=1}^{N_d} \nabla_{x_j} \mathbf{v} \frac{dx_j}{dt} \tag{7a}$$

$$= \sum_{j=1}^{N_d}\sum_{i=1}^{N_d} \frac{\partial v_i^s}{\partial x_j}\mathbf{e}_i \frac{dx_j}{dt} + \sum_{j=1}^{N_d}\sum_{i=1}^{N_d}\sum_{k=1}^{N_d} v_i^s \Gamma_{i,j}^k \mathbf{e}_k \frac{dx_j}{dt} \tag{7b}$$

$$= \sum_{i=1}^{N_d} \frac{dv_i^s}{dt}\mathbf{e}_i + \sum_{j=1}^{N_d}\sum_{i=1}^{N_d}\sum_{k=1}^{N_d} v_i^s \frac{\partial \mathbf{e}_i}{\partial x_j}\cdot\left(\frac{\mathbf{e}_k}{\|\mathbf{e}_k\|}\right)\mathbf{e}_k \frac{dx_j}{dt} \tag{7c}$$

$$= \sum_{i=1}^{N_d} \frac{dv_i^s}{dt}\mathbf{e}_i + \sum_{i=1}^{N_d}\sum_{k=1}^{N_d} v_i^s \frac{d\mathbf{e}_i}{dt}\cdot\left(\frac{\mathbf{e}_k}{\|\mathbf{e}_k\|}\right)\mathbf{e}_k. \tag{7d}$$

Parallel transport occurs when the v_i terms evolve such that $\nabla_t \mathbf{v} = \mathbf{0}$. If t is time, then the terms dx_j/dt are related to the velocity of the observer in the observer's local coordinate system.

The elements of $\nabla_t \mathbf{v}$ are in a global Cartesian coordinate system and are thus independent of how the x terms parameterize $\mathbf{r}$ to obtain the tangential basis vectors $\mathbf{e}$. For the purpose of this discussion, a parameterization of $\mathbf{r}$ is chosen such that the basis vectors in the tangent plane are always orthonormal. It is not necessary to know how to obtain such a set of basis vectors practically, only to know that they exist and could be used. That the magnitudes of the basis vectors remain constant implies that the derivative of a basis vector with respect to t must be orthogonal to the basis vector. The time evolution of the basis vectors as a function of t must be a rotation lest the magnitudes of the vectors change, and all basis vectors must experience the same rotation to remain orthogonal.[10] A 3D rotational dynamic model allows the derivative of a basis vector to be written as

$$\frac{d\mathbf{e}_i}{dt} = \Omega \times \mathbf{e}_i, \tag{8}$$

where × denotes the cross product, the direction of Ω is the axis of rotation, and the magnitude of Ω specifies the rotation rate. (How Ω is determined for a particular surface is discussed in Subsection II-B.) Consequently, when considering a 3D space, if $\nabla_t \mathbf{v} = \mathbf{0}_{3,1}$, where $\mathbf{0}_{3,1}$ is a 3×1 vector of zeros, (7d) becomes

$$\mathbf{0}_{3,1} = \sum_{i=1}^{N_d} \frac{dv_i^s}{dt}\mathbf{e}_i + \sum_{i=1}^{N_d}\sum_{k=1}^{N_d} v_i^s(\mathbf{e}_k \cdot (\Omega \times \mathbf{e}_i))\mathbf{e}_k \tag{9a}$$

$$= \sum_{i=1}^{N_d} \frac{dv_i^s}{dt}\mathbf{e}_i + \sum_{i=1}^{N_d}\sum_{k=1}^{N_d} v_i^s(\Omega \cdot (\mathbf{e}_i \times \mathbf{e}_k))\mathbf{e}_k, \tag{9b}$$

where the vector triple product identity $\mathbf{a}\cdot(\mathbf{b}\times\mathbf{c}) = \mathbf{b}\cdot(\mathbf{c}\times\mathbf{a})$ is used. Rearranging terms yields

$$\sum_{i=1}^{N_d} \frac{dv_i^s}{dt}\mathbf{e}_i = -[\mathbf{e}_1 \dots \mathbf{e}_{N_d}] \begin{bmatrix} \sum_{i=1}^{N_d} v_i^s((\mathbf{e}_i \times \mathbf{e}_1)\cdot\Omega) \\ \vdots \\ \sum_{i=1}^{N_d} v_i^s((\mathbf{e}_i \times \mathbf{e}_{N_d})\cdot\Omega) \end{bmatrix}. \tag{10}$$

In three dimensions, two tangent vectors are sufficient to parameterize a surface ($N_d = 2$). On applying the identity $\mathbf{a}\times\mathbf{b} = -\mathbf{b}\times\mathbf{a}$, (10) reduces to

$$\mathbf{e}_1\frac{dv_1^s}{dt} + \mathbf{e}_2\frac{dv_2^s}{dt} = -[\mathbf{e}_1 \quad \mathbf{e}_2]\begin{bmatrix} -v_2^s(\mathbf{e}_1\times\mathbf{e}_2)\cdot\Omega \\ v_1^s(\mathbf{e}_1\times\mathbf{e}_2)\cdot\Omega \end{bmatrix} \tag{11a}$$

$$= -(\Omega - (\Omega\cdot\mathbf{e}_i)\mathbf{e}_i - (\Omega\cdot\mathbf{e}_j)\mathbf{e}_j)\times\mathbf{v}. \tag{11b}$$

When assuming that $\mathbf{e}_1$ and $\mathbf{e}_2$ are mutually orthonormal, the components of $d\mathbf{v}^s/dt$ can be extracted using dot products with the basis vectors to get

$$\frac{dv_1^s}{dt} = -(\Omega\times\mathbf{v})\cdot\mathbf{e}_1 = v_2\Omega\cdot(\mathbf{e}_1\times\mathbf{e}_2), \tag{12}$$

$$\frac{dv_2^s}{dt} = -(\Omega\times\mathbf{v})\cdot\mathbf{e}_2 = -v_1\Omega\cdot(\mathbf{e}_1\times\mathbf{e}_2). \tag{13}$$

That is, the local coordinates change according to the projection of the cross product of the axis of rotation of the basis vectors onto the local coordinates. The magnitude of the local coordinates is preserved through this operation, which can be demonstrated by writing

$$\begin{bmatrix} \frac{dv_1^s}{dt} \\ \frac{dv_2^s}{dt} \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -\Omega\cdot(\mathbf{e}_1\times\mathbf{e}_2) \end{bmatrix} \times \begin{bmatrix} v_1^s \\ v_2^s \\ 0 \end{bmatrix}, \tag{14}$$

where, due to the cross product, it can be seen that the derivative is orthogonal to the component vector. Since the basis vectors are assumed to maintain unit magnitude, the vector $\mathbf{v}$ maintains a constant magnitude through parallel transport.

After substituting the derivatives of the local coordinates in (12) and (13) into the total derivative in (3), the evolution in the direction of $\mathbf{v}$ over time is

$$\frac{d\mathbf{v}}{dt} = -\sum_{i=1}^{2}((\Omega\times\mathbf{v})\cdot\mathbf{e}_i)\mathbf{e}_i + \sum_{i=1}^{2} v_i^s\Omega\times\mathbf{e}_i \tag{15}$$

$$= \underbrace{\Omega\times\mathbf{v}}_{\substack{\text{General}\\ \text{3D Rotation}}} \underbrace{+v_2\Omega\cdot(\mathbf{e}_1\times\mathbf{e}_2)\mathbf{e}_1 - v_1\Omega\cdot(\mathbf{e}_1\times\mathbf{e}_2)\mathbf{e}_2}_{\text{Undo Rotation in Tangent Plane}}. \tag{16}$$

[10] The ability to express the evolution of the basis vectors as a continuous rotation assumes that there is no bifurcative changes in the basis vectors that redefine the relative relationship between the vectors. For example, suddenly changing from a right-handed coordinate system to a left-handed coordinate system would be a bifucative change in the relationship between the vectors. A bifurcation would also arise when crossing the North Pole using normalized ENU coordinates.

Thus, as **v** parallel transports along the surface, the change in the direction of **v** depends only on how the normal vector to the tangent plane changes, not on how the bases are rotated within the tangent plane. If the orientation of the tangent plane does not change, any rotation of the bases does not matter; **v** does not change. The irrelevance of rotations outside of the local tangent plane means that the rotation vector Ω to describe the evolution of the orientation of the tangent plane is not unique.

B. Naturally Evolving Coordinates

If the vector **v** being parallel transported in (16) is, for example, a velocity vector, then the local coordinates must evolve according to (12) and (13). However, forcing the local coordinates to evolve depending on the motion on the surface can be inconvenient if one wishes to use more than a straight-line dynamic model. It is more convenient if the local coordinates never changes as the target moves.

The vector **v** does not need to be a velocity vector. For example, it could represent an axis (a basis vector) in the coordinate system of the local observer moving on the surface. If motion along the surface does not induce any change in the components of **v** (any basis vector) in the *local* coordinate system of the target, v_1 and v_2, then the target does experience any local acceleration beyond that induced by the flat-Earth dynamic model. Examining (14), v_1 and v_2 remain constant if the axis of rotation is orthogonal to the local normal vector (if there is no rotation in the local tangent plane). When the basis vectors parameterizing **r**, $\mathbf{e}_1$, and $\mathbf{e}_2$, are chosen to have zero rotation in the local tangent plane, then (16) implies that

$$\frac{d\mathbf{v}}{dt} = \Omega \times \mathbf{v}. \tag{17}$$

The elimination of the extra terms in (16) to obtain (17) means that the basis vectors in which the coordinates of **r** are expressed depend on the path taken by the object. Such a path-dependent local coordinate system is defined to be "naturally evolving coordinates." The set of local tangent-plane basis vectors for natural coordinates are designated as $\mathbf{u}_1$ and $\mathbf{u}_2$ to differentiate them from the vectors parameterization of the surface that have thus been expressed as $\mathbf{e}_k$.

Let **g** be a vector orthogonal to the local tangent plane that defines the local "downward" direction on the two-dimensional surface. The down vector **g** is not necessarily a unit vector and could be acceleration due to gravity. To define a right-handed local coordinate system in terms of three orthonormal basis vectors, let $\mathbf{u}_3$ be a unit vector defining the local vertical, which is expressed in terms of **g** as

$$\mathbf{u}_3 = -\frac{\mathbf{g}}{\|\mathbf{g}\|}. \tag{18}$$

In other words, the unit "up" vector is the negative of the "down" vector divided by its magnitude. To complete a right-handed coordinate system, the remaining orthogonal unit vectors $\mathbf{u}_1$ and $\mathbf{u}_2$ in the local tangent plane must satisfy the defining cross product relations

$$\mathbf{u}_1 \times \mathbf{u}_3 = -\mathbf{u}_2 \qquad \mathbf{u}_1 \times \mathbf{u}_2 = \mathbf{u}_3 \qquad \mathbf{u}_2 \times \mathbf{u}_3 = \mathbf{u}_1. \tag{19}$$

One coordinate system that satisfies all of these relations is a local East-North-Up (ENU) coordinate system. One would like to have vectors that are parallel transported with a moving target naturally evolve. Though an ENU coordinate system can be used to define a target's initial local coordinate system, it does not naturally evolve as a target moves. Natural evolution of a vector is synonymous with parallel transport reducing to (17) and the basis vectors evolving with no rotation in the tangent plane.

Expressions for how the orthonormal basis vectors $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$ evolve over time as a target moves shall be derived. The derivative of (18) with respect to time t is

$$\dot{\mathbf{u}}_3 = -\frac{\dot{\mathbf{g}}}{\|\mathbf{g}\|} + \frac{\mathbf{g}}{\|\mathbf{g}\|^3}(\dot{\mathbf{g}} \cdot \mathbf{g}) = \Omega \times \mathbf{u}_3, \tag{20}$$

where the dot indicates differentiation with respect to t. The simplification to a cross product expression is possible, because the derivative must be orthogonal to the vector, lest the magnitude of the basis vector change. An infinite number of solutions for the rotation vector Ω exist. For the naturally evolving coordinate system, the vector Ω is constrained to the local tangent plane (so that $\mathbf{u}_3$ has no rotational components in the local tangent plane):

$$\Omega = c_1\mathbf{u}_1 + c_2\mathbf{u}_2. \tag{21}$$

Consequently,

$$\dot{\mathbf{u}}_3 = (c_1\mathbf{u}_1 + c_2\mathbf{u}_2) \times \mathbf{u}_3. \tag{22}$$

Using (19), $\dot{\mathbf{u}}_3$ becomes

$$\dot{\mathbf{u}}_3 = \overbrace{[-\mathbf{u}_2 \quad \mathbf{u}_1]}^{\mathbf{A}} \begin{bmatrix} c_1 \\ c_2 \end{bmatrix}. \tag{23}$$

Equation (23) is an overdetermined system for which a unique solution should exist. Thus, the coefficients c_1 and c_2 can be found using

$$\begin{bmatrix} c_1 \\ c_2 \end{bmatrix} = \mathbf{A}^\dagger \left(-\frac{\dot{\mathbf{g}}}{\|\mathbf{g}\|} + \frac{\mathbf{g}}{\|\mathbf{g}\|^3}(\dot{\mathbf{g}} \cdot \mathbf{g}) \right), \tag{24}$$

where the † operator represents the matrix pseudoinverse. The matrix pseudoinverse $\mathbf{A}^\dagger$ can be evaluated using `pinv` in Matlab, which is based on a singular value decomposition technique for stability. A similar matrix pseudoinverse algorithm is described in [24, Ch. 5] and [23]. A simpler but less numerically robust alternative is just to use $\mathbf{A}^\dagger = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$, where, as described in [23], the possible numerical instability arises when the matrix $\mathbf{A}^T\mathbf{A}$ is nearly singular.

To evaluate (24) to get Ω in (21), one must know $\dot{\mathbf{g}}$, which is given by

$$\dot{\mathbf{g}} = \begin{bmatrix} \frac{\partial \mathbf{g}}{\partial r_1} & \frac{\partial \mathbf{g}}{\partial r_2} & \frac{\partial \mathbf{g}}{\partial r_3} \end{bmatrix} \dot{\mathbf{r}}, \tag{25}$$

where $\mathbf{r} = [r_1, r_2, r_3]'$ are in global Cartesian coordinates and $'$ denotes the transpose operator. The partial derivatives in (25), which are constrained to the surface, can be written using the chain rule

$$\frac{\partial \mathbf{g}}{\partial r_i} = \sum_{j=1}^{2} \frac{\partial \mathbf{g}}{\partial x_j} \frac{\partial x_j}{\partial r_i}, \tag{26}$$

where the x_j are the terms parameterizing $\mathbf{r}$ on the surface. However, the evaluation of the partial derivatives in (26) can be problematic, because the parameterization of the surface can have singularities. For example, when using ENU coordinates, singularities exist at the poles, because longitude does not have any meaning at the poles. Such singularities can be avoided either by obtaining a closed-form solution for the product of the partial derivatives, which is difficult, or by using a numerical differentiation algorithm. Section IV-C describes how, following the technique of [78], an explicit solution for Ω can be found when using an ellipsoidal Earth model.

In summary, the naturally evolving local coordinate system of a target moving on a surface is defined as an orthonormal coordinate system at an initial point on the surface with axes given by the unit vectors $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$. For example, one might use the ENU axes described in Section IV when on an ellipsoidal Earth. Then, the axes evolve according to

$$\dot{\mathbf{u}}_i = \Omega \times \mathbf{u}_i,, \tag{27}$$

where Ω is given by (21) and (24) (or for an ellipsoidal Earth model from Section IV-C), which depend on the derivative of the local "down vector" $\mathbf{g}$ in (25). Each component of the partial derivatives can be evaluated using (26), which must be simplified to avoid singularities, or using numerical differentiation to avoid singularities. Techniques for numerical differentiation, besides simple differencing of two points, are derived using Taylor series expansions [6, Ch. 4.1]. As described in [80], finite difference formulae with $2N - 1$ points can be automatically generated in Mathematica.

