View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Woods Hole Open Access Server

Rotation Identification in Geometric Algebra: Theory and Application to the Navigation of Underwater Robots in the Field

M. Jordan Stanway
Research & Development Division, Monterey Bay Aquarium Research Institute, Moss Landing, California 95039
e-mail: m.j.stanway@alum.mit.edu
James C. Kinsey
Department of Applied Ocean Physics and Engineering, Woods Hole Oceanographic Institution, Woods Hole, Massachusetts 02543
e-mail: jkinsey@whoi.edu

Received 6 December 2013; accepted 6 November 2014

We report the derivation and experimental evaluation of a stable adaptive identifier to estimate rigid body rotations using rotors in Geometric Algebra (GA). This work is motivated by the need for *in situ* estimation of the alignment between sensors commonly used in underwater vehicle navigation. Here we derive an adaptive identifier using a geometric interpretation of the error to drive first-order rotor kinematics. We prove that it is Lyapunov stable, and we show that it is asymptotically stable in the presence of persistent excitation. We use the identifier to estimate the alignment between the Doppler velocity log sonar and the fiber optic gyrocompass used by underwater vehicles for dead reckoning (DR). We evaluate this method in the laboratory with a remotely operated vehicle (ROV), and then with an autonomous underwater vehicle (AUV) operating in the field at 1,200 m depth. Our results show that this technique reduces dead reckoning navigation errors on these platforms and provides comparable performance to previously reported $SO(3)$ constrained Linear Algebra (LA) approaches. The rotor identifier has a number of advantages over these previously reported methods, including a more straightforward derivation, simpler gain tuning, increased computational efficiency, and reduced data manipulation. © 2015 The Authors. *Journal of Field Robotics* published by Wiley Periodicals, Inc.

1. INTRODUCTION

Rotations are ubiquitous in robotics and engineering. Examples include sensor alignment, manipulator kinematics, and attitude control. Here we focus on the sensor alignment problem, and specifically the task of identifying an unknown rigid body rotation from measured input-output data. This problem has been solved previously in various contexts using Linear Algebra (LA) or quaternion algebra, but the existing methods possess shortcomings and are difficult to visualize geometrically. This paper casts the sensor alignment problem into the language of Geometric Algebra (GA) and reports a novel adaptive identifier on the group of rotors. Solving the problem in GA addresses the shortcomings of existing alignment identification methods and leads to a simpler formulation and stability analysis.

Consider the linear mapping between an input $u(t) \in \mathbb{R}^3$ and an output $y(t) \in \mathbb{R}^3$. In LA, this transformation is written

$$y = \mathbf{M}u, \quad \mathbf{M} \in \mathbb{R}^{3\times 3}. \tag{1}$$

The vector signals $u(t)$ and $y(t)$ are known but $\mathbf{M}$ is unknown and must be identified. This problem is simple in two dimensions—the rotation is fully described by a single parameter (Figure 1a). But it is important to note that the problem is underconstrained in higher dimensions—two vectors in 3-space do not uniquely identify a rotation (Figure 1b) (that would require two *pairs* of nonparallel vectors).

Given sufficient measurements of $u(t)$ and $y(t)$, $\mathbf{M}$ can be estimated using the least squares (LS) techniques (Lawson & Hanson, 1974; Strang, 1986). If the mapping is known to be a rotation, it can be restricted to the group of rigid body rotations by enforcing orthogonality and normality constraints. In LA, a rotation is represented by a matrix $\mathbf{R} \in SO(3)$, which can be estimated using existing constrained least squares (CLS) (Wahba, 1965; Farrell et al., 1966; Arun, Huang, & Blostein, 1987; Markley, 1988; Umeyama, 1991) or adaptive identifier (Kinsey & Whitcomb, 2007a) techniques.

Here we use rotors in GA to formulate a novel adaptive identifier for unknown rotations. We prove that the rotor identifier is stable using Lyapunov methods, and we show that the system is asymptotically stable in the presence of persistent excitation. Then we validate the identifier in experiments with underwater robots in the laboratory and in the field.

 © 2015 The Authors. *Journal of Field Robotics* published by Wiley Periodicals, Inc.
View this article online at wileyonlinelibrary.com • DOI: 10.1002/rob.21572
This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

(a) Two dimensions (b) Three dimensions

Figure 1. The rotation between two vectors in two dimensions (a) is described by a single parameter, the angle φ. The rotation is inherently constrained to the two-dimensional plane. Identifying this rotation becomes more difficult in three dimensions—the plane of rotation is no longer constrained by the space (b). The input vector (red) could rotate along an infinite number of paths (purple) to reach the output vector (blue).

This work is motivated by the practical problem of *in situ* estimation of navigation sensor alignment for dead reckoning (DR) navigation for underwater vehicles originally reported in Brokloff (1994), Whitcomb, Yoerger, & Singh (1999), and Kinsey & Whitcomb (2004) and solved using constrained LS in Kinsey & Whitcomb (2007b), and an adaptive identifier on the group of rotation matrices in Kinsey & Whitcomb (2007a). The $SO(3)$ adaptive identifier successfully identified the unknown alignment, but it possessed several shortcomings:

1. The rotation matrix adaptive identifier has a 3×3 gain matrix that makes manual tuning tedious.
2. Chains of rotation matrices are computationally inefficient when compared to alternative rotation representations such as quaternions and rotors.
3. The geometry of the alignment problem is difficult to see when using rotation matrices—in Kinsey & Whitcomb (2007a), the parameter error is a rotation matrix, but the output error is a vector, and it is not clear how the two relate geometrically.
4. Depending on the input/output data, additional preprocessing may be necessary for the rotation matrix identifier to work in practice.

In terms of position residuals in DR navigation, the rotor identifier proposed in this paper performs comparably to the existing constrained least squares and rotation matrix adaptive identifier methods, but it offers several advantages over the rotation matrix adaptive identifier from Kinsey & Whitcomb (2007a). First, the rotor identifier has a single scalar gain as opposed to a 3×3 gain matrix, which solves shortcoming 1. The rotor identifier is more computationally efficient than the rotation matrix identifier, thus addressing shortcoming 2. While computational efficiency could also be improved by using quaternions, we prefer using rotors because GA enables a geometric understanding of the problem. In GA, we are able to define three related metrics for both the parameter and output error, and we describe the relationships between them, solving shortcoming 3. Finally, the rotor identifier requires no preprocessing, so it avoids shortcoming 4. These advantages, discussed in more detail in Section 7, make our rotor identifier superior to previously reported methods.

The remainder of this introduction reviews existing approaches to rotation identification. Section 2 presents the mathematical preliminaries required for the identifier formulation and stability proof. Section 3 reports the formulation of the rotor identifier, and its asymptotic stability is proven. In Section 4, we apply the identifier to estimate the alignment between two sensors in underwater navigation. We then validate the identifier and characterize its performance on data from a remotely operated vehicle (ROV) in controlled laboratory conditions (Section 5) and on field data from an autonomous underwater vehicle (AUV) (Section 6). Section 7 discusses the advantages of this technique over previously reported $SO(3)$ constrained alignment rotation identifiers.

1.1. Related Work

The alignment identification problem is a specific example of the more general problem of estimating rigid body rotations from a collection of uncertain data. Approaches to this problem can be loosely divided into batch methods and iterative methods.

Least squares (Lawson & Hanson, 1974; Strang, 1986) methods are perhaps the most common techniques for identifying an unknown linear map between known input-output data, but they do not guarantee that the resulting map is a rigid-body rotation, which must be orthonormal with a positive unit determinant. The first constraint can be satisfied by solving the orthogonal Procrustes' problem (Schönemann, 1966). The singular value decomposition (SVD) technique proposed by Arun et al. (Arun, Huang, & Blostein, 1987) constrains the solution to have a unit determinant, but it may be negative—this would produce a rotoreflection instead of a rotation. Many fully constrained techniques have been produced as solutions to Wahba's problem (Wahba, 1965), including the TRIAD and QUEST algorithms (Shuster & Oh, 1981), as well as SVD-based methods in Farrell et al. (1966) and Markley (1988). We use Umeyama's SVD implementation (Umeyama, 1991), which refines the technique by Arun, as the LS-$SO(3)$ method for comparison.

General iterative methods in LA, such as adaptive identification (Narendra & Annaswamy, 1988; Sastry & Bodson, 1989), also do not guarantee that the solution will be a rotation.

The large body of existing work on attitude control [see, e.g., Meyer (1971); Wen & Kreutz-Delgado (1991); Bullo, Murray, & Sarti (1995); and Bullo & Murray (1995)] and tracking [see, e.g., Koditschek (1988); Caccavale & Villani (1999); and Akella (2001)] provides many useful

contributions toward the rotation identification problem, particularly with respect to kinematics, dynamics, and stability on $SO(3)$. Some cast their solutions in quaternion algebra [see, e.g., Shuster & Oh (1981); Bar-Itzhack & Oshman (1985); and Choukroun, Bar-Itzhack, & Oshman (2006)], while others use the tools of Lie group theory [see, e.g., Bullo, Murray, & Sarti (1995) and Bullo & Murray (1995)]. A comprehensive survey of several existing nonlinear attitude estimation methods is given in Crassidis, Markley, & Cheng (2007). However, most of these existing attitude control, tracking, and estimation methods [see, e.g., Shuster & Oh (1981); Koditschek (1988); Caccavale & Villani (1999); Akella (2001); Wen & Kreutz-Delgado (1991); Bullo, Murray, & Sarti (1995); and Bullo & Murray (1995)] assume a unique estimate of the orientation at each time step, and therefore they do not directly apply to the specific alignment identification problem addressed here. Mahony et al. developed complementary filters directly on $SO(3)$ to estimate attitude using angular rate, acceleration, and magnetic measurements (Mahony, Hamel, & Pflimlin, 2005). They adapted that approach in Hamel & Mahony (2006) and Mahony, Hamel, & Pflimlin (2008) to handle direction inputs from a single sensor, but they still require angular rates. Sanyal et al. developed another attitude estimator using single direction measurements and treating uncertainty with ellipsoids in Lee et al. (2007) and Sanyal et al. (2008), but they also assumed knowledge of attitude dynamics (i.e., angular rates). Some properties of the work by Mahony and Sanyal are similar to the specific alignment identification problem addressed here, but neither work applies directly. Kinsey & Whitcomb solved the same alignment identification problem with an adaptive identifier on $SO(3)$, and they proved its stability using Lyapunov theory (Kinsey & Whitcomb, 2007a). This adaptive identifier performs favorably when compared to the batch LS methods in Kinsey & Whitcomb (2007b).

Batch methods for rotor identification in GA have also been developed recently. Doran studied the estimation of an unknown rotor from noisy data in the context of camera localization in Doran (2000). Buchholz and Sommer investigated the related problem of averaging in Clifford groups (Buchholz & Sommer, 2005).

Here we report an iterative online GA method for identifying a constant and unknown alignment between two point sets. The formulation of the asymptotically stable adaptive identifier was originally reported in Stanway & Kinsey (2011), along with preliminary simulation results. The application to underwater navigation, including laboratory and field experiments in Section 5 and Section 6 and the discussion in Section 7, are presented here for the first time. The alignment identification problem discussed in this paper is subtly different from the related problems of attitude estimation and control. Most methods for attitude estimation and control [see, e.g., Bullo, Murray, & Sarti (1995); Bullo & Murray (1995); Akella (2001); and Caccavale & Villani (1999)] assume sufficient information at each time step for a complete, unique observation of the orientation. In the problem addressed in this paper, that assumption does not apply. Two vectors in three dimensions do not uniquely describe a rotation, as illustrated in Figure 1b.

2. PRELIMINARIES

This section defines notation and reviews preliminaries in GA necessary for the derivation of the adaptive rotor identifier in Section 3. The presentation is geared primarily toward $\mathbb{G}_3$, the GA of the three-dimensional Euclidean model, but basic principles apply to algebras of other dimensions as well. Refer to Hestenes (2002); Doran & Lasenby (2003); Dorst, Fontijne, & Mann (2007); and Gull, Lasenby, & Doran (1993) for a more in-depth introduction to GA. Refer to Bayro-Corrochano and Sobczyk (2001), Sommer (2001), Bayro-Corrochano (2005), Bayro-Corrochano (2010), and Bayro-Corrochano & Scheuermann (2010) for examples of GA applied to problems in robotics, computer vision, and engineering.

2.1. Nomenclature

This paper uses both Linear Algebra (LA) and Geometric Algebra (GA), so some discussion on notation is necessary to avoid confusion. Scalar quantities are denoted by lowercase Greek letters (e.g., α), while vectors are in lowercase italic (e.g., a). Both scalars and vectors are directly interchangeable between LA and GA. Matrices in LA are uppercase bold Roman (e.g., $\mathbf{R}$). GA multivectors are uppercase italic (e.g., R). Juxtaposition of two factors represents the natural product in that algebra: $y = \mathbf{R}u$ is a matrix product, while $M = ab$ is a geometric product.

Sections 4–6 discuss rotations between different reference frames. In those sections, we use the same notation as Kinsey & Whitcomb (2007a,b). A leading superscript indicates the frame of reference for a vector quantity: ${}^{w}\dot{p}_d(t)$, ${}^{v}\dot{p}_d(t)$, and ${}^{i}\dot{p}_d(t)$ represent the Doppler velocity log (DVL) velocity in the world, vehicle, and instrument frames of reference, respectively. A trailing subscript denotes the sensor source: ${}^{w}\dot{p}_d(t)$ represents the DVL velocity and ${}^{w}p_l(t)$ represents the measurement from long baseline (LBL) acoustic positioning, both in world coordinates. Two leading subscripts on a rotation matrix indicate the frames of reference involved in a rotation. For example, ${}^{w}_{v}\mathbf{R}(t)$ is the rotation matrix from the vehicle frame to the world frame, i.e., ${}^{w}\dot{p}_d(t) = {}^{w}_{v}\mathbf{R}(t)\ {}^{v}\dot{p}_d(t)$.

