UC Berkeley
Research Reports

Title
Design and Error Analysis of Accelerometer-Based Inertial Navigation Systems

Permalink
https://escholarship.org/uc/item/9mc9q9p5

Authors
Tan, Chin-Woo
Park, Sungsu

Publication Date
2002-06-01

Design and Error Analysis of Accelerometer-Based Inertial Navigation Systems

Chin-Woo Tan
Sungsu Park

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the State of California. This report does not constitute a standard, specification, or regulation.

CALIFORNIA PARTNERS FOR ADVANCED TRANSIT AND HIGHWAYS

Design and Error Analysis of Accelerometer-Based Inertial Navigation Systems

Chin-Woo Tan and Sungsu Park
PATH, University of California, Berkeley
E-mails: tan@robotics.eecs.berkeley.edu, spark@me.berkeley.edu

Abstract

We examine the feasibility of designing an accelerometer-based (or gyroscope-free) inertial navigation system that uses only accelerometers to compute the linear and angular motions of a rigid body. The accelerometer output equation is derived to relate the linear and angular motions of a rigid body relative to a fixed inertial frame. A sufficient condition is given to determine if a configuration of accelerometers is feasible. If the condition is satisfied, the angular and linear motions can be computed separately using two decoupled equations of an input-output dynamical system; a state equation for angular velocity and an output equation for linear acceleration. This simple computation scheme is derived from the corresponding dynamical system equations for a special cube configuration for which the angular acceleration is expressed as a linear combination of the accelerometer outputs. The effects of accelerometer location and orientation errors are analysed. Algorithms that identify and compensate these errors are developed.

Keywords gyroscope-free, configuration of accelerometers, feasibility, input-output dynamical system realisation, error sensitivity analysis.

Nomenclature

$\hat{F}$: force per unit mass (specific force) on a rigid body

g: gravitational force

IF: inertial frame $\{O_I; e_1, e_2, e_3\}$

MF: moving frame $\{O; f_1, f_2, f_3\}$

F: rotation matrix that transforms a coordinate in the MF to its coordinate in the IF

ω: angular rate of moving (or body) frame with respect to inertial frame, with coordinates in the moving frame

Ω: skew-symmetric matrix corresponding to $\omega \in R^3$

u: location of an accelerometer in the moving frame

θ: orientation of an accelerometer in the moving frame

$J \in R^{N \times 6}$: configuration matrix corresponding to a configuration of N accelerometers

$Q \in R^{6 \times N}$: left inverse of configuration matrix J

1 Introduction

Inertial navigation systems (INS) are used in many diverse applications, including automobiles, aviation aircrafts, yachts, and submarine fleet. Recently, robotics and image stabilisation are among the emerging areas that use micro-machined navigation systems. Most inertial navigation systems use *accelerometers* to sense linear accelerations and *gyroscopes* to sense angular velocity. In this paper, we examine the feasibility of designing an accelerometer-based or gyroscope-free INS that uses *only* accelerometers to compute the linear and angular motions of a rigid body. In theory, a minimum of six accelerometers is required for a complete description of rigid body motion. The key to a solution of the feasibility problem is the choice of *location* and *orientation* of the accelerometers. It will be shown that for some "nice" (or feasible) configurations of accelerometers, the angular and linear motions can be computed separately using two decoupled equations associated with an input-output (I/O) dynamical system.

The main motivation for developing a gyroscope-free navigation system is that low-cost gyroscopes lack the accuracy needed for precise navigation applications. The accuracy is usually characterised by the resolution, or the minimum detectable angular rate (in degree/second), of the gyroscope. In general, gyroscopes can be classified into three different categories based on their performace: inertial-grade (10^{-3} deg/sec), tactical-grade (10^{-2} deg/sec), and rate-grade devices (1 deg/sec). Inertial-grade gyroscopes, such as a laser ring gyroscope that measures the earth's rotation, cost in the order of tens of thousands of US dollars. Micro-machined silicon rate-grade gyroscopes, such as those for automotive applications, cost about US$10-20 with a resolution of 0.5 deg/sec in a bandwidth of 50 Hz ([1]). It has also been reported in ([2], [3]) that due to challenges associated with micro-miniaturisation of gyroscopes, inexpensive batch-processed gyroscopes cannot achieve the required levels of precision in the near future. A precise micro-machined accelerometer, on the other hand, is more affordable. This type of accelerometer has become one of the most notable applications of poly-silicon surface micro-machining ([4]). Due to recent breakthrough developments in micro-machining technology, the costs of micro-machined accelerometers are decreasing while the accuracy is being improved ([1], [5]). Also, less fundamental physical constraints inhibit the precision of a micro-machined accelerometer than the precision of a micro-machined gyroscope. So there is a promising market for accelerometer-based inertial navigation

systems.

Semiconductor micro-machined accelerometers are used as basic sensing devices for an accelerometer-based INS ([6], [7], [8]). Using measurements from accelerometers strategically distributed on a rigid body, an INS algorithm computes the linear displacement and rotation of the body relative to a fixed inertial reference frame. In Section 2, the accelerometer output (measurement) equation is derived, and the acceleration of a body is described by the body linear acceleration and a rotation matrix. In Section 3, a sufficient condition is used to determine if a configuration of accelerometers is feasible. Feasibility means that with specified accelerometer locations and orientations, one can compute the linear and angular motions using the accelerometer outputs. A feasible cube configuration of six accelerometers is then considered, and based on this configuration, a basic INS algorithm is developed. The cube configuration was first considered in [6]. In our approach, the motion equations are formulated as an I/O dynamical system in such way that the algorithm involves solving a state equation for angular velocity and computing an output equation for linear acceleration. Our I/O dynamical system formulation results in a general INS algorithm, which is computationally as simple as the basic INS algorithm, for all other feasible configurations of six accelerometers. All other configurations are considered as distorted configurations of the special cube configuration. So in Section IV, we analyse the effects of location and orientation errors on accelerometer output. We also discuss how these misalignment errors can be identified and compensated.

The main result is that as long as the feasibility condition stated in Section 3 is satisfied, we can compute the linear and angular motions using a general INS algorithm that has the same computational *simplicity* as the basic INS algorithm for the cube configuration. The feasibility condition checks the invertibility of a 6×6 matrix, hence will be satisfied "almost surely". So the beauty of the main result is that except of a "measure zero" set of configurations of six accelerometers, any other sets of six accelerometers can be placed and oriented *arbitrarily* on a rigid body, and the six accelerometer measurements are used to compute the linear and angular motions using our simple algorithm. These are discussed in Section 5. We illustrate our results with a simulation example in Section 6. Some concluding remarks are discussed in Section 7.

2   Accelerometer Output Equation

A single-axis accelerometer mounted on a rigid body is a device with one input and one output that measures the force, *per unit mass*, acting on the body along a specific sensing direction. This force vector is also known as the *specific force*. The vector sum of this force $\hat{F}$ and the gravity g, per unit mass, is the inertial acceleration of the body. That is, $\hat{F} + g = ma$, $m = 1$. So the specific force is $\hat{F} = a - g$. The device is designed to measure this force quantity projected along the sensing axis (or orientation) of the accelerometer. Let θ_I and r_I be the sensing axis and the coordinate of the centre of an accelerometer relative to a fixed inertial frame. The vector r_I is time-dependent, and the vector θ_I varies when the body rotates. The output of the accelerometer is given by:

$$A = \langle a - g, \theta_I \rangle = \langle \ddot{r}_I - g, \theta_I \rangle \tag{2.1}$$

where $\langle \cdot, \cdot \rangle$ is the inner product, $g = -|g|e_3$ and e_3 is the column vector $[0\ 0\ 1]^T$ expressed in a local tangent plane coordinate system. One can also model a single-axis accelerometer as a spring-mass system to derive the output equation (2.1) (see [9]).

Next we derive the equations that relate the body acceleration, $a = \ddot{r}_I$, to the linear and angular motions of the body. It will be shown that the body acceleration can be expressed in terms of the linear acceleration at a point and a rotation matrix. In Figure 1, let O_I be the centre of a *fixed inertial frame* (IF) with a right-handed orthonormal basis $\{e_1, e_2, e_3\}$ (so the cross products $e_1 \times e_2 = e_3$, $e_2 \times e_3 = e_1$, $e_3 \times e_1 = e_2$ hold). Typically, the basis is the standard basis in R^3 . The *moving frame* (also called body frame) (MF) associated with the rigid body is a coordinate system $\{O; f_1, f_2, f_3\}$, where O is the centre and $\{f_1, f_2, f_3\}$ is a right-handed orthonormal basis.