III. APPLYING NATURALLY EVOLVING COORDINATES TO 2D AND 3D DYNAMIC MODELS

Let $\mathbf{v}^{\text{local}} = [v_1^l, v_2^l, v_3^l]^T$ be a vector in the local coordinate system of the target, for example, velocity or acceleration. A vector $\mathbf{v}^{\text{global}}$, which is $\mathbf{v}^{\text{local}}$ as seen in the global observer's coordinate system, can be expressed as

$$\mathbf{v}^{\text{global}} = \sum_{i=1}^{3} v_i^l \mathbf{u}_i. \tag{28}$$

Let $\mathbf{x}_t^g$ and $\mathbf{x}_t^l$ be the state vectors of the target in the global (curved-Earth) and local (flat-Earth) coordinate systems at time t, respectively. A noise-free flat-Earth dynamic model is generally expressed in terms of a differential equation,

$$\dot{\mathbf{x}}_t^l = \mathbf{a}^l(\mathbf{x}_t^l, t), \tag{29}$$

where $\dot{\mathbf{x}}_t^l$ is the derivative of the local state with respect to time and $\mathbf{a}^l$ is a possibly time-variant function of $\mathbf{x}_t^l$ known as the drift.

It is assumed that the local dynamic model under consideration obtains the position component of the model by integrating a velocity component of the state. The rest of the dynamic model is arbitrary, but is assumed to be independent of position. Consequently, $\mathbf{x}_t^l$ and $\mathbf{a}^l$ have the forms

$$\mathbf{x}_t^l = \begin{bmatrix} \mathbf{r}_t^l \\ \dot{\mathbf{r}}_t^l \\ \mathbf{x}_t^{l,\text{rest}} \end{bmatrix} \qquad \mathbf{a}^l(\mathbf{x}_t^l, t) = \begin{bmatrix} \dot{\mathbf{r}}_t^l \\ \mathbf{a}^{l,\text{rest}}(\dot{\mathbf{r}}_t^l, \mathbf{x}_t^{l,\text{rest}}, t) \end{bmatrix}, \tag{30}$$

where $\mathbf{r}_t^l$ is the position of the target in its local coordinate system at time t, $\dot{\mathbf{r}}_t^l$ is the velocity of the target in its local coordinate system, and the vector $\mathbf{x}_t^{l,\text{rest}}$ is an arbitrary collection of additional state elements.

The simplest approach for adapting (29) to a global, curved-Earth model is to use a mixed state, where the flat-Earth position $\mathbf{r}_t^l$ is replaced by the curved-Earth position $\mathbf{r}_t^g$. The mixed state $\mathbf{x}_t^m$ with associated dynamic model is thus

$$\mathbf{x}_t^m = \begin{bmatrix} \mathbf{r}_t^g \\ \dot{\mathbf{r}}_t^l \\ \mathbf{x}_t^{l,\text{rest}} \end{bmatrix} \qquad \dot{\mathbf{x}}_t^m = \mathbf{a}^m(\mathbf{x}_t^m, t), \tag{31}$$

where

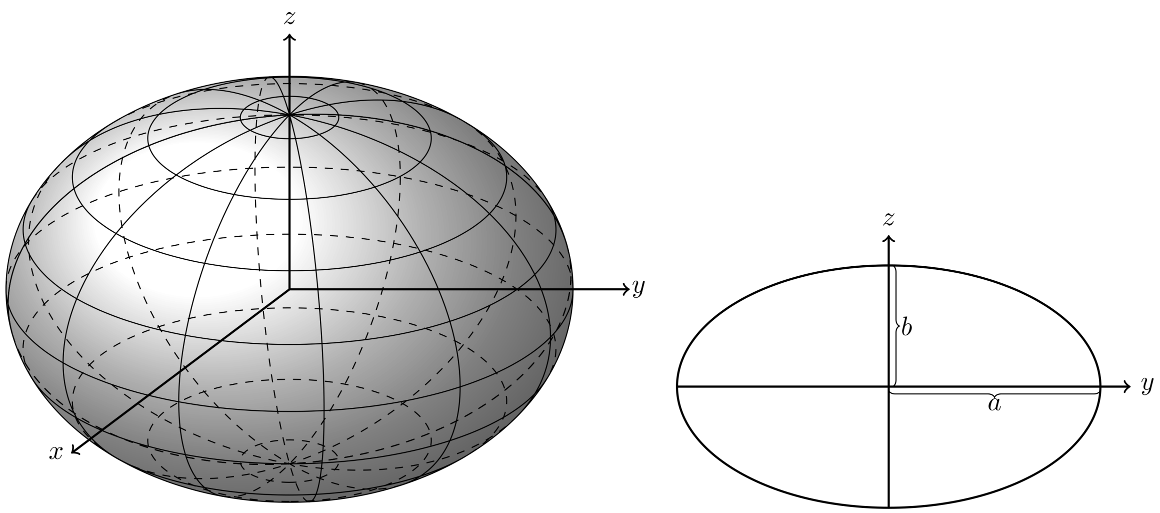
$$\mathbf{a}^m(\mathbf{x}_t^m, t) = \begin{bmatrix} \dot{\mathbf{r}}_t^g \\ \mathbf{a}^{l,\text{rest}}(\dot{\mathbf{r}}_t^l, \mathbf{x}_t^{l,\text{rest}}, t) \end{bmatrix}. \tag{32}$$

Since the global velocity can be linked to the local velocity using (28), (32) can be rewritten as

$$\mathbf{a}^m(\mathbf{x}_t^m, t) = \begin{bmatrix} [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \mathbf{u}_3]\dot{\mathbf{r}}_t^l \\ \mathbf{a}^{l,\text{rest}}(\dot{\mathbf{r}}_t^l, \mathbf{x}_t^{l,\text{rest}}, t) \end{bmatrix}. \tag{33}$$

If in the target's local coordinate system the target experiences no vertical motion (the velocity components multiplying $\mathbf{u}_3$ in (33) are always zero), then the target can be viewed as remaining at a constant altitude and, ignoring air pressure, would be expected to follow a surface of constant gravitational potential. Thus, the basis vectors should evolve "naturally" as in (27), because the effects of the surface curvature enter the dynamic equations only through a rotation in the local bases, not by changing the local states, and the target is constrained to the surface.

On the other hand, suppose that $\dot{\mathbf{r}}_t^l$ has a nonzero component in the $\mathbf{u}_3$ direction. The target no longer remains at a constant gravitational potential, since it is



(a) Ellipsoidal Earth Approximation

(b) $y - z$ Cut ($x = 0$)

Fig. 4. The Earth is approximately shaped like an oblate spheroid; that is an ellipse rotated about the z axis (a type of ellipsoid). The non-spheroidal nature of the Earth's shape has been exaggerated in the diagrams. The horizontal lines in (a) represent lines of latitude; vertical lines represent lines of longitude. Longitude is an angle measured with respect to the x-axis in the xy plane. The DoD's WGS-84 standard [13] and IERS conventions [58], among many other sources, define reference ellipsoids for the Earth. In (b), a is the semi-major axis (equatorial radius), and b is the semi-minor axis (polar radius).

ascending or descending along the local vertical. The local surface of constant gravitational potential should be what defines the local tangent plane. However, **g** changes orientation as one moves vertically. The change of direction of **g** with motion in the direction of **g** is known in geodesy as the "curvature of the plumb line" [29, Ch. 2.3]. Curvature of the plumb line implies that the parallel transport equations of Section II must be rederived to determine the "straightest" path when moving in an arbitrary direction through a field where the local coordinate system rotates everywhere. However, when using the ellipsoidal Earth approximation of Section IV, the plumb line does *not* curve.

In an ENU coordinate system, moving up does not change the direction of what is considered to be up. In this instance, if one uses (27) in Section II-B to predict forward the axes of the local coordinate system, $\mathbf{u}_1$ and $\mathbf{u}_2$ always remain in the local tangent plane, and the projection of a straight-line motion target onto the reference ellipsoid follows a geodesic. Consequently, the suggested technique for handling 3D flat-Earth dynamic models on a curved Earth under an ellipsoidal approximation is to make **g** the negative of the normal vector of the closest point on the reference ellipsoid.

IV. THE ELLIPSOIDAL APPROXIMATION SOLUTION

A. The Gravitational Potential and the Reference Ellipsoid

The Earth is approximately oblate spheroidal in shape (an ellipsoidal shape), as illustrated in Fig. 4. However, the shape of the Earth as defined by the geoid, a hypothetical surface of equal gravitational potential defining mean sea level (MSL), is only approximately ellipsoidal.

The global coordinate system used by the U.S. DoD is the WGS-84 standard [13]. The latest version of WGS-84 [51] agrees with the alignment and location of the axes in the ITRF2008, the latest version of the International Terrestrial Reference Frame (ITRF). The ITRF is set by the International Earth Rotation and Reference Systems Service (IERS) [13], [51] and is described in the IERS conventions in [58]. Parameters for the ITRF can be downloaded from [67]. The x-axis of the ITRF points in the direction of the IERS reference meridian, which is the same as the prime meridian, zero degrees longitude, which passes through Greenwich, England. The z-axis points in the direction of the IERS reference pole, which is the geographic North Pole, and is approximately the rotational axis of the Earth. Differences between WGS-84 and ITRF arise from differences in points of reference used to implement the standards, as discussed in [29, Ch. 2.11], and are due to the WGS-84 standard using a different semi-major radius for the reference ellipsoid.

The WGS-84 standard defines a reference ellipsoid for the shape of the Earth, which is derived from parameters for the Earth's gravitational field and is a low-order approximation of the geoid. Table II lists the parameters for the reference ellipsoid, and Fig. 4(b) shows how the parameters relate to the ellipse of revolution. The flattening factor is related to the semi-major (a) and semi-minor (b) axes through the equation

$$f = \frac{a-b}{a}. \tag{34}$$

The reference ellipsoid can be directly related to an approximation of the gravitational potential including the effects of the Earth's rotation, ignoring the direction-dependent Coriolis effect (see [29, Ch. 2.7]).

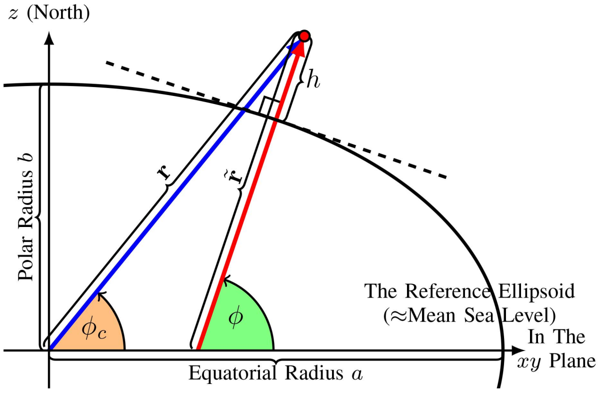


Fig. 5. An illustration of how latitude is measured for a point an altitude h above the reference ellipsoid. Unless specified otherwise, the term latitude generally means the geodetic latitude ϕ, which is taken with respect to a normal to the reference ellipsoid. On the other hand, the geocentric latitude ϕ_c, which is seldom used, is the angle measured with respect to the origin. The horizontal axis is anywhere in the xy plane, which comes out of the page in the diagram.

TABLE II
Defining parameters of the reference ellipsoid in the WGS-84 standard [13]. The presence of GM and ω allow one to reconstruct the gravitational potential of the equipotenial surface implied by the ellipsoid, as derived in [29, Ch. 2.7].

Quantity	Symbol	Value
Gravitational Constant times Earth's Mass (with atmosphere)	GM	$3.986004418 \times 10^{14} (\mathrm{m}^3/\mathrm{s}^2)$
Earth's Semi-Major Axis	a	6378137.0 m
Earth's Inverse Flattening Factor	$1/f$	298.257223563
Nominal Angular Velocity of Earth's Rotation	ω	$7.2921150 \times 10^{-5} (\mathrm{rad/s})$

B. Coordinates on the Reference Ellipsoid

Geodetic and geocentric latitudes are angles measured with respect to the equatorial plane. The geocentric latitude ϕ_c of a point is the angle formed between the equatorial plane and a line from the center of the Earth to the point. The geodetic latitude ϕ is the angle formed between the equatorial plane and a normal to the reference ellipsoid that passes through the point. Figure 5 illustrates the difference between geocentric and geodetic latitudes. Unless otherwise specified, maps generally show geodetic latitude. On the other hand, if one draws a line from the coordinate origin to a point and projects the line onto the xy (equatorial) plane, the angle the line forms with the x-axis (the prime meridian) is the longitude λ. Geocentric latitude and longitude are equivalent to elevation and azimuth in a spherical coordinate system.

When using the ellipsoidal gravitational model, the normal to the reference ellipsoid coincides with the vertical direction defined by the negative of the acceleration due to gravity. That is, "up" is in the direction of $\tilde{\mathbf{r}}$ in Fig. 5. Latitude, longitude, and altitude (height) form an orthogonal set of coordinates. Thus, a set of basis vectors derived from local ellipsoidal coordinates can be used to define $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$ initially, but not over time. Rather, the basis vectors must evolve as discussed in Section II-B.

A target's position is given in Earth-Centered, Earth-Fixed (ECEF) coordinates in terms of a vector taken from the center of the Earth, such as $\mathbf{r}$ in Fig. 5. Such a position can be converted into geodetic latitude, longitude, and height coordinates (ϕ, λ, h) using the formulae of [66], which are summarized in Appendix B. Figure 5 shows a point above the reference ellipsoid with its height h measured with respect to the normal to the reference ellipsoid. The conversion of a point given in geodetic latitude, longitude, and height coordinates (ϕ, λ, h) to ECEF Cartesian coordinates can be done as

$$x = (N_e + h)\cos(\phi)\cos(\lambda) \tag{35}$$

$$y = (N_e + h)\cos(\phi)\sin(\lambda) \tag{36}$$

$$z = (N_e(1 - e^2) + h)\sin(\phi) \tag{37}$$

$$N_e = \frac{a}{\sqrt{1 - e^2(\sin(\phi))^2}}, \tag{38}$$

where the first numerical eccentricity of the ellipsoid is e,

$$e \triangleq \sqrt{2f - f^2}, \tag{39}$$

and N_e is the normal radius of curvature [29, Ch. 5.6], also known as the radius of curvature in the prime vertical.

A set of local orthogonal basis vectors obtained by differentiating (35), (36), and (37) with respect to ϕ, λ, and h are

$$\frac{d\mathbf{r}}{d\lambda} = c_1\mathbf{u}_1 = \begin{bmatrix} -(N_e + h)\cos(\phi)\sin(\lambda) \\ (N_e + h)\cos(\phi)\cos(\lambda) \\ 0 \end{bmatrix} \tag{40}$$

$$\frac{d\mathbf{r}}{d\phi} = c_2\mathbf{u}_2 = \begin{bmatrix} \left(\cos(\phi)\frac{dN_e}{d\phi} - (N_e + h)\sin(\phi)\right)\cos(\lambda) \\ \left(\cos(\phi)\frac{dN_e}{d\phi} - (N_e + h)\sin(\phi)\right)\sin(\lambda) \\ (N_e(1 - e^2) + h)\cos(\phi) + (1 - e^2)\frac{dN_e}{d\phi}\sin(\phi) \end{bmatrix} \tag{41}$$

$$\frac{d\mathbf{r}}{dh} = \mathbf{u}_3 = \begin{bmatrix} \cos(\phi)\cos(\lambda) \\ \cos(\phi)\sin(\lambda) \\ \sin(\phi) \end{bmatrix}, \tag{42}$$

where

$$\frac{dN_e}{d\phi} = \frac{ae^2\cos(\phi)\sin(\phi)}{(1 - e^2(\sin(\phi))^2)^{3/2}}. \tag{43}$$

The constants c_1 and c_2 in (41) and (40) represent the fact that the vectors do not have unit magnitudes. It can be verified that the vectors form an orthogonal basis. This basis is ENU, because $d\mathbf{r}/dh$ points up, $d\mathbf{r}/d\phi$

points in the direction of geographic North along the surface of the ellipse, and $d\mathbf{r}/d\lambda$ points East along the surface of the ellipse. The ENU coordinate system refers to the normalized forms of the bases in (40) through (42).