2.2. Multiplying Vectors

Every vector has two intrinsic properties: magnitude and direction. GA defines inner, outer, and geometric products to describe relationships between these properties. These products expand the language of GA to include elements of

lower and higher grade (i.e., dimension), and even elements of mixed grade.

2.2.1. Inner Product

The inner product of two vectors a, b in GA is essentially the same as in LA—it is commutative, and it defines a scalar α (grade 0):

$$a \cdot b = b \cdot a = \alpha = |a||b| \cos\varphi, \tag{2}$$

where φ is the angle between a and b.

2.2.2. Outer Product

The outer product of two vectors is anticommutative and defines a new entity called a bivector B (grade 2):

$$a \wedge b = -b \wedge a = B = |a||b| \sin\varphi \hat{B}. \tag{3}$$

This bivector should be thought of as a directed plane segment in much the same way as a vector is a directed line segment.

2.2.3. Geometric Product

The inner and outer products complement each other: one lowers the grade while the other raises it, one is commutative while the other is anticommutative. They are combined in the geometric product:

$$ab = a \cdot b + a \wedge b = b \cdot a - b \wedge a = \alpha + B = M, \tag{4}$$

which produces a multivector M that contains both scalar and bivector parts. It may seem odd at first to add together elements of different grades, but it is no different than adding together the real and imaginary parts of a complex number (Gull, Lasenby, & Doran, 1993).

The geometric product is actually the most basic product in GA. The inner and outer products can be derived axiomatically as the symmetric and antisymmetric components of the geometric product (Hestenes, 2002; Doran & Lasenby, 2003),

$$a \cdot b = \frac{1}{2}(ab + ba), \tag{5a}$$

$$a \wedge b = \frac{1}{2}(ab - ba). \tag{5b}$$

The geometric product itself is generally neither commutative nor anticommutative. Reversing the order of factors is still an important operation, known as reversion:

$$\text{reverse}(ab) = ba = (ab)^{\sim} = \tilde{M}. \tag{6}$$

2.3. Useful Bivector Properties

2.3.1. Reversion

Since a pure bivector has no scalar part, the reverse of a bivector is its negative:

$$B = a \wedge b = \frac{1}{2}(ab - ba) \tag{7a}$$

$$\Rightarrow \tilde{B} = \frac{1}{2}(ba - ab) = b \wedge a = -B. \tag{7b}$$

2.3.2. Magnitude

Bivectors in $\mathbb{G}_3$ are like imaginary numbers; they square to a negative scalar:

$$B^2 = BB = -\beta^2. \tag{8}$$

Since the reverse of a bivector is its negative, and any bivector is parallel to itself, the magnitude β can be written several ways:

$$\beta = \sqrt{B\tilde{B}} = \sqrt{-BB} = \sqrt{-B \cdot B}. \tag{9}$$

Using the scalar α to define another bivector A parallel to B,

$$A \parallel B \Rightarrow A = \alpha B, \tag{10}$$

shows that the geometric product of any two parallel bivectors with the same sign is negative-definite:

$$AB = (\alpha B)(B) = -\alpha\beta^2 < 0 \quad \forall \quad \alpha > 0. \tag{11}$$

2.3.3. Commutativity

Bivectors neither commute nor anticommute in the geometric product. However, any pair of $\mathbb{G}_3$ elements in the same grade can be decomposed into parallel and perpendicular parts. For two bivectors, A, B:

$$A = A_{\parallel B} + A_{\perp B};\ B = B_{\parallel A} + B_{\perp A}. \tag{12}$$

Parallel (i.e., dependent) bivectors commute:

$$A_{\parallel B} B = B A_{\parallel B}, \tag{13a}$$

and orthogonal (i.e., independent) bivectors anticommute:

$$A_{\perp B} B = -B A_{\perp B}. \tag{13b}$$

The symmetric part of the geometric product of two bivectors in $\mathbb{G}_3$ therefore reduces to their inner product:

$$\frac{1}{2}(AB + BA) = \frac{1}{2}(A_{\parallel B}B + A_{\perp B}B + BA_{\parallel B} + BA_{\perp B}) \tag{14a}$$

$$= A_{\parallel B} B = A \cdot B. \tag{14b}$$

If $A_{\|B}$ has the same sign as B, this inner product is negative-definite:

$$A \cdot B = A_{\|B} B = (\alpha B)(B) = -\alpha\beta^2 < 0 \quad \forall \quad \alpha > 0, \quad (15)$$

if the sign is opposite, the inner product is positive-definite. In the case in which either bivector is actually zero, the inner product becomes zero.

2.4. Representing Rotations

Rotations in GA are encoded by special elements called rotors. These are normalized elements of the even subalgebra $\mathbb{G}^+$. In the three-dimensional Euclidean model $\mathbb{G}_3$, rotors form the group

$$Spin(3) = \left\{ R : R \in \mathbb{G}_3^+, R\tilde{R} = \tilde{R}R = 1 \right\}, \quad (16)$$

which provides a double-cover over $SO(3)$.

Three-dimensional rotors are isomorphic to the quaternions $\mathbb{H}$ [see, e.g., Murray, Li, & Sastry (1994) and Kuipers (1999)], but rotors emphasize geometric interpretation and extend to higher dimensions more easily (Doran & Lasenby, 2003; Dorst, Fontijne, & Mann, 2007). Using the richer language of GA also explicitly includes scalars, vectors, and bivectors from the start. The bivectors, in particular, are important in our application—they play a principal role in the identifier formulation and in its stability proof. Bivectors are not present in quaternion algebra.

Bivectors are linked to rotors via the exponential map (i.e., the bivector exponential and rotor logarithm). This exponential map highlights an important correspondence between GA and Lie group theory—every Lie algebra can be represented as a bivector algebra, hence every Lie group can be represented as a spin group (Doran, Hestenes, Sommen, & van Acker, 1993).

Three-dimensional rotors have four coefficients: one scalar and three bivectors. The orthogonality constraint is implicit in the algebraic structure. The unit norm constraint is simple to enforce, but often unnecessary because of good numerical stability properties shared with quaternions (Brown, 1989; Breckenridge, 1999; Farrell, 2008). In contrast, rotation matrices have nine coefficients, and enforcing orthonormality is more complicated.

Rotors can be composed by chaining together multiple rotors, using repeated reflections, or as the exponential of a bivector (Figure 2).

2.4.1. Rotor Chains

The rotor group is closed under multiplication in the geometric product, so the geometric product of two or more rotors will always produce another rotor:

$$R_a R_b = R_c \in Spin(3) \quad \forall \quad R_a, R_b \in Spin(3). \quad (17)$$

2.4.2. Repeated Reflections

The Cartan-Dieudonné theorem states that any Euclidean transformation can be expressed by a combination of reflections (Cartan, 1966). An even number of reflections preserves the chirality (i.e., handedness) of a coordinate frame, and produces a proper rotation.

Forming a sandwich product with a unit vector $\hat{n}$ on each side produces a reflection:

$$z = \hat{n}u\hat{n}, \quad (18)$$

so the geometric product of an even number of unit vectors is a rotor:

$$y = \hat{m}\hat{n}u\hat{n}\hat{m} = Ru\tilde{R}, \quad R = \hat{m}\hat{n}. \quad (19)$$

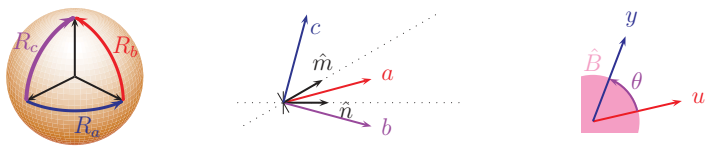
Since this composition method produces a rotation in the plane spanned by two vectors, it can be used to calculate the shortest rotor between them [see, e.g., Doran & Lasenby (2003) and Dorst, Fontijne, & Mann (2007)]. This is used in Section 3.4 to define the output error rotor.

2.4.3. Bivector Exponential and Rotor Logarithm

A bivector B can encode the plane $\hat{B}$ and angle θ (i.e., the direction and magnitude) of a rotation. The rotor that generates this rotation is the exponential of the bivector:

$$R = \exp\left(-\frac{1}{2}\theta\hat{B}\right) = \mathrm{e}^{-\frac{1}{2}B}, \quad (20)$$

where the factor of $1/2$ is because we multiply the rotor twice (once on each side) and the negative sign is a convention so that positive angles produce a counterclockwise (i.e., right-handed) rotation. The concept of an exponential function is



(a) rotor chain (b) repeated reflections (c) bivector exponential

Figure 2. Three main methods of composing rotors and generating rotations: (a) from two or more other rotors as a chain, (b) from two vectors using the concept of repeated reflections, or (c) from a bivector and generating rotations using the exponential map.

easily generalized to operate on multivectors; see Hestenes (2002) or Doran & Lasenby (2003).

The bivector B is invariant in the rotation generated by R since that rotation is in the same plane as B:

$$RB\tilde{R} = B \quad \forall \quad R = \mathrm{e}^{-\frac{1}{2}B}. \tag{21}$$

It is useful to define a multivector logarithm so that the bivector B can be determined from the rotor R. We adopt the definition given in Hestenes (2002) and denote the rotor logarithm:

$$B = -2\ln R. \tag{22}$$

If $R = 1$, $B = 0$, like the familiar natural logarithm for scalars. Otherwise, B is a bivector describing the direction and magnitude of the rotation produced by R.

2.4.4. First-order Rotor Kinematics

Since B describes the rotation angle, dB/dt describes the rotational velocity. We adopt the notation $\dot{B} = \Omega$, where Ω is the natural bivector form of the rotational velocity.

Using the chain rule to differentiate the bivector exponential (20) gives the first-order rotor differential equation:

$$\frac{d}{dt}(R(t)) = \frac{dB}{dt}\frac{dR}{dB}, \tag{23a}$$

$$\dot{R} = -\frac{1}{2}\Omega R. \tag{23b}$$

3. AN ADAPTIVE IDENTIFIER ON THE GROUP OF ROTORS

Here we propose an adaptive identifier based on first-order rotor kinematics and a geometric interpretation of identification error. We begin by defining the rotation plant in terms of input and output vectors. Then we discuss several relevant error metrics, commenting on the geometric meaning of each of them. We use first-order rotor kinematics and a proportional gain error state feedback regulator to formulate a rotor identifier, and then we prove its asymptotic stability using Lyapunov theory.

3.1. Plant

Given known vector input u and output y signals, the plant

$$y = Ru\tilde{R}, \tag{24}$$

describes the rotation from input to output. The unknown rotor $R \in Spin(3)$ is a stationary element of the even subalgebra, and it encodes a rigid body rotation. The goal of the method proposed here is to identify this unknown rotor, R.

3.2. Identification Plant

With the time-varying estimate of the rotor defined as $S(t) \in Spin(3)$, the expected plant output v is

$$v = Su\tilde{S}. \tag{25}$$

The goal of the identifier is to drive the estimated rotor S toward the actual rotor R. Since each time step provides an incomplete observation of the actual rotor, we achieve this using feedback on the output error, driving the expected output v toward the observed output y.

3.3. Parameter Error

The parameter error rotor is

$$Q = R\tilde{S} : \; Q \in Spin(3). \tag{26}$$

This describes the difference between the estimated rotor and the actual rotor. Applying the parameter error rotor directly to the estimated output will rotate it into the actual output:

$$Qv\tilde{Q} = (R\tilde{S})(Su\tilde{S})(S\tilde{R}) = Ru\tilde{R} = y.$$

In this way, the parameter error rotor resolves the difference between the identification plant and the actual plant.

The evolution of the estimate S is characterized by the time derivative of the parameter error rotor:

$$\dot{Q} = \dot{R}\tilde{S} + R\dot{\tilde{S}} = R\dot{\tilde{S}}. \tag{27}$$

The parameter error bivector is derived from the natural logarithm of the parameter error rotor:

$$X = -2\ln(Q) \; : \; Q = \mathrm{e}^{-\frac{1}{2}X}. \tag{28}$$

This metric encodes both the magnitude and the direction of the error, which can be interpreted as the angle and the rotation plane.

The magnitude of the parameter error is a scalar:

$$\chi = \sqrt{X\tilde{X}}. \tag{29}$$

This is the natural distance measure on the three-dimensional unit sphere $\mathbb{S}^3$, and concisely summarizes the error state. It is distilled down to the angle between the estimate and the truth, and it is analogous to the distance measure defined in Bullo & Murray (1995). As a scalar, χ is convenient for visualization and for stability analysis, but the lack of directional information makes it less useful for feedback.

3.4. Output Error

Since a pair of input/output vectors does not uniquely define a rotation in three dimensions (Figure 1b), the parameter error is only partially observed at each time step. We define output errors to characterize the observable part of the misalignment.

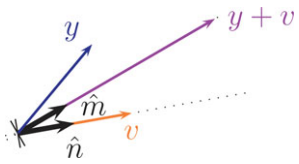


Figure 3. We compose the output error rotor P using repeated reflections (Section 2.4.2). If the expected output v is parallel to the observed output y, then $\hat{m} = \hat{n}$, and $P = 1$. Otherwise, P rotates v into y along the shortest path on the sphere.

We use repeated reflections (19) to compose the output error rotor:

$$P = \hat{m}\hat{n} = \frac{y+v}{|y+v|}\frac{v}{|v|} \; : \; P \in Spin(3), \tag{30}$$

so that it rotates the estimated output v directly into the measured output y along the shortest path on the sphere (Figure 3).

The output error bivector is analogous to its parameter error counterpart:

$$Y = -2\ln(P). \tag{31}$$

This bivector form of the output error encodes information on both magnitude and direction—we will use it in the update law (34) as a rotational velocity to drive the identification plant kinematics.

Similar to the parameter error angle (29), the output error angle is

$$\psi = \sqrt{Y\tilde{Y}}. \tag{32}$$

Again, this scalar provides a useful summary of the output error, but the lack of directional information makes it unsuitable for feedback.