We shall see that the body motion can be described by the linear acceleration, R_I, of the centre O and the rotation of the moving frame relative to the inertial frame. The rotation is given by a 3×3 matrix F as shown in Figure 1. This is a coordinate transformation that expresses the coordinate in the moving frame in terms of its coordinate in the inertial frame. That is, $f_k = Fe_k$, $k = 1, 2, 3$. The matrix F is orthogonal with $F^T F = I$ (identity matrix). It follows that $\det(F) = \pm 1$. Since the bases are right-handed, we have $\det(F) = 1$. The matrix F is called a *rotation matrix*. Next consider a point M on the body as shown in Figure 1. We have the following equation
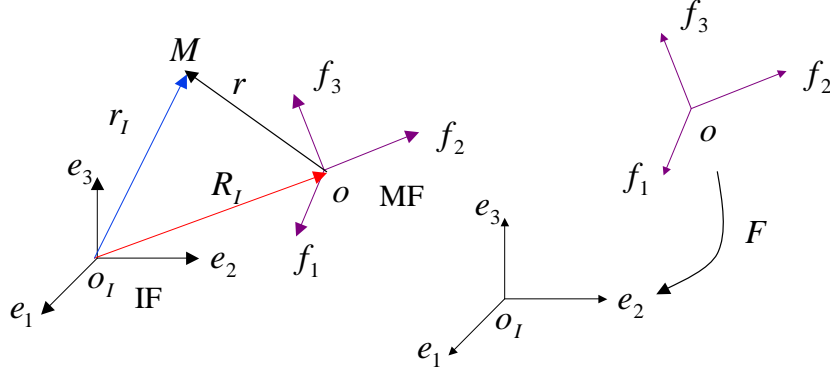
Figure 1: Rigid body motion relative to an inertial frame

relating the displacement vectors shown in Figure 1:

$$r_I = R_I + r \tag{2.2}$$

All the vectors in (2.2) are expressed in the inertial frame. Since the body is *rigid*, the magnitude of r remains constant. However, the direction of r changes when the body rotates. Let u be the coordinate of the vector r in the moving frame. So u is *time-independent* and we have $r = Fu$. Substituting this into (2.2) gives:

$$r_I = R_I + Fu \tag{2.3}$$

Thus the motion at point M can be described by the translation R_I and the rotation F relative to the inertial frame. The acceleration at M (relative to IF) is:

$$\ddot{r}_I = \ddot{R}_I + \ddot{F}u \tag{2.4}$$

Time differentiation of $F^T F = I$ gives $F^T \dot{F} + \dot{F}^T F = 0$. Define the matrix $\Omega := F^T \dot{F}$, so Ω is *skew-symmetric* with $\Omega^T = -\Omega$. The orthogonality of F implies:

$$\dot{F} = F\Omega \tag{2.5}$$

Time differentiation of (2.5) gives $\ddot{F} = F(\dot{\Omega} + \Omega^2)$. So from (2.4) the acceleration at M can also be expressed as:

$$\ddot{r}_I = \ddot{R}_I + F(\dot{\Omega} + \Omega^2)u \tag{2.6}$$

Next consider an accelerometer mounted on the rigid body, with its centre at M. Let θ be the orientation of the sensor in the moving frame. So θ is

time-independent and $\theta_I = F\theta$. Using (2.1) and (2.6) the accelerometer output is:

$$
\begin{aligned}
A(u,\theta) &= \left\langle (\ddot{R}_I - g) + F(\dot{\Omega} + \Omega^2)u, F\theta \right\rangle \\
&= \left\langle F^T(\ddot{R}_I - g) + (\dot{\Omega} + \Omega^2)u, \theta \right\rangle \qquad (2.7) \\
&= \langle L, \theta \rangle + \langle Gu, \theta \rangle
\end{aligned}
$$

where $L := F^T(\ddot{R}_I - g)$ and $G := \dot{\Omega} + \Omega^2$. The term $\langle L, \theta \rangle$ computes the *linear* acceleration (along θ). L is the specific force at the center O expressed in the body frame. The other term $\langle Gu, \theta \rangle$ computes the *angular* acceleration consists of the tangential (skew-symmetric Ω) and centripetal (symmetric Ω^2) accelerations. The authors in ([6], [7]) have obtained a similar accelerometer output equation, but that equation does not consider the transformation of coordinates between the inertial frame and moving frame. So their rotation matrix is always the identity.

It is known that every 3×3 skew-symmetric matrix Ω has a *unique cross-product* representation. So there exists a unique $\omega \in R^3$ such that $\Omega a = \omega \times a$ for all $a \in R^3$. We use $\Omega \leftrightarrow \omega$ or $\Omega = S(\omega)$ to denote this equivalence. The vector $\omega(t)$ is the instantaneous angular velocity of the rotation described by the matrix $F(t)$. Geometrically, a 3×3 skew-symmetric matrix corresponds to an axis of rotation (via the mapping $\Omega \leftrightarrow \omega$). It can be shown ([10]) that a rotation about an axis ω (relative to an inertial frame) by a specified angle θ is equivalent to the matrix exponential $e^{\Omega\theta}$. So for a rotation about a *fixed* axis, the matrix differential equation (2.5) has a simple closed-form solution. Indeed, it is easy to check that if a rigid body rotates about a fixed axis ω_0 with an angular speed $\dot{\theta}(t)$ and $\Omega_0 \leftrightarrow \omega_0$, then $F(t) = F(t_0)e^{\theta(t)\Omega_0}$ solves $\dot{F} = F\Omega$, $t \geq t_0$. For angular motions other than a simple rotation about a fixed axis, (2.5) can be solved numerically using the quaternion equations ([11]).

3 Feasibility of Gyroscope-Free INS Design

A gyroscope-free inertial navigation system (GF-INS) is a system that uses *only* accelerometer measurements to compute the linear displacement and angular motion of a rigid body. To achieve this, the accelerometers need to be strategically distributed on the body. A set of accelerometers whose outputs

are sufficient to compute the linear and angular motions of a moving body is called a *feasible* configuration of accelerometers. It is obvious that not all configurations are feasible. For example, those with all the accelerometers having the same orientation are not feasible. However, it turns out that "almost all" configurations are feasible. We first present a sufficient condition to determine if a configuration is feasible.

3.1   Feasible Configurations of Accelerometers

Consider N accelerometers mounted at locatons $u_1, \cdots, u_N$ with orientations $\theta_1, \cdots, \theta_N$, respectively. The pair (u_i, θ_i) is expressed in the moving frame and is thus time-independent.

Definition 1: A configuration $C = \{(u_i, \theta_i); 1 \leq i \leq N\}$ of N accelerometers is *feasible* if the outputs $A_i = A(u_i, \theta_i)$ and the initial conditions $\{R_I(t_0), \dot{R}_I(t_0), F(t_0), \dot{F}(t_0)\}$ are sufficient to determine the linear and angular motions, $R_I(t)$ and $F(t)$, for $t \geq t_0$.

Define two $3 \times N$ matrices $J_1 := [u_1 \times \theta_1, \cdots, u_N \times \theta_N]$ and $J_2 := [\theta_1, \cdots, \theta_N]$. Let $J = [J_1^T \quad J_2^T]$, a $N \times 6$ matrix. We call J the *configuration matrix*. Using the output equation (2.7) and $\Omega \leftrightarrow \omega$, we get:

$$
\begin{aligned}
A_i &= A(u_i, \theta_i) = \left\langle L + (\dot{\Omega} + \Omega^2)u_i, \theta_i \right\rangle \\
&= \theta_i^T L + (u_i \times \theta_i)^T \dot{\omega} + \theta_i^T \Omega^2 u_i \\
&= \left[\ (u_i \times \theta_i)^T \ \ \theta_i^T \ \right] \left[\begin{array}{c} \dot{\omega} \\ L \end{array} \right] + \theta_i^T \Omega^2 u_i
\end{aligned}
$$

So we have:

$$
A = \left[\begin{array}{c} A_1 \\ \vdots \\ A_N \end{array} \right] = J \left[\begin{array}{c} \dot{\omega} \\ L \end{array} \right] + \left[\begin{array}{c} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_N^T \Omega^2 u_N \end{array} \right] \tag{3.1}
$$

If J has a *left inverse* $Q \in R^{6 \times N}$, then (3.1) becomes:

$$
\left[\begin{array}{c} \dot{\omega} \\ L \end{array} \right] = QA - Q \left[\begin{array}{c} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_N^T \Omega^2 u_N \end{array} \right] \tag{3.2}
$$

The matrix differential equation $\dot{F} = F\Omega$ is embedded in (3.2) to relate F and Ω, so we can view (3.2) as an *input-output (I/O) dynamical system* where

8

the input is A, the state equation consists of the top three rows of (3.2):

$$\dot{\omega} = f(\omega, A); \qquad \dot{F} = F\Omega, \qquad \Omega \leftrightarrow \omega \qquad (3.3)$$

and the output equation consists of the bottom three rows of (3.2):

$$L = F^T(\ddot{R}_I - g) = g(\omega, A) \qquad (3.4)$$

Note that the angular and linear motions are computed separately using two *decoupled* equations: the state equation for solving the angular motion, and the algebraic output equation for computing the linear motion. We summarise this I/O representation in a proposition.