Since (40) has zero magnitude at the poles, but (41) and (42) have nonzero magnitudes, $\mathbf{u}_1$ is defined by

$$\mathbf{u}_1 = \mathbf{u}_2 \times \mathbf{u}_3 \tag{44}$$

to guarantee that it always exists. The longitude λ determines the orientation of the North and East axes at the poles.

C. Computing the Tangent Plane Rotation Vector

In the case of an ellipsoidal Earth model, the rotation vector Ω needed to propagate naturally evolving coordinates in Section II-B can be found directly as is done as an intermediate step in propagating wander coordinates in [50, Pgs. 54–56]. The instantaneous turn rate ω of an object that is rotating about an axis is equal to the speed of the object divided by the radius of the object from the axis. Here, we wish to find the instantaneous turn rate of an observer moving above an ellipsoidal Earth. Under an ellipsoidal approximation, the Earth has a different radius of curvature going North than going East. In [61, Sec. 3.5], the different radii of curvature of an ellipsoid are derived in detail. The radius of curvature in the prime vertical, which is the one relevant for North-South motion, is given by N_e in (38). The radius of curvature in the meridian, which is relevant to East-West motion, is given by

$$M_e = N_e \frac{1-e^2}{1-e^2\sin(\phi)^2} \tag{45}$$

Given a global velocity vector of a target expressed in terms of local East, North, and Up coordinates

$$\mathbf{v} = v_{\text{East}}\mathbf{u}_{\text{East}} + v_{\text{North}}\mathbf{u}_{\text{North}} + v_{\text{Up}}\mathbf{u}_{\text{Up}} \tag{46}$$

the instantaneous (right-handed) turn rate about the local East axis (which is needed to correct for North-south motion) is

$$\omega_{\text{East}} = \frac{-v_{\text{North}}}{M_e + h} \tag{47}$$

where h is the height of the target above the reference ellipsoid. Similarly, the instantaneous rotation rate about the local North-axis, which is needed to correct for (East-West motion) is

$$\omega_{\text{North}} = \frac{v_{\text{East}}}{N_e + h} \tag{48}$$

Thus, the total rotation vector needed to propagate naturally evolving coordinates is

$$\Omega = \omega_{\text{East}}\mathbf{u}_{\text{East}} + \omega_{\text{North}}\mathbf{u}_{\text{North}} \tag{49}$$

As the velocity vector $\mathbf{v}$ will be parameterized in terms of the naturally evolving basis vectors

$$\mathbf{v} = v_1^s\mathbf{u}_1 + v_2^s\mathbf{u}_2 + v_3^s\mathbf{u}_3 \tag{50}$$

the local East North and Up components necessary to use (49) can be found using dot products:

$$v_{\text{East}} = \mathbf{v}'\mathbf{u}_{\text{East}} \tag{51}$$

$$v_{\text{North}} = \mathbf{v}'\mathbf{u}_{\text{North}} \tag{52}$$

$$v_{\text{Up}} = \mathbf{v}'\mathbf{u}_{\text{Up}} \tag{53}$$

On the surface of the reference ellipsoid, the naturally evolving coordinate system coincides with the wander coordinate system of navigation.

On the other hand, in Section V, to compare with continuous-time extensions of the algorithms of [43], [81], which rotate a local East-North-Up coordinate system rather than using naturally evolving coordinates. In such a coordinate system, $\mathbf{u}_1 = \mathbf{u}_{\text{East}}$, $\mathbf{u}_2 = \mathbf{u}_{\text{North}}$, and $\mathbf{u}_3 = \mathbf{u}_{\text{Up}}$ and (33) can be used to propagate the state without an explicit differential equation for the basis vectors, because Subsection IV-B provides formulae for the East, North, and Up vectors all over the globe.

D. Constructing the Full Propagation Algorithm

Figure 6 consolidates the concepts in the previous sections to summarize the state propagation algorithm for fitting a flat-Earth dynamic model to a curved Earth under an ellipsoidal gravitational approximation. The Runge-Kutta algorithm, a standard technique for solving deterministic differential equations, is described in [6, Ch. 5.4, 5.5] as well as in the tutorial [9], and an adaptive function is built into Matlab under the name `ode45`.

V. DEMONSTRATING THE ALGORITHMS THROUGH SIMULATION

The validity of the method of adapting a flat-Earth dynamic model to a curved Earth causes the path of a constant-velocity flat-Earth motion model to follow the same geodesic curve that is obtained using established techniques given the same initial heading. Given a target state of $\mathbf{x}_t = [\mathbf{r}_t', \dot{\mathbf{r}}_t']'$, where $\mathbf{r}_t$ is the Cartesian position as a function of time and $\dot{\mathbf{r}}_t$ is the Cartesian velocity vector, the drift term for constant-velocity, flat-Earth state propagation is

$$\mathbf{a}^l(\mathbf{x}_t, t) = \begin{bmatrix} \dot{\mathbf{r}}_t^l \\ \mathbf{0}_{3,1} \end{bmatrix}. \tag{54}$$

Using the notation of Section III, (54) implies that $\mathbf{a}^{l,\text{rest}}(\dot{\mathbf{r}}_t^l, \mathbf{x}_t^{l,\text{rest}}, t) = \mathbf{0}_{3,1}$.

Given two points on the surface of the Earth (height h in ellipsoidal coordinates is zero) whose longitudes neither coincide nor are 180° apart, the geodesic algorithm of [57] provides latitude and longitude pairs along a geodesic curve connecting the points, along with a derivative of latitude with respect to longitude. That derivative can be transformed into an initial heading for a geodesic curve that would connect those points. Such a heading is generally expressed as radians or degrees North of East. North and East define the local tangent

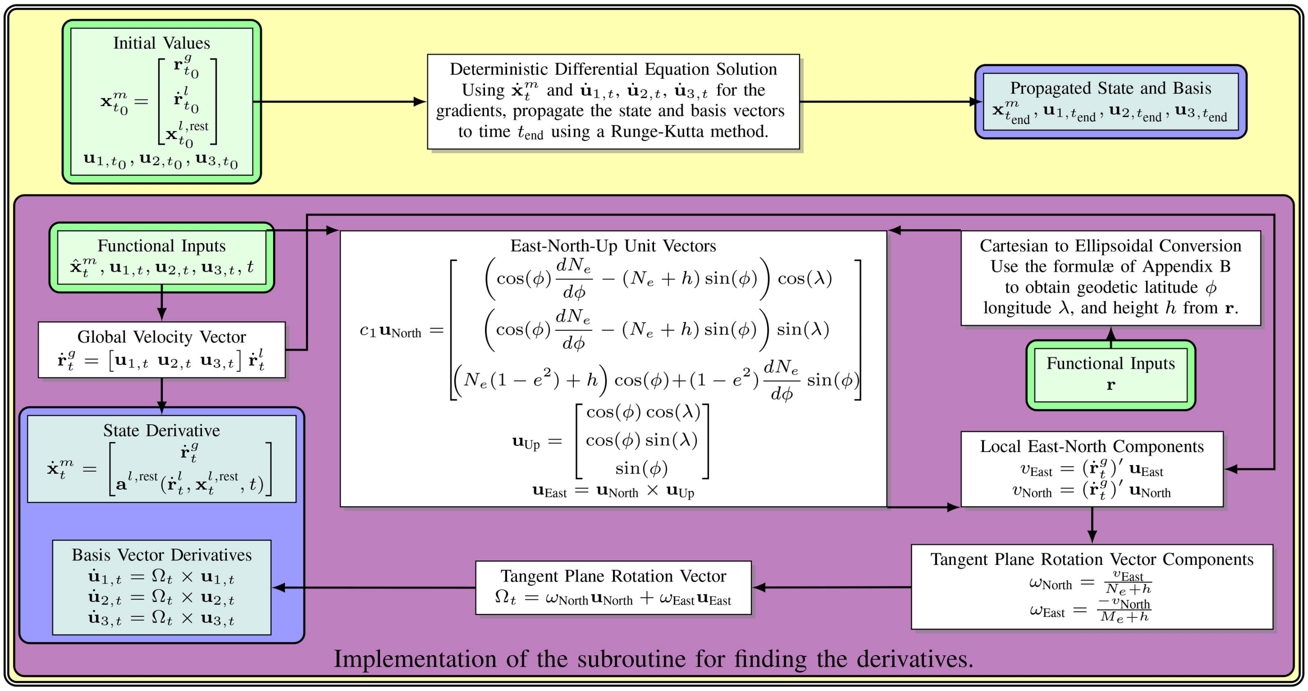

Initial Values
$\mathbf{x}_{t_0}^m = \begin{bmatrix} \mathbf{r}_{t_0}^g \\ \dot{\mathbf{r}}_{t_0}^l \\ \mathbf{x}_{t_0}^{l,\text{rest}} \end{bmatrix}$
$\mathbf{u}_{1,t_0}, \mathbf{u}_{2,t_0}, \mathbf{u}_{3,t_0}$
Deterministic Differential Equation Solution
Using $\dot{\mathbf{x}}_t^m$ and $\dot{\mathbf{u}}_{1,t}$, $\dot{\mathbf{u}}_{2,t}$, $\dot{\mathbf{u}}_{3,t}$ for the gradients, propagate the state and basis vectors to time t_{end} using a Runge-Kutta method.
Propagated State and Basis
$\mathbf{x}_{t_{\text{end}}}^m, \mathbf{u}_{1,t_{\text{end}}}, \mathbf{u}_{2,t_{\text{end}}}, \mathbf{u}_{3,t_{\text{end}}}$
Functional Inputs
$\hat{\mathbf{x}}_t^m, \mathbf{u}_{1,t}, \mathbf{u}_{2,t}, \mathbf{u}_{3,t}, t$
Global Velocity Vector
$\dot{\mathbf{r}}_t^g = [\mathbf{u}_{1,t}\ \mathbf{u}_{2,t}\ \mathbf{u}_{3,t}]\, \dot{\mathbf{r}}_t^l$
State Derivative
$\dot{\mathbf{x}}_t^m = \begin{bmatrix} \dot{\mathbf{r}}_t^g \\ \mathbf{a}^{l,\text{rest}}(\dot{\mathbf{r}}_t^l, \mathbf{x}_t^{l,\text{rest}}, t) \end{bmatrix}$
Basis Vector Derivatives
$\dot{\mathbf{u}}_{1,t} = \Omega_t \times \mathbf{u}_{1,t}$
$\dot{\mathbf{u}}_{2,t} = \Omega_t \times \mathbf{u}_{2,t}$
$\dot{\mathbf{u}}_{3,t} = \Omega_t \times \mathbf{u}_{3,t}$
East-North-Up Unit Vectors
$c_1 \mathbf{u}_{\text{North}} = \begin{bmatrix} \left(\cos(\phi)\frac{dN_e}{d\phi} - (N_e + h)\sin(\phi)\right)\cos(\lambda) \\ \left(\cos(\phi)\frac{dN_e}{d\phi} - (N_e + h)\sin(\phi)\right)\sin(\lambda) \\ \left(N_e(1-e^2) + h\right)\cos(\phi) + (1-e^2)\frac{dN_e}{d\phi}\sin(\phi) \end{bmatrix}$
$\mathbf{u}_{\text{Up}} = \begin{bmatrix} \cos(\phi)\cos(\lambda) \\ \cos(\phi)\sin(\lambda) \\ \sin(\phi) \end{bmatrix}$
$\mathbf{u}_{\text{East}} = \mathbf{u}_{\text{North}} \times \mathbf{u}_{\text{Up}}$
Cartesian to Ellipsoidal Conversion
Use the formulæ of Appendix B to obtain geodetic latitude ϕ longitude λ, and height h from $\mathbf{r}$.
Functional Inputs
$\mathbf{r}$
Local East-North Components
$v_{\text{East}} = (\dot{\mathbf{r}}_t^g)' \mathbf{u}_{\text{East}}$
$v_{\text{North}} = (\dot{\mathbf{r}}_t^g)' \mathbf{u}_{\text{North}}$
Tangent Plane Rotation Vector Components
$\omega_{\text{North}} = \frac{v_{\text{East}}}{N_e + h}$
$\omega_{\text{East}} = \frac{-v_{\text{North}}}{M_e + h}$
Tangent Plane Rotation Vector
$\Omega_t = \omega_{\text{North}}\mathbf{u}_{\text{North}} + \omega_{\text{East}}\mathbf{u}_{\text{East}}$
Implementation of the subroutine for finding the derivatives.

Fig. 6. The steps for predicting forward a flat-Earth dynamic model on a curved Earth when using an ellipse of revolution approximation for the gravitational shape of the Earth. $\mathbf{r}_t^g$ is the target position in the global (curved-Earth) coordinate system; $\dot{\mathbf{r}}_t^l$ is the target velocity in the flat Earth dynamic model and $\mathbf{x}_t^{l,\text{rest}}$ are other state components in the flat-Earth dynamic model. $\mathbf{u}_{\text{North}}$ is obtained by dividing the quantity on the right by its magnitude. $\mathbf{a}^{l,\text{rest}}$ is the function providing derivatives of the velocity and other state components in the flat-Earth dynamic model. The simplest initial set of basis vectors is $\mathbf{u}_{2,t_0} = \frac{d\mathbf{r}}{d\phi} / \left\| \frac{d\mathbf{r}}{d\phi} \right\|$ (North), $\mathbf{u}_{3,t_0} = \frac{d\mathbf{r}}{dh} / \left\| \frac{d\mathbf{r}}{dh} \right\|$ (Up), and $\mathbf{u}_{1,t_0} = \mathbf{u}_{2,t_0} \times \mathbf{u}_{3,t_0}$ (East), using the expressions of Section IV-B.

plane in the ENU coordinate system, and an angle North of East is an angle measured from the East axis in the direction of the North axis. By initializing the local coordinate axes $\mathbf{u}_1$, $\mathbf{u}_2$, and $\mathbf{u}_3$ at time 0 to be ENU, then if θ is the initial bearing provided by the algorithm of [57], the corresponding initial local (flat-Earth) velocity vector is[11]

$$\dot{\mathbf{r}}_t^l = v \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \\ 0 \end{bmatrix}, \tag{55}$$

where the speed $v = \|\dot{\mathbf{r}}_t^l\|$ can be set arbitrarily and determines how long it takes to travel between the points.

When using the algorithm in Fig. 6, the speed and number of steps needed for the Runge-Kutta algorithm to propagate the state a certain distance are related through the time duration of the step size. If the geodesic distance between the points on the surface of the Earth provided by the algorithm of [57] is s, and one wishes to use N_{steps} of time duration Δ_t for the Runge-Kutta method to integrate from the starting point to the desired ending point, then

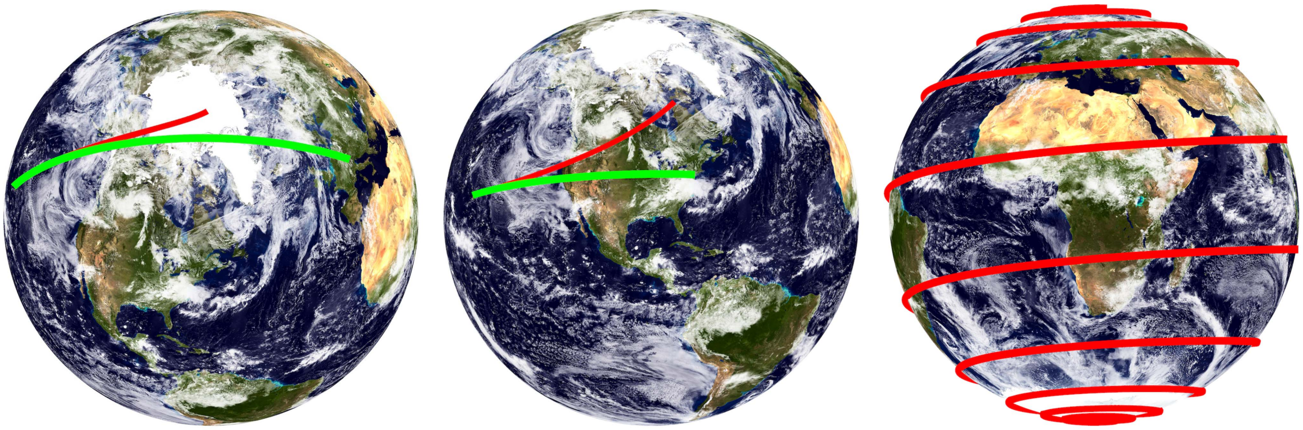
$$\|\dot{\mathbf{r}}_t^l\| = v = \frac{s}{N_{\text{steps}}\Delta_t} \tag{56}$$

is the speed needed for the target state.