Note that the error rotors tend toward unity as $S \to R$, while the bivectors and angles tend toward zero. When $y \parallel v$, $\hat{m} = \hat{n}$, so the output error rotor $P = 1$, the output error bivector $Y = 0$, and the output error scalar $\psi = 0$.

Incidentally, this formulation allows the rotor identifier to avoid the unwinding phenomenon sometimes encountered in quaternion controllers and estimators. Since the output error rotor P is explicitly defined in (30) to trace the shortest path on the sphere from the expected output v to the observed output y, it will always describe a rotation with the output error angle $\psi < \pi$. So if $S_0 = 1$, then S will stay closer to 1 than to -1 for all time.

3.5. Update Law

We formulate the rotor identifier as an output error feedback regulator with proportional gain, using the output error bivector to drive the rotational velocity of the estimated rotor:

$$\Omega(t) = \kappa_p(t) Y(t), \tag{33}$$

where $\kappa_p(t)$ is a positive-definite scalar, i.e., $\kappa_p(t) > 0$. For the remainder of the paper, we will use constant κ_p, but the stability proof still holds for a time-varying gain, as long as κ_p is always positive.

Feeding the rotational velocity (33) into the first-order rotor kinematics (23b) gives the continuous-time update law:

$$\dot{S} = -\frac{\kappa_p}{2} Y S \text{ and } \dot{\tilde{S}} = \frac{\kappa_p}{2}\tilde{S}Y. \tag{34}$$

Error due to finite precision numerical integration is of little concern—the algebraic structure of the rotor enforces orthogonality, and the unit norm constraint can be enforced by periodic renormalization if necessary. This paper also provides an exact discrete-time update in Section 3.8.

3.6. System

Substituting the update law (34) into the parameter error (27) gives the continuous-time system:

$$\dot{Q} = R\left(\frac{\kappa_p}{2}\tilde{S}Y\right) = \frac{\kappa_p}{2} QY. \tag{35}$$

This looks very similar to the first-order rotor differential equation (23b). In fact, (35) simply states that the rotational velocity of the parameter error is proportional and opposite to the rotational velocity of the estimate:

$$\dot{Q} = -\frac{1}{2}\dot{X}Q, \tag{36}$$

$$-\dot{X} = 2\dot{Q}\tilde{Q} = \kappa_p QY\tilde{Q} = \Omega', \tag{37}$$

where Ω' is the rotational velocity expressed in the reference frame of R. Note that the time derivative of a bivector ($\dot{X}$) is still a bivector.

3.7. Stability

We apply Lyapunov methods (Murray, Li, & Sastry, 1994; Khalil, 1996) to prove the asymptotic stability of the continuous time system (35). The proof is broken into three parts: a general Lyapunov candidate function for rotor identifiers; a geometric discussion of the criteria for stability; and the algebraic proof of asymptotic stability for the specific update law and system proposed here.

3.7.1 Lyapunov Candidate Function

Consider a Lyapunov candidate function based on the parameter error angle χ:

$$V = \frac{1}{2}\chi^2. \tag{38a}$$

This function is analogous to the Lyapunov candidate functions employed in Bullo & Murray (1995) and Kinsey & Whitcomb (2007a). By expanding the parameter error angle χ, the Lyapunov candidate function can also be written directly in terms of the parameter error bivector:

$$V = \frac{1}{2}\|X\|^2 = \frac{1}{2}X\tilde{X}. \tag{38b}$$

The time derivative of the Lyapunov candidate function (38b) is

$$\dot{V} = \frac{d}{dt}\left(\frac{1}{2}X\tilde{X}\right) \tag{39a}$$

$$= \frac{1}{2}\left(\dot{X}\tilde{X} + X\dot{\tilde{X}}\right). \tag{39b}$$

Recalling that the reverse of a bivector is its negative (7b):

$$\dot{V} = -\frac{1}{2}\left(\dot{X}X + X\dot{X}\right) = -\dot{X}\cdot X. \tag{39c}$$

This form of the Lyapunov derivative applies to all rotor identifiers using the candidate function (38)—the specific identifier update law and resulting system have not yet entered into the equation. To produce a stable system, the update law must make (39c) negative-semidefinite.

3.7.2 Geometric Understanding

The Lyapunov derivative (39c) shows that any component of $\dot{X}$ orthogonal to X is inconsequential in the stability analysis. Furthermore, for the Lyapunov derivative (39c) to be negative-semidefinite, the component of rotational velocity $\dot{X}$ that is not orthogonal to the parameter error bivector X must be antiparallel to it, i.e., the part of $\dot{X}$ parallel to X must have opposite sign; see (15). This is equivalent to an extension of Gauss' Lemma reported in Bullo, Murray, & Sarti (1995). With this geometric understanding of the $\dot{V} \leq 0$ stability criterion, consider the update law (33) and resulting system (37).

3.7.3 Stability of the proposed system

Substituting the system (37) into the Lyapunov derivative (39c) gives

$$\dot{V} = \frac{1}{2}\kappa_p\left(QY\tilde{Q}X + XQY\tilde{Q}\right). \tag{40a}$$

Recall (21), which for X and Q is $X = QX\tilde{Q}$. Thus,

$$\dot{V} = \frac{1}{2}\kappa_p\left(QY\tilde{Q}QX\tilde{Q} + QX\tilde{Q}QY\tilde{Q}\right) \tag{40b}$$

$$= \frac{1}{2}\kappa_p\left(QYX\tilde{Q} + QXY\tilde{Q}\right) \tag{40c}$$

$$= \frac{1}{2}\kappa_p Q\left(YX + XY\right)\tilde{Q}. \tag{40d}$$

Since both sides of the equation are scalars, the rotation by Q does nothing, leaving

$$\dot{V} = \frac{1}{2}\kappa_p\left(YX + XY\right). \tag{40e}$$

Recalling (14b), this reduces to the inner product:

$$\dot{V} = \kappa_p Y\cdot X \leq 0. \tag{40f}$$

Since Y has been defined so that the part parallel to X has the same sign as X, the inner product is a negative-definite scalar (15). If the output error had been defined in the opposite sense, the stability criterion (40f) would simply require that κ_p be negative instead of positive.

Three conditions can cause $Y\cdot X = 0$ and the inner product $\dot{V}$ to be negative-semidefinite:

1. When the parameter error X is zero and the rotor is identified;
2. when the output error Y is zero—this corresponds to an unstable equilibrium where the expected output is parallel and directly opposed to the observed output (a topological limitation sometimes referred to as almost global stability); and
3. when the remaining parameter error is orthogonal to the observed output error.

In these cases, the system is still stable, but not asymptotically stable. Condition (1) is achieved when the estimated rotor equals the actual rotor and the identification is complete. Condition (2) can be avoided by proper selection of the initial condition, i.e., the initial estimated rotor generates an expected output that is not parallel and directly opposed to the observed output. In practice, measurement noise and persistent excitation quickly move the system out of this unstable equilibrium. Persistent excitation is also necessary to avoid condition (3) and ensure that the parameter error is completely observed (see the more detailed discussion below). Thus both (a) persistent excitation and (b) proper initial conditions are theoretically necessary for (40f) to become strictly negative-definite, although in practice persistent excitation is sufficient.

If this identifier were operating in two dimensions, it would have a complete observation of the parameter error at every time step—Y and X would always be on the same plane, and could never be orthogonal. But in three dimensions, X can be orthogonal to Y, which makes $Y\cdot X = 0$, and part of the parameter error remains unobserved. Recall (Figure 1b), which showed that two vectors do not completely define a rotation in three dimensions—that requires two pairs of vectors. The identifier can estimate the components of the alignment in the planes it has observed, but it needs observations in different directions to be fully excited. This means that the plant input u has to be varied in all directions in order to accurately estimate all components of the alignment—if that happens, (40f) becomes negative-definite.

Consequently, the proposed Lyapunov candidate function (38) satisfies the following conditions:

1. V is a continuous, scalar function.
2. V is positive-definite when $\chi \neq 0$.
3. When $\chi = 0$, $V = 0$.
4. $\dot{V} \leq 0$.
5. In the presence of persistent excitation, $\dot{V} < 0$.

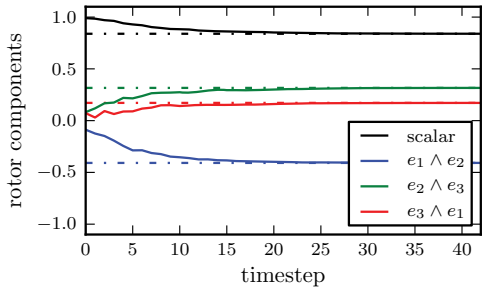


Figure 4. The components of the estimated rotor S computed by the proportional gain rotor identifier converge to the components of the unknown rotor R (marked by dash-dot lines).

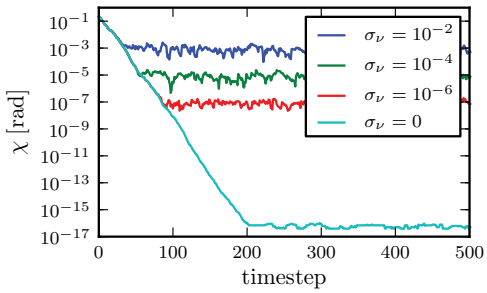


Figure 5. Evolution of the scalar parameter error when the input/output vector pairs are corrupted by additive white noise. The estimated rotor converges to the actual rotor until χ approaches the noise floor. The estimate then oscillates around the truth with magnitude controlled by the gain and the noise level.

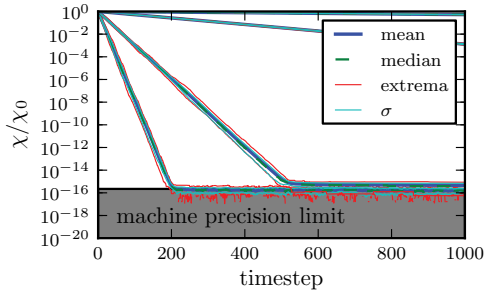


Figure 6. Evolution of the scalar parameter error, normalized by its initial value. Results are ensemble-averaged over 20 randomly generated datasets and compared for four proportional gains: $\kappa_p \in \{0.001, 0.01, 0.1, 0.25\}$.

When all of the above conditions are satisfied, the system is asymptotically stable and the identified alignment converges to the actual alignment as time approaches infinity.

3.8. Discrete-time Implementation

Most applications will provide measurements of the input u and the output y in discrete time, so it makes sense to develop a discrete-time version of the proportional update law (34).

Define the discrete update of the rotor estimate:

$$\delta S = S_k \tilde{S}_{k-1} : S_k = (\delta S)(S_{k-1}). \tag{41a}$$

Assuming the gain and rotational velocity of the rotor estimate are constant over the time step, we use Euler's method to integrate the update law (34) directly:

$$\delta S = \int_0^{\delta t} \left(-\frac{\kappa_p}{2} Y S\right) d\tau = \mathrm{e}^{-\frac{1}{2}(\kappa_p \delta t) Y}, \tag{41b}$$

$$S_k = \mathrm{e}^{-\frac{1}{2}(\kappa_p \delta t) Y} S_{k-1}. \tag{41c}$$

This discrete-time form of the rotor identifier was implemented in C using a base library generated by `Gaigen2.5` (Fontijne, 2010), the latest version of the GA implementation generator (Fontijne, Bouma, & Dorst, 2001; Fontijne, 2006).

Figure 4 shows the four components of the estimated rotor S converging toward the true rotor R in a simulation using the discrete-time rotor identifier. In other simulations, several levels of random Gaussian noise were added to the output y. Figure 5 illustrates how the scalar parameter error χ decreases in time until it reaches the level of the noise.

3.8.1. Gain Selection

The proportional gain on this identifier operates as would be expected on any system driven with proportional feedback: higher κ_p speeds up convergence, but it leaves the system more susceptible to measurement noise; lower κ_p rejects noise better, but it produces longer transients and slower convergence. Figure 6 illustrates this convergence speed tradeoff—it shows ensemble averages of the scalar parameter error χ for four different values of κ_p. The ensembles are averaged over 20 randomly generated datasets, and χ is normalized by its initial value to show that the identifier behaves similarly across the random datasets in the ensemble, and it converges at different rates for different κ_p. Optimal or adaptive gain selection is outside the scope of the current paper, but it could be an interesting topic for future research.

4. APPLICATION TO UNDERWATER ROBOT NAVIGATION

Our motivation for investigating this class of adaptive identifiers arises from challenges in underwater navigation. Navigation remains difficult for robots operating underwater, where a Global Positioning System (GPS) is unavailable, and the limited precision and update rate of acoustic positioning systems is often insufficient for closed loop feedback control of underwater robotic vehicles. Historically, underwater controllers have been developed and implemented

for the heading, altitude, depth, and attitude degrees of freedom (Yoerger and Slotine, 1991; Fossen, 1994; Choi & Yuh, 1996; Whitcomb & Yoerger, 1996). Only in the past decade has closed-loop control in the horizontal degrees of freedom become common in ocean robots (Whitcomb et al., 2003), and this is the direct result of the development of new navigation systems with sufficient resolution and update rate (Whitcomb, Yoerger, & Singh, 1999; Kinsey & Whitcomb, 2004). Navigation is also crucial in obtaining important measurements for oceanographic research. For example, AUVs use sonar and cameras to make high-resolution seafloor maps that are not obtainable from surface ships (Singh et al., 2007; Caress et al., 2008; Pizarro, Eustice, & Singh, 2009). Improved navigation enhances both the control of underwater vehicles and the scientific value of data obtained by these vehicles.

