Daniel Choukroun
Yaakov Oshman
Julie Thienel
Moshe Idan *Editors*

Advances in Estimation, Navigation, and Spacecraft Control

Selected Papers of the Itzhack Y. Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control

Springer

Advances in Estimation, Navigation, and Spacecraft Control

Daniel Choukroun · Yaakov Oshman
Julie Thienel · Moshe Idan
Editors

Advances in Estimation, Navigation, and Spacecraft Control

Selected Papers of the Itzhack Y. Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control

Springer

Editors
Daniel Choukroun
Mechanical Engineering Department
Ben-Gurion University of the Negev
Beer Sheva
Israel

Yaakov Oshman
Faculty of Aerospace Engineering
Technion - Israel Institute of Technology
Haifa
Israel

Julie Thienel
NASA Goddard Space Flight Center
Greenbelt
USA

Moshe Idan
Faculty of Aerospace Engineering
Technion - Israel Institute of Technology
Haifa
Israel

ISBN 978-3-662-44784-0 ISBN 978-3-662-44785-7 (eBook)
DOI 10.1007/978-3-662-44785-7

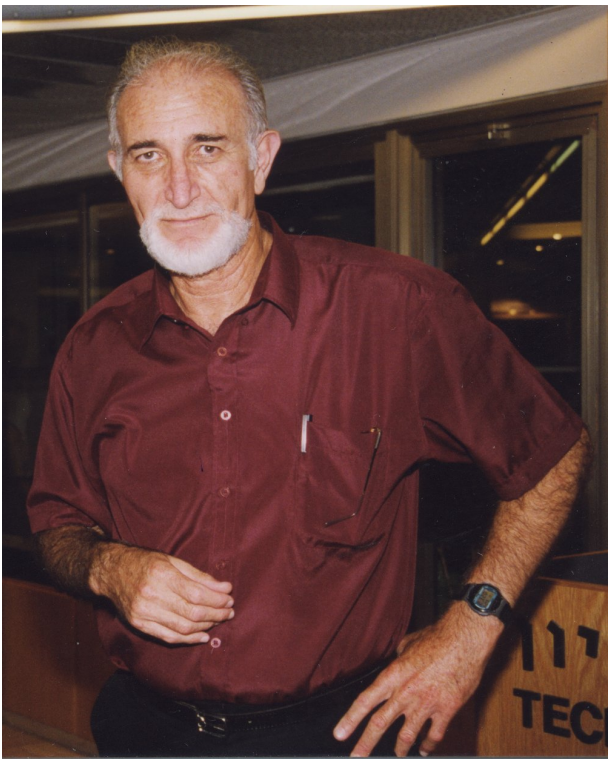
Library of Congress Control Number: 2014953108

Springer Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer-Verlag GmbH Berlin Heidelberg is part of Springer Science+Business Media (www.springer.com)

Preface



Background

Itzhack Y. Bar-Itzhack, professor Emeritus of Aerospace Engineering at the Technion – Israel Institute of Technology, was a prominent and world-renowned member of the applied estimation, navigation, and spacecraft attitude determination communities. He touched the lives of many. He had a love for life, an incredible sense of humor, and wisdom that he shared freely with everyone he met.

To honor Professor Bar-Itzhack's memory, as well as his numerous seminal professional achievements, an international symposium was held in Haifa, Israel, on October 14–17, 2012, under the auspices of the Faculty of Aerospace Engineering at the Technion and the Israeli Association for Automatic Control.

Itzhack Yoav Bar-Itzhack (1937-2007)

Since receiving his Ph.D. degree in electrical engineering from the University of Pennsylvania in 1968, with a dissertation titled "Strapdown Inertial Navigation", Professor Bar-Itzhack had made seminal contributions to the art of Inertial and Aided Navigation Systems, and was one of the pioneers of the theory of strapdown Inertial Navigation Systems (INS). Prof. Bar-Itzhack was also a distinguished member of a select group of international experts in the field of spacecraft attitude determination. He published extensively in this area, often with NASA co-workers, covering problems of attitude and angular rates estimation from vector measurements, attitude determination from GPS measurements, and combined attitude and orbit determination. Professor Bar-Itzhack contributed to the theory of applied optimal filtering as well, by developing, with students, batch-recursive data compression filters, and eigenfactor-based square root filters. Between 1968-1971 he was employed by Bellcomm Inc. in Washington DC, where he worked on the Apollo-project, analyzing the navigation system of the Lunar Roving Vehicle (LRV). In 1977-8, during his sabbatical, he served as a member of the technical staff at the Analytic Sciences Corporation (TASC) in Reading MA, where he was involved in the Improved Accuracy Program on the Trident missile guidance system. Prof. Bar-Itzhack also spent several sabbaticals and summers at the NASA Goddard Space Flight Center. There he was instrumental in developing the first real time attitude filter for use in the spacecraft control center and he supported the development of attitude estimation systems for several NASA spacecraft.

Professor Bar-Itzhack was very active in national and international professional communities, serving, among others, as an International Advisor of the Journal of Guidance, Control, and Dynamics, a member of the AIAA Guidance, Navigation and Control Technical Committee, and as a President of the Israeli Association for Automatic Control (national member organization of IFAC). He contributed immensely to the security of Israel by sharing knowledge and wisdom with the Israeli MoD and with Israeli defense related industries, including Rafael-Advanced defense Systems, Ltd., IAI-Israel Aerospace Industries, Ltd., and IMI-Israel Military Industries, Ltd. At the Technion-Israel Institute of Technology, he served as Dean of the Faculty of Aerospace Engineering, among numerous other positions.

Professor Bar-Itzhack graduated 33 doctoral and master's students, and published over 80 papers in major archival journals and 140 papers in conference proceedings. He received numerous professional honors, including the IEEE Third Millenium Award, the IEEE Kershner Award, the NASA Exceptional Technology Achievement Medal, and the NASA Goddard Space Flight Center Group Achievement Award. Along with two NASA colleagues, he was awarded a US patent in 2004 for an autonomous navigation system. He was a Fellow of AIAA "...in recognition of professional distinction and valuable contributions made to the arts, sciences and technology of aeronautics and astronautics," and a Fellow of IEEE "...for contributions to the development of inertial navigation system." Prof. Bar-Itzhack was chosen by the IEEE Aerospace and Electronic Systems Society to serve as a member

of the Distinguished Lecturers Program, with his talk on "The Evolution of Inertial Navigation."

About the Book

For the Itzhack Y. Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control, the International Program Committee conducted a formal review process. Each paper was reviewed in compliance with good journal practices by at least two independent and anonymous reviewers. The papers published in this book were selected based on the results and recommendations from the reviewers.

The book contains 27 selected, revised, and edited contributed chapters written by eminent international experts. The book is organized in three parts: (1) Estimation, (2) Navigation and (3) Spacecraft Guidance, Navigation and Control. The volume was prepared as a reference for research scientists and practicing engineers from academy and industry in the fields of estimation, navigation, and spacecraft GN&C.

About the Symposium

The international Itzhack Y. Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control, chaired by

- Yaakov Oshman (Technion-Israel Institute of Technology)
- Julie Thienel (NASA Goddard Space Flight Center)

was held in Haifa, Israel, on October 14–17, 2012. Along this four days single-track event, attended by more than 100 participants, 50 talks were presented with contributors from 18 different countries. Six eminent experts from academy and industry delivered the following keynote talks during the technical sessions:

- Mark J. Balas (Embry-Riddle Aeronautical University)
  "The Beautiful Simplicity of Direct Adaptive Control in Theory and (maybe) in Practice"
- F. Landis Markley (NASA Goddard Space Flight Center)
  "I Have Some Attitude Issues"
- Mark. L. Psiaki (Cornell University)
  "Securing Civilian GNSS Services in the Presence of Jamming and Spoofing"
- Hector P. Rotstein (Rafel-Advanced Defense Systems, Ltd.)
  "FOG, MEMS, and all the rest-Some thoughts on Inertial Navigation technologies and algorithms"
- Jason L. Speyer (University of California, Los Angeles)
  "Stochastic Estimation and Control for Linear Systems with Cauchy Noise"

and two keynote talks were given during a social session at the Gala dinner

- Daniel Weihs (Technion - Israel Institute of Technology)
  "Learning Engineering from Nature"
- Yoav Medan (Technion - Israel Institute of Technology)
  "StarTrek Surgery-Science Meets Fiction"

Acknowledgments

It is our pleasure to thank for their valuable and patient support

- members of the National Organizing Committee:
  - Nadav Berman (Ben-Gurion University of the Negev; Prof. Berman, a dear friend and colleague passed away on April 11, 2012)
  - Per-Olof Gutman (Technion-Israel Institute of Technology)
  - Moshe Idan (Technion-Israel Institute of Technology)
  - Leonid Mirkin (Technion-Israel Institute of Technology)
  - Tal Shima (Technion-Israel Institute of Technology)
- members of the International Program Committee:
  - Amit Ailon (Ben-Gurion University of the Negev)
  - Zvi Artstein (Weizmann Institute of Science)
  - Mark J. Balas (Embry-Riddle Aeronautical University)
  - Yaakov Bar-Shalom (University of Connecticut)
  - Anthony J. Calise (Georgia Institute of Technology)
  - Avishy Carmi (Ben-Gurion University of the Negev)
  - Daniel Choukroun (Ben-Gurion University of the Negev)
  - John L. Crassidis (University at Buffalo, SUNY)
  - Bernard Friedland (New Jersey Institute of Technology)
  - Hari B. Hablani (Indian Institute of Technology, Bombay)
  - Richard R. Harman (NASA Goddard Space Flight Center)
  - F. Landis Markley (NASA Goddard Space Flight Center)
  - Mark L. Psiaki (Cornell University)
  - Ilia Rapoport (Elbit Systems, Electro-Optics ELOP, Ltd.)
  - Hector P. Rotstein (Rafael-Advanced Defense Systems, Ltd.)
  - Uri Shaked (Tel-Aviv University)
  - Jason L. Speyer (UCLA)
  - Min-Jea Tahk (Korea Advanced Institute of Science and Technology)
  - Haim Weiss (Rafael-Advanced Defense Systems, Ltd.)
  - Isaac Yaesh (Israel Military Industries, Ltd.)
  - Paul Zarchan (MIT Lincoln Laboratory)
- all symposium participants including speakers and sessions chairs

The present book would not have been possible without the support and careful work of the international college of reviewers purposely nominated for this task. The quality of any scientific book is largely due to the will of reviewers to share their expertise and knowledge with unknown colleagues from all over the world.

Last but not least thanks are due to our sponsors without which the symposium would not have taken place:

- Technion-Israel Institute of Technology
- AIAA Guidance Navigation and Control Technical Committee
- Bernard M. Gordon Center for Systems Engineering at the Technion
- IMI-Israel Military Industries, Ltd.
- Rafael-Advanced Defense Systems, Ltd.
- IAI, Tamam Division-Israel Aerospace Industries, Ltd.
- European Office of Aerospace research and Development, Air Force Office of Scientific research, United States Air Force Research Laboratory
- Technion Autonomous Systems Program
- IEEE Israel

The Editors

Daniel Choukroun
Yaakov Oshman
Julie Thienel
Moshe Idan

Contents

Part I: Estimation

Part II: Navigation

Part III: Spacecraft Guidance, Navigation and Control

Part I
Estimation

Steady-State Time Constant of the Kalman Filter

Joseph Etzion

Self employed
etzion_y@netvision.net.il

Abstract. Under conditions of constant sampling rate and constant sampling errors, after a sufficiently long tracking time, all terms of the covariance matrix converge to their corresponding asymptotic magnitudes. After convergence, all variables of the Riccati equation remain unchanged. In such conditions the only unknown in this equation is the steady-state covariance matrix. An analytic solution of the resulting steady-state Riccati equation for "short" sampling times is derived. After solution all terms of the resulting covariance matrix are then expressed as functions of a particularly defined "time constant". The application of the time constant produces several practical advantages: it greatly facilitates the determination of the model-noise parameter (q), provides a fair assessment of the tracking accuracies, and determines the steady-state transfer function of the filter.

Keywords: Time constant, tracking accuracy, transfer function.

1 Kalman Equations in Steady State

A sequential tracking by a Kalman filter is done by a repeated execution of prediction and update of both the covariance matrix and the state vector. In the analysis that follows the *discretized continuous-time white noise acceleration* (*CWNA*) *model* is assumed.

Assume that at a given time t the covariance matrix, P, and the state vector, X, are known. Also assume that at time $t+T$ some measurement has been made. At this time the variables of the filter are predicted to the new time instant by the following equations:

$$P_p = \Phi P \Phi^T + Q \qquad \text{Covariance,} \qquad (1)$$

$$X_p = \Phi X \qquad \text{State vector,} \qquad (2)$$

where Φ is the transition matrix and Q is the covariance matrix of model-noise. Both these matrices depend on the sampling time T (see definite expressions in the following sections).

Then, the data of the new measurement are used for updating the predicted variables as follows:

$$K = P_p H^T \left(H P_p H^T + S \right)^{-1} \qquad \text{Gain matrix,} \qquad (3)$$

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_1

$$P_u = (I - KH) P_p \qquad \text{Covariance,} \qquad (4)$$

$$X_u = X_p + K\varepsilon \qquad \text{State vector,} \qquad (5)$$

where H is the measurement matrix, S is the covariance matrix of the measurement errors, I is an identity matrix, and ε is the innovation vector (see definition in the following sections).

Assume a scenario where the H and S matrices are time independent. If, in addition, the sampling time is also kept constant, then Φ and Q remain time independent as well. In such scenario, after long enough cycling of (1), (3), and (4) the covariance matrix, P, and the gain matrix, K, remain virtually constant. In other words, after the variance stabilization, the updated covariance, P_u, returns to its unpredicted magnitude, P, or

$$P_u = P. \qquad (6)$$

By a substitution of (1) in (3), and a further substitution in (4), and then a substitution of the resulting expression of P_u in (6), we get an equation with a single unknown, which is the covariance matrix P. This matrix equation breaks down into a number of scalar equations with the same number of scalar unknowns. For instance, in a first-order filter applied to a single coordinate the covariance matrix contains three distinct (unknown) elements, which can be evaluated by solving the three scalar equations. For a second-order filter, the number of unknowns, or the number of equations, is six.

Approximate solutions of the first and the second order filters are presented in the following three sections. For convenience, all covariances are expressed in terms of a time constant. Despite the approximations used, the resulting accuracy of filter characteristics is remarkable. The simple covariance expressions derived here and the concept of the time constant are greatly instrumental in a preliminary estimation of filter performance, even in cases of multiple-coordinate dynamic scenarios.

2 Steady-State Solution of First-Order Filter

In stead-steady conditions the process of first-order Kalman filter becomes identical to the alpha-beta filter or the Wiener filter ([1], Sections 6.5.4 and 9.5). The solution presented here, however, is aimed at deriving at a linkage between the time constant, τ, and the model-noise parameter, q. Such linkage is helpful for a practical determination of the required q-parameter. In addition, as will be shown below, the time constant is advantageous for a preliminary assessment of the expected tracking errors.

Consider Kalman filtering on a single coordinate. Assume the position in this single coordinate is sampled at constant time intervals, T. The "true" position of the tracked body will be called y. This time-dependent position is sampled with measurement errors of a constant standard deviation (STD), σ_y. The sampled y, with its measurement error, is the input to the filter. The resulting output from the filter will be x for position and v for velocity.

In such single-coordinate scenario, the arrays used in the preceding section are defined as follows.

$$P = \begin{pmatrix} p_x & p_{xv} \\ p_{xv} & p_v \end{pmatrix} \qquad \text{Covariance matrix,} \qquad (7)$$

$$\Phi = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \qquad \text{Transition matrix,} \qquad (8)$$

$$Q = \begin{pmatrix} T^3/3 & T^2/2 \\ T^2/2 & T \end{pmatrix} q \qquad \text{Model-noise,} \qquad (9)$$

$$X = \begin{pmatrix} x \\ v \end{pmatrix} \qquad \text{State vector,} \qquad (10)$$

$$H = (1 \quad 0) \qquad \text{Measurement matrix,} \qquad (11)$$

$$S = \sigma_y^2 \qquad \text{Meas. variance,} \qquad (12)$$

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \qquad \text{Identity matrix.} \qquad (13)$$

Note that X is the state vector, whereas x is the position variable, which is an element of X.

The model-noise matrix, Q, used here (equation 9), is in the form of the popular Singer model [2]. In this form, q is the model-noise parameter, defined by $q = 2\sigma_m^2 \tau_m$, where σ_m is an acceleration STD of the tracked target's maneuver, and τ_m is the time constant of the maneuvers. In practice, however, the two parameters, σ_m and τ_m, are extremely vague, and even meaningless. As a result of this uncertainty, the q parameter is being routinely determined empirically, by way of extensive simulations or field experimentations. In contrast, the time constant (measured in seconds) is a lot easier to assess by considering the expected maneuverability of the tracked object. For instance, appropriate time constant for tracking a maneuvering aircraft by radar is typically between four and ten seconds; for vessel tracking the optimal time constant is somewhat larger that that.

In the present analysis, in any case, the q-parameter is considered as a given number, expressed in units of $\mathrm{m}^2/\mathrm{sec}^3$, which can be interpreted as a per-second growth of the velocity variance. In the analysis that follows, the derived covariance terms will be expressed as functions of q.

By using (7), (8), and (9) in (1), we get the following predicted covariance terms:

$$p_{xp} = p_x + 2Tp_{xv} + T^2 p_v + T^3 q/3 \qquad \text{Position,} \qquad (14)$$

$$p_{xvp} = p_{xv} + Tp_v + T^2 q/2 \qquad \text{Covariance,} \qquad (15)$$

$$p_{vp} = p_v + Tq \qquad \text{Velocity.} \qquad (16)$$

We know in advance that all terms of the covariance matrix in steady-state conditions of the filter's operation are approximately proportional to the sampling time, *T*. For sufficiently small sampling times, most terms in the above three equations become negligible. For the needs of the present analysis, in these equations we only retain the terms which exceed the magnitude of T^2. After such omission, these equations reduce to the following approximate form:

$$p_{xp} \approx p_x + 2Tp_{xv} \qquad \text{Position,} \tag{17}$$

$$p_{xvp} \approx p_{xv} + Tp_v \qquad \text{Covariance,} \tag{18}$$

$$p_{vp} \approx p_v + Tq \qquad \text{Velocity.} \tag{19}$$

Next, we simplify the expression of the gain matrix, *K*, given by (3). In this equation, the product HPH^T becomes a scalar, given by

$$HPH^T = \sigma_y^2 . \tag{20}$$

By using this result in (3), the gain matrix becomes

$$K = \frac{1}{p_{xp} + \sigma_y^2} \begin{pmatrix} p_{xp} \\ p_{xvp} \end{pmatrix} \approx \frac{1}{\sigma_y^2} \begin{pmatrix} p_{xp} \\ p_{xvp} \end{pmatrix}, \tag{21}$$

where p_{xp} has been neglected relative to σ_y^2.

Further, the resulting gain, *K*, is substituted in (4), which likewise can be broken down into its separate scalar components. Again after ignoring all powers of *T* greater than 2, the following updated expressions are obtained:

$$p_{xu} \approx p_x + 2Tp_{xv} - p_x^2 / \sigma_y^2 \qquad \text{Position,} \tag{22}$$

$$p_{xvu} \approx p_{xv} + Tp_v - p_x p_{xv} / \sigma_y^2 \qquad \text{Covariance,} \tag{23}$$

$$p_{vp} \approx p_v + Tq - p_{xv}^2 / \sigma_y^2 \qquad \text{Velocity.} \tag{24}$$

In steady-state conditions, the updated covariances return to their previous magnitudes, before the prediction stage, which are the same as after the previous update. Such equality is expressed by (6), which is in this case equivalent to three scalar equations. By using these equalities in the last three equations, the dominant terms (on the left) cancel out, and we are left with their following reduced form:

$$2Tp_{xv} = p_x^2 / \sigma_y^2 , \tag{25}$$

$$Tp_v = p_x p_{xv} / \sigma_y^2 , \tag{26}$$

$$Tq = p_{xv}^2 / \sigma_y^2 . \tag{27}$$

These simple equations provide the following solution for the three steady-state covariance terms:

$$p_x = \sqrt[4]{4T^3\sigma_y^6 q} \qquad \text{Position,} \tag{28}$$

$$p_v = \sqrt[4]{4T\sigma_y^2 q^3} \qquad \text{Velocity,} \tag{29}$$

$$p_{xv} = T\sigma_y^2 q \qquad \text{Covariance.} \tag{30}$$

We now introduce the following definition of the time constant:

$$\tau = \sqrt[4]{T\sigma_y^2 / q}\ , \tag{31}$$

or the inverse relation

$$q = T\sigma_y^2 / \tau^4\ . \tag{32}$$

The time constant defined by (31) is also related to "target maneuvering index," λ_c, defined in [1], (6.5.4-10). This particular relation becomes

$$\lambda_c = T^2 / \tau^2\ . \tag{33}$$

In real applications, however, a preliminary assessment of an optimal τ magnitude is practically easier than an assessment of both λ_c and q before some simulation testing can be accomplished. While optimal values of λ_c or q can vary by orders of magnitude depending on the system parameters, the optimal τ varies remarkably little. For instance, for tracking a maneuvering aircraft the optimal time constant can vary between about 4 and 6 seconds; for tracking a ship the time constant can become about 10 seconds. In contrast, the high power of τ in (32) or (33) make the other alternative parameters a lot more uncertain. It is therefore advisable, for practical system design, to first make an estimate of an optimal time constant, and then to determine the corresponding q magnitude by means of (32).

The particular definition of the time constant by means of (31) will become clearer by considering the dynamic response of the filter, presented in the next section.

Equation (32) is now substituted in (28) and (29) in order to provide alternative expressions for the position and the velocity variances. The square roots of these variances are the STDs of the corresponding estimation errors:

$$\sigma_x = \sigma_y \sqrt{\frac{T\sqrt{2}}{\tau}} \qquad \text{Position,} \tag{34}$$

$$\sigma_v = \frac{\sigma_y}{\tau} \sqrt{\frac{T\sqrt{2}}{\tau}} \qquad \text{Velocity,} \tag{35}$$

where σ_y is the STD of the measurement error, T is the sampling time, and τ is the time constant.

In addition, the position-velocity correlation coefficient is given as follows:

$$\rho = \frac{p_{xv}}{\sqrt{p_x p_v}} = \frac{1}{\sqrt{2}}. \tag{36}$$

For the dynamic specification of the filter, expressions of the filter gains will be necessary. For the first-order filter, the gains are given by the two terms of the gain matrix, K, according to (21). By substituting the results of p_x and p_{xv} (from (28) and (30)) for p_{xp} and p_{xvp} in (21), the following steady-state gains are obtained:

$$k_x = T\sqrt{2}/\tau \qquad \text{Position gain,} \tag{37}$$

$$k_v = T/\tau^2 \qquad \text{Velocity gain.} \tag{38}$$

3 Dynamic Response of First-Order Filter

The response of the Kalman filter is determined by a sequential execution of (2) and (5). For the first-order filter, these equations are expressed by the following two scalar equations:

$$x_{n+1} = x_n + Tv_n + k_x\varepsilon \qquad \text{Position,} \tag{39}$$

$$v_{n+1} = v_n + k_v\varepsilon \qquad \text{Velocity,} \tag{40}$$

where n is the sample number, and ε is the innovation, given by the difference between the last measurement and the predicted position:

$$\varepsilon = y_{n+1} - x_p. \tag{41}$$

In steady-state conditions, the gains used in (39) and (40) are given by (37) and (38). In the limit $T \to 0$, (39) and (40) convert to integrals, where the sampling time, T, is replaced by the infinitesimal time increment dt. The resulting continuous response of the filter can then be described by a corresponding flow diagram, as shown in Fig. 1.

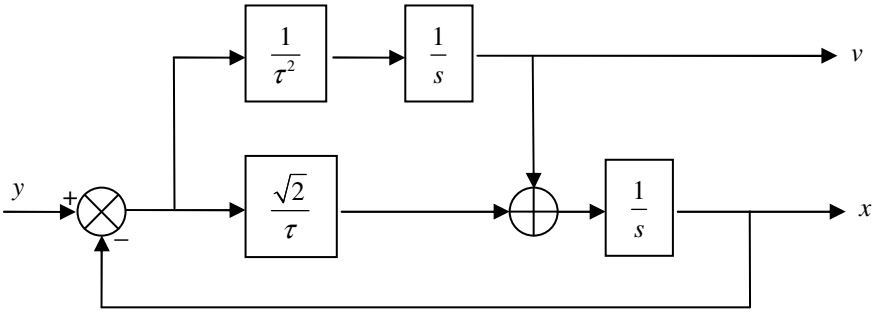


Fig. 1. Transfer function of first-order filter

The diagram of Fig. 1 provides the following transfer functions of the first-order Kalman filter:

$$\frac{x}{y} = \frac{1+\sqrt{2}\tau s}{1+\sqrt{2}\tau s+\tau^2 s^2} \qquad \text{Position,} \tag{42}$$

$$\frac{v}{y} = \frac{s}{1+\sqrt{2}\tau s+\tau^2 s^2} \qquad \text{Velocity,} \tag{43}$$

where τ is the time constant (equation (31)) and s is the Laplace variable.

It can be seen that the transfer function given by (42) is analogous to the transfer function of the Wiener filter ([1], (9.5.2-28)).

In time domain, (42) and (43) convert to the following differential equations:

$$\tau^2\ddot{x}+\sqrt{2}\tau\dot{x}+x = y+\sqrt{2}\tau\dot{y} \qquad \text{Position,} \tag{44}$$

$$\tau^2\ddot{v}+\sqrt{2}\tau\dot{v}+v = \dot{y} \qquad \text{Velocity.} \tag{45}$$

The homogeneous part of any of the last two equations (their left-hand sides) has the following two complex zeros:

$$z = -\frac{1}{\tau\sqrt{2}} \pm \frac{i}{\tau\sqrt{2}} . \tag{46}$$

Hence, the homogeneous solution of (44) is given by

$$x_h(t) = \left(C_1 \sin\frac{t}{\tau\sqrt{2}} + C_2 \cos\frac{t}{\tau\sqrt{2}} \right) e^{-t/\tau\sqrt{2}} , \tag{47}$$

where C_1 and C_2 are constants to be determined by the appropriate initial conditions.

As an example, a step response of the filter is presented for a case of a unit step, introduced at time zero. The step function at the input to the filter is defined by

$$y(t) = \begin{cases} 0, & t<0 \\ 1, & t\ge 0 \end{cases}, \tag{48}$$

where t is time, measured from the unit step.

It is assumed, however, that at time zero the filter has already stabilized in the steady-state conditions, where the filter gains remain constant at their asymptotic values, given by (37) and (38).

After the introduction of the unit step, the general solution of the position function is given by

$$x(t) = 1 + x_h(t) , \tag{49}$$

where $x_h(t)$ is the homogeneous solution, given by (47).

The initial conditions, necessary for the constants' determination, can be deduced from the diagram in Fig. 1. They are given by

$$x(0)=0, \quad \dot{x}(0)=\sqrt{2}/\tau\,. \tag{50}$$

By using the initial conditions in conjunction with (49) and its derivative, the following constants are obtained:

$$C_1=1, \quad C_2=-1\,, \tag{51}$$

which provide the following position function for the step response:

$$x(t)=1+\left(\sin\frac{t}{\tau\sqrt{2}}-\cos\frac{t}{\tau\sqrt{2}}\right)e^{-t/\tau\sqrt{2}}\,. \tag{52}$$

For velocity, the solution of (45) is similar to the position solution, except that in this case the asymptotic velocity magnitude is zero. The initial conditions for velocity are

$$v(0)=0, \quad \dot{v}(0)=1/\tau^2\,. \tag{53}$$

The general solution for velocity becomes

$$v(t)=\frac{\sqrt{2}}{\tau}\sin\frac{t}{\tau\sqrt{2}}e^{-t/\tau\sqrt{2}}\,. \tag{54}$$

Both the position and the velocity responses are of the form of damped oscillations. The first position maximum is $1+e^{-\pi/2}=1.2079$, and it occurs at time $t=\tau\pi/2=2.2214\tau$. The first velocity maximum is $e^{-\pi/4}/\tau=0.4459/\tau$, and it occurs at time $t=\tau\pi/2\sqrt{2}=1.1107\tau$.

Figs. 2 and 3 present a comparison between the approximate analytic solution of the filter response and the actual sequential solution of the first-order Kalman filter. Both solutions take place in steady-state conditions, which become valid after long enough filter operation with constant measurement errors and constant time steps, when the covariance matrix stabilizes at its asymptotic values. In order to achieve the steady-state conditions of the actual Kalman filter prior to the unit-step introduction, (1), (3), and (4) have been cycled until all terms of the covariance matrix stopped changing up to their sixth decimal significant figure. The initial covariance matrix in this stabilization was constructed from the expressions given by (34), (35), and (36), according to the definition given by (7). For convergence to six significant figures a time span of about 6τ was required. However, no detectable effect on the results shows up even after convergence to four figures. The extra two figures were left for the sake of safety.

Fig. 2 shows a comparison between the actual and the analytic response of the first-order Kalman filter to a unit step in steady-state conditions. This example is given for a time step of $T = 0.2\tau$. For such time step a maximum position difference of about 2% between the two calculations is detected, while in velocity the difference is too small to be seen at the given scale. For time steps smaller than 0.05τ, even the position differences practically vanish.

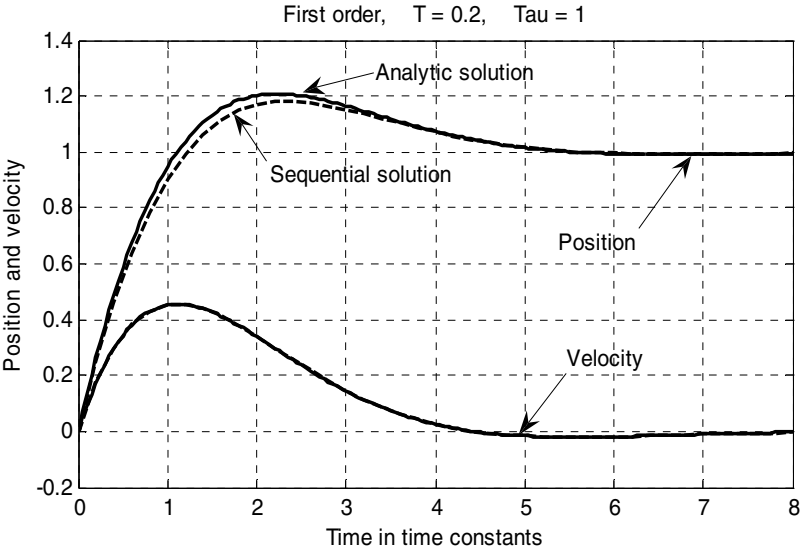


Fig. 2. Step response of first-order Kalman filter

Even though the analytic solution was derived for "small" time steps, in practice, time steps as large as $T = \tau$ provide remarkable similarity between the two solutions, with differences of only 10% or less. One reason of such closeness of solution is the radical insensitivity of the Kalman filter to most elements of the model-noise matrix, Q (equation (9)). In the derivation of the analytic approximation, the *only* non-negligible term of Q was its bottom-right element. This means that virtually the same results are obtained with a degenerate matrix of the following form:

$$Q = \begin{pmatrix} 0 & 0 \\ 0 & T \end{pmatrix} q \,. \tag{55}$$

The use of the degenerate form of the model-noise matrix, instead of its original form taken from the Singer model [2], does not create any detectable differences in the filter's response even for time steps as large as $T = \tau$ in any of the above calculations. This sort of test of the filter's performance clearly shows that the "high sophistication" of the popular Singer model is actually an illusion, meaningless in its practical applications.

Fig. 3 presents a comparison between the analytic and the true filtering errors of the first-order Kalman filter in steady-state conditions for varying magnitudes of time step. This comparison is done between the analytic STDs, given by (34) and (35), and the square roots of the diagonal elements of the asymptotic covariance matrix. For moderate time steps, of $T = 0.2\tau$, the differences are 7% and 4% for position and velocity, respectively. For the extreme case of $T = \tau$ the differences amount to 37% and 17% for the two functions.

The results presented in Figs. 2 and 3 demonstrate a remarkable reliability of the approximate solution of Riccati equation, summarized by (34) and (35), even for

relatively large time steps (T), relative to the time constant (τ) of the Kalman filter. In particular, the results show that a preliminary estimate of model-noise parameter, q, can confidently be done by means of (32).

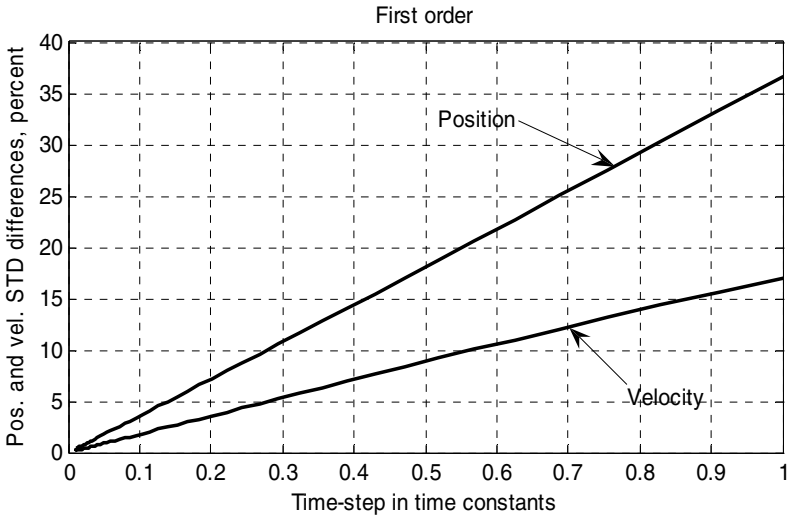


Fig. 3. Error differences between analytic and actual solutions of first-order Kalman filter

For very large time steps, when $T > \tau$, the Kalman filter in fact degenerates into a straight line drawn between the latest two samples, disregarding all preceding measurements. In such cases the solution presented above is no longer necessary; rather, the filter's performance can then be assessed by a much simpler analysis of the straight-line fit.

Even though the analysis presented above has been derived for a sufficiently long tracking time, the resulting equations provide a fair approximation of the filter performance even when the tracking time is comparable to the time constant. For shorter tracking times, of less than one time constant, the filter becomes equivalent to least-squares fit of a straight line to all measurements collected so far. In this case, again, the solution presented above becomes unnecessary because it can be replaced by the simpler approach of the least-squares principle.

4 Steady-State Solution of Second-Order Filter

In steady-state conditions the second-order Kalman filter becomes identical to the alpha-beta-gamma filter ([1], Section 6.5.5).

The derivation of an approximate analytic solution in steady state is repeated here for the second-order Kalman filter. For the second order, all equations shown in Section I remain valid in this case as well, and only the matrices involved increase their dimensions. The previous matrices, given above by (7) through (13), are replaced here by the following expressions:

$$P = \begin{pmatrix} p_x & p_{xv} & p_{xa} \\ p_{xv} & p_v & p_{va} \\ p_{xa} & p_{va} & p_a \end{pmatrix} \qquad \text{Covariance,} \qquad (56)$$

$$\Phi = \begin{pmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{pmatrix} \qquad \text{Transition,} \qquad (57)$$

$$Q = \begin{pmatrix} T^5/20 & T^4/8 & T^3/6 \\ T^4/8 & T^3/3 & T^2/2 \\ T^3/6 & T^2/2 & T \end{pmatrix} q \qquad \text{M. noise} \qquad (58)$$

$$X = \begin{pmatrix} x \\ v \\ a \end{pmatrix} \qquad \text{State vector,} \qquad (59)$$

$$H = (1 \quad 0 \quad 0) \qquad \text{Measurement matrix,} \qquad (60)$$

$$S = \sigma_y^2 \qquad \text{Measurement variance,} \qquad (61)$$

$$I = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad \text{Identity matrix,} \qquad (62)$$

where x, v and a stand for position, velocity and acceleration, respectively.

In the second-order filter, the units of the model-noise parameter, q, are $\mathrm{m}^2/\mathrm{sec}^5$, which can be interpreted as a per-second growth of the acceleration variance.

The derivation of the approximate solution of the second-order filter exactly follows the steps used in the case of the first-order filter (Sections 2 and 3), only with a higher algebraic complication. Here we skip the lengthy algebra, and proceed to the presentation of the analytic results.

For the second-order filter, the time constant is defined by

$$\tau = \sqrt[6]{T\sigma_y^2/q}\,, \qquad (63)$$

or the inverse relation

$$q = T\sigma_y^2/\tau^6\,. \qquad (64)$$

Here, due to the sixth power of τ in (64), the contrast between the magnitudes of τ and q is even higher than in the case of the first-order filter. As explained above (see paragraph after (33)), there are typical values of τ that correspond to certain targets. Surprisingly, the same typical values mentioned before remain almost unchanged for the second order as well. Hence again, it is highly preferable to first select an appropriate magnitude of an optimal time constant, and then make a determination of q by means of (64).

By the (63) definition of the time constant, the approximate steady-state output STDs of the second-order filter are given by

$$\sigma_x = \sigma_y \sqrt{\frac{2T}{\tau}} \qquad \text{Position STD,} \qquad (65)$$

$$\sigma_v = \frac{\sigma_y}{\tau} \sqrt{\frac{3T}{\tau}} \qquad \text{Velocity STD,} \qquad (66)$$

$$\sigma_a = \frac{\sigma_y}{\tau^2} \sqrt{\frac{2T}{\tau}} \qquad \text{Acceleration STD.} \qquad (67)$$

In the second-order filter there are three correlation coefficients between the three output variables:

$$\rho_{xv} = \sqrt{2/3} = 0.816 \qquad \text{Pos.-vel.,} \qquad (68)$$

$$\rho_{xv} = 1/2 \qquad \text{Pos.-acc.,} \qquad (69)$$

$$\rho_{va} = \sqrt{2/3} = 0.816 \qquad \text{Vel.-acc.} \qquad (70)$$

In this case, the following steady-state Kalman gains are obtained:

$$k_x = 2T/\tau \qquad \text{Position,} \qquad (71)$$

$$k_v = 2T/\tau^2 \qquad \text{Velocity,} \qquad (72)$$

$$k_a = T/\tau^3 \qquad \text{Acceleration.} \qquad (73)$$

In the limit $T \longrightarrow 0$, the recursive filter equations convert to integrals, which can be presented by a flow diagram. The flow diagram in this case is an extension of the first-order flow diagram of Fig. 1. The extended diagram is shown in Fig. 4.

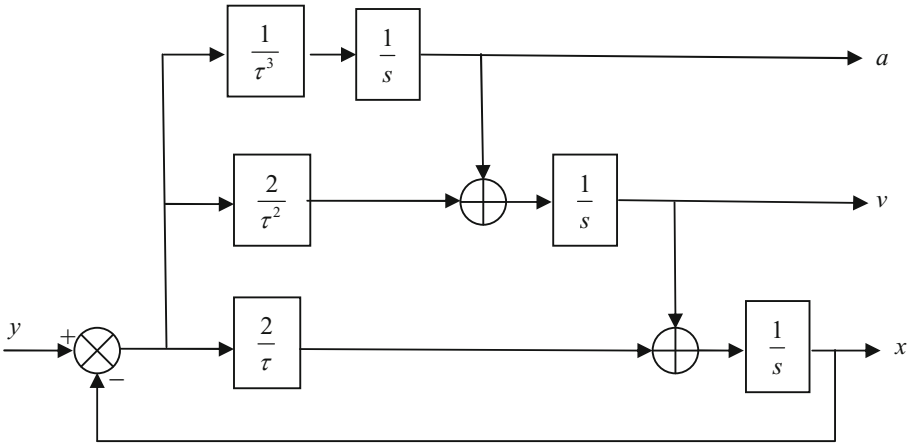


Fig. 4. Transfer function of second-order filter

The diagram of Fig. 4 provides the following transfer functions of the second-order Kalman filter:

$$\frac{x}{y}=\frac{1+2\tau s+2\tau^2 s^2}{1+2\tau s+2\tau^2 s^2+\tau^3 s^3} \qquad \text{Position,} \qquad (74)$$

$$\frac{v}{y}=\frac{s\left(1+2\tau s\right)}{1+2\tau s+2\tau^2 s^2+\tau^3 s^3} \qquad \text{Velocity,} \qquad (75)$$

$$\frac{a}{y}=\frac{s^2}{1+2\tau s+2\tau^2 s^2+\tau^3 s^3} \qquad \text{Acceleration,} \qquad (76)$$

where τ is the time constant (equation (63)) and s is the Laplace variable.

The differential equations, determined by the above three transfer functions, will be determined by the zeros of their denominators, which are third-degree polynomials of s. These polynomials have one real zero and two complex zeros, as follows:

$$z_1=-\frac{1}{\tau}, \quad z_{2,3}=-\frac{1}{2\tau}\pm\frac{i\sqrt{3}}{2\tau}. \qquad (77)$$

These zeros provide the following homogeneous solution of the differential equations:

$$x_h\left(t\right)=C_1 e^{-t/\tau}+\left(C_2\sin\frac{t\sqrt{3}}{2\tau}+C_3\cos\frac{t\sqrt{3}}{2\tau}\right)e^{-t/2\tau}. \qquad (78)$$

The general solution for the step response of $x(t)$ is given as before by (49).

For a determination of the three constants in (78), the following initial conditions of the position variable are applied:

$$x\left(0\right)=0, \quad \dot{x}\left(0\right)=2/\tau, \quad \ddot{x}(0)=-2/\tau^2. \qquad (79)$$

These initial conditions, and the constants that follow, determine the following general solution for the position step-response of the second-order filter:

$$x\left(t\right)=1-e^{-t/\tau}+\frac{2}{\sqrt{3}}\sin\frac{t\sqrt{3}}{2\tau}e^{-t/2\tau}. \qquad (80)$$

For velocity and acceleration the homogeneous equation is the same as (78), but their asymptotic values are zero. The initial conditions for velocity are

$$v\left(0\right)=0, \quad \dot{v}\left(0\right)=2/\tau^2, \quad \ddot{v}(0)=-3/\tau^3, \qquad (81)$$

and its general solution becomes

$$v\left(t\right)=\frac{1}{\tau}\left[-e^{-t/\tau}+\left(\sqrt{3}\sin\frac{t\sqrt{3}}{2\tau}+\cos\frac{t\sqrt{3}}{2\tau}\right)e^{-t/2\tau}\right]. \qquad (82)$$

The initial conditions for acceleration are the following:

$$a(0)=0, \quad \dot{a}(0)=1/\tau^3, \quad \ddot{a}(0)=-2/\tau^4, \tag{83}$$

and its unit-step response is then

$$a(t)=\frac{1}{\tau^2}\left[-e^{-t/\tau}+\left(\frac{1}{\sqrt{3}}\sin\frac{t\sqrt{3}}{2\tau}+\cos\frac{t\sqrt{3}}{2\tau}\right)e^{-t/2\tau}\right]. \tag{84}$$

A comparison between the analytic solutions, given by (80), (82) and (84), and the sequential solution of the Kalman equations is shown in Fig. 5. The time step is same as before (in Fig. 2), $T = 0.2\tau$. Here again, like in the first-order case, the differences between the two solutions are remarkably small. At much greater time steps, even as large as $T = \tau$, the differences between the two solutions also remain moderate (up to about 20% in this case).

Fig. 6 presents a comparison between the analytic and the true filtering errors of the second-order Kalman filter in steady-state conditions for varying magnitude of time step. For moderate time steps, such as $T = 0.2\tau$, the relative STD-differences remain small, about 10%, but they grow substantially for much larger time steps. In addition, unlike the case of the first-order filter (Fig. 3), they do not converge to zero at the limit of $T \longrightarrow 0$, even though the absolute STD magnitudes do converge to zero at that limit. Such characteristics take place apparently because at the small time steps there still remains some non-negligible effect of higher powers of T, which were ignored in the approximate solution of the second-order filter.

The almost complete insensitivity of the filter performance to the form of the model-noise matrix, Q, is manifest in the case of the second-order filter as well. By replacing all elements except one in the matrix in (58) by zeros, no meaningful change in the filter's response can be detected. Such insensitivity exists even with the large time step of $T = \tau$. The degraded matrix in this case is given by

$$Q=\begin{pmatrix}0&0&0\\0&0&0\\0&0&T\end{pmatrix}q. \tag{85}$$

By using (85) instead of (58) for the model-noise matrix, no change can be detected in the sequential solution in Fig. 5. In Fig. 6 there is some change only in the acceleration curve in case of such replacement.

This insensitivity again emphasizes the futility of attempts of formulating an advanced model-noise matrix.

Similar to the first-order case, for the second-order filter as well a remarkable reliability of the approximate solution of the Riccati equation is demonstrated. This means that in this case the model-noise parameter, q, can confidently be evaluated by means of (64), and the expected performance of the second-order Kalman filter can be represented by (65), (66) and (67).

Also similar to the first-order filter, for time steps larger than one time constant the filter this time degenerates into a second-order polynomial drawn between the latest

three samples. In the other situation, when the tracking time is shorter than one time step, the filter becomes equivalent to least-squares fit of a second-order polynomial to all measurements collected up to that point.

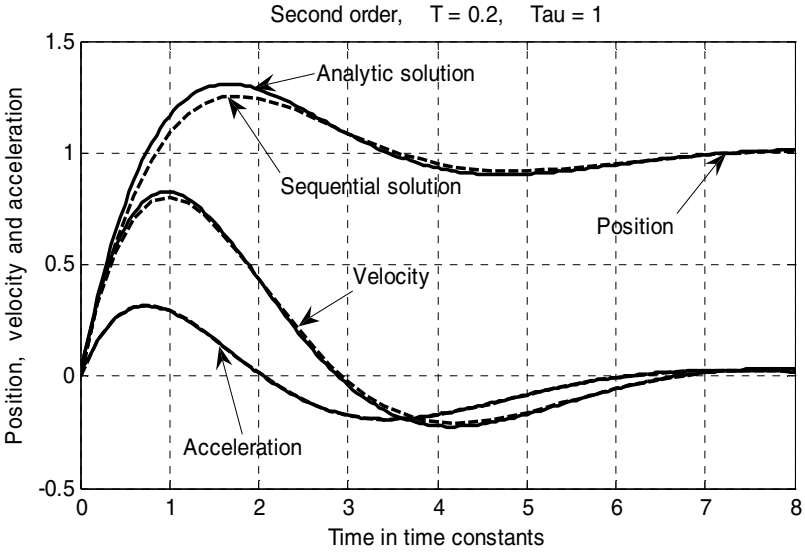


Fig. 5. Step response of second-order Kalman filter

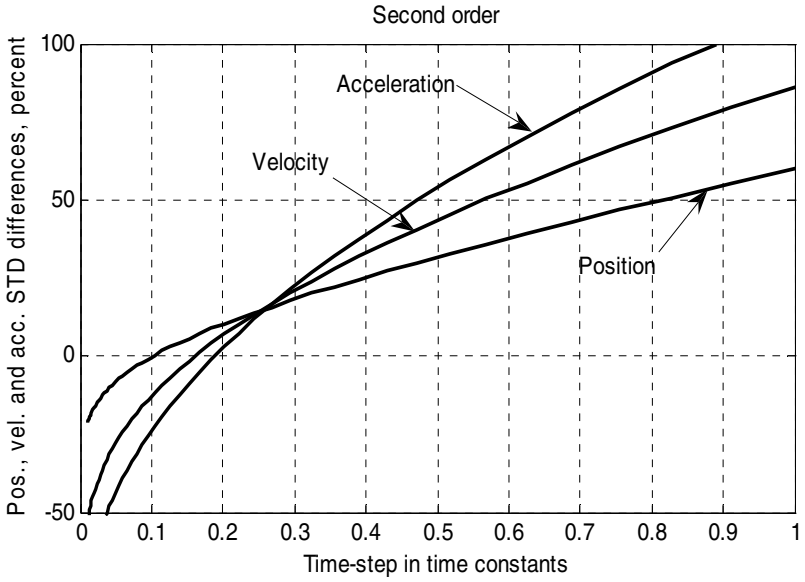


Fig. 6. Error differences between analytic and actual solutions of second-order Kalman filter

5 Summary

The analytic solution of the first- and second-order Kalman filter has been derived for a single coordinate in steady-state conditions, with constant sampling time and constant measurement errors. However, despite those limitations, the analytic solutions for the estimation errors and the dynamic response of the filter are extremely useful in preliminary assessments of the filter's performance in much broader operation conditions. In a variety of actual scenarios the magnitude of the time constant, τ, can safely be estimated by using (31) or (63) by assuming quasi-static conditions, separately for each measured coordinate. For example, in a three-dimensional radar tracking, a separate time constants can be obtained for the range, azimuth and elevation coordinates by using the actual measurement errors in those directions at a given time. True, the three line-of-sight coordinates are not entirely uncorrelated, but except for cases of very short ranges the correlation is quite small. In addition, in dynamic scenarios the measurement errors may change in time, but again, except for very short ranges the dynamics are not very high in relation to the magnitude of the time constant.

In reality, in a three-dimensional tracking system, because of the different measurement errors in (32) and (64) for each coordinate, there will be three possible values of the q-parameter. The selected magnitude of q should be the largest one of the three. Later, at the practical filter application, the actual magnitudes of τ will automatically be determined for each coordinate separately in accordance with (31) or (63).

A possible variation of the sampling time in a given scenario is also not very disturbing with respect to the time-constant estimate. As seen in (31) and (63), the dependence of the time constant on T is very weak, so that a use of some "average" value of the sampling time can still provide very useful performance estimates. Then, on the base of the representative time constants, the anticipated output errors from the filter can be calculated by means of (34) and (35) for the first-order filter, or by (65), (66), and (67) for the second-order filter.

The transfer functions of the filter ((42) and (43) for the first-order filter, and (74), (75), and (76) for the second-order filter) can also be very useful for a quick estimation of the effect of various tracking disturbances. For example, a systematic error caused by a presence of acceleration when the first-order filter is applied, can be estimated by means of (42). Similar estimates can be done for other deviations, such as a ramp response (a velocity step) or an acceleration step.

The approximate analytic results of the filter performance, presented above, provide possibilities for very fast estimates of the filter's performance, before an investment of big efforts and resources in developing extensive simulation programs and field experiments. The preliminary analysis is capable of a timely identification of unexpected problems in the early stages of development of technological projects.

Another by-product of the present analysis is the demonstration of the filter's insensitivity to the exact form of the model-noise matrix. Instead of the "advanced" form of this matrix, given by (9) or (58), the degenerate form by (55) or (85) can be used without fear of any unwanted side effects. In addition, instead of the usual

guesswork involved in the "exact" determination of the model-noise parameter, q (see description after (13)), a much more practical approach is to first select a reasonable magnitude for the time constant, and then calculate the corresponding value of q by using (32) or (64), according to the case.

References

1. Bar-Shalom, Y., Rong Lee, X., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation, pp. 286, 287, 289, 362, 366. John Wiley and Sons (2001)
2. Blackman, S., Popoli, R.: Design and Analysis of Modern Tracking Systems. Section 4.2. Artech House (1999)

Track-to-Track Fusion in Linear and Nonlinear Systems*

Xin Tian, Ting Yuan, and Yaakov Bar-Shalom

University of Connecticut
Storrs, CT 06269-4157, USA
{xin.tian,tiy,ybs}@engr.uconn.edu

Abstract. This chapter starts with a review of the architectures for track-to-track fusion (T2TF). Based on whether the fusion algorithm uses the track estimates from the previous fusion and the configuration of information feedback, T2TF is categorized into six configurations, namely, T2TF with no memory with no, partial and full information feedback, and T2TF with memory with no, partial and full information feedback. The exact algorithms of the above T2TF configurations and the impact of information feedback on fusion accuracy are presented. Although (under the Linear Gaussian assumption) the exact T2TF algorithms yield theoretically consistent fusion results, their major drawback is the need of the crosscovariances of the tracks to be fused, which drastically complicates their implementation. The information matrix fusion (IMF) is a special case of T2TF with memory. Although it is heuristic when not conducted at full rate, it was shown to have consistent and near optimal fusion performance for practical tracking scenarios. Due to its simplicity, it is a good candidate for practical tracking systems. For the problem of asynchronous T2TF (AT2TF), a generalized version of the IMF is presented. It supports information feedback for AT2TF in the presence of communication delay, and was shown to have good consistency and close to optimal fusion accuracy. Finally the fusion of heterogenous tracks where the states at the local trackers are nonlinearly related and of different dimension is discussed. For the problem of the fusion of the track from an Interacting Multiple Model (IMM) estimator from an active sensor with the track from a passive sensor, a counterintuitive phenomenon that heterogenous T2TF may have better performance than the centralized measurement-to-track fusion approach (which is the known optimum in the linear case) is demonstrated and explained.

1 Introduction

In tracking applications, when more than one sensor is used to obtain measurements, there are several possible configurations for information processing, which are summarized in Fig. 1. They differ in the sequence in which the data association and tracking are carried out and the information available to the various processors [6].

* This work was supported by grants ARO W911NF-10-1-0369 and ONR N00014-10-1-0029.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_2

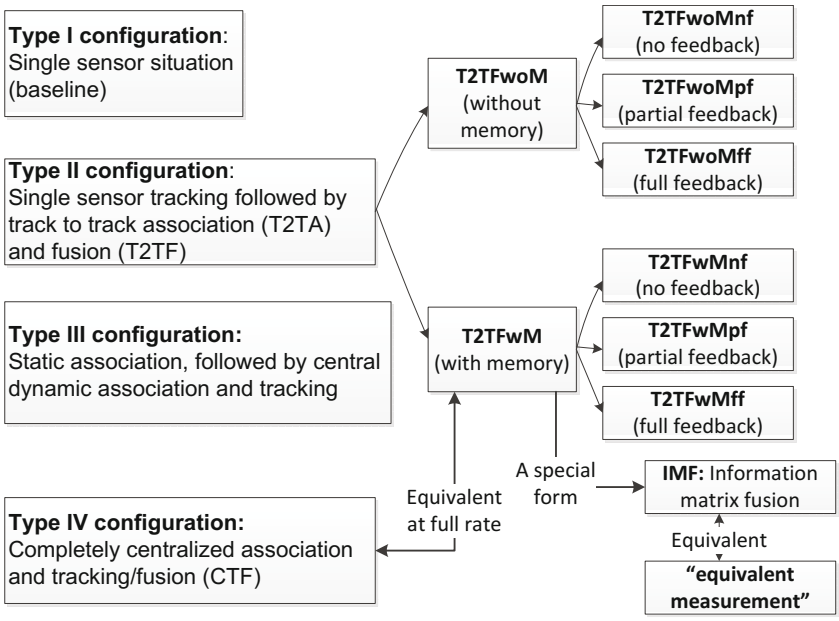


Fig. 1. Configurations for Multisensor Information Processing

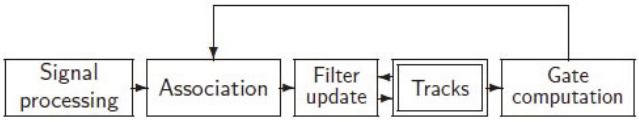


Fig. 2. Type I configuration — single sensor tracking

Type I configuration refers to the (standard) tracking system using a single sensor, which has the flowchart depicted in Fig. 2. In a multisensor situation this corresponds to reporting responsibility (RR). Each sensor operates alone and has responsibility for a certain sector of the surveillance region – no fusion of the data (measurements or tracks) from the multiple sensors is done. As targets move from one sector to another, they are handed over – handoff – in a manner that depends on the system. Generally, the mechanism is to assign responsibility to the sensor with the highest expected accuracy, although workload and communication constraints can also play a role.

Type III configuration is the static intersensor association and fusion followed by central processing which consists of two stages. In the first stage, the measurements from the various sensors, assumed to be from the same time, i.e., the sensors are assumed synchronized, are first associated and fused. This is a static intersensor measurement association and centralized measurement fusion (CMF)

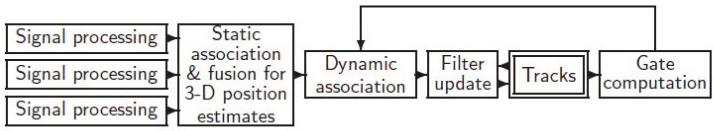


Fig. 3. Type III configuration — static association, extraction of composite measurements (fusion), followed by dynamic association/tracking

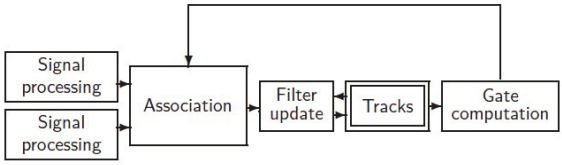


Fig. 4. Type IV configuration — centralized multisensor tracking (Centralized Tracker/Fuser — CTF)

that yields composite measurements, also known as supermeasurements. Then in the second stage, these composite measurements are processed by a (central) dynamic association and tracking algorithm. Fig. 3 shows such a configuration for 3 sensors.

The Type IV configuration, which is the centralized tracking (also called centralized tracker/fuser — CTF), is depicted in Fig. 4. In this configuration all the measurements are sent to the center, which carries out the association with all the available information and then uses these measurements to update the tracks. Since this configuration uses the maximum available information, it will provide (subject to the limitations of the specific data association algorithm it uses) the best results [6]. In the absence of the need for data association, it will yield, for linear systems, the globally optimal estimates [6]. As it will be shown later, in a nonlinear problem with heterogenous trackers, HT2TF (heterogenous T2TF) can be superior to CTF.

The Type II configuration, which is the main subject of this chapter, is the single sensor tracking followed by track fusion. Its importance stems from the fact that it can run at a low rate, e.g., on demand. This is important in situations where the communication bandwidth from the sensors to the fusion center is limited. This configuration is distributed and decentralized, where each sensor has its own information processor local data associator/tracker and yields full tracks. A Fusion Center (FC) carries out the association and fusion of the local tracks into system tracks. These steps are designated as Track-to-Track Association (T2TA) and Track to Track Fusion (T2TF), respectively. The Type II configuration is very important in distributed tracking systems. Compared to

the CTF, which requires the transmission of local sensor measurements to the FC at the full rate, T2TF can be conducted at much lower rates, which can significantly reduce communication requirements.

As shown in Fig. 1, depending on whether the fusion algorithm uses the track estimates from the previous fusion and the configuration of information feedback, T2TF configurations are further categorized as: T2TF with no memory with no information feedback (T2TFwoMnf), partial information feedback (T2TFwoMpf) and full information feedback (T2TFwoMff), and T2TF with memory with no information feedback (T2TFwMnf), partial information feedback (T2TFwMpf) and full information feedback (T2TFwMff). These configurations as well as the impact of information feedback on the fusion accuracy are discussed in detail in Sect. 2.

The impact of the crosscorrelation between local estimation errors on fused estimate for T2TFwoMnf is discussed in Sect. 3.

Also shown in Fig. 1, the information matrix fusion (IMF) [10,8,15,11,6] is a special form of T2TFwM. Operating at full rate the IMF is equivalent to the Type IV Configuration, i.e., the CTF, while, at a reduced rate, the IMF is heuristic. However, for the practical range of system process noises levels, it was shown to yield consistent[1] fusion results and close to the optimal fusion accuracy [9]. Also note that the IMF is algebraically equivalent to the equivalent measurement approach [6]. Compared to the exact T2TF fusion algorithms, the IMF has the advantage of not requiring the crosscovariances of the errors of the tracks to be fused, which significantly simplifies the implementation. Sect. 4 reviews the IMF at full and reduced rate, as well as a generalized IMF (GIMF) for the fusion of asynchronous tracks which supports information feedback in the presence of communication delay and was shown to have consistent and close to optimal fusion results [18].

Another special type of T2TF that may occur in practical tracking systems is the fusion of tracks from heterogeneous trackers that are using different state vectors, discussed in Sect. 5. In [19] the HT2TF problem was investigated, where the track from an interacting multiple model (IMM) filter with states in Cartesian coordinates and using an active sensor was fused with the track from a passive sensor with angular states. Counterintuitively, it was shown that when the IMM tracker is involved, HT2TF yielded better performance than the CTF approach. These results are presented in Sect. 6. Sect. 7 summarizes the chapter with concluding remarks.

2 Track-to-Track Fusion Configurations and the Impact of Information Feedback

Depending on whether the track estimates from the previous fusion are used for the current fusion and the configuration of information feedback, T2TF can be categorized as the following configurations:

[1] Its errors were commensurate with its calculated covariance [2].

- T2TFwoM with no information feedback (T2TFwoMnf)
- T2TFwoM with partial information feedback (T2TFwoMpf)
- T2TFwoM with full information feedback (T2TFwoMff)
- T2TFwM with no information feedback (T2TFwMnf)
- T2TFwM with partial information feedback (T2TFwMpf)
- T2TFwM with full information feedback (T2TFwMff)

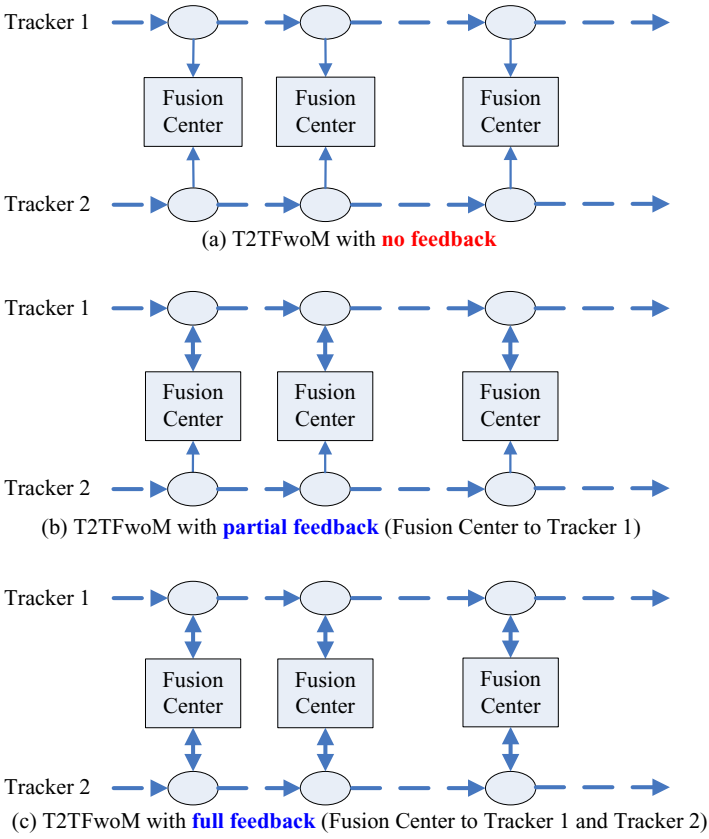


Fig. 5. Information configurations for T2TFwoM

2.1 T2TF without Memory

In T2TFwoM, the FC uses only the current track estimates with no memory of the track estimates from the previous fusion. Fig. 5 illustrates the three information configurations of T2TFwoM, where two local tracks (that pertain to the same target) are fused at certain times. The first configuration is the T2TFwoMnf [5], designated as Config. IIa for multisensor tracking in [1]. As

indicated in Fig. 5(a), the two tracks evolve independently without any information from each other, thus the improved accuracies are achieved only at the fusion times at the FC. The second configuration is the T2TFwoM with partial information feedback (T2TFwoMpf) which belongs to the Config. IIb in [1]. In this case, as shown in Fig. 5(b), track 1 is fused with track 2 and continues with the fused track (feedback) from the FC. However, track 2 does not receive the fused track in view of the partial information feedback. The third configuration is the T2TFwoM with full information feedback (T2TFwoMff), which also belongs to the Type IIb configuration in [1]. As shown in Fig. 5(c), in this case both trackers receive and continue with the fused track.

The exact algorithms for T2TFwoM can be found in [6,16]. The key is to evaluate the crosscovariances between the local tracks. As shown in [3], although the measurements at different local trackers have independent noises, the local tracks are correlated due to common process noises of the target's motion. Ignoring the crosscovariances in T2TF will lead to over-optimistic fused covariance and track inconsistency [6]. In [16] it was shown that, compared to CTF, T2TFwoM always has a certain loss in fusion accuracy and, counterintuitively, information feedback has a negative impact on the accuracy of T2TFwoM.

To illustrate this phenomenon, consider the following generic T2TF example. The target state is defined as $[x \;\; \dot{x}]'$. The target motion is modeled as the discrete white noise acceleration (DWNA) model in [2], Sect. 6.3.2. It is assumed that two sensors obtain position measurements of the target with a sampling interval of $T = 1\,\mathrm{s}$. The standard deviation of the measurement noise is $\sigma_w = \sqrt{R_l} = 30\,\mathrm{m}$ for each sensor (i.e., at each local tracker) and the process noise variance is $q = 1\,\mathrm{m}^2/\mathrm{s}^4$. T2TFwoM takes place every 5 s, i.e., at a reduced rate.

Table 1. Fuser variances (at fusion times) in steady state (fusion interval: 5 s)

Fusion Type	FC track at fusion time	
	Position	Velocity
T2TFwoMff	133	6.29
T2TFwoMpf	131	6.30
T2TFwoMnf	125	6.30
CTF	119	6.03
Single sensor tracker	205	7.26

Table 1 shows the steady state variances of position and velocity at the FC. All the fused tracks are more accurate than the single-sensor (local) tracks without fusion, which have steady state variances as 205 in position and 7.26 in velocity. Note that at the fusion time the position estimates of all the fused tracks have a small degradation compared to the CTF: 5% for T2TFwoMnf, 10% for T2TFwoMpf, 12% for T2TFwoMff. This shows that T2TFwoM has a degradation in fusion accuracy compared to CTF and this degradation increases in the presence of information feedback.

To explain this phenomenon consider the gains of the steady state filter for the above problem, namely, the alpha-beta filter. In steady state the filter gain is a *monotonically increasing function* of the maneuvering index λ [2].

Under T2TFwoMnf, the filter gain of each measurement in the fused track (with two equal-accuracy sensors) is

$$W^{\text{T2TFwoMnf}} = \frac{1}{2}[\alpha(\lambda_l),\ \beta(\lambda_l)/T]' \tag{1}$$

where $\lambda_l = \frac{\sqrt{q}T^2}{\sqrt{R_l}}$ is the local maneuvering index of the two trackers ($\sqrt{q}$ and $\sqrt{R_l}$ are the standard deviations of the process noise and measurement noise, respectively).

Under CTF, $z_c = \frac{1}{2}(z_1 + z_2)$ and $R_c = \frac{1}{2}R_l$. Thus, the central maneuvering index is $\lambda_c = \sqrt{2}\lambda_l$, i.e., larger. For each measurement, the centralized (globally optimal) filter gain in steady state for each measurement — its weighting — is

$$\begin{aligned} W^C &= \frac{1}{2}[\alpha(\lambda_c),\ \beta(\lambda_c)/T]' = \frac{1}{2}[\alpha(\sqrt{2}\lambda_l),\ \beta(\sqrt{2}\lambda_l)/T]' \\ &> W^{\text{T2TFwoMnf}} = \frac{1}{2}[\alpha(\lambda_l),\ \beta(\lambda_l)/T]' \end{aligned} \tag{2}$$

With information feedback, the local filter gains will be even smaller than without feedback, i.e., *they will deviate further* from the globally optimal gains.[2] This is because the local trackers have more accurate information due to the feedback (compared to the no feedback case) and this *reduces their filter gains* for the new measurements.

2.2 T2TF with Memory

In the configuration of T2TFwM, the fusion involves both the track estimates at the current fusion time and those from the previous fusion time. Fig. 6 illustrates T2TFwM with no, partial and full information feedback.

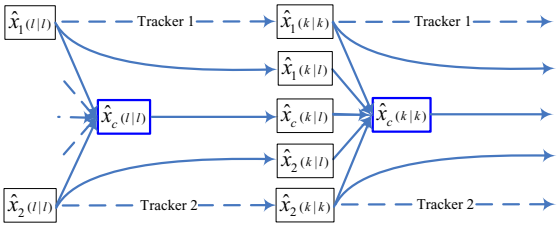
The exact fusion algorithms for the three T2TFwM configurations were presented in [16]. It was shown that, at full rate, T2TFwM has equivalent fusion performance with the CTF (for a linear system). However, at a reduced rate, compared to the CTF there is a certain loss of fusion accuracy, which is unavoidable [16]. And unlike the case with T2TFwoM, information feedback improves the fusion accuracy of T2TFwM. This phenomenon is illustrated with the following example.

The state of the target (taken as a scalar for simplicity) evolves according to

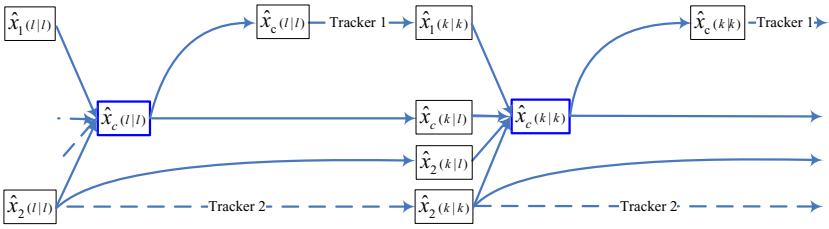
$$x(k) = x(k-1) + v(k) \qquad k = 2, 3, \ldots \tag{3}$$

where $v(k)$ is the process noise with variance $q = 0.3$.

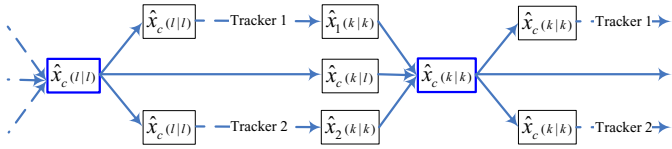
[2] Gains smaller or larger than the optimal gains (which yield the minimum MSE) will lead to a MSE larger than the minimum [2]. The relationship between the optimal gain and the optimal state estimation MSE is discussed in detail in [2].



(a) T2TFwM with no information feedback (one cycle: from fusion time l to the next fusion time k)



(b) T2TFwM with partial information feedback (one cycle: from fusion time l to the next fusion time k)



(c) T2TFwM with full information feedback (one cycle: form fusion time l to the next fusion time k)

Fig. 6. T2TFwM at arbitrary rate

Two trackers, 1 and 2, take measurements of the target with independent measurement noises w_1 and w_2, namely,

$$z_i(k) = x(k) + w_i(k) \qquad i = 1, 2 \tag{4}$$

where $w_i(k)$ are zero-mean Gaussian noises with variance $R_i = 1, i = 1, 2$. The two trackers calculate tracks of the target with their own measurements using a Kalman filter. Each local track is initialized at time 1 with the first local measurement. The first T2TF happens at time 1. Then T2TFwM occurs every $N_f = 3$ sampling times.

Table 2 shows the fuser- and tracker-calculated variances when the fuser (with memory) is operating at reduced rate. These results verify the conclusions on the impact of information feedback on T2TFwM, namely, that feedback is beneficial. Note that due to the reduced rate, T2TFwM is suboptimal compared to CTF, but only slightly.

Table 2. Fuser and tracker 1 calculated variances at fusion times for $N_f = 3$ (reduced rate), $q = 0.3$, $R_1 = R_2 = 1$

Time		1	3	6	9	12	15
T2TFwMnf	Tracker1	1.0000	0.4639	0.4196	0.4180	0.4179	0.4179
	Fuser	0.5000	0.2772	0.2698	0.2694	0.2694	**0.2694**
T2TFwMpf	Fuser	0.5000	0.2763	0.2690	0.2688	0.2688	**0.2688**
T2TFwMff	Fuser	0.5000	0.2755	0.2683	0.2682	0.2682	**0.2682**
CTF/CMF		0.5000	0.2743	0.2654	0.2653	0.2653	**0.2653**

Note that the IMF, detailed in the Sect. 4, also uses the previous track estimates (i.e., it has memory). When operating at full rate, IMF is algebraically equivalent to CTF and also to the algorithms for T2TFwM (for a linear system). However, at a lower rate, the IMF is heuristic.

Limitations of the exact T2TF algorithms discussed above include i) the exact fusion algorithms only exist under the Linear Gaussian assumption, ii) the algorithms require the crosscovariances of the tracks to be fused, which are generally difficult to obtain and greatly increase the complexity of the algorithms' implementation. For T2TF in practical tracking systems, approximate algorithms with near optimal fusion performance and less complexity are desirable. The IMF has been shown as a good candidate for the purpose and will be discussed in the next section.

3 Impact of Crosscorrelation on the Track-to-Track Fusion without Memory and No Feedback

A simple approximate approach for T2TF is to ignore the crosscovariances among different tracks. However this will lead to over-optimistic fused covariances, which, especially when full information feedback is used, could lead to large errors and filter divergence [16]. Here we use a simple example to show the effect of the crosscovariance on T2TF.

Two sensors tracking the same target, each with its local processor, are considered. The target is modelled by the kinematic model

$$x(k+1) = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 1/2 \\ 1 \end{bmatrix} v(k) \tag{5}$$

This is a nearly constant velocity (NCV) or white noise acceleration (WNA) motion model with unity sampling time with "piecewise constant" process noise (see [2], Sect. 6.3) with variance q entering into the system. A range of values for q will be considered.

The measurements at the two sensors are

$$z^m(k) = [1 \;\; 0]\, x(k) + w^m(k) \qquad m = 1, 2 \tag{6}$$

with the two measurement noises mutually independent and with unity variance.

The maneuvering index for this target (see Eq. (6.5.3-14) in [2]), is thus

$$\lambda = \sqrt{q} \tag{7}$$

Fig. 7 presents in the first column the "2σ" steady-state estimation error ellipses (corresponding to 85% probability mass under Gaussian assumption) in the state space x_2-x_1 (velocity-position) *for each sensor*. The values of q considered are 0.01, 0.1, 0.5, 1, 2 and 5. These correspond to maneuvering indices from 0.1 to $\sqrt{5}$.

The second column presents the error ellipses one obtains by combining these estimates *assuming they are independent*. These ellipses are exactly half (in area) compared to the single-sensor error ellipses — they are, however, optimistic because the half variance (in each coordinate) is *unachievable*.

The third column presents the exact ellipses corresponding to the combined estimates obtained by *taking into account the dependence* between the two tracks due to the common process noise, quantified by the crosscovariance [6].

Fig. 8 shows the ratio of the areas of the ellipses of uncertainty of

- the fused estimates from the two sensors *accounting for their dependence* and
- the single sensor estimates

for various values of the target maneuvering index. This ratio would be equal to 1/2 *in the absence of the dependence*. However, due to the dependence, it is around 0.7 for a wide range of process noise variances.

Fig. 9 shows the decreases in the elements of the covariance matrix for the same situation — as before, in the absence of dependence, each element would be halved after fusion.

For linear systems the crosscorrelations are positive and ignoring them leads to optimistic covariance for the fused estimates.

4 The Information Matrix Fusion

This section reviews the information fusion algorithm [10,8,15,11,6] and its extensions. The IMF operates similarly to the Information Matrix form of the KF — the Information Filter — and, consequently, it is designated as Information Matrix Fusion (IMF). The following versions of the IMF will be discussed, which are the IMF with full communication rate (IMFfcr), the IMF with reduced communication rate (IMFrcr) and the Generalized IMF (GIMF) for asynchronous T2TF (AT2TF) in the presence of communication delay.

4.1 IMF with Full Communication Rate – IMFfcr

The IMF, when operating at full rate, is equivalent to the optimal CTF [6]. The fused (central) estimate follows (for simplicity, N synchronized local trackers are assumed here)

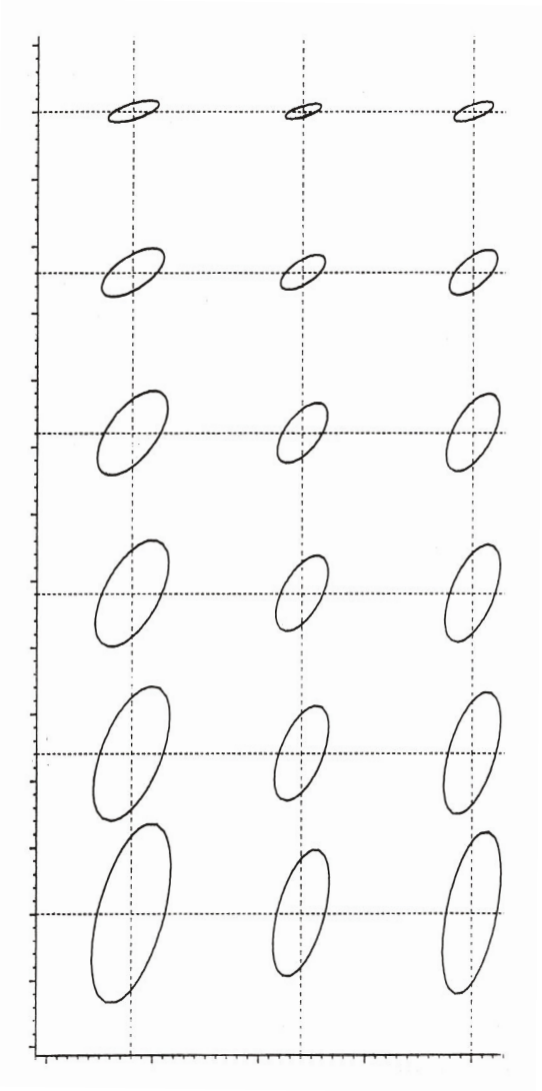


Fig. 7. Error ellipses for various levels of maneuvering index (Column 1: for each sensor alone; Column 2: for two sensors combined assuming independence; Column 3: for two sensors combined correctly)

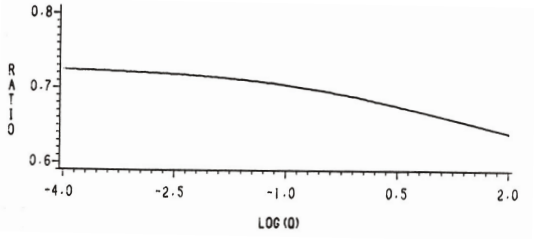


Fig. 8. Decrease in the error ellipse area in view of the dependence of the errors in the fusion process

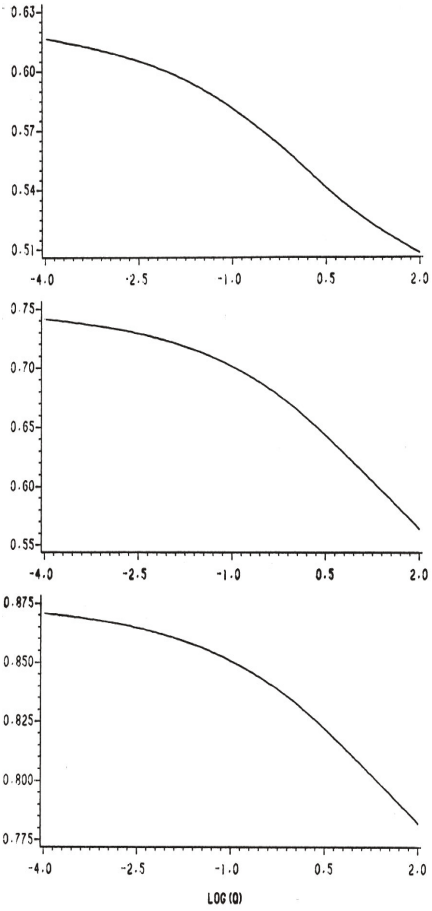


Fig. 9. Decrease of the state covariance matrix elements after fusion in view of the dependence of the errors

$$P(k|k)^{-1}\hat{x}(k|k) = P(k|k-1)^{-1}\hat{x}(k|k-1) + \sum_{i=1}^{N}[P^i(k|k)^{-1}\hat{x}^i(k|k) - P^i(k|k-1)^{-1}\hat{x}^i(k|k-1)] \quad (8)$$

The updated fused covariance needed above is obtained as

$$P(k|k)^{-1} = P(k|k-1)^{-1} + \sum_{i=1}^{N}[P^i(k|k)^{-1} - P^i(k|k-1)^{-1}] \quad (9)$$

In IMFfcr each local estimate/covariance has to be available at the center, i.e., *full communication rate* is necessary. The implementation of information feedback to local trackers is trivial in this configuration. Also note that the IMF is algebraically equivalent to the equivalent measurement approach [6].

4.2 IMF with Reduced Communication Rate – IMFrcr

If the communication occurs only every n sampling times, equations (8)–(9) are used with the following modification. The fused (central) estimate follows from

$$P(k|k)^{-1}\hat{x}(k|k) = P(k|k-n)^{-1}\hat{x}(k|k-n) + \sum_{i=1}^{N}[P^i(k|k)^{-1}\hat{x}^i(k|k) - P^i(k|k-n)^{-1}\hat{x}^i(k|k-n)] \quad (10)$$

The updated central covariance needed above is obtained in terms of the local covariances as

$$P(k|k)^{-1} = P(k|k-n)^{-1} + \sum_{i=1}^{N}[P^i(k|k)^{-1} - P^i(k|k-n)^{-1}] \quad (11)$$

It should be noted that the above is *no longer equivalent to the CTF* — the modified equations (10)–(11) are heuristic. As shown in [9], with full information feedback, the IMFrcr diverges for extremely large values of process noise variance. However, for practical levels of process noises, the IMFrcr was shown to have consistent and close optimal fusion performance. Compared to the exact T2TF fusion algorithms, the IMFrcr does not require the evaluation of the crosscovariances between the tracks to be fused, which significantly simplifies the implementation and makes it a good candidate for T2TF in practical tracking systems. Next the generalization of the IMF for the problem of asynchronous T2TF will be discussed.

4.3 Generalized Information Matrix Fusion for Asynchronous T2TF

The T2TF algorithms mentioned above assume that the local tracks are synchronized. In practical distributed tracking systems, the synchronicity assumption

can hardly be satisfied, which raises the problem of asynchronous track-to-track fusion (AT2TF). In addition local tracks arrive at the FC with transmission delays, which further complicates the fusion problem and the implementation of information feedback. To address this problem, a generalized IMF (GIMF) for AT2TF was presented in [18], and is reviewed next.

Without loss of generality consider the fusion of tracks from two asynchronous local trackers 1 and 2. Tracker 1 is collocated with the FC (no communication delay and information feedback to tracker 1). Tracker 2 is a remote tracker (with communication delay; no feedback to it). Suppose at the fusion time t_f one has

- track $(\hat{x}^1(t_f|t_f),\ P^1(t_f|t_f))$ from tracker 1 (same as FC) and
- tracks $(\hat{x}^2(t_1|t_1),\ P^2(t_1|t_1))$ and $(\hat{x}^2(t_2|t_2),\ P^2(t_2|t_2))$ from tracker 2, $t_1 < t_2 \leq t_f$,

where t_1 and t_2 are the previous and current communication times from sensor 2.

According to the Generalized Information Matrix fusion (GIMF) the fused track is given by

$$P(t_f)^{-1} = P^1(t_f|t_f)^{-1} + \left[P^2(t_f|t_2)^{-1} - P^2(t_f|t_1)^{-1}\right] \tag{12}$$

$$\begin{aligned} P(t_f)^{-1}\hat{x}(t_f) = {} & P^1(t_f|t_f)^{-1}\hat{x}^1(t_f|t_f) \\ & + \left[P^2(t_f|t_2)^{-1}\hat{x}^2(t_f|t_2) - P^2(t_f|t_1)^{-1}\hat{x}^2(t_f|t_1)\right] \end{aligned} \tag{13}$$

where $\hat{x}(t_f)$ is the fused track at t_f, $P(t_f)$ is its covariance, $\hat{x}^1(t_f|t_f)$ and $P^1(t_f|t_f)$ are the track and its covariance from tracker 1 at the fusion time t_f, $\hat{x}^2(t_f|t_i)$ and $P^2(t_f|t_i)$ are the predicted local track 2 from t_i to the fusion time t_f and the corresponding covariance, $i = 1, 2$.

In the presence of communication delay, information feedback to the remote tracker 2 needs to be carefully handled. See [18] for the details of the implementation for both AT2TF with partial and AT2TF with full information feedback. It was shown that the proposed GIMF based AT2TF algorithms yield consistent and close to optimal fusion results. The following reasons contribute to the applicability of the GIMF:

- The predicted information gain from track 2 quantified by $\left[P^2(t_f|t_2)^{-1} - P^2(t_f|t_1)^{-1}\right]$ in (12), is due to the local measurements from $(t_1\ t_2]$ and can be viewed as approximately independent from the other tracks.
- The subtraction structure of the information gain $\left[P^2(t_f|t_2)^{-1} - P^2(t_f|t_1)^{-1}\right]$ provides a desirable feature that cancels (approximately) its crosscorrelation with other local tracks caused by the common process noises with the use of prediction.

5 Heterogenous Track-to-Track Fusion

The previously discussed T2TF configurations and algorithms assume that the local trackers use the same target state vector. In practical tracking systems, local

trackers may use different motion models and state vectors, due to observability issues (e.g., active vs. passive sensors).

A concrete example of such a situation is when (i) tracker 1 uses an active sensor which is able to obtain range and azimuth measurements (full 2D or 3D position, the latter requires elevation measurements as well) and its target state vector comprises Cartesian position, velocity, etc. (ii) tracker 2 uses a passive sensor with angle only measurements and its target state vector comprises angular position, velocity and possibly acceleration.

5.1 The Heterogenous Track-to-Track Fusion Problem

Without loss of generality, consider the following state-space models

$$\mathbf{x}^i(k+1) = f^i[\mathbf{x}^i(k)] + \mathbf{v}^i(k) \tag{14}$$

$$\mathbf{z}^i(k) = h^i[\mathbf{x}^i(k)] + \mathbf{w}^i(k) \tag{15}$$

at sensor i and

$$\mathbf{x}^j(k+1) = f^j[\mathbf{x}^j(k)] + \mathbf{v}^j(k) \tag{16}$$

$$\mathbf{z}^j(k) = h^j[\mathbf{x}^j(k)] + \mathbf{w}^j(k) \tag{17}$$

at sensor j. In the above, $f^s[\cdot]$ and $h^s[\cdot]$, $s = i, j$, are different and can be nonlinear; $\mathbf{v}^s(\cdot)$ and $\mathbf{w}^s(\cdot)$, $s = i, j$, are the process and measurement noises, respectively.

The state vectors $\mathbf{x}^i$ and $\mathbf{x}^j$ are in different spaces. Let $\mathbf{x}^i$ be the larger dimension state. For example, one can have full Cartesian position and velocity in 2-dimensional space for tracking with an active sensor

$$\mathbf{x}^i = [\ x \quad \dot{x} \quad y \quad \dot{y}\ ]' \tag{18}$$

and $\mathbf{x}^j$ be the smaller dimension state (e.g., angular position and velocity for tracking with a passive sensor)

$$\mathbf{x}^j = [\ \theta \quad \dot{\theta}\ ]' \tag{19}$$

These state vectors, in general, have the nonlinear relationship

$$\mathbf{x}^j \stackrel{\Delta}{=} g(\mathbf{x}^i) \tag{20}$$

The two sensors are assumed synchronized[3] and the time index k for sampling time t_k will be omitted if there is no ambiguity.

[3] Generalization to asynchronous sensors is possible [17], but the notations become very cumbersome. Without considering the crosscovariance matrix, the extension to asynchronous case is straightforward. If the crosscovariance matrix is considered (for the configuration with no memory at the FC and no information feedback to the local trackers), each track's latest estimate available at the FC is predicted to the fusion time and then they are fused using the appropriate covariance matrices.

The corresponding estimates (approximate conditional means) at these heterogeneous local sensors are $\hat{\mathbf{x}}^i$ with (conditional) covariance matrix (the conditioning is omitted for brevity)

$$P^i \triangleq E[(\mathbf{x}^i - \hat{\mathbf{x}}^i)(\mathbf{x}^i - \hat{\mathbf{x}}^i)'] \tag{21}$$

and $\hat{\mathbf{x}}^j$ with (conditional) covariance matrix

$$P^j \triangleq E[(\mathbf{x}^j - \hat{\mathbf{x}}^j)(\mathbf{x}^j - \hat{\mathbf{x}}^j)'] \tag{22}$$

The problem is how to carry out the fusion of the estimate $\hat{\mathbf{x}}^i$ with covariance P^i and the estimate $\hat{\mathbf{x}}^j$ with covariance P^j to achieve better estimation performance for the full state of interest $\mathbf{x}^i$.

5.2 The LMMSE Fuser

The approach to HT2TF discussed next is to use the linear technique based on the fundamental equations of LMMSE estimation [2]. Considering the full state estimate $\hat{\mathbf{x}}^i$ as the prior and the smaller state estimate $\hat{\mathbf{x}}^j$ as the measurement, we have the LMMSE fused estimate

$$\hat{\mathbf{x}}^i_{\text{LMMSE}} = \hat{\mathbf{x}}^i + P_{\mathbf{xz}} P_{\mathbf{zz}}^{-1} \left[\hat{\mathbf{x}}^j - g\left(\hat{\mathbf{x}}^i\right)\right] \tag{23}$$

with the corresponding fused covariance matrix

$$P^i_{\text{LMMSE}} = P^i - P_{\mathbf{xz}} P_{\mathbf{zz}}^{-1} P'_{\mathbf{xz}} \tag{24}$$

where (the details can be found in [19])

$$\begin{aligned} P_{\mathbf{xz}} &\triangleq E\left[\left(\mathbf{x}^i - \hat{\mathbf{x}}^i\right)\left(\hat{\mathbf{x}}^j - g(\hat{\mathbf{x}}^i)\right)'\right] \\ &\approx P^i (G^i)' - P^{ij} \qquad (25) \\ P_{\mathbf{zz}} &\triangleq E\left[\left(\hat{\mathbf{x}}^j - g(\hat{\mathbf{x}}^i)\right)\left(\hat{\mathbf{x}}^j - g(\hat{\mathbf{x}}^i)\right)'\right] \\ &\approx P^j - G^i P^{ij} - P^{ji}(G^i)' + G^i P^i (G^i)' \qquad (26) \end{aligned}$$

with G^i the Jacobian of $g(\mathbf{x}^i)$

$$G^i \triangleq \left[\nabla_{\mathbf{x}^i} g(\mathbf{x}^i)'\right]'_{\mathbf{x}^i = \hat{\mathbf{x}}^i} \tag{27}$$

and P^{ij} the crosscovariance matrix

$$P^{ij} \triangleq E[(\mathbf{x}^i - \hat{\mathbf{x}}^i)(\mathbf{x}^j - \hat{\mathbf{x}}^j)'] \tag{28}$$

5.3 The ML Fuser

Under the Gaussian assumption, the heterogeneous T2TF problem can be solved by minimizing the negative log-likelihood function[4] (NLLF) based on (20). The LF of $\mathbf{x}^i$ (the full state) is [6]

$$\Lambda(\mathbf{x}^i) = p(\hat{\mathbf{x}}^i, \hat{\mathbf{x}}^j | \mathbf{x}^i) = \mathcal{N}\left\{ \begin{bmatrix} \hat{\mathbf{x}}^i \\ \hat{\mathbf{x}}^j \end{bmatrix}; \begin{bmatrix} \mathbf{x}^i \\ \mathbf{x}^j \end{bmatrix}, P^{-1} \right\} \tag{29}$$

where

$$P = \begin{bmatrix} P^i & P^{ij} \\ P^{ji} & P^j \end{bmatrix} \tag{30}$$

Then the ML fused estimate is the solution of

$$\nabla_{\mathbf{x}^i} L(\mathbf{x}^i) = 0 \tag{31}$$

where

$$L(\mathbf{x}^i) \stackrel{\Delta}{=} -\ln \Lambda(\mathbf{x}^i) \tag{32}$$

Because of the nonlinearity of the function $g(\mathbf{x}^i)$, there is no explicit expression for the solution of (31). Consequently, (32) is minimized by a numerical search, e.g., the gradient projection algorithm. The result is denoted as $\hat{\mathbf{x}}^i_{\mathrm{ML}}$ and the corresponding covariance matrix is

$$P^i_{\mathrm{ML}} = \left(\begin{bmatrix} I & G^i \end{bmatrix} P^{-1} \begin{bmatrix} I \\ G^i \end{bmatrix} \right)^{-1} \tag{33}$$

where G^i is defined in (27) and I is the identity matrix (4×4 in our case).

6 Example

Fig. 10 shows an example scenario (see [19] for details), where the active sensor is located at $\left(-6 \times 10^4,\ 2 \times 10^4\right)$ m with sampling interval $T_a = 5$ s and the passive sensor located at $\left(-5 \times 10^4,\ 4 \times 10^4\right)$ m with sampling interval $T_p = 1$ s. Measurement noises from the two sensors are assumed to be mutually independent zero mean white Gaussian noises with standard deviations $\sigma_r = 20$ m, $\sigma_a = 5$ mrad for the active sensor, and $\sigma_p = 0.5$ mrad for the passive sensor.

The tracker at the active sensor uses an IMM estimator with two modes: mode 1, a linear nearly constant acceleration (NCV) model [2], and mode 2, a nonlinear nearly coordinate turn (NCT) model [2]. The tracker at the passive sensor uses a linear KF (rather than IMM estimator, because target maneuvers are practically unobservable by the passive sensor). The motion model is the discretized continuous Wiener process acceleration (CWPA) model (with angle, angle rate and angle acceleration).

In [19] it was observed that for this (nonlinear) HT2TF problem

[4] As it is pointed out in [8], the LMMSE T2TF approach is, in the linear Gaussian case, optimal in ML sense.

- The crosscorrelation of the estimation errors from heterogeneous local trackers is too complicated to capture — it can be *positive or negative*. The estimation errors' crosscorrelation has been examined by MC simulations.
- When using a LMMSE fuser, neglecting the track crosscovariance in HT2TF leads to sometimes optimistic, sometimes pessimistic fused covariance. Note this is different from the homogenous T2TF where neglecting the crosscovariance between the local tracks will always result in optimistic fused covariance. This is because for linear systems the crosscorrelation coefficients are always positive.
- When the configuration of fusion without memory and no information feedback is used, neglecting the track crosscovariance is a reasonable practical choice, which yields little loss in fusion performance.

The results using $\hat{\mathbf{x}}^i_{\text{LMMSE}}$ with P^i_{LMMSE} and $\hat{\mathbf{x}}^i_{\text{ML}}$ with P^i_{ML} were found to be practically the same so only the former is compared with the CTF which processes all the measurements (from both the active and the passive sensor) in the FC.

Fig. 11 compares the position RMSE of the HT2TF algorithm to that of the CTF IMM tracker from 400 Monte Carlo runs. Surprisingly, the results show that, in the scenario considered, the HT2TF is superior to the centralized IMM tracker (CTF IMM) during the maneuver periods when the latter experiences a spike in its error.

To explain this apparently counterintuitive result, Fig. 12 shows the maneuvering mode probabilities (NCT) in the active sensor IMM and CTF IMM. It turns out that the use of the passive measurements in the CTF IMM "clouds" the maneuvers because of low maneuvering index, which leads to the degraded

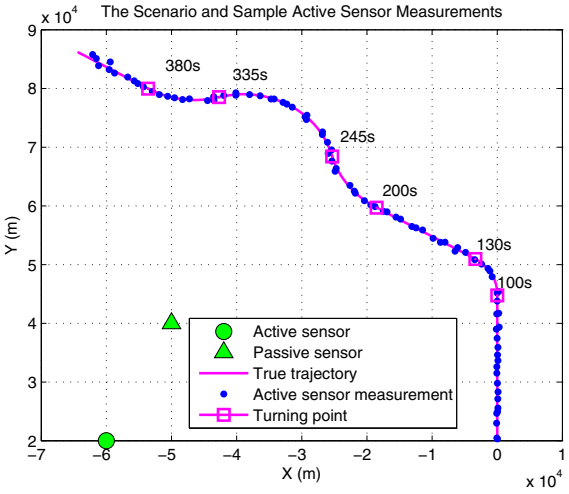


Fig. 10. The scenario, with the target true speed 250 m/s

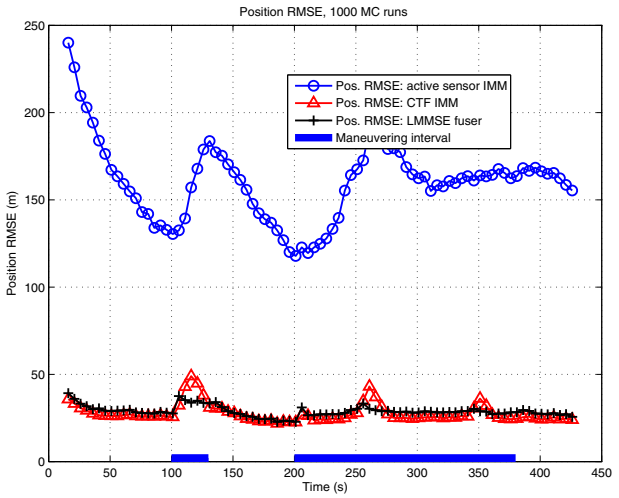


Fig. 11. Performance comparison: HT2TF (with LMMSE fuser) vs. CTF IMM in position RMSE

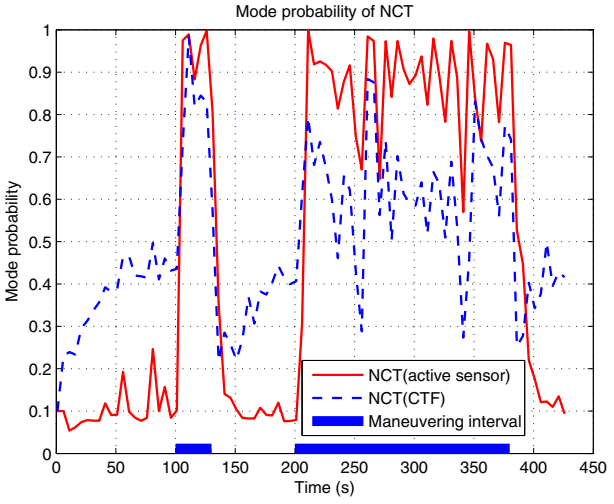


Fig. 12. Maneuvering mode probability (NCT) in the active sensor's IMM and CTF IMM

performance of the CTF IMM filter. Specifically, as can be seen from Fig. 12, when there is a maneuver, the maneuvering mode probability in the CTF IMM rises slower and to a lower level than in the active sensor's IMM. (Note that in this example the passive sensor has a higher sampling rate than the active sensor with $T_a = 5\,\mathrm{s}$ and $T_p = 1\,\mathrm{s}$.) In the scenario considered, the freedom available to

each local sensor to flexibly design a more suitable local estimator is what allows the heterogeneous T2TF approach to achieve a better estimation performance than the CTF IMM.

7 Conclusions

This chapter discusses the various architectures for track-to-track fusion (T2TF). Based on whether the track estimates from the previous fusion are used by the current fusion and the presence of information feedback, T2TF can be further categorized into six configurations, namely, T2TF without memory with no, partial and full information feedback, and T2TF with memory with no, partial and full information feedback.

T2TF without Memory (T2TFwoM) uses only the latest local tracks. It has a small loss in fusion accuracy compared to the centralized tracker/fuser (CTF) regardless of fusion rate. It was shown that information feedback has a negative impact on the accuracy of T2TFwoM.

T2TF with Memory (T2TFwM) uses both the latest local tracks and the track estimates from the previous fusion. It is algebraically equivalent to the CTF when operating at full rate. At reduced rates, it has a slight loss in fusion accuracy compared to the CTF. In contrast to T2TFwoM, information feedback has a positive impact on the fusion accuracy of T2TFwM.

The Information Matrix Fusion (IMF) is a special form of T2TFwM. It is equivalent to the CTF at full rate. At reduced rates it is heuristic but performs well for practical levels of process noises: it has near optimal fusion performance and is consistent. For the fusion of asynchronous tracks in the presence of communication delay, fusion algorithms were developed based on a generalized IMF, which are easy to implement, support the use of information feedback and yield consistent, close to optimal fusion results.

The Heterogenous Track-to-Track Fusion (HT2TF) problem involves the fusion of tracks in different state spaces. Unlike in the conventional T2TF, the crosscorrelations of local tracks have indefinite impact on the fuser-calculated accuracy of the fused track. When an IMM tracker is used, HT2TF allows each local sensor to flexibly design a more suitable local estimator which can lead to a better estimation performance than the CTF.

References

1. Bar-Shalom, Y., Li, X.R.: Multitarget-Multisensor Tracking: Principles and Techniques. YBS Publishing (1995)
2. Bar-Shalom, Y., Li, X.R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation: Algorithms and Software for Information Extraction. Wiley (2001)
3. Bar-Shalom, Y.: On the Track-to-Track Correlation Problem. IEEE Trans. on Automatic Control 26(2), 571–572 (1981)

4. Bar-Shalom, Y., Campo, L.: The Effect of the Common Process Noise on the Two-Sensor Fused-Track Covariance. IEEE Trans on Aerospace and Electronic Systems 22(6), 803–804 (1986)
5. Bar-Shalom, Y.: On Hierarchical Tracking for the Real World. IEEE Trans. on Aerospace and Electronic Systems 42(3), 846–850 (2006)
6. Bar-Shalom, Y., Willet, P.K., Tian, X.: Target Tracking and Data Fusion: A Handbook of Algorithms. YBS Publishing (2011)
7. Challa, S., Legg, J., Wang, X.: Track-to-Track Fusion of Out-of-Sequence Tracks. In: Proc. 5th International Conference on Information Fusion, pp. 919–926 (2002)
8. Chang, K.C., Saha, R.K., Bar-Shalom, Y.: On Optimal Track-to-Track Fusion. IEEE Transactions on Aerospace and Electronic Systems 33(4), 1271–1276 (1997)
9. Chang, K.C., Tian, Z., Saha, R.: Performance Evaluation of Track Fusion with Information Matrix Filter. IEEE Trans. on Aerospace and Electronic Systems 38(2), 455–466 (2002)
10. Chong, C.Y.: Hierarchical Estimation. In: Proc. MIT/ONR Workshop on C3, Monterey, CA (1979)
11. Chong, C.Y., Mori, S., Chang, K.C.: Distributed Multitarget Multisensor Tracking. In: Bar-Shalom, Y. (ed.) Multitarget-Multisensor Tracking: Advanced Applications, ch. 8. Artech House, MA (1990)
12. Li, X.R., Zhu, Y.M., Wang, J., Han, C.Z.: Unified Optimal Linear Estimation Fusion–PartI: Unified Model and Fusion Rules. IEEE Transactions on Information Theory 49(9), 2192–2207 (2003)
13. Mallick, M., Schimdt, S., Pao, L.Y., Chang, K.C.: Out-of-sequence track filtering using the decorrelated pseudo measurement approach. In: Proc. SPIE Conf. on Signal and Data Processing for Small Targets, vol. 5428(1), pp. 154–166 (2004)
14. Novoselsky, A., Sklarz, S.E., Dorfan, M.: Track to track Fusion using Out-of-Sequence Track Information. In: Proc. 10th International Conference on Information Fusion, Quebec City, Canada (2007)
15. Speyer, J.L.: Computation and Transmission Requirements for a Decentralized Linear-Quadratic-Gaussian Control Problem. IEEE Transactions on Automatic Control 24(2), 54–57 (1979)
16. Tian, X., Bar-Shalom, Y.: Track-to-Track Fusion Configurations and Association in a Sliding Window. J. Advances in Information Fusion 4(2), 146–164 (2009)
17. Tian, X., Bar-Shalom, Y.: The optimal algorithm for asynchronous track-to-track fusion. In: Proc. SPIE Conference on Signal and Data Processing of Small Targets. #7698-46, Orlando, FL (2010)
18. Tian, X., Bar-Shalom, Y.: Algorithms for Asynchronous Track-to-Track Fusion. J. Advances in Information Fusion 5(2), 128–138 (2010)
19. Yuan, T., Bar-Shalom, Y., Tian, X.: Heterogeneous Track-to-Track Fusion. J. Advances in Information Fusion 6(2), 131–149 (2011)

State Dependent Difference Riccati Equation Based Estimation for 2D Spiraling Target Maneuver

Ilan Rusnak[1,2,*,†] and Liat Peled-Eitan[1,*]

[1] RAFAEL, P.O. Box 2250, Haifa 31021, Israel
[2] Faculty of Electrical Engineering, Technion, Haifa 32000, Israel
{ilanru,liatap}@rafael.co.il

Abstract. Estimation of target maneuvers with unknown varying turning rate is considered. The modeling of the target's equations of motion takes into account the rotation of the velocity and acceleration vectors as the target maneuvers. This is more complex target maneuver than the simple turning maneuver. The inclusion of the more detailed kinematic behavior of the maneuvering target creates nonlinear equations of motion. The position, velocity, acceleration, jerk, angular rate and angular acceleration of the velocity vector are estimated. This is done without inclusion of the angular rate and angular acceleration into the state vector, but in separate equations. As the equations of motion are nonlinear the State Dependent Differential-Difference Riccati Equation based estimator (SDDRE) is implemented and compared to the Kalman Filter based on the constant-step acceleration target maneuver model. It is demonstrated via simulations for the two dimensional spiraling target maneuvers that the detailed modeling of the maneuvering target based filter-estimator has improved performance with respect to the Kalman Filter based on the constant-step acceleration target maneuver model.

1 Introduction

The issue of estimating a maneuvering target is widely treated subject. A comprehensive survey of models and estimators is presented in [1-5]. The simplest approach is to implement three independent Constant-Step acceleration filters (CA) or Exponentially Correlated Acceleration (ECA) filters [6], one filter for each coordinate.

However these filters may not achieve the required performance for spiraling target maneuvers as they are not matched to these maneuvers, i.e. steady state errors are created.

For more advanced estimators it has been understood that incorporating detailed information on the target dynamics and kinematics into the estimator's equations has the potential to increase the quality of estimation. However, the inclusion of more detailed target maneuver model and the related constraints lead to nonlinear models.

[*] Research Fellow, RAFAEL.
[†] Adjunct Senior Lecturer, Faculty of Electrical Engineering, Technion.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_3

Thus the Kalman Filter is not directly applicable. The most common approach to deal with nonlinear systems is the Extended Kalman Filter (EKF).

In [7] the issue of pseudo measurements had been introduced and the Extended Kalman Filter was applied. In [7] it was pointed out that inclusion of a constraint is usually difficult to incorporate into the dynamic equation and it is much easier to incorporate them into the measurement equations.

In [29] filtering techniques for spiraling target with almost constant angular rate and drag are considered.

In this paper spiraling target maneuvers [9, 26] with unknown angular rate and angular acceleration are considered. One of the main and important issues is the estimation of the angular rate and angular acceleration. This is needed for achieving better matching of the estimator to the target maneuver. Derivation of high performance guidance law for this type of evading maneuver see [8].

Although there are many publication on estimation of barrel roll target maneuvers, i.e. constant turning rate, there are no publication dealing with estimation of spiraling target maneuvers with unknown varying angular rate and angular acceleration.

There are several options to model Spiraling Maneuver. Following [9] the target's spiraling maneuver is modeled as a constant absolute value (modulus) velocity rotating vector with increasing rotation frequency. This creates the acceleration and frequency profile as described in [9, figure 5].

The current approaches to estimation of nonlinear systems include many methods. A comprehensive survey of such methods applied to maneuvering target estimation is presented in [1-5]. For example: in [10] a multiple model approach is applied; in [11] algebraic constraint is incorporated in the state equations; and in [12] the IMM approach is applied for estimation of maneuvering target.

In this paper the State Dependent Differential-Difference Riccati Equation (SDDRE) based estimator is applied to the nonlinear equations of motion based on onboard sensors. The authors are unaware of any publication estimating a maneuvering target by the SDDRE method. The SDDRE approach is very intuitive, although it is not optimal as shown for the State Dependent Algebraic Riccati Equation (SDARE) approach in [13-16]. In [27, 28] the SDARE has been used for angular rate estimation of satellites. The optimal filter requires additional terms for optimality [14-17]. Albeit the sub-optimality of the direct SDDRE approach it is known that the SDDRE based estimator is BIBO stable [18].

The novelty in this paper is:

i) Modeling of spiraling target maneuver based on the rotating constant modulus velocity vector with increasing frequency;
ii) Derivation of the differential equations of the Snap (derivative of the Jerk);
iii) Estimation of the turning rate and turning acceleration;
iv) Application of the SDDRE to estimation of a spiraling target maneuver.

The mathematical derivations are presented in the three dimensional space, however for simplicity the simulations are presented in two dimensions, i.e. planar target's maneuver. Simulations show that estimator based on the rotating velocity

kinematics equations of motion give better performance than the Kalman Filter based on constant acceleration target model. These simulations demonstrate the effectiveness of the proposed SDDRE based estimator.

2 Problem Statement and Approach to the Solution

The problem considered here is the state estimation of the nonlinear stochastic system

$$\begin{aligned} \dot{x}(t) &= f\big(x(t)\big) + \Gamma w(t), \quad x(t_o) = x_o, \\ z(t) &= g\big(x(t)\big) + v(t) \end{aligned} \tag{2.1}$$

where $x(t)$ is the state vector, $z(t)$ is the measurement, $w(t)$, $v(t)$ are the white Gaussian stochastic processes representing the system driving noise and the measurement noise, respectively, $x(t_o)$ is a Gaussian random vector, and

$$\begin{aligned} &E[x_o] = \bar{x}_o, E[w(t)] = 0, E[v(t)] = 0, \\ &E[w(t)w(\tau)^T] = W\delta(t-\tau), E[v(t)v(\tau)^T] = V\delta(t-\tau), \\ &E[w(t)v(\tau)^T] = 0, E[w(t)x_o^T] = 0, E[v(t)x_o^T] = 0, \\ &E[(x_o - E(x_o))(x_o - E(x_o)^T] = Q_o. \end{aligned} \tag{2.2}$$

All vectors and matrices are of appropriate dimensions.

The problem being considered here is finding the optimal estimate $\hat{x}(t)$ as a functional of $\{z(t),\ t_o \le t \le t_f\}$ that minimizes the quadratic criterion:

$$J = E\Big\{\big[x(t) - \hat{x}(t)\big]^T \Theta \big[x(t) - \hat{x}(t)\big]\Big\}, \quad \Theta > 0. \tag{2.3}$$

2.1 Estimator for Linear System

For the linear system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + \Gamma w(t), \quad x(t_o) = x_o, \\ z(t) &= Cx(t) + v(t) \end{aligned} \tag{2.4}$$

The solution of the preceding problem is the Kalman filter [19, 20].

$$\begin{aligned} &\dot{\hat{x}}(t) = A\hat{x}(t) + K(t)[z(t) - C\hat{x}(t)], \quad \hat{x}(t_o) = \bar{x}_o, \\ &K(t) = Q(t)CV^{-1} \\ &\dot{Q}(t) = AQ(t) + Q(t)A^T + \Gamma W \Gamma^T - Q(t)C^T V^{-1} CQ(t), Q(t_o) = Q_o. \end{aligned} \tag{2.5}$$

2.2 Estimators for Nonlinear System

For nonlinear systems there are several approaches. Here the State Dependent Riccati Equation (SDRE) [13-16] approach is considered. The SDRE approach is based on the dual of the SDRE based nonlinear control [16]. This approach parameterizes the state equation (2.1) into a linear structure called the State Dependent Coefficient Form, and is also called Extended Linearization. This approach includes the State Dependent Algebraic Riccati Equation (SDARE) based estimation and the State Dependent Differential-Difference Riccati Equation (SDDRE) based estimation. Then for linear measurement of the state, i.e. $g\big(x(t)\big)=Cx(t)$, the state equations (2.5) can be represented in the State Dependent Coefficients Form (SDC form) as

$$\begin{aligned}&\dot{x}(t)=A_s\big(x(t)\big)x(t)+\Gamma w(t),\quad x(t_o)=x_o,\\&z(t)=Cx(t)+v(t)\end{aligned}\tag{2.6}$$

The SDRE/SDARE based estimator is [13]

$$\begin{aligned}&\dot{\hat{x}}(t)=A_s\big(\hat{x}(t)\big)\hat{x}(t)+K\big(\hat{x}(t)\big)\big[z(t)-C\hat{x}(t)\big],\quad \hat{x}(t_o)=\bar{x}_o,\\&K\big(\hat{x}(t)\big)=Q(t)CV^{-1}\\&0=A_s\big(\hat{x}(t)\big)Q(t)+Q(t)A_s^T\big(\hat{x}(t)\big)+\Gamma W\Gamma^T-Q(t)C^TV^{-1}CQ(t),\ \ Q(t_o)=Q_o.\end{aligned}\tag{2.7}$$

This is a suboptimal estimator. The optimal estimator has additional terms as detailed in [14,15,16]. Here the discrete version of the following SDDRE based state estimator [18] is implemented.

$$\begin{aligned}&\dot{\hat{x}}(t)=A_s\big(\hat{x}(t)\big)\hat{x}(t)+K\big(\hat{x}(t),t\big)\big[z(t)-C\hat{x}(t)\big],\quad \hat{x}(t_o)=\bar{x}_o,\\&K\big(\hat{x}(t),t\big)=Q(t)CV^{-1}\\&\dot{Q}(t)=A_s\big(\hat{x}(t)\big)Q(t)+Q(t)A_s^T\big(\hat{x}(t)\big)+\Gamma W\Gamma^T-Q(t)C^TV^{-1}CQ(t),\ \ Q(t_o)=Q_o.\end{aligned}\tag{2.8}$$

The difference between the SDARE as applied in [14,15,16] , eq. (2.7), and eq. (2.8), the one implemented in this paper, is that in this paper the differential-difference Riccati equation is used for computation of the gains (2.8) and not the algebraic Riccati equation (2.7) that is solved each sampling interval.

3 The Constant Acceleration Target Maneuver Model

For comparison in this paper the performance of a most common estimator of target motion is presented. This estimator is based on the target Constant Acceleration (CA)

step target maneuver. The following dynamic model of the target-missile encounter [1,2,6] is assumed for the CA target manuver model.

For each inertial coordinate it is assumed that

$$\frac{d}{dt}\begin{bmatrix} x(t) \\ \dot{x}(t) \\ a_T(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} x(t) \\ \dot{x}(t) \\ a_T(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} w_T(t),$$
$$z(t) = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}\begin{bmatrix} x(t) \\ \dot{x}(t) \\ a_T(t) \end{bmatrix} + v(t) \tag{3.1}$$

where

x - target position [m]
$\dot{x}$ - target velocity [m/s]
a_T - target acceleration [m/s^2]
$w_T(t)$ - target process driving noise (jerk) [m/s^3]
$v(t)$ - target measurement noise (glint) [m]
z - measured target position[m]

The preceding assumes that the target performs an evasive maneuver (a stochastic process), that is, a step acceleration maneuver of amplitude a_{T0} whose initiation instant is uniformly distributed in the interval [t_o,t_f]. The continuous shaping filter [1,2,6] of this process is represented by

$$\frac{d}{dt}a_T = w_{Tcontinuous}(t), \tag{3.2}$$

where the spectral density of the target maneuver (the process noise), $w_T(t)$, is

$$W_{AScontinuous} = W_T = \frac{a_{T0}^2}{t_m} \quad [(m/\sec^3)^2/Hz] = [m^2/s^5] \tag{3.3}$$

where

a_{T0} - target step maneuver value [m/s^2]

$t_m = t_f - t_o$ is the time interval over which the target is expected to take an evasive maneuver.

4 Kinematic Equations of Spiraling Target

Comprehensive survey of modeling the behavior of a maneuvering target can be found in [1-5]. Here one specific case is considered.

A redundant set of assumptions with respect to maneuvering target behavior is:

i. Constant absolute value of target velocity, $\|\vec{v}_T\| = constant$;

ii. $\vec{v}_T \cdot \vec{a}_T = 0$, velocity perpendicular to acceleration (aerodynamically controlled aircraft);

iii. $\vec{v}_T \cdot \vec{\omega}_T = 0$, velocity perpendicular to turning rate (angular velocity);

iv. $\vec{v}_T \cdot \dot{\vec{\omega}}_T = 0$, velocity perpendicular to angular acceleration;

v. no loss of target energy (constant altitude).

4.1 The Kinematics as a Function of Velocity

It is assumed that the target's velocity is expressed as

$$\vec{v}_T = v_T 1_v \tag{4.1}$$

where

$\vec{v}_T$ - target velocity vector [m/sec]

v_T - absolute value of the target's velocity

1_v - unit vector in the target velocity direction

A. Acceleration equations. First order equations of motion based on the target's velocity (4.1) are derived. As the target is maneuvering the velocity vector is rotating. The target's acceleration is given by [21]

$$\vec{a}_T = \frac{d\vec{v}_T}{dt} = \dot{v}_T 1_v + \vec{\omega}_T \times \vec{v}_T \tag{4.2}$$

$$\vec{\omega}_T = \frac{\vec{v}_T \times \vec{a}_T}{\|\vec{v}_T\|^2}; \tag{4.3}$$

$\vec{a}_T$ - target acceleration vector [m/s^2]

$\vec{\omega}_T$ - angular rate of the target velocity direction [rad/sec]

B. Jerk equations – VJ. Second order equations of motion based on the targets velocity (4.1) are derived. As the target is maneuvering the velocity and acceleration vectors are rotating. The jerk (derivative of acceleration) is the given by [21,22]. This is called here Velocity based Jerk (VJ) equations of motion. The target's jerk is [21,22]

$$\vec{j}_T = \frac{d\,\vec{a}_T}{dt} = \frac{d^2\,\vec{v}_T}{dt^2} = \ddot{v}_T 1_v + \dot{\vec{\omega}}_T \times \vec{v}_T + 2\left(\vec{\omega}_T \times \dot{v}_T 1_v\right) + \vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right) \quad (4.4)$$

$\vec{j}_T$ - target's jerk vector [m/sec^3]

Substituting $\dot{v}_T 1_v$ from (4.2) gives [22]

$$\begin{aligned} \vec{j}_T &= \ddot{v}_T 1_v + \dot{\vec{\omega}}_T \times \vec{v}_T + 2\vec{\omega}_T \times \frac{d\,\vec{v}_T}{dt} - \vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right) \\ &= \ddot{v}_T 1_v + \dot{\vec{\omega}}_T \times \vec{v}_T + 2\vec{\omega}_T \times \vec{a}_T - \vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right) \end{aligned} \quad (4.5)$$

$\dot{\vec{\omega}}_T$ - angular acceleration of the target velocity direction [rad/s^2]

and

$$\dot{\vec{\omega}}_T = \frac{\vec{v}_T \times \vec{j}_T}{\left\|\vec{v}_T\right\|^2}$$

C. Snap equations – VS. Third order equations of motion based on the targets velocity (4.1) are derived. As the target is maneuvering the velocity, acceleration and jerk vectors are rotating. The snap (derivative of jerk) is derived-presented here. This is called here Velocity based Snap (VS) equations of motion. The target's snap is

$$\begin{aligned} \vec{s}_T &= \frac{d\,\vec{j}_T}{dt} = \frac{d^2\,\vec{a}_T}{dt^2} = \\ &= \dddot{v}_T 1_v + 3\left(\vec{\omega}_T \times \ddot{v}_T 1_v\right) + 3\,\vec{\omega}_T \times \left(\vec{\omega}_T \times \dot{v}_T 1_v\right) + \vec{\omega}_T \times \left(\vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right)\right) \\ &+ 2\,\dot{\vec{\omega}}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right) + \vec{\omega}_T \times \left(\dot{\vec{\omega}}_T \times \vec{v}_T\right) + 3\,\dot{\vec{\omega}}_T \times \dot{v}_T 1_v + \ddot{\vec{\omega}}_T \times \vec{v}_T \end{aligned} \quad (4.6)$$

$\vec{s}_T$ - target's jerk vector [m/s^4]

$\ddot{\vec{\omega}}_T$ - angular jerk of the target velocity direction [rad/s^3]

Substituting $\dot{v}_T 1_v$ from (4.2) and $\ddot{v}_T 1_v$ from (4.4) gives

$$\begin{aligned}\vec{s}_T = \ddot{\vec{v}}_T 1_v + 3\vec{\omega}_T \times \vec{j}_T - 3\vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{a}_T\right) + \vec{\omega}_T \times \left(\vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right)\right) \\ + 3\dot{\vec{\omega}}_T \times \vec{a}_T - 2\vec{\omega}_T \times \left(\dot{\vec{\omega}}_T \times \vec{v}_T\right) - \dot{\vec{\omega}}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right) + \ddot{\vec{\omega}}_T \times \vec{v}_T\end{aligned} \tag{4.7}$$

5 The Variance and Spectrum of the Glint Noise

For simplicity it is assumed assume that the only measurement noise is the glint noise. The standard deviation of the glint noise, σ_g, for uniformly distributed reflectors, is [23]

$$\begin{aligned}\sigma_g^2 &= \frac{1}{12}D^2 \\ D &= \sqrt{\frac{2}{\pi}}L\end{aligned} \tag{5.1}$$

where

D - the effective linear dimension of the target perpendicular to the target-missile LOS.

L - the linear dimension of the target perpendicular to the target-missile LOS.

When frequency agility is applied at rate of $f_s=1/T_s$ [Hz], the spectral density of (stair type random stochastic process - i.i.d. sequence) the glint is given by [24]

$$\begin{aligned}V_{go} &= \sigma_g^2 T_s, \left[\frac{m^2}{Hz}\right] \\ V_g(\omega) &= V_{go}\left[\frac{\sin\left(\frac{\omega T_s}{2}\right)}{\left(\frac{\omega T_s}{2}\right)}\right]^2, \left[\frac{m^2}{Hz}\right]\end{aligned} \tag{5.2}$$

where T_s is the sampling interval of the frequency agile radar. Therefore, the spectral density of the measurement noise, *v(t)*, is $V_{go}\left[m^2/Hz\right]$.

6 Incorporation of the Kinematic Constraint

The kinematic equation in section 4 is a constraint that can be incorporated into the estimator equations. It is possible to incorporate the kinematic constraint into the state equation or measurement equation and the "unknown" quantities are interpreted as either a measurement noise or system driving noise. The approach here has the advantage that the kinematic constraint is incorporated in the system equations, the unknown is the system driving noise and the measurements are linear. The derivation here is in three dimensions although the simulations are performed in two dimensions. The following modeling and the assumptions in section 4 mean that the target's maneuver is modeled as piece-wise constant accelerating turning rate maneuver in target's coordinates.

6.1 Velocity Based Snap Kinematic Equation

With velocity based snap equations (4.6) it is assumed that $\ddot{\vec{\omega}} = 0$. Thus from (4.7) the kinematics of maneuvering target is modeled as

$$\begin{aligned}\vec{s}_T &= 3\vec{\omega}_T \times \vec{j}_T - 3\vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{a}_T\right) + \vec{\omega}_T \times \left(\vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right)\right) \\ &\quad + 3\dot{\vec{\omega}}_T \times \vec{a}_T - 2\vec{\omega}_T \times \left(\dot{\vec{\omega}}_T \times \vec{v}_T\right) - \dot{\vec{\omega}}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right) + w_T \\ w_T &= \ddot{v}_T 1_v + \ddot{\vec{\omega}}_T \times \vec{v}_T\end{aligned} \tag{6.1}$$

$$\vec{\omega}_T = \frac{\vec{v}_T \times \vec{a}_T}{\left\|\vec{v}_T\right\|^2};$$

$$\dot{\vec{\omega}}_T = \frac{\vec{v}_T \times \vec{j}_T}{\left\|\vec{v}_T\right\|^2}$$

where w_T represents the deviation of the actual behavior of the target from the constant angular turning acceleration and constant absolute value of the velocity assumptions.

The state space representation in the State Dependent Coefficient Form is in three dimensions

$$\frac{d}{dt}\begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \\ y \\ \dot{y} \\ \ddot{y} \\ \dddot{y} \\ z \\ \dot{z} \\ \ddot{z} \\ \dddot{z} \end{bmatrix} = \begin{bmatrix} & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & A & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \\ & & & & & & & & & & & \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \\ y \\ \dot{y} \\ \ddot{y} \\ \dddot{y} \\ z \\ \dot{z} \\ \ddot{z} \\ \dddot{z} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} w_x \\ w_y \\ w_z \end{bmatrix}$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{4,2} & 3\left(\omega_y^2+\omega_z^2\right) & 0 & 0 & a_{4,6} & -3\omega_x\omega_y-3\dot{\omega}_z & -3\omega_z & 0 & a_{4,10} & -3\omega_x\omega_z+3\dot{\omega}_y & 3\omega_y \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & a_{8,2} & -3\omega_x\omega_y+3\dot{\omega}_z & 3\omega_z & 0 & a_{8,6} & 3\left(\omega_x^2+\omega_z^2\right) & 0 & 0 & a_{8,10} & -3\omega_z\omega_y-3\dot{\omega}_x & -3\omega_x \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_{12,2} & -3\omega_x\omega_z-3\dot{\omega}_y & -3\omega_y & 0 & a_{12,6} & -3\omega_y\omega_z+3\dot{\omega}_x & 3\omega_x & 0 & a_{12,10} & 3\left(\omega_x^2+\omega_y^2\right) & 0 \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix}_m = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \\ y \\ \dot{y} \\ \ddot{y} \\ y \\ z \\ \dot{z} \\ \ddot{z} \\ \dddot{z} \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

$$
\begin{aligned}
a_{4,2} &= 3\left(\omega_y\dot{\omega}_y + \omega_z\dot{\omega}_z\right) \\
a_{8,2} &= -\left(\left(\omega_x^2 + \omega_y^2 + \omega_z^2\right)\omega_z + 2\omega_x\dot{\omega}_y + \dot{\omega}_x\omega_y\right) \\
a_{12,2} &= -\left(-\left(\omega_x^2 + \omega_y^2 + \omega_z^2\right)\omega_y + 2\omega_x\dot{\omega}_z + \dot{\omega}_x\omega_z\right)
\end{aligned}
$$

$$
\begin{aligned}
a_{4,6} &= -\left(-\left(\omega_x^2 + \omega_y^2 + \omega_z^2\right)\omega_z + 2\dot{\omega}_x\omega_y + \omega_x\dot{\omega}_y\right) \\
a_{8,6} &= 3\left(\omega_x\dot{\omega}_x + \omega_z\dot{\omega}_z\right) \\
a_{12,6} &= -\left(\left(\omega_x^2 + \omega_y^2 + \omega_z^2\right)\omega_x + 2\omega_y\dot{\omega}_z + \dot{\omega}_y\omega_z\right)
\end{aligned}
$$

$$
\begin{aligned}
a_{4,10} &= -\left(\left(\omega_x^2 + \omega_y^2 + \omega_z^2\right)\omega_y + 2\dot{\omega}_x\omega_z + \omega_x\dot{\omega}_z\right) \\
a_{8,10} &= -\left(-\left(\omega_x^2 + \omega_y^2 + \omega_z^2\right)\omega_x + 2\omega_z\dot{\omega}_y + \dot{\omega}_z\omega_y\right) \\
a_{12,10} &= 3\left(\omega_x\dot{\omega}_x + \omega_y\dot{\omega}_y\right)
\end{aligned}
$$

where

$$
\dot{\vec{\omega}}_T = \frac{(\vec{v}_T \times \dot{\vec{a}}_T)}{\left\|\vec{v}_T\right\|^2} = \frac{(\vec{v}_T \times \vec{j}_T)}{\left\|\vec{v}_T\right\|^2} = \frac{1}{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}\begin{bmatrix} \dot{y}\dddot{z} - \dot{z}\dddot{y} \\ \dot{z}\dddot{x} - \dot{x}\dddot{z} \\ \dot{x}\dddot{y} - \dot{y}\dddot{x} \end{bmatrix}
$$

In two dimensions this reduces to

$$\frac{d}{dt}\begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \\ y \\ \dot{y} \\ \ddot{y} \\ \dddot{y} \end{bmatrix} = \left[\begin{array}{cccc:cccc} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 3\omega_z\dot{\omega}_z & 3\omega_z^2 & 0 & 0 & \omega_z^3 & -3\dot{\omega}_z & -3\omega_z \\ \hdashline 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & -\omega_z^3 & 3\dot{\omega}_z & 3\omega_z & 0 & 3\omega_z\dot{\omega}_z & 3\omega_z^2 & 0 \end{array}\right] \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \\ y \\ \dot{y} \\ \ddot{y} \\ \dddot{y} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} w_x \\ w_y \end{bmatrix}$$

$$\begin{bmatrix} x \\ y \end{bmatrix}_m = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \ddot{x} \\ \dddot{x} \\ y \\ \dot{y} \\ \ddot{y} \\ \dddot{y} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \end{bmatrix}$$

where

$$\omega_z = \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{\dot{x}^2 + \dot{y}^2}$$

$$\dot{\omega}_z = \frac{\dot{x}\dddot{y} - \dot{y}\dddot{x}}{\dot{x}^2 + \dot{y}^2}$$

7 Planar Spiraling Target Maneuver Trajectory Model

This section presents a model of a planar spiraling target trajectory. It is called here the Spiraling Evading Maneuver (SEM). The target moves with constant speed. At certain moment it starts a spiraling maneuver at constant speed (absolute value of the velocity) and constant altitude, i.e. in the (x-y) plane only.

The velocity of planar spiraling maneuver, similarly to [25], is modeled as

$$\begin{aligned}\dot{x}_T &= V_T \sin\left(\omega_{To}t + \frac{1}{2}\alpha_T t^2 - \varphi\right)\\ \dot{y}_T &= V_T \cos\left(\omega_{To}t + \frac{1}{2}\alpha_T t^2 - \varphi\right)\\ \dot{z}_T &= 0\end{aligned} \tag{7.1}$$

where

V_T - Speed of the target [m/s]

ω_{To} - The initial turning rate of the target's velocity [rad/s]

α_T - The turning acceleration of the target's velocity [rad/s^2]

φ - Initial phase [rad]

Then the position is given by

$$\begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} = \int_0^t \begin{bmatrix} \dot{x}_T \\ \dot{y}_T \\ \dot{z}_T \end{bmatrix} d\tau + \begin{bmatrix} x_{To} \\ y_{To} \\ z_{To} \end{bmatrix} \tag{7.2}$$

where (x,y,z) are the target's coordinates, and the rest of the variables are self evident.

8 The SDDRE Based Discrete Estimator

The discrete nonlinear stochastic system in the State Dependent Coefficient Form is

$$\begin{aligned}&x(t+1) = A\big(x(t)\big)x(t) + \Gamma w_d(t), \quad x(t_o) = x_o,\\ &z(t) = Cx(t) + v_d(t)\end{aligned} \tag{8.1}$$

where $x(t)$ is the state vector, $z(t)$ is the measurement, $u(t)$ is the input, $w(t)$, $v(t)$ are white Gaussian stochastic sequences representing the system driving noise and the measurement noise, respectively, x_o is a Gaussian random vector, and

$$
\begin{aligned}
&E[x_o]=\bar{x}_o, E[w_d(t)]=0, E[v(t)]=0,\\
&E[w_d(t)w_d(\tau)^T]= W_d\delta_{t\tau}, E[v_d(t)v_d(\tau)^T]=V_d\delta_{t\tau},\\
&E[w_d(t)v_d(\tau)^T]=0, E[w_d(t)x_o^T]=0, E[v_d(t)x_o^T]=0,\\
&E[(x_o-E(x_o))(x_o-E(x_o)^T]=Q_o.
\end{aligned}
\tag{8.2}
$$

All vectors and matrices are of appropriate dimensions.

The State Dependent Difference Riccati Equation (SDDRE) based estimator is for *k=0,1,2,…,* an adaptation of the State Dependent Differential Riccati Equation (SDDRE)

$$
\begin{aligned}
&\hat{x}(k+1) = A(\hat{x}(k))\,\hat{x}(k) + K(k)[z(k) - C\hat{x}(k)], \quad \hat{x}(t_o) = \bar{x}_o,\\
&K(k) = \tilde{Q}(k)C^T\left(C\tilde{Q}(k)C^T + V_d\right)^{-1}\\
&\tilde{Q}(k+1) = A(\hat{x}(k))Q(k)A(\hat{x}(k))^T + \Gamma W_d\Gamma^T, Q(k_o) = Q_o. / \ predictor - prior /\\
&Q(k) = [I - K(k)C]\tilde{Q}(k) / corrector - posterior /
\end{aligned}
\tag{8.3}
$$

9 Simulation Results

In this section the performance of the constant acceleration (CA) (3.1), and the velocity based snap (VS) equation based filters (section 6) are compared via simulations. The simulations are performed against the spiraling evasive maneuver. In effort to perform the comparison on common basis all simulation are performed for system driving noise (process noise) with power spectral density of

$$
\begin{aligned}
W_T\big|_{CA} &= 100\ [(m/s^3)^2/Hz]\\
W_T\big|_{VS} &= 100\ [(m/s^4)^2/Hz]
\end{aligned}
$$

for sampling interval

$$T_s = 10\,m\sec$$

and for measurement noise level of σ_g =3m.

The instantaneous frequency of the spiraling maneuver (7.1) is

$$\omega_z = \omega_{To} + \alpha_T t$$

Table 9.1 presents the parameters of the examples that are presented in this paper. The signal-to-noise ratio is the "power" of the target position divided by the noise power (ratio of half of the squared ratio of the radius of the maneuvering trajectory to the measurement noise standard deviation). However for spiraling maneuver this value changes. Therefore here the initial signal-to-noise ratio, $(S/N)_I$, is used as a measure of the signal-to-noise ratio.

$$\left(\frac{S}{N}\right)_I = \frac{\frac{1}{2}\left(\frac{V_T}{\omega_{To}}\right)^2}{\sigma_g^2};$$

The "bandwidth" of the "SEM" filter can be approximately estimated by

$$\omega_o = \sqrt[8]{\frac{W_T}{V_{go}}} = \sqrt[8]{\frac{W_T\big|_{VS}}{\sigma_g^2 T_s}}\ .$$

Table 9.1. Parameters of the examples

	VT [m/s]	ω_{To} [rad/s]	α_T [rad/s2]	σ_g [m]	W_T $[(m/s^4)^2/Hz]$	$(S/N)_I$ [dB]	ω_o [rad/s]
SEM	40	1	1	3	100	19	2.4
SEM	100	1	1	3		27	2.4

Simulation results in this section are presented for the spiraling evasive maneuver (SEM) for two cases as presented in table 9.1.

Figure 9.1 presents the trajectories of the coordinated turn target maneuver. Figure 9.2 presents the deterministic position tracking error. One can see that the CA filter develops a diverging tracking error while the VS filters have zero steady state error. Figure 9.3 presents the deterministic estimation error of the turning rate (ω) and acceleration (ω dot) of the maneuver. Again the CA filter develops a diverging estimation error while the VS filter has zero steady state estimation error.

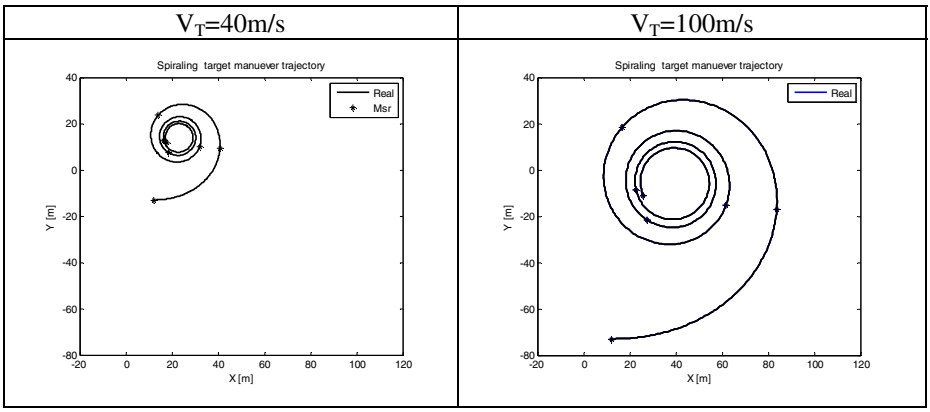


Fig. 9.1. Trajectory of spiraling evasive maneuver

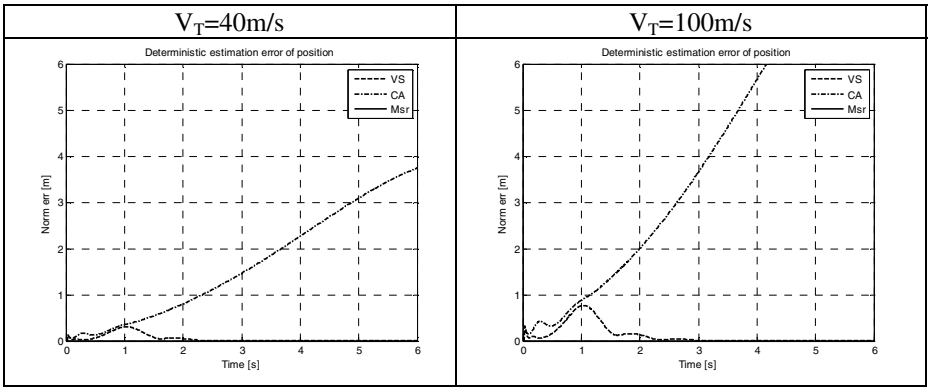


Fig. 9.2. Deterministic tracking position error of spiraling evasive maneuver

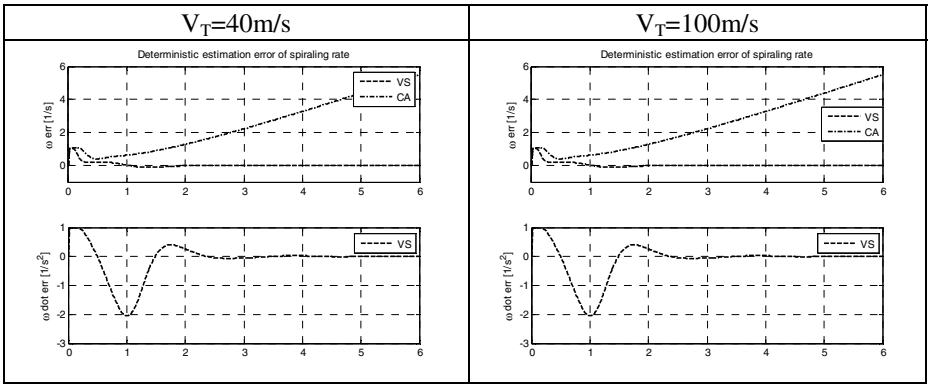


Fig. 9.3. Deterministic estimation error of the turning rate (ω) and acceleration (ω dot) of spiraling evasive maneuver

Figure 9.4 presents the stochastic tracking. One can see that the CA filter develops higher tracking error than the VS filter thus demonstrating the improved performance of the based filter. Figure 9.5 presents the mean of the estimation error of the turning rate (ω) and acceleration (ω dot) of the maneuver with noisy measurements. Again the CA filter develops higher estimation error than the VS based filters.

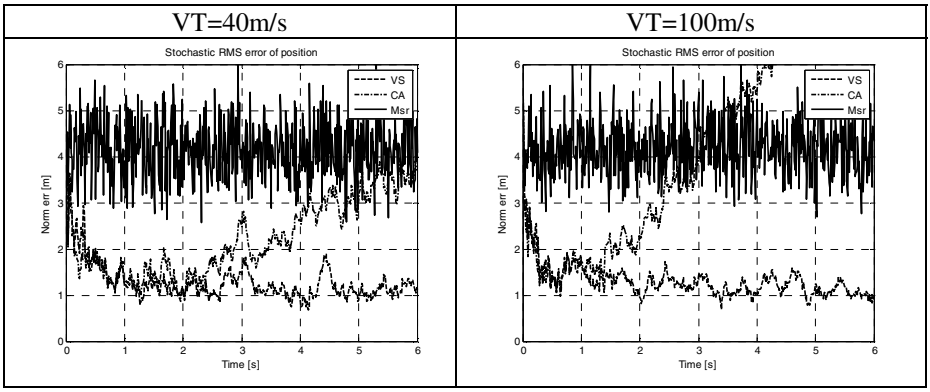


Fig. 9.4. Stochastic RMS tracking position error of spiraling evasive maneuver

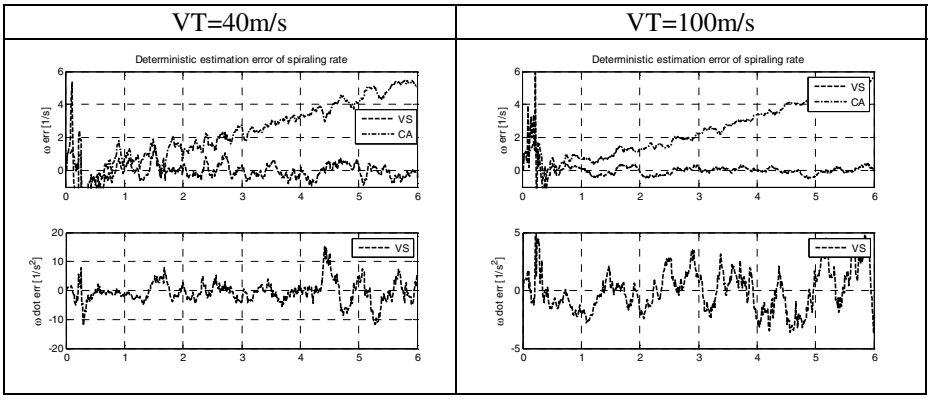


Fig. 9.5. Mean estimation error of the turning rate (ω) and acceleration (ω dot) for spiraling evasive maneuver with noisy measurements

Figure 9.6 presents sample run of the estimates of the state in the x-coordinate for spiraling evasive maneuver with noisy measurements. The y-axis exhibits similar behavior with the corresponding shift of 90 degrees. The better performance of the VS based filters is demonstrated.

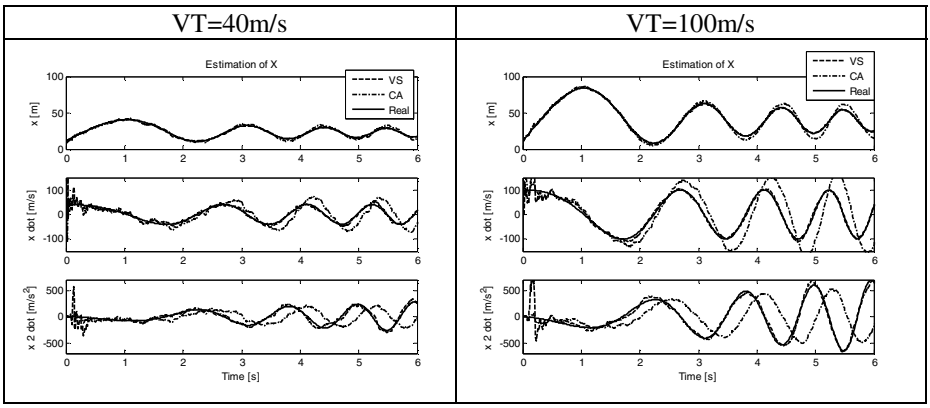


Fig. 9.6. Sample run of the estimates of the state in the x-coordinate for spiraling evasive maneuver with noisy measurements

10 Conclusions

Estimation of evading target maneuvers with unknown varying turning rate is considered. The equations of motion take into account the rotation of the velocity and acceleration vector as the target maneuvers. The inclusion of the more detailed kinematic behavior of the maneuvering target creates nonlinear equations of motion. The estimates of the target's state- position, velocity and acceleration, and the angular rate and acceleration of the velocity vector are derived. This is done without inclusion of the angular rate and acceleration into the state vector. As the equations of motion are nonlinear the State Dependent Differential-Difference Riccati Equation based estimator (SDDRE) is implemented and compared to the Kalman Filter based on the constant-step acceleration target maneuver model. It is demonstrated via simulations that the detailed modeling of the maneuvering target based filter-estimator have improved performance with respect to the Kalman Filter based on the constant-step acceleration target maneuver model.

Appendix

This appendix presents for the completeness of presentation derivation of the formulas of the turning rate and acceleration, together with the assumptions that are presented in section 4.

The turning rate is derived

$$\vec{v}_T = v_T 1_v$$

$$\vec{a}_T = \frac{d\,\vec{v}_T}{dt} = \dot{v}_T 1_v + \vec{\omega}_T \times \vec{v}_T$$

$$\vec{v}_T \times \vec{a}_T = \vec{v}_T \times \left(\dot{v}_T 1_v + \vec{\omega}_T \times \vec{v}_T\right) = 0 + \vec{v}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right)$$
$$= \vec{\omega}_T \left(\vec{v}_T \cdot \vec{v}_T\right) - \vec{v}_T \left(\vec{v}_T \cdot \vec{\omega}_T\right) = \vec{\omega}_T \left\|\vec{v}_T\right\|^2 - \vec{v}_T \left(\vec{v}_T \cdot \vec{\omega}_T\right)$$
$$\vec{\omega}_T = \frac{\vec{v}_T \times \vec{a}_T - \vec{v}_T \left(\vec{v}_T \cdot \vec{\omega}_T\right)}{\left\|\vec{v}_T\right\|^2}$$

if we assume that

$$\vec{v}_T \cdot \vec{\omega}_T = 0$$

then

$$\vec{v}_T \times \vec{a}_T = 0 + \vec{\omega}_T \left\|\vec{v}_T\right\|^2$$
$$\vec{\omega}_T = \frac{\vec{v}_T \times \vec{a}_T}{\left\|\vec{v}_T\right\|^2}$$

The rate of the turning rate (turning acceleration)

$$\vec{v}_T = v_T 1_v$$
$$\vec{a}_T = \frac{d\,\vec{v}_T}{dt} = \dot{v}_T 1_v + \vec{\omega}_T \times \vec{v}_T$$
$$\vec{j}_T = \ddot{v}_T 1_v + \dot{\vec{\omega}}_T \times \vec{v}_T + 2\vec{\omega}_T \times \vec{a}_T - \vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right)$$

$$\vec{v}_T \times \vec{j}_T = \vec{v}_T \times \left(\ddot{v}_T 1_v + \dot{\vec{\omega}}_T \times \vec{v}_T + 2\vec{\omega}_T \times \vec{a}_T - \vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right)\right)$$
$$= v_T 1_v \times \ddot{v}_T 1_v + \vec{v}_T \times \left(\dot{\vec{\omega}}_T \times \vec{v}_T\right) + 2\vec{v}_T \times \left(\vec{\omega}_T \times \vec{a}_T\right) - \vec{v}_T \times \left(\vec{\omega}_T \times \left(\vec{\omega}_T \times \vec{v}_T\right)\right)$$

Then

$$\vec{v}_T \times \vec{j}_T = 0 + \dot{\vec{\omega}}_T \left(\vec{v}_T \cdot \vec{v}_T\right) - \vec{v}_T \left(\vec{v}_T \cdot \dot{\vec{\omega}}_T\right) + 0 - \vec{v}_T \times \left(\vec{\omega}_T \times 0\right)$$
$$= \dot{\vec{\omega}}_T \left\|\vec{v}_T\right\|^2 - \vec{v}_T \left(\vec{v}_T \cdot \dot{\vec{\omega}}_T\right)$$
$$\dot{\vec{\omega}}_T = \frac{\vec{v}_T \times \vec{j}_T + \vec{v}_T \left(\vec{v}_T \cdot \dot{\vec{\omega}}_T\right)}{\left\|\vec{v}_T\right\|^2}$$

assume that $\vec{v}_T \cdot \dot{\vec{\omega}}_T = 0$ *then* $\dot{\vec{\omega}}_T = \frac{\vec{v}_T \times \vec{j}_T}{\left\|\vec{v}_T\right\|^2}$

then $\vec{v}_T \cdot \dot{\vec{\omega}}_T = \frac{\vec{v}_T \cdot \left(\vec{v}_T \times \vec{j}_T\right)}{\left\|\vec{v}_T\right\|^2} = 0$ *Q.E.D.*

References

[1] Li, X.R., Jilkov, V.P.: Survey of Maneuvering Target Tracking. Part I: Dynamic Models. IEEE Trans. on AES 39(4), 1333–1364 (2003), Also: In: Proceedings of the 2000 SPIE Conference on Signal and Data Processing of Small Targets, vol. 4048, pp. 212–234

[2] Li, X.R., Jilkov, V.P.: Survey of Maneuvering Target Tracking. Part II: Motion Models of Ballistic and Space Targets. IEEE Trans. on AES 46(1), 96–119 (2010), Also: In: Proceedings of the 2001 SPIE Conference on Signal and Data Processing of Small Targets, vol. 4473

[3] Li, X.R., Jilkov, V.P.: Survey of Maneuvering Target Tracking. Part III: Measurement Models. In: Proceedings of the 2001 SPIE Conference on Signal and Data Processing of Small Targets, vol. 4473, pp. 423–446 (2001)

[4] Li, X.R., Jilkov, V.P.: Survey of Maneuvering Target Tracking. Part IV: Decision –Based Methods. In: Proceedings of the 2002 SPIE Conference on Signal and Data Processing of Small Targets, vol. 4728, pp. 511–534 (2002)

[5] Li, X.R., Jilkov, V.P.: Survey of Maneuvering Target Tracking. Part V: Multiple-Models Methods. IEEE Trans. on AES 41(4), 1255–1321 (2005)

[6] Fitzgerald, R.J.: Simple tracking filters: Closed-form solutions. IEEE Trans. on Aerospace and Electronic Systems AES-17, 781–785 (1981)

[7] Tahk, M., Speyer, J.L.: Target Tracking Problems Subject to Kinematic Constraints. IEEE Trans. on AC 35(3), 324–326 (1990)

[8] Aggarwal, R.K.: Optimal Missile Guidance for Weaving Targets. In: Proceedings of the 35th Conference on Decision and Control, Kobe, Japan, pp. 2775–2779 (December 1997)

[9] Chadwick, W.R., Zarchan, P.: Interception of Spiraling Ballistic Missiles. In: Proceedings of the American Control Conference, ACC 1996, Seattle, WA, pp. 4476–4483 (June 1996)

[10] Marks, G.M.: Multiple Models Adaptive Estimation (MMAE) for Improving Guidance Performance against Weaving Targets. In: AIAA Guidance, Navigation and Control Conference, Keystone, CO, August 21-24, pp. 2006–6697. AIAA (2006)

[11] Chen, R.H., Speyer, J.L., Lianos, D.: Homing Missile Guidance and Estimation under Agile Target Acceleration. Journal of Guidance, Control and Dynamics 30(6), 1577–1589 (2007), doi:10.25141/1.30107

[12] Dezert, J., Pannetier, B.: A PCR-BIMM filter for Maneuvering Target Tracking. In: Fusion 2010, 13th International Conference on Information Fusion, July 26-29. EICC, Edinburgh (2010)

[13] Mracek, C.P., Cloutier, J.R., D'Souza, C.A.: A New Technique for Nonlinear Estimation. In: Proceedings of the 1996 IEEE International Conference on Control Applications, Dearborn, MI, September 15-18 (1996)

[14] Shue, S.P., Agarwal, R.K.: Extension of Kalman Filter Theory to Nonlinear Systems with Application to Wing Rock Motion. Asian Journal of Control 2(1), 42–49 (2000)

[15] Xin, M., Balakrishnan, S.N.: A New Filtering Technique for a Class of Nonlinear Systems. In: Proceedings of the 41st IEEE Conference on Decision and Control, Las Vegas, Nevada USA (December 2002)

[16] Lam, Q.M., Anderson, B., Xin, M.: θ-D Filter vs. Nonlinear Filtering Techniques in Nonlinear Target Tracking Applications. In: AIAA 2010 InfoTech Conference @Aerospace 2010, Atlanta, Georgia, April 20-22, pp. 2010–3521. AIAA (2010)

[17] Xin, M., Balakrishnan, S.N., Ohlmeyer, E.J.: Integrated Guidance and Control of Missile with θ-D Method. IEEE Trans. on Control Systems Technology 14(6), 981–992 (2006)

[18] Haessig, D.A., Friedland, B.: State Dependent Differential Riccati Equation for Nonlinear Estimation and Control. In: 2002 IFAC, 15th Triennial World Congress, Barcelona, Spain (2002)
[19] Bryson, A.E., Ho, Y.C.: Applied Optimal Control. Hemisphere, New-York (1975)
[20] Kwakernaak, H., Sivan, R.: Linear Optimal Control Systems. Wiley-Interscience (1972)
[21] Blakelock, J.H.: Automatic Control of Aircraft and Missiles. John Wiley & Sons, Inc. (1991)
[22] Asseo, S.J., Ardila, R.J.: Sensor-Independent Target State Estimator Design and Evaluation. In: Proc. National Aerospace and Electronics Conference, NAECON 1982, pp. 916–924 (1982)
[23] Barton, D.K., Ward, H.R.: Handbook of Radar Measurement. Prentice-Hall (1969)
[24] Papoulis, A.: Probability, Random Variables, and Stochastic Processes. McGraw-Hill, Inc. (1991)
[25] Imado, F., Miwa, S.: Missile Guidance Algorithm against High-g Barrel Roll maneuvers. Journal of Guidance, Control and Dynamics 17(1), 123–128 (1994)
[26] Zarchan, P.: Tracking and Intercepting Spiraling Ballistic Missiles. In: Proceedings of IEEE Plans Symposium, Position Location and Navigation Symposium, San Diego, CA, pp. 277–284 (March 2000)
[27] Harman, R.R., Bar-Itzhack, I.Y.: Pseudolinear and State-Dependent Riccati Equation Filters for Angular Rate Estimation. Journal of Guidance, Control and Dynamics 22(5), 1577–1589 (1999)
[28] Azor, R., Bar-Itzhack, I.Y., Deutschmann, J., Harman, R.R.: Angular-Rate Estimation using Delayed Quaternion Measurements. Journal of Guidance, Control and Dynamics 24(3), 436–443 (2001)
[29] Kim, J., Vaddi, S.S., Menon, P.K., Ohlmeyer, E.J.: Comparison Between Nonlinear Filtering Techniques for Spiraling Ballistic Missile State Estimation. IEEE Trans. on AES 48(1), 313–328 (2012)

A Guaranteed Bound Filter for Polynomial Systems

György Hexner, Ilan Rusnak, and Haim Weiss

RAFAEL, Advanced Defense Systems, Haifa, Israel
{georgeh,ilanru,haimw}@rafael.co.il

Abstract. A recursive filter for polynomial systems is derived, where the bound on the mean square estimation error is explicitly calculated. The derivation relies on the recently introduced theory of positive polynomials. The general form of the filter is similar to the extended Kalman filter, but the filter gain is calculated differently.

1 Introduction

The purpose of this paper is to present a recursive filter for a polynomial system. The filter is in the form of the extended Kalman filter, but it inherently includes bounds on the mean square error of the estimated states. The derivation relies on the recently introduced positive polynomials [9], [10], [1], and the software SOSTOOLS [11]. SOSTOOLS translates the problem to a semi-definite program, which is readily solved by SeDuMi [13]. This set of software makes possible the numerical calculation of the bounds used in this paper routine.

There have been many attempts to use polynomial approximations for non-linear Kalman filters [14], [2], the latter also includes a bibliography of earlier work. A barrier to these efforts has been the moment closure problem [12], [6]. The problem refers to the fact that the calculation of nth order moments of a stochastic process requires the calculation of at least $(n+1)$th order moments, eventually leading to an infinite sequence of moments. In any practical application of polynomial approximation in filtering, some rule for truncating this infinite sequence is required. A popular method has been to assume that moments of higher order are related to lower order moments as if the underlying probability density were Gaussian, [6].

An approach to estimation for cone bounded non-linearities was proposed in [5,4,3]. The special feature of these papers, compared to the many publications that deal with estimation for non-linear systems, is the derivation of an *analytic bound* on the performance of the estimator, without requiring any sort of truncation approximation. A different approach to nonlinear estimation is [8], which is based on a special type of discretization of the exact equations of nonlinear filtering.