The thick, green line in Fig. 7(a) is the geodesic curve on the surface of the Earth between the Mauna Kea Observatory in Hilo, Hawaii, which is located at approximately −155.470° East longitude and 19.823° North latitude [55, Sec. J], and Neuschwanstein Castle in Füssen, Germany, which, according to Google Maps,[12] is approximately located at 47.5575° latitude and 10.7500° longitude. The geodesic curve (which assumes a constant zero altitude above the reference ellipsoid) is determined using the algorithm of [57]. When the algorithm of Fig. 6 is used with a constant-velocity motion model propagated for the same distance (approximately 12,416 km) and the same heading as that determined by the algorithm of [57], the results differ by 0.0276 cm (1,000 Runge-Kutta steps spaced one second apart were used for the propagation). In Fig. 7(b), the geodesic trajectory under consideration goes from the Mauna Kea Observatory to New York City, which according to Google Earth is located at approximately 40.67° latitude and −73.94° longitude. Again, the difference between the stopping point of the geodetic estimation algorithm and the propagation algorithm of Fig. 6 is 0.00727 cm after a travel distance of 7,903 km.

[11] The algorithm of [35] and [34] is more robust than [57], but does not directly provide the latitude versus longitude of the curve to plot.

[12] Note that although Google Maps can be used to get approximate positions, its maps use the "web Mercator projection," which can result in positional biases to be as high as 40 km in some areas compared to a correct WGS-84-coordinates [53].



(a) Hilo to Füssen

(b) Hilo to New York

(c) A Constant-Bearing Path

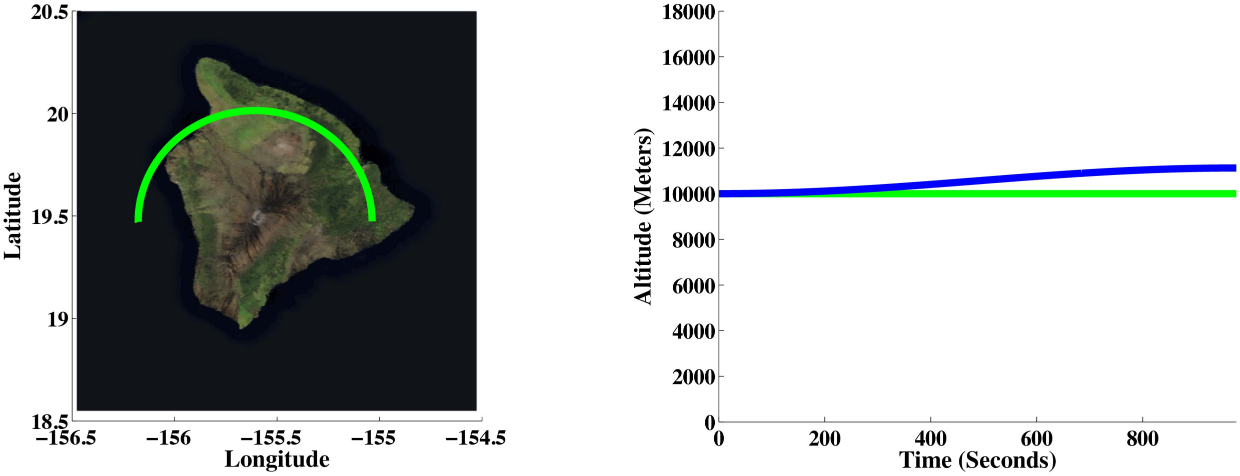
Fig. 7. In (a) and (b), the thicker, green curve is a geodesic curve from Hilo, Hawaii to Füssen, Germany in (a) and to New York City in (b). At the scale shown, the geodesic curve, which is determined using the algorithm of [57], is indistinguishable from the prediction of a flat-Earth constant-velocity motion model applied to the spheroidal Earth when initialized with the same bearing. The red curve in (a) and (b) is a curve of constant bearing starting with the same bearing as the geodesic curve. That is the curve that the algorithms of [43], [81] mapping a non-maneuvering flat-Earth model to a curved Earth would follow. The curve gets stuck at the North Pole in (a) and deviates greatly North in (b). In (c), a curve of constant bearing heading 5° North of East starting in Antarctica is shown and clearly does not follow a geodesic path. The globe image is derived from an equirectangular projection provided by Visible Earth, which is run by the EOS Project Science Office at the NASA Godard Space Flight Center. Full credits for the image are at [68], [71]. The projection is re-mapped to a spherical Earth using the code of [44] (though the code is labeled for a Mercator projection, it actually uses an equirectangular projection), with minor modifications to use the WGS-84 ellipsoid. The images were made in Matlab and touched-up in Gimp, since Matlab tends to render three-dimensional lines in a broken, jagged manner.

The primary source of the small difference in the estimates between the traditional geodetic algorithm of [57] and the algorithm of Fig. 6 appears to stem from numerical precision errors that accumulate over the integration period. For example, after converting the location of Hilo to Cartesian coordinates using the formulae of Section IV-B and then back to ellipsoidal coordinates using the formulae of Appendix B, the heights differ by 9.3×10^{-10} m and the latitudes differ by 4.2×10^{-11} degrees. In comparison, the ratio of the error of the final Cartesian location to the distance traveled for the algorithm of Fig. 6 is 2.2×10^{-11} for the trajectory to Neuschwanstein and 9.1×10^{-12} for the path to New York. Consequently, the algorithm in this paper, which is implemented using double-precision arithmetic under a constant-velocity motion model, agrees with the geodesic algorithm of Appendix B to a reasonable precision.

The solid red lines in Fig. 7 are constant-bearing trajectories. Using the tracking techniques of [43] and [81], a target under a constant-velocity motion model propagated forward in time would follow a constant-bearing trajectory. Similarly, for purposes of trajectory generation, one might consider using a constant-bearing trajectory on a curved Earth. In Figures 7(a) and 7(b), the initial bearing matches that of the geodesic. However, it is clear that the constant-bearing trajectories deviate greatly from the geodesic path, because following a constant heading in a local ENU coordinate system always deviates toward the poles, except when following lines of constant latitude. In Fig. 7(c) a track maintaining a constant-bearing trajectory is started at a low angle of 5° North of East at a latitude of $\phi = -89°$, a longitude of $\lambda = 0°$, and a height of 0 m, which places it somewhere in Antarctica. The resulting rhumb line is observed to spiral greatly going up the globe, and is clearly not a geodesic curve. In other words, constant-bearing trajectories are not suitable for modeling non-maneuvering target motion near the poles or over long distances.

Next, the algorithm in this paper is considered for mapping a maneuvering target state to a curved Earth. Three models representing well-known maneuvers are chosen. The first model is a level coordinated turn model, as described in Appendix C. The second is a weaving motion model derived in Appendix D, and the third is a spiraling motion model, as derived in Appendix E. Appendices D and E also describe how to make the flat-Earth weaving and spiraling models travel a certain distance in a desired direction, since those designing simulations will generally want to know how to make a target go from one desired location on the Earth to another. It will be shown that while the algorithm derived in this paper correctly keeps the targets from flying off into space, the curvature of the Earth induces a slight drift in the trajectories in the local tangent plane compared to a geodesic starting in the same direction when adapting flat-Earth navigation methods to a curved Earth. Nonetheless, the flat-Earth equations for the overall trajectory to have a particular offset along an initial bearing as given in the appendices are good approximations to the curved-Earth result.

For the coordinated turn model, an example of an aircraft circling the Mauna Loa Volcano in Hawaii is considered. The Mauna Loa volcano is located at a



(a)

(b)

Fig. 8. The trajectory of a coordinated-turning plane around Hawaii as found using the algorithm of Fig. 6 is plotted in (a). At this angle and scale, the flat-Earth approximation would appear to be the same. Using the algorithm of Fig. 6, the target only differs from its starting altitude by 2.8 nm at the end due to numerical precision errors. On the other hand, using the flat-Earth model, the plane ends 1.13 km too high, as illustrated in (b), where the green line is the correct curved-Earth model and the blue-line is a flat-Earth approximation centered at the start of the target trajectory. The land image is again from Visible Earth [68], [72] and the image's contrast was increased to make the land more visible. The land image was displayed in Matlab using the `imread` and `imagesc` commands. To save the resulting plot while preventing Matlab from rendering the trajectory and axes as (pixellated) bitmaps, the trajectory and axes were saved separately from the land in encapsulated postscript (EPS) format, which is a format that supports vector graphics. The axes and trajectory were then combined with the plotted land image using PowerPoint, and the mixed vector/bitmap image was exported in portable document format (PDF). Had everything been plotted together, Matlab would have rasterized the vector graphics, even when saved in EPS format, causing the axes either to be very pixellated or the file size to be excessively large.

latitude of 19.475°, a longitude of −155.608° and is 4,170 m high [76]. The aircraft circling the volcano starts at a location 60 km along a geodesic East of the volcano (19.4741° latitude,−155.0365° longitude) at a heading of due North and an altitude of 10 km.[13]

To circle the volcano under a flat-Earth model, the radius of rotation r in Appendix C, should be approximately 60 km. The radius of rotation and the target speed are related by

$$\|\dot{\mathbf{r}}^l\| = v = r\omega, \tag{57}$$

where ω is the turn rate parameter in the coordinated turn model. Assuming that $v = 193$ m/s, which is a reasonable speed for a chartered private jet touring the island, as it is approximately the maximum cruise speed of an Eclipse 550 [17], the corresponding turn rate is approximately 0.184301°/s. The turn rate is only approximate when mapped to a curved surface, because Euclidean geometry is no longer valid and the time to go in a complete circle mapped to the surface of the ellipsoidal Earth is not necessarily 360° divided by the turn rate. Nonetheless, the flat-Earth (Euclidean) geometric approximation for the travel time is a reasonable approximation. Under Euclidean geometry, it would take about 33 minutes to circle the volcano.

If the aircraft is initially heading due North, at a constant speed for 16 minutes, the trajectory computed by the algorithm of Fig. 6 is plotted in Fig. 8 as a function of latitude and longitude using $N_{\text{steps}} = 1{,}000$ Runge-Kutta steps of duration $\Delta_t = 160$ ms. This trajectory maintains a constant altitude above the ground to within 2.8 nm at the final point in the trajectory (due to numerical precision errors). On the other hand, when using a flat-Earth coordinated turn model centered at the starting point[14] as described in Appendix C, the final target location is 1.13 km too high, with a total Cartesian offset of 1.15 km.

The second model under consideration is the weaving motion model of Appendix D. A target might travel in a weaving or other erratic manner over long distances to avoid detection while approaching a strategic destination. Here, two flat-Earth weaving models are mapped to the curved Earth. In the first model, the target weaves within the local tangent plane (xy plane) and should maintain a constant altitude above the ground. In the second model, the target performs weaves in the local vertical direction, and the weaves should not drift upward or downward over time.

[13] The location a desired distance along a geodesic East of the volcano can be found in the same way that the geodesic curves for Fig. 7 are evaluated. In this instance, the initial point is the volcano location, but set to zero altitude. One establishes an initial set of axes as an ENU coordinate system, and the initial local velocity is in the positive x direction. The algorithm in Fig. 6 is then used to travel at an arbitrary speed until the desired distance is traversed.

[14] The use of a local level coordinate system for putting flat-Earth models at any point on the Earth is described in [84, Ch. 3.2.2.7], where the flat-Earth model is placed on the local tangent plane of the starting point.

When using the weaving motion model of Appendix D, a target is considered flying a weaving trajectory at an altitude of 10 km from Mauna Loa to Honolulu (21.3000° latitude, $\lambda = -157.8167°$ longitude), according to Google Maps. The target's speed is set to 680 m/s, which is approximately Mach 2 and might just be attainable by the MiG in Table I. The relative amplitude of the weave is $\beta = 0.5$ with an initial phase of $\theta_0 = 0$ starting from time $t = 0$. The target performs $N_w = 6$ weaves along the path.

The geodesic distance from Mauna Loa to Honolulu is determined using the algorithm of [57] and is approximately $s \approx 306.552$ km. With the given weave parameters over this distance, Appendix D specifies the maximum acceleration felt by the pilot, ignoring gravity. Using the maximum gravity-free acceleration coupled with the G-force equations of Appendix C-B that account for gravity, the maneuver in the local tangent plane corresponds to maximum 4.2 G (load factor) turns. When the maneuver is in the vertical plane using the maximum acceleration of the turn with the maximum G-force equation of a vertical turn in Appendix C-B, one obtains an upper bound on the G-force felt by the pilot of 5.2 G. Both of those bounds are well within the structural limits of the aircraft listed in Table I, highlighting the realism of the maneuver.

The duration of the maneuver to travel the geodesic distance between Mauna Loa and Honolulu is determined using the method of Appendix D-B and is found to be approximately 529 s. However, when the target maintains an altitude of 10 km, it will stop short of the destination—the indirect geodetic problem solved with [57] is only valid for paths on the surface of the Earth. At higher altitudes, the target must travel farther to reach the same latitude and longitude.

To perform weaving motion in the local tangent plane, the axis of rotation used in the weaving model (the direction of Ω_t) is the local vertical $[0,0,1]'$, the z-axis. When the weaves are in the local vertical plane, then the axis of rotation is the normalized cross product of the initial (level-flight) velocity and the local vertical. In the simulations, 3500 Runge-Kutta steps were taken, which corresponds to a duration of approximately $\Delta_t =$ 151.2 ms per step in (56).