Underwater vehicles employ a variety of sensors to estimate the XY position. GPS provides three-dimensional navigation for both surface and air vehicles, however the rapid attenuation of GPS radio-frequency signals prevents them from being used by submerged vehicles. The accuracy, update rate, and range of acoustic positioning systems depend on many factors, including the nominal frequency of the acoustic signals used. Long-baseline (LBL) acoustic systems estimate vehicle position based on ranges computed from two-way travel times to and from fixed transponders deployed on the seafloor. LBL systems used in the field typically have a center frequency of 12 kHz, with a precision of ± 0.25–10 m and update periods as long as 20 s (Hunt et al., 1974; Milne, 1983). Ultra-short baseline (USBL) acoustic tracking is becoming more widely used in navigation for scientific, industrial, and military underwater vehicles. The geographic position of the submerged vehicle is calculated by adding the acoustic range, azimuth, and elevation measurements from a single ship-mounted transducer to the GPS position of the ship. However, as with LBL systems, the limited precision and update rate of USBL measurements precludes their use for underwater robot control. Inertial navigation systems (INSs) offer excellent strap-down navigation capabilities, but they require help from additional sensors to limit integration drift in the position estimates (Britting, 1971; Titterton & Weston, 1997). To date, power consumption and cost have precluded the widespread use of high-grade INSs in civilian underwater vehicles. Refer to Kinsey, Eustice, & Whitcomb (2006) for an extensive review of underwater vehicle navigation.

DVL-based DR navigation is commonly used in underwater vehicles (Brokloff, 1994; Whitcomb, Yoerger, & Singh, 1999; Kinsey & Whitcomb, 2004). The DVL measures the velocity of the vehicle with respect to the seafloor, and it reports it as a 3×1 vector in the instrument frame:

$$ {}^{i}\dot{p}_d(t) = \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{z}(t) \end{bmatrix}. \tag{42} $$

The instrument-frame velocity, ${}^{i}\dot{p}_d(t)$, is converted to the world-frame velocity, ${}^{w}\dot{p}_d(t)$, by the linear transformation

$$ {}^{w}\dot{p}_d(t) = {}^{w}_{v}\mathbf{R}(t)\ {}^{v}_{i}\mathbf{R}\ {}^{i}\dot{p}_d(t), \tag{43} $$

where ${}^{w}_{v}\mathbf{R}(t)$ is the time-varying rotation matrix from the vehicle frame to the world frame, measured by a three-axis orientation sensor. The accuracy and precision of this orientation sensor are crucial (Joyce, 1989; Pollard & Read, 1989; Brokloff, 1994; Münchow et al., 1995; Whitcomb, Yoerger, & Singh, 1999; McEwen et al., 2005). A three-axis gyrocompass based on fiber-optic gyroscopes (FOGs) provides dynamic accuracy on the order of 0.1°, and using a FOG gyrocompass for DR navigation gives better results than using a gyrostabilized magnetic compass (Kinsey & Whitcomb, 2004). The DVL alignment ${}^{v}_{i}\mathbf{R}$ is the constant rotation matrix from the DVL instrument frame to the vehicle frame—our goal is to use the rotor identifier presented in Section 3 to estimate this rotation.

The DR position estimate, ${}^{w}p_d$, is computed by integrating the world velocities ${}^{i}\dot{p}$ in time:

$$ {}^{w}p_d(t) = {}^{w}p_d(t_0) + \int_{t_0}^{t} {}^{w}_{v}\mathbf{R}(\tau)\ {}^{v}_{i}\mathbf{R}\ {}^{i}\dot{p}(\tau)d\tau, \tag{44} $$

where ${}^{w}p(t_0)$ is the initial position estimate. For a more detailed discussion of DR navigation in underwater vehicles, see Brokloff (1994); Whitcomb, Yoerger, & Singh (1999); Kinsey & Whitcomb (2004); and Stanway (2012).

4.1. The Alignment Calibration Problem

If a FOG is used to estimate the vehicle's orientation, then the principal source of error in the DR position estimate (44) is the alignment, ${}^{v}_{i}\mathbf{R}$, between the DVL and the FOG (Whitcomb, Yoerger, & Singh, 1999; Kinsey & Whitcomb, 2004). In cases in which the DVL and the attitude sensor can be assembled as a single mechanical unit, e.g., Alameda (2002), the alignment can be calibrated once under controlled laboratory conditions by the manufacturer. The *in situ* alignment problem arises when the DVL and the orientation sensors cannot be assembled as a single unit because of size, weight, power, pressure housing limitations, or reconfigurability requirements. These concerns often necessitate that the DVL and FOG be mounted at different locations on the vehicle. For example, on the *Sentry* AUV, the DVL is mounted forward on the keel of the vehicle, while the FOG is mounted inside a pressure housing approximately 1.5 m away. This arrangement precludes calibrating the alignment of the sensors during manufacture, thus necessitating *in situ* alignment calibration from data available during normal submerged vehicle operations.

In situ estimation of FOG/DVL alignment using LS techniques has been previously reported by researchers using the DVL for navigation (Brokloff, 1994; Kinsey &

Whitcomb, 2007b) and for water velocity measurements (Joyce, 1989; Pollard & Read, 1989; Münchow et al., 1995). Kinsey and Whitcomb derive an $SO(3)$ constrained adaptive identifier and apply it to this problem in Kinsey & Whitcomb (2007a). Recently, Troni and Whitcomb have adapted the LS methods to estimate the alignment between DVL and INS (Troni & Whitcomb, 2010, 2011; Troni, Kinsey, Yoerger, & Whitcomb, 2012). Their method eliminates the need for external position measurements, effectively making it a strapdown calibration technique.

The sensor alignment problem differs from the internal gyrocompass calibration problem (e.g., identification of bias, temperature dependence, and nonorthogonality). As with previously reported results, we assume the presence of an internally calibrated FOG gyrocompass, and we are concerned with estimating the alignment between the FOG and DVL. This paper differs from previously reported results in that it uses GA to formulate a rotor adaptive identifier to estimate the DVL/FOG alignment.

In Kinsey & Whitcomb (2007b), the authors integrate (44) by parts to obtain

$$\underbrace{{}^{v}_{w}\mathbf{R}(t)\,{}^{w}p_l(t) - \int {}^{v}_{w}\dot{\mathbf{R}}(\tau)\,{}^{w}p_l(\tau)d\tau}_{y(t)} = \underbrace{{}^{v}_{i}\mathbf{R}}_{\mathbf{R}} \underbrace{\int {}^{i}\dot{p}_d(\tau)d\tau}_{u(t)}, \quad (45)$$

which is the LA form of the identification plant (24). The input signal, $u(t)$, and the output signal, $y(t)$, are functions of *in situ* sensor measurements.

In our experiments, these signals are based on data obtained from the following three navigation sensors:

1. **DVL**: A bottom-lock DVL providing three-axis translational velocities, ${}^{i}\dot{p}_d(t_i)$, in instrument coordinates. Commercially available units can provide velocity measurements with single-ping standard deviations less than 0.3% with an accuracy of 0.2–0.4% and update rates up to 7 Hz.
2. **FOG**: A three-axis North-seeking FOG gyrocompass providing absolute orientation, ${}^{w}_{v}\mathbf{R}(\tau)$, and orientation rate, ${}^{w}_{v}\dot{\mathbf{R}}(\tau)$, with respect to true North and the local gravitational field. Commercially available units can provide dynamic measurements at up to 100 Hz with a precision of about 0.1°.
3. **LBL**: An LBL acoustic positioning system providing absolute three-dimensional vehicle position measurements, ${}^{w}p_l(t_k)$, at relatively low update rates. Commercially available 10–40 kHz systems can provide absolute precision of 0.1–10 m with update rates varying from 1.0 to 0.05 Hz over ranges of approximately 10 km. Higher-frequency 200–500 kHz systems are capable of subcentimeter precision and update rates of up to 10 Hz, but they are limited to ranges less than 100 m.

Our goal is to use these sensor signals to experimentally determine an estimate of the unknown sensor alignment. The rotor identifier will use the integrated vector input and output signals ($u(t)$ and $y(t)$, respectively) from (45) in the identification plant (24):

$$y = \mathbf{R}u \quad \mapsto \quad y = Ru\tilde{R}, \quad (46)$$

to estimate the alignment rotor R. Once the alignment is known, the DVL and FOG signals can precisely determine the vehicle position by DR (44), or with stochastic state estimation techniques such as Kalman filtering (Kalman, 1960).

For simplicity of exposition, we have assumed that (i) the LBL transponder is colocated with the DVL at the origin of the vehicle's reference frame, and (ii) the coordinate axes of the FOG define the coordinate axes of the vehicle frame. While these assumptions greatly simplify the navigation equations, they can easily be relaxed without requiring any fundamental change in the proposed methodology. This assumption was also made in the experiments reported in Kinsey & Whitcomb (2007a,b).

The DVL, FOG, and LBL all provide asynchronous measurements in discrete time. We first resample the data to provide synchronous measurements, and we use (45) to calculate the input and output signals of the rotation plant. We then estimate the unknown DVL/FOG alignment using the discrete-time form of the rotor identifier (41c).

In the following sections, we report the experimental validation of the rotor identifier using data collected by underwater robots in the laboratory and in the field. For all experiments, the initial condition of the rotor estimate was $S = 1$. Analysis starting with alternate initial conditions also converged to the same alignment estimate.

For the following experiments, we numerically tested several gains and chose $\kappa_p = 0.001$. This value was not overly sensitive to noise in the input measurements, and it provided a convergence speed roughly comparable to the $SO(3)$ adaptive identifier method. As mentioned in Section 3.8.1, the effect of the gain on the rotor identifier is consistent with any other proportional identifier or controller: higher proportional gain renders the system more sensitive to noise, but lower proportional gain slows the system response.

5. LABORATORY EXPERIMENTS

We first evaluate the performance of the rotor identifier by using it to estimate the DVL/FOG alignment on a laboratory ROV. The experiments were conducted in the Hydrodynamics Test Facility at Johns Hopkins University (Kinsey, Smallwood, & Whitcomb, 2003) using the *JHUROV* (Figure 7), a testbed vehicle for dynamics, control, and navigation studies for deep submergence vehicles (Smallwood, Bachmayer, & Whitcomb, 1999). *JHUROV* is instrumented with sensors similar to those found on vehicles in the field, except that some of the laboratory systems have higher precision and shorter range. The sensors relevant to this study are listed in Table I.

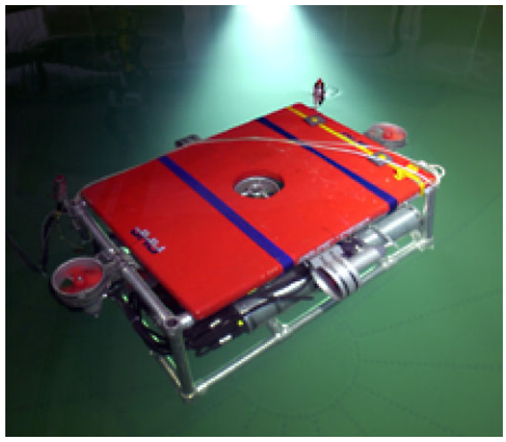


Figure 7. The Johns Hopkins University ROV is a laboratory system outfitted with sensors similar to those typically found on oceanographic research vehicles. It is a testbed vehicle used for dynamics, control, and navigation studies for deep submergence vehicles (photo credit: JHU DSCL).

The rotor identifier was used to estimate DVL/FOG alignment on *JHUROV* during six laboratory experiments. Each experiment was 40 min long, during which the robot was in closed-loop position control following predefined trajectories. The duration of the data acquisition is long enough to converge on a solution but short enough to avoid potential problems from unmodeled DR integration drift. These experiments are discussed extensively in Kinsey & Whitcomb (2007b).

5.1. Methodology

Using these data, four techniques were used to compute the DVL/FOG alignment:

- **Visual**: This is a visual alignment estimate of the DVL alignment with the instrument. Relative to the FOG, the DVL rotated 45° in heading and 0° in roll and pitch.
- **LS-SO(3)**: The $SO(3)$ constrained least squares method originally reported in Umeyama (1991) and applied to DVL/FOG alignment in Kinsey & Whitcomb (2007b).
- **ID-SO(3)**: The adaptive identifier on $SO(3)$ reported in Kinsey & Whitcomb (2007a). This has been slightly modified by normalizing the input u and output y fed to the identifier. This change preserves directional information in the measurements, but it changes the behavior of the identifier with respect to gains. It was necessary for the identifier to work on the field experiments.
- **ID-Spin(3)**: The adaptive identifier on $Spin(3)$, the group of rotors in $\mathbb{G}_3$, as introduced in Stanway & Kinsey (2011) and detailed in this paper.

To experimentally evaluate the rotor identifier, we compare DR trajectories (44) calculated with the rotor alignment estimate to LBL observations. The DR trajectories were also computed for the visual, LS-$SO(3)$, and ID-$SO(3)$ alignment estimates. This analysis uses the same renavigation implementation as in Kinsey & Whitcomb (2007a,b). Rotor alignment estimates were translated directly into equivalent rotation matrices using the basis vector image method (Dorst, Fontijne, & Mann, 2007).

Having recomputed the DR trajectories, we calculate the *position residual* $\Delta\mathbf{p}$ which is the difference between the DR estimate and the LBL measurement of vehicle position:

$$\Delta\mathbf{p} = \mathbf{p}^{\mathrm{n}}_{\mathrm{DR}} - \mathbf{p}^{\mathrm{n}}_{\mathrm{LBL}}, \tag{47a}$$

$$\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} = \begin{bmatrix} x^{\mathrm{n}}_{\mathrm{DR}} - x^{\mathrm{n}}_{\mathrm{LBL}} \\ y^{\mathrm{n}}_{\mathrm{DR}} - x^{\mathrm{n}}_{\mathrm{LBL}} \\ z^{\mathrm{n}}_{\mathrm{DR}} - z^{\mathrm{n}}_{\mathrm{LBL}} \end{bmatrix}. \tag{47b}$$

Figure 8 shows the components of the position residual in laboratory experiment 070.