Proposition 1 *(A Sufficient Condition for Feasibility): If the $N \times 6$ matrix J has a left inverse, then the configuration $C = \{(u_i, \theta_i); 1 \leq i \leq N\}$ is feasible. Also if $N < 6$, then the configuration is not feasible.* $\qquad \square$

So a feasible configuration must have at least six accelerometers. For configurations with six accelerometers, the feasibility condition in Proposition 1 checks if the configuration matrix J is invertible. This condition is satisfied "almost surely" since the set of singular 6×6 matrices form a "measure zero" set in $R^{6 \times 6}$. For those invertible J matrices, it is nonetheless computationally inefficient to use (3.3)-(3.4) to compute the angular and linear motions. The main contribution of our work is the development of a simple, computationally efficient, algorithm that computes angular and linear motions for all feasible configurations. The key idea is to obtain an I/O realisation of the dynamical system (3.2). From now on, we will only consider configurations of $N = 6$ accelerometers, but all the results are valid for $N > 6$. The first step of our algorithm development is to consider a special cube configuration.

3.2   A Cube Configuration

In general, the state equation $\dot{\omega} = f(\omega, A)$ in (3.3) does not have a closed form solution. A numerical solution for $\omega(t)$ depends on its value calculated from previous time steps because of the implicit dependence of $f(\omega, A)$ on ω. This results in an accumulation of numerical errors in solving for $\omega(t)$. We consider a special configuration where $\dot{\omega}$ is a linear combination of the accelerometer outputs. So $\omega(t)$ has a closed form solution and this makes numerical integration much easier. (For example, if $\dot{\omega}(t) = A_1(t)$, forward

Euler approximation of the integral gives $\omega(t_i) = \omega(t_0) + \Delta t \cdot \sum_{j=0}^{i-1} A_1(t_j)$.)
Consider the cube-shaped GF-INS first examined in [6] (however, their accelerometer output equation is incorrect). The design has one accelerometer at the centre of each face of a cube of length $2l$. The sensing direction of each accelerometer is along the respective cube face diagonal, in such a way that these diagonals form a regular tetrahedron (see Figure 2).
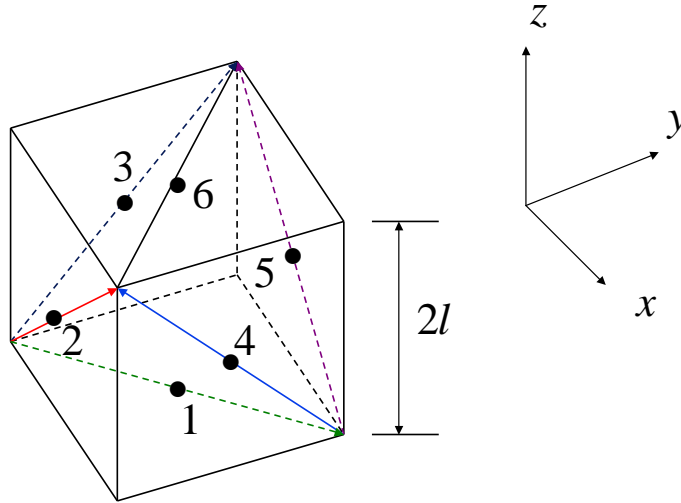

Figure 2: A six-accelerometer cube configuration C_{cube}

Let the origin of the MF be the centre of the cube. The locations and orientations of the six accelerometers (in the MF coordinate) are:

$$U = [u_1 \cdots u_6] = l \begin{bmatrix} 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$J_2 = [\theta_1 \cdots \theta_6] = \frac{l}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 & -1 & -1 \\ 1 & 0 & 1 & -1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

It is easy to check that:

$$\begin{aligned} J_1 &= [u_1 \times \theta_1 \cdots u_6 \times \theta_6] \\ &= \frac{l}{\sqrt{2}} \begin{bmatrix} 1 & -1 & 0 & 0 & 1 & -1 \\ -1 & 0 & 1 & -1 & 0 & -1 \\ 0 & 1 & -1 & -1 & 1 & 0 \end{bmatrix} \end{aligned}$$

10

Lemma 1 *For $i, j \in \{1, 2, \cdots, 6\}$, $i \neq j$,*

$$\frac{1}{l^2} \langle u_i \times \theta_i, u_j \times \theta_j \rangle + \langle \theta_i, \theta_j \rangle = 0 \quad .$$

And for each $1 \leq i \leq 6$, we have $||u_i \times \theta_i||^2 = l^2$ and $||\theta_i||^2 = 1$.

Proof: The assertions follow immediately from straight-forward calculations using the J_1 and J_2 matrices. □

Proposition 2 *The inverse of $J = [J_1^T \; J_2^T]$ is:*

$$Q = J^{-1} = \frac{1}{2l^2} \begin{bmatrix} J_1 \\ l^2 J_2 \end{bmatrix} \quad . \tag{3.5}$$

Proof: The proof is in the Appendix. □

By Proposition 1, the configuration $C_{cube} = \{(u_i, \theta_i); 1 \leq i \leq 6\}$ is feasible. We can use (3.2) and (3.5) to obtain the corresponding state and output equations for C_{cube}.

Lemma 2 *For the cube configuration, the second term on the right-hand side of (3.2) is:*

$$\begin{bmatrix} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_N^T \Omega^2 u_6 \end{bmatrix} = -l J_2^T \begin{bmatrix} \omega_2 \omega_3 \\ \omega_1 \omega_3 \\ \omega_1 \omega_2 \end{bmatrix} \quad , \tag{3.6}$$

where $\Omega \leftrightarrow \omega = [\omega_1 \; \omega_2 \; \omega_3]^T$.

Proof: Let $\hat{\omega} = [\omega_2 \omega_3 \; \omega_1 \omega_3 \; \omega_1 \omega_2]^T$. It is easy to check that:

$$\theta_1^T \Omega^2 u_1 = \frac{l}{2}(-\omega_1 \omega_3 \; - \; \omega_2 \omega_3) = -l \theta_1^T \hat{\omega}$$

Indeed, we have $\theta_i^T \Omega^2 u_i = -l \theta_i^T \hat{\omega}$, for each $1 \leq i \leq 6$. Therefore (3.6) holds since $J_2 = [\theta_1, \cdots, \theta_6]$. □

Proposition 3 *The decoupled state and output equations (3.3)-(3.4) for the cube configuration are:*

$$\begin{bmatrix} \dot{\omega}_1 \\ \dot{\omega}_2 \\ \dot{\omega}_3 \end{bmatrix} = \frac{1}{2l^2} J_1 A = \frac{1}{2\sqrt{2}l} \begin{bmatrix} A_1 - A_2 + A_5 - A_6 \\ -A_1 + A_3 - A_4 - A_6 \\ A_2 - A_3 - A_4 + A_5 \end{bmatrix} \tag{3.7}$$

11

$$L = \frac{1}{2}J_2 A + l \begin{bmatrix} \omega_2\omega_3 \\ \omega_1\omega_3 \\ \omega_1\omega_2 \end{bmatrix}$$

$$= \frac{1}{2\sqrt{2}} \begin{bmatrix} A_1 + A_2 - A_5 - A_6 \\ A_1 + A_3 - A_4 + A_6 \\ A_2 + A_3 + A_4 + A_5 \end{bmatrix} + l \begin{bmatrix} \omega_2\omega_3 \\ \omega_1\omega_3 \\ \omega_1\omega_2 \end{bmatrix} \qquad (3.8)$$

Proof: The proof is in the Appendix. $\square$

From (3.7) we see that the angular acceleration is independent of the angular rate ω. Furthermore, it is a *linear combination* of the accelerometer outputs, thus making numerical integration much easier. This is expected from the choice of the locations and orientations. Recall that in (2.7) the term $\langle Gu, \theta \rangle$, where $G = \dot{\Omega} + \Omega^2$, computes the angular acceleration consisting of the tangential (skew-symmetric $\dot{\Omega}$) and centripetal (symmetric Ω^2) accelerations. The sensors at u_3 and u_4 do not sense x-axis motion, so $\dot{\omega}_1$ does not depend on A_3 or A_4. The asymmetry of $\{\theta_1, \theta_6\}$ implies that the symmetric components of the angular motions measured at these two locations cancel each other. Similarly for the angular motions measured at u_2 and u_5. So $\dot{\omega}_1$, the asymmetric component of the angular motion, is a linear combination of the outputs A_1, A_2, A_5 and A_6.