The present paper derives similar performance bounds, without resorting to truncating an infinite sequence, for estimators for the state of a non-linear system whose right hand sides are polynomials. The use of SOSTOOLS to derive bounds

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_4

on the moments and other nonlinear functions of jump diffusion processes was initiated in [7]. The paper derived time varying bounds on nonlinear functions, including moments, of the state of a polynomial system. In contrast to this, here the derived bounds are solutions of differential inequalities. The use of these bounds is more suited to state estimation than the time functions derived in [7].

The present paper is restricted to the simplest possible case. The system is assumed to be first order, and only the simplest extended Kalman filter type formulas for the propagation of the estimate. The observation equation is restricted to being linear. The data processing uses a formula similar to the one in the extended Kalman filter, but the filter gain is computed differently. The contribution of the paper is the derivation of precomputable bounds on the mean square error of the resulting filter. Thus the class of systems with precomputable mean square bounds is expanded from the cone bounded non-linearities in [5,4,3] to systems with polynomial right hand sides.

2 Problem Statement

Let

$$dx = f(x)dt + g(x)dw \tag{1}$$

be a scalar diffusion, $f(x)$ and $g(x)$ are polynomial functions, and w is a standard Wiener process. The polynomials $f(x)$ and $g(x)$ are arbitrary, but the next assumption ensures that all the moments of the random variable x exist, provided that all the moments exist at the initial time.

Assumption 1. *The highest order term of $f(x)+g^2(x)$ is odd, and its coefficient is negative.*

The variable y

$$y(T_k) = h_1 x(T_k) + v_k \tag{2}$$

is observed at regular intervals, defined by the sequence of times, $\{T_k\}$; and, v_k is a zero mean white noise sequence, independent of the state x, whose variance is R. The aim is to derive an estimate for x, and bounds on its mean square error from the observations y.

The functions $f(x)$, $g(x)$ are defined as

$$f(x) = \sum_{k=1}^{N_f} f_k x^k \tag{3}$$

$$g(x) = \sum_{k=0}^{N_g} g_k x^k \tag{4}$$

3 The Data Processing Step

The update rule here is restricted to being linear, that is of the form

$$\hat{x}(T_k) = \tilde{x}(T_k) + K\left[y(T_k) - h_1\tilde{x}(T_k)\right] \tag{5}$$

for some K, which is to be determined. Here the $\tilde{x}$ represents the predicted value of the state variable and the $\hat{x}$ the estimate of the state, after processing the observation at time T_k. This rule is sub-optimal, nevertheless, a bound is here derived for the mean square error after the update. Let

$$n(T_k) = h_1 x(T_k) - h_1\tilde{x}(T_k) = -h_1\tilde{e}(T_k) \tag{6}$$

where

$$\tilde{e}(T_k) = \tilde{x}(T_k) - x(T_k) \tag{7}$$

is the prediction error at time T_k. Then the error in the estimate $\hat{e}$ after the data processing step, using the update rule (5), is

$$\begin{aligned}\hat{e}(T_k) = \hat{x}(T_k) - x(T_k) &= \tilde{x}(T_K) + K\left[n(T_k) + v_k\right] - x(T_k)\\ &= \tilde{e}(T_k) + K\left[n(T_k) + v_k\right]\end{aligned} \tag{8}$$

The mean square error after the update is

$$\boldsymbol{E}_k\{\hat{e}^2\} = \boldsymbol{E}_{k-1}\{\tilde{e}^2\} + K^2\boldsymbol{E}_{k-1}\{n^2\} + K^2R + 2K\boldsymbol{E}_{k-1}\{\tilde{e}n\} \tag{9}$$

where the subscript k on the expectation operator indicates conditioning on all observations up to the kth observation, and the time subscript on all the variables is T_k, but has been suppressed for readability. Let $\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)$ be the bound on the prior mean square error of the estimate, $\tilde{e}$, at time T_k. Since only partial information is available on the prior mean square error, a reasonable criterion for choosing the gain K of the update rule is

$$\min_K \max_{\boldsymbol{E}_{k-1}\{\tilde{e}^2\} \le \overline{\sigma}^2_{\tilde{e}\tilde{e}}} \boldsymbol{E}_k\{\hat{e}^2\} \tag{10}$$

The expectations on the right hand side of (9) are readily calculated,

$$\boldsymbol{E}_{k-1}\{n^2\} = h_1^2\boldsymbol{E}_{k-1}\{\tilde{e}^2\} \tag{11}$$

$$\boldsymbol{E}_{k-1}\{\tilde{e}n\} = -h_1\boldsymbol{E}_{k-1}\{\tilde{e}^2\} \tag{12}$$

The mean square error after the update evaluates to

$$\boldsymbol{E}_k\{\hat{e}^2\} = (1 - Kh_1)^2\,\boldsymbol{E}_{k-1}\{\tilde{e}^2\} + K^2R \tag{13}$$

The maximum for any K such that

$$1 - Kh_1 > 0 \tag{14}$$

is achieved at the maximum value of $\boldsymbol{E}_{k-1}\{\tilde{e}^2\}$; namely, by substituting

$$\boldsymbol{E}_{k-1}\{\tilde{e}^2\} = \overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k) \tag{15}$$

The minimizing value of K is

$$K = \frac{\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1}{\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1^2 + R} \tag{16}$$

so that (14) is satisfied for all $\boldsymbol{E}_{k-1}\{\tilde{e}^2\} \leq \overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)$. The value of the mean square error after the update is

$$\boldsymbol{E}_k\{\hat{e}^2\} = R\left(\frac{R\boldsymbol{E}_{k-1}\{\tilde{e}^2\} + \overline{\sigma}^4_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1^2}{\left(\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1^2 + R\right)^2}\right) \tag{17}$$

which, when the bound for $\boldsymbol{E}_{k-1}\{\tilde{e}^2\}$ is substituted, becomes

$$\boldsymbol{E}_k\{\hat{e}^2\} \leq \sigma^2_{\hat{e}\hat{e}}(\tilde{x}, T_k) \tag{18}$$

where,

$$\sigma^2_{\hat{e}\hat{e}}(\tilde{x}, T_k) = \frac{R\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x})}{\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1^2 + R} \tag{19}$$

The $\sigma^2_{\hat{e}\hat{e}}(\tilde{x}, T_k)$ serves as the initial condition for the propagation of the bound on the mean square error during the time update step. Note that

$$\boldsymbol{E}_k\{\hat{e}^2\} < \boldsymbol{E}_{k-1}\{\tilde{e}^2\} \tag{20}$$

so that a decrease in the mean squared error after each data processing step is ensured.

4 The Time Update Step

Between measurements the estimate is propagated as

$$d\tilde{x} = f(\tilde{x})dt \tag{21}$$

The initial condition for (21) is

$$\tilde{x}(T_k) = \hat{x}(T_k) \tag{22}$$

where $\hat{x}$ was calculated in (5). This estimate propagation rule is not optimal, but is the simplest to use. It forms the basis of the analysis here. The estimation error propagates according to

$$d\tilde{e} = [f(\tilde{x}) - f(\tilde{x} - \tilde{e})]\, dt - g(\tilde{x} - \tilde{e})dw \tag{23}$$

Note that $\tilde{x}$ can be calculated in (21), so that it is a known time function between measurements. Suppose that functions $A(\tilde{x})$ and $B(\tilde{x})$ can be found such that

$$2\tilde{e}\,[f(\tilde{x}) - f(\tilde{x} - \tilde{e})] + g^2(\tilde{x} - \tilde{e}) \leq 2A(\tilde{x})\tilde{e}^2 + B^2(\tilde{x}) \tag{24}$$

Using the Ito rule [6] the error squared propagates according to

$$d\tilde{e}^2 = \left[2\tilde{e}\left[f(\tilde{x}) - f(\tilde{x}-\tilde{e})\right] + g^2(\tilde{x}-\tilde{e})\right]dt - g(\tilde{x}-\tilde{e})dw \tag{25}$$

Then taking expectations,

$$d\boldsymbol{E}_{k-1}\{\tilde{e}^2\} = \boldsymbol{E}_{k-1}\left\{2\tilde{e}\left[f(\tilde{x}) - f(\tilde{x}-\tilde{e})\right] + g^2(\tilde{x}-\tilde{e})\right\}dt \tag{26}$$

Hence the bound $\overline{\sigma}^2_{\tilde{e}\tilde{e}}$ on $\boldsymbol{E}_{k-1}\{\tilde{e}^2\}$ evolves according to,

$$d\overline{\sigma}^2_{\tilde{e}\tilde{e}} = \begin{cases} \left[2A(\tilde{x})\overline{\sigma}^2_{\tilde{e}\tilde{e}} + B^2(\tilde{x})\right]dt, & A(\tilde{x}) \geq 0 \\ B^2(\tilde{x})dt, & A(\tilde{x}) < 0 \end{cases} \tag{27}$$

The initial condition for $\overline{\sigma}^2_{\tilde{e}\tilde{e}}$ is from the last data processing step, (19), with

$$\overline{\sigma}^2_{\tilde{e}\tilde{e}}(T_k) = \sigma^2_{\hat{e}\hat{e}}(T_k) \tag{28}$$

When $A(\tilde{x}) \leq 0$ equation (27) is a Lyapunov equation. An interpretation of $A(\tilde{x})$ and $B(\tilde{x})$ is that between measurements, the covariance of $\tilde{e}$ in the linear stochastic differential equation

$$d\tilde{e} = A(\tilde{x})\tilde{e}dt + B(\tilde{x})dw \tag{29}$$

dominates the covariance of $\tilde{e}$ in (23), for a given value of the covariance. Equation (27) is the differential equation normally used to propagate the covariance in the Kalman filter.

4.1 Calculating $A(\tilde{x})$ and $B(\tilde{x})$

Ideally the most desirable $A(\tilde{x})$ and $B(\tilde{x})$ is the one that results in the tightest bound in (24), that is the least upper bound is sought. The first step in the exposition is to show that there always exists $A(\tilde{x})$ and $B(\tilde{x})$ that satisfies (24).

When f is a linear function of x, and g is constant, there is no need to calculate a bound, therefore this possibility is not considered here. Choose $A(\tilde{x})$ be an arbitrary function. Then from Assumption 1 there exists $\tilde{e}_M(\tilde{x})$, such that

$$2\tilde{e}\left[f(\tilde{x}) - f(\tilde{x}-\tilde{e})\right] + g^2(\tilde{x}-\tilde{e}) \leq 2A(\tilde{x})\tilde{e}^2,\ \forall |\tilde{e}| > e_M(\tilde{x}) \tag{30}$$

since the highest power of the polynomial on the left hand side of (30) is even and its coefficient is negative, the polynomial is bounded above, and for sufficiently large $|\tilde{e}|$ the highest power, and its coefficient dominate. Since the left hand side of (30) is bounded from above, there exists $B(\tilde{x})$ such that (24) holds $\forall \tilde{e}$.

To calculate the least upper bound, the MATLAB toolbox SOSTOOLS [11] is available. Using the toolbox the problem is formulated as

$$\min_{A,B} \lambda_A A(\tilde{x}) + \lambda_B B^2(\tilde{x}) \tag{31}$$

subject to (24). For each value of λ_A and λ_B a different bound is obtained. Alternatively, $A(\tilde{x})$ may be fixed at some value and the criterion is simplified to

$$\min_{B} B^2(\tilde{x}) \tag{32}$$

Assumption 1 ensures that both problems have solutions. When implementing the filter this calculation needs to be carried out for all the relevant values of $\tilde{x}$ off-line, and stored in tables.

There are two special cases of some interest. Suppose that $A(\tilde{x})$ is chosen as

$$A(\tilde{x}) = \frac{\partial f(\tilde{x})}{\partial \tilde{x}} \tag{33}$$

Then $A(\tilde{x})$ is the value often used in covariance propagation of the extended Kalman filter. Then the optimization problem (32) amounts to calculating the minimum value of the process noise for the extended Kalman filter to ensure that the mean square error of the extended Kalman filter is less than the calculated bound. Note that the present filter is not necessarily unbiased.

Another interesting value for $A(\tilde{x})$ is

$$A(\tilde{x}) = \frac{f(\tilde{x})}{\tilde{x}} \tag{34}$$

Now the minimum value of the process noise is calculated to ensure that the mean square error of the state dependent Riccati equation (SDRE) filter is less than the calculated bound.

5 Outline of the Filter Algorithm

In this section , the algorithm is summarized. The summary of the algorithm is shown in Table 1. Note that it includes an off line step, where the functions $A(\tilde{x})$ and $B(\tilde{x})$ are calculated. Once the algorithm is started after the first measurement, it cycles between the steps "Measurement update", and "Time update", similar to the standard extended Kalman filter. A crucial difference between the extended Kalman filter and the new proposed algorithm is that the extended Kalman filter calculates an approximation to the estimate covariance, whereas the present filter calculates a bound on the mean squared error of the estimation error. The estimate may be biased, nevertheless the calculated mean square error bound is valid. In the following paragraph the equations of the algorithm are collected.

Table 1. Summary of the algorithm

Step	**Description**	**Equation**
Off line	Bound calculation	(31), (24)
Measurement Update	Gain calculation	(16)
	Update state	(5)
	RMS bound calculation	(19)
Time Update	Propagation of the estimate	(21), (22)
	RMS error bound propagation	(27), (28)

Off Line
Calculate the bound

$$\min_{A,B} \lambda_A A(\tilde{x}) + \lambda_B B^2(\tilde{x})$$

subject to

$$2\tilde{e}\left[f(\tilde{x}) - f(\tilde{x} - \tilde{e})\right] + g^2(\tilde{x} - \tilde{e}) \leq 2A(\tilde{x})\tilde{e}^2 + B^2(\tilde{x})$$

for all relevant values of $\tilde{x}$.

Measurement Update
Gain Calculation:

$$K = \frac{\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1}{\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1^2 + R}$$

Update estimate

$$\hat{x}(T_k) = \tilde{x}(T_k) + K\left[y(T_k) - h_1\tilde{x}(T_k)\right]$$

Mean square bound calculation:

$$\sigma^2_{\hat{e}\hat{e}}(\tilde{x}, T_k) = \frac{R\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)}{\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k)h_1^2 + R}$$

Time Update
Propagation of the estimate

$$d\tilde{x} = f(\tilde{x})dt$$

with the initial condition

$$\tilde{x}(T_k) = \hat{x}(T_k)$$

Mean square error bound propagation:

$$d\overline{\sigma}^2_{\tilde{e}\tilde{e}} = \begin{cases} \left[2A(\tilde{x})\overline{\sigma}^2_{\tilde{e}\tilde{e}} + B^2(\tilde{x})\right]dt, & A(\tilde{x}) \geq 0 \\ B^2(\tilde{x})dt, & A(\tilde{x}) < 0 \end{cases}$$

with the initial condition

$$\overline{\sigma}^2_{\tilde{e}\tilde{e}}(\tilde{x}, T_k) = \sigma^2_{\hat{e}\hat{e}}(\tilde{x}, T_k)$$

6 An Example

The example chosen to illustrate the bound filter is the following: The underlying continuous time process is generated by the stochastic differential equation,

$$dx = x(c^2 - x^2)dt + gdw \tag{35}$$

while the discrete time observations are generated by the equation

$$y(T_k) = x(T_k) + v(T_k) \tag{36}$$

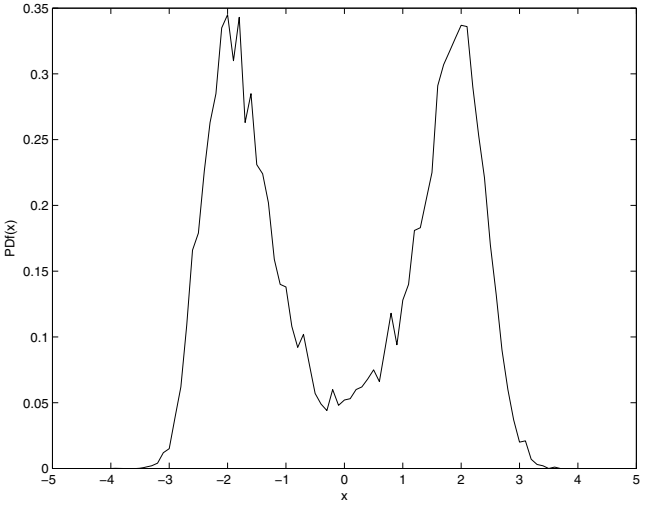


Fig. 1. The stationary probability density of continuous time process

Table 2. Parameter values in the example

Parameter	Description	Value
c	—	2
g	process noise intensity	2
R	observation noise variance	16
$T_k - T_{k-1}$	time between observations	0.025
λ_A	weight for calculating bound	1
λ_B	weight for calculating bound	0.1

Note that in (35) there are three critical points, at $x = -c$, $x = 0$, $x = c$. The critical point at $x = 0$ is repulsive, while the remaining two are attractive. The stationary probability density of x is shown in Fig. 1. It was obtained from a 10000 sample Monte-Carlo simulation of (35) between 0 and 10 s, with initial condition $x = 0$ at $t = 0$. The time interval is sufficiently long to achieve stationary probability density. The values of the parameters used in the example are shown in table 2.

To illustrate the bound the left and right hand sides of the inequality (24) were plotted for $\lambda_A = 1$ and $\lambda_B = 0.005, 0.01, 0.05, 0.1$. The resulting plots are shown in Fig. 2, for $\tilde{x} = 0$. The bold line is a plot of the left side of (24), and the dotted lines are the values of the bounding quadratic function. The values of A and B as a function of $\tilde{x}$ are plotted in Figs. 3 and 4.

Figs. 5 and 6 show the estimation errors and the 1σ values of the error for the new bound based filter and for the traditions extended Kalman filter. These graphs were obtained by averaging the results of 250 Monte-Carlo simulations. The mean sample squared error and the average standard deviation of the error

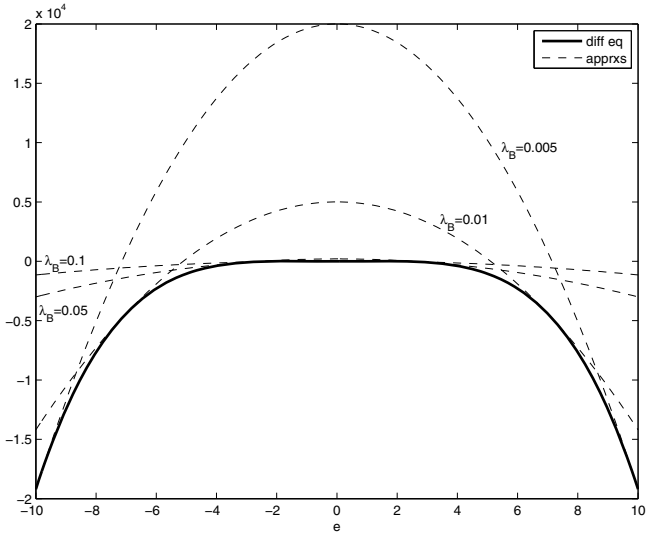


Fig. 2. Illustration of the bound for the example

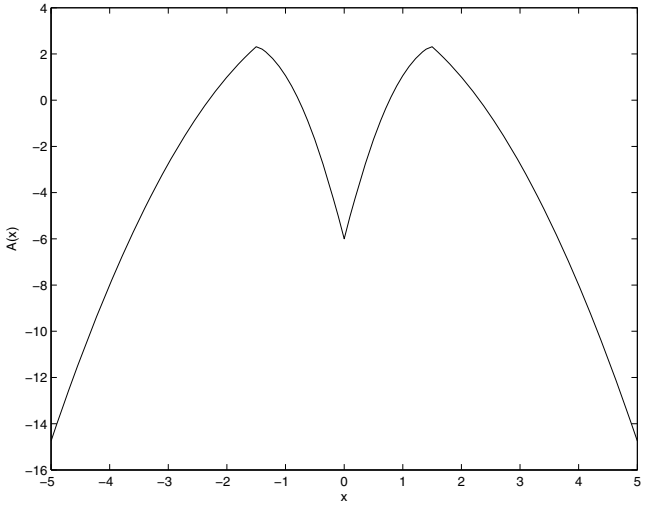


Fig. 3. The function $A(\tilde{x})$ for the example

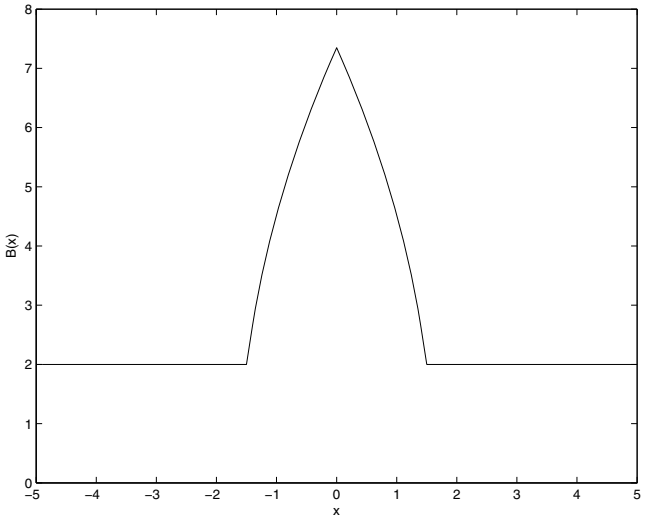


Fig. 4. The function $B(\tilde{x})$ for the example

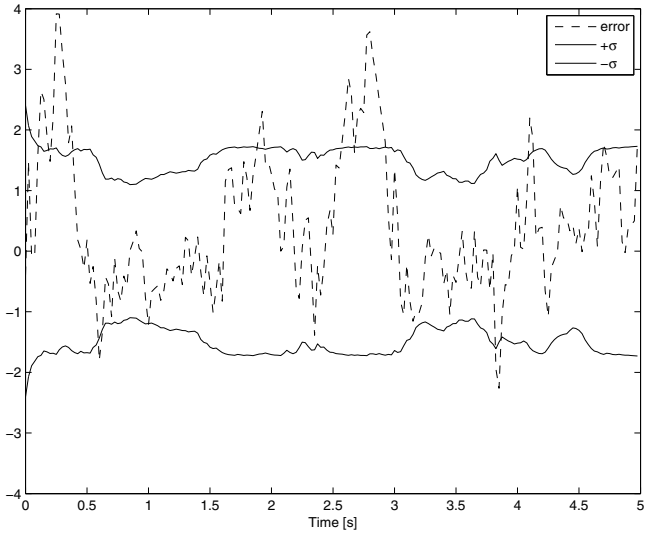


Fig. 5. Sample run for the bound based filter

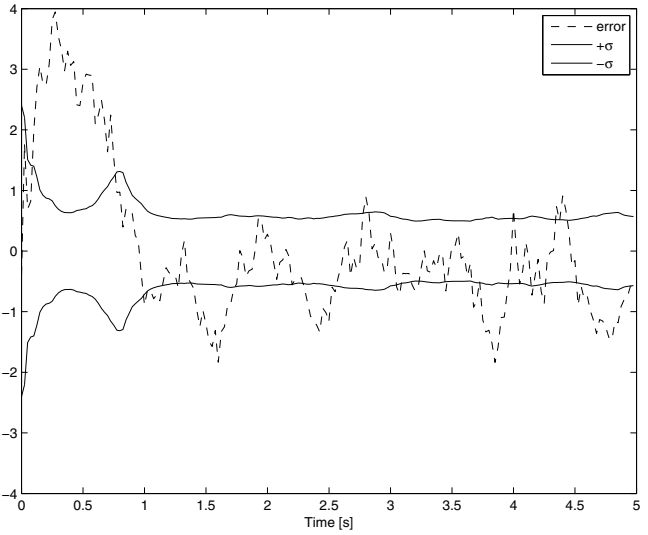


Fig. 6. Sample run for the extended Kalman filter

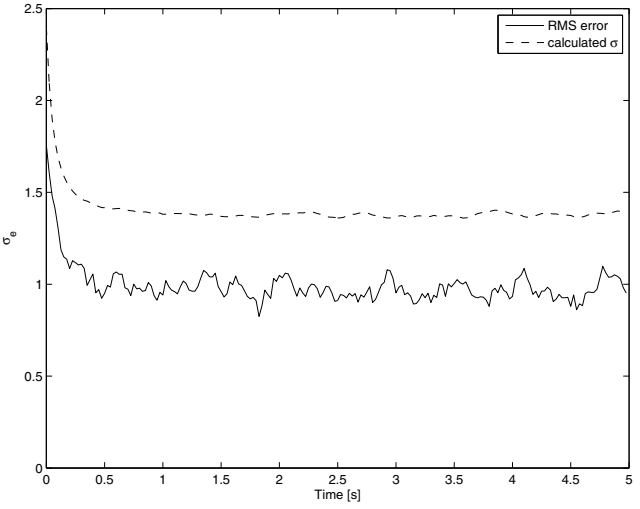


Fig. 7. Bound based filter: Root mean square error (dotted line) and average filter calculated error standard deviation (full line)

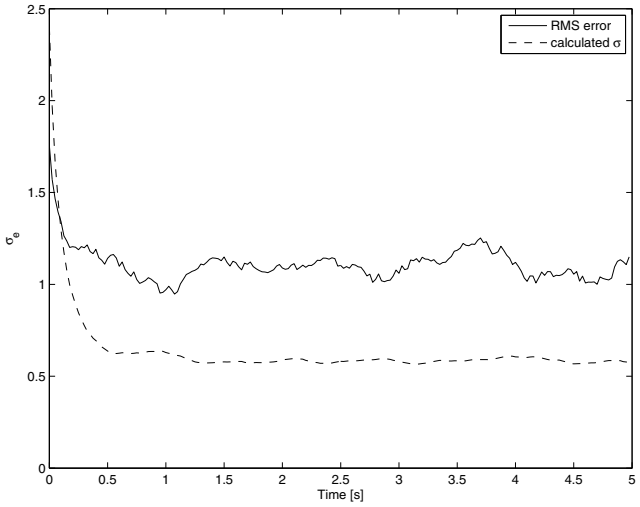


Fig. 8. Extended Kalman filter: Root mean square error (dotted line) and average filter calculated error standard deviation (full line)

in calculating the gains of the bound based filter and the extended Kalman filter are shown in Figs. 7 and 8 respectively. The level of process noise used in the extended Kalman filter was that given by the value of g. Note that for the bounded filter case the filter calculates a conservative estimate for the error standard deviation; whereas, for the extended Kalman filter the calculated error standard deviation severely *underestimates* the sample standard deviation. It may be possible to increase the value of the process noise and thereby improve the correspondence between the sample mean squared error and the covariance calculated by the filter.

7 Summary and Conclusions

The paper presents a new filter for estimating the state of a system with polynomial dynamics, and linear measurements. The filter is in the form of an extended Kalman filter, but the calculation of the gain differs from the commonly used filter gain calculation of the extended Kalman filter. An integral part of the new gain calculation is the generation of guaranteed bounds for the mean squared estimation error. The calculation of the bound relies on the recently introduced theory of positive polynomials, and the sum of squares approximation for testing their positiveness. The estimate is not necessarily unbiased, nevertheless the mean square error between the estimate and the true value, of the state, satisfies the calculated bound.

It is worth noting that for the class of systems considered in this paper the last part of section 4, which considers the $A(\tilde{x})$ derived from the EKF and

SDRE based filter, has a special meaning. It can be interpreted as a proof of the existence of a process noise value that renders the calculated filter error variance a valid bound of the filter error.

References

1. Chesi, G.: LMI techniques for optimization over polynomials in control: a survey. IEEE Transactions on Automatic Control AC 55(11), 2500–2510 (2010)
2. Germani, A., Manes, C., Paulmbo, P.: Polynomial extended Kalman filter. IEEE Transactions on Automatic Control 50(12), 2059–2064 (2005)
3. Gilman, A.S., Rhodes, I.A.: Cone-bounded nonlinearities and mean square bounds—estimation upper bound. IEEE Transactions on Automatic Control AC-18(3), 260–265 (1973)
4. Gilman, A.S., Rhodes, I.A.: Cone-bounded nonlinearities and mean square bounds—estimation lower bound. IEEE Transactions on Automatic Control AC-20(5), 632–641 (1975)
5. Gilman, A.S., Rhodes, I.A.: Cone-bounded nonlinearities and mean-square bounds — quadratic regulation bounds. IEEE Transactions on Automatic Control AC-21(4), 472–483 (1976)
6. Jazwinsky, A.H.: Stochastic Processes and Filtering Theory. Academic Press, New York (1970)
7. Kashima, K., Kawai, R.: An optimization approach to weak approxmation of stochastic differential equations with jumps. Applied Numerical Mathematics 61(5), 641–650 (2011)
8. Kumar, M., Chakravorty, S.: Nonlinear filter based on the Fokker–Planck equation. Journal of Guidance, Control and Dynamics 35(1), 68–79 (2012)
9. Papachristodoulou, A., Prajna, S.: A tutorial on sum of squares techniques for system analysis. In: American Control Conference (2005)
10. Parrilo, P.A., Lall, S.: Semidefinite programming relaxations and algebraic optimization in control. European Journal of Control 9(2-3), 307–321 (2003)
11. Prajna, S., Papachristodoulo, A., Seiler, P., Parrilo, P.A.: SOSTOOLS (2004), `http://www.cds.caltech.edu/sostools/`
12. Sorenson, H.W.: On the development of practical nonlinear filters. Information Sciences 7, 253–270 (1974)
13. Strum, J.F.: SeDuMi (2010), `http://sedumi.ie.lehigh.edu/`
14. Sue, S.P., Agarwal, R.K.: Extension of Kalman filter theory to nonlinear systems with application to wing rock motion. Asian Journal of Control 2(1) (2000)

Reference Basis Model Updating – Following the Baruch and Bar-Itzhack Approach

Yoram Halevi

Faculty of Mechanical Engineering,
Technion, Haifa 32000, Israel
yoramh@technion.ac.il

Abstract. The area known as model updating is concerned with the correction of analytical models of flexible structures, mostly finite element by processing test data. The topic is of great practical and theoretical importance and a large number of methods have been suggested over the last decades. Reference basis is one of the main approaches to model updating. Its underlying principle is to fix certain quantities and minimize the changes of the others while matching the measured eigendata exactly. The method was first introduced in a seminal paper of Baruch and Bar-Itzhack and immediately became a subject of extensive research activity. The current paper presents the classical reference basis method and some later results that were inspired by it.

Keywords: Model updating, flexible structures.

1 Introduction

Accurate models are essential in analyzing flexible structures, such as airplanes, space equipment, car bodies and modern bridges systems. The model enables prediction of the behavior under various excitations, boundary conditions and parameter changes and design of control systems. Analytical models, typically obtained by Finite Element method, inevitably deviate from the true model due to uncertainties in geometry, material properties, boundary conditions, etc. Consequently the natural frequencies and modeshapes that are extracted from the test results do not agree with the predicted values from the analytical model. It should be noted that the experimental results are always partial, so even if they are assumed to be absolutely accurate, which is far from being true, they still cannot be the sole source of the final model. Model updating is the process of using the test results to correct the model so that it agrees, either completely or approximately, with the experimental data. In broader terms, it can be described as fusion of two sources of information, both inaccurate and possibly incomplete, to obtain a better model.

A wide variety of model updating methods have been suggested, and a comprehensive survey of them can be found in [2]. They range from formal use of system identification, which, to a certain degree, completely disregards the analytical model [3], to

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_5

methods that use the entire frequency response function (FRF) [4]. Most approaches though do incorporate the analytical model and their starting point is the eigenvalue equation that should be satisfied. The notion of satisfying the equation is not unique however and has three main interpretations. Sensitivity methods, e.g. [5-7] aim at matching the solution of the equation, i.e. natural frequencies and modeshapes, with the experimental data. The dependence of the natural frequencies and modeshapes on the physical parameters is calculated either numerically or analytically. This method is presumably the most popular among practitioners.

A second approach is minimizing the equation error. This is the common theme in the class of methods known as Minimization of the Error in the Constitutive Equations (MECE), which minimize the difference between the static stiffness forces and the dynamic inertia forces [8-10]. With parameterization that is similar to the one in sensitivity methods, the various methods in this class differ mainly in the weighting of the error (residual force).

In Reference Basis (RB) methods perfect satisfaction of the eigenvalue equation, with the measured data, is required. With certain parameters that are assumed to be accurate, hence reference basis, the minimal change of the free quantities, typically stiffness, from their analytical values is sought. In mathematical terms, the minimal deviation requirement defines the optimization criterion while the characteristic equation is the constraint. The method was first introduced in the seminal paper of Baruch and Bar-Itzhack [1] and soon after was followed by several other publications of Baruch [11, 12] and Berman [13-15]. The main advantages of the reference basis methods are their ability to match the measured natural frequencies and modeshapes and the mathematical and numerical convenience. They provide closed form solutions, requiring only elementary matrix operations and inversion of low dimension matrices. The main disadvantage of the original methods is that they do not take into account any dependence of the matrices on physical properties, i.e. do not preserve the connectivity of the system. Perhaps the first attempt to solve this problem was given in [16] that minimizes a similar criterion yet imposes certain connectivity on the stiffness matrix. The updated stiffness matrix is closer to the true one, provided that the assumed connectivity is correct, however the method requires an extremely large number of parameters and involves the solution of a set of equations whose dimension is orders of magnitude larger than the number of measured modes. A similar problem was solved in [17] with a projection based, more efficient, algorithm.

This paper presents some improved and generalized versions of the main results in the reference basis methods, presented by the author and his colleagues in the last fifteen years. In [18], Kenigsbuch and Halevi introduced the generalized reference basis (GRB) that offers more flexibility by using a general weight instead of the mass matrix in the original RB. This result was heavily used by Halevi and Bucher in [19] in which RB and the parametric optimization, used in other methods, were merged in the connectivity constrained reference basis (CCRB) method. The updated model is found using RB methods but has the required connectivity. Taking the same idea one step further, Halevi, Vilensky and Datta [20] derived a direct method, called manifold distance minimization (MDM) that defines the sets of models that have the correct structure and models that yield the measured data, and look for the minimum distance

between them. A technical simple example is used throughout the paper to demonstrate the use of the various methods and for comparison of their performance.

The material is organized as follows. The model updating problem is stated in section 2. Section 3 gives a short introduction to reference basis following Baruch and Bar-Itzhack [1]. The generalized reference basis method is given in section 4, the connectivity constrained reference basis in 5 and the manifold distance minimization in section 6. The results are briefly summarized and discussed in section 7.

2 The General Model Updating Problem

Consider a conservative, i.e. without damping, flexible structure with N degrees of freedom (dof). Let the 'true' equations of the system be

$$M_T \ddot{x}(t) + K_T x(t) = f(t) \tag{1}$$

where $x \in R^N$ is a vector of generalized displacements and $M_T > 0$ and $K_T \geq 0$ are the mass and stiffness matrices respectively. Clearly the absolutely accurate model of the system is non-linear, infinite dimensional, damped, etc., so 'true' means here accurate enough for all practical purposes. The natural frequencies ω_i and the modeshapes ϕ_i satisfy the eigenvalue equation

$$(K_T - M_T \omega_i^2)\phi_i = 0 \quad , \quad i = 1, \ldots, N \tag{2}$$

It is well known that the modeshapes are orthogonal w.r.t the mass matrix so with appropriate normalization of their length

$$\phi_i^T M_T \phi_j = \delta_{ij} \tag{3}$$

The analytic model of the system, usually obtained by finite elements, is given by

$$M_A \ddot{x}(t) + K_A x(t) = f(t) \tag{4}$$

where in general $M_A \neq M_T$ and $K_A \neq K_T$.

The result of a modal test of the system are m ($m<N$) natural frequencies, ω_{xi}, and m modeshapes ϕ_{xi}. Since not all dof's of the model are measured, in general the measured modeshapes contain only a partial set of $N_x \leq N$ of values of those vectors. The modal experimental data thus consists of two matrices: A diagonal matrix $\Omega_x \in R^{m \times m}$ of the measured natural frequencies and a full matrix $\Phi_x \in R^{N_x \times m}$ containing the values of the modeshapes at the measured dof's.

The most general statement of the model updating problem is how to combine the analytic information M_A, K_A and the experimental results Ω_x, Φ_x to obtain a model which is more accurate. More specifically, how to find $M(M_A, K_A, \Omega_x, \Phi_x)$ and $K(M_A, K_A, \Omega_x, \Phi_x)$ that are close to M_T, K_T.

3 Reference Basis Model Updating

In this section the results of Baruch and Bar-Itzhack [1] are repeated. Three assumptions are first made.

Assumption 1: The measured modeshapes contain all the dof's, i.e. N_x=N. This does not necessarily mean that all dof's are measured, which is usually unrealistic, but rather that a preliminary step of expansion or reduction [21-24] has already been applied to the partial (dof wise) set of modeshapes or to the model.

Assumption 2: The measured natural frequencies are correct, i.e. $\Omega_x = \Omega_T = \Omega$. The physical justification for this is that natural frequencies are global variables, which are common to all measurements in the system and therefore after processing their error is relatively small.

Assumption 3: The mass matrix is correct, i.e. M_A=M_T =M. The physical justification for this is that the mass is usually known better than the stiffness since it involves mainly geometry. However, in some cases that will be discussed later this assumption was replaced by another or even removed.

Assumptions 2 and 3 provide the "reference basis" for the updating algorithms in the next two sub-sections.

3.1 Orthogonalization

The first step towards stiffness updating is correcting the measured modeshapes so that they satisfy the orthogonality condition. Though not required specifically by the algorithm it is assumed for practical reasons that each modeshape has already been scaled individually so that $\phi_{xi}^T M \phi_{xi} = 1$. With M as a reference basis, the problem is formulated as follows

$$\begin{aligned} \min_{\Phi} \quad & J = \left\| M^{1/2} \left(\Phi - \Phi_X \right) \right\|_F \\ s.t \quad & \Phi^T M \Phi = I \end{aligned} \tag{5}$$

Where F denotes the Frobenius norm. The solution was found using Lagrange multipliers and is given by

$$\Phi = \Phi_X \left(\Phi_X{}^T M \Phi_X \right)^{-1/2} \tag{6}$$

3.2 Stiffness Updating

With M and Φ, that are assumed to be accurate, as reference basis, the updating of K follows two principles. First, the eigenvalue equation should be satisfied exactly, which means that the natural frequencies and modeshapes of the updated model coincide with the measured values. Secondly, among all stiffness matrices satisfying that the method seeks the one with minimum change from the analytical model. The optimization problem is therefore as follows.

$$\begin{aligned} \min_{K} \quad & J = \left\| M^{-1/2} \left(K - K_A \right) M^{-1/2} \right\|_F^2 \\ s.t \quad & K\Phi = M\Phi\Omega^2 \\ & K = K^T \end{aligned} \tag{7}$$

The solution, which was obtained by means of Lagrange's multipliers, is given by

$$K = K_A - \left(K_A\Phi - M\Phi\Omega^2 \right)\Phi^T M - M\Phi\left(K_A\Phi - M\Phi\Omega^2 \right)^T + M\Phi\left(\Phi^T K_A\Phi - \Omega^2 \right)\Phi^T M \tag{8}$$

To summarize, the reference basis has the following main advantages.

- It matches the measured (processed) eigendata exactly.
- It provides closed form expressions.
- The amount of calculation is small. Only matrix multiplications and summation in the stiffness updating and a single inversion in the dimension of the measurements in the modeshapes orthogonalization are required.

3.3 Alternative Formulation – Mass Matrix Update

The procedure in sub-section 3.1 is based on the assumption that the mass matrix is known exactly and the modeshapes are corrected to ensure an orthogonal set with respect to it. Berman [13] claimed that after model reduction this is not necessarily so and suggested to use the stiffness matrix in the orthogonalization. In [15] the roles of the mass matrix and the measured modeshapes are reversed. It is assumed that the measured modeshapes are exact, and the mass matrix is corrected accordingly. The optimization problem of that formulation becomes

$$\begin{aligned} \min_{M} \quad & J = \left\| M_A^{-1/2} \left(M - M_A \right) M_A^{-1/2} \right\|_F \\ s.t \quad & \Phi^T M\Phi = I \end{aligned} \tag{9}$$

With the solution

$$M = M_A + M_A\Phi(\Phi^T M_A\Phi)^{-1}(I - \Phi^T M_A\Phi)(\Phi^T M_A\Phi)^{-1}\Phi^T M_A \tag{10}$$

The updated M is then used in the stiffness updating in sub-section 3.2 without any further change.

4 Generalized Reference Basis

The advantages of the reference basis method were listed in the previous section. However, its lack of ability, in its original form, to maintain the correct connectivity was labeled by both researchers and practitioner as a major disadvantage. This is demonstrated by the following simple example.

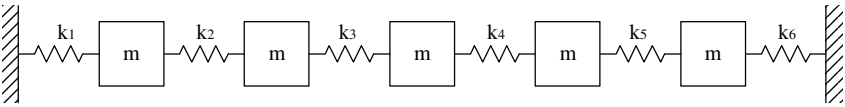


Fig. 1. A simplified model of a flexible system

Example: The system shown in Fig. 1 contains five masses and six springs and can be regarded as an approximation of a rod in tension or torsion. The mass and stiffness matrices are given by

$$M = m \cdot I_5 \quad , \quad K = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 & 0 & 0 \\ -k_2 & k_2 + k_3 & -k_3 & 0 & 0 \\ 0 & -k_3 & k_3 + k_4 & -k_4 & 0 \\ 0 & 0 & -k_4 & k_4 + k_5 & -k_5 \\ 0 & 0 & 0 & -k_5 & k_5 + k_6 \end{bmatrix}$$

The true numerical values are m=1, k_1=k_3=k_4=k_5=k_6=50, k_2=70, whereas in the analytical model, m=1, and k_i=50, i=1,…6. The true and analytic stiffness matrices are then

$$K_T = \begin{bmatrix} 120 & -70 & 0 & 0 & 0 \\ -70 & 120 & -50 & 0 & 0 \\ 0 & -50 & 100 & -50 & 0 \\ 0 & 0 & -50 & 100 & -50 \\ 0 & 0 & 0 & -50 & 100 \end{bmatrix} \quad , \quad K_A = \begin{bmatrix} 100 & -50 & 0 & 0 & 0 \\ -50 & 100 & -50 & 0 & 0 \\ 0 & -50 & 100 & -50 & 0 \\ 0 & 0 & -50 & 100 & -50 \\ 0 & 0 & 0 & -50 & 100 \end{bmatrix}$$

The experimental data consists of the first two natural frequencies and modeshapes. This technical example will be used throughout the paper to demonstrate the properties of the various updating algorithms. Applying the reference basis algorithm to this example resulted in

$$K_{RB} = \begin{bmatrix} 97.88 & -50.61 & 1.96 & -1.72 & -1.01 \\ -50.61 & 103.01 & -48.28 & 1.51 & 0.88 \\ 1.96 & -48.28 & 99.83 & -50.15 & -0.09 \\ -1.72 & 1.51 & -50.15 & 99.87 & -50.08 \\ -1.01 & 0.88 & -0.09 & -50.08 & 99.95 \end{bmatrix}$$

As can be seen the updated matrix is not close to the true one neither in structure nor in its values. The change from the analytical model is minimal, yet sufficient to match exactly the first two natural frequencies and modeshapes. Furthermore, the matrix contains stiffness elements that are clearly non-physical.

One of the key features of reference basis is the use of the inverse of the mass matrix as the weight in the optimization criterion (7). This choice has a physical justification, and it is convenient for the derivation because of the orthogonality condition $\Phi^T M \Phi = I$. However, as was shown by the example, it does not always lead to good

results. Kenigsbuch and Halevi [18] introduced the generalized reference basis (GRB) that offers more flexibility by using a general weight $W>0$. The problem is defined as

$$\begin{aligned} \min_{L} \quad & J = \left\| W^{-1/2}\left(K - K_A\right)W^{-1/2}\right\|_F^2 \\ s.t \quad & K\Phi = M\Phi\Omega^2 \\ & K = LL^T \end{aligned} \tag{11}$$

The second constraint requires non-negativity of the updated stiffness matrix rather than only symmetry as in previous works. The solution of this problem, that involves some matrix operations, is given by

$$K = K_A - \left(K_A\Phi - M\Phi\Omega^2\right)R^T - R\left(K_A\Phi - M\Phi\Omega^2\right)^T + R\left(\Phi^T K_A\Phi - \Omega^2\right)R^T \tag{12}$$

where

$$R = W\Phi\left(\Phi^T W\Phi\right)^{-1} \tag{13}$$

R is a right inverse of Φ^T whose 'angle' depends on the weight W. If $W=M$, as in RB, then $R=M\Phi$ and (12) coincides with (8). A second form of expressing the same updated K is

$$K = M\Phi\Omega^2\Phi^T M + \left(I - \Phi R^T\right)^T\left(K_A - M\Phi\Omega^2\Phi^T M\right)\left(I - \Phi R^T\right) \tag{14}$$

Both (12) and (14) have a structure of predictor-corrector. In (12) K_A is interpreted as the *a priori* estimate and the experiment results Φ and Ω are the new data. In (14) the roles are reversed and $M\Phi\Omega^2\Phi^T M$ is the *a priori* estimate, partial in this case, and K_A is the new data. The form (14) has also a geometrical interpretation. To see this, define the complete set of natural frequencies and modeshapes as $\Phi_F=[\Phi \;\; \Phi_u]$ and $\Omega_F=diag\{\Omega, \Omega_u\}$ where Φ_u and Ω_u are the unmeasured quantities. Then

$$\begin{aligned} K_T &= M\Phi_F\Omega_F{}^2\Phi_F{}^T M \\ &= M\Phi\Omega^2\Phi^T M + M\Phi_u\Omega_u^2\Phi_u{}^T M \end{aligned} \tag{15}$$

Notice that $K_A - M\Phi\Omega^2\Phi^T M$ is an approximation of the contribution of the unmeasured part that is given in the last term in (15) and is equal to it when K_A is accurate, i.e. $K_A=K_T$. Since R is a right inverse of Φ^T, $I-\Phi R^T$ is a projection into a subspace orthogonal to Φ. Thus K has an accurate part (the first term) in a subspace defined by the measured modeshapes Φ and a correction part, projected into a subspace orthogonal to Φ. In case $K_A=K_T$, the difference already belongs to that subspace and therefore remains unchanged by the projection. The weight W affects only the 'angle' of the projection but not its image. However, as will be demonstrated in the sequel, this has a strong effect on the results.

GRB is now applied to the same example. Suppose that it is somehow known in advance that the inaccuracy lies in the first two dof's and therefore the weight is chosen as

$$W = diag\{a, a, 1, 1, 1\} \tag{16}$$

a=1 is the same as RB, and as it increases the updated stiffness K gets closer to the true values and structure of the stiffness matrix. For example, for a=100 and a=1000 we have

$$K_{100} = \begin{bmatrix} 112.98 & -64.02 & -0.37 & -0.59 & -0.45 \\ -64.02 & 114.91 & -49.68 & 0.50 & 0.38 \\ -0.37 & -49.68 & 99.98 & -50.03 & -0.02 \\ -0.59 & 0.50 & -50.03 & 99.95 & -50.04 \\ -0.45 & 0.38 & -0.02 & -50.04 & 99.97 \end{bmatrix}$$

$$K_{1000} = \begin{bmatrix} 119.72 & -69.76 & -0.01 & -0.02 & -0.02 \\ -69.76 & 119.80 & -49.99 & 0.02 & 0.02 \\ -0.01 & -49.99 & 100.00 & -50.00 & -0.00 \\ -0.02 & 0.02 & -50.03 & 100.00 & -50.00 \\ -0.02 & 0.02 & -0.00 & -50.00 & 100/00 \end{bmatrix}$$

The generalized reference basis method [18] has also two other results that are not given here; a counterpart of the orthogonalization procedure with general weight instead of the matrix M and an algorithm for simultaneous updating of the mass and stiffness matrices.

5 Generalized Reference Basis with Connectivity Constraints

5.1 Connectivity

The stiffness matrix has usually inherent connectivity properties. In their simplest form, which we call zero-nonzero connectivity, some of the entrees should be identically zero. Let I_0 be the set of entries of the stiffness matrix which are known to be identically zero. Then this type of connectivity is defined formally by

$$K_{ij} = 0 \quad \forall\, i, j \in I_0 \tag{17}$$

Further connectivity requirements exist in the structured connectivity case, where certain relationships between the nonzero elements of the stiffness matrix should hold. Hence the model is actually determined by a smaller set of parameters. As an example, in the case of n masses connected serially by springs in a fixed-fixed form, as in our example, the stiffness matrix is completely determined by the n+1 spring constants with the requirement that the sum of each row, except the first and last, is zero. Similar conditions arise from the Finite Element structure. Following the

framework suggested in many publications, e.g. [5], it is assumed that the stiffness matrix depends on the free parameters linearly

$$K = K_A + \sum_{i=1}^{p} \alpha_i K_i \tag{18}$$

where K_i are given matrices and α_i are scalar parameters, nominally equal zero. In case the true dependence on the parameters is nonlinear, eq. (18) can be viewed as linearization, assuming small deviations. In [5-7], as well as in other publications, the problem was solved using sensitivity methods which minimizes the error between the measured eigendata and the same variables when obtained using $K(\alpha)$. Setting up the problem with the RB criterion and the parameterization (18) is straightforward.

$$\begin{aligned} \min_{\alpha} \quad & J = \left\| W^{-1/2} \left(K - K_A \right) W^{-1/2} \right\|_F^2 \\ s.t. \quad & K\Phi = M\Phi\Omega^2 \\ & K = K_A + \sum_{i=1}^{p} \alpha_i K_i \end{aligned} \tag{19}$$

Getting a solution, on the other hand, is hard if not impossible. The main problem with (19) is that the number of parameters that are required to satisfy the constraint equation is in the order of the number of dof's, which is unrealistic. Furthermore, all the nice properties of RB, including closed form solution, and low order matrix inversion are no longer valid. In this section we follow the results of Halevi and Bucher [19] that adopted the parametric modeling philosophy, but used different means to solve the problem.

5.2 Implicit Parameterization by Weighting Matrix Adjustment

Considering for simplicity a diagonal weighting matrix $W=diag\{w_i\}$, the cost function in eq. (11) is a weighted sum of the deviations of K from K_A. More specifically,

$$J = \sum_{i=1}^{n} \sum_{j=1}^{n} w_i^{-1} w_j^{-1} \left(K_{ij} - K_{A,ij} \right)^2 \tag{20}$$

If w_i is large, a unit change in the i-th degree of freedom results in a small contribution to J and vice versa. The optimization machinery will therefore automatically try to direct the changes to areas with large w_i and to avoid changes in areas where w_i is small. The conclusion from this intuitive analysis is that the weighting matrix W provides means of restricting the updating to the desired dof's in a soft manner. This set of dof's can be a result of either *a priori* or *a posteriori* knowledge. Sometimes it is clear that certain areas are better modeled than others. In such cases w_i^{-1} is a confidence measure of the model of that dof.

In this paper we consider the other possibility, i.e. *a priori* equal confidence in all dof's. The selection of the weighting matrix is then based on detection, rather than

expectation, and the specific tool which is going to be used is error localization. There exist a variety of error localization methods, e.g. [8, 25], and the output of all of them is, or can be translated to, a numerical value describing the magnitude of the error in that dof. We consider here the eigenvalue equation error as such indicator but other, more sophisticated, methods can be applied as well without any effect on the general scheme. Let d_i be proportional to the norm of the *i-th* row of the eigenvalue equation error matrix.

$$d_i = c_i \left\| \left(K_A \Phi - M \Phi \Omega^2 \right)_{i,*} \right\| \tag{21}$$

where the c_i are constants that can be used as normalization factors for non-identical dof's, such as displacements and angles, or to contain any *a priori* knowledge. W is selected as

$$W = diag\left\{ d_i^{\beta} \right\} \tag{22}$$

Where β is a parameter for adjusting the updating procedure. For β=0, W becomes the identity matrix and all dof's are treated equally. As β increases, areas with larger errors in the eigenvalue equation are more and more emphasized. In case the mode-shapes contain some noise, too large values of β tend to increase it, hence in general there is a finite optimal β.

5.3 *A Posteriori* Connectivity Assignment

The main idea of connectivity constrained reference basis (CCRB) [19] is applying the connectivity constraints *after* the standard GRB updating procedure. Clearly this is wrong from pure mathematical considerations; however from practical point of view the procedure combines the best features of both reference basis and parametric approaches. The starting point is a GRB update K, given by eq. (12) or (14), that does not satisfy the connectivity constraints. This update will be replaced by K_{con} that satisfies the connectivity constraints, and is closest to K. The following optimization problem yields that K_{con}.

$$\begin{aligned} \min_{\alpha} \quad & J = \left\| K_{con} - K \right\|_F^2 \\ s.t. \quad & K_{con} = K_A + \sum_{i=1}^{p} \alpha_i K_i \end{aligned} \tag{23}$$

This problem is easily transformed into least squares with the closed form solution

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} tr(K_1 K_1) & \cdots & tr(K_1 K_p) \\ \vdots & \ddots & \vdots \\ tr(K_p K_1) & \cdots & tr(K_p K_p) \end{bmatrix}^{-1} \begin{bmatrix} tr(K_1 (K - K_A)) \\ \vdots \\ tr(K_p (K - K_A)) \end{bmatrix} \tag{24}$$

Let us define now two measures of the quality of the update. The 'updating cost' J_{up} is the distance between the updated stiffness matrix and the true one.

$$J_{up} = \frac{\left\| K_T - K_{con} \right\|_F}{\left\| K_{con} \right\|_F} \tag{25}$$

This cost requires knowledge of the unknown K_T, thus cannot be calculated in real applications. It can serve in simulations as a measure of the quality of the updated model.

The difference between K, that satisfies all the updating requirements, except for connectivity, and K_{con} can be regarded as 'payment' for invoking connectivity. It is termed the 'connectivity cost'. Several criteria can be defined for that purpose [19], e.g. the deviation in natural frequencies (unlike K, K_{con} is not completely compatible with the measured eigendata). No matter how the cost J_{con} is defined, it involves only known quantities and can be calculated in real applications. The cost used here is again the normalized distance, which is a measure of how much the assumed connectivity was violated in the updated K.

$$J_{con} = \frac{\left\| K - K_{con} \right\|_F}{\left\| K_{con} \right\|_F} \tag{26}$$

The entire updating procedure still has one free parameter, the power β in the weighting matrix W in eq. (22). Since K depends on W, and K_{con} on K, it follows that with given analytical model and measurement data, the connectivity cost is a function of β.

$$J_{con}(\beta) = J_{con}\left(W = diag\left\{ d_i^{\beta} \right\} \right) \tag{27}$$

To decide what value of β to use, consider first the following scenario: The correct connectivity is known but not imposed, and the unconstrained updated model happens to conform to it. This is of course the ideal situation. The fundamental premise of the suggested algorithm is therefore stated as follows.

Unconstrained updated models with smaller connectivity cost are closer to the true model.

The meaning of that idea is that a weighting matrix is considered to be better if it leads to an unconstrained updated stiffness matrix which is closer to the assumed connectivity. In other words, the amount of connectivity violation of K can serve as an indicator for the quality of the β. This abstract reasoning was confirmed by extensive simulations showing good correlation between J_{con} that can be computed and J_{up} that is available only in simulations. The algorithm is based on searching for β which minimizes J_{con}. With perfect measurements, larger values of β always yield better results. When the measurements contain some noise, larger β tend to increase it, and usually there is a finite optimal value.

Consider once more the example and define as parameters the changes in all the springs. Fig. 2 shows the updating cost and the connectivity cost in two cases, perfect

and noisy measurement. The correlation between the unknown J_{up} and the known J_{con} is evident. Without noise, any $\beta > 0.3$ is practically optimal. With β=0.5 the algorithm found exactly the correct values of 70 for k_2 and 50 for the others. With 1% noise in the modeshapes, the optimum occurs at β=3.3 with the following reasonably accurate values for the springs

$$k_1 \div k_6 = \{49.19, 70.27, 49.05, 50.07, 49.98, 50.03\}$$

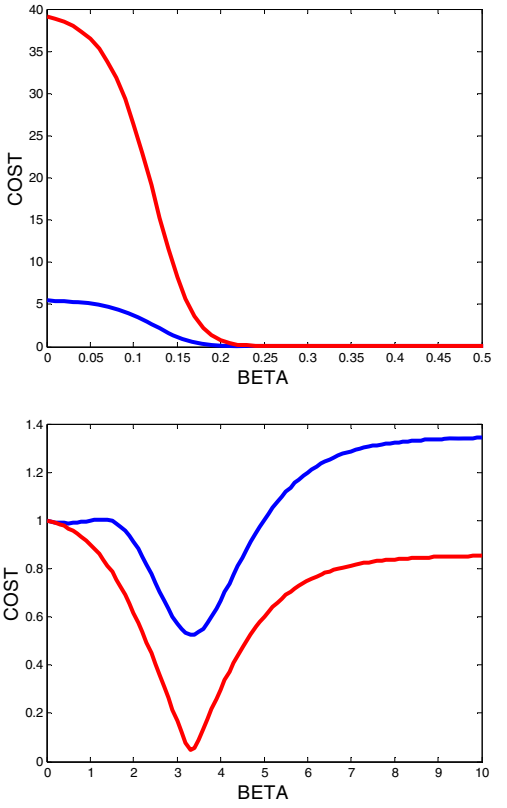


Fig. 2. The updating cost J_{up} (red) and the connectivity cost J_{con} (blue). Top – without noise, Bottom (normalized) – 1% noise in modeshapes.

6 Direct Method – Manifold Distance Minimization

The discussion in the previous section revolved around the inherent conflict, existing in practically all methods of model updating, between satisfying the characteristic equation and complying with the connectivity that is known to exist in the model. CCRB tries to bridge that gap by going through GRB. In [20], Halevi, Vilensky and

Datta went one step further and suggested a direct method called manifold distance minimization (MDM). The main results of that paper are presented in this section.

Consider two sets of stiffness matrices. The first is the set of all matrices that satisfy exactly the eigenvalue equation with the m measured natural frequencies and modeshapes.

$$S_E = \left\{ K : K\Phi = M\Phi\Omega^2 \right\} \tag{28}$$

It can be shown that an equivalent definition of the set is

$$S_E = \left\{ K : K = M\Phi\Omega^2\Phi^T M + BXB^T, \qquad X = X^T \geq 0 \right\} \tag{29}$$

where B=null(Φ^T). S_E is a linear manifold in the elements of X. Since N, the number of dof's, is in practice very large, and m<<N, the actual calculation of B can require heavy computation. However, it is shown later that B needs not be calculated at all. The second set is that of all matrices that satisfy the required connectivity by having the standard linear parameterization.

$$S_C = \left\{ K : K = K_A + \sum_{i=1}^{p} \alpha_i K_i \right\} \tag{30}$$

If the two manifolds intersect, there exists a stiffness matrix with the correct connectivity that is in full agreement with the measurement and the problem has a unique and perfect solution. Since this is almost never the case, we look for the point in S_C that is closest to S_E, i.e., the minimum distance between the manifolds. We therefore define, with some positive weighting matrix W, the following optimization problem.

$$\begin{aligned} &\min_{K_1, K_2} \; J = \left\| W^{-1/2}(K_1 - K_2)W^{-1/2} \right\|_F^2 \\ &s.t. \quad K_1 \in S_E \\ &\qquad K_2 \in S_C \end{aligned} \tag{31}$$

In more specific terms, the problem is

$$\min_{X,\alpha} \qquad J = \left\| W^{-1/2} \left(\left(K_A + \sum_{i=1}^{p} \alpha_i K_i \right) - \left(M\Phi\Omega^2\Phi^T M + BXB^T \right) \right) W^{-1/2} \right\|_F^2 \tag{32}$$

An explicit expression for the solution of the optimal parameters is given by

$$\begin{bmatrix} \alpha_1 \\ \vdots \\ \alpha_p \end{bmatrix} = \begin{bmatrix} tr(K_1 W \tilde{K}_1) & \cdots & tr(K_1 W \tilde{K}_p) \\ \vdots & \ddots & \vdots \\ tr(K_p W \tilde{K}_1) & \cdots & tr(K_p W \tilde{K}_p) \end{bmatrix}^{-1} \begin{bmatrix} tr(K_1 W \tilde{K}_U) \\ \vdots \\ tr(K_p W \tilde{K}_U) \end{bmatrix} \tag{33}$$

where

$$K_U = K_A - M\Phi\Omega^2\Phi^T M \quad , \quad P = I - W\Phi\left(\Phi^T W\Phi\right)^{-1}\Phi^T$$
$$\tilde{K}_i = K_i - PK_iP^T \quad , \quad \tilde{K}_U = K_U - PK_UP^T \tag{34}$$

Notice that the actual calculation of the null space B is not required as the projection P is defined by the problem data. In [20] the method was compared to connectivity constrained reference basis (CCRB) and to minimization of the error in the constitutive equation (MECE). It was shown that MDM covers a larger set of potential matrices and that it is more efficient computationally and less sensitive to noise.

It is clear from the MDM problem definition that if there is a stiffness matrix with the correct connectivity that satisfies the characteristic equation, the method will find it. This is, by the way, not true for many existing methods, including reference basis and sensitivity. The appropriate comparison is therefore in the presence of noise. The method was applied to the same example, with different levels of noise and was compared to CCRB and its iterative version (see [19]). The results, which are averages of extensive simulation, are shown in Fig. 3. It is evident that the error in MDM is much smaller. Similar results were obtained for MECE. A qualitative explanation is that in CCRB the noisy data is used twice, not only as the modeshape, but also for computing the optimal weight for the generalized reference basis problem. Similar argument can be made about MECE. However these points still need to be investigated rigorously.

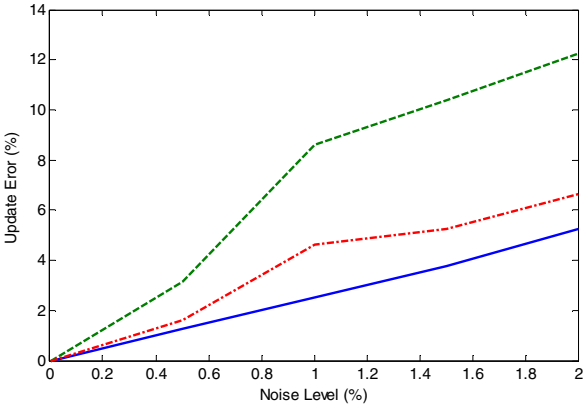


Fig. 3. Update error vs. noise level. MDM – blue, CCRB – green, CCRB with 25 iterations – red.

7 Conclusion

The reference basis model updating method was revisited. The seminal results of Baruch and Bar-Itzhack were discussed and several extensions of the method were presented. Most of these later results try to combine, one way or another, the mathematical and computational advantages of reference basis with the requirement that the updated model should have a certain structure. An example that was used for all the

methods demonstrated the progress from no connectivity in classical reference basis in section 3, through ad hoc weighting selection in generalized reference basis in section 4 to the systematic connectivity constrained reference basis in section 5. Though not sharing exactly the same philosophy, the manifold distance minimization in section 6 can be regarded as reference basis because one of the two manifolds is defined by satisfying the characteristic equation accurately. As a concluding remark, it is worth noting that the principles of reference basis, which was developed for conservative systems, were applied also to systems with damping [26-28].

References

1. Baruch, M., Bar-Itzhack, I.Y.: Optimal Weighted Orthogonalization of Measured Modes. AIAA Journal 16(4), 346–351 (1978)
2. Friswell, M.I., Mottershead, J.E.: Finite Element Model Updating in Structural Dynamics. Kluwer Academic Publication, Dordrecht (1995)
3. Natke, H.G. (ed.): Identification of Vibrating Structures. Springer (1982)
4. Sastieri, A., D'Ambrogio, W.: Why Be Modal?, How To Avoid The Use of Modes in The Modification of Vibrating System. International Journal of Analytical and Experimental Modal Analysis 4(1), 25–30 (1989)
5. Lallement, G., Piranda, J.G., Cogan, S.: Parametric Correction of Finite Element Models by Minimization of an Output Residual: Improvement of The Sensitivity Method. In: Proceedings of the 9th International Modal Analysis Conference (IMAC) Florence, Italy, pp. 363–368 (1991)
6. Link, M.: Updating of Analytical Models – Basic Procedures and Extensions. In: Silva, J.M.M., Maia, N.M.M. (eds.) Modal Analysis and Testing. Nato Science Series. Kluwer Academic Publication (1999)
7. Maia, N.M.M., Silva, J.M.M. (eds.): Theoretical and Experimental Modal Analysis. Research Study Press, Baldoc (1997)
8. Fissette, E., Ibrahim, S., Sravrinidis, C.: Error Localization and Updating of an Analytic Model Using a Force Balance Method. In: Proceedings of the 6th International Modal Analysis Conference (IMAC), Orlando, FL, pp. 1063–1070 (1988)
9. Alvin, F.A.: Finite Element Model Updating via Bayesian Estimation and Minimization of Dynamic Residuals. AIAA Journal 35(5), 879–886 (1997)
10. Tarazaga, P.A., Halevi, Y., Inman, D.J.: The Quadratic Compression Method for Model Updating and Its Noise Filtering Properties. Mechanical Systems and Signal Processing 21(1), 58–73 (2007)
11. Baruch, M.: Optimal Correction of Mass and Stiffness Matrices Using Measured Modes. AIAA Journal 20(11), 1623–1626 (1982)
12. Baruch, M.: Methods of Reference Basis for Identification of Linear Dynamic Structures. AIAA Journal 22(4), 561–564 (1984)
13. Berman, A.: Comments on 'Optimal Weighted Orthogonalization of Measured Modes'. AIAA Journal 17(8), 927–928 (1979)
14. Berman, A.: Mass Matrix Correction using an Incomplete Set of Measured Modes. AIAA Journal 17(11), 1147–1148 (1979)
15. Berman, A., Nagy, E.J.: Improvement of a Large Analytical Model using Test Data. AIAA Journal 21(8), 1168–1173 (1983)

16. Kabe, A.M.: Stiffness Matrix Adjustment using Mode Data. AIAA Journal 23(9), 1431–1436 (1985)
17. Kammer, D.C.: Optimum Approximation for Residual Stiffness in Linear System Identification. AIAA Journal 26(1), 104–112 (1988)
18. Kenigsbuch, R., Halevi, Y.: Model Updating in Structural Dynamics: A Generalised Reference Basis Approach. Mechanical Systems and Signal Processing 12(1), 75–90 (1998)
19. Halevi, Y., Bucher, I.: Model Updating via Weighted Reference Basis with Connectivity Constraints. Journal of Sound and Vibration 265, 561–581 (2003)
20. Halevi, Y., Vilensky, D., Datta, B.N.: Model Updating via Manifold Distance Minimization. In: Proceedings of ISMA, Leuven, Belgium (2010)
21. Guyan, R.J.: Reduction of Stiffness and Mass Matrices. AIAA Journal 3(2), 280 (1965)
22. O'Callahan, J., Avitable, P., Riemer, R.: System Equivalent Reduction Expansion Process. In: Proceedings of the 7th International Modal Analysis Conference (IMAC), Las Vegas, NV, pp. 17–21 (1989)
23. Levin-West, M., Minham, M., Kissil, A.: Evaluation of Mode Shape Expansion Techniques for Prediction. NASA report, pp. 94–158 (1994)
24. Halevi, Y., Morales, C.A., Inman, D.J.: Combined Expansion and Orthogonalization of Experimental Modeshapes. ASME Journal of Vibration and Acoustics 127, 188–196 (2005)
25. Lieven, N.A., Ewins, D.J.: Error Localization and Updating Finite Element Models Using Singular Value Decomposition. In: Proceedings of the 8th International Modal Analysis Conference (IMAC), Orlando, FL, pp. 768–773 (1990)
26. Minas, C., Inman, D.J.: Identification of a Nonproportional Damping Matrix from Incomplete Modal Information. ASME Journal of Vibration and Acoustics 113, 219–224 (1991)
27. Halevi, Y., Kenigsbuch, R.: Updating of the Complex Modeshapes and the Damping Matrix. Inverse Problems in Engineering 8, 143–162 (2000)
28. Datta, B.N., Deng, S., Sokolov, V.O., Sarkissian, D.R.: An Optimization Technique for Damped Model Updating with Measured Data Satisfying Quadratic Orthogonality Constraint. Mechanical Systems and Signal Processing 23(6), 1759–1772 (2009)

Stochastic Estimation and Control for Linear Systems with Cauchy Noise

Jason L. Speyer[1], Moshe Idan[2], and Javier Fernández[1]

[1] Mechanical and Aerospace Engineering,
University of California, Los Angeles
{speyer,jhf}@seas.ucla.edu
[2] Aerospace Engineering,
Technion - Israel Institute of Technology
moshe.idan@technion.ac.il

Abstract. The light-tailed Gaussian paradigm has dominated the foundation of estimation and control algorithms. However, in many realistic applications the system can experience large impulsive noises far more often than the Gaussian would admit. In this paper the Cauchy probability density function (pdf) is used to develop a new class of estimation and control algorithms. First, the scalar Cauchy estimation problem is addressed which entails the generation of the state pdf conditioned on the measurement history. Next, based on this Cauchy conditional pdf, a model predictive optimal controller is developed. Finally, the vector stated estimator is derived by recursively propagating the characteristic function of the unnormalized conditional pdf through measurement updates and dynamic state propagation. The conditional mean and variance are easily computed from the first and second derivatives of this characteristic function.

Keywords: Estimation, Stochastic control, Optimal Controller Synthesis for Systems with Uncertainties, Heavy-Tailed Distributions.

1 Introduction

The Gaussian paradigm has dominated the foundation of estimation and control algorithms for over fifty years. These algorithms were developed for linear system models that assume additive Gaussian noises. They extended classical control system design from single-input/single-output using frequency domain analysis to multi-inpt/multi-output systems based on state space formulations. State space methods produced general estimation and control algorithms such as the Kalman filters, the linear-quadratic-Gaussian (LQG or H_2) controller, and the linear-exponential-Gaussian (LEG or H_∞) controller [6]. The Kalman filter is optimal only for Gauss-Markov processes, where the filter gains operate linearly on the residuals. The LQG and LEG controllers operate on a state estimate or the worst case state estimate, respectively, leading to a linear controllers with

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_6

respect to the measurements. These controllers assume Gaussian uncertainty and noises where most of the probability is concentrated near the mean.

However, in many realistic applications, i.e., radar and sonar measurements, atmospheric turbulence, adversarial motion effecting missile guidance, or abrupt changes in financial markets, the system experiences large impulsive noise far more often then the light-tailed Gaussian would admit [10]. As a result these controllers track impulsive noise inputs, be it process or measurement noise, leading to large deterioration in performance. To mitigate these effects, ad hoc procedures are introduced into the algorithms which may help, but sometimes induce even poorer performance.

In this work, the Cauchy probability density function (pdf) is suggested to model the system uncertainty, hence leading to the development of a new class of estimation and control algorithms. Heavy-tailed pdfs capture physical phenomena with a more impulsive character [11]. The Cauchy pdf is a member of a class of heavy-tailed distributions that are represented by their characteristic functions, called symmetric α-stable distributions (or sub-Gaussian). The characteristic function for the symmetric α-stable distributions [5], in its simplest form, is $\phi(\nu) = \exp(-\sigma^\alpha|\nu|^\alpha)$. For $\alpha = 2$, the characteristic function represents the Gaussian, for $\alpha = 1$ it represents the Cauchy pdf. The pdf can be expressed in closed form only for $\alpha = .5$, 1, 2, and can be shown to have an infinite variance for $\alpha \in (0, 2)$. This paper summarized the results we have attained in the area of estimation and control of dynamic linear discrete systems with additive Cauchy noises.

The paper is structured as follows. In section 2 we formulate the scalar dynamic estimation problem. The scalar Cauchy estimator is derived in sections 3 and 4. Comparing the Cauchy solution to the Gaussian reveals interesting differences between them. [2]. A Cauchy model-predictive controller (MPC) based on the conditional pdf of a *scalar* state [8] is presented in section 5. In its simplest form, when addressing a single measurement and a one-step prediction, this controllers exhibits a nonlinear behavior in strong contrast to the linear Gaussian controller. Then, in the multi-stage formulation, the advantage of Cauchy controller is demonstrated when handling impulsive measurement noise, outperforming the linear Gaussian model-predictive controller. The vector state estimation problem is addressed in section 6. This solution is based on propagating the characteristic function of the unnormalized conditional pdf [3]. Numerical results for the two-state problem is given and contrasted with an equivalent Gaussian estimator [9]. Conclusions and future research directions are offered in section 7.

2 Scalar Estimation Problem Formulation

Consider the discrete-time scalar linear dynamic system

$$x_{k+1} = \Phi x_k + \Gamma u_k + w_k \tag{1}$$

$$z_k = H x_k + v_k \tag{2}$$

where x_k is the scalar state, z_k is the measurement, Φ, Γ, H are know parameters, u_k is the scalar control, w_k is the process noise and v_k is the measurement noise. The system uncertainties are assumed to be characterized by Cauchy pdfs

$$f_{X_1}(x_1) = \frac{\alpha/\pi}{(x_1 - \bar{x}_1)^2 + \alpha^2}, \tag{3}$$

$$f_{W_k}(w_k) = \frac{\beta/\pi}{w_k^2 + \beta^2}, \quad f_{V_k}(v_k) = \frac{\gamma/\pi}{v_k^2 + \gamma^2}, \tag{4}$$

where the parameters x_1, α, β, γ are given.

To compare Cauchy and Gaussian pdfs a least square fit is made between Cauchy and Gaussian,

$$f_X^G(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}, \; f_X^C(x) = \frac{\gamma/\pi}{x^2 + \gamma^2} \tag{5}$$

with the result that the parameters are related as $\sigma \approx 1.4\gamma$. The least square fit is shown in Fig. 1 where the heavy tail of the Cauchy pdf is to be contrasted with the light tail of the Gaussian pdf.

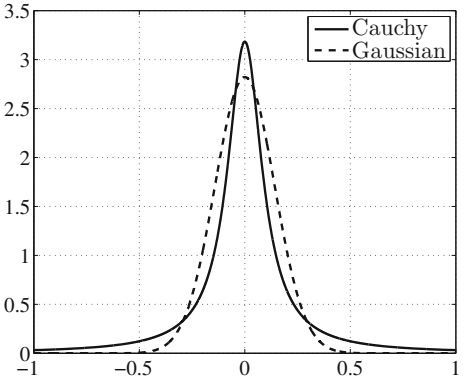


Fig. 1. Least square fit between Cauchy and Gaussian pdfs

Our objective is to find the conditional mean of the system state given the measurement history $y_k = \{z_1, z_2, \cdots, z_k\}$ or equivalently, the minimum conditional-variance estimator

$$\hat{x}(k) = E[x_k | y_k], \tag{6}$$

where $E[\cdot]$ denotes the expectation operator.

3 One Step Measurement and Time Update

First the pdf conditioned on a single measurement is obtained. Then, this conditional pdf is propagated to the next time step. It is then observed that a similar structure can be associated with these pdfs. This notion is used to generate the pdf conditioned on the measurement history at any time step k.

3.1 Measurement Update

From the measurement equation at $k = 1$

$$z_1 = Hx_1 + v_1 \tag{7}$$

the conditional mean $\hat{x}_1 = E\left[x_1|z_1\right]$ is to be determined. The conditional pdf is obtained by Bayes' rule as

$$f_{X_1|Z_1}(x_1|z_1) = \frac{f_{X_1,Z_1}(x_1, z_1)}{f_{Z_1}(z_1)}. \tag{8}$$

Since z_1 is known, $f_{Z_1}(z_1)$ is also known and the joint pdf, referred to as the un-normalized conditional pdf, is

$$f_{X_1,Z_1}(x_1, z_1) = f_{X_1}(x_1)\, f_{V_1}(z_1 - Hx_1)\,. \tag{9}$$

By integrating over x_1, the marginal pdf is computed analytically to be

$$f_{Z_1}(z_1) = \frac{(|H|\alpha + \gamma)/\pi}{(z_1 - H\bar{x}_1)^2 + (|H|\alpha + \gamma)^2}. \tag{10}$$

Then, the conditional pdf becomes

$$f_{X_1|Z_1}(x_1|z_1) = \frac{\alpha\gamma/\pi}{|H|\alpha + \gamma} \frac{(z_1 - H\bar{x}_1)^2 + (|H|\alpha + \gamma)^2}{\left[(x_1 - \bar{x}_1)^2 + \alpha^2\right]\left[(z_1 - Hx_1)^2 + \gamma^2\right]}, \tag{11}$$

where the denominator is a quartic in x_1. Therefore, even though the *a priori* densities have infinite moments, this conditional pdf has two *computable* moments, $E\left[x_1|z_1\right]$ and $E\left[x_1^2|z_1\right]$. After some manipulations, the conditional mean is found to be a linear function of the measurement as

$$\hat{x}_1 = E\left[x_1|z_1\right] = \bar{x}_1 + \frac{\alpha\,\mathrm{sign}(H)}{|H|\alpha + \gamma}\,(z_1 - H\bar{x}_1)\,. \tag{12}$$

The conditional error variance is found to be a quadratic function of the measurement as

$$E\left[\tilde{x}_1^2|z_1\right] = \frac{\alpha\gamma}{|H|}\left[\frac{(z_1 - H\bar{x}_1)^2}{(|H|\alpha + \gamma)^2} + 1\right], \tag{13}$$

where $\tilde{x}_1 = x_1 - \hat{x}_1$. The conditional variance being measurement dependant should be contrasted with the conditional variance for Gaussian pdf, which does not depend on the measurement and is known *a priori*.

For numerical values $\alpha = 3$, $\bar{x}_1 = 5$, $H = 2$, $\gamma = 5$, and $z_1 = -3$, the initial state, measurement noise, and measurement updated pdfs are depicted in Fig. 2. Shown also are the the estimate, $\hat{x}_1 = 1.45$, and the estimation error standard deviation, $\sqrt{E[\tilde{x}_1^2|z_1]} = 4.24$. It is interesting to note that unlike in the Gaussian, the conditional pdf is non-symmetric and may even become bimodal when the difference between z_1 and $H\bar{x}_1$ is large.

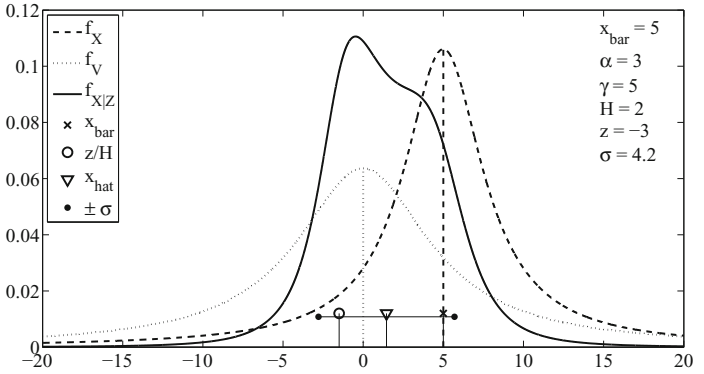


Fig. 2. Conditional pdf for one step measurement update

3.2 Time Update

Starting with $f_{X_1|Z_1}(x_1|z_1)$ and the system dynamics

$$x_2 = \Phi x_1 + \Gamma u_1 + w_1 \tag{14}$$

the time updated density $f_{X_2|Z_1}(x_2|z_1)$ is computed using the Chapman-Kolmogorov equation

$$f_{X_2|Z_1}(x_2|z_1) = \int_{-\infty}^{\infty} f_{X_2|X_1}(x_2|x_1)\, f_{X_1|Z_1}(x_1|z_1)\, dx, \tag{15}$$

where the transition pdf is

$$f_{X_2|X_1}(x_2|x_1) = f_{W_1}(x_2 - \Phi x_1 - \Gamma u_1).$$

The resulting time propagated conditional pdf is

$$f_{X_2|Z_1}(x_2|z_1) = \frac{A\left(x_2 - B\right)^2 + C^2}{\left[\left(x_2 - \bar{x}_{21}\right)^2 + \alpha_1^2\right]\left[\left(x_2 - \bar{x}_{22}\right)^2 + \alpha_2^2\right]}. \tag{16}$$

A, B and C are complicated functions of the various system parameters and the signals z_1 and u_1. Since the exact expressions do not provide valuable information or insight, they are not made explicit. Note that this conditional pdf, which is referred to as the *a priori* conditional pdf, has no computable moments [2].

3.3 Conditional pdf Factorization

The conditional pdf for the one step measurement update (11) and time propagation process (16) can be put into factored form by using partial fractions as

$$f_{X_1|Z_1}(x_1|z_1) = \frac{a_{1,1}x_1 + b_{1,1}}{(x_1 - \bar{x}_1)^2 + \alpha^2} + \frac{a_{1,2}x_1 + b_{1,2}}{(z_1 - Hx_1)^2 + \gamma^2} \tag{17}$$

$$f_{X_2|Z_1}(x_2|z_1) = \frac{a_{2,1}x_2 + b_{2,1}}{(x_2 - \bar{x}_{21})^2 + \alpha_1^2} + \frac{a_{2,2}x_2 + b_{2,2}}{(x_2 - \bar{x}_{22})^2 + \alpha_2^2}. \tag{18}$$

where

$$\bar{x}_{21} = \Phi\bar{x}_1 + \Gamma u_1 \qquad \alpha_1 = |\Phi|\alpha + \beta \tag{19a}$$

$$\bar{x}_{22} = \frac{\Phi}{H}z_1 + \Gamma u_1 \qquad \alpha_2 = \left|\frac{\Phi}{H}\right|\gamma + \beta \tag{19b}$$

The parameters $a_{j,i}, b_{j,i}$, $j = 1, 2$, $i = 1, 2$ can be determined from (11) and (16). Motivated by the structure of these rational pdfs, we propose to represent the conditional pdf of x at any time step k in a factored form given by

$$f_X(x) = \sum_{i=1}^{N} \frac{a_i x + b_i}{(x - \sigma_i)^2 + \omega_i^2}, \tag{20}$$

where N is to be explicitly determined in the sequel. Here, for presentation simplicity, in the above we have suppressed the dependance of the state x on the time index k and the fact that this pdf may be a conditional one. The conditions for $f_X(x)$ in (20) to be a valid pdf are

$$\sum_{i=1}^{N} a_i = 0, \quad \pi \sum_{i=1}^{N} \frac{a_i\sigma_i + b_i}{\omega_i} = 1. \tag{21}$$

For the pdf to have a well defined first moment, the condition is

$$\sum_{i=1}^{N} (2a_i\sigma_i + b_i) = 0. \tag{22}$$

The pdf will have a well defined second moment if

$$\sum_{i=1}^{N} \left[a_i\left(3\sigma_i^2 - \omega_i^2\right) + 2b_i\sigma_i\right] = 0. \tag{23}$$

For details see [2]. Note, it is immediate to verify that these conditions hold for the initial pdf, for which $N = 1$, $a_1 = 0$, $b_1 = \alpha/\pi$, $\sigma_1 = \bar{x}_0$, and $\omega_1 = \alpha$.

4 The General Form of the Cauchy Estimator

Based on the form of (20) motivated by the partial fraction decomposition of (17) and (18), we propose to represent the conditional pdf of x_k given past data y_{k-1}, i.e., before the z_k measurement is processed, in a factored form given by the propagated conditional pdf

$$f_{X_k|Y_{k-1}}(x_k|y_{k-1}) = \sum_{i=1}^{k+1} \frac{a_i(k|k-1)x_k + b_i(k|k-1)}{\left(x_k - \sigma_i(k|k-1)\right)^2 + \omega_i^2(k|k-1)} \tag{24}$$

where $y_{k-1} = \{z_1 \cdots z_{k-1}\}$. The linear update formulas for $a_i(k|k-1)$ and $b_i(k|k-1)$ are given in [2]. Similarly, for the measurement update, the conditional pdf of x_k given the measurement history y_k as

$$f_{X_k|Y_k}(x_k|y_k) = \sum_{i=1}^{k+2} \frac{a_i(k|k)x_k + b_i(k|k)}{\left(x_k - \sigma_i(k|k)\right)^2 + \omega_i^2(k|k)}. \tag{25}$$

Note that the number of terms in the sum increases by one during a measurement update. The updates for the parameters is $\sigma_i(k|k) = \sigma_i(k|k-1)$, $\omega_i(k|k) = \omega_i(k|k-1), i = 1, \ldots, k+1$, $\sigma_{k+2}(k|k) = z_k/H$, $\omega_{k+2}(k|k) = \gamma/|H|$. Simple, linear expressions for $a_i(k|k), b_i(k|k)$, $i = 1, \ldots, k+2$ are given in [2]. The dynamic properties of $a_i(k|k), b_i(k|k)$, $i = 1, \ldots, k+2$ are used to show the decay of these parameters, and are thereby used to prune terms from the sum.

Using the conditional pdf of x_k given in (25), the conditional mean, conditional second moment and conditional error variance can be calculated as

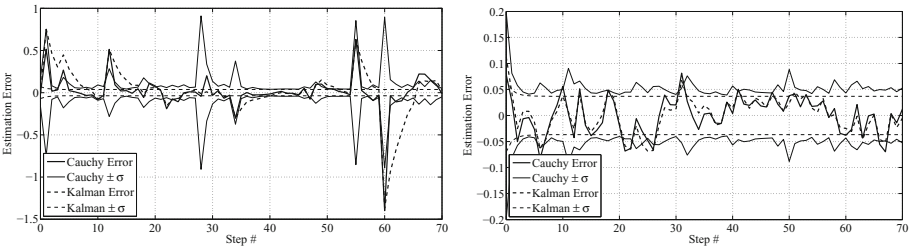
$$E(x_k|y_k) = \hat{x}(k|k) = \pi \sum_{i=1}^{k+2} \frac{a_i\left(\sigma_i^2 - \omega_i^2\right) + b_i\sigma_i}{\omega_i}, \tag{26}$$

$$E\left(x_k^2|y_k\right) = \pi \sum_{i=1}^{k+2} \frac{\left(a_i\sigma_i + b_i\right)\left(\sigma_i^2 - \omega_i^2\right) - 2a_i\sigma_i\omega_i^2}{\omega_i}, \tag{27}$$

$$E\left(\tilde{x}_k^2|y_k\right) = E\left(x_k^2|y_k\right) - \hat{x}^2(k|k). \tag{28}$$

4.1 Numerical Example

A numerical example is given with the system data $\bar{x}_1$=5, α=0.5, Φ=0.9, Γu_k=1, β=0.02, H=2, γ=0.1. The Cauchy conditional estimator is first compared with a Kalman filter when the simulation uses Cauchy noise, as shown in Fig. 3(a). The least square fit of the Gaussian densities with the Cauchy densities produces the Kalman filter design, i.e., σ_{x_1}=1.4α, σ_w=1.4β, σ_v=1.4γ. In the Cauchy simulation the Cauchy estimator is the exact minimum variance estimator. Even though both filters react to impulsive noise, the Cauchy estimation error reduces very rapidly compared to the Gaussian estimator. Note that the Cauchy standard deviation reacts to impulsive noise, whereas the Gaussian error variance is *a priori* known.



(a) Cauchy Noises Simulation (b) Gaussian Noises Simulation

Fig. 3. Performance of the Cauchy and Gausian estimators

The Cauchy conditional estimator is compared with a Kalman filter when the simulation uses Gaussian noise, as shown in Fig. 3(b). Both estimators perform in a very similar manner. Furthermore, even though the standard deviation for the cauchy estimator is a bit ragged in response to the measurement and process noise sequence, it closely upper bounds the Gaussian standard deviation, which is optimum for the Gaussian simulation.

As was shown in section 4, the number of terms representing the conditional pdf of interest is constantly growing. This may cause numerical difficulties. It is shown in [2] that the linear update formulas for $a_i(k|k-1)$ and $b_i(k|k-1)$ are exponentially convergent in k for a constant i. When $a_i(k|k-1)$ and $b_i(k|k-1)$ become sufficiently small, their associate term can be removed from the sum in (25), thus enabling a low order approximation of the conditional pdf representation. Simulation results carried out with such truncation are practically identical to those presented earlier. The number of terms used in the truncated sum for the conditional pdf is plotted in Fig. 4, showing that they are fluctuating slightly around the values of 18 and 10 for the two truncation cases examined. Most importantly it was demonstrated that the difference between the estimation results attained with the non-truncated and truncated pdfs, i.e., numerical truncation errors, do not increase with time. Hence, using such truncations allows propagating the Cauchy estimator for any number of steps while keeping the number of terms in the pdf sum nearly constant.

5 Stochastic Control Problem for Cauchy Uncertainties

In this section a stochastic control problem is formulated and a controller developed for a scalar linear system with additive Cauchy noise. The performance index is chosen so that the unconditional expectation remains finite so it can be determined *a priori*. Since expectation and minimization can be interchanged, the conditional expectation of the performance index is minimized. This expectation is found in closed-form using the conditional pdf of (25). However, the conditional expectation of the performance index is a complicated function of the control and does not lend itself to a dynamic programming solution. Therefore, a model predictive controller (MPC) is proposed which has sub-optimal

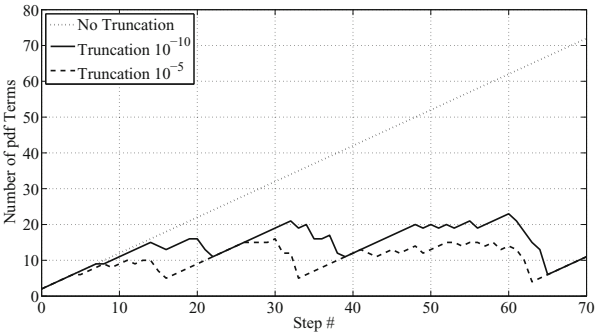


Fig. 4. Cauchy Noises - Decay of pdf Sum Elements

performance. For zero mean noises and initial uncertainty, the performance of this controller is similar to the linear-exponential-Gaussian MPC when the measurements are small and dramatically different when the measurements are large.

5.1 Deterministic-Stochastic Decomposition of the State Dynamics

For the control problem, we consider the dynamic scalar stochastic system

$$x_k = \Phi x_{k-1} + u_{k-1} + w_{k-1} \tag{29}$$

$$z_k = \mathrm{H} x_k + v_k \tag{30}$$

with measurement history

$$y_k = \{z_1, \ldots, z_k\}. \tag{31}$$

To simplify the subsequent derivations, the system dynamics is decomposed into a dynamic system forced by the control and a dynamic system forced by the underlying random variables. Given this decomposition, we show that the control has to be adaptive to only the σ-algebra generated by the measurement history associated with the stochastic part of the state decomposition.

Consider the linear, discrete-time, scalar stochastic system of (29) with the measurement history given by (31). Let u_k be adaptive to the filtration σ-algebra σ_k generated by the measurement history z_k. Filtration implies that the collection of σ-algebras σ_k have the property that if $j \leq k$, then $\sigma_j \subseteq \sigma_k$ [1]. Therefore, filtration is the evolution of the σ-algebra generated by measurement history through time. Adaptation means that the control is a measurable function of events on this σ-algebra, i.e., this ensures that the control sequence is causal. Now consider the decomposition $x_k = \tilde{x}_k + \bar{x}_k$ where

$$\tilde{x}_k = \Phi \tilde{x}_{k-1} + w_{k-1}, \quad \tilde{z}_k = H \tilde{x}_k + v_k, \tag{32a}$$

$$\bar{x}_k = \Phi \bar{x}_{k-1} + u_{k-1}, \quad \bar{z}_k = H \bar{x}_k. \tag{32b}$$

Here, $\tilde{x}_k$ and $\tilde{z}_k$ are the state and the measurement of the subsystem containing all the underlying random variables, i.e., w_k, v_k, with pdfs given in (4), and the initial condition $\tilde{x}_0$ that is Cauchy distributed with zero median, i.e.,

$$f_{X_1}(\tilde{x}_1) = \frac{\alpha/\pi}{\tilde{x}_1^2 + \alpha^2}. \tag{33}$$

Similarly, $\bar{x}_k$ and $\bar{z}_k$ are the state and measurement of a dynamic system driven by u_k with initial condition $\bar{x}_1$.

The measurement history can be decomposed as $y_k = \tilde{y}_k + \bar{y}_k$ where

$$\tilde{y}_k = \{\tilde{z}_1, \cdots, \tilde{z}_k\}, \quad \bar{y}_k = \{\bar{z}_1, \cdots, \bar{z}_k\}. \tag{34}$$

In the following it is shown that the control is measurable on events generated by $\tilde{y}_k$ only.

Theorem 1. *Consider the filtration σ − algebra $\tilde{\sigma}_k$ generated by $\tilde{y}_k$, with the decomposition $y_k = \tilde{y}_k + \bar{y}_k$. For $\tilde{y}_k \in \tilde{\sigma}_k$ and $\tilde{\sigma}_{k-1} \subset \tilde{\sigma}_k$, $\bar{y}_k$ is adapted to $\tilde{\sigma}_{k-1}$ and u_k is adapted to $\tilde{\sigma}_k$.*

Proof. Start with $k = 1$. The initial state is decomposed as $x_1 = \tilde{x}_1 + \bar{x}_1$, where $\bar{x}_1$ is a given non-random parameter. The measurement decomposes as $z_1 = \tilde{z}_1 + \bar{z}_1$, where $\bar{z}_1 = H\bar{x}_1$ is a given non-random parameter and $\tilde{z}_1 = \tilde{y}_1 \in \tilde{\sigma}_1$. Then, u_1, which is determined by z_1, is adapted to $\tilde{\sigma}_1$. At $k = 2$, both $\bar{x}_2 = \Phi\bar{x}_1 + u_1$ and $\bar{z}_2 = H\bar{x}_2$ are adapted to $\tilde{\sigma}_1$, and thus $\bar{y}_2$ is adapted to $\tilde{\sigma}_1$. For the measurement at $k = 2$, $\tilde{z}_2 \in \tilde{\sigma}_2$, $\tilde{y}_2 \in \tilde{\sigma}_2$, and $\tilde{\sigma}_1 \subset \tilde{\sigma}_2$. Hence, since u_2 is determined by $y_2 = \tilde{y}_2 + \bar{y}_2$, it is adapted to $\tilde{\sigma}_2$. Recursively to any k, $\bar{y}_k$ is adapted to $\tilde{\sigma}_{k-1}$. With $\tilde{y}_k \in \tilde{\sigma}_k$, and $\tilde{\sigma}_{k-1} \subset \tilde{\sigma}_k$, u_k that is determined by $y_k = \tilde{y}_k + \bar{y}_k$ is adapted to $\tilde{\sigma}_k$.

5.2 Performance Index

Assuming the expectation is finite, the optimal value of the performance index is stated as

$$J^* = \max_{\mathcal{U}_k^{p-1} \in \mathcal{F}} E\left[\psi\left(\mathcal{X}_{k+1}^p, \mathcal{U}_k^{p-1}\right)\right] = E\left[\max_{\mathcal{U}_k^{p-1} \in \mathcal{F}} E\left[\psi\left(\mathcal{X}_{k+1}^p, \mathcal{U}_k^{p-1}\right) \middle| \tilde{y}_k\right]\right] \tag{35}$$

where $\mathcal{F}$ is the class of feedback open-loop control functions of $\tilde{y}_k$ adapted to $\tilde{\sigma}_k$. The state history $\mathcal{X}_{k+1}^p$ and control history $\mathcal{U}_k^{p-1}$ are defined as

$$\mathcal{X}_{k+1}^p := \{x_{k+1}, \ldots, x_p k\}, \tag{36}$$

$$\mathcal{U}_k^{p-1} := \{u_k, \ldots, u_{p-1}\},\ \mathcal{U}_k^{p-1} \in \mathcal{F}, \tag{37}$$

where k is the current time, and p = k + m so that m is the size of a moving horizon. In (35) the expectation and maximization operations are interchanged due to the Fundamental Lemma [6].

Since the expectation is taken over all underlying random variables, for a viable performance index this expectation must remain finite in the presence of the heavy tailed Cauchy densities. Hence, the cost function $\psi(\cdot)$ in (35) is chosen to have the form

$$\psi\left(\mathcal{X}_{k+1}^{p}, \mathcal{U}_{k}^{p-1}\right) = \prod_{i=k}^{p-1}\left[\frac{\eta_{i+1}^{2}}{x_{i+1}^{2}+\eta_{i+1}^{2}} \cdot \frac{\zeta_{i}^{2}}{u_{i}^{2}+\zeta_{i}^{2}}\right]. \tag{38}$$

When the expectation is taken over Gaussian underlying random variables, the cost function is assumed to be of an exponential form

$$\psi\left(\mathcal{X}_{k+1}^{p}, \mathcal{U}_{k}^{p-1}\right) = \prod_{i=k}^{p-1}\left[e^{-\frac{1}{2}q_{i+1}x_{i+1}^{2}} \cdot e^{-\frac{1}{2}r_{i}u_{i}^{2}}\right]. \tag{39}$$

The resulting controller is referred to as Linear Exponential Gaussian (LEG) MPC [7].

The expectation $E\left[\psi\left(\mathcal{X}_{k+1}^{p}, \mathcal{U}_{k}^{p-1}\right)\middle|\, \tilde{y}_{k}\right]$ in (35) is taken with respect to the conditional joint pdf $f_{\tilde{X}_{p}\cdots\tilde{X}_{k+1}|\tilde{Y}_{k}}\left(\tilde{x}_{p}, \ldots, \tilde{x}_{k+1}|\tilde{y}_{k}\right)$, which is the extension of the conditional density of (24). It includes the projected states influenced by the process noise as [8]

$$\begin{aligned} f_{\tilde{X}_{p}\cdots\tilde{X}_{k+1}|\tilde{Y}_{k}}&\left(\tilde{x}_{p}, \ldots, \tilde{x}_{k+1}|\tilde{\mathcal{Y}}_{k}\right) \\ &= f_{\tilde{X}_{p}|\tilde{X}_{p-1}}\left(\tilde{x}_{p}|\tilde{x}_{p-1}\right) f_{\tilde{X}_{p-1}|\tilde{X}_{p-2}}\left(\tilde{x}_{p-1}|\tilde{x}_{p-2}\right)\cdots \\ &\qquad f_{\tilde{X}_{k+2}|\tilde{X}_{k+1}}\left(\tilde{x}_{k+2}|\tilde{x}_{k+1}\right) f_{\tilde{X}_{k+1}|\tilde{Y}_{k}}\left(\tilde{x}_{k+1}|\tilde{y}_{k}\right). \end{aligned} \tag{40}$$

In the above, the transition pdf is given by

$$f_{\tilde{X}_{j+1}|\tilde{X}_{j}}\left(\tilde{x}_{j+1}|\tilde{x}_{j}\right) = \frac{\beta/\pi}{(\tilde{x}_{j+1}-\Phi\tilde{x}_{j})^{2}+\beta^{2}}. \tag{41}$$

5.3 Cauchy Model Predictive Controller

The m-step optimal multi-step predictive controller is formed by determining the optimal open-loop control sequence over the interval k to $p-1$ given the measurement history $\tilde{y}_{k}$ and using only the optimal control value at time stage k. This process is repeated at each time stage. To do this, the optimal performance index $J_{\tilde{y}_{k}}^{*}$ is evaluated as

$$\begin{aligned} J_{\tilde{y}_{k}}^{*} &= \max_{\mathcal{U}_{k}^{p-1}\in\mathcal{F}} E\left[\psi\left(\mathcal{X}_{k+1}^{p}, \mathcal{U}_{k}^{p-1}\right)\middle|\, \tilde{y}_{k}\right] = \max_{\mathcal{U}_{k}^{p-1}\in\mathcal{F}} E\left[\prod_{i=k}^{p-1} M_{x}(x_{i+1})M_{u}(u_{i})\middle|\, \tilde{y}_{k}\right] \\ &= \max_{\mathcal{U}_{k}^{p-1}\in\mathcal{F}} \int_{-\infty}^{\infty}\cdots\int_{-\infty}^{\infty}\prod_{i=k}^{p-1} M_{x}(x_{i+1})M_{u}(u_{i}) \\ &\qquad \times f_{\tilde{X}_{p}\cdots\tilde{X}_{k+1}|\tilde{Y}_{k}}\left(\tilde{x}_{p}, \ldots, \tilde{x}_{k+1}|\tilde{y}_{k}\right) d\tilde{x}_{p}, \ldots, d\tilde{x}_{k+1}, \end{aligned} \tag{42}$$

where $f_{\tilde{X}_p \cdots \tilde{X}_{k+1}|\tilde{Y}_k}\left(\tilde{x}_p, \ldots, \tilde{x}_{k+1}|\tilde{y}_k\right)$ is given by (40). Currently, results are given for the case where only the terminal state is weighted in the performance index, i.e.

$$J^*_{\tilde{y}_k} = \max_{\mathcal{U}_k^{p-1} \in \mathcal{F}} \prod_{i=k}^{p-1} \frac{\zeta_i^2}{u_i^2 + \zeta_i^2} \int_{-\infty}^{\infty} \frac{\eta_p^2}{x_p^2 + \eta_p^2} \, f_{\tilde{X}_p|\tilde{Y}_k}\left(\tilde{x}_p|\tilde{y}_k\right) d\tilde{x}_p$$

$$= \max_{\mathcal{U}_k^{p-1} \in \mathcal{F}} \prod_{i=k}^{p-1} \frac{\zeta_i^2}{u_i^2 + \zeta_i^2} \cdot \eta_p \pi \sum_{i=1}^{k+2} \frac{1}{\omega_i} \frac{a_i(\bar{x}_p + \sigma_i)\eta_p + (b_i - a_i\bar{x}_p)(\eta_p + \omega_i)}{(\bar{x}_p + \sigma_i)^2 + (\eta_p + \omega_i)^2}, \quad (43)$$

where

$$\bar{x}_p = \Phi^m \bar{x}_k + \sum_{i=1}^{m} \Phi^{m-1} \Gamma u_{k+i-1}. \quad (44)$$

The maximum value of the performance index in (43) uses explicitly the conditional pdf in the expectation operation. The optimization of the analytic function in (43), which is not concave, must be done numerically.

5.4 LEG Model Predictive Control

To compare the common Gaussian controller approach with the results of the Cauchy controller, the LEG controller is presented [6]. The objective is to maximize the expected value of an exponential cost

$$J^*_G = E\left[\max_{\mathcal{U}_k^{p-1} \in \mathcal{F}} E\left[e^{-\frac{1}{2}\left(q_p x_p^2 + \sum_{i=k}^{p-1} r_i u_i^2\right)} \Big| \tilde{y}_k\right]\right], \quad (45)$$

where the expectation is taken with respect to the Gaussian conditional density with conditional mean and variance

$$\hat{x}_k = \bar{x}_k + P_k H/V(z_k - H\bar{x}_k), \quad P_k = \frac{VM_k}{V + H^2 M_k}. \quad (46)$$

and propagated to time p from time k using

$$\bar{x}_{k+1} = \Phi\hat{x}_k + u_k, \quad M_{k+1} = \Phi^2 P_k + W. \quad (47)$$

The LEG controller from [6] is

$$u_k^* = -\frac{S_{k+1}\Phi}{S_{k+1} + r_k + S_{k+1} r_k M_{k+1}} \hat{x}_k, \quad S_i = \frac{S_{i+1} r_i \Phi^2}{S_{i+1} + r_i + S_{i+1} r_i W}, \; S_p = q_p. \quad (48)$$

As stated earlier, the least square fit of the Gaussian and the Cauchy pdf yield a linear relation between the standard deviation of the Gaussian and the scale parameter of the Cauchy. This linear relation relates also the parameters in the cost function.

5.5 Numerical Examples

To obtain insight into the properties of the Cauchy stochastic controller, the one-step one-measurement example is first analyzed. Then the multi-step numerical simulation results are given, which illustrate the behavior of the Cauchy controllers in the presence of dominate Cauchy measurement noise and then dominant Cauchy process noise.

One-Step One-Measurement Example. The value of the optimal control signal at $k = 1$, i.e., u_1^*, as a function of the first measurement z_1, is determined. Specifically, we examine the value of the optimal control signal at $k = 1$, i.e., $u_1^*(z_1)$, as a function of the first measurement $\tilde{z}_1$, that varies due to the measurement noise v_1 [7], while considering the one step horizon, i.e., $m = 1$. The parameters for the system and Cauchy signals are first chosen as

$$\Phi = 1,\ H = 1,\ \alpha = 0.1,\ \beta = 0.02,\ \gamma = 0.5,\ \bar{x}_1 = 0. \tag{49}$$

Initially, no penalty is introduced on the control signal, i.e., the term $\zeta_1^2/(u_1^2+\zeta_1^2)$ is removed from the objective function in (43), i.e. $\zeta_1 \to \infty$ while the state at $k = 2$ is weighted with $\eta_2 = 0.7$. Substituting these parameters into (43), the performance index becomes

$$J_{\tilde{y}_1}^* = \frac{0.1148(4.1667z_1^2 - 1.0163u_1z_1 + 1)}{(z_1^2 + 0.16)(u_1^2 + 0.6724)} + \frac{0.03416(7.5820z_1^2 + 3.4153u_1z_1 - 1)}{(z_1^2 + 0.16)\left((u_1 + z_1)^2 + 1.22\right)} \tag{50}$$

The optimal controller can be obtained by minimizing (50) with respect to u_1. The necessary optimality condition, $\partial J_{\tilde{y}_1}^*/\partial u_1 = 0$, reduces to finding the roots of the fifth-order polynomial

$$l_5u_1^5 + l_4u_1^4 + l_3u_1^3 + l_2u_1^2 + l_1u_1 + l_0 = 0, \tag{51}$$

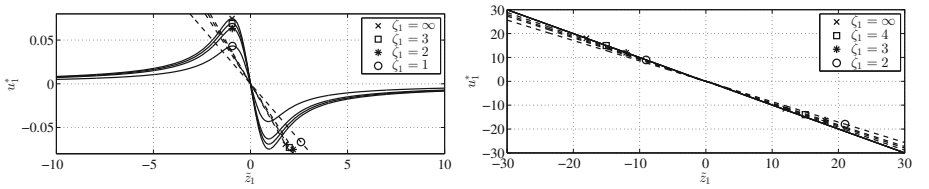
where

$$\begin{aligned} &l_5 = 1, \quad l_4 = 3.5z_1, \quad l_3 = (5.2315z_1^2 + 3.6681), \quad l_2 = (3.6806z_1^2 + 6.6305)z_1, \\ &l_1 = (0.9491z_1^4 + 3.2124z_1^2 + 2.9623), \quad l_0 = (0.07782z_1^2 + 0.3992)z_1. \end{aligned} \tag{52}$$

This polynomial always has at least one real root. If three roots are real, then there are two local maximum values and the larger of the two gives the optimal control.

Solving the polynomial numerically, the optimal control signal is plotted versus the measurement $\tilde{z}_1$ in Fig. 5(a) for $\zeta_1 = \infty$ as well with weighting $\zeta_1 = 1,\ 2,\ 3$ and $\gamma > \alpha$. Also included is the LEG controller where $u_1^* = -\hat{x}_1$, i.e., it is linear in $\tilde{z}_1$ (see [7] for details of the LEG model predictive controller). The Cauchy controller in Fig. 5(a) for large $\tilde{z}_1$ goes toward zero. This is in sharp contrast with the LEG controller, which remains linear in the measurement. This is a significant difference in behavior between the Cauchy and Gaussian optimal controllers that can be deduced analytically from (51). If $u_1^*(\tilde{z}_1)$ is finite, the

dominant term in (51) as $|\tilde{z}_1| \to \infty$ is $l_1 u_1^*(\infty)$, or $\lim_{|\tilde{z}_1|\to\infty} u_1^*(\tilde{z}_1) \to 0$. Therefore, the problem of handling outliers, which occur for the Cauchy pdf, appears to be resolved by the Cauchy controller explicitly, and not in some filter as has been done traditionally. Note that the controller design process explicitly uses the parameters γ and α, and thus their relative size $\gamma > \alpha$, i.e., it should expect more impulsive measurement uncertainty than process uncertainty. If $\gamma < \alpha$, the Cauchy controller behaves approximately like the LEG linear controller in Fig. 5(b), i.e., it expects more impulsive process uncertainty than measurement uncertainty. The effect of reducing ζ_1 from ∞ to 2 has a small effect on the control strategy, as is clearly seen in Figs. 5.



(a) $\alpha = 0.1,\ \beta = 0.02,\ \gamma = 0.5,\ \eta_2 = 0.7$ (b) $\alpha = 0.5,\ \beta = 0.02,\ \gamma = 0.1,\ \eta_2 = 0.7$

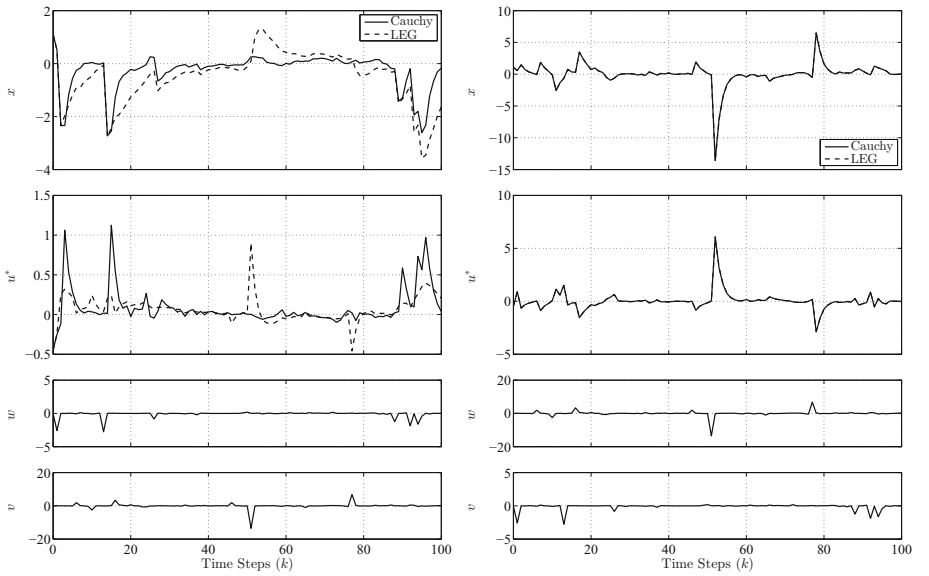
Fig. 5. Scalar Cauchy (solid line) and Gaussian (dashed line) one-step controllers with parameters variations in ζ_1

5.6 Multi-step Numerical Examples

The dynamic characteristics of the Cauchy optimal controller, obtained by maximizing the performance index in (43), are explored through several multi-step numerical examples. The Cauchy optimal control results are compared against the least-squares equivalent LEG controller, obtained from (48), and the Kalman filter from (46). The example that is discussed in this section is a stable system with $\Phi = 0.95$, $H = 1$, and a horizon length of $m = 2$. The state weight parameter is chosen as $\eta_p = \eta_{k+2} = 0.7$, while the control weights are chosen as $\zeta_i = 8$, $i = k, k+1$. The noise parameter values β and γ are interchanged to see how the controller performance changes when it is designed for a large measurement noise impulse in contrast to when it is designed for a large process noise impulse. The values used are $\beta = 0.1, \gamma = 0.02$ or $\beta = 0.02, \gamma = 0.1$, while the initial condition parameters are $\alpha = 0.5$ and $\bar{x}_1 = 0$. Substituting these parameters into (43), the performance index is maximized numerically with respect to the control at each time using a homotopy optimization method.

The simulations results are depicted in Figs. 6. First, for $\gamma = 0.1$ and $\beta = .02$, depicted in Fig. 6(a) when the noises are small, the Cauchy and the LEG controllers exhibit similar performance. However, they behave rather differently when a large measurement pulse occurs. A measurement noise pulse does not represent a state deviation and thus, for proper regulation, the controller should ignore that measurement. The Cauchy predictive controller, designed for $\gamma > \beta$,

is able to make this distinction, whereas the LEG predictive controller reacts linearly to all the pulses and does not differentiate as shown in Fig. 6(a). At time steps $k = 2$ and $k = 13$ process noise pulses occur, and although both controllers react to them and are able to overcome this deviation, the Cauchy controller does so much quicker than the LEG by applying a much larger control effort. The Cauchy applies a larger control because its gain for small measurement values are higher than that of the LEG. Conversely, when a large measurement pulse occurs at $k = 51$, the Cauchy controller ignores it, applying almost zero control, whereas the LEG controller applies a very large control input that causes the state to deviate away from zero, which then required additional control effort to correct. This way the Cauchy controller manages to avoid unnecessary actuation and thus maintains the system performance. When $\gamma < \beta$, the behavior of the Cauchy and LEG controllers are similar, as is shown in Fig. 6(b). This demonstrates the same linear behavior as was seen in Fig. 5(b).



(a) $\alpha = 0.5,\ \beta = 0.02,\ \gamma = 0.1,$ $\Phi = 0.95,\ \eta_2 = 0.7, \zeta_1 = 8$

(b) $\alpha = 0.5,\ \beta = 0.1,\ \gamma = 0.02,$ $\Phi = 0.95,\ \eta_2 = 0.7, \zeta_1 = 8$

Fig. 6. 2-step Cauchy and Gaussian controllers with β and γ parameters interchanged

6 Vector-State Cauchy Estimation

So far we have only considered scalar dynamic systems. When addressing multivariable systems, the pdf method cannot be used, mainly because the partial fraction expansions utilized in the scalar case do not apply here. To overcome this

difficulty, for the multivariable system case we recursively generate the characteristic function of the unnormalized conditional pdf, from which the conditional mean and conditional error variance can be recovered. In this section we summarize our current efforts in vector-state cauchy estimation.

6.1 Formulation

The single-input single output multivariable linear dynamic system is

$$x_{k+1} = \Phi x_k + \Gamma w_k,\ z_k = H x_k + v_k, \tag{53}$$

where the state vector $x_k \in \mathbb{R}^n$, scalar measurement z_k, and known matrices $\Phi \in \mathbb{R}^{n\times n}$, $\Gamma \in \mathbb{R}^{n\times 1}$, and $H \in \mathbb{R}^{1\times n}$. The noise inputs are assumed to be independent with know Cauchy pdf given in (4). The characteristic functions of these scalar noises are assumed to be time independent and given by

$$\phi_W(\bar{\nu}) = e^{-\beta|\bar{\nu}|}, \quad \phi_V(\bar{\nu}) = e^{-\gamma|\bar{\nu}|}, \tag{54}$$

where these characteristic functions have a scalar argument $\bar{\nu}$. The initial conditions at $k = 1$ are also assumed to be independent and Cauchy distributed. Specifically, each i-th element x_{1i} of the initial state vector x_1 has a Cauchy pdf with a zero median and a scaling parameter $\alpha_i > 0,\ i = 1, \ldots, n$. The characteristic function of the joint pdf of the initial conditions, which is a function of a n-dimensional spectral variable $\nu \in \mathbb{R}^n$, is given by

$$\begin{aligned}\phi_{X_1}(\nu) = \prod_{i=1}^{n} e^{-\alpha_i|\nu_i|} &= \exp\left(-\sum_{i=1}^{n}\alpha_i|\nu_i|\right) \\ &\triangleq \exp\left[\left(-\sum_{i=1}^{n} p_i^1 |\langle a_i^1, \nu\rangle|\right) + j\langle b_1^1, \nu\rangle\right].\end{aligned} \tag{55}$$

The last form was introduced for notational convenience to be used in the sequel. We used the definitions

$$p_i^1 = \alpha_i,\ \ a_i^1 = e_i, \quad i = 1, \ldots, n, \quad b_1^1 = \{0\}_n, \tag{56}$$

where e_i is a n-dimensional i-th unity vector and $\{0\}_n$ is n-dimensional vector of zeros.

6.2 Characteristic Function for the Un-normalized Conditional pdf

Our goal is to compute the minimum variance estimate of x_k given the measurement measurement history or $y_k = \{z_1\ z_2 \cdots\ z_k\}$ [4]. We begin by determining the characteristic function for the un-normalized conditional pdf at $k = 1$, where the conditional pdf at $k = 1$ is

$$\begin{aligned} f_{X_1|Z_1}(x_1|z_1) &= \frac{f_{X_1 Z_1}(x_1, z_1)}{f_{Z_1}(z_1)} \\ &= \frac{f_{Z_1|X_1}(z_1|x_1)\, f_{X_1}(x_1)}{f_{Z_1}(z_1)} = \frac{f_V(z_1 - Hx_1)\, f_{X_1}(x_1)}{f_{Z_1}(z_1)}. \end{aligned} \tag{57}$$

The unnormalized conditional pdf (ucpdf) at $k = 1$ is simply the joint pdf of the vector state and scalar measurement as

$$f_{X_1 Z_1}(x_1, z_1) = f_{Z_1|X_1}(z_1|x_1) f_{X_1}(x_1) = f_V(z_1 - Hx_1) f_{X_1}(x_1). \tag{58}$$

The characteristic function of the ucpdf is obtained as

$$\begin{aligned}\bar{\phi}_{X_1|Z_1}(\nu) &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_{X_1}(x_1) f_V(z_1 - Hx_1) e^{j x_1^{\mathrm{T}} \nu} dx_1 \\ &= \frac{1}{(2\pi)^n} \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \phi_{X_1}(\nu - \eta) \hat{\phi}_V(\eta) d\eta. \end{aligned} \tag{59}$$

The first integral is a Fourier transform of a product of two functions. Using the dual convolution property, the second integral is a convolution in the ν domain between the associated characteristic functions $\phi_{X_1}(\nu)$ given in (55) and $\hat{\phi}_V(\eta)$, the characteristic function of $f_{Z_1|X_1}(z_1|x_1) = f_V(z_1 - Hx_1)$ determined as [4]

$$\begin{aligned}\hat{\phi}_V(\nu) &= \int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} f_V(z_1 - Hx_1) e^{j x_1^T \nu} dx_1 \\ &= K(\nu) \phi_V(-\frac{e_n^T \nu}{He_n}) \prod_{i=1}^{n-1} 2\pi \delta(e_i P_n \nu), \end{aligned} \tag{60}$$

where $He_n \neq 0$, $P_n = I - \dfrac{H^T e_n^T}{He_n}$, $K(\nu) = \exp\left(\dfrac{j z_1 e_n^T \nu}{He_n}\right)$, and $\phi_V\left(-\dfrac{e_n^T \nu}{He_n}\right) = \exp\left(-\gamma \left|\dfrac{e_n^T \nu}{He_n}\right|\right)$. Substitution of the $n-1$ delta functions of (60) into (59) reduces the n integrals to one as

$$\begin{aligned}\bar{\phi}_{X_1|Z_1}(\nu) &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \phi_{X_1}(\nu - H^{\mathrm{T}} \sigma) \phi_V(-\sigma) e^{j z_1 \sigma} d\sigma \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \phi_{X_1}(\nu - H^{\mathrm{T}} \sigma) \phi_V(-\sigma) e^{j z_1 \sigma} d\sigma, \end{aligned} \tag{61}$$

where $\phi_V(-\sigma) = e^{-\gamma|\sigma|}$.

The convolution integral in (61) is solved in closed form. Here we assume that each element in H is non-zero, i. e., $He_i \neq 0$, $i = \{1, \ldots, n\}$. (See [4] regarding relaxing this condition.) To compute the integral, we define $\rho_i = \alpha_i |He_i|$, $\mu_i = e_i^T \nu / He_i$, $i = \{1, \ldots, n\}$, $\rho_{n+1} = \gamma$, $\mu_{n+1} = 0$. The convolution integral becomes

$$\bar{\phi}_{X_1|Z_1}(\nu) = \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-\sum_{i=1}^{n+1} \rho_i |\mu_i - \sigma| + j z_1 \sigma} d\sigma. \tag{62}$$

To carry out the integration, assume that there exists an arbitrary ν for which the μ_is are ordered as $\mu_\ell(\nu) \leq \mu_i(\nu)$ for all $\ell \leq i$ with $(\ell, i) \in \{1, \ldots, n+1\}$. The convolution integral (62) can be decomposed into a sum as

$$\bar{\phi}_{X_1|Z_1}(\nu) = \frac{1}{2\pi}\sum_{\ell=0}^{n}\int_{\mu_\ell}^{\mu_{\ell+1}} e^{-\sum_{i=1}^{n+1}\rho_i|\mu_i-\sigma|+jz_1\sigma}d\sigma$$
$$= \frac{1}{2\pi}\sum_{\ell=0}^{n}\int_{\mu_\ell}^{\mu_{\ell+1}} e^{-\sum_{i=1}^{n+1}\rho_i(\mu_i-\sigma)\,\mathrm{sign}(\mu_i-\sigma)+jz_1\sigma}d\sigma$$

where $\mu_0 = -\infty$ and $\mu_{n+1} = \infty$. Note that $\mathrm{sign}(\mu_\ell - \sigma)$ is *constant* over the interval $\sigma \in (\mu_i, \mu_{i+1}), \forall i$. Therefore, we can define the sign function as

$$\mathrm{sign}(\mu_\ell - \sigma) \triangleq s_i^\ell = \begin{cases} \mathrm{sign}(\mu_\ell - \mu_i) & \text{if } i \neq \ell \\ -1 & \text{if } i = \ell \end{cases}$$

For a given i, the discrete, two-indexed function s_i^ℓ is constant for all ℓ except for one switch at $\ell = i$, i.e., $s_i^\ell = -1 \;\forall\; \ell \leq i$ and $s_i^\ell = 1 \;\forall\; \ell > i$. Since μ_i can be ordered for any value of ν, the solution to the integral is independent of the value assumed for ν.

The closed-form solution to the convolution integral for the characteristic function of the unnormalized conditional density function, $\bar{\phi}_{X_1/Z_1}(\nu)$, is

$$\bar{\phi}_{X_1/Z_1}(\nu) = \frac{1}{2\pi}\sum_{\ell=1}^{n+1} g_i^{1|1}(y_{gi}^{1|1}(\nu))e^{y_{ei}^{1|1}(\nu)}, \tag{63}$$

where

$$g_i^{1|1}(y_{gi}^{1|1}(\nu)) = \left[\frac{1}{jz_1 + \rho_\ell + y_{gi}^{1|1}(\nu)} - \frac{1}{jz_1 - \rho_\ell + y_{gi}^{1|1}(\nu)}\right] \tag{64}$$

and

$$y_{gi}^{1|1}(\nu) = \sum_{\substack{i=1\\ i\neq\ell}}^{n+1}\rho_i\,\mathrm{sign}(\mu_i - \mu_\ell), \quad y_{ei}^{1|1}(\nu) = -\sum_{\substack{i=1\\ i\neq\ell}}^{n+1}\rho_i|\mu_i - \mu_\ell| + jz_1\mu_\ell. \tag{65}$$

The first two derivatives of $\bar{\phi}_{X_1/Z_1}(\nu)$ can be shown to be continuous. For continuity of the first derivative of $\bar{\phi}_{X_1|Z_1}(\nu)$ in (61), a piecewise continuous function is convolved with a continuous function. For continuity of the second derivative of $\bar{\phi}_{X_1|Z_1}(\nu)$ in (61), two piecewise continuous functions are convolved.

6.3 Conditional Mean and Variance

The conditional mean and variance are computed by evaluating $\bar{\phi}_{X_1/Z_1}(\nu)$ and its first two derivatives at $\nu = \{0\}_n$ or alternatively as $\nu \to \{0\}_n$. Since $\bar{\phi}_{X_1/Z_1}(\nu)$ and its first two derivatives are continuous, they can be evaluated along a fixed direction $\nu = \epsilon\hat{\nu}$ while letting $\epsilon \to 0$. The pdf of the measurement variable

$$f_{Z_1}(z_1) = \bar{\phi}_{X_1|Z_1}(\epsilon\hat{\nu})\Big|_{\epsilon=0} = \frac{1}{\pi}\frac{\sum_{\ell=1}^{n}\alpha_\ell|h_\ell| + \gamma}{z_1^2 + \left(\sum_{\ell=1}^{n}\alpha_\ell|h_\ell| + \gamma\right)^2}.$$

The minimum conditional-variance estimate is given by

$$\hat{x}_1(z_1) = \frac{1}{j\, f_{Z_1}(z_1)} \left(\frac{\partial \bar{\phi}_{X_1|Z_1}(\epsilon\hat{\nu})}{\epsilon\hat{\nu}} \right)^{\mathrm{T}} \Bigg|_{\epsilon=0} = z_1 \frac{[\alpha_1 \operatorname{sign}(h_1) \cdots \alpha_n \operatorname{sign}(h_n)]^{\mathrm{T}}}{\sum_{\ell=1}^{n} \alpha_\ell |h_i| + \gamma},$$

The conditional variance is ($\tilde{x_1} = x_1 - \hat{x}_1$)

$$E[\tilde{x}_1 \tilde{x}_1^{\mathrm{T}} | z_1] = \left[1 + \frac{z_1^2}{\left(\sum_{i=1}^{n} \alpha_i |h_i| + \gamma\right)^2} \right] \times \begin{bmatrix} \frac{\alpha_1}{|h_1|} \left(\sum_{i=2}^{n} \alpha_i |h_i| + \gamma \right) & \cdots -\alpha_1 \alpha_n \operatorname{sign}(h_1) \operatorname{sign}(h_n) \\ \vdots & \vdots \\ -\alpha_1 \alpha_n \operatorname{sign}(h_1) \operatorname{sign}(h_n) & \cdots \frac{\alpha_n}{|h_n|} \left(\sum_{i=1}^{n-1} \alpha_i |h_i| + \gamma \right) \end{bmatrix}. \quad (66)$$

Note that with one measurement, the $n \times n$ conditional variance is bounded and positive definite. Furthermore, the conditional variance is an explicit function of the measurement.

6.4 Propagation to $k = 2$ and the Second Measurement Update

The time propagated characteristic function to $k = 2$ is found in [4] as

$$\bar{\phi}_{X_2|Z_1}(\nu) = \bar{\phi}_{X_1|Z_1}(\Phi^T \nu) e^{-\beta |\Gamma^T \nu|}. \quad (67)$$

The convolution integral for the second measurement update is

$$\bar{\phi}_{X_2|Y_2}(\nu) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \bar{\phi}_{X_2|Z_1}(\nu - H^T \sigma) e^{-\gamma|\sigma| + j z_2 \sigma} d\sigma$$

where $y_2 = \{z_1, z_2\}$. Since $\bar{\phi}_{X_1|Z_1}(\nu)$ is twice differentiable, then for the linear transformation Φ $\bar{\phi}_{X_1|Z_1}(\Phi^{\mathrm{T}} \nu)$ is also continuous. By assuming $H\Gamma \neq 0$, it can be shown that the first two derivatives of $\bar{\phi}_{X_2|Y_2}(\nu)$ are continuous. If $H\Gamma = 0$, then $e^{-\beta|\Gamma^{\mathrm{T}}(\nu - H^{\mathrm{T}}\sigma)|}$ comes out of the convolution integral and $\bar{\phi}_{X_2|Y_2}(\nu)$ does not have a continuous derivative since the derivative of $e^{-\beta|\Gamma^T \nu|}$ is piecewise continuous and there is no estimate of x.

6.5 General Form of $\bar{\phi}_{X_k|Y_k}(\nu)$

In general, the convolution integral for the k^{th} measurement update is

$$\bar{\phi}_{X_k|Y_k}(\nu) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \bar{\phi}_{X_k|Y_{k-1}}(\nu - H^T \sigma) e^{-\gamma|\sigma| + j z_k \sigma} d\sigma$$

where the measurement history is $y_k = \{z_1, \ldots, z_k\}$. It is assumed that $H\Gamma \neq 0$ at each time stage and by induction $\bar{\phi}_{X_{k-1}|Y_{k-1}}(\nu)$ is twice differentiable. Therefore, $\bar{\phi}_{X_k|Y_k}(\nu)$ is twice continuously differentiable.

The twice differentiable $\bar{\phi}_{X_k|Y_k}(\nu)$ has a closed form [4] as

$$\bar{\phi}_{X_k|Y_k}(\nu) = \sum_{i=1}^{n_t^{k|k}} g_i^{k|k}\left(y_{gi}^{k|k}(\nu)\right) \exp\left(y_{ei}^{k|k}(\nu)\right)$$

where

$$y_{gi}^{k|k}(\nu) = \sum_{\ell=1}^{n_{ei}^{k|k}} q_{i\ell}^{k|k} \operatorname{sign}(\langle a_{i\ell}^{k|k}, \nu\rangle) \in \mathbb{R}^k$$

$$y_{ei}^{k|k}(\nu) = -\sum_{\ell=1}^{n_{ei}^{k|k}} p_{i\ell}^{k|k} |\langle a_{i\ell}^{k|k}, \nu\rangle| + j\langle b_i^{k|k}, \nu\rangle$$

where $q_{i\ell}^{k|k} \in \mathbb{R}^k$, $p_{i\ell}^{k|k}$, $a_{i\ell}^{k|k} \in \mathbb{R}^n$, and $b_i^{k|k} \in \mathbb{R}^n$ are all parameters computed recursively up to time k. The form given in (55) is consistant with this notation.

6.6 The pdf of the Measurement History, the Conditional Mean, and the Conditional Variance

To construct the conditional mean and variance, choose $\nu = \epsilon\hat{\nu}$ where $\epsilon > 0$ and $\hat{\nu}$ is a fixed direction. Assuming the condition that $\langle a_{i\ell}^{k|k}, \hat{\nu}\rangle \neq 0 \; \forall\, (i,\, \ell)$, then

$$\operatorname{sign}(\langle a_{i\ell}^{k|k}, \epsilon\hat{\nu}\rangle) = \operatorname{sign}(\langle a_{i\ell}^{k|k}, \hat{\nu}\rangle) \triangleq s_\ell^i,$$

$$y_{gi}^{k|k}(\epsilon\hat{\nu}) = \sum_{\ell=1}^{n_{ei}^{k|k}} q_{i\ell}^{k|k} s_\ell^i = y_{gi}^{k|k}(\hat{\nu}),\; y_{ei}^{k|k}(\epsilon\hat{\nu}) = \epsilon\langle \bar{y}_{ei}^{k|k}(\hat{\nu}), \hat{\nu}\rangle$$

where s_ℓ^i is a piecewise constant and $\langle \bar{y}_{ei}^{k|k}(\hat{\nu}), \hat{\nu}\rangle$ is a constant. The pdf of the measurement history is

$$f_{Y_k}(y_k) = \bar{\phi}_{X_k|Y_k}(\epsilon\hat{\nu})\Big|_{\epsilon=0} = \sum_{i=1}^{n_t^{k|k}} g_i^{k|k}\left(y_{gi}^{k|k}(\hat{\nu})\right).$$

The conditional mean of the state x_k is

$$\hat{x}_k = E[x_k|y_k] = \frac{1}{j\, f_{Y_k}(y_k)} \left(\frac{\partial \bar{\phi}_{X_k|Y_k}(\epsilon\hat{\nu})}{\partial(\epsilon\hat{\nu})}\right)^T \Bigg|_{\epsilon=0}$$

$$= \frac{1}{j\, f_{Y_k}(y_k)} \sum_{i=1}^{n_t^{k|k}} g_i^{k|k}\left(y_{gi}^{k|k}(\hat{\nu})\right) \bar{y}_{ei}^{k|k}(\hat{\nu}).$$

The second moment

$$E[x_k x_k^T|y_k] = \frac{1}{j^2\, f_{Y_k}(y_k)} \frac{\partial^2 \bar{\phi}_{X_k|Y_k}(\epsilon\hat{\nu})}{\partial(\epsilon\hat{\nu})\partial(\epsilon\hat{\nu})^T}\bigg|_{\epsilon=0}$$
$$= \frac{1}{j^2\, f_{Y_k}(y_k)} \sum_{i=1}^{n_t^{k|k}} g_i^{k|k} \left(y_{gi}^{k|k}(\hat{\nu})\right) \left(\bar{y}_{ei}^{k|k}(\hat{\nu})\right) \left(\bar{y}_{ei}^{k|k}(\hat{\nu})\right)^T.$$

The error $\tilde{x}_k = x_k - \hat{x}_k$ variance can be evaluated as

$$E[\tilde{x}_k \tilde{x}_k^T|y_k] = E[x_k x_k^T|y_k] - \hat{x}_k \hat{x}_k^T.$$

6.7 The Two State Estimator

There are some simplifications that occur in the two-state estimation problem [9]. A recursion in the terms in the argument of the exponential can be made explicit. A basis $\mathcal{B}_k$ for each term in the sum is recursive as

$$\mathcal{B}_k = \begin{bmatrix} \mathcal{B}_{k-1}\Phi^{\mathrm{T}} \\ \Gamma^{\mathrm{T}} \\ \mathrm{HA} \end{bmatrix},\ \mathcal{B}_1 = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \mathrm{HA} \end{bmatrix},\ \mathrm{A} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$$

This basis is related to $a_{i\ell}^{k|k} \in \mathbb{R}^n$ in the general solution in that every $a_{i,l}^{k|k}$ vector is coaligned with a row in $\mathcal{B}_k$.. The number of terms in the sum N_k is reduced by combining terms with the same argument in the exponential. For the two-state estimator with no process noise, $N_k = \frac{(k+2)(k+1)}{2}$. For the two-state estimator with process noise,

$$N_k = \begin{bmatrix} 1\,0\,0\,0 \end{bmatrix} \cdot \begin{bmatrix} 2\,1\,1 & 0 \\ 0\,1\,0 & 2 \\ 1\,0\,1 & -1 \\ 0\,0\,0 & 1 \end{bmatrix}^{k-1} \cdot \begin{bmatrix} 3 \\ 3 \\ 0 \\ 1 \end{bmatrix}.$$

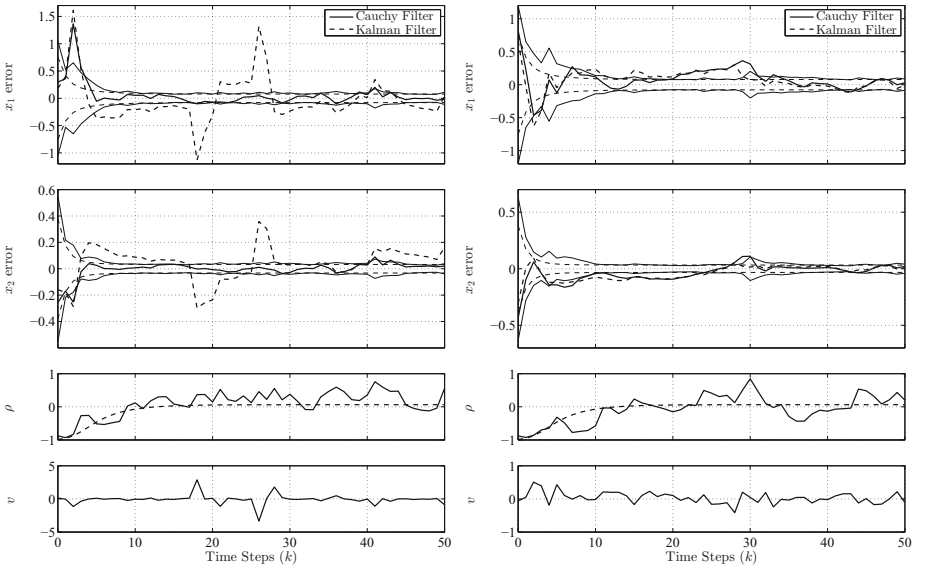
where for $k = 1$ the identity matrix is used.

6.8 Numerical Examples of the Vector-State Cauchy Estimator

First, the no process noise case is generated for an unstable system so that the estimator conditional variance remains finite. Simulation results for 50 steps are given in Figs. 7(a) and Fig. 7(b). In Figs. 7(a) we can clearly see how the impulsive nature of the data increases the estimation inaccuracy of the Kalman filter. However, contrary to the Kalman filter, the Cauchy estimator's standard deviation of the error increases in the presence of impulses in the measurement

noise. The conditional standard deviation of the error for the Cauchy filter, also plotted in Fig. 7(a), is minimal. However, the standard deviation of the error plotted for the Kalman filter is calculated assuming Gaussian noise variances and thus is neither related to the actual estimation errors nor it is minimal.

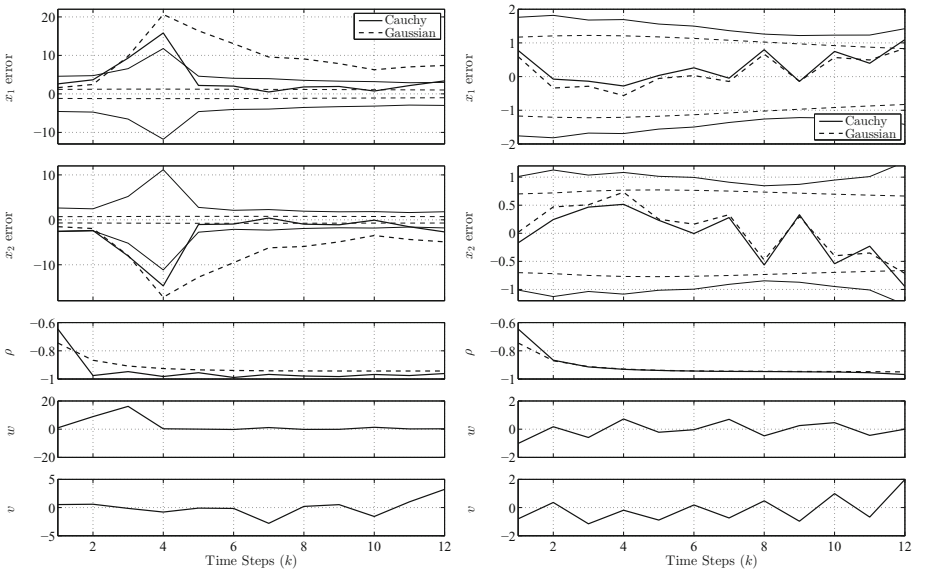
Figure 7(b) presents the estimation errors together with the computed estimation error standard deviations for the Cauchy and Kalman estimators when the process and measurement noise sequences are Gaussian. The error variance for the Kalman filter is now the minimal, since the process and measurement noise sequence are Gaussian. It is remarkable how close the standard deviation of the estimation error generated from the Cauchy cpdf approximates that computed from the Kalman filter, demonstrating the robustness of the former.



(a) Cauchy Noise (b) Gaussian Noise

Fig. 7. Cauchy and Kalman Estimators for an unstable system. Simulation parameters are: $\alpha_1 = 0.5$, $\alpha_2 = 0.6$, $\gamma = 0.1$, $\mathrm{H} = [1 \;\; 2]$, $\Gamma^{\mathrm{T}} = [1 \;\; -1]$, and the eigenvalues of Φ are 1.3 and 1.15.

The two-state system with both measurement and process noise is shown in Figs. 8(a) and Fig. 8(b). Again we see in Figs. 8(a) that the Gaussian filter estimate error deviates from the Cauchy conditional mean error in the Cauchy simulation Fig. 8(a). Also, note that the standard deviation fluctuates dramatically with the Cauchy noises. For the Gaussian simulation Fig. 8(b) both the Cauchy and the Gaussian perform almost identically.



(a) Cauchy Noise (b) Gaussian Noise

Fig. 8. Cauchy and Kalman Estimators for a stable system. Simulation parameters are: $\alpha_1 = 0.9$, $\alpha_2 = 1$, $\gamma = 0.7$, $\beta = 0.3$, H = [1 2], and the eigenvalues of Φ are 0.98 and 0.97.

7 Conclusions: Vector Cauchy Estimation and Control

For scalar stochastic dynamics, developed a Cauchy estimator by generating the conditional pdf. Based on this conditional pdf a m-step optimal MPC for Cauchy noises was developed. This Cauchy optimal MPC can differentiate between process and measurement noise spikes and thereby is a formal method for handling outliers. For the n-vector stochastic dynamics, a Cauchy estimator was developed by propagating and updating the characteristic function of the unnormalized conditional pdf. Numerical results were given for a two state recursive estimator. However, the scheme produces many terms. This issue is being addressed by analyzing the coefficient terms of the exponential to prune away small elements. In addition we are developing schemes for parallel processing.

Acknowledgments. This work was partially supported by Air Force Office of Scientific Research, Award No. FA9550-10-1-0570, and by the United States - Israel Binational Science Foundation, Grant 2008040.

References

1. Fleming, W.H., Rishel, R.W.: Deterministic and Stochastic Optimal Control. Springer, New York (1975)
2. Idan, M., Speyer, J.L.: Cauchy Estimation for Linear Scalar Systems. IEEE Transactions on Automatic Control 55(6), 1329–1342 (2010)
3. Idan, M., Speyer, J.L.: State Estimation for Linear Scalar Dynamic Systems With Additive Cauchy Noises: Characteristic Function Approach. SIAM Journal on Control and Optimization 50(4), 1971–1994 (2012)
4. Idan, M., Speyer, J.L.: Multivariate cauchy estimator with a scalar measurement and process noise. SIAM J. Control and Optimization (submitted)
5. Samorodnitsky, G., Taqqu, M.S.: Stable Non-Gassian Random Processes: Stochastic Models with Infinite Variance. Chapman & Hall, New York (1994)
6. Speyer, J.L., Chung, W.H.: Stochastic Processes, Estimation, and Control. SIAM (2008)
7. Speyer, J.L., Idan, M., Fernández, J.: A stochastic controller for a scalar linear system with additive cauchy noise. Automatica Submitted
8. Speyer, J.L., Idan, M., Fernández, J.: Multi-Step Prediction Optimal Control for a Scalar Linear System with Additive Cauchy Noise. In: IEEE Conference on Decision and Control, Atlanta, Georgia (December 2010)
9. Speyer, J.L., Idan, M., Fernández, J.: The two-state estimator for linear system with additive measurement and process cauchy noise. In: IEEE Conference on Decision and Control, Maui (December 2012)
10. Taleb, N.N.: The Black Swan: The Impact of the Highly Improbable. Random House (2007)
11. Tsakalides, P., Nikias, C.L.: Deviation from Normality in Statistical Signal Processing: Parameter Estimation with Alpha-Stable Distributions. In: A Practical Guide to Heavy Tails: Statistical Techniques and Applications. Birkhauser (1998)

Planar-Feature Based 3D SLAM Using Randomized Sigma Point Kalman Filters

Cihan Ulas* and Hakan Temeltas

Istanbul Technical University, Department of Control Engineering, Istanbul, 34469, Turkey
{culas,hakan.temeltas}@itu.edu.tr

Abstract. In this study, a novel filtering method called Randomized Sigma Point Kalman Filter (RSPKF) is introduced for feature based 3D Simultaneous Localization and Mapping (SLAM). Conventional SLAM methods are mostly based on Extended Kalman Filters (EKF) for 'mild' nonlinear processes and Unscented KF (UKF) or Cubature KF (CKF) for 'aggressive' nonlinear processes. A critical problem of the existing filtering methods is that they lead to biased estimates of the state and measurement statistics. The main purpose of this study is to propose a new local filter, RSPKF, based on stochastic integration rules providing an unbiased estimate of an integral for feature based SLAM. The simulation based on point features in 2D and experimental results based on planar features in 3D show that the RSPKF based SLAM method provides more accurate results than the traditional methods.

Keywords: SLAM, feature extraction, randomized sigma point filters.

1 Introduction

Simultaneous Localization and Mapping plays a central role for fully autonomous system when the Global Navigation Satellite System is not available or denied. SLAM is an active research area of the last decade and its solution is considered as the "holy grail" by the robotics researchers [1]. Feature based SLAM (Fb-SLAM) methods requires sophisticated feature extraction methods. These features are principally considered as rotation and translation independent and can be distinguished when they are exists in the two consecutive observations. The aim of the feature based SLAM methods is to estimate the robot pose and landmark locations combined in a state vector.

A traditional representation in SLAM is to use state space model with additive Gaussian noise, which leads to the local filters such as EKF, UKF, and CKF. EKF is the well-known filtering method using the first order approximation of the nonlinear functions [2]. Therefore, it is appropriate for 'mild' nonlinear processes and measurement models. In order to overcome the linearization problem of 'aggressive' nonlinearities, Julier and Uhlman [3] proposed Unscented Kalman Filter (UKF) known as

* Corresponding author.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_7

derivative-free approach. The UKF, instead of linearization of the nonlinear functions, estimate the mean values and covariance matrices with sigma points, which are obtained by a deterministic sampling approach. Cubature Kalman Filters (CKF) is proposed as a more accurate filtering method and a more mathematically principled method than UKF for nonlinear state estimation by Arasaratnam and Haykin [4]. CKF is also more stable filter than the UKF and has a square root solution providing numerical advantages and maintains the positive definiteness the of covariance matrix.

The UKF and CKF methods can be jointly considered as sigma point or derivative-free Kalman filters. The difference between these local filter is originated from the approximation used in computation of the integrals. The approximations based on the Taylor expansion, unscented transform, and cubature transform has a significant weakness which is the systematic error emerged by the approximate solution to the integrals [5]. To solve this problem, a randomized unscented Kalman filter (RUKF) has been proposed for solving the integrals without systematic errors very recently [5]. RUKF is based on the stochastic integration rule for infinite regions proposed by the Genz and Monohan [6].

In this paper, Randomized Sigma Point Kalman Filters (RSPKF) are used in simultaneous localization and mapping problem. To test the method in a more challenging SLAM problem, we introduce a novel landmark extraction method based on plane detection. Unlike the conventional methods, the 4D infinite plane parameters are encoded into the state vector and they are estimated with the latest 6D robot pose. The proposed observation model consists of dense trigonometric functions and cannot be considered as a mild nonlinear function; therefore, the RSPKF is obviously suitable for this type of problem. The appropriateness of the proposed SLAM method is validated through both simulations and experimental datasets in 2D and 3D, respectively. In 2D, point features are used as landmarks, and in 3D planes are used as landmarks in the SLAM state vector representation.

In Section 2, the Sigma Point Kalman Filter is introduced. Then the Randomized Sigma Point Kalman Filter (RSPKF) based on stochastic integration rule and RSPKF based SLAM method is presented in Section 3. Finally, the simulation and experimental results are given in Section 4 and a conclusion is drawn in Section 5.

2 Sigma Point Kalman Filter

The Sigma Point Kalman Filters (SPKFs) based on the unscented transform (UT) or cubature transform (CT) is introduced in this section.

2.1 Unscented Transformation

The aim of the unscented transformation is to calculate first two moments of a known nonlinear function $\mathbf{y}=\mathbf{g}(\mathbf{x})$ where $\mathbf{x}$ and $\mathbf{y}$ are the random vector variables. The mean vector $\bar{\mathbf{y}}$, the covariance matrix $\mathbf{P}_y$, and the cross-covariance matrix $\mathbf{P}_{xy}$ are described by

$$\begin{aligned}
\bar{\mathbf{y}} &= E[\mathbf{y}] = E[\mathbf{g}(\mathbf{x})] \\
\mathbf{P}_y &= \mathrm{cov}[y] = E[(\mathbf{y} - \bar{\mathbf{y}})(\mathbf{y} - \bar{\mathbf{y}})^T] \\
\mathbf{P}_{xy} &= E[(\mathbf{x} - \bar{\mathbf{x}})(\mathbf{y} - \bar{\mathbf{y}})^T].
\end{aligned} \tag{1}$$

The solution to the problem with UT is based on the approximation of the random variable $\mathbf{x}$ by using a deterministically chosen set of sigma points, χ_i, and their corresponding weights w_i.

$$\begin{aligned}
\chi_0 &= \bar{\mathbf{x}},\ w_0 = \frac{\kappa}{n+\kappa}, \\
\chi_i &= \bar{\mathbf{x}} + \left(\sqrt{(n+\kappa)\mathbf{P}}\right)_i,\ w_i = \frac{1}{2(n+\kappa)}, \\
\chi_{n+i} &= \bar{\mathbf{x}} - \left(\sqrt{(n+\kappa)\mathbf{P}}\right)_i,\ w_{i+n} = w_i,
\end{aligned} \tag{2}$$

where i=1,2, … n, and n is the dimension of the state vector. The term $(\bullet)_i$ represents the i^{th} column of the matrix. The covariance matrix satisfy the definition of $\mathbf{P}=\mathbf{S}\mathbf{S}^T$ where $\mathbf{S}$ is the square root of $\mathbf{P}$. Then sigma points are propagated based on the nonlinear function $\mathbf{g}(\mathbf{x})$ as

$$y_i = \mathbf{g}(\chi_i),\ \forall i. \tag{3}$$

Then the mean and covariance values are approximated as follows

$$\begin{aligned}
\bar{\mathbf{y}}^{ut} &= \sum_{i=0}^{2n} w_i y_i \\
\mathbf{P}_y^{ut} &= \sum_{i=0}^{2n} w_i (y_i - \bar{\mathbf{y}}^{ut})(y_i - \bar{\mathbf{y}}^{ut})^T \\
\mathbf{P}_{xy}^{ut} &= \sum_{i=0}^{2n} w_i (\chi_i - \bar{\mathbf{x}})(y_i - \bar{\mathbf{y}}^{ut})^T
\end{aligned} \tag{4}$$

The variable κ is the scaling parameter and suggested setting is the $\kappa = 3 - n$ [3]. However, the positive semi-definiteness is lost for multi-dimensional variable $\mathbf{x}$, which is the indispensable occasion of SLAM methods, because of negative κ ($n > 3$, $\kappa < 0$). For that reason, a possible practical solution is to choose $\kappa = 0$ for the multi-dimensional case although there is no mathematical justification. Moreover, the adaptive setting of the scaling parameters may improve the estimation accuracy of the UT [7].

2.2 Cubature Transformation

Cubature transformation, a more accurate and mathematically principled transformation than the UT, is proposed by Arasaratnam and Haykin [4]. The cubature transformation is based on the cubature theory and it is summarized as follows.

The key point of the Cubature theory [8] is to find multi-dimensional integrals using cubature rules since its integrands are in the form of,

$$non\text{-}linear\ function \times Gaussian.$$

Thus, the Bayesian filter solution is approximated by the help of cubature theory.

Cubature Rules

The cubature rule is used to approximate an *n*-dimensional Gaussian weighted integral as

$$\int_{R^n} f(x)N(\mathbf{x};\bar{\mathbf{x}},\mathbf{P})dx \approx \frac{1}{2n}\sum_{i=1}^{2n} f(\bar{\mathbf{x}}+\sqrt{\mathbf{P}}\zeta_i) \tag{5}$$

where N is the normal distribution of $\mathbf{x}$ with mean $\bar{\mathbf{x}}$ and covariance matrix $\mathbf{P}$. The relation for covariance matrix $\mathbf{P} = \sqrt{\mathbf{P}}\sqrt{\mathbf{P}}^T$ is satisfied. The $2n$ set of cubature array set is defined by ζ, and ζ_i the i^{th} element of the set ζ,

$$\zeta = \sqrt{n}\left\{\begin{pmatrix}1\\.\\.\\.\\0\end{pmatrix},...,\begin{pmatrix}0\\.\\.\\.\\1\end{pmatrix},\begin{pmatrix}-1\\.\\.\\.\\0\end{pmatrix},...,\begin{pmatrix}0\\.\\.\\.\\-1\end{pmatrix}\right\}. \tag{6}$$

2.3 Sigma Point Kalman Filters: UKF and CKF

The UKF and CKF are jointly called as Sigma Point Kalman filters (SPKF). The SPKF methods are based on either the unscented or the cubature transformations. Consider the following discrete time process and observation models.

$$\mathbf{x}_k = f(\mathbf{x}_{k-1},\mathbf{u}_{k-1}) + w_{k-1} \tag{7}$$

where w_{k-1} denotes the zero mean Gaussian distribution noise vector with covariance matrix Q, and $\mathbf{u}_{k-1}$ the control signal or odometry data. The two fundamental steps of the Kalman filters are explained as follows.

Time Update

In the time update step, SPKFs computes the predicted mean $\bar{\mathbf{x}}^-$ and covariance matrix $\mathbf{P}^-$ depending on the transformation.

$$\bar{\mathbf{x}}_k^- = E[f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + w_{k-1} \mid D_{k-1}] \tag{8}$$

where D_{k-1} denotes the history of the input and measurement pairs up to *k-1*. Since w_{k-1} is assumed to be zero mean and independent of the measurement sequence, one can write

$$\begin{aligned} \bar{\mathbf{x}}_k^- &= E[f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \mid D_{k-1}] \\ &= \int_{R^n} f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) p(\mathbf{x}_{k-1} \mid D_{k-1}) d\mathbf{x}_{k-1} \\ &= \int_{R^n} f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) N(\mathbf{x}_{k-1}; \bar{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1}) d\mathbf{x}_{k-1}. \end{aligned} \tag{9}$$

The corresponding error covariance matrix can be written as

$$\begin{aligned} \mathbf{P}_k^- &= E[(\mathbf{x}_k - \bar{\mathbf{x}}_k^-)(\mathbf{x}_k - \bar{\mathbf{x}}_k^-)^T \mid \mathrm{z}_{k-1}] \\ &= \int_{R^n} f(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) f^T(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \\ &\quad x\, N(\mathbf{x}_{k-1}; \bar{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1}) d\mathbf{x}_{k-1} - \bar{\mathbf{x}}_{k-1} \bar{\mathbf{x}}_{k-1}^T + Q_{k-1}. \end{aligned} \tag{10}$$

Measurement Update. The predicted measurement vector, the corresponding covariance and cross covariance matrices are given by

$$\begin{aligned} \bar{\mathbf{z}}_k^- &= \int_{R^n} h(\mathbf{x}_k) N(\mathbf{x}_k, \bar{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1})\, d\mathbf{x}_k \\ P_{zz,k} &= \int_{R^n} h(\mathbf{x}_k) h^T(\mathbf{x}_k) N(\mathbf{x}_k; \bar{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1}) d\mathbf{x}_k - \bar{\mathbf{z}}_k \bar{\mathbf{z}}_k^T + R_k \\ P_{xz,k} &= \int_{R^n} \mathbf{x}_k h^T(\mathbf{x}_k) N(\mathbf{x}_k; \bar{\mathbf{x}}_{k-1}, \mathbf{P}_{k-1}) d\mathbf{x}_k - \bar{\mathbf{x}}_{k-1} \bar{\mathbf{z}}_k^T \end{aligned} \tag{11}$$

After new $\mathbf{z}_k$ measurements are obtained, the sigma point Kalman filter updates the state vector and covariance matrix as

$$\begin{aligned} \bar{\mathbf{x}}_k^+ &= \bar{\mathbf{x}}_k^- + K_k(\mathbf{z}_k - \bar{\mathbf{z}}_k) \\ P_k &= P_k^- - K_k P_{zz,k} K_k^T \end{aligned} \tag{12}$$

where the K_k is the Kalman gain given by

$$K_k = P_{xz,k} P_{zz,k}^{-1} \tag{13}$$

The main difference between the UKF and CKF is the approximations used to solve the given integrals. While the UKF filters uses unscented transform, CKF uses the cubature transform for solving the integrals. In the next subsection, we state the problem of the systematic error caused by approximations.