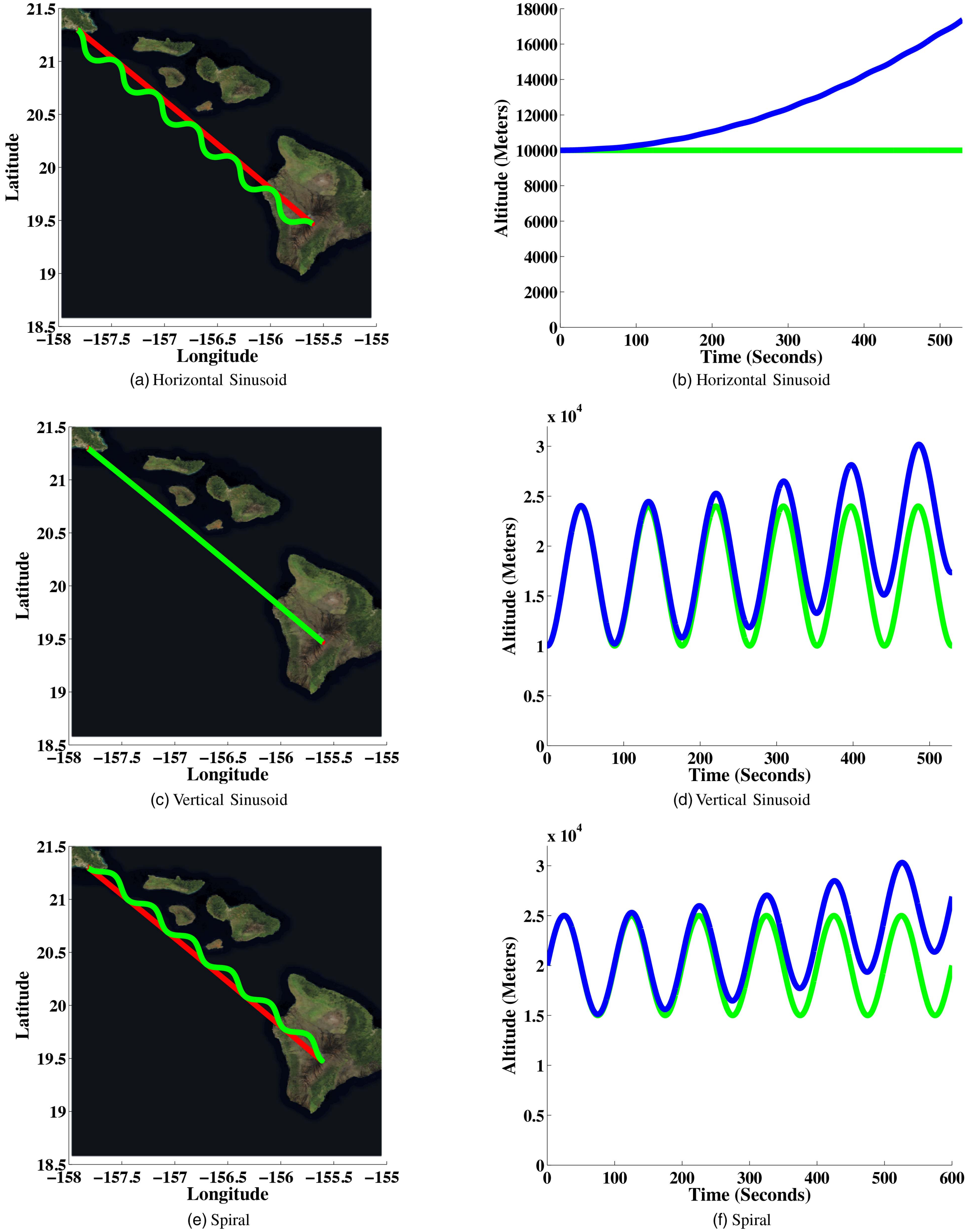
Figures 9(a) and 9(c) show the weaving trajectory after traveling the ground distance of the geodesic curve (as computed on the surface of the reference ellipsoid) for a target at an initial altitude of 10 km using the algorithm of Fig. 6. In Figures 9(a) and 9(b), the target weaves in the local tangent plane. At the final point, the target's altitude differs from the initial point by approximately 1.86 μnm when performing horizontal weaves and 57 nm when performing vertical weaves, which one would expect to be entirely due to numerical precision problems. However, the target's Cartesian location at the end differs from the end of the geodesic curve raised to the same altitude by 481 m when performing horizontal weaves and 798 m when performing vertical weaves. This offset in the local tangent plane is primarily because the geodesic curve is computed at zero altitude whereas the target flies at 10 km altitude, but also because the flattening factor of the Earth means that deviations in one direction do not perfectly cancel by deviation in another.

The third model considered is the spiraling model of Appendix E. The spiraling model is of interest, because full physics-based dynamic model derivations, such as in [73, Sec. 5], often omit expressions for a spiraling model on a curved Earth as being too difficult. While the flat-Earth model of Appendix E might not be the best physical representation of a spiraling target, it is sufficient to demonstrate the efficacy of the algorithm. As shown in Figures 9(e) and 9(f), the spiraling model varies its position in the horizontal and vertical planes, but does not drift upward.

The same trajectory is chosen for the spiraling model as for the weaving models, though the initial altitude is increased to 20 km. The spiraling model is implemented with 3500 Runge-Kutta steps over a period of $T = 600$ s. The angular velocity of the turn component is chosen to be $\omega = 2\pi N_w/T$ with $N_w = 6$ so that there would be 6 spiral cycles over the trajectory. The linear speed of the spiral track is the distance between the points divided by the time, which is approximately $\|\mathbf{v}_l\| = 481$ m/s. The magnitude of the orthogonal velocity causing the rotation is set to cause the radius of the spiral to be 5 km, which is approximately $\|\mathbf{v}_s\| = 314$ m/s. The total speed of the target is consequently 574 m/s, which is less than Mach 2 and is attainable by two of the fighter jets in Table I. The upper bound on the G-force of the target as described in Appendix E is approximately 3 G and is tolerable by all of the aircraft in Table I.

VI. CONCLUSIONS

This paper shows how deterministic flat-Earth dynamic models can be used on a curved Earth via a naturally evolving coordinate system. The technique avoids the highly unrealistic tendency of many other algorithms to spiral toward the poles. When the target is constrained to the surface of an arbitrary-shaped Earth, the solution is exact, assuming that a set of coordinates can be defined on the surface. Similarly, when the Earth is approximated as an ellipse of revolution, as in the WGS-84 standard, then a precise algorithm for simulating targets at arbitrary altitudes is available, because no curvature of the plumb line exists in the ellipsoidal model. The moving local coordinate system is equivalent to the wander coordinate system from the inertial navigation literature. The algorithm of Fig. 6 is a summary of the technique for use on a reference ellipsoid. When used with a non-maneuvering motion model, the algorithm can be used to trace out geodesic curves given an initial heading without encountering singularities in any direction or at the poles. Unlike traditional geodesic



(a) Horizontal Sinusoid

(b) Horizontal Sinusoid

(c) Vertical Sinusoid

(d) Vertical Sinusoid

(e) Spiral

(f) Spiral

Fig. 9. A target traveling between Mauna Loa and Honolulu. In (a) and (b), the target follows a weaving trajectory in the local tangent plane; in (c) and (d), the weaving is in the vertical plane, and in (e) and (f) the target spirals. In (a), (c) and (e) the straight red curve is the geodesic curve; in (c), the geodesic is covered by the trajectory. In (b), (c) and (d) the green trajectory, which does not rise, is the model correctly accounting for the curvature of the Earth; the blue lines, which rise, are local flat-Earth models centered at the starting point of the trajectory. It can be seen that ignoring the curvature of the Earth introduces a large bias in altitude. Again, the land image is from Visible Earth [68], [72].

algorithms, it can compute geodesic curves at constant altitudes above the reference ellipsoid.

Through simulation, the algorithm of Fig. 6 is shown to agree with a traditional technique for determining geodesic curves when applied to a non-maneuvering, constant-velocity motion model. The algorithm did not exhibit a tendency to veer toward the poles, as occurs when using a constant heading model to adapt tracks to a curved Earth as is the case when using more traditional ENU methods. The generality of the algorithm is demonstrated with coordinated turn, weaving, and spiraling motion models. The limiting factor in the approach appears to be the accumulation of numerical precision errors over time, resulting in small (nanometer) deviations from the target height.

In addition to presenting a technique for mapping arbitrary flat-Earth dynamic models for aircraft to a curved Earth, this paper also analyzes the extent to which the ellipsoidal Earth approximation is valid when accounting for the fact that aircraft generally use pressure altitude for navigation purposes. This analysis is in Appendix A. This paper also derived flat-Earth weaving and spiraling models in Appendices D and E that do not appear to exist in the literature. The models were derived along with methods for determining the G-forces felt by a pilot and determining how to travel a desired distance in a particular direction while maneuvering.

The most advanced algorithms for simulating target dynamics are probably those built into commercial and military flight simulators, such as the many listed in Jane's online database of Simulation and Training Systems (part of https://janes.ihs.com). However, for simple design of target tracking algorithms, the cost and computational resources required for the high-fidelity models in Janes might not be practicable, which is probably why such advanced models appear to be absent from the target tracking literature, where simple, generic flat-Earth models are commonly used. Moreover, highly precise models often have a large number of parameters to be estimated, not all of which are observable by a radar (comments on the observability of one sophisticated model are given in [40]). Consequently, medium-fidelity dynamic models might be more useful in many tracking algorithm designs. Thus, the technique presented in this paper can fall into line with those of [50], [62] for long-range maneuvering target simulations. However, unlike the methods of [50], [62], the algorithm of this paper can also simulate targets on non-ellipsoidal surfaces.

Future work can focus on developing state prediction algorithms that use meteorological data to model properly the altitude variations experienced by aircraft navigating using pressure altitude. Similarly, real aircraft do not fly deterministic trajectories. Ergo, future work can generalize the technique of this paper to map stochastic motion models to a curved Earth. Nonetheless, deterministic flat-Earth trajectories mapped to a curved Earth can specify nominal maneuvering trajectories used by real control systems on aircraft or missiles.

ACKNOWLEDGEMENTS

This research is supported by the Office of Naval Research through the Naval Research Laboratory (NRL) Base Program. The author would like to thank Dr. Eric Mokole, and Mr. Steven Brockett from NRL for their editorial feedback on this manuscript, Dr. Jimmy Alatishe and Mr. Jerry Kim for their discussions on differential geometry, and Mr. Scott R. Duncan for showing where free, high-resolution Earth imagery could be obtained.

APPENDIX A. HOW AIR PRESSURE VARIATIONS AFFECT CIVILIAN AVIATION

Aircraft in controlled airspace normally fly at altitudes computed from measuring a local barometric pressure and comparing it to a reference mean sea level value. The reference pressure at sea level in controlled airspace is usually set to the standard pressure of 29.92 inches of mercury [21]. The pressure altitude is normally determined using a standard atmospheric model. A number of global and regional standard atmospheric models are reviewed in [2], not all of which are relevant to aviation.[15] For civil aviation, the International Civil Aviation Organization (ICAO) Standard Atmosphere, the International Standard Atmosphere (ISA), from the International Organization for Standardization (ISO), and the U.S. Standard Atmosphere are the most relevant. As noted in [2], the ISA is identical to the ICAO standard atmosphere and the World Meteorological Organization (WMO) Standard Atmosphere up to an altitude of 32 km. The development of the U.S. Standard atmosphere is described in [47], where it is noted that the U.S. Standard Atmosphere and the ICAO standard atmosphere are identical up to an altitude of 32 km. The defining document for the U.S. Standard atmosphere is [54].[16] In the United States, civilian aircraft are limited to an altitude of 40,000 ft (12,192 m), with a single exemption for the Airbus A380-800, which can fly up to 43,000 ft (13,106.4 m) [14].

The U.S. Standard Atmosphere [54] provides equations relating altitude and pressure within the altitudes concerning civilian aviation as well as beyond those bounds. Three equations relating altitude and pressure are given in the standard. One is for altitudes from 11 km to 20 km and 47 km to 51 km. Another is for

[15] Additionally, some of the models are for Mars, Neptune, Titan, and Venus, rather than for the Earth.

[16] The ICAO standard atmosphere and the ISA are not directly cited, because the U.S. Standard Atmosphere can be freely downloaded, whereas the other two standards are rather expensive. For example, one can purchase the ISA from the ISO at http://www.iso.org/iso/catalogue_detail?csnumber=7472 for 210 Swiss Francs (≈ 230 U.S. dollars). Since all three standards agree from ground level to 32 km altitude, the freely available U.S. Standard Atmosphere generally suffices for civilian aviation purposes worldwide.

altitudes above 85 km, and the third is for all other altitudes, but with different parameters in different regions. Consequently, two different sets of equations are necessary to describe all altitudes up to 43,000 ft, the maximum altitude of a commercial aircraft in the United States. However, to simplify the analysis here, only the model from ground level to 11 km (36,089 ft) is considered.

The relationship between the altitude above mean sea level z and the atmospheric pressure for altitudes from ground level to 11 km is

$$z = \frac{r_0 h}{r_0 - h}, \tag{58}$$

where h is the geopotential altitude given by

$$h = \frac{T_0}{L_0}\left(\left(\frac{P}{P_0}\right)^{-L_0 R/g_0 M_0} - 1\right) \tag{59}$$

and

$$r_0 = 6{,}356{,}766 \text{ m} \qquad g_0 = 9.80665\frac{\text{m}}{\text{s}^2} \tag{60}$$

$$R = 8314.32\frac{\text{N m}}{\text{kmol K}} \quad M_0 = 18.9644\frac{\text{kg}}{\text{kmol}} \tag{61}$$

$$P_0 = 101{,}325.0 \text{ Pa} \qquad T_0 = 288.15 \text{ K} \tag{62}$$

$$L_0 = -6.5\times 10^{-3}\frac{\text{K}}{\text{m}}. \tag{63}$$

The value r_0 is the effective radius of the Earth and g_0 is the magnitude of the acceleration due to gravity at sea level, which differ from the more modern values derived from the WGS-84 standard [13]. The value R is the ideal gas constant, also known as the molar gas constant. The value of R in (61) is provided in [54], which differs slightly from the modern value standardized by the Committee on Data for Science and Technology (CODATA) in [48]. The quantities M_0 and P_0 are respectively the mean molecular weight of the atmospheric constituents at the surface of the Earth and the standard atmospheric pressure at sea level. The air pressure at the surface of the Earth P_0 is approximately equal to 29.92 inches of mercury on a barometer at the standard temperature $T_{M,0} = 15°\text{C}$. Note that many sources erroneously use the geopotential altitude in (59) as the true altitude instead of (58).

The relationship between the geopotential altitude h and the true altitude above sea level z is

$$h = \frac{1}{g_0}\int_0^z \|\mathbf{g}\| dz, \tag{64}$$

where $\mathbf{g}$ is the acceleration due to gravity at altitude z and g_0 is the magnitude of the acceleration due to gravity at mean sea level.[17] Equation (59) comes from using a very simple approximation to the magnitude of the acceleration due to gravity at height z, specifically,

$$\|\mathbf{g}\| = g_0\left(\frac{r_0}{r_0 + z}\right)^2. \tag{65}$$

In other words, by modern standards, the U.S. Standard atmosphere is not a high-precision model. However, it forms the basis of pressure-based altimeters for civilian aviation and is a good approximation for basic analysis.

To determine how much local pressure variations can affect the true altitude of an airplane flying according to a pressure altitude, an example of an aircraft flying at a pressure altitude of 7,620 m (25,000 ft) is considered. Using (58) with the standard parameters shown, the pressure at the aircraft's altitude is $P = 52{,}991.6$ Pa. The maximum and minimum (excluding tornadoes) recorded sea-level air pressures are 108,330 Pa and 87,000 Pa [39]. Using $P = 52{,}991.6$ Pa for an aircraft at a constant pressure altitude of 7,620 m, but varying P_0 to both of the extremes, the true altitude varies from 5,953.45 m (19,532 ft) to 8,328.2 m (27,323 m).

In comparison, in the United States, aircraft are placed in 500 ft (≈ 152 m) flight levels. A flight level is the height in feet divided by 100. The use of flight levels begins at 18,000 ft, corresponding to flight level 180, denoted FL180. For pilots operating under instrument flight rules (IFR), which is generally all pilots flying in flight levels, air traffic controllers space the aircraft through FL410 a distance of 1,000 ft apart vertically. That is, they are two flight levels apart. Above FL410, aircraft are spaced 2,000 ft apart, except for supersonic and military aircraft [21]. If a pilot sets his altimeter to use the standard MSL air pressure and the air pressure is at the record minimum, then the pilot will fly about 10.9 flight levels lower, compared to the ground, than on a normal day. On the other hand, if the air pressure is the record high, then the pilot will fly about 4.6 flight levels too high.

In practice, the extreme low is unrealistically extreme, since it was taken in the middle of Typhoon Tip in 1979, and generally only research and military aircraft would knowingly venture into a strong hurricane. A more realistic lower bound for the air pressure is 98,000 Pa, which is the very bottom of the pressure range of a category 1 hurricane on the Saffir-Simpson scale [3], and is the value used as a lower bound on the atmospheric pressure in [18], where the feasibility of reducing the minimum vertical separation between planes (which was larger back then) was studied. In this instance, the minimum air pressure extreme leads to an altitude of 7261.4 m (23,823 ft), which is an offset of 2.4 flight levels. Additionally, when the air pressure extrapolated to mean sea level in a location is above 31 inches of mercury (≈ 104,966 Pa), air traffic control in the United States is supposed to instruct pilots to set their altimeters to use 31 inches of mercury as P_0 [21];

[17] Mean sea level is defined as a surface of constant gravitational potential. Surprisingly, it is *not* a surface where the magnitude of the gravitational acceleration $\|\mathbf{g}\|$ is constant. In [29, Ch. 2.9], the magnitude of the gravitational acceleration on an ellipsoidal approximation of the Earth is derived and varies with latitude. Thus, the assumption that g_0 is a single constant everywhere is just an approximation.

so in many instances the vertical offset due to pressure changes also spans fewer flight levels.