3.3   Basic Algorithm

For the cube configuration C_{cube}, we have a simple *Basic INS Algorithm* for computing $\omega(t)$ (or $F(t)$) and $R_I(t)$.

Basic INS Algorithm:
Step 1: Integrate $\dot{\omega} = f(A)$ in (3.7) to obtain $\omega(t)$. Use the correspondence $\Omega \leftrightarrow \omega$ to obtain Ω.
Step 2: Solve the matrix differential equation $\dot{F} = F\Omega$ (numerically) to obtain $F(t)$. The numerical solution $F(t_i)$ must be a rotation matrix. One can solve this using the quaternion equations ([11]).
Step 3: Compute $L(t)$ using the algebraic output equation (3.8). The linear displacement is computed by integration:

$$R_I(t) = R_I(t_0) + \dot{R}_I(t_0)(t - t_0) \int_{t_0}^{t} \left[\int_{t_0}^{s} F(\tau)(L(\tau) + g)d\tau \right] ds$$

3.4 Gyroscope-Free Cube INS Versus Conventional INS

A conventional INS has three rate gyroscopes that measure the angular rates ω (yaw, roll, pitch), and three accelerometers that measure the specific forces $\hat{F}$ along the three axes. Integration of ω gives the attitude (i.e. angles), so the (numerical) attitude error grows linearly with time. The position estimate involves double-integration of the specific force and using the attitude information, so the position error grows with time in an order of t^3. On the other hand, a cube GF inertial measurement unit (IMU) gives only the angular acceleration, which is a linear combination of the accelerometer outputs. So an additional integration is needed to obtain the angular rate. This is illustrated in Figure 3. The error growth rates are thus an order higher than those of a conventional INS. Error growth rates: (i) conventional INS: attitude $\sim t$, position $\sim t^3$, (ii) GF-INS: attitude $\sim t^2$, position $\sim t^4$.
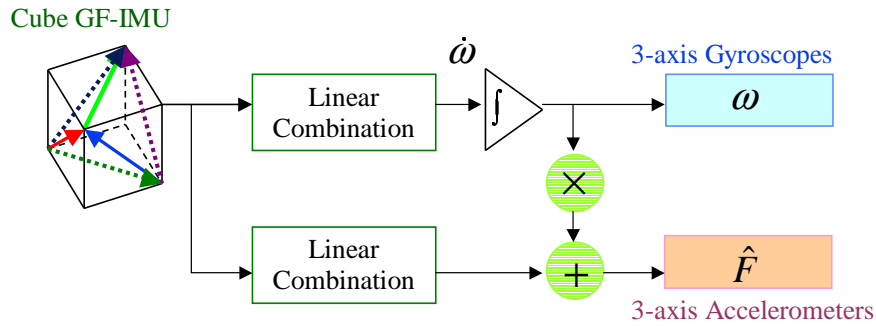

Figure 3: An additional integration for GF-IMU

4 Effects of Location and Orientation Errors

The simple Basic Algorithm presented in Section 3.3 assumes a special cube configuration of accelerometers. So the next logical design question is:

> If six accelerometers are *arbitrarily* placed at six locations on a rigid body with *arbitrary* orientations, can one compute the linear and angular motions using an algorithm that has the *same computational simplicity* as the Basic INS Algorithm for the cube configuration C_{cube}?

13

The answer to this question is "yes" as long as the configuration matrix J is invertible. The first step in the development of such an algorithm is to consider any configuration of six accelerometers as a distorted configuration of the cube configuration C_{cube}. That is, it is C_{cube} with location and orientation errors. We first derive the accelerometer output (measurement) error caused by location and orientation errors.

4.1   Accelerometer Output Error and Sensitivity

There are several error sources that cause an accelerometer output to deviate from its correct value. They are configuration (or misalignment) errors and the accelerometer errors embedded in the device itself. The configuration errors of an accelerometer are the *location* and *orientation* errors of the accelerometer. The error sources of a MEMS accelerometer are: *scale factor error*, *bias*, and *noise*. All of these errors need to be identified and calibrated either on-line or off-line as accurately as possible. In this paper, we will consider only the configuration errors and propose schemes to identity them. The identification schemes are presented in Sections 4.2 and 4.3. Our schemes can easily be extended to identify scale factor and bias.

Suppose one would ideally like to place an accelerometer at location u with orientation θ. The expected (or ideal) accelerometer measurement output is given by (2.7). However, due to misalignment (or calibration) errors, the sensor location and orientation are actually u_r and θ_r. So the *actual* accelerometer output is given by:

$$A_r(u_r, \theta_r) = \langle L, \theta_r \rangle + \langle Gu_r, \theta_r \rangle \tag{4.1}$$

By using (2.7), the change in the output, $A_e = A_r - A$, is:

$$
\begin{aligned}
A_e(u_r, \theta_r; u, \theta) &= \begin{bmatrix} (u \times \theta_e + u_e \times \theta_e)^T & \theta_e^T \end{bmatrix} \begin{bmatrix} \dot{\omega} \\ L \end{bmatrix} \\
&\quad + \begin{bmatrix} \theta_e^T \Omega^2 u + \theta_r^T \Omega^2 u_e \end{bmatrix}
\end{aligned}
\tag{4.2}
$$

where $u_e = u_r - u$ and $\theta_e = \theta_r - \theta$ are the configuration (or misalignment) errors. Note that the output error depends on both the linear motion (described by the specific force L) and the angular rate ω.

By taking partial derivatives of the error equation (4.2), the sensitivity with respect to location and orientation errors can easily be calculated. It is

14

not difficult to check that:

$$\frac{\partial A_e}{\partial \theta_e} \big|_{(u,\theta)} = (\dot{\omega} \times u)^T + L^T + u^T \Omega^2 \tag{4.3}$$

$$\frac{\partial A_e}{\partial u_e} \big|_{(u,\theta)} = -(\dot{\omega} \times \theta) + \theta^T \Omega^2 \tag{4.4}$$

Note that these error sensitivity terms are motion-dependent. Also from (4.4) the sensitivity relative to location error depends *only* on the angular motion. This is intuitively obvious because if there is only linear motion and there is no orientation error, then the accelerometer output will be the same regardless of where the accelerometer is mounted on the body.

Consider the cube GF-INS. If there is no misalignment error, the accelerometer outputs are $A = [A_1, \cdots, A_6]^T$. We can then use the Basic Algorithm to compute the correct angular rate $\omega(t)$. However, if there are misalignment errors u_{ei} and θ_{ei}, the *actual* output of accelerometer i is A_{ri}. If the errors are not identified and compensated, the angular rate computed by the Basic Algorithm using the distorted output A_r will be $\tilde{\omega}(t)$. By using the chain rule and (4.3)-(4.4), it follows that:

$$\frac{\partial \tilde{\omega}}{\partial \theta_{ei}} = \frac{\partial \tilde{\omega}}{\partial A_{ei}} \cdot \frac{\partial A_{ei}}{\partial \theta_{ei}} = \frac{1}{2l^2} (u_i \times \theta_i) \left[(\dot{\omega} \times u_i)^T + L^T + u_i^T \Omega^2 \right]$$

$$\frac{\partial \tilde{\omega}}{\partial u_{ei}} = \frac{\partial \tilde{\omega}}{\partial A_{ei}} \cdot \frac{\partial A_{ei}}{\partial u_{ei}} = \frac{1}{2l^2} (u_i \times \theta_i) \left[\theta_i^T \Omega^2 - \dot{\omega} \times \theta_i \right]$$

So if the misalignment errors are not identified and compensated, the incorrect solution $\tilde{\omega}(\cdot)$ depends on the correct motion and is also sensitive to the cube length. We next discuss how we can identify location and orientation errors.