2.4 Problem Statement

The sigma point filters provides an approximate solutions to the nonlinear functions. However, these approximations are biased and generate systematic errors. To keep the equations more certain, we elucidate this situation on the UT. The error ε^{ut} is expressed by means of the Taylor expansion of the actual mean and the approximate mean [3].

$$\begin{aligned}\varepsilon^{ut} &= E\left[\frac{D^4_{\Delta x}\mathbf{g}(\mathbf{x})}{4!}+\frac{D^6_{\Delta x}\mathbf{g}(\mathbf{x})}{6!}+\ldots\right] \\ &-\frac{1}{2(n+\kappa)}\sum_{p=1}^{2n}\left(\frac{D^4_{\varepsilon_p}\mathbf{g}(\chi_p)}{4!}+\frac{D^6_{\varepsilon_p}\mathbf{g}(\chi_p)}{6!}+\ldots\right)\end{aligned} \tag{14}$$

where

$$\frac{D^k_{\varepsilon_p}\mathbf{g}(\mathbf{x})}{k!}=\frac{1}{k!}\left(\sum_{i=1}^{n}(\chi_p(i)-\bar{\mathbf{x}}_i)\frac{\partial}{\partial\chi_p(i)}\right)^k\mathbf{g}(\chi_p)\,|_{\chi_p=\bar{x}} \tag{15}$$

is the k^{th} term of the Taylor series expansion of the p^{th} sigma point $\mathbf{g}(\chi_p)$ and $\chi_p(i)$ is the ith element of χ_p. The error ε^{ut} is different from zero if the function **g** is not a polynomial of degree *2n*. This systematic error is also appears in the computations of the covariance matrices in a similar fashion.

In the next section, the randomized sigma point Kalman filter is presented to eliminate the mentioned systematic error.

3 Randomized SPKF Based Slam

Randomized Sigma Point Kalman Filter (RSPKF) proposed by Dunik et al. [5] uses the stochastic integration rule (SIR) introduced by Genz and Monohan [6]. SIR is explained as follows.

3.1 Stochastic Integration Rule (SIR)

SIR is appropriate for solving the integral of the form

$$\mu=\int_{R^n}\mathbf{g}(\mathbf{x})\left(\frac{1}{2\pi}\right)^{n/2}\mathrm{e}^{-\frac{1}{2}\mathbf{x}^T\mathbf{x}}. \tag{16}$$

This relation can be considered as a computation of the expected value of the function **g** where x is a random variable with $p(\mathbf{x})=N(\mathbf{x};\bar{\mathbf{x}},\mathbf{P})$. The algorithm to solve the integral (16) based on SIR is given by the Algorithm 1 in Table 1.

Table 1. Algorithm 1. Stochastic Integration Rule

Algorithm 1. $\mu = \mathbf{SI}(\bar{\mathbf{x}}, \mathbf{P}, \mathbf{g}(\mathbf{x}))$

1: Define N_{max}
2: Set $\mu = \mathbf{0}$ and compute $\chi_0 = \mathbf{g}(\bar{\mathbf{x}})$
3: **for** i=1 to N_{max} do
4: Generate a uniformly random orthogonal matrix $\mathbf{Q} \in R^{nxn}$ and generate a random number ρ form Chi-distribution with $n+2$ degrees of freedom.
5: Compute a set of points $\boldsymbol{\chi}_i$ and corresponding weights w_i according to

$$\boldsymbol{\chi}_i = -\rho\sqrt{P}\mathbf{Q}\mathbf{e}_i$$

$$\boldsymbol{\chi}_{n+i} = \rho\sqrt{P}\mathbf{Q}\mathbf{e}_i$$

$$w_0 = 1 - \frac{n}{\rho^2},\ w_i = w_{n+i} = \frac{1}{2\rho^2}$$

where i=1,2…n and $\mathbf{e}_i$ is the i^{th} column of the identity matrix.

6: Compute the value **S** of the integral at current iteration

$$\mathbf{S} = -\boldsymbol{\chi}_0\omega_0 + \sum_{i=0}^{n_x}\left(\mathbf{g}(\boldsymbol{\chi}_i) + \mathbf{g}(\boldsymbol{\chi}_{i+n_x})\right)\omega_i$$

and use it to update the approximate mean μ

$$\mu = \mu + (\mathbf{S} - \mu)/i$$

7: **end for**
8: return μ

The matrix **Q** can be generated using a product of appropriately chosen random reflections [6].

3.2 Randomized Sigma Point Kalman Filter

The time update and the measurement update steps of the filter are given as follows.

Time Update
The relations of the time update step is previously given by the equations (8) and (10). Here the integrals are solved by the SIR algorithm.

$$\begin{aligned}\bar{\mathbf{x}}_k^- &= \mathbf{SI}\left(\bar{\mathbf{x}}_k, \mathbf{P}_k, f_k(\mathbf{x}_k)\right)\\ \mathbf{P}_k^- &= \mathbf{SI}\left(\bar{\mathbf{x}}_k, \mathbf{P}_k, (f_k(\mathbf{x}_k) - \bar{\mathbf{x}}_k)(f_k(\mathbf{x}_k) - \bar{\mathbf{x}}_k)^T\right) + Q_k\end{aligned} \tag{17}$$

Measurement Update
The relations of the measurement update step is previously given by the equations (11) and the integral obtained by the SIR as

$$\begin{aligned}
\bar{\mathbf{z}}_k^- &= \mathbf{SI}\left(\bar{\mathbf{x}}_k^-, \mathbf{P}_k^-, h_k(\mathbf{x}_k)\right) \\
\mathbf{P}_{zz,k} &= \mathbf{SI}\left(\bar{\mathbf{x}}_k^-, \mathbf{P}_k^-, (h_k(\mathbf{x}_k) - \bar{\mathbf{z}}_k^-)(h_k(\mathbf{x}_k) - \bar{\mathbf{z}}_k^-)^T\right) + R_k \\
\mathbf{P}_{xz,k} &= \mathbf{SI}\left(\bar{\mathbf{x}}_k^-, \mathbf{P}_k^-, (\mathbf{x}_k - \bar{\mathbf{x}}_k^-)(h_k(\mathbf{x}_k) - \bar{\mathbf{z}}_k^-)^T\right) + R_k
\end{aligned} \tag{18}$$

Finally, the estimated state vector and covariance matrix is computed as in (12).

3.3 SLAM Based on RSPKF

The conventional Fb-SLAM representation consists of three models, which are vehicle model *f*, observation model *h*, and the augmentation model *g*. These representations are expressed as follows.

Vehicle Model
The vehicle model is given by *f*

$$\mathbf{x}_{v_k} = f(\mathbf{x}_{v_{k-1}}, \mathbf{u}_{k-1}) + w_{k-1} \tag{19}$$

where w_{k-1} denotes the zero mean Gaussian distribution noise vector with the covariance matrix Q, and the control signal **u.**

Landmark Model
The landmarks are assumed as stationary $\mathbf{x}_{m_k} = \mathbf{x}_{m_{k+1}}$ and represented in world (W) frame. The SLAM map is augmented with the following state vector representation,

$$\mathbf{x}^a{}_{k+1} = \begin{bmatrix} \mathbf{x}_{v_k} & \mathbf{x}_{m_{k+1}} \end{bmatrix} \in \mathrm{R}^{6+4\mathrm{N}}. \tag{20}$$

Observation Model
Measurement or observation model parameters, z_k are provided by the feature extraction method and stated as

$$z_k = h(\mathbf{x}, \mathbf{u}_{k-1}) + v_{k-1} \tag{21}$$

where *h* is the measurement model and v_{k-1} is the zero mean observation noise with *R* error covariance matrix.

Motion Update

Motion update step is based on the vehicle model (19). The state and covariance matrix is augmented as

$$\begin{aligned} \overline{\mathbf{x}}_k &= \begin{bmatrix} \overline{\mathbf{x}}_{v_k} & \mathbf{u}_k \end{bmatrix}^T \\ \mathbf{P}_k &= \begin{bmatrix} \mathbf{P}_k & 0 \\ 0 & Q_k \end{bmatrix} \end{aligned} \tag{22}$$

where $\overline{\mathbf{x}}_k$ is the state vector in the k^{th} time step and $\mathbf{u}_k$ is the applied control signal at this time. $\mathbf{P}_k$ denotes the state covariance matrix and augmented as in (22). The square root, **S**, of the covariance matrix, $\mathbf{P}_k$, is obtained by the Cholesky decomposition **S**=*chol*($\mathbf{P}_k$). Then the time prediction step of the RSPKF algorithm is applied to the augmented vectors (22). The error covariance matrix of the motion is shown by Q_k .

$$\begin{aligned} \overline{\mathbf{x}}_k^- &= \mathbf{SI}\left(\overline{\mathbf{x}}_k, \mathbf{P}_k, f_k(\mathbf{x}_k)\right) \\ \mathbf{P}_k^- &= \mathbf{SI}\left(\overline{\mathbf{x}}_k, \mathbf{P}_k, (f_k(\mathbf{x}_k) - \overline{\mathbf{x}}_k)(f_k(\mathbf{x}_k) - \overline{\mathbf{x}}_k)^T\right) + Q_k \end{aligned} \tag{23}$$

Measurement Update

The measurement update step is based on the observation model (21) and the state and covariance estimations are obtained using the SIR as in (18)

$$\begin{aligned} \overline{\mathbf{x}}_k^+ &= \overline{\mathbf{x}}_k^- + K_k(\mathbf{z}_k - \overline{\mathbf{z}}_k) \\ P_k &= P_k^- - K_k P_{zz,k} K_k^T \end{aligned} \tag{24}$$

where the K_k is the Kalman gain given by

$$K_k = P_{xz,k} P_{zz,k}^{-1} \tag{25}$$

State Augmentation

The state augmentation is based on the augmentation given by (20) and operated in every new landmark observations. The state augmentation is applied in two steps. Firstly, the state vector and covariance matrix is augmented with the new observations as follows.

$$\begin{aligned} \overline{\mathbf{x}}_k^+ &= \begin{bmatrix} \overline{\mathbf{x}}_{v_k} & z_k \end{bmatrix}^T \\ \mathbf{P}_k &= \begin{bmatrix} \mathbf{P}_k & 0 \\ 0 & R_k \end{bmatrix} \end{aligned} \tag{26}$$

where R_k is the error covariance matrix of the measurement.

The augmented state model $g^a = [\mathbf{x}_v \quad x_m]$ is constructed, and then the augmented state vector and covariance matrix are computed by following the same procedure in motion update step with (22) and (23) which are restated here to save space.

4 Performance Evaluatıons

RSPKF based SLAM performance is compared to the SPKF based SLAM in both simulations and experimental in 2D and 3D respectively.

4.1 Simulation Results in 2D

In this section, an artificial environment containing landmark position in 2D is generated and the robot way points are defined. The aim of the SLAM algorithm is to estimate the landmark positions and last robot pose information using the range and bearing observation model.

Vehicle Model
The explicit vehicle model (19) is given by

$$\begin{aligned} x_{v_k} &= x_{v_{k-1}} + Vdtcos(\gamma_{k-1} + \phi_{v_{k-1}}) + w_{x_{k-1}} \\ y_{v_k} &= y_{v_{k-1}} + Vdtsin(\gamma_{k-1} + \phi_{v_{k-1}}) + w_{y_{k-1}} \\ \phi_{v_k} &= \phi_{v_{k-1}} + Vdtsin(\gamma_{k-1}) / L + w_{\phi_{k-1}} \end{aligned} \tag{27}$$

where V and γ are the control input representing the constant velocity and steering angle with zero mean Gaussian noise w, respectively, and ϕ_{v_k} denotes the vehicle heading angle at time k.

Observation Model
The range and bearing observation model (21) is

$$\begin{aligned} r &= \sqrt{(x_m - x_{v_k})^2 + (y_m - y_{v_k})^2} + v_{r_k} \\ b &= tan^{-1}(\frac{y_m - y_{v_k}}{x_m - x_{v_k}}) - \phi_{v_k} + v_{b_k} \end{aligned} \tag{28}$$

where r is the range and b represents the bearing measurements, and the measurements are with zero mean Gaussian noise v.

State Augmentation Model

The state augmentation model $\mathbf{x}_{m_k}$ is given by

$$\begin{aligned} x_{m_k} &= x_{v_k} + r\cos(\phi_{v_k} + b_k) \\ y_{m_k} &= y_{v_k} + r\sin(\phi_{v_k} + b_k) \end{aligned} \tag{29}$$

The Monte Carlo simulations are carried out and the average position and orientation error norms for UKF and RSPKF SLAM methods are shown in Fig. 1. The control noise is 1 m/s in speed and 1 degree in steering angle. Similarly, the measurement noise in range and bearing is assumed as 1 meter and 1 degree, respectively. Therefore, the process covariance matrix Q=R=*diag*(1, π/180). The vehicle speed is taken as 2 m/s and time interval between two control signals is set by 0.05 seconds. The time interval between the two observations is assumed as 2.5 seconds.

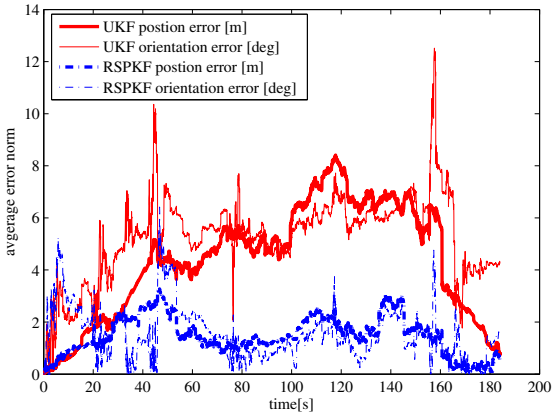


Fig. 1. Average position error norm

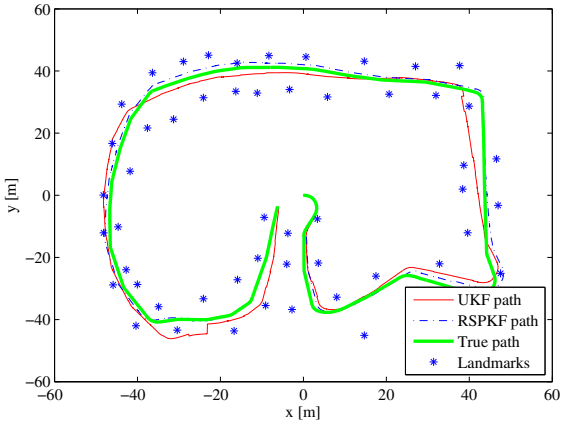


Fig. 2. The feature map and the estimated robot path

The feature map and the estimated paths based on the filtering methods are shown in Fig. 2.

4.2 Experimental Results in 3D

In this section, an experimental data set provided by Oliver Wulf is used [9]. This data set was recorded at the Leibniz University Campus and contains 468 3D scans, each with approximately 20,000 data points. A scan is given by three columns in *x*, *y*, and *z-axes*. The initial pose estimates are given by x_v, y_v as position, and θ_v as orientation in 3D. The ground truth pose data is available in 6D, $x_v = [x_{v_p} \quad x_{v_o}]$, and the size of the map is about 30 meter by 60 meter. The proposed RPSKF-SLAM method requires the Gaussian noise; however, the relative Odometry error variation is neither zero-mean nor Gaussian as shown in Fig. 3. Therefore, the problem becomes more challenging with respect to the Gaussian case. The vehicle model, observation model, and the augmentation models are expressed below.

Vehicle Model

The vehicle model function given by *f,* and it can be disclosed explicitly as in (30) for the odometry data having the relative rigid body transformation parameters. The odometry data is provided by the relations of $\mathbf{u} = [\delta x \ \delta y \ \delta z \ \delta\alpha \ \delta\beta \ \delta\gamma]$. The vehicle state vector is represented by $\mathbf{x}_v = [\mathbf{x}_{v_p} \quad \mathbf{x}_{v_o}]$ where $\mathbf{x}_{v_p} = [x \ y \ z]$ and $\mathbf{x}_{v_o} = [\alpha \ \beta \ \gamma]$ denote the robot position and the orientation, respectively. In the vector representations, the transpose *T* symbol is dropped for convenience.

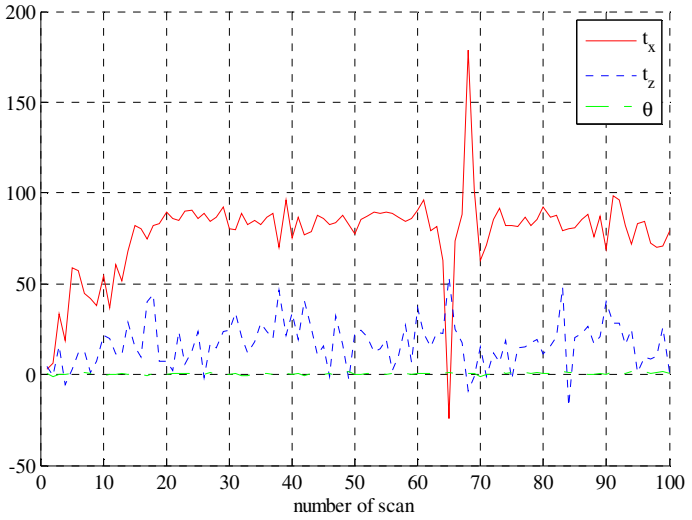


Fig. 3. Odometry error variation for the first 100 scans in Hannover dataset [9]. Translation errors are in cm and rotation error is in radian.

$$\begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ z_{k-1} \end{bmatrix} + Rot(x_{v_{o,k}}) \begin{bmatrix} \delta x_{k-1} \\ \delta y_{k-1} \\ \delta z_{k-1} \end{bmatrix}$$
$$\begin{bmatrix} \alpha_k \\ \beta_k \\ \gamma_k \end{bmatrix} = \begin{bmatrix} \alpha_{k-1} \\ \beta_{k-1} \\ \gamma_{k-1} \end{bmatrix} + \begin{bmatrix} \delta\alpha_{k-1} \\ \delta\beta_{k-1} \\ \delta\gamma_{k-1} \end{bmatrix} \tag{30}$$

where *Rot* matrix represent the three successive rotations defined by the Euler angles in *x*, *y*, and *z-axes*.

Observation Model

The observation model is based on the feature extraction method proposed by Ulas and Temeltas [10]. The plane features are used as landmarks and are encoded in the state vector with their infinite plane representations. The observation model *h* is given by

$$z_k = h(\mathbf{x}, \mathbf{u}_{k-1}) + v_{k-1}$$
$$z_k = \begin{bmatrix} n_{F_k}^L \\ d_{F_k}^L \end{bmatrix} = \begin{bmatrix} Rot^T(x_{v_{o,k}}) n_{F_k}^W \\ d_{F_k}^W + (n_{F_k}^W)^T x_{v_{p,k}} \end{bmatrix} + v_{k-1} \tag{31}$$

where $n_{F_k}^L$ is the plane normal vector represented in the local (*L*) frame, and $d_{F_k}^L$ is the plane minimum distance to the robot location $x_{v_{p,k}}$ provided by the feature extraction method. The reader is referred to [10] for more information about the feature extraction method. Based on the robot orientation $x_{v_{o,k}}$ and location $x_{v_{p,k}}$ the plane patch parameters are transformed to the world (W) frame for state augmentation.

State Augmentation Model

The state augmentation model is given by

$$n_F^W = Rot(x_{v_o}) n_F^L$$
$$d_F^W = d_F^L - (n_F^W)^T x_{v_p} \tag{32}$$

The infinite plane representations in local and world frame are shown in Fig. 4.

For the data association purpose, the other plane properties such as center of gravity of the planes G_F^L, the covariance matrix C_F^L of the plane points and convex hull points $\Delta^L{}_{XYZ,F}$ are also transferred to the world frame by using the estimated robot position.

$$
\begin{aligned}
C_F^W &= Rot(x_{v_o})C_F^L \\
G_F^W &= Rot(x_o)G_F^L + x_{v_p} \\
\Delta^W{}_{XYZ,F} &= Rot(x_{v_o})\Delta^L{}_{XYZ,F} + x_{v_p}
\end{aligned}
\tag{33}
$$

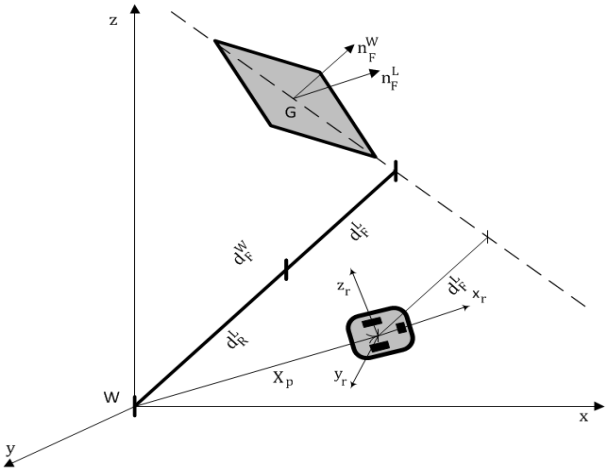


Fig. 4. Infinite Plane representation in local and world frame

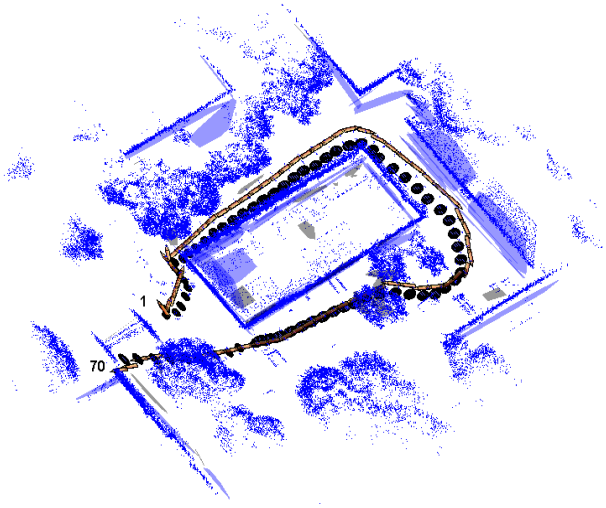


Fig. 5. Estimated planar map of the environment. The estimated robot position with uncertainty ellipsoids and ground truth path (orange) are shown.

In Fig. 5, the planar map constructed form the SLAM and the actual robot path is shown. In addition, the error uncertainty ellipsoids of the robot 3D position with their mean are shown on the map. Here, the point cloud is registered based on the ground truth as the reference.

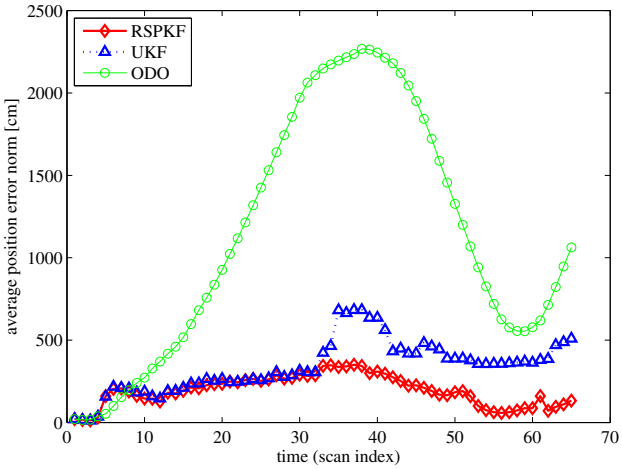


Fig. 6. Average position error norm

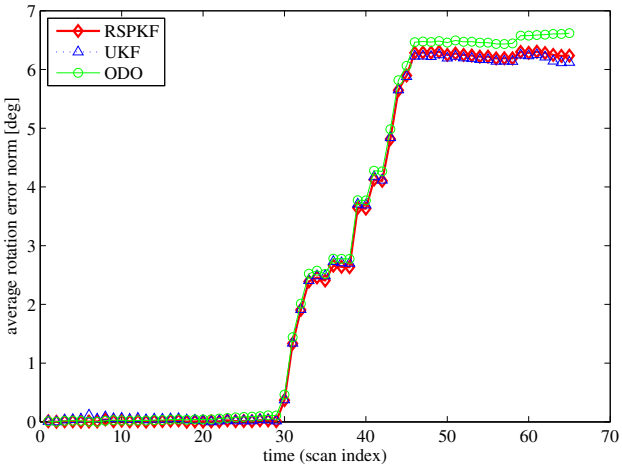


Fig. 7. Average rotation error norm

The results show that UKF and CKF based SLAM satisfy the similar results with a maximum position error around 6.5 meters (in the 35 time index). On the other hand, the RSPKF based SLAM has a maximum of 4 meter position error norm and more accurate than the conventional sigma point approaches. The rotation error norm looks similar for all filter types.

5 Conclusion

In this study, a new filtering method based on randomized sigma point sampling is introduced for localization mapping problem. The advantage of the proposed method is that the estimations are unbiased and does not yield the systematic error which is always the case of the classical filtering approaches. The performance evaluations are given for both simulations and experimental data. The proposed method is more accurate than the traditional sigma point Kalman filters like UKF and CKF which have similar performances. Although this approximation takes more computational time, it can be used accurately in SLAM problems without any systematic error.

Acknowledgments. This work was supported in part by the TUBITAK under Grant 110E194.

References

1. Dissanayake, M.W.M.G., Newman, P., Clark, S., Durrant-Whyte, H.F., Csorba, M.: A solution to the simultaneous localization and map building (SLAM) problem. IEEE Transactions on Robotics and Automation 17, 229–241 (2001)
2. Thrun, W.B.S., Fox, D.: Probabilistic Robotics. MIT Press (2005)
3. Julier, S.J., Uhlmann, J.K.: New extension of the Kalman filter to nonlinear systems. In: Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 3068, pp. 182–193 (1997)
4. Arasaratnam, I., Haykin, S.: Cubature Kalman Filters. IEEE Transactions on Automatic Control 54, 1254–1269 (2009)
5. Dunik, J., Straka, O., Simandl, M.: The Development of a Randomised Unscented Kalman Filter. In: Proceedings of the 18th IFAC World Congress, Milano, Italy (2011)
6. Genz, A., Monahan, J.: Stochastic Integration Rules for Infinite Regions. SIAM J. Sci. Comput. 19, 426–439 (1998)
7. Dunik, J., Simandl, M., Straka, O.: Adaptive choice of scaling parameter in derivative-free local filters. In: 13th Conference on Information Fusion (FUSION), pp. 1–8 (2010)
8. Fischler, M.A., Bolles, R.C.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM 24, 381–395 (1981)
9. Wulf, O.: Hannover, Leibniz University Campus, `http://kos.informatik.uni-osnabrueck.de/3Dscans/`
10. Ulas, C., Temeltas, H.: A fast and robust scan matching algorithm based on ML-NDT and feature extraction. In: International Conference on Mechatronics and Automation (ICMA), pp. 1751–1756 (2011)

A Novel Linear, Unbiased Estimator to Fuse Delayed Measurements in Distributed Sensor Networks with Application to UAV Fleet

Ronan Arraes Jardim Chagas and Jacques Waldmann

Instituto Tecnológico de Aeronáutica,
São José dos Campos, SP, 12.228-900, Brazil
ronan.jardim@gmail.com,
jacques@ita.br

Abstract. This paper proposes a novel methodology to fuse delayed measurements in a distributed sensor network. The algorithm derives from the linear minimum mean square error estimator and yields a linear, unbiased estimator that fuses the delayed measurements. Its performance regarding the estimation accuracy, computational workload and memory storage needs is compared to the classical Kalman filter reiteration that achieves the minimum mean square error in linear and Gaussian systems. The comparison is carried out using a simulated distributed sensor network that consists of a UAV fleet in formation flight in which the GPS measurements and relative positions are exchanged among neighboring network nodes. The novel technique yields similar performance to the reiterated Kalman filtering, which is the optimal linear Gaussian solution, while demanding less storage capacity and computational throughput in the problems of interest.

Keywords: Delayed measurements, measurement transportation, distributed Kalman filter, sensors network, UAV fleet.

Notation and Abbreviations

DCM	Direction Cossine Matrix.
MMSE	Minimum Mean Square Error.
y	Scalar.
$\mathbf{y}$	Vector.
$\mathbf{A}$	Matrix.
$\mathbf{I}_n$	Identity matrix of size n.
$[\mathbf{y}]_\times \mathbf{x}$	Matrix representation of the cross product $\mathbf{y} \times \mathbf{x}$.
$\mathbf{D}_b^a$	DCM that rotates from the a coordinate frame to the b coordinate frame.
$\boldsymbol{\rho}_l$	Transport rate represented in the local horizontal frame.
$\boldsymbol{\Omega}_{e,l}$	Earth's angular rate represented in the local horizontal frame.
$\mathbf{Asp}_l$	Specific force represented in the local horizontal frame.
$\boldsymbol{\Delta}\mathbf{R}_l$	INS position error represented in the local horizontal frame.
$\boldsymbol{\Delta}\mathbf{V}_l$	INS velocity error represented in the local horizontal frame.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_8

$\boldsymbol{\Psi}$	Misalignment from the computed coordinate frame to the platform coordinate frame.
$\boldsymbol{\nabla}$	Accelerometer triad bias.
$\boldsymbol{\epsilon}$	Rate-gyro triad drift.
R_e	Earth radius at the latitude of the vehicle.
g_e	Gravitation at the latitude of the vehicle.
$\boldsymbol{\Omega}_{k-1,i}$	Set of all measurements received by the i-th node up to instant $k-1$.
$\hat{\mathbf{x}}_{k\|k-1,i}$	Estimate of the vector $\mathbf{x}_{k,i}$ using all measurement up to instant $k-1$.
$\mathbf{P}_{k\|k-1,i}$	Estimation error covariance of the vector $\mathbf{x}_{k,i}$ using all measurement up to instant $k-1$.

1 Introduction

A network consisting of spatially distributed sensor nodes with local processing units for acquiring local measurements and estimating the state vector of a dynamic system can produce more accurate estimates when information is exchanged among the nodes. Such distributed estimation approach is less susceptible to a single point failure that can cripple centralized estimation schemes [1, 2].

The processing unit at each sensing node iterates a Kalman filter. Two distinct possibilities were evaluated in literature: exchanging sensor measurements among nodes [3–5] and exchanging state vector estimates produced by the local Kalman filters [1, 2].

Distributed filtering has been widely investigated when network nodes share the dynamic model [1–4]. However, interesting problems call for algorithms that can perform the distributed estimation when the nodes do not share the state-model, e.g. an UAV fleet [6], a set of satellites in orbit [7, 8], and spacecrafts flying into the deep space [9–11]. To the best knowledge of the authors, the first approach to information fusion in such a network was [12]. A very similar algorithm was proposed in [13]. In both investigations the nodes' states should be related by a linear transformation. Here, the subject is probed further to deal with delayed measurements in a distributed network wherein a particular dynamic model is embedded in each node. It is shown here that in such a scenario the exchanged measurements can only be fused if additional information is gathered to relate measurements from neighboring nodes with a node's state.

It is expected that the required information will arrive with delays. The accuracy of the distributed estimation by the network could be severely degraded had the delayed measurements been processed without adequate caution. There is a myriad of techniques in the literature to fuse delayed measurements in a non-distributed estimation. Reference [14] compared many methods regarding performance, storage necessity, and computational workload. Among the techniques discussed, references [15–17] need the knowledge that a delayed measurement has not been received at a node so that parallel computations can be started to optimally fuse the delayed measurement when it arrives. On the other hand, the algorithms in [18, 19] have not been extended to handle multiple delays. These inconveniences preclude the use of such algorithms in distributed

estimation problems. Reference [20] compared algorithms to fuse out-of-sequence measurements in sensor networks, which can be used in the distributed filtering problem addressed here. These algorithms were developed to achieve the minimum mean square error (MMSE) optimality in linear and Gaussian systems and thus require recursions that yield a heavy computational burden.

In the problems of interest, it is expected that measurements may be received with a delay on the order of thousands of the sampling step. Thus, algorithms that call for recursions will be time consuming. Here, a novel approach, thereafter called measurement transportation, has been developed based on the delayed-state Kalman filter [21] and on the suboptimal technique in [22]. This novel algorithm has been compared with a classical methodology for delayed measurement fusion: the reiterated Kalman filter [23]. This technique assures MMSE optimality if the system is linear and Gaussian.

A simulated UAV fleet in formation flight is the sensor network scenario in which GPS measurements and relative positions are exchanged among the aircraft. It turns out that this novel technique needs less storage capacity than any algorithm in [20], the computational workload is lighter in comparison with the reiterated Kalman filter, and a good overall performance is achieved for the problems of interest.

Section 2 presents the coordinate frames. The distributed estimation problem when the nodes do not share the same state dynamics model is presented in section 3. The two algorithms to fuse the delayed measurements are described in section 4. The distributed filtering in a UAV fleet is described in section 5. Simulations and results are presented in section 6. Finally, the conclusions are written in section 7.

2 Coordinate Frames

The **true local horizontal frame** is used to represent the INS errors. In the true vehicle position, its X-axis points towards north, its Y-axis points towards east, and its Z-axis points down. This coordinate system is thereafter indicated with the l subscript.

The **computed coordinate frame** is defined as the local horizontal frame at the position computed by the INS.

The **platform coordinate frame** is defined as the local horizontal frame computed by the INS.

The **body coordinate frame** is defined as the inertial sensors coordinate frame. It is usually assumed to be aligned with the vehicle coordinate frame in strapdown IMUs or aligned with the platform coordinate frame in IMUs mounted on a stabilized platform. This coordinate frame is thereafter indicated with the b subscript.

The **Earth-Centered-Earth-Fixed** coordinate frame has its origin at the center of the Earth, its X axis lies on the equatorial plane and points to the Greenwich meridian, its Z axis is aligned with the Earth's rotation axis, and its Y axis completes the right-hand coordinate frame. It is thereafter indicated with the e subscript.

The **WGS-84 ellipsoid Earth-fixed coordinate frame** is used to represent the GPS data.

3 Distributed Estimation

Distributed estimation has been widely studied in the literature using a set of sensors that measure states components from a common process dynamics [1–4]. Many algorithms have been developed to fuse the network data to improve the overall estimation accuracy and to provide robustness. The exchanged information can be the state vector estimates from the neighboring nodes or the measurements from the corresponding sensors. However, many interesting problems call for algorithms that can perform distributed estimation when each node observes a different process, e.g. an UAV fleet. This scenario is modeled as follows for the i-th node:

$$\begin{aligned} \mathbf{x}_{k+1,i} &= \mathbf{F}_{k,i}\mathbf{x}_{k,i} + \mathbf{B}_{k,i}\mathbf{u}_{k,i} + \mathbf{G}_{k,i}\mathbf{w}_{k,i} \\ \mathbf{y}_{k,i} &= \mathbf{H}_{k,i}\mathbf{x}_{k,i} + \mathbf{v}_{k,i} \ , \end{aligned} \tag{1}$$

where $\mathbf{F}_{k,i}$ is a $M_i \times M_i$ state-transition matrix, $\mathbf{B}_{k,i}\mathbf{u}_{k,i}$ is a deterministic and known control vector, $\mathbf{G}_{k,i}\mathbf{w}_{k,i}$ is the model noise assumed to be a zero-mean, white Gaussian random vector with $\mathbf{Q}_{k,i}$ covariance matrix, $\mathbf{H}_{k,i}$ is a $N_i \times M_i$ measurement matrix, and $\mathbf{v}_{k,i}$ is the measurement noise modeled as a zero-mean, white Gaussian random vector with $\mathbf{R}_{k,i}$ covariance matrix. The initial state $\mathbf{x}_{0,i}$ is a Gaussian random vector with mean $\mathbf{m}_{0,i}$ and covariance $\mathbf{P}_{0,i}$. Additionally it is assumed that all measurement and model noises through the network are independent to each other and are also independent to the initial state $\mathbf{x}_{0,i}$ at every node.

As mentioned before, the nodes does not share the same dynamics, thus the j-th node measurement cannot be directly used by the i-th node. If the latter receives, in the instant k, a measurement from the former, then the posterior probability density function is

$$p(\mathbf{x}_{k,i}|\mathbf{y}_{k,i}, \mathbf{y}_{k,j}, \boldsymbol{\Omega}_{k-1,i}) \ , \tag{2}$$

where $\boldsymbol{\Omega}_{k-1,i}$ is the set of all fused measurements up to instant $k-1$ at node i. Using Bayes rules, one can verify that

$$\begin{aligned} &p(\mathbf{x}_{k,i}|\mathbf{y}_{k,i}, \mathbf{y}_{k,j}, \boldsymbol{\Omega}_{k-1,i}) = \\ &\quad C_k p(\mathbf{y}_{k,j}|\mathbf{x}_{k,i}, \mathbf{y}_{k,i}, \boldsymbol{\Omega}_{k-1,i}) p(\mathbf{y}_{k,i}|\mathbf{x}_{k,i}) p(\mathbf{x}_{k,i}|\boldsymbol{\Omega}_{k-1,i}) \ , \end{aligned} \tag{3}$$

with C_k being a normalizing constant that yields

$$\int_{\mathbb{R}^{M_i}} p(\mathbf{x}_{k,i}|\mathbf{y}_{k,i}, \mathbf{y}_{k,j}, \boldsymbol{\Omega}_{k-1,i}) d\mathbf{x}_{k,i} = 1 \ .$$

The computation of the p.d.f. $p(\mathbf{y}_{k,j}|\mathbf{x}_{k,i}, \mathbf{y}_{k,i}, \boldsymbol{\Omega}_{k-1,i})$ will eventually need some sort of additional information to relate the j-th neighboring node measurement to the i-th node states as in the following function:

$$\mathbf{y}_{k,j} = \mathbf{h}_k^{i,j}(\mathbf{x}_{k,i}) \ . \tag{4}$$

If such function can be constructed, then the j-th node measurement can be fused at the i-th node as if it was actually produced by a sensor that is local to the i-th node. This methodology was used in [24] in which the exchanging of position measurements from robots was proposed. The authors verified that it could only be possible if the relative position vectors had to be available. Additionally, in case function $\mathbf{h}_k^{i,j}(\cdot)$ is nonlinear, then it should be linearized about $\hat{\mathbf{x}}_{k|k-1,i}$ as in the extended Kalman filter algorithm.

4 Delayed Measurements

In a distributed sensor network, it is expected that the exchanged measurements will spread across the network and reach distinct nodes with varying time delays. Here, the sample step is small enough such that the node dynamics has been assumed constant between two consecutive sample steps. Thus a delay smaller than the sampling step is negligible [25]. Additionally, if a measurement is received at instant t_l in which $t_{k-n} \leq t_l \leq t_{k-n+1}$, then it has been considered as delayed by n sample steps. Under these assumptions, a measurement with a delay lower than one sample step can be fused as usual. However if a measurement happens to reach a node with a delay higher than the sampling interval, then the local estimate could be severally degraded had the measurement been naïvely fused.

It should be noticed that all previous methodologies need to store information to accomplish the delayed measurement fusion [14, 20, 23]. Thus one must define a maximum allowed delay, thereafter called max. If any measurement with a delay higher than max is received, then it will be discarded.

4.1 Measurement Grouping

Let $\boldsymbol{\nu}_k^i$ be the set of all measurements that the i-th node received in the instant k. It has been considered that these measurements from the neighboring nodes as described in eq. 4 depend on a function $\mathbf{h}_k^{i,j}(\mathbf{x}_{k,i})$ that is either linear or has been linearized about $\hat{\mathbf{x}}_{k|k-1,i}$, thus $\mathbf{y}_{k,j} = \mathbf{H}_k^{i,j}\mathbf{x}_{k,i} + \mathbf{v}_{k,j}$. The aforementioned set can be partitioned into subsets according to the measurement delay, hereafter called $\boldsymbol{\nu}_{k,\Delta_n}^i$. Thus the subset $\boldsymbol{\nu}_{k,\Delta_n}^i$ is composed of all measurements received by the i-th node in the instant k delayed by Δ_n sample steps, where $0 \leq \Delta_0 < \Delta_1 < \cdots < \Delta_L \leq max$.

The measurements in the same subset $\boldsymbol{\nu}_{k,\Delta_n}^i$ can be fused into one single vector to reduce the computational burden. In case the sensors do not share the same measurement matrix, the measurements vectors are projected into the state-space as follows [1]:

$$\begin{aligned}
\mathbf{y}^{f}_{k,\Delta_n,i} &= \sum_{j\in\boldsymbol{\nu}^{i}_{k,\Delta n}} \mathbf{H}^{i,j,T}_{k-\Delta_n}\mathbf{R}^{-1}_{k-\Delta_n,j}\mathbf{y}_{k-\Delta_n,j} = \\
&= \left(\sum_{j\in\boldsymbol{\nu}^{i}_{k,\Delta_n}} \mathbf{H}^{i,j,T}_{k-\Delta_n}\mathbf{R}^{-1}_{k-\Delta_n,j}\mathbf{H}^{i,j}_{k-\Delta_n}\right)\mathbf{x}_{k-\Delta_n,i} + \\
&\qquad + \sum_{j\in\boldsymbol{\nu}^{i}_{k,\Delta_n}} \mathbf{H}^{i,j,T}_{k-\Delta_n}\mathbf{R}^{-1}_{k-\Delta_n,j}\mathbf{v}_{k-\Delta_n,j} = \\
&= \mathbf{H}^{f}_{k,\Delta_n,i}\mathbf{x}_{k-\Delta_n,i} + \mathbf{v}^{f}_{k,\Delta_n,i} \ , \qquad (5a)\\
\mathbf{R}^{f}_{k,\Delta_n,i} &= cov\{\mathbf{y}^{f}_{k,\Delta_n,i}\mathbf{y}^{f,T}_{k,\Delta_n,i}\} = \\
&= \left(\sum_{j\in\boldsymbol{\nu}^{i}_{k,\Delta_n}} \mathbf{H}^{i,j,T}_{k-\Delta_n}\mathbf{R}^{-1}_{k-\Delta_n,j}\mathbf{H}^{i,j}_{k-\Delta_n}\right) = \mathbf{H}^{f}_{k,\Delta_n,i} \ , \qquad (5b)
\end{aligned}$$

in which the information form of the Kalman filter should be used. On the other hand, if all measurements in the set share the same measurement matrix, then measurement fusion can be carried out by [26]

$$\begin{cases}
\mathbf{R}^{f,-1}_{k,\Delta_n,i} = \displaystyle\sum_{j\in\boldsymbol{\nu}^{i}_{k,\Delta n}} \mathbf{R}^{-1}_{k-\Delta_n,j} \\
\mathbf{y}^{f}_{k,\Delta_n,i} = \mathbf{R}^{f}_{k,\Delta_n,i}\left[\displaystyle\sum_{j\in\boldsymbol{\nu}^{i}_{k,\Delta n}} \mathbf{R}^{-1}_{k-\Delta_n,j}\mathbf{y}_{k-\Delta_n,j}\right] \ .
\end{cases} \qquad (6)$$

Finally, the problem is reduced to fuse the measurements $\mathbf{y}^{f}_{k,\Delta_n,i}$, where $n \in [0, 1, 2, \cdots, L]$, and to compute (or to approximate) the p.d.f.

$$p(\mathbf{x}_{k,i}|\mathbf{y}^{f}_{k,\Delta_0,i}, \mathbf{y}^{f}_{k,\Delta_1,i}, \cdots, \mathbf{y}^{f}_{k,\Delta_L,i}, \boldsymbol{\Omega}_{k-1,i}) = p(\mathbf{x}_{k,i}|\boldsymbol{\Omega}_{k,i}) \ .$$

One should note that a consensus over the network is not pursued here as in [1] or [2]. The nodes send to the neighbors the local measurements by the time they are acquired. Additionally, a node can retransmit the information received to permit that a measurement reach, even if delayed, nodes outside its neighborhood. The idea is to fuse all available information (delayed or not) without waiting for the communication steps to achieve the network consensus.

4.2 The Reiterated Kalman Filter

When node of a distributed sensor network receives a delayed measurement, then the optimal fusion is accomplished if the posterior estimate is exactly the same as it would be if the measurement had been received at the time of its production, without the delay. The most direct way to accomplish that is to reiterate the Kalman filter from the instant when the measurement was produced until the

present time [23]. However, one should notice that the MMSE optimality is assured only in linear and Gaussian systems.

The aforementioned methodology can be only used if the updated estimates and covariance matrices together with the fused measurements and respective statistics are stored from instant $k - max$ up to instant $k - 1$. Let $\mathbf{y}^u_{k-n,i}$ and $\mathbf{R}^u_{k-n,i}$ be, respectively, the measurement vector and its covariance that was used in the Kalman filter update step at instant $k - n$ by the i-th node. Thus the algorithm can be written as follows:

- $j = \Delta_L$, $n = L$
- WHILE $j \geq 0$
  - IF $j = \Delta_n$ THEN
    - $n = n - 1$
    - Fuse the measurement vectors $\mathbf{y}^u_{k-j,i}$ and $\mathbf{y}^f_{k,j,i}$ into $\mathbf{y}^d_{k,j,i}$.
    - Fuse the statistics of the measurement vectors $\mathbf{R}^u_{k-j,i}$ and $\mathbf{R}^f_{k,j,i}$ into $\mathbf{R}^d_{k,j,i}$.
  - ELSE
    - $\mathbf{y}^d_{k,j,i} = \mathbf{y}^u_{k-j,i}$
    - $\mathbf{R}^d_{k,j,i} = \mathbf{R}^u_{k-j,i}$
  - ENDIF
  - Using $\mathbf{y}^d_{k,j,i}$, $\mathbf{R}^d_{k,j,i}$, $\hat{\mathbf{x}}_{k-j|k-j-1,i}$, and $\mathbf{P}_{k-j|k-j-1,i}$ apply the update step of the Kalman filter and overwrite $\hat{\mathbf{x}}_{k-j|k-j,i}$ and $\mathbf{P}_{k-j|k-j,i}$.
  - Apply the propagation step of the Kalman filter and overwrite $\hat{\mathbf{x}}_{k-j+1|k-j,i}$ and $\mathbf{P}_{k-j+1|k-j,i}$.
  - $\mathbf{y}^u_{k-j,i} = \mathbf{y}^d_{k,j,i}$
  - $\mathbf{R}^u_{k-j,i} = \mathbf{R}^d_{k,j,i}$
  - j = j - 1
- ENDWHILE

If the node does not receive any measurements delayed more than max sampling steps, then the posterior estimate will be optimal in the MMSE sense in a linear and Gaussian system. However the computational workload is huge. If the most delayed measurement was produced n sampling steps in the past, then this algorithm will iterate the Kalman filter $n + 1$ times.

4.3 Measurement Transportation [27]

A novel approach to the fusion of delayed measurements in the Kalman filter is proposed here: the measurement transportation. The algorithm, though suboptimal in the MMSE sense, has achieved a good performance in the situations of interest with lighter computational load and less storage necessity than the reiterated Kalman filter. The approach is based on the technique in [22] and on the delayed-state Kalman filter [21], which was constructed to fuse a measurement composed of two consecutive states, $\mathbf{x}_{k-1}$ and $\mathbf{x}_k$.

Using the model in eq. 1, the state $\mathbf{x}_{k-n,i}$ can be related to $\mathbf{x}_{k,i}$ as follows

$$\begin{aligned}\mathbf{x}_{k-n,i} = &\left[\prod_{l=0}^{n-1}\mathbf{F}_{k-(n-l),i}^{-1}\right]\mathbf{x}_{k,i} - \sum_{j=1}^{n}\left(\left[\prod_{l=0}^{n-j}\mathbf{F}_{k-(n-l),i}^{-1}\right]\mathbf{G}_{k-j,i}\mathbf{w}_{k-j,i}\right) - \\ &- \sum_{j=1}^{n}\left(\left[\prod_{l=0}^{n-j}\mathbf{F}_{k-(n-l),i}^{-1}\right]\mathbf{B}_{k-j,i}\mathbf{u}_{k-j,i}\right) \ .\end{aligned} \tag{7}$$

Thus the fused delayed measurement in each subset $\boldsymbol{\nu}_{k,\Delta_n}^{i}$ can be transported to the present instant by

$$\mathbf{y}_{k,\Delta_n,i}^{f} = \mathbf{H}_{k,\Delta_n,i}^{p}\mathbf{x}_k + \mathbf{u}_{k,\Delta_n,i}^{p} + \mathbf{v}_{k,\Delta_n,i}^{p} = \mathbf{y}_{k,\Delta_n,i}^{p} \ , \tag{8}$$

where

$$\mathbf{H}_{k,\Delta_n,i}^{p} = \mathbf{H}_{k,\Delta_n,i}^{f}\left[\prod_{l=0}^{\Delta_n-1}\mathbf{F}_{k-(\Delta_n-l),i}^{-1}\right] \ , \tag{9}$$

$$\mathbf{u}_{k,\Delta_n,i}^{p} = \mathbf{H}_{k,\Delta_n,i}^{f}\sum_{j=1}^{\Delta_n}\left(\left[\prod_{l=0}^{\Delta_n-j}\mathbf{F}_{k-(\Delta_n-l),i}^{-1}\right]\mathbf{B}_{k-j,i}\mathbf{u}_{k-j,i}\right) \ , \tag{10}$$

$$\mathbf{v}_{k,\Delta_n,i}^{p} = \mathbf{v}_{k,\Delta_n,i}^{f} - \mathbf{H}_{k,\Delta_n,i}^{f}\underbrace{\sum_{j=1}^{\Delta_n}\left(\left[\prod_{l=0}^{\Delta_n-j}\mathbf{F}_{k-(\Delta_n-l),i}^{-1}\right]\mathbf{G}_{k-j,i}\mathbf{w}_{k-j,i}\right)}_{\star} \ . \tag{11}$$

Notice that the fused delayed measurement noise $\mathbf{v}_{k,\Delta_n,i}^{p}$ has a covariance ellipsoid larger than that of the original measurement due to the summation of the model noise samples from instants $k-\Delta_n$ up to $k-1$ ($\star$). Thus, the fused delayed measurement signal-to-noise ratio degrades with respect to the instantaneous measurement.

By stacking the measurements in each subset $\boldsymbol{\nu}_{k,\Delta_n}^{i}$, $n \in [0,1,2,\cdots,L]$, the measurement vector to be fused at instant k by the i-th node is

$$\mathbf{y}_{k,i}^{e,p} = \mathbf{H}_{k,i}^{e,p}\mathbf{x}_{k,i} + \mathbf{v}_{k,i}^{e,p} \ , \tag{12}$$

where

$$\mathbf{y}_{k,i}^{e,p} = \left[\mathbf{y}_{k,\Delta_0,i}^{p,T}\ \mathbf{y}_{k,\Delta_1,i}^{p,T}\cdots\mathbf{y}_{k,\Delta_L,i}^{p,T}\right]^T - \mathbf{u}_{k,i}^{e,p} \ , \tag{13}$$

$$\mathbf{u}_{k,i}^{e,p} = \left[\mathbf{u}_{k,\Delta_0,i}^{p,T}\ \mathbf{u}_{k,\Delta_1,i}^{p,T}\cdots\mathbf{u}_{k,\Delta_L,i}^{p,T}\right]^T \ , \tag{14}$$

$$\mathbf{H}_{k,i}^{e,p} = \left[\mathbf{H}_{k,\Delta_0,i}^{p,T}\ \mathbf{H}_{k,\Delta_1,i}^{p,T}\cdots\mathbf{H}_{k,\Delta_L,i}^{p,T}\right]^T \ , \tag{15}$$

$$\mathbf{v}_{k,i}^{e,p} = \left[\mathbf{v}_{k,\Delta_0,i}^{p,T}\ \mathbf{v}_{k,\Delta_1,i}^{p,T}\cdots\mathbf{v}_{k,\Delta_L,i}^{p,T}\right]^T \ . \tag{16}$$

Under the foregoing assumptions, it is clear that $\mathbf{v}_{k,i}^{e,p}$ has zero mean. Thus its covariance matrix is

$$\mathbf{R}_{k,i}^{e,p} = E\{\mathbf{v}_{k,i}^{e,p}\mathbf{v}_{k,i}^{e,p,T}\} = \begin{bmatrix} E\{\mathbf{v}_{k,\Delta_0,i}^{p}\mathbf{v}_{k,\Delta_0,i}^{p,T}\} & \cdots & E\{\mathbf{v}_{k,\Delta_0,i}^{p}\mathbf{v}_{k,\Delta_L,i}^{p,T}\} \\ \vdots & \ddots & \vdots \\ E\{\mathbf{v}_{k,\Delta_L,i}^{p}\mathbf{v}_{k,\Delta_0,i}^{p,T}\} & \cdots & E\{\mathbf{v}_{k,\Delta_L,i}^{p}\mathbf{v}_{k,\Delta_L,i}^{p,T}\} \end{bmatrix} , \quad (17)$$

where the expectations $E\{\mathbf{v}_{k,\Delta_n,i}^{p}\mathbf{v}_{k,\Delta_m,i}^{p,T}\}$, $n, m \in [0, 1, 2, \cdots, L]$, can be computed as follows

$$E\{\mathbf{v}_{k,n,i}^{p}\mathbf{v}_{k,m,i}^{p,T}\} = \\ = \begin{cases} \mathbf{R}_{k,n,i}^{f} \cdot \delta(n-m) + \\ \quad + \mathbf{H}_{k,n,i}^{f} \cdot \sum_{j=1}^{min(n,m)} \left(\left[\prod_{i=0}^{n-j} \mathbf{F}_{k-(n-i)}^{-1} \right] \mathbf{Q}_{k-j} \left[\prod_{i=0}^{m-j} \mathbf{F}_{k-(m-i)}^{-1} \right]^{T} \right) \mathbf{H}_{k,m,i}^{T,f}, \\ \qquad\qquad n > 0, m > 0, \\ \mathbf{R}_{k,0,i}^{f} \cdot \delta(n) \cdot \delta(m), \quad m = 0 \text{ or } n = 0 \ , \end{cases} \quad (18)$$

where $\delta(n)$ is the Kronecker's delta.

The usual Kalman filter algorithm cannot be used with the measurement in eq. 12, because the measurement noise $\mathbf{v}_{k,i}^{e,p}$ is correlated with the model noise. Thus, the linear MMSE estimate produced by the Kalman filter update step, defined as

$$\begin{cases} \hat{\mathbf{x}}_{k|k,i} = E\{\mathbf{x}_{k,i}|\mathbf{y}_{k,i}^{e,p}, \boldsymbol{\Omega}_{k-1,i}\} = \\ \qquad = E\{\mathbf{x}_{k,i}|\boldsymbol{\Omega}_{k-1,i}\} + \mathbf{C}_{k,i}^{xy}\mathbf{C}_{k,i}^{yy,-1}(\mathbf{y}_{k,i}^{e,p} - E\{\mathbf{y}_{k,i}^{e,p}|\boldsymbol{\Omega}_{k-1,i}\}) \\ \mathbf{C}_{k,i}^{xy} = E\{(\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}|\boldsymbol{\Omega}_{k-1,i}\}) \cdot (\mathbf{y}_{k,i}^{e,p} - E\{\mathbf{y}_{k,i}^{e,p}|\boldsymbol{\Omega}_{k-1,i}\})^{T}|\boldsymbol{\Omega}_{k-1,i}\} \\ \mathbf{C}_{k,i}^{yy} = E\{(\mathbf{y}_{k,i}^{u} - E\{\mathbf{y}_{k,i}^{e,p}|\boldsymbol{\Omega}_{k-1,i}\}) \cdot (\mathbf{y}_{k,i}^{e,p} - E\{\mathbf{y}_{k,i}^{e,p}|\boldsymbol{\Omega}_{k-1,i}\})^{T}|\boldsymbol{\Omega}_{k-1,i}\} \\ \mathbf{P}_{k|k,i} = \mathbf{P}_{k|k-1,i} - \mathbf{C}_{k,i}^{xy}\mathbf{C}_{k,i}^{yy,-1}\mathbf{C}_{k,i}^{xy,T} \ , \end{cases} \quad (19)$$

needs to be rewritten [28].

One can see that [28, 29]

$$E\{\mathbf{y}_{k,i}^{e,p}|\boldsymbol{\Omega}_{k-1,i}\} = \mathbf{H}_{k,i}^{e,p}E\{\mathbf{x}_{k,i}|\boldsymbol{\Omega}_{k-1,i}\} = \mathbf{H}_{k,i}^{e,p}\hat{\mathbf{x}}_{k|k-1,i} \ , \quad (20)$$

$$\mathbf{C}_{k,i}^{xy} = \mathbf{P}_{k|k-1,i}\mathbf{H}_{k,i}^{e,p,T} + \mathbf{S}_{k,i} \ , \quad (21)$$

$$\mathbf{C}_{k,i}^{yy} = \mathbf{H}_{k,i}^{e,p}\mathbf{P}_{k|k-1,i}\mathbf{H}_{k,i}^{e,p,T} + \mathbf{R}_{k,i}^{e,p} + \mathbf{H}_{k,i}^{e,p}\mathbf{S}_{k,i} + \mathbf{S}_{k,i}^{T}\mathbf{H}_{k,i}^{e,p,T} \ , \quad (22)$$

where $\mathbf{S}_{k,i} = E\{(\mathbf{x}_{k,i} - \hat{\mathbf{x}}_{k|k-1,i})\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1}^{i}\}$. In the usual Kalman filter algorithm, $\mathbf{S}_{k,i}$ is zero due to the assumption that the model and measurement noise vectors are uncorrelated. However, the transported measurement, defined in eq. 12, carries the model noises from the instant it was produced up to the instant $k-1$.

The $\mathbf{S}_{k,i}$ matrix can be rewritten as

$$\begin{aligned}\mathbf{S}_{k,i} = E\{(\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\} - \\ - E\{(\hat{\mathbf{x}}_{k|k-1,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\} \ .\end{aligned} \tag{23}$$

It can be verified that $\hat{\mathbf{x}}_{k|k-1,i}$ is a random vector that depends just on all the measurement vectors fused up to instant $k-1$ at the i-th node and on the random vector $\mathbf{x}_{0,i}$. The latter has been assumed to be independent with respect to all measurements and model noise sequences throughout the network. Thus, conditioned on $\boldsymbol{\Omega}_{k-1,i}$, $\hat{\mathbf{x}}_{k|k-1,i}$ and $\mathbf{v}_{k,i}^{e,p}$ are independent, which leads to

$$\begin{aligned}&E\{(\hat{\mathbf{x}}_{k|k-1,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\} = \\ &= E\{\hat{\mathbf{x}}_{k|k-1,i}\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\} - E\{E\{\mathbf{x}_{k,i}\}\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\}\} = \\ &= E\{\hat{\mathbf{x}}_{k|k-1,i}|\boldsymbol{\Omega}_{k-1,i}\}E\{\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\} - E\{\mathbf{x}_{k,i}\}E\{\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\} \ .\end{aligned} \tag{24}$$

It can be concluded that $E\{\mathbf{v}_{k,i}^{e,p}\} = \mathbf{0}_{(L+1)M_i\times 1}$ using eqs. 11 and 16 and the assumptions regarding the model and measurement noises. However, since $\mathbf{v}_{k,i}^{e,p}$ and the fused measurements in $\boldsymbol{\Omega}_{k-1,i}$ depend on the model noise sequence, then $E\{\mathbf{v}_{k,i}^{e,p}|\boldsymbol{\Omega}_{k-1,i}\} \neq \mathbf{0}_{(L+1)M_i\times 1}$ in general. To compute this vector, the algorithm needs to store all fused measurements from instant $k-max$ up to instant $k-1$. Thus, to decrease the computational burden and the memory needs, the following approximation is used

$$E\{\mathbf{v}_{k,i}^{e,p}|\boldsymbol{\Omega}_{k-1,i}\} \approx \mathbf{0}_{(L+1)M_i\times 1} \ , \tag{25}$$

which together with eq. 24 leads to

$$E\{(\hat{\mathbf{x}}_{k|k-1,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,i}^{e,p,T}|\boldsymbol{\Omega}_{k-1,i}\} \approx \mathbf{0}_{M_i\times(L+1)M_i} \ . \tag{26}$$

The approximation in eq. 25 neglects the effect of the model noise sequence on the fused measurements in the set $\boldsymbol{\Omega}_{k-1,i}$. Notice that if the model noise is absent, then $\mathbf{v}_{k,i}^{e,p}$ and the vectors in $\boldsymbol{\Omega}_{k-1,i}$ are independent and the approximation in eq. 25 is exact. Thus, if the effect of the model noise sequence on the state vector is negligible compared to that of the system dynamics and the initial state, then the approximation in eq. 25 is accurate. Hence, the accuracy of the proposed approximation requires a sufficiently small model noise covariance matrix, which is assumed to be valid in the proposed scenario of a UAV fleet to be simulated later on.

Additionally it can be shown by induction that

$$\begin{aligned}\mathbf{x}_{k,i} = &\left[\prod_{t=1}^{n}\mathbf{F}_{k-t,i}\right]\mathbf{x}_{k-n,i} + \mathbf{G}_{k-1,i}\mathbf{w}_{k-1,i} + \mathbf{B}_{k-1,i}\mathbf{u}_{k-1,i} + \\ &+ \sum_{j=2}^{n}\left(\left[\prod_{t=1}^{j-1}\mathbf{F}_{k-t,i}\right]\mathbf{G}_{k-j,i}\mathbf{w}_{k-j,i}\right) + \sum_{j=2}^{n}\left(\left[\prod_{t=1}^{j-1}\mathbf{F}_{k-t,i}\right]\mathbf{B}_{k-j,i}\mathbf{u}_{k-j,i}\right)\end{aligned} \tag{27}$$

and setting $n = k$, one can see that

$$\begin{aligned}\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}\} = \left[\prod_{t=1}^{k} \mathbf{F}_{k-t,i}\right](\mathbf{x}_{0,i} - \mathbf{m}_{0,i}) + \\ + \mathbf{G}_{k-1,i}\mathbf{w}_{k-1,i} + \sum_{j=2}^{k}\left(\left[\prod_{t=1}^{j-1}\mathbf{F}_{k-t,i}\right]\mathbf{G}_{k-j,i}\mathbf{w}_{k-j,i}\right) .\end{aligned} \tag{28}$$

A white sequence has been assumed as the model noise $\mathbf{G}_k\mathbf{w}_k$, $k \in \mathbb{N}$. Thus

$$E\{\mathbf{G}_n\mathbf{w}_n\mathbf{w}_m^T\mathbf{G}_m^T\} = \mathbf{Q}_n \cdot \delta(n-m) . \tag{29}$$

This result together with the assumption that $\mathbf{x}_0$ is independent with respect to all model and measurement noise sequences leads to

$$\begin{aligned}E\{(\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,\Delta_n,i}^{p,T}|\boldsymbol{\Omega}_{k-1,i}\} = \left\{-\mathbf{Q}_{k-1}\left[\prod_{l=0}^{\Delta_n-1}\mathbf{F}_{k-(\Delta_n-l),i}^{-1}\right]^T\right. \\ \left. -\sum_{j=2}^{\Delta_n}\left(\left[\prod_{t=1}^{j-1}\mathbf{F}_{k-t,i}\right]\cdot\mathbf{Q}_{k-j}\left[\prod_{l=0}^{\Delta_n-j}\mathbf{F}_{k-(\Delta_n-l),i}^{-1}\right]^T\right)\right\}\cdot\mathbf{H}_{k,\Delta_n,i}^{f,T} ,\end{aligned} \tag{30}$$

for $n \in [0, 1, 2, \cdots, L]$ and $\Delta_n \neq 0$. If $\Delta_0 = 0$, then

$$E\{(\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,\Delta_0,i}^{p,T}|\boldsymbol{\Omega}_{k-1,i}\} = \mathbf{0}_{M_i\times N_i} . \tag{31}$$

The results in eqs. 26, 30 and 31 allow the computation of the matrix $\mathbf{S}_{k,i}$ as follows

$$\mathbf{S}_{k,i} = \begin{bmatrix}\left(E\{(\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,\Delta_0,i}^{p,T}|\boldsymbol{\Omega}_{k-1,i}\}\right)^T \\ \left(E\{(\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,\Delta_1,i}^{p,T}|\boldsymbol{\Omega}_{k-1,i}\}\right)^T \\ \vdots \\ \left(E\{(\mathbf{x}_{k,i} - E\{\mathbf{x}_{k,i}\})\mathbf{v}_{k,\Delta_L,i}^{p,T}|\boldsymbol{\Omega}_{k-1,i}\}\right)^T\end{bmatrix}^T . \tag{32}$$

Finally the Kalman filter update step can be performed using the linear MMSE estimate in eq. 19, which can be computed using eqs. 20, 21, and 22.

This method needs the product of the inverse of consecutive state-transition matrices, which imposes a heavy computational load. However, the state-transition matrix inverses from instant $k - max$ up to instant $k - 1$ can be stored to decrease the computational burden of computing these products. One should notice that the state-transition matrix is always invertible for discretized continuous linear systems [30].

If it is assumed that the local nodes have access to the state-transition matrix and model noise covariance matrices from instant $k-max$ up to instant $k-1$ [20], then the measurement transportation must store:

- The control signals from instant $k-max$ up to instant $k-1$: that is, $max \cdot M_i$ elements;
- (OPTIONAL) The state-transition matrix inverses from instant $k-max$ up to instant $k-1$ to decrease the computational burden: that is, $max \cdot M_i^2$ elements.

Thus the proposed algorithm needs to store $max \cdot (M_i + M_i^2)$ elements. Hence it can verified from Table 2 in [20], which shows the memory needs for nine algorithms that can fuse delayed measurements in a multisensor network, that the measurement transportation is the method with the lowest memory needs among all those analyzed in [20].

The MMSE estimate could be achieved in a linear and Gaussian system if all the measurements fused with Kalman filtering were not delayed, which is not the case of the investigated scenarios. No claim of optimality is made regarding the use of either the reiterated Kalman filtering or the measurement transportation approach in the investigated scenarios where error dynamics are linearized, noise sequences are non-Gaussian, and delayed measurements transit throughout the network. However, since the equations have been derived based on linear MMSE estimation, it can be claimed that the measurement transportation approach is an unbiased estimator for the fusion of delayed measurements [28].

5 UAV Fleet Problem Formulation

If a fleet of UAVs is modeled as nodes of a distributed sensor network with links to exchange information, then it has been indicated by eq. 4 and its further development that GPS measurements from one UAV can be used by another UAV if their relative positions are available measurements as well. This information sharing can be used to increase the robustness of the fleet formation flight, e.g. if a UAV loses GPS signal lock, then INS solution errors can be limited if the neighboring nodes' GPS measurements are correctly fused. However, it is likely that these network data will arrive with varying, possibly high delays across the network nodes. Thus, the scenario motivates the use of the algorithms in the last section to properly fuse the delayed information in transit throughout the network.

Omitting model noise, the continuous-time INS error model dynamics for the i-th UAV is described as follows [31, 32]

$$
\begin{aligned}
\dot{\mathbf{x}}_i(t) &= \mathbf{A}_i(t) \cdot \mathbf{x}_i(t) \\
\mathbf{A}_i &= \left[\begin{array}{c|c|c|c|c}
[\boldsymbol{\rho}_{l,i}]_\times & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\
\mathbf{g}_{e,i} & \boldsymbol{\alpha}_i & \boldsymbol{\Gamma}_i & \mathbf{D}^b_{l,i} & \mathbf{0}_{3\times3} \\
\mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \boldsymbol{\beta}_i & \mathbf{0}_{3\times3} & -\mathbf{D}^b_{l,i} \\
\mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\
\mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3}
\end{array}\right] \\
\mathbf{x}_i &= \left[\, \Delta\mathbf{R}^T_{l,i} \; \Delta\mathbf{V}^T_{l,i} \; \boldsymbol{\psi}^T_i \; \boldsymbol{\nabla}^T_{b,i} \; \boldsymbol{\epsilon}^T_{b,i} \,\right]^T \;,
\end{aligned}
\tag{33}
$$

where

- $\boldsymbol{\alpha}_i = [\boldsymbol{\rho}_{l,i} + 2\boldsymbol{\Omega}_{e,l,i}]_\times$,
- $\boldsymbol{\beta}_i = [\boldsymbol{\rho}_{l,i} + \boldsymbol{\Omega}_{e,l,i}]_\times$,
- $\boldsymbol{\Gamma}_i = [\mathbf{A}\mathbf{s}\mathbf{p}_{l,i}]_\times$,
- $\mathbf{g}_{e,i} = diag(-g_{e,i}/R_{e,i} \quad -g_{e,i}/R_{e,i} \quad 2g_{e,i}/R_{e,i})$,

and $diag(\cdot)$ is a diagonal matrix. Thus the discrete form of this model can be written as

$$\mathbf{x}_{k+1,i} = \mathbf{F}_{k,i}\mathbf{x}_{k,i} + \mathbf{u}_{k,i} + \mathbf{G}_{k,i}\mathbf{w}_{k,i} \ , \tag{34}$$

where $\mathbf{x}_{k,i} = \mathbf{x}_i(t_k)$, $\mathbf{F}_{k,i} = e^{\mathbf{A}(t_{k-1})\cdot\Delta}$, Δ is the sample step, $\mathbf{G}_{k,i}\mathbf{w}_{k,i}$ is the model noise as described in eq. 1, and $\mathbf{u}_{k,i}$ is a virtual control vector used to remove the mean of $\mathbf{x}_{k,i}$ when the Kalman filter estimates are fed back to correct the INS.

The GPS measurement is assumed to directly provide UAV position and velocity in the WGS-84 ellipsoid Earth-fixed coordinate frame. Thus, these data are compared to the INS solution to produce a measurement vector of the state-error. Receiver clock errors have not been involved in this investigation. Under these considerations, the discrete GPS measurement equation for the INS error model is

$$\begin{aligned} \mathbf{y}_{k,i}^{GPS} &= \begin{bmatrix} \mathbf{D}_{l,k,i}^{e}(\mathbf{p}_{k,e,i}^{GPS} - \mathbf{p}_{k,e,i}^{INS}) \\ \mathbf{D}_{l,k,i}^{e}(\mathbf{v}_{k,e,i}^{GPS} - \mathbf{v}_{k,e,i}^{INS}) \end{bmatrix} \\ \mathbf{y}_{k,i}^{GPS} &= \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \mathbf{x}_{k,i} + \begin{bmatrix} \mathbf{D}_{l,k,i}^{e} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{D}_{l,k,i}^{e} \end{bmatrix} \mathbf{v}_{k,i}^{GPS} \ , \end{aligned} \tag{35}$$

where $\mathbf{p}_{k,e,i}^{GPS}$ and $\mathbf{v}_{k,e,i}^{GPS}$ are, respectively, the position and velocity of the vehicle given by the GPS receiver on board and represented in the Earth-Centered-Earth-Fixed coordinate frame; $\mathbf{p}_{k,e,i}^{INS}$ and $\mathbf{v}_{k,e,i}^{INS}$ are, respectively, the position and velocity of the vehicle given by the INS and represented in the Earth-Centered-Earth-Fixed coordinate frame; and $\mathbf{v}_{k,i}^{GPS}$ is assumed to be a white Gaussian noise sequence with covariance $\mathbf{R}_{k,i}^{GPS}$. The DCM $\mathbf{D}_{l,k,i}^{e}$ can be computed using either the GPS data or the INS solution. One should notice that the computation of $\mathbf{D}_{l,k,i}^{e}$ is approximated, since navigation errors have been neglected.

If the i-th UAV receives the GPS position from the j-th UAV ($\mathbf{p}_{k,e,j}^{GPS}$), then the former can use this information if the measurement of the relative position between the two UAVs ($\mathbf{p}_{k,e,j\to i}$) is available as follows

$$\mathbf{p}_{k,e,j}^{GPS} + \mathbf{p}_{k,e,j\to i} = \mathbf{p}_{k,e,j}^{GPS,i} \ , \tag{36}$$

where $\mathbf{p}_{k,e,j}^{GPS,i}$ is a position measurement of the i-th UAV using the GPS measurement data from j-th UAV and the measurement of the relative position between the UAVs. The latter cannot be constructed using the GPS information from both UAVs, since then no additional information would be available. The relative position can be obtained, for example, from an imaging pod and proper image processing, or by a RF range measurement as studied in [33]. Finally,

the state-error measurement using the information from the j-th UAV can be constructed as follows

$$\begin{aligned} \mathbf{y}_{k,j}^{GPS,i} &= \mathbf{D}_{l,k,i}^{e}(\mathbf{p}_{k,e,j}^{GPS,i} - \mathbf{p}_{k,e,i}^{INS}) \\ \mathbf{y}_{k,j}^{GPS,i} &= \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \mathbf{x}_{k,i} + \mathbf{D}_{l,k,i}^{e} \mathbf{v}_{k,j}^{GPS,i} \ , \end{aligned} \tag{37}$$

where $\mathbf{v}_{k,j}^{GPS,i}$ is assumed to be a white Gaussian noise sequence with $\mathbf{R}_{k,j}^{GPS,i}$ covariance matrix.

If the measurement from the j-th UAV arrives with delay, then the algorithms showed previously can be used to correctly fuse it into the i-th UAV Kalman filter. One should notice that the relative positions and the GPS data does not need to be transmitted at the same time. However, the UAVs must store the relative positions received from instant $k - max$ up to instant k to properly convert the neighboring nodes' GPS measurements as described in eq. 36 when needed.

6 Simulations and Results

The simulations have been carried out using a swarm of 5 UAVs. The INS solution was given by the algorithm in [34]. Additionally, a magnetometer as described in [35] has been added to each UAV to limit the misalignment due to the low-quality inertial sensors. The communication links are shown in fig. 1, the simulations parameters are presented in Table 1, and the UAVs velocities and angular rates are described in the Appendix.

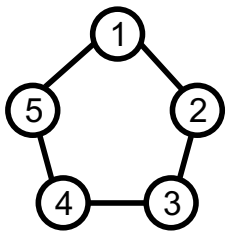


Fig. 1. UAV fleet communication links

The UAVs do not share any information before $t = 151$ s. Additionally, all the measurements reach the neighboring nodes at a fixed 60 s (6,000 sampling steps) after measurement transmission, which is the maximum allowed delay. This is assumed to be the worst scenario possible, and provides a tough test for the algorithms. Furthermore, UAV 5 loses its GPS lock after $t = 190$ s, and thus the embarked magnetometer data and the delayed measurements vectors from neighboring UAVs 1 and 4 are the only information available to UAV 5. The RMS error of each component of UAV 5's state vector has been used to compare the performance of both the reiterated Kalman filter and the measurement transportation approach.

Table 1. Simulation parameters

Sensors	
$\boldsymbol{\nabla}$	$\begin{bmatrix} 3 & 3 & 3 \end{bmatrix}^T$ mg
$\boldsymbol{\epsilon}$	$\begin{bmatrix} 1000 & 1000 & 1000 \end{bmatrix}^T$ $\circ$/h
Accelerometers covariance ($\mathbf{R}_{\boldsymbol{\nabla}}$)	$diag\begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$ $(mg)^2$
Rate-gyros covariance ($\mathbf{R}_{\boldsymbol{\epsilon}}$)	$diag\begin{pmatrix} 500 & 500 & 500 \end{pmatrix}$ $(\circ/\mathrm{h})^2$
$\mathbf{R}_{GPS}$	$diag\begin{pmatrix} 81 & 81 & 81 & 0.1 & 0.1 & 0.1 \end{pmatrix}$ SI units^2
$\mathbf{R}_{magnetometer}$	$diag\begin{pmatrix} (2\cdot 10^{-5})^2 & (2\cdot 10^{-5})^2 & (2\cdot 10^{-5})^2 \end{pmatrix}$ Gauss^2
Covariance of relative position measurement	$5 \cdot diag\begin{pmatrix} 81 & 81 & 81 \end{pmatrix}$ m^2
GPS and magnetometer data frequency	1 Hz
INS	
Initial position	Latitude: $(23^\circ 12' \ S + 0.05 \cdot \mathcal{G}_1)$, Longitude: $(45^\circ 52' \ W + 0.05 \cdot \mathcal{G}_2)$, where $\mathcal{G}_1$ and $\mathcal{G}_2$ are zero-mean Gaussian variables with standard deviation of 1".
Initial altitude	700 m $+ \mathcal{H}$ where $\mathcal{H}$ is a zero-mean Gaussian variable with standard deviation of 1 m.
Initial velocity	$\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ m/s
Initial alignment	TRIAD algorithm [36]
INS solution sampling rate (t_{ins})	0.01 s
Kalman filter	
Feedback start	95 s
$\mathbf{Q}$, $t < 95$ s	$1/50 \cdot t_{ins} \cdot \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{R}_{\boldsymbol{\nabla}} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{R}_{\boldsymbol{\epsilon}} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{bmatrix}^T$ SI Units^2
$\mathbf{Q}$, $t \geq 95$ s	$1/150 \cdot t_{ins} \cdot \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{R}_{\boldsymbol{\nabla}} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{R}_{\boldsymbol{\epsilon}} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{bmatrix}^T$ SI Units^2
Initial covariance	$diag(50^2 \ \ 50^2 \ \ 50^2 \ \ 2^2 \ \ 2^2 \ \ 2^2 \ \ 0.05 \ \ 0.05 \ \ 0.05 \ \ 0.09 \ \ 0.09 \ \ 0.09 \ \ 0.015 \ \ 0.015 \ \ 0.015)$ SI Units^2
Initial estimate	$\mathbf{0}_{15\times 1}$ SI units

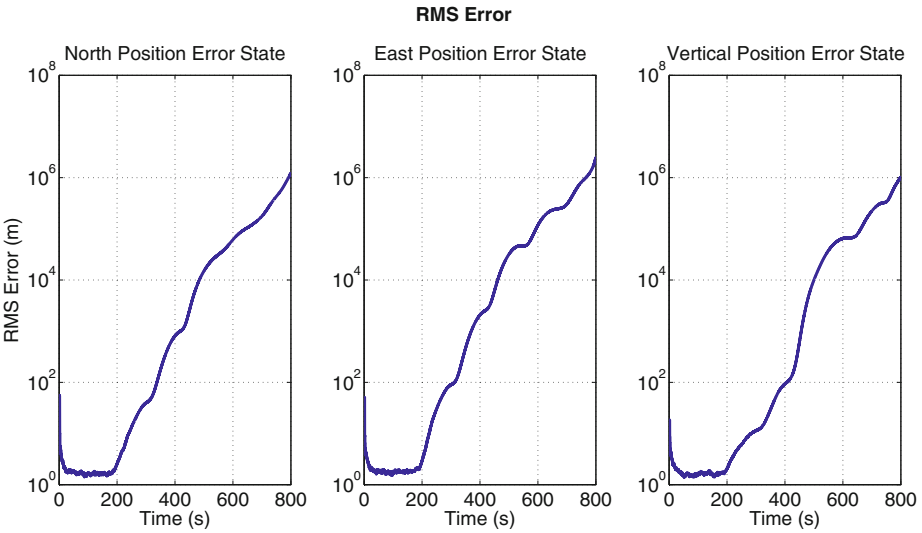


Fig. 2. Scenario 01: Delayed measurements naïvely fused - RMS errors of the position error components from 100 realizations

In the first scenario, all the measurements were fused neglecting the measurement delay. The RMS errors of the position error components computed via a Monte Carlo simulation with 100 realizations are presented in fig. 2. The estimate diverges after the local GPS fault when the delayed measurements are naïvely fused.

In the second scenario, the measurements were fused using the algorithms presented previously. The RMS errors of the state vector components computed via a Monte Carlo simulation with 100 realizations are presented in figs. 3 to 7. The computational load of the reiterated Kalman filter was 3.42 times that of the measurement transportation approach.

6.1 Results Analysis

Firstly, fig. 2 clearly shows that the fusion of delayed measurements without adequate processing yields estimation divergence. On the other hand, the results obtained with scenario 02 and displayed in figs. 3 to 7 show that the algorithms described here correctly fuse the delayed measurements and provide limited navigation errors.

The fusion of neighboring nodes' measurements received by UAV 5 when its GPS observables were available did not provide any noticeable improvement in estimation accuracy ($151s < t < 190s$) because the simulated scenario had the relative position measurement covariance with much larger eigenvalues than those of the GPS measurement covariance. On the other hand, adequate processing of delayed network data from neighboring nodes successfully eliminated estimation divergence when the GPS signal was denied to UAV 5.

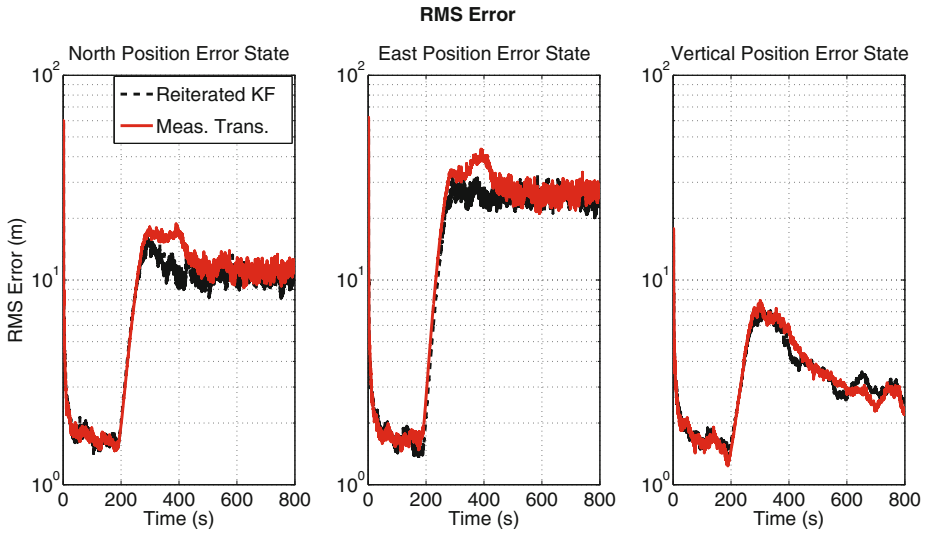


Fig. 3. Scenario 02: Delayed measurements correctly fused - RMS errors of the position error components from 100 realizations

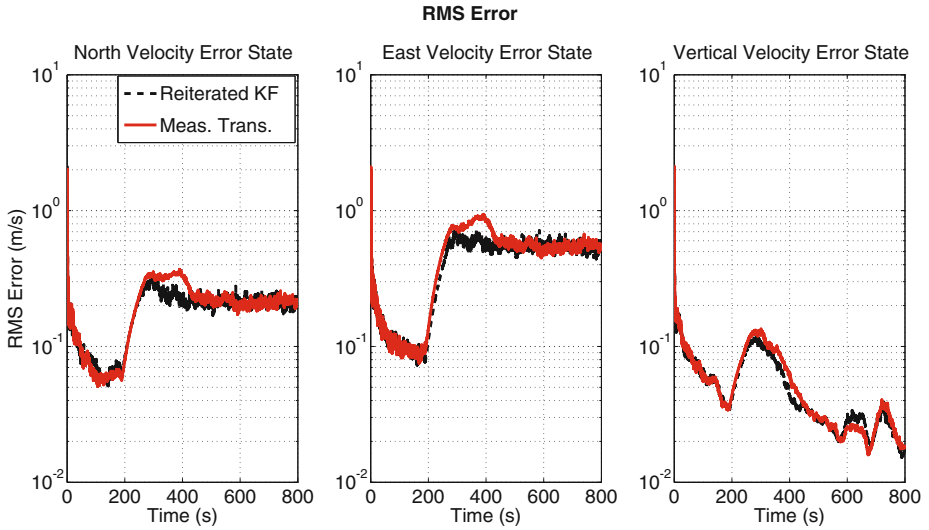


Fig. 4. Scenario 02: Delayed measurements correctly fused - RMS errors of the velocity error components from 100 realizations

Figures 3 to 7 also show that the Kalman filter reiteration, which is optimal in the linear Gaussian case, most times achieved better performance than that of the measurement transportation approach. The cost-benefit ratio of the latter is far more attractive, however, due to its reduced computational workload and statistically similar estimation performance.

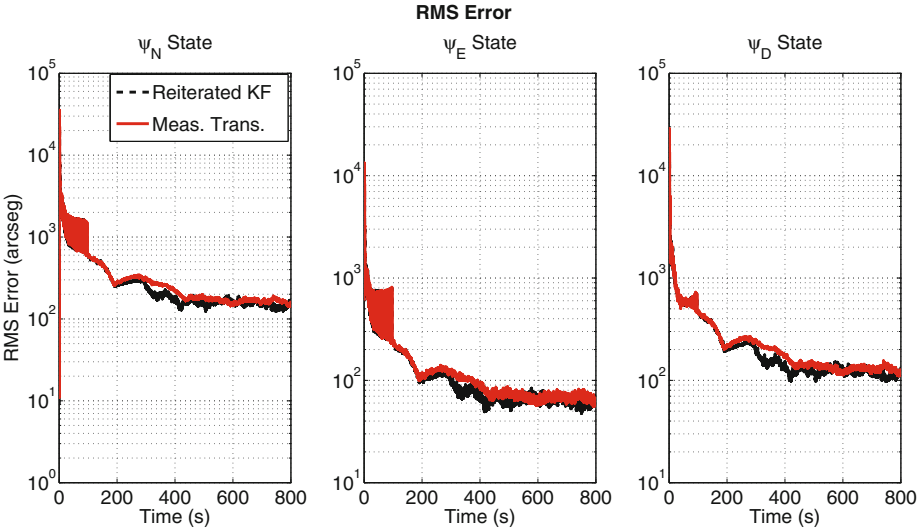


Fig. 5. Scenario 02: Delayed measurements correctly fused - RMS errors of the misalignment error components from 100 realizations

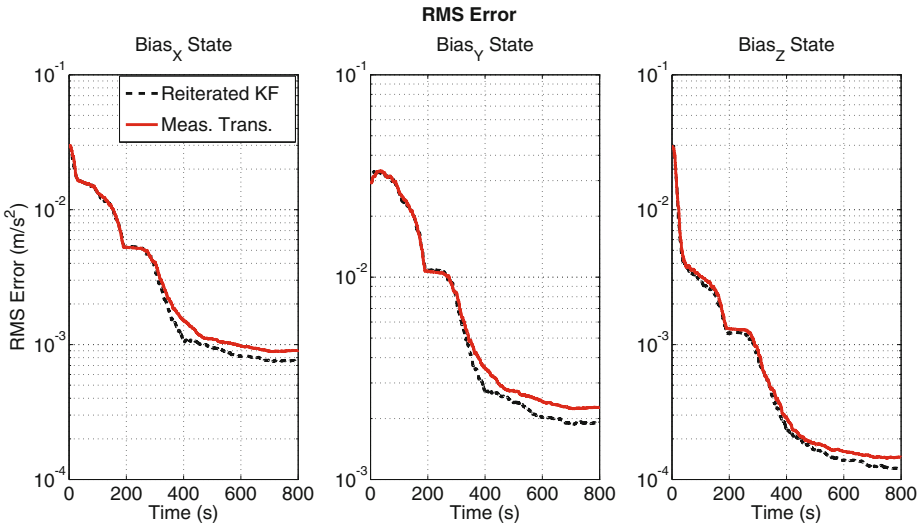


Fig. 6. Scenario 02: Delayed measurements correctly fused - RMS errors of the accelerometer bias components from 100 realizations

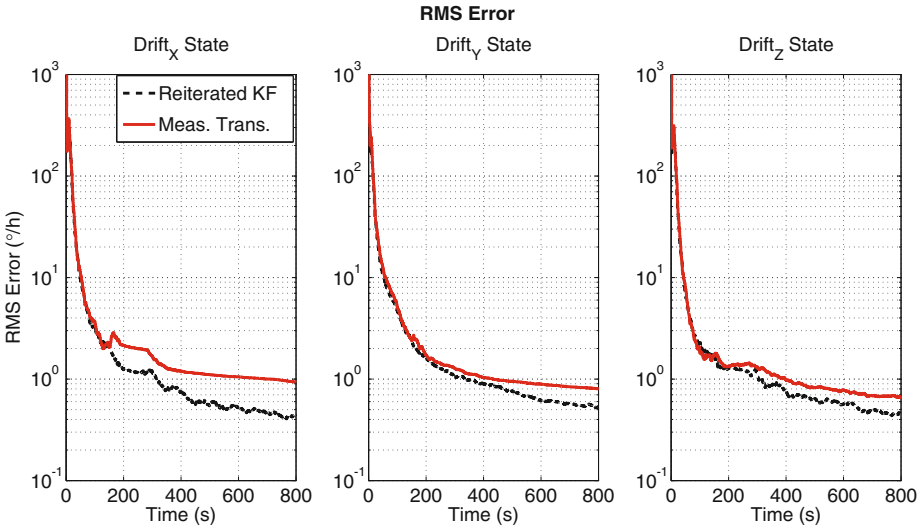


Fig. 7. Scenario 02: Delayed measurements correctly fused - RMS errors of the rate-gryo drift components from 100 realizations

7 Conclusions

This paper presented a novel suboptimal approach, called measurement transportation, to fuse delayed measurements in distributed sensor networks. This new technique has less memory needs than the usual algorithms investigated in [20] and a good overall performance is achieved for the problems of interest.

The novel algorithm was compared with the classical approach to fuse delayed measurements in distributed sensor networks: the reiterated Kalman filter. The comparison was carried out using a simulated distributed sensor network that consists of a UAV fleet in formation flight in which the GPS measurements and relative positions are exchanged among neighboring network nodes.

The results shows that both algorithms could correctly fuse the delayed measurements in the proposed scenario and produced similar estimation accuracies. The measurement transportation approach demands a much lower computational load and requires less memory.

One must notice that the distributed estimation problem tackled here is neither Gaussian nor linear. Thus, one should expect that other estimators may yield improved accuracy with respect to that of the reiterated Kalman filter in the scenario simulated here. For example, enhanced accuracy is expected if the INS algorithm is reiterated together with the Kalman filter and the linearization of the error dynamics model about the reiterated INS solution is also carried out at every step. However, the computational load of such estimator algorithm has shown to be prohibitive even to computationally resourceful desktop PCs.

Acknowledgments. The authors acknowledge the generous support granted by project FINEP/CTA/INPE SIA (Inertial Systems for Aerospace Application,

in Portuguese), embodied through the logistical support, acquisition of the computational resources needed for the research reported here, and the scholarship awarded to the first author.

Appendix: UAVs Trajectory and Angular Movement

The UAV trajectory is composed of several segments with a distinct, constant specific force during each one. They are described in Table 2 in which $\mathcal{A}_1$ and $\mathcal{A}_2$ are uniformly distributed random variables on the interval $[-3, 3]$ m/s^2 that have been sampled at the beginning of each realization for each UAV.

The IMU attitude evolves in terms of the Euler angles that rotate the local coordinate frame into alignment with the body coordinate frame (yaw, pitch, and roll rotation sequence) as follows

$$\begin{aligned}
\psi &= 0.1 \sin\left(2\pi \frac{t}{300}\right) + 0.05 \sin\left(2\pi \frac{t}{1.7}\right) + 0.2 \text{ rad} \\
\theta &= 0.1 \sin\left(2\pi \frac{t}{300}\right) + 0.05 \sin\left(2\pi \frac{t}{1.7}\right) - 0.4 \text{ rad} \\
\phi &= 0.1 \sin\left(2\pi \frac{t}{300}\right) + 0.05 \sin\left(2\pi \frac{t}{0.85}\right) + 0.5 \text{ rad} .
\end{aligned} \tag{38}$$

One should notice that this trajectory and angular movement yield a fully observable system [35, 37, 38].

Table 2. UAVs trajectory

		Specific forces		
Start (s)	**End** (s)	**N** (m/s^2)	**E** (m/s^2)	**D** (m/s^2)
0	30	0	0	-g
30	70	$\mathcal{A}_1$	0	-g
70	110	0	$\mathcal{A}_1$	-g
110	150	$\mathcal{A}_1$	$\mathcal{A}_1$	-g
150	190	0	0	-g-$\mathcal{A}_1$
190	240	0	0	-g
240	280	-$\mathcal{A}_1$	0	-g
280	320	0	-$\mathcal{A}_1$	-g
320	360	0	$\mathcal{A}_2$	-g
360	500	0	0	-g+$\mathcal{A}_2$
500	520	0	$\mathcal{A}_2$	-g
520	540	-$\mathcal{A}_2$	0	-g
540	560	-$\mathcal{A}_2$	$\mathcal{A}_2$	-g
560	600	0	-$\mathcal{A}_2$	-g
600	660	0	0	-g-$\mathcal{A}_2$
660	720	0	$\mathcal{A}_2$	-g
720	800	-$\mathcal{A}_2$	0	-g

References

1. Olfati-Saber, R.: Distributed Kalman Filtering for Sensor Networks. In: 46th IEEE Conference on Decision and Control, pp. 5492–5498. IEEE Press, New York (2007)
2. Cattivelli, F.S., Lopes, C.G., Sayed, A.H.: Diffusions Strategies for Distributed Kalman Filtering: Formulation and Performance Analysis. In: 2008 IARP Workshop on Cognitive Information Processing. IAPR Press, New York (2008)
3. Carlson, N.A.: Federated Filter for Fault-Tolerant Integrated Navigation Systems. In: 1998 IEEE Position, Location and Navigation Symposium, pp. 110–119. IEEE Press, New York (1998)
4. Felter, S.C.: An Overview of Decentralized Kalman Filter Techniques. In: 1990 IEEE Southern Tier Technical Conference, pp. 79–87. IEEE Press, New York (1990)
5. Qui, H.Z., Zhang, H.Y., Jin, H.: Fusion Algorithm of Correlated Local Estimates. Aerospace Science and Technology 8, 619–629 (2004)
6. Borreli, F., Keviczky, T., Balas, J.G.: Collision-free UAV Formation Flight using Decentralized Optimization and Invariant Sets. In: 43th IEEE Conference on Decision and Control, vol. 1, pp. 1099–1104. IEEE Press, New York (2004)
7. Fergunson, P., Yang, T., Tillerson, M., How, J.: New Formation Flying Testbed for Analyzing Distributed Estimation and Control Architectures. In: 2002 AIAA Guidance, Navigation and Control Conference. AIAA Press, Reston (2002)
8. Fergunson, P., How, J.: Decentralized Estimation Algorithms for Formation Flying Spacecraft. In: 2003 AIAA Guidance, Navigation and Control Conference. AIAA Press, Reston (2003)
9. Smith, R.S., Hadaegh, F.Y.: Parallel Estimation and Control Architectures for Deep-space Formation Flying Spacecraft. In: 2006 IEEE Aerospace Conference, pp. 1–10. IEEE Press, New York (2006)
10. Smith, R.S., Hadaegh, F.Y.: A Distributed Parallel Estimation Architecture for Cooperative Vehicle Formation Control. In: 2006 IEEE Aerospace Conference. IEEE Press, New York (2006)
11. Azizi, S.M., Khorasani, K.: A Distributed Kalman Filter for Actuator Fault Estimation of Deep Space Formation Flying Satellites. In: 3rd Annual IEEE International Systems Conference, pp. 354–359. IEEE Press, New York (2009)
12. Alouani, A.T.: Linear Distributed Estimation. In: 1987 IEEE American Control Conference, pp. 412–417. IEEE Press, New York (1987)
13. Berg, T.M., Durrant-Whyte, H.F.: General Decentralized Kalman Filters. In: 1994 IEEE American Control Conference, vol. 2, pp. 2273–2274. IEEE Press, New York (1994)
14. Tasoulis, D.K., Adams, N.M., Hand, D.J.: Selective Fusion of Out-of-Sequence Measurements. Information Fusion 11(2), 183–191 (2010)
15. Matveev, A., Savkin, A.: The Problem of State Estimation via Asynchronous Communication Channels with Irregular Transmission Times. IEEE Transactions on Automatic Control 48(4), 670–676 (2003)
16. Zhang, K., Li, X.R., Zhu, Y.: Optimal Update with Out-of-Sequence Measurements. IEEE Transactions on Signal Processing 56(6), 1992–2004 (2005)
17. Lu, X., Zhang, H., Wang, W., Teo, K.: Kalman Filtering for Multiple Time-Delay Systems. Automatica 41(86), 1455–1461 (2005)

18. Alexander, H.L.: State Estimation for Distributed Systems with Sensing Delay. In: Libby, V. (ed.) Data Structures and Target Classification, vol. 1470, pp. 103–111. SPIE (1991)
19. Larsen, T.D., Andersen, N.A., Ravn, O., Poulsen, N.K.: Incorporation of Time Delayed Measurements in a Discrete-time Kalman Filter. In: 37th IEEE Conference on Decision and Control, pp. 3972–3977. IEEE Press, New York (1998)
20. Besada-Portas, E., Lopez-Orozco, J.A., Besada, J.A., de la Cruz, J.M.: Multisensor Out of Sequence Data Fusion for Estimating the State of Discrete Control Systems. IEEE Transactions on Automatic Control 54(7), 1728–1732 (2009)
21. Brown, R.G., Hartman, G.L.: Kalman Filter with Delayed States as Observables. In: 1968 National Electronics Conference. National Electronics Conference, Oak Brook (1968)
22. Bar-Shalom, Y., Mallick, M., Chen, H., Washburn, R.: One-step Solution for the General Out-of-Sequence Measurement Problem in Tracking. In: 2002 IEEE Aerospace Conference, vol. 4, pp. 1551–1559. IEEE Press, New York (2002)
23. Gopolakrishnan, A., Kaisare, N.S., Narasimhan, S.: Incorporating Delayed and Infrequent Measurements in Extended Kalman Filter based Nonlinear State Estimation. Journal of Process Control 21, 119–129 (2011)
24. Leung, K.Y.K., Barfoot, T.D., Liu, H.H.T.: Decentralized Localization of Sparsely-Communicating Robot Networks: A Cetralized-Equivalent Approach. IEEE Transactions on Robotics 26(1), 62–77 (2010)
25. Thomopoulos, S., Zhang, L.: Decentralized Filtering with Random Sampling and Delay. Information Fusion 81(1-2), 117–131 (1994)
26. Maybeck, P.S.: Stochastic Models, Estimation and Control, vol. 1. Academic Press, New York (1979)
27. Chagas, R.A.J.: Distributed Estimation of Aided INS Errors. Ph.D. thesis. Instituto Tecnológico de Aeronáutica. SP, Brazil (2012) (in Portuguese)
28. Anderson, B.D.O., Moore, J.B.: Optimal Filtering. Prentice-Hall, Inc., Englewood Cliffs (1979)
29. Brown, R.G., Hwang, P.Y.C.: Introduction to Random Signals and Applied Kalman Filtering. John Wiley & Sons, Inc., Hoboken (1997)
30. VanValkenburgh, M.: The Matrix Exponential Function, `http://math.berkeley.edu/~mjv/H54Lec22.pdf`
31. Weinred, A., Bar-Itzhack, I.Y.: The Psi-Angle Error Equation in Strapdown Inertial Navigation Systems. IEEE Transactions on Aerospace and Electronic Systems 14(3), 539–542 (1979)
32. Lee, J., Park, C.G., Park, H.W.: Multiposition Alignment of Strapdown Inertial Navigation System. IEEE Transactions on Aerospace and Electronic Systems 29(4), 1323–1328 (1993)
33. Lapid-Maoz, J., Bar-Itzhack, I.Y.: Relative-Location Determination of Cooperating Aircraft. In: 2000 AIAA Guidance, Navigation and Control Conference. AIAA Press, Reston (2000)
34. Salychev, O.: Applied Inertial Navigation: Problems and Solutions. BMSTU Press, Moscow (2004)
35. Chagas, R.A.J., Waldmann, J.: Geometric Inference-Based Observability Analysis Digest of INS Error Model with GPS/Magnetometer/Camera Aiding. In: 19th Saint Petersburg International Conference on Integrated Navigation Systems. CSRI Elektropribor, JSC, Saint Petersburg, Russia (2012)
36. Shuster, M., Oh, S.: Three-Axis Attitude Determination from Vector Observations. Journal of Guidance, Control, and Dynamics 4(1), 70–77 (1981)

37. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Observability Analysis of Piece-Wise Constant Systems - Part II: Application to Inertial Navigation In-Flight Alignment. IEEE Transactions on Aerospace and Electronic Systems 28(4), 1068–1075 (1992)
38. Chung, D., Park, C.G., Lee, J.G.: Observability Analysis of Strapdown Inertial Navigation System using Lyapunov Transformation. In: 35th IEEE Conference on Decision and Control, pp. 23–28. IEEE Press, New York (1995)

Higher Order Sliding Mode Controllers with Adaptation

Daniel Yitzjak Negrete-Chavez and Leonid M. Fridman

National Autonomous University of Mexico
Mexico City C.P. 58089, Mexico
danielnegrete@gmail.com,
fridman@servidor.unam.mx

Abstract. Three different second order and one fourth order sliding mode controllers are considered in this chapter. An adaptive gain is implemented which adjusts the level of scalar control action on-line based on direct measurements of the equivalent control obtained by a low-pass filter. It is shown that the adaptive algorithm converge in finite time, thus the chattering is reduced in amplitude and the amount of energy demanded by the controller is reduced too. The results of a real implementation of the adaptive sliding mode controllers in a spring-mass-damper system are presented. Also a comparison between the adaptation methodology and augmented order methodology is presented.

Keywords: Higher Order Sliding Mode Controllers, Adaptive Gain, Variable Structure.

1 Introduction

The concept of adaptation in the context of control engineering refers to the variation of certain parameters with respect to a certain signal based on utilization of current information. It involves modifying the control law used by the controller in order to cope with the fact the parameters of the system being controlled are uncertain, or, to improve the performance of the controller and its effectiveness exhibiting the same dynamics properties under uncertainty conditions. Even more, adaptive control implies improving dynamic characteristics while properties of a controlled plant or environment are varying [3, 4].

1.1 Motivation

The main obstacle of *Sliding mode Control* (SMC) application is the *chattering* which is an oscillatory phenomenon inherent in sliding motions (see, for example, [5–7]). The chattering phenomenon is caused due to the high frequency switching nature of the controller. The phenomenon is well-known from literature on power converters and referred as "*ripple*" [8].

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_9

The amplitude of the chattering is proportional to the gain of the controller. In classical SMC the gain of the control should be a constant value greater than the bound of the uncertainty/perturbation. Thus the controller demands a constant amount of energy that may not be needed to maintain the system in sliding mode. Also the chattering presents a constant amplitude that can be harmful for the actuator and the plant.

1.2 Objective

The objective of this paper is to implement an adaptive gain with higher order SMCs, in order to reduce both the chattering amplitude and the amount of energy demanded by the controller.

2 Main Result

2.1 System Description

Consider the following system

$$\dot{z}(t) = h(z(t)) + g(z(t))u_t(z(t)), \tag{1}$$

where

$$\begin{aligned} z &\in \mathbb{R}^n \\ h &: \mathbb{R}^n \to \mathbb{R}^n, \\ u_t &: \mathbb{R}^n \to \mathbb{R}, \\ g &: \mathbb{R}^n \to \mathbb{R}^n \end{aligned}$$

g, h, u_t are continuous functions.

For the system (1) the following assumptions hold

- **A1** A diffeomorphism $T(z(t))$ exist, such that the system (1) can be expressed as

$$\begin{aligned} \dot{x}_i &= x_{i+1} \\ \dot{x}_n &= f(t,x) + u(k(t),x) \\ i &= 1, 2..., n-1 \end{aligned} \tag{2}$$

where

$$\begin{aligned} f &: \mathbb{R}^+ \times \mathbb{R}^n \to \mathbb{R}, \\ u &: \mathbb{R}^n \to \mathbb{R}, \\ x &\in \mathbb{R}^n \end{aligned} \tag{3}$$

a, f, u_t are continuous functions.

– **A2** The control

$$u(k(t), x) = -k(t)w(x) \tag{4}$$

with adaptive gain

$$0 < k_{min} \leq k(t) \leq k_{max} \tag{5}$$

is a sliding mode controller. Therefore it enforce the system into some sliding surface $\sigma(x) = 0$ ($\sigma(x) \in C^1$) in finite time, where x is the state vector. The structure of the sliding surface depends on the design of the controller.

The upper bound of $k(t)$ is related with the maximum power of the actuator. The lower bound is chosen in order to maintain the actuator active with a minimum value of the gain.

– **A3** For the unknown, smooth and bounded function $f(t, x)$ the following conditions hold:

$$|f(t, x)| < A < k_{max} \tag{6}$$

$$\left| \frac{d|f(t, x)|}{dt} \right| < L \tag{7}$$

– **A4** The equivalent control $u_{eq}(t, x)$ is available and can be obtained by filtering the high frequency component of the discontinuos function $u(k(t), x)$ by means of the low-pass filter

$$\tau \dot{u}_{eq}(t, x) + u_{eq}(t, x) = u(k(t), x), \qquad u_{eq}(0) = 0 \tag{8}$$

with a small constant $\tau > 0$ and the function $u_{eq}(t, x)$ as an output.

Remark 1. The objective of sliding mode controllers is to design a control law such that the constraint $\sigma(x) = 0$ holds, where x is the state vector and $\sigma(x)$ is called the sliding surface. Some choices of this constraint are discussed in [10] [11] [12]. In this paper, it is assumed that the sliding surface has been previously design. It is assumed in (A2) that the controller has been design to enforce the trajectories of the system into the sliding surface.

2.2 Description of the Adaptive Algorithm

In [9] an adaptive methodology is presented for the super-twisting algorithm. A similar methodology is presented here for second order and higher order sliding mode controllers.

Consider the system (2). The adaptation law for the gain of control (4) is described as

$$\dot{k}(t) = \begin{cases} \gamma k_{max} + M(k(t)), & if \quad \sigma(x) \neq 0 \\ \gamma k(t) sign(\delta) + M(k(t)), & if \quad \sigma(x) = 0 \end{cases}$$

$$M(k(t)) = \begin{cases} -\gamma\rho k(t), & if \;\; k(t) > k_{max} \\ \gamma\rho k(t), & if \;\; k(t) < k_{min} \\ 0, & if \;\; k_{min} \leq k(t) \leq k_{max} \end{cases} \tag{9}$$

$$\delta(t,k) = \frac{|u_{eq}(t, x_i, x_n)|}{k(t)} - \alpha \tag{10}$$

$\rho > 1$, $\gamma > 0$ and $\alpha \in (0, 1)$ is the desired proportion between the magnitude of the unknown function $f(t, x)$ and the gain $k(t)$. The function $M(k(t))$ is needed to ensure that the gain $k(t)$ remains bounded.

The idea of the algorithm is to increase the gain $k(t)$, during the reaching phase, i.e. when $\sigma(x) \neq 0$. Once the sliding mode is established, i.e. $\sigma(x) = 0$, the dynamics of the gain depends on the proportion α between the unknown function $f(t, x)$ and the gain $k(t)$. The gain decreases if $|u_{eq}/k(t)| < \alpha$ and increases if $|u_{eq}/k(t)| > \alpha$ until $|u_{eq}/k(t)| = \alpha \rightarrow \delta(t, k) = 0$.

The equivalent control can be thought as an equivalent signal of the perturbation, thus when $\delta(t, k) = 0$ the proportion between the unknown function $f(t, x)$ and the gain $k(t)$ is α.

Remark 2. Sliding mode controllers, theoretically, has to switch infinitely to reach exactly the constraint $\sigma(x) = 0$. Practically, it is impossible to achieve an infinite switching, thus the sliding mode is detected when $\sigma(x)$ switches about zero with $|\sigma(x)| < \epsilon$ for a sufficiently small constant $\epsilon > 0$. Moreover ϵ is the amplitude of the chattering phenomenon. The objective of adapting the gain is to reduce ϵ.

Remark 3. The adaptation law is based on the measurement of the equivalent control that is obtained by filtering out the control signal when the sliding mode is established. In fact it is an estimation of the control signal. This estimation is good enough for the adaptation law to work. The disadvantage of using a filtered signal to adapt the gain is that there is a delay on the equivalent control signal that is discussed on the results section.

2.3 $\delta(t, k(t))$ Stability Proof

The adaptation process is over when the desired proportion between the magnitude of the perturbation and the magnitude of the gain is achieve, i.e. $\delta(t, k) = 0$.

Before the sliding mode is established, the adaptive gain will grow until it reaches a value which leads the system into sliding mode. For the following proof it is assumed that the system is already in sliding mode.

To show that the variable $\delta(t,k)$ converge to zero in finite time, when the system is in sliding mode, the following Lyapunov function is considered

$$V(\delta(t,k(t))) = \frac{\delta^2(t,k(t))}{2} \tag{11}$$

and its time derivative

$$\dot{V}(\delta(t,k(t))) = \delta(t,k(t))\dot{\delta}(t,k(t)) = \delta(t,k(t))\left(\frac{d|u_{eq}(t,x)|}{dt}\frac{1}{k} - \frac{\dot{k}}{k^2}|u_{eq}(t,x)|\right) \tag{12}$$

The function $u_{eq}(t,x)$ is an approximation of $f(t,x)$, as mentioned on remark 3. It can be stated that when the system is in sliding mode, $u_{eq}(t,x) = f(t,x)$. Therefore

$$\begin{aligned}
\dot{V}(\delta(t,k(t))) &= \delta(t,k(t))\left(\frac{d|f(t,x)|}{dt}\frac{1}{k(t)} - \frac{\gamma k(t) sign(\delta(t,k(t)))}{k^2(t)}|f(t,x)|\right) \\
&= \frac{\delta(t,k(t))}{k(t)}\left(\frac{d|f(t,x)|}{dt} - \gamma sign(\delta(t,k(t)))|f(t,x)|\right) \\
&= -\frac{|\delta(t,k(t))|}{k(t)}\left(\gamma|f(t,x)| - \frac{d|f(t,x)|}{dt}sign(\delta(t,k(t)))\right) \\
&\leq \frac{-|\delta(t,k(t))|}{k(t)}\left(\gamma|f(t,x)| - \frac{d|f(t,x)|}{dt}\right) \\
&< \frac{-|\delta(t,k(t))|}{k(t)}\left(\gamma|f(t,x)| - L\right)
\end{aligned}$$

The adaptation of the gain is only required when the magnitude of the perturbation is greater than αk_{min}, thus adaptation will exists when$|f(t,x)| > \alpha k_{min}$, therefore

$$\dot{V}(\delta(t,k(t))) < -\frac{|\delta(t,k(t))|}{k(t)}\left(\gamma\alpha k_{min} - L\right) \tag{13}$$

if

$$\gamma > \frac{L}{\alpha k_{min}} \tag{14}$$

$\delta(t,k(t))$ converge to zero and the proportion between the magnitude of the perturbation and the gain is α, i.e. $\frac{|f(t,x)|}{k(t)} = \alpha$. Also if (14) holds, then

$$\dot{V}(\delta(t,k(t))) \leq -\sqrt{2}\frac{\gamma\alpha k_{min} - L}{\alpha k_{min}}\sqrt{V(\delta(t,k(t)))} \tag{15}$$

$$0 \leq \sqrt{V(\delta(t,k(t)))} \leq \sqrt{V(\delta(0,k(t))} - \frac{\gamma\alpha k_{min} - L}{\sqrt{2}\alpha k_{min}}t \tag{16}$$

Is evident from the solution of the differential inequality (15) that $\sqrt{V(\delta(t,k(t)))} = 0$ at least after

$$t_f = \frac{k_{max}}{\gamma\alpha k_{min} - L}\sqrt{2V(\delta(0,k(t)))} = \frac{\alpha k_{min}}{\gamma\alpha k_{min} - L}|\delta(0,k(t))| \tag{17}$$

and, as a result, $\delta(t,k(t))$ is zero in time less or equal than t_f.

2.4 Higher Order Adaptive Sliding Mode Controller

Consider the arbitrary order system (2), where the controller is given by

$$u(k(t),x) = k(t)w(x). \tag{18}$$

Let the function $w(x)$ be a sliding mode control algorithm with arbitrary order n defined as the following recursive procedure called *nested n-sliding controller* [1]

$$\begin{aligned} w(x) &= \psi_{n-1,n}(x) \\ \psi_{0,n} &= sign(x_1) \\ \psi_{i,n} &= sign\left(x_i + \beta_i N_{i,n}\psi_{i-1,n}\right) \\ N_{i,n} &= \left(|x_1|^{p/n} + |x_2|^{p/(n-1)} + ... + |x_{i-1}|^{p/(n-i+1)}\right)^{\frac{n-i}{p}} \\ i &= 1,2,...,n-1 \end{aligned} \tag{19}$$

where p is the least common multiple of 1,2,...,n. The dynamics of the gain $k(t)$ are defined by (9).

It is has been proven in [1] that if values of gains $\beta_1, \beta_2, ..., \beta_i$ are properly chosen and are sufficiently large, then the system (2) converges to sliding mode.

The adaptive gain $k(t)$ will grow until it gets a value sufficiently large to ensure that the set of gains $\beta_1, \beta_2, ..., \beta_i$ and the adaptive gain $k(t)$ will be large enough to establish the sliding mode. Once it is established the gain $k(t)$ will be adapted by the law (9).

Adding the adaptive gain to the *nested algorithm* it is obtained a higher order sliding mode controller with adaptation. This controller with the adaptive mechanism is the main result of this paper. It can be applied to n order systems that can be or are expressed in the form (2). By adapting the gain the chattering amplitude, that is the main disadvantage of sliding mode controller, is decrease.

3 Experimental Results

In this section the results of the implementation of three sliding mode controllers with adaptive gain in a spring-mass-damper system are presented. It is shown in figure (1) the spring-mass-damper system where the controllers were implemented. The system consist on one spring one mass and one damper. The video of each controller implemented is available at the following address *www.negrete.webs.com.*

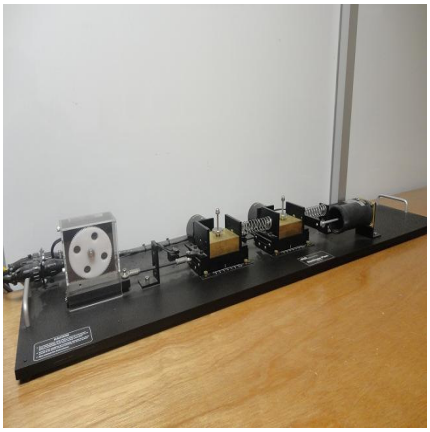


Fig. 1. Spring-mass-damper system

Consider the following second order system as a model of the spring-mass-damper system

$$\begin{aligned} \dot{x} &= y \\ \dot{y} &= a(t) + b(t)u_2(t, x, y) \\ u_2 &= -k(t)w(x, y), \qquad 0 < k_{min} \leq k \leq k_{max} \end{aligned} \tag{20}$$

where k_{min} and k_{max} are preselected minimum and maximum values of gain k, respectively. The functions $a(t)$ and $b(t)$ are unknown smooth bounded functions that include the parameters of the spring-mass-damper system which are assumed unknown. The state x represents the position of the mass measured in centimeters and the state y represents the derivative of the position of the mass measured in centimeter per hour. Suppose that assumptions A1, A2 and A3 holds.

3.1 Adaptive Twisting Control (ATWC)

Consider the system (20), where

$$w(x, y) = sign(x) + \beta sign(y), \beta \in (0, 1) \tag{21}$$

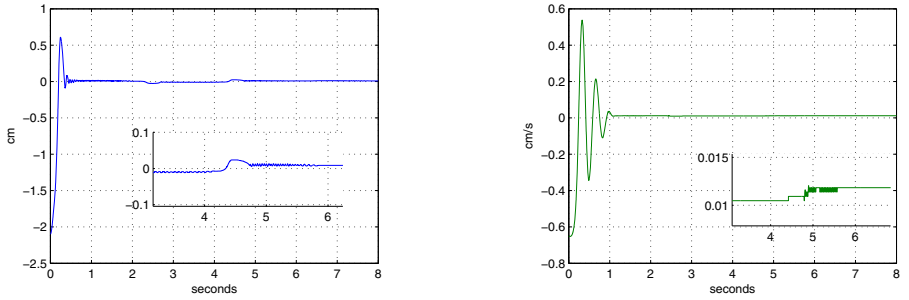
is a version of the so-called twisting algorithm.

The parameters used in the implementation are $\gamma = 4\pi$,$k_{min} = 1$,$k_{max} = 15$, $\tau = \sqrt{.001}$, $\beta = 0.5$, $\alpha = 0.45$.

3.2 Adaptive Terminal Control (ATEC)

Consider the system (20) where

$$w(x, y) = sign\left(y + \lambda |x|^{1/2} sign(x)\right), \lambda > 0 \tag{22}$$



(a) (b)

Fig. 2. a) Position of the mass (state x)-ATWC and b) Velocity of the mass (state y)-ATWC

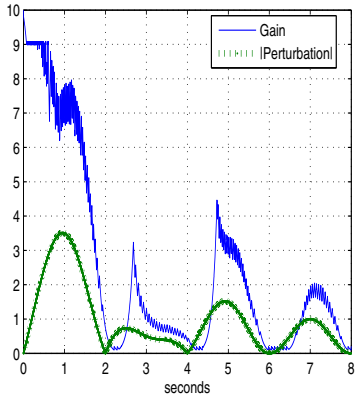


Fig. 3. Gain $k(t)$ and perturbation-ATWC

is a version of the so called terminal algorithm. The values of the implementation are $\alpha = 0.95$, $k_{min} = 1$, $k_{max} = 15$, $\lambda = 1$, $\tau = \sqrt{.001}$

3.3 Adaptive Sub-optimal Control (ASC)

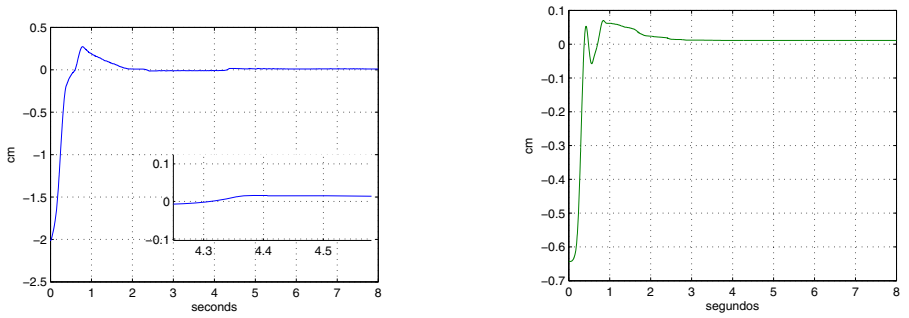
Consider the system (20) where

$$w(x, y) = \eta(t) sign(x - \beta x_m(t)) \tag{23}$$

were

$$\eta(t) = \begin{cases} 1, & if x_m(x - \eta x_m \geq 0 \\ \eta^*, & if x_m(t)(x - \eta x_m) < 0 \end{cases} \tag{24}$$

where $x_m(t)$ is a piece-wise function representing the value of the last singular point of x, i.e. the most recent value of x where $y = 0$.



(a) (b)

Fig. 4. a) Position of the mass (state x)-ATEC and b) Velocity of the mass (state y)-ATEC

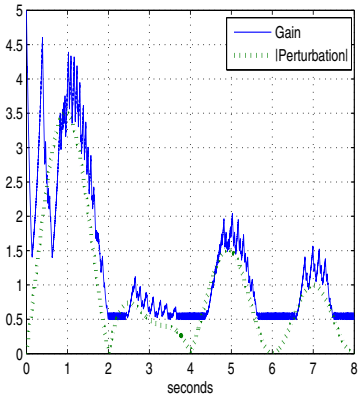


Fig. 5. Gain $k(t)$ and perturbation-ATEC

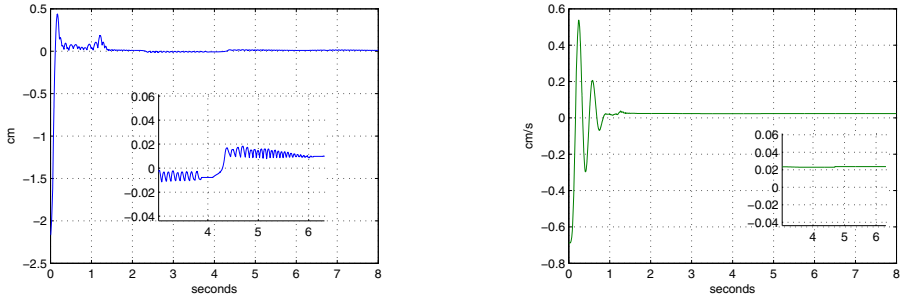
The algorithm was implemented with the following values of parameters $\eta^* = 3$, $k_{min} = 0.5$, $k_{max} = 8$, $\alpha = .9$ and $\gamma = \frac{5}{2}\pi$.

3.4 Adaptive Fourth Order Sliding Mode Controller (AFOSMC)

Consider the system (2) and the control (4), where $w(x)$ is defined by (19) with $n = 4$

$$w(x) = sign(x_4 + \beta_1(x_3^6 + x_2^4 + |x_1|^3)^{1/12} \\ sign[x_3 + \beta_2(x_2^4 + |x_1|^3)^{1/6} \\ sign(x_2 + \beta_3|x_1|^{3/4} sign(x_1))])$$

in [2] the value of gains are proposed as $\beta_1 = 3, \beta_2 = 1, \beta_3 = 0.5$.



(a) (b)

Fig. 6. a) Position of the mass (state x)-ASC and b) Velocity of the mass (state y)-ASC

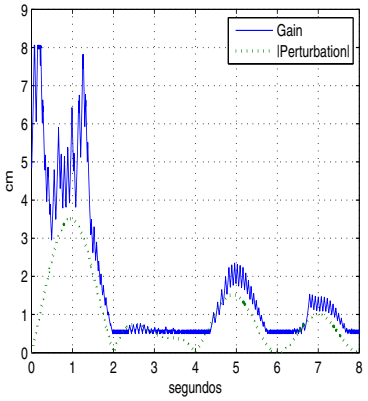


Fig. 7. Gain $k(t)$ and perturbation-ASC

3.5 Discussion of Results

It is observed in figures (2, 4, 6, 8, 9) that the system converge to sliding mode in finite time. It is clear from the same figures that the chattering amplitude is at most of the order of micrometers and is proportional to the amplitude of the perturbation, thus the objective to reduce the amplitude chattering to a minimum level is achieve.

The dynamics of the gain of the three controllers is observed in figures (3, 5, 3.3, 10). It is clear that the amplitude of the gain varies with respect to the amplitude of the perturbation. As a consequence of the adaptation of the gain, the energy demanded by the controllers is only the amount needed to compensate the perturbation saving energy unlike the classical sliding mode controllers where the gain is constant and the controllers demands a fixed amount of energy the may not be needed for the control objective.

A delay on the gain signal with respect to the perturbation is observed in figures (3, 5, 3.3, 10) due to the implementation of the filter that is needed to obtain the equivalent control (u_{eq}). The delay leads to a loss of the sliding mode that is observed in figures (2, 4, 6, 8, 9). For example in figure (3) between second 4 and 5 the amplitude of the perturbation is greater than the amplitude of the gain leading to a destruction of the sliding mode that is observed in figure (2). This is the main disadvantage of this adaptation method.

4 A Comparison between Adaptive SMC and Augmented Order SMC (AOSMC)

Consider the following second order system

$$\begin{aligned} \dot{x}_a &= y_a \\ \dot{y}_a &= d(t) + u_a(x_a, y_a) \end{aligned} \tag{25}$$

where $|d(t)| < D$ is a bounded perturbation and $u_a(t, x_a, y_a)$ is the control. To drive the system into sliding mode the order of the system is increased in order to smoothen the control signal applying an integrated control signal. Introducing the virtual state z_a the system can be expressed as

$$\begin{aligned} \dot{x}_a &= y_a \\ \dot{y}_a &= z_a \\ \dot{z}_a &= \frac{d}{dt} d(t) + v_a(t, x_a, y_a, z_a) \end{aligned} \tag{26}$$

where

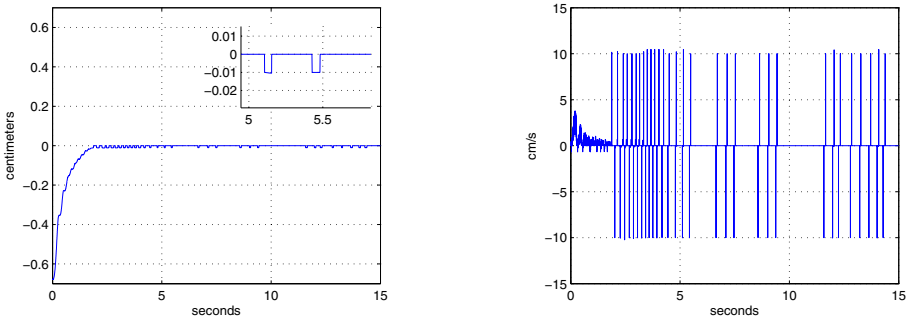
$$\begin{aligned} v_a(t, x_a, y_a, z_a) = \\ = -g[z_a + 2\left(|y_a|^3 + |x_a|^2\right)^{1/6} \\ sign(y_a + |x_a|^{2/3} sign(x_a))] \end{aligned}$$

is a third order control signal defined by (19) with $n = 3$ that enforces the system into sliding mode. Driving the variable z_a into sliding mode the integral of the control $u_(x_a, y_a)$ equal to the perturbation $d(t)$. Therefore the perturbation $d(t)$ is compensated by a smooth control signal reducing the chattering amplitude.

Consider the system(25) where the following ATEC is implemented

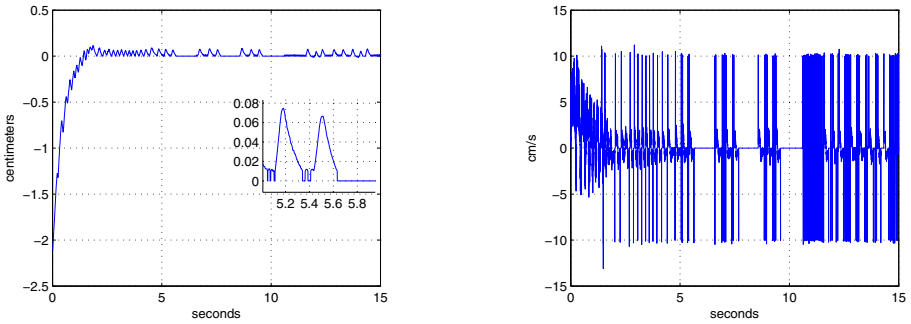
$$u_a(x_a, y_a) = -k(t) sign\left(y + |x|^{1/2} sign(x)\right). \tag{27}$$

The dynamics of the adaptive gain $k(t)$ are defined by the algorithm (9).



(a) (b)

Fig. 8. a) Position of mass 1 (state x_1)-AFOSMC and b) Velocity of mass 1 (state x_2)-AFOSMC



(a) (b)

Fig. 9. a) Position of the mass 2 (state x_3)-AFOSMC and b) Velocity of the mass 2 (state x_4)-AFOSMC

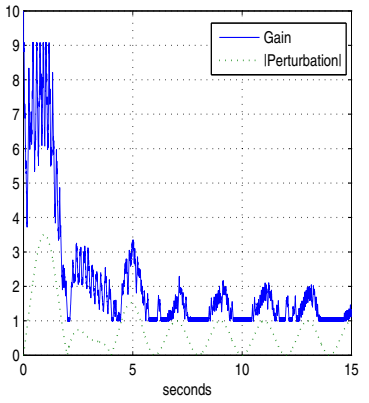
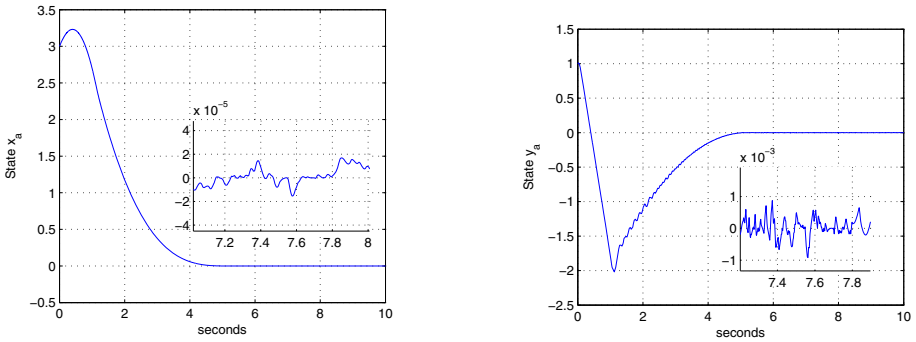
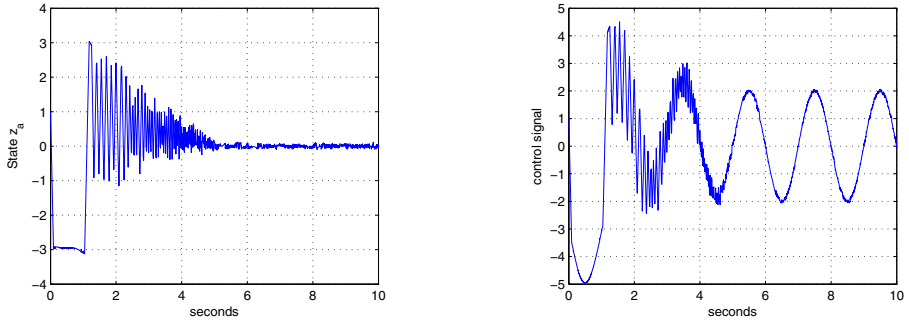


Fig. 10. Gain $k(t)$ and perturbation-AFOSMC



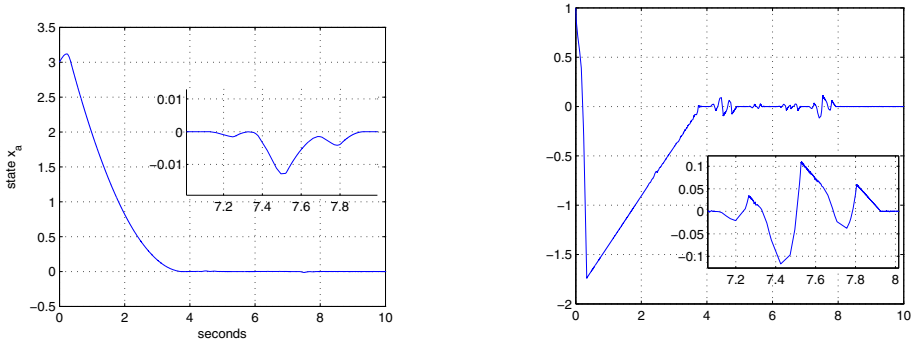
(a) (b)

Fig. 11. a) State x_a-AOSMC and b) State y_a-AOSMC



(a) (b)

Fig. 12. a) Virtual state z_a-AOSMC and b) Control signal $u(x_a, y_a)$-AOSMC



(a) (b)

Fig. 13. a) State x_a-ATEC and b) State y_a-ATEC

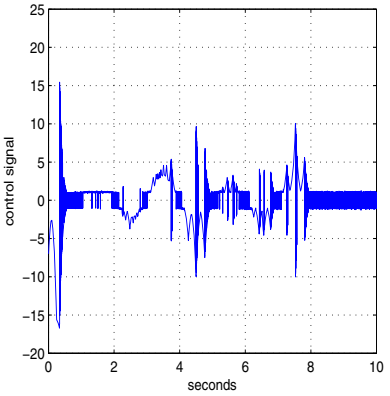


Fig. 14. Control signal $u_a(x_a, y_a)$-ATEC

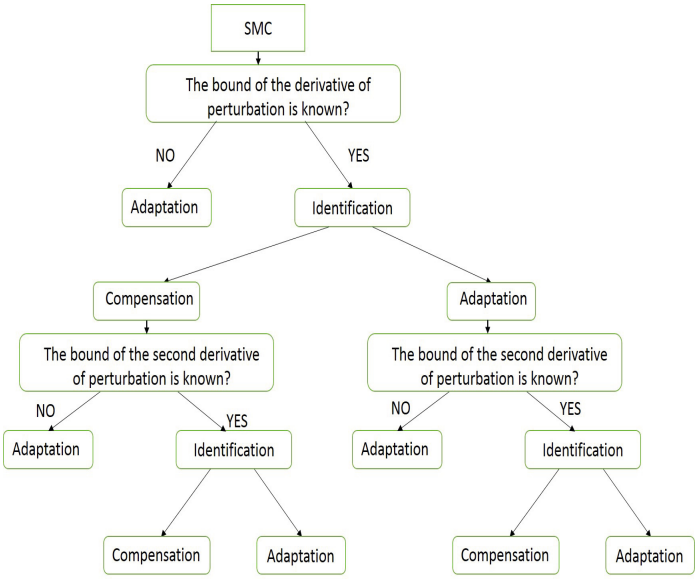


Fig. 15. Procedure to reduce the chattering amplitude

Both systems (25) and (26) are simulated with the controllers mentioned. The parameters for system (25) are $k_{min} = 1, k_{max} = 15, \alpha = 0.95, \gamma = 4\pi, \tau = \sqrt{.001}$ and the initial conditions $x_a(0) = 3, y_a(0) = 1, k(0) = 7$. The value of the gain of the system (26) is $g = 25$ and the initial conditions are $x_a(0) = 3, y_a(0) = 1, z_a(0) = 2$. For both systems the perturbation signal applied is $d(t) = 2\sin \pi t$.

By comparing figures (11)and (13)it can be observed that both the adaptive SMC and the augmented order SMC reduce the chattering amplitude. Also in both cases the control signal is similar to the perturbation (figures (12(b),)(14)).

It is clear that the augmented order SMC reduce the amplitude of the chattering better than the adaptive SMC and the amount of energy demanded by the augmented order SMC is less than the energy demanded by the adaptive SMC.

It it clear that both methodologies are based on the identification of the perturbation. This identification can be achieved if the bound of the derivative of the perturbation is known. Then if the identification is possible two approaches can be used. The first approach is the one that is presented in this work. Its advantage is that an ideal-sliding mode is achieve theoretically, also the gain is always greater, in a proportion, than the magnitude of the perturbation. The disadvantages are that a filter time constant is needed, also, in application, the equivalent control is delayed because it is a filtered signal.

The second approach that can be used when the identification of the perturbation is possible, is to use higher order sliding mode to compensate directly the perturbation. Also with the estimated signal can be used as a SMC gain. This approach is presented in ([14]). The advantage is that the perturbation can be compensated, theoretically, exactly. The disadvantage is that with some noise in the input, the precision of estimation is lost.

The two approaches mentioned required the knowledge of the derivative of the perturbation. If the derivative of the perturbation is unknown, only adaptation of the controller gain can be used to compensate the perturbation. For example, in ([13]), the gain of the controller is increased until the sliding mode is achieve, then the gain is decreased until the sliding mode is lost. The advantage of this approach is that the bound of the derivative of the perturbation is not required. The disadvantage is that only a *real-sliding mode* is achieved.

The question now, is when to adapt the controller gain? The proposed answer is illustrated in the following diagram. The aim is to reduce the chattering effect.

The adaptation is required only when there is no information about the derivative of the perturbation. The proposed method is to identify the perturbation as many times as the information of the derivatives of perturbation available, and compensate the perturbation or adapt based on the estimation. Once the information is not available, then adaptation is required.

5 Conclusions

The implementation of an adaptive gain increases the efficiency of sliding mode controller. With the adaptive gain, the controller demands a minimum amount of energy necessary to compensate the perturbation. Also the chattering amplitude is reduce to a minimum value and its proportional to the amplitude of the perturbation.

The use of a low-pass filter to obtain the equivalent control produce a delay on the dynamics of the gain. The delay could produce a momentary loss of the sliding mode. This is the main disadvantage of this method. In order to decrease the effect of the delay, a data acquisition system with a smaller sampling time is required to reduce constant of the low-pass filter, improving the accuracy of the adaptive algorithm for the gain. The smaller the filter constant, the greater the accuracy of the adaptive algorithm.

The adaptive SMC is compared to an augmented order SMC. It is concluded that the performance of the augmented order SMC is better than the adaptive SMC, not only because the augmented order SMC presents a better reduction of the chattering amplitude and in the amount of energy demanded by the controller, but also because the control signal does not present a delay like the adaptive SMC does. In both cases the knowledge of the bound of the derivative of the perturbation is required.

The main conclusion of this work is that if the bound of the derivative of the perturbation is known it is reasonable to identify the perturbation and then compensate it or adapt based on the estimation. The adaptation is required when the bound of the derivative of the perturbation is not known. In this work it is proposed to identify as many times as possible and then adapt the gain when the identification is not possible.

References

1. Levant, A.: Higher-order sliding modes, differentiation and output-feedback control. Int. J. of Control 76(9/10), 924–941, Special issue on Sliding-Mode Control (2003)
2. Levant, A., Michael, A.: Adjustment of high-order sliding-mode controllers. Int. J. Robust Nonlinear Control 19, 1657–1672 (2009), www.interscience.wiley.com, doi:10.1002/rnc.1397 (published online November 7, 2008 in Wiley InterScience)
3. Astrom, K.J., Wittenmark, B.: Adaptive Control, 2nd edn. Addison-Wesley, NY (1989)
4. Sastry, S., Bodson, M.: Adaptive Control: Stability, Convergence, and Robustness. Prentice-Hall, NY (1994)
5. Bartolini, G., Ferrara, A., Usai, E., Utkin, V.I.: On Multi-input Chattering-free Second-order Sliding Mode Control. IEEE Transactions on Automatic Control 45, 1711–1717 (2000)
6. Boiko, I., Fridman, L.: Analysis of Chattering in Continuous Sliding-mode Controllers. IEEE Transaction on Automatic Control 50, 1442–1446 (2005)
7. Boiko, I., Fridman, L., Pisano, A., Usai, E.: Analysis of Chattering in Systems with Second Order Sliding Modes. IEEE Transactions on Automatic Control 52, 2085–2102 (2007)
8. Perreault, D.J., Selders Jr., R.L.: Switching- Ripple-Based Current Sharing for Paralleled Power Converters. IEEE Transaction on Circuit and Systems 46(10), 1264–1274 (1999)
9. Utkin, V.I., Poznyak, A.S., Ordaz, P.: Adaptive super-twist control with minimal chattering effect. In: 2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC), December 12-15, pp. 7009–7014 (2011), doi:10.1109/CDC.2011.6160720
10. Cruz-Zavala, E., Moreno, J., Fridman, L.: Uniform sliding mode controllers and uniform sliding surfaces. IMA Journal of Mathematical Control and Information 29(4), 491–505 (2012), doi:10.1093/imamci/dns005
11. Deepak, F., Bandyopadhyay, B., Fridman, L.: Non-linear sliding surface: towards high performance robust control. IET Control Theory and Applications 6(2), 235–242 (2012), doi:10.1049/iet-cta.2010.0727

12. Castañosa, F., Fridman, L.: Dynamic Switching Surfaces for Output Sliding Mode Control: an H^{∞} Approach. Automatica 47(7), 1957–1961 (2011)
13. Shtessel, Y., Taleb, M., Plestan, F.: A novel adaptive-gain supertwisting sliding mode controller: Methodology and application. Automatica 48(5), 759–769 (2012)
14. Ferreira, A., Bejarano, F.J., Fridman, L.: Robust Control With Exact Uncertainties Compensation: With or Without Chattering? IEEE Transactions on Control Systems Technology 19(5), 969–975 (2011)

Part II
Navigation

The Early Eighties: Development of In-Flight Transfer Alignment – Challenges and Methods

Zeev Berman

Berman Consulting and Training Ltd,
Kinneret, 15105, Israel
berman.consult@gmail.com
http://www.bermanconsult.com

Abstract. In the late seventies and early eighties, Rafael devolved a very advanced (at the time) transfer alignment algorithm. This required the development of an entire infrastructure for navigation work: strapdown navigation equations, navigation error model, Kalman filter implementation, system level error model, inertial measurement unit, real-time, floating-point computer, test design, implementation and analysis. This paper tells the story of this enterprise, from the preliminary studies to successful operational deployment, by pointing out the different phases and lessons learned.

1 Introduction

This paper is about the development of a navigation system, and touches on a very wide range of engineering topics related to navigation systems. The main goal of the paper is to describe the people, decisions, challenges, problems, and solutions during this project. The dilemma was deciding on the extent to which the underlying engineering and mathematical topics should be detailed. The decision was ultimately made to write a paper without equations, mainly because the subject is so broad that once I started writing equations, I would not know where to stop. Moreover, the main objective of the paper is to provide the reader, who is not necessarily a navigation expert, a history background of the technical achievements. Those who are missing the equations are directed to references in which the relevant ones are described. Of course, the best descriptions are in Itzhack Bar-Itzhack's technical documents.

Itzhack Bar-Itzhack is one of the key individuals in this story. Like many others, I learned a lot from Bar-Itzhack: I took his course at the Technion and read his brochures, which were always on his desk. We knew each other pretty well: Bar-Itzhack worked as a consultant at Rafael; we even played volleyball together. Nevertheless, although the paper is being published in the Itzhack Y. Bar-Itzhack Memorial Symposium, it is not a dedicated memorial to the man. Its main purpose is to depict a picture of engineering challenges and the progress in their achievements.

To maintain a continuous flow in the paper, we present here, for non-experts, a short description of the technical terms used in the sequel (see [14] for more details).

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_10

Accelerometer – an instrument that measures a specific force (acceleration combined with gravity effect).

Gyro – an instrument that measures angular rates.

IMU (Inertial Measurement Unit) – a unit composed of accelerometers and gyros to measure specific forces and angular rates in three orthogonal axes.

Inertial Navigation – a method to calculate position, velocity and angular position from initial conditions and accelerometer and gyro outputs.

Stabilized Platform Navigation System – an inertial system with accelerometers installed on a stabilized gimbal.

SD (Strapdown) Navigation System – an inertial system without any stabilizing gimbals. It is stiff with respect to the body to which it is attached.

SD Navigation Algorithm – an algorithm that integrates the IMU outputs to provide position, velocity and angular position where the IMU is installed rigidly with the body to be navigated.

Transfer Alignment – a method of finding attitude (orientation) of a navigation system from a velocity (or position) reference. To achieve this, the reference data are provided for a certain time and some maneuver during this phase is required. The standard implementation is based on Kalman filtering (see [2] for further description).

Quaternion representation – a method to describe attitude by four normalized *numbers.*

ARU (Attitude Reference Unit) – a method to calculate orientation from direct measurements of gyros and accelerometers, with the underlying assumption that the mean value of acceleration is zero, and therefore the accelerometer's mean value is related to a gravity vector. These units are usually integrated with heading gyro and optional magnetometer.

Captive Flight – the phase in the missile's mission when it is operating but connected to the aircraft.

Free Flight – the phase in the missile's mission when it flies without any connection to the aircraft.

Inertial Mid-Course – the part of free flight when the missile is steered by its navigation system.

2 The Early Years (1974–1980): From Conceptual Study to Design and Implementation

Our story begins in 1974, the year that saw the creation of a missile model with its 6DOF simulation for a medium-range precise air-to-surface missile, later called Popeye. The relatively long range was due to the requirement for a standoff. By standoff we mean that the missile should be dropped beyond the range of most air-defense ammunition.. At that time, Rafael had already gained some experience with precise TV-guided weapons. Their principle of operation was to present a pilot with an image from the target vicinity; the pilot's task was to recognize the target on this image and correct the missile's course until it hits the target. Implementation of such a system required a seeker, a high-quality TV camera mounted on gimbals to provide

stabilization and movement capability, and two-way communication links. Having mastered the concept of terminal guidance, the open question was the midcourse—how to guide the missile with such accuracy that indeed the seeker would point close to the target, and the pilot could recognize it with high confidence. The new challenges were related to the several tens of kilometers range and relatively long flight time. Many guidance concepts, based on the classical ARU approach, that combined accelerometers, gyros, and perhaps an additional sensor to calculate the attitude directly, were analyzed. It took until mid-1975 to understand that for this type of range and for the required electro-optical performance, inertial mid-course was absolute necessary. Moreover, as an outcome of this work, the goal of 200-m accuracy (2–2.5 sigma) was stated as a primary requirement for the navigation subsystem.

No one knows who proposed the strapdown (SD) implementation for such a problem—it could have been Itzhack Bar-Itzhack, or someone influenced by him.

In any case, before the end of 1976, two navigation pioneers went to the United States to visit companies that had made some progress in SD technology. Of course, travel preparations were much more involved back then, without the benefit of Google and Internet searches. They needed to read a lot of professional literature and to consult with every available expert.

They visited three companies (names are withheld for private reasons), and the responses and impressions were diverse:

- At one company, the vendor representatives refused to discuss the implementation issue with guests from Israel. Their assertion was that no business could come of this meeting and they were not interested in teaching the team from Israel how to implement the SD system.
- At the second company, the team found a system that was over 10 times less precise than required. Although the company was interested in cooperation, the low performance did not justify the effort.
- At the third company, the team found a nice prototype of an inertial system that closely matched the Popeye's requirements. They found an engineering and management team that was willing to cooperate. The Israeli team understood that this company would provide good support for the entire navigation system composed of hardware (IMU), navigation software and a Kalman filter for transfer alignment.

Nevertheless, the Israeli Department of Defense decided to develop the Popeye's navigation system in Israel. The Tamam division of Israel Aircraft Industries, Ltd. proposed to use a stabilized platform which they were producing at the time, but the cost ($1M) and size were prohibitive. After a long discussion between all involved parties, three important decisions were made:

- The Popeye navigation system would be of the SD type.
- The hardware (IMU) would be developed by Tamam.
- The navigation algorithms would be developed at Rafael.

So the task of implementing a SD IMU with a price of around $100K, weighing less than 6 kg with errors of 1 deg/h for gyro drift and 1 mg bias for the accelerometer

was assigned to Tamam. The assigned project manager just completed M.Sc. degree under the supervision of Itzhack Bar-Itzhack. It was an important project with enormous challenges: sensor development, accompanying electronics, mechanical design, all to be the first of their kind. I am sure that Taman engineers would add the relationships with Rafael's people as one additional and demanding challenge. Eventually the IMU was ready, on time and with outstanding performance. I am certain that the history of this development would provide an excellent foundation for a dedicated paper.

The first man that started to develop navigation algorithms at Rafael tells: "We got the main idea from Itzhack Bar-Itzhack; he presented us with differential equations and explained all of the details. For example, it was his recommendation to use quaternion integration. We performed the detailed work, but every time we progressed, Itzhack Bar-Itzhack was already ahead of us; as we got into a problem or dilemma, he had already studied the issue and came prepared with an analysis, explanations and recommendations."

The task was quite demanding: to design an algorithm that discretizes the navigation differential equations in a way that can be implemented in the proposed Rafael-homemade computer, the μ-Remez. Every multiplication and load was counted and optimized. Eventually a very effective navigation algorithm that fit well into the constraints was developed. The resulting procedure was a multi-rate integration with very carefully selected and optimized discrete integration methods. One interesting example was the quaternion normalization, which was required at quite a high rate. The optimization result was to use a linear approximation for normalization instead of the "standard" deviation by square root of the sum of squares. The testing of the navigation algorithm was combined with 6DOF simulation. From the very beginning, their models matched each one; the same person was in charge of both, 6 DOF simulation and navigation algorithms, so the differences were only because of numerical errors (integration rate and computer resource-saving navigation algorithm). In addition, this platform (6DOF simulation and navigation algorithm) served for an analysis of sensor error effect. One important remark is needed here: the approach then, and during the entire development phase, was not to use "blind" simulation for analysis but to associate it to an analytical (usually simplified) analysis. One could not simply present a result and state: "these are the results that I got from the simulation." Everyone was expected to explain why these results appeared to be reasonable. This work ended with a well-known report describing the proposed algorithm, the main trade-offs, sensitivities and tests. This document is well known in Rafael and is still in use for training the younger generations. The algorithm proposed then has remained basically unchanged, and every navigation system developed at Rafael is based on it.

The challenge of transfer alignment was even more daunting: the differential equations describing the inertial navigation error models were not yet available and it was clear that the discrete time and efficient implementation issues were very demanding. This task was combined with M.Sc. degree dissertation under the supervision of Itzhack Bar-Itzhack. During 1977 to 1979, the navigation error model was developed and formulated it into the framework of in-flight transfer alignment. In particular, the attitude error model required special attention. This study led to a SD version of psi model, which was already known for platform navigation. It was somewhat intriguing

that they developed the same equation as in the case of a platform, but with the opposite sign of the drift term. In addition, they proposed an extended sensor error model and integrated it into one combined system. Special effort was dedicated to the random noise integration formula required by the Kalman filter. The analysis distinguished direct sampling from integral sampling, and he provided rigorous analyses for both cases. One of the conclusions was the need for integral sampling, and Tamam developed a mechanism called V/F to provide angular and velocity increments instead of angular rate and accelerations. This dissertation was very extensive, several hundred pages long, with some of the equations written in A3 format, it could in effect be considered a Transfer Alignment Handbook, as it completely covered all models required to develop the transfer alignment algorithm. Unfortunately, this work was defined as classified and was never published. The reason for this was that part of the work dealt directly with the Popeye transfer alignment algorithm. It was based on a Kalman filter with a 12-state vector: velocity error, attitude error, gyro drift and accelerometer bias (all of them in three axes). The velocity error measurement was chosen, based on a comparison between aircraft and SD velocities, with 1-sec intervals between measurements. Those critical decisions were based on common sense and good engineering insight into the dominating phenomena, and were later justified by simulations. For example, acceleration measurements were rejected due to the high flexibility of the wing that Popeye missile was installed underneath and the long time between measurements, related to computer-resource limitations. The states for gyro drift and accelerometer bias were added to allow tracking and compensating for these slow-changing error terms. The time between measurements, 1 sec, was set as the longest time (to save on computer resources) that would presumably allow the required tracking quality. The performance analysis was based on S-shape maneuver during in-flight alignment. The Kalman filter calculations were very computationally expensive and required working with a floating point machine, which was not available in feasible sizes.

The project manager was dedicated to meeting the time schedule for a series of system tests. His message was that the navigation system's development, after 4 years of effort, was still fraught with huge uncertainties; therefore, in the event of a delay or critical problem, he would replace the proposed SD navigation system with a backup one, based on a simple ARU that had already been developed for airframe configuration tests. In this atmosphere, the need to reduce development risks and efforts was vital.

The next task was to build a simulation that would combine SD navigation and in-flight alignment. During this task, an important achievement was found. The observation was that at the cost of a minor approximation in the stochastic part of the model, but without sacrificing the accuracy of its deterministic part, the algorithm's complexity can be reduced. The key observation was that the simplified system transition matrix is nilpotent. The precise calculation of the discrete transition matrix in a time-varying system is related to matrix multiplication, which is computationally heavy. The nilpotent property states that those multiplications come to zero. This observation opened the possibility of calculating the transition matrix by simple integrations (summations). The term nilpotent, that may create negative connotations, was very

attractive in the eyes of the project leaders, because it eliminated the need of a special-purpose computer; the new Intel 8086 processor with floating point 8087 coprocessor was able (when working at almost 100% capacity) to carry out the calculations required for all of the navigation algorithms.

3 Years 1980–1985: It Works! Integrations, Tests and First Improvements

The Project Test Plan consisted of several phases, each phase dedicated to testing one of the major subsystems, while the subsystems tested in the previous phases served as the infrastructure for those tests. Table 1 describes the main test plan from the perspective of navigation subsystems

All navigation test analyses were based on the principle of data recording and off-line reconstruction. The idea of recording was to store the entire stream of IMU outputs (6 numbers at 60 Hz) and aircraft navigation blocks (9 numbers at 20 Hz). The requirements for data rate and storage volume were high, but the most demanding requirement was with respect to data quality. To successfully calculate the navigation data, the stream of IMU data had to be close to perfect. Because the navigation algorithm was based on integration, lack of even a single IMU block (for example during maneuvering) could harm the entire task.

Table 1. The project test plan

Test name	Type	Goal	Configuration
B	Captive	navigation concept	Big commercial aircraft, alternative IMU hardware, alternative aircraft navigation system, recording system
104/2	Captive	navigation algorithm + hardware	Dedicated aircraft with its navigation system, dedicated IMU (from Tamam), recording system
104/3	Captive	navigation implementation	Dedicated aircraft with its navigation system, Popeye electronic box (navigation and all other operational computers, IMU) installed in the missile envelope, recording system
107	Captive	seeker tests	As above + seeker
109	Captive	communication tests	As above + communication pod
1004	Free	inertial mid-course	As above but missile with its motor, control and guidance subsystems. Telemetry.
1007	Free	final system test	The complete system

The proposed solution was quite complex, but presumably the best available given early days hardware and software limitations. The recording medium was analog video; dedicated hardware to convert digital data into video stream (and vice versa) was designed and built. To increase data quality, two parallel video recorders were installed. The process of preparing the digital stream data was difficult and long: the first phase was to play the video data and convert it into a digital stream of blocks, and sometimes some reiteration was required. A dedicated mini-computer was involved in this process. The second phase was to merge the blocks from two video recorders; the third was manual corrections of some blocks that were erroneous after the merge. In retrospect, this effort was critical to acquiring essential knowledge and an understanding of navigation system performance and sensitivities.

The analysis of the first navigation test B was difficult; the data-reduction process was very lengthy and then it turned out that the sensor calibration was inconsistent. It took several weeks to find the reason for the problem encountered in this test: bad time synchronization. The surprise was twofold: indeed the temporal synchronization between IMU data and aircraft data was worse than expected, but the system sensitivity to these phenomena was much higher than one might have intuitively expected. Since then, synchronization has become one of the most important integration issues, the topic of many discussions and much testing. We can clarify this theme with a simple example. Assume that we make a 0.5 g (5 m/sec^2) turn with 3 cm/sec velocity noise and we are willing to estimate bias up to 0.5 mg (0.5 cm/sec^2). For acceleration of 0.5 g = 5 m/sec2, 1 msec of synchronization error will produce a measurement error of 0.5 cm/sec, significantly less than the velocity error of 3 cm/sec. However, after 1 sec, the bias error of 0.5 mg will cause a velocity error of 0.5 cm/sec, so the same as the error due to miss-synchronization The observation was that the comparison of error due to miss-synchronization with velocity noise is misleading. The correct comparison is between the synchronization error and the error caused by bias that we are willing to estimate. Indeed, these errors appear with similar correlation to the trajectory. In this case, since we would like to keep the synchronization error well below the error due to the estimated bias, the allowed synchronization error should be on the order of 0.1 msec.

Test series 104/2 was less problematic; integration with the real IMU went smoothly, and the performances and sensor calibration behaved well. This was a great opportunity to optimize the transfer alignment maneuver. The starting point was relatively long S-shaped maneuver, required for a good estimation of the heading error and z-axis drift. It turned out that in a Popeye-type missile, the sensitivity to heading error is relatively low; the simple (albeit non-intuitive) explanation was as follows: if a missile is launched and does not perform any maneuvers, than at the end of the mission its position error will be zero, even if the heading error was large (in this case all other errors are assumed to be zero). In other words, heading error influences system error only if there is acceleration. The most significant acceleration (due to the rocket motor) was at the beginning of the mission. This observation allowed to relax the requirements for z-axis drift estimation and significantly reduced the need to estimate heading error. As a result, a shorter maneuver was proposed which was much easier from an operational point of view. It was based on a single, relatively small turn and

the time of captive flight was reduced significanly. An additional result of the 104/2 series was the tuning of the Kalman filter.

On Itzhack Bar-Itzhack, the first navigation group leader relates the following: "In those days, Itzhack Bar-Itzhack was coming in to Rafael once a week to work with us. For me, he was one of the team members, perhaps younger than many of the others. He was always smiling, in a good mood and ready to tell new jokes, some of them not necessary politically correct by today's standards. Nevertheless, from a professional point of view, he was always serious, well-organized and very precise. He was always ready to carry out new assignments, whatever was needed, once he had studied the details of aircraft navigation systems in depth; on one occasion, for example, he proposed a missile trajectory generator algorithm. We always discussed the current issues, problems and plans with him."

The 104/3 tests, performed with the actual navigation system (hardware and software), were relatively extensive; they aimed to cover the entire operational envelope. The software integration was very successful; it worked well from the first flight test. The most important lesson learnt here was the system's sensitivity to flight conditions: for quiet flights very good results were obtained, whereas for low-level, fast flights, the results were worse but still within requirements. The long-flight 104/3 test series, followed by the even longer 107 series, was used to create a huge library of navigation data: real-life trajectories, sensor performance together with post-processing analysis, and error sensitivities for a very broad family of error sources. This library, which was continuously updated with new tests, served as an excellent platform to learn navigation systems and develop new algorithms.