This variability in altitude of an aircraft traveling at a constant barometric altitude exceeds the variability in the height above the reference ellipsoid due to gravitational variations when using an ellipsoidal approximation of the Earth (Using the data in [52], the reference ellipsoid differs from mean seal level by up to +84 m and −107 m). Consequently, if one wants more precise aircraft prediction models on a curved Earth than the technique presented in this paper, one must consider the use of meteorological data, not just the use of a more precise geoid approximation than an ellipsoid.

APPENDIX B. CONVERTING FROM CARTESIAN TO GEODETIC ELLIPSOIDAL COORDINATES

The conversion from Cartesian (x_0, y_0, z_0) to ellipsoidal (ϕ, λ, h) coordinates is a difficult problem. A table of 80 references addressing the topic is given in [20]. Generally, an explicit, numerically stable solution is the most desirable. In this appendix, the solution of [66], which is a stabler form of [65], is reviewed. When using typical parameters for the Earth's ellipse, it is noted in [65] and [66] that this conversion is valid for all points outside of a small ellipsoid around the center of the Earth, whose radius is about 43 km. Considering that no one has ever managed to drill through the 6 km of crust of the Earth to reach the mantle, though it might be technically feasible [70], this restriction is in practice meaningless for all applications of this coordinate conversion outside of, perhaps, seismic research.

It is assumed that the semi-major axis a and semi-minor axis b of the reference ellipsoid are known and that the reference ellipsoid is centered on the Cartesian origin. Section IV describes the relationship between the different formulations of ellipsoidal parameters. Given the Cartesian point (x_0, y_0, z_0), the corresponding latitude, longitude, and ellipsoidal height are

$$\phi = \arcsin\left(\frac{z(\epsilon^2+1)}{N_e}\right) \tag{66}$$

$$\lambda = \arctan 2(y_0, x_0) \tag{67}$$

$$h = r_0\cos(\phi) + z_0\sin(\phi) - \frac{a^2}{N_e}, \tag{68}$$

where z, ϵ, N_e, and r_0 are found by computing

$$e^2 = 1 - \frac{b^2}{a^2} \qquad \epsilon^2 = \frac{a^2}{b^2} - 1 \tag{69}$$

$$r_0 = \sqrt{x_0^2 + y_0^2} \qquad p = \frac{|z_0|}{\epsilon^2} \tag{70}$$

$$s = \frac{r_0^2}{e^2\epsilon^2} \qquad q = p^2 - b^2 + s \tag{71}$$

$$u = \frac{p}{\sqrt{q}} \qquad v = \frac{b^2u^2}{q} \tag{72}$$

$$P = 27\frac{vs}{q} \qquad Q = \left(\sqrt{P+1} + \sqrt{P}\right)^{2/3} \tag{73}$$

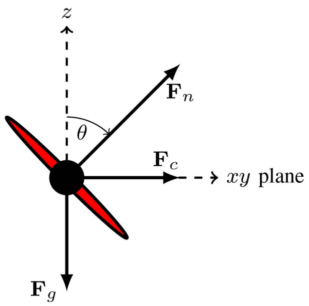


Fig. 10. The force diagram used to derive the relationship between the turn rate and the load factor. The force $\mathbf{F}_g$ is the force due to gravity and is aligned with the negative z-axis. The force $\mathbf{F}_c$ is the centripetal force, which points toward the center of the circle of rotation when turning. During a level, coordinated turn, the aircraft remains in the xy plane, which is perpendicular to the z-axis, with θ being the bank angle of the aircraft. In the ideal scenario, a lift force $\mathbf{F}_n$ is only produced in a direction perpendicular to the wings of the aircraft. When the aircraft is not turning, the lift is straight up and counters $\mathbf{F}_g$. When the aircraft is in a coordinated turn, the lift is such that the plane does not move in the z direction during the turn and such that the required centripetal force $\mathbf{F}_c$ is present to make the turn occur.

$$t = \tfrac{1}{6}(1 + Q + Q^{-1})$$

$$c = \sqrt{u^2 - 1 + 2t} \tag{74}$$

$$w = \frac{c-u}{2} \tag{75}$$

$$z = \text{sign}(z_0)\sqrt{q}\left(w + \sqrt{\sqrt{t^2+v} - uw - \frac{t}{2} - \frac{1}{4}}\right) \tag{76}$$

$$N_e = a\sqrt{1 + \frac{\epsilon^2 z^2}{b^2}}. \tag{77}$$

The term e is known as the first numerical eccentricity; ϵ is the second numerical eccentricity. The four-quadrant inverse tangent in (67) is not uniquely defined at the poles, when $x = 0$ and $y = 0$. However, many implementations of the four-quadrant inverse tangent, such as the `atan2` function in Matlab, will return zero in that instance. On the other hand, the `ArcTan` function in Mathematica correctly returns an indeterminate quantity. In a practical system, it is often preferable for a zero longitude to be returned at the poles rather than something indeterminate.

Care must be taken with the outer square root in (76), because when the argument is near zero, finite precision problems have been observed to cause the argument to become slightly negative. A practical implementation should check for negativity and insert zero instead. A similar check might be necessary for the square root in (74).

APPENDIX C. A FLAT-EARTH PLANAR TURN MODEL

A. Deriving a Model for Circular, Planar Turns

A coordinated turn is one where an aircraft maintains a constant altitude and speed. Figure 10 is an approximate free body diagram (ignoring drag, wind, and the Coriolis force) of the forces affecting the center of mass of a turning aircraft. The primary forces are gravity $\mathbf{F}_g$, which in a flat-Earth model can be aligned with the local z-axis, and lift $\mathbf{F}_n$, which is assumed to act normally to the body of the aircraft. To maintain a constant altitude, $\mathbf{F}_g + \mathbf{F}_c$ should act perpendicularly to the gravity vector. If $\|\mathbf{F}_c\|$ is constant, then the shape of the aircraft's maneuver will be a circle.

Constant-speed circular motion causes the acceleration to be orthogonal to the velocity. If $\ddot{\mathbf{r}}_t$ denotes acceleration, then

$$\ddot{\mathbf{r}}_t = \Omega_t \times \dot{\mathbf{r}}_t, \tag{78}$$

where × denotes the cross product, the direction of Ω is the axis of rotation, and the magnitude of Ω_t determines the turn rate at time t. For a coordinated turn on a flat Earth, the turn must remain in the xy plane. Consequently, the full dynamic model is

$$\mathbf{a}(\mathbf{x}_t, t) = \begin{bmatrix} \dot{\mathbf{r}}_t \\ \Omega_t \times \dot{\mathbf{r}}_t \\ 0 \end{bmatrix} \qquad \Omega_t = \begin{bmatrix} 0 \\ 0 \\ \omega_t \end{bmatrix} \tag{79}$$

for a target state $\mathbf{x}_t = [\mathbf{r}_t, \dot{\mathbf{r}}_t, \omega_t]'$. The model in (79) is the same as the basic two-dimensional horizontal coordinated turn model in [40]. This model can easily be expanded to a turn in an arbitrary plane by setting $\Omega_t = \omega_t \mathbf{u}_{\text{rot}}$, where $\mathbf{u}_{\text{rot}}$ is the rotation axis desired. To make the model a flat-Earth approximation in a local tangent plane on a curved Earth, set $\mathbf{u}_{\text{rot}}$ to the local vertical at the point of reference. In Section V, the point of reference was taken to be the starting point of the trajectory.

B. Relating Turn Rates to G-Forces

Section V related the turns in the simulations to the force felt by the pilot so as to justify the realism of the turn rates used. One feels 1 G when standing on the ground. A pilot will typically lose consciousness for turns producing between 4.5 and 5 Gs, whereas military pilots wearing "anti-G" suits can sustain up to 8 Gs. Combat aircraft are usually designed for load factors of at least 8 Gs [45]. Obviously, unmanned aircraft could be designed to withstand turns that would render a pilot unconscious. A reasonable baseline for simulation design can be the parameters for the F-16 Fighting Falcon, which are given in Table I.

The term ω_t is the turn rate in radians per second and can be related to the force felt by the pilot using simple equations for centripetal acceleration. In the coordinated turn model, ignoring wind and drag, the force that the aircraft must generate to maintain altitude and speed during the turn must counteract both the gravitational force and the centripetal force, as shown in Fig. 10. The centripetal force can be written from the acceleration in (79) as

$$\mathbf{F}_c = m\Omega_t \times \dot{\mathbf{r}}, \tag{80}$$

where m is the mass of the aircraft. The force that the aircraft must generate to maintain altitude and speed during the turn must counteract both the gravitational force and the centripetal force, as shown in Fig. 10. This normal force, designated by $\mathbf{F}_n$, is the lift produced by the plane such that a centripetal force of $\mathbf{F}_c$ is obtained. The normal force is thus

$$\mathbf{F}_n = \mathbf{F}_c - \mathbf{F}_g \tag{81a}$$

$$= m\Omega_t \times \dot{\mathbf{r}} - m\mathbf{g}, \tag{81b}$$

where $\mathbf{g}$ is the acceleration due to gravity ($\|\mathbf{g}\| \approx 9.81\text{m/s}^2$) on the surface of the Earth.

Dividing out the mass in (81b), the acceleration is obtained. When the magnitude of the acceleration is divided by $\|\mathbf{g}\|$, the load factor, which is the G-force felt by the pilot, is obtained[18]

$$\text{G-force} = \frac{\|\Omega_t \times \dot{\mathbf{r}} - \mathbf{g}\|}{\|\mathbf{g}\|}. \tag{82}$$

Equation (82) is useful in that it can express the G-force felt by the pilot not only for coordinated turns, but also for circular turns in any axis. In such an instance, the vector Ω_t need not point to the local vertical. Equation (82) was used in Section V to find the maximum G-force of the turns under the weaving maneuver model of Appendix D for weaves outside of the local tangent plane.

When performing a level coordinated turn, that is Ω_t is as in (79), then (82) can be simplified to

$$\text{G-force} = \frac{\sqrt{(\omega_t)^2\|\dot{\mathbf{r}}\|^2 + \|\mathbf{g}\|^2}}{\|\mathbf{g}\|}. \tag{83}$$

In such an instance, the G-force can be related to the bank angle θ of the aircraft in Fig. 10 by substituting (83) into the expression for the tangent of θ to obtain

$$\text{G-force} = \sqrt{(\tan[\theta])^2 + 1}. \tag{84}$$

On the other hand, if the turn is in the vertical direction, then the maximum G-force is felt when the cross product of the turn axis and velocity vector aligns with the gravitational acceleration and is

$$\text{G-force} = 1 + \omega\frac{\|\dot{\mathbf{r}}\|}{\|\mathbf{g}\|}. \tag{85}$$

More sophisticated aircraft dynamic models, such as [49] and other models based on aircraft aerodynamics as surveyed in [40], require significantly more elements in the target state. Aircraft turns need not always be circular in nature, especially if the turn is not in the tangent plane. For example, parabolic trajectories are often used

[18] Equation (82) does not differentiate between positive (feel pushed down in one's seat) and negative (feeling pulled out of one's seat) G-forces (load factors). Aircraft typically have different tolerances for positive and negative load factors before they break. As per Part 23, Section 337 of the FAA's Federal Aviation Regulations, the negative load factor that small aircraft in the United States must withstand is only 40%–50% the magnitude of the maximum tolerable positive load factor.

to obtain weightless environments without going into space [33]. Nonetheless, the model in this appendix can be useful, because it requires fewer parameters than the sophisticated model of [49] and can thus be initialized with a smaller number of measurements if used in a target tracker. For general circular turns, the orientation of Ω_t would also have to be estimated.

APPENDIX D. A FLAT-EARTH WEAVING TARGET MODEL

A. Deriving a Model for Variable-Rate Planar Turns

Weaving target models often arise when designing proportional navigation systems for missiles, such as in the examples discussed in [82, Chs. 21, 26, 30] and [56], where weaving is considered as a countermeasure. A simple weaving target model can be obtained by varying the turn rate ω_t in the coordinated turn model of Appendix C as a function of time.

Consider setting

$$\omega_t = A\cos(\alpha t + \theta_0), \tag{86}$$

where α is the weave rate, θ_0 is a phase offset, and A is the weave amplitude. The weave amplitude determines how much the target varies from a straight trajectory. For simplicity, it is assumed that $A > 0$. A reasonable upper bound on the weave amplitude would be such that the target traces out a series of 180° turns that alternate in direction so that the target moves forward. A weave amplitude resulting in total turns greater than 180° would have the target doubling back on its path in parts and not necessarily following a clear trajectory.

To determine the upper bound on A, consider the case where $\theta_0 = 0$, without loss of generality. In such an instance, at time $t = 0$, ω_t is at its maximum. The integral of ω_t over time gives the total turn. For a target to make a series of 180° turns of alternating directions, the accumulated angular change should be a quarter of a circular rotation; that is

$$\frac{\pi}{2} = \int_0^{\pi/2\alpha} A\cos(\alpha t)dt. \tag{87}$$

Consequently, the maximum allowable value of A before the trajectory starts describing loops is

$$A \leq \frac{\pi\alpha}{2}. \tag{88}$$

Smaller values of A make the track straighter. To make the overall direction of the weaving track follow a desired path, set the initial velocity point in the overall direction of the path and set $\theta_0 = -\alpha t$ at the initial time t when the weaving trajectory begins.

B. Determining How to Arrive At a Desired Point

In the simulations, the parameters were chosen such that a desired integer number of weaves were made from the starting position to get to an ending position a known distance away. In this section, the time and speed needed to perform the desired number of weaves and end up at the desired ending position when the weaves are made is derived. It is assumed that the track begins with $\alpha t + \theta_0 = 0$ and, without loss of generality, that the x-axis is aligned with the initial direction of motion and the weave is in the xy plane. The assumption that $\alpha t + \theta_0 = 0$ can be replaced by setting the initial time to 0 and $\theta_0 = 0$. These assumptions mean that the trajectory reaches the desired point at a time when $y = 0$. Using (86) and (79), along with these assumptions, the accelerations in the x and y coordinates can be written

$$\ddot{r}_{x,t} = -A\dot{r}_{y,t}\cos(\alpha t) \tag{89}$$

$$\ddot{r}_{y,t} = A\dot{r}_{x,t}\cos(\alpha t). \tag{90}$$

Solving for $\dot{r}_{y,t}$ in (89),

$$\dot{r}_{y,t} = -\frac{\ddot{r}_{x,t}}{A\cos(\alpha t)}, \tag{91}$$

and taking the derivative of (91) with respect to time yields

$$\ddot{r}_{y,t} = -\frac{\dddot{r}_{x,t} + \alpha\ddot{r}_{x,t}\tan(\alpha t)}{A\cos(\alpha t)}. \tag{92}$$

By combining equations (90) and (92), one obtains the differential equation

$$0 = \frac{\dddot{r}_{x,t}}{A\cos(\alpha t)} + \frac{\ddot{r}_{x,t}}{A\cos(\alpha t)}(\alpha\tan(\alpha t)) + A\dot{r}_{x,t}\cos(\alpha t). \tag{93}$$

It can be verified that a solution to (93) is

$$\dot{r}_{x,t} = c_1\cos\left(\frac{A\sin(\alpha t)}{\alpha}\right) + c_2\sin\left(\frac{A\sin(\alpha t)}{\alpha}\right). \tag{94}$$

The values of c_1 and c_2 can be directly found from the initial conditions at time $t = 0$: $\dot{r}_{x,t}|_{t=0} = \dot{r}_{x,0}$ and $\dot{r}_{y,t}|_{t=0} = 0$. Applying these initial conditions to (91) and (94) yields

$$c_1 = \dot{r}_{x,0} \qquad c_2 = 0, \tag{95}$$

so that (94) becomes

$$\dot{r}_{x,t} = \dot{r}_{x,0}\cos\left(\frac{A\sin(\alpha t)}{\alpha}\right). \tag{96}$$

Next, it is known that at the final time on the trajectory t_{end}, which is still unknown, the y position of the target should be equal to the initial y position (it ends with an integer number of weaves). Without loss of generality, assume that $r_{y,0} = 0$, which implies

$$r_{y,t_{\text{end}}} = \int_0^{t_{\text{end}}} \dot{r}_{y,t}dt = \int_0^{t_{\text{end}}} \dot{r}_{x,0}\sin\left(\frac{A}{\alpha}\sin(\alpha t)\right)dt = 0. \tag{97}$$

The ending position with respect to the y axis is zero when

$$\alpha = \frac{2\pi N_w}{t_{\text{end}}}, \tag{98}$$

where N_w is the positive integer number of weaves desired, because the integral is over an integer number of periods of the inner sine in (97) and the outer sine has no phase offset. Thus, for every t such that the argument of the integral has a positive value, there exists a t offset by a fraction of the period of the inner sine to cancel that.