4.2 Identification of Orientation Error

Suppose there is an orientation error so that the actual location and orientation are u and $\theta_r = \theta + \theta_e$, respectively. This error can be identified by examining the resultant gravity effect on the accelerometer output. When the body is *stationary*, the only force acting on the accelerometer is the gravity. This gravity effect can be obtained by adding its components along three independent axes. So an identification scheme is to *flip* the rigid body along

15

three independent axes (e.g. the standard basis axes $\{e_1, e_2, e_3\}$) and record the corresponding accelerometer outputs. This implies $\ddot{R}_I = 0$, $\Omega = 0$, and F is a constant corresponding to the specific flip rotation. So we have $G = 0$, $L = -F^T g$, and the accelerometer output equation is reduced to:

$$A_r(u, \theta_r) = -\left\langle F^T g, \theta_r \right\rangle = |g| \left\langle F^T e_3, \theta_r \right\rangle \tag{4.5}$$

Let $F^{(1)}$, $F^{(2)}$ and $F^{(3)}$ be the flip rotation of the body by $+\pi/2$ (anticlockwise flip by $\pi/2$) about the axes e_1, e_2 and e_3, respectively. The corresponding (steady-state) accelerometer outputs are $A_r^{(1)}$, $A_r^{(2)}$ and $A_r^{(3)}$. It is easy to check that $F^{(1)}e_3 = e_2$, $F^{(2)}e_3 = -e_1$ and $F^{(3)}e_3 = e_2$. So from (4.5) we obtain:

$$A_r^{(1)} = |g| \left\langle e_2, \theta_r \right\rangle, \; A_r^{(2)} = -|g| \left\langle e_1, \theta_r \right\rangle, \; A_r^{(3)} = |g| \left\langle e_3, \theta_r \right\rangle$$

Therefore, we can estimate the actual orientation $\theta_r = \sum_i \left\langle e_i, \theta_r \right\rangle e_i$ as:

$$\theta_r = \frac{1}{|g|} \left(-A_r^{(2)} e_1 + A_r^{(1)} e_2 + A_r^{(3)} e_3 \right) \tag{4.6}$$

4.3   Identification of Location Error

If there are also location errors, the acceleromter locations and orientations for the cube GF-INS are u_{ri} and θ_{ri}, $1 \le i \le 6$. The location errors cannot be identified unless the body rotates (see (4.4)). So to estimate them, we consider three cases of constant speed (ω_0) rotational motion of different surfaces of the (cube) body frame with respect to the inertial frame (or a navigation frame). In all three cases, the rotation is parallel to the gravity direction. The six surfaces of the cube are numbered in the same way as shown in Figure 2. That is, accelerometer i is mounted on surface i. For example, surface 2 is the $x - z$ plane of the cube body frame. The three cases are: the rotation axis lies in (1) $x - z$ plane, (2) $y - z$ plane, and (3) $x - y$ plane of the body frame. The first case is shown in Figure 4. As shown in the figure, the cube is placed on a wedge of angle β. The combined cube-wedge body is then placed on a rate table for performing the rotations.

For a constant speed rotation, $\ddot{R}_I = 0$ and $\dot{\Omega} = 0$. So from (4.1), the steady-state outputs of accelerometer i corresponding to the three cases ($j = 1, 2, 3$) of constant speed rotation are:

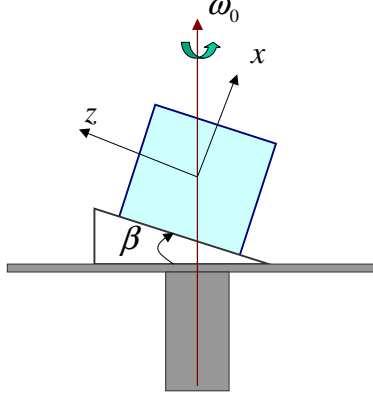$$A_{ri}^{(j)} = \theta_{ri}^T \hat{F}^{(j)} + \theta_{ri}^T (\Omega^{(j)})^2 u_{ri} , \; 1 \le i \le 6, \; 1 \le j \le 3 \tag{4.7}$$

Figure 4: Identification of location error

where $\Omega^{(j)} \leftrightarrow \omega^{(j)}$ and

$$\omega^{(1)} = \omega_0 \left[\cos\beta \quad 0 \quad \sin\beta\right]^T \, , \ \hat{F}^{(1)} = |g| \left[\cos\beta \quad 0 \quad \sin\beta\right]^T$$

$$\omega^{(2)} = \omega_0 \left[0 \quad \cos\beta \quad \sin\beta\right]^T \, , \ \hat{F}^{(2)} = |g| \left[0 \quad \cos\beta \quad \sin\beta\right]^T$$

$$\omega^{(3)} = \omega_0 \left[\cos\beta \quad \sin\beta \quad 0\right]^T \, , \ \hat{F}^{(3)} = |g| \left[\cos\beta \quad \sin\beta \quad 0\right]^T$$

Here ω_0 is a constant angular rate and $\omega^{(j)}$ is the axis of rotation in the body frame for case (j). Assuming that the orientation error θ_{ei} has been identified (see Section 4.2), the actual location of accelerometer j is estimated by:

$$u_{ri} = \left[\begin{array}{c} \theta_{ri}^T(\Omega^{(1)})^2 \\ \theta_{ri}^T(\Omega^{(2)})^2 \\ \theta_{ri}^T(\Omega^{(3)})^2 \end{array}\right]^{-1} \left[\begin{array}{c} A_{ri}^{(1)} - \theta_{ri}^T\hat{F}^{(1)} \\ A_{ri}^{(2)} - \theta_{ri}^T\hat{F}^{(2)} \\ A_{ri}^{(3)} - \theta_{ri}^T\hat{F}^{(3)} \end{array}\right] \qquad (4.8)$$

The matrix inverse in (4.8) exists if the rotation axis is not parallel or orthogonal to the sensing direction of accelerometer j, which corresponds to $\beta = 0°, 45°, 90°$.

5   Design of Gyroscope-Free INS

The Basic Algorithm presented in Section 3.3 for the cube configuration has the simplicity that $\dot{\omega}$ is independent of ω and is a linear combination of the

accelerometer outputs. The main result of this paper is that for any configuration of six accelerometers, if the configuration matrix is invertible (i.e. the configuration is feasible), then one can compute the linear and angular motions using an algorithm that has the *same computational simplicity* as the Basic Algorithm.

Consider a set of six accelerometers located at u_{ri} with orientations θ_{ri}, $1 \leq i \leq 6$. The accelerometer outputs are A_{ri}, $1 \leq i \leq 6$. If the configuration matrix $J_r = \begin{bmatrix} J_{r1}^T & J_{r2}^T \end{bmatrix}$ is *invertible*, then the state and output equations are:

$$\begin{bmatrix} \dot{\omega} \\ L \end{bmatrix} = J_r^{-1} A_r - J_r^{-1} \begin{bmatrix} \theta_{r1}^T \Omega^2 u_{r1} \\ \vdots \\ \theta_{r6}^T \Omega^2 u_{r6} \end{bmatrix} \tag{5.1}$$

Note that the angular acceleration $\dot{\omega}$ depends on product terms $\omega_i \omega_j$, $i, j \in \{1, 2, 3\}$. So numerical integration of (5.1) is more complex than that of (3.7) for the cube configuration. To overcome this complexity problem, we design an I/O dynamical system Σ that realises the same I/O relation in (5.1). That is, for all inputs A_r, the solution of (5.1) and that of Σ are identical. The idea is to consider the configuration $C_r = \{(u_{ri}, \theta_{ri})\}$ as a distortion of the cube configuration, such that the location and orientation errors are $u_{ei} = u_{ri} - u$ and $\theta_{ei} = \theta_{ri} - \theta$, respectively. These misalignment errors can be identified using the methods discussed in Sections 4.2 and 4.3. For a gyroscope-free INS with the configuration C_r, the *actual* accelerometer outputs, corresponding to a body motion (ω, R_I), are $A_r = [A_{r1}, \cdots, A_{r6}]^T$. Our approach is to derive an equation that computes the "ideal" accelerometer outputs $A = [A_1, \cdots, A_6]^T$ that would be measured by the cube-shaped INS, so that we can use the Basic Algorithm to compute the motion (ω, R_I). By using the output error equation (4.2), it can be shown that there is a 6×6 matrix K and a 6×1 column vector $H(\omega)$ (a function of the angular rate ω) such that

$$A = K(A_r + H(\omega)) \tag{5.2}$$

So from the accelerometer measurement A_r, we can use (5.2) to obtain the measurement A that would be the accelerometer output if there were no misalignment errors. This measurement A is the input to the *Basic Algorithm* for computing the motion (ω, R_I) (see Section 3.3). Finally, a feedback system consisting of (3.7)-(3.8) for the cube configuration and the feedback path defined by (5.2) realises the I/O dynamical system (5.1). We summarise these ideas in the following theorem.