The seeker test series 107 showed a problem that appeared to be related to the navigation system. The seeker pointing errors were too large. Rafael people claimed that this phenomenon was due to aircraft navigation errors, whereas the air force claimed that after position update the aircraft navigation error was about 50 m, and such large pointing errors must therefore be related to the missile. The first task was to plan a test that would separate the error sources. The idea was to keep the seeker tracking a target at known positions, so the system pointing errors could be measured continuously for several tens of seconds. A Kalman filter would be designed with measurements of pointing errors and states of position error, attitude error (in an inertial reference frame) and misalignment between the seeker and the navigation system (related to the body reference system). The Kalman filter implementation was off-line and based on the already existing reconstruction infrastructure. This test showed with high confidence that the pointing errors were due to position errors. Later analysis showed that although the position updates were quite accurate, they did not properly correct for the velocity error and therefore, after 1 min or more, the aircraft accuracy was significantly worse than the specified 50 m. This fact endangered the entire project, but a solution was proposed almost immediately. The initiative was to make auxiliary target updates. The idea was to find, close to the actual target, an auxiliary target with good visibility and known location. Then, from pointing to this target, the system could estimate the pointing errors and correct them (assuming that they came from horizontal position errors). This algorithm was accepted and applied in the system. Years later, pilots still use this procedure with every aircraft working with GPS,

claiming that it improves performance; the reason for this is unknown, but my impression is that this procedure survived the GPS era mainly because it makes the mission less boring. In any case, the lessons learned in line-of-sight analyses and the integration with navigation errors were later used to propose installation of the navigation system inside the seeker gimbal [10]. A well-known application of this approach is Rafael's Litening Airborne Navigation and Targeting Pod.

The preparation for the first free flight test, test 1004, went according to plan, but one day before the test, during the final captive flight test, a new feature was tested. During the airframe configuration test, immediately after the launch, strong roll movement was observed. Therefore, the pilot performed a strong roll movement (as fast as he could) to simulate the release roll movement. As a result, in this route, large navigation error appeared. It happened only once and there was a big debate whether to stop the test launch or not. Rafael people (at least the senior ones) wanted to continue; it was the test pilot who persuaded the project managers (from the Ministry of Defense) to postpone the test and require, from the Rafael team, a solid analysis of the phenomena.

The result of the analysis was surprising: it was essentially a system/hardware problem. It turned out that the actual maximal angular acceleration is much higher than the one specified. Eventually the spec was changed, Tamam made the necessary changes in their design and after several months of intensive work, test 1004 was ready to restart. At the very end of the last test before the launch, an IMU error message appeared. Again, long discussions, collection of all available data, consultations with experts, lasted till late-night hours.

This time the decision was to continue the testing. On the following day, early in the morning, the missile was launched. Before the successful happy ending, the missile was almost terminated by safety personnel due to a lengthy lack of communication with the operations room, first due to telemetry problems, then due to a tape-recorder, that someone had put in the room and created acoustic noise oscillations. At the very last moment the problems were fixed, and everyone in the operations room, including the safety personnel, were able to witness the missile's precision in following its designated route.

4 Post 1985: Still a Lot to Do

After the success of test 1004, the navigation group's involvement in the project has gradually reduced. Then, in 1986, great excitement spread from the Popeye project management: we were going to demonstrate the system in the United States. Project management's view was that since the missile's development had been completed, in order to reduce our costs and time schedule, the aircraft to be demonstrated in the US should mimic the interface that was already integrated in the Popeye missile. At this stage, schedule was very tight and intensive work was required: a few months for implementations and integrations, then a few captive flights for testing and operational training and finally, a full operational launch. In the course of the meetings and tests, we understood that our system had excellent characteristics: the quality of the

seeker, the navigation accuracy and the very short and easy transfer alignment procedure were world-class outstanding. From a technical point of view, the process ran very smoothly with no critical failures. One improvement of note was the use of the aircraft simulator to create dynamic trajectories, in order to record the blocks sent to the Popeye and analyze them. In this way we were able to fix some problems well ahead of the flight tests.

During the integration in the US, we understood that changing an aircraft's interface to simulate another interface was an once-in-a-lifetime scenario and that the next time we would need to change our interface. Then we understood that our design lacks the flexibility to support such changes. The problem was related to Kalman filter propagation and the time window for measurements. The first implementation (due to a lack of spare computer resources) was very rigid; the covariance matrix was propagated to a known a-priori measurement time, creating a narrow time window in which the measurement should appear; if it didn't, the measurement was discarded and the system was moved to the next second. In the next version of the transfer alignment algorithms, due to the change in the covariance propagation scheme, the system was ready to receive asynchronous measurements, with the only limitation being the minimal time between measurements.

Another great challenge faced the team when it started work on the inertial navigation system for air-to-air missiles. It was clear that a dedicated maneuver for transfer alignment was out of the question. The approach was to perform a continuous transfer alignment, namely to turn on the missile navigation system before takeoff and keep it working continuously such that the missile would be ready to launch all the time. Since in every aircraft flight there are always periods of some acceleration, sensor errors and pitch and roll angles can be estimated and kept accurate during the flight—after intensive work for proper Kalman filter tuning of course. The only problem is that for long flight periods with no accelerations, the heading error can grow. In an air-to-air missile, due to its huge acceleration, the sensitivity to heading error is much more severe than that in air-to-surface missiles. The solution came from a very interesting direction: the reason that we could not perform a direct alignment (i.e. simply copy aircraft orientation onto that of the missile) was that the missile had been installed under the wing, and its relative orientation was changing during the flight. Indeed this was true, but only for pitch and roll; the heading misalignment (the difference between the aircraft and missile orientations) was almost fixed during the flight. This observation led to adding another state to the Kalman filter, the new state being the heading misalignment, and the resulting 13-state Kalman filter provided excellent continuous transfer alignment. This 13-state Kalman filter has become a standard solution that is implemented in all relevant airborne systems.

The stringent requirement for time synchronization between the aircraft and missile has always been a key issue in aircraft integrations. The number of types of aircrafts which needed to be integrated was constantly increasing, as was the number of Rafael airborne systems that included navigation units. As a result, more and more cases of an aircraft navigation system not providing the required synchronization accuracy began to appear. The obvious consequence was performance degradation, until an important observation was made. It was clear that precise synchronization is required

to accurately interpret the system errors during maneuvers. However, one could omit the measurements during the maneuvers and estimate the system errors after a maneuver had been completed. This non-trivial observation claimed that due to the integral nature of the system, there is no significant harm to performance when no measurements are performed during the maneuver itself (but with enough measurements after the maneuver) compared to the case in which all measurements are taken.

At this point our paper is complete; the basic solution for transfer alignment and its essential improvements have been described. During those years, Rafael provided state-of-the-art systems, with a constant line of improvements. Of course the real story never ends, and the navigation group at Rafael continues to develop navigation systems, based on the foundations described here, facing new challenges and producing new achievements.

In my opinion, the secret to accomplishment lies in following these three guidelines, as well illustrated in the story described herein:

- Recruit capable people
- Provide a challenge
- Build an infrastructure and culture to analyze integrations and tests properly and in depth

Acknowledgments. First I must thank Yaakov Oshman who encouraged me, from the very beginning, to write this paper, and for his help and support which were vital in surmounting several difficulties (for example, delays due to issues of security).

Then I must thank my colleagues from the navigation group who told me stories from the different project phases, helped me formulate a complete picture, and provided helpful remarks. During these interviews and during the writing of this paper, I must admit that Itzhack Bar-Itzhack's absence was painfully evident: I had so many questions to ask him…

Lastly, and most importantly, I must acknowledge all of the people who worked on this project: the project managers, the engineers: systems, software, hardware, control, navigation and integration, the technicians, the Israeli Air Force flight test team, and so many others whose coordinated work allowed everything to come together. For me, they were simply wonderful people who I was lucky to work with and from whom I learned a great deal.

References

1. Bar-Itzhack, I.Y., Levinger, E.: An analysis of errors in navigation inertial systems. Technion (1975) (in Hebrew)
2. Gelb, A.: Applied Optimal Estimation. The M.I.T. Press (1974)
3. Bar-Itzhack, I.Y., Porat, B.: Azimuth Observability Enhancement during Inertial Navigation System In-Flight-Alignment. AIAA J. Guidance and Control 3(4), 337–344 (1980)
4. Porat, B., Bar-Itzhack, I.Y.: Effect of Acceleration Switching during INS In-Flight-Alignment. AIAA J. Guidance and Control 4(4), 385–389 (1981)

5. Weinreb, A., Bar-Bar-Itzhack, I.Y.: The Psi Angle Error Equation in Strapdown Inertial Navigation Systems. IEEE Transactions on Aerospace and Electronic Systems AES-14(3) (1978)
6. Bar-Itzhack, I.Y., Vitek, Y.: The Enigma of False Bias Detection in a Strapdown System during T.A. AIAA J. of Guidance (March-April 1985)
7. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Unified Approach to Inertial Navigation System Error Modeling 15(3), 648–653 (1992)
8. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Observability Analysis of Piece Wise Constant Systems, Part I – Theory. IEEE Transactions of Aerospace and Electronic Systems AES-28(4), 1056–1067 (1992)
9. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Observability Analysis of Piece Wise Constant Systems, Part II – Application to Inertial Navigation In-flight Alignment. IEEE Transactions of Aerospace and Electronic Systems AES-28(4), 1068–1075 (1992)
10. Berman, Z.: On Range and Attitude Estimation. In: Proceedings of the IEEE Position, Location and Navigation Symposium, Las Vegas (1994)
11. Reiner, J.: In-Flight Transfer Alignment Using Aircraft-to-Wing Stiff Angle Estimation. In: AIAA Conference on GNC, San Diego, CA (July 1996)
12. Reiner, J., Ben-Jaacov, J., Rotstein, H.: Transfer Alignment in the Presence of Unknown Aircraft Measurement Delays. In: IBA Conference on Navigation, Herzlyia, Israel (May 1999)
13. Bar-Itzhack, I.Y.: In Flight Alignments of Inertial Navigation Systems. Series of Control and Dynamic Systems 38, 369–396 (1990)
14. Titterton, D.H., Weston, J.L.: Strapdown Inertial Navigation Technology, Peter Peregrinus Ltd., UK (1997)

Efficient Error Model Construction

Zeev Berman

Berman Consulting and Training Ltd, Kinneret, 15105, Israel
berman.consult@gmail.com
http://www.bermanconsult.com

Abstract. This paper describes a new method for error model construction. Instead of the standard local slope analysis of the Allan variance, two major modifications are proposed: (1) the Direct Bound principle, i.e. finding an entire error model that generates analyzing tool values that tightly bound the analyzing tool values generated by the real data; (2) instead of using Allan variance as a unique analyzing tool, a variety of analyzing tools termed Direct Predictor (DP) types 0, 1, 2, and 3 are introduced. In the paper, a uniform structure of DPs is developed and their parameterization is extended. For a nominal model that consists of a Markov process with additive white noise, the analytical functions for DPs are presented (for infinite data length). The errors due to the final data length are analyzed. Using these results, a reliable optimization problem is presented to implement the Direct Bound approach. The flexibility of working with hard and soft bounds is introduced. The presented simulation results show that the proposed method is indeed efficient and provides satisfactory results for model parameter estimations. The paper concludes with a description of an entire engineering process to cover test design and its analysis.

1 Introduction

This paper is a part of ongoing research on inertial sensor calibration under changing temperature. In it, we address the question of how to find a model (time invariant) that can be used to bound the performance of the underlying navigation system, even if at the sensor level, the random (residual after calibration) errors may follow time-variant dynamics. This is the precise reason why system identification models are not popular in the navigation community. The standard approach is to use Allan variance (AVAR) analysis and construct the error model from different local slopes of the Allan plot. A survey of the literature on inertial sensor calibration and error model analysis reveals many papers on the subject (see for example [1-3]). In this context, the IEEE Standard group's attempt to create a common terminology and framework for gyro modeling is very promising. The standard approach [4] divides the error sources into two groups: stochastic and environmental. For stochastic errors, significant effort has been invested in creating a suitable stochastic model, using the AVAR and related power spectral densities (PSDs). The attempt to present methodologies for the development of models for post-calibration residual errors [5] marks important progress.

D. Choukroun et al. (eds.), Advances in Estimation, Navigation, and Spacecraft Control,
DOI: 10.1007/978-3-662-44785-7_11

The models related to PSD, for example flicker noise—termed bias instability, are well adapted for time-invariant, stationary systems and work well at room temperature, but they are not well-suited to dealing with environmental (mainly thermal) sensitivity. Our goal is to propose an efficient method of constructing an error model under environmental sensitivity, with relatively short data-collection time and satisfactory accuracy. This new approach [6, 7] is based on two principles:

- Instead of looking for local slopes of AVAR (or other analyzing functions), we propose finding an entire model that generates analyzing tool values that tightly bound those generated by the real data. This is called the Direct Bound approach.
- Instead of using AVAR as a unique analyzing tool, a variety of analyzing tools, termed Direct Predictor (DP) types 0, 1, 2, and 3, are proposed.

The applicability of the proposed approach was verified with real-life data taken from MEM's gyro calibration. For further details see [6].

The main drawback in the proposed approach [6] was that the analysis and application were carried out using simulations and trial and error search. The purpose of this paper is to convert the ideas presented and verified in [6] into a concrete engineering procedure that can be easily implemented by the navigation community.

The paper is structured as follows. In section 2, the DPs are defined as linear operators acting on the data. It is shown that all DPs have the same unified structure with different parameters for types 0, 1, 2, and 3. The nominal model in this paper is a first-order Gaussian Markov process with additive white noise. In Section 3, a unified analytical function to describe the DP values for the nominal model and infinite data length is developed. In section 4, an approximation for DP errors due to finite length is proposed and verified for a wide range of cases. Section 5 presents an optimization problem that implements the Direct Bound approach. It turns out that in order to get reliable solutions from optimization algorithms; one must carefully define the underlying optimization problem, with proper scaling and normalization. After doing so, we present the error model accuracy for instrumental gyro model, and for all four types of DPs. In this case, the DP type 3 outperforms the Allan Variance. Although the Direct Bound approach defines only hard bounds over the test data, in this section hard bound and soft bound applications are discussed and presented. Section 6 is devoted to a discussion of the entire engineering procedure, error model construction, evaluation of its accuracy, and the selection of design parameters—mainly duration of data collection.

2 The General Structure of Direct Predictors

The AVAR $\sigma_y{}^2(\tau)$ for infinite data length is defined as follows:

$$\sigma_y{}^2(\tau) = \frac{1}{2} E\left(\Delta(\tau, y)\right)^2 \tag{1}$$

Where E is the expectation operator and $\Delta(\tau, y)$ is defined as

$$\Delta(\tau, y) = \frac{1}{n}\sum_{i=n+1}^{2n} y(i) - \frac{1}{n}\sum_{i=1}^{n} y(i) \qquad n = \frac{\tau}{dt} \tag{2}$$

dt is the sampling time. The Allan deviation (ADEV) is the square root of AVAR and is denoted $\sigma_y(\tau)$.

The first interpretation for $\Delta(\tau, y)$ is the difference between successive mean values over two sampling periods. We will provide an additional interpretation for $\Delta(\tau, y)$ in the context of mean prediction error. Refer to $i = n$ as the current time; then $D^E = \frac{1}{n}\sum_{i=1}^{n} y(i)$ is the estimated mean data value (drift in gyro nomenclature) over the past. We can interpret $\Delta(\tau, y)$ using the following observation:

$$\Delta(\tau, y) = \frac{1}{n}\sum_{i=n+1}^{2n} \left(y(i) - D^E \right) \tag{3}$$

$\left\{ y(i) - D^E \right\}_{i=n+1}^{2n}$ is the sequence of prediction errors for "future" indices $i = n+1, \cdots, 2n$.

So $\Delta(\tau, y)$ is the mean prediction error (over the sampling period with length τ) and AVAR (for infinite data length) is half the covariance of the mean prediction error. The DPs introduced in [6] are generalizations of this structure; all of them are half the covariance of the mean prediction error over the sampling interval, but they are distinct in terms of estimation and prediction methods.

DP type 0 is simply the AVAR, for which the estimator is given by:

$$D_0{}^E = \frac{1}{n}\sum_{i=1}^{n} y(i) \tag{4}$$

and the mean prediction error is denoted by:

$$\Delta(\tau, y, 0) = \frac{1}{n}\sum_{i=n+1}^{2n} y(i) - \frac{1}{n}\sum_{i=1}^{n} y(i) \tag{5}$$

The third argument in Δ is the predictor type.

DP type 1 is based on estimations calculated over a fixed interval (independent of τ).

$$D_1{}^E = \frac{1}{m}\sum_{i=1}^{m} y(i) \tag{6}$$

$$\Delta(\tau, y, 1) = \frac{1}{n}\left(\sum_{i=m+1}^{m+n} y(i) - \frac{1}{m}\sum_{i=1}^{m} y(i) \right) \tag{7}$$

DP type 2 is based on the Kalman filter estimation; this calculation requires definition of a nominal model. The nominal model assumed in this paper is a first-order Gaussian Markov process with additive white noise. To describe ${D_2}^E$ as a function of data samples, some analysis is needed. For Kalman filter implementation, the measurement is defined by $y(i) = x(i) + v(i)$; $x(i)$ is the first-order Gaussian Markov process, given by $x(i) = \alpha_f x(i-1) + w(i-1)$, and $v(i), w(i)$ are measurement and process random noise processes, respectively. The Kalman filter iterations (see [8] for notations) are:

$$\begin{aligned} x_e^{-}(i) &= a_f x_e(i-1) \\ x_e(i) &= x_e^{-}(i) + k\left(y(i) - x_e^{-}(i) \right) \end{aligned} \tag{8}$$

$x_e(i)$ is the estimate, k is the Kalman gain.

Combining them:

$$x_e(i) = (1-k)\alpha_f x_e(i-1) + ky(i) = \mu x_e(i-1) + ky(i) \tag{9}$$

with $\mu = (1-k)\alpha_f$. In the context of this paper, the Kalman filter gain k is taken as its steady-state value. See the Appendix for the steady-state Kalman filter gain calculation in this case. By inserting the above iteration for $i = 1, 2, ..m$, using $x_e(0) = 0$, we get:

$${D_2}^E = x_e(m) = k\sum_{i=1}^{m} \mu^{m-i} y(i) \tag{10}$$

and therefore

$$\Delta(\tau, y, 2) = \frac{1}{n}\left(\sum_{i=m+1}^{m+n} y(i) - nk\sum_{i=1}^{m} \mu^{m-i} y(i) \right) \tag{11}$$

DP type 3 applies both the optimal estimator and the optimal predictor (for a nominal model). Consider again the prediction error series $\left\{ y(i) - D^E \right\}_{i=m+1}^{m+n}$. For an optimal predictor that applies the underlying system model (first-order Gaussian Markov process with additive white noise), we can calculate the prediction error series as $\left\{ y(i) - {\alpha_f}^{i-m} D^E \right\}_{i=m+1}^{m+n}$ (see [8] for details). In this case, DP type 3 can be calculated as:

$$\Delta(\tau, y, 3) = \frac{1}{n}\left(\sum_{i=m+1}^{m+n} y(i) - k\left(\alpha_f + {\alpha_f}^2 + \cdots + {\alpha_f}^n\right) \sum_{i=1}^{m} \mu^{m-i} y(i) \right) \tag{12}$$

$$\Delta(\tau, y, 3) = \frac{1}{n}\left(\sum_{i=m+1}^{m+n} y(i) - k\alpha_f \frac{1-{\alpha_f}^n}{1-\alpha_f} \sum_{i=1}^{m} \mu^{m-i} y(i) \right) \tag{13}$$

After describing the four types of predictors, we can deduce that all of them obey the same general structure:

$$\Delta(\tau, y) = \frac{1}{n}\left(\sum_{i=m+1}^{m+n} y(i) - \beta \sum_{i=1}^{m} \mu^{m-i} y(i) \right) \tag{14}$$

with the following parameter settings:

DP type 0: $m = n, \mu = 1, \beta = 1$

DP type 1: $\mu = 1, \beta = \frac{n}{m}$

DP type 2: $\mu = (1-k)\alpha_f, \beta = nk$

DP type 3: $\mu = (1-k)\alpha_f, \beta = k\alpha_f \frac{1-{\alpha_f}^n}{1-\alpha_f}$

Note that since all DPs satisfy the same structure, they are equivalent in terms of computational complexity.

3 Direct Predictor Analysis

In this section we take the general structure of the mean prediction error described in Eq. (14) and insert the structure of the signal generated by the nominal system model that consists of the first-order Gaussian Markov process with additive white noise:

$$\begin{aligned} y(i) &= x(i) + v(i) \\ x(i) &= \alpha x(i-1) + w(i-1) \end{aligned} \tag{15}$$

where the sensor additive $v(i)$ is white noise (which after integration causes random walk) and $w(i)$ is process noise related to the Markov process. First we will develop an equation for the mean prediction error $\Delta(\tau, y)$ for this particular signal. Then its covariance will be calculated to provide DP values (for infinite data length).

Since the predictor is a linear operator and our signal $y(i)$ can be considered the sum of two signals $x(i), v(i)$, we can analyze them separately. We start with the Markov process.

One can solve the recursion defined in Eq. (15) to get:

$$x(i) = \alpha^{i-1}x(1) + \alpha^{i-2}w(1) + \alpha^{i-3}w(2) + \ldots + \alpha w(i-2) + w(i-1) \quad (16)$$

Subjecting Eq. (16) to a straightforward but relatively long and involved algebraic manipulation, the following expression is obtained:

$$\sum_{i=1}^{m} \mu^{m-i} x(i) = \mu^{m-1} S\left(\frac{\alpha}{\mu}, m\right) x(1) + \sum_{i=1}^{m-1} \mu^{m-1-i} S\left(\frac{\alpha}{\mu}, m-i\right) w(i) \quad (17)$$

where $S(q,n)$ is a sum of geometric series with ratio q, n elements and 1 as the first element.

$$S(q,n) = 1 + q + \cdots + q^{n-1} = \frac{1-q^n}{1-q} \quad (18)$$

Note that $S(q,1) = 1$.

Similarly, one can show that

$$\sum_{i=m+1}^{m+n} x(i) = S(\alpha,n)\, x(m+1) + \sum_{i=m+1}^{m+n-1} S(\alpha, m+n-i) w(i) \quad (19)$$

Now we need to find a substitution for $x(m+1)$, which is

$$x(m+1) = \alpha^{m-1}x(1) + w(m) + \sum_{i=1}^{m-1} \alpha^{m-i} w(i) \quad (20)$$

Finally, after omitting some details which are straightforward but quite involved, we get the following general structure for the mean prediction error of the Markov process for all types of DPs:

$$\begin{aligned} \Delta(\tau, x) = {} & \frac{1}{n}\left\{ S(\alpha,n)\alpha^m - \beta\mu^{m-1} S\left(\frac{\alpha}{\mu}, m\right) \right\} x(1) + \\ & + \frac{1}{n} \sum_{i=1}^{m-1} \left[S(\alpha,n)\alpha^{m-i} - \beta\mu^{m-1-i} S\left(\frac{\alpha}{\mu}, m-2-i\right) \right] w(i) + \\ & + \frac{1}{n} \sum_{i=m}^{m+n-1} S(\alpha, m+n-i) w(i) \end{aligned} \quad (21)$$

From Eq. (21), we can calculate half the variation of the mean prediction error; its root square is the DP value for infinite data length, and in particular, for type 0 we will have the exact AVAR expression for the Markov process.

Recall that $x(1), w(i)\ i = 1, 2, \cdots, m+n-1$ are independent. $x(1)$ has variance ${S_m}^2$ and $w(i)\ i = 1, 2, \ldots, N$ has covariance ${S_w}^2 = {S_m}^2(1-\alpha^2)$. Using this information, the covariance calculation is direct.

$$\sigma_x^{\,2}(\tau)=\frac{1}{2}Var\left(\Delta(\tau,x)\right)=\frac{1}{2n^2}\left\{S(\alpha,n)\alpha^m-\beta\mu^{m-1}S\left(\frac{\alpha}{\mu},m\right)\right\}^2 S_m^{\,2}+$$
$$+\frac{1}{2n^2}\sum_{i=1}^{m-1}\left[S(\alpha,n)\alpha^{m-i}-\beta\mu^{m-1-i}S\left(\frac{\alpha}{\mu},m-2-i\right)\right]^2 S_w^{\,2}+ \quad (22)$$
$$+\frac{1}{2n^2}\sum_{i=m}^{m+n-1}S(\alpha,m+n-i)^2 S_w^{\,2}$$

Using $S_w^{\,2}=S_m^{\,2}\left(1-\alpha^2\right)$ we get the general formula for DP values (for infinite data length) for the first-order Gaussian Markov process:

$$\sigma_x^{\,2}(\tau)=\frac{S_m^{\,2}}{2n^2}\left[\begin{array}{l}\left(S(\alpha,n)\alpha^m-\beta\mu^{m-1}S\left(\frac{\alpha}{\mu},m\right)\right)^2+\\ +\left(1-\alpha^2\right)\sum_{i=1}^{m-1}\left[S(\alpha,n)\alpha^{m-i}-\beta\mu^{m-1-i}S\left(\frac{\alpha}{\mu},m-2-i\right)\right]^2+\\ +\left(1-\alpha^2\right)\sum_{i=m}^{m+n-1}S(\alpha,m+n-i)^2\end{array}\right] \quad (23)$$

To complete the calculation for our process, we need to calculate the DP for v, the white noise component. We start with Eq. (14) for this case:

$$\Delta(\tau,v)=\frac{1}{n}\left(\sum_{i=m+1}^{m+n}v(i)-\beta\sum_{i=1}^{m}\mu^{m-i}v(i)\right) \quad (24)$$

Recall that $v(i)$ is independent with variance $S_v^{\,2}$; the square of the DP values (half covariance) for white noise is given as:

$$\sigma_v^{\,2}(\tau)=\frac{S_v^{\,2}}{2n^2}\left(\sum_{i=m+1}^{m+n}1+\beta^2\sum_{i=1}^{m}\mu^{2(m-i)}\right)=\frac{S_v^{\,2}}{2n^2}\left(n+\beta^2 S(\mu^2,m)\right) \quad (25)$$

Now, since the Markov process and white noise components are independent, we can combine Eqs. (23) and (25) to obtain:

$$\sigma_y(\tau)=\sqrt{\sigma_x^{\,2}(\tau)+\sigma_v^{\,2}(\tau)} \quad (26)$$

Eq. (26), together with Eqs. (23) and (25), provide a closed-form solution for $\sigma_y(\tau)$ of the first-order Gaussian Markov process with additive white noise. The model parameters are S_m, S_v - the standard deviations of the Markov process and

white noise components, respectively; the third model parameter is α, related to the Markov process time constant t_m by $\alpha = \exp(-dt / t_m)$, where dt is sampling time. The other parameters, $m.\mu, \beta$, are related to DP structure (type and parameters). The parameter n is related to the argument τ by $n = \tau / dt$.

Note that the notation $\sigma_y(\tau,...)$ is used in this paper for DPs calculated for infinite data length by Eqs. (26), (23), and (24). The DP value calculated for final data length will be denoted $DP(\tau,...)$.

4 Direct Predictor Errors Due to Finite Data Length

We begin by describing how the DP for finite data length is calculated, using a non-overlapping approach. The total number of samples is denoted by N; for any τ we divide the data sample into N_τ groups (windows), every window with $m+n$ elements, and every group starting with the index n_k, such that $n_1 = 1, n_2 = m+n+1, n_3 = 2(m+n)+1, ...n_{N_\tau} = (N_\tau - 1)(m+n)-1$. The mean prediction error, calculated for window k, is defined by:

$$\Delta(\tau, y, k) = \frac{1}{n}\left(\sum_{i=n_k+m}^{n_k+m+n-1} y(i) - \beta \sum_{i=n_k}^{n_k+m-1} \mu^{m-i} y(i) \right) \tag{27}$$

The square of DP, denoted DQ, is estimated as follows:

$$DQ(\tau, y, N) = DP(\tau, y, N)^2 = \frac{1}{2N_\tau} \sum_{k=1}^{N_\tau} \Delta(\tau, y, k)^2 \tag{28}$$

To compare DQ with a Chi-square distribution, we make the following normalization

$$DQ(\tau, y, N) = \frac{2\sigma_y(\tau)^2}{2N_\tau} \sum_{k=1}^{N_\tau} \left(\frac{\Delta(\tau, y, k)}{\sqrt{2}\sigma_y(\tau)} \right)^2 \tag{29}$$

If $\Delta(\tau, y, k)\ \ k = 1, 2, ..., N_\tau$ is independent, $\sum_{k=1}^{N_\tau} \left(\frac{\Delta(\tau, y, k)}{2\sigma_y(\tau)} \right)^2$ has a Chi-squared distribution with N_τ degrees of freedom. In this case, the mean is N_τ and variation is $2N_\tau$ (see [9]). Therefore, the mean of $DQ(\tau, y, N)$ is $\sigma_y(\tau)^2$ and its

variance is $\frac{2\sigma_y(\tau)^4}{N_\tau}$. Moreover, for large N_τ, a Gaussian distribution for $DQ(\tau, N, y)$ can be assumed.

Similarly DP, which is the square root of DQ, can be described using the Chi distribution of $\sqrt{\sum_{k=1}^{N_\tau}\left(\frac{\Delta(\tau, y, k)}{\sqrt{2}\sigma_y(\tau)}\right)^2}$ with mean $m_p = \sqrt{2}\frac{\Gamma\left(0.5\left(N_\tau+1\right)\right)}{\Gamma\left(0.5N_\tau\right)}$ and variance $\sigma_p{}^2 = N_\tau - m_p{}^2$.

$\Gamma(x)$ denotes the gamma function. For the purpose of the analysis required here, we will avoid the use of gamma functions, mainly because we found a that linear approximation provides a simpler and satisfactory result. The following analysis is then performed.

$$y_0 = \sqrt{x_0}, \quad y_0 + \delta y \approx \sqrt{x_0} + \frac{1}{2\sqrt{x_0}}\delta x \tag{30}$$

In our case $x_0 = \sigma_y(\tau)^2, y_0 = \sigma_y(\tau)$ are the mean values of DQ, DP, and $\delta x, \delta y$ are the standard deviations of DQ, DP, respectively. To conclude the above discussion:

$$mean\left(DP\left(\tau, y, N\right)\right) = \sigma_y(\tau) \tag{31}$$

$$std\left(DP\left(\tau, y, N\right)\right) = \frac{1}{\sqrt{2N_\tau}}\sigma_y(\tau) \tag{32}$$

Eqs. (31) and (32) provide us with the error analysis due to the final data length that we were looking for. However, before we can accept these results we need to examine our assumptions:

The sequence $\Delta(\tau, y, k)$ is assumed to be independent, due to the underlying Markov process; in general, this is not true. However, the mean dependence is expected to be weak because the dependence exists only for adjacent windows and the number of windows is large.

The linear approximation for square root should be valid; this means that the standard deviation of DQ should be much smaller than its mean value, and indeed, it is valid for large N_τ.

The final verification of this assumption was carried out by simulation. We considered a variety of systems and our conclusion was that for a collection time of one hour, $dt = 1, 1 \le \tau \le 300$, this approximation provides poor results. If the collection time is on the order of two hours or more and the quality of the sensors is in the range of $0.1 \div 100\ ^\circ / h$, Eqs. (31) and (32) are certainly valid. The following plot compares the values calculated by Eqs. (31) and (32) (precise) with the results of 500 Monte Carlo runs, each describing a two-hour data collection. The presented case is for DP type 3, and instrument model $S_m = 100^\circ / h, S_v = 1^\circ / \sqrt{h}, t_m = 25\,\text{sec}$

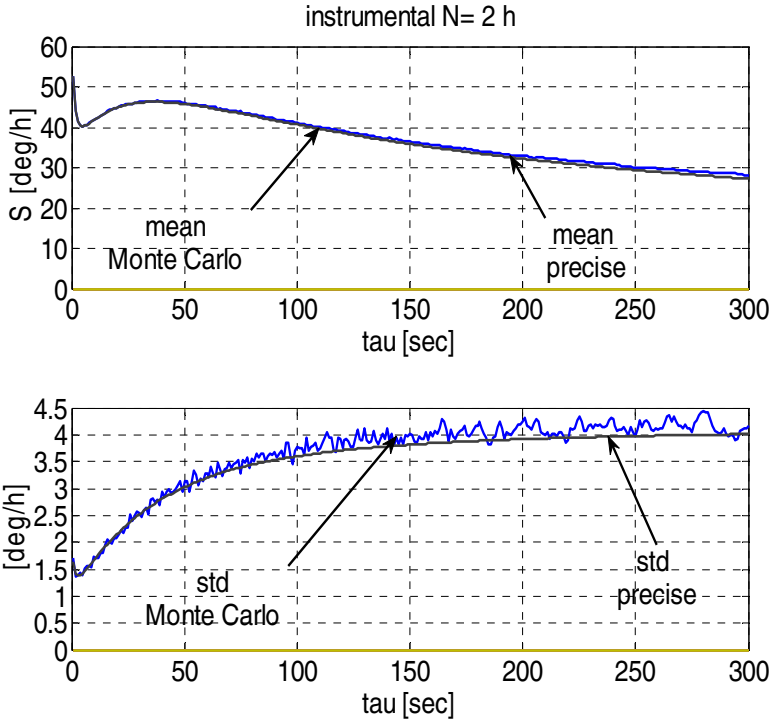


Fig. 1. Errors due to finite length: instrumental system

Figure 2 describes the same results for tactical level system $S_m = 1^\circ / h, S_v = 0.05^\circ / \sqrt{h}, t_m = 100\,\text{sec}$.

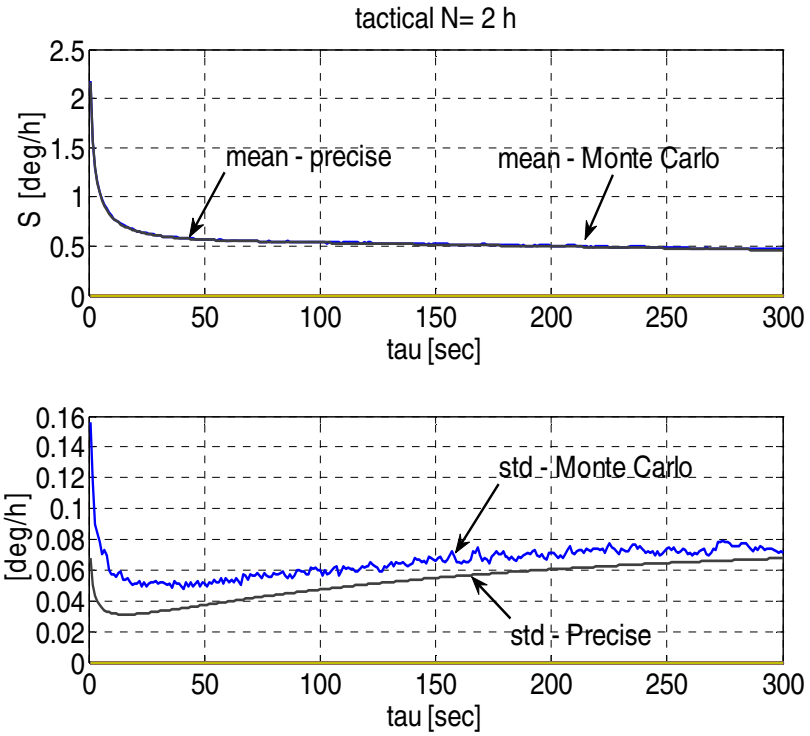


Fig. 2. Errors due to finite length: tactical system

5 Optimization Techniques for Direct Bound Implementation

Given DP values calculated from the real data, $DP(\tau)$ (arguments of the signal y and data length N are omitted), we are looking for parameters S_m, S_v, t_m such that the model $M_D(S_m, S_v, t_m)$ will generate $\sigma(\tau, M_D, T)$ to be a tight bound for $DP(\tau)$. Observe that the notation for $\sigma_y(\tau)$ was modified to $\sigma(\tau, M_D, T)$, with T representing the DP type (with its parameters). Where the DP type is not important, the notation $\sigma(\tau, M_D)$ will be used as well.

Let us write the formal problem definition. We consider a discrete sequence of $\{\tau_i\}_{i=1}^{K_\tau} = \tau_{\min}, \tau_{\min} + d\tau, \tau_{\min} + 2d\tau, ..., \tau_{\max}$. $DP(\tau_i)$ and we need to solve the following optimization problem:

$$\min_{S_m, S_v, t_m} \left\{ \sum_{i=1}^{K_\tau} \left(\sigma(\tau_i, M_D(S_m, S_v, t_m)) - DP(\tau_i) \right)^2 \right\} \tag{33}$$

$$subject\ to\ \sigma\left(\tau_i, M_D\left(S_m, S_v, t_m\right)\right) \geq DP(\tau_i)\ \ i = 1, 2, ..., K_\tau \tag{34}$$

For this problem, we were surprised to get poor, unstable results with a significant number of outlier solutions, even with different optimization methods. We thus understood that some modifications in the above definition are required. In the following we describe the modification required to get stable solutions.

The physical values of S_m, S_v are very small. To get a reliable optimization method we need to introduce proper scaling. The choice is to work with a vector of engineering units: $^\circ/h, ^\circ/\sqrt{h}, \sec$ and to introduce the following notation:

$$M_D(S_m, S_v, t_m) = M_D{}^S\left(X\right) \text{ with } X(1) = s_1 S_m, X(2) = s_2 S_v, X(3) = t_m$$

such that $s_1 = \frac{180}{\pi} 3600, s_2 = \frac{180}{\pi}\sqrt{3600}$.

Now observe that the minimization defined in Eq. (33) is related to the norm of residuals for the following equation:

$$DP(\tau_i) = \sigma_y\left(\tau_i, M_D{}^S\left(X^*\right)\right) + v(\tau_i) \tag{35}$$

where X^* represents the true model parameters and $v(\tau_i)$ is the error due to finite data length. We know that standard deviation of this error is given by $std\left(DP(\tau)\right) = \frac{1}{\sqrt{2N_\tau}}\sigma_y(\tau)$. Therefore, to deal with normalized residuals we need to consider

$$\min_{S_m, S_v, t_m}\left\{\sum_{i=1}^{K_\tau}\left(r_i\right)^2\right\} \tag{36}$$

$$r_i = \sqrt{2N_\tau}\,\frac{\sigma_y\left(\tau_i, M_D{}^S\left(X\right)\right) - DP(\tau_i)}{\sigma_y\left(\tau_i, M_D{}^S\left(X\right)\right)} \tag{37}$$

We can define a penalty factor ρ and combine the constraints defined in Eq. (34) into the minimization defined in Eq. (36) by replacing r_i by its bounded version r_i^B:

$$r_i^B = \begin{cases} r_i & if \quad r_i > 0 \\ \rho r_i & else \end{cases} \tag{38}$$

The penalty factor ρ is selected to ensure the condition:

$$\sigma_y\left(\tau_i, M_D{}^S\left(X^*\right)\right) \geq DP(\tau_i) \tag{39}$$

Moreover, ρ can provide flexibility from hard bound (high ρ), via soft bound (moderate ρ) to best match (standard least square: $\rho = 1$). The optimization problem defined by the cost function described in Eqs. (36–39) was tested for a variety of cases and reliable, stable results were obtained.

Having developed a robust optimization method, we approached the problem of DP selection and the accuracy of the estimated model parameters. Our first test was for the instrumental system, with parameters: $S_m = 100°/h, S_v = 1°/\sqrt{h}, t_m = 25\,\text{sec}$. The test duration was two hours, sampling time $dt = 1\text{sec}$ and the range of prediction time τ was $1 \le \tau \le 300$, termed in the sequel tau range. The following plot provides insight into how the optimization works. We present four plots:

- DP – a sample of DPs calculated from sample data—actual
- $\sigma_y\left(\tau, M_D^{\,*}\right)$ – DP calculation based on the true model—precise
- $\sigma_y\left(\tau, M_D^{\,EH}\right)$ – DP calculated using the model found by hard bound optimization, $\rho = 100$ —hard bound.
- $\sigma_y\left(\tau, M_D^{\,ES}\right)$ – DP calculated using the model found by soft bound optimization, $\rho = 10$ —soft bound.

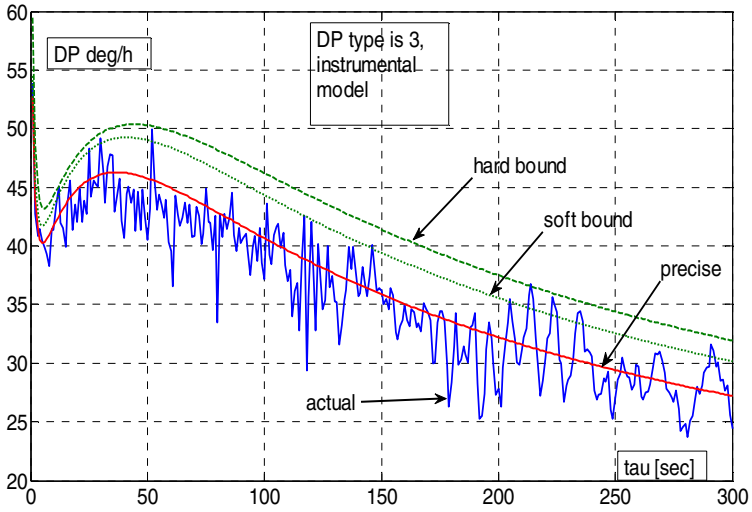


Fig. 3. DPs found by the optimization algorithm

The following table summarizes the results of 100 Monte Carlo runs. For every run, we calculated the estimated parameters for all four types of DP. After all runs, mean value and standard deviation (std) were calculated for each type. The value "total rel" is rss (root sum square) of the mean and standard deviation, divided by the nominal parameter. Using this table we can evaluate performance and compare the accuracy of each DP type.

Table 1. The errors of estimated model parameters: instrumental system

The estimated model parameters errors				DP type			
				0	1	2	3
HARD BOUND	S_m	$\circ / h$	mean	22	27.8	45.2	12.6
			std	6.1	8.7	51.0	4.8
			total rel	*22%*	*29%*	*68%*	*13%*
	S_v	$\circ / \sqrt{h}$	mean	0.02	0.34	0.10	0.07
			std	0.04	0.4	0.2	0.1
			total rel	*4%*	*52%*	*22%*	*12%*
	t_m	sec	mean	10.3	22.5	30.8	5.7
			Std	5.7	12.8	36.5	5.6
			total rel	*47%*	*103%*	*191%*	*32%*
SOFT BOUND	S_m	$\circ / h$	mean	15.1	17.1	23.4	8.7
			std	5.2	6.6	6.7	4.2
			total rel	*16%*	*18%*	*24%*	*9.6%*
	S_v	$\circ / \sqrt{h}$	mean	0.01	0.21	0.08	0.05
			std	0.03	0.24	0.08	0.08
			total rel	*3%*	*31%*	*11%*	*9%*
	t_m	sec	mean	6.8	12.5	14.4	3.6
			std	4.6	8.0	6.8	4.8
			total rel	*33%*	*59%*	*63%*	*24%*

The following observations can be made from the table:

- The mean of estimated errors shows that the estimators are strongly biased. After some consideration, this should not be a surprise. The nature of bounding is that it provides higher DP plots than the actual one. The bias estimator is the price for our attempt to bound the performance of a system that is not necessarily time-invariant. Of course, we can reduce this price by allowing some crossing of the actual DP (soft bound).
- In principle, we have three different quality criteria: the accuracies of S_m, S_v, t_m. The best DP in one category is not necessarily the best one in another.

- In the case presented here, DP type 3 is the best for the Markov process. Its errors for Markov process parameters are about 50% better than the second-best DP, which is type 0, the AVAR. For white noise the AVAR is better, but DP type 3 can be considered satisfactory with 10% accuracy.
- DP types 1 and 2 show relatively poor performance in this case.
- In general, for soft bound optimization, two hours of data collection to get 10% accuracy in S_m, S_v and 25% in the time constants seems to be a very efficient and precise approach to error model construction.

6 Recommendations for Practical Applications and Summary

Good engineering practice requires answering the following questions:

1. How should the error model parameters be calculated?
2. How should the test be designed to obtain satisfactory accuracy of these parameters?
3. How should the test be monitored and sensor malfunctions detected?

In this paper, tools to deal with all of these questions were developed and presented.

With respect to #1, our position is clear: never use local slopes, use matching techniques, such as Eqs. (36–38). The decision of whether to use hard bound, soft bound or even best matching depends on the application tradeoff between the estimation accuracy and sensitivity to detecting outliers with respect to the time-invariant model. In the preparation phase, one can analyze what kind of accuracy degradation is related to higher outlier detection. The preparation phase, which is based on simulations, only provides the right answers for #2. In this phase, we need to select the proper collection time, range of tau, kind of matching used by the estimation (hard bound, soft bound, best matching), and type of DP to be used for real data. Our recommendation is to select the minimum collection time, the maximum tau range, and the hardest bound that provides satisfactory estimation accuracy. The minimum collection time saves costs, the maximum tau range and hardest bound provide good detectability for outliers. Of course we will select the best DP for the case. Recall that beside freedom of selection from four different types, there also exists freedom in DP parameter selection. Figure 4 describes the flow of the simulation, applied for test design.

After defining the test parameters, the real-data test appears straightforward; it is such in the path of model parameter estimations. An additional path of quality of matching, which measures the residuals of the optimization function and compares them with the statistics of errors due to finite length (Eq. 32) provides some, perhaps partial, answers to problem #3.

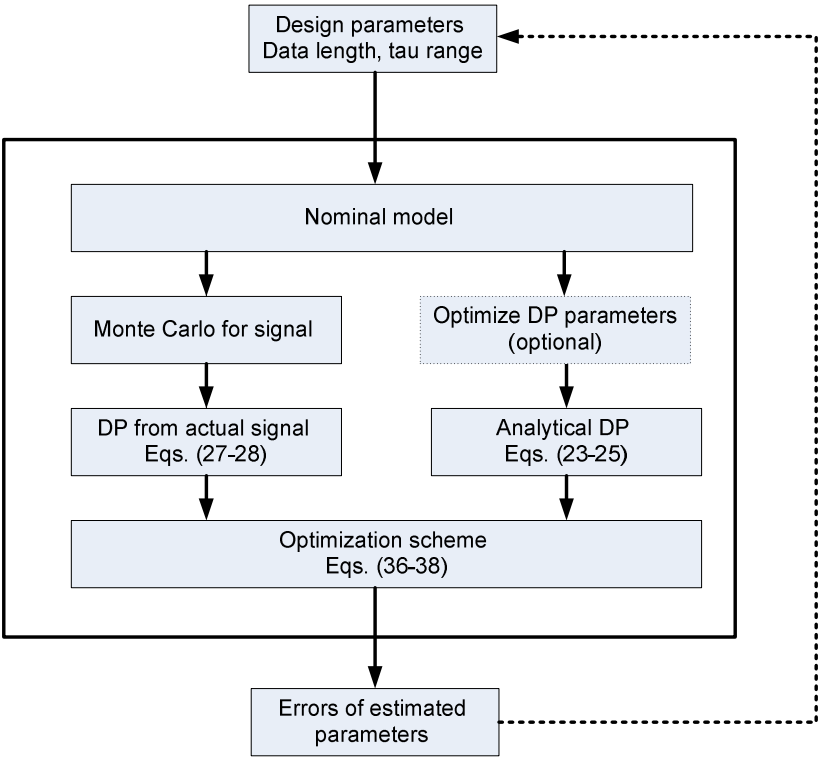


Fig. 4. Diagram block for simulation phase

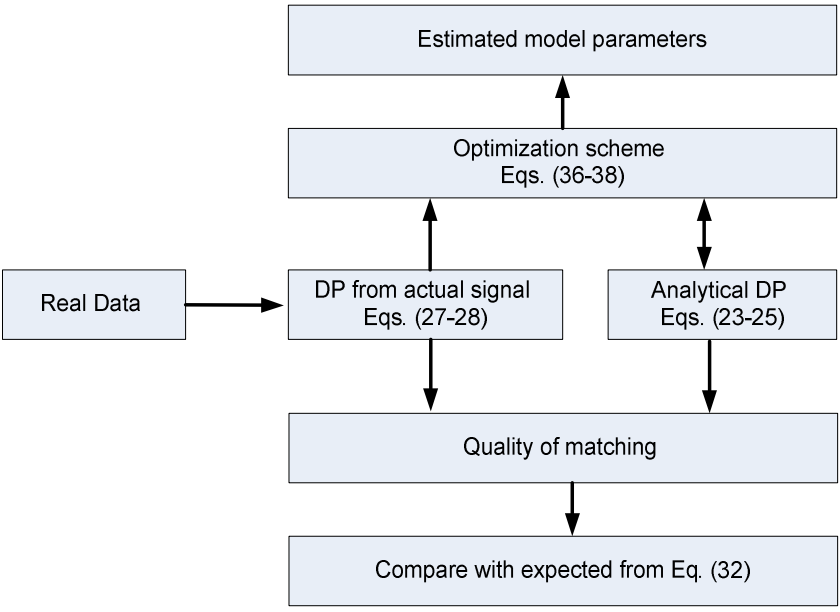


Fig. 5. Block diagram for real-data test

Appendix

Calculation of the Steady-State Kalman Gain. The Development Here is Based on Notations and the Equation Presented in [8].

The measurement matrix is scalar equal to 1:

$$z(i) = y(i) + v(i) \Rightarrow H = 1 \tag{40}$$

The propagation system matrix is scalar equal to α

$$y(i) = \alpha y(i-1) + w(i-1) \Rightarrow \Phi = a \tag{41}$$

The covariance of measurement noise error $v(i)$ is r and the covariance of process noise $w(i)$ is q.

$$K_k = P_k^- H^T \left(H P_k^- H^T + R \right)^{-1} \Rightarrow k_k = \frac{p_k^-}{p_k^- + r} \tag{42}$$

$$P_k^+ = \left(I - K_k H \right) P_k^- \Rightarrow p_k^+ = (1 - k_k) p_k^- \tag{43}$$

$$P_{k+1}^- = \Phi P_k^+ \Phi^T + Q \Rightarrow p_{k+1}^- = \alpha^2 p_k^+ + q \tag{44}$$

The steady-state gain is denoted by k. Under steady-state conditions $p_{k+1}^- = p_k^- = p$.

$$p = \alpha^2 (1-k) p + q \tag{45}$$

Using

$$k = \frac{p}{p+r} \Rightarrow (1-k) = \frac{r}{p+r} \tag{46}$$

one gets

$$p^2 + \left(r - \alpha^2 r - q \right) p - qr = 0 \tag{47}$$

The positive solution of Eq. (47) is given by

$$p^* = \frac{r - \alpha^2 r - q}{2} + \sqrt{\left(\frac{\left(r - \alpha^2 r - q \right)^2}{4} + qr \right)} \tag{48}$$

The final solution is obtained by inserting Eq. (48) into Eq. (46)

$$k = \frac{p^*}{p^* + r} \tag{49}$$

References

1. Aggarwal, P., Syed, Z., Niu, X., El-Sheimy, N.: A Standard Testing and Cali-bration Procedure for Low Cost MEMs Inertial Sensors and Units. Journal of Navigation 61, 323–336 (2008)
2. Titterton, D.H., Weston, J.L.: Strapdown Inertial Navigation Technology, Peter Peregrinus Ltd., UK (1997)
3. Flenniken IV, W.S., Wall, J.H., Bevly, D.M.: Characterization of Various IMU Error Sources and the Effect on Navigation Performance. ION GNSS (2005)
4. IEEE. IEEE Standard Specification Format Guide and Test Procedure for Single-Axis Interferometric Fiber Optic Gyros. IEEE Std 952-1997 (2003)
5. Xing, Z., Gebre-Egziabher, D.: Modeling and Bounding Low Cost Inertial Sensor Errors. In: Proceedings of IEEE/ION PLANS 2008, May 6-8 (2008)
6. Berman, Z.: Inertial Sensors—Further Developments in Low Cost Calibration and Testing. In: PLANS 2012 (April 2012)
7. Berman, Z.: Inertial sensors—A New Approach for Low Cost Calibration and Testing. In: Inertial Sensors and Systems Gyro Technology Symposium, Karlsruhe (2011)
8. Gelb, A.: Applied Optimal Estimation. M.I.T. Press, Cambridge (1974)
9. Papoulis, A.: Probability, Random Variables and Stochastic Processes, 2nd edn. McGraw-Hill, New York (1984)

Time-Critical Cooperative Path Following of Multiple UAVs: Case Studies

Isaac Kaminer[1], Enric Xargay[2], Venanzio Cichella[2], Naira Hovakimyan[2], António Manuel Pascoal[3], A. Pedro Aguiar[4], Vladimir Dobrokhodov[1], and Reza Ghabcheloo[5]

[1] Naval Postgraduate School, Monterey, CA 93943, USA
{kaminer,vldobr}@nps.edu
[2] University of Illinois at Urbana-Champaign, Urbana, IL 61801, USA
{xargay,cichell2,nhovakim}@illinois.edu
[3] Instituto Superior Técnico, University of Lisbon, LX 1049-001, Portugal
antonio@isr.ist.utl.pt
[4] Faculty of Engineering, University of Porto, P.O. 4200, Portugal
pedro.aguiar@fe.up.pt
[5] Tampere University of Technology, Tampere, FI-33101, Finland
reza.ghabcheloo@tut.fi

Abstract. This paper describes a multi-vehicle motion control framework for time-critical cooperative missions and evaluates its performance by considering two case studies: a simultaneous arrival mission scenario and a sequential auto-landing of a fleet of UAVs. In the adopted setup, the UAVs are assigned nominal spatial paths and speed profiles along those paths; the vehicles are then tasked to execute cooperative path following, rather than "open-loop" trajectory tracking. This cooperative strategy yields robust behavior against external disturbances by allowing the UAVs to negotiate their speeds along the paths in response to information exchanged over a supporting communications network.

1 Introduction

Unmanned Aerial Vehicles (UAVs) are ubiquitous in military reconnaissance and strike operations, border patrol missions, forest fire detection, and recovery operations. In simple missions, a single vehicle can be managed by a crew using a ground station provided by the vehicle manufacturer. The execution of more challenging missions, however, requires the use of multiple vehicles working in cooperation to achieve a common objective. Such missions require vehicles to execute maneuvers in close proximity to each other, and to effectively exchange information so as to meet desired spatial and temporal constraints. The flow of information among vehicles is often severely restricted, either for security reasons or because of tight bandwidth limitations. A key enabling element for the execution of such missions is thus the availability of cooperative motion control strategies that can yield robust performance in the face of external disturbances and communications limitations, while ensuring collision-free maneuvers.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_12

The range of relevant, related topics addressed in literature includes parallel computing [1], synchronization of oscillators [2], study of collective behavior and flocking [3], multi-system consensus mechanisms [4], multi-vehicle system formations [5–8], coordinated motion control [9–11], cooperative path and trajectory planning [12–15], asynchronous protocols [16], dynamic graphs [17], stochastic graphs [17–19], and graph-related theory [20,21]. Especially relevant are the applications of the theory developed in the area of multi-vehicle control: spacecraft formation flying [22], UAV control [23,24], coordinated control of land robots [9], and control of multiple autonomous marine vehicles [25–30]. In spite of significant progress in the field, much work remains to be done to develop strategies capable of providing guaranteed levels of performance in the presence of complex vehicle dynamics, communications constraints, and partial vehicle failures.

In [31], we addressed the problem of *steering a fleet of UAVs along desired spatial paths while meeting relative temporal constraints*. Representative examples of such missions are sequential auto-landing and coordinated ground target suppression; in both cases, only *relative* –rather than *absolute*– temporal constraints are given a priori. In the proposed framework, the vehicles are assigned nominal paths and speed profiles along those, obtained from an appropriately formulated optimization problem. The paths are judiciously parameterized, and the vehicles are requested to execute cooperative path following, rather than "open-loop" trajectory-tracking maneuvers. The reader is referred to [31–35] for a general perspective of key ideas that are at the root of this distributed cooperative approach. In the present paper, we present simulation results of two multi-vehicle time-critical missions that exploit the cooperative control framework developed in [31]. In the first mission, three UAVs must follow spatially-deconflicted paths and arrive at predefined locations at the same time. The second mission considers the case of sequential auto-landing, in which three UAVs must arrive at the glide path separated by prespecified safe-guarding time-intervals and maintain this separation as they fly along the glide slope.

The paper is organized as follows. Section 2 formulates the time-critical cooperative path-following problem and introduces a set of assumptions on the supporting network. Section 3 presents a path-following control algorithm for UAVs in 3D space. Section 4 derives a strategy for time-critical cooperative path following of multiple UAVs that relies on the adjustment of the speed profile of each vehicle. Section 5 presents simulation results that demonstrate the effectiveness of the algorithms. Finally, Section 6 summarizes concluding remarks.

The following notation is used throughout the paper. Uppercase calligraphic letters are used to denote reference frames, e.g. $\mathcal{F}$; $\{\boldsymbol{v}\}_F$ is used to denote vector $\boldsymbol{v}$ resolved in frame $\mathcal{F}$; $\{\hat{\boldsymbol{e}}\}_F$ represents versor $\hat{\boldsymbol{e}}$ resolved in frame $\mathcal{F}$; $\boldsymbol{\omega_{F1/F2}}$ denotes the angular velocity of frame $\mathcal{F}1$ with respect to frame $\mathcal{F}2$; the rotation matrix from frame $\mathcal{F}1$ to frame $\mathcal{F}2$ is represented by $\boldsymbol{R}_{\boldsymbol{F1}}^{\boldsymbol{F2}}$; $\dot{\boldsymbol{v}}\,]_F$ indicates that the time-derivative of vector $\boldsymbol{v}$ is taken in frame $\mathcal{F}$. The notation $\|\cdot\|$ is used for the 2-norm of a vector. Finally, $\mathsf{SO(3)}$ denotes the Special Orthogonal group of all rotations about the origin of three-dimensional Euclidean space $\mathbb{R}^3$, while $\mathfrak{so}(3)$ represents the set of 3×3 skew-symmetric matrices over $\mathbb{R}$.

2 Problem Formulation

This section formulates in a concise manner the problem of time-critical cooperative path-following control of multiple UAVs in 3D space, in which a fleet of UAVs is tasked to converge to and follow a set of desired feasible paths so as to meet spatial and temporal constraints. The section also introduces a set of assumptions and constraints on the supporting communications network.

We note that the problem of cooperative *trajectory generation* is not addressed in this paper. In fact, it is assumed that a set of desired 3D time-trajectories $\boldsymbol{p_{d,i}}(t_d) : [0, t_d^*] \to \mathbb{R}^3$, $i = 1, \ldots, n$, conveniently parameterized by a single time-variable t_d, is known for all the n UAVs involved in the mission. The variable t_d represents a *desired mission time* (distinct from the actual mission time that evolves as the mission unfolds), with t_d^* being the *desired mission duration*. For a given t_d, $\boldsymbol{p_{d,i}}(t_d)$ defines the desired position of the ith UAV t_d seconds after the initiation of the mission. These time-trajectories can be reparameterized in terms of arc length to obtain *spatial paths* $\boldsymbol{p_{d,i}}(\tau_{\ell,i}) : [0, \ell_{fi}] \to \mathbb{R}^3$ –with no temporal specifications– and the corresponding *desired speed profiles* $v_{d,i}(t_d) : [0, t_d^*] \to \mathbb{R}$. For convenience, each spatial path is parameterized by its arc length $\tau_{\ell,i}$, with ℓ_{fi} denoting the total length of the ith path, whereas the desired speed profiles are parameterized by the desired mission time t_d. It is assumed that both the paths and the speed profiles satisfy collision-avoidance constraints as well as appropriate boundary and feasibility conditions, such as those imposed by the physical limitations of the UAVs. The problem of generating feasible time-critical trajectories for multiple vehicles is described in [36, 37].

2.1 Path Following for a Single UAV

The solution to the path-following problem described in this paper extends the algorithm in [38] to the 3D case, and relies on the insight that a UAV can follow a given path using only its attitude, thus leaving its linear speed as a degree of freedom to be used at the coordination level. Following the approach developed in [38], this section introduces a *virtual target vehicle* running along the 3D path, defines a frame attached to this virtual target, and characterizes a generalized error vector between this moving coordinate system and a frame attached to the actual UAV. With this setup, the path-following problem is reduced to driving this generalized error vector to zero by using only the UAV's attitude control effectors, while following an arbitrary feasible speed profile.

Figure 1 captures the geometry of the problem at hand. The symbol $\mathcal{I}$ denotes an inertial reference frame $\{\hat{\mathbf{e}}_\mathbf{1}, \hat{\mathbf{e}}_\mathbf{2}, \hat{\mathbf{e}}_\mathbf{3}\}$ and $\boldsymbol{p_d}(\cdot)$ is the desired path assigned to one of the vehicles, with ℓ_f being its total path length. Vector $\boldsymbol{p_I}(t)$ denotes the position of the center of mass Q of the vehicle in this inertial frame. Further, we let P be an arbitrary point on the desired path that plays the role of the virtual target, and denote by $\boldsymbol{p_d}(\ell)$ its position in the inertial frame. Here $\ell \in [0, \ell_f]$ is a free length variable that defines the position of the virtual target vehicle along the path. In the setup adopted, the total rate of progression of the virtual target along the path, $\dot{\ell}(t)$, is an additional design parameter. Endowing point P with

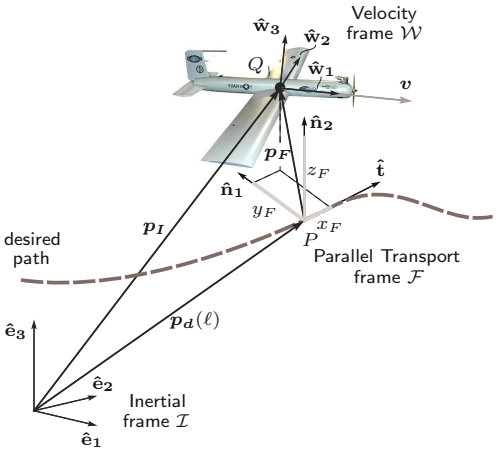


Fig. 1. Following a virtual target vehicle; problem geometry

an extra degree of freedom is the key to the path-following algorithm presented in [38] and its extension to the 3D case described in this paper.

For our purposes, it is convenient to define a *parallel transport frame* $\mathcal{F}$ [39] attached to point P on the path and characterized by vectors $\{\hat{\mathbf{t}}(\ell), \hat{\mathbf{n}}_\mathbf{1}(\ell), \hat{\mathbf{n}}_\mathbf{2}(\ell)\}$. These vectors define an orthonormal basis for $\mathcal{F}$, in which the unit vector $\hat{\mathbf{t}}(\ell)$ defines the tangent direction to the path at the point determined by ℓ, while $\hat{\mathbf{n}}_\mathbf{1}(\ell)$ and $\hat{\mathbf{n}}_\mathbf{2}(\ell)$ define the normal plane perpendicular to $\hat{\mathbf{t}}(\ell)$. Unlike the Frenet-Serret frame, parallel transport frames are well defined when the path has a vanishing second derivative. Moreover, let $\boldsymbol{p_F}(t)$ be the position of the vehicle's center of mass Q in this moving frame, and let $x_F(t)$, $y_F(t)$, and $z_F(t)$ be the components of vector $\boldsymbol{p_F}(t)$ with respect to the basis $\{\hat{\mathbf{t}}, \hat{\mathbf{n}}_\mathbf{1}, \hat{\mathbf{n}}_\mathbf{2}\}$.

Let $\mathcal{W}$ denote a vehicle-carried velocity frame $\{\hat{\mathbf{w}}_\mathbf{1}, \hat{\mathbf{w}}_\mathbf{2}, \hat{\mathbf{w}}_\mathbf{3}\}$ with its origin at the UAV center of mass and its x-axis aligned with the velocity vector of the UAV. The z-axis is chosen to lie in the plane of symmetry of the UAV, and the y-axis is determined by completing the right-hand system. In this paper, $q(t)$ and $r(t)$ are the y-axis and z-axis components, respectively, of the vehicle's rotational velocity resolved in the $\mathcal{W}$ frame. With a slight abuse of notation, $q(t)$ and $r(t)$ will be referred to as *pitch rate* and *yaw rate*, respectively.

We also introduce an auxiliary frame $\mathcal{D}$ defined by $\{\hat{\mathbf{b}}_\mathbf{1D}, \hat{\mathbf{b}}_\mathbf{2D}, \hat{\mathbf{b}}_\mathbf{3D}\}$, which is used to shape the approach attitude to the path as a function of the cross-track error components y_F and z_F. Frame $\mathcal{D}$ has its origin at the UAV center of mass and vectors $\hat{\mathbf{b}}_\mathbf{1D}(t)$, $\hat{\mathbf{b}}_\mathbf{2D}(t)$, and $\hat{\mathbf{b}}_\mathbf{3D}(t)$ are defined as

$$\hat{\mathbf{b}}_\mathbf{1D} := \frac{d\,\hat{\mathbf{t}} - y_F\,\hat{\mathbf{n}}_\mathbf{1} - z_F\,\hat{\mathbf{n}}_\mathbf{2}}{\left(d^2 + y_F^2 + z_F^2\right)^{\frac{1}{2}}} \quad , \qquad \hat{\mathbf{b}}_\mathbf{2D} := \frac{y_F\,\hat{\mathbf{t}} + d\,\hat{\mathbf{n}}_\mathbf{1}}{\left(d^2 + y_F^2\right)^{\frac{1}{2}}} \quad , \qquad \hat{\mathbf{b}}_\mathbf{3D} := \hat{\mathbf{b}}_\mathbf{1D} \times \hat{\mathbf{b}}_\mathbf{2D} \quad ,$$

with $d > 0$ being a constant *characteristic distance* that plays the role of a design parameter. The basis vector $\hat{\mathbf{b}}_\mathbf{1D}(t)$ defines the desired direction of the UAV's velocity vector. Clearly, when the vehicle is far from the desired path,

vector $\hat{\mathbf{b}}_{\mathbf{1D}}(t)$ becomes perpendicular to $\hat{\mathbf{t}}(\ell)$. As the vehicle comes closer to the path and the cross-track error becomes smaller, vector $\hat{\mathbf{b}}_{\mathbf{1D}}(t)$ tends to $\hat{\mathbf{t}}(\ell)$.

Finally, let $\tilde{\boldsymbol{R}}(t) \in \mathsf{SO(3)}$ be the rotation matrix from $\mathcal{W}$ to $\mathcal{D}$, that is,

$$\tilde{\boldsymbol{R}} := \boldsymbol{R}_{\boldsymbol{W}}^{\boldsymbol{D}} = \boldsymbol{R}_{\boldsymbol{F}}^{\boldsymbol{D}}\, \boldsymbol{R}_{\boldsymbol{W}}^{\boldsymbol{F}} = (\boldsymbol{R}_{\boldsymbol{D}}^{\boldsymbol{F}})^{\top}\, \boldsymbol{R}_{\boldsymbol{W}}^{\boldsymbol{F}} \ ,$$

and define the real-valued attitude error function on $\mathsf{SO(3)}$

$$\Psi(\tilde{\boldsymbol{R}}) := \tfrac{1}{2}\mathrm{tr}\big[\big(\mathbb{I}_{\mathbf{3}} - \boldsymbol{\Pi}_{\boldsymbol{R}}^{\top}\boldsymbol{\Pi}_{\boldsymbol{R}}\big)\big(\mathbb{I}_{\mathbf{3}} - \tilde{\boldsymbol{R}}\big)\big] \ , \tag{1}$$

where $\boldsymbol{\Pi}_{\boldsymbol{R}}$ is defined as $\boldsymbol{\Pi}_{\boldsymbol{R}} := \left[\begin{smallmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{smallmatrix}\right]$. The function $\Psi(\tilde{\boldsymbol{R}})$ in (1) can be expressed in terms of the entries of $\tilde{\boldsymbol{R}}(t)$ as $\Psi(\tilde{\boldsymbol{R}}) = (1/2)(1 - \tilde{R}_{11})$, where $\tilde{R}_{11}(t)$ denotes the $(1,1)$ entry of $\tilde{\boldsymbol{R}}(t)$. Therefore, $\Psi(\tilde{\boldsymbol{R}})$ is a positive-definite function about $\tilde{R}_{11} = 1$. Note that $\tilde{R}_{11} = 1$ corresponds to the situation where the velocity vector of the UAV is aligned with the basis vector $\hat{\mathbf{b}}_{\mathbf{1D}}(t)$.

With the above notation, as shown in [31], the path-following kinematic-error dynamics between the UAV and its virtual target vehicle can be written as

$$\dot{\boldsymbol{p}}_{\boldsymbol{F}}]_F = -\ \dot{\ell}\,\hat{\mathbf{t}} \ -\ \boldsymbol{\omega}_{\boldsymbol{F/I}} \times \boldsymbol{p}_{\boldsymbol{F}} \ +\ v\,\hat{\mathbf{w}}_{\mathbf{1}} \ , \tag{2a}$$

$$\dot{\Psi}(\tilde{\boldsymbol{R}}) = \boldsymbol{e}_{\tilde{\boldsymbol{R}}} \cdot \left(\begin{bmatrix} q \\ r \end{bmatrix} - \boldsymbol{\Pi}_{\boldsymbol{R}} \tilde{\boldsymbol{R}}^{\top} \left(\boldsymbol{R}_{\boldsymbol{F}}^{\boldsymbol{D}} \{\boldsymbol{\omega}_{\boldsymbol{F/I}}\}_F + \{\boldsymbol{\omega}_{\boldsymbol{D/F}}\}_D \right) \right) \ , \tag{2b}$$

where $v(t)$ denotes the magnitude of the UAV's ground velocity vector and $\boldsymbol{e}_{\tilde{\boldsymbol{R}}}(t)$ is the attitude kinematic-error vector defined as

$$\boldsymbol{e}_{\tilde{\boldsymbol{R}}} := \tfrac{1}{2}\boldsymbol{\Pi}_{\boldsymbol{R}} \left(\left(\mathbb{I}_{\mathbf{3}} - \boldsymbol{\Pi}_{\boldsymbol{R}}^{\top}\boldsymbol{\Pi}_{\boldsymbol{R}}\right) \tilde{\boldsymbol{R}} - \tilde{\boldsymbol{R}}^{\top} \left(\mathbb{I}_{\mathbf{3}} - \boldsymbol{\Pi}_{\boldsymbol{R}}^{\top}\boldsymbol{\Pi}_{\boldsymbol{R}}\right) \right)^{\vee} \ ,$$

where $(\cdot)^{\vee} : \mathfrak{so}(3) \to \mathbb{R}^3$ denotes the *vee map* (see Appendix). In the kinematic-error model (2), $q(t)$ and $r(t)$ play the role of control inputs, while the rate of progression $\dot{\ell}(t)$ of point P along the path becomes an extra variable that can be manipulated at will. At this point, the path-following generalized error vector $\boldsymbol{x}_{\boldsymbol{pf}}(t)$ can be formally defined as

$$\boldsymbol{x}_{\boldsymbol{pf}} := \big[\, \boldsymbol{p}_{\boldsymbol{F}}^{\top} \,,\, \boldsymbol{e}_{\tilde{\boldsymbol{R}}}^{\top} \,\big]^{\top} \ .$$

Notice that, within the region where $\Psi(\tilde{\boldsymbol{R}}) < 1$, if $\boldsymbol{x}_{\boldsymbol{pf}} = 0$, then both the path-following position error and the path-following attitude error are equal to zero, that is, $\boldsymbol{p}_{\boldsymbol{F}} = \mathbf{0}$ and $\Psi(\tilde{\boldsymbol{R}}) = 0$.

Using the above formulation, and given a spatially defined feasible path $\boldsymbol{p}_{\boldsymbol{d}}(\cdot)$, the problem of path following for a single vehicle can now be defined accordingly.

Definition 1 (Path-Following Problem). *For a given UAV, design feedback control laws for pitch rate $q(t)$, yaw rate $r(t)$, and rate of progression $\dot{\ell}(t)$ of the virtual target along the path such that the path-following generalized error vector $\boldsymbol{x}_{\boldsymbol{pf}}(t)$ converges to a neighborhood of the origin with a guaranteed rate of convergence, regardless of the (feasible) temporal assignments of the mission.*

2.2 Time-Critical Coordination and Network Model

To enforce the temporal constraints of the mission, we formulate a consensus problem, in which the objective of the fleet of vehicles is to reach agreement on some distributed variables of interest. Appropriate coordination variables need thus to be defined that capture the temporal assignments of the mission.

For this purpose, let $\ell'_{d,i}(t_d)$ be the desired normalized curvilinear abscissa of the ith UAV along its path at the desired mission time t_d, which is given by

$$\ell'_{d,i}(t_d) := \tfrac{1}{\ell_{fi}} \int_0^{t_d} v_{d,i}(\tau)\, d\tau \ .$$

The trajectory-generation algorithm ensures that the desired speed profiles $v_{d,i}(\cdot)$ satisfy feasibility conditions, which implies that the following bounds hold:

$$0 < v_{\min} \leq v_{d,i}(\cdot) \leq v_{\max} \ , \qquad i = 1, \ldots, n \ , \tag{3}$$

where $v_{\min}$ and $v_{\max}$ denote, respectively, minimum and maximum operating speeds of the UAVs involved in the mission. From the definition of $\ell'_{d,i}(t_d)$ and the bounds in (3), it follows that $\ell'_{d,i}(t_d)$ is a strictly increasing continuous function of t_d mapping $[0, t_d^*]$ onto $[0, 1]$. Let $\eta_i : [0, 1] \to [0, t_d^*]$ be the inverse function of $\ell'_{d,i}(t_d)$. Clearly, $\eta_i(\cdot)$ is also a strictly increasing continuous function of its argument. Then, letting $\ell'_i(t) := \ell_i(t)/\ell_{fi}$, we introduce the time-variables

$$\xi_i(t) := \eta_i(\ell'_i(t)) \ , \qquad i = 1, \ldots, n \ .$$

Note that, for any two vehicles i and j, if $\xi_i(t) = \xi_j(t) = t'_d$ at a given time t, then $\ell'_i(t) = \ell'_{d,i}(t'_d)$ and $\ell'_j(t) = \ell'_{d,j}(t'_d)$, which implies that at time t the target vehicles corresponding to UAVs i and j have the desired relative position at the desired mission time t'_d. Moreover, if $\dot{\xi}_i(t) = 1$, then at time t the ith virtual target travels at the desired speed, $\dot{\ell}_i(t) = v_{d,i}(\xi_i(t))$. The variables $\xi_i(t)$ represent thus an appropriate measure of vehicle coordination and will be referred to as *coordination states*, while the functions $\eta_i(\cdot)$ will be called *coordination maps*.

To reach agreement on these coordination states, the UAVs need to exchange information over the supporting communications network. Next, tools and facts from *algebraic graph theory* [40] are used to model the information exchange over the network as well as the constraints imposed by the communications topology.

First, it is assumed that the ith UAV can only exchange information with a neighboring set of vehicles, denoted by $\mathcal{N}_i(t)$. It is also assumed that communications are bidirectional and, for simplicity, that information is transmitted continuously with no delays. Moreover, each vehicle is only allowed to exchange its coordination state $\xi_i(t)$ with its neighbors. Finally, we assume that the connectivity of the graph $\Gamma(t)$ that captures the underlying communications topology of the fleet at time t satisfies the persistency of excitation (PE)-like condition [41]

$$\frac{1}{n}\frac{1}{T}\int_t^{t+T} \boldsymbol{Q}\, \boldsymbol{L}(\tau)\, \boldsymbol{Q}^\top d\tau \geq \mu\, \mathbb{I}_{\boldsymbol{n-1}} \ , \qquad \text{for all } t \geq 0 \ , \tag{4}$$

where $\boldsymbol{L}(t) \in \mathbb{R}^{n \times n}$ is the Laplacian of the graph $\Gamma(t)$, and $\boldsymbol{Q}$ is an $(n-1) \times n$ matrix such that $\boldsymbol{Q}\mathbf{1}_{\boldsymbol{n}} = \mathbf{0}$ and $\boldsymbol{Q}\boldsymbol{Q}^\top = \mathbb{I}_{\boldsymbol{n-1}}$, with $\mathbf{1}_{\boldsymbol{n}}$ being the vector in $\mathbb{R}^n$

whose components are all 1. Parameters $T > 0$ and $\mu \in (0, 1]$ characterize the *quality of service* (QoS) of the communications network, which in the context of this paper represents a measure of the level of connectivity of the communications graph. Note that the PE-like condition (4) requires the communications graph $\Gamma(t)$ to be connected only in an integral sense, not pointwise in time. In fact, the graph may be disconnected during some interval of time or may even fail to be connected at all times.

Using the formulation above, one can now define the problem of time-critical cooperative path following for a fleet of n UAVs.

Definition 2 (Time-Critical Cooperative Path-Following Problem). *Given a fleet of n vehicles supported by an inter-vehicle communications network and a set of desired 3D time trajectories $\boldsymbol{p_{d,i}}(t_d)$, design feedback control laws for pitch rate $q_i(t)$, yaw rate $r_i(t)$, and speed $v_i(t)$ for all vehicles such that*

1. *for each vehicle i, $i = 1, \ldots, n$, the path-following error vector $\boldsymbol{x_{pf,i}}(t)$ converges to a neighborhood of the origin; and*
2. *for each pair of vehicles i and j, $i, j = 1, \ldots, n$, the coordination errors $(\xi_i(t) - \xi_j(t))$ and $(\dot{\xi}_i(t) - 1)$ converge to a neighborhood of the origin.*

3 3D Path Following Control law

To solve the path-following problem described in Sect. 2.2.1, we first let the rate of progression of point P along the path be governed by

$$\dot{\ell} = (v\, \hat{\mathbf{w}}_{\mathbf{1}} + k_\ell \boldsymbol{p_F}) \cdot \hat{\mathbf{t}} \ , \qquad k_\ell > 0 \ . \tag{5}$$

Then, the rate commands $q_c(t)$ and $r_c(t)$ given by

$$\begin{bmatrix} q_c \\ r_c \end{bmatrix} := \boldsymbol{\Pi_R} \tilde{\boldsymbol{R}}^\top \left(\boldsymbol{R_F^D} \{\boldsymbol{\omega_{F/I}}\}_F + \{\boldsymbol{\omega_{D/F}}\}_D \right) \ - \ 2k_{\tilde{R}} \boldsymbol{e_{\tilde{R}}} \ , \qquad k_{\tilde{R}} > 0 \ , \tag{6}$$

drive the path-following generalized error vector $\boldsymbol{x_{pf}}(t)$ to a neighborhood of zero with a guaranteed rate of convergence. More precisely, it can be shown that if the speed of the vehicle satisfies $0 < v_{\min} \leq v(t) \leq v_{\max}$, then the origin of the kinematic-error dynamics (2) with the controllers $q_c(t)$ and $r_c(t)$ in (6) is locally exponentially stable. A formal statement of this result can be found in [31], while insights into this path-following algorithm can be found in [36].

The use of the Special Orthogonal group $\mathsf{SO(3)}$ in the formulation of the attitude control problem avoids the geometric singularities and complexities that appear when dealing with local parameterizations of the vehicle's attitude. See, for example, the path-following control algorithm reported in [35].

Finally, we notice that the path-following control laws $q_c(t)$ and $r_c(t)$ represent outer-loop guidance commands to be tracked by the UAV. In this sense, the proposed solution departs from standard backstepping techniques in that the final path-following control laws can be seamlessly tailored to vehicles that are equipped with commercial autopilots.

4 Time-Critical Coordination

We now address the problem of time-critical cooperative path-following control of multiple vehicles. To this effect, the speeds of the UAVs are adjusted based on coordination information exchanged among the vehicles over the supporting communications network. The distributed coordination control law described next is intended to provide a correction to the desired speed profile $v_{d,i}(\cdot)$ obtained in the trajectory-generation step, and to generate a speed command $v_{c,i}(t)$. This speed command is then to be tracked by the ith vehicle to achieve coordination.

4.1 Distributed Coordination Law

We start by noting that the evolution of the ith coordination state is given by

$$\dot{\xi}_i = \frac{\dot{\ell}_i}{v_{d,i}(\xi_i)} \ .$$

Recalling from the solution to the path-following problem that the evolution of the ith virtual target vehicle is described by $\dot{\ell}_i = (v_i\, \hat{\mathbf{w}}_{\mathbf{1},\boldsymbol{i}} + k_\ell\, \boldsymbol{p}_{\boldsymbol{F},\boldsymbol{i}}) \cdot \hat{\mathbf{t}}_{\boldsymbol{i}}$, where for simplicity we have kept k_ℓ without indexing, the dynamics of the ith coordination state can be rewritten as

$$\dot{\xi}_i = \frac{(v_i\, \hat{\mathbf{w}}_{\mathbf{1},\boldsymbol{i}} + k_\ell\, \boldsymbol{p}_{\boldsymbol{F},\boldsymbol{i}}) \cdot \hat{\mathbf{t}}_{\boldsymbol{i}}}{v_{d,i}(\xi_i)} \ .$$

Then, to solve the time-coordination problem we use dynamic inversion and define the speed command for the ith vehicle as

$$v_{c,i} := \frac{u_{\mathrm{coord},i}\, v_{d,i}(\xi_i) - k_\ell\, \boldsymbol{p}_{\boldsymbol{F},\boldsymbol{i}} \cdot \hat{\mathbf{t}}_{\boldsymbol{i}}}{\hat{\mathbf{w}}_{\mathbf{1},\boldsymbol{i}} \cdot \hat{\mathbf{t}}_{\boldsymbol{i}}} \ , \tag{7}$$

where $u_{\mathrm{coord},i}(t)$ is a coordination control law to be defined later. With this speed command, the coordination dynamics for the ith target vehicle become

$$\dot{\xi}_i = u_{\mathrm{coord},i} + \tfrac{e_{v,i}}{v_{d,i}(\xi_i)}\, \hat{\mathbf{w}}_{\mathbf{1},\boldsymbol{i}} \cdot \hat{\mathbf{t}}_{\boldsymbol{i}} \ ,$$

where $e_{v,i}(t) := v_i(t) - v_{c,i}(t)$ denotes the speed tracking error for the ith UAV.

Recall now that each UAV is allowed to exchange its coordination state $\xi_i(t)$ only with its neighbors $\mathcal{N}_i(t)$, which are defined by the communications topology. To observe this constraint, the following distributed coordination control law is proposed:

$$u_{\mathrm{coord},1}(t) = -k_P \textstyle\sum_{j\in\mathcal{N}_1} (\xi_1(t) - \xi_j(t)) + 1 \ , \tag{8a}$$

$$u_{\mathrm{coord},i}(t) = -k_P \textstyle\sum_{j\in\mathcal{N}_i} (\xi_i(t) - \xi_j(t)) + \chi_{I,i}(t) \ , \qquad i = 2,\ldots,n \ , \tag{8b}$$

$$\dot{\chi}_{I,i}(t) = -k_I \textstyle\sum_{j\in\mathcal{N}_i} (\xi_i(t) - \xi_j(t)) \, , \qquad \chi_{I,i}(0) = 1 \ , \quad i = 2,\ldots,n \ , \tag{8c}$$

where vehicle 1 is elected as the formation leader (which can be a *virtual vehicle*), and k_P and k_I are positive coordination control gains. Note that the coordination control law has a proportional-integral structure, which provides disturbance rejection capabilities at the coordination level [42].

4.2 Overall Time-Critical Cooperative Path-Following System

Figure 2 shows the overall cooperative path-following control architecture for the ith vehicle. With the approach adopted, the control architecture exhibits a multiloop control structure in which an inner-loop controller stabilizes the vehicle dynamics, while guidance outer-loop controllers are designed to control the vehicle kinematics, providing path-following and coordination capabilities.

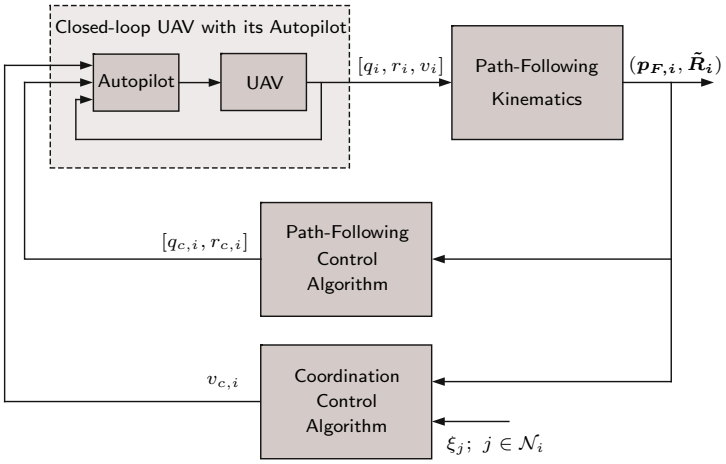


Fig. 2. Coordinated path-following closed-loop for the ith vehicle

It is proven in [31] that, if the connectivity of the communications graph verifies the PE-like condition (4) and the initial conditions are within a given domain of attraction, then there exist control gains for the path-following control law (5)-(6) and the coordination control law (7)-(8) that solve the time-critical cooperative path-following problem with guaranteed rates of exponential convergence, while ensuring at the same time that the speed of each UAV satisfies $v_{\min} \leq v_i(t) \leq v_{\max}$ for all $t \geq 0$.

Additionally, it is shown in [31] that the QoS of the network, characterized by parameters T and μ, limits the guaranteed rate of convergence of the coordination-error dynamics. The results in this paper also imply that, as the communications graph becomes connected pointwise in time, the convergence rate of the coordination-error dynamics can be set arbitrarily high by increasing the coordination control gains. This fact is consistent with results obtained in previous work; see [43, Lemma 2].

Finally, we notice that similar results have been derived for the case of a coordination control law with multiple leaders [42]; in this case, the convergence rate of the coordination dynamics depends not only on the QoS of the network, but also on the number of leaders. The work reported in [42] also analyzes the convergence properties of control law (8) when the vehicles exchange quantized information, and proves the existence of undesirable "zero-speed" attractors in the presence of coarse quantization.

5 Simulation Results

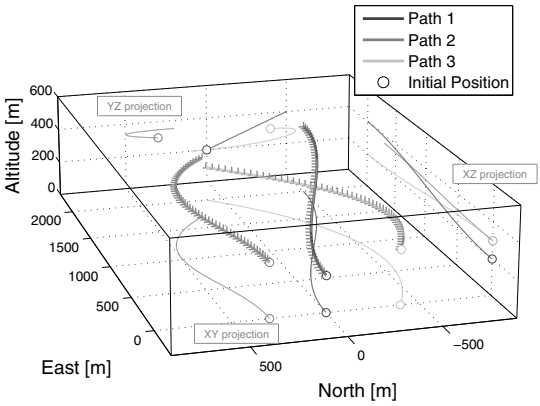
This section presents simulation results of two cooperative multi-vehicle mission scenarios that show the efficacy of the cooperative framework in this paper. In the first mission, three UAVs must execute a coordinated maneuver to arrive at predefined positions at the same time. We then consider a second mission in which three UAVs must execute sequential auto-landing while maintaining a prespecified safe-guarding separation along the glide slope. Both missions are designed to be executed by small tactical UAVs equipped with an autopilot providing angular-rate and speed tracking capabilities; see Fig. 3.

Fig. 3. SIG Rascal 110 research aircraft operated by the Naval Postgraduate School for time-critical cooperative missions. Onboard avionics include the Piccolo Plus autopilot, two PC-104 industrial embedded computers, and a wireless MANET link for air-to-air and air-to-ground communications. (See [36] for a detailed description of these avionics.)

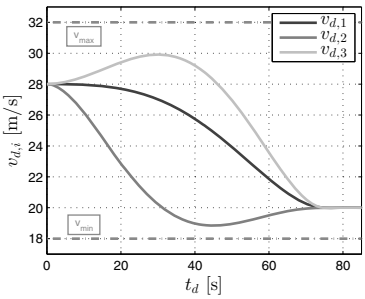
5.1 Path-Following with Simultaneous Arrival

In this mission scenario, three UAVs are tasked to converge to and follow three spatially-deconflicted paths and arrive at their final destinations at the same time. A representative example of such mission is simultaneous suppression of multiple targets located at different positions. Note that this mission imposes only *relative* temporal constraints on the arrival of the UAVs.

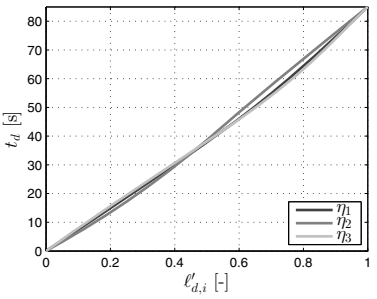
Figure 4 shows the three paths with the parallel transport frames as well as the corresponding desired speed profiles, which assume a final desired speed of 20 m/s for all UAVs. The beginning of each path is indicated in this figure with a circle. The figure also shows the coordination maps η_i relating the desired normalized curvilinear abscissa $\ell'_{d,i}$ to the desired mission time t_d. The paths have lengths $\ell_{f1} = 2,084.8$ m, $\ell_{f2} = 1,806.4$ m, and $\ell_{f3} = 2,221.0$ m, and the desired time of arrival is $t_d^* = 85.0$ s. Figure 5 presents the path separations, which show a minimum spatial clearance of 125 m, and the desired inter-vehicle separations for this particular mission.



(a) Framed 3D paths



(b) Desired speed profiles

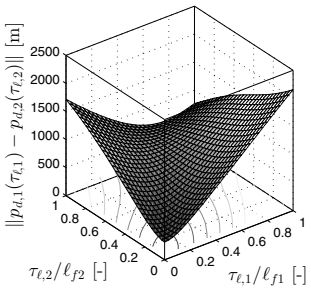


(c) Coordination maps

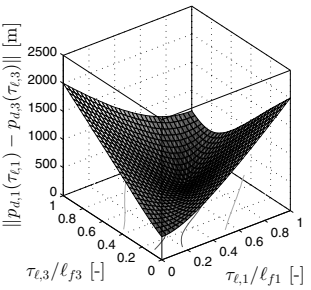
Fig. 4. Path-following with simultaneous arrival. Framed 3D spatial paths along with the corresponding desired speed profiles and coordination maps.

The cooperative motion-control algorithms described in this paper are used to solve this multi-vehicle simultaneous-arrival path-following problem. In order to achieve coordination, the UAVs rely on a supporting communications network. The information flow is assumed to be time-varying and, at any given time t, is characterized by one of the graphs in Fig. 6.

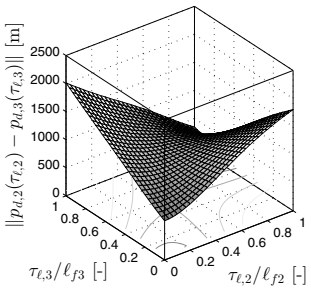
Simulation results for this particular mission are presented next. Figure 7 illustrates the evolution of the UAVs (black) as well as the virtual targets (mid gray) moving along the paths (light gray). This figure also includes the $\mathcal{W}$ frame attached to each UAV (black) as well as the $\mathcal{F}$ frame attached to the virtual targets (mid gray). The UAVs start the mission with an initial offset in both position and attitude with respect to the beginning of the framed paths. As can be seen in the figure, the path-following algorithm eliminates this initial offset and steers the UAVs along the corresponding paths, while the coordination algorithm ensures simultaneous arrival at the end of the path at $t = 84.2$ s.



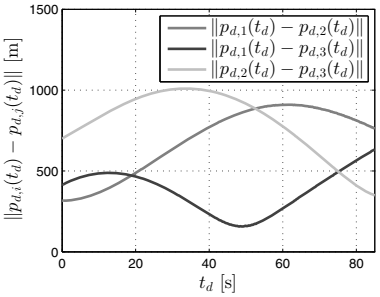
(a) Separation between paths 1 and 2



(b) Separation between paths 1 and 3



(c) Separation between paths 2 and 3



(d) Desired inter-vehicle separation

Fig. 5. Path-following with simultaneous arrival. Path separation and desired inter-vehicle separation; the three paths are spatially deconflicted with a minimum clearance of 125 m.

(a) Topology 1 (b) Topology 2 (c) Topology 3 (d) Topology 4

(e) Topology 5 (f) Topology 6 (g) Topology 7 (h) Topology 8

Fig. 6. Network topologies. At any given time t, the dynamic information flow is characterized by one of these graphs.

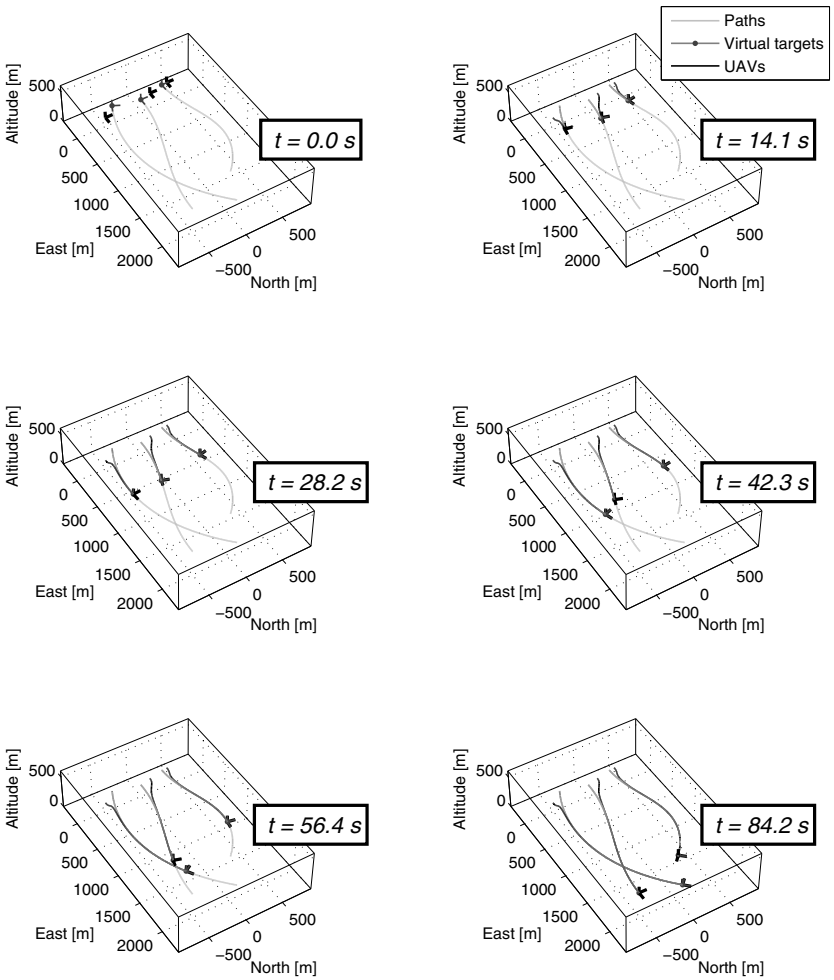
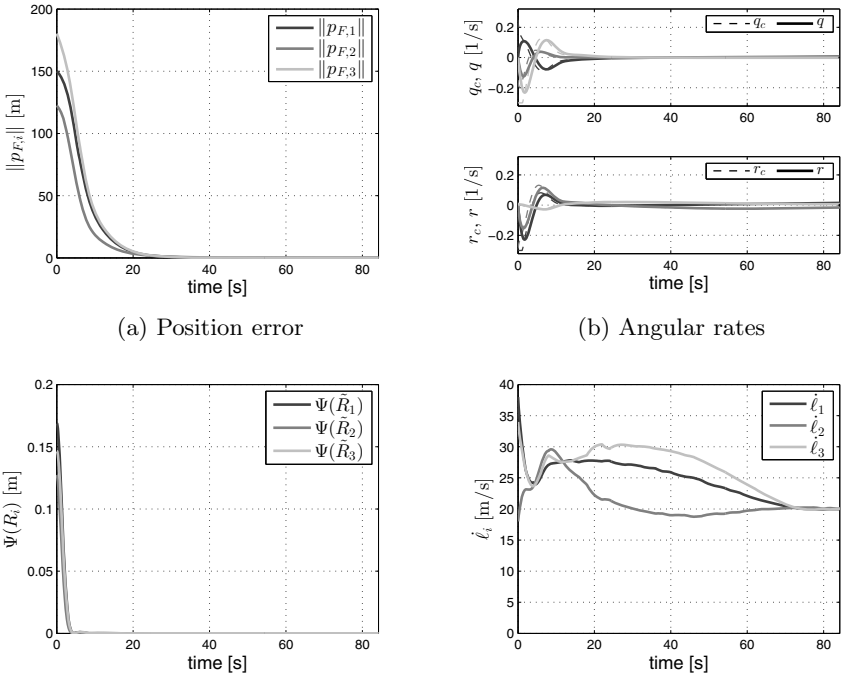


Fig. 7. Path-following with simultaneous arrival. The three UAVs achieve simultaneous arrival at their final destinations at $t = 84.2$ s.



(a) Position error

(b) Angular rates

(c) Attitude error

(d) Progression of the virtual targets

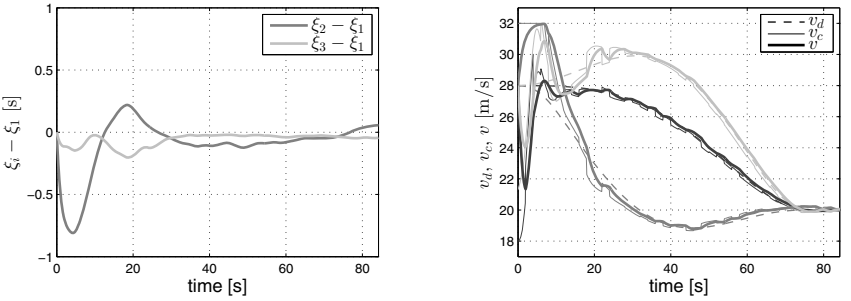
Fig. 8. Path-following with simultaneous arrival. The path-following algorithm drives the path-following position and attitude errors to a neighborhood of zero.

Details about the performance of the path-following algorithm are shown in Fig. 8; the path-following position and attitude errors, $\boldsymbol{p_{F,i}}$ and $\Psi(\tilde{\boldsymbol{R}}_{\boldsymbol{i}})$, converge to a neighborhood of zero within 30 s. The figure also presents the angular-rate commands, $q_{c,i}$ and $r_{c,i}$, as well as the rate of progression $\dot{\ell}_i$ of the virtual targets along the path.

The evolution of both the coordination errors $(\xi_i - \xi_j)$ and the rate of change of the coordination states $\dot{\xi}_i$ are illustrated in Fig. 9, along with the resulting UAV speeds and the integral states implemented on the follower vehicles. The figure shows that the coordination errors converge to a neighborhood of zero, while the rate of change of the coordination states converges to the desired rate $\dot{\xi}_{\text{ref}} = 1$. In particular, Figure 9b illustrates how the vehicles adjust their speeds (with respect to the desired speed profile) to achieve coordination. Finally, Figure 10 describes the evolution of the information flow as the mission unfolds, and presents an estimate of the QoS of the network, computed as

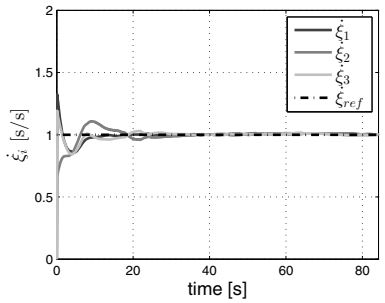
$$\hat{\mu}(t) := \lambda_{\min}\left(\frac{1}{3}\frac{1}{T}\int_{t-T}^{t} \boldsymbol{Q_3 L}(\tau)\boldsymbol{Q_3}^{\top} d\tau\right) \ , \qquad t \geq T \ , \tag{9}$$

with $T = 10$ s.

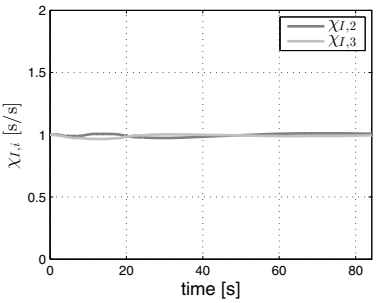


(a) Coordination errors

(b) UAV speeds

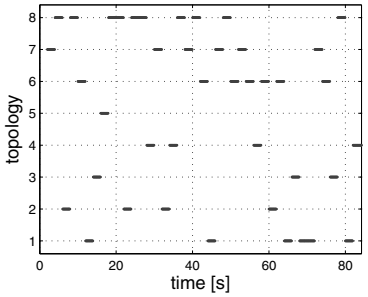


(c) Coordination-state rates

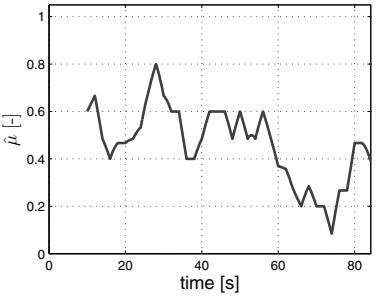


(d) Integral states

Fig. 9. Path-following with simultaneous arrival. The coordination control law ensures that the coordination errors converge to a neighborhood of zero and also that the rate of change of the coordination states evolves at about the desired rate $\dot{\xi}_{\mathrm{ref}} = 1$.



(a) Topology



(b) QoS estimate

Fig. 10. Path-following with simultaneous arrival. At a given time instant, the information flow is characterized by one of the topologies in Fig. 6. The resulting graph is only connected in an integral sense, and not pointwise in time.

5.2 Sequential Auto-Landing

Here, three UAVs must arrive at the assigned glide slope separated by prespecified safe-guarding time-intervals, and then follow the glide path at a constant approach speed while maintaining the safe-guarding separation. To this end, time-deconflicted transition trajectories are generated from prespecified initial conditions to the beginning of the glide path, satisfying the desired inter-vehicle arrival schedule and taking the UAVs to the desired approach speed. Again, this mission imposes only *relative* temporal constraints on the arrival of the UAVs.

Figure 11 shows the three transition paths with the parallel transport frames as well as the framed 3-deg glide path. The beginning of each transition path is indicated with a circle, while the beginning of the glide path is indicated with a triangle. The figure also presents the desired speed profiles for the initial transition phase that ensure a desired safe-guarding arrival separation of 30 s, trajectory deconfliction, as well as a final approach speed of 20 m/s. The transition coordination maps are shown in Fig. 11c. Finally, the figure also includes the desired speed profile for the approach along the glide slope as well as the corresponding coordination map. The transition paths have lengths $\ell_{f1} = 1,609.0$ m, $\ell_{f2} = 1,962.7$ m, and $\ell_{f3} = 2,836.7$ m, and the desired times of arrival at the glide slope are $t^*_{d1} = 65.0$ s, $t^*_{d2} = 95.0$ s, and $t^*_{d3} = 125.0$ s. Figure 12 presents the path separations, which show that the three transition paths meet at their end positions (beginning of the glide slope), whereas the desired inter-vehicle separations for this particular mission are never less than 350 m.

The cooperative motion-control algorithms described in this paper can be used to solve this sequential auto-landing problem. In this case, however, since the UAVs are required to maintain a safe-guarding separation during the approach along the glide path, the coordination states have to be redefined as the vehicles reach the glide slope. Hence, while the ith UAV is flying along its transition path, its coordination state is defined as

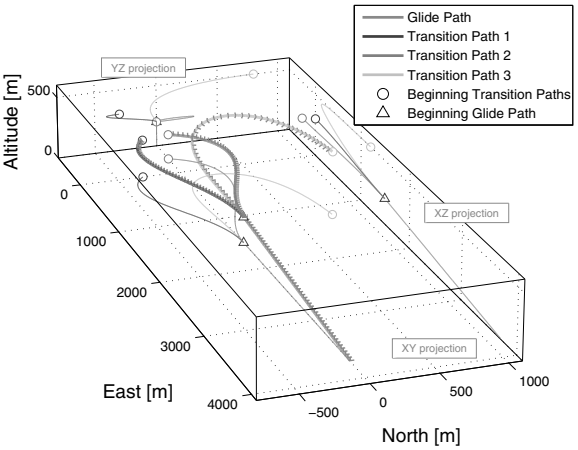
$$\xi_i(t) = \eta_i(\ell'_i(t)) \ , \qquad i = 1, 2, 3 \ ,$$

where $\ell'_i(t)$ is the normalized curvilinear abscissa of the ith virtual target along the corresponding transition path. When the UAV reaches the beginning of the glide path, then its coordination state is (re)defined as

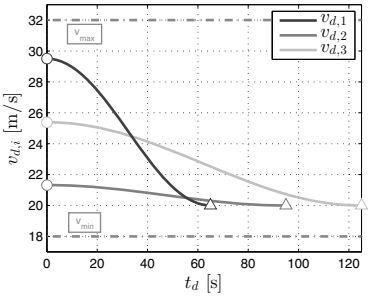
$$\xi_i(t) = \eta_{\rm gs}(\ell'_i(t)) + t^*_{di} \ , \qquad i = 1, 2, 3 \ ,$$

where $\ell'_i(t)$ is now the normalized curvilinear abscissa of the ith virtual target along the glide path, and t^*_{di} is the desired time of arrival of the ith UAV at the beginning of the glide slope. Note that, with the above definitions, the coordination states $\xi_i(t)$ are continuous, as $\eta_i(1) = t^*_{di}$ and $\eta_{\rm gs}(0) = 0$.

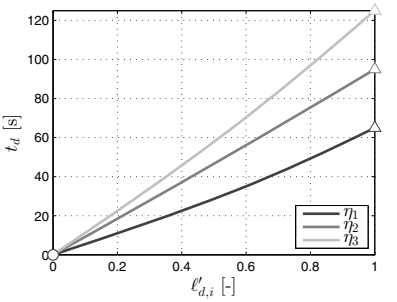
Next, we present simulation results for this mission scenario. Figure 13 illustrates the evolution of the UAVs (black) as well as the virtual targets (mid gray) moving along the paths (light gray). Similar to the previous scenario, the



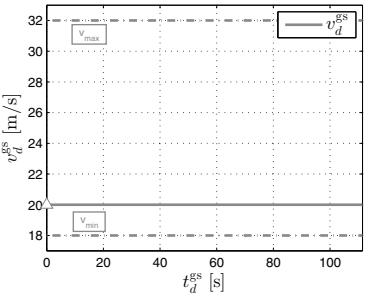
(a) Framed 3D paths



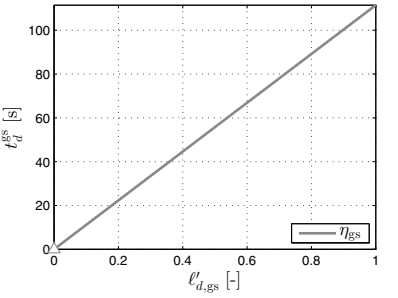
(b) Transition: Desired speed profiles



(c) Transition: Coordination maps

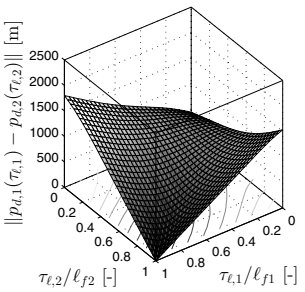


(d) Glide slope: Desired speed profile

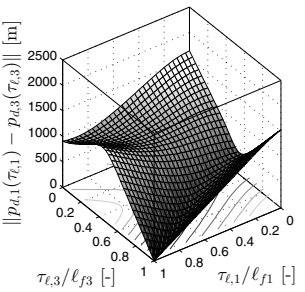


(e) Glide slope: Coordination map

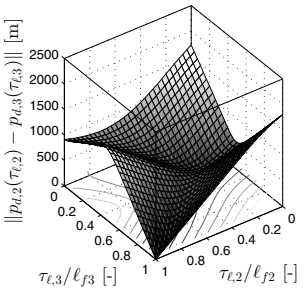
Fig. 11. Sequential auto-landing. Framed 3D spatial paths along with the corresponding desired speed profiles and coordination maps for both the transition trajectories and the glide slope.



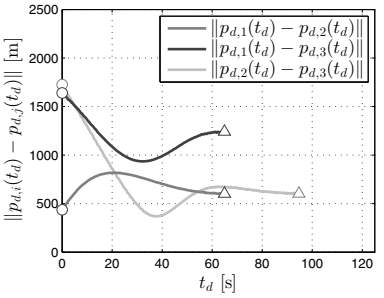
(a) Separation between paths 1 and 2



(b) Separation between paths 1 and 3



(c) Separation between paths 2 and 3



(d) Desired inter-vehicle separation

Fig. 12. Sequential auto-landing. Path separation and desired inter-vehicle separation during the transition phase; the speed profiles ensure deconfliction of the three desired trajectories with a minimum clearance of 350 m.

UAVs start the mission with an initial offset in both position and attitude with respect to the beginning of the transition paths. As can be seen in the figure, the path-following algorithm eliminates this initial offset and steers the UAVs along the corresponding transition paths, while the coordination algorithm ensures that the UAVs reach the glide slope separated by a desired time-interval. The UAVs reach the glide slope at $t = 67.0$ s, $t = 97.0$ s, and $t = 127.0$ s, meeting the desired 30 s inter-vehicle separation. After reaching the glide slope, the path-following algorithm ensures that the UAVs stay on the glide path as the coordination algorithm maintains the safe-guarding separation. The simulation is stopped when the first UAV reaches the end of the glide path.

Figure 14 shows the path-following position and attitude errors, $\boldsymbol{p_{F,i}}$ and $\Psi(\tilde{\boldsymbol{R}}_{\boldsymbol{i}})$, as well as the angular-rate commands, $q_{c,i}$ and $r_{c,i}$, and the rate of progression of the virtual targets along the path $\dot{\ell}_i$. The path-following errors converge to a neighborhood of zero within 40 s.

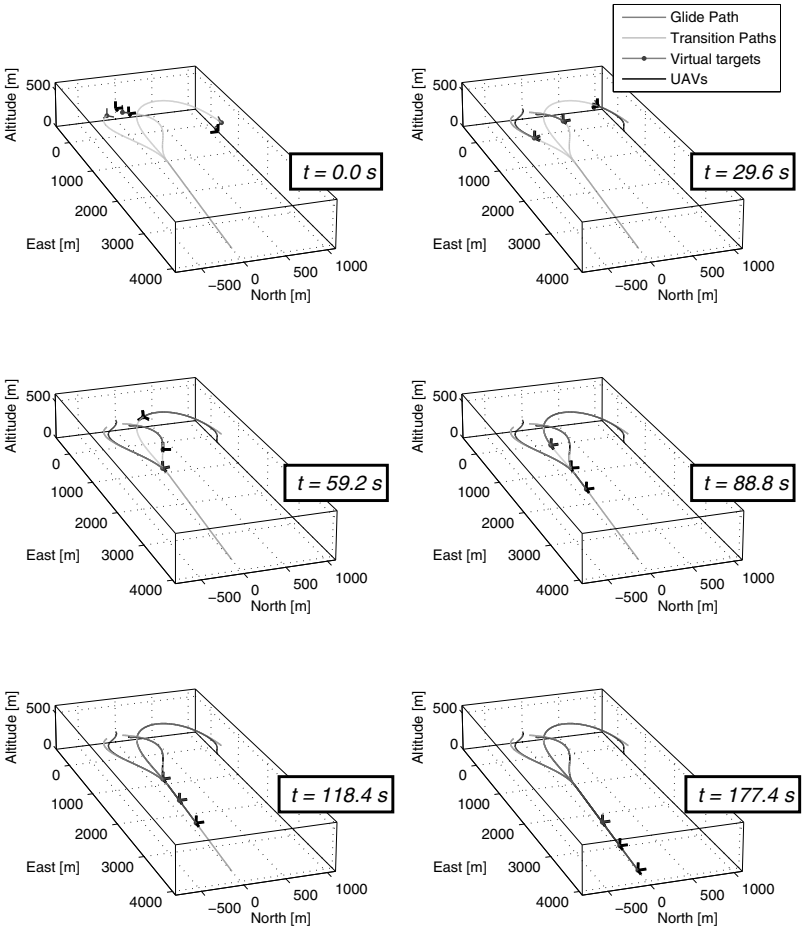
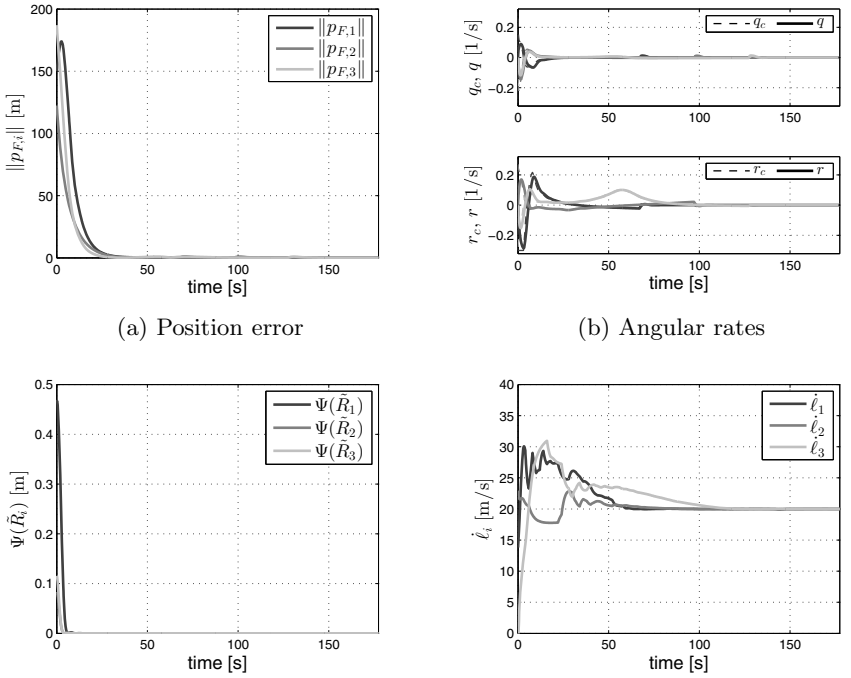


Fig. 13. Sequential auto-landing. The three UAVs arrive at the beginning of the glide path separated by approximately 30 s and maintain this safe-guarding separation as they fly along the glide slope.



(a) Position error

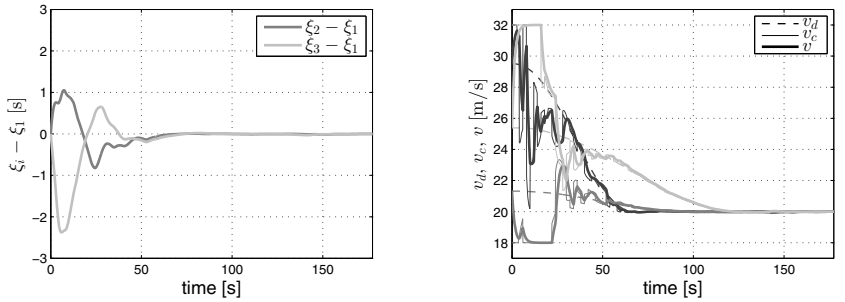
(b) Angular rates

(c) Attitude error

(d) Progression of the virtual targets

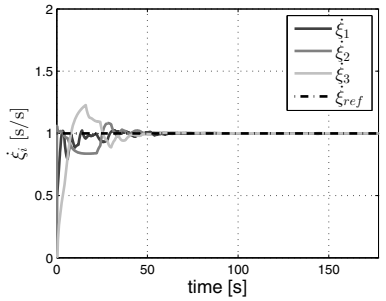
Fig. 14. Sequential auto-landing. The path-following algorithm drives the path-following position and attitude errors to a neighborhood of zero.

The coordination errors $(\xi_i - \xi_j)$ also converge to a neighborhood of zero, while the rate of change of the coordination states $\dot{\xi}_i$ converges to neighborhood of the desired rate $\dot{\xi}_{\text{ref}} = 1$; see Fig. 15. This figure also shows the UAV speeds and the integral states implemented on the follower vehicles. In particular, Figure 15b shows that, after a transient caused by the initial path-following errors as well as the speed corrections introduced by the coordination control law, the speed of each UAV converges to its desired speed and, as the vehicles enter the glide path, their speeds converge to the desired approach speed of 20 m/s. Finally, Figure 16 shows the evolution of the time-varying network topology along with an estimate of the QoS of the network, computed as in (9).

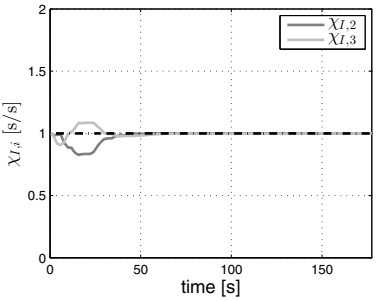


(a) Coordination errors

(b) UAV speeds

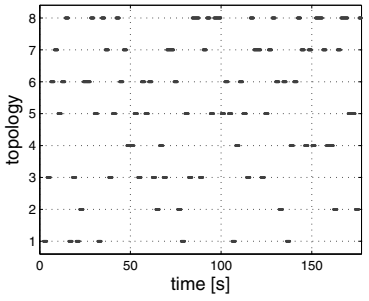


(c) Coordination-state rates

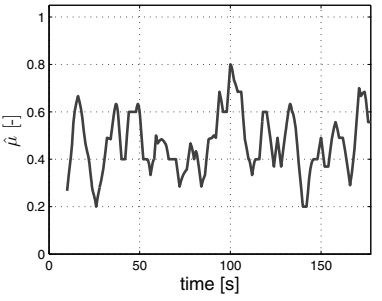


(d) Integral states

Fig. 15. Sequential auto-landing. The coordination control law ensures that the coordination errors converge to a neighborhood of zero, thus ensuring trajectory deconfliction, and also that the rate of change of the coordination states evolves at about the desired rate $\dot{\xi}_{\text{ref}} = 1$.



(a) Topology



(b) Quality of service

Fig. 16. Sequential auto-landing. At a given time instant, the information flow is characterized by one of the topologies in Fig. 6. The resulting graph is only connected in an integral sense, and not pointwise in time.

6 Conclusions

The paper presented simulation results of two multi-vehicle time-critical missions that exploit a distributed cooperative control framework proposed by the authors in [31]. The simulation study illustrated the efficacy of the algorithms developed and verified the main theoretical claims. Our current research efforts go well beyond concept; in fact, the framework described in this paper has already been tested in a cooperative road-search mission involving two small tactical UAVs equipped with commercial off-the-shelf autopilots. These preliminary flight-test results, which have been reported in [31,36], demonstrate the effectiveness of the proposed theoretical framework in a specific realistic application as well as the feasibility of the onboard implementation of the algorithms.

Acknowledgments. Research was supported in part by Office of Naval Research, Air Force Office of Scientific Research, Army Research Office, European Commission under the FP7 MORPH Project (grant agreement No. 288704), and Fundação para a Ciência e a Tecnologia under the CONAV project (PTDC/EEACRO/113820/2009) and the project PEst-OE/EEI/LA0009/2011.

Appendix: The *hat* and *vee* Maps

The *hat map* $(\cdot)^\wedge : \mathbb{R}^3 \to \mathfrak{so}(3)$ is defined as

$$(\boldsymbol{x})^\wedge = \begin{bmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{bmatrix}$$

for $\boldsymbol{x} = [x_1,\ x_2,\ x_3]^\top \in \mathbb{R}^3$. The inverse of the hat map is referred to as the *vee map* $(\cdot)^\vee : \mathfrak{so}(3) \to \mathbb{R}^3$. The reader is referred to [44] for details on these maps.

References

1. Tsitsiklis, J.N., Athans, M.: Convergence and asymptotic agreement in distributed decision problems. IEEE Transactions on Automatic Control 29(1), 42–50 (1984)
2. Sepulchre, R., Paley, D., Leonard, N.: Collective Motion and Oscillator Synchronization. In: Kumar, V., Leonard, N., Stephen Morse, A. (eds.) Cooperative Control. LNCIS, vol. 309, pp. 189–206. Springer, Heidelberg (2005)
3. Jadbabaie, A., Lin, J., Morse, A.S.: Coordination of groups of mobile autonomous agents using nearest neighbor rules. IEEE Transactions on Automatic Control 48(6), 988–1001 (2003)
4. Lin, Z., Francis, B.A., Maggiore, M.: State agreement for continuous-time coupled nonlinear systems. SIAM Journal on Control and Optimization 46(1), 288–307 (2007)
5. Egerstedt, M., Hu, X.: Formation constrained multi-agent control. IEEE Transactions on Robotics and Automation 17(6), 947–951 (2001)
6. Olfati Saber, R., Dunbar, W.B., Murray, R.M.: Cooperative control of multi-vehicle systems using cost graphs and optimization. In: American Control Conference, Denver, CO, pp. 2217–2222 (June 2003)

7. Fax, J.A., Murray, R.M.: Information flow and cooperative control of vehicle formations. IEEE Transactions on Automatic Control 49(9), 1465–1476 (2004)
8. Dunbar, W.B., Murray, R.M.: Distributed receding horizon control for multi-vehicle formation stabilization. Automatica 42(4), 549–558 (2006)
9. Ghabcheloo, R., Pascoal, A.M., Silvestre, C., Kaminer, I.: Coordinated path following control of multiple wheeled robots using linearization techniques. International Journal of Systems Science 37(6), 399–414 (2006)
10. Stevenson, D., Wheeler, M., Campbell, M.E., Whitacre, W.W., Rysdyk, R.T., Wise, R.: Cooperative tracking flight test. In: AIAA Guidance, Navigation and Control Conference, Hilton Head, SC. AIAA 2007-6756 (August 2007)
11. Keviczky, T., Borrelli, F., Fregene, K., Godbole, D., Balas, G.J.: Decentralized receding horizon control and coordination of autonomous vehicle formations. IEEE Transactions on Control System Technology 16(1), 19–33 (2008)
12. Schouwenaars, T., How, J.P., Feron, E.: Decentralized cooperative trajectory planning of multiple aircraft with hard safety guarantees. In: AIAA Guidance, Navigation and Control Conference, Providence, RI. AIAA 2004-5141 (August 2004)
13. McLain, T.W., Beard, R.W.: Coordination variables, coordination functions, and cooperative timing missions. AIAA Journal of Guidance, Control and Dynamics 28(1), 150–161 (2005)
14. Scholte, E., Campbell, M.E.: Robust nonlinear model predictive control with partial state information. IEEE Transactions on Control System Technology 16(4), 636–651 (2008)
15. Kuwata, Y., How, J.P.: Cooperative distributed robust trajectory optimization using receding horizon MILP. IEEE Transactions on Control System Technology 19(2), 423–431 (2011)
16. Fang, L., Antsaklis, P.J., Tzimas, A.: Asynchronous consensus protocols: Preliminary results, simulations and open questions. In: IEEE Conference on Decision and Control, Seville, Spain, pp. 2194–2199 (December 2005)
17. Mesbahi, M.: On state-dependent dynamic graphs and their controllability properties. IEEE Transactions on Automatic Control 50(3), 387–392 (2005)
18. Stilwell, D.J., Bishop, B.E.: Platoons of underwater vehicles. IEEE Control Systems Magazine 20(6), 45–52 (2000)
19. Stilwell, D.J., Bollt, E.M., Roberson, D.G.: Sufficient conditions for fast switching synchronization in time-varying network topologies. SIAM Journal of Applied Dynamical Systems 5(1), 140–156 (2006)
20. Cao, M., Spielman, D.A., Morse, A.S.: A lower bound on convergence of a distributed network consensus algorithm. In: IEEE Conference on Decision and Control, Seville, Spain, pp. 2356–2361 (December 2005)
21. Kim, Y., Mesbahi, M.: On maximizing the second smallest eigenvalue of state-dependent graph Laplacian. IEEE Transactions on Automatic Control 51(1), 116–120 (2006)
22. Mesbahi, M., Hadaegh, F.Y.: Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching. AIAA Journal of Guidance, Control and Dynamics 24(2), 369–377 (2001)
23. Song, Y.D., Li, Y., Liao, X.H.: Orthogonal transformation based robust adaptive close formation control of multi-UAVs. In: American Control Conference, Portland, OR, vol. 5, pp. 2983–2988 (June 2005)
24. Stipanović, D.M., İnalhan, G., Teo, R., Tomlin, C.J.: Decentralized overlapping control of a formation of unmanned aerial vehicles. Automatica 40(8), 1285–1296 (2004)

25. Skjetne, R., Moi, S., Fossen, T.I.: Nonlinear formation control of marine craft. In: IEEE Conference on Decision and Control, Las Vegas, NV, vol. 2, pp. 1699–1704 (December 2002)
26. Fossen, T.I.: Marine Control Systems: Guidance, Navigation and Control of Ships, Rigs and Underwater Vehicles. Marine Cybernetics, Norway (2002)
27. Pereira, F.L., de Sousa, J.B.: Coordinated control of networked vehicles: An autonomous underwater system. Automation and Remote Control 65(7), 1037–1045 (2004)
28. Ihle, I.A.F.: Coordinated Control of Marine Craft. PhD thesis, Norwegian University of Science and Technology, Trondheim, Norway (September 2006)
29. Ihle, I.A.F., Jouffroy, J., Fossen, T.I.: Robust formation control of marine craft using Lagrange multipliers. In: Pettersen, K.Y., Gravdahl, J.T., Nijmeijer, H. (eds.) Group Coordination and Cooperative Control. LNCIS, vol. 336, pp. 113–129. Springer, Heidelberg (2006)
30. Breivik, M., Hovstein, V.E., Fossen, T.I.: Ship formation control: A guided leader-follower approach. In: IFAC World Congress, Seoul, South Korea (July 2008)
31. Xargay, E., Kaminer, I., Pascoal, A., Hovakimyan, N., Dobrokhodov, V., Cichella, V., Aguiar, A.P., Ghabcheloo, R.: Time-critical cooperative path following of multiple unmanned aerial vehicles over time-varying networks. AIAA Journal of Guidance, Control and Dynamics 36(2), 499–516 (2013)
32. Aguiar, A.P., Pascoal, A.M.: Coordinated path-following control for nonlinear systems with logic-based communication. In: IEEE Conference on Decision and Control, New Orleans, LA, pp. 1473–1479 (December 2007)
33. Ghabcheloo, R., Kaminer, I., Aguiar, A.P., Pascoal, A.M.: A general framework for multiple vehicle time-coordinated path following control. In: American Control Conference, St. Louis, MO, pp. 3071–3076 (June 2009)
34. Ghabcheloo, R., Aguiar, A.P., Pascoal, A.M., Silvestre, C., Kaminer, I., Hespanha, J.P.: Coordinated path-following in the presence of communication losses and delays. SIAM Journal on Control and Optimization 48(1), 234–265 (2009)
35. Kaminer, I., Pascoal, A.M., Xargay, E., Hovakimyan, N., Cao, C., Dobrokhodov, V.: Path following for unmanned aerial vehicles using $\mathcal{L}_1$ adaptive augmentation of commercial autopilots. AIAA Journal of Guidance, Control and Dynamics 33(2), 550–564 (2010)
36. Xargay, E., Dobrokhodov, V., Kaminer, I., Pascoal, A.M., Hovakimyan, N., Cao, C.: Time-critical cooperative control for multiple autonomous systems. IEEE Control Systems Magazine 32(5), 49–73 (2012)
37. Choe, R., Cichella, V., Xargay, E., Hovakimyan, N., Trujillo, A.C., Kaminer, I.: A trajectory-generation framework for time-critical cooperative missions. In: AIAA Infotech@Aerospace, Boston, MA. AIAA 2013-4582 (August 2013)
38. Lapierre, L., Soetanto, D., Pascoal, A.M.: Non-singular path-following control of a unicycle in the presence of parametric modeling uncertainties. International Journal of Robust and Nonlinear Control 16(10), 485–503 (2006)
39. Bishop, R.L.: There is more than one way to frame a curve. The American Mathematical Monthly 82(3), 246–251 (1975)
40. Biggs, N.: Algebraic Graph Theory. Cambridge University Press, New York, NY (1993)
41. Arcak, M.: Passivity as a design tool for group coordination. IEEE Transactions on Automatic Control 52(8), 1380–1390 (2007)
42. Xargay, E., Choe, R., Hovakimyan, N., Kaminer, I.: Multi-leader coordination algorithm for networks with switching topology and quantized information. Automatica 50(3), 841–851 (2014)

43. Aguiar, A.P., Kaminer, I., Ghabcheloo, R., Pascoal, A.M., Xargay, E., Hovakimyan, N., Cao, C., Dobrokhodov, V.: Time-coordinated path following of multiple UAVs over time-varying networks using $\mathcal{L}_1$ adaptation. In: AIAA Guidance, Navigation and Control Conference, Honolulu, HI. AIAA 2008-7131 (August 2008)
44. Lee, T., Leok, M., McClamroch, N.H.: Control of complex maneuvers for a quadrotor UAV using geometric methods on $\mathsf{SE}(3)$. IEEE Transactions on Automatic Control (2010), available online:arXiv:1003.2005v3

Observability Analysis for the INS Error Model with GPS/Uncalibrated Magnetometer Aiding

Ronan Arraes Jardim Chagas and Jacques Waldmann

Instituto Tecnológico de Aeronáutica,
São José dos Campos, SP, 12.228-900, Brazil
ronan.jardim@gmail.com,
jacques@ita.br

Abstract. A commercial inertial navigation system (INS) yields time-diverging solutions due to errors in the inertial sensors, which can inhibit long term navigation. To circumvent this issue, a set of non-inertial sensors is used to limit these errors. The fusion between additional data and INS solution is often done by means of an extended Kalman filter using a state-error model. However, the Kalman filter estimates should be used when full observability produces small estimation uncertainty. This paper has analyzed conditions to achieve full observability using as non-inertial sensors a GPS receiver and an uncalibrated magnetometer combined with either a locally horizontal-stabilized IMU or with a strapdown IMU. The magnetometer bias was considered constant and augmented the error-state space. Observability analysis based on concepts of linear algebra provided a geometric insight on the requirements for attaining full observability when assuming piece-wise constant system dynamics. The novel analysis has been validated by covariance analysis of simulation results. Also, simulation results indicate that fusion with uncalibrated magnetometer data without proper processing gives rise to estimation divergence.

Keywords: Observability analysis, INS error model, GPS, magnetometer.

Notation and Abbreviations

$\mathbb{R}$	Set of real numbers.
DCM	Direction Cossine Matrix.
PWC	Piece-wise constant.
y	Scalar.
$\mathbf{y}$	Vector.
$\mathbf{A}$	Matrix.
$diag(\mathbf{A}\ \mathbf{B})$	Block-diagonal matrix constructed by the matrices $\mathbf{A}$ and $\mathbf{B}$.
$\mathbf{I}_n$	Identity matrix of size n.
$[\mathbf{y}]_\times \mathbf{x}$	Matrix representation of the cross product $\mathbf{y} \times \mathbf{x}$.
$\mathbf{D}_b^a$	DCM that rotates from the a coordinate frame to the b coordinate frame.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_13

$\boldsymbol{\omega}_c^{ab}$	Angular rate of the a coordinate frame with respect to the b coordinate frame represented in the c coordinate frame.
$\boldsymbol{\rho}_l$	Transport rate represented in the local horizontal frame.
$\boldsymbol{\Omega}_{e,l}$	Earth's angular rate represented in the local horizontal frame.
$\mathbf{Asp}_l$	Specific force represented in the local horizontal frame.
$\boldsymbol{\Delta}\mathbf{R}_l$	INS position error represented in the local horizontal frame.
$\boldsymbol{\Delta}\mathbf{V}_l$	INS velocity error represented in the local horizontal frame.
$\boldsymbol{\Psi}$	Misalignment from the computed coordinate frame to the platform coordinate frame.
$\boldsymbol{\nabla}$	Accelerometer bias.
$\boldsymbol{\epsilon}$	Rate-gyro drift.
$\boldsymbol{\delta}$	Magnetometer bias.
R_N	North-south radius of curvature of the Earth.
R_E	East-west radius of curvature of the Earth.
R_e	Earth radius at the latitude of the vehicle.
g_e	Gravitation at the latitude of the vehicle.
λ	Latitude of the vehicle.
h	Altitude of the vehicle.
$\boldsymbol{\omega}_l^{li}$	Angular rate of the local horizontal frame with respect to the inertial coordinate frame represented in the local horizontal frame, which is equal to $\boldsymbol{\Omega}_{e,l} + \boldsymbol{\rho}_l$.

1 Introduction

A commercial, stand-alone inertial navigation system (INS) quickly yields a diverging solutions due to errors in the low-quality inertial measurement unit (IMU) sensors [1]. In myriad applications, such errors can preclude the use of the navigation solution in the long term. Thus, a set of non-inertial sensors often aid the INS by means of a state-error model embedded in an extended Kalman filter [1]. The state-error model employs a state vector that comprises position and velocity errors, misalignment angles with respect to the locally horizontal coordinate frame, accelerometer biases, and rate-gyro drifts [1]. Consequently, observability analysis is called for to ensure that the filter estimates are accurate. Full observability yields the estimation error covariance attaining the minimum in all state-error space directions [2], and then Kalman filter estimates can be used to correct the INS errors and calibrate the inertial sensors.

Navigation becomes unfeasible when the aiding sensors are lacking with respect to limiting the misalignment error. These errors can be bounded by a magnetometer or a camera. The latter requires a pointing apparatus and calls for image processing. Hence, for low cost systems, the magnetometer is the usual choice of aiding sensor, along with a pressure altimeter, and embedded in many commercially available IMUs [3, 4].

A previous observability analysis and validation by simulation of the INS error model with GPS/magnetometer aiding in [5] have shown that useful measurements of position and misalignment errors result from comparing the calibrated

magnetometer readings to an embedded geomagnetic field model. However, incorrect magnetometer calibration impairs the estimation performance reported in [5]. The errors in the magnetometer readings should be estimated to prevent the degradation of estimation accuracy as shown later in the simulation results.

For the sake of modeling simplicity, it has been assumed that the magnetometer measurement error can be modeled as a constant bias. The magnetometer bias is appended to the state-error and the observability analysis is then carried out for an IMU mounted on a stabilized platform and for a strapdown IMU. In both cases, piece-wise constant dynamics for the INS error model have been assumed as in [6, 7]. The theoretical results have been validated by covariance analysis of a simulated aided INS with an ideal IMU. Also, to cover more realistic situations, two simulations have considered a non-ideal IMU and a time-varying magnetometer bias. In the first, the magnetometer bias has been neglected and, in the second, it has been estimated by the extended Kalman filter.

Section 2 presents the coordinate frames. The INS error model and the sensors model are presented in sections 3 and 4, respectively. The observability analysis is shown in section 5 . Simulations and results are presented in section 6. Finally, the conclusions are reported in section 7.

2 Coordinate Frames

The **true local horizontal frame** is used to represent the INS errors. In the true vehicle position, its X-axis points towards North, its Y-axis points towards East, and its Z-axis points down. This coordinate system is thereafter indicated with the l subscript.

The **computed coordinate frame** is defined as the local horizontal frame at the position computed by the INS. It is thereafter indicated with the c subscript.

The **platform coordinate frame** is defined as the local horizontal frame computed by the INS. It is thereafter indicated with the p subscript.

The **body coordinate frame** is defined as the inertial sensors coordinate frame. It is usually assumed to be aligned with the vehicle coordinate frame in strapdown IMUs or aligned with the platform coordinate frame in IMUs mounted on a stabilized platform. This coordinate frame is thereafter indicated with the b subscript.

3 INS Error Model

INS errors are increasing and unbounded, thus navigation can be seriously compromised in a long-term mission even with high-quality inertial sensors [1, 5, 8]. Non-inertial sensors provide additional information that can limit such errors. The fusion between the non-inertial sensors and the INS solution is often accomplished by an extended Kalman filter using a state-error model. The usual state vector for the INS error model is composed of position and velocity errors, misalignment from the computed coordinate frame to the platform coordinate

frame, accelerometers triad bias, and rate-gyros triad drift [1, 8]. The magnetometer bias is appended to the state-error in section 4.

For the sake of completeness, the standard state-error model for an IMU mounted on a stabilized platform and for a strapdown IMU are presented in eqs. 1 and 2, respectively:

$$\dot{\mathbf{x}} = \left[\begin{array}{c|c|c|c|c} [\boldsymbol{\rho}_l]_\times & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{g}_e & \boldsymbol{\alpha} & \boldsymbol{\Gamma} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \boldsymbol{\beta} & \mathbf{0}_{3\times3} & -\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{array}\right] \mathbf{x} \qquad (1)$$

$$\mathbf{x} = \left[\, \Delta\mathbf{R}_l^T \;\; \Delta\mathbf{V}_l^T \;\; \boldsymbol{\psi}^T \;\; \boldsymbol{\nabla}_l^T \;\; \boldsymbol{\epsilon}_l^T \,\right]^T \;,$$

$$\dot{\mathbf{x}} = \left[\begin{array}{c|c|c|c|c} [\boldsymbol{\rho}_l]_\times & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{g}_e & \boldsymbol{\alpha} & \boldsymbol{\Gamma} & \mathbf{D}_l^b & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \boldsymbol{\beta} & \mathbf{0}_{3\times3} & -\mathbf{D}_l^b \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{array}\right] \mathbf{x} \qquad (2)$$

$$\mathbf{x} = \left[\, \Delta\mathbf{R}_l^T \;\; \Delta\mathbf{V}_l^T \;\; \boldsymbol{\psi}^T \;\; \boldsymbol{\nabla}_b^T \;\; \boldsymbol{\epsilon}_b^T \,\right]^T \;,$$

where $\mathbf{g}_e = diag\left(-g_e/R_e \;\; -g_e/R_e \;\; 2g_e/R_e\right)$, $\boldsymbol{\alpha} = [\boldsymbol{\rho}_l + 2\boldsymbol{\Omega}_{e,l}]_\times$, $\boldsymbol{\beta} = [\boldsymbol{\rho}_l + \boldsymbol{\Omega}_{e,l}]_\times$, and $\boldsymbol{\Gamma} = [\mathbf{A}\mathbf{s}\mathbf{p}_l]_\times$.

4 Sensors Model

This investigation concentrates on INS error-state observability analysis when a GPS receiver and an uncalibrated magnetometer aid the INS, which extends a previous investigation [5]. A measurement model for each non-inertial sensor is described next.

The GPS solution is assumed to directly provide position and velocity errors. In practice, GPS raw data can be post-processed to yield vehicle position and velocity in the WGS84 ellipsoid coordinate frame as in a loosely-coupled implementation. Alternatively, the raw data composed of, for example, pseudoranges and Doppler shift from the receiver to each satellite are employed in a tightly-coupled implementation [8]. The GPS solution is then compared to the INS solution to produce a measurement vector of the state-error. Receiver clock errors have not been involved in this investigation.

The GPS measurement equation under the aforementioned assumption and neglecting measurement noise is

$$\mathbf{y}_{GPS} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \mathbf{x} \;. \qquad (3)$$

The magnetic pseudomeasurement is composed of the difference between the magnetometer raw data and the prediction from the geomagnetic field model.

Considering Pinson's model [10], the DCM from the body coordinate frame to the true local horizontal frame can be approximated by neglecting second order terms as

$$\mathbf{D}_b^l = \mathbf{D}_b^p \cdot \mathbf{D}_p^c \cdot \mathbf{D}_c^l \approx \mathbf{D}_b^p \cdot (\mathbf{I}_{3\times 3} - [\boldsymbol{\psi}]_\times) \cdot (\mathbf{I}_{3\times 3} - [\boldsymbol{\Delta\theta}]_\times) \approx \mathbf{D}_b^p \cdot (\mathbf{I}_{3\times 3} - [\boldsymbol{\psi}]_\times - [\boldsymbol{\Delta\theta}]_\times) \,, \tag{4}$$

where $\boldsymbol{\Delta\theta}$ is the misalignment from the true coordinate frame to the computed coordinate frame and $\boldsymbol{\psi}$ is misalignment from the computed coordinate frame to the platform coordinate frame. Thus, the calibrated magnetometer measurement neglecting noise can be approximated by [5]

$$\begin{aligned} &\mathbf{B}_{mag} \approx \mathbf{D}_b^p \cdot (\mathbf{I}_{3\times 3} - [\boldsymbol{\psi}]_\times - [\boldsymbol{\Delta\theta}]_\times) \cdot \mathbf{B}_l \\ &\mathbf{D}_p^b \mathbf{B}_{mag} - \mathbf{B}_l \approx -[\boldsymbol{\psi}]_\times \mathbf{B}_l - [\boldsymbol{\Delta\theta}]_\times \mathbf{B}_l \\ &\mathbf{y}_{mag}^{cal} = [\mathbf{B}_l]_\times \boldsymbol{\psi} + [\mathbf{B}_l]_\times \boldsymbol{\Delta\theta} \,, \end{aligned} \tag{5}$$

where $\mathbf{B}_{mag}$ is the calibrated magnetometer measurement and $\mathbf{B}_l$ is the local geomagnetic field vector represented in the true local horizontal frame, which is not accessible. Hence the use of the geomagnetic field model using the position solution computed by either the GPS or the INS. $\mathbf{D}_b^p = \mathbf{I}_3$ if the IMU and the magnetometer are mounted on a stabilized platform. Moreover, vector $\boldsymbol{\Delta\theta}$ can be related to the position error represented in the local horizontal frame as

$$\boldsymbol{\Delta\theta} = \begin{bmatrix} 0 & \dfrac{1}{R_E + h} & 0 \\ -\dfrac{1}{R_N + h} & 0 & 0 \\ 0 & -\dfrac{\tan\lambda}{R_E + h} & 0 \end{bmatrix} \cdot \boldsymbol{\Delta}\mathbf{R}_l = \mathbf{C} \cdot \boldsymbol{\Delta}\mathbf{R}_l \,. \tag{6}$$

Finally, the calibrated magnetic pseudo-measurement can be approximated by

$$\mathbf{y}_{mag}^{cal} = \begin{bmatrix} [\mathbf{B}_l]_\times \cdot \mathbf{C} & \mathbf{0}_3 & [\mathbf{B}_l]_\times & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \mathbf{x} = \mathbf{H}_{mag}^{cal} \mathbf{x} \,. \tag{7}$$

If the geomagnetic field model $\mathbf{B}_l$ is not accurate enough or the magnetometer calibration has not been executed correctly, then the magnetic pseudo-measurement $\mathbf{y}_{mag}^{cal}$ in eq. 7 will be in error. Thus, the uncalibrated magnetometer raw measurement can be approximated as follows

$$\mathbf{B}_{mag} \approx \mathbf{D}_b^p \cdot (\mathbf{I}_{3\times 3} - [\boldsymbol{\psi}]_\times - [\boldsymbol{\Delta\theta}]_\times) \cdot \mathbf{B}_l + \boldsymbol{\delta}_b \,, \tag{8}$$

where $\boldsymbol{\delta}_b$ is the magnetometer bias that comprises the uncalibrated magnetic pseudo-measurement errors.

For the sake of modeling simplicity, $\boldsymbol{\delta}_b$ has been modeled as a constant because it is expected to undergo a slow change when represented in the body coordinate frame. However, later in the simulations, this restriction is relaxed and the measurement error $\boldsymbol{\delta}_b$ is simulated using a time-varying function. Hence, the magnetometer bias dynamics neglecting modeling uncertainty can be written as

$$\dot{\boldsymbol{\delta}}_b = \mathbf{0}_{3\times 1} \,. \tag{9}$$

The magnetometer bias is appended to the state-error vector to produce the augmented INS error model presented in eqs. 10 and 11 for an IMU mounted on a stabilized platform and for a strapdown IMU, respectively:

$$\dot{\mathbf{x}}_e = \left[\begin{array}{c|c|c|c|c|c} [\boldsymbol{\rho}_l]_\times & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{g}_e & \boldsymbol{\alpha} & \boldsymbol{\Gamma} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \boldsymbol{\beta} & \mathbf{0}_{3\times3} & -\mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{array}\right] \mathbf{x}_e \tag{10}$$

$$\mathbf{x}_e = \left[\, \Delta\mathbf{R}_l^T \;\; \Delta\mathbf{V}_l^T \;\; \boldsymbol{\psi}^T \;\; \boldsymbol{\nabla}_l^T \;\; \boldsymbol{\epsilon}_l^T \;\; \boldsymbol{\delta}_l^T \,\right]^T \;,$$

$$\dot{\mathbf{x}_e} = \left[\begin{array}{c|c|c|c|c|c} [\boldsymbol{\rho}_l]_\times & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{g}_e & \boldsymbol{\alpha} & \boldsymbol{\Gamma} & \mathbf{D}_l^b & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \boldsymbol{\beta} & \mathbf{0}_{3\times3} & -\mathbf{D}_l^b & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{array}\right] \mathbf{x}_e \tag{11}$$

$$\mathbf{x}_e = \left[\, \Delta\mathbf{R}_l^T \;\; \Delta\mathbf{V}_l^T \;\; \boldsymbol{\psi}^T \;\; \boldsymbol{\nabla}_b^T \;\; \boldsymbol{\epsilon}_b^T \;\; \boldsymbol{\delta}_b^T \,\right]^T \;.$$

Thus, the uncalibrated magnetic pseudo-measurement equation can be written as follows

$$\begin{aligned} \mathbf{y}_{mag}^{uncal} &= \mathbf{H}_{mag}^{cal}\mathbf{x} + \mathbf{D}_p^b\boldsymbol{\delta}_b \\ \mathbf{y}_{mag}^{uncal} &= \left[\, \mathbf{H}_{mag}^{cal} \middle| \mathbf{D}_p^b \,\right] \mathbf{x}_e \;. \end{aligned} \tag{12}$$

If the position and velocity errors are directly measured, then, by definition, those state vector components are observable. The position error is dynamically coupled only with the velocity error, which shows that the position error dynamics fails to bring any unmeasured component into the observable subspace. Hence, for the sake of simplicity of the observability analysis, the position error component can be neglected [9].

For the purpose of observability analysis, one calls models 1 and 2 the contents of eqs. 10 and 11 without the position errors, respectively, and the measurement equations 3, and either 7 or 12. The analysis had those measurement equations properly changed to match the reduced error-state vector wherein position errors have been neglected.

5 Observability Analysis

The INS error dynamics in eqs. 10 and 11 is a time-varying, linear system. The most general way to check observability is to compute the observability Grammian [11]. However, it leads to a complicated mathematical treatment.

In the literature, observability has been verified by three main methods. The first is to analyze conditions that turn the model into a time-invariant system, e. g. the vehicle is stationary on the Earth's surface, or to find an adequate Lyapunov transformation that also leads to a time-invariant system [12]. Thus the

observability can be checked by means of rank computation of the observability matrix [11]. This approach was used in [1, 12].

The second method is applicable to piece-wise constant (PWC) systems. The observability analysis is still done by rank computation, but it turns out that it can be greatly simplified when a certain condition holds as is presented in Theorem 1 [9]. For the INS error model 1, the system can be considered PWC if the specific force is piece-wise constant [6]. For the INS error model 2, the system can be approximated by a PWC system if the specific force and attitude with respect to the local horizontal frame are piece-wise constant [7]. In this case, it is also expected that the DCM $\mathbf{D}_p^b$ will be PWC, since, during a maneuver segment, it will vary only due to numerical errors and imperfections of the inertial sensors. Thus, for sufficiently short segments, the uncalibrated magnetic pseudo-measurement matrix in eq. 12 can be approximated as a PWC matrix.

The third method has tried to investigate observability without assuming piece-wise constant dynamics [12–16]. Since the observability analysis of time-varying systems is not easily applied under general conditions, several restrictions can be considered, for example, constant specific forces and angular velocities, or a C-shaped path.

For the sake of simplicity, the second method based on assuming piece-wise constant dynamics has been preferred here. It can be shown that under this constraint, an observability analysis with linear algebra concepts provides sufficient conditions for full observability that hold for practically all situations of interest [5]. Additionally, the approach yields a geometrical insight of the kinematics involved in the observability analysis [5]. Let a vehicle move at constant altitude according to three consecutive trajectory segments: 1) towards North; 2) in a C-shaped trajectory; and 3) towards East. The composed movement leads to a time-varying INS error model. However, if segment 2 is neglected, then the above INS error models turn into PWC systems. Thus, if full observability by the end of segment 3 can be proved, then, by definition, the time-varying system composed of the three segments is also fully observable. However, if full observability cannot be claimed from the analysis of the first and the third PWC segments, then further analysis including the second segment is needed. Fortunately, the vehicle can move in such a manner that the INS error model will remain constant during certain time intervals in all the situations of interest. Thus the analysis of just the PWC segments using the aforementioned method can provide sufficient conditions for full observability [5].

5.1 Observability Analysis of Piece-Wise Constant Systems

A PWC system is defined as

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}_j\mathbf{x} + \mathbf{B}_j\mathbf{u} \\ \mathbf{y} &= \mathbf{C}_j\mathbf{x} \ , \end{aligned} \tag{13}$$

where $j \in [0, 1, 2, \cdots]$ and matrices $\mathbf{A}_j$, $\mathbf{B}_j$, and $\mathbf{C}_j$ are constants for all j. Observability can be checked by rank analysis of the total observability matrix (TOM), defined for the first $r+1$ segments as [9]

$$\bar{\mathbf{Q}}(r) = \begin{bmatrix} \bar{\mathbf{Q}}_0 \\ \bar{\mathbf{Q}}_1 \cdot e^{A_0 \Delta_0} \\ \vdots \\ \bar{\mathbf{Q}}_r \cdot e^{\mathbf{A}_{r-1}\Delta_{r-1}} \cdot \ \cdots \ \cdot e^{\mathbf{A}_1 \Delta_1} \cdot e^{\mathbf{A}_0 \Delta_0} \end{bmatrix} , \tag{14}$$

with matrix $\mathbf{Q}_i$ defined as

$$\bar{\mathbf{Q}}_i = \begin{bmatrix} \mathbf{C}_i^T & [\mathbf{C}_i\mathbf{A}_i]^T & \cdots & \left[\mathbf{C}_i\mathbf{A}_i^{n-1}\right]^T \end{bmatrix}^T , \tag{15}$$

where n is the state vector dimension.

The computation of the exponential matrices leads to tedious and complicated algebraic calculations, but it can be avoided if the following theorem holds [9].

Theorem 1. *If*

$$NULL(\mathbf{Q}_j) \in NULL(\mathbf{A}_j), \quad \forall\ j\ \in [0, 1, 2, \cdots, r] ,$$

then the following holds:

$$\begin{aligned} NULL(\bar{\mathbf{Q}}(r)) &= NULL(\bar{\mathbf{Q}}_s(r)) \\ RANK(\bar{\mathbf{Q}}(r)) &= RANK(\bar{\mathbf{Q}}_s(r)) , \end{aligned}$$

where $\bar{\mathbf{Q}}_s(r)$ is the stripped observability matrix (SOM) defined as follows

$$\bar{\mathbf{Q}}_s(r) = \begin{bmatrix} \bar{\mathbf{Q}}_0^T & \bar{\mathbf{Q}}_1^T & \cdots & \bar{\mathbf{Q}}_r^T \end{bmatrix}^T . \tag{16}$$

Thus the computation of the exponential matrices can be avoided in the observability analysis.

Proof. See [9]. □

References [7, 9] stated that the Theorem 1 holds for both models in eqs. 1 and 2 when the INS state-error model neglects the position error dynamics and terrestrial velocity measurements are available from GPS solution. On the other hand, [5] showed that this theorem only holds for these models if the specific force $\mathbf{Asp}_l$ is not aligned with the angular rate of the local horizontal frame with respect to the inertial coordinate frame $\boldsymbol{\omega}_l^{li}$ at any segment. However, to the best knowledge of the authors, the validity of Theorem 1 has not been checked for the models 1 and 2, in which the magnetometer bias is appended to the state-error vector.

Regarding model 2, the observability matrix for the j-th segment can be written, after elementary row operations, as

$$\bar{\mathbf{Q}}_j = \left[\begin{array}{ccccc} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \boldsymbol{\Gamma}_j & \mathbf{D}^b_{l,j} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \boldsymbol{\Gamma}_j\boldsymbol{\beta} & \mathbf{0}_3 & -\boldsymbol{\Gamma}_j\mathbf{D}^b_{l,j} & \mathbf{0}_3 \\ \mathbf{0}_3 & \boldsymbol{\Gamma}_j\boldsymbol{\beta}^2 & \mathbf{0}_3 & -\boldsymbol{\Gamma}_j\boldsymbol{\beta}\mathbf{D}^b_{l,j} & \mathbf{0}_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hline \mathbf{0}_3 & [\mathbf{B}_l]_\times & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{D}^b_{p,j} \\ \mathbf{0}_3 & [\mathbf{B}_l]_\times\boldsymbol{\beta} & \mathbf{0}_3 & -[\mathbf{B}_l]_\times\mathbf{D}^b_{l,j} & \mathbf{0}_3 \\ \mathbf{0}_3 & [\mathbf{B}_l]_\times\boldsymbol{\beta}^2 & \mathbf{0}_3 & -[\mathbf{B}_l]_\times\boldsymbol{\beta}\mathbf{D}^b_{l,j} & \mathbf{0}_3 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{array}\right] \begin{array}{l} \text{GPS} \\ \\ \\ \\ \\ \\ \text{Mag.} \\ \\ \\ \end{array} , \tag{17}$$

where the subscript j indicates a matrix at the j-th segment. Notice that $\boldsymbol{\beta}$, which is the cross product matrix representation of the angular rate vector $\boldsymbol{\omega}^{li}_l$, and $[\mathbf{B}_l]_\times$ are time-varying matrices. However, they can be approximated as constants for a short term analysis when the terrestrial velocity is small enough.

A vector $\mathbf{x} \in NULL(\bar{\mathbf{Q}}_j)$ must satisfy the following conditions:

$$\begin{cases} (1) & \mathbf{x}_1 = \mathbf{0}_{3\times 1} \\ (2) & \boldsymbol{\Gamma}_j\mathbf{x}_2 + \mathbf{D}^b_{l,j}\mathbf{x}_3 = \mathbf{0}_{3\times 1} \\ (3) & \boldsymbol{\Gamma}_j\boldsymbol{\beta}^n(\boldsymbol{\beta}\mathbf{x}_2 - \mathbf{D}^b_{l,j}\mathbf{x}_4) = \mathbf{0}_{3\times 1} \quad n \in [0,1,2,\cdots] \\ (4) & [\mathbf{B}_l]_\times\boldsymbol{\beta}^n(\boldsymbol{\beta}\mathbf{x}_2 - \mathbf{D}^b_{l,j}\mathbf{x}_4) = \mathbf{0}_{3\times 1} \quad n \in [0,1,2,\cdots] \\ (5) & [\mathbf{B}_l]_\times\mathbf{x}_2 + \mathbf{D}^b_{p,j}\mathbf{x}_5 = \mathbf{0}_{3\times 1} \ . \end{cases} \tag{18}$$

If $\mathbf{Asp}_{l,j}$ is not aligned with $\boldsymbol{\omega}^{li}_l$, then condition (3) and Theorem 5 (see Appendix) lead to $\boldsymbol{\beta}\mathbf{x}_2 - \mathbf{D}^b_{l,j}\mathbf{x}_4 = \mathbf{0}_{3\times 1}$.