Using (96) and (98), and assuming without loss of generality that the $r_{x,0} = 0$, the distance traveled in the x direction up to time t_{end} is

$$\begin{aligned} r_{x,t_{\text{end}}} &= \int_0^{t_{\text{end}}} \dot{r}_{x,t}dt \\ &= \dot{r}_{x,0}\int_0^{t_{\text{end}}} \cos\left(\frac{At_{\text{end}}}{2\pi N_w}\sin\left(\frac{2\pi N_w}{t_{\text{end}}}t\right)\right)dt. \quad (99) \end{aligned}$$

The distance traveled $r_{x,t_{\text{end}}}$ is a known quantity that gets one to the desired location along a geodesic curve from the initial location. The only unknown term in (99) is the travel time t_{end}.

From (98) and (88), the bound on A becomes

$$A \leq \frac{\pi^2 N_w}{t_{\text{end}}}. \quad (100)$$

If A is equal to the maximum value in (100), the argument of the cosine function in (99) is within the range $\pm\pi/2$, so the cosine function is always positive. However, if a larger A is used, then the cosine function can be negative, causing the target to go backwards in the x direction. The backwards motion of the target with A larger than the limit in (100) confirms the limit on A of Subsection D-A. To simplify the notation in the following optimization routine, A is expressed as a scale factor β $(0 < \beta \leq 1)$ times the maximum allowable value of A; that is,

$$A = \frac{\pi^2 N_w}{t_{\text{end}}}\beta. \quad (101)$$

Figure 11 shows how the shape of the curve changes with different values of β.

Given bounds on t_{end} ($t_{\text{end}} < 3r_{x,t_{\text{end}}}/\dot{r}_{x,0}$ was used as an *ad-hoc* upper bound in the simulations) and noting that (99) increases monotonically for allowable values of β, one can numerically estimate t_{end} with the optimization

$$\hat{t}_{\text{end}} = \arg\min_{t_{\text{end}}}\left|r_{x,t_{\text{end}}} - \dot{r}_{x,0}\int_0^{t_{\text{end}}} \cos\left(\beta\frac{\pi}{2}\sin\left(\frac{2\pi N_w}{t_{\text{end}}}t\right)\right)dt\right|. \quad (102)$$

Equation (102) can be numerically solved using a line search technique, such as the golden section search described in [4, Appendix C], with a numerical integration technique, such as those described in [6, Ch. 4]. In Matlab, one can use the function `fminbnd` to perform the minimization with `integral` for the numerical integration. In Mathematica, one can use the `NMinimize` command for the minimization with the `NIntegrate` command for the integration. Consequently, given a speed $\dot{r}_{x,0}$, a desired distance traveled along the initial direction of motion $r_{x,t_{\text{end}}}$, a value of β determining the relative amplitude of the weave, and the number of weaves N_w, the time necessary to travel the desired distance can be determined.

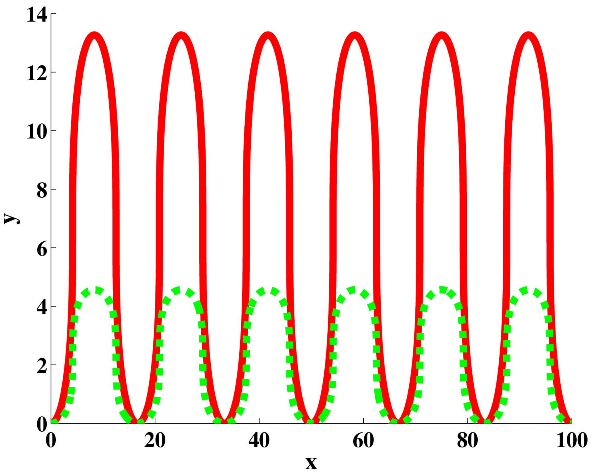


Fig. 11. The weaving motion for the same displacement, speed, and number of weaves varying only β, whereby t_{end} also changes so the same endpoint is reached. The solid red line is $\beta = 1$; the dashed green line is $\beta = 1/2$. The axes are in unit of meters, the speed is a constant 10 m/s, and $N_w = 6$.

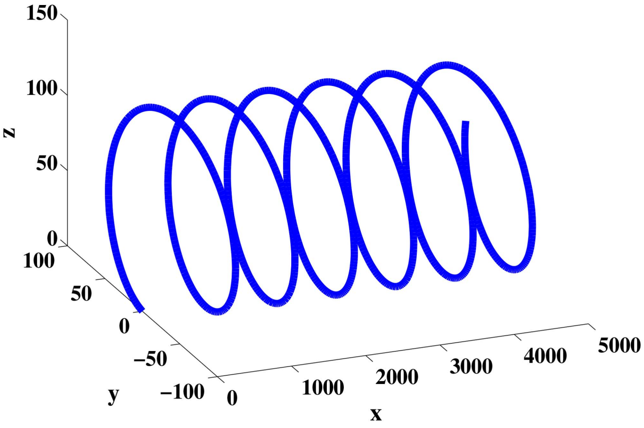


Fig. 12. An example of a spiraling trajectory starting from the origin with $\mathbf{v}_l = [100 \text{ m/s}, 0, 0]'$, $\mathbf{v}_s = [0, 50 \text{ m/s}, 0]'$, and $\omega = \pi/4$. The axes are in units of meters.

On the other hand, if one wishes to create a simulation that goes a particular distance $r_{x,t_{\text{end}}}$ weaving N_w times with amplitude scale factor β over a fixed period of time t_{end}, then the necessary speed is simply

$$\dot{r}_{x,0} = \frac{r_{x,t_{\text{end}}}}{\int_0^{t_{\text{end}}}\cos\left(\beta\frac{\pi}{2}\sin\left(\frac{2\pi N_w}{t_{\text{end}}}t\right)\right)dt}. \quad (103)$$

APPENDIX E. A FLAT-EARTH SPIRALING DYNAMIC MODEL

In [9], a non-ballistic flat-Earth spiraling dynamic model is derived by rotating the axis of a coordinates turn about the velocity vector.[19] One formulation of this

[19] Readers are advised to consider [59] if a ballistic spiraling model is desired.

model is

$$\ddot{\mathbf{r}} = \boldsymbol{\Omega} \times \dot{\mathbf{r}} \tag{104}$$

$$\dot{\boldsymbol{\Omega}} = \tilde{\omega}\left(\boldsymbol{\Omega} \times \frac{\ddot{\mathbf{r}}}{\|\dot{\mathbf{r}}\|}\right), \tag{105}$$

where $\boldsymbol{\Omega}$ is the instantaneous turn axis that is orthogonal to the velocity and $\tilde{\omega}$ is a constant affecting the rate of the spiral. Generally, $\boldsymbol{\Omega}$ does not have unit magnitude. The spiraling dynamic model of [9] is a constant-velocity model and ignores the effects of gravity. Unlike the weaving model of Appendix D, it is difficult to determine the actual direction in which the target will travel during the spirals.

Thus, an alternative constant-velocity spiraling model is presented here. Again, this is an approximate model that is not directly derived from aerodynamic and gravitational equations. The new spiraling model is

$$\dot{\mathbf{r}} = \mathbf{v}_l + \mathbf{v}_s \tag{106}$$

$$\dot{\mathbf{v}}_s = \omega\left(\frac{\mathbf{v}_l}{\|\mathbf{v}_l\|}\right) \times \mathbf{v}_s. \tag{107}$$

A drift equation corresponding to this model is

$$\mathbf{a}(\mathbf{x}_t, t) = \begin{bmatrix} \dot{\mathbf{r}} \\ \dot{\mathbf{v}}_s \\ \dot{\mathbf{v}}_s \end{bmatrix} \tag{108}$$

for a target state of $\mathbf{x}_t = [\mathbf{r}, \dot{\mathbf{r}}, \mathbf{v}_s]'$ ($\dot{\mathbf{r}}$ and $\mathbf{v}_s$ differ by a constant offset, so two of the derivative vectors in the drift equation are equal). Figure 12 shows an example of a trajectory arising from the spiraling dynamic model.

In practice, since $\mathbf{v}_s = \dot{\mathbf{r}} - \mathbf{v}_l$, the $\mathbf{v}_s$ term does not need to be stored directly in the state. In the model of (108), the velocity $\dot{\mathbf{r}}$ is the sum of orthogonal linear $\mathbf{v}_l$ and turning $\mathbf{v}_s$ components. The linear component is time-invariant. The turning component rotates about the linear component with a turn rate of ω rad/s. To make a target spiral in a particular direction, $\mathbf{v}_l$ should point in the desired direction of motion. The magnitude of $\mathbf{v}_l$ is the speed at which the target advances along the average trajectory about which the spiral turns, and $\mathbf{v}_s$ has to be orthogonal to $\mathbf{v}_l$. When programming a simulation of a spiraling target where the spiral neither ascends nor descends, it is reasonable to set the initial direction of $\mathbf{v}_s$ to the local vertical. In a flat-Earth model, this is generally the z-axis, since $\mathbf{v}_l$ would usually be placed in the xy plane. The magnitude of $\mathbf{v}_s$ determines the distance of the target from the center of the spiral as well as the G-forces felt by the pilot.

The relationship between the distance of the target from the center of the spiral and the magnitude of $\mathbf{v}_s$ can be determined by solving the differential equations for position. This determination can be made by assuming that $\mathbf{v}_l = [v_x^l, 0, 0]'$ and $\mathbf{v}_s = [0, v_s^y, v_s^z]'$. In such an instance, it can be verified that a solution to the differential equation in (107) has the form

$$v_s^y = c_1 \cos(\omega t) + c_2 \sin(\omega t) \tag{109}$$

$$v_s^z = c_2 \cos(\omega t) - c_1 \sin(\omega t), \tag{110}$$

where c_1 and c_2 are constants determined by the initial conditions and t is time. Assuming that at time $t = 0$, $v_s^y = 0$ and $v_s^z = v_M$, then $c_1 = 0$ and $c_2 = v_M$. Substituting $\mathbf{v}_s$, with y and z components given by (109) and (110), into (106) and integrating from time $t = 0$, at time t the target will have moved from its initial position by

$$\mathbf{r} = \begin{bmatrix} v_x^l t \\ \frac{v_M}{\omega}(1 - \cos(\omega t)) \\ \frac{v_M}{\omega}\sin(\omega t) \end{bmatrix}. \tag{111}$$

The magnitude of the y and z components, which are the only components affected by $\mathbf{v}_s$, is a constant v_M/ω and the rotational velocity is a constant $\|\mathbf{v}_s\| = v_M$. Consequently, the ratio of the rotational velocity to the turn rate determines the radius of the spiral. As for setting realistic bounds on the values so that the G-forces tolerable by a pilot will not be exceeded, the values in Appendix C-B are relevant given ω and the magnitude of $\mathbf{v}_s$, since those are the only velocity components that experience acceleration in the model.

Unlike with the weaving model of Appendix D, it is fairly simple to have the simplified flat-Earth spiraling target model go a desired distance in the xy plane: If $\mathbf{v}_l$ is in the xy plane and ω times the traveling time T is a multiple of 2π, then the target will be located at $\mathbf{v}_l T$ from its original position.

REFERENCES

[1] J. Alexander
Loxodromes: A rhumb way to go,
Mathematics Magazine, vol. 77, no. 5, pp. 349–350, Dec. 2004.

[2] "AIAA guide to reference and standard atmosphere models,"
American Institute of Aeronautics and Astronautics, Reston, VA, Tech. Rep. G-003B-2004, 17 Nov. 2004.

[3] E. L. Andreas
"Handbook of physical constants and functions for use in atmospheric boundary layer studies,"
U.S. Army Corps of Engineers, Engineer Research and Development Center, Cold Regions Research and Engineering Laboratory, Hanover, NH, Tech. Rep. ERDC/CRREL M-05-1, Oct. 2005.

[4] D. P. Bertsekas
Nonlinear Programming,
2nd ed. Belmont, MA: Athena Scientific, 2003.

[5] R. L. Bishop
There is more than one way to frame a curve,
The American Mathematical monthly, vol. 82, no. 3, pp. 246–251, Mar. 1975.

[6] R. L. Burden and J. D. Faires
Numerical Analysis,
9th ed. Boston, MA: Brooks/Cole, 2011.

[7] K. Carlton-Wippern
"Surface navigation and geodesy a parametric approach,"
Air Force Space Command, Tech. Rep., 1 Mar. 1988. [Online]. Available: http://www.dtic.mil/docs/citations/ADA227907.

[8] D. Cressey
Arctic melt opens northwest passage,
Nature, vol. 449, 20 Sep. 2007.

[9] D. F. Crouse
Basic tracking using nonlinear continuous-time dynamic models,
IEEE Aerospace and Electronic Systems Magazine, accepted, December 2013.

[10] ———
Basic tracking using nonlinear 3D monostatic and bistatic measurements,
IEEE Aerospace and Electronic Systems Magazine, vol. 29, no. 8, Part II, pp. 4–53, Aug. 2014.

[11] ———
Simulating targets near a curved earth,
in *Proceedings of the 17th International Conference on Information Fusion*, Salamanca, Spain, 7–10 Jul. 2014.

[12] R. E. Deakin and M. N. Hunter
"Geodesics on an ellipsoid–Bessel's method,"
RMIT University, Melbourne, Australia, Tech. Rep., Oct. 2009. [Online]. Available: http://user.gs.rmit.edu.au/rod/files/publications/Geodesics%20-%20Bessel's%20method.pdf.

[13] Department of Defense
"Department of Defense world geodetic system 1984: Its definition and relationships with local geodetic systems,"
National Imagery and Mapping Agency, Tech. Rep. NIMA TR8350.2, Jun. 2004, third Edition, Amendment 2. [Online]. Available: http://earth-info.nga.mil/GandG/publications/tr8350.2/wgs84fin.pdf.

[14] "In the matter of the petition of Airbus SAS,"
Department of Transportation Federal Aviation Administration, 24 Mar. 2006, Regulatory Docket No. FAA-2005-20139. [Online]. Available: http://rgl.faa.gov/Regulatory_and_Guidance_Library/rgEX.nsf/0/ 9929ce16709cad0f8625713f00551e74/$FILE/8695.doc.

[15] M. P. Do Carmo
Differential Geometry of Curves and Surfaces.
Upper Saddle River: Prentice-Hall, Inc., 1976.

[16] C. Dougan
The parallel transport frame,
in *Game Programming Gems 2*, M. DeLoura, Ed. Charles River Media, 2001, ch. 2.5.