The position residual serves as the basis for several other performance metrics:

1. The components of the position residual (47b) are sorted into bins, and the distribution is plotted for the residual along each direction. These are the *residual component distributions*. The horizontal axis shows the residual, and the vertical axis shows the normalized frequency (Figure 9).
2. We summarize the residual component distributions by computing the standard deviations of the residual in the x, y, and z degrees of freedom. These are the *residual*

Table I. Navigation sensors on *JHUROV* used in laboratory experiments [adapted from Kinsey & Whitcomb (2007a, 2007b)].

Quantity	Sensor	Sample rate	Precision	Range
(x, y) position	300 kHz LBL (Marquest SHARPS)	5 Hz	$5e^{-3}$ m	100 m
depth	pressure transducer (Paroscientific Digiquartz)	1.4 Hz	$1e^{-4}$ m	10 m
orientation	FOG gyrocompass (IXSEA PHINS)	10 Hz	0.01°	360°
velocity	1200 kHz DVL (TRDI WHN)	7 Hz	0.3%	~30 m

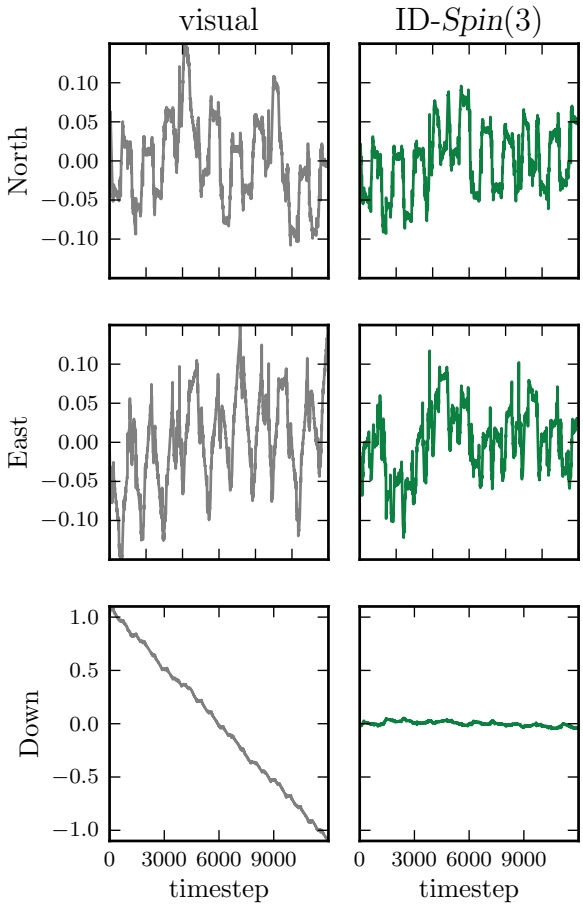


Figure 8. Components of the position residual from laboratory experiment 070. In the left column, residuals from DR with the visual alignment estimate are plotted in gray. In the right column, residuals from DR with the ID-*Spin*(3) alignment estimate are plotted in green. Note that the vertical component of the residual from the visual alignment is much larger than the others, and that the ID-*Spin*(3) alignment reduces it considerably.

component standard deviations. These are collected in the *residual component standard deviation vector*

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \end{bmatrix} = \begin{bmatrix} \text{std}\left(x^{\text{n}}_{\text{DR}} - x^{\text{n}}_{\text{LBL}}\right) \\ \text{std}\left(y^{\text{n}}_{\text{DR}} - y^{\text{n}}_{\text{LBL}}\right) \\ \text{std}\left(z^{\text{n}}_{\text{DR}} - z^{\text{n}}_{\text{LBL}}\right) \end{bmatrix} \in \mathbb{R}^3. \tag{48}$$

The Euclidean norm of this vector is termed the *standard deviation vector norm*. This provides a scalar metric for computing ensemble statistics.

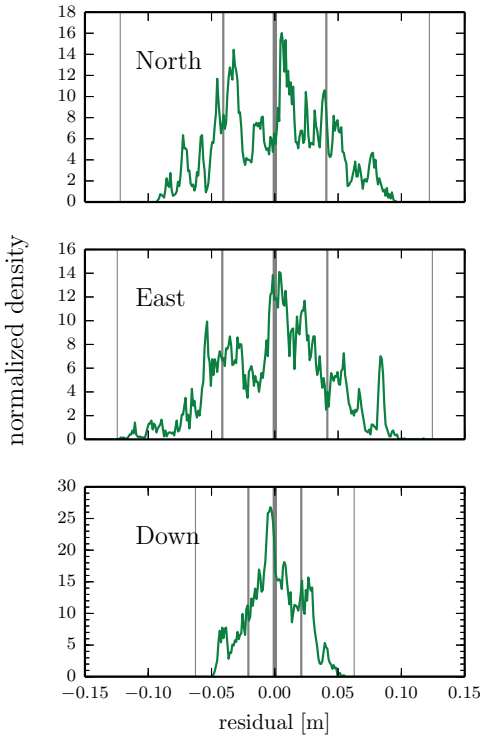


Figure 9. Distributions of the position residual components from laboratory experiment 070. The thick vertical line marks the mean of the residual, the medium vertical lines show one standard deviation (1σ), and the thin vertical lines show three standard deviations (3σ).

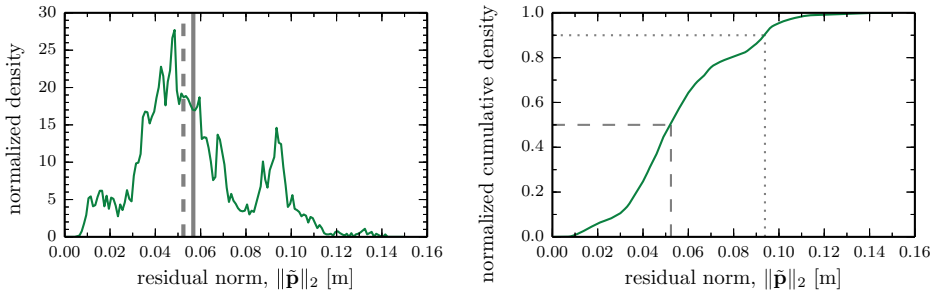
3. We also compute the Euclidean norm of the position residual at each point in time:

$$\|\Delta\mathbf{p}\|_2 = \sqrt{(\Delta\mathbf{p})\,(\Delta\mathbf{p})^T} \in \mathbb{R}^{n\times 1}. \tag{49}$$

This is the *residual magnitude*. Since the magnitude of $\Delta\mathbf{p}$ is always positive, this produces a one-sided distribution (Figure 10a). The frequency axis is normalized so that the distribution integrates to 1. The integral of this distribution is the *cumulative residual magnitude distribution*—it shows the fraction of the residuals that fall below a certain magnitude. For example, (Figure 10b) shows that half of the position residual vectors are less than 0.053 m, and 90% of them are less than 0.094 m. We use the mean and median to summarize this metric.

5.2. Alignment Identification

Figure 11 shows the normalized distribution of the residual magnitude for experiment 070. Table II shows the standard deviations. For the visual alignment, the range of the residuals exceeds 1 m. Applying the rotor alignment estimate decreases the range to 0.14 m and is comparable to the LS-*SO*(3) and ID-*SO*(3) methods.



(a) Distribution (b) Cumulative distribution

Figure 10. Distribution (a) and cumulative distribution (b) of the position residual magnitude from laboratory experiment 070, using the alignment estimated by the rotor identifier. On the distribution in the left panel (a): the thick vertical line marks the mean, and the dashed vertical line marks the median. On the cumulative distribution in the right panel (b): the dashed lines show that 50% of the residuals fall below 0.053 m, and 90% fall below 0.094 m.

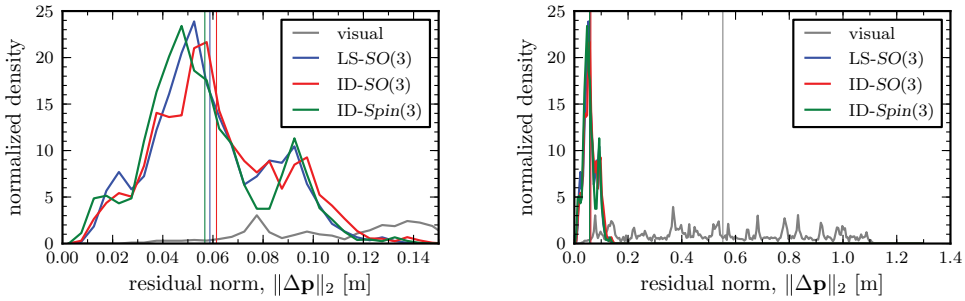


Figure 11. Residual magnitude distribution on *JHUROV* laboratory dataset 070, renavigated using different DVL/FOG alignment estimates. The left figure is a zoom into the left side of the right figure, to see the small differences between the alignment methods. Vertical lines indicate the mean of the residual magnitude. The rotor identifier presented here performs comparably to the LS-$SO(3)$ batch method in Kinsey & Whitcomb (2007b) and the ID-$SO(3)$ method with normalized inputs, adapted from Kinsey & Whitcomb (2007a). All three methods perform significantly better than the visual alignment estimate.

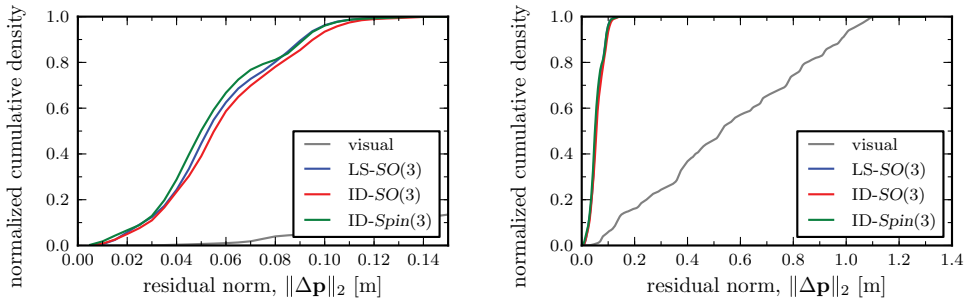


Figure 12. Cumulative residual magnitude distribution for *JHUROV* laboratory dataset 070, renavigated using different DVL/FOG alignment estimates. The left figure is a zoom into the left side of the right figure, to see the small differences between the alignment methods. The rotor identifier presented here performs comparably to the LS-$SO(3)$ batch method in Kinsey & Whitcomb (2007b) and the ID-$SO(3)$ method with normalized inputs, adapted from Kinsey & Whitcomb (2007a). All three methods perform significantly better than the visual alignment estimate.

Figure 12 shows the normalized cumulative distribution of the residual magnitude. For example, 50% of the residuals in expt070 are below 0.053 m for the rotor identifier, but that same fraction of residuals extends as high as 0.53 m using the visual alignment estimate. Again, the ID-$Spin(3)$ case provides comparable performance to the LS-$SO(3)$ and ID-$SO(3)$ methods.

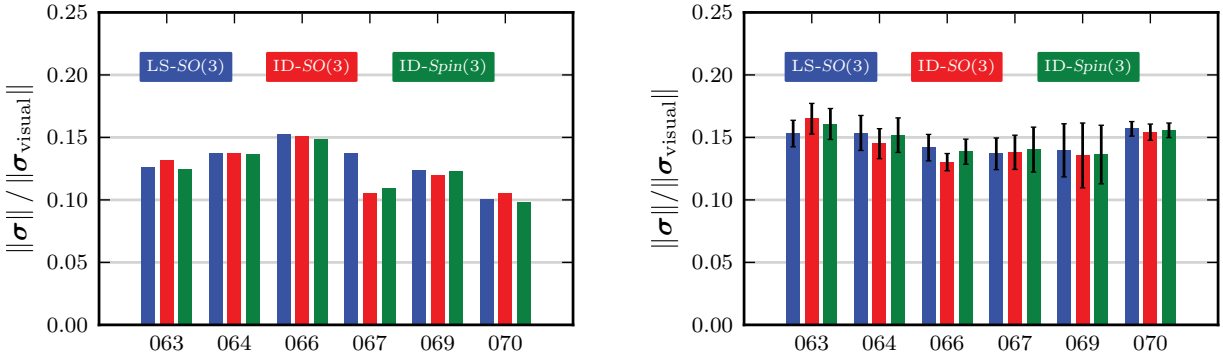


Figure 13. Experimental validation of alignment estimates on laboratory data. Self-validation (left) is illustrated by the normalized standard deviation vector norm calculated using data from the same experiment as was used to estimate the alignment. All datasets show an improvement in navigation over the original (visual) alignment estimate. Cross-validation statistics (right) were computed using the alignment estimate from one dataset to renavigate all other datasets. Bars show the mean normalized magnitude of the standard deviation vector, and whiskers indicate the standard deviation of this magnitude across the other datasets.