Furthermore, the following conditions must be satisfied if the vector $\mathbf{x}$ also lies in the null space of the j-th segment dynamics matrix $\mathbf{A}_j$:

$$\begin{aligned} &\boldsymbol{\alpha}_j\mathbf{x}_1 + \boldsymbol{\Gamma}_j\mathbf{x}_2 + \mathbf{D}^b_{l,j}\mathbf{x}_3 = \mathbf{0}_{3\times 1} \\ &\boldsymbol{\beta}_j\mathbf{x}_2 - \mathbf{D}^b_{l,j}\mathbf{x}_4 = \mathbf{0}_{3\times 1} \ . \end{aligned} \tag{19}$$

It is straightforward to check that if $\mathbf{x} \in NULL(\bar{\mathbf{Q}}_j)$ and $\mathbf{Asp}_{l,j}$ is not aligned with $\boldsymbol{\omega}^{li}_l$, then $\mathbf{x}$ also lies in the $\mathbf{A}_j$ null space. Hence, if this is valid for all $j \geq 0$, then Theorem 1 holds. One should notice that this proof can be extended to model 1 by substituting $\mathbf{D}^b_l = \mathbf{I}_3$ and $\mathbf{D}^b_p = \mathbf{I}_3$.

Hereafter, the assumption that $\mathbf{Asp}_{l,j}$ is not aligned with $\boldsymbol{\omega}^{li}_l$ for all $j \geq 0$ is called assumption $\star$. Thus, if assumption $\star$ holds, the observability analysis for the models 1 and 2 with GPS solution and uncalibrated magnetic pseudo-measurements can be carried out by means of rank analysis of the SOM.

5.2 Observability Analysis for an IMU Mounted on a Stabilized Platform

The observability analysis is now presented for the model 1 when GPS and uncalibrated magnetic pseudo-measurements as in eqs. 3 and 12, respectively, are available. The SOM for the first three segments can be assembled after elementary row and column operations as follows

$$\bar{\mathbf{Q}}_s^{'}(2) = \left[\begin{array}{ccccc} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma}_0 & \mathbf{0}_{3\times3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma}_0\boldsymbol{\beta}^{n-1} & \mathbf{0}_{3\times3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -[\mathbf{B}_l]_\times & \mathbf{0}_{3\times3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -[\mathbf{B}_l]_\times\boldsymbol{\beta}^{n-1} & \mathbf{0}_{3\times3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \hline \multicolumn{5}{c}{\mathcal{A}_1} \\ \hline \multicolumn{5}{c}{\mathcal{A}_2} \end{array}\right], \tag{20}$$

where

$$\mathcal{A}_i = \begin{bmatrix} \mathbf{0}_{3\times3} & \boldsymbol{\Gamma}_i - \boldsymbol{\Gamma}_0 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma}_i & \mathbf{0}_{3\times3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma}_i\boldsymbol{\beta}^{n-1} & \mathbf{0}_{3\times3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}. \tag{21}$$

Thus, if $\mathbf{x} \in NULL(\bar{\mathbf{Q}}_s^{'}(2))$, then the following must hold:

$$\begin{array}{ll} (1) & \mathbf{x}_1 = \mathbf{0}_{3\times1} \\ (2) & \mathbf{x}_3 = \mathbf{0}_{3\times1} \\ (3) & \boldsymbol{\Gamma}_i\boldsymbol{\beta}^n\mathbf{x}_4 = \mathbf{0}_{3\times1}, \quad n \in [0,1,2,\cdots],\ i \in [0,1,2] \\ (4) & [\mathbf{B}_l]_\times\boldsymbol{\beta}^n\mathbf{x}_4 = \mathbf{0}_{3\times1}, \quad n \in [0,1,2,\cdots] \\ (5) & (\boldsymbol{\Gamma}_1 - \boldsymbol{\Gamma}_0)\mathbf{x}_2 = \mathbf{0}_{3\times1} \\ (6) & (\boldsymbol{\Gamma}_2 - \boldsymbol{\Gamma}_0)\mathbf{x}_2 = \mathbf{0}_{3\times1}\ . \end{array} \tag{22}$$

Since this analysis is only valid if the assumption $\star$ holds, thus condition (3) and Theorem 5 (see Appendix) lead to $\mathbf{x}_4 = \mathbf{0}_{3\times1}$. Additionally, condition (5) restricts $\mathbf{x}_2$ to be aligned with $\mathbf{Asp}_{l,1} - \mathbf{Asp}_{l,0}$. Likewise, condition (6) also restricts $\mathbf{x}_2$ and $\mathbf{Asp}_{l,2} - \mathbf{Asp}_{l,0}$ to be linearly dependent vectors. Thus, if $\mathbf{Asp}_{l,1} - \mathbf{Asp}_{l,0}$ is not aligned with $\mathbf{Asp}_{l,2} - \mathbf{Asp}_{l,0}$, then the conditions (5) and (6) can only be valid with $\mathbf{x}_2 = \mathbf{0}_{3\times1}$, which leads to a fully observable system.

Finally, the full observability of the piece-wise constant INS error model for an IMU mounted on a stabilized platform with GPS/uncalibrated magnetometer aiding is achieved when:

- The specific force $\mathbf{Asp}_{l,j}$ is not aligned with the angular rate of the local horizontal frame with respect to the inertial coordinate frame $\boldsymbol{\omega}_l^{li}$ at any segment j, $j \in [0, 1, 2, 3, \cdots]$;
- There are at least three segments in which the specific force difference from segment 0 to segment 1 $\mathbf{Asp}_{l,1} - \mathbf{Asp}_{l,0}$ is not aligned with the specific force difference from segment 0 to segment 2 $\mathbf{Asp}_{l,2} - \mathbf{Asp}_{l,0}$.

One should notice that these are the same conditions to achieve full observability when the GPS is the only aiding sensor [9]. If the magnetometer is calibrated and its residual bias can be neglected, then the full observability can be achieved with just two specific force segments [5]. Hence, an uncalibrated magnetometer does not help to improve observability, but the magnetometer bias can be made observable by means of specific force changes if the IMU is mounted on a stabilized platform.

5.3 Observability Analysis for a Strapdown IMU

The observability analysis is hereafter presented for the model 2 when GPS and uncalibrated magnetic pseudo-measurements as in eqs. 3 and 12, respectively, are available. If the strapdown IMU angular rate with respect to the local coordinate frame, when represented in this same frame, is zero, then the observability analysis can be carried out as in the previous section [5]. Moreover, the dynamics of the INS error model of a strapdown IMU can be also stimulated by rotational motion. Thus additional excitation signals are available to increase the dimension of the observable subspace. The analysis when the IMU undergoes PWC attitude is presented next.

The SOM for the first two segments can be assembled after elementary row and column operations as follows

$$\bar{\mathbf{Q}}_s^{'}(1) = \left[\begin{array}{cccc|c} & & \mathcal{A}_0 & & \begin{matrix}\mathbf{0}_{9\times 3}\\ \vdots\\ \mathbf{0}_{3\times 3}\\ \vdots\end{matrix} \\ \hline & & \mathcal{A}_1 & & \begin{matrix}\mathbf{0}_{9\times 3}\\ \vdots\\ \mathbf{0}_{3\times 3}\\ \vdots\end{matrix} \\ \hline \mathbf{0}_{3\times 3} & [\mathbf{B}_l]_\times & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{D}_{p,0}^b \\ \mathbf{0}_{3\times 3} & [\mathbf{B}_l]_\times & \mathbf{0}_{3\times 3} & \mathbf{0}_{3\times 3} & \mathbf{D}_{p,1}^b \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{array}\right], \tag{23}$$

where

$$\mathcal{A}_0 = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{\Gamma} & \mathbf{D}^b_{l,0} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \boldsymbol{\Gamma\beta} & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma}\mathbf{D}^b_{l,0} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{3\times3} & \boldsymbol{\Gamma\beta}^n & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma\beta}^{n-1}\mathbf{D}^b_{l,0} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix}, \tag{24}$$

$$\mathcal{A}_1 = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{D}^b_{l,1} - \mathbf{D}^b_{l,0} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma}(\mathbf{D}^b_{l,1} - \mathbf{D}^b_{l,0}) \\ \vdots & \vdots & \vdots & \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\boldsymbol{\Gamma\beta}^{n-1}(\mathbf{D}^b_{l,1} - \mathbf{D}^b_{l,0}) \\ \vdots & \vdots & \vdots & \end{bmatrix}. \tag{25}$$

Thus if $\mathbf{x} \in NULL(\bar{\mathbf{Q}}'_s(1))$, then the following must hold:

$$\begin{aligned}
&(1) \quad \mathbf{x}_1 = \mathbf{0}_{3\times1} \\
&(2) \quad \boldsymbol{\Gamma}\mathbf{x}_2 + \mathbf{D}^b_{l,0}\mathbf{x}_3 = \mathbf{0}_{3\times1} \\
&(3) \quad \boldsymbol{\Gamma\beta}^n(\boldsymbol{\beta}\mathbf{x}_2 - \mathbf{D}^b_{l,0}\mathbf{x}_4) = \mathbf{0}_{3\times1}, \quad n \in [0,1,2,\cdots] \\
&(4) \quad (\mathbf{D}^b_{l,1} - \mathbf{D}^b_{l,0})\mathbf{x}_3 = \mathbf{0}_{3\times1} \\
&(5) \quad \boldsymbol{\Gamma\beta}^n(\mathbf{D}^b_{l,1} - \mathbf{D}^b_{l,0})\mathbf{x}_4 = \mathbf{0}_{3\times1}, \quad n \in [0,1,2,\cdots] \\
&(6) \quad [\mathbf{B}_l]_\times\mathbf{x}_2 + \mathbf{D}^b_{p,0}\mathbf{x}_5 = \mathbf{0}_{3\times1} \\
&(7) \quad [\mathbf{B}_l]_\times\mathbf{x}_2 + \mathbf{D}^b_{p,1}\mathbf{x}_5 = \mathbf{0}_{3\times1}\ .
\end{aligned} \tag{26}$$

Conditions (3) and (5), assumption $\star$, and Theorem 5 (see Appendix) lead to (A) $\boldsymbol{\beta}\mathbf{x}_2 - \mathbf{D}^b_{l,0}\mathbf{x}_4 = \mathbf{0}_{3\times1}$ and (B) $(\mathbf{D}^b_{l,1} - \mathbf{D}^b_{l,0})\mathbf{x}_4 = \mathbf{0}_{3\times1}$. Condition (B) and Theorem 6 (see Appendix) claim that $\mathbf{x}_4$ must lie in the Euler axis in which a single rotation aligns the body coordinate frame in segment 0 (*b0*) with the body coordinate frame in segment 1 (*b1*) [5]. This axis is thereafter called $\mathbf{e}_{b0\rightleftarrows b1}$. Likewise, $\mathbf{x}_3$ must also lie in the same axis due to condition (4) [5]. Then condition (2) leads to $\boldsymbol{\Gamma}\mathbf{x}_2 = -\mathbf{D}^b_{l,0}\mathbf{x}_3$, which constrains $\mathbf{x}_2$ and the specific force $\mathbf{Asp}_l$ to lie in a plane perpendicular to the Euler axis $\mathbf{e}_{b0\rightleftarrows b1}$ if $\mathbf{x}_3$ is not $\mathbf{0}_{3\times1}$ [5]. In the same way, if $\mathbf{x}_4$ is not $\mathbf{0}_{3\times1}$, then condition (A) above constrains $\mathbf{x}_2$ and the angular rate of the local horizontal frame with respect to the inertial coordinate frame $\boldsymbol{\omega}^{li}_l$ to also lie in a plane perpendicular to the same Euler axis $\mathbf{e}_{b0\rightleftarrows b1}$ [5]. Hence, if either the specific force $\mathbf{Asp}_l$ **or** the angular rate of the local horizontal frame with respect to the inertial coordinate frame $\boldsymbol{\omega}^{li}_l$ is not perpendicular to the Euler axis $\mathbf{e}_{b0\rightleftarrows b1}$, then the components $\mathbf{x}_2$, $\mathbf{x}_3$, and $\mathbf{x}_4$ must all be $\mathbf{0}_{3\times1}$ [5]. If it holds, then it is straightforward to check that $\mathbf{x}_5$ must also be $\mathbf{0}_{3\times1}$, since both $\mathbf{D}^b_{p,0}$ and $\mathbf{D}^b_{p,1}$ are full rank matrices. In such case, the INS error model with GPS and uncalibrated magnetometer aiding is fully observable.

Finally, the full observability of the INS error model for a strapdown IMU with constant specific force undergoing PWC attitude with GPS/uncalibrated magnetometer aiding is achieved when:

- The specific force $\mathbf{Asp}_l$ is not aligned with the angular rate of the local horizontal frame with respect to the inertial coordinate frame $\boldsymbol{\omega}_l^{li}$;
- The specific force $\mathbf{Asp}_l$ **or** the angular rate of the local horizontal frame with respect to the inertial coordinate frame $\boldsymbol{\omega}_l^{li}$ is not perpendicular to the Euler axis in which a single rotation aligns the body coordinate frame in segment 0 ($b0$) with the body coordinate frame in segment 1 ($b1$) $\mathbf{e}_{b0 \rightleftarrows b1}$.

6 Simulation and Results

The five scenarios for validation of foregoing theoretical results are presented as follows. All numerical simulations were obtained with a simulated INS coded in Matlab.

The three first scenarios validated the INS error model observability by covariance analysis. In these cases, the accelerometers and rate-gyros have been considered ideal and thus the model noise covariance has been set to zero. The initial EKF covariance and GPS and magnetometer measurement noise covariance matrices are presented, respectively, in eqs. 27, 28, and 29 [5]:

$$\begin{aligned}\mathbf{P}_0 = diag\big(&10^{-6}\ \ 10^{-6}\ \ 10^{-6}\ \ 10^{-4}\ \ 10^{-4}\ \ 10^{-4}\ \ 10^{-6}\ \ 10^{-6}\ \ 10^{-6}\\ &10^{-10}\ \ 10^{-10}\ \ 10^{-10}\ \ 10^{-3}\ \ 10^{-3}\ \ 10^{-3}\big)\ ,\end{aligned} \tag{27}$$

$$\mathbf{R}_{GPS} = diag\left(10^{-10}\ \ 10^{-10}\ \ 10^{-10}\right)\ , \tag{28}$$

$$\mathbf{R}_{MAG} = diag\left(10^{-20}\ \ 10^{-20}\ \ 10^{-20}\right)\ , \tag{29}$$

where the units are:

- $(\mathrm{m/s})^2$ for the velocity-related components;
- $(\mathrm{rad})^2$ for the misalignment components;
- $(\mathrm{m/s^2})^2$ for the accelerometer bias components;
- $(\mathrm{rad/s})^2$ for the rate-gyro drift components;
- $(\mathrm{Gauss})^2$ for the magnetometer bias components.

For the sake of simplicity, the local geomagnetic field vector has been assumed to point towards north with 230.60 mGauss of intensity, which is the geomagnetic field intensity at the city of São José dos Campos, Brazil. Additionally, the actual magnetometer bias has been assumed constant as follows

$$\boldsymbol{\delta}_b = [10\ \ 10\ \ 10]^T\ \ \mathrm{mGauss}\ . \tag{30}$$

The first scenario simulates an IMU mounted on a locally horizontal-stabilized platform in a GPS/Magnetometer-aided INS subjected to the trajectory in Table 1. The standard deviations of the state-error estimation error are presented component-wise in fig. 1.

Table 1. IMU trajectory for scenario 01

		Specific forces		
Start (s)	**End** (s)	**N** (m/s^2)	**E** (m/s^2)	**D** (m/s^2)
0	20	0	0	-g
20	40	0	0.5	-g
40	60	0.5	0	-g+0.5
60	80	0	0.5	-g+0.5

The second and third scenarios simulate a position-stationary strapdown IMU aided by GPS and magnetometer. The IMU has been subjected to piece-wise attitude changes in which the Euler angles that rotate the local coordinate frame into alignment with the body coordinate frame (yaw, pitch, and roll rotation sequence) are described in the figs. 2a and 2b, respectively. In both cases, the angles undergo a 0.5 s of rise time to avoid the discontinuities of an instantaneous rate of change. Additionally, the standard deviations of the state-error estimation error are presented component-wise in fig. 3 for the second scenario and in fig. 4 for the third scenario.

The fourth and fifth scenarios consider a more realistic situation. The position error is appended to the EKF state-error and the various simulation parameters are presented in Table 2. UAV ground velocity and angular rate are described in the Appendix. Magnetometer bias for these both scenarios was simulated according to fig. 2c. In the fourth scenario, eq. 7 was used as the calibrated magnetic pseudo-measurement equation, thus the magnetometer data were fused neglecting the magnetometer bias. On the other hand, in the fifth scenario, the pseudo-measurement equation used magnetometer data as in eq. 12 and the magnetometer bias was estimated by the EKF. The estimation errors of the position and velocity error components for the fourth scenario are presented in fig. 5, and the estimation errors for each state-error component for the fifth scenario are presented in figs. 6 and 7.

6.1 Results Analysis

In scenario 01 (fig. 1), when an IMU mounted on a stabilized platform is aided by GPS and uncalibrated magnetometer, the INS error model achieves full observability only in the third segment ($t > 40s$).

In scenarios 02 (fig. 3) and 03 (fig. 4), in which a strapdown IMU aided by GPS and uncalibrated magnetometer was simulated, it can be seen that the full observability is achieved for scenario 02 at the third segment ($t > 40s$) and for scenario 03 at the second segment ($t > 20s$). In scenario 02, the Euler axis that rotates the body coordinate frame at the first segment to the body coordinate frame at the second segment points towards East. Since the IMU is stationary, both the specific force and the angular rate of the local coordinate frame with respect to the inertial coordinate frame lie in the XZ plane of the local coordinate

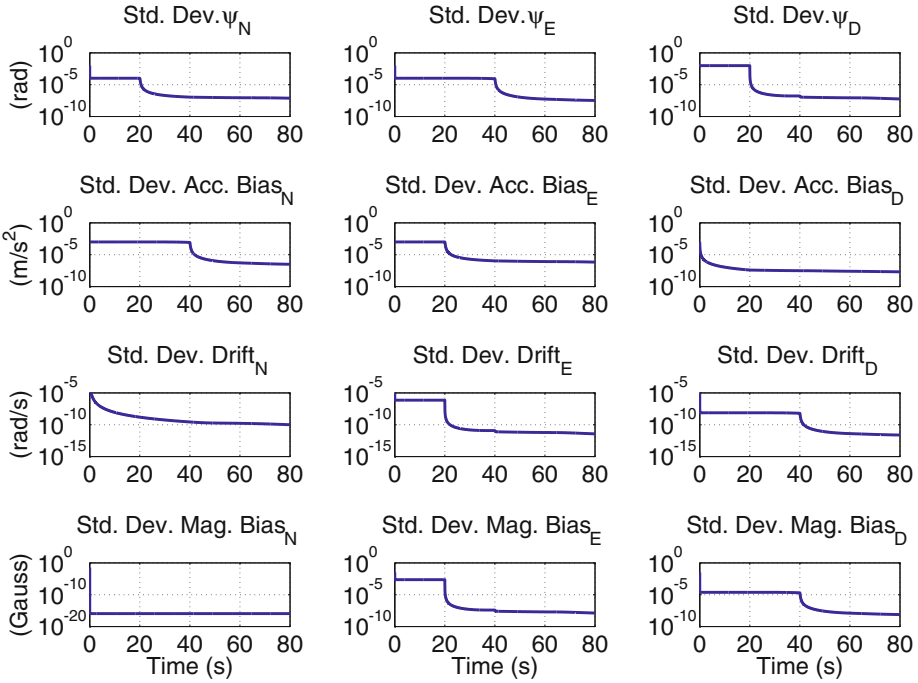
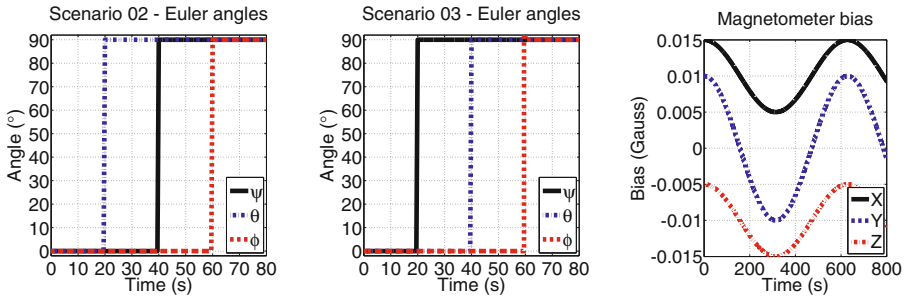


Fig. 1. Scenario 01 - Standard deviations for observability analysis with an IMU mounted on a stabilized platform

frame. Thus, these two vectors are perpendicular to the Euler axis. Hence, as the theoretical analysis predicted, this rotation does not bring all state components to the observable subspace. Thus, another rotation maneuver with a different Euler axis is required to attain full observability. On the other hand, the first rotation in scenario 03 has its Euler axis aligned with the vertical axis of the local horizontal frame. Thus, the Euler axis is not simultaneously perpendicular to the specific force and to the angular rate of the local coordinate frame with respect to the inertial coordinate frame, which brings all state components to the observable subspace after the first rotation.

The fourth scenario shows that use of magnetometer data without proper processing yields estimation divergence when the magnetometer bias is relevant. On the other hand, even assuming a constant magnetometer bias, the fifth scenario shows that the technique described here successfully estimated a slowly varying magnetometer bias and prevented the estimation divergence seen in scenario 04. Therefore, figs. 1 to 7 confirm the theoretical results of the observability analysis.



(a) Scenario 02 - Euler angles.

(b) Scenario 03 - Euler angles.

(c) Scenario 04 and 05 - Magnetometer bias.

Fig. 2. Euler angles for scenarios 02 and 03 and magnetometer bias for scenario 04

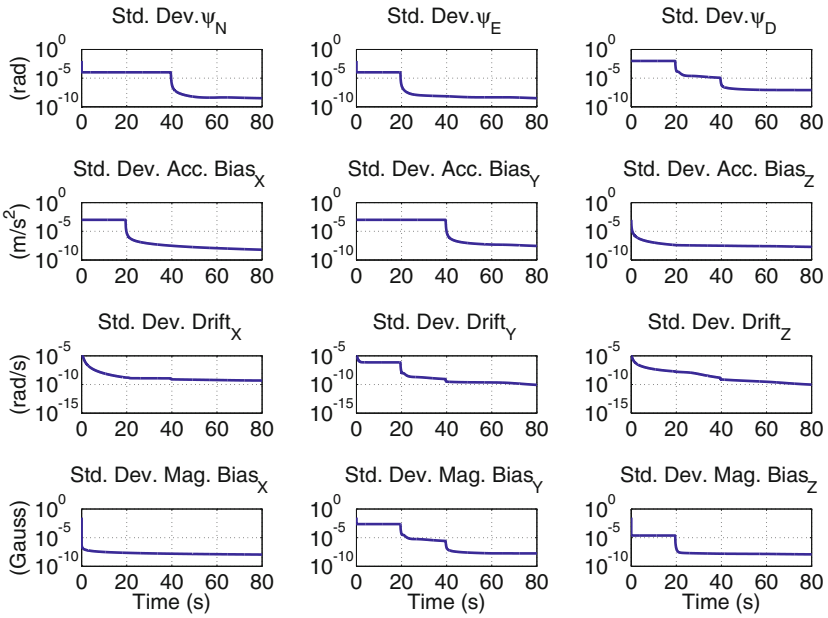


Fig. 3. Scenario 02 - Standard deviations for observability analysis with a strapdown IMU subjected to piece-wise constant rotations about Y-Z-X axes

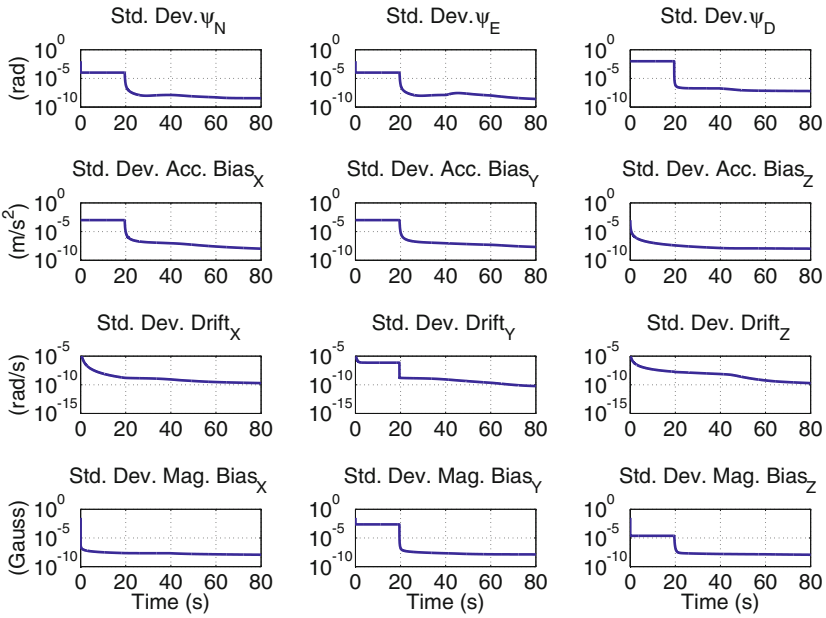


Fig. 4. Scenario 03 - Standard deviations for observability analysis with a strapdown IMU subjected to piece-wise constant rotations about Z-Y-X axes

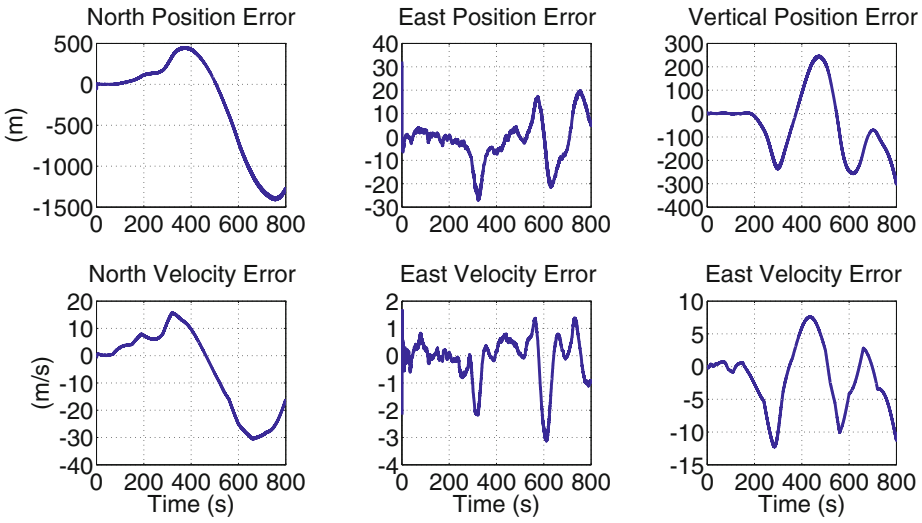


Fig. 5. Scenario 04 - Estimation errors of the position and velocity error components when the magnetometer bias is neglected

Table 2. Simulation parameters for scenarios 04 and 05

Sensors	
$\boldsymbol{\nabla}$	$\begin{bmatrix} 3 & 3 & 3 \end{bmatrix}^T$ mg
$\boldsymbol{\epsilon}$	$\begin{bmatrix} 1000 & 1000 & 1000 \end{bmatrix}^T$ $\circ$/h
Accelerometers covariance ($\mathbf{R}_{\boldsymbol{\nabla}}$)	$diag\begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$ $(mg)^2$
Rate-gyros covariance ($\mathbf{R}_{\boldsymbol{\epsilon}}$)	$diag\begin{pmatrix} 500 & 500 & 500 \end{pmatrix}$ $(\circ/\text{h})^2$
$\mathbf{R}_{GPS}$	$diag\begin{pmatrix} 81 & 81 & 81 & 0.1 & 0.1 & 0.1 \end{pmatrix}$ SI units^2
$\mathbf{R}_{magnetometer}$	$diag\begin{pmatrix} (2\cdot 10^{-5})^2 & (2\cdot 10^{-5})^2 & (2\cdot 10^{-5})^2 \end{pmatrix}$ Gauss^2
GPS and magnetometer data frequency	1 Hz
INS	
Initial position	$23^\circ 12' \ S \ \ 45^\circ 52' \ W$
Initial altitude	700 m
Initial velocity	$\begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ m/s
Initial alignment	TRIAD algorithm
INS solution sampling rate (t_{ins})	0.01 s
Kalman filter	
Feedback start	95 s
$\mathbf{Q}$, $t < 95$ s	$diag($ $\mathbf{Q}^*_{t<95\text{ s}}$ $4\cdot 10^{-10}$ $4\cdot 10^{-10}$ $4\cdot 10^{-10}$ $)$ SI Units^2
$\mathbf{Q}$, $t \geq 95$ s	$diag($ $\mathbf{Q}^*_{t>95\text{ s}}$ $4\cdot 10^{-10}$ $4\cdot 10^{-10}$ $4\cdot 10^{-10}$ $)$ SI Units^2
$\mathbf{Q}^*_{t<95\text{ s}}$	$1/50 \cdot t_{ins} \cdot \left[\begin{array}{cc} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{array}\right] \cdot \begin{bmatrix} \mathbf{R}_{\boldsymbol{\nabla}} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{R}_{\boldsymbol{\epsilon}} \end{bmatrix} \cdot \left[\begin{array}{cc} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{array}\right]^T$ SI Units^2
$\mathbf{Q}^*_{t\geq 95\text{ s}}$	$1/150 \cdot t_{ins} \cdot \left[\begin{array}{cc} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{array}\right] \cdot \begin{bmatrix} \mathbf{R}_{\boldsymbol{\nabla}} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{R}_{\boldsymbol{\epsilon}} \end{bmatrix} \cdot \left[\begin{array}{cc} \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{D}_l^b & \mathbf{0}_3 \\ \mathbf{0}_3 & -\mathbf{D}_l^b \\ \hline \multicolumn{2}{c}{\mathbf{0}_6} \end{array}\right]^T$ SI Units^2
Initial covariance	$diag(50^2 \ \ 50^2 \ \ 50^2 \ \ 2^2 \ \ 2^2 \ \ 2^2 \ \ 0.05 \ \ 0.05 \ \ 0.05 \ \ 0.09 \ \ 0.09 \ \ 0.09 \ \ 0.015 \ \ 0.015 \ \ 0.015 \ \ 1 \ \ 1 \ \ 1)$ SI Units^2
Initial estimate	$\mathbf{0}_{18\times 1}$ SI units

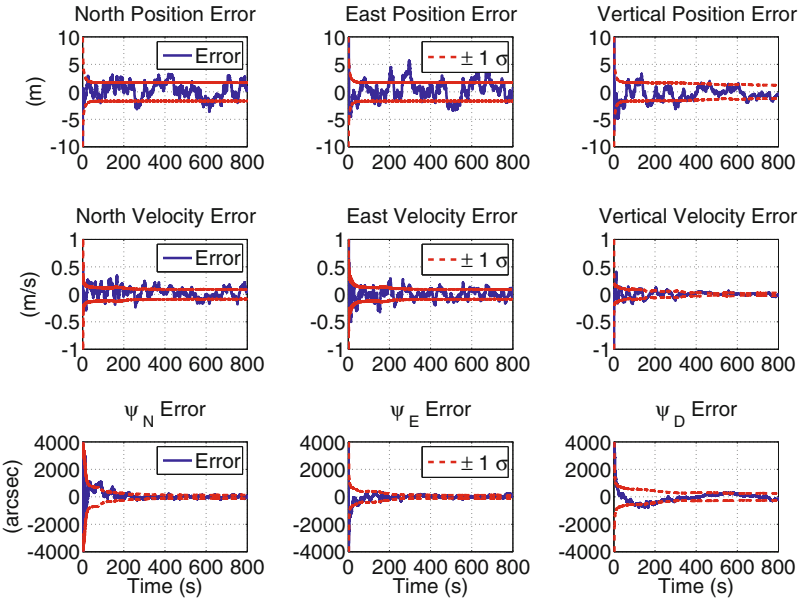


Fig. 6. Scenario 05 - Estimation errors and EKF standard deviation of the position and velocity errors and misalignment components when the magnetometer bias is considered

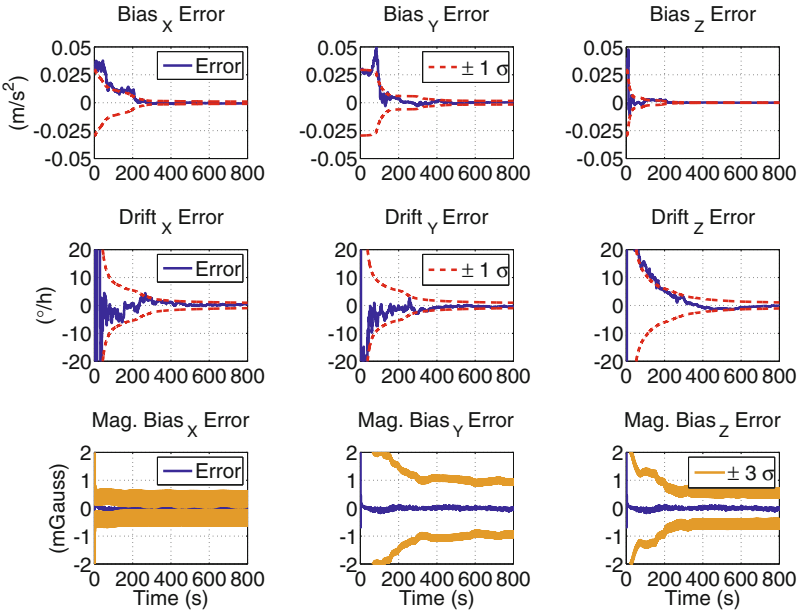


Fig. 7. Scenario 05 - Estimation errors and EKF standard deviation of the accelerometer bias, rate-gyro drift, and magnetometer bias components

7 Conclusions

The observability of a linear INS error model has been analyzed with aiding sensors that involved the use of the GPS solution and an uncalibrated magnetometer when the vehicle trajectory yielded piece-wise constant error dynamics. The analysis has dealt with both a gyro-stabilized platform undergoing piece-wise constant specific force segments and a strapdown IMU that was also subjected to piece-wise constant rotation segments.

The magnetometer errors have been modeled as a random constant bias vector appended to the INS error model. Thus, the theoretical analysis provided the sufficient conditions to achieve full observability from IMU maneuvers. All theoretical results have been validated by simulations.

The simulations revealed that the estimation accuracy was severely degraded when a relevant magnetometer bias was ignored. On the other hand, the extended Kalman filter properly estimated the state-error vector augmented with a random constant model of magnetometer bias even when the latter showed slow dynamics.

Acknowledgments. The authors acknowledge the generous support granted by project FINEP/CTA/INPE SIA (Inertial Systems for Aerospace Application, in Portuguese), embodied through the logistical support, acquisition of the computational resources needed for the research reported here, and the scholarship awarded to the first author.

Appendix: Theorems

The theorems on this appendix were already stated in [5], but they are also presented here for the sake of completeness.

Theorem 2. *Let* $\mathbf{x} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$ *and* $\mathbf{y} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$. *If* $[\mathbf{x}]_\times \mathbf{y} = \mathbf{0}_{3\times1}$, *then* $\mathbf{x}$ *and* $\mathbf{y}$ *must be aligned.*

Proof. The proof is trivial considering that $[\mathbf{x}]_\times \mathbf{y} = \mathbf{x} \times \mathbf{y}$. □

Theorem 3. *Let* $\mathbf{x} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$ *and* $\mathbf{y} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$. *Thus* $NULL([\mathbf{y}]_\times [\mathbf{x}]_\times{}^n) = NULL([\mathbf{x}]_\times), n \in [1, 2, 3, 4, \cdots]$, *iff* $\mathbf{x}$ *and* $\mathbf{y}$ *are not orthogonal.*

Proof. Left to the reader due to lack of space. □

Theorem 4. *Let* $\mathbf{x} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$ *and* $\mathbf{y} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$ *be two orthogonal vectors, then the set of vectors* $[\mathbf{x}, \mathbf{x} \times \mathbf{y}]$ *spans* $NULL([\mathbf{y}]_\times [\mathbf{x}]_\times)$.

Proof. Left to the reader due to lack of space. □

Theorem 5. *Let* $\mathbf{x} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$ *and* $\mathbf{y} \in \mathbb{R}^3 \neq \mathbf{0}_{3\times1}$ *be two non-collinear vectors. If* $[\mathbf{y}]_\times [\mathbf{x}]_\times{}^n \mathbf{v} = \mathbf{0}_{3\times1}$ *holds for all* $n \in [0, 1, 2, \cdots, L]$, $L \geq 1$, *then* $\mathbf{v} = \mathbf{0}_{3\times1}$ *is the only possible solution.*

Proof. The proof is trivial considering Theorems 2, 3, and 4. □

Theorem 6. *Let* $\mathbf{x} \in \mathbb{R}^3$ *and* $\mathbf{D}_b^a$ *and* $\mathbf{D}_c^a$ *be the DCMs from the a coordinate frame to, respectively, the b coordinate frame and c coordinate frame. If* $(\mathbf{D}_a^b - \mathbf{D}_a^c)\mathbf{x} = \boldsymbol{0}_{3\times 1}$ *holds, then* $\mathbf{x}$ *lies in the Euler axis in which a single rotation aligns the b coordinate frame to the c coordinate frame.*

Proof. The condition can be rewritten as $\mathbf{x} = \mathbf{D}_b^a \mathbf{D}_a^c \mathbf{x} = \mathbf{D}_b^c \mathbf{x}$. Thus $\mathbf{x}$ is a vector that has the same representation in the b and c coordinate frames, then $\mathbf{x}$ must lie in the Euler axis in which a single rotation rotates the b coordinate frame into alignment with the c coordinate frame. □

Appendix: IMU Trajectory and Angular Movement for Scenarios 04 and 05

The UAV motion is such that the IMU trajectory is composed of several segments with a distinct, constant specific force during each one. They are described in Table 3.

Table 3. IMU trajectory

		Specific forces		
Start (s)	**End** (s)	**N** (m/s^2)	**E** (m/s^2)	**D** (m/s^2)
0	30	0	0	-g
30	70	+3	0	-g
70	110	0	+3	-g
110	150	+3	+3	-g
150	190	0	0	-g-3
190	240	0	0	-g
240	280	-3	0	-g
280	320	0	-3	-g
320	360	0	+2	-g
360	500	0	0	-g
500	520	0	+2	-g
520	540	-2	0	-g
540	560	-2	-2	-g
560	600	0	-2	-g
600	660	0	0	-g
660	720	0	2	-g
720	800	-2	0	-g

The IMU attitude evolves in terms of the Euler angles that rotate the local coordinate frame into alignment with the body coordinate frame (yaw, pitch, and roll rotation sequence) as follows:

$$\begin{aligned}
\psi &= 0.1 \sin\left(2\pi \frac{t}{300}\right) + 0.05 \sin\left(2\pi \frac{t}{1.7}\right) \text{ rad} \\
\theta &= 0.1 \sin\left(2\pi \frac{t}{300}\right) + 0.05 \sin\left(2\pi \frac{t}{1.7}\right) \text{ rad} \\
\phi &= 0.1 \sin\left(2\pi \frac{t}{300}\right) + 0.05 \sin\left(2\pi \frac{t}{0.85}\right) \text{ rad} .
\end{aligned} \tag{31}$$

One should notice that this trajectory and angular movement are in agreement with the sufficient conditions for full observability according to the theoretical results derived in this paper.

References

1. Bar-Itzhack, I.Y., Berman, N.: Control Theoretic Approach to Inertial Navigation Systems. Journal of Guidance, Control, and Dynamics 11, 237–247 (1988)
2. Brammer, K., Siffling, G.: Kalman-Bucy Filters. Artech House Publishers, Boston (1989)
3. Xsens Technologies B.V., Pantheon 6a, P.O. Box 559, 7500 AN Enschede, The Netherlands: MTi-G User Manual and Technical Documentation (2008)
4. Analog Devices, Inc., One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.: ADIS16400/ADIS16405: Triaxial Inertial Sensor with Magnetometer (2009)
5. Chagas, R.A.J., Waldmann, J.: Geometric Inference-Based Observability Analysis Digest of INS Error Model with GPS/Magnetometer/Camera Aiding. In: 19th Saint Petersburg International Conference on Integrated Navigation Systems. CSRI Elektropribor, JSC, Saint Petersburg, Russia (2012)
6. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Observability Analysis of Piece-Wise Constant Systems - Part II: Application to Inertial Navigation In-Flight Alignment. IEEE Transactions on Aerospace and Electronic Systems 28(4), 1068–1075 (1992)
7. Lee, J., Park, C.G., Park, H.W.: Multiposition Alignment of Strapdown Inertial Navigation System. IEEE Transactions on Aerospace and Electronic Systems 29(4), 1323–1328 (1993)
8. Salychev, O.: Applied Inertial Navigation: Problems and Solutions. BMSTU Press, Moscow (2004)
9. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Observability Analysis of Piece-Wise Constant Systems - Part I: Theory. IEEE Transactions on Aerospace and Electronic Systems 28(4), 1056–1067 (1992)
10. Pinson, J.C.: Inertial Guidance for Cruise Vehicles. In: Leondes, C.T. (ed.) Guidance and Control of Aerospace Vehicles. McGraw-Hill, New York (1963)
11. Chen, C.-T.: Linear System Theory and Design. CBS College Publishing, New York (1984)
12. Chung, D., Park, C.G., Lee, J.G.: Observability Analysis of Strapdown Inertial Navigation System using Lyapunov Transformation. In: 35th IEEE Conference on Decision and Control, pp. 23–28. IEEE Press, New York (1995)

13. Rhee, I., Abdel-Hafez, M.F., Speyer, J.L.: Observability of an Integrated GPS/INS during Maneuvers. IEEE Transactions on Aerospace and Electronic Systems 40(2), 526–535 (2004)
14. Hong, S., Lee, M.H., Chun, H.-H., Kwon, S.-H., Speyer, J.L.: Observability of Errors States in GPS/INS Integration. IEEE Transactions on Vehicular Technology 54(2), 731–743 (2005)
15. Lee, M.K., Hong, S., Lee, M.H., Kwon, S., Chun, H.-H.: Observability Analysis of Alignment Erros in GPS/INS. Journal of Mechanical Science and Technology 19(6), 1253–1267 (2005)
16. Tang, Y., Wu, Y., Wu, M., Wu, W., Hu, X., Shen, L.: INS/GPS Integration: Global Observability Analysis. IEEE Transactions on Vehicular Technology 58(3), 1129–1142 (2009)

Leveling Loop Design and State Multiplicative Noise Kalman Filtering

Isaac Yaesh[1] and Adrian-Mihail Stoica[2]

[1] IMI Advanced Systems Division, Ramat Hasharon 47100, Israel
iyaesh@imi-israel.com
[2] Faculty of Aerospace Engineering, University "Politehnica" of Bucharest, Ro-011061, Romania
adrian.stoica@upb.ro

Abstract. A leveling loop for initialization of an inertial navigation system mounted on a moving platform is considered. The leveling loop is designed by exact modeling of the sensors errors as state-multiplicative noise processes. Such modeling allows application of a State Multiplicative Kalman Filter and is shown to outperform the standard Kalman filter based on ad-hoc analysis ignoring the state-multiplicative noise. The design considerations include both estimation error covariance minimization and error decay rate. Both design goals are integrated into a single design, using a trade-off parameter. A couple of numerical examples illustrate the benefits of the State Multiplicative Kalman Filter with and without the decay rate requirement. The first example which deals with a leveling loop focuses on the decay rate parameter effect, whereas the second example deals with a more standard inertial navigation and demonstrates the benefits of incorporating the state-multiplicative noise effect, rather than neglecting it.

1 Introduction

Strap Down Inertial Navigation Systems (SDINS) require initialization of position, velocity and attitude. When the platform on which the SDINS is stationary the roll and pitch of the SDINS may be measured directly from the accelerometers readings. When the platform moves and there is another board (e.g. on a marine application) accurate INS which is already aligned, one may apply transfer alignment. However, if the platform is moving and no transfer alignment is possible, one may resort to the leveling loop approach where the roll and pitch angles are estimated using the Inertial Measurement Unit (IMU) measurements aided with a speed log sensor. By applying the combination of speed and acceleration measurements, the platform motions are eliminated, and corrected accelerations are then applied to produce the roll and pitch estimates. These ideas are not new and have been pursued in [1] and [2] and the references therein. Specifically in [2] it has been argued that if the sensors errors effect is modeled as white noise, the direction cosines turn out to be modeled as a Wiener process, which diverges in time. Therefore, a finite-energy modeling of the sensors errors effect has been suggested in [2], which led to an H_∞ filtering approach

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_14

which had also the benefit of designing a filter with a prescribed bandwidth (i.e. leading to adequate time response) by using an appropriate dynamic weighting of the estimation error.

In the present paper, the design problem of leveling loops is revisited first by re modeling the sensors errors. We show that when the gyros are modeled as having white noise, the leveling loop state equations are subject to a state multiplicative noise (e.g. [3]). Under this model, a variety of exact filters can be designed for the leveling loop. One can either design a (discrete-time) state-multiplicative Kalman filter (MKF) first suggested in [4] and [5], design a State-Multiplicative Noise (see e.g. [3]) H_∞ Filter (SMH), with or without dynamic error weighting to achieve a prescribed bandwidth, or consider solutions which are 'in between' these extreme cases of H_2 and H_∞ filters (see e.g. [6]). The present paper is aimed at suggesting exact modeling of the leveling loop design problem, by explicitly taking into account the effect of the state-multiplicative noise. We note that in contrast to [3] which focuses only on the pitch leveling loop design, both pitch and roll loops are considered here. As described above, the solution of the leveling problem requires the MKF. However, since the estimation error decay rate is of practical importance, the MKF design equations are re-derived to comply with such decay rate specifications. It is worth mentioning that state-multiplicative noise appears is other navigation related estimation problems as well, as was recognized in [7] where the special filtering problem encountered in the quaternion estimation problems was solved in the presence of state-multiplicative noise in the model.

The paper is organized as follows: the leveling loop design problem is presented in Section 2 in continuous-time, and in Section 3 in discrete-time. Section 4 formulates the general estimation problem which may be used to design the leveling loop, and Section 5 presents the solution of this problem. Sections 6 and 7 include numerical examples illustrating the benefits of incorporating a decay-rate requirement in the design and of using MKF, and Section 8 summarizes and concludes the paper.

2 Leveling Loop Problem

The leveling loop utilizes as in [1] the speed log sensor output v, two of the three IMU acceleration measurements of a_x and a_y and the three gyros outputs p, q, r (namely the x, y, z gyros respectively). Taking into account the platform movement, the x and y components of total sensed acceleration in the body fixed frame of reference are given by:

$$a_x = gsin(\theta) + A_x$$

$$a_y = -gcos(\theta)sin(\phi) + A_y$$

and

$$a_z = -gcos(\theta)cos(\phi) + A_z$$

where A_x, A_y, A_z are the platform inertial accelerations along its x, y, z axes respectively and a_x, a_y, a_z are the corresponding sensed accelerations. The speed along the x axis is measured by the log speed sensor, namely

$$\dot{v} = A_x + \epsilon_x$$

where ϵ_x is the error in the log speed derivative resulting from the misalignment between the platform x axis and the speed log measurement axis. Combining these equations, and defining the direction cosines to be:

$$c_1 = -\sin(\theta)\ ,\ c_2 = \cos(\theta)\sin(\phi) \text{ and } c_3 = \cos(\theta)\cos(\phi)$$

we readily obtain by differentiating the latter equations with respect to time (see e.g. also [1]) the following state-space description of the leveling loop design problem:

$$\begin{array}{l} \frac{dc_1}{dt} = c_2 r - c_3 q \\ \frac{dc_2}{dt} = c_3 p - c_1 r \\ \frac{dc_3}{dt} = c_1 q - c_2 p \\ \frac{dv}{dt} = g c_1 + a_x + \epsilon_x \end{array}$$

We are now in the position of selecting state-vector components. If we include the body rates p, q, r in the state-vector, we need to include the corresponding differential equations to which they obey in the state equations, and to have state-equation dynamics nonlinearity stemming from the products $c_2 r$, $c_3 q$, $c_3 p$ and $c_1 r$. On the other hand, we can choose (following also [3]) to assume:

$$p = p_m + \epsilon_p$$

$$q = q_m + \epsilon_q$$

$$r = r_m + \epsilon_r$$

where p_m, q_m, r_m are the measured roll pitch and roll rates respectively, and ϵ_p, ϵ_q and ϵ_r are the corresponding measurement errors. In such a case, we summarize the resulting state equations as follows:

$$\begin{array}{l} \frac{dc_1}{dt} = c_2 r_m + \ c_2 \epsilon_r - c_3 q_m - c_3 \epsilon_q \\ \frac{dc_2}{dt} = c_3 p_m + c_3 \epsilon_p - c_1 r_m - c_1 \epsilon_r \\ \frac{dc_3}{dt} = c_1 q_m + c_1 \epsilon_q - c_2 p_m - c_2 \epsilon_p \\ \frac{dv}{dt} = g c_1 + a_{x,m} + \epsilon_x + n_x \end{array}$$

where $a_{x,m}$ and n_x are the measured x acceleration and the corresponding measurement error. We note now that the measurement errors $\epsilon_p, \epsilon_q, \epsilon_r, n_x$ may include both bias and white noise. We, however, focus on the white noise components of the sensors errors, assuming some calibration of the sensors has been performed to remove the bias errors. Obviously, any calibration cannot possibly remove the noise errors.

We define the state-vector for the estimation problem to be:

$$x = [v, c_1, c_2, c_3]^T\ . \tag{1}$$

Having formed the state vector and equations, the estimation problem definition requires also the following measurement equations:

$$\begin{aligned} y_1 &= v + v_1 \\ y_2 &= gc_2 + \epsilon_y + v_2 \\ y_3 &= gc_3 + \epsilon_z + v_3 \end{aligned}$$

where v_1 is the error in the speed log measurement and where v_2 and v_3 are the y, z acceleration measurement errors respectively. The additional error sources ϵ_y and ϵ_z express the residual unmodelled accelerations in the y, z directions. One could add additional velocity measurement in these axes to eliminate these errors (e.g. GPS receiver) but we prefer maintaining first the original setup of [1] which suits to marine applications. Another setup which utilizes such measurements will be discussed on Section 7.

We note again that due to the pre-calibration assumption, the errors $\epsilon_p, \epsilon_q, \epsilon_r$, n_x are white noise signals. So are also v_1 and v_2. However, $\epsilon_x, \epsilon_y, \epsilon_z$ are usually non white processes which may be modeled by the output of some low pass filter driven by a white noise process. Namely, $\epsilon_x, \epsilon_y, \epsilon_z$ are Markov processes. To complete the problem definition, one may e.g. assume that $\epsilon_x, \epsilon_y, \epsilon_z$ are first order processes with a given or uncertain bandwidth and prescribed covariance. Another option is to assume, in contrast to the underlying physical model, that $\epsilon_x, \epsilon_y, \epsilon_z$ are white noise processes with covariances that can serve as tuning parameters so as to achieve the required filter agility. This approach is in fact inspired by the common practice in target tracking literature [8] of modeling target maneuvers or jerks as white noise (in spite the fact they are not such) and treat their covariance matrices as tuning parameters. Since, however, such a tuning involves a try and error procedure, we will include in the filter design algorithm, filter agility (i.e. estimation error decay-rate) considerations. We next define:

$$\xi_1 = \epsilon_p, \xi_2 = \epsilon_q, \ \xi_3 = \epsilon_r$$

and notice that the filtering problem is a special case of the estimation problem defined in the next section. Before we re-state the estimation problem, we have one more decision to make. We can either choose to solve the estimation problems for c_1, c_2, c_3 in the three different 'decoupled' channels or to solve it as a fully coupled problem. In either of these cases, we define the useful inputs

$$\begin{aligned} u_1 &:= c_2 r_m - c_3 q_m \\ u_2 &:= a_{x,m} \\ u_3 &:= c_3 p_m - c_1 r_m \\ u_4 &:= c_1 q_m - c_2 p_m \, . \end{aligned}$$

For the decoupled estimation case we define the exogenous disturbance signal components

$$\begin{aligned} w_1 &:= c_2 \epsilon_r - c_3 \epsilon_q \\ w_2 &:= \epsilon_x + n_x \\ w_3 &:= c_3 \epsilon_p - c_1 \epsilon_r \\ w_4 &:= c_1 \epsilon_q - c_2 \epsilon_p \end{aligned}$$

whereas for the coupled estimation problem we just define

$$w := \epsilon_x + n_x$$

2.1 The Uncoupled Estimation

The three uncoupled estimation problems are defined by:

Estimation of c_1 [2]:

$$\begin{aligned} \frac{dc_1}{dt} &= u_1 + w_1 \\ \frac{dv}{dt} &= gc_1 + u_2 + w_2 \\ y_1 &= v + v_1 \end{aligned}$$

Estimation of c_2:

$$\begin{aligned} \frac{dc_2}{dt} &= u_3 + w_3 \\ y_2 &= gc_2 + v_2 \end{aligned}$$

Estimation of c_3:

$$\begin{aligned} \frac{dc_3}{dt} &= u_4 + w_4 \\ y_3 &= gc_3 + v_3\,. \end{aligned}$$

Each of the three estimation problems is solved as if they are decoupled from the two others. This implementation, however, ignores the fact that the overall estimation problem includes the augmented state vector x off all the three 'decoupled' problems, and, therefore, neglects the effect of the state-multiplicative noise.

2.2 Coupled Estimation

The alternative approach, explored in the present paper, treats all three problems together, and does not ignore the multiplicative noise effect. The state-equations for the latter approach are as follows :

$$\begin{aligned} \frac{dv}{dt} &= gc_1 + u_2 + w \\ \frac{dc_1}{dt} &= c_2\xi_3 - c_3\xi_2 + u_1 \\ \frac{dc_2}{dt} &= c_3\xi_1 - c_1\xi_3 + u_3 \\ \frac{dc_3}{dt} &= c_1\xi_2 - c_2\xi_1 + u_4 \end{aligned}$$

whereas the measurement equations are

$$\begin{aligned} y_1 &= v + w_1 \\ y_2 &= gc_2 + v_2 \\ y_3 &= gc_3 + v_3. \end{aligned}$$

3 The Discrete–Time Model

Consider the continuous-time model of Section 2.2. Note that it can be brought to the form of the stochastic Itô type differential equation

$$dx(t) = A_c x(t)dt + \sum_{i=1}^{N} H_{c,i} x(t) d\nu_i(t) + B_c d\omega(t)$$

where $\nu_i(t), \omega_i(t)$ and $\eta_i(t)$ are zero mean real valued Wiener processes, so that $E\{d\nu_i^2\} = dt$, $E\{d\omega_i^2\} = dt$. The discrete time counter part of this stochastic differential equation, which describes its evolution at times $kh, k = 1, 2, ...$ for small enough $0 < h << 1$ can be written defining $x(k) := x(kh)$ and using the result of Appendix A.9 in [3] as follows:

$$x(k+1) = (I + A_c h)x(k) + \sum_{i=1}^{N} H_{c,i} x(t) \xi_k \sqrt{h} + B_c w_k \sqrt{h}$$

where w_k and ξ_k are zero mean unit variance sequences. Defining $A := I + A_c h$, $H_i := H_{c,i}\sqrt{h}$, $B := B_c\sqrt{h}$ we obtain the discrete-time state equation of the next section. The continuous-time measurement equation

$$dy(t) = C_c x(t)dt + D_c d\omega(t)$$

similarly yields the discrete-time model:

$$y(k) = C_c h x(k) + D_c w_k / \sqrt{h}.$$

4 The General Estimation Problem

Consider the following system with state multiplicative noise:

$$x_{k+1} = \left(A + \sum_{i=1}^{N} H_i \xi_{i,k} \right) x_k + B w_k + \Gamma u_k \tag{2}$$

and the measurements

$$y_k = C x_k + D w_k \tag{3}$$

where $x_k \in R^n$ is the state vector, $w_k \in R^m$ is a white noise sequences of unity covariance and $\xi_k \in R^1$ is another white noise sequence independent of w_k. The matrix $\Gamma = I_4$ is the driver of the useful signal $u_k := u(kh)$ where $u := col\{u_2, u_1, u_3, u_4\}$. The matrices A, B, C, D, H are constant matrices of appropriate dimensions defined in the sequel.

In the coupled estimation problem of Section 2 B, these matrices are given by

$$A = \begin{bmatrix} 1+gh & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, B = \sigma_w \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & g & 0 \\ 0 & 0 & 0 & g \end{bmatrix},$$

$$H_1 = \sigma\sqrt{h} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix}, H_2 = \sigma\sqrt{h} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, H_3 = \sigma\sqrt{h} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

where σ is the gyros random walk value in $rad/\sqrt{sec}$. We note that for the sake of filters development, we will take just nonzero $H := H_1$ and will assume zero $H_i, i > 1$. The general result will be later deduced. We also note that

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & g & 0 \\ 0 & 0 & 0 & g \end{bmatrix} \text{ and } D = \begin{bmatrix} 0 & \sigma_v & 0 & 0 \\ 0 & 0 & \sigma_a & 0 \\ 0 & 0 & 0 & \sigma_a \end{bmatrix}$$

where σ_v is the velocity error standard deviation and σ_a is the accelerometer noise standard deviation.

We wish to estimate x_k utilizing the measurements set $Y_{k-1} = \{y_j, j \leq k-1\}$ using the following filter

$$\hat{x}_{k+1} = A\hat{x}_k + K\left(y_k - C\hat{x}_k\right)$$

where K should be designed. We also note that having multiple state-multiplicative uncertainties $H_1\xi_{1,k}+H_2\xi_{2,k} + H_3\xi_{3,k}$ as needed to solve the leveling problem of the previous section, can be made by simple adaptation of the solution given in the next section.

Our aim is to design the filter gain matrix K which will the minimize

$$P_k := E\left\{e_k e_k^T\right\}$$

where e_k is the estimation error

$$e_k = x_k - \hat{x}_k\,.$$

The following preliminary result which proof may be found in [4] will be used in the sequel regarding the mean square stability and the covariance of χ_k which satisfies:

$$\chi_{k+1} = (\alpha + \eta\xi_k)\chi_k + \beta w_k\,. \tag{4}$$

Lemma 1. *The above system is exponentially stable in mean square iff the following Lyapunov algebraic equation has a positive definite solution*

$$\alpha X \alpha^T - X + \eta X \eta^T + \beta\beta^T = 0\,. \tag{5}$$

Note that the covariance $X_k = E\{\chi_k\chi_k^T\}$ *satisfies*

$$X_{k+1} = \alpha X_k \alpha^T + \eta X_k \eta^T + \beta\beta^T\,. \tag{6}$$

Remark 1. The relation between the recursion for X_k and the equation for X was explored in [3]. It is shown there, applying monotonicity properties of the solutions of discrete-time Lyapunov equations, that solving (5) while minimizing $Tr\{X\}$ leads to to the steady-state value of X_k.

5 Estimation Problem Solution

The above estimation problem has already been solved in [4] using a completing the square argument for $D = 0$. A version of this result but in the case $D = I$ is derived in [5]. For the sake of completeness we will nevertheless provide a slightly modified of the results of [4] and [5] for the nonzero general D case.

Lemma 2. *The optimal filter gain K for which the estimation error is mean square stable and P_k is minimized, is given by:*

$$K = \left(APC^T + BD^T\right)\left(DD^T + CPC^T\right)^{-1}$$

where

$$\begin{aligned} P = APA^T + HYH^T + BB^T - \left(APC^T + BD^T\right) \\ \times\left(DD^T + CPC^T\right)^{-1}\left(CPA^T + DB^T\right) \end{aligned} \tag{7}$$

and where

$$Y = AYA^T + HYH^T + BB^T\,. \tag{8}$$

Proof. The proof is obtained following the lines of [4] and [5], by substituting

$$\alpha := \begin{bmatrix} A - KC & 0 \\ 0 & A \end{bmatrix},\ \eta := \begin{bmatrix} 0 & H \\ 0 & H \end{bmatrix},\ \beta := \begin{bmatrix} B - KD \\ B \end{bmatrix},$$

partitioning X according the dimensions of $\chi_k := col\{e_k, x_k\}$, completing the square and invoking monotonicity properties of the stabilizing solutions of the resulting Riccati equations.

Remark 2. It is easily shown that in the time varying case, the filter is to be implemented as

$$\hat{x}_{k+1} = A_k\hat{x}_k + K_k\left(y_k - C\hat{x}_k\right), \qquad \hat{x}_0 = 0$$

$$\begin{aligned} P_{k+1} = A_kP_kA_k^T + HY_kH^T + BB^T \\ -\left(A_kP_kC^T + BD^T\right)\left(DD^T + CP_kC^T\right)^{-1} \\ \times\left(CP_kA_k^T + DB^T\right) \end{aligned}$$

and

$$Y_{k+1} = A_kY_kA_k^T + HY_kH^T + BB^T$$

where $P_0 = Y_0 = E\left\{x_0x_0^T\right\}$.

We next integrate into the design, agility considerations. To do so, we reconsider the system of (4) and $X_k = E\{\chi_k \chi_k^T\}$. We note that the requirement

$$X_{k+1} < \rho^2 X_k, |\rho| < 1$$

for exponential decay of the estimation error covariance, is satisfied iff

$$\alpha X_k \alpha^T + \eta X_k \eta^T + \beta\beta^T - \rho^2 X_k < 0\,. \tag{9}$$

We obtain the following result.

Lemma 3. *The optimal filter gain K for which the estimation error is mean square stable and its covariance decays as $|\rho|^k$, is given by:*

$$K = \left(APC^T + BD^T\right)\left(DD^T + CPC^T\right)^{-1}$$

where

$$\rho^2 P = APA^T + HYH^T + BB^T$$
$$-\left(APC^T + BD^T\right)\left(DD^T + CPC^T\right)^{-1}\left(CPA^T + DB^T\right)$$

and where

$$\rho^2 Y = AYA^T + HYH^T + BB^T\,.$$

Proof. The proof is obtained following the lines of the proof of Lemma 2, by replacing (5) with (9), to get the necessary and sufficient condition for mean square stability and exponential decay as:

$$-\rho^2 P + APA^T + HYH^T + BB^T - \left(APC^T + BD^T\right)$$
$$\times\left(DD^T + CPC^T\right)^{-1}\left(CPA^T + DB^T\right) < 0$$

and

$$-\rho^2 Y + AYA^T + HYH^T + BB^T < 0$$

where the mean square stability follows from the fact that $-\rho^2 X + \alpha X \alpha^T + \eta X \eta^T + \beta\beta^T < 0$ implies, when $|\rho| < 1$ that $-X + \alpha X \alpha^T + \eta X \eta^T + \beta\beta^T < 0$.

Noting that the equations of Lemma 3 that replace the inequalities for P and Y are then obtained as sufficient conditions, using the monotonicity properties (see e.g. [1]) of the stabilizing solutions of the algebraic Riccati equations.

Remark 3. We observe that the design equations of Lemma 2 are obtained from those of Lemma 3, when $\rho = 1$. Naturally, when $|\rho| < 1$ the estimation error covariance will be no longer minimal. Therefore, $|\rho| \leq 1$ can be utilized as a design parameter to trade off between the decay rate and the error covariance.

Remark 4. We note that the Lyapunov equation for Y is very conservative when the system is marginally stable in the mean square. Note also that when $\rho = 1$, Y is just $E\{x_k x_k^T\}$. To allow filter design also in mean square marginally stable cases with $\rho < 1$ we modify (9) to be

$$\alpha X_k \alpha^T + \eta X_k \eta^T + \beta\beta^T - \Gamma X_k \Gamma^T < 0. \tag{10}$$

We then re-derive the result of Lemma 3 by choosing $\Gamma = diag\{\sqrt{\rho}I, I\}$ and obtain the following design equations

$$K = \left(APC^T + BD^T\right)\left(DD^T + CPC^T\right)^{-1}$$

where

$$\rho^2 P = APA^T + HYH^T + BB^T$$
$$-\left(APC^T + BD^T\right)\left(DD^T + CPC^T\right)^{-1}\left(CPA^T + DB^T\right)$$

and where

$$Y = AYA^T + HYH^T + BB^T\,.$$

6 Example–Coupled Leveling

We next consider the problem of coupled leveling which was described in the previous sections. The state equations are given by

$$x_{k+1} = \left(A + \sum_{i=1}^{N} H_i \xi_{i,k}\right) x_k + Bw_k + Bu_k$$

and the measurements are given by

$$y_k = Cx_k + Dw_k$$

where the matrices A, B, C, D, H are given in Section 4 above,

$$u_k = col\{u_1(k), u_2(k), u_3(k), u_4(k)\}$$

and where

$$\begin{aligned} u_1(k) &:= c_2(k)r_m(k) - c_3(k)q_m(k) \\ u_2(k) &:= a_{x,m}(k) \\ u_3(k) &:= c_3(k)p_m(k) - c_1(k)r_m(k) \\ u_4(k) &:= c_1(k)q_m(k) - c_2(k)p_m(k)\,. \end{aligned}$$

The estimator is described by

$$\hat{x}_{k+1} = A\hat{x}_k + K(y_k - C\hat{x}_k) + B\hat{u}_k$$

where $\hat{u}_k = col\{\hat{u}_1(k), \hat{u}_2(k), \hat{u}_3(k), \hat{u}_4(k)\}$ and where

$$\begin{aligned} \hat{u}_1(k) &:= \hat{c}_2(k)r_m(k) - \hat{c}_3(k)q_m(k) \\ \hat{u}_2(k) &:= a_{x,m}(k) \\ \hat{u}_3(k) &:= \hat{c}_3(k)p_m(k) - \hat{c}_1(k)r_m(k) \\ \hat{u}_4(k) &:= \hat{c}_1(k)q_m(k) - \hat{c}_2(k)p_m(k)\,. \end{aligned}$$

Remark 5. We are obliged to replace u_k by $\hat{u}_k$ in the estimator implementation, since c_i are not available. Only their estimates $\hat{c}_i$ are available. Therefore, unfortunately, the estimation problem we have in this coupled problem can not be purely modelled by the simple time invariant state-multiplicative noise model, unless p_m, q_m and r_m are near zero. This is however, nearly the case when the platform is in steady state (i.e. the angles θ and ϕ remain constant). To obtain true description of the coupled estimation problem, a time varying model (but still with multiplicative noise) should be invoked, where the A matrix depends on p_m, q_m and r_m. In such a case, one can use the time-varying version of Lemma 2 above, where the algebraic equations for P and Y are replaced by recursive equations where in the LHS P is replaced by $P(k+1)$ and in the RHS P is replaced by $P(k)$, and where similar modifications are applied to the Y equation.

For the numerical simulations we took $h = 0.01sec$ and $\sigma = 0.001rad/\sqrt{sec}$. The sensed acceleration noise was taken as $0.01m/sec^2$, whereas the velocity error was taken, for design purposes, as a zero mean white noise with standard deviation of $1m/sec$. Since the velocity noise is usually much smaller (i.e. the main error is a bias), it was not applied in the simulations, but just affected the estimator gains. The estimator gain was designed for $\rho = 1$ and $\rho = 10^{-h/\tau}$ for decay time constant $\tau = 1sec$. We note that with the above system parameters the merit of using MKF rather than KF is not very significant, and we just study here the effect of ρ which is seen when comparing in Fig. 1 the convergence of $\hat{\theta}$ to θ for the above couple of cases. In Fig. 2 we see the estimation errors and in Fig. 3 we see the roll angle ϕ which apparently, is not affected by ρ since the estimator is fast and does not need tuning to be faster. The latter fact stems from the low noise in the sensed accelerations measurements. The next example focuses on $\rho = 1$ and explores the benefits of using MKF rather than KF in a closely related leveling problem.

7 Example–Leveling by Inertial Navigation

We next consider a numerical example that demonstrates the advantages of using the new estimator derived in the present paper, for a simplified navigation problem where a vehicle's attitude is estimated from its noisy position and velocity measurements (e.g. from GPS) utilizing inaccurate inertial sensors. The example is a modified version of the three-axis simplified navigation model of [9].

The three axis model of [9] is given by

$$\begin{array}{ll} \ddot{x} = \beta_x - g(-\phi_y), & -\dot{\phi}_y = \dot{x}/R_e + (\omega_x + \epsilon_x)\phi_z - \epsilon_y \\ \ddot{y} = \beta_y - g\phi_x, & \dot{\phi}_x = \dot{y}/R_e + (\omega_y + \epsilon_y)\phi_z + \epsilon_z \\ \ddot{z} = \beta_z, & \dot{\phi}_z = \epsilon_z \end{array} \tag{11}$$

where x, y, z are the components of the vehicle position error, ϕ_x, ϕ_y, ϕ_z are the tilt errors and $\omega_x, \omega_y, \omega_z$ are the angular rates. In [9] it was assumed that the constant bias and drifts of the accelerometers and rate sensors have been compensated via calibration and, therefore, the driving terms $\beta_x, \beta_y, \beta_z$ and $\epsilon_x, \epsilon_y, \epsilon_z$

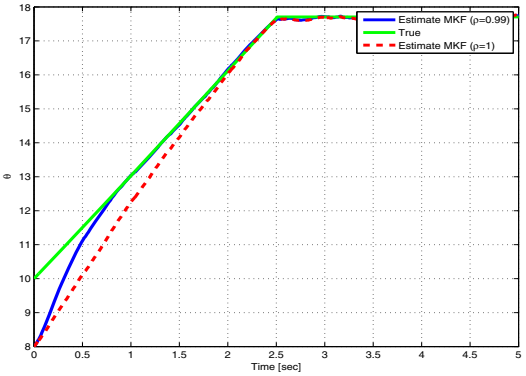


Fig. 1. True Pitch angle and its estimates and the effect of ρ

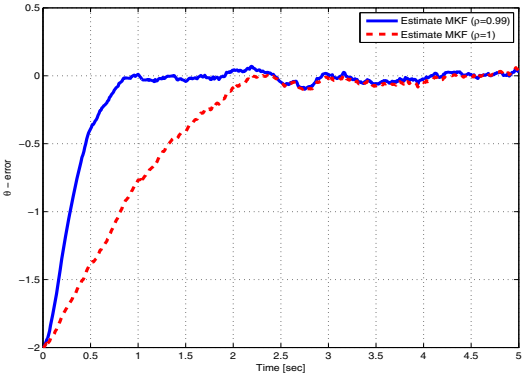


Fig. 2. Pitch angle estimation error and the effect of ρ

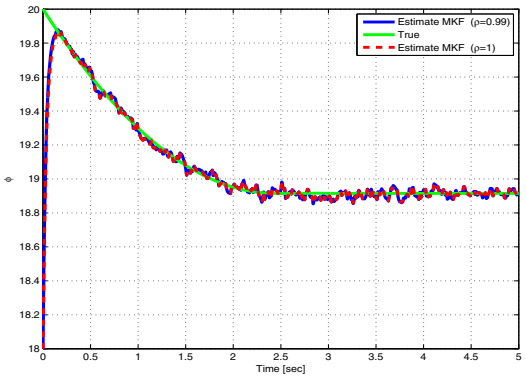


Fig. 3. True Roll angle and its estimates and the effect of ρ

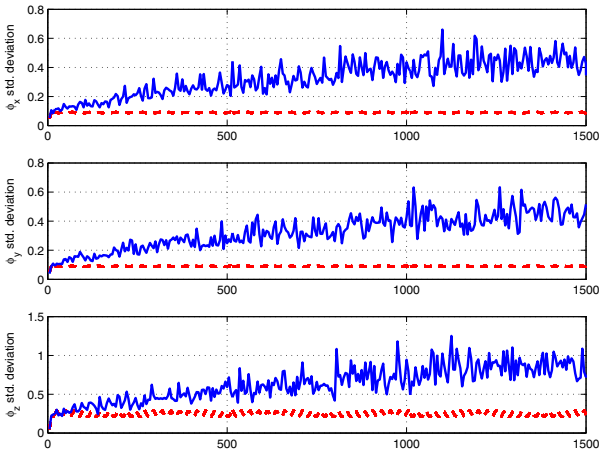


Fig. 4. Kalman Filter - 50 Monte-Carlo Runs

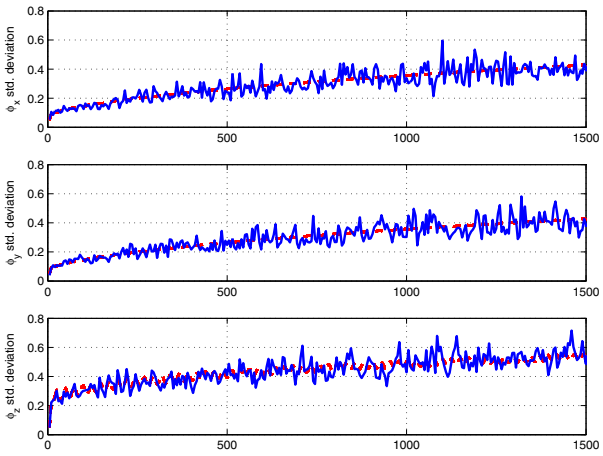


Fig. 5. State-Multiplicative Kalman Filter - 50 Monte-Carlo Runs

are white noise processes. In our case, we assume low cost noisy measurement devices for the angular rates (e.g. by differentiation of angles computed from magnetometers) and we have, therefore,added the noise terms ϵ_x to ω_x and ϵ_y to ω_y. Defining the state–vector to be $x = \left[x\ \dot{x}\ -\phi_y\ y\ \dot{y}\ \phi_x\ z\ \dot{x}\ \phi_z \right]^T$ and the measurements to be $y = \left[x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z} \right]^T$ and considering the discrete-time version of the above system with a sampling time of $h = 0.1sec$ we obtain the system of form (2), (3) where

$$A \approx I + h \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0\,0 & 0 \\ 0 & 0 & -g & 0 & 0 & 0 & 0\,0 & 0 \\ 0 & 1/R_e & 0 & 0 & 0 & 0 & 0\,0 & \omega_x \\ 0 & 0 & 0 & 0 & 1 & 0 & 0\,0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -g & 0\,0 & 0 \\ 0 & 0 & 0 & 0 & 1/R_e & 0 & 0\,0 & \omega_y \\ 0 & 0 & 0 & 0 & 0 & 0 & 0\,1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0\,0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0\,0 & 0 \end{bmatrix}, B\beta_k = \begin{bmatrix} 0\,0\,0 & 0 & 0 & 0 \\ 1\,0\,0 & 0 & 0 & 0 \\ 0\,0\,0 & 0 & -\epsilon_d & 0 \\ 0\,0\,0 & 0 & 0 & 0 \\ 0\,1\,0 & 0 & 0 & 0 \\ 0\,0\,0 & \epsilon_d & 0 & 0 \\ 0\,0\,0 & 0 & 0 & 0 \\ 0\,0\,1 & 0 & 0 & 0 \\ 0\,0\,0 & 0 & 0 & \epsilon_d \end{bmatrix} \begin{bmatrix} \beta_x \\ \beta_y \\ \beta_z \\ \epsilon_x \\ \epsilon_y \\ \epsilon_z \end{bmatrix},$$

$u_k \equiv 0$.

We also define the terms with state–dependent noise:

$$H_1 x_k \epsilon_x = \begin{bmatrix} 0_{2\times 8} & 0_{2\times 1} \\ 0_{1\times 8} & \epsilon_d \\ 0_{6\times 8} & 0_{6\times 1} \end{bmatrix} x_k \epsilon_x,\ H_2 x_k \epsilon_y = \begin{bmatrix} 0_{5\times 8} & 0_{5\times 1} \\ 0_{1\times 8} & \epsilon_d \\ 0_{3\times 8} & 0_{3\times 1} \end{bmatrix} x_k \epsilon_y$$

The covariance of the measurement noise is taken as

$$DD^T = diag\{100, 100, 100, 0.01, 0.01, 0.01\}.$$

We took $\varepsilon_d = 0.0483\ rad/\sqrt{sec}$. Although this level of random walk is very high and beyond commonly encountered practical values, it may represent cases where angular rates are obtained with very cheap and noisy components or under severe environmental conditions.

The performances of two estimators have been compared. One is the KF which ignores the state-multiplicative noise and is derived by solving the discrete-time recursive Riccati equation obtained from Lemma 2 above by nulling H and Y and by replacing P in the left-hand-side of (7) by $P(k+1)$ and by replacing Y in the left-hand-side of (8) by $Y(k+1)$. All values in the right-hand-sides of (7) and (8) correspond to time index k. Similarly, the new filter of the present paper, which we refer to as the state-multiplicative Kalman Filter (MKF) is obtained by taking the above non-zero value for H. The vehicle maneuvers are assumed to behave according $\omega_x = \omega_y = \omega_z = 0.5sin(0.5kh)$. The results of 50 Monte-Carlo simulation runs depicting the standard deviations of the tilt errors ϕ_x, ϕ_y, ϕ_z for the KF and MKF are are given in Fig. 4-5 respectively. The solid (blue) lines are the actual ensemble based standard deviations whereas the dashed (red) lines are the standard deviations predicted by the filter - namely $\sqrt{P_{3,3}}, \sqrt{P_{6,6}}, \sqrt{P_{9,9}}$. Clearly the prediction by the MKF is considerably more accurate and tighter. Moreover, the standard deviations of the tilt errors are smaller with the MKF, where the benefit of using MKF over using KF is best observed in the estimation of ϕ_z where the errors are smaller by a factor of 2 with the MKF with respect to the KF. Note also that with both filters, the errors in ϕ_z are larger than those in ϕ_x and ϕ_y due to the weaker observability in ϕ_z due to lack of the direct relation between ϕ_z and the measured velocities that exists, in contrast, between $\ddot{x}$ and ϕ_y and $\ddot{y}$ and ϕ_x.

8 Conclusions

Design considerations for a three axis leveling loop have been presented. The apparent effect of state-multiplicative noise which couple the three channels can be either ignored from, or kept in the filter design equations. The latter case, has motivated development of a new version of the Kalman filter which takes into account the state-multiplicative noise and complies with a decay rate requirement for the filtering errors. The performance of the new filter has been demonstrated in a couple of examples where one is of leveling loop design and the other of leveling using standard inertial navigation. In these examples, the superiority of the new filter, with respect to the Kalman filter, in the aspects of decay rate and response to the state-multiplicative noise has been illustrated.

References

1. Krogman, U.K.: Design Aspects, Performance Analysis and Test Results of a Strapdown Single Gyro Attitude and Heading Reference. In: Navigation Symposium, Stutgart (1984)
2. Yaesh, I., Priel, B.: Design of Leveling Loop for Marine Navigation System. IEEE Transactions on Aerospace and Electronic Systems 29, 599–604 (1993)
3. Gershon, E., Shaked, U., Yaesh, I.: H_∞ Control and Estimation of State-Multiplicative Linear Systems. LNCIS, vol. 318. Springer, Heidelberg (2005)
4. Stoica, A.-M., Yaesh, I.: Kalman Type Filtering for Discrete-Time Stochastic Systems with State-Dependent Noise. In: Proceedings of the MTNS 2008 (2008)
5. Stoica, A.–M., Dragan, V., Yaesh, I.: Kalman-type filtering for stochastic systems with state dependent noise and Markovian jumping. In: IFAC-SYSID, Saint-Malo, France, July 6-8 (2009)
6. Rotstein, H., Sznaier, M.: An Exact Solution to General Four Block Discrete-Time Mixed H_2/H_∞ problems via COnvex Optimization. IEEE Transactions on Automatic Control 43, 1475–1481 (1998)
7. Choukroun, D., Weiss, H., Bar-Itzhak, I., Oshman, Y.: Quaternion estimation from vector observations using matrix Kalman filter. In: Itzhack, Y. (ed.) Bar-Itzhack Memorial Symposium on Estimation, Navigation, and Spacecraft Control (October 2012)
8. Bar-Shalom, Y., Li, X.-R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation. John Wiley and Sons (2001)
9. Bar-Shalom, Y., Li, X.-R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation. John Wiley and Sons (2001)
10. Stoorvogel, A.A., Saberi, A.: The Discrete Algebraic Riccati Equation and Linear Matrix Inequality. Linear Algebra and its Applications 274, 317–365 (1998)

A Novel Imaging Measurement Model for Vision and Inertial Navigation Fusion with Extended Kalman Filtering

Leandro Ribeiro Lustosa and Jacques Waldmann

Instituto Tecnológico de Aeronáutica, São José dos Campos, Brazil
{lustosa,jacques}@ita.br

Abstract. It is well-known that stand-alone inertial navigation systems (INS) have their errors diverging with time. The traditional approach for solving such incovenience is to resort to position and velocity aiding such as global navigation satellite systems (GNSS) signals. However, misalignment errors in such fusion architecture are not observable in the absence of maneuvers. This investigation develops a novel sighting device (SD) model for vision-aided inertial navigation for use in psi-angle error based extended Kalman filtering by means of observations of *a priori* mapped landmarks. Additionally, the psi-angle error model is revisited and an extended Kalman filter datasheet-based tuning is explained. Results are obtained by computer simulation, where an unmanned aerial vehicle flies a known trajectory with inertial sensor measurements corrupted by a random constant model. Position and velocity errors, misalignment, accelerometer bias, rate-gyro drift and GNSS clock errors with respect to ground-truth are estimated by means of INS/GNSS/SD fusion and tested for statistical consistency.

Keywords: inertial navigation, vision, Kalman filter, unmanned aerial vehicles.

1 Introduction

Advances in microelectromechanical inertial sensors (MEMs) made low-cost inertial navigation systems (INS) commercially available. On the other hand, their errors quickly diverge with time and set an upper bound on the duration of autonomous operations and thus such systems become improper for use in low-cost unmanned aerial vehicle (UAV) missions. The traditional approach for solving such inconvenience is to resort to a global navigation satellite system (GNSS) receiver as position and velocity aiding device. Hence, INS/GNSS fusion yields bounded navigation errors. However, misalignment errors in such fusion architecture are not observable in the absence of maneuvers [1,2]. In the light of such restriction, the present study develops a novel model for INS/GNSS and sighting device (SD) integration for use in outdoor navigation with known landmarks. In general, outdoor navigation in structured environments requires some sort of

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_15

road-following. Herein, *a priori* mapped landmarks are imaged by a camera and tracked in the image plane to aid the INS.

An initial study on the matter was conducted in [3], which developed two distinct strategies for INS/SD fusion by means of psi-angle error model [4] based extended Kalman filtering (EKF). One of them is the inspiration for this paper and explores a relationship between the INS errors and the position of a landmark in the field of view relative to the line of sight (LOS) of the SD, after the latter is aimed at the assumed position of the landmark. The shortcoming of such procedure is the restriction to have the camera maintaining LOS pointing to the landmark. Such restriction has been recently overcome by means of resorting to a generalized SD model in which the difference between measured and estimated positions of the tracked landmark in the plane of image are correlated with INS errors [5].

However, the psi-angle error framework has received only a modicum of attention in INS/SD integration [6] since [3]. The present investigation revisits such scheme, which has proven sucessful for INS/GNSS fusion [7], by means of developing an INS/SD fusion formulation within the psi-angle error based EKF framework.

Initially, the INS psi-angle error model is revisited, and a datasheet-based nominal EKF tuning is explored alongside. Ultimately, an INS/GNSS/SD EKF-based fusion strategy is proposed for the estimation of navigation and sensors errors, and evaluated by means of Monte Carlo simulation and statistical consistency tests [8].

2 Reference Frames and Earth Model

Reference frames and the Earth model are here briefly discussed. The WGS-84 ellipsoid has been used due to its accuracy and simplicity [9]. The local reference frames at the true and computed positions differ [4], and are respectively denoted by $T = \{\hat{\boldsymbol{t}}_1, \hat{\boldsymbol{t}}_2, \hat{\boldsymbol{t}}_3\}$ and $C = \{\hat{\boldsymbol{c}}_1, \hat{\boldsymbol{c}}_2, \hat{\boldsymbol{c}}_3\}$ (see figure 1).

Additionally, equally important frames are I, E, B and P, respectively, inertial, Earth-fixed, vehicle body and platform coordinate systems. The latter is the local reference frame computed by the inertial navigation system at its estimated position and affected by attitude estimation errors [4].

3 Mathematical Notation

The chosen notation [10] is illustrated by table 1.

Furthermore, the decomposition of a vector $\boldsymbol{v} \in \mathbb{R}^3$ into its components in a R coordinate system is denoted by means of the right subscript position, e.g.

$$\boldsymbol{v}_R = \begin{pmatrix} v_{r1} & v_{r2} & v_{r3} \end{pmatrix}^T \quad . \tag{1}$$

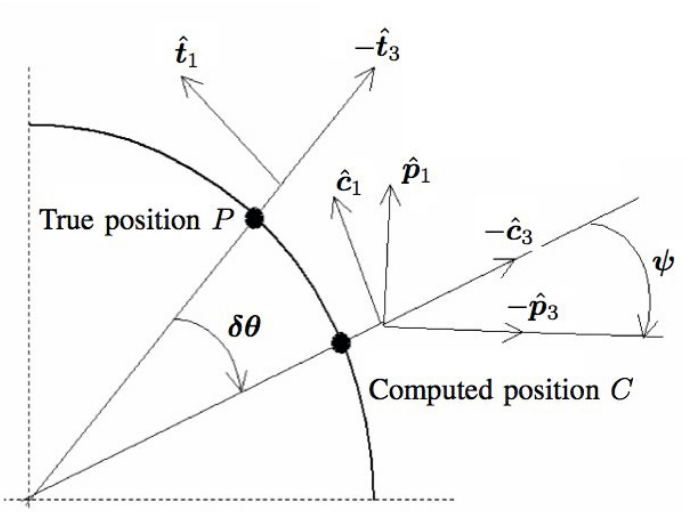


Fig. 1. Illustration of true (P) and computed (C) positions; and platform (P), computed (C) and true (T) reference frames

4 Kalman Filter Formulation

For the purpose of EKF-based INS/GNSS/SD fusion, the INS psi-angle error, GNSS receiver and camera linear models are outlined, i.e., linear state and covariance propagation and update are formulated.

4.1 INS Psi-Angle Error Model Revisited

Consider strapdown accelerometers and rate-gyros measurements corrupted, respectively, by unknown constant bias $\boldsymbol{\nabla}$ and drift $\boldsymbol{\epsilon}$, modelled as random normal variables with σ_∇ and σ_ϵ standard deviations. Additionally, additive zero-mean white additive noise $\boldsymbol{\omega}_{\text{accel}}$ and $\boldsymbol{\omega}_{\text{gyro}}$ are considered with σ_{accel} and σ_{gyro} standard deviations. The measured specific force $\boldsymbol{A}_{\mathbf{sp,m}}$ is given [4] by a rotation of $\boldsymbol{A}_{\mathbf{sp}}$ by the misalignment vector $\boldsymbol{\psi}$ (see figure 1) from C to P reference frame, and biased by $\boldsymbol{\nabla}$ according to

$$\boldsymbol{A}_{\mathbf{sp,m}} = \boldsymbol{\nabla} + \boldsymbol{A}_{\mathbf{sp}} - \boldsymbol{\psi} \times \boldsymbol{A}_{\mathbf{sp}} + \boldsymbol{\omega}_{\text{accel}} \tag{2}$$

whereas [4]

$$\boldsymbol{A}_{\mathbf{sp}} = \overset{ii}{\boldsymbol{R}} - \boldsymbol{g}(\boldsymbol{R}) - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{R}) \tag{3}$$

and

$$\boldsymbol{A}_{\mathbf{sp,m}} = \overset{ii}{\hat{\boldsymbol{R}}}_{\mathbf{INS}} - \boldsymbol{g}(\hat{\boldsymbol{R}}_{\mathbf{INS}}) - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \hat{\boldsymbol{R}}_{\mathbf{INS}}) \tag{4}$$

Table 1. Kinematics Notation

Notation	Meaning
$\boldsymbol{p}^{X/Y}$	Position of point X w.r.t. point Y
$\boldsymbol{R}$	INS position with respect to Earth centre
$\hat{\boldsymbol{R}}_{\mathbf{INS}}$	INS-computed position with respect to Earth centre
$\overset{e}{\boldsymbol{R}} = \boldsymbol{v}$	Terrestrial velocity
$\overset{e}{\hat{R}}_{\mathbf{INS}} = \hat{\boldsymbol{v}}_{\mathbf{INS}}$	INS-computed terrestrial velocity
$\overset{ss}{\boldsymbol{R}}$	Acceleration w.r.t. S reference frame
$\boldsymbol{\omega}^{\boldsymbol{xy}}$	Angular velocity of X coordinate frame w.r.t. Y frame
$\boldsymbol{A}_{\mathbf{sp}}$	Specific force
$\boldsymbol{g}(\cdot)$	Earth gravity at designated point
D_B^A	Direction cosine matrix: rotates from A coordinate frame into alignment with B frame

where $\boldsymbol{\psi}$, $\boldsymbol{g}(\boldsymbol{R})$, $\boldsymbol{g}(\hat{\boldsymbol{R}}_{\mathbf{INS}})$, $\boldsymbol{\Omega}$, $\boldsymbol{R}$, $\hat{\boldsymbol{R}}_{\mathbf{INS}}$ denote, respectively, the misalignment rotation vector from the computed to the platform reference frame, Earth gravity at the true and computed positions, Earth angular velocity, and true and computed positions with respect to the Earth centre.

If $\boldsymbol{\delta p}$ and $\boldsymbol{\delta v}$ are defined as INS computed errors in position and velocity, i.e.,

$$\begin{cases} \boldsymbol{\delta p} = \hat{\boldsymbol{R}}_{\mathbf{INS}} - \boldsymbol{R} \\ \boldsymbol{\delta v} = \hat{\boldsymbol{v}}_{\mathbf{INS}} - \boldsymbol{v} \end{cases} \tag{5}$$

then equations 2, 3 and 4 imply

$$\boldsymbol{\nabla} - \boldsymbol{\psi} \times \boldsymbol{A}_{\mathbf{sp}} + \boldsymbol{\omega}_{\text{accel}} = \overset{ii}{\boldsymbol{\delta p}} - \underbrace{[\boldsymbol{g}(\overset{ii}{\hat{\boldsymbol{R}}}_{\mathbf{INS}}) - \boldsymbol{g}(\boldsymbol{R})]}_{\boldsymbol{\delta g}} - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{\delta p}) \tag{6}$$

which can be rewritten as

$$\begin{aligned} \boldsymbol{\nabla} - \boldsymbol{\psi} \times \boldsymbol{A}_{\mathbf{sp}} + \boldsymbol{\omega}_{\text{accel}} &= \left(\overset{ie}{\boldsymbol{\delta p}} + \boldsymbol{\Omega} \times \overset{i}{\boldsymbol{\delta p}} \right) - \boldsymbol{\delta g} - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{\delta p}) = \\ &= \overset{i}{\boldsymbol{\delta v}} + \boldsymbol{\Omega} \times \overset{i}{\boldsymbol{\delta p}} - \boldsymbol{\delta g} - \boldsymbol{\Omega} \times (\boldsymbol{\Omega} \times \boldsymbol{\delta p}) = \overset{c}{\boldsymbol{\delta v}} + \boldsymbol{\omega}^{\boldsymbol{ci}} \times \boldsymbol{\delta v} + \boldsymbol{\Omega} \times \boldsymbol{\delta v} - \boldsymbol{\delta g} \ . \end{aligned} \tag{7}$$

Ultimately,

$$\overset{c}{\boldsymbol{\delta v}} = \boldsymbol{\delta g} - (2\boldsymbol{\Omega} + \boldsymbol{\omega}^{\boldsymbol{ce}}) \times \boldsymbol{\delta v} + \boldsymbol{A}_{\mathbf{sp}} \times \boldsymbol{\psi} + \boldsymbol{\nabla} + \boldsymbol{\omega}_{\text{accel}} \ . \tag{8}$$

On the other hand, the derivative of $\boldsymbol{\delta p}$ with respect to frame C yields

$$\overset{c}{\boldsymbol{\delta p}} = \overset{e}{\boldsymbol{\delta p}} - \boldsymbol{\omega}^{\boldsymbol{ce}} \times \boldsymbol{\delta p} = \boldsymbol{\delta v} - \boldsymbol{\omega}^{\boldsymbol{ce}} \times \boldsymbol{\delta p} \ . \tag{9}$$

Additionally, it can be shown that [4]

$$\boldsymbol{\epsilon} = -\overset{i}{\boldsymbol{\psi}} - \boldsymbol{\omega}_{\text{gyro}} \tag{10}$$

hence

$$\overset{c}{\boldsymbol{\psi}} = \overset{i}{\boldsymbol{\psi}} - \boldsymbol{\omega^{ci}} \times \boldsymbol{\psi} = -\boldsymbol{\epsilon} - (\boldsymbol{\Omega} + \boldsymbol{\omega^{ce}}) \times \boldsymbol{\psi} - \boldsymbol{\omega}_{\text{gyro}} \ . \tag{11}$$

Finally, according to [4], assuming $\boldsymbol{\nabla}_B$ and $\boldsymbol{\epsilon}_B$ are random constants, representing vector equations 8, 9 and 11 in the C coordinate frame, and then employing the D_P^B estimated by the INS, the psi-angle-based INS error dynamics model is formulated as

$$\boldsymbol{\delta \dot{p}}_C = \boldsymbol{\delta v}_C - [\boldsymbol{\omega}_C^{\boldsymbol{ce}} \times] \boldsymbol{\delta p}_C \tag{12}$$

$$\boldsymbol{\delta \dot{v}}_C = \boldsymbol{\delta g}_C - [(2\boldsymbol{\Omega}_C + \boldsymbol{\omega}_C^{\boldsymbol{ce}}) \times] \boldsymbol{\delta v}_C + [D_P^B \boldsymbol{A}_{\mathbf{sp,m,B}} \times] \boldsymbol{\psi}_C + D_P^B \boldsymbol{\nabla}_B + D_P^B \boldsymbol{\omega}_{\text{accel,B}} \tag{13}$$

$$\boldsymbol{\dot{\psi}}_C = -[(\boldsymbol{\Omega}_C + \boldsymbol{\omega}_C^{\boldsymbol{ce}}) \times] \boldsymbol{\psi}_C - D_P^B \boldsymbol{\epsilon}_B - D_P^B \boldsymbol{\omega}_{\text{gyro,B}} \tag{14}$$

$$\dot{\boldsymbol{\nabla}}_B = \mathbf{0}_{3\times 1} \tag{15}$$

$$\dot{\boldsymbol{\epsilon}}_B = \mathbf{0}_{3\times 1} \tag{16}$$

where the notation $[\boldsymbol{\zeta}_R \times]$ means

$$[\boldsymbol{\zeta}_R \times] = \begin{bmatrix} 0 & -\zeta_{r3} & \zeta_{r2} \\ \zeta_{r3} & 0 & -\zeta_{r1} \\ -\zeta_{r2} & \zeta_{r1} & 0 \end{bmatrix} \tag{17}$$

and $\boldsymbol{\delta g}_C$ is approximated by [11]

$$\boldsymbol{\delta g}_C = \frac{g_{\text{e}} R_{\text{e}}^2}{(R_{\text{e}} + h_{\text{c}})^3} \begin{pmatrix} -1 \\ -1 \\ 2 \end{pmatrix} \tag{18}$$

with g_{e} and R_{e} denoting, respectively, the gravity and the radius of the Earth as computed with the WGS-84 Earth model, at the geographic location of the INS sensors, according to [11]

$$R_{\text{e}} = R_0(1 - esin^2\lambda_{\text{c}}) \tag{19}$$

and

$$g_{\text{e}} = (1 + 0.0053sin^2\lambda_c) g_0 \tag{20}$$

where R_0, e, g_0 and λ_{c} denote, respectively, the Earth equatorial radius, eccentricity and equatorial gravity, and the INS-computed latitude. The Earth model parameters are found in [9].

In the light of the foregoing development, it is convenient to define the EKF state vector as

$$\boldsymbol{x}_{\text{EKF}} = \left(\boldsymbol{\delta p}_C^T\ \boldsymbol{\delta v}_C^T\ \boldsymbol{\psi}_C^T\ \boldsymbol{\nabla}_B^T\ \boldsymbol{\epsilon}_B^T\ c\Delta t\right)^T \tag{21}$$

where $c\Delta t$ denotes the random constant GNSS clock error model with $\sigma_{c\Delta t}$ standard deviation, whose compensation is fundamental in INS/GNSS integration [11].

The zero-order hold (ZOH) discretization of equations 12, 13, 14, 15 and 16 yields the error-state transition matrix F_k and process noise covariance Q_k for the EKF. In practice, the discretization which yields Q_k is [12]

$$Q_k = G_k \begin{bmatrix} \sigma^2_{\text{accel}} I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \sigma^2_{\text{gyro}} I_{3\times 3} \end{bmatrix} \Delta t G_k^T \tag{22}$$

where

$$G_k = \begin{bmatrix} 0_{3\times 3} & 0_{3\times 3} \\ D_P^B & 0_{3\times 3} \\ 0_{3\times 3} & -D_P^B \\ 0_{7\times 3} & 0_{7\times 3} \end{bmatrix} \tag{23}$$

and Δt denotes the discretization sample time. Finally, EKF tuning is addressed with inertial sensors' datasheet specifications.

4.2 GNSS Tightly Coupled Integration

GNSS integration equations are listed in the following without further explanations due to the broad extension of available literature on the matter [7]. Integration is performed in a tightly coupled architecture involving pseudorange and deltarange measurements. For each satellite S_i, pseudorange and deltarange innovations are incorporated by means of equations 24 and 25, where $\hat{\boldsymbol{u}}_i$ denotes the estimated line-of-sight (LOS) unit vector from the user's receiver antenna to satellite S_i. Pseudorange and deltarange measurements are corrupted by additive Gaussian noise $\boldsymbol{\omega}_{S_i,\text{p}} \sim N(0, \sigma_\text{p})$ and $\boldsymbol{\omega}_{S_i,\text{v}} \sim N(0, \sigma_\text{v})$. The antenna lever arm has been considered as exactly compensated. Furthermore, GNSS clock error dynamics is modelled according to equation 26. In practice, a more complex model that accounts for clock drift should be implemented [7]. However, for the sake of simplicity, the present work implements a simpler model and focuses on camera integration.

$$\boldsymbol{p}^{S_i/P} - \boldsymbol{p}^{S_i/C} = \hat{\boldsymbol{u}}_i \cdot \boldsymbol{\delta p} + c\Delta t + \boldsymbol{\omega}_{S_i,\text{p}} \tag{24}$$

$$\overset{\boldsymbol{e}}{\boldsymbol{p}}^{S_i/P} - \overset{\boldsymbol{e}}{\boldsymbol{p}}^{S_i/C} = \hat{\boldsymbol{u}}_i \cdot \boldsymbol{\delta v} + \boldsymbol{\omega}_{S_i,\text{v}} \tag{25}$$

$$\frac{d}{dt} c\Delta t = 0\ . \tag{26}$$

4.3 Sighting Device

As previously stated, the basic idea of the proposed INS/SD fusion architecture is based upon tracking mapped landmarks $L_i \in L, i = 1..N_\mathrm{l}$, each one with *a priori* known LLA coordinates (latitude, longitude, altitude). For each L_i, its position with respect to the camera V, $\boldsymbol{p}^{L_i/V}$, (see figure 2) is described in the V coordinate frame as [5]

$$\boldsymbol{p}_V^{L_i/V} = D_V^B[D_B^T D_T^E(\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P) - \boldsymbol{p}_B^{V/P}] \ . \tag{27}$$

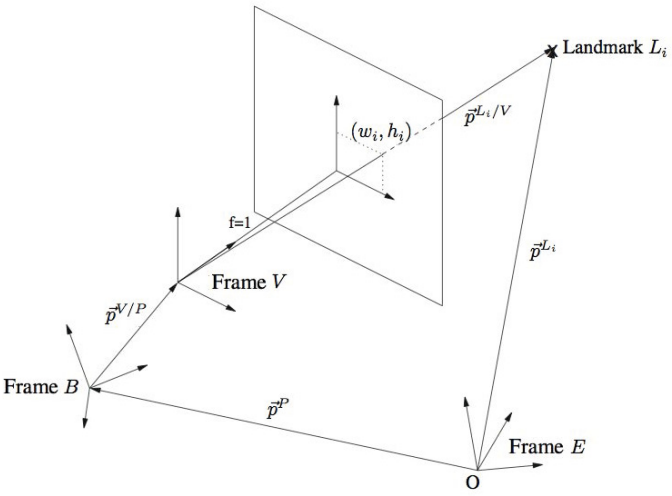


Fig. 2. Perspective projection geometry in the plane of image. Adapted from [5].

The camera V is assumed installed next to the inertial sensors' position P and has its axes aligned according to figure 3, thus

$$\boldsymbol{p}_V^{L_i/V} = D_V^B D_B^T D_T^E(\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P) \ . \tag{28}$$

Hence the adimensional normalized measurement[1] $\boldsymbol{z}_{L_i} = (w_i, h_i)$ provided by the camera is given by

$$\boldsymbol{z}_{L_i} = \mathit{\Pi} \frac{\boldsymbol{p}_V^{L_i/V}}{[1\,0\,0] \cdot \boldsymbol{p}_V^{L_i/V}} + \boldsymbol{\omega}_\mathrm{cam} \tag{29}$$

where $\boldsymbol{\omega}_\mathrm{cam} \in M(\mathbb{R})_{2\times 1}$ is white gaussian noise with standard deviation σ_cam, and $\mathit{\Pi}$ is defined by

$$\mathit{\Pi} = \begin{bmatrix} 0\,1\,0 \\ 0\,0\,1 \end{bmatrix} \ . \tag{30}$$

[1] It is assumed, without loss of generality, camera focal length $f = 1$.

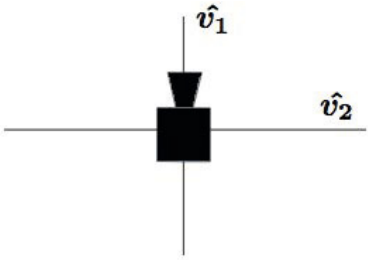


Fig. 3. Definition of camera coordinate system (top view)

$\boldsymbol{\omega}_{\text{cam}}$ shall be modelled to account for uncertainties in target tracking algorithms, servomechanism control and misalignment between camera and vehicle body.[2] The estimated normalized measurement vector is estimated by means of the navigation algorithm [13] computed variables (i.e., position, velocity and attitude) by

$$\hat{\boldsymbol{z}}_{L_i} = \Pi \frac{\boldsymbol{p}_{V_c}^{L_i/C}}{\begin{bmatrix}1 & 0 & 0\end{bmatrix} \cdot \boldsymbol{p}_{V_c}^{L_i/C}} \tag{31}$$

where V_c is the computed camera reference frame. Similarly to equation 28, $\boldsymbol{p}_{V_c}^{L_i/C}$ is expressed as

$$\boldsymbol{p}_{V_c}^{L_i/C} = D_{V_c}^{B_c} D_{B_c}^{C} D_{C}^{E} (\boldsymbol{p}_{E}^{L_i} - \boldsymbol{p}_{E}^{C}) \ . \tag{32}$$

The orientation of the camera V in relation to the vehicle body B is assumed known with great accuracy, thus $D_{B_c}^{V_c} = D_{B}^{V}$. In addition, notice that INS attitude estimates are employed and thus follows the approximation $D_{C}^{B_c} \approx D_{P}^{B}$. In this scope, equation 32 can be rewritten as

$$\boldsymbol{p}_{V_c}^{L_i/C} = D_{V}^{B} D_{B}^{P} D_{C}^{E} (\boldsymbol{p}_{E}^{L_i} - \boldsymbol{p}_{E}^{C}) \ . \tag{33}$$

Notice that all terms in the right-hand side of equation 33 are available from INS navigation algorithm and $\hat{\boldsymbol{z}}_{L_i}$ can be readily computed. For use in the EKF, the difference $\boldsymbol{r}_{L_i} = \boldsymbol{z}_{L_i} - \hat{\boldsymbol{z}}_{L_i}$, and how it relates to navigation errors, is explored in the following.

It has been assumed that $\boldsymbol{r}_{L_i}$ is function solely of $\boldsymbol{\delta p}$ and $\boldsymbol{\psi}$, disregarding electrooptical distortions in the sighting device. Hence, the Jacobian $J = \frac{\partial \boldsymbol{r}_{L_i}}{\partial \boldsymbol{x}}$ is in the sparse form

$$J = - \begin{bmatrix} \frac{\partial \hat{\boldsymbol{z}}_{L_i}}{\partial \boldsymbol{\delta p}_C} & 0_{2\times 3} & \frac{\partial \hat{\boldsymbol{z}}_{L_i}}{\partial \boldsymbol{\psi}_C} & 0_{2\times 7} \end{bmatrix} \ . \tag{34}$$

[2] Servomechanism control and misalignment between camera and vehicle body errors usually cannot be accurately modelled as white gaussian noise hence inflation in Kalman filter noise covariance statistics should take place in practice

First, $\frac{\partial \hat{\boldsymbol{z}}_{L_i}}{\partial \boldsymbol{\delta p}_C}$ is calculated noticing that equation 33 can be rewritten as

$$\begin{aligned}
\boldsymbol{p}_{V_C}^{L_i/C} = D_V^B D_B^P D_C^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^C) = D_V^B (D_B^T D_T^C D_C^P)(D_C^T D_T^E)(\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^C) \approx \\
\approx D_V^B D_B^T (I_{3\times 3} + [\boldsymbol{\psi}_C \times]) D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^C) = \\
= D_V^B D_B^T (I_{3\times 3} + [\boldsymbol{\psi}_C \times]) D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P - \boldsymbol{\delta p}_E) = \\
= D_V^B D_B^T D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P) + D_V^B D_B^T [\boldsymbol{\psi}_C \times] D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P) + \cdots \\
\cdots - D_V^B D_B^T (I_{3\times 3} + [\boldsymbol{\psi}_C \times]) D_T^E D_E^C \boldsymbol{\delta p}_C \approx \\
\approx \boldsymbol{p}_V^{L_i/V} + D_V^B D_B^T [\boldsymbol{\psi}_C \times] D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P) - D_V^B D_B^T (I_{3\times 3} + [\boldsymbol{\delta\theta}_C \times]) \boldsymbol{\delta p}_C \approx \\
\approx \boldsymbol{p}_V^{L_i/V} + D_V^B D_B^T [\boldsymbol{\psi}_C \times] D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P) - D_V^B D_B^T \boldsymbol{\delta p}_C \ . \quad (35)
\end{aligned}$$

Therefore, considering the partial derivative $\frac{\partial \hat{\boldsymbol{z}}_{L_i}}{\partial \boldsymbol{\delta p}_C}$ at point $\boldsymbol{\psi}_C = \mathbf{0}$, above equations deliver

$$\boldsymbol{p}_{V_C}^{L_i/C} = \boldsymbol{p}_V^{L_i/V} - D_V^B D_B^T \boldsymbol{\delta p}_C \ . \quad (36)$$

For the computation of $\frac{\partial \hat{\boldsymbol{z}}_{L_i}}{\partial \boldsymbol{p}_C}$, $\hat{\boldsymbol{z}}_{L_i}$ is rewritten as

$$\hat{\boldsymbol{z}}_{L_i} = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix}, \quad z_1 = \frac{z_{1,\text{num}}}{z_{\text{den}}}, \quad \frac{z_{2,\text{num}}}{z_{\text{den}}} \quad (37)$$

$$z_{1,\text{num}} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} (\boldsymbol{p}_V^{L_i/V} - D_V^B D_B^T \boldsymbol{\delta p}_C) \quad (38)$$

$$z_{2,\text{num}} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} (\boldsymbol{p}_V^{L_i/V} - D_V^B D_B^T \boldsymbol{\delta p}_C) \quad (39)$$

$$z_{\text{den}} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} (\boldsymbol{p}_V^{L_i/V} - D_V^B D_B^T \boldsymbol{\delta p}_C) \ . \quad (40)$$

Hence,

$$\frac{\partial \hat{\boldsymbol{z}}_{L_i}}{\partial \boldsymbol{\delta p}_C} = \begin{bmatrix} \frac{\partial z_1}{\partial \delta p_{c1}} & \frac{\partial z_1}{\partial \delta p_{c2}} & \frac{\partial z_1}{\partial \delta p_{c3}} \\ \frac{\partial z_2}{\partial \delta p_{c1}} & \frac{\partial z_2}{\partial \delta p_{c2}} & \frac{\partial z_2}{\partial \delta p_{c3}} \end{bmatrix} \quad (41)$$

where

$$\frac{\partial z_k}{\partial \delta p_{cj}} = \frac{1}{z_{\text{den}}^2} \left(z_{\text{den}} \frac{\partial z_{\text{k,num}}}{\partial \delta p_{cj}} - z_{\text{k,num}} \frac{\partial z_{\text{den}}}{\partial \delta p_{cj}} \right), \quad \begin{matrix} k = 1..2 \\ j = 1..3 \end{matrix} \quad (42)$$

and

$$\begin{pmatrix} \frac{\partial z_{1,\text{num}}}{\partial \delta p_{c1}} \\ \frac{\partial z_{2,\text{num}}}{\partial \delta p_{c1}} \\ \frac{\partial z_{\text{den}}}{\partial \delta p_{c1}} \end{pmatrix} = - \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} D_V^B D_B^T \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad (43)$$

$$\begin{pmatrix} \frac{\partial z_{1,\text{num}}}{\partial \delta p_{c2}} \\ \frac{\partial z_{2,\text{num}}}{\partial \delta p_{c2}} \\ \frac{\partial z_{\text{den}}}{\partial \delta p_{c2}} \end{pmatrix} = - \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} D_V^B D_B^T \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \quad (44)$$

$$\begin{pmatrix} \frac{\partial z_{1,\mathrm{num}}}{\partial \delta p_{c3}} \\ \frac{\partial z_{2,\mathrm{num}}}{\partial \delta p_{c3}} \\ \frac{\partial z_{\mathrm{den}}}{\partial \delta p_{c3}} \end{pmatrix} = - \begin{bmatrix} 0\,1\,0 \\ 0\,0\,1 \\ 1\,0\,0 \end{bmatrix} D_V^B D_B^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \ . \tag{45}$$

Above computations deliver the first 2×3 block of the Jacobian in equation 34. Similarly, for the computation of the remaining nonzero block, notice that the $\boldsymbol{\psi}$ misalignment affects the camera measurement (for small angles) according to

$$\boldsymbol{p}_{V_C}^{L_i/C} = \boldsymbol{p}_V^{L_i/V} + D_V^B D_B^T [\boldsymbol{\psi}_C \times] D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P) \ . \tag{46}$$

By means of equation 46 and the same strategy used for the computation of the Jacobian with respect to position, follows

$$\hat{\boldsymbol{z}}_{L_i} = \begin{pmatrix} \gamma_1 \\ \gamma_2 \end{pmatrix}, \quad \gamma_1 = \frac{\gamma_{1,\mathrm{num}}}{\gamma_{\mathrm{den}}}, \quad \gamma_2 = \frac{\gamma_{2,\mathrm{num}}}{\gamma_{\mathrm{den}}} \tag{47}$$

$$\gamma_{1,\mathrm{num}} = \begin{bmatrix} 0\,1\,0 \end{bmatrix} (\boldsymbol{p}_V^{L_i/V} + D_V^B D_B^T [\boldsymbol{\psi}_C \times] D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P)) \tag{48}$$

$$\gamma_{2,\mathrm{num}} = \begin{bmatrix} 0\,0\,1 \end{bmatrix} (\boldsymbol{p}_V^{L_i/V} + D_V^B D_B^T [\boldsymbol{\psi}_C \times] D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P)) \tag{49}$$

$$\gamma_{\mathrm{den}} = \begin{bmatrix} 1\,0\,0 \end{bmatrix} (\boldsymbol{p}_V^{L_i/V} + D_V^B D_B^T [\boldsymbol{\psi}_C \times] D_T^E (\boldsymbol{p}_E^{L_i} - \boldsymbol{p}_E^P)) \tag{50}$$

$$\frac{\partial \hat{\boldsymbol{z}}_{L_i}}{\partial \boldsymbol{\psi}_C} = \begin{bmatrix} \frac{\partial \gamma_1}{\partial \psi_{c1}} & \frac{\partial \gamma_1}{\partial \psi_{c2}} & \frac{\partial \gamma_1}{\partial \psi_{c3}} \\ \frac{\partial \gamma_2}{\partial \psi_{c1}} & \frac{\partial \gamma_2}{\partial \psi_{c2}} & \frac{\partial \gamma_2}{\partial \psi_{c3}} \end{bmatrix} \tag{51}$$

$$\frac{\partial \gamma_k}{\partial \psi_{cj}} = \frac{1}{\gamma_{\mathrm{den}}^2} \left(\gamma_{\mathrm{den}} \frac{\partial \gamma_{k,\mathrm{num}}}{\partial \psi_{cj}} - \gamma_{k,\mathrm{num}} \frac{\partial \gamma_{\mathrm{den}}}{\partial \psi_{cj}} \right), \quad \begin{matrix} k = 1..2 \\ j = 1..3 \end{matrix} \tag{52}$$

$$\begin{pmatrix} \frac{\partial \gamma_{1,\mathrm{num}}}{\partial \psi_{c1}} \\ \frac{\partial \gamma_{2,\mathrm{num}}}{\partial \psi_{c1}} \\ \frac{\partial \gamma_{\mathrm{den}}}{\partial \psi_{c1}} \end{pmatrix} = \begin{bmatrix} 0\,1\,0 \\ 0\,0\,1 \\ 1\,0\,0 \end{bmatrix} D_V^B D_B^P \begin{bmatrix} 0\,0\,\ \ 0 \\ 0\,0\,-1 \\ 0\,1\,\ \ 0 \end{bmatrix} D_C^E \boldsymbol{p}_E^{L_i/C} \tag{53}$$

$$\begin{pmatrix} \frac{\partial \gamma_{1,\mathrm{num}}}{\partial \psi_{c2}} \\ \frac{\partial \gamma_{2,\mathrm{num}}}{\partial \psi_{c2}} \\ \frac{\partial \gamma_{\mathrm{den}}}{\partial \psi_{c2}} \end{pmatrix} = \begin{bmatrix} 0\,1\,0 \\ 0\,0\,1 \\ 1\,0\,0 \end{bmatrix} D_V^B D_B^P \begin{bmatrix} \ \ 0\,\ 0\,1 \\ \ \ 0\,\ 0\,0 \\ -1\,0\,0 \end{bmatrix} D_C^E \boldsymbol{p}_E^{L_i/C} \tag{54}$$

$$\begin{pmatrix} \frac{\partial \gamma_{1,\mathrm{num}}}{\partial \psi_{c3}} \\ \frac{\partial \gamma_{2,\mathrm{num}}}{\partial \psi_{c3}} \\ \frac{\partial \gamma_{\mathrm{den}}}{\partial \psi_{c3}} \end{pmatrix} = \begin{bmatrix} 0\,1\,0 \\ 0\,0\,1 \\ 1\,0\,0 \end{bmatrix} D_V^B D_B^P \begin{bmatrix} 0\,-1\,0 \\ 1\,\ \ 0\,\ 0 \\ 0\,\ \ 0\,\ 0 \end{bmatrix} D_C^E \boldsymbol{p}_E^{L_i/C} \ . \tag{55}$$

Above computations deliver the Jacobian J which quantitatively describes how navigation errors affect camera pointing and is used as sensor model in the extended Kalman filter.

$$\boldsymbol{r}_{L_i} = J \boldsymbol{x}_{\mathrm{EKF}} + \boldsymbol{\omega}_{\mathrm{cam}} \ . \tag{56}$$

5 Simulation Results

Consider a simulated scenario where an unmanned aerial vehicle (UAV), initially positioned at $LLA = (0, 0, 100\text{m})$, equipped with 3-axis sensitive triads of strapdown rate-gyros and accelerometers, a GNSS receiver and a camera flies a trajectory with cruising speed 300m/s toward North and altitude 100m. Sensors specifications are illustrated by table 2.

Table 2. Sensors Imperfections

INS	σ_{accel}	σ_{∇}	σ_{gyro}	σ_{ϵ}
	1mg	25mg	1^{o}/h	25^{o}/h
GNSS/SD	$\sigma_{c\Delta t}$	σ_{v}	σ_{p}	σ_{cam}
	300m	0.01m/s	15m	0.01

GNSS/SD updates were made at 100ms intervals. After the instant $t = 5$sec, EKF estimated navigation variables and sensors errors are used for in-flight correction of the INS computed position, velocity and attitude and to calibrate the inertial sensors and GNSS receiver clock error. INS correction is of utmost importance in such systems due to the linearization of the INS error dynamics at the computed INS navigation solution.

With respect to the INS navigation algorithm, [13] and [14] provide cost-effective multiple-rate integration methods to compute position and velocity, and attitude, respectively. However, such algorithms incur in errors in position, velocity and attitude, commonly known in the literature as scrolling, sculling [16] and coning [15]. Scrolling errors added to the position channel process noise (see equation 22) precludes EKF optimality, and hence *ad-hoc* Q_k inflation takes place in the following manner

$$Q_k \leftarrow Q_k + 10^{-1}\Delta t \begin{bmatrix} I_{3\times 3} & 0_{3\times 13} \\ 0_{13\times 3} & 0_{13\times 13} \end{bmatrix} \quad . \tag{57}$$

On top of that, the initial extended Kalman filter covariance matrix $P(0|0)$ is also inflated. Such procedure is commonplace in real applications since initial errors are often not known. This practice yields a non-optimal pessimist filter initiation which can be detected by substantial differences in root mean squared (RMS) estimation and EKF computed covariance during the KF's early working stages as can be seen later on.

The impact of the number of available landmarks on INS/GNSS/SD system performance will be evaluated by comparing single-sided and double-sided observation scenarios. Only one landmark update is made at each updating step of the EKF. In the case of single-sided observations, only landmark L_1 is considered in the filter update. In the double-sided observation mode, L_1 and L_2 are alternately available for the update stage. Landmarks are defined in table 3. The vehicle's trajectory and the observed landmarks are illustrated in figure 4.

Table 3. Landmarks Position Coordinates

Landmark	Latitude (deg)	Longitude (deg)	Altitude (m)
1	10^{-2}	50×10^{-1}	10
2	50×10^{-1}	10^{-2}	10

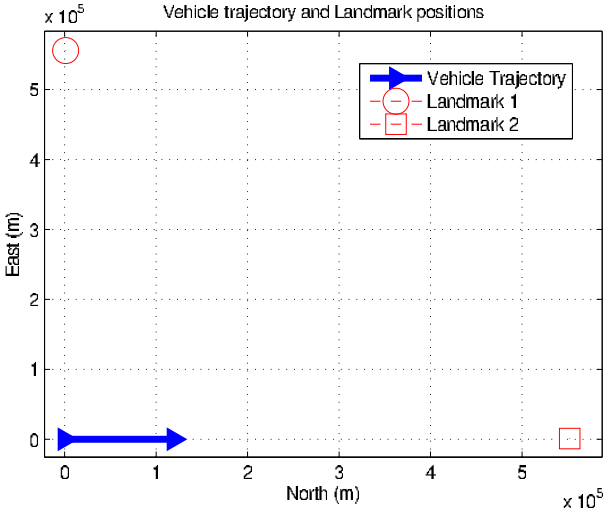


Fig. 4. Vehicle trajectory and landmark positions (top view)

For the sake of simplicity, constant visibility to 4 GNSS satellites is assumed, each with a fixed position with respect to Earth during the simulation time interval. LLA satellite coordinates are given in table 4.

Table 4. GNSS Satellites Position Coordinates

Satellite	Latitude (deg)	Longitude (deg)	Altitude (km)
1	20	−20	20,000
2	40	−20	20,000
3	−30	40	20,000
4	−25	30	20,000

The evaluation of INS/GNSS/SD fusion algorithm performance with single or double-sided observations is based on a Monte Carlo simulation [8] with 50 realizations and two statistical tests. These are the normalized estimation error squared (NEES) and normalized innovation squared (NIS), which are described in [8] and used hereafter with a 5% alarm rate. The number of realizations

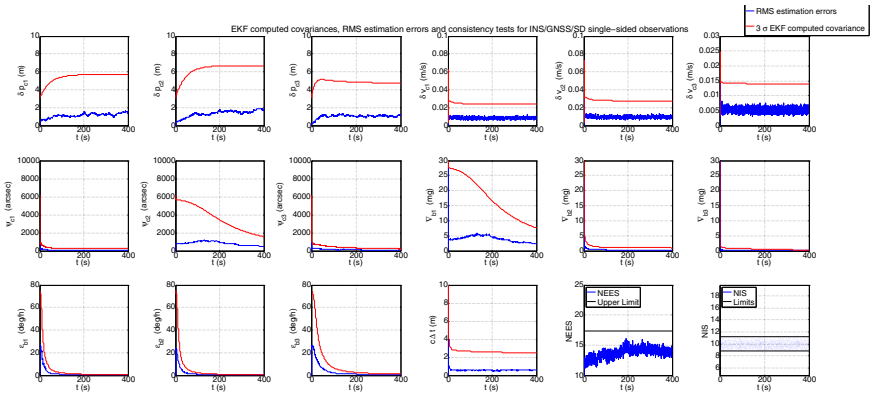


Fig. 5. EKF computed covariances, RMS estimation errors and consistency tests for INS/GNSS/SD single-sided observations

is chosen so that a balance between reliable statistical study and practicable simulation time is obtained.

5.1 Single-Sided Observations

Figure 5 displays INS/GNSS/SD fusion with single-sided observations EKF computed covariances and root mean squared (RMS) estimation errors (denoted by $\boldsymbol{\delta\tilde{p}}$, $\boldsymbol{\delta\tilde{v}}$, $\tilde{\boldsymbol{\psi}}$, $\tilde{\boldsymbol{\nabla}}$, $\tilde{\boldsymbol{\epsilon}}$ and $c\tilde{\Delta t}$) for each component of $\boldsymbol{x}_{\text{EKF}}$. Additionally, NEES and NIS consistency tests are shown with the corresponding alarm limits.

Notice, in sharp contrast with INS/GNSS fusion, the observability of ψ_{c1} and ψ_{c3} without resorting to maneuvers. Notwithstanding, ψ_{c2} is weakly observable due to landmark L_1 location East of the vehicle. It is, indeed, intuitive to expect the inadequacy of sighting devices to yield attitude information about the LOS axis due to the assumed punctual nature of the landmark projection on the image plane. Similarly, ∇_{b1} is weakly observable.

5.2 Double-Sided Observations

Figure 6 displays INS/GNSS/SD fusion with double-sided observations EKF computed covariances and RMS estimation errors for each component of $\boldsymbol{x}_{\text{EKF}}$. Additionally, NEES and NIS consistency tests are shown with the corresponding alarm limits.

The addition of a geometrically favorable landmark, namely L_2, positioned North of the vehicle, enhances ψ_{c2} and ∇_{b1} observability. Thus, the RMS error quickly diminishes and filter tuning is accomplished as far as the NEES and NIS tests can evaluate.

Furthermore, this work suggests as future work further investigation on the impact of number and geometry of landmarks on Kalman filter observability, which can be analytically performed in a fashion similar to [1,2].

As a last remark, the EKF process noise in the position channel must be inflated according to equation 57 to accomplish NEES and NIS statistical consistency. Otherwise, the resulting small EKF computed covariance and the corresponding RMS error become statistically inconsistent, which may render the estimation process unreliable. The tuning of the position channel noise to reach statistical consistency in the NEES test can be addressed by self-tuning algorithms [17,18].

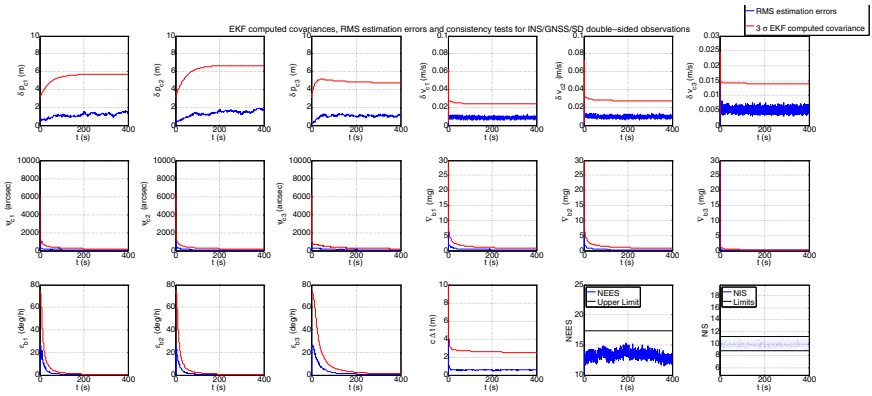


Fig. 6. EKF computed covariances, RMS estimation errors and consistency tests for INS/GNSS/SD double-sided observations

6 Conclusions

An imaging measurement model is formulated for use in a psi-angle error based extended Kalman filter (EKF) that yields the fusion of global navigation satellite observables with vision-aided inertial navigation. At first, the EKF uses datasheet-based nominal tuning and the estimation performance is evaluated by means of Monte Carlo simulation. The resulting performance motivates the use of a process noise inflation scheme to attain statistical consistency. Position and velocity errors, misalignment, accelerometer bias, rate-gyro drift and GNSS clock errors with respect to ground-truth are then effectively estimated and pass the tests for statistical consistency. Ultimately, in-flight INS correction and the calibration of inertial sensors and GNSS receiver clock error are successfully accomplished.

Acknowledgments. This material is based upon research supported by project FINEP/SIA 11382*2, conducted at SIA Lab, Department of Systems and Control, Faculty of Electronics and Computer Engineering, Instituto Tecnológico de Aeronáutica - ITA, São José dos Campos - SP, Brazil.

References

1. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Observability analysis of piece-wise constant systems with application to inertial navigation. In: Proc. 29th IEEE Conf. Decision and Control, pp. 821–826 (1990)
2. Goshen-Meskin, D., Bar-Itzhack, I.Y.: Observability analysis of piece-wise constant systems. ii. application to inertial navigation in-flight alignment. IEEE Trans. Aerospace and Electronic Systems 28(4), 1068–1075 (1992)
3. Bar-Itzhack, I.Y.: Optimal updating of INS using sighting devices. Journal of Guidance and Control 1(5), 305–313 (1978)
4. Weinred, A., Bar-Itzhack, I.Y.: The psi-angle error equation in strapdown inertial navigation systems. IEEE Trans. Aerospace and Electronic Systems AES-14(3), 539–542 (1978)
5. Trawny, N., Mourikis, A.I., Roumeliotis, S.I., Johnson, A.E., Montgomery, J.: Vision-aided inertial navigation for pin-point landing using observations of mapped landmarks. Journal of Field Robotics 24(5), 357–378 (2007)
6. Wang, J., Garratt, M., Lambert, A., Wang, J.J., Han, S., Sinclair, D.: Integration of GPS/INS/vision sensors to navigate unmanned aerial vehicles. In: IAPRS&SIS, Beijing, China, vol. 37, pp. 963–969 (2008)
7. Farrell, J.A., Barth, M.: The Global Positioning System and Inertial Navigation. McGraw-Hill (1998)
8. Bar-Shalom, Y., Li, X.R., Kirubarajan, T.: Estimation with Applications to Tracking and Navigation - Theory Algorithms and Software. Wiley (2001)
9. National Imagery and Mapping Agency, Technical Report TR8350.2 Department of Defense World Geodetic System 1984, Its Definitions and Relationships with Local Geodetic Systems, 3rd edn. (2000)
10. Tenenbaum, R.A.: Fundamentals of Applied Dynamics. Springer (2004)
11. Pitman, G.R. (ed.): Inertial Guidance. Wiley, New York (1962)
12. Maybeck, P.S.: Stochastic Models, Estimation, and Control. Academic Press (1979)
13. Bar-Itzhack, I.Y.: Navigation computation in terrestrial strapdown inertial navigation systems. IEEE Transactions on Aerospace and Electronic Systems (6), 679–689 (1977)
14. Salychev, O.S.: Applied Inertial Navigation: Problems and Solutions. BMSTU Press (2004)
15. Savage, P.G.: Strapdown inertial navigation integration algorithm design, Part 1: Attitude algorithms. Journal of Guidance, Control, and Dynamics 21(1), 19–28 (1998)
16. Savage, P.G.: Strapdown inertial navigation integration algorithm design, Part 2: Velocity and position algorithms. Journal of Guidance, Control, and Dynamics 21(2), 208–221 (1998)
17. Lustosa, L.R., Waldmann, J.: Addressing the statistical consistency of extended Kalman filtering for aided inertial navigation using adaptive techniques. In: Proceedings of 19th Saint Petersburg International Conference on Integrated Navigation Systems. Saint Petersburg, Russia (May 2012)
18. Lustosa, L.R.: Sintonia automática de filtro de Kalman para navegacão inercial auxiliada. M.Sc. thesis. Instituto Tecnológico de Aeronáutica - ITA, Brazil (2012)

Combined Vision – Inertial Navigation with Improved Outlier Robustness

Francesco Di Corato, Mario Innocenti, and Lorenzo Pollini

University of Pisa, Pisa, 56122, Italy
{dicorato,minnoce,lpollini}@dsea.unipi.it

Abstract. This paper describes a loosely coupled approach for the improvement of state estimation in autonomous inertial navigation, using image-based relative motion estimation for augmentation. The augmentation system uses a recently proposed pose estimation technique based on a *Entropy-Like* cost function, which was proven to be robust to the presence of noise and outliers in the visual features. Experimental evidence of its performance is given and compared to a state-of-the-art algorithm. Vision-inertial integrated navigation is achieved using an Indirect Kalman Navigation Filter in the framework of stochastic cloning, and the proposed robust relative pose estimation technique is used to feed a relative position fix to the navigation filter. Simulation and Experimental results are presented and compared with the results obtained via the classical RANSAC – based Direct Linear Transform approach.

1 Introduction

Inertial navigation suffers from drifts due to several factors, in particular inertial sensor errors. As a matter of fact, usually additional sensors like GPS, air data sensors or Doppler speedometers are employed to provide corrections to the navigation system. A viable augmentation alternative is the adoption of a vision system; these were employed in the past for air and land vehicle automation, like car driving [1], obstacle avoidance ([2], [3]) or formation flight ([4], [5], [6], [7]). More recently, mainly due to the increased computational power available, they are receiving more interest in the field of navigation. The use of vision for navigation is often referred to as visual odometry, which core tool is the estimation of the pose of the vision system with respect to the observed scene. Pose estimation is often the concluding step in a sequence of different phases including: detection of significant features in the scene from camera images, and tracking of them between successive frames. The presence of noise and outliers in the acquired data represents the main, in the sense of most challenging, issue in solving the Pose Estimation problem. The presence of outliers depends mainly on inaccurate key points matching and/or tracking between left and right images, in the stereo vision case, and in successive time instants. The outliers rejection problem is often solved via linear/nonlinear minimization techniques (L_2,L_∞, etc) ([8]) or via iterative refinements ([9], [10], [11]), that is via images pre/post-processing

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_16

techniques. Well-known robust approaches in estimating camera pose are RANSAC-type algorithms [9], [10], which have no guarantee of optimality. Almost all outlier rejection schemes proposed in the literature act in a pre/post-processing phase, and most of them perform the pose estimation algorithm by minimizing a squared norm of the estimation error.

The concept of Entropy is not new in the field of estimation; it has already been applied in the last decade in the field of autonomous navigation and robotics, and the most well-known and recent works in such direction can be found in [12], where the concept of alignment via maximization of the Mutual Information is used to perform robust visual servoing and autonomous guidance tasks, in a previously visited scenario. Recently, integrated vision-inertial navigation systems are appearing in the literature; they differ mainly in the adopted coupling approach between vision and inertial measurements. Two large family exists: in the tightly coupled approach [13], [14], [15], each collected key point is added to the navigation filter state, its position is refined over time and cooperates to the estimation phase. The second large family is the loosely coupled approach [16] [17], in which the navigation filter is provided with position fixes computed by the vision system, used in this case as an external aiding sensor like it happens with GPS or altimeters. In [16] the stochastic cloning approach is introduced and used and the relative pose estimation is computed via a classical Least Square minimization. [17] uses a similar approach, but the filter is provided with relative pose measurements, which are obtained via a robust 2-norm minimization, using the Huber cost function [18], in a framework of M-estimation. The work in [19] instead, reverses the point of view and uses the stereo vision system as the main navigation sensor, while the processed IMU measurements are used to feed attitude corrections to an EKF.

In the present paper, we propose a loosely coupled approach, which uses a Stereo Vision system and an Inertial Measurement Unit. The relative pose estimations given by the vision system are computed using an *Entropy-Like* cost function, which is robust by nature with respect to the outliers in the data. The estimated pose is then used to give relative position fixes to the Indirect Kalman Navigation Filter in the framework of the stochastic cloning [20] [16]. The main contribution of the paper is showing that the adoption of the proposed robust pose estimation algorithm, which is robust to a large class of disturbances, provides a net improvement to the navigation accuracy and that there is still room for further improvements that better exploit the peculiar characteristics of the proposed pose estimation algorithm.

The paper is organized as follows: Section 2 introduces the adopted notation and the necessary perturbed inertial navigation background; Section 3 describes the application of the proposed Entropy-based cost function to pose estimation and Section 4 presents a static comparison of performance with a state-of- the-art pose estimation algorithm. Section 5 describes an error-state Extended Kalman Filter for integration of the proposed pose estimation algorithm with inertial navigation; finally Section 6 presents experimental results performed with a ground vehicle.

2 Background on Perturbed Inertial Navigation Dynamics

This paper proposes a vision-inertial integrated navigation system that, as common in precise inertial navigation, makes use of an error-state formulation where navigation errors, rather than navigation states are estimated by the filter[21]. The adopted notation is very common in the Inertial Navigation Literature: define χ as generic motion/sensitivity variable, then $\hat{\chi}$ indicates the estimated value of the true value χ, and $\tilde{\chi}$ indicates the measured value. Thus, the relationship between true values and their measurements is defined as follows:

$$\begin{aligned} \tilde{\chi} &= \chi + \nu_\chi, \\ \chi &= \hat{\chi} - \delta\chi \end{aligned} \tag{1}$$

where $\delta\chi$ is the actual navigation error, and ν_χ is the measurement error. In this work, the measurement errors is modeled as a zero-mean Gaussian process with variance $\mathrm{E}[\nu_\chi^T \nu_\chi]$. With the above notation, it is possible to write a set of perturbed navigation equations for attitude (represented here by the direction cosine matrix R_b^n), velocity in some navigation frame (we used the NED reference frame for filter implementation but any geodetic frame may be used) V^n , and position in ECEF frame r^e as:

$$\begin{cases} \hat{R}_b^n = (I - \delta\gamma \wedge) R_b^n \\ \hat{V}^n = V^n + \delta V^n \\ \hat{r}^e = r^e + \delta r^e \\ \tilde{\omega}_{ib}^b = \omega_{ib}^b + \delta\omega_{ib,b}^b + \nu_\omega \\ \tilde{f}_{ib}^b = f_{ib}^b + \delta f_{ib,b}^b + \nu_f \end{cases} \tag{2}$$

where $\delta\gamma \wedge$ denotes the skew symmetric matrix whose elements are the components of the errors vector $\delta\gamma$, which are functions of the attitude error [21]. Moreover, $\delta\omega_{ib,b}^b$ and $\delta f_{ib,b}^b$ are the bias terms in the measurements of gyroscopes and accelerometers, while ν_ω and ν_f are gyroscope and accelerometer noises, represented here as zero-mean Gaussian processes with variances $E[\nu_\omega \nu_\omega^T] = Q_\omega$ and $E[\nu_f \nu_f^T] = Q_f$. Finally δV^n and δr^e are velocity and (global) position errors respectively.

Given the definition above of the navigation error variables, the continuous time e error dynamics of the navigation equations resolved in the navigation frame [21] can be locally approximated by a compact Linear Parameter Varying (LPV) system, as in Eq. (3) :

$$\delta\dot{x}(t) = F(t)\delta x(t) + G(t)u_c(t) \tag{3}$$

The state vector $\delta x(t)$ and the input vector $u_c(t)$ are defined respectively as (we dropped the function of time for compactness of notation):

$$\delta x(t) = [\delta\gamma^{nT} \quad \delta V^{nT} \quad \delta r^{eT} \quad \delta\omega_{ib,b}^{b\,T} \quad \delta f_{ib,b}^{b\,T}]^T \tag{4}$$

$$u_c(t) = [\nu_\omega^T \quad \nu_f^T \quad \nu_{r\omega}^T \quad \nu_{rf}^T]^T \tag{5}$$

The system matrices $F(t)$ and $G(t)$ in Equation (3) come from linearization of the error dynamics, thus they change with the selected navigation frame, and locally relate the evolution of the state i to the current estimation of the state j. The reader interested in the derivation of the above equations can find all the details in [21]. The IMU biases dynamics in Eq. 3 were modeled as Brownian motions, with trivial dynamics:

$$\begin{cases} \delta\dot{\omega}_{ib,b}^{b} = \nu_{r\omega} \\ \delta\dot{f}_{ib,b}^{b} = \nu_{rf} \end{cases} \tag{6}$$

where $\mathrm{E}[\nu_{r\omega}\nu_{r\omega}^{T}] = Q_{r\omega}$ and $\mathrm{E}[\nu_{rf}\nu_{rf}^{T}] = Q_{rf}$. The covariance matrix of the Gaussian process noise $u_c(t)$, considering the sensors' noises uncorrelated and having the same noise characteristics, is given by:

$$\mathrm{E}[u_c(t)u_c(t)^{T}] = \begin{bmatrix} Q_f & 0 & 0 & 0 \\ 0 & Q_\omega & 0 & 0 \\ 0 & 0 & Q_{r\omega} & 0 \\ 0 & 0 & 0 & Q_{rf} \end{bmatrix} \delta(t-\tau) \tag{7}$$

In order to implement the filter dynamics in real-time, it is necessary to discretize the continuous time dynamics; in the remainder we will consider a time-discretized version of the above dynamics using the Euler integration method, with sample time ΔT. The final discrete-time form of the LPV perturbed system of Equation (3) can then be written as:

$$\delta x_{k+1} = \Psi_k \delta x_k + \Gamma_k u_{c,k} \tag{8}$$

where:

$$\begin{aligned} \Psi_k &= (I - F(t_k)\Delta T) \\ \Gamma_k &= G(t_k)\Delta T \end{aligned} \tag{9}$$

3 Least-Entropy Like Pose Estimation

Loosely coupled vision-aided inertial navigation with relative measurements requires the estimation of the camera motion in between successive frames. This section presents first a general framework for pose estimation, then cast this problem into the framework of Least-Entropy Like (LEL) estimation[22][23], finally presents an analysis of performance using static images.

3.1 Stereo Vision and Pose Estimation

In a stereo vision system, each camera acquires an image, relevant 2-dimensional features (points in the image plane) $\{p_{i,k}\}$ are automatically extracted from the images (for the purpose of this work we used the SIFT algorithm), identical features, that is image points belonging to the same object in the observed scene, are searched for in

the right and left images, and finally a cloud of N 3D keypoints $\{P_{i,k}\}$ is obtained by triangulation of the two corresponding sets (one for the left and one for the right images) of N 2D features $\{p_{i,k}\}$. Several techniques exist for selection and tracking of image features[5][4]; the feature selection and tracking approach used in the later simulations use stereo vision and the Scale Invariant Feature Transform (SIFT) algorithm[24][3], that easily allows both to detect, and to match features for successive triangulation. The stereo matching of features between left and right images is performed by comparing the squared distance between the SIFT descriptors of each feature in the two images, and selecting the couple with the lowest distance. Only those features that are both in the left and right images are considered valid for triangulation and tracking. With the same distance-based approach it is possible to track the features that are present in the current and past images; this makes the selection of 3D keypoints $\mathrm{P_{i,k}}$ and $P_{i,k+h}$ possible. Figure 1 shows a sample of two images with matched features (red circles), unmatched features (blue circles) and green lines representing left-right matches.

Tracking of 2D features in two successive time instants t_k and t_{k+h} produces two clouds of 3D keypoints $P_{i,k}$ and $P_{i,k+h}$ that are related by a rigid motion relationship. This relationship represents, essentially, the camera motion, in terms of translation T_k^{k+h} and rotation R_k^{k+h}, between times t_k and t_{k+h}. Thus the following relationship holds:

$$P_{i,k+h} = R_k^{k+h} P_{i,k} + T_k^{k+h} = g_k^{k+h} P_{i,k} \tag{10}$$

where $g_k^{k+h} = \left\{R_k^{k+h},\ T_k^{k+h}\right\} \in SE(3)$ is the transformation mapping the pose of the camera at the time t_k in the pose of the camera at the time t_{k+h}. The notation $g_k^{k+h} P_{i,k}$ is not actually a vector or matrix multiplication but denotes the *application* of the translation and rotation transformations $R_k^{k+h},\ T_k^{k+h}$ to the point $P_{i,k}$, as described in Eq. (10). Given any 3 dimensional parameterization of the rotation matrix, the transformation matrix in Eq. (10) can be written as: $g_k^{k+h}(\theta_p), \theta_p \in \Theta_p \subset \mathbb{R}^6$, being Θ_p the set of all possible motion parameters (angular displacements and translations). The Pose Estimation problem then becomes the estimation of the unknown motion parameters vector θ_p, given two clouds of N features at the time t_k and t_{k+h}. The solution of the problem can be found by using a minimization approach (either linear or non-linear) over the estimation residuals $E_{i,k+h}$:

$$E_{i,k+h} = P_{i,k+h} - g_k^{k+h}(\theta_p) P_{i,k} \tag{11}$$

that is:

$$\hat{\theta}_p = arg\,min_{\theta_p} \sum_{i=1}^{N} \mathcal{L}\left\{P_{i,k+h} - g_k^{k+h}(\theta_p) P_{i,k}\right\} \tag{12}$$

where $\mathcal{L}\{\cdot\}$ is a suitable cost function built upon the pose estimation residual; common choices for $\mathcal{L}\{\cdot\}$ are the 2-norm or the infinity-norm. Due to triangulation and calibration errors a number $N \geq 4$ of non-aligned points, tracked along the camera motion, are necessary for the problem to have a solution.

3.2 Robust Camera Pose Estimation Using LEL

A very relevant and desirable behavior for any feature detector and tracker is its ability to recognize features in different images even if they were taken from viewpoints distant one from the other (this means the capability to track features during camera motion for long time). When the camera moves and rotates, the same objects of the pictured scene produce different images on the camera plane: deformations and warping happens due to camera motion, change of the point of view, and perspective projection. Thus a good feature detector and tracker must be able to recognize exactly the same warped image regions. In order to achieve this property, covariant feature detectors, such as SIFT, are designed to mod-out the effects of transformations belonging to some group [25]. Such characteristic induces a certain amount of loss of information in the detected features, thus some ambiguities could raise. Figure 1 shows one example where this information loss leads to a mismatch. As a result, the whole set of features collected during the acquisition, matching and tracking phases may be affected by a certain amount of outliers. In the following, a technique which is able to give a measure of the degree of dispersion of the data will be used to design a robust pose estimator.

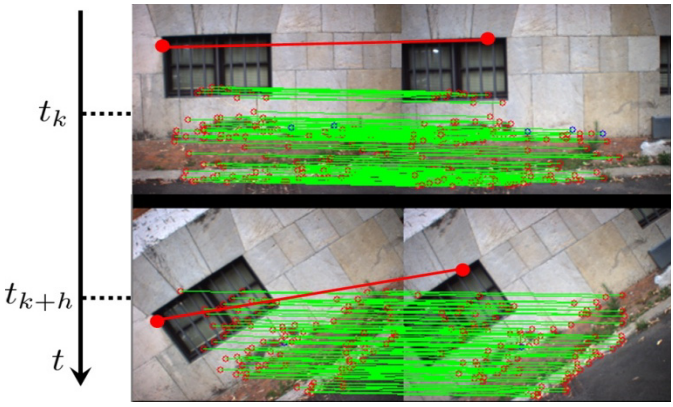


Fig. 1. Example of left-right matched features (red circles connected by the green lines) and an example of a possible matching ambiguity that may happen with the use of co-variant feature detectors (e.g. with SIFT). The matching ambiguity contaminates the data used for pose estimation with outliers (the large red dots).

A robust nonlinear alternative to Least-Square estimation was recently proposed [22]. The aim of such estimator is to give a representation of the dispersion of the residuals; such function is built on the concept of Gibbs' entropy ([26]): this is the reason why such estimator was named *Least-Entropy Like* (*LEL*) estimator. Given a reference model, which allows to match given inputs with measured outputs, minimizing the *LEL* metric of the residuals means to drive the solution toward such directions in which such residuals are in one *configuration* where not all the points have the same probability to belong to the chosen model. In [22], [27] and [23] it is shown that this *selectivity* turns out to be very important in such cases in which data are (heavily) corrupted by noise and outliers. All the implementation considerations

regarding the parameterization and minimization of the *Entropy-Like* cost function are described in detail in the references cited above.

The robust solution $\hat{\theta}_{LEL}$ to the problem of the stereo camera pose estimation between two consecutive acquisitions can be solved by minimizing the *Normalized Entropy-Like* function:

$$\hat{\theta}_{LEL} = \underset{\theta}{argmin}\left(-\frac{1}{log\, N}\sum_{i=1}^{N}\pi_i\, log\, \pi_i\right) \tag{13}$$

where:

$$\pi_i = \frac{\psi(r_i)}{\sum_{j=1}^{N}\psi(r_j)} \tag{14}$$

$$r_i = p_{i,k+h} - K_c\, proj\left\{g_k^{k+h}(\theta_p)P_{i,k}\right\} \tag{15}$$

Notice that this approach uses the re-projected pose estimation residuals r_i in 2D, instead of the pose estimation residual in 3D as in the most general form of equation (11). The re-projection error r_i involves image coordinates only that are invariant to changes in depth [28]; this leads to a better estimation accuracy. The adopted pin-hole camera model is represented, as common in computer vision, by the calibration matrix K_c and the perspective projection operator $proj\{\cdot\}$: given a generic 3D point P with coordinates P_x, P_y, P_z, the perspective projection operator is defined as:

$$P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \Rightarrow proj\{P\} = \begin{bmatrix} P_x/P_z \\ P_y/P_z \\ 1 \end{bmatrix} \tag{16}$$

In addition, this formulation of the *Entropy-Like* function employs the Huber-like [18] function $\psi(\cdot)$ to reduce the risk of incurring into a local minimum during solution of Eq. (13).

$$\psi(a) = \begin{cases} \|a\|^2, & if\ \|a\| \le d \\ d^2, & otherwise \end{cases} \tag{17}$$

The employment of the Huber-like function allows avoiding bad conditionings of the *Entropy-Like* cost function by limiting the upper bound of the denominator in (14), and thus by avoiding the uncontrolled growth of the sum of the residuals norm due to numerical sensitivities.

Numerical solution of the optimization problem in Eq. (13) can be done in several ways. The simulations and experiments presented in this paper adopted the Levenberg-Marquardt as in [23][28]. As explained in [22] and [27], the *Entropy-Like* penalty function is nonlinear and multiple local minima may exist. Thus, the minimization must be computed with particular attention to the initial conditions. The scope of this work is such that we expect to have an acceptable local estimate of the motion given by inertial mechanization alone performed over a short period of time (between two

successive frames). Therefore, it is possible to initialize the nonlinear estimation with the parameter $\hat{\theta}_{p,0}$ that can be extracted by the best available estimate of the relative transformation:

$$\hat{g}_k^{k+h} = (\hat{g}_{k+h}^-)^{-1}\hat{g}_k^+ \rightarrow \hat{\theta}_{p,0} \tag{18}$$

where $\hat{g}_k^+$ is the best (corrected by the filter in the past) estimate of the navigation at time t_k, $\hat{g}_{k+h}^-$ is the navigation prediction at the current time t_{k+h}. The arrow symbol in Eq. (18) means that the value of $\theta_{p,0}$ is extracted by the transformation $\hat{g}_k^{k+h}$.

3.3 Experimental Results for Pose Estimation Only

LEL has already been shown to perform better than ICP [27], and a Monte Carlo Analysis have shown that it can outperform the RANSAC-based Direct Linear Transform (DLT) [29], with nonlinear refinement via Bundle Adjustment [30]. The main results are summarized here for completeness. Tests were performed both with simulated features and various level of image noise, and with real imagery; experiments were performed outdoor with a hand-held fire wire stereo camera system at a resolution of 516×388 pixels (a good trade-off between speed of image processing and accuracy of features selection and matching). An industrial 1.6 GHz PC with 1 GB RAM was used to collect the test videos; then, the video frames were processed off – line, together with the estimation algorithm.

Figure 2 show a sample image pair from an outdoor experiment; the green dots are the matched SIFT features, the red circles are the re-projected features by using the LEL pose estimation result. Figure 3 shows the sorted 2-norm of the re-projection residuals:

$$\left\|e_{r,i}\right\|^2 = \left\|p_{i,2} - K_c \,\mathrm{proj}\left\{g_1^2\left(\hat{\theta}_p^{LEL,DLT}\right)P_{i,1}\right\}\right\|^2 \tag{19}$$

computed using the motion parameters $\hat{\theta}_p$ estimated by the two methods, LEL and robust DLT. The camera calibration matrix $\mathrm{K_c}$ was determined experimentally, $\mathrm{P_{i,1}}$ are the 3D keypoints triangulated in the first position of the camera (at time $\mathrm{t_1}$), and $\mathrm{p_{i,2}}$ are the image-space coordinates of the corresponding features on the image plane of the image acquired in the final position of the camera (at time $\mathrm{t_2}$). The measurement unit of points $\mathrm{p_{i,2}}$ is pixels. The features re-projections (red circle in Fig. 3) were computed as:

$$\hat{p}_{i,2} = K_c \,\mathrm{proj}\left\{g_1^2\left(\hat{\theta}_p^{LEL}\right)P_{i,1}\right\} \tag{20}$$

In addition, Figure 3 highlights the mismatching between the measured and estimated projection, once the optimal transformation is applied to an outlier (marked with two red 'x' connected via the red line). It should be noticed that the robust DLT algorithm tries to exclude the outliers and the noisiest points from the dataset before solving the pose estimation problem, while LEL performs both pose estimation and outlier rejection in one step. Furthermore, it can be stated that LEL (which is run on the whole dataset) is able to perform as good as a 2-norm approach like DLT (that needs the

dataset to be purged by outliers and ambiguous points) [22]. Although no analytical guarantee is available yet, the LEL algorithm performs in general as good as the robust DLT algorithm with nonlinear refinement (tuned with our best efforts), which was used as benchmark. In some particular experiments the accuracy of the methods cannot be stated in an absolute fashion, since no ground truth was available in order to compare algorithms.

Finally, a Monte Carlo analysis was performed to assess the robustness of the proposed algorithm to outliers; LEL provided less re-projection error then DLT for all the tested percentage of presence of outliers [28].

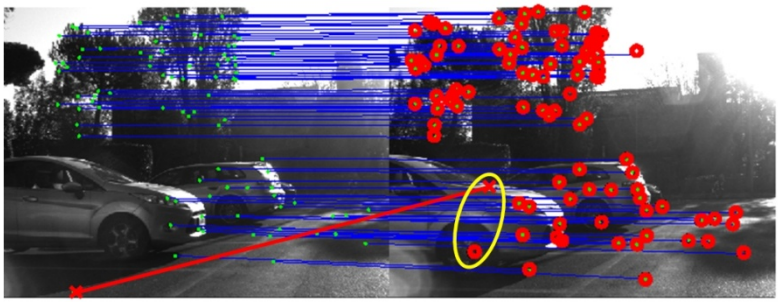


Fig. 2. Outdoor experiment. Image pair with points correspondences and estimation results. The green dots are the matched SIFT features. The red circles are the re-projected features by using the LEL pose estimation result.

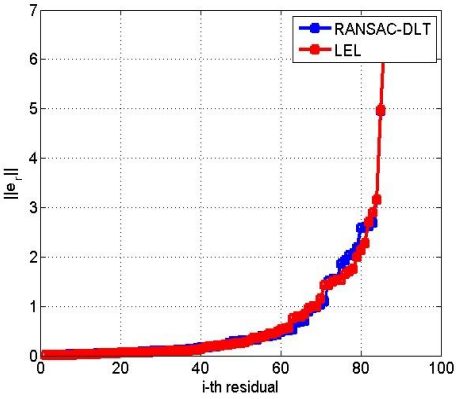


Fig. 3. Outdoor experiment. Sorted re-projection errors.

4 Navigation and Kalman Filtering with Relative Pose Measurements

Usually all aiding sensors produce absolute measures (with respect to a known and fixed reference) of the estimated variables (e.g. GPS measures $\tilde{r}_e$ and $\tilde{V}_e$) while, in the case of visual odometry, the motion measurements are relative only (i.e. only the relative displacement between two successive images is measured). This section

summarizes the equations used for the fusion of the relative motion measurements, given from the pose estimation algorithm, and the inertial data.

4.1 Definition of the Relative Pose Pseudo-Measurement Error

The camera pose at time $t_k$can be related, with respect to initial position (at time t_0), to the inertial mechanization states (position and attitude) as:

$$g_k^0 = \{R_k^0,\ T_k^0\} \tag{21}$$

where $R_k^0 = R_{b,k}^n = R_b^n(t_k)$ and T_k^0 represents the position of the origin of the Navigation/Body frame at time t_k seen from the initial Navigation frame (at time t_0). For the purposes of this paper, we assumed that the relative displacement (latitude and longitude) between successive images is small enough so that the navigation frame (NED) can be considered orientation-invariant (with respect to the ECEF frame); thus a simple planar projection can be used (approximation of flat surface), to approximate motion in the neighborhood of starting point. Thus the camera position can be obtained with Eq. (22):

$$T_k^0 = \xi(r^e(t_k)) = \begin{bmatrix} P_n(\phi_k - \phi_0) \\ P_n \cos\phi_k\,(\lambda_k - \lambda_0) \\ h_k \end{bmatrix} \tag{22}$$

where $r^e(0) = [\phi_0 \quad \lambda_0 \quad 0]$ represents the vector of coordinates in the ECEF frame (latitude, longitude, altitude) corresponding to the initial position of the vehicle, when the navigation task began its execution (at time t_0). P_n is the radius of curvature normal to the ellipsoid surface at the point of tangency at the given latitude ϕ [21].

Given two pairs of successive images at time t_k and t_{k+h}, the relative motion, that must be computed by the vision system, g_k^{k+h} is related to the absolute poses at time t_k and t_{k+h} by:

$$g_k^{k+h} = \left\{R_k^{k+h}, T_k^{k+h}\right\} = \{R_0^{k+h}R_k^0\,, R_0^{k+h}(T_k^0 - T_{k+h}^0)\} \tag{23}$$

It is now necessary to define a filter output that can be used to construct a measurement residual with the vision system output. Thus, first we construct a navigation position error estimate using the planar projection operator $\xi(\cdot)$ and an attitude error:

$$\delta y_k = H_k \delta x_k \triangleq \begin{bmatrix} \delta T_k^0 \\ \delta \gamma_k^0 \end{bmatrix} \tag{24}$$

where:

$$H_k = \begin{bmatrix} 0 & 0 & R_{n,k}^b \left.\frac{\partial T_k^0}{\partial r_k^e}\right|_{\hat{r}_k^e} & 0 & 0 \\ \left.\frac{\partial R_{n,k}^b}{\partial \varphi_k}\right|_{\hat{\varphi}_k} & 0 & 0 & 0 & \left.\frac{\partial R_{n,k}^b}{\partial \omega_{ib,b,k}^b}\right|_{\hat{\omega}_{ib,b,k}^b, \hat{\varphi}_k} \end{bmatrix} \tag{25}$$

It is worth to highlight that $\left.\frac{\partial T_k^0}{\partial r_k^e}\right|_{\hat{r}_k^e}$ is the Jacobian of the function $\xi(\cdot)$ with respect to the estimated position in ECEF frame around $\hat{\mathrm{r}}_{\mathrm{k}}^{\mathrm{e}}$. Then we can estimate the relative navigation error $\Delta\delta y_{k+h}$ between time t_{k+h} and t_k as:

$$\Delta\delta y_{k+h} = H_{k+h}\delta x_{k+h} - H_k \delta x_k \tag{26}$$

Note that H_{k+h} takes the same form of H_{k} except that it is computed with respect to the state at time $k+h$. $\Delta\delta\mathrm{y}_{\mathrm{k+h}}$ is the estimate of the navigation error of the value $\hat{\mathrm{T}}_{\mathrm{k}}^{\mathrm{k+h}}$ computed by the inertial mechanization.