[17] Eclipse Aerospace, Inc. (2013, 12 Sep.)
The Eclipse 550 twin-engine jet.
[Online]. Available: http://www.eclipse.aero/550.php.

[18] M. E. Eshelby
"Pressure based separation in the upper airspace,"
College of Aeronautics, Cranfield Institute of technology, Cranfield, England, Tech. Rep. 8614, Feb. 1986. [Online]. Available: http://repository.tudelft.nl/assets/uuid:851bb984-e88c-4fc1-90c1-a98a11b040c1/Cranfield_College_of_Aeronautics_Report_No._8614-1986.pdf.

[19] G. T. Fairley and S. M. McGovern
Mathematical formulation of a fast-time geometric heading navigation model,
in *Proceedings of the 29th IEEE/AIAA Digital Avionics Systems Conference*, Salt Lake City, UT, 3–7 Oct. 2010, pp. 4.E.2-1–4.E.2-10.

[20] W. E. Featherstone and S. J. Claessens
Closed-form transformation between geodetic and ellipsoidal coordinates,
Studia Geophysica et Geodaetica, vol. 52, no. 1, pp. 1–18, Jan. 2008.

[21] ———
"Air traffic control,"
Federal Aviation Administration (United States), 9 Feb. 2012. [Online]. Available: http://www.faa.gov/documentLibrary/media/Order/ATC.pdf.

[22] M. L. Foucault
Sur les phénomènes d'orientation des corps tournants entraˆnés par un axe fixe à la surface de la terre.—nouveaux signes sensibles du mouvement diurne,
Comptes rendus hebdomadaires des séances de l'Académie des Sciences, pp. 424–427, 5 Jul. 1852. [Online]. Available: http://gallica.bnf.fr/ark:/12148/bpt6k2992n/f428.image.

[23] G. Golub and W. Kahan
Calculating the singular values and pseudo-inverse of a matrix,
Journal of the Society for Industrial and Applied Mathematics Series B Numerical Analysis, vol. 2, no. 2, pp. 205–224, 1965.

[24] G. E. Golub and C. F. van Loan
Matrix Computations,
4th ed. Baltimore, MD: The Johns Hopkins University Press, 2013.

[25] Ø. Grøn. and A. Næss
Einstein's Theory: A Rigorous Introduction for the Mathematically Untrained.
New York: Springer, 2011.

[26] K. T. Guthrie
"Linear parameter-varying path following control of a small fixed wing unmanned aerial vehicle,"
Ph.D. dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, 31 Jul. 2013.

[27] J. E. Hammett
"Evaluation of a proposed INS Kalman filter in a dynamic flight environment,"
Master's thesis, Air Force Institute of Technology, Wright-Patterson AFB, OH, Dec. 1974.

[28] A. J. Hanson and H. Ma
"Parallel transport approach to curve framing,"
Department of Computer Science, Indiana University, Bloomington, IN, Tech. Rep. TR425, 11 Jan. 1995. [Online]. Available: ftp://cgi.soic.indiana.edu/pub/techreports/TR425.pdf.

[29] B. Hofmann-Wellenhof and H. Moritz
Physical Geodesy,
2nd ed. SpringerWienNewYork, 2006.

[30] IHS, Inc. (2013)
Ihs janes'.
[Online]. Available: https://janes.ihs.com.

[31] G. H. Kaplan
Determining the position and motion of a vessel from celestial observations,
Navigation, vol. 42, no. 4, pp. 633–650, Winter 1995–1996. [Online]. Available: http://aa.usno.navy.mil/publications/reports/GK_posmo.pdf.

[32] ———
A navigation solution involving changes to course and speed,
Navigation, vol. 43, no. 4, pp. 469–482, 1996. [Online]. Available: http://oai.dtic.mil/oai/oai?verb=getRecord&metadataPrefix =html&identifier=ADA434100.

[33] F. Karmali and M. Shelhamer
The dynamics of parabolic flight: Flight characteristics and passenger percepts,
Acta Astronaut, vol. 63, no. 5–6, pp. 594–602, Sep. 2008.

[34] C. F. F. Karney. (2013, 31 Aug.)
Addenda and errata for papers on geodesics.
[Online]. Available: http://geographiclib.sourceforge.net/geod-addenda.html.

[35] ———
Algorithms for geodesics,
Journal of Geodesy, vol. 87, no. 1, pp. 43–45, Jan. 2013.

[36] ———. (2013, 2 Jul.)
Geodesics on an ellipsoid of revolution.
Matlab Central. [Online]. Available: http://www.mathworks.com/matlabcentral/fileexchange/39108.

[37] ———. (2013, 30 Jun.)
GeographicLib.
[Online]. Available: http://geographiclib.sourceforge.net.

[38] L. A. Kivioja
Computation of geodetic direct and indirect problems by computers accumulating increments from geodetic line elements,
Bulletin Géodésique, vol. 99, no. 1, pp. 55–63, Mar. 1971.

[39] P. F. Krause and K. L. Flood
"Weather and climate extremes,"
U.S. Army Corps of Engineers, Topographic Engineering Center, Alexandria, VA, Tech. Rep. TEC-0099, Sep. 1997. [Online]. Available: http://www.dtic.mil/docs/citations/ADA346058.

[40] X. R. Li and V. P. Jilkov
Survey of maneuvering target tracking. part I: Dynamic models,
IEEE Transactions on Aerospace and Electronic Systems, vol. 39, no. 4, pp. 1333–1364, Oct. 2003.

[41] ———
Survey of maneuvering target tracking. part II: Motion models of ballistic and space targets,
IEEE Transactions on Aerospace and Electronic Systems, vol. 46, no. 1, pp. 96–119, Jan. 2010.

[42] X. Li and H.-J. Götze
Ellipsoids, geoid, gravity, geodesy, and geophysics,
Geophysics, vol. 66, no. 6, pp. 1660–1668, Nov.–Dec. 2001, an errata wash published in Volume 67, Number 3.

[43] L. Lin, T. Kirubarajan, and Y. Bar-Shalom
3-D track initiation in clutter using 2-D radar measurements,
IEEE Transactions on Aerospace and Electronic Systems, vol. 38, no. 4, pp. 1434–1441, Oct. 2002.

[44] B. Luong. (2012, 18 Oct.)
Plot Earth. Matlab Central.
[Online]. Available: http://www.mathworks.com/matlab central/fileexchange/25048-plot-earth.

[45] W. A. Mair and D. L. Birdsall
Aircraft Performance,
ser. Cambridge Aerospace Series 5. Cambridge, UK: Cambridge University Press, 1992.

[46] P. S. Maybeck
"Wander azimuth implementation algorithm for a strapdown inertial system,"
Air Force Avionics Laboratory, Wright-Patterson AFB, OH, Tech. Rep. AFFDL-TR-73-80, Oct. 1973.

[47] R. A. Minzer
The 1976 standard atmosphere and its relationship to earlier standards,
Reviews of Geophysics and Space Physics, vol. 15, no. 3, pp. 375–384, Aug. 1977.

[48] P. J. Mohr, B. N. Taylor, and D. B. Newell
CODATA recommended values of the fundamental physical constants: 2010,
Reviews of Modern Physics, vol. 84, no. 4, pp. 1527–1605, Oct.–Dec. 2012. [Online]. Available: http://physics.nist.gov/cuu/Constants/Preprints/lsa2010.pdf.

[49] D. J. Mook and I.-M. Shyu
Nonlinear aircraft tracking filter utilizing control variable estimation,
Journal of Guidance, Control, and Dynamics, vol. 15, no. 1, pp. 228–237, Jan.–Feb. 1992.

[50] S. H. Musick
"PROFGEN: A computer program for generating flight profiles,"
Air Force Avionics Laboratory, Wright-Patterson AFB, OH, Tech. Rep. AFAL-TR-76-247, Nov. 1976. [Online]. Available: http://www.dtic.mil/docs/citations/ADA034993.

[51] National Geospatial Intelligence Agency. (2012, 1 Jul.)
WGS 84 G1647 geodetic control network upgrade for areas of white sands missile range and Holloman AFB, NM.
[Online]. Available: http://www.wsmr.army.mil/pdf/G1674-Upgrade-01Jul2012.pdf.

[52] ———. (2013, 29 Apr.)
Office of geomatics: World geodetic system 1984 (WGS 84).
[Online]. Available: http://earth-info.nga.mil/GandG/wgs84/.

[53] NGA Office of Geomatics. (2014, 16 May)
WGS 84 and the web mercator projection.
National Geospatial Intelligence Agency. A slide presentation on the linked page. [Online]. Available: http://earth-info.nga.mil/GandG/wgs84/web_mercator/index.html.

[54] N. Oceanic and A. Administration
"U.S. standard atmosphere 1976,"
Washington, D.C., Oct. 1976. [Online]. Available: http://www.dtic.mil/docs/citations/ADA035728.

[55] N. A. Office
Astronomical Almanac for the Year 2013 and its Companion, the Astronomical Almanac Online.
United States Defense Department, Navy, Nautical Almanac Office, 2013.

[56] E. J. Ohlmeyer
Root-mean-square miss distance of proportional navigation missile against sinusoidal target,
Journal of Guidance, Control, and Dynamics, vol. 19, no. 3, pp. 563–568, May–Jun. 1996.

[57] G. Panou, D. Delikaraoglou, and R. Korakitis
Solving the geodesics on the ellipsoid as a boundary value problem,
Journal of Geodetic Science, vol. 3, no. 1, pp. 40–47, Mar. 2013.

[58] G. Petit and B. Luzum
IERS Conventions (2010),
International Earth Rotation and Reference Systems Service Std. 36, 2010.

[59] D. H. Platus
Ballistic re-entry vehicle flight dynamics,
Journal of Guidance, Control, and Dynamics, vol. 5, no. 1, pp. 4–16, Jan.–Feb. 1982.

[60] P. Pokorny
Geodesics revisited,
Chaotic Modeling and Simulation International Journal of Nonlinear Science, pp. 281–298, Jan. 2012. [Online]. Available: http://www.cmsim.eu/papers_pdf/january_2012_papers/25_CMSIM_2012_Pokorny_1_281-298.pdf.

[61] R. H. Rapp
"Geometric geodesy, part I,"
Ohio State University Department of Geodetic Science and Surveying, Tech. Rep., Apr. 1991. [Online]. Available: http://hdl.handle.net/1811/24333.

[62] M. Sankowski
Reference model of aircraft movements in geodetic coordinates,
in *Proceedings of the International radar Symposium*, Leipzig, Germany, 7–9 Sep. 2011, pp. 874–879.

[63] ———
Target course estimation in tracking filters using local Cartesian coordinates,
in *Proceedings of the 19th International Radar Symposium*, Warsaw, Poland, 23–25 May 2012, pp. 515–520.

[64] L. E. Sjöberg and M. Shirazian
Solving the direct and inverse geodetic problems on the ellipsoid by numerical integration,
Journal of Surveying Engineering, vol. 138, no. 1, pp. 9–16, Feb. 2012.

[65] I. Sofair
Improved method for calculating exact geodetic latitude and altitude,
Journal of Guidance, Control, and Dynamics, vol. 20, no. 4, pp. 824–826, Jul.–Aug. 1997.

[66] ———
Improved method for calculating exact geodetic latitude and altitude revisited,
Journal of Guidance, Control, and Dynamics, vol. 23, no. 2, p. 369, Mar. 2000.

[67] J. Souchay and R. Gaume. (2013, 3 Sep.)
ICRS product center.
[Online]. Available: http://hpiers.obspm.fr/icrs-pc/.

[68] R. Stöckli, E. Vermote, N. Saleous, R. Simmon, and D. Herring. (2005, 22 Aug.)
The blue marble next generation—a true color earth dataset including seasonal dynamics from MODIS.
[Online]. Available: http://eoimages.gsfc.nasa.gov/images/imagerecords/73000/73751/readme.pdf.

[69] K. R. Symon
Mechanics,
3rd ed. Reading, MA: Addison-Wesley, 1971.

[70] D. Teagle and B. Ildefonse
Journey to the mantle of the earth,
Nature, vol. 471, pp. 437–439, 24 Mar. 2011.

[71] The Visible Earth. (2013, 11 Sep.)
The blue marble: Land surface, ocean color, sea ice and clouds.
NASA Goddard Space Flight Center. [Online]. Available: http://visibleearth.nasa.gov/view.php?id=57735.

[72] ———. (2013, 11 Sep.)
NASA visible Earth: July, blue marble next generation w/ topography and bathymetry.
NASA Goddard Space Flight Center. [Online]. Available: http://visibleearth.nasa.gov/view.php?id=73751.

[73] J. A. Thelander
"Aircraft motion analysis,"
Air Force Flight Dynamics Laboratory, Tech. Rep. FLD-TDR-64-70, Mar. 1965. [Online]. Available: http://www.dtic.mil/docs/citations/AD0617354.

[74] C. M. Thomas and W. E. Featherstone
Validation of Vincenty's formulas for the geodesic using a new fourth-order extension of Kivioja's formula,
Journal of Surveying Engineering, pp. 20–26, Feb. 2005.

[75] United States Air Force. (2007, 8 Oct.)
F-16 fighting falcon.
[Online]. Available: http://www.af.mil/AboutUs/FactSheets/Display/tabid/224/Article/104505/f-16-fighting-falcon.aspx.

[76] United States Geological Survey. (2006, 2 Feb.)
Mauna Loa Volcano, Hawaii.
[Online]. Available: http://hvo.wr.usgs.gov/maunaloa/.

[77] T. Vincenty
Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations,
Survey Review, vol. 23, no. 176, pp. 88–93, Apr. 1975. [Online]. Available: http://www.ngs.noaa.gov/PUBS_LIB/inverse.pdf.

[78] M. Vinnins
"PROFGEN: An aircraft flight profile generation program," Research and Development Branch, Department of national Defence, Ottawa, Canada, Tech. Rep. 78-14, Sep. 1978.

[79] R. Williams
The great ellipse on the surface of the spheroid,
Journal of Navigation, vol. 49, no. 2, pp. 229–234, May 1996.

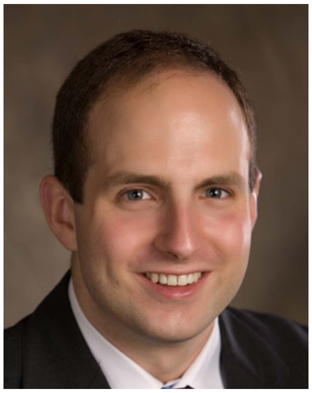
[80] Wolfram, Inc. (2013, 10 Sep.)
Numerical solution of partial differential equations—Mathematica 9 documentation.
[Online]. Available: http://reference.wolfram.com/mathematica/tutorial/NDSolvePDE.html.

[81] M. Yeddanapudi, Y. Bar-Shalom, and K. Pattipati
IMM estimation for multitarget-multisensor air traffic surveillance,
Proceedings of the IEEE, vol. 85, no. 1, pp. 80–94, Jan. 1997.

[82] P. Zarchan
Tactical and Strategic Missile Guidance,
5th ed. Reston, VA: American Institute of Aeronautics and Astronautics, Inc., 2007.

[83] P. Zarchan and H. Musoff
Fundamentals of Kalman Filtering: A Practical Approach,
3rd ed. Reston, VA: American Institute of Aeronautics and Astronautics, Inc., 2009.

[84] P. H. Zipfel
Modeling and Simulation of Aerospace Vehicle Dynamics,
2nd ed. Reston, VA: American Institute of Aeronautics and Astronautics, Inc., 2007.



David Frederic Crouse received B.S., M.S., and Ph.D. degrees in Electrical Engineering in 2005, 2008, and 2011 from the University of Connecticut (UCONN). He also received a B.A. degree in German from UCONN for which he spent a year at the Ruprecht-Karls Universität in Heidelberg, Germany.

He is currently employed at the Naval Research Laboratory in Washington, D.C. and serves as an associate editor at the IEEE Aerospace and Electronic Systems Magazine. His interests lie in the areas of stochastic signal processing and tracking.