Table II. Position residual statistics after renavigation of lab datasets collected by *JHUROV*.

dataset	alignment method	σ_{North}	σ_{East}	σ_{depth}	$\|\boldsymbol{\sigma}\|$	$\frac{\|\boldsymbol{\sigma}\|}{\|\boldsymbol{\sigma}_{\text{visual}}\|}$	mean $\|\Delta\mathbf{p}\|$	median $\|\Delta\mathbf{p}\|$
expt063	visual	0.048941	0.051171	0.562006	0.566449	1.000000	0.493259	0.477453
	LS-*SO*(3)	0.048044	0.045893	0.026068	0.071372	0.125999	0.065268	0.062575
	ID-*SO*(3)	0.049499	0.045983	0.031374	0.074491	0.131505	0.068100	0.064871
	ID-*Spin*(3)	0.049080	0.045934	0.021666	0.070627	0.124683	0.063832	0.061798
expt064	visual	0.054306	0.055148	0.559550	0.564878	1.000000	0.491591	0.477783
	LS-*SO*(3)	0.050364	0.052372	0.026778	0.077437	0.137086	0.071144	0.069237
	ID-*SO*(3)	0.051481	0.052184	0.024861	0.077405	0.137030	0.070971	0.068505
	ID-*Spin*(3)	0.051152	0.052090	0.023962	0.076838	0.136026	0.070451	0.068020
expt066	visual	0.055127	0.063670	0.634868	0.640429	1.000000	0.561986	0.539055
	LS-*SO*(3)	0.051213	0.081422	0.016910	0.097664	0.152498	0.089230	0.087373
	ID-*SO*(3)	0.049993	0.080407	0.018780	0.096526	0.150721	0.088191	0.085575
	ID-*Spin*(3)	0.049093	0.079424	0.018137	0.095117	0.148521	0.086825	0.083930
expt067	visual	0.045764	0.054911	0.652066	0.655972	1.000000	0.575595	0.593576
	LS-*SO*(3)	0.056585	0.067803	0.018521	0.090234	0.137558	0.080677	0.077294
	ID-*SO*(3)	0.037244	0.053945	0.021890	0.069111	0.105356	0.060573	0.056939
	ID-*Spin*(3)	0.038036	0.057167	0.019969	0.071509	0.109012	0.062514	0.057791
expt069	visual	0.050956	0.061569	0.593581	0.598937	1.000000	0.523565	0.527018
	LS-*SO*(3)	0.047439	0.053742	0.019076	0.074179	0.123850	0.067636	0.065466
	ID-*SO*(3)	0.045226	0.052778	0.018236	0.071857	0.119974	0.065144	0.063387
	ID-*Spin*(3)	0.046041	0.053130	0.021236	0.073441	0.122618	0.066976	0.065171
expt070	visual	0.060111	0.053238	0.624291	0.629433	1.000000	0.552794	0.531055
	LS-*SO*(3)	0.040129	0.040330	0.027860	0.063349	0.100645	0.058784	0.054766
	ID-*SO*(3)	0.042558	0.040951	0.030262	0.066363	0.105432	0.061415	0.057304
	ID-*Spin*(3)	0.041485	0.040701	0.020898	0.061760	0.098120	0.056765	0.052385

The other laboratory datasets show similar results. When using the rotor identifier alignment:

- The residual component distributions possess lower standard deviations—the standard deviation vector norm is 9–15% of the value using the visual alignment.
- The mean residual magnitude is on the order of 10^{-2} m, compared to 10^{-1} m using the visual alignment.
- Half of the residuals have a magnitude less than 0.09 m, compared to 0.60 m using the visual alignment.

Similar results are observed for the LS-$SO(3)$ or ID-$SO(3)$ alignment estimates showing that the rotor identifier provides comparable performance.

Table II summarizes the statistics of the renavigation residuals for all six laboratory experiments. The rotor identifier provides modest reductions in the navigation errors for the horizontal components ($\sigma_{\text{North}}, \sigma_{\text{East}}$) and significant reductions in the vertical direction (σ_{Depth}). Figure 13 shows the normalized residual component standard deviation vector norm normalized by the value for the visual estimate (i.e., this value equals unity for the visual estimate). Using the alignment estimated by the rotor identifier reduces the standard deviation vector norm to 10–15% of its original value with the visual alignment estimate. The mean and median of the magnitude of the residuals are similarly reduced. Again, the rotor identifier performs comparably to the previously reported rotation matrix methods, and all of these perform significantly better than the original visual alignment.

5.3. Cross-validation

Given that the *JHUROV* is a rigid-body and the sensors are not moved between experiments, we expect that an alignment estimate from one experiment should perform comparably on the other experiments. To test this hypothesis, we renavigated each experiment using alignment estimates from the other five experiments. We computed the position residuals and the accompanying normalized residual component standard deviation, which is itself a vector. The results are summarized by computing the statistics of this vector for each experiment. The right plot in Figure 13 shows the mean and standard deviation of the magnitude of the residual standard deviation vector. When compared to an experiment's own alignment estimate, the alignment estimates from other experiments provide a comparable reduction in the navigation errors. We also tested the cross-validation hypothesis on the LS-$SO(3)$ and ID-$SO(3)$ techniques, and we observed a similar result. This is consistent with results reported in Kinsey & Whitcomb (2007a,b).

6. FIELD EXPERIMENTS

Having validated the performance of the rotor identifier with laboratory data, we then evaluated its performance on data obtained with an AUV operating in the field. The data were obtained by *Sentry* (Figure 14), a 6,000 m depth-rated AUV used for scientific research (Kinsey et al., 2011) during an expedition to the Håkon Mosby mud volcano, at 1,200 m depth on the continental slope of the Barents Sea. Table III summarizes the navigation sensors used on this cruise. The velocity and attitude sensors are identical to those used in the laboratory experiments. The depth and XY position sensors are similar to those used in the laboratory experiments, but with increased range to allow for use in the deep-ocean. For example, instead of using the 300 kHz acoustic positioning system used with the *JHUROV*, a network of three 12 kHz LBL transponders was deployed and surveyed prior to the dives. The tradeoff for this increase in range is a degradation in precision. For example, the precision of the LBL systems goes from $5e^{-3}$ m in the laboratory experiments to $2.5e^{-1}$ m for the field experiments and the update rate decreases from 5 to 0.1 Hz.

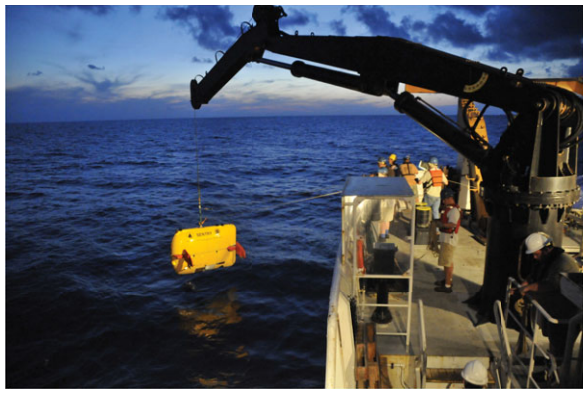


Figure 14. The National Deep Submergence Facility (NDSF) AUV *Sentry* is an unmanned oceanographic research vehicle that dives as deep as 6,000 m to perform various scientific missions. *Sentry* navigates primarily by DR with a FOG and DVL. The FOG is located in the main pressure housing, roughly amidship and just below the name label. The DVL is on the keel just aft from the bow. The large separation between these sensors on a vehicle that may deform under extreme pressure (even slightly) motivates our study of stable, *in situ* sensor alignment methods (photo credit: D Yoerger, *ABE/Sentry* Group, WHOI).

The reported rotor identifier was applied to navigation data from five dives on this cruise. We used the same analysis methodology as for the laboratory experiments (Section 5). Since the measurement period of the field LBL

Table III. Sensors on *Sentry* used in field experiments.

Quantity	Sensor	Sample rate	Precision	Range
(x, y) position	12 kHz LBL (WHOI)	0.1 Hz	~0.25 m	>10,000 m
depth	pressure transducer (Paroscientific Digiquartz)	1.4 Hz	0.02 m	6,500 m
orientation	FOG gyrocompass (IXSEA PHINS)	10 Hz	0.01°	360°
velocity	300 kHz DVL (TRDI WHN)	2 Hz	0.3%	~200 m

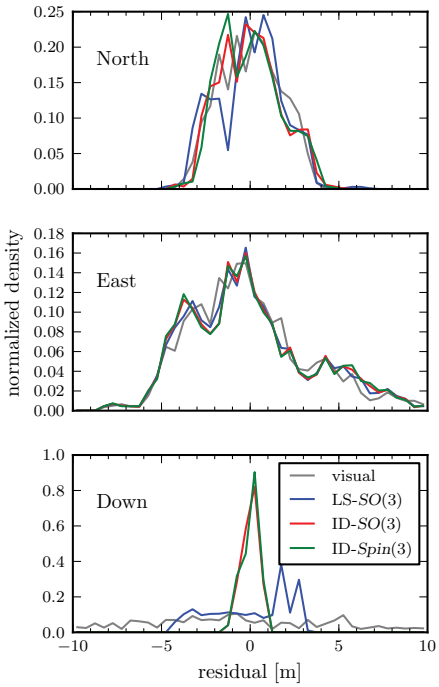


Figure 15. Distributions of residual components for *Sentry*075, renavigated using different DVL/FOG alignment estimates. The rotor identifier presented here performs comparably to the LS-*SO*(3) batch method in Kinsey & Whitcomb (2007b) and the ID-*SO*(3) method with normalized inputs, adapted from Kinsey & Whitcomb (2007a).

system is an order of magnitude longer than the laboratory system, the analyzed datasets are 2–4 h long.

6.1. Alignment Identification and Renavigated Trajectory

Figure 15 shows the normalized distribution of each component of the residual, and Figure 16a shows the normalized distribution of the residual magnitude for dive *Sentry*075. The magnitude distribution of the residuals for the rotor identifier alignment is slightly smaller than those for the visual estimate of the alignment. This is also reflected in the mean of the distribution. Again, the rotor identifier performs comparably to the previously reported rotation matrix identification methods.

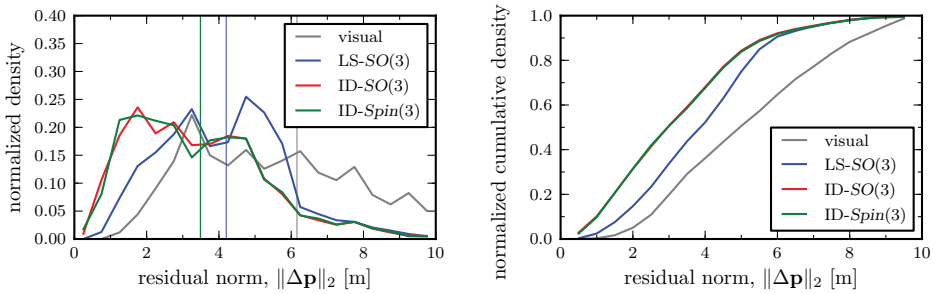
Figure 16b shows the normalized cumulative distribution of the residual magnitude. For *Sentry* 075, 50% of the residuals are lower than 3.3 m for the rotor identifier. The ID-SO(3) provides comparable performance. For the visual alignment estimate, 50% of the residuals are above 5.7 m magnitude.

Table IV lists the summarized statistics of the renavigation residuals for all five field experiments. Figure 17 shows these quantities graphically. As in the laboratory experiments, the rotor identifier performs comparably to the previously reported rotation matrix methods. For some dives, these perform only slightly better than the original visual alignment method. This suggests that the visual alignment estimate is very close to the actual DVL/FOG alignment, and a finer alignment is difficult to identify in these trajectories with the field LBL system.

6.2. Cross-validation

Again, it is reasonable to ask how the alignment estimated using one dataset performs on another. We cross-validated the alignment estimates using the same methodology used in the lab experiments. The right plot in Figure 17 shows the normalized standard deviation vector magnitude. It varies across alignment estimates—staying close to 1 for the most part.

The alignment estimates from dive 076 perform poorly, probably because *Sentry* was operating at low altitude and came into contact with the seafloor a few times. The DR data from that dive are suspect, but they are still included here for completeness.



(a) Distribution (b) Cumulative distribution

Figure 16. Distribution (a) and cumulative distribution (b) of the position residual magnitude from *Sentry*075, renavigated using different DVL/FOG alignment estimates. The rotor identifier presented here performs comparably to the ID-*SO*(3) method with normalized inputs, adapted from Kinsey & Whitcomb (2007a). On the distribution in the left panel (a): the vertical lines indicate the mean of the residual magnitude. The cumulative distribution in the right panel (b) shows that 50% of the residuals fall below 3.25 m for the adaptive identifier methods, while the LS and visual methods produce larger position residuals.

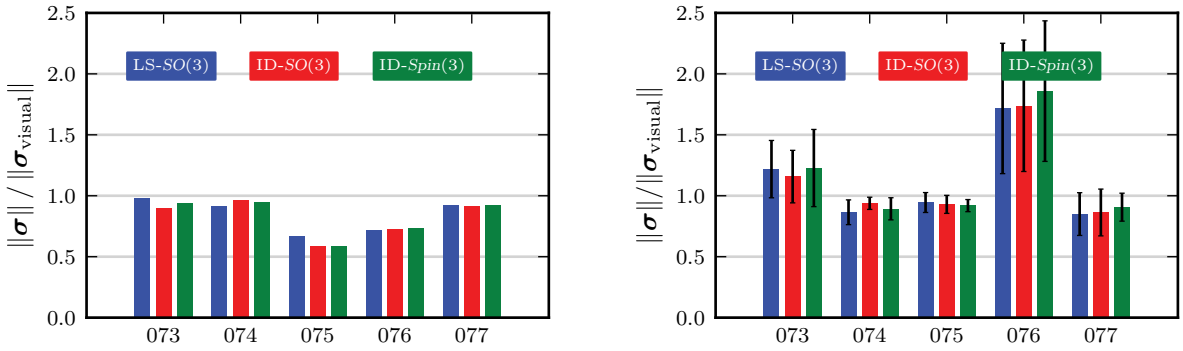


Figure 17. Experimental validation of alignment estimates on field data. Self-validation (left) is illustrated by the normalized standard deviation vector norm calculated using data from the same experiment as was used to estimate the alignment. All datasets show an improvement in navigation over the original (visual) alignment estimate. Cross-validation statistics (right) were computed using the alignment estimate from one dataset to renavigate all other datasets. Bars show the mean normalized magnitude of the standard deviation vector, and whiskers indicate the standard deviation of this magnitude across the other datasets.