Since the vision system actually measures $\tilde{g}_k^{k+h} = \left\{\tilde{R}_k^{k+h}, \tilde{T}_k^{k+h}\right\}$, it is possible to compute a pseudo-measure of the relative pose error from the measured relative pose $\tilde{\mathrm{g}}_{\mathrm{k}}^{\mathrm{k+h}}$ and its estimation $\hat{\mathrm{g}}_{\mathrm{k}}^{\mathrm{k+h}}$ reconstructed from the navigation equations, as a function of the filter state. In our case, such error can be written as:

$$\Delta\delta y_{k+h}^* = \begin{bmatrix} \tilde{T}_k^{k+h} - \hat{T}_k^{k+h} \\ \tilde{\varphi}_k^{k+h} - \hat{\varphi}_k^{k+h} \end{bmatrix} = \begin{bmatrix} \Delta\delta y_{T,k+h}^* \\ \Delta\delta y_{R,k+h}^* \end{bmatrix} \tag{27}$$

We aim at writing the pseudo-measure of relative pose error $\Delta\delta y_{k+h}^*$ as a function of the filter state.

The estimated relative translation is:

$$\hat{T}_k^{k+h} = \hat{R}_0^{k+h}\left(\hat{T}_k^0 - \hat{T}_{k+h}^0\right) \tag{28}$$

while, the measured relative translation is, by definition, equal to the actual data corrupted by noise ν_k:

$$\begin{aligned} \tilde{T}_k^{k+h} &= T_k^{k+h} + \nu_{T,k+h} \\ &= R_0^{k+h}(T_k^0 - T_{k+h}^0) + \nu_{T,k+h} \\ &= R_0^{k+h}\left(\hat{T}_k^0 - \delta T_k^0 - \hat{T}_{k+h}^0 - \delta T_{k+h}^0\right) + \nu_{T,k+h} \\ &= R_0^{k+h}\left(\hat{T}_k^0 - \hat{T}_{k+h}^0\right) - R_0^{k+h}(\delta T_k^0 - \delta T_{k+h}^0) + \nu_{T,k+h} \end{aligned} \tag{29}$$

The pseudo-measure of the relative translation error can be rewritten as a function of the states (current and of the past) of the indirect Kalman Filter only:

$$\begin{aligned} \Delta\delta y_{T,k+h}^* &= \tilde{T}_k^{k+h} - \hat{T}_k^{k+h} \\ &= R_0^{k+h}\left(\hat{T}_k^0 - \hat{T}_{k+h}^0\right) - R_0^{k+h}\left(\delta T_{k+h}^0 - \delta T_k^0\right) + \nu_{T,k+h} \\ &\quad - \hat{R}_0^{k+h}\left(\hat{T}_k^0 - \hat{T}_{k+h}^0\right) \\ &\approx -\hat{R}_0^{k+h}\left(\hat{T}_k^0 - \hat{T}_{k+h}^0\right) \wedge \delta\gamma_{k+h} - \hat{R}_0^{k+h}\left(\delta T_{k+h}^0 - \delta T_k^0\right) \\ &\quad + \nu_{T,k+h} \end{aligned} \tag{30}$$

where $\delta\gamma_{k+h}$ is the attitude error at the time step $k+h$. The previous equation was obtained by neglecting the cross products between error terms and by using the fact that the attitude error $\delta\gamma_{k+h}$ is defined for the matrix R_0^{k+h}, via Equation (2). Thus, by taking the transpose, we have:

$$R_0^{k+h} = (R_{k+h}^0)^T = \left(\hat{R}_{k+h}^0\right)^T (I - \delta\gamma_{k+h}\wedge)^T = \hat{R}_0^{k+h}(I + \delta\gamma_{k+h}\wedge) \tag{31}$$

It is not straightforward to obtain in the same manner, i.e. algebraically, the pseudo-measure of the relative rotation error $\Delta\delta y_{R,k+h}^*$ as a function of the filter state. It is convenient to derive the equation relative to $\Delta\delta y_{R,k+h}^*$ via partial derivatives instead, that is:

$$\Delta\delta y_{R,k+h}^* \approx \left.\frac{\partial R_k^{k+h}}{\partial \gamma_k}\right|_{\hat{R}_k^0, \hat{R}_{k+h}^0} \delta\gamma_k + \left.\frac{\partial R_k^{k+h}}{\partial \gamma_{k+h}}\right|_{\hat{R}_k^0, \hat{R}_{k+h}^0} \delta\gamma_{k+h} + \nu_{R,k+h} \tag{32}$$

The relative pose measurement error do depend on the motion variables corresponding to the current time (via $\hat{r}_{k+h}^e$ and $\hat{R}_0^{k+h}$) and to some steps in the past (via $\hat{r}_k^e$ and $\hat{R}_0^k$). Thus it is necessary to augment the filter state with a memory of the past; this allow to keep track of the cross covariance of estimated navigation between the two time instants [20][23].

The state of the error navigation filter is augmented with one exact copy $\delta\breve{x}_k$ of itself when a reference frame is acquired. Suppose a new reference frame arrives at time t_k, the state of the Kalman Filter will be set to:

$$\delta\bar{x}_k = \begin{bmatrix} \delta\breve{x}_k \\ \delta x_k \end{bmatrix} \tag{33}$$

and the state covariance matrix is set to:

$$\bar{P}_k = \begin{bmatrix} P_k & P_k \\ P_k & P_k \end{bmatrix} \tag{34}$$

being $P_k = E\{\delta x_k \delta x_k^T\}$. The state copy $\delta\breve{x}_k$ is initialized to δx_k and is kept constant during the filter propagation, whereas the state vector δx_k is propagated according to error dynamics.

4.2 Kalman Filter Prediction Step

At each time step inertial mechanization is performed to obtain a new estimate of the vehicle state (position, velocity and accelerometer biases):

$$\hat{x}_{k+1}^- = f(\hat{x}_k, u_k) \tag{35}$$

where $f(\cdot)$ represents the discretized version of the standard INS mechanization [21], which maps corrected navigation states on the states at the next time step; variable

$\hat{x}_{k+1}^{-}$ is the estimation of the vehicle position and velocity (at the time t_{k+1}), before the corrections, if any, produced by Kalman Filter are applied (i.e. the a priori estimate).

According to the above discussion, the indirect Kalman Filter prediction step is performed using:

$$\delta\bar{x}_{k+1} = \begin{bmatrix}\delta\bar{x}_{k+1}\\ \delta x_{k+1}^{-}\end{bmatrix} = \begin{bmatrix} I & 0\\ 0 & \Psi_k\end{bmatrix}\begin{bmatrix}\delta\bar{x}_k\\ \delta x_k\end{bmatrix} + \begin{bmatrix}0\\ \Gamma_k\end{bmatrix} w_k = \bar{\Psi}_k \delta\bar{x}_k + \bar{\Gamma}_k w_k \tag{36}$$

The propagation equation for the covariance matrix is, like for standard Kalman filtering:

$$\bar{P}_{k+1} = \bar{\Psi}_k \bar{P}_k \bar{\Psi}_k^T + \bar{\Gamma}_k Q \bar{\Gamma}_k^T \tag{37}$$

where Q is the process noise covariance matrix. After h steps (the time span needed to obtain the second image) the covariance matrix becomes:

$$\bar{P}_{k+h} = \begin{bmatrix} P_k & P_k\left(\prod_{i=1}^{h}\Psi_{k+i}\right)^T\\ P_k\prod_{i=1}^{h}\Psi_{k+i} & P_{k+h}\end{bmatrix} \tag{38}$$

Note the off-diagonal blocks that represent the cross-correlation between the navigation errors at the time $\mathrm{t_k}$ and $\mathrm{t_{k+h}}$.

4.3 Kalman Filter Correction Step

When the vision system provides a new relative pose measurement, the update step is performed, as follows:

$$\begin{aligned}
S_{k+h} &= [H_k \quad H_{k+h}]\bar{P}_{k+h}^{-}\begin{bmatrix}H_k^T\\ H_{k+h}^T\end{bmatrix} + R_k\\
K_{k+h} &= \bar{P}_{k+h}^{-}\begin{bmatrix}H_k^T\\ H_{k+h}^T\end{bmatrix} S_{k+h}^{-1}\\
\Delta\delta y_{k+h}^{*} &= \bar{T}_k^{k+h} - \hat{T}_k^{k+h}\\
\delta x_{k+h}^{+} &= \delta x_{k+h}^{-} + K_{k+h}|_{\delta x_{k+h}}(\Delta\delta y_{k+h}^{*} - \Delta\delta y_{k+h})\\
\hat{x}_{k+h}^{+} &= \hat{x}_{k+h}^{-} + \delta x_{k+h}^{+}\\
\bar{P}_{k+h}^{+} &= \bar{P}_{k+h}^{-} - K_{k+h}[H_k \quad H_{k+h}]\bar{P}_{k+h}^{-}
\end{aligned} \tag{39}$$

where R_k is the measurement noise covariance matrix. Variable $\hat{x}_{k+h}^{+}$ is the estimation of the vehicle position and velocity (at the time t_{k+h}), given the corrections produced by Kalman Filter (i.e. the a posteriori estimate).

5 Experimental Results

Simulation results with a comparison of the proposed navigation filter with the RANSAC-based Direct Linear Transform, with nonlinear refinement via Bundle

Adjustment demonstrated already the viability of the LEL approach [28] where an inertial grade gyroscope unit was assumed available, and only vision-estimated translational motion was used to correct filter state. The simulations were performed by generating sample (noisy) accelerations and clean angular velocities. The result was a sample camera trajectory in 6DOF. The accelerations and angular velocities movements were generated by using a VTOL quad rotor aircraft simulator, and, in order to emulate the presence of outliers in the data, random numbers were added to the image-space 2D coordinates. Both algorithms produced small errors (few centimeters) but the DLT visual odometry solution resulted to be noisier.

This section presents a sample experiment performed outdoor in the Univ. of Pisa Faculty of Engineering parking lot using a wheeled ground vehicle. The cameras and hardware used was the same of the static experiments. A snapshot of about 80 seconds, where recognition of the actual travelled path was easier, was extracted from a longer recording. The filter state was initially coarse aligned with gravity to estimate initial roll and pitch angles of the camera-IMU system; then motion began and the vehicle was driven along a straight path, followed by a 180 degrees turn, and a successive almost straight path that brought the vehicle back to its initial position.

Figure 4 shows the time histories of the estimated position, velocity and attitude angles during the motion of the vehicle. It appears clearly that the navigation filter produces smooth estimation with minimal drift. The expected drift in pure inertial navigation (i.e. without any aiding), according to the characteristics of the low-cost inertial sensor suite used, would be of several tens of meters in the same time range.
Figure 5 shows the estimated vehicle trajectory in the local geodetic frame. Three trajectories are shown: the output of the integrated vision-inertial system, the result of running the visual odometry algorithm (integration of relative position fixes only, and no inertial data) on the pose estimation results provided by DLT and LEL. By knowing the actual path followed by the vehicle, it appears clearly that the best estimate in terms of navigation accuracy is given by the integrated visual-inertial navigation: the path starts and returns to the same point. The result of visual odometry for both DLT and LEL show instead a relevant drift in the position estimation. Nevertheless the integrated navigation filter succeeds in filtering out these drifts.

Figure 6 shows a comparison of the estimates of relative camera motion performed by: DLT algorithm using visual features only, LEL algorithm using visual features only, inertial mechanization. The latter represents the translation and rotation parameters that are actually estimated by the filter just before a new image is acquired, and that are used to initialize the solver for the LEL minimization problem. The figure proposes selections of the time range where the differences between the three are large. It appears that LEL and DLT pose estimation solutions are often very near to each other, even if LEL is often less noisy then DLT. In addition, the smoothing effect performed by the Kalman filter on the noisy visual measurements is noticeable throughout the entire time range of the experiment.

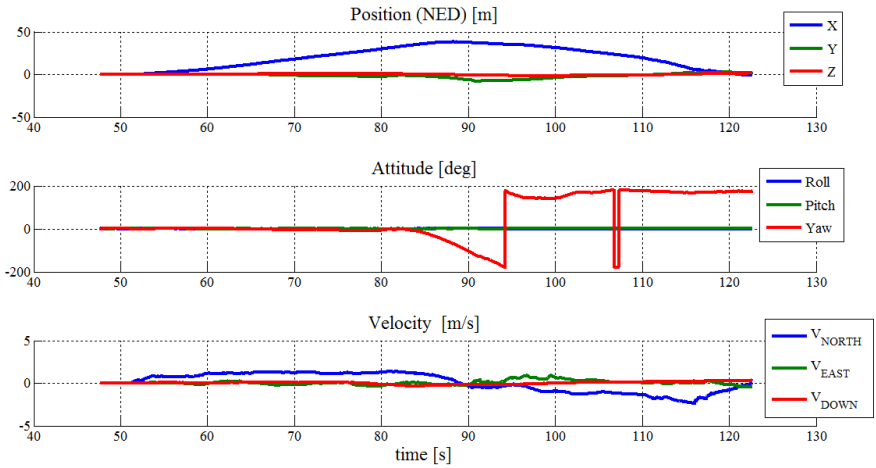


Fig. 4. Time histories of vehicle position (meters from a geodetic fixed reference frame), attitude and velocity in NED

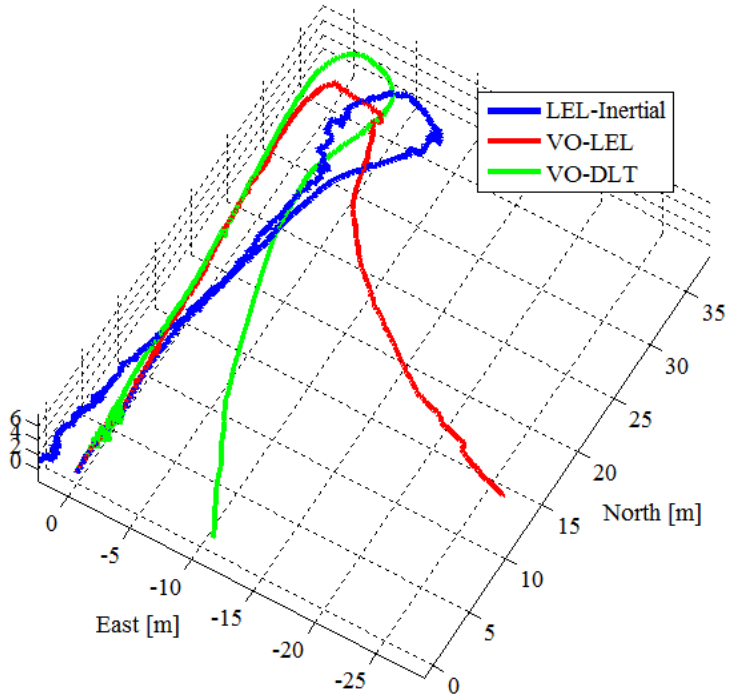


Fig. 5. Trajectories in the local geodetic frame. Comparison of the output of the vision-inertial navigation filter, with visual odometry (VO) performed integrating only the LEL and DLT relative position fixes.

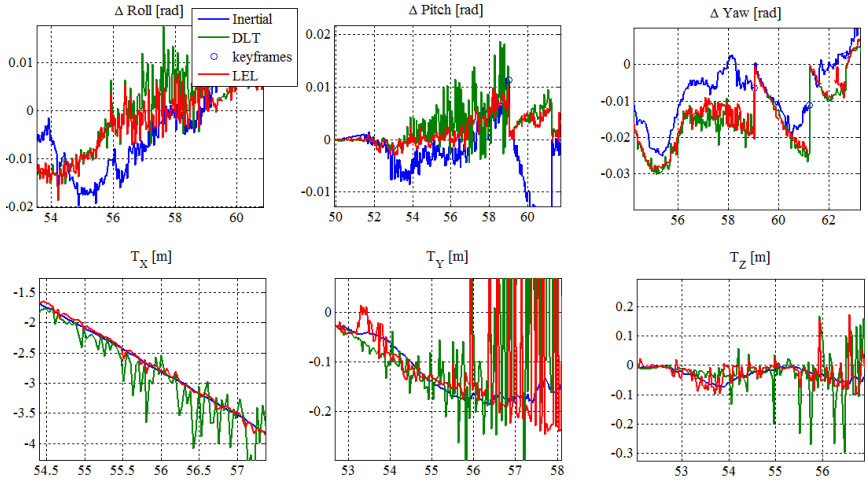


Fig. 6. Comparison of Inertial, DLT, and LEL estimates of relative camera motion

6 Conclusions

A robust loose-coupling approach to vision-augmented inertial navigation, which makes use of a novel cost function, the Entropy of relative squared residuals, was proposed. The LEL algorithm was shown with simulations and experimental tests to be robust to the presence of noise and outliers in the visual features. An error-state Kalman filter was designed and experimental results were presented; these show that using the LEL approach for pose estimation, although may produce noisy estimates, allows to reduce the navigation drift, with respect to a robust technique based on 2-norm minimization plus nonlinear refinement via Bundle Adjustment.

Acknowledgments. Support for the work of the first author was provided by Northrop Grumman Italia Spa.

References

1. Broggi, A.: Robust Real-Time Lane and Road Detection in Critical Shadow Conditions. In: Proceedings of IEEE International Symposium on Computer Vision, Coral Gables, Florida (1995)
2. Watanabe, Y., Calise, A.J., Johnson, E.N.: Vision-Based Obstacle Avoidance for UAVs. In: AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, South Carolina (2007)
3. Pollini, L., Greco, F., Mati, R., Innocenti, M., Tortelli, A.: Stereo Vision Obstacle Detection based on Scale Invariant Feature Transform Algorithm. In: AIAA Guidance Navigation and Control Conference, Hilton Head, South Carolina (2007)

4. Innocenti, M., Mati, R., Pollini, L.: Vision Algorithms for Formation Flight and Aerial Refueling with Optimal Marker Labeling. In: AIAA Modeling and Simulation Technologies Conference, vol. 1, pp. 1–15 (2005)
5. Campa, G., Mammarella, M., Napolitano, M.R., Fravolini, M.L., Pollini, L., Stolarik, B.: A comparison of Pose Estimation algorithms for Machine Vision based Aerial Refueling for UAVs. In: Mediterranean Control Conference 2006, vol. 1, pp. 1–6 (2006)
6. Giulietti, F., Pollini, L., Innocenti, M., Napolitano, M.: Dynamic and control issues of formation flight. Aerospace Science and Technology 9(1), 65–71 (2005)
7. Pollini, L., Innocenti, M., Giulietti, F.: Formation Flight: a Behavioral Approach. In: AIAA Guidance, Navigation and Control Conference, Montreal, Canada, vol. 1 (2001)
8. Hartley, R.I., Kahl, F.: Optimal algorithms in multiview geometry. In: Yagi, Y., Kang, S.B., Kweon, I.S., Zha, H. (eds.) ACCV 2007, Part I. LNCS, vol. 4843, pp. 13–34. Springer, Heidelberg (2007)
9. Milella, A., Siegwart, R.: Stereo-Based Ego-Motion Estimation Using Pixel Tracking and Iterative Closest Point. In: Proceedings of the Fourth IEEE International Conference on Computer Vision Systems (2006)
10. Nistér, D.: Preemptive RANSAC for Live Structure and Motion Estimation. In: IEEE International Conference on Computer Vision (2003)
11. Olson, C.F., Matthies, L.H., Schoppers, M., Maimone, M.W.: Rover navigation using stereo ego-motion. Robotics and Autonomous Systems 43, 215–229 (2003)
12. Dame, A., Marchand, E.: Entropy-based visual servoing. In: IEEE International Conference on Robotics and Automation, Kobe, Japan (2009)
13. Jones, E., Soatto, S.: Visual-Inertial Navigation, Mapping and Localization: A Scalable Real-Time Causal Approach. International Journal of Robotics Research (2010)
14. Mourikis, A., Trawny, N., Roumeliotis, S., Johnson, A., Ansar, A., Matthies, L.: Vision-Aided Inertial Navigation for Spacecraft Entry, Descent, and Landing. IEEE Transactions on Robotics 25(2), 264–280 (2009)
15. Bryson, M., Reid, A., Ramos, F., Sukkarieh, S.: Airborne vision-based mapping and classification of large farmland environments. J. Field Robot. 27(5), 632–655 (2010)
16. Roumeliotis, S., Johnson, A., Montgomery, J.: Augmenting inertial navigation with image-based motion estimation. In: Proceedings of IEEE International Conference on Robotics and Automation (2002)
17. Tardif, J.-P., George, M., Laverne, M., Kelly, A., Stentz, A.: A new approach to vision-aided inertial navigation. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (2010)
18. Huber, P.J., Ronchetti, E.M.: Robust Statistics, 2nd edn. John Wiley & Sons, Inc. (2009)
19. Konolige, K., Agrawal, M., Solà, J.: Large-Scale Visual Odometry for Rough Terrain. Robotics Research; Springer Tracts in Advanced Robotics 66, 201–212 (2011)
20. Roumeliotis, S., Burdick, J.: Stochastic Cloning: A generalized framework for processing relative state measurements. In: Proceedings of IEEE International Conference on Robotics and Automation (2002)
21. Rogers, R.: Applied Mathematics in Integrated Navigation Systems. American Institute of Aeronautics and Astronautics (2007)
22. Lowe, D.: Object Recognition from Local Scale-Invariant Features. In: Proc. of the International Conference on Computer Vision (ICCV) (1999)
23. Le, H., Kendall, D.G.: The Riemannian Structure of Euclidean Shape Spaces: A Novel Environment for Statistics. The Annals of Statistics 21(3), 1225–1271 (1993)
24. Indiveri, G.: An Entropy-Like Estimator for Robust Parameter Identification. Entropy 11, 560–585 (2009)

25. Chakrabarti, C., De, K.: Boltzmann-Gibbs entropy: axiomatic characterization and application. Journal of Mathematics and Mathematical Sciences 23(4), 243–251 (2000)
26. Di Corato, F., Innocenti, M., Indiveri, G., Pollini, L.: An Entropy-Like Approach to Vision Based Autonomous Navigation. In: The Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China (2011)
27. Di Corato, F., Innocenti, M., Pollini, L.: An Entropy-Like Approach to Vision-Aided Inertial Navigation. In: The Proceedings of the 18th IFAC World Congress, Milan, Italy (2011)
28. Di Corato, F., Innocenti, M., Pollini, L.: Robust Vision-Aided Inertial Navigation via Entropy-Like Relative Pose Estimation. Journal of Gyroscopy and Navigation 4(1), 1–13 (2013)
29. Nistér, D., Naroditsky, O., Bergen, J.: Visual Odometry. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (2004)
30. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press (2000)
31. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment – A modern synthesis. In: Triggs, B., Zisserman, A., Szeliski, R. (eds.) Vision Algorithms 1999. LNCS, vol. 1883, pp. 298–372. Springer, Heidelberg (2000)
32. Vedaldi, A., Fulkerson, B.: Vlfeat: an open and portable library of computer vision algorithms. In: Proceedings of the International Conference on Multimedia (MM 2010) (2010)
33. Lee, T.: Vision Lab Geometry Library. UCLA VisionLab (2008), `http://vision.ucla.edu/vlg/`
34. Innocenti, M., Pollini, L.: A Synthetic Environment for Dynamic Systems Control and Distributed Simulation. IEEE Control Systems Magazine 20(2), 49–61 (2000)
35. Pollini, L., Greco, F., Mati, R., Innocenti, M.: Stereo Vision Obstacle Detection based on Scale Invariant Feature Transform Algorithm. In: Guidance Navigation and Control Conference (2007)
36. Hornegger, J., Tomasi, C.: Representation issues in the ML estimation of camera motion. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision (1999)
37. Schmidt, J., Niemann, H.: Using Quaternions for Parametrizing 3-D Rotations in Unconstrained Nonlinear Optimization. In: Proceedings of the Vision Modeling and Visualization Conference (2001)

A Covariance Analysis of Vision-Aided Inertial Navigation: Free Fall Case

Andrew Relyea[1] and Meir Pachter[2]

[1] Air Force Research Laboratory, Eglin AFB, 32542, United States of America
andrew.relyea.1@us.af.mil
[2] Air Force Institute of Technology, Wright-Patterson AFB, 45433, United States of America
meir.pachter@afit.edu

Abstract. Inertial Navigation System (INS) aiding using bearing measurements taken over time of stationary ground features is investigated. A vertical munition drop is considered. The objective is to quantify the temporal development of the uncertainty in the navigation states provided by a munition's INS which is aided by taking bearing measurements of ground objects which have been geolocated using ownship position. It is shown that a munition in "free fall," by tracking its geolocated ground features, will have the benefit of a considerable reduction in the uncertainty in the INS-provided navigation state.[1]

Keywords: navigation, estimation, Kalman Filtering.

1 Introduction

Vision, that is, optical measurements, has been extensively used in navigation in the past [5], but with the necessity of possibly operating in a Global Positioning System (GPS) denied environments looming, research in vision aided navigation has surged in popularity. In previous work [6], the location of the tracked ground objects used for Inertial Navigation System (INS) aiding had been assumed known and the degree of observability of the optical measurement arrangement was quantified. In [2] INS aiding using optical tracking of a ground feature while orbiting the ground feature was experimentally demonstrated. In the same vein, in [1], an algorithm called MonoSLAM was developed that used a single camera to perform Simultaneous Localization and Mapping (SLAM) which was aided by a 3-axes gyroscope. It was determined that positioning error growth was stopped once the system returned to a previously tracked feature and therefore a stationary robot could repeat a task indefinitely without any degradation in localization accuracy.

This paper addresses the use of bearing measurements of unknown ground objects to aid the INS of a munition in "free fall". In [6] it was shown that

[1] The views expressed in this paper are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_17

tracking two known ground objects allows for full observability of the navigation state and hence strong INS aiding action. In [8] it was shown that an aircraft, flying at a constant altitude and wings-level in the vertical plane, can use its own position, in a forward looking way, to geolocate ground objects as they come into view à la SLAM, and then, while flying towards and overflying these objects, track them to aid its own navigation state estimate. This vision-based navigation method which is akin to "bootstrapping" was mechanized through the use of a linearized Kalman filter (KF), with the states coming from the INS serving as the nominal values in the entries of the time-dependent measurement matrix used in the KF. When flying in the vertical plane cross country, this INS aiding scheme greatly decreased the uncertainty of the navigation state's estimate. This work was extended in [7] to show that the results of [8] apply in three dimensional space for a wings-level cross country flight. This paper extends that work to apply during free fall.

In this paper the focus is on INS aiding using SLAM, but not on SLAM per se; issues of image processing, to include autonomous, without human assistance, feature detection and feature correspondence, are not addressed. It is assumed that autonomous feature detection and tracking are possible-think of the SIFT image processing algorithm [3] or by the stochastic process set forth in [10]. This paper focuses exclusively on gauging the performance of INS during a vertical drop when the munition is using its own position to geolocate stationary ground objects near the target for the purpose of using said features to aid the munition's INS, and thus improve the munition's guidance and accuracy using passive means.

2 Development

The following three dimensional scenario is considered

- A vertical drop is considered and the munition has an initial velocity vector $\mathbf{v}(0) = 0$.
- The Earth is flat and nonrotating.
- The munition's initial INS alignment is perfect.
- The ground objects' elevations are known (assumed zero).

2.1 Dynamics

The navigation frame is the "inertial" (X_n, Y_n, Z_n) frame and the munition's body axes are (X_b, Y_b, Z_b). The munition's position is (x, y, z), and ψ, θ and ϕ are the munition's Euler angles. A strapdown [9] INS arrangement is considered. When falling towards a non-rotating and flat Earth as shown in Figure 1, the dynamics of the INS errors, also known as the error equations, are shown in state space notation as $\delta\dot{\mathbf{x}} = \mathbf{A}\delta\mathbf{x} + \mathbf{\Gamma}\delta\mathbf{u}$, where the navigation state's position, velocity and angles error

$$\delta\mathbf{x} = [\,\delta\mathbf{p}\ \delta\mathbf{v}\ \delta\psi\,]^T \tag{1}$$

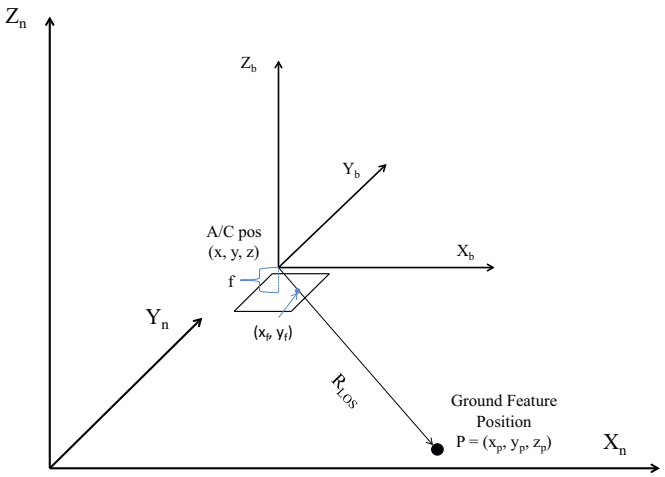


Fig. 1. The munition in 3-D space. The munition's longitudinal body axis is $-Z_b$.

and the disturbances are the three acceleromters' and the three rate gyroscopes' biases

$$\delta \mathbf{u} = [\, \delta f_x^{(b)} \; \delta f_y^{(b)} \; \delta f_z^{(b)} \; \delta \omega_x^{(b)} \; \delta \omega_y^{(b)} \; \delta \omega_z^{(b)} \,]^T \tag{2}$$

Concerning the angular errors vector $\delta\psi$:

$$\delta \mathbf{\Psi} = -\delta \mathbf{C}_b^n \mathbf{C}_n^b \tag{3}$$

and

$$\delta \mathbf{\Psi} = \delta \boldsymbol{\psi} \times \tag{4}$$

where $\delta\mathbf{\Psi}$ is the skew symmetric matrix formed from the vector $\delta\psi$ according to Eq. (4).

For small Euler angles ψ, θ, ϕ, the Directions Cosine Matrix (DCM) is approximated as:

$$\mathbf{C}_b^n(\psi, \theta, \phi) = \begin{bmatrix} 1 & -\psi & \theta \\ \psi & 1 & -\phi \\ -\theta & \phi & 1 \end{bmatrix} \tag{5}$$

and therefore its perturbation

$$\delta \mathbf{C}_b^n = \begin{bmatrix} 0 & -\delta\psi & \delta\theta \\ \delta\psi & 0 & -\delta\phi \\ -\delta\theta & \delta\phi & 0 \end{bmatrix} \tag{6}$$

During the vertical drop along the Z_n axis, the nominal $\mathbf{C}_n^b = \mathbf{I}_3$. Thus, using Eq. (3) we calculate

$$\delta \mathbf{\Psi} = \begin{bmatrix} 0 & \delta\psi & -\delta\theta \\ -\delta\psi & 0 & \delta\phi \\ \delta\theta & -\delta\phi & 0 \end{bmatrix} \tag{7}$$

and since $\delta\boldsymbol{\Psi} = \delta\boldsymbol{\psi}\times$ we recover the errors in the A/C Euler angles

$$\delta\boldsymbol{\psi} = [\,-\delta\phi\ -\delta\theta\ -\delta\psi\,]^T \tag{8}$$

Hence, the navigation state's error vector is

$$\delta\mathbf{x} = [\,\delta x\ \delta y\ \delta z\ \delta v_x\ \delta v_y\ \delta v_z\ -\delta\phi\ -\delta\theta\ -\delta\psi\,]^T \tag{9}$$

The INS error state equations are

$$\delta\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{F}^{(n)}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \delta\mathbf{x} + \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{C}^b_n & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & -\mathbf{C}^b_n \end{bmatrix} \delta\mathbf{u} \tag{10}$$

where $\mathbf{F}^{(n)} = \mathbf{f}^{(n)}\times$ is the skew symmetric matrix form of the specific force vector. The nominal specific force components during a perfect vertical drop will be time varying. Since $\mathbf{f} = \mathbf{a} - \mathbf{g}$, where $\mathbf{g}$ is the gravity vector sensed by the accelerometer cluster, until the munition reaches terminal velocity all of the nominal specific forces are zero because the munition is accelerating at a rate g in the vertical direction. Therefore

$$f^{(n)} = \begin{bmatrix} f^{(n)}_x \\ f^{(n)}_y \\ f^{(n)}_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \quad \forall \ \ 0 \le t < t_{term} \tag{11}$$

Once the munition reaches terminal velocity, at $t = t_{term}$, the accelerometers will detect nominal specific forces such that $f^{(n)}_x = a_x$, $f^{(n)}_y = a_y$, and $f^{(n)}_z = g$ where g is the acceleration of gravity and a_x and a_y are the acceleration components of the munition along its X_b and Y_b axes. Since a purely vertical drop is considered, $a_x = a_y = 0$. Therefore,

$$f^{(n)} = \begin{bmatrix} f^{(n)}_x \\ f^{(n)}_y \\ f^{(n)}_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \quad \forall \ \ t_{term} \le t \le T \tag{12}$$

Eqs. (10) - (12) represents the time-varying dynamics of navigation state's error, $(\delta\mathbf{p}, \delta\mathbf{v}, \delta\boldsymbol{\psi})$, under the assumption that the Earth is flat and non-rotating. The meaning of the angular errors' vector $\delta\boldsymbol{\psi}$, that is, its relationship to the Euler angles' errors, is determined by the munition's trajectory, that is, the nominal DCM, $\mathbf{C^n_b}$.

Having negative angle error states is unorthodox. In order for the navigation state error to be

$$\delta\mathbf{x} = [\,\delta x\ \delta y\ \delta z\ \delta v_x\ \delta v_y\ \delta v_z\ \delta\phi\ \delta\theta\ \delta\psi\,]^T \tag{13}$$

the dynamics model from Eq. (10) is modified as follows

$$\delta\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & -\mathbf{F}^{(n)}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \delta\mathbf{x} + \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{C}^b_n & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{C}^b_n \end{bmatrix} \delta\mathbf{u} \tag{14}$$

and with perfect INS alignment and initialization

$$\dot{\mathbf{C}}_b^n = \mathbf{0}, \;\; 0 \leq t \leq T$$
$$\delta\mathbf{x}(0) = \mathbf{0}$$

where T is the duration of the flight.

Since this is a vertical drop and the nominal trajectory is such that the body axes are aligned with the navigation axes, the time history of the nominal navigation variables is
$x = 0$, $y = 0$, and

$$z(t) = \left\{ \begin{array}{cc} h_{rel} - \frac{g}{2}t^2 & \forall\ 0 \leq t < t_{term} \\ h_{term} + v_{term}(t_{term} - t) & \forall\ t_{term} \leq t \leq T \end{array} \right\}$$

$\phi = \theta = \psi = 0$, where h_{rel} is the release altitude, and h_{term} and v_{term} are the altitude and velocity of the munition when the munition reaches terminal velocity. These variables are non-dimensionalized as follows

$$x \to \frac{x}{h}, \qquad y \to \frac{y}{h}, \qquad z \to \frac{z}{h},$$
$$v_x \to \frac{v_x}{v}, \qquad v_x \to \frac{v_x}{v}, \qquad v_z \to \frac{v_z}{v},$$
$$\delta f_x \to \frac{\delta f_x}{g}, \qquad \delta f_y \to \frac{\delta f_y}{g}, \qquad \delta f_z \to \frac{\delta f_z}{g},$$
$$\delta\omega_x^{(b)} \to h\frac{\delta\omega_x^{(b)}}{v}, \qquad \delta\omega_y^{(b)} \to h\frac{\delta\omega_y^{(b)}}{v}, \qquad \delta\omega_z^{(b)} \to h\frac{\delta\omega_z^{(b)}}{v},$$
$$t \to t\frac{v}{h}, \qquad T \to T\frac{v}{h}$$
$$h_{term} \to \frac{h_{term}}{h} \qquad v_{term} \to \frac{v_{term}}{v},$$

where h and v are a typical altitude and velocity when flying wings-level, t is the current time, and $T = 7$ is the nondimensional fall duration.

The non-dimensional parameters are

$$g \triangleq \frac{hg}{v^2} \qquad \text{and} \qquad a_x \triangleq \frac{ha_x}{v^2}, \qquad a_y \triangleq \frac{ha_y}{v^2}$$

Choosing the reference altitude and reference velocity

$$h = 1000[m], \qquad v = 100\left[\frac{m}{sec}\right],$$

and taking $g = 10\left[\frac{m}{sec^2}\right]$, the non-dimensional parameter
$g = 1$.

It is assumed that the sensor errors consist only of random biases that are Gaussian distributed, and that the biases are constant over time. This allows the state error vector to be augmened with the vector $\delta\mathbf{u}$; the augmented state vector is

$$\delta\mathbf{x_a} = \begin{bmatrix} \delta\mathbf{x} \\ \cdots \\ \delta\mathbf{u} \end{bmatrix}_{15\times 1} \tag{15}$$

and the dynamics matrix is augmented by the $\mathbf{\Gamma}$ matrix, as shown

$$\mathbf{A_a} = \begin{bmatrix} \mathbf{A} & \mathbf{\Gamma} \\ \mathbf{0}_{6\times 9} & \mathbf{0}_{6\times 6} \end{bmatrix}_{15\times 15} \tag{16}$$

One obtains a dynamic system in "free fall". When converted to discrete time, $\mathbf{A_a} \to \mathbf{A_{ad}} = e^{\mathbf{A_a}\Delta t}$, where Δt is the sampling interval. The augmented discrete time state dynamics become

$$\delta\mathbf{x_a}(l+1) = \mathbf{A_{ad}}\delta\mathbf{x_a}(l), \;\; l = 0, \ldots, N-1 \tag{17}$$

where l is the discrete time step counter and the non-dimensional time step is $\Delta T = \frac{T}{N} := \Delta T \frac{v}{h}$. The discrete-time dynamics matrix can be analytically derived.

This dynamics equation applies as long as the ground objects' positions are known. Assuming the ground objects are stationary, but their position is not known, two additional states, the x and y horizontal coordinates of the tracked ground objects, must be added for each tracked ground object whose position will be estimated on the fly. If the number of unknown ground features being tracked is n, then the augmented navigation state is

$$\delta\mathbf{x_a} := \begin{bmatrix} \delta\mathbf{x_a} \\ \cdots \\ \delta x_{p1} \\ \vdots \\ \delta y_{pn} \end{bmatrix}_{(15+2n)\times 1} \tag{18}$$

and

$$\mathbf{A_{ad}} := \begin{bmatrix} \mathbf{A_{ad}} & \mathbf{0}_{15\times 2n} \\ \mathbf{0}_{2n\times 15} & \mathbf{I}_{2n\times 2n} \end{bmatrix}_{(15+2n)\times(15+2n)} \tag{19}$$

If, for example, one unknown ground feature is being tracked, then the dimension of the augmented navigation state's error is 17 and if two ground features are being tracked during, then the dimension of the navigation state's error is 19. On one hand, state augmentation reduces the degree of observability, which decreases the strength of INS aiding action. On the other hand, however, the inclusion of additional features to be tracked increases the number of measurement equations, which helps wash out the measurement error.

2.2 Modeling/Calibrating the Free INS

With the dynamics from Subsection 2.1, the values for σ_a and σ_g, the uncertainty in the bias of the accelerometers and gyroscopes, respectively, are set such that during wings level horizontal flight the munition's free INS is a $100\frac{km}{hr}$ class navigation system; note that a non-dimensional hour is 360 units long. Since the dynamics are not forced, that is, there is no process noise since the latter has been relegated to the uncertainty in the initial state by augmenting the

dynamics, the calibration is performed by using the solution to the Lyapunov difference equation

$$\mathbf{P}(l+1) = \mathbf{A_{ad}}\mathbf{P}(l)\mathbf{A_{ad}^T}, \quad 0 \leq l \leq 360N - 1 \tag{20}$$

with

$$\mathbf{P}(\mathbf{0}) = \begin{bmatrix} \mathbf{0}_{9\times 9} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & diag(\sigma_a^2, \sigma_a^2, \sigma_a^2) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & diag(\sigma_g^2, \sigma_g^2, \sigma_g^2) \end{bmatrix}_{15\times 15} \tag{21}$$

The Lyapunov difference equation is linear and therefore there is a linear relationship between the uncertainty in the sensors' biases and the ensuing uncertainty in the munition's x position:

$$P_{1,1}(360N) = \alpha\sigma_a^2 + \beta\sigma_g^2 \tag{22}$$

where the coefficients α and β are constants. Therefore, Eq. (20) was solved for one non-dimensional hour twice to calculate the values of the constants α and β. The first time, σ_a was set to 1 and σ_g was set to 0. The second time, σ_a was set to 0 and σ_g was set to 1. Then assigning the errors in the accelerometers and gyroscopes an equal role/"guilt" in the uncertainty of the munition's position at time 360, the values for the variances of the sensors' biases are calculated as

$$\sigma_a = \frac{1}{\sqrt{2\alpha}} = 1.0912 \times 10^{-3} \tag{23}$$

$$\sigma_g = \frac{1}{\sqrt{2\beta}} = 9.0935 \times 10^{-6} \tag{24}$$

2.3 Measurement Equation

From the geometry in Figure 1 the relationship of the inertial position of the munition to that of the ground object P is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} - \frac{|R_{LOS}|}{\sqrt{x_f^2 + y_f^2 + f^2}} \mathbf{C}_b^n \begin{bmatrix} x_f \\ y_f \\ -f \end{bmatrix} \tag{25}$$

where x_f and y_f are the projections of the ground feature's respective x and y coordinates onto the focal plane of the camera and f is the camera's focal length. For the case when the munition falls wings level with its longitudinal axis aligned with the X_n axis, that is to say that the angle of attack is $-\frac{\pi}{2}$, the Euler angles are small. As such the DCM for relating the body frame to the navigation frame is given in Eq. (5). The first two equations in the relationship given by Eq. (25) are non-linearly dependent on the third. Now, the third equation yields

$$z_p - z = \frac{|R_{LOS}|}{\sqrt{x_f^2 + y_f^2 + f^2}} \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \mathbf{C}_b^n \begin{bmatrix} x_f \\ y_f \\ -f \end{bmatrix}$$

and thus

$$\frac{|R_{LOS}|}{\sqrt{x_f^2+y_f^2+f^2}} = \frac{z_p - z}{\begin{bmatrix}0\ 0\ 1\end{bmatrix}\mathbf{C}_b^n \begin{bmatrix}x_f\\ y_f\\ -f\end{bmatrix}} \tag{26}$$

Substituting Eq. (26) into Eq. (25) yields the two measurement equations for the three dimensional case:

$$\begin{bmatrix}x\\ y\end{bmatrix} = \begin{bmatrix}x_p\\ y_p\end{bmatrix} - \frac{z_p - z}{\begin{bmatrix}0\ 0\ 1\end{bmatrix}\mathbf{C}_b^n \begin{bmatrix}x_f\\ y_f\\ -f\end{bmatrix}} \begin{bmatrix}1\ 0\ 0\\ 0\ 1\ 0\end{bmatrix}\mathbf{C}_b^n \begin{bmatrix}x_f\\ y_f\\ -f\end{bmatrix}$$

Multiplying out the matrices yields

$$\begin{bmatrix}x\\ y\end{bmatrix} = \begin{bmatrix}x_p\\ y_p\end{bmatrix} - (z_p - z)\frac{1}{-f - \theta x_f + \phi y_f}\begin{bmatrix}x_f - \psi y_f - f\theta\\ y_f + x_f\psi + f\phi\end{bmatrix}$$

and nondimensionalizing such that

$$x_f \to \frac{x_f}{f} \qquad\qquad y_f \to \frac{y_f}{f}$$

yields

$$\begin{bmatrix}x\\ y\end{bmatrix} = \begin{bmatrix}x_p\\ y_p\end{bmatrix} - (z_p - z)\frac{1}{-1 - \theta x_f + \phi y_f}\begin{bmatrix}x_f - \psi y_f - \theta\\ y_f + x_f\psi + \phi\end{bmatrix}$$

We obtain two separate measurement equations

$$x_p - x = -(z_p - z)\frac{x_f - \psi y_f - \theta}{1 + \theta x_f - \phi y_f} \tag{27}$$

$$y_p - y = -(z_p - z)\frac{y_f + x_f\psi + \phi}{1 + \theta x_f - \phi y_f} \tag{28}$$

Due to the small angles assumption, the denominator in Eqs. (27) and (28) can be moved up such that

$$x_p - x \approx -(z_p - z)(x_f - \psi y_f - \theta)(1 - \theta x_f + \phi y_f) \tag{29}$$

$$y_p - y \approx -(z_p - z)(y_f + x_f\psi + \phi)(1 - \theta x_f + \phi y_f) \tag{30}$$

Since the munition is using ground objects to aid its INS, it can be assumed, without loss of generality, that $z_p = 0$. Due to the small values of the angles, when the former fraction is distributed out, the products of the angles are negligible, yielding

$$x_p - x = z[x_f - \theta(1 + x_f^2) + \phi x_f y_f - \psi y_f] \tag{31}$$

$$y_p - y = z[y_f - \theta x_f y_f + \phi(1 + y_f^2) + \psi x_f] \tag{32}$$

Next, perturb the states and the measurements

$$\begin{aligned} x &= x_c - \delta x & y &= y_c - \delta y & z &= z_c - \delta z \\ \theta &= \theta_c - \delta\theta & \phi &= \phi_c - \delta\phi & \psi &= \psi_c - \delta\psi \\ x_p &= x_{pc} - \delta x_p & y_p &= y_{pc} - \delta y_p & & \\ x_f &= (x_{fm} - \delta x_f) & y_f &= (y_{fm} - \delta y_f) & & \end{aligned}$$

where the subscript $_c$ indicates the navigation states components provided by the INS and the subcript $_m$ indicates measured quantities. Inserting the perturbation equations into Eq. (31)(32) yields

$$\begin{aligned} x_{pc} - x_c + \delta x - \delta x_p &= (z_c - \delta z)\Big((x_{fm} - \delta x_f) - (\theta_c - \delta\theta)(1 + (x_{fm} - \delta x_f)^2) \\ &+ (\phi_c - \delta\phi)(x_{fm} - \delta x_f)(y_{fm} - \delta y_f) - (\psi_c - \delta\psi)(y_{fm} - \delta y_f) \Big) \end{aligned}$$

Again, due to the small error in the measurements, the products of these terms can be neglected.

$$\begin{aligned} x_{pc} - x_c + \delta x - \delta x_p &= (z_c - \delta z)\Big((x_{fm} - \delta x_f) - (\theta_c - \delta\theta)(1 + x_{fm}^2 - 2x_{fm}\delta x_f) \\ &+ (\phi_c - \delta\phi)(x_{fm} y_{fm} - y_{fm}\delta x_f - x_{fm}\delta y_f) - (\psi_c - \delta\psi)(y_{fm} - \delta y_f) \Big) \\ &= (z_c - \delta z)\Big(x_{fm} - \theta_c(1 + x_{fm}^2) + \phi_c x_{fm} y_{fm} - \psi_c y_{fm} + (1 + x_{fm}^2)\delta\theta \\ &- x_{fm} y_{fm}\delta\phi + y_{fm}\delta\psi + (2\theta_c x_{fm} - \phi_c y_{fm} - 1)\delta x_f + (\psi_c - \phi_c x_{fm})\delta y_f \Big) \end{aligned}$$

Similarly, inserting the perturbations in the second measurement equation, Eq.-(32), yields

$$\begin{aligned} y_{pc} - y_c + \delta y - \delta y_p &= (z_c - \delta z)\Big((y_{fm} - \delta y_f) - (\theta_c - \delta\theta)(x_{fm} - \delta x_f)(y_{fm} - \delta y_f) \\ &+ (\phi_c - \delta\phi)(1 + (y_{fm} - \delta y_f)^2) + (\psi_c - \delta\psi)(x_{fm} - \delta x_f) \Big) \\ &= (z_c - \delta z)\Big((y_{fm} - \delta y_f) - (\theta_c - \delta\theta)(x_{fm} y_{fm} - y_{fm}\delta x_f - x_{fm}\delta y_f) \\ &+ (\phi_c - \delta\phi)(1 + y_{fm}^2 - 2y_{fm}\delta y_f) + (\psi_c - \delta\psi)(x_{fm} - \delta x_f) \Big) \\ &= (z_c - \delta z)\Big(y_{fm} - \theta_c x_{fm} y_{fm} + \phi_c(1 + y_{fm}^2) + \psi_c x_{fm} + x_{fm} y_{fm}\delta\theta \\ &- (1 + y_{fm}^2)\delta\phi - x_{fm}\delta\psi + (\theta_c y_{fm} - \psi_c)\delta x_f + (\theta_c x_{fm} - 2\phi_c y_{fm} - 1)\delta y_f \Big) \end{aligned}$$

Moving all the error terms to the Right Hand Side (RHS) of the equation and all the non-error terms to the Left Hand Side (LHS) yields

$$\begin{aligned} &x_{pc}-x_c - z_c(x_{fm} - \theta_c(1 + x_{fm}^2) + \phi_c x_{fm} y_{fm} - \psi_c y_{fm}) = -\delta x + \delta x_p \\ &- \delta z(x_{fm} - \theta_c(1 + x_{fm}^2) + \phi_c x_{fm} y_{fm} - \psi_c y_{fm}) - z_c x_{fm} y_{fm} \delta\phi + z_c y_{fm} \delta\psi \\ &+ z_c(1 + x_{fm}^2)\delta\theta + z_c(2\theta_c x_{fm} - \phi_c y_{fm} - 1)\delta x_f + z_c(\psi_c - \phi_c x_{fm})\delta y_f \end{aligned} \tag{33}$$

and

$$\begin{aligned} &y_{pc}-y_c - z_c(y_{fm} - \theta_c x_{fm} y_{fm} + \phi_c(1 + y_{fm}^2) + \psi_c x_{fm}) = -\delta y + \delta y_p \\ &- \delta z(y_{fm} - \theta_c x_{fm} y_{fm} + \phi_c(1 + y_{fm}^2) + \psi_c x_{fm}) + z_c x_{fm} y_{fm} \delta\theta - z_c x_{fm} \delta\psi \\ &- z_c(1 + y_{fm}^2)\delta\phi + z_c(\theta_c y_{fm} - \psi_c)\delta x_f + z_c(\theta_c x_{fm} - 2\phi_c y_{fm} - 1)\delta y_f \end{aligned} \tag{34}$$

Finally, nondimensionalizing such that

$$x_p \to \frac{x_p}{h} \qquad y_p \to \frac{y_p}{h} \qquad z_p \to \frac{z_p}{h},$$

the nondimensional altitude is $z_c = z_l$. In addition, for the purpose of covariance analysis, set all of the calculated values on the RHS equal to the nominal values. This causes all of the angles to go to zero, simplifying the measurement Eqs. (33) and (34). Also, on the RHS set $x_{fm} := x_f$ and $y_{fm} := y_f$.

$$\begin{aligned} &x_{pc}-x_c - z_c(x_{fm} - \theta_c(1 + x_{fm}^2) + \phi_c x_{fm} y_{fm} - \psi_c y_{fm}) = \\ &- \delta x - \delta z x_f + \delta\theta(1 + x_f^2) z_l - \delta\phi x_f y_f z_l + \delta\psi y_f z_l + \delta x_p - \delta x_f z_l \end{aligned} \tag{35}$$

and

$$\begin{aligned} &y_{pc}-y_c - z_c(y_{fm} - \theta_c x_{fm} y_{fm} + \phi_c(1 + y_{fm}^2) + \psi_c x_{fm}) = \\ &- \delta y - \delta z y_f + \delta\theta x_f y_f z_l - \delta\phi(1 + y_f^2) z_l - \delta\psi x_f z_l + \delta y_p - \delta y_f z_l \end{aligned} \tag{36}$$

Now the measurement equation can be written in state space form,

$$\delta\mathbf{Z} = \mathbf{H}_u(l)\delta\mathbf{x} \tag{37}$$

where $\delta\mathbf{Z}$ is the difference between the expected measurement and the actual measurement, and the time dependent observation matrix $\mathbf{H}(l)$ for one unknown ground feature is

$$\mathbf{H}_u(l) = \begin{bmatrix} -1 & 0 \\ 0 & -1 \\ -x_f & -y_f \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -x_f y_f z_l & -(1+y_f^2) z_l \\ (1+x_f^2) z_l & x_f y_f z_l \\ y_f z_l & -x_f z_l \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}^T \tag{38}$$

where the subscript $_u$ indicates that the position of the ground object being tracked is unknown. The nondimensional measurement error is $[\delta x_f, \delta y_f]^T$.

For the sake of observability [6] two ground objects will be tracked. The observation matrix for tracking two unknown ground features

$$\mathbf{H}_{uu}(l) = \begin{bmatrix} -1 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 \\ -x_{f1} & -y_{f1} & -x_{f2} & -y_{f2} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -x_{f1} y_{f1} z_l & -(1+y_{f1}^2) z_l & -x_{f2} y_{f2} z_l & -(1+y_{f2}^2) z_l \\ (1+x_{f1}^2) z_l & x_{f1} y_{f1} z_l & (1+x_{f2}^2) z_l & x_{f2} y_{f2} z_l \\ y_{f1} z_l & -x_{f1} z_l & y_{f2} z_l & -x_{f2} z_l \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}^T \tag{39}$$

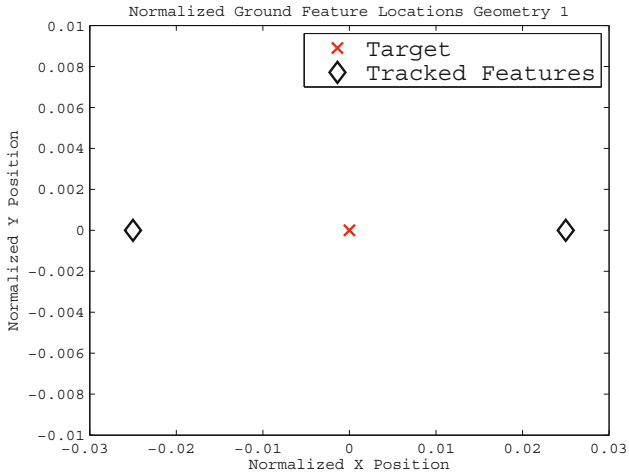


Fig. 2. The locations of the tracked ground features in the first geometry

3 Performance of Aided INS

The munition is in a free fall, that is, $v_z(0) = 0$ and its acceleration is $-g$ along the Z_n axis until it reaches terminal velocity. The nominal vertical drop is such that $h_{rel} = 7500\ [m]$. The terminal $v_{term} = 100\ \left[\frac{m}{s}\right]$ so that the free fall is 10 seconds and the total duration of the vertical drop is $T = 80$ seconds. The nominal trajectory is $x(t) = 0$, $y(t) = 0$ and

$$z(t) = \left\{ \begin{array}{cc} 7500 - 5t^2 & \forall\ 0 \leq t < t_{term} \\ 7000 - 100(t - t_{term}) & \forall\ t_{term} \leq t \leq T \end{array} \right\}$$

Once the terminal velocity has been reached there is no further acceleration and the vertical speed is constant. Two ground feature geometries were considered. The first is shown in Figure 2. For this geometry, in the observation matrix

$$x_{f1}(l) = \frac{x_{p1}}{z_l} = \frac{.025}{z_l}$$
$$x_{f2}(l) = \frac{x_{p2}}{z_l} = \frac{-.025}{z_l}$$
$$y_{f1}(l) = y_{f2}(l) = 0, \quad l = 0, \ldots, N$$

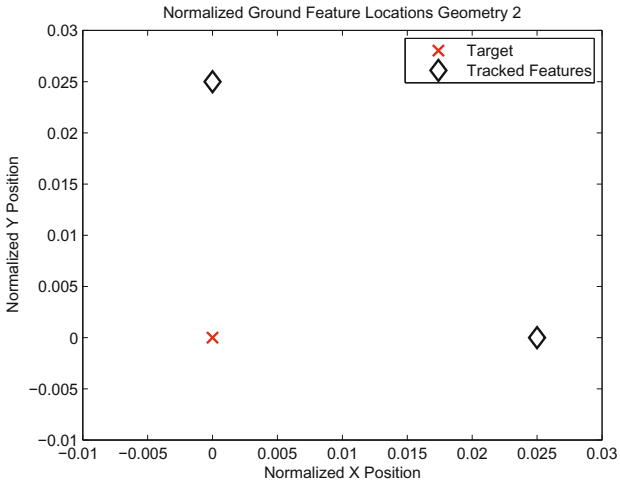


Fig. 3. The locations of the tracked ground features in the second geometry

The second geometry is shown in Figure 3. For this geometry, in the observation matrix

$$x_{f1}(l) = \frac{x_{p1}}{z_l} = \frac{.025}{z_l}$$
$$x_{f2}(l) = y_{f1}(l) = 0$$
$$y_{f2}(l) = \frac{y_{p2}}{z_l} = \frac{.025}{z_l}, \quad l = 0, \ldots, N$$

3.1 Initialization

It is stipulated that initially, the INS has zero error in the navigation states, that is, the INS alignment was perfect, and the states representing the biases in the sensors are

$$\delta f_x^{(b)} \sim N(0, \sigma_a^2) \qquad \delta f_y^{(b)} \sim N(0, \sigma_a^2) \qquad \delta f_z^{(b)} \sim N(0, \sigma_a^2)$$
$$\delta \omega_x^{(b)} \sim N(0, \sigma_g^2) \qquad \delta \omega_y^{(b)} \sim N(0, \sigma_g^2) \qquad \delta \omega_z^{(b)} \sim N(0, \sigma_g^2)$$

The x, y and z accelerometers are of the same quality, and also the x, y and z gyroscopes are of the same quality. Thus

$$\delta \mathbf{x}(\mathbf{0}) \sim N(0, \mathbf{P}(\mathbf{0})) \tag{40}$$

with the initial covariance matrix $\mathbf{P}(0)$ given by Eq. (21).

3.2 Transitioning between Measurement Epochs

In the pure free fall case, there are three measurement epochs. The first is while the munition is accelerating toward the earth. The duration of the first epoch is one nondimensional second. There is $N = 100$ discrete steps per nondimensional second.

Therefore, in epoch 1 the observation matrix $\mathbf{H_{uu}}(l)$, and the dynamics matrix $\mathbf{A_{ad}}_{19\times 19}$ based on the nominal accelerometer values from Eq. (11) were used. In the first epoch, the uncertainty of the states were propagated for one hundred steps using the covariance propagate and update equations of the Kalman filter [4]

$$\mathbf{P}(l+1)^{-} = \mathbf{A_{ad}}\mathbf{P(l)}^{+}\mathbf{A_d} \tag{41}$$

$$\mathbf{K} = \mathbf{P}(l+1)^{-}\mathbf{H_{uu}}(l)^{T}[\mathbf{H_{uu}}(l)\mathbf{P}(l+1)^{-}\mathbf{H_{uu}}(l)^{T} + \mathbf{R}]^{-1} \tag{42}$$

$$\mathbf{P}(l+1)^{+} = (\mathbf{I_{19}} - \mathbf{K}\mathbf{H_{uu}}(l))\mathbf{P}(l+1)^{-} \tag{43}$$

where $\mathbf{R}$ is the measurement uncertainty caused by one pixel in the camera's focal plane

$$\begin{bmatrix} \delta x_{f1} \\ \delta y_{f1} \\ \delta x_{f2} \\ \delta y_{f2} \end{bmatrix} \sim N(0, \mathbf{R})$$

We assume a 9 Megapixel camera with an aspect ratio of 1. However, recall from Eqs. (35) and (36) that the measurement error terms on the RHS are multiplied by the time-varying altitude z_l. Therefore, the nondimensional

$$\mathbf{R} = z_l^2 \begin{bmatrix} \frac{1}{9} & 0 & 0 & 0 \\ 0 & \frac{1}{9} & 0 & 0 \\ 0 & 0 & \frac{1}{9} & 0 \\ 0 & 0 & 0 & \frac{1}{9} \end{bmatrix} \times 10^{-6} \tag{44}$$

where z_l is squared to match $\mathbf{R}$. If $\mathbf{R}$ was standard deviation of the noise instead of the variance then z_l would not need to be squared.

At the conclusion of the first one hundred steps/the first measurement epoch the munition reached terminal velocity. Thus the next time block required the use of the dynamics matrix $\mathbf{A_{ad}}_{19\times 19}$ to be based on the nominal accelerometer values from Eq. (12). The second measurement epoch was 6 nondimensional seconds long. Without any further transitions, Eqs. (41)-(43) were repeated for $101 \leq l \leq 700$, using the updated dynamics matrix.

We assumed a camera Field of View (FOV) of 50 milliradians. Based on the geometry set forth in Figure 2, the ground features leave the camera FOV when the munition is 1000 meters above the ground, h_{final}. Thus the third and final measurement epoch started when the munition reached h_{final} in 70 seconds. In the final 10 seconds of its flight the INS was not aided, and we reverted to a free INS. As such the calculations using the KF equations, (41)-(43), were terminated and the calculation for the free INS, Eq. (20), was used for the last 10 seconds.

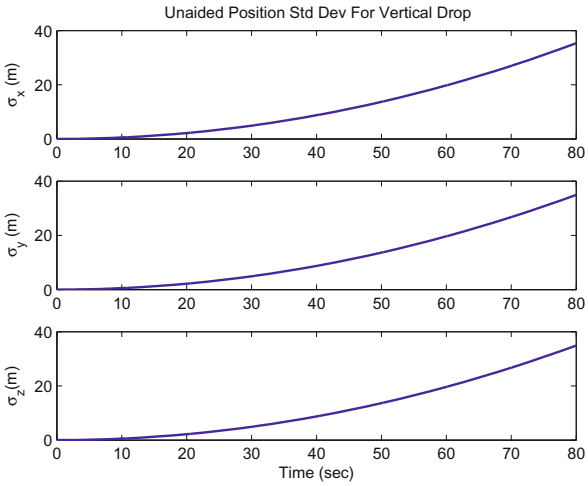


Fig. 4. The development of the standard deviation of the position errors in the unaided, free, INS

This process was completed for both geometries listed at the beginning of Section 3.

4 Calculations

Initially the standard deviations of the navigation states of the unaided INS were plotted as a baseline in Figures 4-6. The standard deviations of the x, y and z positions were almost 27 meters after 70 dimensional seconds. There was a large degree of aiding achieved with the optical tracking scheme, shown in Figures 7-9. After falling for 70 seconds the standard deviations in the x and y positions are 10 and 12 centimeters, respectively, with an uncertainty in the z position of about 3.5 meters. But the position states were not the only navigation states that were aided. Four of the other six navigation states showed significant reduction in uncertainty as well. What is truly remarkable is how much removing the symmetry of the feature location geometry reduced the uncertainty, shown in Figures 10-12. The most significant reductions had to do with the z states. The z position uncertainty peaked at .13 millimeters, without degrading the other two position states. The same happened with the velocity states. The ϕ angular state's peak uncertainty was cut almost in half from the max uncertainty in the first geometry.

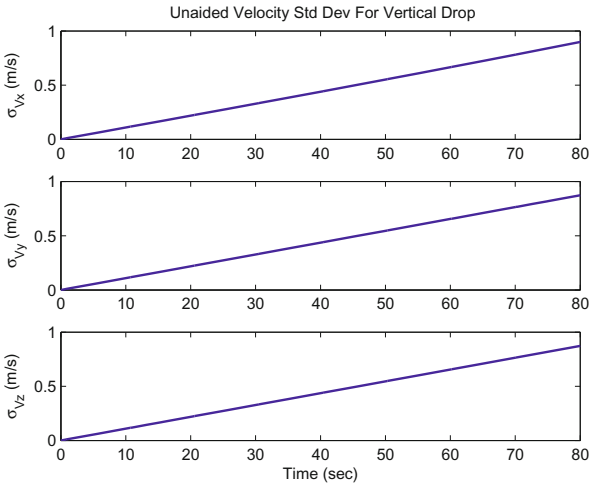


Fig. 5. The development of the standard deviation of the velocity errors in the unaided, free, INS

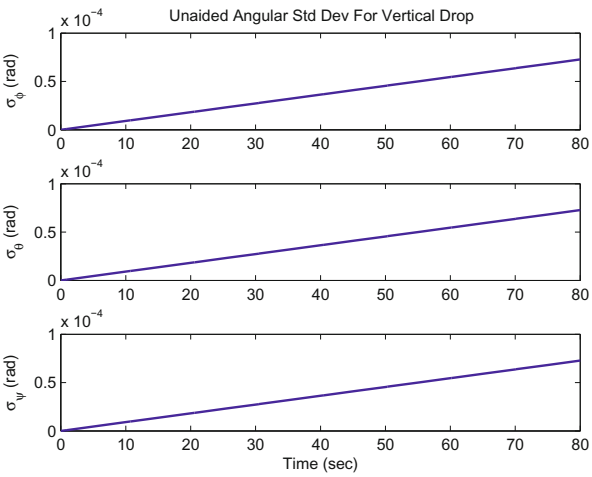


Fig. 6. The development of the standard deviation of the angle errors in the unaided, free, INS

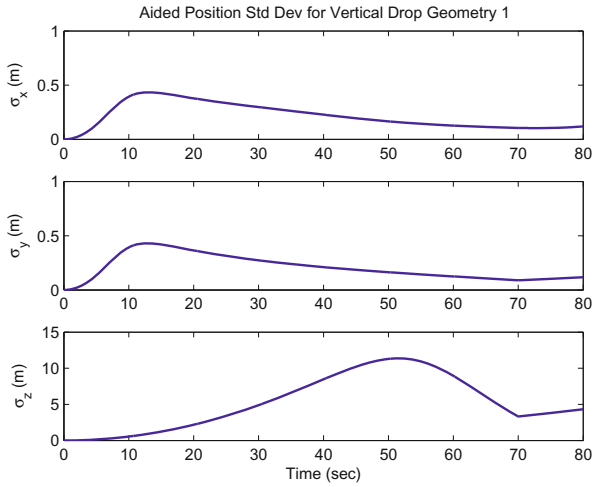


Fig. 7. The development of the standard deviation of the position errors in the aided INS for geometry 1

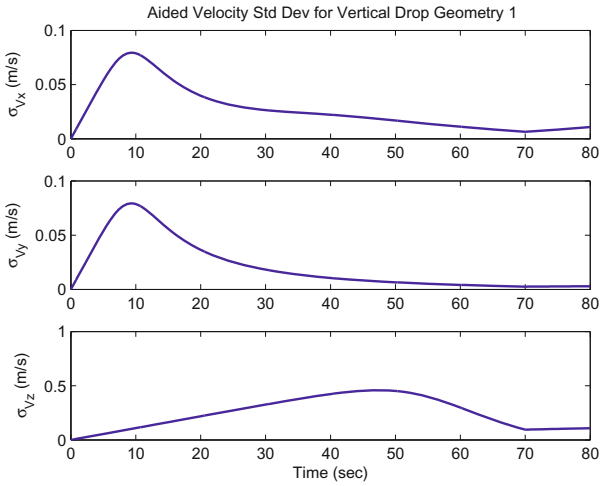


Fig. 8. The development of the standard deviation of the velocity errors in the aided INS for geometry 1

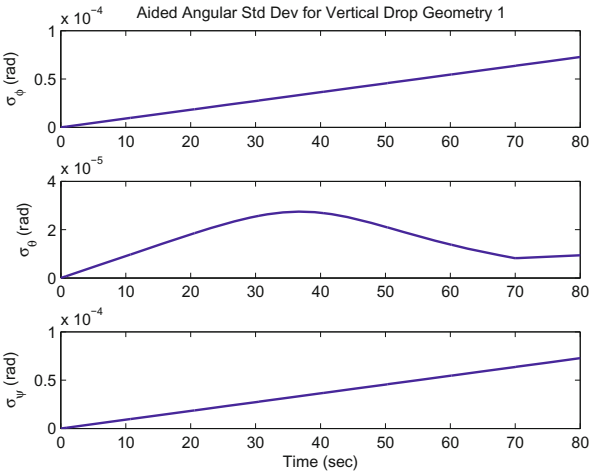


Fig. 9. The development of the standard deviation of the angle errors in the aided INS for geometry 1

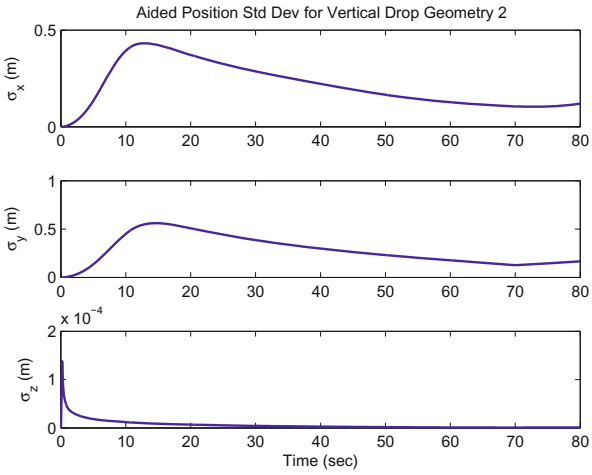


Fig. 10. The development of the standard deviation of the position errors in the aided INS for geometry 2

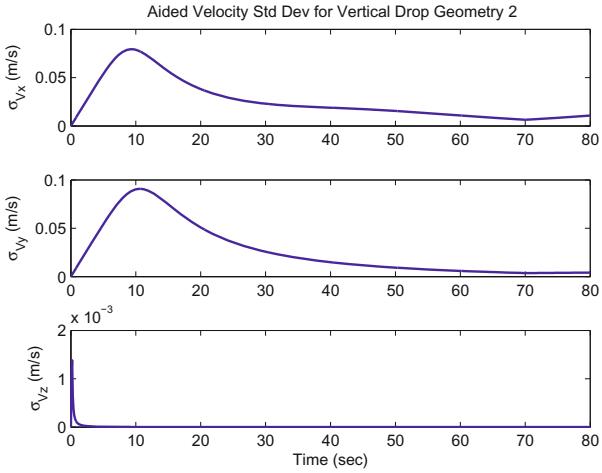


Fig. 11. The development of the standard deviation of the velocity errors in the aided INS for geometry 2

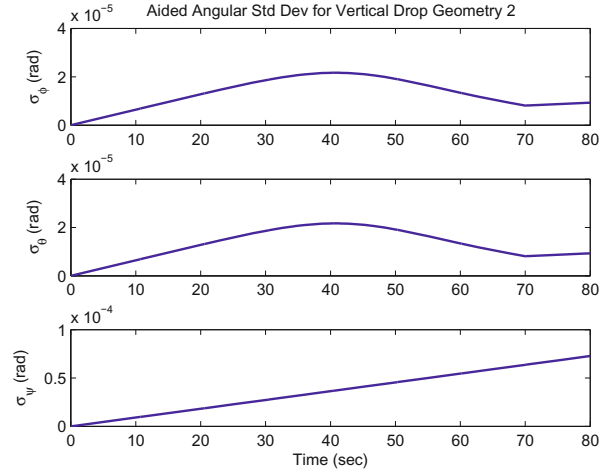


Fig. 12. The development of the standard deviation of the angle errors in the aided INS for geometry 2

5 Conclusion

It is clearly shown that using visual bearings-only measurements greatly reduces the uncertainty in the INS provided navigation state estimate during a vertical drop. This improvement will allow for more accurate guidance of the munition, and therefore a greater chance of the munition hitting its target. This is achieved using a passive means and autonomous guidance.

References

1. Davison, A., Reid, I., Molton, N., Stasse, O.: Monoslam: Real-time single camera slam. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(6), 1052–1067 (2007)
2. Giebner, M.: Tightly-Coupled Image-Aided Inertial Navigation System via a Kalman Filter. Master's thesis, AFIT, WPAFB, OH (2003)
3. Ke, Y., Sukthankar, R.: Pca-sift: A more distinctive representation for local image descriptors. In: 2004 IEEE Computer Society Conference on cumputer Vision and Pattern Recognition (CVPR 2004), vol. 2, pp. 506–513 (2004)
4. Maybeck, P.S.: Stochastic Models, Estimation, and Control, vol. 2. Academic Press, New York (1983)
5. Pacher, M., Porter, A., Polat, M.: Inertial navigation system aiding using bearings-only measurements of an unknown ground object. ION Journal Navigation 53(1), 1–20 (2006)
6. Pachter, M., Mutlu, G.: The Navigation Potential of Ground Feature Tracking. In: Dynamics of Information Systems: Theory and Application, pp. 287–303. Springer (2010)
7. Relyea, A., Pachter, M.: A covariance analysis of vision-aided inertial navigation: 3-d case (November 2011) (not currently submitted for publication)
8. Relyea, A., Pachter, M.: A covariance analysis of vision-aided inertial navigation. In: 51st IEEE Conference on Decision and Control, Maui, HI (December 2012) (submitted to CDC12)
9. Titterton, D., Weston, J.: Strapdown Inertial Navigation Technology, 2nd edn. The Institution of Engineernig and Technology, London (2004)
10. Veth, M., Raquet, J., Pachter, M.: Stochastic constraints for efficient image correspondence search. IEEE Transactions on Aerospace and Electronic Systems 42(3), 973–982 (2006)

Investigation of a Symmetric Vibrating Gyroscope Characteristics Using a Simplified Dynamic Model

Ilia Rapoport[1] and Daniel Choukroun[2,3]

[1] Control and Navigation, Elbit Systems, Electro-Optics ELOP Ltd, Rehovot, Israel
ilia.rapoport@gmail.com
[2] Mechanical Eng. Dept., Ben-Gurion University, POB 653, Beer-Sheva, Israel
danielch@bgu.ac.il
[3] Faculty of Aerospace Eng., Delft Univ. of Techn., 2629 HS, Delft, Netherlands

Abstract. In this work a dynamical model for MEMS vibrational gyroscopes is developed that generalizes a previous work, allows for simpler but accurate qualitative and quantitative analysis of several sources of angular velocity measurement errors, and opens avenues for future developments in MEMS vibrational gyroscopes designs. The proposed model equations govern the dynamics of the amplitudes rather than the dynamics of the rapid oscillatory processes. The characteristics of this approximate model are significantly slower than the driving frequency. It allows a linear time-invariant analysis of the angular velocity measurement errors. These may be direct like a bias caused by the structural damping, or indirect, due e.g. to the unmatched frequencies between the drive and the sense channels. The approximate model was validated on a particular numerical example by examination and comparison of the frequency responses. A simple proportional feedback control was designed for both the drive and the quadrature loops, showing the potential impact of the feedback gains on the low-frequencies error. Being a linear time-invariant model, this model will easily lend itself to the development of more advanced control strategies.

1 Introduction

For over three decades, microelectromechanical system (MEMS) gyroscopes have been a growing focus of research and development in science and technology (see a review in [1]), many have been developed [2], and a few are on the market. In the aerospace engineering field, their scope essentially focus on tactical systems. They find very successful applications, such as attitude determination on-board spacecraft with complementary sensors like star-trackers (see e.g.[3]). This work is concerned with the derivation of a dynamical model for MEMS vibrational gyroscopes that generalizes a previous work[4], allows for simpler but accurate qualitative and quantitative analysis of several sources of angular velocity measurement errors, and opens avenues for future developments in MEMS vibrational gyroscopes designs.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_18

The paper presents an extended model of the vibrational gyroscope where coupling in the damping and stiffness properties of the drive and sense axes are allowed. Following an alternative technique to "stochastic averaging"[5], a simpler model is developed for the dynamics of the in-phase and quadrature sense vibration amplitudes, which keeps the slow dynamics essential characteristics. This simple but accurate model permits straightforward qualitative and quantitative analysis of systematic errors, such as a bias related to the coupling damping coefficient, or of issues in operational mode, such as a narrow bandwidth in open-loop operations. The current results illustrate how the proposed model lends itself to a linear analysis of the impact of electronic sensing noise on the gyro performances. Preliminary results on the noise to angular error frequency response emphasize the need for a careful design of the quadrature loop. The proposed model clearly provides a promising platform for future developments of filtering and control strategies allowing e.g. mitigation of the gyro drift error.

2 Mathematical Model

The operation principle of vibrational gyroscopes relies, in essence, on the sensing of a Coriolis force. This force, which is acting on a mass driven in a controlled vibration along an axis (the drive axis) perpendicular to the angular velocity vector, is directed, and can thus be sensed, along an axis perpendicular to both directions (the sense axis). The Coriolis force, which can be sensed by measuring the vibration amplitude along the sense axis, is proportional to the angular velocity. The observation of the amplitude of the sense axis vibration may thus provide a measurement of the angular velocity.

2.1 Full Gyroscope Model

Let $x(t)$ and $y(t)$ denote the drive axis and the sense axis vibration signals, respectively. The dynamics of $x(t)$ and $y(t)$ are governed by the following generic system of equations:

$$m\begin{bmatrix}\ddot{x}\\ \ddot{y}\end{bmatrix}+\begin{bmatrix}c_{11}\ c_{12}\\ c_{12}\ c_{22}\end{bmatrix}\begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix}+\begin{bmatrix}k_{11}\ k_{12}\\ k_{12}\ k_{22}\end{bmatrix}\begin{bmatrix}x\\ y\end{bmatrix}+4A_g m\Omega\begin{bmatrix}0\ -1\\ 1\ \ 0\end{bmatrix}\begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix}=\begin{bmatrix}f_x\\ f_y\end{bmatrix} \qquad (1)$$

where $f_x(t)$ denotes the control force used to maintain the drive axis vibration at a specified amplitude (called also the drive force), $f_y(t)$ denotes the sense force, which is zero in the open-loop operation and is used to keep the sensing signal $y(t)$ at zero in a closed-loop operational mode [4], m is the mass, c_{ij} and k_{ij} are the damping and stiffness coefficients, $\Omega(t)$ is the rotation rate and A_g is a geometry term taking e.g. the value of 0.5 for a suspended mass [4] and approximately 0.4 for a vibrating ring. As opposed to [4], where the coupling between the drive and sense axes only occurs as a result of the Coriolis force, we assume that additional coupling happens as a result of the off-diagonal damping and stiffness

coefficients. However, resulting from the symmetry of the vibrational structure, the following assumptions are made:

$$|k_{11} - k_{22}| \ll k_{11}, \quad |k_{12}| \ll k_{11} \tag{2}$$

$$|c_{11} - c_{22}| \ll c_{11}, \quad |c_{12}| \ll c_{11} \tag{3}$$

2.2 Approximate Gyroscope Model

Following the usual approach [2,4], the drive force is assumed to excite the dynamics of the drive axis at the resonance frequency, i.e.

$$f_x(t) = F_x(t)\, \sin(\omega_x t) \tag{4}$$

where

$$\omega_x \triangleq \sqrt{\frac{k_{11}}{m}} \tag{5}$$

and the drive axis vibration is assumed to perfectly track a desired pattern thanks to a phase-locked loop and an automatic gain control that adjust $F_x(t)$ until the desired amplitude is reached (an example of such an excitation scheme can be found in [7]). Thus, it will be assumed that the drive axis vibration is expressed as follows:

$$x(t) = X_0 \cos(\omega_x t) \tag{6}$$

Notice that the amplitudes reached by $x(t)$ are typically orders of magnitude greater than those of $y(t)$. This allows neglecting the influence of $y(t)$ on the dynamics of $x(t)$, and facilitates the phase-locked loop regulation task. An approximate gyroscope model is here developed for the dynamics of the sense axis vibration, $y(t)$. This analysis will assume the following expressions for the sense force $f_y(t)$ and for the vibration $y(t)$:

$$f_y(t) = F_{Ys}(t) \sin(\omega_x t) + F_{Yc}(t) \cos(\omega_x t) \tag{7}$$

$$y(t) = Y_s(t) \sin(\omega_x t) + Y_c(t) \cos(\omega_x t) \tag{8}$$

The term $Y_c(t) \cos(\omega_x t)$ is called the "in-phase" part of the y-signal, which is in phase with $x(t)$ (see Eq. (6)), while the term $Y_s(t) \sin(\omega_x t)$ is called the "quadrature" part of the y-signal, since it has a 90 deg phase difference with $x(t)$. Using Eqs. (6)-(8), expressing the time differentials of $y(t)$ as given in Eq. (8), and substituting the resulted expressions into the sense axis equations in Eq. (1) yields the following two ordinary differential equations for the amplitudes $Y_s(t)$ and $Y_c(t)$:

$$m(\ddot{Y}_s - 2\dot{Y}_c \omega_x - Y_s \omega_x^2) + c_{22}(\dot{Y}_s - Y_c \omega_x) + k_{22} Y_s - \\ - \omega_x (c_{12} + 4A_g m\Omega) X_0 = F_{Ys} \tag{9}$$

$$m(\ddot{Y}_c + 2\dot{Y}_s \omega_x - Y_c \omega_x^2) + c_{22}(\dot{Y}_c + Y_s \omega_x) + k_{22} Y_c + k_{12} X_0 = F_{Yc} \tag{10}$$

Because of the symmetry assumption (yielding Eqs. (2)(3)), it can be shown that the characteristic frequency of the amplitude signals $Y_s(t)$ and $Y_c(t)$ is very slow compared to the vibration frequency ω_x. This is sometimes called the *slow envelope approximation* [6, p. 114]. As a consequence, it can be assumed that the following relations hold:

$$|\dot{Y}| \ll |Y|\,\omega_x \tag{11}$$

$$|\ddot{Y}| \ll |\dot{Y}|\,\omega_x \tag{12}$$

where Y might denote Y_s or Y_c. Using Eqs. (11)(12) in Eqs. (9)(10) yields the following approximate model for the dynamics of $Y_c(t)$ and $Y_s(t)$:

$$m(-2\dot{Y}_c\,\omega_x - Y_s\,\omega_x^2) + c_{22}(-Y_c\,\omega_x) + k_{22}Y_s - \omega_x(c_{12} + 4A_g m\Omega)X_0 = F_{Ys} \tag{13}$$

$$m(2\dot{Y}_s\,\omega_x - Y_c\,\omega_x^2) + c_{22}(Y_s\,\omega_x) + k_{22}Y_c + k_{12}X_0 = F_{Yc} \tag{14}$$

Finally, defining the resonance frequency in the sense axis, ω_y, and the "quality factor", Q_y, as follows:

$$\omega_y \triangleq \sqrt{\frac{k_{22}}{m}} \tag{15}$$

$$Q_y \triangleq \frac{\omega_y}{(c_{22}/m)} \tag{16}$$

Equations (13)(14) are rewritten as follows

$$\dot{Y}_s + \frac{\omega_y}{2Q_y}Y_s + \left(\frac{\omega_y^2 - \omega_x^2}{2\,\omega_x}\right)Y_c = -\frac{k_{12}}{2m\,\omega_x}X_0 + \frac{1}{2m\,\omega_x}F_{Yc} \tag{17}$$

$$\dot{Y}_c + \frac{\omega_y}{2Q_y}Y_c - \left(\frac{\omega_y^2 - \omega_x^2}{2\,\omega_x}\right)Y_s = -\left(\frac{c_{12}}{2m} + 2A_g\Omega\right)X_0 - \frac{1}{2m\,\omega_x}F_{Ys} \tag{18}$$

Equations (17)(18) provide a simple approximate model governing the dynamics of Y_c and Y_s, i.e., the in-phase and quadrature amplitudes of the sense signal, respectively. The inputs to this simple model are F_{Yc}, which denotes the in-phase signal control force, and F_{Ys}, which is the quadrature signal control force. It is noticed that the angular velocity Ω (that we want to measure) enters the in-phase amplitude dynamics. It is also noticed that the coefficient c_{12} enters the in-phase amplitude dynamics in such a way that it can not be distinguished from the angular velocity. On the other hand, the coefficient k_{12} perturbs the dynamics of the quadrature signal. The proposed model features the following advantages over the model introduced in [4]: the stiffness and damping coupling between the drive and the sense axes is not neglected ($k_{12} \neq 0, c_{12} \neq 0$), and the quadrature signal can be controlled (via $F_{Yc}(t)$). A more compact expression for

Eqs. (15)(18) is provided below:

$$\begin{bmatrix}\dot{Y}_s\\ \dot{Y}_c\end{bmatrix} = \begin{bmatrix}-\alpha & -\beta\\ \beta & -\alpha\end{bmatrix}\begin{bmatrix}Y_s\\ Y_c\end{bmatrix} + \begin{bmatrix}0 & -\gamma X_0 & 0 & 0 & \gamma\\ -2A_g X_0 & 0 & \frac{-X_0}{2m} & -\gamma & 0\end{bmatrix}\begin{bmatrix}\Omega\\ k_{12}\\ c_{12}\\ F_{Ys}\\ F_{Yc}\end{bmatrix} \tag{19}$$

where

$$\alpha \triangleq \frac{\omega_y}{2Q_y} \tag{20}$$

$$\beta \triangleq \frac{\omega_y^2 - \omega_x^2}{2\,\omega_x} \tag{21}$$

$$\gamma \triangleq \frac{1}{2m\,\omega_x} \tag{22}$$

Notice that the coefficient β can be approximated as follows:

$$\beta = \frac{(\omega_y^2 - \omega_x^2)}{2\,\omega_x} \approx \omega_y - \omega_x \tag{23}$$

In the above model, the parameters X_0 and ω_x are in direct control of the designer. The parameters α (which depends on the mass and the damping coefficient c_{22}), A_g, and γ can be relatively accurately determined, and are thus considered known to the gyroscope designer. On the other hand, parameters like the frequency mismatch, β, and coupling coefficients like k_{12} and c_{12} should be considered uncertain, possibly time-varying, e.g. under the influence of the temperature. In Eq. (19), the parameters k_{12} and c_{12} appear as inputs. Indeed, these signals being time invariant or time varying, one objective consists in investigating the response of the system and to check how well the approximate model can predict the full model behavior. For that purpose the various single-input-single-output transfer functions are required. They are provided next in the form of a 2×5 transfer matrix.

$$\frac{1}{d_c(s)}\begin{bmatrix}2A_g X_0\beta & -\gamma X_0(s+\alpha) & \frac{X_0\beta}{2m} & \beta\gamma & \gamma(s+\alpha)\\ -2A_g X_0(s+\alpha) & -\gamma X_0\beta & -\frac{X_0}{2m}(s+\alpha) & -\gamma(s+\alpha) & \beta\gamma\end{bmatrix} \tag{24}$$

where

$$d_c(s) = (s+\alpha)^2 + \beta^2 \tag{25}$$

The appealing simplicity of this approximate model will help in analyzing the effect of these uncertain terms on the gyroscope's performances.

2.3 Mode of Operation for Angular Velocity Measurement

There are two modes of operations: an open-loop mode and a closed-loop mode. In the open-loop mode no control forces are applied, i.e. $F_{Ys} = F_{Yc} = 0$, and the

angular velocity results from the observation of the in-phase signal in steady-state. In the ideal case, where $k_{12} = c_{12} = 0$ and $\omega_y = \omega_x$, and for a step Ω_0 in Ω, the approximate model allows for an easy development of the steady-state equation for Y_c:

$$Y_c(\infty) = -4A_g \frac{Q_y}{\omega_x} X_0 \, \Omega_0 \tag{26}$$

Inspection of Eq. (26) suggests a usual approach of extracting the angular velocity from the observed $Y_c(t)$[4]:

$$\hat{\Omega}(t) = -\frac{\omega_x}{4A_g Q_y X_0} \, Y_c(t) \tag{27}$$

which is done at any time t. Notice from Eq. (27) that the quality factor, i.e. the damping coefficient, needs to be known in the open-loop operational mode. The resulting open-loop transfer functions are provided in Table 1. In the closed-loop mode, the objective is to bring the in-phase and quadrature signals to zero via feedback. A simple manipulation of the approximate model, in the ideal case ($k_{12} = c_{12} = 0$ and $\omega_y = \omega_x$) for a step Ω_0 yields the following steady state relationship:

$$0 = 2A_g \Omega_0 X_0 - \frac{1}{2m \, \omega_x} \, F_{Ys}(\infty) \tag{28}$$

Solving Eq. (28) for Ω_0 provides the measurement equation for the closed-loop mode of operation:

$$\hat{\Omega}(t) = \frac{1}{4A_g \, \omega_x m X_0} \, F_{Ys}(t) \tag{29}$$

at anytime t. Any deviation from this ideal case, i.e. $c_{12} \neq 0$, or $k_{12} \neq 0$, or $\omega_x \neq \omega_y$ is expected to yield an error in the gyro angular velocity measurement.

Table 1. Open-loop transfer functions with output $\hat{\Omega}$

Input	$H^{OL}(s) = \frac{\hat{\Omega}(s)}{\cdot}$
$\Omega \, [rad/sec]$	$\frac{\omega_x}{2Q_y} \frac{(s+\alpha)}{(s+\alpha)^2+\beta^2}$
$k_{12} \, [N/m]$	$\frac{1}{8mA_gQ_y} \frac{\beta}{(s+\alpha)^2+\beta^2}$
$c_{12} \, [Ns/m]$	$-\frac{\omega_x}{8mA_gQ_y} \frac{(s+\alpha)}{(s+\alpha)^2+\beta^2}$
$F_{Ys} \; [N]$	$\frac{1}{8mA_gX_0Q_y} \frac{(s+\alpha)}{(s+\alpha)^2+\beta^2}$
$F_{Yc} \; [N]$	$\frac{1}{8mA_gX_0Q_y} \frac{\beta}{(s+\alpha)^2+\beta^2}$

3 Numerical Study

The objective of this section is to compare the full and the approximate models via their dynamical and frequency responses for the signals $Y_s(t)$, $Y_c(t)$ (equivalently the measured angular velocity), to inputs in Ω, F_{Ys}, F_{Yc}, k_{12}, and c_{12}. The following performance measures are used:

$$e_C = \max_{[0,T]}(|Y_c^F(t) - Y_c(t)|) \tag{30}$$

$$e_S = \max_{[0,T]}(|Y_s^F(t) - Y_s(t)|) \tag{31}$$

$$e_\Omega = \max_{[0,T]}(|\hat{\Omega}^F(t) - \hat{\Omega}(t)|) \tag{32}$$

The superscript F denotes variables that are computed via the full model equations, $\hat{\Omega}$ denotes the measured angular velocity, and T is the simulation time, which is larger than the characteristic times of the simulated signals.

Full Model Simulation. The computation of the angular velocity from the sense signal, $y(t)$, requires computing the in-phase and quadrature amplitudes, Y_c^F and Y_s^F. These amplitudes are computed following standard steps of demodulation and low-pass filtering [4], which are detailed next for the sake of clarity.

$$\ddot{y} + \frac{\omega_y}{Q_y}\dot{y} + \omega_y^2 y = \frac{1}{m}\left[F_{Ys} + (c_{12} + 4A_g m\,\Omega)X_0\,\omega_x\right]\sin(\omega_x t) + \tag{33}$$

$$+ \frac{1}{m}(F_{Yc} - k_{12}X_0)\cos(\omega_x t) \tag{34}$$

$$z_C(t) = y(t)\cos(\omega_x t) \tag{35}$$

$$z_S(t) = y(t)\sin(\omega_x t) \tag{36}$$

$$\ddot{Y}_c^F + 2\zeta_l\omega_l\dot{Y}_c^F + \omega_l^2 Y_c^F = A_l\,\omega_l^2\,z_C \tag{37}$$

$$\ddot{Y}_s^F + 2\zeta_l\omega_l\dot{Y}_s^F + \omega_l^2 Y_s^F = A_l\,\omega_l^2\,z_S \tag{38}$$

where Eq. (34) results from Eq. (6) inserted in Eq. (1). The purpose of the demodulation steps is to extract the in-phase and quadrature amplitudes from the sense signal $y(t)$. Equation (37) provides the output of the in-phase loop, while Eq. (38) provides the output of the quadrature loop. The second-order low-pass filter parameters, ω_l, ζ_l, and A_l, are chosen in order to filter out the high frequencies [i.e. $(\omega_x + \omega_y)$ and $(2\,\omega_x)$] components that arise from the demodulation steps, Eqs. (35)(36), to maintain a cut-off frequency that is high enough (not to interfere with the slow dynamics components), and to provide a zero *dB* Bode gain. The computation of the angular velocity is then performed *as if we were in the ideal case*, i.e. $c_{12} = k_{12} = \omega_y - \omega_x = 0$, by applying the same scaling gains to $Y_c^F(t)$ and $F_{Ys}^F(t)$, as is done in Eqs. (27)) and Eq. (29) for the open-loop and closed-loop modes respectively. Therefore, in open-loop:

$$\hat{\Omega}^F(t) = -\frac{\omega_x}{4A_g Q_y X_0} Y_c^F(t) \tag{39}$$

and in closed-loop,

$$\hat{\Omega}^F(t) = \frac{1}{4A_g \omega_x m X_0} F_{Ys}^F(t) \tag{40}$$

The numerical values for the simulation parameters appear in Table 2.

Table 2. Simulation parameters values

m [kg]	A_g [-]	k_{11} [N/m]	k_{22} [N/m]	k_{12} [N/m]	c_{11}, c_{22} [Ns/m]	c_{12} [Ns/m]	X_0 [m]	ω_l [rad/sec]	ζ_l [-]	A_l [-]
$5 \cdot 10^{-9}$	0.4	177.65	177.67	$4 \cdot 10^{-7}$	$2 \cdot 10^{-7}$	$2 \cdot 10^{-12}$	10^{-5}	6280	0.707	2

3.1 Open-Loop Response

Step Response. Several step responses were simulated for various values of the input angular velocity, Ω_0, in the ideal case where $c_{12} = k_{12} = \omega_y - \omega_x = 0$. The results are summarized in Table 3 and Fig. 1. Table 3 shows that the adequation of the in-phase and quadrature signals in the full and approximate models is excellent. Figure 1 depicts a typical response to a step of 1 $[deg/sec]$ in the input angular velocity. Both the responses of the full and of the approximate models are fitting very well, up to an error due to remaining high frequency oscillations in the full model variables that are not totally filtered out by the second-order low pass filter. The lack of coupling between the in-phase and quadrature signals yields the (expected) exponential dynamics shown on Fig. 1. This is a clear illustration of the "slow envelope approximation" mentioned earlier. For this particular choice of parameters, the resulting steady state error is of the order of 0.1 $[deg/hr]$.

Table 3. Deviations between Full and Approximate models for responses to steps in the angular velocity. Ideal case. $c_{12} = k_{12} = \omega_x - \omega_y = 0$.

Ω_0 $[deg/sec]$	e_C $[m]$	e_S $[m]$	e_Ω $[rad/sec]$
10	3×10^{-12}	2×10^{-12}	6×10^{-6}
3	2×10^{-12}	1×10^{-12}	6×10^{-6}
0	2×10^{-10}	1×10^{-12}	6×10^{-6}
-3	2×10^{-10}	1×10^{-12}	6×10^{-6}
-10	3×10^{-10}	2×10^{-12}	6×10^{-6}

Frequency Response. The frequency response of the gyroscope in open-loop was investigated for each transfer function between the five inputs Ω, k_{12}, c_{12}, F_{Ys}, and F_{Yc}, and the two outputs $\hat{\Omega}$ and Y_s. Figure 2 shows the Bode plots of the magnitude and phase introduced by the full model and the approximate model between the input angular velocity and the measured angular velocity for the ideal case $\alpha = 23$ and $\beta = 0$. There is a very good fit between the curves over all frequencies, except for the phase in the higher frequencies domain. There seems to be an additional lag around $1000\,[rad/sec]$, which might stem from the demodulator low-pass filter, whose crossover frequency is $6280\,[rad/sec]$. Using the approximate model, the transfer function from the input to the measured angular velocity can easily be determined as the following function:

$$H_{\Omega}^{OL}(s) = \frac{\hat{\Omega}(s)}{\Omega(s)} = \frac{1}{1 + \frac{s}{\alpha}} \tag{41}$$

which nicely explains the first-order shape of the frequency response of the full model. The next Bode plots correspond to the case of a coupled system where $\beta = 10\,[rad/sec]$. Figures 3 and 4 depict the plots of the transfer functions from Ω to $\hat{\Omega}$ and Y_s. Here also, the approximate model is fitting the full model plots, except for the phase in the high frequencies. We notice, on the magnitude plot in Fig. 3, that the low frequency gain differs from 1 because of the discrepancy between ω_x and ω_y. Figures 5 and 6 depict the Bode plots of the transfer functions from k_{12}. Figures 7 and 8 depict the Bode plots of the transfer functions from c_{12}. Figures 9 and 10 depict the Bode plots of the transfer functions from F_{Ys}. The Bode plots of the transfer functions related to F_{Yc} were omitted for the sake of brevity. They are very similar to the plots of the input F_{Ys}. As a concluding remark, the open-loop transfer functions of the open-loop system are in a very good adequation with the full model frequency response, as testified via numerical simulations.

4 Insight on Non-ideal Cases, Open-Loop Issues, and Improvement Directions

4.1 Bias

As opposed to the stiffness coefficient k_{12}, which can be canceled to a large extent by proper feedback control in the quadrature loop, the damping coefficient c_{12} is traditionally not well known. Assume for simplicity that the angular velocity is zero and that the gyroscope works in closed-loop. If it works successfully, then the vibrations are canceled, i.e., $Y_s = Y_c = 0$, and the approximate model provides the following relationship between the control force F_{Ys} and c_{12}:

$$F_{Ys} = -c_{12}\,\omega_x X_0 \tag{42}$$

Using Eq. (42) in Eq. (29) yields a non-zero measurement value for the angular velocity and, thus, a bias:

$$\Omega_{bias} = \frac{c_{12}}{4\,m\,A_g} \tag{43}$$

This result, originally presented in [6, p.117], indicates the relation between the gyroscope bias and the unbalance in the structural damping. This term may vary with the temperature. As a result, the bias value might vary during the gyro operation time. Simulations show that the performance measures e_C, e_S, e_Ω hardly changed, as compared with the ideal case (see Table 3), meaning that the approximate and the full model remain very close. Hence, the approximate model can easily demonstrate the effect of c_{12} on the gyro bias, while such an analysis would not be as straightforward using the full model.

4.2 Bandwidth

The approximate model for the in-phase amplitude, $Y_c(t)$, is rewritten here for convenience, assuming no frequency mismatch ($\beta = 0$), no damping unbalance ($c_{12} = 0$), and no feedback forces. In this special case originally presented in [6, p.114] the resulting dynamical system is

$$\dot{Y}_c + \alpha Y_c = -2\, A_g\, X_0\, \Omega \tag{44}$$

The resulting dynamical system is a 1st order low-pass filter with the following bandwidth:

$$BW_{OL} = \frac{\omega_{_y}}{2Q_y} \tag{45}$$

Typical values for the drive frequency $\omega_{_x}$ and the quality factor vary around $2 \times 10^5\, [rad/sec]$ and 4×10^4, respectively. Hence, a typical bandwidth in open-loop is around the value of $2\, [rad/sec]$. The open-loop dynamics is thus expected to have a narrow band. The associated long settling time results in unpractical gyro output time response characteristics during quick dynamics. In order to augment the gyroscope bandwidth a closed loop architecture with a proper feedback control design is usually exploited.

4.3 Importance of the Quadrature Loop Control

When there is no frequency mismatch ($\omega_{_x} = \omega_{_y}$), there is no coupling between the in-phase and the quadrature signals dynamics, the latter is thus not excited, and there is no need to cancel it. But when the coupling exists, Y_c will not only react to the "correct" signal in Ω, but also to the disturbance

$$\beta\, Y_s = \frac{(\omega_{_y}^2 - \omega_{_x}^2)}{2\,\omega_{_x}}\, Y_s \;\simeq (\omega_{_y} - \omega_{_x}) Y_s \tag{46}$$

For even a small frequency mismatch of a few Hertz, the contribution of the above coupling term can be stronger than that of Y_c on itself. Excited by the permanent input due to X_0, the vibration Y_s will enter, via the coupling term, the dynamics of Y_c and act as a perturbation. It is thus essential to properly address the quadrature effect. Controlling the quadrature term might be achieved by either canceling the value of Y_s or by creating a decoupling between the dynamics of Y_s and Y_c.

4.4 Closed-Loop Design and Response to Noisy Inputs

In this section we show how the approximate model can be conveniently applied to investigate the impact of the disturbances and noises on the response. Let $m_s(t)$ and $m_c(t)$ denote input perturbations that are added to the control forces $F_{Ys}(t)$ and $F_{Yc}(t)$, respectively. They are essentially mechanical-thermal noises, modeled as white noise. Let $n_s(t)$ and $n_c(t)$ denote output noises, which are added to the signals coming from the in-phase and from the quadrature loops. They essentially consist of electronic noise from measurement electronics and are modeled as white noise. The approximate model is re-written here for convenience:

$$\dot{Y}_s + \alpha Y_s + \beta Y_c = -\gamma k_{12} X_0 + \gamma F_{Yc} + m_s(t) \tag{47}$$

$$\dot{Y}_c + \alpha Y_c - \beta Y_s = -\left[\frac{c_{12}}{2m} + 2A_g \Omega(t)\right] X_0 - \gamma F_{Ys} + m_c(t) \tag{48}$$

$$Y_{s,m}(t) = Y_s(t) + n_s(t) \tag{49}$$

$$Y_{c,m}(t) = Y_c(t) + n_c(t) \tag{50}$$

Since the influence of the mechanical-thermal noise has been studied in previous works [1], we focus our discussion on the investigation of the measurement electronic noise impact. A conventional feedback loop architecture consists in closing separately the loops for the in-phase and for the quadrature signals. Thus, the dynamics of Y_s and Y_c will be controlled via the input signals F_{Yc} and F_{Ys}, respectively. The transfer functions from the noises $n_s(t)$ and $n_c(t)$ to the gyro output, i.e. the measured angular velocity, $\hat{\Omega}$ will be investigated next. Assume for simplicity that proportional feedbacks are applied in both loops, with gains K_s and K_c, i.e.

$$F_{Ys} = K_c 2\,\omega_x m Y_{c,m} = K_c 2\,\omega_x m(Y_c + n_c) \tag{51}$$

$$F_{Yc} = K_s 2\,\omega_x m Y_{s,m} = K_s 2\,\omega_x m(Y_s + n_s) \tag{52}$$

and assume that the thermal-mechanical noises are zero. Then the measured angular velocity is expressed as follows:

$$\hat{\Omega}(s) = \frac{-K_c}{2A_g X_0} \frac{(s+\alpha+K_s)(s+\alpha)+\beta^2}{(s+\alpha+K_s)(s+\alpha+K_c)+\beta^2} N_c(s) \tag{53}$$

$$+ \frac{K_s K_c \beta}{2A_g X_0} \frac{1}{(s+\alpha+K_s)(s+\alpha+K_c)+\beta^2} N_s(s) \tag{54}$$

$$+ \frac{(s+\alpha+K_s)K_c}{(s+\alpha+K_s)(s+\alpha+K_c)+\beta^2} \Omega(s) \tag{55}$$

In order to understand the impact of the input noises, a numerical investigation of the corresponding frequency response gains is presented next. The following values were chosen: $\alpha = 3$, $\beta = 10$ (corresponding to a frequency mismatch

of $10\,[rad/sec]$). The feedback gain for the in-phase loop, K_c, is chosen equal to 628. This ensures that the closed-loop bandwidth increase to approximately $628\,[rad/sec]$. The following values of K_s were investigated: $0, 1, 10, 100, 1000$. The noises $n_c(t)$ and $n_s(t)$ are assumed white, identically distributed, uncorrelated one another. Figure 11 shows the gyroscope closed-loop Bode plots, i.e., it shows the transfer function

$$\frac{\hat{\Omega}(s)}{\Omega(s)} = 628\frac{(s+3+K_s)}{(s+3+K_s)(s+631)+100} \tag{56}$$

Figure 11 shows that the Bode gain is positively impacted by the increase in the quadrature control gain. The higher is the value of K_s, the closer is the Bode gain to $1\,(0\,dB)$. Figure 12 depicts several Bode plots of the transfer function from $N_c(s)$ to $\hat{\Omega}(s)$, which is expressed as

$$\left(-\frac{1}{7.5}10^{-7}\right)\frac{\hat{\Omega}(s)}{N_c(s)} = 628\frac{(s+3+K_s)(s+3)+100}{(s+3+K_s)(s+631)+100} \tag{57}$$

for each K_s. For high values of K_s, the behavior is satisfactory with relatively low values of the Bode gain. On the other hand, lower values in the gain K_s yield high values in the Bode gain at low frequencies, resulting in a poor gyroscope drift performance. Notice that this phenomenon was observed in [10], together with the idea that the level of the low frequencies gain is related to the difference between ω_x and ω_y. This phenomenon clearly emphasizes the need to design efficient quadrature loop control. In addition, Fig. 11 illustrates the known phenomenon of noise amplification due to feedback. The magnitude Bode plot features a lead compensation effect above the open-loop crossover frequency and until the closed-loop crossover frequency. When applied to the measurement white noise, after being integrated, this element re-creates a white noise, which becomes a gyro measurement error. Above the closed-loop crossover frequency, it is common practice to add a low-pass filter in order to avoid unnecessary noise amplification. Figure 13 presents the same type of plots but for the quadrature component of the measurement error, $n_s(t)$. Its transfer function is

$$\left(-\frac{1}{7.5}10^{-8}\right)\frac{\hat{\Omega}(s)}{N_s(s)} = 628\frac{K_s}{(s+3+K_s)(s+631)+100} \tag{58}$$

Figure 13 illustrates the limitation attached to the increase of the control gain K_s: the measured angular velocity will include more high frequencies (noise) since an increase of K_s increases the bandwidth. Figure 14 presents the Bode plots for the equivalent total transfer function from the measurement noises, $n_c(t)$ and $n_s(t)$. Since these noises are assumed independent and identically distributed, the resulting Bode plot was obtained as the root-sum-of-squares of the two previous Bode plots.

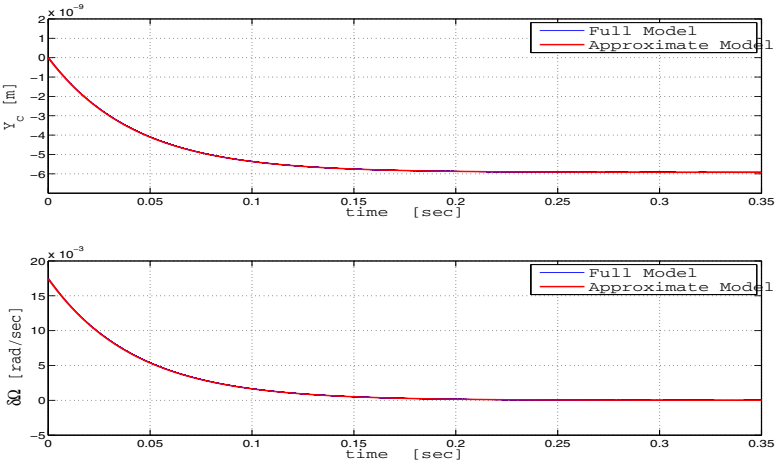


Fig. 1. Comparison of the Full and Approximate models. Time histories of the in-phase amplitudes, Y_c, and of the angular velocity measurement errors. Ideal case: $c_{12} = k_{12} = \omega_x - \omega_y = 0$. The signals present the same transient, stay very close to each other, and the measurement errors converge to small values.

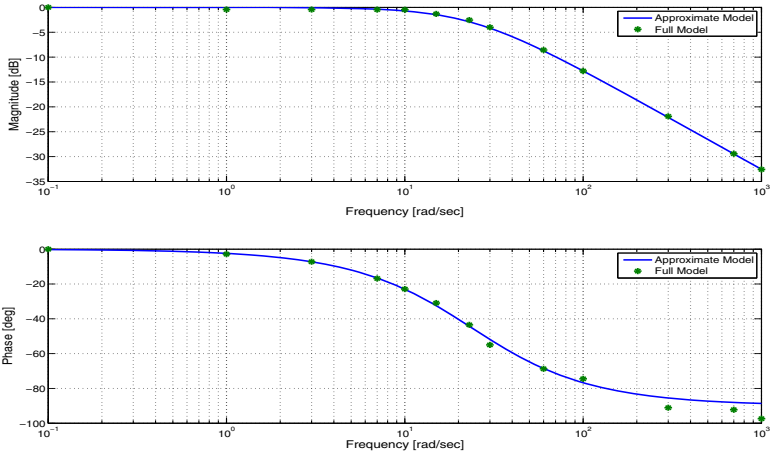


Fig. 2. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the input angular velocity Ω to the measured angular velocity $\hat{\Omega}$. Ideal case: $c_{12} = k_{12} = \omega_x - \omega_y = 0$.

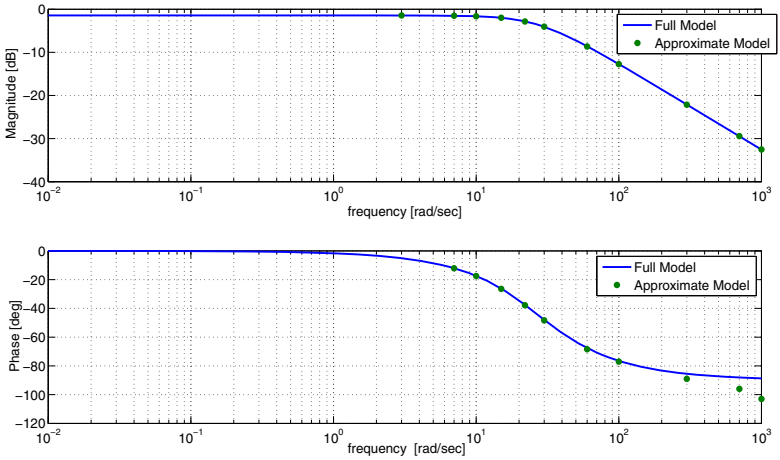


Fig. 3. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the input angular velocity Ω to the measured angular velocity $\hat{\Omega}$. $\omega_x - \omega_y = 10\,[rad/sec]$.

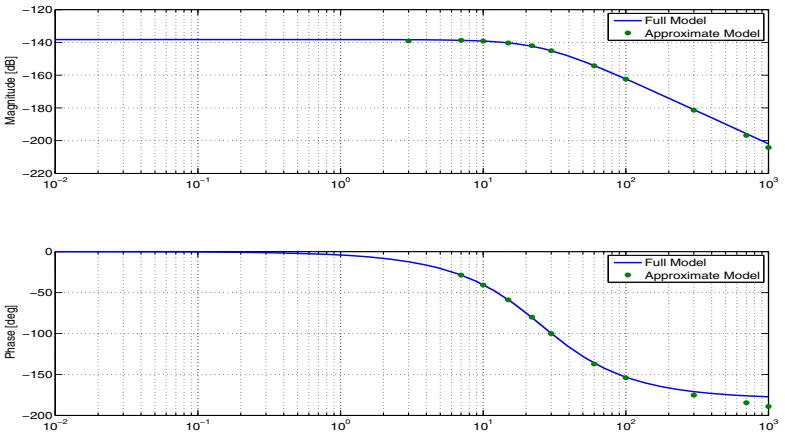


Fig. 4. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the input angular velocity Ω to the quadrature amplitude Y_s. $\omega_x - \omega_y = 10\,[rad/sec]$.

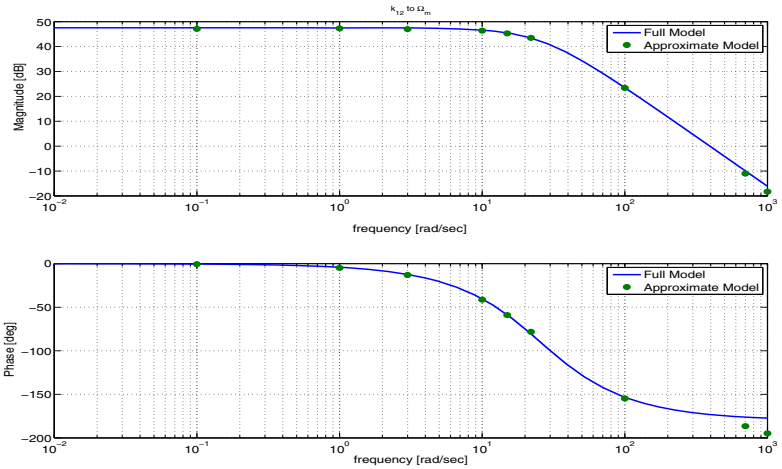


Fig. 5. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the parameter k_{12} to the measured angular velocity $\hat{\Omega}$. $\omega_x - \omega_y = 10\,[rad/sec]$.

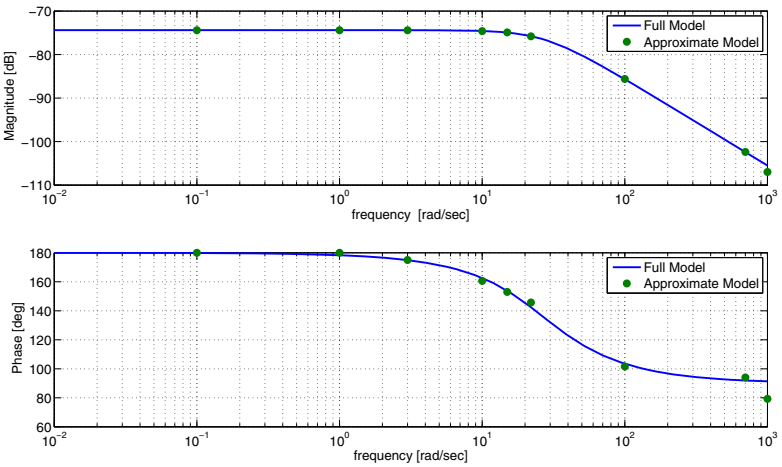


Fig. 6. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the parameter k_{12} to the quadrature amplitude Y_s. $\omega_x - \omega_y = 10\,[rad/sec]$.

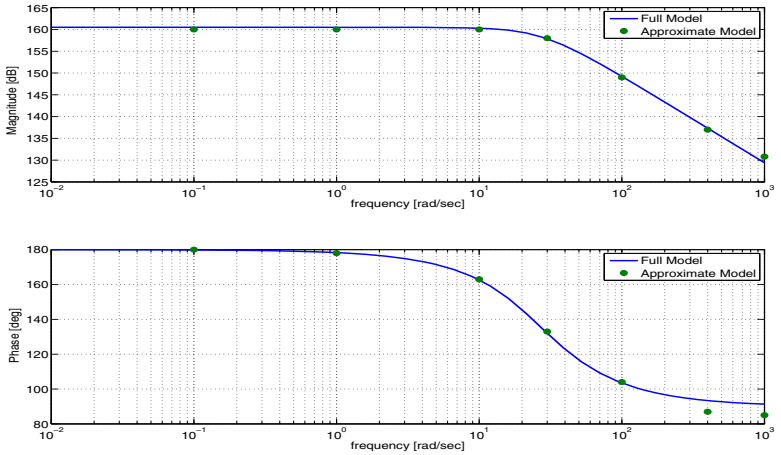


Fig. 7. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the parameter c_{12} to the measured angular velocity $\hat{\Omega}$. $\omega_x - \omega_y = 10\,[rad/sec]$.

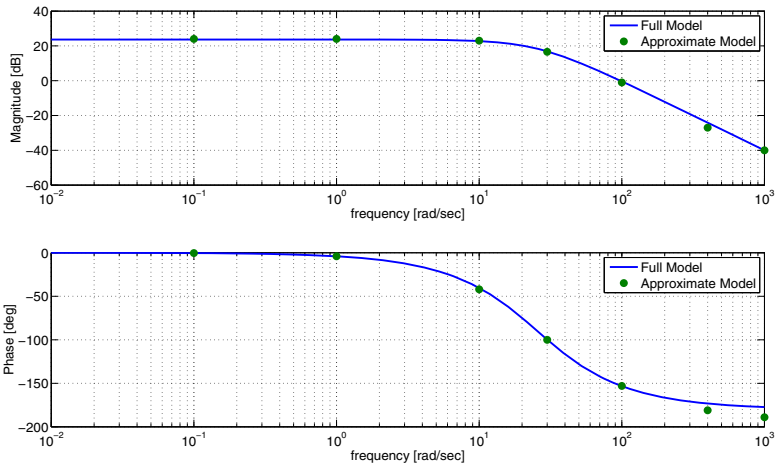


Fig. 8. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the parameter c_{12} to the quadrature amplitude Y_s. $\omega_x - \omega_y = 10\,[rad/sec]$.

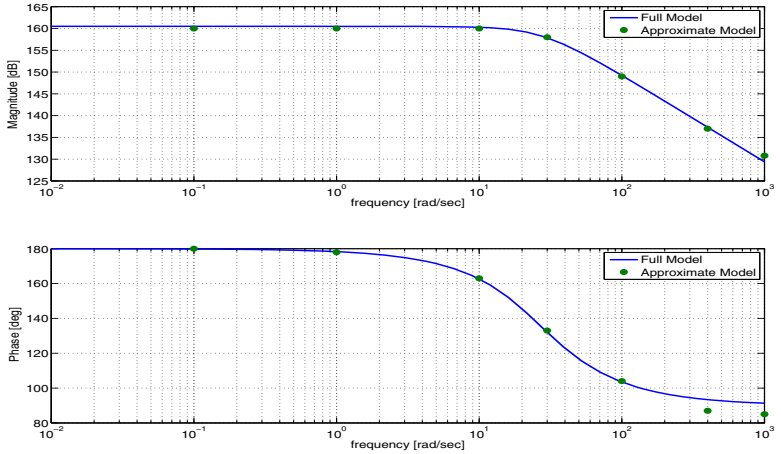


Fig. 9. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the force quadrature amplitude F_{Ys} to the measured angular velocity $\hat{\Omega}$. $\omega_x - \omega_y = 10\,[rad/sec]$.

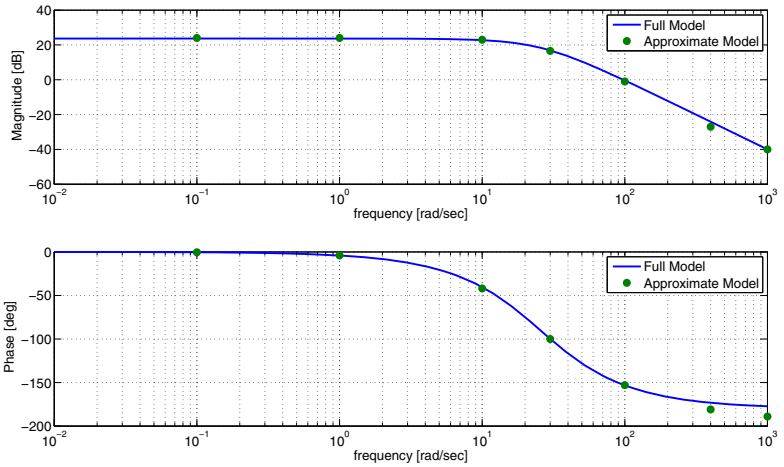


Fig. 10. Comparison of the full and approximate models. Bode plots of the open-loop transfer function from the force quadrature amplitude F_{Ys} to the quadrature amplitude Y_s. $\omega_x - \omega_y = 10\,[rad/sec]$.

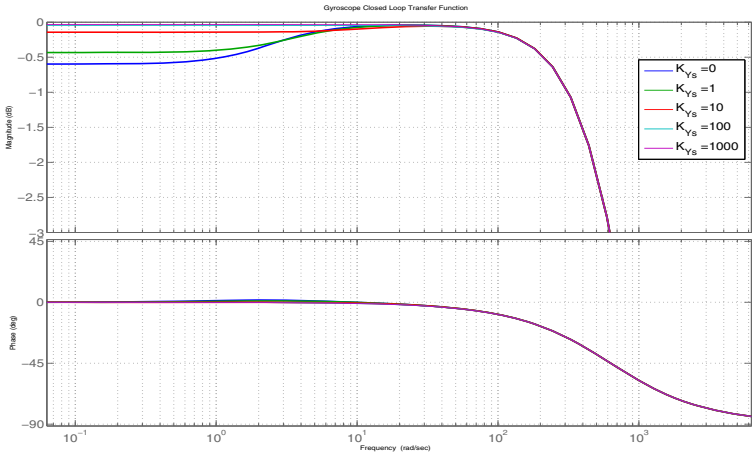


Fig. 11. Bode plots of the closed-loop transfer function from the input angular velocity to the measured angular velocity. Performances comparison of the approximate model for various values of the quadrature channel control gain.

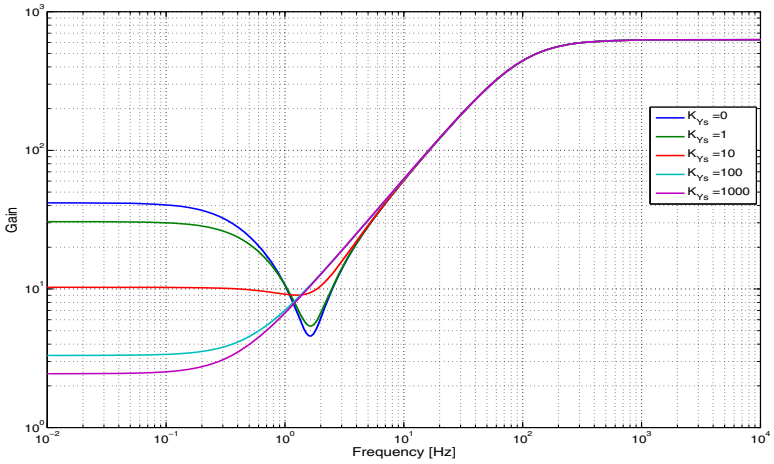


Fig. 12. Bode plots of the closed-loop transfer function from the in-phase measurement noise to the measured angular velocity. Performances comparison of the approximate model for various values of the quadrature channel control gain.

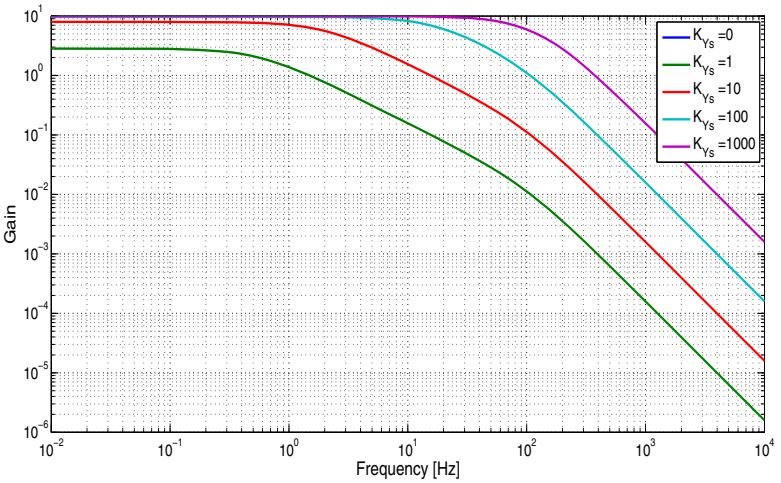


Fig. 13. Bode plots of the closed-loop transfer function from the quadrature measurement noise to the measured angular velocity. Performances comparison of the approximate model for various values of the quadrature channel control gain.

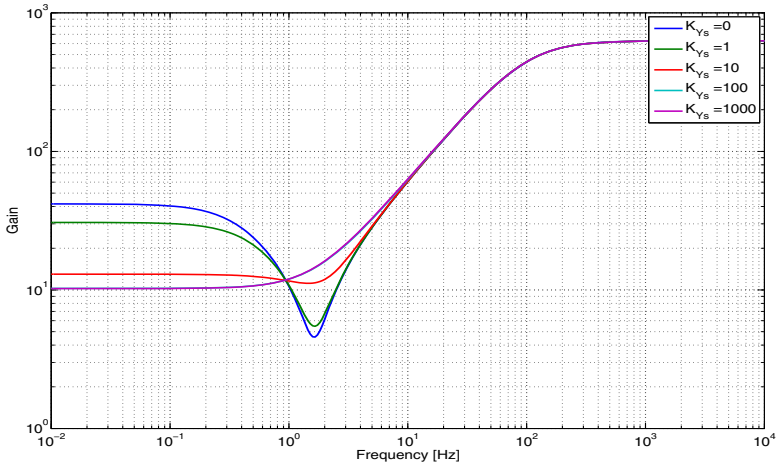


Fig. 14. Bode plots of the closed-loop transfer function for the total (in-phase and quadrature) measurement noise to the measured angular velocity. Performances comparison of the approximate model for various values of the quadrature channel control gain.

5 Conclusion

This work presented an approximate dynamic model for vibratory gyros. The model has the merits of simplicity and accuracy. The simplicity allows a relatively easy analysis of imperfection effects in the gyro measurement output. It provides a clear insight on fundamental known phenomenons like the presence of a drift, which originates in the combined action of structural damping and unmatched frequencies between the drive and the sense channels. The accuracy of the model was thoroughly illustrated vis simulations of the frequency responses of the full and the approximate models. It is suggested that this model will prove useful in the development of a control strategy for gyro performance improvement, in particular for drift mitigation. A future work will tackle the problem of control synthesis for this type of systems, including a sensitivity analysis with respect to the measurement noise.

References

1. Yazdi, N., Ayazi, F., Najafi, K.: Micromachined Inertial Sensors. Proceedings of the IEEE 86(8), 1640–1659 (1998)
2. Ayazi, F., Najafi, K.: A HARPSS Polysilicon Vibrating Ring Gyroscope. Journal of Microelectromechanical Systems 10(2), 169–179 (2001)
3. Brady, T., Buckley, S., Dennehy, C.J., Gambino, J., Maynard, A.: The Intertial Stellar Compass: a Multifunction, Low Power, Attitude Determination Technology Breakthrough. In: AAS-2003-003, 26th AAS Guidance and Control Conference Advances in the Astronautical Sciences, Guidance and Control 2003, Breckenridge, CO, February 5–9, vol. 113, pp. 39–56 (2003)
4. Leland, R.P.: Mechanical Thermal Noise in MEMS Gyroscopes. IEEE Sensors Journal 5(3), 493–500 (2005)
5. Roberts, J.B., Spanos, P.D.: Stochastic Averaging: an approximate method of solving random vibration problems. Int. J. Non-linear Mech. 21(2), 111–134 (2007)
6. Putty, M.W.: A Micromachined Vibrating Ring Gyroscope, Ph.D. Dissertation, Faculty of Electrical Engineering, University of Michigan, Ann Arbor, MI (1995)
7. Gallacher, B.J., Burdess, J.S., Harish, K.M.: A Novel Excitation Scheme for MEMS Gyroscopes Using Parametric Pumping for Near Inertial Grade Performance. Proc. of SPIE 6113, 611305 (2006)
8. Zheng, Q., Dong, L., Gao, Z.: Control and Rotation Rate Estimation of Vibrational MEMS Gyroscopes. In: IEEE International Conference on Control Applications, October 1-3 (2007)
9. Izadian, A., Dawson, J., Famouri, P.: Input-Output Synchronization for Bias Drift Reduction oF MEMS Gyroscopes. In: American Control Conference, June 11-13, pp. 3751–3754 (2008)
10. Gorman, J., Shapiro, B. (eds.): Feedback Control of MEMS to Atoms, Chapter 8 - Dissecting Tuned MEMS Vibratory Gyros, Kim,D., M'Closkey, R.T., pp. 211–267. Springer (2012)

Part III
Spacecraft Guidance, Navigation and Control

Zero Δv Solution to the Angles-Only Range Observability Problem during Orbital Proximity Operations

Itzik Klein[1,⋆] and David K. Geller[2,⋆⋆]

[1] Rafael, P.O., Box 2250, Haifa 31021
itzikkl@rafael.co.il
[2] Utah State University, Logan, Utah, 84322, USA
david.geller@usu.edu

Abstract. During orbital proximity operations, research has shown that angles-only navigation during coasting flight suffers from a lack of range observability. To circumvent this deficiency, previous research has required a prior information on the target geometry or the implementation of special translational maneuvers. This paper shows that the range observability problem during coasting flight can be solved by properly including the offset of the camera from the vehicle center-of-mass in the problem formulation, and by applying appropriate vehicle rotations. Range observability without translational maneuvers (zero Δv) or a priori knowledge of the target geometry is clearly demonstrated using a pseudo 6 degree-of-freedom simulation. Results for v-bar station-keeping, flyby orbits, and circumnavigation (football) orbits are presented.

Keywords: Angle Only Navigation.

1 Introduction

Many past space programs have required relative navigation for rendezvous and proximity operations including Gemini, Apollo, Soyuz and the Space Shuttle[13], [20], [26]. In all of these programs, radar systems were used as the primary relative navigation device. Soyuz used multiple radar systems to determine range, range rate and relative attitude [4]. The Space Shuttle used radar as well as a laser ranging device and a centerline camera for close-in proximity operations [14]. In all cases, these radar-based systems are too heavy and power intensive for smaller spacecraft. Recently, a growing interest in autonomous rendezvous and docking[10], [25],[9],[7],[8],[24],[15] has produced a series of experimental spacecraft in an attempt to develop rendezvous and proximity operations technology that would be more appropriate for smaller unmanned spacecraft [6].

⋆ Dr., Research Fellow, Rafael, itzikkl@rafael.co.il
⋆⋆ Visiting Professor, Technion's Department of Aerospace Engineering.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_19

The eXperimental Satellite System -11 (XSS-11) was a micro-satellite used to successfully demonstrate rendezvous and proximity operations relying primarily on a Light Detection and Ranging (LIDAR) sensor for relative position measurements [1].

The Demonstration of Autonomous Rendezvous Technology (DART) spacecraft [17] was equipped with a camera and a laser to illuminate reflectors placed on the target. Unfortunately software errors caused DART to collide with the target satellite before it could accomplish its mission objectives.

Another vehicle, Orbital Express [22], was equipped with three cameras, the same laser based sensor used on DART, and a laser range finder. Orbital Express was able to successfully complete several proximity and docking operations [3].

A more recent example includes the European Space Agency (ESA) Automated Transfer Vehicle (ATV) [16]. The ATV program is designed to perform automated phasing, approach, rendezvous and docking to the ISS, followed by departure and deorbit maneuverings [11]. Recently, three ATVs successfully completed their missions and several more ATVs are planned to be launched in the near future.

Using a camera is one of the simplest and useful sensors to perform relative navigation. It is a simple low-cost, low power, and low mass/volume solution to the relative orbital navigation problem. Such an approach is known as angles-only relative navigation for rendezvous and proximity operations.

Angles-only navigation relies on the ability of a sensor to measure the azimuth and elevation angles to an object of interest relative to the sensor location. By measuring these angles under the right circumstances, the relative position and velocity of the object may be estimated [21,18]. However, angles-only navigation generally lacks the ability to determine the range between the sensor and the object of interest. This drawback has been well demonstrated [23]. By measuring the azimuth and elevation angles to the target, the LOS from the camera to the target satellite can be determined, but it is generally not possible to determine where the chaser is located along the LOS, even when accounting for the dynamics of the problem.

In the recent past, two approaches have been suggested to overcome this range observability problem. In the first approach, a nominal or special translational maneuver is executed to help determine the unknown range parameter [19,5]. Such translational maneuvers however require additional propellant and increase satellite Δv requirements. In the second approach, knowledge of target spacecraft size, shape, or location of known features is used to determine the unknown range parameter. This however requires a priori information about the target spacecraft [24].

In this paper, a solution to the range observability problem that does not require translational maneuvers nor a priori knowledge of target spacecraft geometry is demonstrated. The proposed solution includes small vehicle rotational maneuvers and a camera center-of-mass offset in the formulation of the estimation problem. To date, all angles-only navigation results have assumed that the camera is either located at the vehicle center-of-mass or that the chaser/target

separation distances are sufficiently large that the center-of-mass to camera offset is of no significance. This paper will show that by properly including the camera offset in the problem formulation, the range observability problem is solved with small vehicle rotations.

In [19] it is shown how range observability is achieved in terrestrial hiking applications. Hikers obtain a bearing measurement to a distant landmark and draw a line on a map emanating from the landmark representing all possible hiker positions. When a second landmark is available, a second line is drawn, and the hiker location is determined by the intersection of the two lines. In the absence of a second landmark (as is the case in rendezvous and proximity operations), the hikers can use a pedometer or knowledge of the length of their stride to walk for a known distance normal to the line-of-sight of the first landmark. By taking a second bearing measurement, only one position will satisfy the known distance they walked. In this way, the hiker can estimate position, using a single landmark.

This hiker analogy is also valid for rotational maneuvers where the distance traveled by the camera between bearing measurements is determined by the camera offset and a known vehicle rotational maneuver. In this way, it will be shown that when the vehicle separation distances are small (e.g., < 100 m), very accurate estimates of the range can be achieved. Results of pseudo 6-degree-of-freedom simulations are presented for v-bar station-keeping, flyby orbits, and circumnavigation orbits, and the sensitivity of the navigation error to key problem parameters is determined.

2 Problem Formulation

2.1 The Hill Clohessy Wiltshire Equations

The Hill Clohessy Wiltshire (HCW) equations [2] represent the chaser satellite dynamics relative to a target satellite in a near-circular orbit:

$$\begin{aligned} \ddot{x} &= -2\omega\dot{z} + a_x \\ \ddot{y} &= -\omega^2 y + a_y \\ \ddot{z} &= 3\omega^2 z + 2\omega\dot{x} + a_z \end{aligned} \tag{1}$$

where $[\, a_x \; a_y \; a_z \,]^T$ is the chaser thrust vector, and ω is the target orbital angular rate given by

$$\omega = \sqrt{\frac{\mu}{a^3}} \tag{2}$$

$\mu = 398601[km^3/s^2]$ is the Earth gravitational parameter, and a is the semi-major axis. The coordinate system associated with the HCW equations is the Local Vertical Local Horizontal (LVLH) coordinate frame depicted in Figure 1. The origin is located at the center-of-mass of the target satellite, the x-axis is the downrange or local horizontal direction, the y-axis is the cross-track direction, and the z-axis is the altitude or local vertical direction.

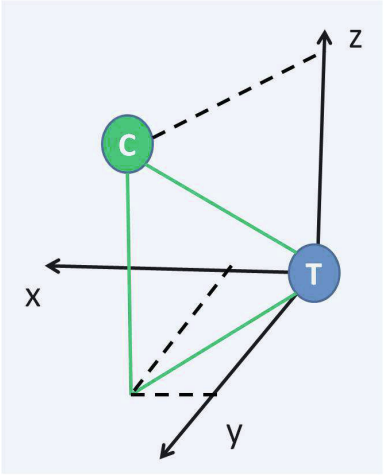


Fig. 1. LVLH coordinate frame

The HCW equations (1) can also be written in state-space form where the state vector $X = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}]^T$ consist of the relative position and velocity vectors. Thus, the state-space model is

$$\dot{X} = AX + Bu \tag{3}$$

where

$$A = \begin{bmatrix} 0_3 & I_3 \\ A_{21} & A_{22} \end{bmatrix}, A_{21} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -\omega^2 & 0 \\ 0 & 0 & 3\omega^2 \end{bmatrix} A_{22} = \begin{bmatrix} 0 & 0 & -2\omega \\ 0 & 0 & 0 \\ 2\omega & 0 & 0 \end{bmatrix} \tag{4}$$

$$B = \begin{bmatrix} 0_3 \\ I_3 \end{bmatrix} u = \begin{bmatrix} a_x\ a_y\ a_z \end{bmatrix}^T \tag{5}$$

In the absence of thrust, the HCW equations have an analytic closed form solution [23]

$$X = \Phi X_0 \tag{6}$$

where Φ is the associated state transition matrix and X_0 is the initial conditions vector.

2.2 Angles-Only Navigation

For angles-only navigation problems, the relative azimuth α and elevation ϵ angles are commonly modeled by assuming that the camera is mounted at the chaser center-of-mass as shown in Figure 2. In this case, α and ϵ are defined by

$$tan\,(\alpha) = \frac{y}{x}\, tan\,(\epsilon) = \frac{z}{\sqrt{y^2 + x^2}} \tag{7}$$

where x, y, and z are the components of the relative position between the target center-of-mass and the chaser satellite center-of-mass in the LVLH frame as given in Eq. (1).

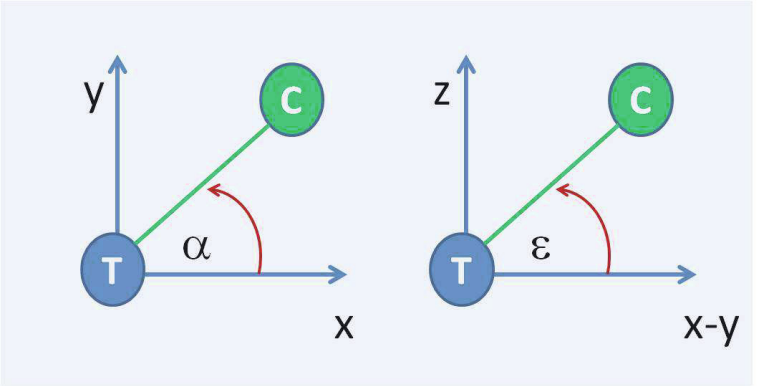


Fig. 2. Azimuth and elevation angles in LVLH coordinate frame

3 The Range Observability Problem

3.1 Range Observability without a Camera Offset

When the camera is assumed to be mounted at the the chaser center-of-mass, the azimuth and elevation angles defined in Figure 2 can be recast in the form of a line-of-sight (LOS) unit vector measurement i_r,

$$i_r = \begin{bmatrix} cos\,(\epsilon)\, cos\,(\alpha) \\ cos\,(\epsilon)\, sin\,(\alpha) \\ sin\,(\epsilon) \end{bmatrix} \tag{8}$$

Similarly, the time-varying azimuth $\alpha\,(\tau)$ and elevation $\epsilon\,(\tau)$ measurements, over an observation time, will produce a LOS time-history $i_r\,(\tau)$,

$$i_r\,(\tau) = \begin{bmatrix} cos\,[\epsilon\,(\tau)]\, cos\,[\alpha\,(\tau)] \\ cos\,[\epsilon\,(\tau)]\, sin\,[\alpha\,(\tau)] \\ sin\,[\epsilon\,(\tau)] \end{bmatrix} \tag{9}$$

The time-history of the LOS unit vector is simply the relative position vector divided by the range separating the two spacecraft

$$i_r(\tau) = \frac{r\,(\tau)}{\|r\,(\tau)\|} \tag{10}$$

Since the solution to the linear HCW equations given in Eq. (6) can be written in the form

$$\begin{bmatrix} r(\tau) \\ v(\tau) \end{bmatrix} = \begin{bmatrix} \Phi_{rr}(\tau) & \Phi_{rv}(\tau) \\ \Phi_{vr}(\tau) & \Phi_{vv}(\tau) \end{bmatrix} \begin{bmatrix} r_0 \\ v_0 \end{bmatrix} \tag{11}$$

the time-history of the LOS unit vector in Eq. (10) can be written as

$$i_r(\tau) = \frac{\Phi_{rr}(\tau) r_0 + \Phi_{rv}(\tau) v_0}{\|\Phi_{rr}(\tau) r_0 + \Phi_{rv}(\tau) v_0\|} \tag{12}$$

The observability problem asks whether or not the initial position and velocity of the chaser can be uniquely determined from the time-history of the LOS unit vector, i.e. from angle measurements alone. To answer this question, Woffinden and Geller [23] show that if the initial conditions are scaled by an arbitrary value, ρ, the resulting LOS time-history, $i_{\rho r}(\tau)$, is no different than the LOS time-history $i_r(\tau)$ produced by the unscaled initial conditions

$$i_{\rho r}(\tau) = \frac{\Phi_{rr}(\tau) \rho r_0 + \Phi_{rv}(\tau) \rho v_0}{\|\Phi_{rr}(\tau) \rho r_0 + \Phi_{rv}(\tau) \rho v_0\|} = \frac{\Phi_{rr}(\tau) r_0 + \Phi_{rv}(\tau) v_0}{\|\Phi_{rr}(\tau) r_0 + \Phi_{rv}(\tau) v_0\|} = i_r(\tau) \tag{13}$$

Thus, the initial position and velocity of the chaser cannot be uniquely determined from the time-history of the LOS unit vector, and, in particular, the range to the target is unobservable.

3.2 Range Observability with a Camera Offset

The purpose of this paper is to show that the state vector, including the range, is observable when the camera is offset from the chaser center-of-mass. Thus, the key to demonstrating observability is to first properly model the position of the camera with respect to the center-of-mass of the chaser.

If the camera is located at the center-of-mass of the chaser, the position vector of the camera with respect to the target in the camera frame r_c^{Camera} is defined by

$$r_c^{Camera} = T^{Body \to Camera} T^{LVLH \to Body} r^{LVLH} \tag{14}$$

where r^{LVLH} denotes the relative position of the chaser center-of-mass with respect to the target satellite center-of-mass in the LVLH frame, $T^{LVLH \to Body}$ is the transformation matrix from the LVLH frame to the body frame, and $T^{Body \to Camera}$ is the transformation matrix from the body frame to the camera frame.

Now, let the position of the camera with respect to the chaser center-of-mass in the chaser body frame be defined as

$$d^{Body} = \begin{bmatrix} d_x & d_y & d_z \end{bmatrix}^T \tag{15}$$

With a camera offset, Eq. (14) becomes

$$r_c^{Camera} = T^{Body \to Camera} T^{LVLH \to Body} r^{LVLH} + T^{Body \to Camera} d^{Body} \tag{16}$$

Without loss of generality, it is assumed that

$$T^{Body \to Camera} = I \tag{17}$$

and Eq. (16) reduces to

$$r_c^{Camera} = r_c^{Body} = T^{LVLH \to Body} r^{LVLH} + d^{Body} \tag{18}$$

Denoting the position of the camera with respect to the target in the LVLH frame simply as r_c, we have

$$r_c = r_c^{LVLH} = r^{LVLH} + T^{Body \to LVLH} d^{Body} \tag{19}$$

The corresponding azimuth and elevation relative angles are then given by

$$tan\left(\alpha_c\right) = \frac{\left[r_c\right]_y}{\left[r_c\right]_x} \, tan\left(\epsilon_c\right) = \frac{\left[r_c\right]_z}{\sqrt{\left[r_c\right]_y^2 + \left[r_c\right]_x^2}} \tag{20}$$

Notice that when the camera offset vector is zero, the angles in Eq.(20) reduce to those without the camera offset as defined in Eq.(7).

Additionally, the time varying azimuth $\alpha_c\left(\tau\right)$ and elevation $\epsilon_c\left(\tau\right)$ measurements, over an observation time, will produce a LOS time-history $i_{r_c}\left(\tau\right)$,

$$i_{r_c}\left(\tau\right) = \begin{bmatrix} cos\left[\epsilon_c\left(\tau\right)\right] cos\left[\alpha_c\left(\tau\right)\right] \\ cos\left[\epsilon_c\left(\tau\right)\right] sin\left[\alpha_c\left(\tau\right)\right] \\ sin\left[\epsilon_c\left(\tau\right)\right] \end{bmatrix} \tag{21}$$

Since the time-history of the LOS unit vector is

$$i_{r_c}(\tau) = \frac{r_c\left(\tau\right)}{\left\|r_c\left(\tau\right)\right\|} \tag{22}$$

Eq. (11) can again be utilized to produce

$$i_{r_c}\left(\tau\right) = \frac{\Phi_{rr}\left(\tau\right) r_0 + \Phi_{rv}\left(\tau\right) v_0 + T^{Body \to LVLH} d^{Body}}{\left\|\Phi_{rr}\left(\tau\right) r_0 + \Phi_{rv}\left(\tau\right) v_0 + T^{Body \to LVLH} d^{Body}\right\|} \tag{23}$$

The aim of this paper is to show that the position/velocity state vector, including the range, is observable when the camera offset d^{Body} is non-zero, i.e., that there exists only one set of initial conditions, r_0, v_0, that will produce the observed LOS time-history, $i_{r_c}\left(\tau\right)$.

4 Extended Kalman Filter

Based on the dynamics model and measurement model in Section 2, an extended Kalman filter is developed for this problem. The dynamic model of the filter is based upon the HCW equations in Eq. (3), hence it is a linear model. However, the measurement model is based on the nonlinear angle measurements as given

in Eq.(20). The first step in the Kalman filter is the prediction of the state and its associated covariance as given by [27]:

The state and state error covariance propagation equations are

$$\hat{x}_{k+1}^{-} = \Phi \hat{x}_{k}^{+} \tag{24}$$

$$P_{k+1}^{-} = \Phi P_{k}^{+} \Phi^{T} + Q_k \tag{25}$$

where P_k is the state error covariance matrix, Φ is the state transition matrix, and Q_k is the power-spectral density of the process noise. The state and state error covariance update equations are given by

$$\hat{x}_{k+1}^{+} = \hat{x}_{k+1}^{-} + K_{k+1}\left[z_{k+1} - h(\hat{x}_{k+1}^{-})\right] \tag{26}$$

$$P_{k+1}^{+} = \left(I - K_{k+1} H_{k+1}\right) P_{k+1}^{-} \tag{27}$$

where the measurement z_{k+1} consist of the angles α_c and ϵ_c, and the estimated measurements $h(\hat{x}_{k+1}^{-})$ are given by Eq. (20).

The Kalman gain is given by

$$K_{k+1} = P_{k+1}^{-} H_{k+1}^{T} \left(H_{k+1} P_{k+1}^{-} H_{k+1}^{T} + R_{k+1}\right)^{-1} \tag{28}$$

where the superscript $(\cdot)^{-}$ represents the predicted quantity (before measurement update), the superscript $(\cdot)^{+}$ represents the updated quantity (after measurement update). The covariance of the measurement noise is denoted by R_{k+1}.

The process noise covariance matrix Q_k can be obtained by

$$Q_k = \int \Phi(\tau) B Q_a B^{T} \Phi^{T}(\tau) d\tau \tag{29}$$

A trapezoidal integration of the above equation yields [12]:

$$Q_k \approx \frac{1}{2}\left[\Phi_k B Q_a B^{T} \Phi_k^{T} + B Q_a B^{T}\right] \Delta t \tag{30}$$

where

$$Q_a = q_a I_{3\times 3} \tag{31}$$

The corresponding measurement sensitivity matrix is obtained by differentiating the relative azimuth and elevation angles given Eq. (20) with respect to the relative position vector

$$H = \begin{bmatrix} \frac{\partial \alpha}{\partial r_c} \\ \frac{\partial \epsilon}{\partial r_c} \end{bmatrix} = \begin{bmatrix} -\frac{[r_c]_y}{[r_c]_{xy}^2} & \frac{[r_c]_x}{[r_c]_{xy}^2} & 0 & 0_{3x3} \\ -\frac{[r_c]_x [r_c]_z}{\|r_c\|^2 \sqrt{[r_c]_{xy}^2}} & -\frac{[r_c]_y [r_c]_z}{\|r_c\|^2 \sqrt{[r_c]_{xy}^2}} & \frac{\sqrt{[r_c]_{xy}^2}}{\|r_c\|^2} & 0_{3x3} \end{bmatrix} \tag{32}$$

where $[r_c]_{xy}^2 = [r_c]_x^2 + [r_c]_y^2$.

The initial state error covariance, P_0, will be defined in the next section. The corresponding initial state of the true orbit is

$$X_0 = x_0^{Orbit} + \delta x_0 \tag{33}$$

where x_0^{Orbit} is the exact initial condition of the desired orbit, and δx_0 is a random initial condition with zero mean and covariance P_0.

To explore the observability of angles-only navigation, with and without camera offset, we examine the observability Gramian of the linearized system and both the variance of the estimated range error and the variance of the position error perpendicular to the LOS.

The observability Gramian is defined as[12]:

$$M = \sum_{i=1}^{N} \Phi^T(i,0) H^T(i) H(i) \Phi(i,0) \tag{34}$$

The variance of the estimated range error, i.e., the LOS position error variance is given by

$$\sigma_{los}^2 = H_{los} P H_{los}^T, \quad H_{los} = \left[r_c^T / \|r_c\| \; 0_{1x3} \right] \tag{35}$$

and the variance of the position error perpendicular to the LOS

$$P_{perp} = H_{perp} P H_{perp}^T, \quad H_{perp} = \left[I_{3\times3} - (r_c)(r_c)^T / \|r_c\|^2 \; 0_{3\times3} \right] \tag{36}$$

$$\sigma_{perp}^2 = Tr\left(P_{perp}\right) \tag{37}$$

5 Relative Orbit Types

In orbital relative motion there are natural relative trajectories that are very useful for rendezvous and inspection. In this research we employ three of them: 1) v-bar, 2) flyby and 3) football orbits.

5.1 V-bar Station-Keeping

V-bar station-keeping refers to the case where the chaser is station-keeping on the target local-horizontal in the direction of the inertial velocity vector at a position forward or behind the target. Figure 3 presents a v-var approach scenario from step 1 to 4, while v-bar station-keeping may be observed when the chaser is fixed in one of the points relative to the target

The initial position/velocity state vector for this scenario is:

$$X_0^{vbar} = \left[x_0 \; 0 \; 0 \; 0 \; 0 \; 0 \right]^T, x_0 = 1m, \; 10m \tag{38}$$

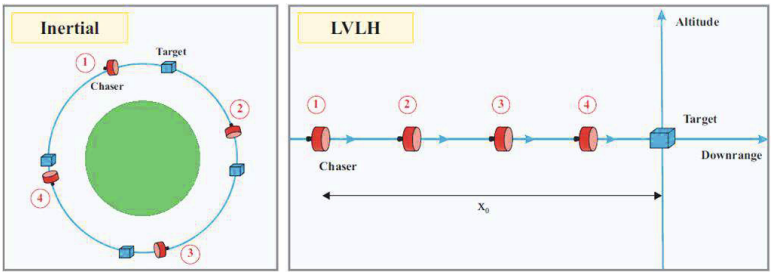


Fig. 3. V-bar approach scenario for rendezvous and docking

5.2 Flyby Orbit

In the flyby trajectory, the chaser approaches or moves away from the target along a linear trajectory above or below the target as observed in the LVLH frame. This trajectory requires the chaser to be in a co-elliptic inertial orbit with the target as shown in Figure 4. The initial state vector for this scenario is:

$$X_0^{flyby} = \left[x_0 \; 0 \; z_0 \; \tfrac{3}{2}h_0\omega \; 0 \; 0\right]^T, \; z_0 = 1m, \; or \; 10m \tag{39}$$

where x_0 is the initial required downrange position.

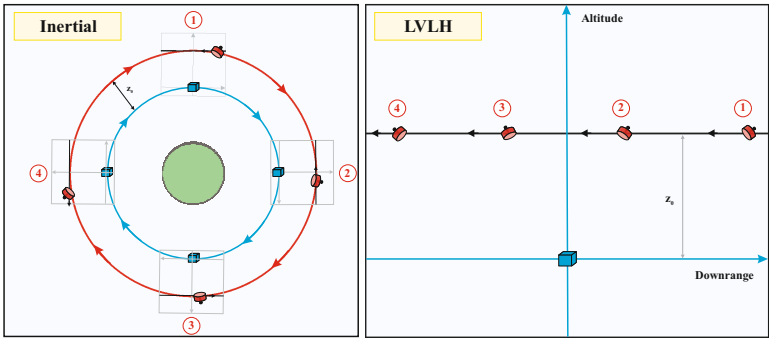


Fig. 4. Flyby, co-elliptic approach scenario

5.3 Circumnavigating (Football) Orbit

When the target and chaser have the same orbital periods, a repeating relative motion of the chaser near the target occurs. This is also known as a football orbit. In the football orbit, the chaser is placed on the v-bar at a given downrange distance from the target with an initial velocity required for a football orbit trajectory as shown in Figure 5. This unique elliptical relative motion is often employed to keep the chaser in a holding pattern downrange from the target for

station keeping, or in a circumnavigation orbit for target inspections. The initial position/velocity state vector for this scenario is:

$$X_0^{football} = \left[x_0 \; 0 \; 0 \; 0 \; 0 \; \tfrac{1}{2} x_0 \omega \right]^T, x_0 = 1m, \; or \; 10m \tag{40}$$

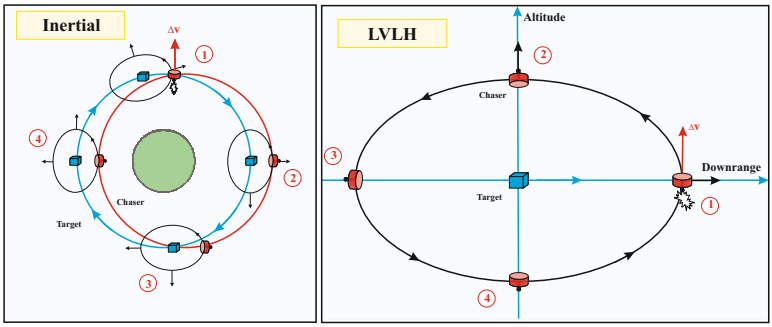


Fig. 5. Circumnavigation football orbit scenario

5.4 Initial Relative Position Errors

The variances of the initial relative state errors are the same for all of the above orbits and depend only on the initial relative distance of target. When the initial chaser location is $1m$ away from the target, the initial state error covariance is

$$P_{0,1m} = Diag\left[(0.1m)^2 \; (0.1m)^2 \; (0.1m)^2 \; (10^{-4}m/s)^2 \; (10^{-4}m/s)^2 \; (10^{-4}m/s)^2 \right] \tag{41}$$

When the initial chaser location is $10m$ away from the target, the initial state error covariance is

$$P_{0,10m} = Diag\left[(1m)^2 \; (1m)^2 \; (1m)^2 \; (10^{-3}m/s)^2 \; (10^{-3}m/s)^2 \; (10^{-3}m/s)^2 \right] \tag{42}$$

While these initial state errors are not associated with any particular mission or navigation system, they are reasonable and primarily used to demonstrate range observability. Values for the initial true trajectory errors are obtained by sampling the above covariance matrices.

6 Range Observability under Vehicle Rotation

The observability problem was examined for three types of relative motion orbits – v-bar station-keeping, flyby orbits, and football orbits (also known as a circumnavigation orbits). In each case, the chaser was commanded to rotate perpendicular to the LOS, back and forth, through a prescribed angle for a period

of 10 minutes, a time period sufficient to determine whether or not the range is observable.

For each type of orbit, a sensitivity analysis was conducted for a range of several key problem parameters. The parameters and their assigned values are summarized in Table 1. The *chaser distance* is the initial range between the chaser center-of-mass and the target center-of-mass. The *measurement noise,* σ_{Noise}, is the 1-σ value of the camera angle measurement noise. The *camera offset* is the distance of the camera from the chaser center-of-mass . The *maximum chaser rotation angle,* $C_{rotation}$, is the maximum chaser rotation angle. The *time to rotate,* C_{time}, is the time it takes for the camera to complete its maximum turn. These last two parameters define the angular rotation rate of the chaser. In all cases, the power-spectral density of the acceleration process noise is given by $q_a = 10^{-10} km^2/sec^3$. This value of q_a produces approximately $6m$ meters of position error in LEO during the 10 minutes rotation period.