18

Theorem 1 *(Input-Output Realisation): The input-output (I/O) dynamical system defined by the state and output equations in (5.1) can be realised by the I/O system Σ depicted in Figure 5. This I/O realisation consists of the basic I/O system (3.7)-(3.8) for the cube-shaped INS, a gain K and feedback $H(\omega)$, where*

$$K = (J_r Q)^{-1} \ , \ Q = J^{-1} \text{ is given in (3.5)} \tag{5.3}$$

$$H(\omega) = J_r Q \begin{bmatrix} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_6^T \Omega^2 u_6 \end{bmatrix} - \begin{bmatrix} \theta_{r1}^T \Omega^2 u_{r1} \\ \vdots \\ \theta_{r6}^T \Omega^2 u_{r6} \end{bmatrix} \tag{5.4}$$
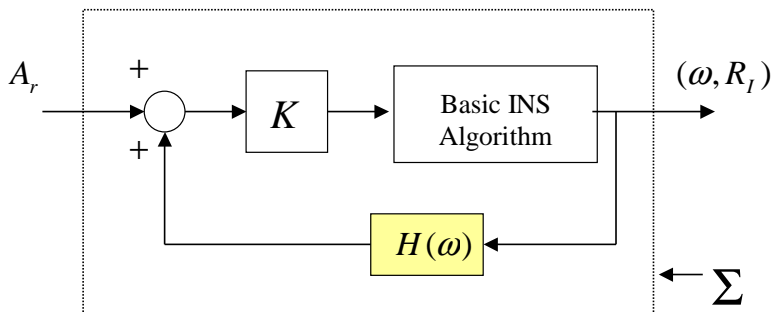

Figure 5: I/O realisation of system (5.1)

Proof: The proof is in the Appendix. □

The system Σ compensates the cube INS system (3.7)-(3.8) for the location and orientation errors. So the design is simply a *compensation scheme.* The design solves the dynamical system (5.1) and involves using only the Basic Algorithm for the cube configuration plus some trivial algebraic computations given in (5.2). Therefore, if the configuration matrix J_r is invertible, we can solve (5.1) using an algorithm that has the *same computational simplicity* as the Basic Algorithm.

We note that the I/O system Σ in Theorem 1 is a continuous-time realisation of the I/O system (5.1). For discrete-time realisation, there should be a one-time-step delay before the input to the feedback path $H(\omega)$. We have the following General INS Algorithm.

19

General INS Algorithm:

Step 1: At the kth time step t_k, use (5.2) to compute

$$A(t_k) = K(A_r(t_k) + H(\omega(t_{k-1})))$$

Step 2: Use $A(t_k)$ as the input to the Basic Algorithm to compute $\omega(t_{k+1})$ and $R_I(t_{k+1})$. $\qquad\Box$

Suppose $\omega(t)$ is the solution of the state equation in (5.1), and $\hat{\omega}(t_k)$ is a numerical solution of the General INS Algorithm. It is crucial that the discrete-time realisation converges to the continuous-time realisation as the simulation time-step Δt decreases to zero. That is, $\hat{\omega}(t_k)$ converges to $\omega(t_k)$ as $\Delta t \downarrow 0$. (We need only consider the state equation since this is where the numerical error occurs.) By using forward Euler approximation for the integral of $\dot{\omega}$, the numerical error $|\hat{\omega}(t_k) - \omega(t_k)|$ is bounded above.

Proposition 4 *Let T be the simulation time. An upper bound for the norm of the numerical error is given by:*

$$|\hat{\omega}(t_k) - \omega(t_k)| \leq (|\hat{\omega}(t_0) - \omega(t_0)| + k_1 \Delta t)\, e^{k_2 T} \qquad (5.5)$$

where $k_1, k_2 > 0$. The term k_1 depends on t_k, and the term k_2 depends the simulation time T.

Proof: The proof is in the Appendix. $\qquad\Box$

Typically, $\hat{\omega}(t_0)$ is set to the given initial angular rate $\omega(t_0)$. So the numerical error in (5.5) converges to zero as $\Delta t \downarrow 0$. This implies the discrete-time realisation converges to the continuous-time realisation. We illustrate the importance of compensating the errors (u_e, θ_e) and the dependence of the numerical solution on the time-step in an example in the next section.

6 Simulation Results: Illustration of Error Compensation and Simplicity of INS Algorithm

In this section, we illustrate the elegance and simplicity of our General INS Algorithm in a simulation example. Consider a vehicle accelerating around a circle of radius $r = 10$ metres. The angular (yaw) acceleration is $\dot{\omega} = \rho e_3$, where $\rho = 0.01$ rad/sec^2. The vehicle is initially stationary with $F(0) = I$.

This is shown in Figure 6. The cube length for the cube GF-INS is $l = 10$ cm. In this example, there is a location error at u_1 with u_{e1} equals 1% of u_1. So the actual location of the accelerometer on surface 1 is $u_{r1} = [0.01 \quad 0 \quad -1]^T$. The simulation time is 1 min. with $\Delta t = 0.01$ sec.
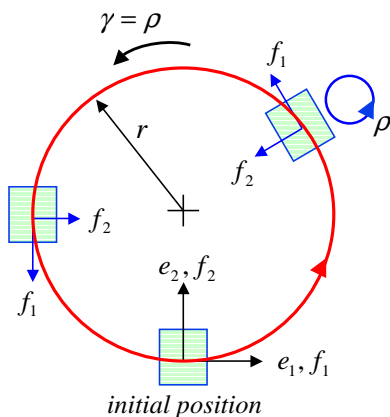

Figure 6: Simulation example

In this example, the motion (ω, R_I) is known, so we can use (4.1) to compute the actual accelerometer output A_r. If the error u_{e1} is not compensated, we use A_r as the input to the *Basic INS Algorithm* for computing the motion trajectory. This is shown in Figure 7. The computed trajectory quickly diverges away from the vehicle trajectory.

If we use the *General INS Algorithm* to compensate for the location error, the computed trajectory agrees well with the actual vehicle trajectory, except for the numerical error. Figure 8 shows the error $\hat{\omega}(t_k) - \omega(t_k)$. This error is the numerical error and its absolute value has an upper bound given in (5.5). The numerical error is sensitive to the time-step Δt. If we increase it to $\Delta t = 0.1$ sec, the numerical error increases about ten times. This is shown in Figure 9. The bound in (5.5) suggests that the error is bounded by an "exponential envelope" which grows "linearly" with Δt.
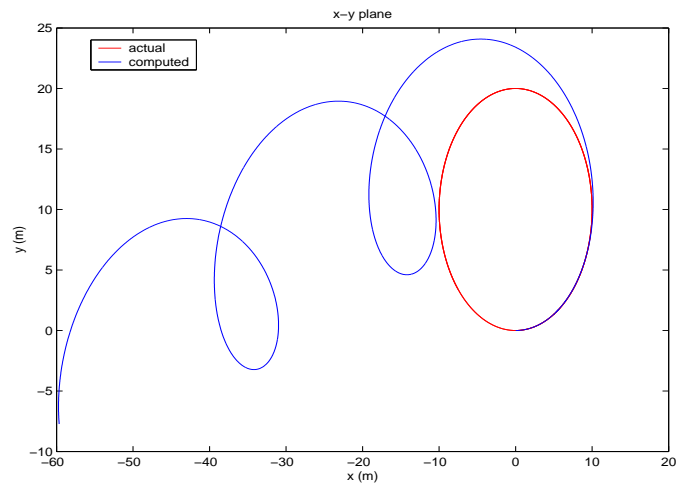
21