Table IV. Position residual statistics after renavigation of field datasets collected by *Sentry*.

dataset	alignment method	σ_{North}	σ_{East}	σ_{depth}	$\|\boldsymbol{\sigma}\|$	$\frac{\|\boldsymbol{\sigma}\|}{\|\boldsymbol{\sigma}_{\text{visual}}\|}$	mean $\|\Delta\mathbf{p}\|$	median $\|\Delta\mathbf{p}\|$
sentry073	visual	4.925604	7.024044	0.199698	8.581297	1.000000	7.626768	6.779823
	LS-*SO*(3)	3.107703	6.747554	4.000094	8.437302	0.983220	7.834738	7.582955
	ID-*SO*(3)	2.821279	6.756833	2.519565	7.743553	0.902376	7.020782	6.495987
	ID-*Spin*(3)	2.886001	6.755998	3.344796	8.072185	0.940672	7.369756	6.960915
sentry074	visual	2.097726	1.492669	0.570155	2.636966	1.000000	2.485443	2.453175
	LS-*SO*(3)	2.082878	1.223801	0.122710	2.418911	0.917308	2.274189	2.264834
	ID-*SO*(3)	2.091771	1.361575	0.483944	2.542360	0.964123	2.396811	2.369539
	ID-*Spin*(3)	2.089810	1.317990	0.316806	2.490938	0.944623	2.342860	2.332514
sentry075	visual	3.464273	1.699159	5.669578	6.858020	1.000000	6.160024	5.700183
	LS-*SO*(3)	3.545474	1.823476	2.246371	4.576203	0.667277	4.201113	4.155020
	ID-*SO*(3)	3.581354	1.714496	0.447683	3.995750	0.582639	3.485687	3.222189
	ID-*Spin*(3)	3.596213	1.696875	0.467718	4.003859	0.583821	3.489983	3.242537
sentry076	visual	5.554410	1.854264	5.938688	8.340131	1.000000	7.277020	6.342527
	LS-*SO*(3)	5.791469	0.743226	1.286456	5.979002	0.716895	4.311404	2.746244
	ID-*SO*(3)	5.761885	0.850114	1.652750	6.054222	0.725914	4.501826	2.912722
	ID-*Spin*(3)	5.764633	0.738832	1.984426	6.141238	0.736348	4.592775	2.975477
sentry077	visual	1.264703	2.807797	1.114158	3.274835	1.000000	2.933720	2.768180
	LS-*SO*(3)	1.217805	2.753298	0.308248	3.026337	0.924119	2.636209	2.552532
	ID-*SO*(3)	1.200990	2.745069	0.204503	3.003266	0.917074	2.607675	2.519493
	ID-*Spin*(3)	1.169840	2.751775	0.422730	3.019849	0.922138	2.643394	2.551747

The statistics from dive 073 also show that the identified estimates tend to do worse than the visual estimate across the other datasets. *Sentry*073 covered more distance at high altitude, with little change in depth—a trajectory that may not have provided enough information for any of the identifiers.

7. DISCUSSION

Looking at these experimental results, we observe a number of trends. First, while all three methods significantly reduced the normalized error for the laboratory experiments, the improvements are less dramatic for the field experiments. There are two possible causes for this: (1) the

position measurements from the 12 kHz LBL system (which has a lower resolution and update rate than the 300 kHz system used in the laboratory experiments) have more noise, limiting the precision that the identifier can achieve; and (2) the visual DVL/FOG alignment estimate on *Sentry* is very close to the actual alignment, and identification techniques can only provide a modest improvement. The second cause is also noted by Troni et al. (Troni & Whitcomb, 2011) in their field validation of a $SO(3)$-based alignment technique using DVL velocity and INS acceleration measurements. Because their method does not use position measurements but provides a similar result, the second cause is likely.

Second, we observe that the laboratory and field results show that the ID-$Spin(3)$ method proposed in this paper is comparable to the previously reported LS-$SO(3)$ and ID-$SO(3)$ techniques. For example, in 6 of the 11 experiments, the ID-$SO(3)$ method marginally outperforms the ID-$Spin(3)$ method. This is consistent with the derivation of these methods—neither of them should be expected to provide significantly better performance than the other. Both methods should identify the alignment up to the level of sensor noise in the measurements, as illustrated in the simulation results for the rotor identifier (Figure 5).

7.1. Advantages over Previously Reported Methods

Although the rotor identifier does not outperform existing methods in terms of navigated position residuals, the GA approach has several theoretical and practical advantages.

7.1.1. Simple Scalar Proportional Gain

The rotor identifier proposed here uses a simple single scalar proportional gain, κ_p, and the acceptable values for a stable system are well-defined. This provides the practitioner with a single parameter to tune the system for desired response time and noise sensitivity. The system responds to changes in κ_p in an intuitive way, similar to other proportional gain systems: higher κ_p leads to faster response at the cost of increased noise sensitivity, and lower κ_p makes the system less susceptible to noise at the cost of slower convergence time. The simplicity of this single scalar gain allowed us to tune the system quickly and easily, but it could also be beneficial for future efforts in gain scheduling or dynamic tuning. This is in contrast to some of the previous work using rotation matrices in LA [see, e.g., Kinsey & Whitcomb (2007a)] where the gain could potentially be a 3×3 matrix, complicating any manual or automated tuning. Furthermore, our approach might allow the addition of a derivative gain that would add damping to the system and provide better noise rejection—this is a subject for future research.

7.1.2. Computational Efficiency

The compactness and computational advantages of quaternions are widely documented [see, e.g., Brown (1989), Kuipers (1999), and Farrell (2008)]. Rotors in $\mathbb{G}_3$ are isomorphic to quaternions and have the same computational advantages, so the rotor identifier provides a more efficient method for estimating rotations in real time than the existing $SO(3)$-based methods. This advantage is important in applications with higher data update rates or for real-time estimation on lower-power computers that are being increasingly used in robots. Furthermore, representing the rotations as rotors instead of rotation matrices allows us to average alignment estimates directly (Buchholz & Sommer, 2005; Doran, 2000).

Existing quaternion-based methods also have these computational advantages, but most assume a complete observation at each time step, and they are not directly applicable to this particular problem. The identifier described in this paper could be implemented using quaternions instead of rotors, but GA adds concepts to simplify the formulation and stability proof (Section 7.1.3), so it makes more sense to use rotors.

7.1.3. Geometric Understanding, Straightforward Formulation, and Stability Proof

One of the main advantages of using GA is the clarity it brings to the mathematical formulation of the identifier and the proof of asymptotic stability. Using rotors in GA inherently constrains the solutions to valid rigid-body rotations. We define both parameter and output error first as rotors, and then we show how those rotors connect to bivector and scalar metrics for the same errors. These error metrics are more geometrically meaningful than the error measures used in previous work using LA, and GA makes the relationships between the error metrics clear.

We compose the output error rotor directly from the expected and observed output vectors using repeated reflections. We derive the output error bivector from the logarithm of the output error rotor. Then we solve the alignment problem by driving the first-order rotor kinematics with a simple proportional feedback regulator on the output error bivector.

The scalar parameter error angle provides a natural starting point for our Lyapunov stability analysis, and expanding the Lyapunov candidate function in terms of the parameter error bivector leads to a geometric understanding of the criteria for stability. This explains why a simple first-order output error feedback regulator can provide asymptotically stable identification, using only the most basic parts of GA.

7.1.4. Reduced Data Manipulation

The ID-$SO(3)$ method reported in Kinsey & Whitcomb (2007a) required the input and output data from the field datasets to be normalized prior to running the adaptive identifier (Section 5.1). Without normalizing first, the magnitude of the input and output vectors defined in (45)

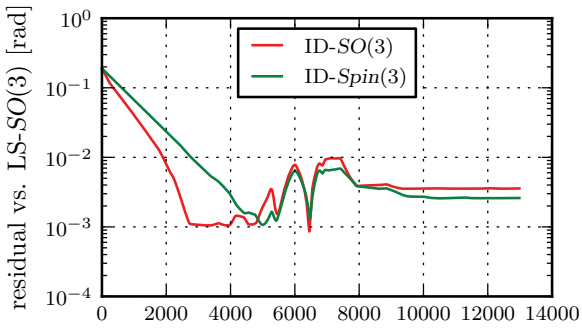


Figure 18. Scalar residual with respect to LS-SO(3) alignment estimate.

varies. Since the output error in Kinsey & Whitcomb (2007a) is defined using the cross-product between expected and observed outputs, this means that the magnitude of the output error varies over an even wider range. Normalizing the inputs and outputs limits the range of magnitudes for the output error while preserving the directional information and minimizing changes to the published algorithm. Effectively, this change means that the adjusted ID-$SO(3)$ method operates on the sine of the angle between expected and observed outputs, where the original ID-$SO(3)$ method operated on the scaled sine of the angle.

In contrast, the ID-$Spin(3)$ method reported here does not require any data modification or preprocessing. Our definition of the output error rotor (30) focuses on the directional information and not the magnitude of the expected and observed outputs. Since the output error rotor is defined to rotate the estimated output directly into the measured output along the shortest path on the sphere, the rotor identifier effectively operates on the angle between expected and observed outputs. This means that the effect of the proportional gain does not change with the angle between expected and observed outputs, as it does in both the original and adjusted forms of the ID-$SO(3)$ method, which operate on the sine of that angle.

7.2. Transient Response

Transient response is an important consideration for all dynamic systems, including the adaptive identifier reported here. Numerous factors affect the transient response—one especially important consideration is the effect of unknown excitations resulting from measurement noise. Figure 18 compares the time history of the ID-$SO(3)$ and ID-$Spin(3)$ identifiers on expt070. Since there is no ground truth available in these experiments, we use a scalar (geodesic) residual with the estimate from the batch LS-$SO(3)$ method as a reference. We choose the LS-$SO(3)$ as the basis for our comparison because it is a batch method and does not have a transient response—not because it provides superior estimates. In fact, for several of the experiments and renavigation performance metrics, the LS-$SO(3)$ method performs worse than one or both of the adaptive identifiers (see, e.g., Fugure 10b and Table II).

Both adaptive identifiers initially converge toward the same rotation estimate as the LS-$SO(3)$ method—the differences between the estimates are approximately 10^{-3} radians. Then, the adaptive identifiers fluctuate between input/output vector pairs 4000 and 8000. The magnitude of the input u and output y is initially large, and both pass through zero near vector pair 6000 before increasing in magnitude again. This is because the means of u and y were set to zero when mapping the DR measurements into the form of the identifier (Section 4.1). As u and y pass through zero, any noise in the measurements becomes proportionally larger, and both online identifiers will react to that noise. The issue here is that the identifiers encounter vanishingly small signal-to-noise ratio partway through the data as a result of preprocessing that was necessary to cast the data into the identifier form. The most obvious way to mitigate this problem is to average the estimated rotor over several time steps, e.g., using methods in Doran (2000) and Buchholz & Sommer (2005). The rotor adaptive identifier formulation presented within provides a potential alternative solution—adding a derivative gain to the update law will introduce damping into the solution; this is a topic for future research.

8. CONCLUSION

This paper detailed a novel alternative approach to the sensor alignment problem. Rotations were encoded using rotors in Geometric Algebra, so that algebraic structure implicitly constrains parameter estimates to describe rigid body rotations. We proposed a stable adaptive identifier in Section 3, and we proved its stability using Lyapunov theory on the continuous-time system. We then provided an equivalent discrete-time implementation of the identifier. Section 4 provided the necessary information to apply the rotor identifier to the DVL/FOG alignment problem for DR in underwater navigation, and several performance metrics were defined. Section 5 applied the rotor identifier to identify DVL/FOG alignment on an ROV operating in controlled laboratory conditions, and Section 6 demonstrated the rotor identifier using field data from the AUV *Sentry*. In the end, the rotor identifier improved DR performance comparably to previously reported rotation matrix methods. The Geometric Algebra formulation provides a clear geometric and kinematic interpretation of the proposed rotor identifier, which provides accurate, stable alignment estimates. It may offer a clearer path forward in improving online alignment identifiers, including, for example, enhanced noise rejection using a proportional-derivative feedback formulation. It may also prove useful in formulating an identifier for full rigid body motions—including rotation and translation—which can be encoded by rotors in higher dimensions.

ACKNOWLEDGMENTS

This work was supported in part by the National Defense Science and Engineering Graduate Program (MJS), the Edwin A. Link Foundation (MJS), the WHOI Academic Programs Office (MJS), The Jessie B. Cox Endowed Fund in Support of Scientific Staff (JCK), and The Penzance Endowed Fund in Support of Assistant Scientists (JCK). We thank Lauren A. Cooney for her editing and helpful comments. Professor Franz Hover (MIT) and our anonymous reviewers also provided constructive feedback. Laboratory data were obtained with funding from NSF award 0427220 (L. Whitcomb). The field data were obtained on the 2010 HMMV cruise on the *FS Maria S. Merian* with support from 6th EU FP ESONET Project no. 036851 (D. de Beer), DFG (Deutsche Forschungsgemeinschaft) and MPG (Max Planck Gesellschaft) grants to A. Boetius, and an award from the WHOI Arctic Research Initiative (C. German, D. Yoerger, R. Camilli), We thank these scientists, the captain and crew of the *Merian*, the science party of the MSM 16-2 Expedition, and especially Andrew Billings and Alan Duester of the *Sentry* operations team for their support in obtaining the field data. We are indebted to Dr. Dana R. Yoerger for his help, and we gratefully appreciate his ongoing support of this research.

REFERENCES

Akella, M. R. (2001). Rigid body attitude tracking without angular velocity feedback. Systems & Control Letters, 42(4), 321–326.

Alameda Jr., W. (2002). Seadevil—A totally integrated inertial navigation system (INS) solution. In Proceedings of the Underwater Intervention Symposium, New Orleans, LA.

Arun, K. S., Huang, T. S., & Blostein, S. D. (1987). Least-squares fitting of two 3-D point sets. IEEE Transactions on Pattern Analysis and Machine Intelligence, 9(5), 698–700.

Bar-Itzhack, I., & Oshman, Y. (1985). Attitude determination from vector observations: Quaternion estimation. IEEE Transactions on Aerospace Electronics Systems, 21(1), 128–136.

Bayro-Corrochano, E. (2005). Handbook of geometric computing: Applications in pattern recognition, computer vision, neural computing, and robotics. Berlin: Springer.