Table 1. Sensitivity analysis parameters

Type	Values
Orbit type	v-bar, flyby, football
Chaser distance $[m]$	$1, 10$
Measurement noise $[rad], 1\sigma$	$0.001, 0.0001$
Camera offset $[m]$	$0.1, 0.2, 0.5$
Maximum chaser rotation angle $[deg]$	$10, 45$
Time to rotate $[sec]$	$30, 60$

To quantify navigation performance, the LOS position error variance and the variance of the position error perpendicular to the LOS are employed as measures of range observability. When the camera is located at the chaser center-of-mass, the range is not observable and the LOS position error variance is constant or diverging. When the camera is offset from the chaser center-of-mass, the LOS position error variance approaches zero indicating range observability. This is verified by examining the condition number of the observability Gramian..

To show this, 10 minute simulations were run. Perfect initial conditions were used as defined for each orbit in Eqs. (38)-(40). Additionally, the measurement noise covariance in the Kalman filter was assigned, but the actual measurements were perfect measurements. The chaser rotation sequence is composed of four rotations: a positive rotation about the body y-axis, , a positive rotation about the body z-axis, and equal but opposite rotation about the body y-axis, and a final equal but opposite rotation about the body z-axis.

6.1 V-bar Station-Keeping

For this first scenario, $C_{rotation} = 10\, deg$, $C_{time} = 30\, sec$, and $\sigma_{Noise} = 0.001\, rad$. The chaser is initial located on the v-bar at a distance $1m$ from the target or $10m$ from the target. Figure 6 shows the 3-σ LOS position estimation errors (left),

the 3-σ position estimation errors perpendicular to the LOS (middle) and log of the condition number of the observability Gramian matrix (right). The results for the $1m$ v-bar position are shown on the top and the results for the $10m$ v-bar position are shown on the bottom. When the camera offset is zero (dark blue curves) the LOS position error diverges regardless of the chaser distance from the target. On the other hand, when the camera offset relative to the chaser center-of-mass is non-zero, the LOS position error convergences as the chaser rotations are employed, regardless of the distance from the target. That is, the range becomes observable. As the camera offset is increased, the LOS position error decreases further providing improved range estimation. A small penalty for obtaining range observability is reflected in a slightly increased variance of the position error perpendicular to the LOS compared with the zero offset case. The results of the Gramian condition number support the observed behavior of the LOS position error (range) variance, and the conclusion that the range is observable.

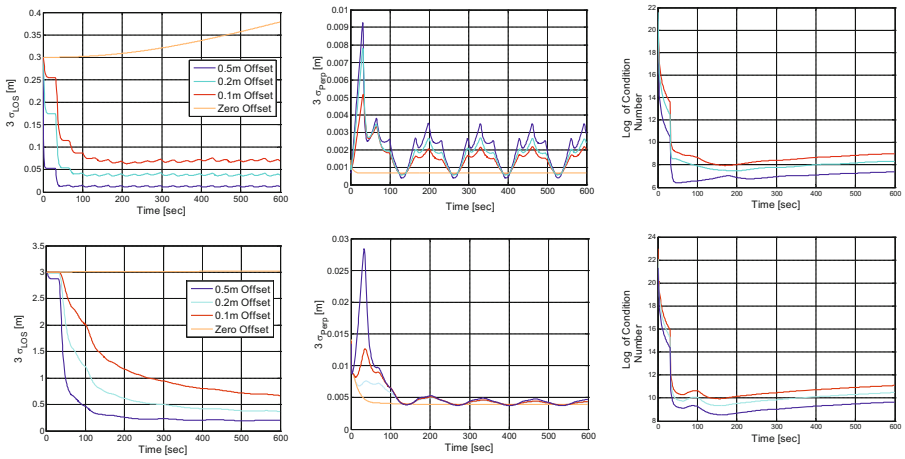


Fig. 6. V-bar station-keeping position estimation performance. LOS $3-\sigma$ position estimation errors (left), the $3-\sigma$ position estimation errors perpendicular to LOS (middle) and log of the condition number of the observability Gramian matrix (right) are shown as a function of the v-bar position, $1m$ (top), and $10m$ (bottom), and the camera offset.

Next, the effect of the rotation angle and the rotation rate on navigation performance was examined. In all cases, the camera measurement noise is $\sigma_{Noise} = 0.001rad$ and the chaser is on the v-bar, $1m$ from the target. In addition, three different values for the camera offset are selected. The results in Figure 7 show that larger chaser rotation angles improve the range observability. This is due to the fact that the larger rotation angles provide a larger camera displacement and a better baseline for the (pseudo-stereo) angle measurements. Figure 7 also

shows that faster rotation rates improve performance. In this case, the faster rotation rates merely provide more useful measurements at large angles. The effect of the rotation angle and the rotation rate on the perpendicular error is opposite their effect on the LOS position error, i.e., larger rotations and faster rotation rates produces slightly larger errors. Finally, it can also be seen that as camera offset is increased, the LOS position variance decreases, while the error perpendicular to the LOS variance increases slightly. The above analysis was repeated with the chaser on the v-bar at a position $10m$ from the target, and the same trends were observed.

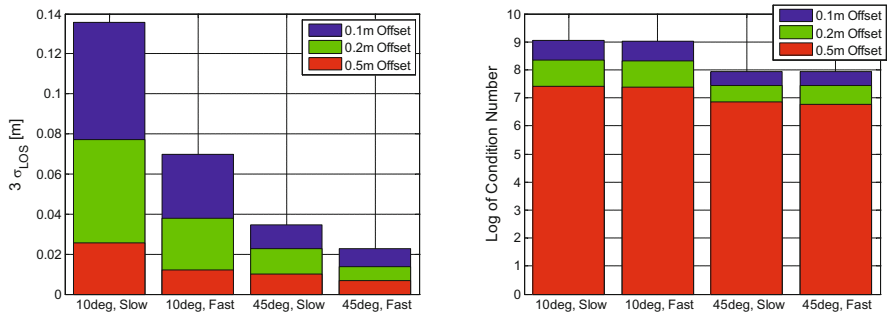


Fig. 7. $3-\sigma$ LOS position estimation errors (left) and log of the condition number of the observability Gramian matrix (right) at the end of the 10 min simulation with the chaser on the v-bar, $1m$ from the target

6.2 Monte Carlo Results

A subset of the scenarios examined above were re-examined with 800 sample Monte-Carlo runs to validate the single run analysis. At the beginning of each run, the initial position and velocity errors were randomly sampled from the covariance matrices given in Eqs. (41)-(42) based on the initial chaser-target separation. The Monte-Carlo results indeed confirmed that range observability is achieved with a camera offset and also provided confirmation of the above single-run analysis.

V-bar Station-Keeping. For example, consider the case where the chaser is on the v-bar, $1m$ from the target, and the camera measurement noise is $\sigma_{Noise} = 0.001rad$. Figure 8 shows the final position estimation errors as determined by the the Monte Carlo analysis. These results can be compared to the single-run results in Figure 6 . The Monte Carlo LOS position errors match the single run results very nicely. The Monte Carlo perpendicular position errors are slightly different, but nonetheless very small.

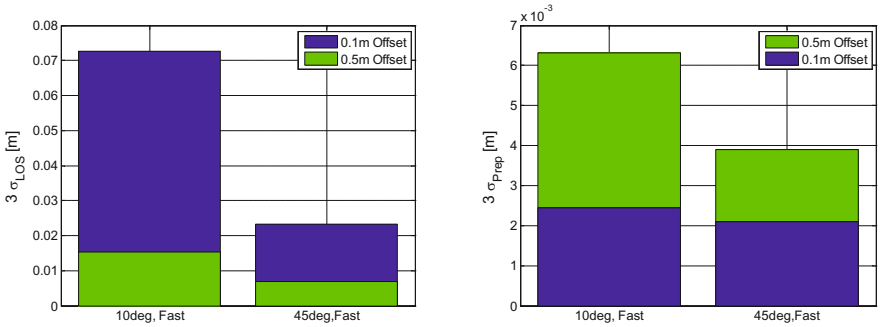


Fig. 8. Monte-Carlo results for the v-bar station keeping trajectory. $3-\sigma$ LOS position errors (left) and $3-\sigma$ position errors perpendicular to LOS (right) at the end of the 10 min simulation with the chaser on the v-bar, $1m$ from the target.

Figure 9 shows the final $3-\sigma$ position errors as a function of rotation angle ($10deg/45deg$), camera offset ($0.1m/0.5m$), initial separation ($1m/10m$), and measurement noise ($0.001rad$ / $0.0001rad$), where the bottom plot is an expanded view of the upper plot. In all cases, a fast rotation rate is being used. The observed trends are the same observed in the single runs: 1) larger rotation angles produce smaller LOS position errors; 2) larger the camera offset produces smaller LOS position errors; 3) smaller measurement noise reduces position and velocity errors, and 4) smaller separation distances produces smaller position and velocity errors.

Figure 10 presents $3-\sigma$ LOS position errors for eight different cases at the end of the 10 min simulation. It is observed that the log of the Gramian matrix condition number trends are the same as the $3-\sigma$ LOS position errors and mentioned earlier.

Flyby Orbit. Figure 11 shows the final $3-\sigma$ position errors as a function of rotation angle ($10deg/45deg$), camera offset ($0.1m/0.5m$), and measurement noise ($0.001rad/0.0001rad$) for a $1m$ flyby orbit above the target. The trends are identical to the results of the single-runs and identical to the results of the v-bar Monte Carlo analysis, and the numerical values of the final position errors are comparable to previous results.

Circumnavigating (Football) Orbit. Figure 12 shows the final $3-\sigma$ position errors as a function of rotation angle ($10deg/45deg$), camera offset ($0.1m/0.5m$), and measurement noise ($0.001rad/0.0001rad$) for a $1m$ football orbit as the chaser passes in front of the target. The trends are again identical to the results of the single-runs and identical to the results of the v-bar Monte Carlo analysis, and the numerical values of the final position errors are comparable to previous results.

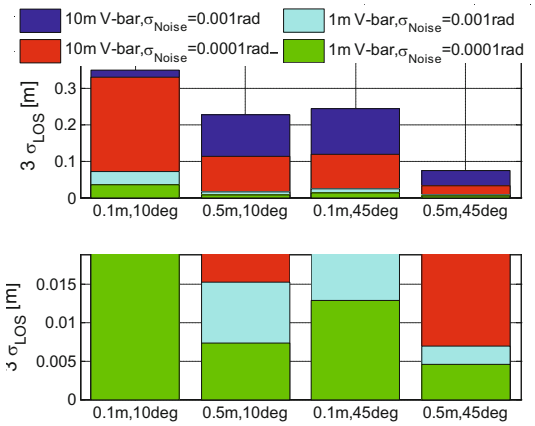


Fig. 9. Monte-Carlo results for the v-bar station keeping trajectory. $3-\sigma$ LOS position errors at the end of the 10 min simulation with the chaser on the v-bar. Bottom plot zooms into the lower part of theis an expanded view of the upper plot.

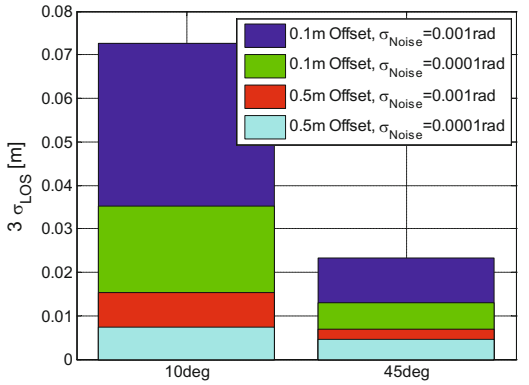


Fig. 10. Monte-Carlo results for the v-bar station keeping trajectory. $3-\sigma$ LOS position errors at the end of the 10 min simulation with the chaser on the v-bar.

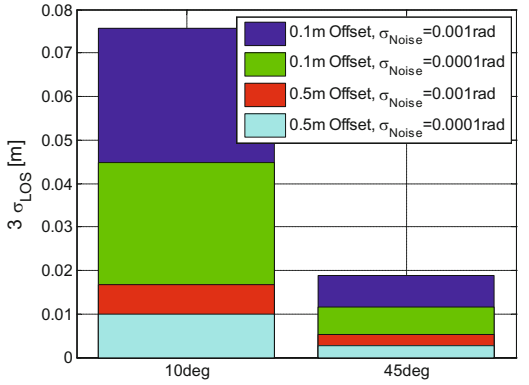


Fig. 11. Monte-Carlo results for flyby orbit. $3-\sigma$ LOS position errors at the end of the 10 min simulation with the chaser on a flyby orbit above the target.

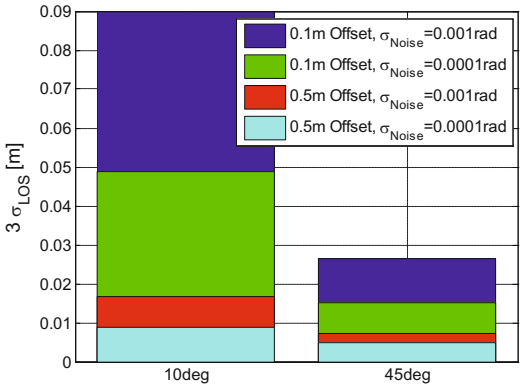


Fig. 12. Monte-Carlo results for a football orbit. $3-\sigma$ LOS position errors at the end of the 10 min simulation with the chaser on a football orbit moving upward in the radial direction from the v-bar.

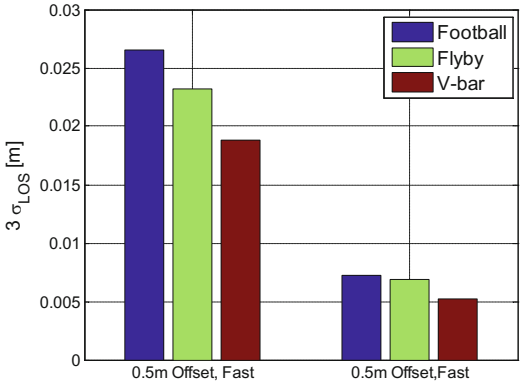


Fig. 13. Monte-Carlo results for v-bar station-keeping, flyby orbit, and football orbit. $3-\sigma$ LOS position errors at the end of the 10 min simulation are shown.

Performance Comparison. Figure 13 shows a comparison of the final LOS position errors for the v-bar, flyby, and football orbits. For the specified rotation sequence, the v-bar trajectory produced the best performance. The flyby and football orbit results can undoubtedly be improved by considering other rotation sequences.

7 Conclusions

During orbital proximity operations, past research has shown that angles-only navigation during coasting flight suffers from a lack of range observability. This paper has shown that the angles-only range observability problem during coasting flight is solved by properly including the offset of the camera from the chaser

center-of-mass in the problem formulation, and by applying appropriate vehicle rotations. Although this result is significant, it is limited, since the ability to estimate range diminishes with chaser-target separation distance. It has also been shown that the ability to estimate range improves with larger camera offset distances, larger maneuvers, and faster maneuvers. In all cases examined, the position estimation error perpendicular to the line-of-sight remains approximately unchanged.

Acknowledgments. This research was conducted with support and funding from the Lady Davis Fellowship Trust and the Asher Space Research Fund.

References

1. Allen, A.M., Langley, C., Mukherji, R., Taylor, A.B., Umasuthan, M., Barfoot, T.: Rendezvous lidar sensor system for terminal rendezvous, capture, and berthing to the international space station. In: Proceedings of SPIE, Vol. 6958, SPIE Conference on Sensors and Systems for Space Applications II. SPIE (2008)
2. Clohessy, W.H., Wiltshire, R.: Terminal guidance system for satellite rendezvous. Journal of the Aero/Space Sciences 27(3), 653–658 (1960)
3. Dennehy, C.J., Carpenter, J.R.: A summary of the rendezvous, proximity operations, docking, and undocking (rpodu) lessons learned from the defense advanced research project agency (darpa) orbital express (oe) demonstration system mission. NASA/TM-2011 217088 (2011)
4. Fehse, W.: Automated Rendezvous and Docking Spacecraft. Cambridge University press, New York (2003)
5. Gillis, R., Geller, D., Chavez, F.R.: Proximity operations using low thrust propulsion and angles-only measurements in geosynchronous orbits. In: AAS Annual Guidance and Control Conference, AAS, Breckenridge (2011)
6. Gillis, W.R.: Low Thrust Assisted Angles Only Navigation. Graduate Thesis, Utah State University (2011)
7. Howard, R.T., Bryan, T.C.: Video guidance sensor for automated capture. In: Space Programs and Technologies Conference, AIAA, Huntsville (1992)
8. Howard, R.T., Bryan, T.C., Book, M.L.: On-orbit testing of the video guidance sensor. In: Proceedings of SPIE, The International Society for Optical Engineering, vol. 3707, pp. 290–300. SPIE, Bellingham (1999)
9. Junkins, J.L., Hughes, D.C., Wazni, K.P., Pariyapong, V.: Vision-based navigation for rendezvous and docking and proximity operations. In: 22nd Annual AAS Guidance and Control Conference, Breckenridge, AAS, CO (1999)
10. Kim, S., Crassidis, J.L., Cheng, Y., Fosbury, A.M.: Kalman filtering for relative spacecraft attitude and position estimation. Journal of Guidance, Control, and Dynamics 30(1), 133–143 (2007)
11. Lee, D., Pernicka, H.: Integrated system for autonomous proximity operations and docking. International Journal of Aeronautical and Space Science (1), 43–56 (2011)
12. Maybeck, P.S.: Stochastic models, estimation and control, vol. 1. Navtech Book and Software store (1994)
13. Parten, R.P., Mayer, J.P.: Development of the gemini operational rendezvous plan. Journal of Spacecraft and Rockets 5(9), 1023–1028 (1968)

14. Don, J.P.: Shuttle rendezvous and proximity operations. In: Proceedings of the International Symposium, Space dynamics, Toulouse, France, pp. 833–851 (1989)
15. Petit, A., Marchand, E., Kanan, i.: Vision-based space autonomous rendezvous: A case study. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp. 619–624. IEEE, San Francisco (2011)
16. Pinard, D., Reynaud, S., Delpy, P., Strandmoe, S.E.: Accurate and autonomous navigation for the atv. Aerospace Science and Technology, 490–498 (2007)
17. Rumford, T.E.: Demonstration of autonomous rendezvous technology (dart) project summary. In: Proceedings of SPIE, SPIE Space System Technology and Operations, vol. 5088, SPIE, Orlando (2003)
18. Sabol, C., Vallado, D.: A fresh look at angles-only orbit determination. In: AAS/AIAA Astrodynamics Specialist Conference, Vol. AAS 99-363, Girdwood, Alaska (1999)
19. Schmidt, J., Geller, D., Chavez, F.R.: Viability of angles-only navigation for orbital rendezvous operation. In: AIAA-2010-7755, AIAA Guidance, Navigation, and Control Conference, Toronto, Ontario, AIAA, Toronto (2010)
20. Smith, D.W., Lampkin, B.A.: Sextant sighting measurements from on board the gemini xi1 spacecraft. NASA Technical Note (1968)
21. Vallado, D.: Fundamentals of Astrodynamics and Applications, vol. 2. Microcosm Press and Kluwer Academic Publishers (2004)
22. Weismuller, T., Leinz, M.: Gnc technology demonstrated by the orbital express autonomous rendezvous and capture sensor system. In: AAS 29th Annual Guidance and Control Conference, AAS, Breckenridge (2006)
23. Woffinden, D., Geller, D.: Observability criteria for angles-only navigation. IEEE Transactions on Aerospace and Electronic Systems 45(3), 1194–1208 (2009)
24. Woffinden, D.C., Geller, K.D.: Relative angles-only navigation and pose estimation for autonomous orbital rendezvous. Journal of Guidance, Control, and Dynamics 30(5), 1455–1469 (2007)
25. Yim, J.R., Crassidis, J.L., Junkins, J.L.: Autonomous orbit navigation of two spacecraft system using relative line of sight vector measurements. American Astronautical Society 4(257) (2004)
26. Young, K.A., Alexander, J.D.: Apollo lunar rendezvous. Journal of Spacecraft and Rockets 7(9), 1083–1086 (1970)
27. Zarchan, P., Musoff, H.: Fundamentals of Kalman filtering: a practical approach, 2nd edn. The American Institute of Aeronautics and Astronautics, Inc. (2005)

Ground Based Navigation of Spacecraft in Lunar Transfer Trajectory, with Application to Chandrayaan-2

Sanat Kumar Biswas* and Hari B. Hablani**

Indian Institute of Technology Bombay, Mumbai, India
sanatb.88@gmail.com, hbhablani@aero.iitb.ac.in

Abstract. Navigation plays a major role in space missions. For a cis-lunar space mission, such as Chandrayaan-2, deep space radar, inertial navigation system and optical navigation system are used to determine the position and velocity of the spacecraft.

In the present work, a mathematical model of Chandrayaan-2 motion is developed which includes J2 effect of the Earth and the Moon, solar gravitational perturbation and solar radiation pressure. Based on this model a nominal lunar transfer trajectory is simulated. Models for ground-based measurements (azimuth, elevation, range and range rate) are described. Time delays due to finite speed of electromagnetic waves are considered in the measurement. Effects of tropospheric and ionospheric refraction on measurements are incorporated in the simulation. Tropospheric errors are corrected using Saastamoinen zenith range correction model and ionospheric range errors are corrected using dual-frequency measurements. MATLAB codes are developed to simulate the observations from four ground station facilities, viz. Indian Deep Space Network, (IDSN) Byalalu, Karnataka, India as well as Deep Space Communication Complex, Goldstone, California, USA, Deep Space Communication Complex, Madrid, Spain and Deep Space Communication Complex, Canberra, Australia.

The polar measurements are used in Extended Kalman Filter (EKF) algorithm to estimate position and velocity of Chandrayaan-2 in the inertial Cartesian frame. To compensate for the time delay in measurements, the states are propagated till the time corresponding to the delayed measurements and then updated. These updated states are then propagated further till the current time to obtain the current estimates. This technique results in 11.79 km of range uncertainty (1σ) and 6.93 m/s of speed uncertainty (1σ) at the time of arrival at the insertion point to the lunar parking orbit of Chandrayaan-2.

Keywords: Spacecraft, ground based navigation, estimation, Kalman filter.

* Former M.Sc. Student, Department of Aerospace Engineering.
** Professor, Department of Aerospace Engineering.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_20

1 Introduction

Navigation is the process of monitoring position and velocity of a vehicle moving from one place to another. Navigational techniques involve locating the position of a vehicle of interest with respect to a suitable reference frame. For missions like Chandrayaan-2, position and velocity are measured using ground tracking system. These quantities along with acceleration due to thrust are also measured by on board accelerometer, gyroscope and celestial navigation. Dynamic model of the spacecraft motion is well established in literature. Highly accurate gravity models of the Earth and the Moon have been developed [5]. Several nonconservative forces like solar radiation pressure, atmospheric drag have been modeled very accurately [5]. The uncertainties in position and velocity of spacecraft are the result of the uncertainties in injection position and velocity, measurement errors and guidance corrections. Another significant factor contributing to the uncertainty in the translunar trajectory is the translational noise, which is the result of imperfectly coupled rotational maneuvers, venting, etc. This may be modeled as process noise [2]. For all these reasons, propagation of initial conditions results in huge uncertainties in final position and velocity. To reduce the uncertainties, various estimation techniques are used. The estimation process involves collecting tracking data (azimuth, elevation, range and range rate) from ground stations and using them batch-wise in least square estimation algorithm or sequentially in Extended Kalman Filter algorithm. For on board navigation, spacecraft position is measured using optical navigation system and Inertial Navigation System (INS) and then Kalman filter is used to estimate position and velocity.

The field of cislunar navigation is researched extensively since sixties. Recent notable works are Chandrayaan-1 precise orbit determination and investigation of on board navigation errors of lunar trajectory of Orion space vehicle [2]. Clark et al have documented design and analysis of navigation system for the Orion translunar and transearth phases of the mission [2]. They have shown that, with infrequent ground station measurement updates, the on board navigation system is capable of estimating the position and velocity with 3 km and 0.06 m/s of uncertainty (1σ) respectively at the end of the lunar transfer trajectory. Lightsey et al have demonstrated autonomous cislunar navigation simulation in [6].

Vighnesam et al [8] have documented precise orbit computation technique used in Chandrayaan-1 mission. For orbit propagation in ground based navigation system, EGM-96 Earth gravitational model and LP100K lunar gravitational model were used. The trajectory was generated by integrating the equation of motion using Gauss-Jackson-Merson's 8th order method. Least square estimation technique was used to estimate position and velocity of Chandrayaan-1.

In the present work gravitational harmonics of the Earth and the Moon higher than J2 are not considered in the mathematical model of spacecraft motion. Additionally solar gravitational perturbation and solar radiation pressure have been considered. For integration, Runge-Kutta 4th order method is used. For estimation of Chandrayaan-2 state vector, Extended Kalman Filter is used.

2 Dynamic Model of Spacecraft Motion in Lunar Transfer Trajectory

The motion of the spacecraft is influenced by the Sun, the Earth and the Moon during the flight along a lunar transfer trajectory. Nonetheless, for high navigation accuracy in a real application, the influence of the Sun's gravity, solar radiation pressure, nonuniform gravitational fields of the Earth and the Moon have to be considered. the principle aim of the project is to study the navigational aspects of Chandrayaan-2 and not trajectory generation, so simpler equation of motion is discussed neglecting higher order gravitational harmonics of the Earth and the Moon to generate nominal trajectory.

2.1 Acceleration of the Spacecraft due to the Gravitational Fields of the Earth, the Moon and the Sun

If it is assumed that, the spacecraft moves entirely under the Earth's gravitational influence, then according to Newton's Law of Gravitation the equation of motion of the spacecraft will be,

$$\ddot{\mathbf{r}} + \frac{GM_{\oplus}}{r^3}\mathbf{r} = \mathbf{0} \tag{1}$$

where, $\mathbf{r}$ is the position vector of the spacecraft with respect to center of the Earth and G is universal gravitational constant. To incorporate gravitational effect of the Moon in the equation of motion Cowell's perturbation technique is used. In Cowell's perturbation technique the Earth and the spacecraft are treated as a single system and the acceleration due to the other bodies (the Sun and the Moon in this case) are considered as a disturbance [1]. The main acceleration due to the central body Earth is given by the equation 1. The perturbation due to external mass M (the Sun or the Moon) is [5]

$$\ddot{\mathbf{r}} = GM\left(\frac{\mathbf{s}-\mathbf{r}}{|\mathbf{s}-\mathbf{r}|^3} - \frac{\mathbf{s}}{s^3}\right) \tag{2}$$

Here s is the position vector of mass M. Using this equation, solar gravitational perturbation and lunar gravitational perturbation is calculated separately and are added with to the right side of the equation 1 and then integrated numerically to obtain the trajectory.

2.2 Gravitational Harmonics

The acceleration due to nonuniform shape and density of the Earth is [5],

$$\ddot{x} = \sum_{n,m} \ddot{x}_{nm} \qquad \ddot{y} = \sum_{n,m} \ddot{y}_{nm} \qquad \ddot{z} = \sum_{n,m} \ddot{z}_{nm} \tag{3}$$

where

$$\begin{aligned}
\ddot{x}_{nm} &= \frac{GM}{{R_\oplus}^2}(-C_{n0}V_{n+1,1}) \qquad\qquad , m = 0 \\
&= \frac{GM}{{R_\oplus}^2}\frac{1}{2}(-C_{nm}V_{n+1,m+1} - S_{nm}W_{n+1,m+1}) \\
&+ \frac{(n-m+2)!}{(n-m)!}(+C_{nm}V_{n+1,m-1} + S_{nm}W_{n+1,m-1}), \qquad\qquad m > 0 \\
\ddot{y}_{nm} &= \frac{GM}{{R_\oplus}^2}(C_{n0}W_{n+1,1}), \qquad\qquad m = 0 \\
&= \frac{GM}{{R_\oplus}^2}\frac{1}{2}(-C_{nm}W_{n+1,m+1} + S_{nm}V_{n+1,m+1}) \\
&+ \frac{(n-m+2)!}{(n-m)!}(-C_{nm}W_{n+1,m-1} + S_{nm}V_{n+1,m-1}), \qquad\qquad m > 0 \\
\ddot{z}_{nm} &= \frac{GM}{{R_\oplus}^2}(n-m+1).(-C_{nm}V_{n+1,m} - S_{nm}W_{n+1,m})
\end{aligned}$$

Here C_{nm} terms are gravitational coefficient terms, Vnm and Wnm terms are calculated using the recursive relations given in [5]. These acceleration terms are given in the earth fixed frame. The same formulation can be used to calculate acceleration due to nonuniform shape of the Moon. In trajectory simulation of Chandrayaan-2 C_{20} terms of the Earth and the Moon (which are the Earth and the Moon J2 terms) are considered. Higher order terms are neglected in our simulation.

2.3 Solar Radiation Pressure

Acceleration due to solar radiation pressure is given by [5]

$$\ddot{\mathbf{r}} = -P_\odot C_R \frac{A}{m}\frac{\mathbf{r}_\odot}{{r_\odot}^3}AU^2 \tag{4}$$

where $P_\odot$ =Solar radiation pressure
C_R =Solar radiation pressure coefficient of the spacecraft
$\mathbf{r}_\odot$ =Position of the spacecraft with respect to the Sun in ECI frame
This formulation is used in the present work to simulate acceleration due to solar radiation pressure.

2.4 Lunar Transfer Trajectory Simulation

Cowell's perturbation technique is used to simulate the nominal trajectory of Chandrayaan-2. In this simulation, solar gravitational perturbation, solar radiation pressure, lunar gravitational perturbation, J2 effect of the Earth and J2

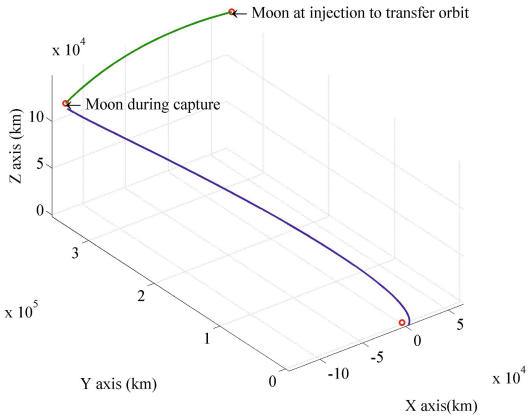


Fig. 1. Trajectory of Chandrayaan-2 using Cowell's perturbation technique

effect of the moon are considered. The gravitational coefficient of the Earth is taken from JGM-3 gravity model [5] and the graviational coefficient of the moon is taken from LP150Q lunar gravity model [7].

Figure 1 shows the transfer trajectory of Chandrayaan-2 in the ECI frame. This trajectory is generated by intigrating the equations mentioned in section 2.1. For the simulation of tracking data and estimation, this trajectory is used.

3 Ground Tracking Station Measurement Model

3.1 Azimuth and Elevation Measurement

If $\mathbf{r}$ and $\mathbf{R}$ be the position vectors of the spacecraft and the ground tracking station respectively in ECI frame then, the angle measurements can be modeled using the vector $\mathbf{d} = \mathbf{r} - \mathbf{R}$ from the ground station to the spacecraft. Due to the finite velocity of electromagnetic wave, the geometric relative position $\mathbf{d_0} = \mathbf{r(t)} - \mathbf{R(t)}$ at the time of signal reception is different from the true signal path, which can be expressed as the vector

$$\mathbf{d} = \mathbf{r}(t - \tau) - \mathbf{R}(t) \tag{5}$$

Here $\mathbf{R}(t)$ is the ground station position at the reception time, $\mathbf{r(t} - \tau)$ is the position of the spacecraft at the transmission time, τ is the signal travel time and c is the speed of electromagnetic wave. For the simulation purpose τ can be calculated by solving the implicit light-time equation [5]

$$c\tau = |\mathbf{r}(t - \tau) - \mathbf{R}(t)| \tag{6}$$

This equation can be solved iteratively and the solution gives the true signal path in the inertial frame which is different from the apparent direction to the

spacecraft for a moving ground station. Neglecting the special relativistic effects, the observed direction is given by the vector [5]

$$\mathbf{d}' = \mathbf{d} + \tau_d \mathbf{V} \tag{7}$$

where $\mathbf{V}$ is the inertial velocity of the ground station relative to the geocenter. Thus the apparent position [5]

$$\mathbf{d}' \approx \mathbf{r}(\mathbf{t} - \tau_\mathbf{d}) - \mathbf{R}(\mathbf{t} - \tau_\mathbf{d}) \tag{8}$$

From this vector $\mathbf{d}'$ the elevation and azimuth are measured. For simulation perpuse, we will consider azimuth and elevation calculated from $\mathbf{d}'$ vector as the angle measurments.

3.2 Range Measurement and Associated Time Delay

In classical two-way ranging technique, signal is radiated from the ground station to the spacecraft. The spacecraft transponder receives the signal and transmits back to the ground station. The ground station receives the transponded signal and determines the signal travel time τ. The signal travel time consists of up-link light time τ_u and down-link light time τ_d. The down-link light time equation is governed by equation 6. The up-link light time equation is expressed as [5]

$$c\tau_u = |\mathbf{r}(t - \tau_d) - \mathbf{R}(t - \tau_d - \tau_u)| \tag{9}$$

This equation has to be solved iteratively after solving down-link light time equation. The two-way range measurement ρ can be modeled as [5]

$$\begin{aligned} \rho &= \frac{1}{2}(\rho_u + \rho_d) \\ &= \frac{c}{2}(\tau_u + \tau_d) \end{aligned} \tag{10}$$

Here ρ_u is the up-link range $\rho(t - \tau)$ and ρ_d is the down-link range $\rho(t)$. Using Taylor series expansion, we can write

$$\begin{aligned} \rho(t - \tau) &= \rho(t - \frac{\tau}{2} - \frac{\tau}{2}) \\ &= \rho(t - \frac{\tau}{2}) - \frac{\tau}{2}\dot{\rho} + \left(\frac{\tau}{2}\right)^2 \ddot{\rho} - ....\infty \end{aligned} \tag{11}$$

and

$$\begin{aligned} \rho(t) &= \rho(t - \frac{\tau}{2} + \frac{\tau}{2}) \\ &= \rho(t - \frac{\tau}{2}) + \frac{tau}{2}\dot{\rho} + \left(\frac{\tau}{2}\right)^2 \ddot{\rho} + ...\infty \end{aligned} \tag{12}$$

From the equations 11 and 12 neglecting higher order terms one can write,

$$\begin{aligned} \rho\left(t - \frac{\tau}{2}\right) &\approx \frac{1}{2}[\rho(t - \tau) + \rho(t)] \\ &= \frac{1}{2}(\rho_u + \rho_d) \end{aligned} \tag{13}$$

This equation shows that, average two-way range at time t is nearly equal to the geometric range between the spacecraft and the ground station at time $\left(t - \frac{\tau}{2}\right)$. From this observation, one can conclude that the range measurement is delayed by $\frac{\tau}{2}$ second.

3.3 Range Rate Measurement and Associated Time Delay

Doppler frequency shift is utilized to measure the range rate of the spacecraft. Two-way Doppler measurements are obtained by integrating Doppler counts over a count-time interval t_c. The measured range rate is modeled as the difference of the two-way ranges at the end and at the beginning of the count-time interval [4]. It can be shown that the average range rate measurement can be modeled as [5]

$$\dot{\bar{\rho}}(t) = \frac{c}{2}\frac{(\tau_{2u} + \tau_{2d}) - (\tau_{1u} + \tau_{1d})}{t_c} \tag{14}$$

It has been shown in [4] that, the average range rate at time t_2 can be approximated as

$$\begin{aligned}\dot{\bar{\rho}}(t_2) &\approx \dot{\rho}\left(t_1 + \frac{t_i}{2} - \frac{1}{2}(\tau_{2d} + \tau_{1d})\right) \\ &= \dot{\rho}\left(t_2 - \frac{1}{2}(t_i + \tau_{2d} + \tau_{1d})\right)\end{aligned} \tag{15}$$

This equation shows that, the Doppler measurement is delayed by $\frac{1}{2}(t_c+\tau_{2d}+\tau_{1d})$ second.

4 Tracking Data from Ground Station

Simulation of tracking data generated with the ground station radars is done assuming Chandrayaan-2 is following the trajectory generate simulation described in section 2.1. The simulation results are obtained using the equations and procedure described in section 3.1 and section 3. The tracking data, time delay and approximation errors are generated for four ground tracking stations: Byalalu, Goldstone, Madrid and Canberra. Tracking data simulation program was verified by generating tracking data of a polar satellite and the data were compared with the tracking data provided in [5]. The following figures show the tracking data of Chandrayaan-2 for IDSN Byalalu. Results for Goldstone, Madrid and Canberra are included in the Appendix. In figures 2 and 3 the azimuths and elevations calculated from true and apparent position vectors (section 3.1) are compared to study the effect of light-time. From the figures 2 and 3 it can be concluded that the errors due to the finite speed of electromagnetic wave (section 3) in azimuth and elevation measurements are negligible. Here zero azimuth in the figure 2 corresponds to zero elevation in figure 3. For zero elevation the spacecraft is out of view of the ground station, so at that time no measurements are available.

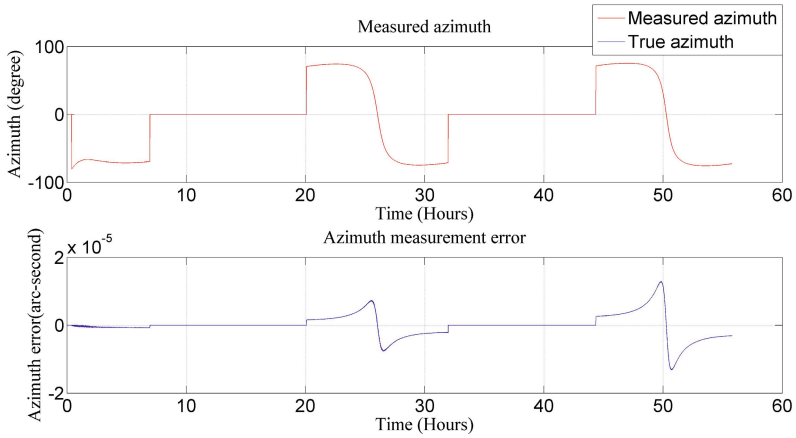


Fig. 2. Azimuth measurement from IDSN, Byalalu and error in measurement due to finite speed of electromagnetic wave

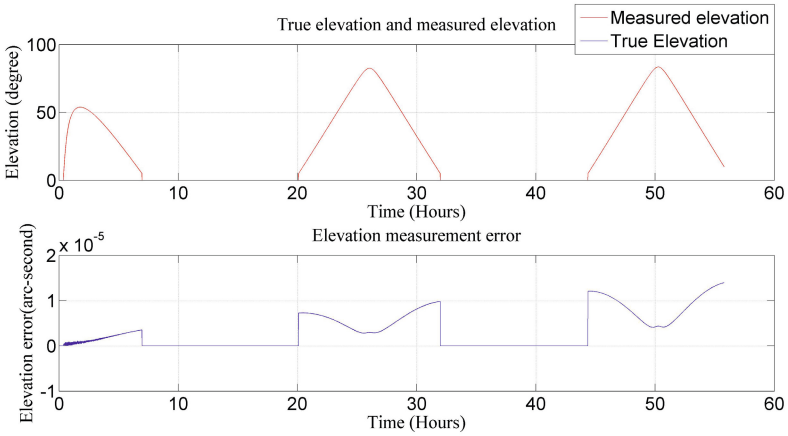


Fig. 3. Elevation measurement from IDSN, Byalalu and error in measurement due to finite speed of electromagnetic wave

during this time the spacecraft is tracked by other ground stations. Here measurements corresponding to elevation less than 15^o are rejected because, lower elevation angle associates with higher error due to troposphere and ionosphere.

Average range and range rate are calculated using equations 10 and 14 respectively. These range and range rates are compared with the instantaneous range and range rate at time $(t - \frac{\tau}{2})$ and $\left(t_2 - \frac{1}{2}(t_i + \tau_{2d} + \tau_{1d})\right)$ respectively. Figures 5 and 6 indicate that range rate delay is always higher than range delay because of the Doppler integration time t_c.Figures 8 and 10 show that the error due to Taylor series approximation is small as expected.

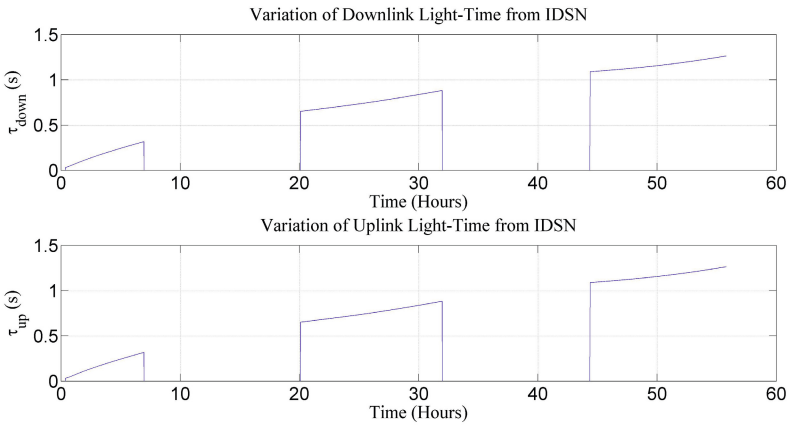


Fig. 4. Up-link and down-link light time delay from IDSN, Byalalu

5 Sources of Error in Ground Station Measurements

The main sources of error in ground station measurements are tropospheric and ionospheric interaction with electromagnetic wave and electronic noise. Electronic noise can be treated as white noise and can be accounted for using filter or estimation algorithms. But the errors in the range and elevation caused by troposphere and ionosphere has to be accounted for with rigorous modeling of the errors as done below.

5.1 Tropospheric Refraction

Tropospheric refraction introduces range and elevation measurement errors. Goad et al modified a general and accurate Hopfield tropospheric refraction model, to use Saastamoinen zenith range correction [5]. According to Hopfield model, the dry troposphere height h_1 is determined from

$$h_1[m] = \frac{0.011385}{N_1 \cdot 10^{-6}} p[hPa] \tag{16}$$

and the wet tropospheric height h_2 is expressed as

$$h_2[m] = \frac{0.011385}{N_2 \cdot 10^{-6}} \left[\frac{1255}{T[K] + 0.05}\right] e[hPa] \tag{17}$$

Here N_1 and N_2 are the dry and wet tropospheric refractivity, respectively, T is average temperature of the troposphere in Kelvin, p is the atmospheric pressure and e is the partial pressure of water vapor. The tropospheric range correction $\Delta\rho$ is given by [5]

$$\Delta\rho = C_\rho \sum_{j=1}^{2} \frac{N_j}{10^6} \sum_{i=1}^{9} \frac{\alpha_{ij} r_j^i}{i} \tag{18}$$

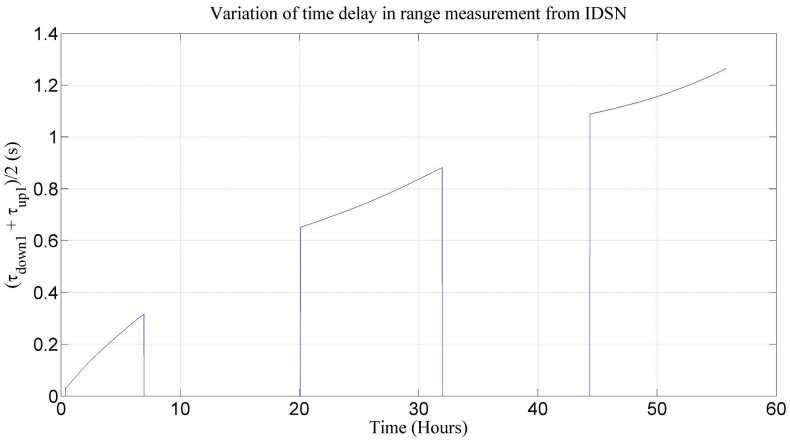


Fig. 5. Delay in range measurement

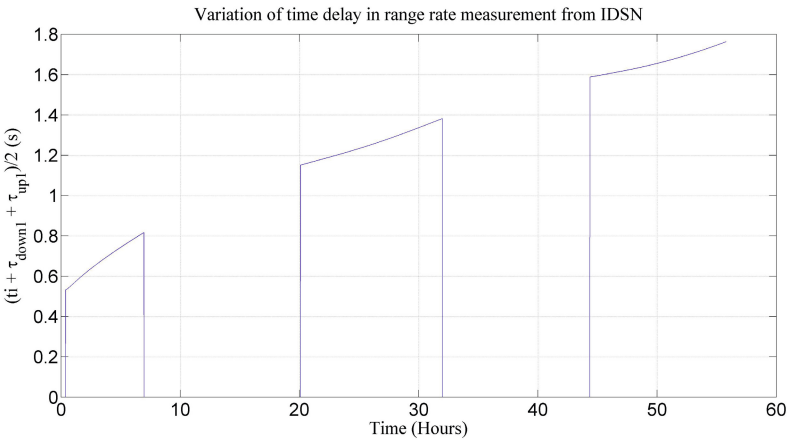


Fig. 6. Delay in range rate measurement

where C_ρ=0.99841 for radio frequencies, and for measured elevation angle E, r_j is defined as

$$r_j = \sqrt{(R_\oplus + h_j)^2 - (R_\oplus cosE)^2} - R_\oplus sinE \tag{19}$$

The coefficients of the polynomial (18) are defined in [5] The tropospheric correction of the elevation angle ΔE is given by [5]

$$\Delta E = C_E \frac{4cosE}{\rho} \left[\sum_{j=1}^{2} \frac{N_j}{10^6 h_j} \left(\sum_{i=1}^{7} \left(\frac{\beta_{ij} r_j^{i+1}}{i(i+1)} + \frac{\beta_{ij} r_j^i}{i} (\rho - r_j) \right) \right) \right] \tag{20}$$

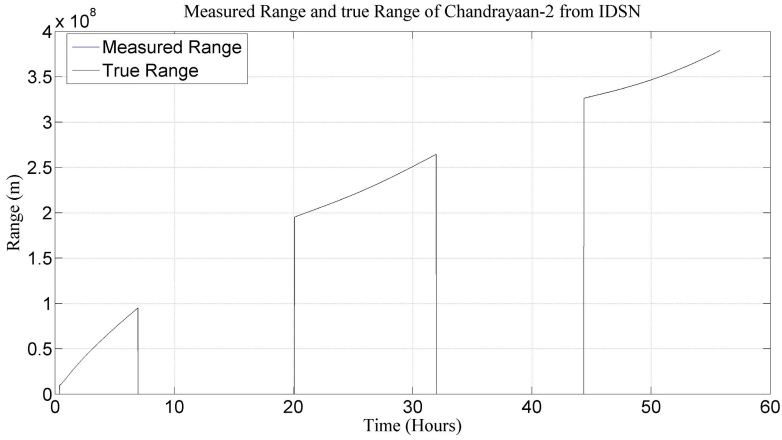


Fig. 7. Range measurement from IDSN, Byalalu

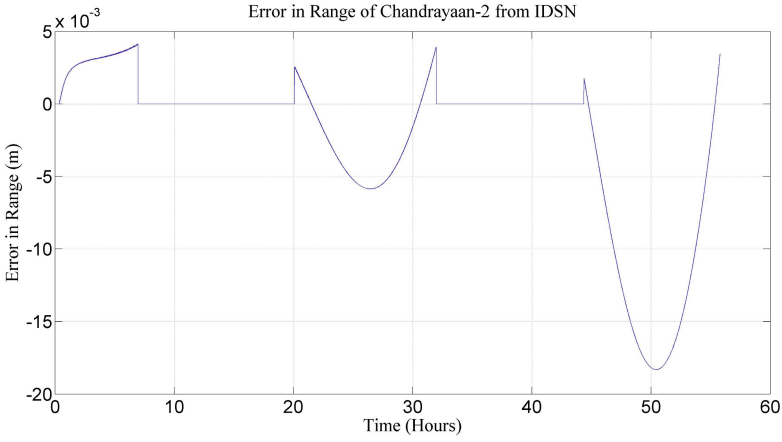


Fig. 8. Range error due to the Taylor series approximation

where, C_E=0.99841 for radio frequencies. The elevation correction coefficients are defined in [5].

5.2 Ionospheric Refraction

Range error due to ionospheric refraction can be calculated from the following equation [3]

$$\Delta\rho = \frac{40.3}{f^2} TEC_{slant} \tag{21}$$

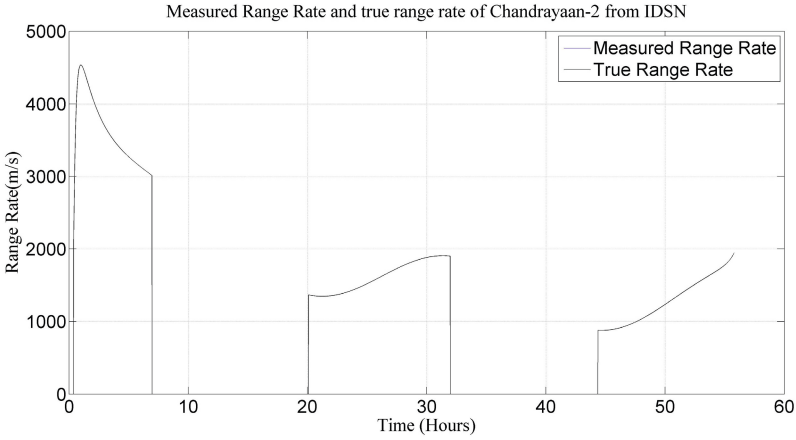


Fig. 9. Range rate measurement from IDSN, Byalalu

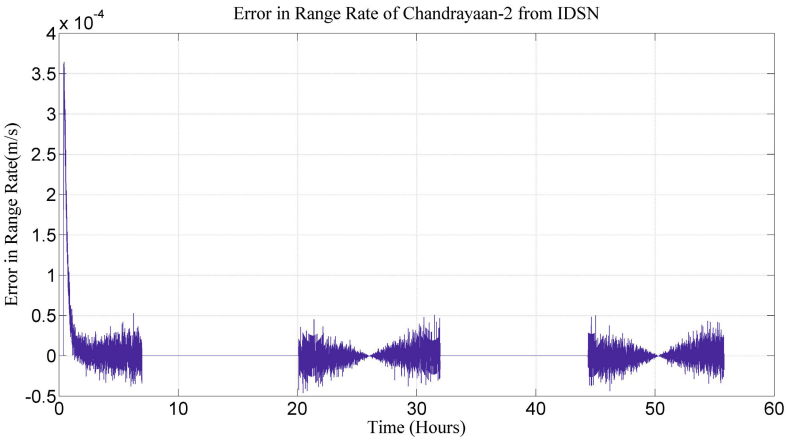


Fig. 10. Range rate error due to Taylor series approximation, Byalalu

where, $TEC_{slant} = \frac{TEC}{cosi}$ and TEC is the total electron count along the signal path. i is defined as [3]

$$i = sin^{-1}\frac{R_{\oplus}}{R_{\oplus}+h}cosE \tag{22}$$

h is the mean ionospheric height and E is the measured elevation. Ionospheric error correction can also be done using two signal frequencies [3]. Let the range measurements for the two frequencies f_1 and f_2 be

$$\rho_{f_1} = \rho_{true} + \frac{40.3}{f_1^2}TEC_{slant} \tag{23}$$

$$\rho_{f_2} = \rho_{true} + \frac{40.3}{f_2^2} TEC_{slant} \tag{24}$$

Here ρ_{true} is the true range. Subtracting 24 from 23,

$$\rho_{f1} - \rho_{f2} = 40.3 TEC_{slant} \left(\frac{1}{f_1^2} - \frac{1}{f_2^2} \right) \tag{25}$$

and

$$TEC_{slant} = (\rho_{f1} - \rho_{f2}) \frac{f_1^2 f_2^2}{40.3(f_1^2 - f_2^2)} \tag{26}$$

After calculating TEC_{slant} one can use equation 21 to determine the range error, or calculate ρ_{true} directly from either equation 23 or 24. But we have assumed in our work here that the ground station have the facility for dual frequency measurements.

5.3 Tropospheric Error Correction

Let the errors due to troposphere in range and elevation be $\Delta\rho$ and ΔE, respectively. If the true range and elevation be ρ_t and E_t, then from section 5.1 we can write

$$\Delta\rho = f_1(E_t + \Delta E) \tag{27}$$

$$\Delta E = f_2(E_t + \Delta E) \tag{28}$$

The functions f_1 and f_2 are described in section 5.1. The above equations can be solved iteratively to calculate the range and elevation error from the true range and elevation measurement. Now the actual measurement from the ground station is $(\rho = \rho_t + \Delta\rho + \Delta\rho_{ion})$ and $E = E_t + \Delta E$. $\Delta\rho_{ion}$ is the range error due to ionosphere. We can use these as measurements in the range and elevation correction model to calculate range and elevation error. The results are shown in the subsequent figures. Here the error after correction corresponding to the elevation greater than 15^o.

5.4 Ionospheric Error Correction

For ionospheric error simulation, electron density data are collected from NASA website for IDSN Byalalu, DSN Goldstone, Madrid and Canberra on September, 20, 2011 for 24 hours with 1 hour interval. We assume that for the next 2-3 days the daily variation of electron density will be same. Electron density is integrated to calculate TEC, and using the formulations described in 5.2 range error is calculated for two frequencies (2GHz and 2.1GHz) assuming all the four ground stations have this facility. Using the range measurements for the two frequencies, the ionospheric error is corrected (section 5.2). The TEC, electron density plot and the errors in range shown here corresponds to Byalalu tracking station. The figures 14 and 15 are not continuous because the electron density data are collected with one hour interval. Nonetheless, simulation using low resolution data demonstrates that using dual frequency correction the ionospheric range error can be corrected significantly (fig. 16).

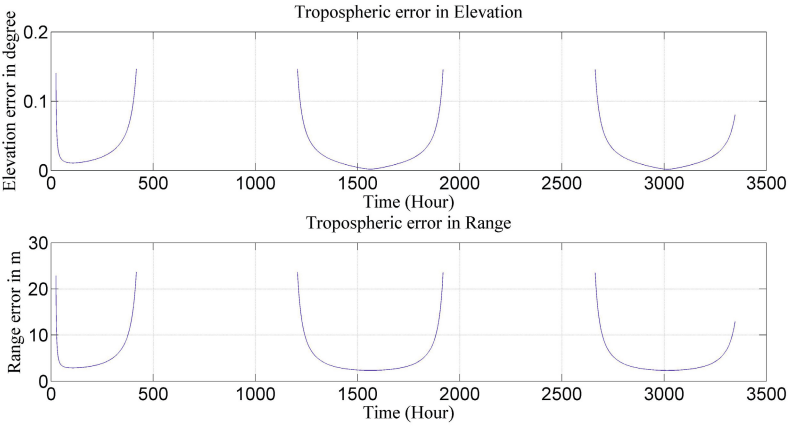


Fig. 11. Error in range and elevation measurement due to troposphere

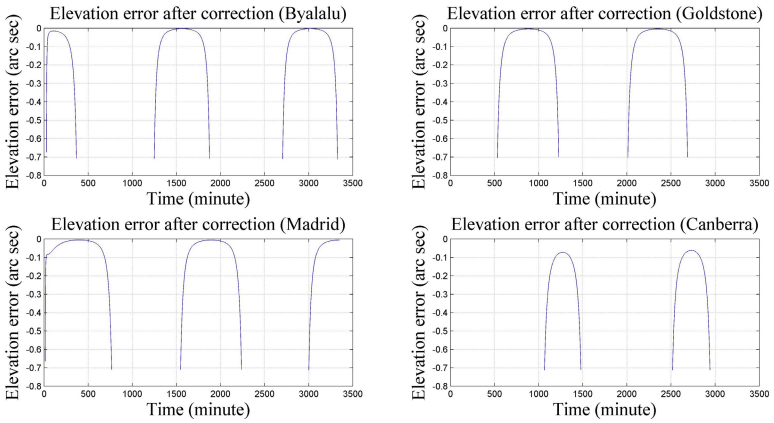


Fig. 12. Error in elevation measurement after correction

6 Spacecraft Position and Velocity Estimation

After correction of the ground station observations using the procedure described in 5 estimation algorithm is used in practice to determine the position and velocity of the spacecraft. This is because the atmospheric corrections may not eliminate the errors perfectly and electronic white noise always contaminate the observations. Generally least square estimation or Kalman Filter algorithm is used for estimation. In the current study, Extended Kalman Filter Algortihm (EKF) is used to estimate the position and velocity of Chandrayaan-2.

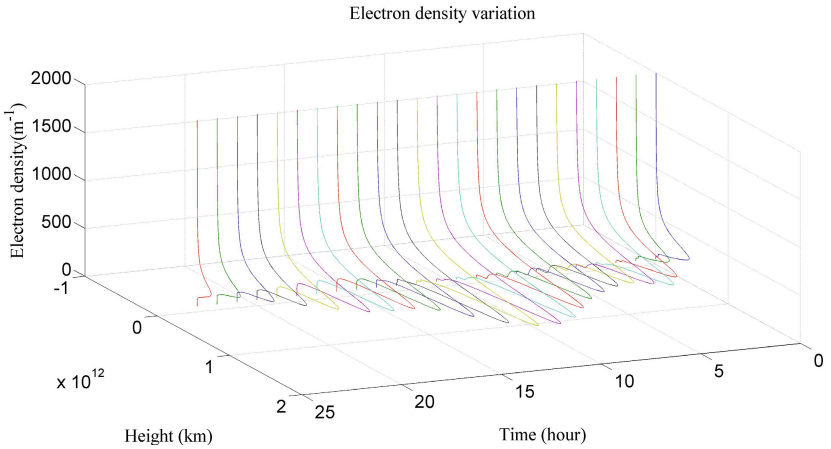


Fig. 13. Electron density variation with height for 20th September 2011

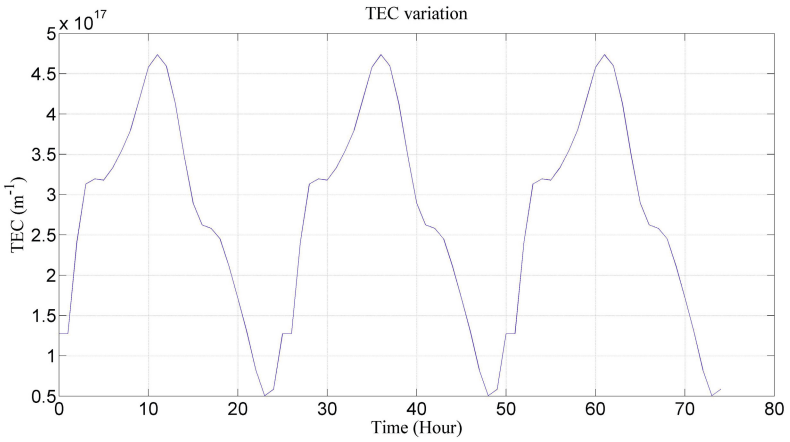


Fig. 14. Variation of TEC during lunar transfer at Byalalu

6.1 Extended Kalman Filter with Delay in Measurements

To compensate for the time delay in the measurements in the current study, the states are propagated till the time corresponding to the measurement and then states are updated. After that, the updated states are propagated till the current time. A block diagram of this scheme is shown in 17. In the block diagram Z_{RR} and Z_{RAE} are the range rate measurement and azimuth, elevation and range measurement vector respectively. τ_{rr} and τ_r are the delay in range rate and range respectively. The range rate delay $\tau_r r$ is greater than range delay τ_r due to the count time interval t_c (section 3.3). For this reason, states are updated using range rate measurement at $(t-\tau_r r)$ and the updated states are propagated

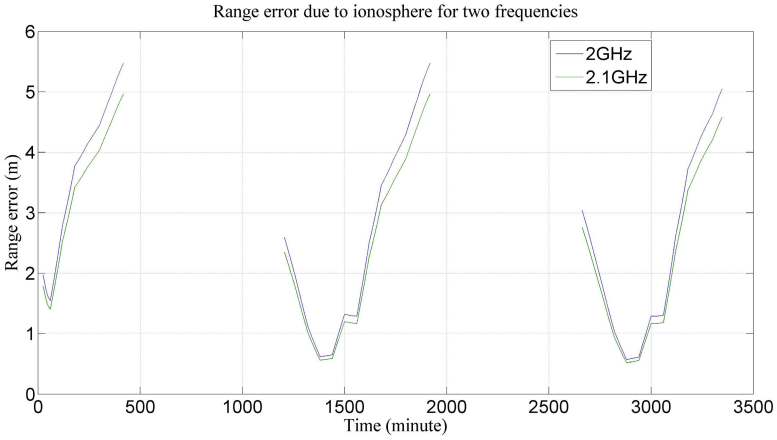


Fig. 15. Range error due to ionosphere for two frequencies

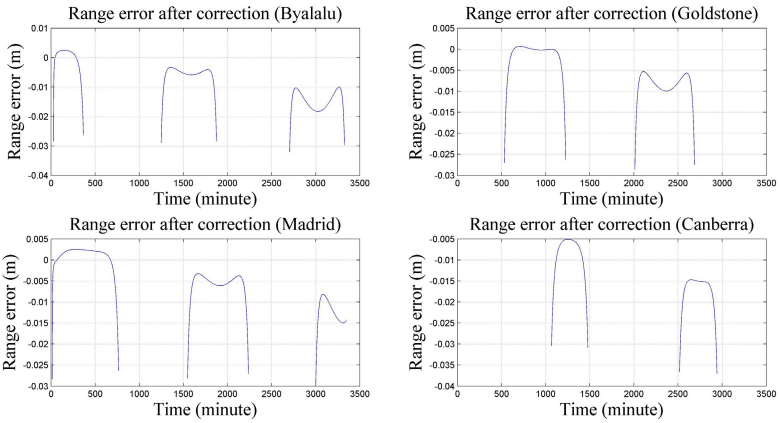


Fig. 16. Range error after tropospheric and ionospheric correction

till $(t-\tau_r)$. The range, azimuth and elevation measurements are associated with the time $(t-\tau_r)$ (section 3.2). The states are updated using these measurements at $(t-\tau_r r)$ and propagated till the time t.

6.2 Estimation of Position and Velocity

The correction for tropospheric and ionospheric errors are done to each set of measurements sequentially and used for estimation of position and velocity of the spacecraft. The algorithm used for estimation is explained in section 6.1. The uncertainties in tracking data are given in the following table [5]. The initial position and velocity vector estimates are based on the first set of measurements.

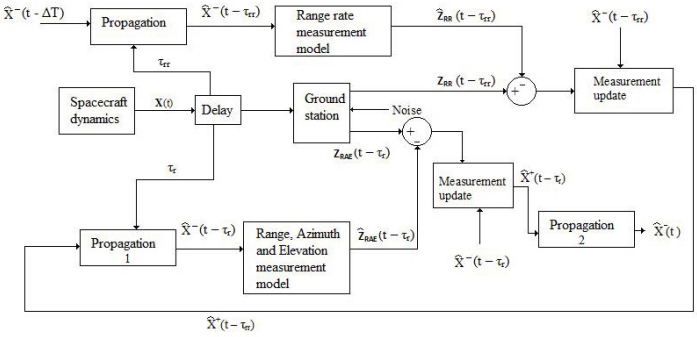


Fig. 17. Block diagarm of Kalman Filter algorithm for delay compensation

Table 1. Measurement uncertainties

Measured quantity	1σ of noise
Range	10m
Range rate	0.1 cm/s
Angles	0.01^o

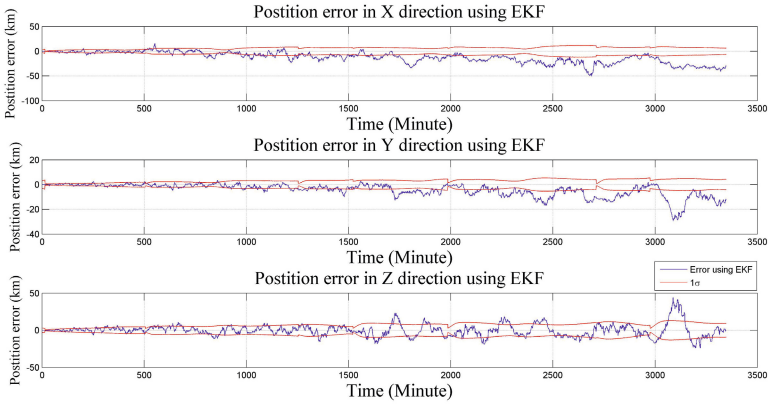


Fig. 18. Position estimation error in the ECI frame

Measurements from Byalalu and other tracking stations are used in the estimation simulation. Simulation results of the estimation process are illustrated and discussed below. Figures 18 and 19 show that the estimation errors are confined within 3σ limit but the accuracy of the filter deteriorates with time. This is because the azimuth and elevation noise. At the arrival to the Moon, the position uncertainty due to azimuth and elevation noise is roughly 76 km.

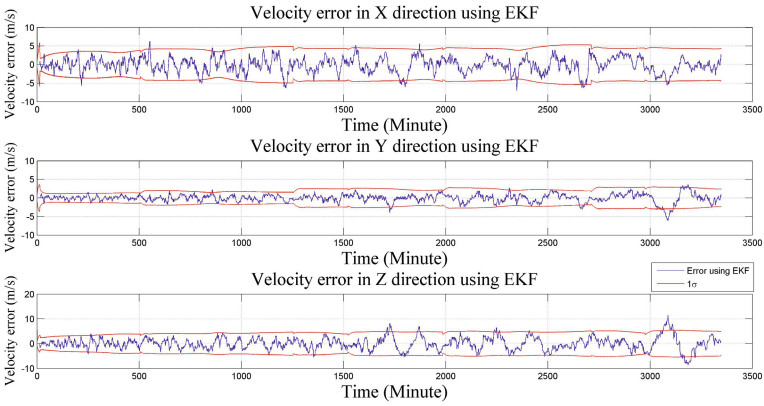


Fig. 19. Velocity estimation error in the ECI frame

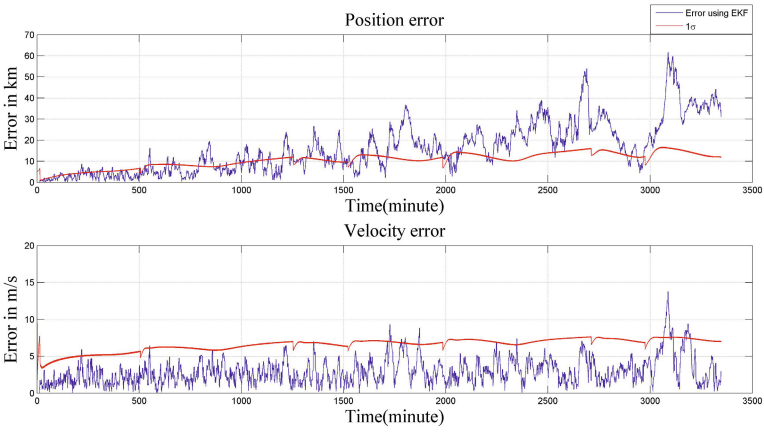


Fig. 20. Magnitude of position and velocity estimation errors in the ECI frame

Table 2. Estimation errors and uncertainties at the time of arrival at the Moon

Axis	Position error (km)	1σ of uncertainty (km)	Velocity error (m/s)	1σ of uncertainty (m/s)
X	-28.77	6.54	1.29	4.36
Y	-14.25	4.22	0.55	2.41
Z	-0.22	9.57	0.83	5.01

Using the Kalman filter this uncertainty is contained within 10 km (fig. 20). The errors and uncertainties at the time of arrival to the vicinity of the Moon after estimation are shown in the table 2.

7 Conclusion

The aim of this project is to understand navigational aspects of Chandrayaan-2 mission, simulate tracking data generated by ground tracking stations considering practical measurement and error models, and estimate position and velocity of Chandrayaan-2 from these data. Lunar transfer trajectory is simulated considering J2 effect of the Earth and the Moon, solar gravitational perturbation and solar radiation pressure. Range, range rate, azimuth and elevation Measurement models for ground tracking station are developed, and time delays due to the finite speed of electromagnetic wave, tropospheric and ionospheric errors are included in the measurement simulation. Correction procedure for tropospheric and ionospheric errors is discussed. It is shown that, as known, using dual frequency measurements, ionospheric range error can be corrected satisfactorily for higher elevation angles. An estimation algorithm is proposed to compensate for the delay in measurements. Final position uncertainty of the spacecraft is 11.79 km and velocity uncertainty is 6.93 m/s. The accuracy of the estimation process is not as per the practical requirements. More advanced filter algorithm is required to improve the accuracy. In this present study the error in elevation due to the ionosphere is not considered and the range noise uncertainty is assumed to be constant. For L-band and S-band signals, this error is comparable to the elevation noise. Further study is required to incorporate the ionospheric error in elevation and range noise characteristics. In recent years, research on autonomous navigation of spacecraft has become very important. Lightsey et al have shown the performance of autonomous navigation system in case of communication failure with ground station [6]. Extensive research has to be done on on-board autonomous navigation system. Injection error in transfer trajectory deviates the spacecraft from the desired trajectory. Several mid-course corrections are performed to keep the spacecraft in the desired trajectory. There is a scope of research on mid-course correction and guidance of the interplanetary spacecraft in transfer trajectory.

References

1. Battin, R.H.: An Introduction to the Mathematics and Methods of Astrodynamics. AIAA (1987)
2. D'Souza, C., Crain, T., Clark, F.D., Getchius, J.: Orion cislunar guidance guidance and navigation. In: AIAA Guidance, Navigation and Control Conference and Exhibit (2007)
3. Grewal, M.S., Weill, L.R., Andrews, A.P.: Global Positioning Systems, Inertial Navigation, and Integration. John Wiley & Sons, Inc. (2007)
4. Kruger, B.: The range rate error due to the averaging techniques of doppler measurements. Tech. rep., Goddard Space Flight Center (1965)
5. Montenbruck, O., Gill, E.: Satellite Orbits Models, Methods, and Applications. Springer (2000)
6. Muñoz, S., Christian, J.A., Lightsey, E.G.: Development of an end to end simulation tool for autonomous cislunar navigation. In: Navigation, and Control Conference. AIAA (2009)

7. Roncoli, R.B.: Lunar constants ans models document. Tech. rep., Jet Propulsion Laboratory (2005)
8. Vighnesam, N.V., Sonney, A., Gopinath, N.S.: India's first lunar mission chandrayaan-1 initial phase orbit determination. Acta Astronautica 67(4) (2010)
9. Vighnesam, N.V., Sonney, A., Subramaniam, B., Gopinath, N.S.: Precise orbit computation of india's first lunar mission chandrayaan-1 using accelerometer and tracking data during early phase. In: 21st ISSFD (2009)

Robust Kalman Filtering with Single and Multiple Scale Factors for Small Satellite Attitude Estimation

Halil Ersin Soken[1,*], Chingiz Hajiyev[2], and Shin-ichiro Sakai[3]

[1] The Graduate University for Advanced Studies (SOKENDAI), Sagamihara, Kanagawa, Japan
ersin_soken@ac.jaxa.jp
[2] Istanbul Technical University, Istanbul, Turkey
cingiz@itu.edu.tr
[3] Japan Aerospace Exploration Agency (JAXA), Sagamihara, Kanagawa, Japan
sakai@isas.jaxa.jp

Abstract. In case of normal operational conditions for a satellite, a conventional Kalman Filter gives sufficiently good attitude estimation results. On the other hand, when there is a fault in the measurements then the Kalman filter fails about providing the required accuracy and may even collapse by time. In this paper, a Robust Kalman filtering method is proposed for the attitude estimation problem. By using the proposed method both the Extended Kalman Filter and Unscented Kalman Filter are modified and the new algorithms, which are robust against the measurement malfunctions, are called as the Robust Extended Kalman Filter (REKF) and Robust Unscented Kalman Filter (RUKF), respectively. The adaptation is performed following both single and multiple scale factor based schemes. As an application example the proposed algorithms are applied for attitude estimation of a small satellite and the performance of the robust Kalman filters are compared in case of different measurement faults.

Keywords: robust Kalman filtering, REKF, RUKF, measurement fault, attitude estimation, small satellite.

1 Introduction

The Kalman Filter (KF) plays an important role in the attitude estimation procedure of the spacecrafts [1, 2]. Regarding the obstacles met during the development process of the attitude estimation systems, various types of KFs have been developed. One of these difficulties is the inherent nonlinear dynamics and kinematics of the satellites similarly to the many real world systems. The Extended Kalman Filter (EKF) was proposed so as to overcome this problem and since then it has been used instead of the linear Kalman filter for estimating the attitude of the satellite [2].

Even though still being a popular spacecraft attitude estimator, the EKF has some disadvantages, especially in case of high nonlinearity, which appears to be a common problem in the attitude determination applications [2]. On the other hand, the Unscented Kalman Filter (UKF) algorithm is a new filtering method which has become popular after the millennium especially because of its advantages over the well known

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control,*
DOI: 10.1007/978-3-662-44785-7_21

EKF. The essence of the UKF is the fact that; the approximation of a nonlinear distribution is easier than the approximation of a nonlinear function or transformation [3]. The UKF introduces sigma points to catch higher order statistic of the system and avoid the linearization step of the EKF. As a result it satisfies both better estimation accuracy and convergence characteristic [4]. Moreover, the UKF is more robust against the initial attitude estimation errors than the EKF [5].

One of the main problems for the attitude estimation via both the EKF and UKF is having an algorithm which does not have capability of the adaptation to the changing conditions of the measurement system. As known, space is a severe environment so malfunctions such as abnormal measurements, increase in the background noise etc. are always a potential threat for the onboard sensors. The main issue is such faults in the measurements affect the instantaneous filter outputs and may cause the attitude estimator diverge in long term. Hence the KF algorithm that is used as the attitude estimator must be built robust via using an adaptive approach [6].

In literature there are several methods to adapt the linear KF. Unquestionably, the pioneering methods in this area have been proposed by Mehra [7, 8]. Specifically the covariance matching technique discussed in [8] may be considered as the fundamental of the algorithm proposed in this paper. The main drawback of these studies, and as well their successors that examine the adaptation of the KF [9-13], they are generally appropriate for discrete-time linear systems and cannot be used as a method for the adaptation of the EKF or UKF without any correction of modification.

In this sense, researches on the Adaptive Extended Kalman Filter (AEKF) or Adaptive Unscented Kalman Filter (AUKF), which can be used for nonlinear systems, should be examined. In [14] basically the method proposed in [11] is generalized for nonlinear systems and a two-stage AEKF is presented for a loosely coupled INS/GPS systems. The main drawback of this AEKF approach is using a single fading factor for the adaptation procedure. This is not a healthy procedure since each term of the innovation sequence has a different effect on the filter performance, especially in a case where one of the sensors is faulty but the others are working properly. In [15], two distinct methods are described as the AUKF algorithms. In the first method, the MIT rule is used to derive the adaptive law and a cost function is defined in order to minimize the difference between the filter computed covariance and the actual innovation covariance. The algorithm is used for the Q-adaptation (process noise covariance adaptation) but it is stated that a similar approach may be followed for the R-adaptation (measurement noise covariance adaptation). As a deficiency, the presented algorithm requires calculation of the partial derivatives and that introduces a relatively large computational burden as it is also stated by authors themselves. In the second method, two UKFs are run in parallel within master and slave filter manner. Its computational demand is lower than the first method but as it is known [5], despite being free of the Jacobian calculations, the computational burden of the UKF is not low because of the sigma point calculations. Therefore, using two UKF algorithms in a parallel manner still increases the required computation burden significantly. Hence the main problem for both of the methods presented in [15] is high computational load. Nonetheless in [16, 17] Saga-Husa noise statistics estimator is integrated with the UKF in order to build an AUKF. Although it may be possible to have satisfactory

results for the target tracking problem, this method has an unstability issue; when the noise covariance loses its semi-positive definiteness the filter diverges.

The EKF and UKF may be also built adaptive by using the fuzzy logic techniques [18-20]. However, although it is possible to get satisfactory results for some specific cases, the essence of this kind of fuzzy methods is human experience and heuristic information; in out of experience cases they may not work.

In this paper, a Robust Kalman filtering method is proposed for the attitude estimation problem. The proposed method is applied for both the EKF and UKF and the new algorithms, which have an extension for the adaptation against measurement malfunctions, are called as the Robust Extended Kalman Filter (REKF) and Robust Unscented Kalman Filter (RUKF), respectively. The applied adaptation scheme is similar to the one given in [6]. However, in this paper the attitude estimation problem is generalized and instead of the Euler angles the quaternions are used as the attitude representation method. As well, the robust Kalman filters are examined for different measurement system failure cases. As introduced, there are some application examples for the UKF process noise adaptation in literature and similar complex procedures are suggested for the measurement noise adaptation. In contrast with these existing studies, this paper introduces a simple method for the measurement noise covariance matrix scaling. Moreover in order to show the clear effects of disregarding only the data of the faulty sensor we performed the adaptation by using both single and multiple scale factors which are two different approaches for the same problem. The algorithms are tested for the attitude and attitude rate estimation problem of a small satellite which has only three magnetometers as the attitude reference source. Using only magnetometers is a common preference for the small satellite applications (especially for the cubic pico satellites); and having only a limited number of sensors onboard increases the significance of the given robust Kalman filtering methods. Throughout the study, results of these proposed algorithms are compared and discussed for different types of measurement malfunctions.

2 Satellite Equations of Motion and the Sensor Model

2.1 Satellite Equations of Motion

The kinematics equation of motion of the satellite via the quaternion attitude representation can be given as [21],

$$\dot{\boldsymbol{q}}(t) = \frac{1}{2}\Omega(\boldsymbol{\omega}_{BR}(t))\boldsymbol{q}(t) \quad . \tag{1}$$

Here $\boldsymbol{q}$ is the quaternion vector formed of four attitude parameters, $\boldsymbol{q} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T$ and $\Omega(\boldsymbol{\omega}_{BR})$ is the skew symmetric matrix as;

$$\Omega(\boldsymbol{\omega}_{BR}) = \begin{bmatrix} 0 & r & -q & p \\ -r & 0 & p & q \\ q & -p & 0 & r \\ -p & -q & -r & 0 \end{bmatrix}, \tag{2}$$

where, p, q and r are the components of $\boldsymbol{\omega}_{BR}$ vector which indicates the angular velocity of the body frame with respect to the reference frame. On the other hand, the body angular rate vector with respect to the inertial axis frame should be stated separately as; $\boldsymbol{\omega}_{BI} = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T$. $\boldsymbol{\omega}_{BI}$ and $\boldsymbol{\omega}_{BR}$ can be related via,

$$\boldsymbol{\omega}_{BR} = \boldsymbol{\omega}_{BI} + A\begin{bmatrix} 0 & -\omega_0 & 0 \end{bmatrix}^T. \tag{3}$$

Here ω_0 denotes the angular velocity of the orbit with respect to the inertial frame, found as $\omega_0 = \left(\mu / r_0^3\right)^{1/2}$; μ is the gravitational constant, r_0 is the distance between the centre of mass of the satellite and the Earth. Moreover A is the attitude matrix which is related to the quaternions by;

$$A = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2(q_1q_2 + q_3q_4) & 2(q_1q_3 - q_2q_4) \\ 2(q_1q_2 - q_3q_4) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2(q_2q_3 + q_1q_4) \\ 2(q_1q_3 + q_2q_4) & 2(q_2q_3 - q_1q_4) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix}. \tag{4}$$

In case of using quaternions for the kinematic modeling of the satellite's motion, the UKF in standard format cannot be implemented straightforwardly. The reason of such drawback is the constraint of quaternion unity given by $\boldsymbol{q}^T\boldsymbol{q} = 1$. If the kinematics equation (Eq.1) is used in the filter directly, than there is no guarantee that the predicted quaternion mean of the UKF will satisfy this constraint.

One of the documented methods to overcome this problem is to use an unconstrained three component vector in order to represent an attitude-error quaternion instead of using all four components of the quaternion vector. In this paper we preferred to use the vector of Generalized Rodrigues Parameters (GRP) for representing the local error-quaternion. For details readers may refer to [5].

First let us rewrite the quaternion vector by $\boldsymbol{q} = \begin{bmatrix} \boldsymbol{g}^T & q_4 \end{bmatrix}^T$, so $\boldsymbol{g} = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^T$. After that when the local error-quaternion is denoted by $\delta\boldsymbol{q} = \begin{bmatrix} \delta\boldsymbol{g}^T & \delta q_4 \end{bmatrix}^T$, the vector of GRP may be given as [5],

$$\delta\boldsymbol{p} = f\left[\delta\boldsymbol{g} / (a + \delta q_4)\right]. \tag{5}$$

Here a is a parameter from 0 to1 and f is the scale factor. When $a = 0$ and $f = 1$ then Eq. (5) gives the Gibbs vector and when $a = 1$ and $f = 1$ then Eq. (5) gives the

standard vector of modified Rodrigues parameters. In [5] - as well as in this paper - f is chosen as $f = 2(a+1)$. The inverse transformation from $\delta \boldsymbol{p}$ to $\delta \boldsymbol{q}$ is given by

$$\delta q_4 = \frac{-a\|\delta \boldsymbol{p}\|^2 + f\sqrt{f^2 + (1-a^2)\|\delta \boldsymbol{p}\|^2}}{f^2 + \|\delta \boldsymbol{p}\|^2} \quad , \tag{6a}$$

$$\delta \boldsymbol{g} = f^{-1}(a + \delta q_4)\delta \boldsymbol{p} \quad . \tag{6b}$$

For a full-state attitude estimator where attitude and attitude rates are estimated jointly, we also need dynamic knowledge. The dynamic equations of the satellite can be derived based on the Euler's equations;

$$J\frac{\boldsymbol{\omega}_{BI}}{dt} = \boldsymbol{N}_d - \boldsymbol{\omega}_{BI} \times \left(J\boldsymbol{\omega}_{BI}\right), \tag{7}$$

where J is the inertia matrix consists of principal moments of inertia as $J = diag\left(J_x, J_y, J_z\right)$ and $\boldsymbol{N}_d$ is the vector of disturbance torque affecting the satellite which can be given as a sum of

$$\boldsymbol{N}_d = \boldsymbol{N}_{gg} + \boldsymbol{N}_{ad} + \boldsymbol{N}_{sp} + \boldsymbol{N}_{md} \quad . \tag{8}$$

Here $\boldsymbol{N}_{gg}$ is the gravity gradient torque, $\boldsymbol{N}_{ad}$ is the aerodynamic disturbance torque, $\boldsymbol{N}_{sp}$ is the solar pressure disturbance torque and $\boldsymbol{N}_{md}$ is the residual magnetic torque which is caused by the interaction of the satellite's residual dipole and the Earth's magnetic field [21].

2.2 Sensor Models

Magnetometer is a favorite sensor type for attitude estimation especially in small satellite applications. The model for the magnetometer measurements is given by (let us assume that magnetometers are already calibrated with one of the in-orbit or on-ground estimation methods [4,22]),

$$\begin{bmatrix} B_x(\boldsymbol{q},t) \\ B_y(\boldsymbol{q},t) \\ B_z(\boldsymbol{q},t) \end{bmatrix} = A \begin{bmatrix} B_1(t) \\ B_2(t) \\ B_3(t) \end{bmatrix} + \eta_1 , \tag{9}$$

where $B_1(t)$, $B_2(t)$ and $B_3(t)$ represent the Earth magnetic field vector components in the orbit frame as a function of time and can be modeled as given in [23].

Therefore, $B_x(\boldsymbol{q},t)$, $B_y(\boldsymbol{q},t)$ and $B_z(\boldsymbol{q},t)$ show the measured Earth magnetic field vector components in the body frame as a function of time and varying quaternion vector. Furthermore, concerning the Eq.(7), η_1 is the zero mean Gaussian white noise with the characteristic of

$$E\left[\eta_{1k}\eta_{1j}^{T}\right] = I_{3x3}\sigma_m^2\delta_{kj}\,. \tag{10}$$

I_{3x3} is the identity matrix with the dimension of 3×3 , σ_m is the standard deviation of each magnetometer error and δ_{kj} is the Kronecker symbol.

3 Robust Kalman Filtering for Satellite Attitude Estimation

The EKF and UKF work accurately when there is no fault in the measurement system. On the contrary, in case of a fault such as abnormal measurements, step-like changes or sudden shifts in the measurement channel etc. the filter deteriorates and the estimation outputs become faulty.

Therefore, a robust algorithm must be introduced such that the filter is insensitive to the measurements in case of malfunctions and the estimation process is corrected without affecting the remaining good estimation behavior.

The robustness of the filter is secured by scaling the measurement noise covariance matrix in case of fault. In this sense two different approaches may be used: Scaling by a single scale factor or scaling by a scale matrix built of multiple scale factors. In general, despite its relative simplicity, using single scale factor is not a healthy procedure since the filter should be insensitive just to the measurements of the faulty sensor, not to the all sensors including the ones working properly [6]. In contrast a matrix built of multiple scale factors might be preferred since in this method the relevant terms of the measurement noise covariance are fixed, individually.

The robust algorithm affects characteristic of the filter only when the condition of the measurement system does not correspond to the model used in the synthesis of the filter. Otherwise the UKF or EKF work with the regular algorithm.

This section gives the detailed discussion about the adaptation procedures of the EKF and UKF which makes them robust against any kind of measurement malfunction. The conventional algorithms for the UKF and EKF are not presented here for brevity and for details reader may refer to [2,5,24].

3.1 Robust Unscented Kalman Filter

It is known that the UKF innovation sequence can be determined by,

$$\boldsymbol{e}\left(k+1\right) = \boldsymbol{y}\left(k+1\right) - \hat{\boldsymbol{y}}\left(k+1\middle|k\right), \tag{11}$$

where, $\boldsymbol{y}\left(k+1\right)$ is the measurements vector and $\hat{\boldsymbol{y}}\left(k+1\middle|k\right)$ is the predicted observation vector.

The essence of the adaptation procedure against the measurement malfunctions is to compare the real and theoretical values of the innovation covariance matrix. When there is a sensor fault in the system, the real error will exceed the theoretical one. In this case we may ensure the robustness of the filter against the sensor fault by adapting the R matrix, which is a diagonal matrix, formed of the measurement process

noise covariances. The adaptation procedure basically aims at finding an appropriate multiplier for the R, such that the real and theoretical values of the innovation covariance match. As discussed, this multiplier might be either as a single factor or a matrix formed of multiple factors. In case we use a single factor, matching the real and theoretical values of the innovation covariance means that we basically increase all the terms of the R matrix and impose to the UKF that the measurements are faulty. However we do not isolate which sensor is malfunctioning. On the contrary when we use multiple scale factors we correct the necessary term of the R matrix (the term which corresponds to the sensor with the faulty measurement). In other word we make UKF disregard just the measurements of this sensor which is not reliable at that sampling time.

Single Scale Factor. As stated the essence of the adaptation is the covariance matching. For single scale factor approach we match the trace of the covariances such that

$$tr\left[e(k+1)e^T(k+1)\right] = tr\left[P_{yy}(k+1|k) + S(k)R(k+1)\right], \tag{12}$$

where $P_{yy}(k+1|k)$ is the observation covariance, $R(k+1)$ is the measurement noise covariance and $S(k)$ is the introduced single scale factor. $tr[\cdot]$ is the trace of the related matrix. We may rewrite the equation as

$$tr\left[e(k+1)e^T(k+1)\right] = tr\left[P_{yy}(k+1|k)\right] + S(k)tr\left[R(k+1)\right]. \tag{13}$$

Then, regarding $tr\left[e(k+1)e^T(k+1)\right] = e^T(k+1)e(k+1)$ the single scale factor can be obtained

$$S(k) = \frac{e^T(k+1)e(k+1) - tr\left[P_{yy}(k+1|k)\right]}{tr\left[R(k+1)\right]}. \tag{14}$$

The scale factor affects the Kalman gain as;

$$K(k+1) = P_{xy}(k+1|k)\left[P_{yy}(k+1|k) + S(k)R(k+1)\right]^{-1}. \tag{15}$$

Here $K(k+1)$ is the Kalman gain and $P_{xy}(k+1|k)$ is the cross correlation matrix.

In case of sensor fault the scalar scale factor will take a larger value and that will increase all terms of the innovation covariance. Eventually the Kalman gain will decrease and the measurements will be disregarded in the state update process (or taken into consideration with lesser weight than the regular case). In such approach the information about the faulty sensor isolation does not have any significance; all of the current information from the measurements is left out and the UKF relies mostly on the propagation information during the estimation.

Multiple Scale Factor. Firstly, we add a matrix built of multiple scale factors, $S(k)$, into the algorithm in order to tune the measurement noise covariance matrix and match the real and theoretical innovation covariances,

$$\frac{1}{\xi}\sum_{j=k-\xi+1}^{k} \boldsymbol{e}(j+1)\boldsymbol{e}^{T}(j+1) = P_{yy}(k+1|k) + S(k)R(k+1). \tag{16}$$

Here, ξ is the width of the moving window. Left hand side of the equation represents the real innovation covariance while the right hand side stands for the theoretical innovation covariance. Then, if we re-arrange the equation, it is clear that we can get the scale matrix by [25],

$$S(k) = \left\{\frac{1}{\xi}\sum_{j=k-\xi+1}^{k} \boldsymbol{e}(j+1)\boldsymbol{e}^{T}(j+1) - P_{yy}(k+1|k)\right\} R^{-1}(k+1). \tag{17}$$

In case of a measurement fault for one of the sensors then the corresponding term of the scale matrix will be a relatively larger term and that will increase the measurement noise covariance of this sensor in the R matrix. Eventually this faulty measurement will be disregarded (or regarded with a lower gain) by the filter. On the other hand, the scale matrix affects the estimation procedure only when the measurements are faulty. Otherwise, in case of normal operation, the scale matrix will be a unit matrix as $S(k) = I_{z\times z}$, where z is the size of the innovation vector.

Nonetheless, as ξ is a limited number because of the number of the measurements and the computations performed with the computer implies errors such as the approximation and round off errors; $S(k)$ matrix that is calculated by the use of Eq. (17) may not be diagonal and may have diagonal elements which are "negative" or lesser than "one". $S(k)$ matrix should be diagonal because only its diagonal terms have significance on the adaptation since each diagonal term corresponds to the noise covariance of each measurement (for the adaptation procedure $S(k)$ matrix is multiplied with the diagonal R matrix). Besides the measurement noise covariance matrix must be positive definite (that is why the multiplier $S(k)$ matrix cannot have negative terms) and also any term of this matrix cannot decrease in time for this specific problem since there is no possibility for increasing the performance of the onboard sensor (that is why the multiplier $S(k)$ matrix cannot have terms less than one).

Therefore, in order to avoid such situations, composing the scale matrix by the following rule is suggested:

$$S^{*} = diag\left(s_1^{*}, s_2^{*}, \ldots, s_z^{*}\right) \tag{18}$$

$$s_i^{*} = \max\{1, S_{ii}\} \qquad i = 1, z\,. \tag{19}$$

Here, S_{ii} represents the ith diagonal element of the matrix $S(k)$. Apart from that point, if the measurements are faulty, $S^*(k)$ will change and so affect the Kalman gain as;

$$K(k+1) = P_{xy}(k+1|k)\left[P_{yy}(k+1|k) + S^*(k)R(k+1)\right]^{-1}. \quad (20)$$

In case of any kind of malfunction, the element(s) of the scale matrix, which corresponds to the faulty component(s) of the innovation vector, increases and so the terms in the related column(s) of the Kalman gain decreases. As a consequence, the effect of the faulty innovation term on the state update process reduces and accurate estimation results can be obtained even in case of measurement malfunctions.

3.2 Robust Extended Kalman Filter

The adaptation procedure of the EKF is not different from the one for the UKF. Basically this time we apply the same method by using the EKF equations.

Firstly we can easily show that the single scale factor for the EKF is calculated as

$$S(k) = \frac{e^T(k+1)e(k+1) - tr\left[H(k+1)P(k+1|k)H^T(k+1)\right]}{tr\left[R(k+1)\right]}. \quad (21)$$

Here $P(k+1|k)$ is the predicted covariance matrix and $H(k+1)$ is the measurement matrix constituted of the partial derivatives.

Nonetheless in order to derive the equation for the multiple scale factor, similarly with the UKF, we compare the real and theoretical values of the innovation covariance matrix and add a scale matrix, $S(k)$, into the algorithm as,

$$\frac{1}{\xi}\sum_{j=k-\xi+1}^{k} \boldsymbol{e}(j+1)\boldsymbol{e}^T(j+1) = H(k+1)P(k+1|k)H^T(k+1) + S(k)R(k+1). \quad (22)$$

Then, the definition for the scale matrix is,

$$S(k) = \left\{\frac{1}{\xi}\sum_{j=k-\xi+1}^{k} \boldsymbol{e}(j+1)\boldsymbol{e}^T(j+1) - H(k+1)P(k+1|k)H^T(k+1)\right\}R^{-1}(k+1). \quad (23)$$

After that we use the Eq. (18, 19) to correct and diagonalize the scale matrix. Finally the Kalman gain is tuned as;

$$K(k+1) = P(k+1|k)H^T(k+1)\left[H(k+1)P(k+1|k)H^T(k+1) + S^*(k)R(k+1)\right]^{-1} \quad (24)$$

3.3 Fault Detection Procedure

As aforementioned, we use the robust Kalman filters only in case of the fault and in all other cases, the filters run following their regular algorithms. The fault detection is realized via a kind of statistical information. In order to achieve that, following two hypotheses may be proposed:

- γ_o ; the system is normally operating
- γ_1 ; there is a malfunction in the estimation system.

Then we may introduce the following statistical functions for the RUKF and REKF respectively,

$$\beta(k) = \boldsymbol{e}^T(k+1)\left[P_{yy}(k+1|k) + R(k+1)\right]^{-1}\boldsymbol{e}(k+1), \tag{25}$$

$$\beta(k) = \boldsymbol{e}^T(k+1)\left[H(k+1)P(k+1|k)H^T(k+1) + R(k+1)\right]^{-1}\boldsymbol{e}(k+1). \tag{26}$$

These functions have χ^2 distribution with z degree of freedom, where z is the dimension of the innovation vector.

If the level of significance, α, is selected as,

$$P\left\{\chi^2 > \chi^2_{\alpha,z}\right\} = \alpha; \qquad 0 < \alpha < 1, \tag{27}$$

the threshold value, $\chi^2_{\alpha,z}$ can be determined. Hence, when the hypothesis γ_1 is correct, the statistical value of $\beta(k)$ will be greater than the threshold value $\chi^2_{\alpha,s}$, i.e.:

$$\begin{aligned} &\gamma_0 : \beta(k) \le \chi^2_{\alpha,s} \qquad \forall k \\ &\gamma_1 : \beta(k) > \chi^2_{\alpha,s} \qquad \exists k \quad . \end{aligned} \tag{28}$$

4 Robust Kalman Filtering for Satellite Attitude Estimation

In this section, the proposed robust Kalman filtering algorithms are tested via simulations for a small satellite model. Besides, the same simulation scenarios are repeated by using the conventional UKF or EKF algorithms and the results are compared.

The simulations are realized for 7000 seconds with a sampling time of $\Delta t = 0.1\text{sec}$. This period coincides with approximately 1 orbit of the satellite. Nonetheless the orbit of the satellite is assumed as circular. Other orbit parameters are; the magnetic dipole moment of the Earth, $M_e = 7.943\ x\ 10^{15}\ Wb.m$; the Earth Gravitational constant, $\mu = 3.98601\ x\ 10^{14}\ m^3/s^2$; the orbit inclination, $i = 31^\circ$; the spin rate of the Earth, $\omega_e = 7.29\ x\ 10^{-5}\ rad/s$; the magnetic dipole tilt, $\varepsilon = 11.7^\circ$; the

distance between the centre of mass of the satellite and the Earth, $r_0 = 7450km.$ The inertia matrix for the used small satellite model is $J = diag(310 \quad 180 \quad 180) kg.m^2$.

For the magnetometer measurements, the sensor noise is characterized by zero mean Gaussian white noise with a standard deviation of $\sigma_m = 300nT$. As the filter parameters for the UKF and RUKF, κ is selected as $\kappa = -2$, where $f = 2(a+1)$ and $a = 1$. For the REKF and RUKF the size of the moving window is taken as $\xi = 30$.

The initial attitude errors in the simulations are set to 50 degrees for all three axes. Besides, the initial value of the covariance matrix is taken as $P_0 = diag\left[0.5 \quad 0.5 \quad 0.5 \quad 10^{-4} \quad 10^{-4} \quad 10^{-4}\right]$ while the process noise covariance matrix is selected as $Q = diag\left[10^{-7} \quad 10^{-7} \quad 10^{-7} \quad 10^{-12} \quad 10^{-12} \quad 10^{-12}\right]$.

Nonetheless, for the fault detection procedure, $\chi^2_{\alpha,z}$ is taken as 7.81 and this value comes from chi-square distribution when the degree of freedom is 3 and the reliability level is 95%.

Three different scenarios are taken into consideration for simulating the fault in the measurements; the continuous bias, fault of zero output and measurement-noise increment. For each scenario a series of simulations are run by the REKF, RUKF and as well the conventional EKF and UKF algorithms.

4.1 Continuous Bias Failure

In this scenario, a constant value is added to the measurements of the magnetometer aligned in the *x* axis between the 3000th and 3200th seconds for a period of 200 seconds such that;

$$B_x(\boldsymbol{q},t) = B_x(\boldsymbol{q},t) + 20000nT \qquad t = 3000 \ldots 3200 \sec$$

The constant term, selected as $20000nT$, almost doubles the magnetometer output.

In Fig.1 the attitude estimation error of the UKF and RUKF are given for the pitch angle. The RUKF that uses single scale factor is plotted with dotted line and labeled as RUKF$_s$ while the RUKF with multiple scale factors is plotted with dashed line and labeled as RUKF$_m$. Apparently, in case of fault the estimation accuracy for the conventional UKF algorithm deteriorates. The RUKF with single scale factor lessens the effect of the fault but still the filter is not fully recovered and after the measurement fault ends at 3200th sec. the RUKF$_s$ estimations show a fluctuating behavior. The reason for a filter that is not fully recovered is disregarding the measurements of all three magnetometers as a result of increasing all terms of the R matrix via multiplication with a single large scale factor (see Fig.2 for the variation of the single scale factor). Instead of isolating the faulty sensor and leaving out just its measurements, the RUKF$_s$ considers all of the measurement as faulty and throughout this period it mostly relies on the propagation values. In this case especially for a filter with relatively higher process noise covariance, Q, the estimation errors accumulate and the filter starts to diverge from the actual values. The Table 1, which gives the absolute values

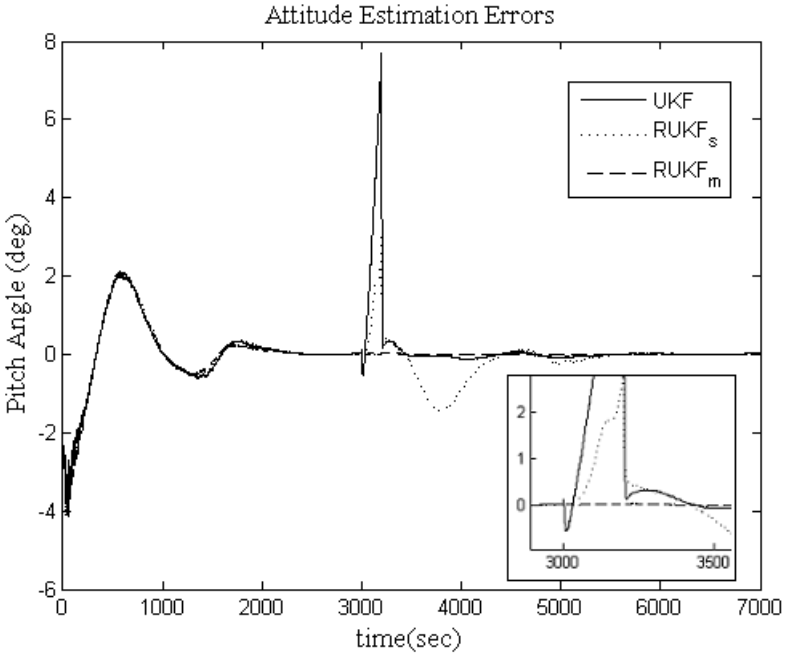


Fig. 1. Pitch angle estimation error for the UKF and RUKF in case of continuous bias

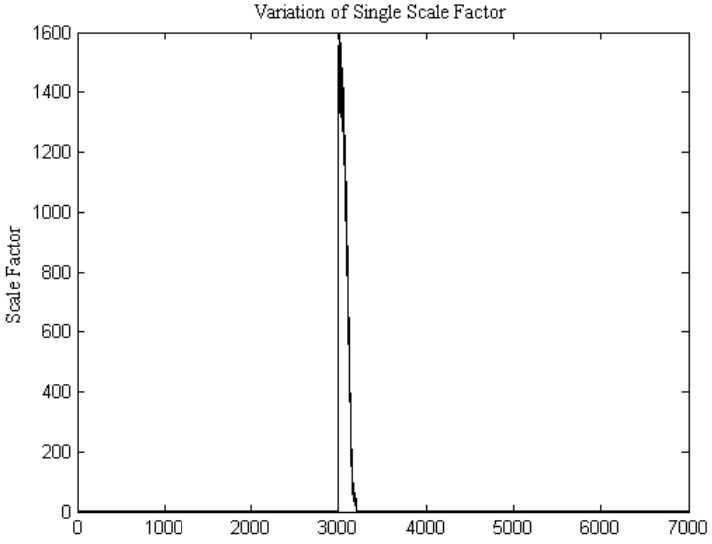


Fig. 2. Variation of the single scale factor for the RUKF in case of continuous bias

Table 1. Absolute Estimation Errors in case of continuous bias: Regular UKF, RUKF with single scale factor (SSF) and RUKF with multiple scale factor (MSF)

	Abs. Err. Values for Regular UKF		**Abs. Err. Values for RUKF with SSF**		**Abs. Err. Values for RUKF with MSF**	
Parameter	3050 s.	3150 s.	3050 s.	3150 s.	3050 s.	3150 s.
φ(deg)	10,580	10,835	0,1901	0,4936	0,0207	0,0430
θ(deg)	0,6945	5,1327	0,1116	1,8268	0,0033	0,0178
ψ(deg)	0,6131	0,8453	0,0862	2,6225	0,0041	0,0030

Table 2. Absolute Estimation Errors in case of continuous bias: Regular EKF, REKF with single scale factor (SSF) and REKF with multiple scale factor (MSF)

	Abs. Err. Values for Regular EKF		**Abs. Err. Values for REKF with SSF**		**Abs. Err. Values for REKF with MSF**	
Parameter	3050 s.	3150 s.	3050 s.	3150 s.	3050 s.	3150 s.
φ(deg)	7,1599	5,9440	3,7358	2,2637	0,5298	0,6140
θ(deg)	2,2395	2,2887	0,1460	1,8108	0,1000	0,2523
ψ(deg)	1,9778	2,1547	1,4322	5,5227	0,1291	0,0056

of error at 3050th and 3150th seconds, supports this interpretation. Clearly the estimation error for the RUKF$_s$ at the 3150th sec is higher than the one at the 3050th sec. Hence the single scale factor approach may be useful only for faults which lasts a short period. On the contrary the RUKF with the multiple scale factors does not have such limitation and keeps its estimation accuracy without being affected from the fault. An examination on the scale matrix at an instant between the 3000th and 3200th seconds of the simulation shows that the algorithm works properly; $S^* = diag(4427 \quad 1 \quad 1.59)$. Since the fault is in the measurements of the magnetometer aligned in the x axis, the correction must be applied to the first term of the R matrix as in this case. The large first diagonal term of the scale matrix decreases the terms in the first column of the Kalman gain and so the faulty innovation term (the first term of the innovation vector) is disregarded in the state update process.

In Fig.3 the estimation results for the REKF and EKF are given. Likewise the RUKF$_m$, the REKF$_m$ is not affected from the measurement fault and sustains reliable estimation results for the whole period. Although the REKF with single scale gives satisfactory results for a short period starting from 3000th sec. it deteriorates later on because of disregarding even the healthy measurements. Table 2 may be seen for more detailed investigation.

Nonetheless, a comparison between the performances of the UKF and EKF (or RUKF vs. REKF) shows that the UKF algorithms outperform the EKF algorithms in

terms of accuracy and the convergence speed. Obviously the EKF can satisfy attitude accuracy with less than 2deg error approximately 1500 seconds later than the UKF. This is an expected result for high initial attitude error as already discussed in literature [5].

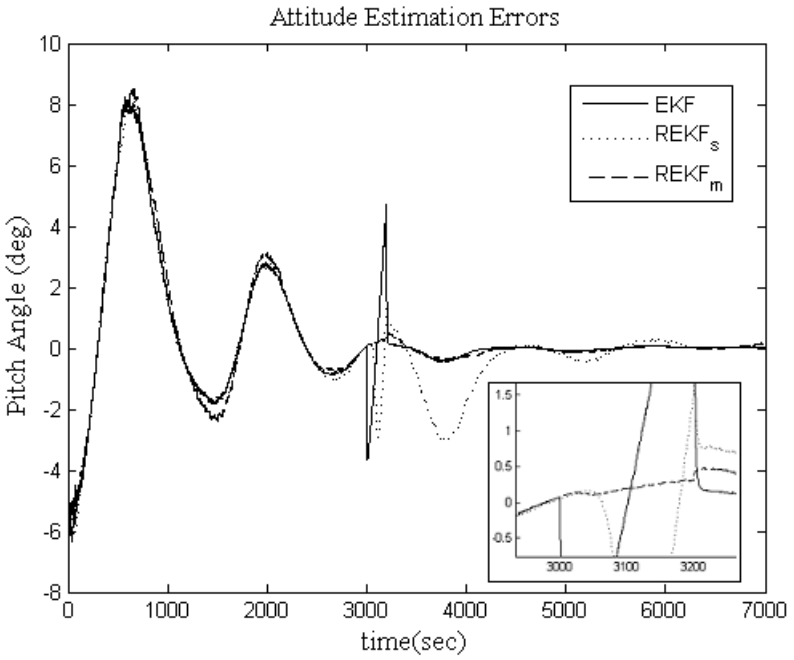


Fig. 3. Pitch angle estimation error for the EKF and REKF in case of continuous bias

4.2 Measurement Noise Increment Failure

In this scenario, the measurement noise of the magnetometer aligned in x axis is multiplied with a constant between the 3000th and 3200th seconds for a period of 200 seconds. In fault case the standard deviation of this magnetometer becomes $\sigma_{m_x}^f = 300 \times 100nT.$

In Fig.4, the estimation results for the UKF and RUKF are given. As can be observed, again the UKF fails about giving accurate estimation results in case of fault and the noisy measurements make the estimations deteriorate for a longer period than the fault itself. For this simulation case it takes 3000 seconds for the filter to satisfy accuracy less than 0.1 deg. after the fault removes. On the other hand in case of measurement noise increment both approaches for robust Kalman filtering that are scaling with single and multiple factors give accurate estimation results. Unlike the first fault case this time single scale factor works properly because of the fault's characteristic. Since the noise in the measurements is random the filter does not work continuously with the robust algorithm for whole 200 seconds and moreover the scale factor may take values that are closer to the one. In other words the filter tunes itself depending on the magnitude of the noise and does not disregard the measurements for the whole

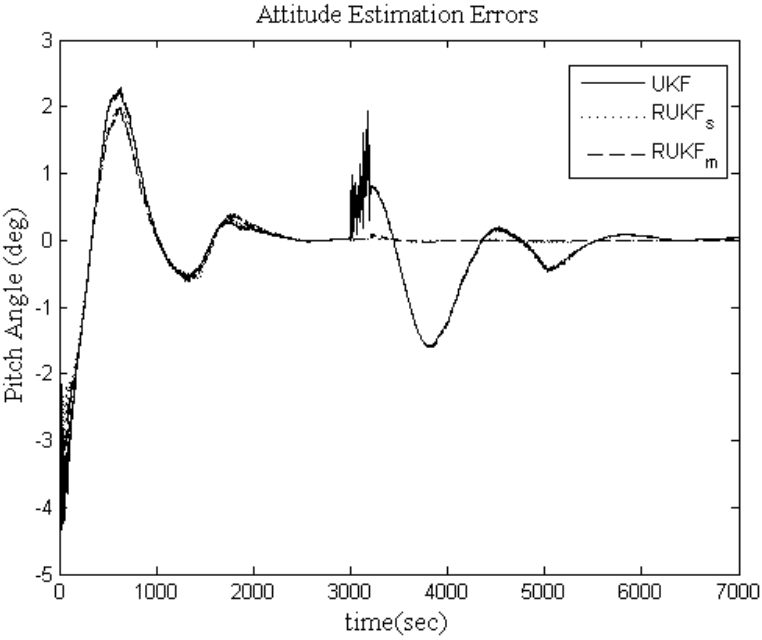


Fig. 4. Pitch angle estimation error for the UKF and RUKF in case of measurement noise increment

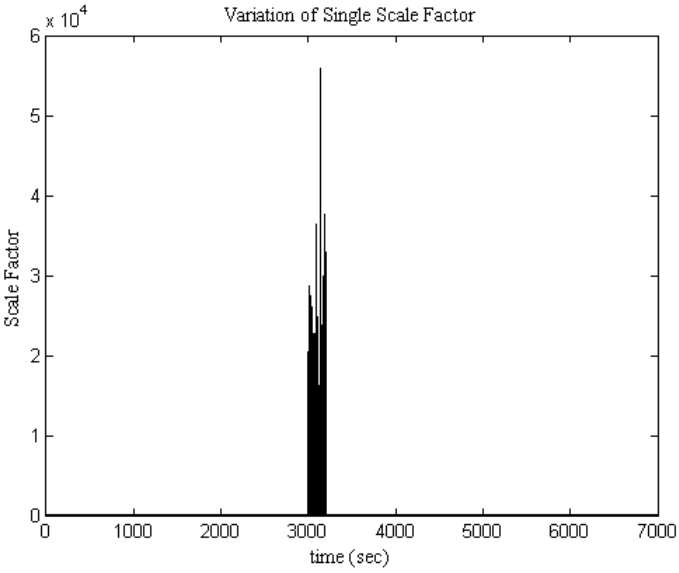


Fig. 5. Variation of the single scale factor for the RUKF in case of measurement noise increment

fault period; it only does when the $\chi^2_{\alpha,z}$ threshold is exceeded and the magnitude of the noise is high such that the scale factor takes large values. Variation of the single scale factor confirms that (Fig. 5). Hence specifically for this fault scenario the $RUKF_s$ and $RUKF_m$ do not have any significant difference in the sense of estimation accuracy (Table 3).

Table 3. Absolute Estimation Errors in case of measurement noise increment: Regular UKF, RUKF with single scale factor (SSF) and RUKF with multiple scale factor (MSF)

	Abs. Err. Values for Regular UKF		**Abs. Err. Values for RUKF with SSF**		**Abs. Err. Values for RUKF with MSF**	
Parameter	3050 s.	3150 s.	3050 s.	3150 s.	3050 s.	3150 s.
φ(deg)	3,4186	4,3199	0,0010	0,0242	0,0269	0,0412
θ(deg)	0,2853	1,3830	0,0134	0,0048	0,0062	0,0174
ψ(deg)	0,9533	0,9003	0,0026	0,0049	0,0017	0,0015

The performed simulations have proved that the REKF is also capable of overcoming the deteriorating effect of the measurement noise increment. For the simulation with the REKF, the $REKF_m$ gives slight more accurate estimations than the $REKF_s$ (Fig. 6 and Table 4).

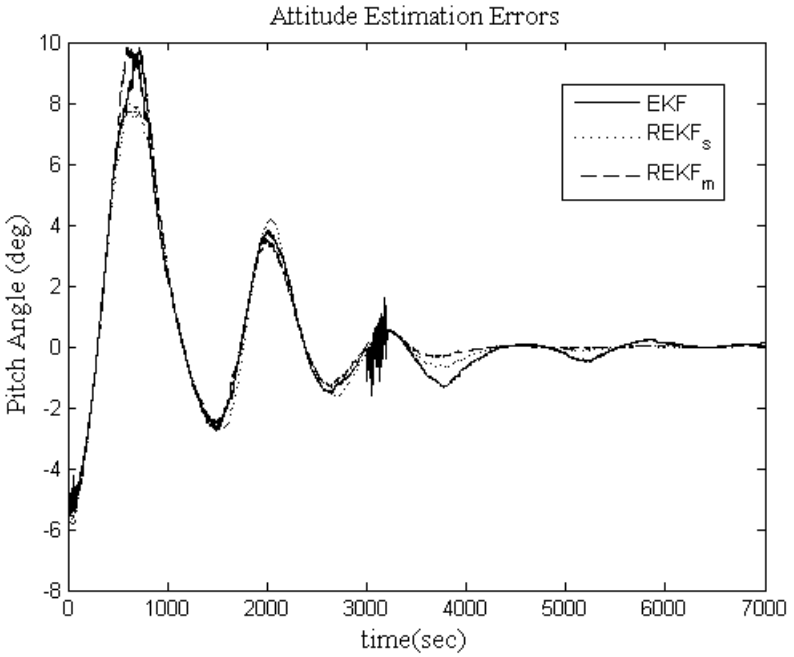


Fig. 6. Pitch angle estimation error for the EKF and REKF in case of measurement noise increment

Table 4. Absolute Estimation Errors in case of measurement noise increment: Regular EKF, REKF with single scale factor (SSF) and REKF with multiple scale factor (MSF)

	Abs. Err. Values for Regular EKF		**Abs. Err. Values for REKF with SSF**		**Abs. Err. Values for REKF with MSF**	
Parameter	3050 s.	3150 s.	3050 s.	3150 s.	3050 s.	3150 s.
φ(deg)	0,1332	0,2110	2,7516	2,3680	0,7954	0,9315
θ(deg)	1,1840	0,1027	0,1554	0,6464	0,1114	0,3182
ψ(deg)	1,5309	0,7592	0,6777	0,3859	0,2131	0,0266

4.3 Zero Output Failure

The third failure case, which is fault of zero output, is simulated by simply making the measurement output of one of the magnetometers zero so it measures $0nT$ for 200 seconds between the 3000^{th} and 3200^{th} seconds. In order to test the algorithm this time the fault is implemented to the magnetometer aligned in the z axis:

$$B_z(\boldsymbol{q},t) = 0 + \eta_{1z} \qquad t = 3000 \ldots 3200 \,\text{sec}$$

In Fig.7 the estimation results for the UKF and RUKF are given. Obviously, same as the first simulation scenario, the UKF cannot achieve accurate estimation whereas the RUKF with the single scale factor can overcome the fault only for a short period. Because of taking none of the measurements into consideration the RUKF_s estimations get worse when the robust algorithm runs longer than 50 seconds. The single scale factor behaves in a similar manner with its trend for the continuous bias fault scenario (Fig. 8). Moreover, the results show us when the filter is not robust the zero output failure has a high detractive impact on the estimation accuracy that lasts for a very long period. Even though the filter's response may vary when it is designed with different parameters (such as the Q matrix) simulations show that a fault may affect the filter for a longer period than its length. Therefore if the magnetometer measures 0 even for just few seconds, it is not possible to compensate that with a filter other than the robust ones. In this sense the estimation results for the RUKF_m clearly signify the importance of using the proposed algorithm. The RUKF_m is not affected from the fault and can perform accurate estimation even when the fault lasts long by simply disregarding the measurements of the faulty magnetometer and working on the basis of the measurements from two properly operating magnetometers (Table 5 may be seen for further examination). The sample for the multiple scale factors in case of fault validates that the RUKF_m disregards the measurements of the magnetometer aligned in the z axis as it supposed to be $S^* = diag(1.44 \quad 1 \quad 4024)$.

In Fig.9 and Table 6 the estimation results for the EKF and REKF are given. Same as the UKF, the EKF fails at giving accurate estimation results for a longer period than the fault itself whereas the REKF_m is superior to REKF_s regarding the estimation accuracy.

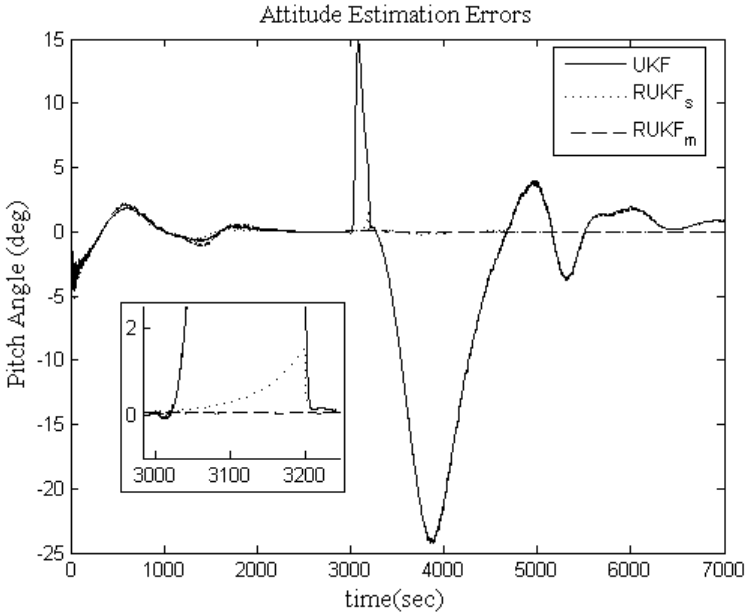


Fig. 7. Pitch angle estimation error for the UKF and RUKF in case of zero output failure

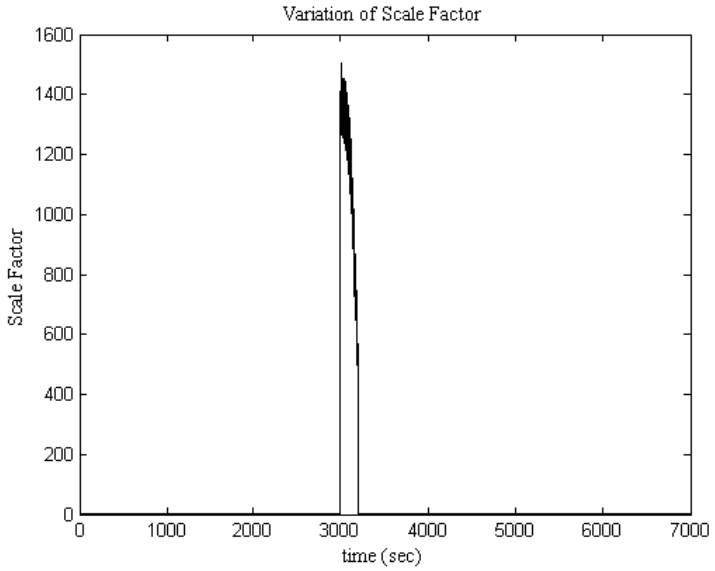


Fig. 8. Variation of the single scale factor for the RUKF in case of zero output failure

Table 5. Absolute Estimation Errors in case of zero output failure: Regular UKF, RUKF with single scale factor (SSF) and RUKF with multiple scale factor (MSF)

	Abs. Err. Values for Regular UKF		**Abs. Err. Values for RUKF with SSF**		**Abs. Err. Values for RUKF with MSF**	
Parameter	3050 s.	3150 s.	3050 s.	3150 s.	3050 s.	3150 s.
φ(deg)	9,7561	42,492	0,0321	0,2176	0,0892	0,0567
θ(deg)	7,0059	8,1779	0,0293	0,3733	0,0521	0,0576
ψ(deg)	9,6897	41,673	0,0114	0,1908	0,0163	0,0191

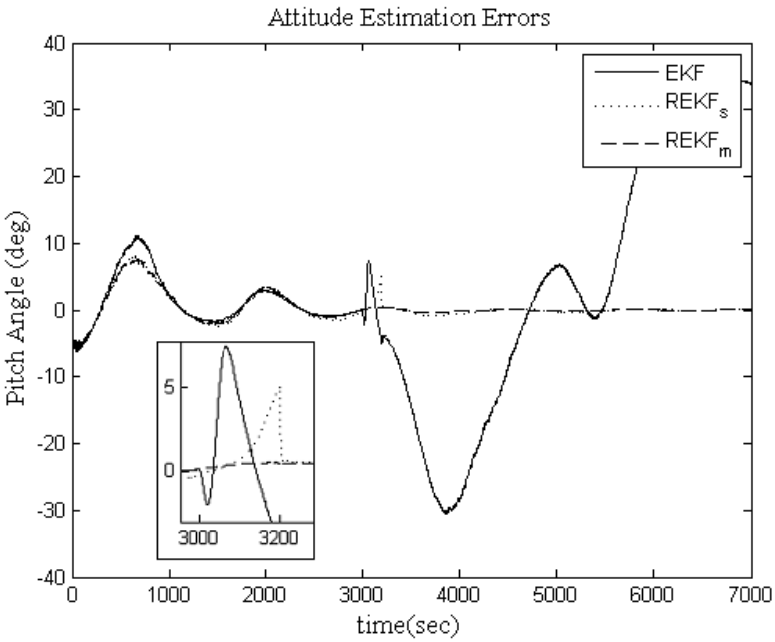


Fig. 9. Pitch angle estimation error for the EKF and REKF in case of zero output failure

Table 6. Absolute Estimation Errors in case of zero output failure: Regular EKF, REKF with single scale factor (SSF) and REKF with multiple scale factor (MSF)

	Abs. Err. Values for Regular EKF		**Abs. Err. Values for REKF with SSF**		**Abs. Err. Values for REKF with MSF**	
Parameter	3050 s.	3150 s.	3050 s.	3150 s.	3050 s.	3150 s.
φ(deg)	53,686	148,50	2,8558	1,8026	1,5884	1,2380
θ(deg)	5,2576	0,8919	0,0760	2,2197	0,2049	0,4120
ψ(deg)	30,024	45,173	0,9518	3,6684	0,3210	0,0242

5 Conclusion

In this paper, a covariance scaling based Robust Kalman filtering method is proposed for the attitude estimation problem and applied to the both EKF and UKF. The simulation results show that both the RUKF and REKF perform well when a specific measurement fault is the point at issue. On the other hand, the conventional filters (the UKF and EKF) fails at giving accurate estimation results for the period of the fault and as well for some additional time that is necessary for filter to converge again. Moreover, the RUKF outperforms all other filters including the REKF for all simulation cases when the initial attitude error is high as in case.

Acknowledgments. This work was supported in part by Japanese Government with MONBUKAGAKUSHO scholarship and also by Japan Aerospace Exploration Agency (JAXA) with a research grant.

References

1. Lefferts, E.J., Markley, F.L., Shuster, M.D.: Kalman Filtering for Spacecraft Attitude Estimation. Journal of Guidance, Control and Dynamics 5, 417–422 (1982)
2. Markley, F.L., Crassidis, J.L., Cheng, Y.: Nonlinear Attitude Filtering Methods. In: AIAA Guidance, Navigation, and Control Conference and Exhibit, San Francisco (2005)
3. Julier, S.J., Uhlmann, J.K., Durrant-Whyte, H.F.: A New Approach for Filtering Nonlinear Systems. In: American Control Conference, Seattle, vol. 3, pp. 1628–1632 (1995)
4. Soken, H.E., Hajiyev, C.: UKF-Based Reconfigurable Attitude Parameters Estimation and Magnetometer Calibration. IEEE Transactions on Aerospace and Electronic Systems 48, 2614–2627 (2012)
5. Crassidis, J.L., Markley, F.L.: Unscented Filtering for Spacecraft Attitude Estimation. Journal of Guidance, Control and Dynamics 26, 536–542 (2003)
6. Soken, H.E., Hajiyev, C.: Pico Satellite Attitude Estimation via Robust Unscented Kalman Filter in the Presence of Measurement Faults. ISA Transactions 49, 249–256 (2010)
7. Mehra, R.K.: On the Identification of Variances and Adaptive Filtering. IEEE Transactions on Automatic Control 15, 175–184 (1970)
8. Mehra, R.K.: Approaches to Adaptive Filtering. IEEE Transactions on Automatic Control 17, 693–698 (1972)
9. Mohamed, A.H., Schwarz, K.P.: Adaptive Kalman Filtering for INS/GPS. Journal of Geodesy 73, 193–203 (1999)
10. Geng, Y., Wang, J.: Adaptive Estimation of Multiple Fading Factors in Kalman Filter for Navigation Applications. GPS Solutions 12, 273–279 (2008)
11. Kim, K.H., Lee, J.G., Park, C.G.: Adaptive Two-Stage Kalman Filter in the Presence of Unknown Random Bias. International Journal of Adaptive Control and Signal Processing 20, 305–319 (2006)
12. Odelson, B.J., Rajamani, M.R., Rawlings, J.B.: A new Autocovariance Least-Squares Method for Estimating Noise Covariances. Automatica 42, 303–308 (2006)
13. Dunik, J., Simandl, M., Straka, O.: Methods for Estimating State and Measurement Noise Covariance Matrices: Aspects and Comparison. In: 15th IFAC Symposium on System Identification, Saint-Malo, pp. 972–977 (2009)

14. Kim, K.H., Lee, J.G., Park, C.G.: Adaptive Two-Stage Extended Kalman Filter for a Fault-Tolerant INS-GPS Loosely Coupled System. IEEE Transactions on Aerospace and Electronic Systems 45, 125–137 (2009)
15. Han, J., Song, Q., He, Y.: Adaptive Unscented Kalman Filter and Its Applications in Control. In: Moreno, V.M., Pigazo, P. (eds.) Kalman Filter: Recent Advances and Applications, pp. 1–24. InTeh Publishing, Rijeka (2009)
16. Liu, J., Lu, M.: An Adaptive UKF Filtering Algorithm for GPS Position Estimation. In: 5th International Conference on Wireless Communications, Networking and Mobile Computing, Beijing, pp. 1–4 (2009)
17. Shi, Y., Han, C., Liang, Y.: Adaptive UKF for Target Tracking with Unknown Process Noise Statistics. In: 12th International Conference on Information Fusion, Seattle, pp. 1815–1820 (2009)
18. Jwo, D., Wang, S.: Adaptive Fuzzy Strong Tracking Extended Kalman Filtering for GPS Navigation. IEEE Sensors Journal 7, 778–789 (2007)
19. Jwo, D., Tseng, C.: Fuzzy Adaptive Interacting Multiple Model Unscented Kalman Filter for Integrated Navigation. In: 18th IEEE International Conference on Control Applications Part of 2009 IEEE Multi-Conference on Systems and Control, Saint Petersburg, pp. 1684–1689 (2009)
20. Jwo, D., Chung, F.: Fuzzy Adaptive Unscented Kalman Filter for Ultra-Tight GPS/INS Integration. In: The International Symposium on Computational Intelligence and Design, Hangzhou, pp. 229–235 (2010)
21. Wertz, J.R.: Spacecraft Attitude Determination and Control. Kluwer Academic Publishers, Dordrecht (1988)
22. Alonso, R., Shuster, M.D.: Complete Linear Attitude-Independent Magnetometer Calibration. Journal of Astronautical Science 50, 477–490 (2002)
23. Sekhavat, P., Gong, Q., Ross, I.M.: Unscented Kalman Filtering: NPSAT I Ground Test Results. In: AIAA Guidance, Navigation, and Control Conference and Exhibit, Colorado (2006)
24. Filipski, M.N., Varatharajoo, R.: Evaluation of a Spacecraft Attitude and Rate Estimation Algorithm. Aircraft Engineering and Aerospace Technology 82, 184–193 (2010)
25. Soken, H.E., Hajiyev, C., Sakai, S.: Robust Kalman Filtering for Small Satellite Attitude Estimation in the Presence of Measurement Faults. European Journal of Control 20, 64–72 (2014)

Spacecraft Attitude Determination with Sun Sensors, Horizon Sensors and Gyros: Comparison of Steady-State Kalman Filter and Extended Kalman Filter

Vaibhav V. Unhelkar and Hari B. Hablani

Indian Institute of Technology Bombay, Powai, Mumbai, India
v.unhelkar@iitb.ac.in, hbhablani@aero.iitb.ac.in

Abstract. Attitude determination, along with attitude control, is critical to functioning of every space mission. In this paper, we investigate and compare, through simulation, the application of two autonomous sequential attitude estimation algorithms, adopted from the literature, for attitude determination using attitude sensors (sun sensor and horizon sensors) and rate-integrating gyros. The two algorithms are: the direction cosine matrix (DCM) based steady-state Kalman Filter, and the classic quaternion-based Extended Kalman Filter. To make the analysis realistic, as well as to improve the attitude determination accuracies, detailed sensor measurement models are developed. Modifications in the attitude determination algorithms for estimation of additional states to account for sensor biases and misalignments are presented. A modular six degree-of-freedom closed-loop simulation, developed in house, is used to observe and compare the performances of the attitude determination algorithms.

Keywords: Attitude Determination, Kalman Filter, Horizon Sensors.

1 Introduction

Maintaining a desired orientation in space, with a specified level of accuracy, is a mission requirement for every spacecraft. Attitude determination along with attitude control is responsible for satisfying this requirement. Based on the function of the spacecraft the level of pointing accuracy required varies. During the past four decades, extensive research has been done in the area of spacecraft attitude determination. Various algorithms exist in the literature, with varied level of complexity and applicability [1]. The choice of algorithm for a mission depends on pointing accuracy requirements, the type of sensors available and capability of the on-board computer.

Here, we consider analysis of the attitude determination subsystem for Low Earth Orbit satellites using sun sensors, horizon sensors and fiber optic gyros to achieve three-axis pointing accuracy requirement of 0.1 deg. To make the analysis of these algorithms realistic, as well as to improve the accuracy of the

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_22

attitude determination algorithms, detailed models of measurements with sun sensor, horizon sensor and gyros are utilized. Instead of treating measurement errors as white noise, an effort is made to develop realistic systemic errors and noise models. For instance, horizon sensor modeling includes errors arising from Earth's oblateness, atmospheric radiance and sensor electronics.

Further, a modular six degree-of-freedom closed-loop MATLAB-Simulink® simulation is developed, which comprises true attitude kinematics and dynamics, sensor models, orbit propagation, attitude determination and control. This simulation setup is used to compare the performances of the attitude determination algorithms developed. The modular design of simulation allows a straight forward approach to include or exclude sensors and to test different attitude estimators and controllers.

Two attitude estimation algorithms are considered, which mainly vary with respect to their attitude representations and computational requirements. First, a steady-state Kalman Filter, adopted from [2], is analyzed and simulated to obtain estimates of satellite attitude and gyro drift rate bias. Direction cosine matrix and Euler angles are used to represent the attitude, for ease of physical interpretation. The steady-state formulation does away with expensive matrix covariance computations, but if dictated by mission requirements the formulation can be easily modified to its recursive gain counterpart. This is followed up with the analysis and simulation of the classic quaternion-based Extended Kalman Filter of Lefferts, Markley and Shuster [3] for the on-board sensor suite considered . The classic EKF is modified to estimate exponentially-correlated radiance error in horizon sensor measurements. Lastly, the effect of sensor misalignments on attitude estimation performance is assessed through simulations, and Pittelkau's remedy to mitigate the performance degradation due to the misalignments, i.e., alignment Kalman filter [4], is presented.

Reference Frames

We consider three frames of reference. The Earth-centered Inertial (ECI) frame is an inertial frame with origin at the Earth's center. The coordinate axes $\mathbf{x}_I$ and $\mathbf{z}_I$ point towards the direction of the vernal equinox and the north pole, respectively, and $\mathbf{y}_I$ completes the right-handed coordinate system. The Local Vertical Local Horizontal (LVLH) frame describes the current orbit frame of the satellite, and has its origin at the center of mass of the satellite. The coordinate axis $\mathbf{z}_L$ points towards the center of the earth (direction of the nadir), $\mathbf{y}_L$ points opposite to the satellite's angular momentum, and $\mathbf{x}_L$ completes the right-handed triad. The instantaneous LVLH frame is used as reference to measure the local attitude of the satellite. Body frame is an orthogonal coordinate system fixed to the satellite body with origin at its center of mass.

The symbols $\hat{\mathbf{x}}$ and $\mathbf{x}^{\times}$ denote the estimate and the cross-product matrix associated with $\mathbf{x}$, respectively.

2 Sensor Models

This section briefly describes the sensor models for the sensor suite considered. As these sensors have been used in various space missions, sufficient technical research exists regarding their characteristics and performance [5,6]. However, in spite of the literature, usually additional analysis is required to arrive at customized and realistic model of these sensors (especially so for the attitude sensors) based on the sensor configuration of a particular spacecraft.

2.1 Rate-Integrating Gyros

Gyros are inertial sensors which measure change in attitude as opposed to the absolute attitude. Gyros are of various types, such as, mechanical gyros, ring laser gyros, fiber optic gyros, and can be classified based on their accuracy, mechanisms and form of output. Rate gyros measure angular rate directly, while the rate-integrating gyros (RIG) measure integrated angular rate [1]. In our attitude determination study, we consider application of fiber optic rate-integrating gyros, which provide incremental angle vector.

The measurement equation which, essentially, corresponds to the inertial rate of the body expressed in body frame $\boldsymbol{\omega}_{BI}^{B}$ corrupted by various noise sources is given as,

$$\boldsymbol{\omega}_m = \mathrm{A}_{malgn}\boldsymbol{\omega}_{BI}^{B} + \mathbf{b} + \boldsymbol{\eta}_g \tag{1}$$

$$\dot{\mathbf{b}} = \boldsymbol{\eta}_u \tag{2}$$

where, the subscript 'm' denotes the measured rate, $\mathbf{b}$ denotes the gyro drift-rate bias, A_{malgn} denotes the misalignment and scale factor matrix matrix, and $\boldsymbol{\eta}_g$ (random-walk rate vector) and $\boldsymbol{\eta}_u$ (drift acceleration) are two continuous time white noise vectors. These equations when converted to discrete time yield [7],

$$\Delta\boldsymbol{\varphi} = \Delta\boldsymbol{\theta} + T_{gyro}\mathbf{b}_k + \boldsymbol{\beta}_k + \boldsymbol{\nu}_{q,k} \tag{3}$$

$$\text{where} \quad \boldsymbol{\omega}_{in} = \mathrm{A}_{malgn}\boldsymbol{\omega}_{BI}^{B} \tag{4}$$

$$\Delta\boldsymbol{\theta} = \int_{kT_{gyro}}^{(k+1)T_{gyro}} \boldsymbol{\omega}_{in}(t)dt \tag{5}$$

The term $\Delta\boldsymbol{\theta}$ expresses the true change in the spacecraft attitude, whereas the $\Delta\boldsymbol{\varphi}$ denotes the rate-integrating gyro output during one gyro sample period (T_{gyro}). The zero-mean noise due to $\boldsymbol{\eta}_g(t)$ and $\boldsymbol{\eta}_u(t)$ is expressed by $\boldsymbol{\beta}_k$. The variance of $\boldsymbol{\beta}_k$ is a 3×3 diagonal matrix for which the diagonal element is $\sigma_\beta^2 = \sigma_v^2 T_{gyro} + \sigma_u^2 T_{gyro}^3/3$, where σ_v^2 (rad^2/s) and σ_u^2 (rad^2/s^3) are power spectral densities of the scalar elements of $\boldsymbol{\eta}_g$ and $\boldsymbol{\eta}_u$, respectively. The gyro drift-rate bias evolves in discrete-time as,

$$\mathbf{b}_k = \mathbf{b}_{k-1} + \boldsymbol{\alpha}_k \tag{6}$$

where, $\boldsymbol{\alpha}_k$ is a zero-mean discrete random-rate noise vector, with variance of each element being $\sigma_\alpha^2 = \sigma_u^2 T_{gyro}$ [2]. The term $\boldsymbol{\nu}_{q,k}$, a discrete-time white noise with

variance σ_e^2, represents the quantization error of the gyro. Lastly, scale factor and misalignment errors occur due to mounting errors of the system, or due to intrinsic sensor errors. We use the notation of Pittelkau [4] to define the scale factor and misalignment matrix, where λ_j denotes the scale factor error, while δ_{ij} denotes the axis misalignment,

$$\mathrm{A}_{malign} = \begin{bmatrix} (1+\lambda_x) & -\delta_{yz} & \delta_{zy} \\ \delta_{xz} & (1+\lambda_y) & -\delta_{zx} \\ -\delta_{xy} & \delta_{yx} & (1+\lambda_z) \end{bmatrix} \tag{7}$$

For simulations presented later, we have used parameters of a fiber optic gyro (Table 1). Initial value of gyro drift rate bias is taken as 0.05 deg/hr.

Table 1. Gyro Parameters

Parameter	Value	Units
σ_v	7.27	μrad/s$^{1/2}$
σ_u	3×10^{-4}	μrad/s$^{3/2}$
σ_e	15	μrad

2.2 Sun Sensors

Sun sensors measure the direction of the sun relative to the spacecraft, and provide an attitude reference. These sensors measure the impinging solar energy on their surface and determine the angle made by the sun with respect to the sensor, which in turn is used to arrive at the sun vector. The sun vector, along with a sun model, can also be used to determine the yaw attitude of the spacecraft. However, as the sun sensors measure the radiation from the sun they can function only in the sun-lit phase of the orbit.

The model of sun sensor being used for a particular satellite depends on the type of the sensor, its positioning and error characteristics. For our analysis, we assume six solar cells (represented by blue circles in Fig. 1) placed on each side of the satellite. This configuration and the sun vector construction algorithm is adapted from that of the Pratham student-satellite [8], and has the advantage

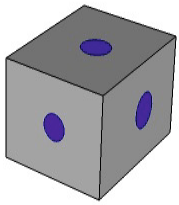


Fig. 1. Sun Sensor Configuration

that in the sun-lit phase the sun is usually visible to three of the sun sensors. Each of the individual cell measures, with some noise, the cosine of the angle made between the Sun vector ($\mathbf{s}$) and the solar cell vector ($\mathbf{c_i}$). The measurement model for one individual solar cell is given by

$$u_i = \mathbf{c_i}_B \cdot \mathbf{s}_B + \nu_c, \forall i = 1:6 \tag{8}$$

The noise in the measurement arises due to the sensor mechanism, electronics, and quantization, the strength of which depends on the type of the sensor. Lastly, each of the cell has a limited field-of-view, hence the measurement model equation (Eq. 8) is valid only within the FOV of each sensor. The sensor parameters for simulation have been taken corresponding to that of a single-axis solar cell [6], as listed in Table 2.

Table 2. Sun Sensor Parameters

Parameter	Value	Units	Remark
σ_c	0.05	deg	Standard Deviation of ν_c
FOV	80	deg	Field-of-View (conical)
T_c	1	s	Sample Time

These measurements are then used to determine the Sun vector through elementary linear transformation [8], which is then used in the attitude determination formulations. The measured Sun vector can be represented as,

$$\mathbf{s}_{m,B} = \mathbf{s}_B + \tilde{\mathbf{s}}_{m,B} \tag{9}$$

where, $\tilde{\mathbf{s}}_{m,B}$ is a random zero-mean Gaussian variable, with noise covariance, $\mathrm{R}_c = \sigma_c^2 \mathbf{I}_{3\times 3}$.

2.3 Horizon Sensors

Horizon sensors, essentially, measure the direction of Earth (nadir) by observing the shape of the Earth's limb as seen from the spacecraft and comparing it with a modeled shape, to arrive at the spacecraft attitude. These can be classified into two types - scanning and static - which differ in their mechanism to sense the Earth's horizon, error characteristics and field-of-view. The horizon sensors measure roll and pitch angles. Thus, in the sun-lit phase, along with the sun sensor the horizon sensor provide complete attitude information. We consider scanning-type horizon sensors, and follow the work presented in [9,10,11] for their analysis and error characteristics.

Scanning type horizon sensors consist of moving optical scanners mounted on the spacecraft, which scan and detect the Earth's limb. The sensor electronics converts the radiation information to scan width of Earth's limb. This measurements of scan width are then processed to obtain measurements of roll and pitch. The scanning type sensors though relatively less accurate have a much larger field-of-view than the static horizon sensors.

Mathematical Model. The horizon sensor configuration considered involves two optical scanners mounted on opposite sides of the spacecraft in the pitch-yaw plane. These two scanners measure four semi-scan angles - ($\theta_{\omega 0R}$,$\theta_{\omega 1R}$,$\theta_{\omega 0L}$, and $\theta_{\omega 1L}$) - corresponding to the space-to-earth and earth-to-space transition of left and right scanners. For a detailed description of the sensor configuration the reader is directed to [11]. In order to generate the scan width measurements based on the semi-scan angles due to an oblate earth model, we use the equations specified in [11] and [12], along with the model of oblate earth by Liu [5], with appropriate modifications to suit our coordinate convention. Once the four semi-scan angles are obtained, the roll and pitch angles are obtained as follows [11],

$$\phi = \frac{1}{4} K \left(\theta_{\omega 0R} + \theta_{\omega 1R} - \theta_{\omega 0L} - \theta_{\omega 1L}\right) \tag{10}$$

$$\theta = \frac{1}{4} \cos\xi \left(\theta_{\omega 0R} - \theta_{\omega 1R} + \theta_{\omega 0L} - \theta_{\omega 1L}\right) \tag{11}$$

where, K and ξ are known parameters which depend on spacecraft's altitude and sensor hardware parameters.

The horizon sensor measurements are affected both by the noises in electronics and the errors arising from the limitation in accurately modeling the Earth's limb. The major sources of horizon sensor errors are Earth's oblateness, variation of Earth's radiation, electronic noise, quantization error, and sensor bias and misalignment [9]. Errors due to Earth's oblateness are systemic and highly predictable. Thus, by using an appropriate model of Earth's shape these errors can be largely eliminated. We follow the approach described in [11] to account for errors due to oblateness. Earth's radiation suffers from seasonal and latitudinal variations, which are partly systemic and partly stochastic [13]. The systemic variations can be largely corrected based on modeling of sensor optics and Earth's radiation. The available horizon sensor hardware largely compensates for these systemic variations internally; however, stochastic errors of the order of 0.06 deg still persist post corrections [9]. Sensor bias, misalignment and electronic noise are inherent sensor errors that arise due to the sensor hardware.

For simulation, the roll and pitch angles obtained from Eq. (10-11) are corrupted with bias, white noise (to simulate the electronic noise) and noise due to

radiance. Radiance models exist in literature based on analytical modeling of the Earth's atmosphere and data observed from various space missions [10,14]. In order to obtain the radiance errors in roll and pitch, these radiance models are then used along with the model of sensor optics, signal processing and electronics. However, this procedure involves detailed knowledge of the sensor hardware. Hence, we approximate the noise due to radiance as an exponentially correlated noise in both the roll and pitch axes [15]. This model, though approximate, models the radiance errors as varying with time with a specified steady-state RMS value. The discrete time equation for horizon sensor noise due to radiance variation (w_ϕ and w_θ) is thus given as,

$$w_{\phi,k+1} = aw_{\phi,k} + l\sqrt{1-a^2}\nu_k \tag{12a}$$

$$w_{\theta,k+1} = aw_{\theta,k} + l\sqrt{1-a^2}\nu_k \tag{12b}$$

where l denotes the steady-state RMS value of radiance variation error and is taken as 0.06 deg [9], ν_k is the discrete time white noise with variance equal to unity, and the parameter a is defined in terms of the horizon sensor sample period (T_{hs}) and the correlation time of exponentially auto-correlated noise (τ_w),

$$a = \exp\left[\frac{-T_{hs}}{\tau_w}\right] \tag{13}$$

To account for the cyclic variation with latitude occuring every orbit, the parameter τ_w is selected as one-eighth of the orbital period. Sensor noise due to radiance variation for a sample simulation run is illustrated in Fig. 2. Errors due to sensor bias and static misalignment errors are added as constant bias (b_ϕ and b_θ) of magnitude 0.02 deg in each axis. Other random errors arising from the sensor hardware are modeled as discrete white (ν_ϕ and ν_θ) noise with its 3σ value as 0.042 deg. The measurements are sampled every 1 s. The values of parameters describing sensor bias and random errors are chosen based on brochure of Sodern Horizon Sensors [16].

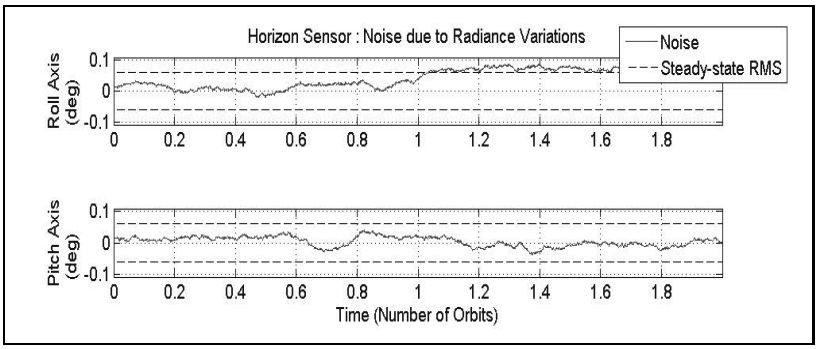


Fig. 2. Horizon sensor noise due to Radiance Variation

Horizon sensor measurement equations containing error from all the sources considered above can be written as,

$$\phi_{m,hs} = \left[\frac{1}{4}K\left(\theta_{\omega 0R} + \theta_{\omega 1R} - \theta_{\omega 0L} - \theta_{\omega 1L}\right)\right] + w_\phi + b_\phi + \nu_\phi \tag{14}$$

$$\theta_{m,hs} = \left[\frac{1}{4}\cos\xi\left(\theta_{\omega 0R} - \theta_{\omega 1R} + \theta_{\omega 0L} - \theta_{\omega 1L}\right)\right] + w_\theta + b_\theta + \nu_\theta \tag{15}$$

Oblateness Corrections. The horizon sensor errors described above contain both systemic as well as stochastic terms. To improve the accuracy of measurements, the systemic errors are predicted through analytical models, and subtracted from the measurements to mitigate the systemic errors. Earth's oblateness is one of the major systemic error which can be largely eliminated by proper modeling. Following the formulation of [11] oblateness corrections are calculated.

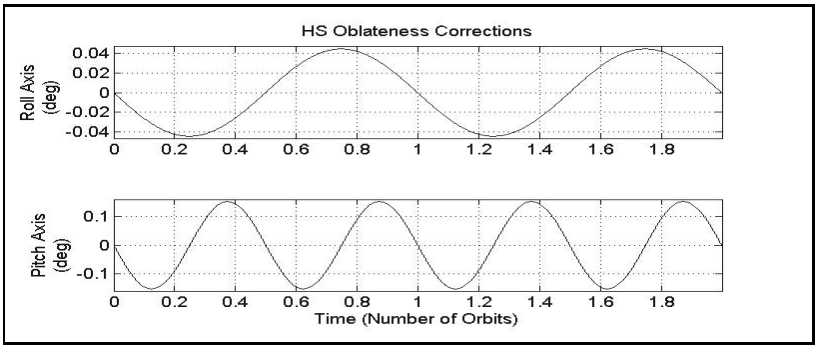


Fig. 3. Horizon Sensor : Oblateness Corrections

These oblateness corrections are added to the sensor measurements to remove the noise due to the Earth's oblateness. The variation of oblateness correction for orbit considered in our simulation is illustrated in Fig. 3. Corrections for other systemic errors, such as bias and residual noise due to radiance variation, can be done through on-board estimation techniques [17] or through post-processing of measurement residuals [9].

Horizon Sensor Parameters. A horizon sensor with clockwise scanning pattern is considered with realistic sensor parameters based on [9,16]. Table 3 lists the sensor parameters used during the simulation of the scanning type horizon sensors. Note that the sensor field-of-view is limited.

Table 3. Scanning Horizon Sensor Parameters

Parameter	Value	Units	Remark
ξ	20	deg	Canting Angle
δ_c	45	deg	Semi-cone Angle
FOV	25	deg	Field-of-View
l	0.06	deg	Radiance noise RMS (each axis)
b_{hs}	0.02	deg	Bias (each axis)
$3\sigma_{hs}$	0.042	deg	White noise (each axis)
T_{hs}	1	s	Sample Time

3 Development of Simulation

In order to validate and compare the attitude determination formulations, a six degree-of-freedom closed-loop simulation setup (Fig. 4), similar to that of Pratham student-satellite [8], is developed using MATLAB®-Simulink. A controller is included to observe the pointing accuracy obtained by the attitude determination and control sub-system.

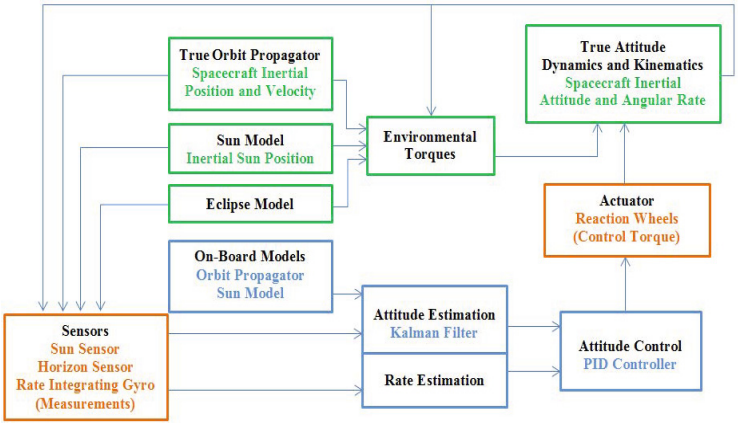


Fig. 4. Overview of Simulation

The design of the simulation is kept modular, so that it can be utilized to test different sets of sensors and attitude determination algorithms. The overall simulation sample rate is selected as 20 Hz, as it is sufficient to capture the system dynamics in simulation. Further, various sub-systems are simulated at different sample rates in order to account for their different sample times (see Table 4).

Standard equations for various simulation blocks as shown in Fig. 4 have been used [5]. Environmental torques due to gravity-gradient, solar radiation pressure and aerodynamic disturbance are considered. A sun-synchronous circular orbit

Table 4. Simulation rates of different sub-systems

Sub-system	Simulation Rate
True Attitude Dynamics	20 Hz
True Orbit Propagation	20 Hz
Sensors (Gyro)	10 Hz
Sensors (SS & HS)	1 Hz
On-board Models	10 Hz
Attitude Estimator	10 Hz
Controller	10 Hz

with altitude of 720 km and 98.28 deg inclination is considered, which is similar to that of the Oceansat-2 satellite [18]. Earth's gravity model which incorporates terms due to the Earth's oblateness upto J_2 zonal harmonics is used for orbit propagation [6]. The complete simulation setup has been validated with the help of conservation of angular momentum check in absence of external torques. Other models being standard, we here describe briefly the controller and rate estimation filter employed in our simulation.

3.1 Controller

A controller is required for the purpose of simulation, in order to observe the closed-loop performance of the attitude estimator. Here, we use a basic PID controller for our simulations. The gains of the PID controller are selected to obtain the desired damping ratio (ζ) of 0.707, and natural frequency (ω_n) corresponding to a time period of half minute, i.e., (0.5)(60)s. The parameter δ influences the integral gain, and helps to eliminate the steady-state error of the controller. The values of the normalized controller gains, for the listed specifications, are listed in Table 6.

Table 5. Controller Parameters

δ	0.7
ζ	0.707
ω_n	$\frac{2\pi}{(0.5)(60)} = 0.2094$

Table 6. Normalized Controller Gains

a_d	$(2+\delta)\zeta\omega_n$	0.3998
a_p	$\omega_n^2(1+2\delta\zeta^2)$	0.0746
a_i	$\delta\zeta\omega_n^3$	0.0045

Based on the values of controller gains the control torque vector is determined using the attitude and rate error, and satellite's moment of inertia, $\boldsymbol{I}$.

$$\mathbf{g}_{con} = \boldsymbol{I}\left(a_p\boldsymbol{\theta}_{err} + a_i\int \boldsymbol{\theta}_{err}dt + a_d\boldsymbol{\omega}_{err}\right) \tag{16}$$

$$\boldsymbol{\theta}_{err} = \boldsymbol{\theta}_{com} - \hat{\boldsymbol{\theta}} \tag{17}$$

$$\boldsymbol{\omega}_{err} = \boldsymbol{\omega}_{com} - \hat{\boldsymbol{\omega}} \tag{18}$$

Lastly, for all the simulation results presented the commanded attitude and angular rates align the spacecraft body frame with the LVLH frame.

3.2 Rate Filter

Estimate of inertial angular velocity of the spacecraft is required for control of the satellite. However, in the attitude estimation algorithms presented next, we only estimate the gyro drift rate bias and satellite attitude. Using the estimates of gyro drift rate bias the gyro measurement can be corrected for bias error; however, the measurements still include the random noise. The rate integrating gyros along with the above Kalman filter provide incremental attitude vector. Since the gyros work at a very high rate the incremental angles are related to $\boldsymbol{\omega}$ as follows

$$\Delta\hat{\boldsymbol{\theta}}_k = \Delta\boldsymbol{\varphi}_k - T_{gyro}\hat{\mathbf{b}}_k \tag{19}$$

$$\hat{\boldsymbol{\omega}}_{kf} \approx \frac{\Delta\hat{\boldsymbol{\theta}}_k}{T_{gyro}} \tag{20}$$

In order to remove the high frequency noise associated with above calculation of $\boldsymbol{\omega}$, a discrete low pass filter with bandwidth (ω_c) is used,

$$\hat{\boldsymbol{\omega}}_{k,lpf} = \frac{T_{gyro}\omega_c}{T_{gyro}\omega_c + 2}\hat{\boldsymbol{\omega}}_{k,kf} + \frac{T_{gyro}\omega_c}{T_{gyro}\omega_c + 2}\hat{\boldsymbol{\omega}}_{k-1,kf} - \frac{T_{gyro}\omega_c - 2}{T_{gyro}\omega_c + 2}\hat{\boldsymbol{\omega}}_{k-1,lpf} \tag{21}$$

This filtered estimate is used to arrive at the control torque, which results in a relatively smoother control action which is beneficial for actuator hardware. In our simulation the filter bandwidth is chosen as ten times that of the controller bandwidth.

4 Steady-State Kalman Filter

A steady-state three-axis Kalman Filter [2,7] is first presented for attitude determination. The filter provides the estimate of the spacecraft attitude and the gyro drift-rate bias. The gyro measurements, which are available at a very high rate, are used as the process model for the filter. The attitude sensors are used for correction of the attitude estimate and gyro drift-rate bias, and represent the measurement model. The prediction step using the rate integrating gyro measurements takes place at a higher rate, while the correction step is used only after a predetermined update interval (T_{up}).

Propagation equations, as they occur at a different rate, are described by using the subscript k. At the n-th gyro interval correction step is applied using the attitude sensors. In the following analysis, the indices $(-)$ and $(+)$ indicate the estimates prior to and post measurement updates from attitude sensors, respectively. The choice of update interval depends on the sensor error characteristics, sensor sampling rate and required pointing accuracy, and is discussed subsequently.