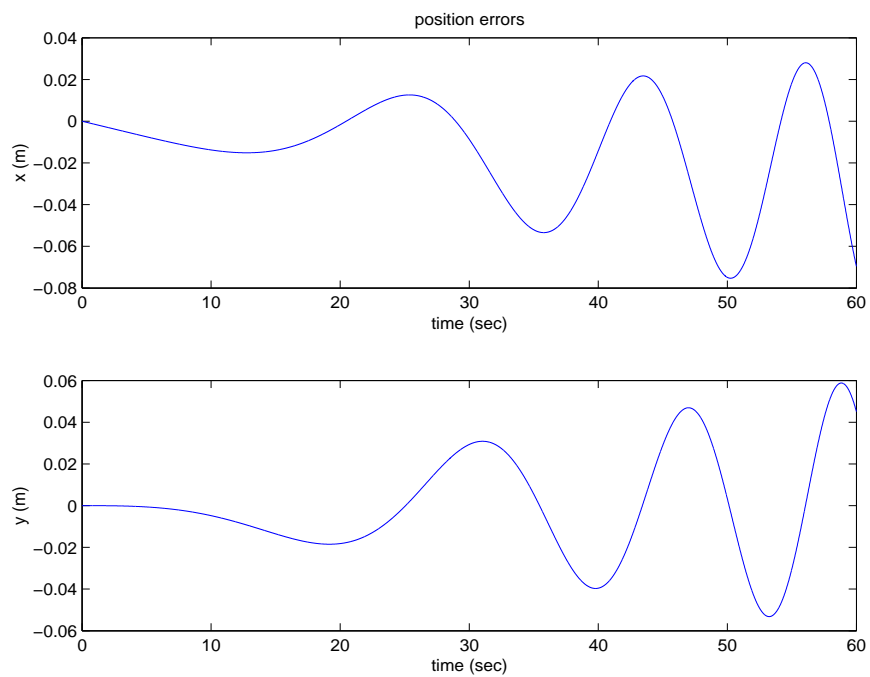Figure 7: Simulation result when u_{e1} is not compensated

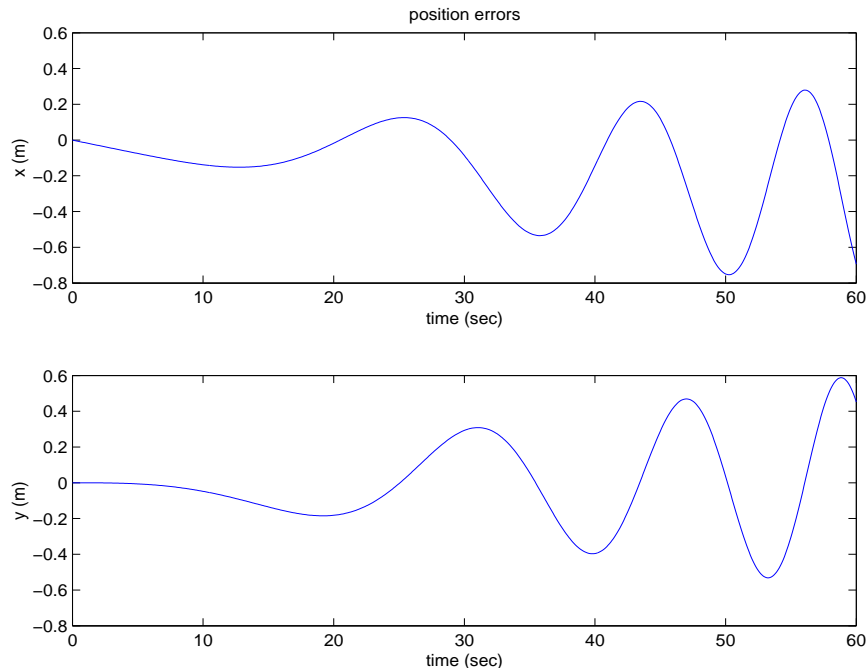Figure 8: Compensation of location error with $\Delta t = 0.01$ sec

23

Figure 9: Compensation of location error with $\Delta t = 0.1$ sec

7 Conclusions

We have examined the feasibility of designing an INS using only accelerometers. A simple sufficient condition - invertibility of the configuration matrix J - is used to determine if the body motion can be computed from the accelerometer outputs. This condition is "almost surely" satisfied since the set of singular 6×6 matrices is a "measure zero" set in $R^{6 \times 6}$. If a configuration of six accelerometers is feasible, there is a state equation (3.3) for angular acceleration and an output equation (3.4) for linear acceleration. While it is feasible to compute the motion, it may be a difficult task to solve the (nonlinear) state equation (3.3) numerically. This problem is tackled by designing a simple feedback compensation system (in Section 5) to realise the I/O system (3.3)-(3.4). The design is based on the corresponding I/O dynamical system equations for a cube configuration for which the angular acceleration is a linear combination of the accelerometer outputs.

The position and velocity estimates of a GF-INS are obtained by simple

computations that involve integration of the accelerometer data. Since the data are error-prone, the integration processes lead to errors that grow with time. So a GF-INS is a rapid diverging system, and indeed its output error diverges at a rate that is an order faster than that of a conventional gyroscope-based system. To enhance the performance of a GF-INS, it is thus necessary to estimate and correct the errors in the INS data by using external reference data, such as those from the Global Positioning System (GPS). An integrated GPS/GF-INS system needs to include the error dynamics of the INS data so that the errors can be estimated, compensated and bounded.

An integrated GPS/GF-INS system has a wide range of navigation applications, especially those where location information is most often safety-critical and time-critical. The location information could be used to provide traffic information to travellers, detect traffic incidents and congestion, or pinpoint the positions of other vehicles near an intersection for collision avoidance application. There are applications where the required level of accuracy and reliability is typically very high (cm-accuracy), and the information update rates are usually fast (100Hz). For example, in the case of controlling the vehicle positions in a platoon of vehicles. An integrated GPS/GF-INS system can provide the accuracy and fast update rate that are needed for this class of applications.

Appendix

The proofs of Proposition 2, Proposition 3, Theorem 1, and Proposition 4 are provided in this appendix.

Proof of Proposition 2
Proof: We have

$$
JQ = \begin{bmatrix} J_1^T & J_2^T \end{bmatrix} \frac{1}{2l^2} \begin{bmatrix} J_1 \\ l^2 J_2 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \frac{1}{l^2} J_1^T J_1 + J_2^T J_2 \end{bmatrix}
$$

Since $J_1 = [u_1 \times \theta_1, \cdots, u_6 \times \theta_6]$ and $J_2 = [\theta_1, \cdots, \theta_6]$, the (i, j) entry of JQ is:

$$
(JQ)_{ij} = \frac{1}{2} \begin{bmatrix} \frac{1}{l^2} (u_i \times \theta_i)^T (u_j \times \theta_j) + \theta_i^T \theta_j \end{bmatrix}
$$

By Lemma 1, $(JQ)_{ij}$ is 0 for $i \neq j$, and is 1 for $i = j$. Hence $Q = J^{-1}$. $\quad\square$

Proof of Proposition 3

Proof: It is easy to check that $J_1 J_2^T = O_3$ and $J_2 J_2^T = 2I_3$, where O_3 is the zero matrix and I_3 is the identity matrix in $R^{3\times3}$. So by Proposition 2 and Lemma 2, we get:

$$
Q \begin{bmatrix} \theta_1^T \Omega^2 u_1 \\ \vdots \\ \theta_6^T \Omega^2 u_6 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \frac{1}{l^2} J_1 \\ J_2 \end{bmatrix} \left(-l J_2^T \begin{bmatrix} \omega_2 \omega_3 \\ \omega_1 \omega_3 \\ \omega_1 \omega_2 \end{bmatrix} \right)
$$

$$
= - \begin{bmatrix} O_3 \\ l I_3 \end{bmatrix} \begin{bmatrix} \omega_2 \omega_3 \\ \omega_1 \omega_3 \\ \omega_1 \omega_2 \end{bmatrix} \tag{A.1}
$$

Substitute (A.1) into (3.2), and use the Q matrix in Proposition 2 and J_1 and J_2, we obtain the state equation (3.7) and output equation (3.8). □