Bayro-Corrochano, E. (2010). Geometric computing: For wavelet transforms, robot vision, learning, control and action. London: Springer.

Bayro-Corrochano, E., & Scheuermann, G. (eds.) (2010). Geometric algebra computing in engineering and computer science. London: Springer.

Bayro-Corrochano, E., & Sobczyk, G. (eds.) (2001). Geometric algebra with applications in science and engineering. Boston: Birkhauser.

Breckenridge, W. G. (1999). Quaternions—Proposed standard conventions. Technical Report IOM 343-79-1199, NASA Jet Propulsion Laboratory.

Britting, K. (1971). Inertial navigation system analysis. John Wiley and Sons.

Brokloff, N. (1994). Matrix algorithm for Doppler sonar navigation. In Proceedings of OCEANS (vol. 3, pp. 378–383). MTS/IEEE.

Brown, C. M. (1989). Some computational properties of rotation representations. Technical Report TR 303, University of Rochester, Computer Science Department, Rochester, NY.

Buchholz, S., & Sommer, G. (2005). On averaging in Clifford groups. In H. Li, P.J. Olver and G. Sommer (eds.). Computer Algebra and Geometric Algebra with Applications (pp. 229–238). Berlin: Springer Heidelberg.

Bullo, F., & Murray, R. M. (1995). Proportional derivative (PD) control on the Euclidean group. Technical Report Caltech/CDS 95–010, California Inst. of Tech., Pasadena, CA.

Bullo, F., Murray, R. M., & Sarti, A. (1995). Control on the sphere and reduced attitude stabilization. Technical Report Caltech/CDS 95–005, California Inst. of Tech., Pasadena, CA.

Caccavale, F., & Villani, L. (1999). Output feedback control for attitude tracking. Systems & Control Letters, 38(2), 91–98.

Caress, D., Thomas, H., Kirkwood, W., McEwen, R., Henthorn, R., Clague, D., Paull, C., Paduan, J., & Maier, K. (2008). High-resolution multibeam, sidescan, and subbottom surveys using the MBARI AUV D. Allan B. In Reynolds, J., & Greene, H. (eds.). Marine Habitat Mapping Technology for Alaska (pp. 47–69). Alaska Sea Grant.

Cartan, E. (1966). The theory of spinors. Cambridge, MA: MIT Press.

Choi, S. K., & Yuh, J. (1996). Experimental study on a learning control system with bound estimation for underwater robots. In Proceedings of the International Conference on Robotics and Automation (ICRA) (vol. 3, pp. 2160–2165). Minneapolis, MN: IEEE.

Choukroun, D., Bar-Itzhack, I. Y., & Oshman, Y. (2006). Novel quaternion Kalman filter. IEEE Transactions on Aerospace Electronics Systems, 42(1), 174–190.

Crassidis, J., Markley, F., & Cheng, Y. (2007). A survey of nonlinear attitude estimation methods. Journal of Guidance, Control, and Dynamics, 30(1), 12–28.

Doran, C., Hestenes, D., Sommen, F., & van Acker, N. (1993). Lie groups as spin groups. Journal of Mathematical Physics, 34(8), 3642–3669.

Doran, C., & Lasenby, A. (2003). Geometric algebra for physicists. Cambridge, UK: Cambridge University Press.

Doran, C. J. L. (2000). Bayesian inference and geometric algebra: An application to camera localization. In Bayro-Corrochano, E., & Sobczyk, G. (eds.). Geometric algebra: A geometric approach to computer vision, neural and quantum computing, robotics and engineering. Birkhäuser Boston.

Dorst, L., Fontijne, D., & Mann, S. (2007). Geometric algebra for computer science: An object-oriented approach to geometry. Morgan Kaufmann series in computer graphics and geometric modeling. Amsterdam: Elsevier.

Farrell, J., Stuelpnagel, J., Wessner, R., Velman, J., & Brook, J. (1966). A least squares estimate of satellite attitude (Grace Wahba). SIAM Review, 8(3), 384–386.

Farrell, J. A. (2008). Aided navigation: GPS with high rate sensors. New York: McGraw-Hill.

Fontijne, D. (2006). Gaigen 2: A Geometric Algebra implementation generator. In Proceedings of the International Conference on Generative Programming and Component Engineering (GPCE) (pp. 141–150). Portland, Oregon: ACM.

Fontijne, D. (2010). Gaigen2.5 Geometric Algebra implementation generator. visited: 29 Sep. 2011, http://g25.sourceforge.net/.

Fontijne, D., Bouma, T., & Dorst, L. (2001). Gaigen: A Geometric Algebra implementation generator. http://www.science.uva.nl/ga/gaigen/.

Fossen, T. I. (1994). Guidance and control of ocean vehicles. New York: John Wiley & Sons.

Gull, S. F., Lasenby, A. N., & Doran, C. J. L. (1993). Imaginary numbers are not real—The geometric algebra of spacetime. Foundations of Physics, 23(9), 1175–1201.

Hamel, T., & Mahony, R. (2006). Attitude estimation on SO(3) based on direct inertial measurements. In Proceedings of the International Conference on Robotics and Automation (ICRA) (pp. 2170–2175). Orlando, FL: IEEE.

Hestenes, D. (2002). New foundations for classical mechanics, vol. 99 of Fundamental theories of physics. The Netherlands: Springer.

Hunt, M. M., Marquet, W. M., Moller, D. A., Peal, K. R., Smith, W. K., & Spindel, R. C. (1974). An acoustic navigation system. Technical Report 74-6, Woods Hole Oceanographic Institution, Woods Hole, MA (unpublished manuscript).

Joyce, T. M. (1989). On in situ calibration of shipboard ADCPs. Journal of Atmospheric and Oceanic Technology, 6(5), 169–172.

Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. Transactions of the ASME—Journal of Basic Engineering, 82(Series D), 35–45.

Khalil, H. K. (1996). Nonlinear systems. Englewood Cliffs, NJ: Prentice-Hall.

Kinsey, J., Smallwood, D., & Whitcomb, L. (2003). A new hydrodynamics test facility for UUV dynamics and control research. In Proceedings of OCEANS (vol. 1, pp. 356–361). MTS/IEEE.

Kinsey, J. C., Eustice, R. M., & Whitcomb, L. L. (2006). A survey of underwater vehicle navigation: Recent advances and new challenges. In Proceedings of the Conference of Manoeuvering and Control of Marine Craft, Lisbon, Portugal. IFAC. Invited paper.

Kinsey, J. C., & Whitcomb, L. L. (2004). Preliminary field experience with the DVLNAV integrated navigation system for oceanographic submersibles. Control Engineering Practice, 12(12), 1541–1548.

Kinsey, J. C., & Whitcomb, L. L. (2007a). Adaptive identification on the group of rigid-body rotations and its application to underwater vehicle navigation. IEEE Transactions on Robotics, 23(1), 124–136.

Kinsey, J. C., & Whitcomb, L. L. (2007b). In situ alignment calibration of attitude and Doppler sensors for precision underwater vehicle navigation: Theory and experiment. IEEE Journal of Oceanic Engineering, 32(2), 286–299.

Kinsey, J. C., Yoerger, D. R., Jakuba, M. V., Camilli, R., Fisher, C. R., & German, C. R. (2011). Assessing the Deepwater Horizon oil spill with the Sentry autonomous underwater vehicle. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS) (pp. 261–267). San Francisco: IEEE/RSJ.

Koditschek, D. E. (1988). Application of a new Lyapunov function to global adaptive attitude tracking. In Proceedings of the Conference on Decision and Control (CDC) (pp. 63–68). Austin, TX: IEEE.

Kuipers, J. B. (1999). Quaternions and rotation sequences. Princeton, NJ: Princeton University Press.

Lawson, C. L., & Hanson, R. J. (1974). Solving least squares problems. Englewood Cliffs, NJ: Prentice Hall.

Lee, T., Leok, M., McClamroch, N. H., & Sanyal, A. (2007). Global attitude estimation using single direction measurements. In Proceedings of the American Control Conference (ACC) (pp. 3659–3664). New York: IEEE.

Mahony, R., Hamel, T., & Pflimlin, J.-M. (2005). Complementary filter design on the special orthogonal group SO(3). In Proceedings of the Conference on Decision and Control (CDC) (pp. 1477–1484). Seville, Spain: IEEE.

Mahony, R., Hamel, T., & Pflimlin, J.-M. (2008). Nonlinear complementary filters on the special orthogonal group. IEEE Transactions on Automatic Control, 53(5), 1203–1218.

Markley, F. L. (1988). Attitude determination using vector observations and the singular value decomposition. Journal of Astronautical Sciences, 36(3), 245–258.

McEwen, R., Thomas, H., Weber, D., & Psota, F. (2005). Performance of an AUV navigation system at Arctic latitudes. IEEE Journal of Oceanic Engineering, 30(2), 443–454.

Meyer, G. (1971). Design and global analysis of spacecraft attitude control systems. Technical report, NASA.

Milne, P. H. (1983). Underwater acoustic positioning systems. Gulf Publishing Company.

Münchow, A., Coughran, C. S., Hendershott, M. C., & Winant, C. D. (1995). Performance and calibration of an acoustic Doppler current profiler towed below the surface. Journal of Atmospheric and Oceanic Technology, 12(2), 435–444.

Murray, R. M., Li, Z., & Sastry, S. S. (1994). A mathematical introduction to robotic manipulation. Washington, DC: CRC Press.

Narendra, K., & Annaswamy, A. (1988). Stable adaptive systems. New York: Prentice Hall.

Pizarro, O., Eustice, R. M., & Singh, H. (2009). Large area 3-D reconstructions from underwater optical surveys. IEEE Journal of Oceanic Engineering, 34(2), 150–169.

Pollard, R., & Read, J. (1989). A method for calibrating ship-mounted acoustic Doppler profilers and the limitations of gyro compasses. Journal of Atmospheric and Oceanic Technology, 6(6), 859–865.

Sanyal, A. K., Lee, T., Leok, M., & McClamroch, N. H. (2008). Global optimal attitude estimation using uncertainty ellipsoids. Systems & Control Letters, 57(3), 236–245.

Sastry, S., & Bodson, M. (1989). Adaptive control: Stability, convergence, and robustness. Prentice Hall.

Schönemann, P. H. (1966). A generalized solution of the orthogonal Procrustes problem. Psychometrika, 31(1).

Shuster, M. D., & Oh, S. D. (1981). Three-axis attitude determination from vector observations. Journal of Guidance, Control, and Dynamics, 4(1), 70–77.

Singh, H., Roman, C., Pizarro, O., Eustice, R., & Can, A. (2007). Towards high-resolution imaging from underwater vehicles. International Journal of Robotics Research, 26(1), 55–74.

Smallwood, D., Bachmayer, R., & Whitcomb, L. L. (1999). A new remotely operated underwater vehicle for dynamics and control research. In Proceedings of the 11th International Symposium on Unmanned Untethered Submersible Technology (pp. 370–377). Durham, NH.

Sommer, G. (2001). Geometric computing with Clifford algebras: Theoretical foundations and applications in computer vision and robotics. Berlin: Springer.

Stanway, M. J. (2012). Contributions to automated real-time underwater navigation. Ph.D. thesis, MIT/Woods Hole Oceanographic Inst. Joint Program, Cambridge and Woods Hole, MA.

Stanway, M. J., & Kinsey, J. C. (2011). Sensor alignment using rotors in Geometric Algebra. In Proceedings of the International Conference on Robotics and Automation (ICRA) (pp. 994–999). Shanghai, China: IEEE.

Strang, G. (1986). Introduction to applied mathematics. Wellesley, MA: Wellesley-Cambridge Press.

Titterton, D., & Weston, J. (1997). Strapdown inertial navigation. London: Peter Peregrinus Ltd.

Troni, G., Kinsey, J., Yoerger, D., & Whitcomb, L. (2012). Field performance evaluation of new methods for in-situ calibration of attitude and Doppler sensors for underwater vehicle navigation. In Proceedings of the International Conference on Robotics and Automation (ICRA). St. Paul, MN: IEEE.

Troni, G., & Whitcomb, L. (2010). New methods for in-situ calibration of attitude and Doppler sensors for underwater vehicle navigation: Preliminary results. In Proceedings of OCEANS. MTS/IEEE.

Troni, G., & Whitcomb, L. (2011). Experimental evaluation of new methods for in-situ calibration of attitude and Doppler sensors for underwater vehicle navigation. In Proceedings of the International Conference on Intelligent Robots and Systems (IROS). San Francisco, CA: IEEE/RSJ.

Umeyama, S. (1991). Least-squares estimation of transformation parameters between two point patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(5), 376–380.

Wahba, G. (1965). A least squares estimate of satellite attitude. SIAM Review, 7(3), 409.

Wen, J. T.-Y., & Kreutz-Delgado, K. (1991). The attitude control problem. IEEE Transactions on Automatic Control, 36(10), 1148–1162.

Whitcomb, L. L., Howland, J., Smallwood, D. A., Yoerger, D., & Thiel, T. (2003). A new control system for the next generation of US and UK deep submergence oceanographic ROVs. In Proceedings of the 1st IFAC Workshop on Guidance and Control of Underwater Vehicles (pp. 137–142). Cardiff, Wales.

Whitcomb, L. L., & Yoerger, D. R. (1996). Preliminary experiments in the model-based dynamic control of marine thrusters. In Proceedings of the International Conference on Robotics and Automation (ICRA). Minneapolis, MN: IEEE.

Whitcomb, L. L., Yoerger, D. R., & Singh, H. (1999). Advances in Doppler-based navigation of underwater robotic vehicles. In Proceedings of the International Conference on Robotics and Automation (ICRA) (vol. 1, pp. 399–406). Detroit, MI: IEEE.

Yoerger, D. R., & Slotine, J. E. (1991). Adaptive sliding control of an experimental underwater vehicle. In Proceedings of the International Conference on Robotics and Automation (ICRA) (pp. 2746–2751). Sacramento, CA: IEEE.