Proof of Theorem 1

Proof: The accelerometer outputs generated by the configurations C_r and C_{cube} are A_r and A, respectively. The difference, $A_e = A_r - A$, is:

$$
A_e = J_e \begin{bmatrix} \dot{\omega} \\ L \end{bmatrix} + \begin{bmatrix} \theta_{e1}^T \Omega^2 u_1 + \theta_{r1}^T \Omega^2 u_{e1} \\ \vdots \\ \theta_{e6}^T \Omega^2 u_6 + \theta_{r6}^T \Omega^2 u_{e6} \end{bmatrix} \tag{A.2}
$$

where

$$
J_e = J_r - J = \begin{bmatrix} (u_1 \times \theta_{e1} + u_{e1} \times \theta_{e1})^T & \theta_{e1}^T \\ \vdots \\ (u_1 \times \theta_{e1} + u_{e1} \times \theta_{e1})^T & \theta_{e1}^T \end{bmatrix}
$$

For simplicity of notation, let $\left[x_i^T y_i \right]^*$ denote the column vector $\left[x_1^T y_1 \cdots x_6^T y_6 \right]^T$. The angular and linear motions can be computed using the I/O equations (3.2) for C_{cube} when there are no misalignment errors. So (A.2) becomes:

$$
A_e = J_e \left(QA - Q \left[\theta_i^T \Omega^2 u_i \right]^* \right) + \left[\theta_{ei}^T \Omega^2 u_i + \theta_{ri}^T \Omega^2 u_{ei} \right]^* \tag{A.3}
$$

Since $J_e = J_r - J$ and $JQ = I$, we get $(I + J_e Q)^{-1} = (J_r Q)^{-1}$. Next we use the relation $A = A_r - A_e$ and (A.3) for A_e to obtain:

$$
A = (J_r Q)^{-1} A_r + (J_r Q)^{-1} \left\{ J_e Q \left[\theta_i^T \Omega^2 u_i \right]^* \right.
$$

26

$$- \left[\theta_{ei}^T \Omega^2 u_i + \theta_{ri}^T \Omega^2 u_{ei} \right]^* \}$$

$$= (J_r Q)^{-1} A_r + (J_r Q)^{-1} \{ (J_r Q - I) \left[\theta_i^T \Omega^2 u_i \right]^*$$

$$- \left[\theta_{ei}^T \Omega^2 u_i + \theta_{ri}^T \Omega^2 u_{ei} \right]^* \}$$

$$= (J_r Q)^{-1} A_r + (J_r Q)^{-1} \{ J_r Q \left[\theta_i^T \Omega^2 u_i \right]^*$$

$$- \left[\theta_{ri}^T \Omega^2 u_{ri} \right]^* \} \tag{A.4}$$

$$=: K(A_r + H(\omega))$$

The feedback compensation system Σ is defined by (A.4) and (3.2) for C_{cube}. If we substitute (A.4) for A into (3.2), we get (5.1). So Σ realises the I/O system (5.1). $\qquad\square$

Proof of Proposition 4
Proof: Using the state equation (3.7) for C_{cube} and the relation (5.2), we get:

$$\dot{\omega}(t) = a(t) + b(\omega(t)) \ , \ t \geq t_0$$

where $a(t) := \frac{1}{2l^2} J_1 K A_r(t)$ and $b(\omega(t)) := \frac{1}{2l^2} J_1 K H(\omega(t))$. Integration of $\dot{\omega}$ gives:

$$\omega(t) = \omega(t_0) + \int_{t_0}^t a(\tau) d\tau + \int_{t_0}^t b(\omega(\tau)) d\tau \ , \ t \geq t_0 \tag{A.5}$$

Let $\hat{\omega}$ be the numerical solution of (A.5) with forward Euler approximation for the integral. That is, $\Delta t \dot{\hat{\omega}}(t_k) = \hat{\omega}(t_{k+1}) - \hat{\omega}(t_k)$, where $k = 0, \cdots, N-1$ and $T = N \Delta t$. From (A.5) the numerical data are:

$$\hat{\omega}(t_k) = \hat{\omega}(t_0) + \Delta t \sum_{j=0}^{k-1} a(t_j) + \Delta t \sum_{j=0}^{k-1} b(\hat{\omega}(t_j)) \tag{A.6}$$

Consider the two piecewise-constant functions below.

$$\hat{a}(t) := a(t_k) \ , \ t \in [t_k, t_{k+1}) \ , \ 0 \leq k \leq N-1$$

$$\hat{\omega}(t) := \hat{\omega}(t_k) \ , \ t \in [t_k, t_{k+1}) \ , \ 0 \leq k \leq N-1$$

These two piecewise-constant functions are integrable, so from (A.6) we get:

$$\hat{\omega}(t_k) = \hat{\omega}(t_0) + \int_{t_0}^{t_k} \hat{a}(\tau) d\tau + \int_{t_0}^{t_k} b(\hat{\omega}(\tau)) d\tau \tag{A.7}$$

27

The function $a(\cdot)$ is a linear combination of the actual accelerometer outputs A_{ri}, so it is continuous and there exist constants $c_k < \infty$ such that:

$$\sup_{t \in [t_k, t_{k+1})} |\hat{a}(t) - a(t)| < c_k \ , \ 0 \le k \le N - 1$$

Let $c = \max\{c_k : 0 \le k \le N - 1\}$. Also for each fixed T, it is not difficult to see that there is a Lipschitz constant $k_T > 0$ such that:

$$|b(\hat{\omega}(t)) - b(\omega(t))| \le k_T |\hat{\omega}(t) - \omega(t)| \tag{A.8}$$

So (A.5)-(A.8) together give:

$$|\hat{\omega}(t_k) - \omega(t_k)| \le |\hat{\omega}(t_0) - \omega(t_0)| + ck\Delta t + k_T \int_{t_0}^{t_k} |\hat{\omega}(\tau) - \omega(\tau)| d\tau$$

Next by the Bellman-Gronwall Lemma ([12]), we obtain:

$$
\begin{aligned}
|\hat{\omega}(t_k) - \omega(t_k)| &\le (|\hat{\omega}(t_0) - \omega(t_0)| + ck\Delta t)\, e^{k_T(t_k - t_0)} \\
&\le (|\hat{\omega}(t_0) - \omega(t_0)| + k_1\Delta t)\, e^{k_2 T}
\end{aligned}
$$

where $k_1 = ck$ and $k_2 = k_T$. This completes the proof. $\square$

Acknowledgments

References

[1] Yazdi, N., Farrokh A., and Khalil N., "Micromachined Inertial Sensors", *Proceedings of IEEE*, Vol. 86, No. 8, August 1998, pp. 1640-1659.

[2] Greenspan, R. L. "Inertial Navigation Technology from 1970-1995", *Journal of the Institute of Navigation*, Vol. 42, No. 1, Spring 1995, pp. 165-185.

[3] Kourepenis, A., Borenstein, J., Connelly, J., Elliott, R., Ward, P., and Weinberg, M., *Performance of MEMS Inertial Sensors*, New York, NY, USA: IEEE, 1998.

[4] Lemkin, M. A., Boser, B. E., "A 3-Axis Force Balanced Accelerometer Using Single Proof Mass", *Proceedings of Transducers*, June 1997, pp. 1185-1188.

[5] Howe, R. T., Boser, B. E., and Horowitz, R., "Integrated Micro-Electro-Mechanical Sensor Development for Inertial Applications", *IEEE Positions, Location and Navigation Symposium*, 1998.

[6] Chen, J-H., Lee, S-C., and DeBra, D., "Gyroscope Free Strapdown Inertial Measurement Unit by Six Linear Accelerometers", *Journal of Guidance, Control, and Dynamics*, Vol. 17,, No. 2, March-April 1994, pp. 286-290.

[7] Padgaonkar, A. J., Krieger, K. W., and King, A. I., "Measurement of Angular Acceleration of a Rigid Body Using Linear Accelerometers", *Journal of Applied Mechanics* in the *Transactions of the ASME*, Vol. 42, September 1975, pp. 552-556.

[8] Mital, N. K. and King, A. I., "Computation of Rigid-Body Rotation in Three-Dimensional Space from Body-Fixed Linear Acceleration Measurements", *Journal of Applied Mechanics* in the *Transactions of the ASME*, Vol. 46, December 1979, pp. 925-930.

[9] Ljung, P. B., *Micromachined Gyroscope with Integrated Electronics*, Doctoral Thesis, University of California, Berkeley, 1997.

[10] Murray, R., Li, Z., and Sastry, S. S., *A Mathematical Introduction to Robotic Manipulation*, Boca Raton, CRC Press, 1994.

[11] Tan, Q. and Balchen, J. G., "General Quaternion Transformation Representation for Robotic Application", Proceedings of IEEE 1993 International Conference on Systems, Man and Cybernetics, Vol. 3, 1993, pp.319-324.

[12] Kailath, T., *Linear Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1980.