4.1 Prediction

Based on the gyro measurement model (Eq. 3) the propagation equations for the filter, which utilize the gyro measurements $\Delta\boldsymbol{\varphi}$, are given as:

$$\Delta\hat{\boldsymbol{\theta}}_k = \Delta\boldsymbol{\varphi}_k - T_{gyro}\hat{\mathbf{b}}_k \tag{22}$$

$$\hat{\mathbf{b}}_k = \hat{\mathbf{b}}_{k-1} \tag{23}$$

In order to obtain incremental inertial attitude from the attitude estimates, following propagation equations for the direction cosine matrix are used [19],

$$\hat{\mathrm{C}}_{k,I} = \hat{\mathrm{C}}_{k,k-1}\hat{\mathrm{C}}_{k-1,I} \tag{24}$$

$$\hat{\mathrm{C}}_{k,k-1} = \mathbf{I}_{3\times3} - \Delta[\hat{\boldsymbol{\theta}}^{\times}] + \frac{\Delta\hat{\boldsymbol{\theta}}_k\Delta\hat{\boldsymbol{\theta}}_k^T - \|\Delta\hat{\boldsymbol{\theta}}_k\|^2\mathbf{I}_{3\times3}}{2} \tag{25}$$

4.2 Steady-State Kalman Gains

To obtain the correction equations, we first need to determine the Kalman Filter gains. The gains depend on innovation covariance, error covariance of the process and measurement noise. Following the steady-state analysis of [2], the Kalman Gains for each axis are represented using three non-dimensional parameters - dependent on the sensor errors $\sigma_u, \sigma_v, \sigma_n$ and the correction update interval T_{up} - characterizing the readout noise ($S_e = \frac{\sigma_e}{\sigma_n}$), random-walk noise ($S_u = \frac{T_{up}^{3/2}\sigma_u}{\sigma_n}$), and drift angle $S_v = \frac{T_{up}^{1/2}\sigma_v}{\sigma_n}$. Based on the steady-state covariance analysis, the steady-state Kalman Filter gains are,

$$K_{hs} = (\zeta\sigma_n)^{-2}\begin{bmatrix} P_{\theta\theta}(-) \\ P_{\theta b}(-) \\ P_{\theta\varphi}(-) \end{bmatrix} = \begin{bmatrix} 1-\zeta^{-2} \\ (\zeta T_{up})^{-1}S_u \\ (S_e/\zeta)^2 \end{bmatrix} \tag{26}$$

where,

$$\gamma = (1 + S_e^2 + \frac{1}{4}S_v^2 + \frac{1}{48}S_u^2)^{\frac{1}{2}} \tag{27}$$

$$\zeta = \gamma + \frac{1}{4}S_u + \frac{1}{2}(2\gamma S_u + S_v^2 + \frac{1}{3}S_u^2)^{\frac{1}{2}} \tag{28}$$

4.3 Correction

In order to utilize the attitude sensors (sun and horizon sensors) the measurements of roll, pitch, and yaw, are transformed and represented as,

$$\hat{\mathrm{C}}_{n.att,I} = \left(\mathbf{I}_{3\times3} - [\boldsymbol{\nu_{att}}^{\times}]\right)\hat{\mathrm{C}}_{n,I} \tag{29}$$

wherein the subscript *'att'* refers to both the horizon and sun sensors and $\boldsymbol{\nu}_{att}$ quantifies the total noise in the attitude measurements. In order to obtain a three-axis equivalent of the small angle error residual, we observe that

$$\theta_{att} - \hat{\theta}_0\,(-) \Leftrightarrow \hat{\mathrm{C}}_{n.att,I}\hat{\mathrm{C}}_{I,n.gyro} \tag{30a}$$

$$\approx \mathbf{I}_{3\times 3} - [\boldsymbol{\nu}_{att} - \hat{\boldsymbol{\nu}}_{n.gyro}]^{\times} \tag{30b}$$

$$= \mathbf{I}_{3\times 3} - [\boldsymbol{\nu}_{\boldsymbol{att/gyro}}{}^{\times}] \tag{30c}$$

Hence, $\boldsymbol{\nu}_{att/gyro}$ characterizes the required difference, and can be obtained in terms of the available matrices $\hat{\mathrm{C}}_{n.att,I}$ (from measurement) and $\hat{\mathrm{C}}_{I,n.gyro}$ (from estimator),

$$\boldsymbol{\nu}^{\times}_{att/gyro} = \mathbf{I}_{3\times 3} - \hat{\mathrm{C}}_{n.att,I}\hat{\mathrm{C}}_{I,n.gyro} \tag{31}$$

The correction equation in terms of $\boldsymbol{\nu}_{att/gyro}$ for attitude and bias are given as,

$$\boldsymbol{\nu}_{att/update} = \begin{bmatrix}(1-\zeta_x^{-2})\nu_{att/gyro,x}\\(1-\zeta_y^{-2})\nu_{att/gyro,y}\\(1-\zeta_z^{-2})\nu_{att/gyro,z}\end{bmatrix} \tag{32}$$

$$\hat{\mathrm{C}}_{0.gyro,I}(+) = (\mathbf{I}_{3\times 3} - [\boldsymbol{\nu}_{att/update}]^{\times})\hat{\mathrm{C}}_{n.gyro,I} \tag{33}$$

$$\hat{\mathbf{b}}_0(+) = \hat{\mathbf{b}}_0(-) - \begin{bmatrix}S_{u,x}(\zeta_x T_{up})^{-1}\nu_{att/gyro,x}\\S_{u,y}(\zeta_y T_{up})^{-1}\nu_{att/gyro,y}\\S_{u,z}(\zeta_z T_{up})^{-1}\nu_{att/gyro,z}\end{bmatrix} \tag{34}$$

The filter thus provides estimates of inertial attitude which can be transformed to other frames as per the requirement of the attitude control sub-system. The estimates of bias are used to correct the gyro measurement. Note that since the KF gains in the three axes are independent of each other, asynchronous sun and horizon sensor measurements can also be used by the steady-state Kalman filter. Next, we discuss the initialization of the filter, selection of the update parameter and the filter's simulated performance.

4.4 Initialization

To reduce the filter transients, the filter should be initialized with the best attitude estimate available. This *a priori* estimate can be obtained from the Sun and horizon sensor measurements. These measurements are used to initialize the attitude states of the filter. The drift bias states of the filter should be initialized with the drift bias value specified in the specification sheet or as obtained through ground testing of the gyro.

4.5 Update Interval

In the above formulation, all but one variables influencing Kalman gains are dependent on the sensor characteristic. The parameter T_{up} also influences Kalman Gains, and can be chosen by the designer. The achievable values of T_{up} will be limited due to the sample time of attitude sensors (T_{hs} and T_c), and computational capability of the on-board computer. Based on steady-state covariance

analysis, variation of achievable estimate covariance with respect to T_{up} is obtained, as shown in Figs. 5-6. Both pre- and post-update covariance estimates at steady-state and corresponding standard deviation of attitude sensor errors are shown. Using the plots, the parameter T_{up} is chosen as 2s, since it provides estimation accuracy of ~ 0.01 deg at steady-state.

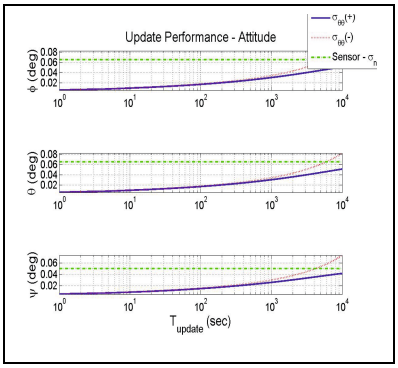


Fig. 5. T_{up} anlaysis : Attitude

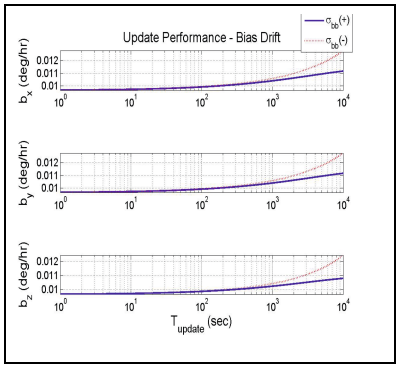


Fig. 6. T_{up} anlaysis : Gyro Bias

4.6 Estimator Performance

The results of the three-axes attitude determination algorithm developed above are now presented. The filter is propagated at a rate of 10 Hz and the attitude sensor measurement corrections are effected every 2s. The rate low-pass filter with cut-off frequency $\omega_c = 0.32$ Hz is used. Horizon sensor measurements are corrected for oblateness prior to being used in the filter and no gyro misalignments are considered. Initial estimation errors, tabulated in Table 7, have been included as per section 4.4.

Table 7. Initial Estimation Errors

Attitude Estimation Error(deg)		
ϕ	θ	ψ
0.1	0.1	0.1
Drift Bias Estimation Error (deg/hr)		
b_x	b_y	b_z
0.03	0.03	0.03

Test Case: As a theoretical test case, we observe the performance of the filter in presence of white noise in attitude sensors, where the model in the Kalman filter completely matches with the measurements. The estimation performance is within the predicted bounds (Figs. 7-8); however, long duration transients

(up to 20% of the orbit period) persist before the steady-state is arrived at. Due to the coupling between the roll and yaw axis, and propagation due to the gyro measurements, performance in the eclipse region is also satisfactory.

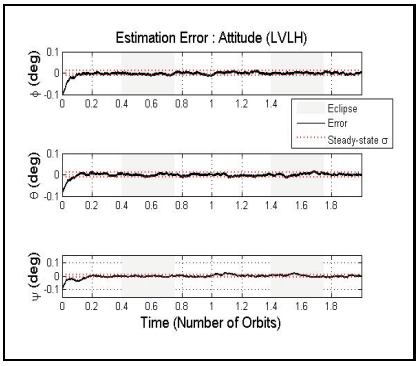


Fig. 7. Steady-state KF with white measurement noise : Attitude

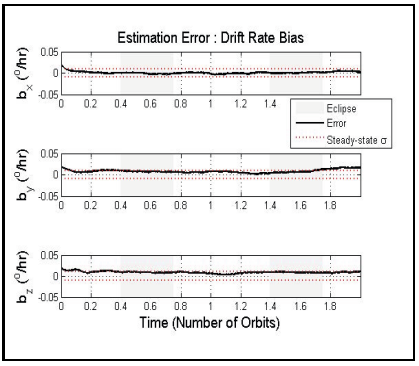


Fig. 8. Steady-state KF with white measurement noise : Gyro Bias

Performance with Realistic Measurement Errors: Next, using the estimator parameters specified earlier, we obtain the performance of the steady-state KF in presence of all sensor errors except gyro misalignments. As observed in Figs. 9-10, the estimation performance has degraded considerably as compared to the theoretical test case. However, this is expected as the bias and radiance variations in horizon sensor measurements are not being compensated for in the estimation algorithm.

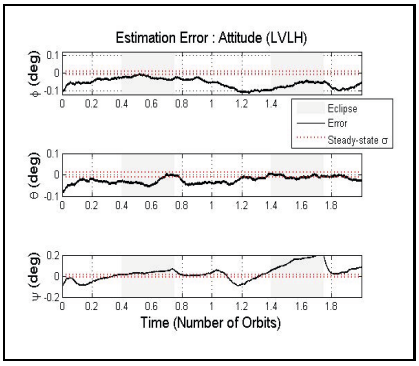


Fig. 9. Steady-state KF : Attitude

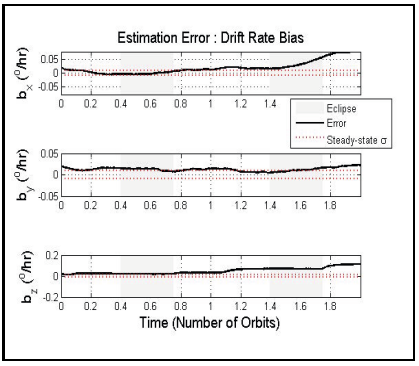


Fig. 10. Steady-state KF : Gyro bias

5 Extended Kalman Filter

This section briefly describes the quaternion-based extended Kalman filter (EKF) of Lefferts, Markley and Shuster [3] for satellite attitude determination. As with the steady-state KF, this filter too estimates the satellite attitude and gyro drift rate bias.

5.1 Formulation

The quaternion-based EKF estimates the attitude quaternion ($\mathbf{q}_{BI}$, 4 elements) and gyro bias ($\mathbf{b}$, 3 elements) resulting in seven states. However, the covariance propagation is achieved through error state to avoid the quaternion singularity. The error states for quaternion ($\delta\mathbf{q} = \mathbf{q} \otimes \hat{\mathbf{q}}^{-1}$) and bias ($\boldsymbol{\Delta b} = \mathbf{b} - \hat{\mathbf{b}}$) result in a six-element state vector: $\boldsymbol{\Delta x} = [\delta\mathbf{q}, \boldsymbol{\Delta b}]'$. Process model for error state is then given as $\boldsymbol{\Delta \dot{x}} = \mathrm{F}\boldsymbol{\Delta x} + \mathrm{G}\mathbf{w}$, where,

$$\mathrm{F} = \begin{bmatrix} -[\boldsymbol{\omega}^{\times}] & -1/2\mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix} \tag{35}$$

$$\mathrm{G} = \begin{bmatrix} -1/2\mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix} \tag{36}$$

$$\mathbf{w} = \begin{bmatrix} \boldsymbol{\eta_v} & \boldsymbol{\eta_u} \end{bmatrix}' \tag{37}$$

The development of the quaternion EKF in [3] provides freedom while using the attitude sensors, in the sense that the measurements can be used either as scalar angles or reference vectors. Through simulation it was observed that use of either approaches produces similar performance; hence, here we present only one of them. Sensor models presented earlier represent the measurement models. Here, we list the measurement noise covariance (R) and sensitivity (H) matrices corresponding to the two attitude sensors,

Sun Sensor

$$\mathrm{R}_k = E[\mathbf{v}_k'\mathbf{v}_k] = \sigma_c^2(\mathbf{I}_{3\times3}) \tag{38}$$

$$\mathrm{H}_k = \begin{bmatrix} 2(\mathbf{r}_1 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} \\ 2(\mathbf{r}_2 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} \\ 2(\mathbf{r}_3 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} \end{bmatrix} \tag{39}$$

where, $\mathbf{r}_i = [\delta_{i1}, \delta_{i2}, \delta_{i3}]'$, the symbol δ_{ij} representing the Kronecker delta; and $\hat{\mathbf{z}}_k (= \hat{\mathbf{s}}_B = [\mathrm{C}(\hat{\mathbf{q}})]\mathbf{s}_I)$ corresponds to the modeled sun vector in the body frame. The sensitivity matrix H_k is derived through application of the corresponding general expressions provided in (Eq. 151-157) of [3].

Horizon Sensor

$$\mathrm{R}_k = \begin{bmatrix} \sigma_\theta^2 & 0 \\ 0 & \sigma_\phi^2 \end{bmatrix} \tag{40}$$

$$\mathrm{H}_k = \begin{bmatrix} 2(\mathbf{r}_1 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} \\ 2(\mathbf{r}_2 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} \end{bmatrix} \tag{41}$$

where $\hat{\mathbf{z}}_k$ corresponds to the modeled value of the nadir vector in the body frame; and horizon sensor measurement is represented as $[-\theta_m, \phi_m]'$ to correspond to the definition of the nadir vector. As expected these measurements depend only on the attitude and not on the gyro bias. For further details of this filter and the standard Kalman Filter equations the reader is directed to reference [3].

5.2 Estimator Performance

We present the performance of the quaternion-based EKF using similar measurement models and parameters as to that for the steady-state KF. As the measurements from different sensors need not be synchronous, each measurement update is applied independently, using Murell's approach [20]. While calculating filter gains we need to specify P_0. When the sun and horizon sensors are used to provide the initial estimates, value of P_0 corresponding to the accuracy of these sensors is used. The initial drift bias state covariances can be obtained from ground testing of the gyro.

Test Case: The Kalman filter is optimal in presence of measurement errors being white noise. Hence, similar to the case of steady-state KF, as a theoretical test case, we first observe the performance of the EKF in presence of only white noise in attitude sensors. The absence of yaw (sun sensor) measurements, during the eclipse phase, results in increased state covariance in yaw estimates. Due to the exact correspondence between the measurement noise and its model being white, the assumptions of EKF are satisfied, resulting in the expected estimation performance, shown in Figs. 11-12.

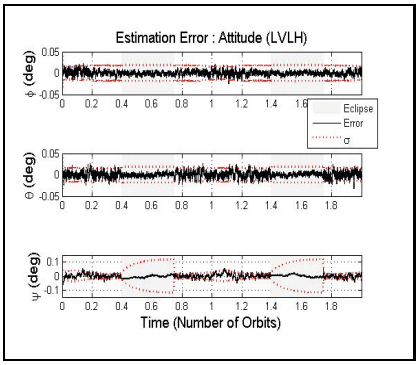


Fig. 11. EKF (white noise) : Attitude

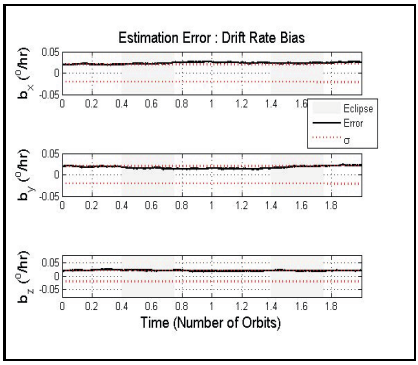


Fig. 12. EKF : Gyro bias

Performance with Realistic Measurement Errors: Having verified the filter through a simplified measurement model, we observe its performance in presence of the complete measurement models except for misalignments. Similar to

the case of steady-state KF, the performance of the filter deteriorates. The observed estimation accuracy in presence of colored noise and bias in measurement is given as seen in Figs. 13-14.

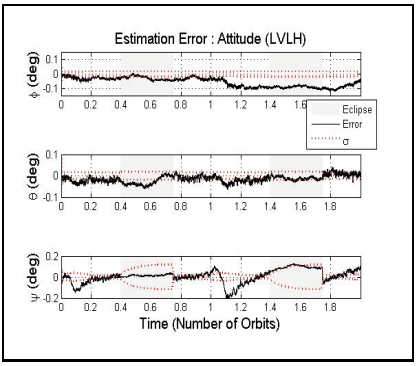


Fig. 13. EKF : Attitude

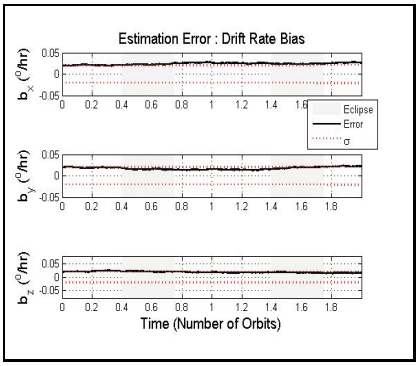


Fig. 14. EKF : Gyro bias

6 Estimation of Additional States

The horizon sensor measurements used in both the steady-state KF and quaternion-based EKF described earlier contain errors such as biases and radiance variations, which are not just white noise. These filters, however, work under the assumption that the errors entering the measurement model are white - resulting therefore in sub-optimal estimation performance. Better estimates of the attitude requires augmentation of the state vector, albeit without any additional measurements. This may result in the system becoming unobservable, degrading the attitude performance. Attempts to estimate biases of attitude sensors [17] and the noise due to radiance variation [21] have been reported earlier in the literature, and mixed results have been obtained. Hence, the augmented state vector filter should be implemented only after verifying the performance of the estimator through analysis and simulation. We proceed with the state vector augmentation for the quaternion-based EKF instead of the steady- state KF, due to its relatively straight-forward formulation and ease of implementation. A similar approach of state vector augmentation for estimation of misalignment and scale factor errors has been developed by Pittelkau.

6.1 Horizon Sensor Error Estimation

The state vector is augmented with the noise terms corresponding to the horizon sensor radiance error $(w_\phi,\ w_\theta)$: $\mathbf{x}(t) = [\mathbf{q}_{BI}, \boldsymbol{\Delta b}_g, w_\phi, w_\theta]'$. The discrete-time model for radiance errors in roll and pitch is given by Eq. (12a-12b). This can be represented as a differential equation,

$$\dot{w_\phi}(t) = -\beta w_\phi + \eta_{\phi,w} \tag{42}$$

$$\dot{w_\theta}(t) = -\beta w_\theta + \eta_{\theta,w} \tag{43}$$

where the parameter β and the PSD of η are obtained by comparing the continuous model with the discrete-time equation: $\beta = \frac{1}{\tau_w}$, PSD of $\eta_{\phi,w}$and $\eta_{\theta,w} = \frac{l^2(1-a^2)}{T_{hs}}$. The evolution of these error parameters is independent of the attitude and gyro bias, and the same is reflected in the modified process model,

$$\frac{d}{dt}\begin{bmatrix} \mathbf{q}_{BI} \\ \boldsymbol{\Delta b}_g \\ w_\phi \\ w_\theta \end{bmatrix} = \begin{bmatrix} -[\boldsymbol{\omega}^\times] & -1/2\mathbf{I}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times1} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & -\beta & 0 \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 0 & -\beta \end{bmatrix} \begin{bmatrix} \mathbf{q}_{BI} \\ \boldsymbol{\Delta b}_g \\ w_\phi \\ w_\theta \end{bmatrix} + \begin{bmatrix} -1/2\boldsymbol{\eta_v} \\ \boldsymbol{\eta_u} \\ \eta_{\phi,w} \\ \eta_{\theta,w} \end{bmatrix} \tag{44}$$

Propagation equation for the quaternion and bias is the same as for the filter described earlier. The propagation for the radiance variation is done using the discrete counterpart of the process model described above. Sun sensor measurement equations and corresponding noise covariance (Eq. 38) remains the same. Horizon sensor measurements are corrected for noise due to oblateness. In the current formulation, an estimate of the horizon sensor noise due to radiance variation is developed so as to subtract it from the horizon sensor measurements. The corresponding sensitivity matrix is given as,

$$\mathrm{H}_k = \begin{bmatrix} 2(\mathbf{r}_1 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} & 0 & -1 \\ 2(\mathbf{r}_2 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} & 1 & 0 \end{bmatrix} \tag{45}$$

Radiance Noise Estimation. The initial estimate of the attitude and gyro bias are specified to be the same as before, whereas since no estimate of residual error due to radiance variation is available its initial value is taken as zero. The steady-state RMS value of radiance noise is used to define P_0, the initial state covariance. As seen in Figs. 15-16, the estimation error of the radiance noise is within the covariance bounds. However, the predicted bounds for the most part of the orbit are the same as l, the steady-state RMS of the radiance noise.

Although the radiance noise is estimated to certain accuracy, the improvements in attitude estimation are not significant even with the application of this modified filter (Fig. 6.1). Similar results were reported in [21] wherein with real attitude data a similar augmented filter was able to estimate the radiance noise but improved the attitude estimation performance only marginally.

Bias Estimation. Apart from radiance noise, the horizon sensor measurements also have a constant bias which may arise due to sensor electronics or static misalignment. Since our filter currently does not estimate the bias, the attitude estimate might improve with estimation of this bias, even though the magnitude of this bias is relatively smaller than the radiance noise. Hence, we try to estimate the horizon sensor bias using the augmented filter, instead of the radiance noise.

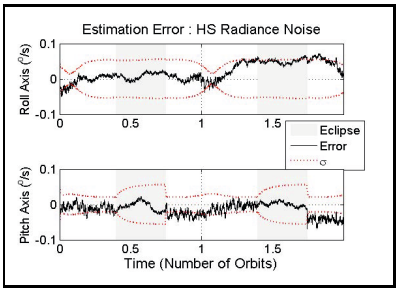


Fig. 15. Horizon Sensor Radiance Noise : Estimation Error

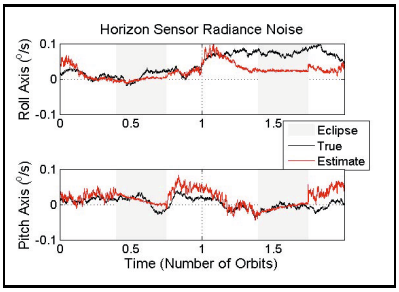


Fig. 16. Horizon Sensor Radiance Noise : True v/s Estimate

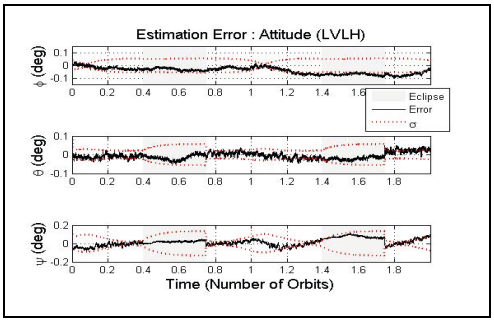


Fig. 17. Effect of Radiance Estimation : Attitude

The formulation to estimate radiance noise can be adapted to estimate horizon sensor bias, by choosing the parameter β in the process model of the filter as zero. The filter is initialized similar to the case of radiance estimation. The horizon sensor bias estimation performance is portrayed in Figs. 18-19.

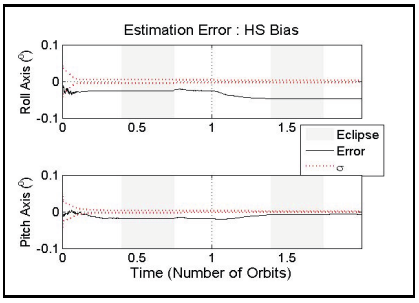


Fig. 18. Horizon Sensor Bias : Estimation Error

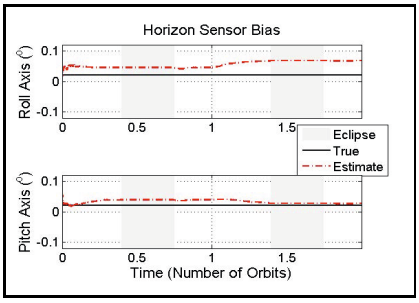


Fig. 19. Horizon Sensor Bias : True v/s Estimate

Improvements in attitude estimation accuracy are obtained due to estimation of the horizon sensor bias. As seen in Fig. 20, the attitude estimation accuracy is generally within the predicted covariance, and yaw estimates do not degrade much during eclipse.

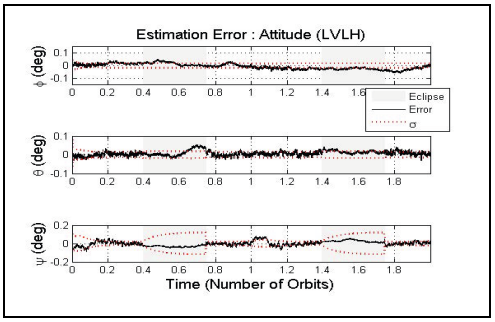


Fig. 20. Effect of Bias Estimation : Attitude

6.2 Misaligned Sensors

In the simulation discussed till now, no sensor misalignments were considered. Presence of misalignment in either the gyro or the attitude sensors may result in incorrect estimates of both the attitude and gyro drift rate bias. Both batch and sequential misalignment estimation method exist in the literature. Here, we consider application of the misalignment estimation Kalman filter developed by Pittelkau [4], which is called the Alignment Kalman Filter (AKF). This filter is suited for on-board real time estimation of sensor misalignments and scale factor errors.

The AKF, too, is an augmentation of the quaternion-based EKF. The augmented state vector is given as $\mathbf{x} = [\boldsymbol{\delta q}_v, \boldsymbol{\Delta b}_g, \boldsymbol{\delta}_g, \mathbf{b}_{HS}]'$, where, the parameter $\boldsymbol{\delta}_g$ $(= [\lambda_x, \delta_{yz}, \delta_{zy}, \delta_{xz}, \lambda_y, \delta_{zx}, \delta_{xy}, \delta_{yx}, \lambda_z]')$ denotes the gyro misalignments and scale factor terms, and $\mathbf{b}_{HS}$ $(= [b_\phi, b_\theta]')$ denotes the two components of horizon sensor bias . The alignment Kalman filter, thus, attempts to estimate additional states - misalignment of sensors - along with the attitude and sensor biases. However, the number of measurements used for correction of the state vector still remain the same. Naturally, this causes concerns of observability of the state vector and potential degradation of the estimate. In order to make the system observable, and prevent ill effects of state augmentation, in-flight attitude maneuvers are performed. These maneuvers are of higher frequency then the spacecraft dynamics, and for on-board implementation require capable actuators. The system is made observable by subjecting the spacecraft to

non-harmonic sinusoidal angular rates. Hence, the rate command for the attitude controller is modified to be as,

$$\boldsymbol{\omega}_{BO}^{B} = (0.05\,\mathrm{deg}\,/\,\mathrm{sec}) \begin{bmatrix} \sin[2\pi(0.0100)] \\ \sin[2\pi(0.0085)] \\ \sin[2\pi(0.0080)] \end{bmatrix} \tag{46}$$

The process model for the above state vector is given as,

$$\frac{d}{dt}\begin{bmatrix} \mathbf{q}_{BI} \\ \boldsymbol{\Delta b}_g \\ \boldsymbol{\delta}_g \\ \mathbf{b}_{HS} \end{bmatrix} = \begin{bmatrix} -[\boldsymbol{\omega}^{\times}] & -1/2\mathbf{I}_{3\times3} & 1/2\Omega_g & \mathbf{0}_{3\times2} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times9} & \mathbf{0}_{3\times2} \\ \mathbf{0}_{9\times3} & \mathbf{0}_{9\times3} & \mathbf{0}_{9\times9} & \mathbf{0}_{9\times2} \\ \mathbf{0}_{2\times3} & \mathbf{0}_{2\times3} & \mathbf{0}_{2\times9} & \mathbf{0}_{2\times2} \end{bmatrix} \begin{bmatrix} \mathbf{q}_{BI} \\ \boldsymbol{\Delta b}_g \\ \boldsymbol{\delta}_g \\ \mathbf{b}_{HS} \end{bmatrix} + \begin{bmatrix} -1/2(\mathbf{I}_{3\times3} + \mathrm{M})\boldsymbol{\eta_v} \\ \boldsymbol{\eta_u} \\ \boldsymbol{\eta_g} \\ \boldsymbol{\eta_b} \end{bmatrix} \tag{47}$$

where, M denotes the gyro misalignment and scale factor matrix,

$$\mathrm{M} = \begin{bmatrix} \lambda_x & -\delta_{yz} & \delta_{zy} \\ \delta_{xz} & \lambda_y & -\delta_{zx} \\ -\delta_{xy} & \delta_{yx} & \lambda_z \end{bmatrix} \tag{48}$$

and Ω_g is defined in terms of the inertial rates,

$$\Omega_g = \begin{bmatrix} \omega_x & -\omega_y & \omega_z & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \omega_x & \omega_y & -\omega_z & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\omega_x & \omega_y & \omega_z \end{bmatrix} \tag{49}$$

The propagation equations for $\boldsymbol{\delta}_g$ and $\mathbf{b}_{HS}$ is the same as that of the $\boldsymbol{\Delta b}_g$ due to analogous continuous time equations. The propagation of the full quaternion is also done similar to that of the 6-state EKF, except that rate-integrating gyro measurements are compensated not just for drift rate bias but also misalignments and scale factors using available estimates. The measurement sensitivity matrix is expanded to account for the additional states; for instance, for horizon sensor the sensitivity matrix is given as,

$$\mathrm{H}_k = \begin{bmatrix} 2(\mathbf{r}_1 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times9} & 0 & -1 \\ 2(\mathbf{r}_2 \times \hat{\mathbf{z}}_k)' & \mathbf{0}_{1\times3} & \mathbf{0}_{1\times9} & 1 & 0 \end{bmatrix} \tag{50}$$

When the spacecraft is subjected to sinusoidal rate maneuvers, as specified in Eq. 46, the misalignment states are expected to become observable, resulting in improvement in attitude estimation performance. Further, the attitude and gyro drift rate bias performance is within the Kalman filter covariance bounds (Figs. 21-22).

The estimation of scale factors and misalignment in all the three axes is possible. Within two orbits the scale factor and miaslignment estimation errors achieve steady-state (Figs. 23-24), after which the maneuvers are terminated and normal spacecraft operation is resumed.

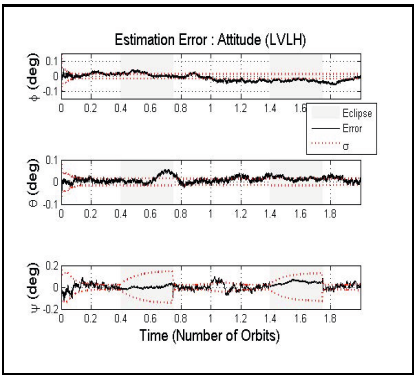


Fig. 21. AKF : Attitude

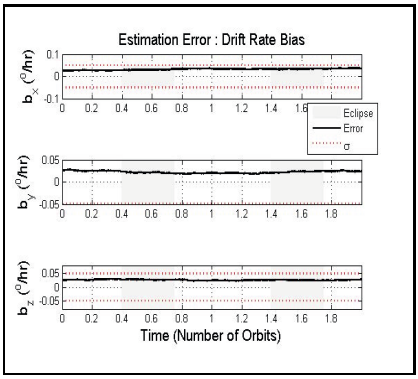


Fig. 22. AKF : Gyro Bias

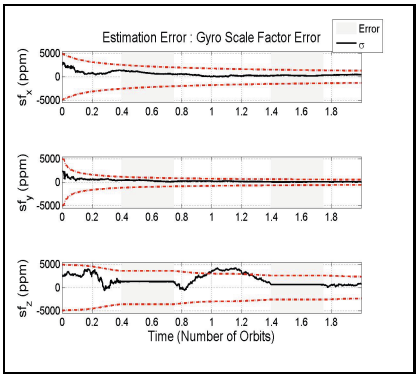


Fig. 23. Gyro Scale Factors : Estimation Error

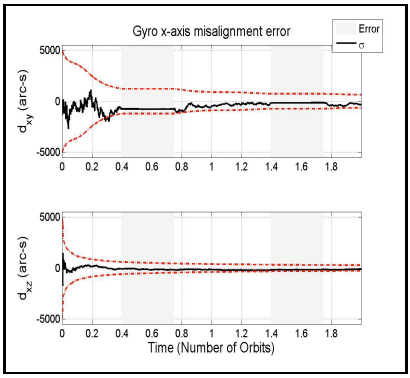


Fig. 24. Gyro Misalignment (x-axis) : Estimation Error

7 Conclusions and Comments

Sequential on-board attitude estimation algorithms for the sensor suite - sun sensor, horizon sensor and gyros - are studied, analyzed and simulated. As the attitude estimation algorithms are tested through simulation, an effort is made to consider detailed models of sensors and to test the algorithms in presence of realistic sensor errors. A standard, realistic model of rate-integrating gyro is used which includes time varying drift-rate bias, static misalignment, scale factor errors and quantization noise. Horizon sensor modeling includes effects of Earth's oblateness, atmospheric radiance, bias and electronic noise. An approximate model of noise due to atmospheric radiance, treating it as an exponentially auto-correlated noise, is considered. A modular six degree-of-freedom closed-loop simulation has been developed, and performance of the attitude estimation algorithms using this simulation is presented.

Two attitude estimation algorithms - the DCM-based steady-state KF and quaternion-based EKF - are simulated. Modifications in the standard algorithms have been made, to customize them for the sensors under consideration. The steady-state filter is observed to work for the current sensor suite if only white noise is present in the attitude sensors. Hence, the steady-state KF, which is computationally efficient, may be a preferred choice in the case of the sensors primarily exhibiting such noise characteristics, such as the star trackers [7]. Further, depending on sensor error characteristics and required estimation accuracy, the steady-state KF may not be suitable during the eclipse phase of the orbit.

The quaternion-based EKF, too, in its original form works only for attitude sensors with white noise. Hence, there is a need to estimate other errors in the attitude sensors through augmentation of the state vector. The quaternion-based EKF offers a natural framework for estimation of additional states. Through estimation of horizon sensor bias, but not that of radiance noise, improvement in estimator performance is observed. Even during the eclipse phase of the orbit, when the sun sensor measurements are not available, the filter performance is satisfactory. The performance of steady-state KF may also be improved by estimation of horizon sensor bias; however, the quaternion-based EKF offers a simpler way of augmenting the state vector and can better handle asynchronous measurements. Hence, we recommend the use of the quaternion-based EKF with augmented state vector. Lastly, to account for sensor misalignment and scale factor errors, the alignment Kalman filter is studied and simulated. Due to satellite maneuvers, the AKF is able to estimate the misalignment parameters, resulting in desired attitude estimation performance.

References

1. Markley, F.L.: Spacecraft attitude determination methods. In: Israel Annual Conference on Aerospace Sciences (2000)
2. Markley, F.L., Reynolds, R.G.: Analytic steady-state accuracy of a spacecraft attitude estimator. Journal of Guidance, Control, and Dynamics 23(6), 1065–1067 (2000)
3. Leerts, E.J., Markley, F.L., Shuster, M.D.: Kalman ltering for spacecraft attitude estimation. Journal of Guidance 5(5), 417–429 (1982)
4. Pittelkau, M.E.: Kalman filtering for spacecraft system alignment calibration. Journal of Guidance, Control, and Dynamics 24(6), 1187–1195 (2001)
5. Wertz, J.R. (ed.): Spacecraft Attitude Determination and Control. Computer Sciences Corporation (1978)
6. Sidi, M.J.: Spacecraft Dynamics and Control. Cambridge University Press (1997)
7. Hablani, H.B.: Autonomous inertial relative navigation with sight-line-stabilized integrated sensors for spacecraft rendezvous. Journal of Guidance, Control, and Dynamics 32(1) (2009)
8. Joshi, J., et al.: Conceptual design report - ADCS, PRATHAM. Technical report, Indian Institute of Technology Bombay (2009)
9. Alex, T.K., Shrivastava, S.K.: On-board corrections of systematic errors of earth sensors. IEEE Transactions on Aerospace and Electronic Systems 25(3), 373–379 (1989)

10. Alex, T.K., Seshamani, R.: Generation of infrared earth radiance for attitude determination. Journal of Guidance, Control, and Dynamics 12(2), 257–277 (1989)
11. Hablani, H.B.: Roll/pitch determination with scanning horizon sensors - oblateness and altitude corrections. Journal of Guidance, Control, and Dynamics 18(6), 1355–1364 (1995)
12. Tekawy, J.A., Wang, P., Gray, C.W.: Scanning horizon sensor attitude corrections for earth oblateness. Journal of Guidance, Control and Dynamics 19(3), 706–708 (1996)
13. Hashmall, J.A., Sedlak, J., Andrews, D., Luquette, R.: Empirical correction for earth sensor horizon radiance variation. In: Proceedings, AAS/GSFC 13th International Symposium on Space Flight Dynamics, Goddard Space Flight Center,Greenbelt, Maryland (May 1998)
14. Phenneger, M.C., Singhal, S.P., Lee, T.H., Stengle, T.H.: Infrared horizon sensor modeling for attitude determination and control: analysis and mission experience. NASA technical memorandum. National Aeronautics and Space Administration,Scientific and Technical Information Branch (1985)
15. Crassidis, J.L., Markley, F.L., Kyle, A.M., Kull, K.: Attitude determination improvements for GOES. In: Proceedings, Flight Mechanics/ Estimation Theory Symposium, Goddard Space Flight Center, Greenbelt, Maryland, NASA Conference Publication 3333, pp. 161–175 (May 1996)
16. STD 16, EADS Sodern Earth Sensor Brochure, `www.sodern.com/sites/docs_wsw/RUB_52/STD16.pdf`
17. Deutschmann, J., Bar-Itzhack, I.Y.: Extended kalman filter for the attitude estimation of the earth radiation budget satellite. In: Proceedings, Flight Me chanics/Estimation Theory Symposium, Goddard Space Flight Center, Greenbelt, Maryland, NASA Conference Publication 3050, pp. 333–346 (1989)
18. ISRO Oceansat-2 Brochure. Oceansat-2-Brochure-1.pdf, `http://www.isro.org/pslv-c14/pdf/`
19. Hughes, P.C.: Spacecraft Attitude Dynamics. Dover Publications (2004)
20. Crassidis, J., Junkins, J.: Optimal Estimation of Dynamic Systems. Chapman & Hall/CRC Applied Mathematics and Nonlinear Science Series. Chapman &Hall/CRC (2004)
21. Sedlak, J.: Improved earth sensor performance using a sequentially correlated noise model. In: Proceedings, Flight Mechanics Symposium, Goddard Space Flight Center, Greenbelt, Maryland, NASA Conference Publication, pp. 71–83 (May 1999)

Quaternion Data Fusion

Yang Cheng[1], William D. Banas[2], and John L. Crassidis[2]

[1] Mississippi State University, Mississippi State, MS 39762
cheng@ae.msstate.edu
[2] University at Buffalo, Buffalo, NY 14260
{wdbanas,johnc}@buffalo.edu

Abstract. A numerical method for solving a class of constrained minimization problems encountered in quaternion data fusion is presented. The quaternion constraints are handled by the method of Lagrange multipliers. The number of the stationary points of the minimization problem is finite and all of them are found by solving via homotopy continuation a system of polynomial equations. The global minimizer is the stationary point that minimizes the loss function of the minimization problem. A numerical example of two-quaternion data fusion is given to illustrate the viability of the method as a global minimization method for quaternion data fusion.

Keywords: quaternion, data fusion, homotopy continuation.

1 Introduction

Attitude quaternion [7] is the attitude parameterization of choice for spacecraft attitude estimation for several reasons: 1) it is free of singularities, 2) the attitude matrix is quadratic in the quaternion components, and 3) the kinematics equations is bilinear and an analytic solution exists for the propagation. However, the components of the attitude quaternion are not independent of each other and the norm of the attitude quaternion must be unity. This unity-norm constraint leads to problems for data fusion involving quaternions. The objective of data fusion is to find the optimal estimate from data of various sources.

Reference [5] addresses the problem of fusing or averaging a set of quaternions. The fused or averaged quaternion is defined as the optimal solution to a constrained minimization problem subject to one equality constraint (the quaternion constraint). The method of Lagrange multipliers is used to convert the constrained minimization problem to an unconstrained minimization problem. The Lagrange multiplier is the maximum eigenvalue of a 4×4 symmetric matrix composed from the quaternions and weights [3]. The optimal average quaternion is the eigenvector corresponding to the maximum eigenvalue [3].

Reference [3] addresses a more general data fusion problem in which the state vector of interest consists of one quaternion and a set of unconstrained parameters, for example, gyro biases. The data fusion problem is also formulated as a constrained minimization problem subject to one equality constraint. The Lagrange multiplier is now the maximum eigenvalue of an 8×8 asymmetric matrix

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_23

or the maximum root of the 8^{th}-degree secular equation [3]. Given the Lagrange multiplier, the optimal state estimate is obtained by solving a linear system of equations [3].

This paper addresses an even more general data fusion problem in which the state vector includes two or more quaternions as well as a set of unconstrained parameters. Such a problem appears in formation flying involving multiple vehicles in which two or more relative attitudes need to be determined or fused[1]. The data fusion problem is again defined as a constrained minimization problem, but now subject to two or more equality constraints. The multiple quaternion constraints significantly increase the difficulty of the minimization problem. To our knowledge, no closed form solution exists for the minimization problem and few properties of the Lagrange multipliers are known. The latter makes it difficult to choose the initial guess of the Lagrange multipliers when solving the problem using an iterative method.

In this paper, a numerical method is presented for the constrained minimization problem subject to multiple quaternion constraints based on the solution of polynomial systems. Unlike the iterative gradient-based methods, which can only find a local minimum of the minimization problem, this method first finds all the stationary points and then selects the global minimum from them. In addition, the method provides insights to the properties, the number of local minima in particular, of the minimization problem.

The organization of the remainder of this paper is as follows. First, one-quaternion data fusion is reviewed. Next, the problem statement and formal solution of the multi-quaternion data fusion problem are given. Then, the numerical solution of the polynomial system is presented. Finally, a numerical example is given, followed by the conclusions.

2 One-Quaternion Data Fusion

The objective of data fusion is to fuse n estimates $\mathbf{x}_i$ of a state vector $\mathbf{x}$ to yield a single (better) estimate of the state vector. Throughout this paper, it is assumed that quaternion is part of the state vector and that the optimal estimate is the solution to a constrained minimization problem of which the loss function is quadratic in the state vector. While the solution to one-quaternion data fusion problem has been studied in [5,3], the problems and solutions of one-quaternion data fusion are reviewed in this section for sake of completeness.

The vector and scalar parts of a quaternion are defined by $\mathbf{q} \triangleq \begin{bmatrix} \boldsymbol{\varrho}^T & q_4 \end{bmatrix}^T$, which are assumed to satisfy the unity-norm constraint $||\boldsymbol{\varrho}||^2 + q_4^2 = \mathbf{q}^T\mathbf{q} = 1$. The attitude matrix is related to the quaternion by

$$A(\mathbf{q}) = \Xi^T(\mathbf{q})\Psi(\mathbf{q}) = \left(q_4^2 - ||\boldsymbol{\varrho}||^2\right) I_{3\times 3} + 2\,\boldsymbol{\varrho}\,\boldsymbol{\varrho}^T - 2\,q_4[\boldsymbol{\varrho}\times] \tag{1}$$

where $I_{3\times 3}$ is a 3×3 identity matrix and $[\boldsymbol{\varrho}\times]$ is the cross-product matrix defined by

$$[\boldsymbol{\varrho}\times] \triangleq \begin{bmatrix} 0 & -q_3 & q_2 \\ q_3 & 0 & -q_1 \\ -q_2 & q_1 & 0 \end{bmatrix} \tag{2}$$

$$\Xi(\mathbf{q}) \triangleq \begin{bmatrix} q_4 I_{3\times 3} + [\boldsymbol{\varrho}\times] \\ -\boldsymbol{\varrho}^T \end{bmatrix}, \Psi(\mathbf{q}) \triangleq \begin{bmatrix} q_4 I_{3\times 3} - [\boldsymbol{\varrho}\times] \\ -\boldsymbol{\varrho}^T \end{bmatrix} \tag{3}$$

The simplest case of one-quaternion data fusion is now reviewed, where the i^{th} state estimate $\mathbf{x}_i = \mathbf{q}_i$. The loss function is chosen as [5]

$$J(\mathbf{q}) = \frac{1}{2}\sum_{i=1}^{n} \mathbf{q}^T \Xi(\mathbf{q}_i)\, \mathcal{W}_{qq_i} \Xi^T(\mathbf{q}_i)\mathbf{q} \tag{4}$$

where $\mathcal{W}_{qq_i}$ is a 3×3 positive definite weighting matrix. The three-dimensional vector $\Xi^T(\mathbf{q}_i)\mathbf{q}$ has been widely to measure the attitude error in spacecraft attitude estimation and the magnitude of $\Xi^T(\mathbf{q}_i)\mathbf{q}$ is the absolute value of the sine of the half-error angle [5]. The quaternion constraint $\mathbf{q}^T\mathbf{q} = 1$ is handled using the method of Lagrange multipliers, which gives the augmented loss function as

$$J(\mathbf{q}) = \frac{1}{2}\sum_{i=1}^{n} \mathbf{q}^T \Xi(\mathbf{q}_i)\, \mathcal{W}_{qq_i} \Xi^T(\mathbf{q}_i)\mathbf{q} + \frac{\lambda}{2}(\mathbf{q}^T\mathbf{q} - 1) \tag{5}$$

where λ is the Lagrange multiplier. The necessary conditions for minimization of Eq. (5) are

$$(\mathcal{F} + \lambda I_{4\times 4})\, \mathbf{q} = \mathbf{0} \tag{6a}$$

$$\mathbf{q}^T\mathbf{q} = 1 \tag{6b}$$

where

$$\mathcal{F} \triangleq \sum_{i=1}^{n} \Xi(\mathbf{q}_i)\, \mathcal{W}_{qq_i} \Xi^T(\mathbf{q}_i) \tag{7}$$

A vector satisfying the the necessary conditions is called a stationary point. Equation (6a) indicates that $\mathbf{q}$ and λ are the eigenvector and eigenvalue of $\mathcal{F}$, respectively. Since the loss function for the quaternion satisfying Eq. (6) equals $-\lambda/2$, the optimal average quaternion is the eigenvector corresponding to the maximum eigenvalue of $-\mathcal{F}$.

When the i^{th} state estimate $\mathbf{x}_i$ is composed of a quaternion $\mathbf{q}_i$, which is subject to the constraint $\mathbf{q}_i^T\mathbf{q}_i = 1$, and other quantities $\mathbf{b}_i$, which are free of constraints, the loss function is given by

$$J(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{n} \Delta\mathbf{x}_i^T \mathcal{W}_i \Delta\mathbf{x}_i \tag{8}$$

with

$$\mathbf{x} \triangleq \begin{bmatrix} \mathbf{q} \\ \mathbf{b} \end{bmatrix} \tag{9a}$$

$$\Delta\mathbf{x}_i \triangleq \begin{bmatrix} \Xi^T(\mathbf{q}_i)\mathbf{q} \\ \\ \mathbf{b} - \mathbf{b}_i \end{bmatrix} \tag{9b}$$

$$\mathcal{W}_i \triangleq \begin{bmatrix} \mathcal{W}_{qq_i} & \mathcal{W}_{qb_i} \\ \\ \mathcal{W}_{qb_i}^T & \mathcal{W}_{bb_i} \end{bmatrix} \tag{9c}$$

where $\mathcal{W}_i$ is positive definite. The vector $\mathbf{b}$ can be of any dimension, denoted by n_b, and the same attitude error vector expression $\Xi^T(\mathbf{q}_i)\mathbf{q}$ has been used as in the previous case. Note that although $\mathbf{q}$ and $-\mathbf{q}$ represent the same attitude, they give attitude error vectors in opposite directions. For nonzero $\mathcal{W}_{qb_i}$, $[\mathbf{q}^T, \mathbf{b}^T]^T$ and $[-\mathbf{q}^T, \mathbf{b}^T]^T$ yield different values of the loss function. Only when $\mathcal{W}_{qb_i} = 0$ or $\mathbf{x} = \mathbf{q}$ ($\mathbf{b}$ is empty), $\pm\mathbf{q}$ or $\pm\Xi^T(\mathbf{q}_i)\mathbf{q}$ yield the same loss function.

The augmented loss function is now

$$J(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{n} \Delta\mathbf{x}_i^T \mathcal{W}_i \Delta\mathbf{x}_i + \frac{\lambda}{2}(\mathbf{q}^T\mathbf{q} - 1) \tag{10}$$

The necessary conditions for minimization of Eq. (10) are

$$\begin{bmatrix} \mathcal{B}_{qq} + \lambda I_{4\times 4} & \mathcal{B}_{qb} \\ \\ \mathcal{B}_{qb}^T & \mathcal{B}_{bb} \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{c} \\ \\ \mathbf{d} \end{bmatrix} \tag{11a}$$

$$\mathbf{q}^T\mathbf{q} = 1 \tag{11b}$$

where

$$\mathcal{B} = \begin{bmatrix} \mathcal{B}_{qq} & \mathcal{B}_{qb} \\ \\ \mathcal{B}_{qb}^T & \mathcal{B}_{bb} \end{bmatrix} \triangleq \begin{bmatrix} \sum_{i=1}^n \Xi(\mathbf{q}_i)\mathcal{W}_{qq_i}\Xi^T(\mathbf{q}_i) & \sum_{i=1}^n \Xi(\mathbf{q}_i)\mathcal{W}_{qb_i} \\ \\ \sum_{i=1}^n \mathcal{W}_{qb_i}^T\Xi^T(\mathbf{q}_i) & \sum_{i=1}^n \mathcal{W}_{bb_i} \end{bmatrix} \tag{12a}$$

$$\mathbf{c} \triangleq \sum_{i=1}^{n} \Xi(\mathbf{q}_i)\mathcal{W}_{qb_i}\mathbf{b}_i \tag{12b}$$

$$\mathbf{d} \triangleq \sum_{i=1}^{n} \mathcal{W}_{bb_i}\mathbf{b}_i \tag{12c}$$

Solving the second subequation of Eq. (11a) leads to

$$\mathbf{b} = \mathcal{B}_{bb}^{-1}(\mathbf{d} - \mathcal{B}_{qb}^T\mathbf{q}) \tag{13}$$

With $\mathbf{b}$ eliminated using Eq. (13), Eq. (11) reduces to

$$(\mathcal{G} + \lambda I_{4\times 4})\mathbf{q} = \mathbf{g} \tag{14a}$$

$$\mathbf{q}^T\mathbf{q} = 1 \tag{14b}$$

where

$$\mathcal{G} \triangleq \mathcal{B}_{qq} - \mathcal{B}_{qb}\mathcal{B}_{bb}^{-1}\mathcal{B}_{qb}^{T} \tag{15a}$$

$$\mathbf{g} \triangleq \mathbf{c} - \mathcal{B}_{qb}\mathcal{B}_{bb}^{-1}\mathbf{d} \tag{15b}$$

Equation (14) is the key equation to solve in one-quaternion data fusion. Given $\mathbf{x}_i$ and $\mathcal{W}_i$, the one-quaternion data fusion procedure is as follows:

1. Compute $\mathcal{B}_{qq}$, $\mathcal{B}_{qb}$, $\mathcal{B}_{bb}$, $\mathbf{c}$, $\mathbf{d}$ using Eq. (12)
2. Compute $\mathcal{G}$ and $\mathbf{g}$ using Eq. (15)
3. Solve Eq. (14) for the optimal $\mathbf{q}^*$ and λ^*
4. Compute the optimal $\mathbf{b}^*$ using Eq. (13)

Note that for nonzero $\mathbf{g}$, $\mathbf{q}^*$ and $-\mathbf{q}^*$ cannot both be solutions of Eq. (14). The Lagrange multiplier λ^* is known to be the maximum real eigenvalue of an 8×8 asymmetric matrix or the maximum root of the 8^{th}-degree secular equation [3]. Given λ^*, the optimal $\mathbf{q}^*$ can be obtained by solving $(\mathcal{G} + \lambda^* I_{4\times 4})\mathbf{q}^* = \mathbf{g}$ [3].

3 Multi-quaternion Data Fusion

In multi-quaternion data fusion, the state vector is assumed to consist of m quaternions $\mathbf{q}^{(j)}$, $j = 1, \ldots, m$, with $\mathbf{q}^{(j)T}\mathbf{q}^{(j)} = 1$, and a set of unconstrained quantities $\mathbf{b}$. Define

$$\mathbf{x} \triangleq \begin{bmatrix} \mathbf{Q} \\ \\ \mathbf{b} \end{bmatrix} \tag{16a}$$

$$\mathbf{x}_i \triangleq \begin{bmatrix} \mathbf{Q}_i \\ \\ \mathbf{b}_i \end{bmatrix} \tag{16b}$$

$$\mathbf{Q} \triangleq \begin{bmatrix} \mathbf{q}^{(1)} \\ \vdots \\ \mathbf{q}^{(m)} \end{bmatrix} \tag{16c}$$

$$\mathbf{Q}_i \triangleq \begin{bmatrix} \mathbf{q}_i^{(1)} \\ \vdots \\ \mathbf{q}_i^{(m)} \end{bmatrix} \tag{16d}$$

$$\Sigma(\mathbf{Q}_i) \triangleq \begin{bmatrix} \Xi\left(\mathbf{q}_i^{(1)}\right) & & \\ & \ddots & \\ & & \Xi\left(\mathbf{q}_i^{(m)}\right) \end{bmatrix} \tag{16e}$$

Note that $\Sigma(\mathbf{Q}_i)$ is a block-diagonal matrix. The loss function is of the form of Eq. (8), repeated here:

$$J(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{n} \Delta\mathbf{x}_i^T \mathcal{W}_i \Delta\mathbf{x}_i \tag{17}$$

where

$$\Delta\mathbf{x}_i \triangleq \begin{bmatrix} \Sigma^T(\mathbf{Q}_i)\mathbf{Q} \\ \\ \mathbf{b} - \mathbf{b}_i \end{bmatrix} \tag{18a}$$

$$\mathcal{W}_i \triangleq \begin{bmatrix} \mathcal{W}_{QQ_i} & \mathcal{W}_{Qb_i} \\ \\ \mathcal{W}_{Qb_i}^T & \mathcal{W}_{bb_i} \end{bmatrix} \tag{18b}$$

Augmenting the constraint function with the m quaternion constraints gives

$$J(\mathbf{x}) = \frac{1}{2}\sum_{i=1}^{n} \Delta\mathbf{x}_i^T \mathcal{W}_i \Delta\mathbf{x}_i + \frac{1}{2}\sum_{j=1}^{m} \lambda_j \left(\mathbf{q}^{(j)T}\mathbf{q}^{(j)} - 1\right) \tag{19}$$

with λ_j, $j = 1, \ldots, m$, the Lagrange multipliers.

The necessary conditions for minimization of Eq. (19) are

$$\begin{bmatrix} \mathcal{B}_{QQ} + \Lambda & \mathcal{B}_{Qb} \\ \\ \mathcal{B}_{Qb}^T & \mathcal{B}_{bb} \end{bmatrix} \begin{bmatrix} \mathbf{Q} \\ \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{C} \\ \\ \mathbf{d} \end{bmatrix} \tag{20a}$$

$$\mathbf{q}^{(j)T}\mathbf{q}^{(j)} = 1,\ j = 1, \ldots, m \tag{20b}$$

where

$$\Lambda = \begin{bmatrix} \lambda_1 I_{4\times 4} & & \\ & \ddots & \\ & & \lambda_m I_{4\times 4} \end{bmatrix} \tag{21}$$

$$\mathcal{B}_{QQ} \triangleq \sum_{i=1}^{n} \Sigma(\mathbf{Q}_i)\mathcal{W}_{QQ_i}\Sigma^T(\mathbf{Q}_i) \tag{22a}$$

$$\mathcal{B}_{Qb} \triangleq \sum_{i=1}^{n} \Sigma(\mathbf{Q}_i)\mathcal{W}_{Qb_i} \tag{22b}$$

$$\mathcal{B}_{bb} \triangleq \sum_{i=1}^{n} \mathcal{W}_{bb_i} \tag{22c}$$

$$\mathbf{C} \triangleq \sum_{i=1}^{n} \Sigma(\mathbf{Q}_i)\mathcal{W}_{Qb_i}\mathbf{b}_i \tag{22d}$$

$$\mathbf{d} \triangleq \sum_{i=1}^{n} \mathcal{W}_{bb_i}\mathbf{b}_i \tag{22e}$$

Solving the second subequation of Eq. (20a) leads to

$$\mathbf{b} = \mathcal{B}_{bb}^{-1}(\mathbf{d} - \mathcal{B}_{Qb}^T\mathbf{Q}) \tag{23}$$

With $\mathbf{b}$ eliminated using Eq. (23), Eq. (20a) reduces to

$$(\mathcal{H} + \Lambda)\mathbf{Q} = \mathbf{h} \tag{24a}$$

$$\mathbf{q}^{(j)T}\mathbf{q}^{(j)} = 1,\ j = 1, \ldots, m \tag{24b}$$

where

$$\mathcal{H} \triangleq \mathcal{B}_{QQ} - \mathcal{B}_{Qb}\mathcal{B}_{bb}^{-1}\mathcal{B}_{Qb}^{T} \tag{25a}$$

$$\mathbf{h} \triangleq \mathbf{C} - \mathcal{B}_{Qb}\mathcal{B}_{bb}^{-1}\mathbf{d} \tag{25b}$$

Equation (24) is the key equation to solve in multi-quaternion data fusion. Given $\mathbf{x}_i$ and $\mathcal{W}_i$, the multi-quaternion data fusion procedure is as follows:

1. Compute $\mathcal{B}_{QQ}$, $\mathcal{B}_{Qb}$, $\mathcal{B}_{bb}$, $\mathbf{C}$, $\mathbf{d}$ using Eq. (22)
2. Compute $\mathcal{H}$ and $\mathbf{h}$ using Eq. (25)
3. Solve Eq. (24) for the optimal $\mathbf{Q}^*$ and λ_j^*, $j = 1, \ldots, m$
4. Compute the optimal $\mathbf{b}^*$ using Eq. (23)

Note that for nonzero $\mathbf{h}$, if $\mathbf{Q}^*$ is the solution of Eq. (24), its "conjugate" with one or more $\mathbf{q}^{(j)*}$ replaced by $-\mathbf{q}^{(j)*}$ is not a solution of Eq. (24) because the attitude error vectors in the loss function given by Eq. (17) are dependent on the signs of the quaternions.

Equation (24) or the equivalent of it may be solved using an iterative gradient-based algorithm, but it cannot guarantee that $\mathbf{Q}^*$ and λ_j^* are globally optimal. Since little is known about λ_j^*, choosing an appropriate initial guess for the gradient-based algorithm is not easy.

Noting that Eq. (24) is a system of polynomial equations in $\mathbf{Q}$ and λ_j, we are motived to solve Eq. (24) using the homotopy continuation method [2,4], which is capable of finding all the isolated solutions, real and complex, of the polynomial system. The globally optimal $\mathbf{Q}^*$ and λ^* are then given by the real solution that minimizes $J(\mathbf{x})$. The advantage of the method is that it guarantees that the globally optimal solution is one of the isolated solutions as long as it exists and is unique. The detail of the numerical solution is given in the next section.

4 Numerical Solution for Multi-quaternion Data Fusion

Define the $5m$-dimensional vector

$$\mathbf{y} \triangleq \begin{bmatrix} \mathbf{Q} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{q}^{(1)} \\ \vdots \\ \mathbf{q}^{(m)} \\ \lambda_1 \\ \vdots \\ \lambda_m \end{bmatrix} \tag{26}$$

where m is the number of quaternions in the state vector. The $5m$-dimensional vector $\mathbf{y}$ is required to satisfy the following $5m$ polynomial equations (also Eq. (24)):

$$(\mathcal{H} + \Lambda)\mathbf{Q} = \mathbf{h} \tag{27a}$$

$$\mathbf{q}^{(j)T}\mathbf{q}^{(j)} = 1,\ j = 1, \ldots, m \tag{27b}$$

For a well-posed data fusion problem, this polynomial system has a finite number of isolated solutions, which can be found via homotopy continuation [2,4]. The idea of homotopy continuation is to cast the target polynomial system in a parameterized family of systems, one of which (the start system) has known or easily found solutions[6,8]. After choosing this family, one chooses a path from the start system to the target system, constructs a homotopy between the two, and tracks the solution paths. The homotopy continuation method finds all the real and complex solutions to a polynomial system. Since only real solutions of the polynomial system are of interest, the complex solutions found by the solver are discarded.

For illustration purposes, an example of homotopy continuation of one equation in one unknown is given[6]. To solve

$$f(x) = x^5 + ax + b = 0 \tag{28}$$

where a and b are two constants, one may construct the continuation equation as

$$h(x,t) = x^5 + atx + [tb - (1-t)q^5], \quad 0 \le t \le 1 \tag{29}$$

where q is a complex constant. The target system (the original equation) corresponds to $t = 1$ and the start system corresponds to $t = 0$, given by

$$h(x,0) = x^5 - q^5 \tag{30}$$

The five complex solutions to the start system are obvious. The solution of $h(x,t)$ can be viewed as a function of t, denoted by $x(t)$, $0 \le t \le 1$. Geometrically, $x(t)$ are paths originating from $x(0)$ and ending at $x(1)$. With $x(0)$ given, $x(\Delta t)$, where Δt is a small step size, can be found using the Euler or Newton method. Step by step, the paths are tracked until $t = 1$.

Although the idea of homotopy continuation is simple, several important issues need to be handled with care. These include but are not limited to determination of the number of paths, path crossing, path divergence to infinity, and singular solutions[6]. A singular solution has a singular Jacobian matrix, defined as the derivatives of the equations with respect to the unknowns[6]. Thanks to homotopy continuation based solvers such as HOM4PS[4], Bertini[2], PHCpack[9], and HomLab[8], solving a system of polynomial equations via homotopy continuation is easy for a user, who only needs to provide a model description of the polynomial system that can be processed by the solver but does not need to provide a starter system.

By definition, the real solutions of the polynomial system in Eq. (27) are the stationary points of the constrained minimization problem. The globally

optimal solution is the one that minimizes the loss function. The local minimizers of the minimization problem are the stationary points with positive definite Hessian. Because the loss function is quadratic in the attitude quaternions, the Hessian without considering the m quaternion constraints is $(\mathcal{H} + \Lambda^*)$, where Λ^* is evaluated at the $\boldsymbol{\lambda}^*$ in $\mathbf{y}^*$. The dimension of $(\mathcal{H} + \Lambda^*)$ is $(4m) \times (4m)$. Since one attitude only has three degrees of freedom and m attitudes only have $3m$ degrees of freedom, this Hessian should not be used. Instead, a $(3m) \times (3m)$ Hessian is defined as

$$H^* = \mathcal{P}^{*T}(\mathcal{H} + \Lambda^*)\mathcal{P}^* \tag{31}$$

with the $(3m) \times (4m)$ $\mathcal{P}^*$ satisfying

$$\mathcal{P}^{*T}\mathcal{P}^* = I_{(3m)\times(3m)} \tag{32}$$

$$\mathcal{P}^{*T}\mathcal{C}^* = 0_{(3m)\times m} \tag{33}$$

where

$$\mathcal{C}^* = \begin{bmatrix} \mathbf{q}^{*(1)} & \cdots & \mathbf{0}_{4\times 1} \\ \vdots & \ddots & \vdots \\ \mathbf{0}_{4\times 1} & \cdots & \mathbf{q}^{*(m)} \end{bmatrix} \tag{34}$$

The columns of the $(4m) \times m$ matrix $\mathcal{C}^*$ correspond to the gradients of the quaternion constraints. The matrix $\mathcal{P}^*$ can be found using the QR decomposition of $\mathcal{C}^*$. A stationary point is a local minimizer if H^* is positive definite.

A stationary point or local minimizer is the global minimizer if gives the global minimum of the loss function of multi-quaternion data fusion, whose data-dependent part is equivalent to

$$J_{eq}(\mathbf{Q}) = \frac{1}{2}\mathbf{Q}^T\mathcal{H}\mathbf{Q} - \mathbf{h}^T\mathbf{Q} \tag{35}$$

The loss functions J and J_{eq} have the same stationary points and minima, but they are not equal to each other in general. Evaluating the equivalent loss function at a local minimum and substituting Eq. (27) gives

$$\begin{aligned} J_{eq}(\mathbf{Q}^*) &= \frac{1}{2}\mathbf{Q}^{*T}\mathcal{H}\mathbf{Q}^* - \mathbf{h}^T\mathbf{Q}^* \\ &= \frac{1}{2}\mathbf{Q}^{*T}\mathcal{H}\mathbf{Q}^* - \mathbf{Q}^{*T}(\mathcal{H} + \Lambda^*)\mathbf{Q}^* \\ &= -\frac{1}{2}\mathbf{Q}^{*T}\mathcal{H}\mathbf{Q}^* - \sum_{j=1}^{m}\lambda_j^* \end{aligned} \tag{36}$$

An alternative form of J_{eq} is given by

$$J_{eq} = -\frac{1}{2}\mathbf{Q}^{*T}\mathbf{h} - \frac{1}{2}\sum_{j=1}^{m}\lambda_j^* \tag{37}$$

It follows that if $\mathbf{h} = \mathbf{0}$, $J_{eq} = -1/2\sum_{j=1}^{m}\lambda_j^*$, completely determined by the optimal Lagrange multipliers λ_j^*. In this case, the globally optimal solution has the maximum sum of the associated Lagrange multiplers.

To find the globally optimal $\mathbf{Q}^*$ and λ_j^*, $j = 1, \ldots, m$, the following procedure is used:

1. Find all real solutions $\mathbf{y}^*$ of the polynomial system given by Eq. (27)
2. Extract $\mathbf{Q}^*$ and λ_j^*, $j = 1, \ldots, m$, from $\mathbf{y}^*$
3. For $j = 1, \ldots, m$, find the set of $\mathbf{Q}^*$ and λ_j^* with the minimum value of the equivalent loss function given by Eq. (35)

5 Numerical Example

An example of fusing two estimates of a 11-dimensional state vector is presented. The 11-dimensional state vector consists of two four-dimensional quaternions and a three-dimensional bias vector (in deg/hr). The optimal quaternions and gyro bias vector minimizes the loss function given by Eq. (17).

The numerical result of a typical run is given, in which the numerical values of $\mathbf{x}_1$ and $\mathbf{x}_2$ are

$$\mathbf{x}_1 = \begin{bmatrix} \mathbf{q}_1^{(1)} \\ \mathbf{q}_1^{(2)} \\ \mathbf{b}_1 \end{bmatrix} = \begin{bmatrix} -0.000277382573244 \\ -0.000204231576920 \\ 0.000192024591025 \\ 0.999999922237461 \\ -0.000351386991909 \\ 0.000409974757656 \\ 0.500076203448043 \\ 0.865981234896502 \\ 2.498888167313556 \\ 1.419147803026932 \\ 1.515444247510682 \end{bmatrix}, \mathbf{x}_2 = \begin{bmatrix} \mathbf{q}_2^{(1)} \\ \mathbf{q}_2^{(2)} \\ \mathbf{b}_2 \end{bmatrix} = \begin{bmatrix} 0.000460180571557 \\ -0.000137458850112 \\ 0.000082907410592 \\ 0.999999881232627 \\ 0.000201400729929 \\ 0.000114725959778 \\ 0.499722080513250 \\ 0.866185770215148 \\ 4.262501700209700 \\ 2.678350976663482 \\ -0.290141003402079 \end{bmatrix} \tag{38}$$

The individual blocks of the weighting matrices for the two estimates as defined by Eq. (18b) are given by

$$\mathcal{W}_{QQ_1} = \begin{bmatrix} 2.6028 & -1.1934 & -0.0338 & 0.0005 & -0.2992 & 0.4473 & -0.4107 & 0.2368 \\ -1.1934 & 2.6220 & 0.0618 & 0.0002 & 0.3700 & -0.3078 & 0.4449 & -0.2566 \\ -0.0338 & 0.0618 & 1.6481 & -0.0003 & 0.0046 & -0.0102 & 0.0071 & -0.0041 \\ 0.0005 & 0.0002 & -0.0003 & 0.0000 & -0.0000 & 0.0001 & -0.0000 & 0.0000 \\ -0.2992 & 0.3700 & 0.0046 & -0.0000 & 1.8740 & 0.3170 & 0.0723 & -0.0412 \\ 0.4473 & -0.3078 & -0.0102 & 0.0001 & 0.3170 & 1.4845 & -0.1206 & 0.0691 \\ -0.4107 & 0.4449 & 0.0071 & -0.0000 & 0.0723 & -0.1206 & 1.0865 & -0.6273 \\ 0.2368 & -0.2566 & -0.0041 & 0.0000 & -0.0412 & 0.0691 & -0.6273 & 0.3622 \end{bmatrix} \times 10^7 \tag{39}$$

$$\mathcal{W}_{QQ_2} = \begin{bmatrix} 2.6470 & 0.2882 & -0.1381 & -0.0012 & 0.7717 & 0.0829 & 1.4650 & -0.8454 \\ 0.2882 & 1.7538 & -0.0554 & 0.0001 & 0.3028 & 0.0258 & 0.3218 & -0.1858 \\ -0.1381 & -0.0554 & 1.2859 & -0.0001 & -0.1252 & -0.0097 & -0.0964 & 0.0556 \\ -0.0012 & 0.0001 & -0.0001 & 0.0000 & -0.0003 & -0.0000 & -0.0006 & 0.0004 \\ 0.7717 & 0.3028 & -0.1252 & -0.0003 & 1.8273 & -0.0658 & 0.4699 & -0.2715 \\ 0.0829 & 0.0258 & -0.0097 & -0.0000 & -0.0658 & 1.2899 & -0.0125 & 0.0071 \\ 1.4650 & 0.3218 & -0.0964 & -0.0006 & 0.4699 & -0.0125 & 2.8015 & -1.6164 \\ -0.8454 & -0.1858 & 0.0556 & 0.0004 & -0.2715 & 0.0071 & -1.6164 & 0.9326 \end{bmatrix} \times 10^7 \tag{40}$$

$$\mathcal{W}_{Qb_1} = \begin{bmatrix} -87.1316 & 718.0253 & -7.1995 \\ 0.1239 & -17.8754 & -34.1459 \\ -8.9145 & 5.1567 & -718.6707 \\ -405.9057 & 29.0107 & -0.2878 \\ 189.8244 & -194.0680 & 237.5290 \\ -136.9964 & -141.7521 & 135.0680 \\ 2.7172 & -0.0123 & 25.9761 \\ -41.2554 & 36.2763 & -20.9183 \end{bmatrix} \tag{41}$$

$$\mathcal{W}_{Qb_2} = \begin{bmatrix} 40.2161 & 52.0082 & -26.7553 \\ -0.0091 & 151.4475 & 18.2024 \\ 360.3686 & -207.9424 & -460.4823 \\ -178.0111 & 73.2052 & 0.1814 \\ -394.2257 & -29.2972 & -255.2308 \\ 147.3439 & -663.7105 & -257.5989 \\ 106.0858 & 0.2612 & -571.5016 \\ -42.6669 & -377.3766 & 217.8556 \end{bmatrix} \tag{42}$$

$$\mathcal{W}_{bb_1} = \begin{bmatrix} 0.2034 & 0.0278 & 0.0002 \\ 0.0278 & 5.4026 & 0.0075 \\ 0.0002 & 0.0075 & 0.1230 \end{bmatrix} \tag{43}$$

$$\mathcal{W}_{bb_2} = \begin{bmatrix} 0.2462 & -0.0091 & -0.0131 \\ -0.0091 & 0.1299 & 0.0334 \\ -0.0131 & 0.0334 & 0.1574 \end{bmatrix} \tag{44}$$

From the estimates and weighting matrices, the corresponding polynomial system in Eq. (27) is formed, where $\mathcal{H}$ and $\mathbf{h}$ are given by

$$\mathcal{H} = \begin{bmatrix} 4.9966 & -0.9375 & -0.1396 & -0.0006 & 0.3153 & 0.5017 & 0.9629 & -0.5558 \\ -0.9375 & 4.2338 & 0.0170 & 0.0003 & 0.6291 & -0.2846 & 0.6949 & -0.4009 \\ -0.1396 & 0.0170 & 2.9294 & -0.0004 & -0.0986 & -0.0165 & -0.0739 & 0.0426 \\ -0.0006 & 0.0003 & -0.0004 & 0.0000 & -0.0003 & 0.0000 & -0.0006 & 0.0004 \\ 0.3153 & 0.6291 & -0.0986 & -0.0003 & 3.5923 & 0.2349 & 0.4678 & -0.2697 \\ 0.5017 & -0.2846 & -0.0165 & 0.0000 & 0.2349 & 2.7710 & -0.1418 & 0.0812 \\ 0.9629 & 0.6949 & -0.0739 & -0.0006 & 0.4678 & -0.1418 & 3.8214 & -2.2053 \\ -0.5558 & -0.4009 & 0.0426 & 0.0004 & -0.2697 & 0.0812 & -2.2053 & 1.2726 \end{bmatrix} \times 10^7 \tag{45}$$

$$\mathbf{h} = \begin{bmatrix} 28.9572 \\ -510.8195 \\ -17.3103 \\ -0.1430 \\ 184.0905 \\ 16.0382 \\ 347.8748 \\ -200.7379 \end{bmatrix} \tag{46}$$

Given $\mathcal{H}$ and $\mathbf{h}$, Eq. (27) is solved using the polyhedral homotopy continuation method of HOM4PS 2.0 [4], which is chosen for its high speed relative to the other homotopy continuation based solvers. On a Macintosh computer with a 3 GHz Intel Core 2 Duo processor, the real and complex solutions of the equations are found in less then 0.5 seconds.

For this specific run, 52 solutions are real and 12 solutions are complex. Note that because $\mathbf{h}$ is nonzero, $[\mathbf{q}^{(1)T}, \mathbf{q}^{(2)T}]^T$, $[-\mathbf{q}^{(1)T}, \mathbf{q}^{(2)T}]^T$, $[\mathbf{q}^{(1)T}, -\mathbf{q}^{(2)T}]^T$, and $[-\mathbf{q}^{(1)T}, -\mathbf{q}^{(2)T}]^T$ are not equivalent. Recall that they yield different attitude error vectors in the loss function. If one of them solves Eq. (27), the other three do not.

It should be pointed out that the number of real solutions depends on both $\mathcal{H}$ and $\mathbf{h}$. For different values of $\mathcal{H}$ and $\mathbf{h}$, the observed number of real solutions varies from 48 to 64 but is always even. The observed total number of real and complex solutions is always 64, however.

The local minimizers are found by checking the positive definiteness of the Hessians given by Eq. (31) at the stationary points. The number of local minimizers (including the global minimizer) is four in all runs and does not depend on the values of $\mathcal{H}$ and $\mathbf{h}$. The global minimizer as well as the other three local minimizers for this specific run are given in Table 1. The data of the global minimizer are given in column 1. The data of the three local minimizers are in columns 2-4.

The 8×8 matrices $(\mathcal{H}+\Lambda^*)$ corresponding to the four local minimizers have six large positive eigenvalues and two small eigenvalues. The two small eigenvalues of $(\mathcal{H}+\Lambda^*)$ are seven orders of magnitude less than the six large eigenvalues. Both small eigenvalues are positive for the global minimizer. For the other three local minimizers, one or two of the small eigenvalues are negative. The eigenvalues of

Table 1. Global and Local Minimizers

$\mathbf{Q}^*$	−0.000082243849	0.000071570723	−0.000086071033	0.000075397906
	0.000070491751	−0.000103146338	0.000077016986	−0.000109671573
	−0.000131920776	0.000131512371	−0.000143351102	0.000142942697
	−0.999999985432	0.999999983471	−0.999999983055	0.999999980927
	0.000017062159	−0.000003434267	−0.000141582193	0.000155210084
	−0.000187251357	0.000187114822	0.000255151458	−0.000255287992
	−0.499941870375	0.499960737563	0.499730976390	−0.499712105367
	−0.866058941927	0.866048050556	0.866180619784	−0.866191504511
$\boldsymbol{\lambda}^*$	−4.953412878868	−5.016727156338	−5.452070929468	−5.515385202667
	−4.015817997239	−4.006374995955	−4.505033071226	−4.514476066556
$\mathbf{b}^*$	3.776978219338	3.184611496826	3.660908235836	3.300681480455
	1.417500286987	1.453817987354	1.394096956034	1.477221318092
	1.442498556379	−0.205862220233	0.712077571903	0.524558761517
J_{eq}	**4.464238890976**	4.518110168723	4.953453955162	5.026211225024
J	**4.815951252845**	4.869822531936	5.305166316980	5.377923588248

the 6×6 Hessians are positive for all four local minimizers (including the global minimizer).

6 Conclusions

In quaternion data fusion, the optimal quaternion estimate is usually obtained as the solution to a minimization problem subject to one or more quaternion norm constraints. The necessary condition for the local minima of the minimization problem is a system of polynomial equations with a finite number of isolated real and complex solutions, which can be all found via homotopy continuation. The global minimizer is then chosen from them as the one that minimizes the loss function of the minimization problem. The homotopy continuation based quaternion data fusion method does not depend on the initial guess of the state or Lagrange multipliers and is guaranteed to find the globally optimal solution of the minimization problem. The method is appropriate for small-scale quaternion data fusion problems.

References

1. Andrle, M.S., Crassidis, J.L., Linares, R., Cheng, Y., Hyun, B.: Deterministic relative attitude determination of three-vehicle formations. Journal of Guidance, Control, and Dynamics 32(4), 1077–1088 (2009)
2. Bates, D.J., Hauenstein, J.D., Sommese, A.J., Wampler, C.W.: Bertini: Software for numerical algebraic geometry, `http://www.nd.edu/~sommese/bertini`
3. Crassidis, J.L., Cheng, Y., Nebelecky, C.K., Fosbury, A.M.: Decentralized attitude estimation using a quaternion covariance intersection approach. Journal of the Astronautical Sciences 57(1-2), 113–128 (2009)
4. Lee, T.L., Li, T.Y., Tsai, C.H.: Hom4PS-2.0, a software package for solving polynomial systems by the polyhedral homotopy continuation. Computing 83, 109–133 (2008)

5. Markley, F.L., Cheng, Y., Crassidis, J.L., Oshman, Y.: Averaging quaternions. Journal of Guidance, Control, and Dynamics 30(4), 1193–1196 (2007)
6. Morgan, A.: Solving Polynomial Systems Using Continuation for Engineering and Scientific Problems. Englewood Cliffs, Prentice Hall (1987)
7. Shuster, M.D.: A survey of attitude representations. Journal of the Astronautical Sciences 41(4), 439–517 (1993)
8. Sommese, A.J., Wampler, C.W.: The Numerical Solution of Systems of Polynomials Arising in Engineering and Science. World Scientific, River Edge (2005)
9. Verschelde, J.: Algorithm 795: PHCpack: a general-purpose solver for polynomial systems. ACM Transactions on Mathematical Software 25(2), 251–276 (1999)

Linear Covariance Techniques for Closed-Loop Attitude Determination and Control Analysis

Noam Leiter[1,⋆] and David K. Geller[2,⋆⋆]

[1] Technion-IIT, Haifa 32000, Israel
noaml@technion.ac.il
[2] Utah State University, Logan, Utah, 84322, USA
david.geller@usu.edu

Abstract. During the 2011-2012 winter semester, graduate students in the Department of Aerospace Engineering, Technion were asked to reproduce the results of an attitude estimation paper by Prof. Itzhack Bar-Itzhack, "True Covariance Simulation of the EUVE Update Filter", as a homework assignment for a special topics course on linear covariance analysis. The students reproduced both the expected filter errors and the true filter errors as reported by Bar-Itzhack using covariance analysis. Bar Itzhack's work was then extended to determine the closed-loop pointing/control errors, again using linear covariance techniques. The control problem included star-tracker and gyro errors, magnetic torquer actuation errors, random disturbance moments, a suboptimal Kalman filter with model replacement, and a simple proportional-derivative control law. Using an augmented state formulation, covariance techniques were used to determine the variances of the expected and true attitude estimation errors, the variances of the true pointing errors of the closed-loop system, and the variance of the required control effort. Results were verified by nonlinear Monte Carlo analysis. The linear covariance analysis proved to be a very useful and fast analysis tool for the preliminary design of attitude determination and control systems.

Keywords: Linear Covariance Techniques, Attitude Control.

1 Introduction

In 1989, Bar-Itzhack and Harman presented a paper, "True Covariance Simulation of the EUVE Update Filter", at the Flight Mechanics/Estimation Symposium at the NASA Goddard Space Flight Center[2]. The Extreme Ultra-Violet Explorer (EUVE) attitude estimation problem involved 24 true states - attitude (3), misalignment states (12) of four sensors (two star-trackers, one sun sensor, and one gyro package), gyro bias (3), gyro scale-factor (3), and gyro non-orthogonal misalignment (3) - as well as gyro noise and gyro bias noise. The proposed suboptimal 6 state attitude estimation filter consisted of 6 states - attitude (3) and gyro bias

⋆ Graduate Student, Department of Aerospace Engineering.
⋆⋆ Visiting Professor, Technion's Department of Aerospace Engineering.

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_24

(3). The Bar-Itzhack paper clearly demonstrated the usefulness of true covariance analysis in determining the difference between the suboptimal filter covariance and the true estimation error covariance.

These results are extended in this paper by adding a closed-loop control system to the problem, formulating the associated closed-loop covariance analysis equations, and determining the covariance of the closed-loop control/pointing errors and the covariance of the control effort. The closed-loop control problem included magnetic torquers, a proportional-derivative (PD) control law, and nine new true states - attitude rate (3), control torque misalignment (3), and control torque biases (3). Random torque disturbances and control torque actuation noise were also incorporated. The original 6-state suboptimal filter was not modified.

Historically, linear covariance theory has been applied to general estimation problems [8,5,16], design and analysis of orbit determination algorithms [24,12,18,7], inertial navigation systems[23,14,6,13], and attitude determination systems[2,15,10]. The covariance analysis approaches are more commonly known as *true covariance analysis*[2], *consider analysis*[27,1,24], or *generalized covariance analysis*[15,4]. In all cases however, the effects of closed-loop control on the overall performance of the system is not considered.

Recent developments in linear covariance theory have combined the developments of Battin [3] and Maybeck [16] with continuous feedback control and model replacement[19] (e.g., state propagation using gyro measurements) to produce linear covariance tools that can be applied to many different types of closed-loop GN&C problems[9]. Specific applications include autonomous rendezvous [9,25], powered lunar descent[17], and launch vehicle ascent trajectory analysis [20]. While linear covariance theory for closed-loop GN&C systems has been successfully applied to these applications, a clear demonstration of its value and usefulness in the context of a closed-loop spacecraft attitude control problem is warranted.

The objectives of this paper are to demonstrate the usefulness of linear covariance analysis in closed-loop attitude control system design and analysis, and to demonstrate the accuracy and efficiency of linear covariance analysis by direct comparison to nonlinear Monte Carlo analysis. The variance of the expected and true attitude estimation errors as reported in Bar Itzhack's original paper [2] will be reproduced, and the variances of the true attitude control errors and required control effort will be presented.

To develop a linear covariance simulation, the truth models for the dynamics, sensors, and actuators must first be defined. This is accomplished in Section 2. Biases, scale-factors, misalignment, etc., and other colored noise processes are incorporated into the truth models as additional true states. Next, the navigation state vector, and the associated navigation algorithms are defined in Section 3. The navigation algorithms are derived from a filter *design model* that is discarded after the navigation algorithms are developed. Note that the navigation design model and the truth models are generally similar, but not identical. Next, the control laws are defined and presented in Section 3. Here again, the guidance

and control algorithms are derived from a design model that is discarded once the algorithms are developed. The output of the guidance and control algorithms, along with potential compensation for actuator modeling errors, represents the actuator command defined in the truth model. All models and algorithms are linearized about the mean or nominal state time-history. Since the dynamics of the true states and the dynamics of the navigation states are coupled, an augmented state vector is created by appending the navigation state vector to the true state vector. This is described in Section 4 and results is a dimensionally large linear time-varying model of the entire closed-loop system. The mathematical details of this process are provided in [9]. The linear covariance analysis and non-linear Monte Carlo analysis of a closed-loop attitude determination and control system are presented in Section 5 followed by conclusions in Section 6.

2 The EUVE Attitude Model[1]

The EUVE was a NASA satellite launched to a low Earth orbit on June 7 1992, operating till January 31 2001. The EUVE was designed to perform two missions: 1) an all-sky survey of extreme ultraviolet sources, and 2) spectroscopy of particular ultraviolet sources.

In this study we examine the overall closed-loop performance of the navigation and control system during the all-sky survey. The EUVE payload for the all-sky survey consists of three telescopes pointing perpendicular to the body x-axis. During the survey the satellite is spin stabilized about the x-axis and is nominally aligned with the Earth-Sun vector, requiring a precession of approximately 1 [deg] per day. In the analysis that follows, the system is examined over a 600 [sec] time period, therefore the precession is negligible and the Earth-Sun vector is assumed constant.

2.1 EUVE Attitude Control Dynamics Model

The actual EUVE attitude control system includes two magneto-torquers, aligned with the x-body and z-body axes, two wobble control actuators and one passive x-axis nutation damper. Assuming rigid body motion, the angular dynamics are given by

$$J\dot{\boldsymbol{\omega}} = \mathbf{T}_{mag} + \mathbf{T}_{nutation} + \mathbf{T}_{wobb} + \mathbf{T}_{dist} - [\boldsymbol{\omega}\times]\, J\boldsymbol{\omega}$$

where $J = J_0 + \delta J$ is the inertia tensor of the EUVE in x-y-z body frame [26]:

$$J_0 = \begin{bmatrix} 850 & 0 & 0 \\ 0 & 700 & 0 \\ 0 & 0 & 700 \end{bmatrix} \left[kg \cdot m^2\right], \quad \delta J = -10 \cdot \begin{bmatrix} 0\,0\,1 \\ 0\,0\,0 \\ 1\,0\,0 \end{bmatrix} \left[kg \cdot m^2\right]$$

[1] The model described is not the model of the final EUVE mission. We have tried to reproduce the mission model as known to Bar-Itzhack 3 years before the launch.

$\mathbf{T}_{nutation}$ is the torque applied by the passive nutation damper with a damping time constant of 10 [hr][26]. This torque is dominant in stabilizing the spin x-axis when the Earth-Sun vector alignment is initially acquired, or during the spectroscopy mission when large magnitude attitude changes between targets are necessary. This torque is however negligible for the spin axis/x-axis regulation problem over the 600 [sec] time period of interest in this paper. $\mathbf{T}_{wobb}$ is the torque applied by the two wobble control actuators, which consist of two movable masses. The position of the masses is adjusted to suppress the wobble effect of the cross product moment of inertia [26].

$$\mathbf{T}_{wobb} \cong [\boldsymbol{\omega}\times]\, \delta J \boldsymbol{\omega} - \delta J \dot{\boldsymbol{\omega}}$$

Therefore the attitude dynamics examined in this study are

$$\dot{\boldsymbol{\omega}} = J_0^{-1} \left(\mathbf{T}_{mag} + \mathbf{T}_{dist} - [\boldsymbol{\omega}\times]\, J_0 \boldsymbol{\omega} \right) \tag{1}$$

where $\mathbf{T}_{dist}$ is a zero mean continuous Gaussian white noise random disturbance moment

$$\mathbf{T}_{dist} = \boldsymbol{\zeta}_d, \qquad E\left[\boldsymbol{\zeta}_d(t)\, \boldsymbol{\zeta}_d^T(t') \right] = S_d \cdot \delta(t - t'). \tag{2}$$

and $\mathbf{T}_{mag}$ is a torque applied by the magneto-torquers

$$\mathbf{T}_{mag} = \mathbf{d} \times \mathbf{B} = \mathbf{T}_N + \delta \mathbf{T}_{mag}$$

where $\mathbf{B}$ is Earth magnetic field and $\mathbf{d}$ is the magnetic dipole vector in the x-y-z body frame.

The nominal angular velocity of the EUVE in the survey mission is $\boldsymbol{\omega}_N = \begin{bmatrix} \Omega_N & 0 & 0 \end{bmatrix}^T$ and the nominal attitude dynamics are

$$\dot{\boldsymbol{\omega}}_N = 0 = J_0^{-1} \left(\mathbf{T}_N - [\boldsymbol{\omega}_N\times]\, J_0 \boldsymbol{\omega}_N \right) = J_0^{-1} \mathbf{T}_N.$$

Therefore the nominal control torque $\mathbf{T}_N$ is zero. The remaining perturbation dynamics are given by

$$\delta\boldsymbol{\omega} \equiv \boldsymbol{\omega} - \boldsymbol{\omega}_N \tag{3}$$

$$\delta\dot{\boldsymbol{\omega}} = J_0^{-1} \left(\delta\mathbf{T}_{mag} + \boldsymbol{\zeta}_d - [\boldsymbol{\omega}\times]\, J_0 \boldsymbol{\omega} \right) \tag{4}$$

and the perturbing torque is

$$\delta\mathbf{T}_{mag} = \left(I - [\boldsymbol{\epsilon}_{act}\times] \right) \delta\mathbf{u} + \mathbf{b}_{act} + \boldsymbol{\zeta}_{act} \tag{5}$$

where $\boldsymbol{\zeta}_{act}$ is zero mean continuous Gaussian white noise with the variance

$$E\left[\boldsymbol{\zeta}_{act}(t)\, \boldsymbol{\zeta}_{act}^T(t') \right] = S_{act} \cdot \delta(t - t'), \tag{6}$$

$\delta\mathbf{u}$ is the desired control torque, and $\mathbf{b}_{act}$ and $\boldsymbol{\epsilon}_{act}$ are the control bias and misalignment, initially random and constant thereafter, i.e., $\dot{\mathbf{b}}_{act} = \dot{\boldsymbol{\epsilon}}_{act} = 0$. The linearized attitude dynamics are then given by

$$\delta\dot{\boldsymbol{\omega}} = J_0^{-1} \left([J_0 \boldsymbol{\omega}_N\times] - [\boldsymbol{\omega}_N\times]\, J_0 \right) \delta\boldsymbol{\omega} + J_0^{-1} \left(\delta\mathbf{u} + \mathbf{b}_{act} + \boldsymbol{\zeta}_{act} + \boldsymbol{\zeta}_d \right) \tag{7}$$

and the attitude kinematics are governed according to the following differential equation[21]

$$\dot{q}_\theta = \frac{1}{2} q_\omega \otimes q_\theta$$

where q_θ is the attitude (unit) quaternion

$$q_\theta = \begin{bmatrix} q_0 \\ \mathbf{q} \end{bmatrix} = \begin{bmatrix} cos\frac{\theta}{2} \\ sin\frac{\theta}{2} \cdot \hat{\mathbf{n}} \end{bmatrix}, \qquad q_\theta^* \otimes q_\theta = \begin{bmatrix} 1 \\ \mathbf{0} \end{bmatrix} \tag{8}$$

$\hat{\mathbf{n}}$ is the axis of rotation and q_ω is the angular velocity quaternion:

$$q_\omega = \begin{bmatrix} 0 \\ \boldsymbol{\omega} \end{bmatrix} \tag{9}$$

The linearized attitude kinematics are given by

$$\delta\boldsymbol{\theta} \equiv \boldsymbol{\theta} - \boldsymbol{\theta}_N = [\delta\theta_x, \delta\theta_y, \delta\theta_z]^T \tag{10}$$

$$\delta\dot{\boldsymbol{\theta}} = \delta\boldsymbol{\omega} - [\boldsymbol{\omega}_N \times]\, \delta\boldsymbol{\theta} \tag{11}$$

where $\delta\boldsymbol{\theta}$ is a small angle rotation from the nominal vehicle attitude to the true vehicle attitude.

2.2 Star-Trackers Measurement Model

The EUVE is equipped with two star-trackers aligned with the body y-axis and z-axis, producing four measurements equivalent to the pixel locations of a single star in each star-tracker focal plane plus random noise ν_i. If the unit vectors to a single star in each of the two star-tracker frames are denoted by $\mathbf{s}_\mathbf{z}^{\mathbf{st_1}}$ and $\mathbf{s}_\mathbf{y}^{\mathbf{st_2}}$, the four star-tracker measurements are given by projections of the star-vectors onto the star-trackers focal planes plus random noise [2].

$$\begin{aligned} \tilde{z}_1 &= \mathbf{i}_{x_{st_1}}^T \mathbf{s}_\mathbf{z}^{\mathbf{st_1}} + \nu_1 \ , \quad \tilde{z}_2 = \mathbf{i}_{y_{st_1}}^T \mathbf{s}_\mathbf{z}^{\mathbf{st_1}} + \nu_2 \\ \tilde{z}_3 &= \mathbf{i}_{x_{st_2}}^T \mathbf{s}_\mathbf{y}^{\mathbf{st_2}} + \nu_3 \ , \quad \tilde{z}_4 = \mathbf{i}_{z_{st_2}}^T \mathbf{s}_\mathbf{y}^{\mathbf{st_2}} + \nu_4 \end{aligned} \tag{12}$$

The star-vector in the star-tracker coordinate frame can be related to the known star location in the inertial frame via the transformation

$$\mathbf{s}^{\mathbf{st}} = T_{\bar{st}2st} T_{B2\bar{st}} T_{\bar{B}2B} T_{I2\bar{B}} \mathbf{s}^{\mathbf{I}} \tag{13}$$

$T_{\bar{st}2st}$ is the transformation from the nominal star-tracker frame to the actual tracker frame

$$T_{\bar{st}2st} \cong I - [\boldsymbol{\phi} \times] \tag{14}$$

[2] It is assumed that at any given time a known star is acquired in the boresight of both trackers.

where $\boldsymbol{\phi}$ is a small unknown orthogonal tracker misalignment:

$$\boldsymbol{\phi} = [\phi_x, \phi_y, \phi_z]^T . \tag{15}$$

For both star-trackers the nominal misalignment is zero and the misalignment is initially random and constant thereafter, i.e., $\dot{\boldsymbol{\phi}}_1 = \dot{\boldsymbol{\phi}}_2 = 0$.

$T_{B2\bar{st}}$ is a known constant transformation from the body frame to the nominal star-tracker frame, $T_{I2\bar{B}}$ is a known transformation from the inertial frame to the nominal body frame, and $T_{\bar{B}2B}$ is the transformation from nominal body frame to the actual body frame

$$T_{\bar{B}2B} \cong I - [\delta\boldsymbol{\theta}\times] \tag{16}$$

The linearized star-tracker measurement equations [2] are then given

$$\delta z_1 = \left(\left[\mathbf{i}_{x_{st_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}}\right)^T \delta\boldsymbol{\theta} + \left(\left[\mathbf{i}_{x_{st_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}}\right)^T \boldsymbol{\phi}_1 + \nu_1 \tag{17}$$

$$\delta z_2 = \left(\left[\mathbf{i}_{y_{st_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}}\right)^T \delta\boldsymbol{\theta} + \left(\left[\mathbf{i}_{y_{st_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}}\right)^T \boldsymbol{\phi}_1 + \nu_2 \tag{18}$$

$$\delta z_3 = \left(\left[\mathbf{i}_{x_{st_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}}\right)^T \delta\boldsymbol{\theta} + \left(\left[\mathbf{i}_{x_{st_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}}\right)^T \boldsymbol{\phi}_2 + \nu_3 \tag{19}$$

$$\delta z_4 = \left(\left[\mathbf{i}_{z_{st_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}}\right)^T \delta\boldsymbol{\theta} + \left(\left[\mathbf{i}_{z_{st_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}}\right)^T \boldsymbol{\phi}_2 + \nu_4 \tag{20}$$

$$\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}} = T_{I2\bar{B}}\mathbf{s}_{\mathbf{1}}^{\mathbf{I}} \ , \ \ \mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}} = T_{I2\bar{B}}\mathbf{s}_{\mathbf{2}}^{\mathbf{I}} \tag{21}$$

where $\mathbf{s}_{\mathbf{1}}^{\mathbf{I}}$ and $\mathbf{s}_{\mathbf{2}}^{\mathbf{I}}$ are the known locations of two stars in the field of view of the z- and y-star-trackers, respectively.

2.3 Gyro Measurement Model

The gyro measurement is given by:

$$\tilde{\boldsymbol{\omega}} = (I - [\boldsymbol{\epsilon}_\omega\times] + U(\boldsymbol{\alpha}_\omega))\,\boldsymbol{\omega} \tag{22}$$

where $\boldsymbol{\epsilon}_\omega$ is a small orthogonal misalignment

$$\boldsymbol{\epsilon}_\omega = [\epsilon_x, \epsilon_y, \epsilon_z]^T \tag{23}$$

and $\boldsymbol{\alpha}_\omega$ is a small non-orthogonal misalignment:

$$\boldsymbol{\alpha}_\omega = [\alpha_x, \alpha_y, \alpha_z]^T , \qquad U(\boldsymbol{\alpha}_\omega) = \begin{bmatrix} 0 & \alpha_z & \alpha_y \\ 0 & 0 & \alpha_x \\ 0 & 0 & 0 \end{bmatrix} . \tag{24}$$

Adding bias, noise and scale factors produces

$$\tilde{\boldsymbol{\omega}} = (I + D(\mathbf{f}_\omega))\,(I - [\boldsymbol{\epsilon}_\omega\times] + U(\boldsymbol{\alpha}_\omega))\,\boldsymbol{\omega} + \mathbf{b}_\omega + \boldsymbol{\zeta}_\omega \tag{25}$$

where $\mathbf{f}_\omega$ is the scale factor vector

$$\mathbf{f}_\omega = [f_x, f_y, f_z]^T, \qquad D(\mathbf{f}_\omega) = diag(f_x, f_y, f_z) \tag{26}$$

and $\boldsymbol{\zeta}_\omega$ is zero mean continuous Gaussian white noise with the variance

$$E\left[\boldsymbol{\zeta}_\omega(t)\,\boldsymbol{\zeta}_\omega{}^T(t')\right] = S_\omega \cdot \delta(t-t'). \tag{27}$$

The gyro bias is a random process driven by zero mean continuous Gaussian white noise

$$\dot{\mathbf{b}}_\omega = \boldsymbol{\zeta}_b, \qquad E\left[\boldsymbol{\zeta}_b(t)\,\boldsymbol{\zeta}_b^T(t')\right] = S_b \cdot \delta(t-t'). \tag{28}$$

All misalignment and scale factors are initially random and constant thereafter

$$\dot{\boldsymbol{\epsilon}}_\omega = \dot{\boldsymbol{\alpha}}_\omega = \dot{\mathbf{f}}_\omega = 0 \tag{29}$$

If we define the measurement perturbation relative to the nominal state as

$$\delta\tilde{\boldsymbol{\omega}} \equiv \tilde{\boldsymbol{\omega}} - \boldsymbol{\omega}_N \tag{30}$$

the linearized gyro measurement equation is

$$\delta\tilde{\boldsymbol{\omega}} \cong \delta\boldsymbol{\omega} + \mathbf{b}_\omega + \boldsymbol{\zeta}_\omega + D(\boldsymbol{\omega}_N)\,\mathbf{f}_\omega + [\boldsymbol{\omega}_N\times]\,\boldsymbol{\epsilon}_\omega + W(\boldsymbol{\omega}_N)\,\boldsymbol{\alpha}_\omega \tag{31}$$

where

$$D(\boldsymbol{\omega}) \equiv diag(\boldsymbol{\omega}), \qquad W(\boldsymbol{\omega}) \equiv \begin{bmatrix} 0 & \omega_z & \omega_y \\ \omega_z & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{32}$$

2.4 State Space Representation

The state space of the true perturbations from the nominal state is:

$$\delta\mathbf{x} \equiv \mathbf{x} - \mathbf{x}_N = \left[\,\delta\boldsymbol{\theta}^T\ \delta\boldsymbol{\omega}^T\ \mathbf{b}_{act}^T\ \boldsymbol{\epsilon}_{act}^T\ \mathbf{b}_\omega^T\ \boldsymbol{\epsilon}_\omega^T\ \boldsymbol{\alpha}_\omega^T\ \mathbf{f}_\omega^T\ \boldsymbol{\phi}_1^T\ \boldsymbol{\phi}_2^T\,\right]^T \tag{33}$$

The linearized dynamics are given by

$$\delta\dot{\mathbf{x}} = A\cdot\delta\mathbf{x} + G\cdot\delta\mathbf{u} + B\cdot\boldsymbol{\zeta} \tag{34}$$

where $\boldsymbol{\zeta}$ is continuous white process noise

$$\boldsymbol{\zeta} = \left[\,\boldsymbol{\zeta}_\omega^T\ \boldsymbol{\zeta}_b^T\ \boldsymbol{\zeta}_{act}^T\ \boldsymbol{\zeta}_d^T\,\right]^T, \qquad E\left[\boldsymbol{\zeta}(t)\,\boldsymbol{\zeta}^T(t')\right] = S_\eta\cdot\delta(t-t'). \tag{35}$$

The linearized star-tracker measurement equation is

$$\delta\tilde{\mathbf{z}} = H\cdot\delta\mathbf{x} + \mathbf{v} \tag{36}$$

where the measurement noise is given by

$$\mathbf{v} = [\nu_1, \nu_2, \nu_3, \nu_4]^T, \qquad E[\mathbf{v}(t_i)\,\mathbf{v}(t_j)] = R\cdot\delta_{ij}. \tag{37}$$

and the linearized gyro measurement equation is

$$\delta\tilde{\boldsymbol{\omega}} = H_{gyro}\cdot\delta\mathbf{x} + \boldsymbol{\zeta}_\omega$$

A, G, B, H and H_{gyro} are provided in the appendix.

3 EUVE Navigation and Control Design

3.1 Reduced Order Filter Algorithm

The filter design model assumes that the true misalignments, scale factors and control bias are zero.

$$\mathbf{b}_{act_f} = \boldsymbol{\epsilon}_{act_f} = \boldsymbol{\epsilon}_{\omega_f} = \boldsymbol{\alpha}_{\omega_f} = \mathbf{f}_{\omega_f} = \boldsymbol{\phi}_{1_f} = \boldsymbol{\phi}_{2_f} = 0 \tag{38}$$

Thus, the design model is a reduction of the full EUVE model with only 9 states, $\delta\boldsymbol{\theta}_f$. $\delta\boldsymbol{\omega}_f$, and $\mathbf{b}_{\omega_f}$. The dynamics of the linearized filter design model are

$$\delta\dot{\boldsymbol{\theta}}_f \cong \delta\boldsymbol{\omega}_f - [\boldsymbol{\omega}_N\times]\,\delta\boldsymbol{\theta}_f \tag{39}$$

$$\delta\dot{\boldsymbol{\omega}}_f \cong J_0^{-1}\left([J_0\boldsymbol{\omega}_N\times] - [\boldsymbol{\omega}_N\times]\,J_0\right)\delta\boldsymbol{\omega}_f + J_0^{-1}\left(\delta\mathbf{u} + \boldsymbol{\zeta}_{act_f} + \boldsymbol{\zeta}_{d_f}\right) \tag{40}$$

$$\dot{\mathbf{b}}_{\omega_f} = \boldsymbol{\zeta}_{b_f} \tag{41}$$

where $\delta\boldsymbol{\theta}_f \equiv \boldsymbol{\theta}_f - \boldsymbol{\theta}_N$, and $\delta\boldsymbol{\omega}_f \equiv \boldsymbol{\omega}_f - \boldsymbol{\omega}_N$. The linearized star-tracker measurement equations are

$$\begin{aligned}
\delta z_{f_1} &= \left(\left[\mathbf{i}_{x_{st_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}}\right)^T\delta\boldsymbol{\theta}_f + \nu_{f_1}\ ,\quad \delta z_{f_2} = \left(\left[\mathbf{i}_{y_{st_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}}\right)^T\delta\boldsymbol{\theta}_f + \nu_{f_2}\\
\delta z_{f_3} &= \left(\left[\mathbf{i}_{x_{st_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}}\right)^T\delta\boldsymbol{\theta}_f + \nu_{f_3}\ ,\quad \delta z_{f_4} = \left(\left[\mathbf{i}_{z_{st_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}}\right)^T\delta\boldsymbol{\theta}_f + \nu_{f_4}
\end{aligned} \tag{42}$$

and the linearized gyro measurements are

$$\delta\tilde{\boldsymbol{\omega}}_f \equiv \tilde{\boldsymbol{\omega}}_f - \boldsymbol{\omega}_N \cong \delta\boldsymbol{\omega}_f + \mathbf{b}_{\omega_f} + \boldsymbol{\zeta}_{\omega_f} \tag{43}$$

A standard linearized Kalman filter algorithm can be developed to estimate $\delta\hat{\boldsymbol{\theta}}_f$, $\delta\hat{\boldsymbol{\omega}}_f$, and $\hat{\mathbf{b}}_{\omega_f}$ from the reduced order linearized dynamics in Eqs. 39-41 and the linearized measurement equations in Eqs. 42-43; however, we postpone the develop of the Kalman filter until later when model replacement [19] will be used to replace the rotational dynamics model with the gyro measurements.

3.2 The Control Law

For this study, a simple continuous regulator is used for the attitude control law[3].

$$\delta\mathbf{u} = -K_\theta\left(\hat{\boldsymbol{\theta}}_f - \boldsymbol{\theta}_N\right) - K_\omega\left(\hat{\boldsymbol{\omega}}_f - \boldsymbol{\omega}_N\right) = -K_\theta\delta\hat{\boldsymbol{\theta}}_f - K_\omega\delta\hat{\boldsymbol{\omega}}_f \tag{44}$$

where K_θ and K_ω are constant gain matrices derived from the controller frequency ω_c, damping ratio ξ, and vehicle inertia tensor J_0:

$$K_\theta = \omega_c^2\cdot J_0\ ,\quad K_\omega = 2\cdot\xi\cdot\omega_c\cdot J_0 \tag{45}$$

[3] This is not the control method used for the actual EUVE mission.

3.3 Reduced Order Filter Algorithm and Model Replacement

When the uncertainties in the rotational dynamics model (Eq. 40) are significant (e.g., spacecraft inertia and external disturbances), it is often prudent to replace the rotational dyanmics model with gyro data. In this situation, the dynamical model for the filter design becomes

$$\delta\dot{\boldsymbol{\theta}}_f \cong \delta\boldsymbol{\omega}_f - [\boldsymbol{\omega}_N\times]\,\delta\boldsymbol{\theta}_f \tag{46}$$

$$\dot{\mathbf{b}}_{\omega_f} = \boldsymbol{\zeta}_{b_f} \tag{47}$$

where

$$\delta\boldsymbol{\omega}_f = \delta\tilde{\boldsymbol{\omega}}_f - \mathbf{b}_{\omega_f} - \boldsymbol{\zeta}_{\omega_f} \tag{48}$$

and the estimate of the spacecraft angular velocity is simply

$$\delta\hat{\boldsymbol{\omega}}_f = \delta\tilde{\boldsymbol{\omega}}_f - \hat{\mathbf{b}}_{\omega_f} \tag{49}$$

When this estimate for the angular velocity is substituted into the control law in Eq. 44, the desired control moment is slightly modified and becomes

$$\delta\mathbf{u} = -K_\theta\delta\hat{\boldsymbol{\theta}}_f + K_\omega\hat{\mathbf{b}}_{\omega_f} - K_\omega\delta\tilde{\boldsymbol{\omega}}_f \tag{50}$$

where $\delta\tilde{\boldsymbol{\omega}}_f$ are the actual gyro measurements given by Eq. 31.

Using this model replacement, the state space and state dynamics for the filter design become

$$\delta\mathbf{x}_f = \left[\,\delta\boldsymbol{\theta}_f^T\ \mathbf{b}_{\omega_f}^T\,\right]^T \tag{51}$$

$$\delta\dot{\mathbf{x}}_f = A_f\cdot\delta\mathbf{x}_f + G_\omega\cdot\delta\tilde{\boldsymbol{\omega}}_f + B_f\cdot\boldsymbol{\zeta}_f \tag{52}$$

where A_f, G_ω, and B_f are provided in the appendix, and $\boldsymbol{\zeta}_f$ is zero mean continuous white noise

$$\boldsymbol{\zeta}_f = \begin{bmatrix}\boldsymbol{\zeta}_{\omega_f}\\ \boldsymbol{\zeta}_{b_f}\end{bmatrix}, \qquad E\left[\boldsymbol{\zeta}_f(t)\,\boldsymbol{\zeta}_f^T(t')\right] = S_f\cdot\delta(t-t') \tag{53}$$

The filter design model for the star-tracker measurements remains unchanged and repeated here for convenience

$$\delta\tilde{\mathbf{z}}_f = H_f\cdot\delta\mathbf{x}_f + \mathbf{v}_f \tag{54}$$

where H_f is provided in the appendix, and $\mathbf{v}_f$ is zero mean discrete white noise

$$\mathbf{v}_f = [\nu_1,\nu_2,\nu_3,\nu_4]^T, \qquad E\left[\mathbf{v}_f(t_i)\,\mathbf{v}_f(t_j)\right] = R_f\delta_{ij} \tag{55}$$

Using this new design model, a Kalman filter algorithm can be developed. First, the state propagation equations are given by

$$\delta\dot{\hat{\mathbf{x}}}_f = A_f\cdot\delta\hat{\mathbf{x}}_f + G_\omega\cdot\delta\tilde{\boldsymbol{\omega}} \tag{56}$$

Since the estimated filter navigation error is given by

$$\hat{\boldsymbol{\varepsilon}}_f \equiv \hat{\mathbf{x}}_f - \mathbf{x}_f = \delta\hat{\mathbf{x}}_f - \delta\mathbf{x}_f \tag{57}$$

$$\dot{\hat{\boldsymbol{\varepsilon}}}_f = A_f \cdot \hat{\boldsymbol{\varepsilon}}_f - B_f \cdot \boldsymbol{\zeta}_f \tag{58}$$

the navigation error is expected to have zero mean and covariance given by

$$P_f = E\left[\hat{\boldsymbol{\varepsilon}}_f \cdot \hat{\boldsymbol{\varepsilon}}_f^T\right]. \tag{59}$$

Thus, the filter state error covariance propagation is given by

$$\dot{P}_f = A_f P_f + P_f A_f^T + B_f S_f B_f^T \tag{60}$$

The Kalman filter update equations for the post measurement state and covariance corrections are

$$K_f = P_f H_f^T \left(H_f P_f H_f^T + R_f\right)^{-1} \tag{61}$$

$$\delta\hat{\mathbf{x}}_f^+ = \delta\hat{\mathbf{x}}_f + K_f\left(\delta\tilde{\mathbf{z}} - \delta\hat{\mathbf{z}}_f\right) \tag{62}$$

$$P_f^+ = \left(I - K_f H_f\right) P_f \left(I - K_f H_f\right)^T + K_f R_f K_f^T \tag{63}$$

where

$$\delta\hat{\mathbf{z}}_f = H_f \cdot \delta\hat{\mathbf{x}}_f \tag{64}$$

This filter is no longer optimal because the covariance propagation and gain calculation are now based on a reduced model [22] with model replacement [11]. One of the objectives of linear covariance analysis is to determine the effects of these suboptimal schemes on the true navigation error. This will be measured by comparing the filter covariance P_f to the true covariance P_{true} of the true filter state error

$$\delta\hat{\mathbf{x}}_f \equiv \hat{\mathbf{x}}_f - M\mathbf{x}_N \tag{65}$$

where M is a mapping from the entire true state to the true values of the reduced filter states. M is also provided in the appendix.

4 Linear Covariance Analysis

4.1 EUVE Linear Covariance Model

For linear covariance analysis, the linearized navigation state vector $\delta\hat{\mathbf{x}}_f$ (65) is appended to the true state vector $\delta\mathbf{x}$ (33) to form the augmented state vector $\mathbf{X} \in \mathbb{R}^{\mathbf{n}+\hat{\mathbf{n}}}$.

$$\mathbf{X} \equiv \begin{bmatrix} \delta\mathbf{x} \\ \delta\hat{\mathbf{x}}_f \end{bmatrix} \tag{66}$$

Thus we have $\delta\mathbf{x} = M_x\mathbf{X}$ and $\delta\hat{\mathbf{x}}_f = M_f\mathbf{X}$ where M_x and M_f are appropriate mappings from the augmenented state to the true state and navigation state, respectively. The dynamics for the augmented state vector are defined as

$$\dot{\mathbf{X}} = \mathcal{A}\mathbf{X} + \mathcal{B}\boldsymbol{\zeta} \tag{67}$$

$$\boldsymbol{\zeta} = \left[\, \boldsymbol{\zeta}_\omega^T \; \boldsymbol{\zeta}_b^T \; \boldsymbol{\zeta}_{act}^T \; \boldsymbol{\zeta}_d^T \,\right]^T \tag{68}$$

where

$$\mathcal{A} = \begin{bmatrix} A + GG_x & GG_f \\ A_{f_X} & A_f \end{bmatrix}, \quad \mathcal{B} = \begin{bmatrix} B + GB_u \\ B_{f_X} \end{bmatrix} \tag{69}$$

and where G_f, G_x, B_u, B_{f_X} and A_{f_X} are provided in the appendix. Note that because

$$E\left[\delta\mathbf{x}\right] = E\left[\mathbf{x} - \mathbf{x}_N\right] = \mathbf{0} \tag{70}$$

$$E\left[\delta\hat{\mathbf{x}}_f\right] = E\left[\hat{\mathbf{x}}_f - M\mathbf{x}_N\right] = \mathbf{0} \tag{71}$$

the mean of the augmented state vector is $E\left[\mathbf{X}\right] = \mathbf{0}$ and the covariance of the augmented system can be calculated as

$$C_A = E\left[\mathbf{X}\left(t\right)\mathbf{X}^T\left(t\right)\right] \tag{72}$$

Thus, the entire closed-loop system is represented by the linear time-varying differential equation in Eq. (67). A variety of linear system analysis can now be applied to this system. For covariance analysis, the following propagation and update equations will be utilized:

$$\dot{C}_A = \mathcal{A}C_A + C_A\mathcal{A}^T + \mathcal{B}S_\eta\mathcal{B}^T \tag{73}$$

$$C_A^+ = \left(I + \mathcal{K}\mathcal{H}\right)C_A\left(I + \mathcal{K}\mathcal{H}\right)^T + \mathcal{K}R_f\mathcal{K}^T \tag{74}$$

$$\mathcal{K} = \begin{bmatrix} 0 \\ K_f \end{bmatrix}, \qquad \mathcal{H} = \left[\, H \,, -H_f \,\right]$$

The initial conditions are defined as[4]:

$$C_{A_0} = \begin{bmatrix} P_{disp_0} & 0_{N_x \times N_f} \\ 0_{N_f \times N_x} & 0_{N_f \times N_f} \end{bmatrix}, \qquad P_{disp_0} = E\left[\delta\mathbf{x_0}\delta\mathbf{x_0}^T\right]$$

[4] The filter state is initiated with the state nominal values and zero variance.

4.2 Performance Evaluation

The overall closed-loop performance of the GN&C system is evaluated by examining the covariance matrix of the augmented state with the appropriate linear mappings. The true state dispersions $\delta\mathbf{x} = M_x\mathbf{X}$ are extracted from the augmented state covariance matrix as follows

$$D_{true} = E\left[\delta\mathbf{x}\left(t\right)\delta\mathbf{x}^T\left(t\right)\right] = M_x C_A M_x^T \tag{75}$$

The covariance of the true state dispersion accounts for navigation errors, guidance/control execution errors, and all external disturbances/noise sources. In a similar manner, the covariance matrix of the navigation state dispersion $\delta\hat{\mathbf{x}}_f = M_f\mathbf{X}$ can be extracted from the augmented covariance matrix.

$$D_{nav} = E\left[\delta\hat{\mathbf{x}}_f\left(t\right)\delta\hat{\mathbf{x}}_f^T\left(t\right)\right] = M_f C_A M_f^T \tag{76}$$

Since the true filter estimation error is the difference between the navigation state vector and the corresponding true state vector,

$$\mathbf{e}_{true} \equiv \hat{\mathbf{x}}_f - M\mathbf{x} = \hat{\delta\mathbf{x}}_f - M\delta\mathbf{x} = M_{true}\mathbf{X} \tag{77}$$

the covariance matrix of the true filter estimation error is given by

$$P_{true} = E\left[\mathbf{e}_{true}\mathbf{e}_{true}^T\right] = M_{true} C_A M_{true}^T \tag{78}$$

where $M_{true} = M_f - MM_x$.

Finally, the control effort can be determined by substituting Eqs. 49 and 31 into Eq. 44 to obtain, $\delta\mathbf{u} = M_u \cdot \delta\mathbf{X} + B_u \cdot \boldsymbol{\zeta}$. The covariance of the control effort is then given by:

$$P_u = E\left[\delta\mathbf{u}\delta\mathbf{u}^T\right] = M_u C_A {M_u}^T + B_u S_\zeta B_u^T \tag{79}$$

where $M_u = G_x \cdot M_x + G_f \cdot M_f$.

5 Results

5.1 Setup

The nominal value of the initial true state is given by $\mathbf{m_0}$ with the only non-zero values being $\theta_x\left(0\right) = \theta_0$ and $\dot{\theta}_x\left(0\right) = \Omega_N$. The initial filter state is $\mathbf{x}_f\left(0\right) = M\mathbf{m}_0$. The initial covariance of the true state, P_{disp_0} , is a 30 X 30 diagonal matrix, and the initial navigation filter covariance is $P_{f_0} = MP_{disp_0}M^T$. P_{disp_0} is provided in the appendix. The power-spectral-density of the true model process noise is

$$S_{\boldsymbol{\eta}} = \begin{bmatrix} q_\omega \cdot I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & q_b \cdot I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & q_{act} \cdot I_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & q_d \cdot I_{3\times3} \end{bmatrix}$$

and the power-spectral density of the filter model process noise is:

$$S_f = \begin{bmatrix} q_\omega \cdot I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & q_b \cdot I_{3\times 3} \end{bmatrix}$$

The variance of the measurement noise is

$$R = R_f = \sigma_{st}^2 \cdot I_{4\times 4}$$

Star-tracker measurements are produced every $10\,[sec]$. All initial non-zero nominal values, initial uncertainties, process noise power-spectral densities and other system parameters are given in Table 1.

Table 1. Setup Values

Symbol	Value	Units	Description
θ_0	-60	deg	Initial nominal x-angle
Ω_N	0.2	deg/sec	Nominal x-angular velocity
σ_θ	$5 \cdot 10^{-2}$	deg	Initial attitude $1-\sigma$
σ_ω	10^{-3}	deg/sec	Initial angular velocity $1-\sigma$
$\sigma_{b_{act}}$	10^{-4}	$N \cdot m$	Initial actuator bias $1-\sigma$
$\sigma_{\epsilon_{act}}$	1	$arcsec$	Initial actuator misalignment $1-\sigma$
σ_{b_ω}	1	$arcsec/sec$	Initial gyro bias $1-\sigma$
σ_{ϵ_ω}	8	$arcsec$	Initial gyro (orthogonal) misalignment $1-\sigma$
σ_{α_ω}	8	$arcsec$	Initial gyro (non-orthogonal) misalignment $1-\sigma$
σ_{f_ω}	10^{-3}		Initial gyro scale factor $1-\sigma$
σ_ϕ	24	$arcsec$	Initial star-tracker misalignment $1-\sigma$
q_ω	$\left(4.25 \cdot 10^{-2}\right)^2$	$arcsec^2/sec$	Gyro drift noise strength
q_b	$\left(4.44 \cdot 10^{-5}\right)^2$	$arcsec^2/sec^3$	Gyro bias noise strength
q_{act}	10^{-4}	$N^2 \cdot m^2 \cdot sec$	Actuator noise strength
q_d	10^{-4}	$N^2 \cdot m^2 \cdot sec$	Disturbance noise strength
σ_{st}	$\frac{14}{3600}\frac{\pi}{180}$	rad	Measurement noise $1-\sigma$
ω_c	$\frac{2\pi}{100}$	$\frac{rad}{sec}$	Controller frequency
ξ	$\sqrt{2}/2$		Controller damping ratio

5.2 Linear Covariance Results

Attitude Navigation Errors. The linear covariance attitude navigation errors are examined in Fig.1. The θ_x steady state filter $1-\sigma$ error is $2.9\,[arcsec]$ while the true navigation $1-\sigma$ error is $17.2\,[arcsec]$. The θ_y and θ_z steady-state filter $1-\sigma$ error is $3.9\,[arcsec]$ while the true navigation $1-\sigma$ error is $24.3\,[arcsec]$. Since both star-trackers measure θ_x and provide only one measurement of θ_y and θ_z each, there are effectively twice as many measurements of θ_x. Thus the steady-state errors for θ_x are reduced by a factor of $\sqrt{2}$. In all cases, it is clear that the suboptimal, reduced order filter is underestimating the true navigation error.

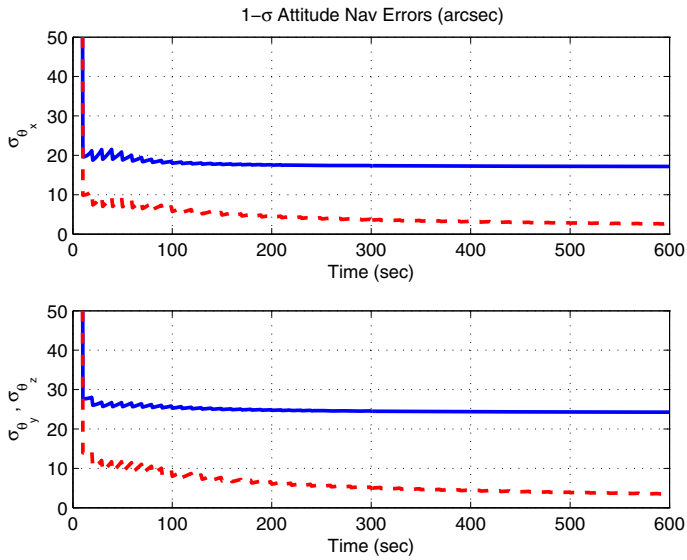
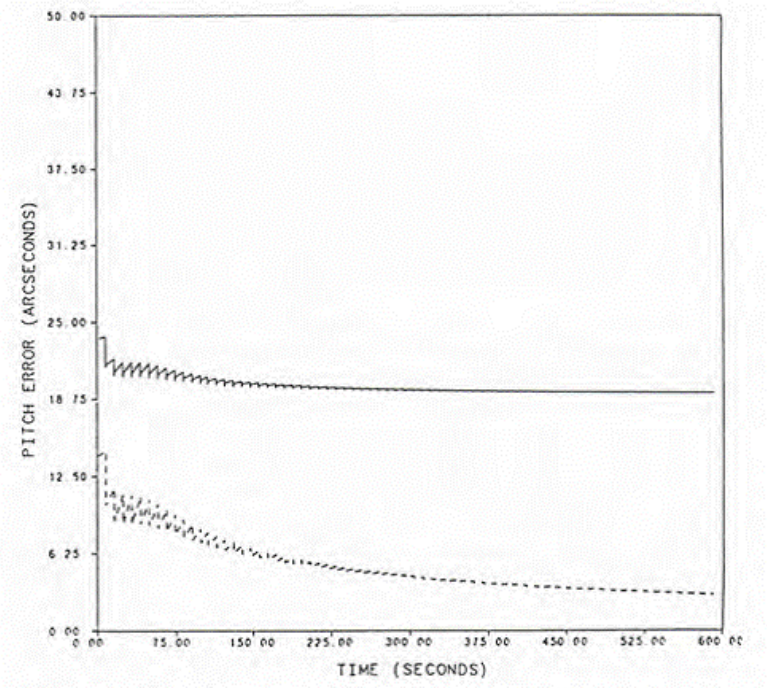


Fig. 1. Attitude navigation error based on linear covariance analysis. The dotted red curve shows the filter $1-\sigma$ error, and the solid blue curve shows the true $1-\sigma$ error.

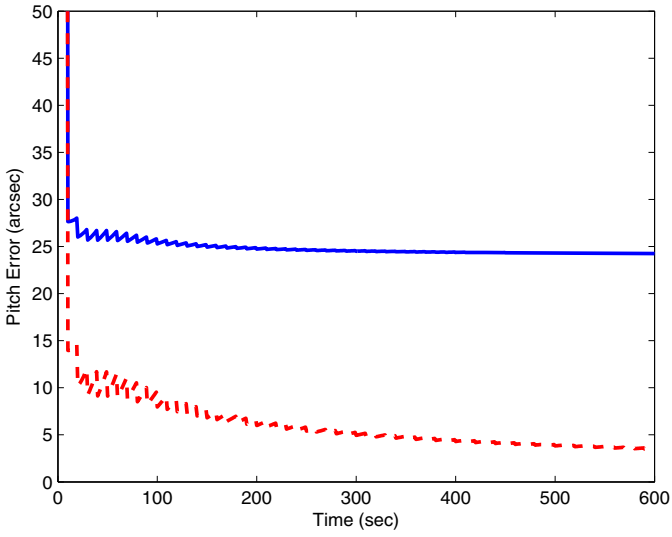
A comparison of the above pitch angle navigation error variance σ_{θ_y} with Bar-Itzhack and Harman's true covariance results is shown in Fig. 2. Although the Bar-Itzhack and Harman's work included a fine sun sensor that was not included in this study, the results are remarkably similar.

Attitude Dispersions. Although the above results for the attitude navigation errors are important, the ultimate goal is to limit the attitude control system pointing errors, i.e., the attitude dispersions from their desired nominal values. The true and filter state $1-\sigma$ attitude dispersions are shown in Fig. 3. These results should be compared to the 180 arcsec pointing accuracy requirement for the all-sky survey. Since the attitude navigation errors are relatively small compared to the pointing errors, it is concluded that the pointing errors are primarily due to actuator errors and random disturbances.

Gyro Bias Errors. The variance of the gyro bias errors are examined in Fig.4. The $1-\sigma$ filter x-axis steady-state gyro bias estimation error is 0.01 $[arcsec/sec]$, significantly less than the true $1-\sigma$ bias estimation error of 0.72 $[arcsec/sec]$. This is a consequence of the reduced filter state. Since the x-axis gyro scale-factor is not included in the filter design model, and since the spacecraft is rotating primarily about the x-axis, the error in the gyro measurement produced by the true scale-factor, $f_x\Omega_N$, incorrectly appears in the filter as an x-axis gyro bias.



(a) Bar Itzhack's true covariance analysis results for the filter and true $1-\sigma$ pitch error



(b) The authors linear covariance analysis results for the filter and true $1-\sigma$ error

Fig. 2. Pitch error comparison - true error (solid), filter error (dashed)

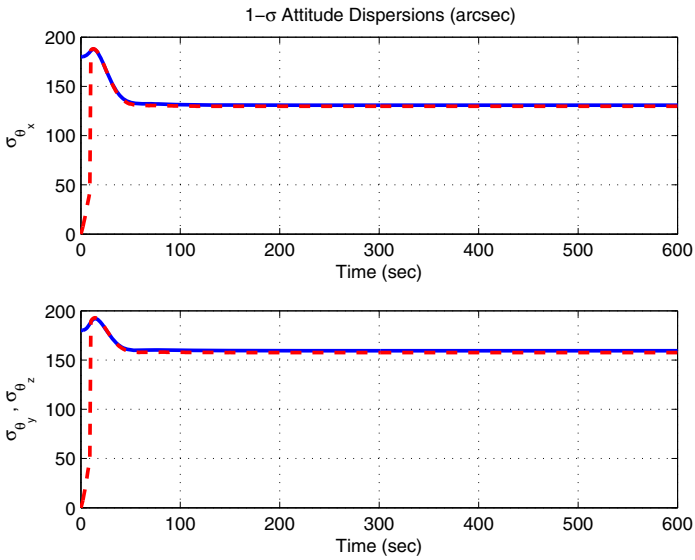


Fig. 3. True attitude dispersions. The dotted red curve shows the filter $1-\sigma$ dispersion, and the solid blue curve shows the true $1-\sigma$ dispersion.

This dramatic difference is not seen in the true and filter steady-state y-axis and z-axis gyro errors.

The above pitch gyro drift estimation errors $\left(b_{\omega_y}\right)$ are also in good agreement with Bar-Itzhack and Harman's true covariance results.

Control Effort and Magnetic Dipole Demand. The variance of the $1-\sigma$ control effort is shown in Fig.5. After an initial transient, the steady-state variance in each of the 3 axes is approximatety $0.0038\,[N \cdot m]$. In this example, the steady-state control effort is primarily in repsonse to actuator errors and random disturbances.

5.3 Nonlinear Monte Carlo Analysis

In Monte Carlo analysis, the covariance of the true dispersion and error states are computed by simply generating N samples of each state as a function of time, and then estimating the covariances as

$$D_{true}(t) \approx \frac{1}{N-1}\sum_{i=1}^{N}\delta\mathbf{x}(t)\delta\mathbf{x}^T(t) \tag{80}$$

$$D_{nav}(t) \approx \frac{1}{N-1}\sum_{i=1}^{N}\delta\hat{\mathbf{x}}(t)\delta\hat{\mathbf{x}}^T(t) \tag{81}$$

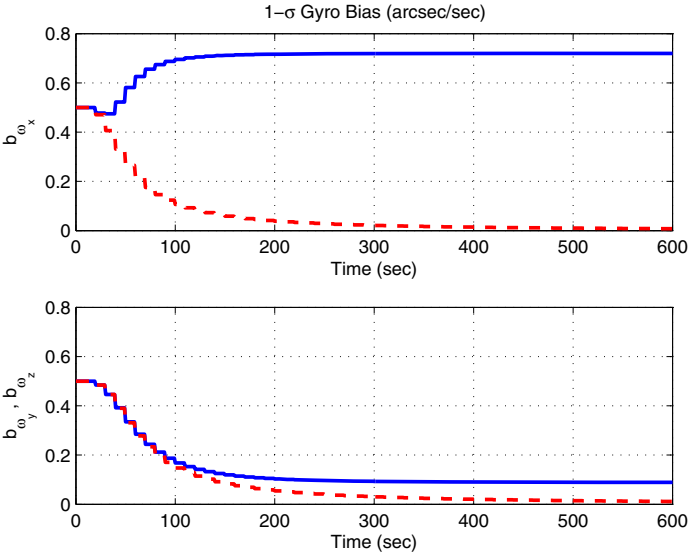


Fig. 4. Gyro drift estimation error based on linear covariance analysis. The dotted red curve shows the filter $1-\sigma$ error, and the solid blue curve shows the true $1-\sigma$ error.

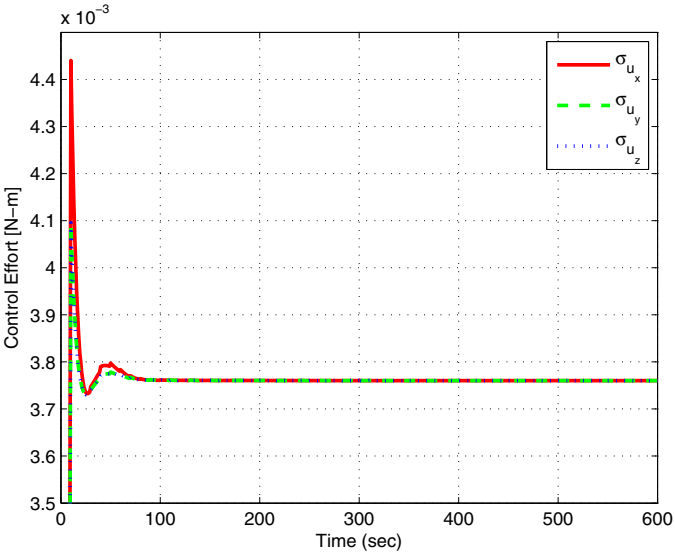


Fig. 5. 3-axis $1-\sigma$ control effort

$$P_{true}(t) \approx \frac{1}{N-1}\sum_{i=1}^{N}\delta\mathbf{e}(t)\delta\mathbf{e}^T(t) \tag{82}$$

For this application, the Monte Carlo simulation consisted of the nonlinear dynamics given in Eq. 1 , the nonlinear gyro and star-tracker measurements given by Eq.12 and 25, an extended Kalman filter with the same reduced filter state and model replacement, and a PD control law based on the estimated attitude quaternion and the estimated angular rate with gyro bias compensation.

The initial conditions for the simulation were generated by sampling the Gaussian probability density function given by

$$\mathbf{x}_0 \sim \mathcal{N}(\mathbf{m}_0, P_{disp_0})$$

$N = 1000$ Monte Carlo runs were performed and system performance was evaluated using Eq. 80, 81 and 82. The Monte Carlo results were then used to verify the linear covariance analysis results.

The x-axis true navigation errors of 10 Monte Carlo runs with the $3-\sigma$ envelope (of the 1000 runs) are illustrated in Fig.6, as well as the comparison with the linear covariance results. The x-axis true estimation error as determined by the linear covariance analysis matches the Monte Carlo results with a maximum deviation $\varepsilon_{max} = 5.23\,[arcsec]$, where ε_{max} is the maximum difference between the Monte Carlo and linear covariance $1-\sigma$ data.

The x-axis true dispersions of 10 Monte Carlo runs with a $3-\sigma$ envelope are examined in Fig.7. The x-axis true dispersions from the linear covariance analysis match the Monte Carlo results with a maximum deviation $\varepsilon_{max} = 7.88\,[arcsec]$

The x-axis control effort dispsersion for 10 Monte Carlo runs and the $3-\sigma$ linear covariance data are illustrated in Fig.7. The $1-\sigma$ x-axis control effort determined by the linear covariance analysis matches the $1-\sigma$ Monte Carlo results with a maximum deviation $\varepsilon_{max} = 1.1 \cdot 10^{-3}\,[N \cdot m]$

5.4 Sensitivity Analysis

Linear covariance analysis also provides the capabilty to quickly examine the sensitivity of the system performance to initial errors, measurement noise, and process noise. This analysis enables a system designer to focus on the most significant error sources. To do this, the covariance matrix of the augmented state is presented as a sum of these three contributions

$$C_A = C_0 + C_Q + C_R \tag{83}$$

The sensitivity dynamics can then be obtained from Eq. 73

$$\dot{C}_0 = \mathcal{A}C_0 + C_0\mathcal{A}^T \tag{84}$$

$$\dot{C}_R = \mathcal{A}C_R + C_R\mathcal{A}^T \tag{85}$$

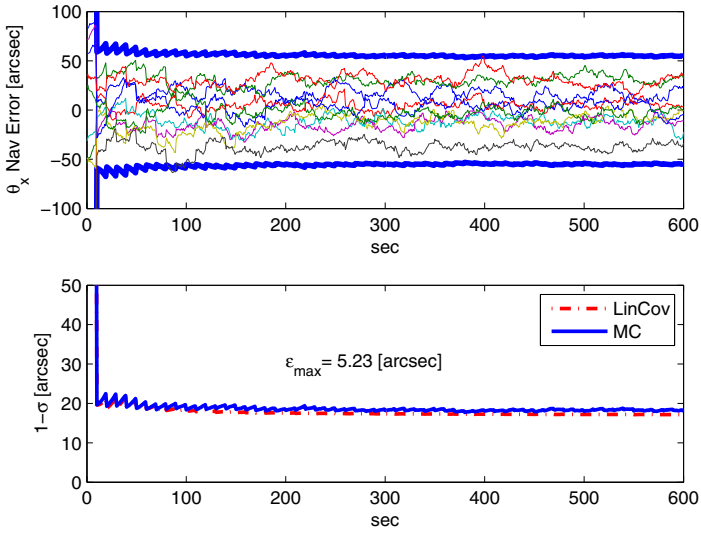


Fig. 6. True attitude estimation errors. Monte Carlo results with $3-\sigma$ statistics (blue) are shown in the top figure. A comparison of the $1-\sigma$ Monte Carlo results to the $1-\sigma$ linear covariance analysis results is shown in the bottom figure.

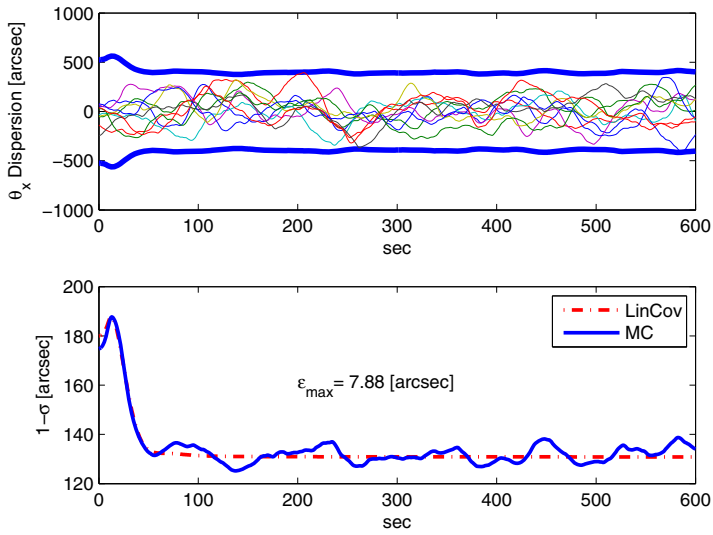


Fig. 7. True attitude dispersions. Monte Carlo results with $3-\sigma$ statistics (blue) are shown in the top figure. A comparison of the $1-\sigma$ Monte Carlo results to the $1-\sigma$ linear covariance analysis results is shown in the bottom figure.

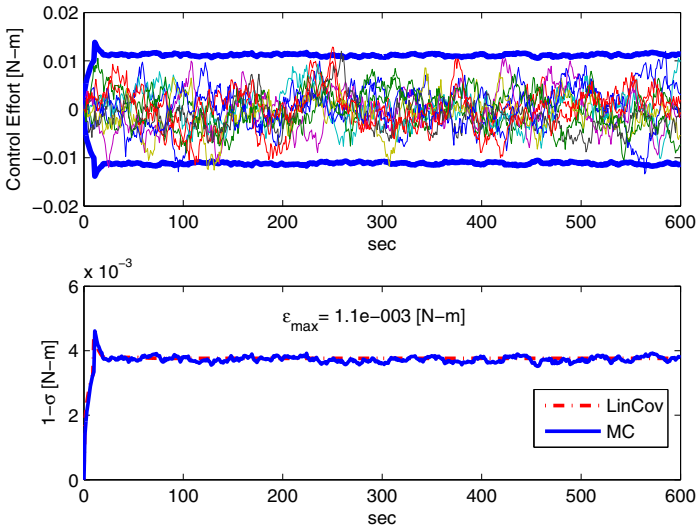


Fig. 8. Control effort dispersion. Monte Carlo results with $3-\sigma$ statistics (blue) are shown in the top figure. A comparison of the $1-\sigma$ Monte Carlo results to the $1-\sigma$ linear covariance analysis results is shown in the bottom figure.

$$\dot{C}_Q = \mathcal{A}C_Q + C_Q\mathcal{A}^T + \mathcal{B}S_\eta\mathcal{B}^T \tag{86}$$

and the updated sensitivity equations are obtained from Eq. 74

$$C_0^+ = (I + \mathcal{K}\mathcal{H})\, C_0\, (I + \mathcal{K}\mathcal{H})^T \tag{87}$$

$$C_R^+ = (I + \mathcal{K}\mathcal{H})\, C_R\, (I + \mathcal{K}\mathcal{H})^T + \mathcal{K}R_f\mathcal{K}^T \tag{88}$$

$$C_Q^+ = (I + \mathcal{K}\mathcal{H})\, C_Q\, (I + \mathcal{K}\mathcal{H})^T \tag{89}$$

Fig. 9 shows how each source of error contributes to the true x-axis attitude estimation error. In this case, the initial errors are the main contribution to the x-axis attitude estimation error. This is a consequence of the initial star-tracker misalignment ($24\,[arcsec]$) being unobservable. Since there are two tracker measurements of θ_x, the steady-state error due to tracker misalignment is $24\,[arcsec]\,/\sqrt{2}$, or approximately $17\,[arcsec]$.

Fig. 10 shows how each source of error contributes to the x-axis pointing dispersion. In this case, the initial errors and the process noise (actuator errors and disurbances) are the main contributions to the x-axis pointing dispersion.

Similar sensitivity results can be obtained for any performance indicator that can be extracted from the augmented state.

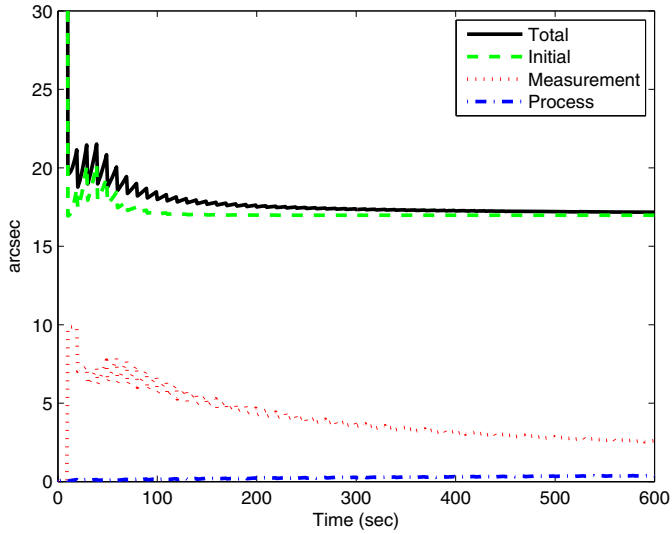


Fig. 9. True x-axis $1-\sigma$ attitude estimation error sensitivity analysis

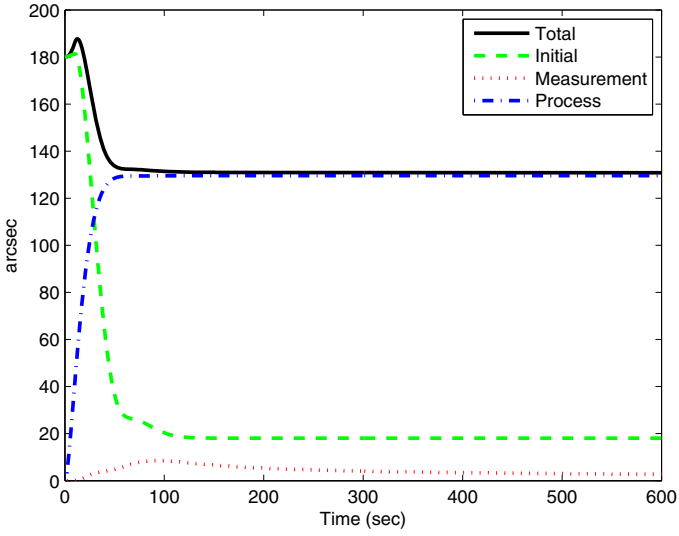


Fig. 10. x-axis $1-\sigma$ pointing dispersion sensitivity analysis

6 Conclusions

It has been shown using nonlinear Monte Carlo analysis that Bar Itzhack and Harman's application of true covariance analysis [2] for estimating the true navigation errors of a reduced order filter with model replacement produces accurate results and can thus be used for preliminary attitude filter design. An extension of Bar Itzhack and Harman's true covariance analysis approach using an augmented state to include true state dispersions, i.e., attitude control and pointing errors, was also implemented. The variances of the true attitude and angular velocity disperison errors based on the extended linear covariance analysis was validated using nonlinear Monte Carlo analysis showing that linear covariance analysis can be a useful tool in preliminary attitude determination and control system design. It was also demonstrated that the extended covariance analysis can provide useful system design sensitivity analysis.

Acknowledgments. This research was conducted with support and funding from the Lady Davis Fellowship Trust and the Asher Space Research Fund.

References

1. Ambrosius, B.A.C., Hesper, E.T., Wakker, K.F.: Application of global positioning system for hermes rendezvous navigation. Journal of Guidance, Control, and Dynamics 16(1), 197–205 (1993)
2. Bar-Itzhack, I.Y., Harmann, R.R.: Covariance simulation of the euve update filter. In: Flight Mechanics/Estimation Theory Symposium, pp. 225–237. NASA Goddard Space Flight Center, Greenbelt (1989)
3. Battin, R.H.: An Introduction to the Mathematics and Methods of Astrodynamics. AIAA Education Series. AIAA, New York (1987)
4. Carpenter, J.R., Markley, F.L.: Generalized linear covariance analysis. In: AAS/AIAA F. Landis Markley Astrodynamics Symposium AAS 08-273. NASA Goddard Space Flight Center, Cambridge (2008)
5. Crassidis, J.L., Junkins, J.L.: Optimal Estimation of Dynamic Systems. Chapman and Hall CRC, New York (2004)
6. Edwards, R.M.: Gravity model performance in inertial navigation. Journal of Guidance, Control, and Dynamics 5(1), 73–78 (1986)
7. Euler, E.A.: Orbit determination covariance analysis for the cassini solstice mission. In: AIAA-2010-8264, AIAA/AAS Astrodynamics Specialist Conference, Toronto, Ontario, August 2-5. NASA Goddard Space Flight Center (2010)
8. Gelb, A.: M.I.T. Press, Cambridge (1974)
9. Geller, D.K.: Linear covariance techniques for orbital rendezvous analysis and autonomous onboard mission planning. Journal of Guidance, Control, and Dynamics 29(6) (November 2006)
10. Ha, J.C.V.D.: Attitude determination covariance analysis for geostationary transfer orbits. Journal of Guidance, Control, and Dynamics 9(2), 156–163 (1986)
11. Haessig, D.: Separate-bias estimation with reduced-order Kalman filters. Automatic Control. IEEE (1998)

12. Ionasescu, R.: Orbit determination covariance analysis for the cassini solstice mission. In: AIAA-2010-8264, AIAA/AAS Astrodynamics Specialist Conference, Toronto, Ontario, August 2-5. NASA Goddard Space Flight Center (2010)
13. Jekili, C.: Precision free-inertial navigation with gravity compensation by an onboard gradiometer. Journal of Guidance, Control, and Dynamics 29(3), 73–78 (2006)
14. Kreigsman, B.A., Mahar, K.B.: Gravity model errors in mobile inertial-navigation systems. Journal of Guidance, Control, and Dynamics 9(3), 312–318 (1986)
15. Markley, F.L., Seidewitz, E., Nicholson, M.: A general model for the attitude determination error analysis. In: NASA Conference Publication 3011: Flight Mechanics/Estimation Theory Symposium. NASA Goddard Space Flight Center, Greenbelt (1988)
16. Maybeck, P.S.: Stochastic Models, Estimation, and Control, vol. 1. Navtech Book and Software Store, New York (1994)
17. Moesser, T., Geller, D., Robinson, S.: Guidance and navigation linear covariance analysis for lunar powered descent. AAS Paper 11-532 (August 2011)
18. Park, R., Chodas, P., Roundhill, I.: Covariance analysis of cassini titan flyby using sar and altimetry data. In: AIAA-2006-6398, AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Keystone, Colorado, August 21-24. NASA Goddard Space Flight Center (2006)
19. Pittelkau, M.E.: Rotation vector in attitude estimation. Journal of Guidance, Control, and Dynamics 26(6), 855–860 (2003)
20. Rose, T., Geller, D.: Linear covariance techniques for powered ascent. AIAA-2010-8175 (August 2010)
21. Shuster, M.: A survey of attitude representations. Navigation (1993)
22. Simon, D.: Reduced Order Kalman Filtering Without Model Reduction. Control and Intelligent Systems 35(2), 1–16 (2007)
23. Stastny, N.B., Bettinger, R.A., Chavez, F.R.: Lincov analysis of an automated celectial inertial navigation approach for geo satellites. In: AIAA 2008-6756, AIAA/AAS Astrodynamics Specialists Conference, Honolulu Hawaii, pp. 184–196. AIAA (2008)
24. Tabley, B.D., Schutz, B.E., Born, G.E.: Statistical Orbit Determination, pp. 387–435. Elsevier Academic Press, London (2004)
25. Woffinden, D.C.: Angles-Only Navigation for Autonomous Orbital Redezvous. Dissertation, Utah Sate University (December 2008), `http://digitalcommons.usu.edu/etd/12/`
26. Wong, E.C.: Attitude Control System for the Extreme Ultrqviolet Explorer Satellite. The Journal of the Astronautical Sciences 33(4), 401–415 (1985)
27. Woodbury, D., Junkins, J.: On the consider kalman filter. In: AIAA-2010-7752, AIAA Guidance, Navigation, and Control Conference, Toronto, Ontario, August 2-5. AIAA (2010)

Appendix - Matrices

$$A = \begin{bmatrix} -[\boldsymbol{\omega}_N\times] & I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times18} \\ 0_{3\times3} & A_{\boldsymbol{\omega}} & J_0^{-1} & 0_{3\times3} & 0_{3\times18} \\ 0_{24\times3} & 0_{24\times3} & 0_{24\times3} & 0_{24\times3} & 0_{24\times18} \end{bmatrix}$$

$$A_{\boldsymbol{\omega}} = J_0^{-1}\left[J_0\boldsymbol{\omega}_N\times\right] - J_0^{-1}\left[\boldsymbol{\omega}_N\times\right]J_0$$

$$G = \begin{bmatrix} 0_{3\times3} \\ J_0^{-1} \\ 0_{24\times3} \end{bmatrix}, \quad B = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{3\times3} & 0_{3\times3} & J_0^{-1} & J_0^{-1} \\ 0_{6\times3} & 0_{6\times3} & 0_{6\times3} & 0_{6\times3} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times3} \\ 0_{15\times3} & 0_{15\times3} & 0_{15\times3} & 0_{15\times3} \end{bmatrix}$$

$$H = \begin{bmatrix} \left[\mathbf{i}_{x_{ST_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{y_{ST_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{x_{ST_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{z_{ST_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}} \\ 0_{21\times1} & 0_{21\times1} & 0_{21\times1} & 0_{21\times1} \\ \left[\mathbf{i}_{x_{ST_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{y_{ST_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}} & 0_{3\times1} & 0_{3\times1} \\ 0_{3\times1} & 0_{3\times1} & \left[\mathbf{i}_{x_{ST_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{z_{ST_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}} \end{bmatrix}^T$$

$$A_f = \begin{bmatrix} -[\boldsymbol{\omega}_N\times] & -I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}, G_\omega = \begin{bmatrix} I_{3\times3} \\ 0_{3\times3} \end{bmatrix}, B_f = \begin{bmatrix} -I_{3\times3} & O_{3\times3} \\ O_{3\times3} & I_{3\times3} \end{bmatrix}$$

$$H_f = \begin{bmatrix} \left[\mathbf{i}_{x_{ST_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{y_{ST_1}}\times\right]\mathbf{s}_{\mathbf{z}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{x_{ST_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}} & \left[\mathbf{i}_{z_{ST_2}}\times\right]\mathbf{s}_{\mathbf{y}}^{\bar{\mathbf{B}}} \\ 0_{3\times1} & 0_{3\times1} & 0_{3\times1} & 0_{3\times1} \end{bmatrix}^T$$

$$G_x = \left[0_{3\times3}\ -K_\omega\ 0_{3\times6}\ -K_\omega\ -K_\omega\left[\boldsymbol{\omega}_N\times\right]\ -K_\omega W\left(\boldsymbol{\omega}_N\right)\ -K_\omega D\left(\boldsymbol{\omega}_N\right)\ 0_{3\times6}\right]$$

$$G_f = \left[-K_\theta\ K_\omega\right]\ \ B_u = \left[-K_\omega\ 0_{3\times9}\right]\ \ B_{f_X} = \begin{bmatrix} I_{3\times3} & 0_{3\times9} \\ 0_{3\times3} & 0_{3\times9} \end{bmatrix}$$

$$A_{f_X} = \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times6} & I_{3\times3} & \left[\boldsymbol{\omega}_N\times\right] & W\left(\boldsymbol{\omega}_N\right) & D\left(\boldsymbol{\omega}_N\right) & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times6} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \end{bmatrix}$$

$$M_x = \left[I_{N_x\times N_x}\ 0_{N_x\times N_f}\right], \quad M_f = \left[O_{N_x\times N_f}\ I_{N_f\times N_f}\right]$$

$$M = \begin{bmatrix} I_{3\times3} & 0_{3\times9} & 0_{3\times3} & 0_{3\times15} \\ 0_{3\times3} & 0_{3\times9} & I_{3\times3} & 0_{3\times15} \end{bmatrix}$$

$$P_{disp_0} = \begin{bmatrix} \sigma_\theta^2 I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & \sigma_\omega^2 I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & \sigma_{b_{act}}^2 I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{\epsilon_{act}}^2 I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{b_\omega}^2 I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{\epsilon_\omega}^2 I_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{\alpha_\omega}^2 I_{3\times3} & 0_{3\times3} & 0_{3\times6} \\ 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & \sigma_{f_\omega}^2 I_{3\times3} & 0_{3\times6} \\ 0_{6\times3} & 0_{6\times3} & 0_{6\times3} & 0_{6\times3} & 0_{6\times3} & 0_{6\times3} & 0_{6\times3} & 0_{6\times3} & \sigma_\phi^2 I_{6\times6} \end{bmatrix}$$

Reaction Wheel Parameter Identification and Control through Receding Horizon-Based Null Motion Excitation

Avishai Weiss[1], Frederick Leve[2], Ilya V. Kolmanovsky[1], and Moriba Jah[2]

[1] University of Michigan, Ann Arbor, MI, USA
{avishai,ilya}@umich.edu
[2] Space Vehicles Directorate, Kirtland Air Force Base, NM, USA
AFRL.RVSV@kirtland.af.mil

Abstract. Additional actuator motion, constrained to the null-space of the Reaction Wheel Array (RWA) of an over-actuated spacecraft, can be exploited for learning system parameters without inducing large perturbations to the controlled body (e.g., spacecraft bus). In this paper a receding horizon optimization approach is developed to generate such a null-motion excitation (NME) that facilitates the identification of the actuator misalignments with perturbations that are local to the nominal trajectory and decreasing with the decrease in size of the parameter estimation error. The receding horizon approach minimizes an objective function that penalizes the parameter error covariance and the null-motion excitation. The potential of the receding horizon approach to outperform the baseline null motion excitation algorithm proposed in an earlier publication is demonstrated through simulations.

Keywords: Receding Horizon Control, Null-Motion, Null-Space, Parameter Identification, Reaction Wheel Assembly.

1 Introduction

The on-orbit estimation of spacecraft parameters, such as Reaction Wheel Array (RWA) alignments, can reduce assembly, integration, and test (AI&T) time and efforts necessary with detailed ground-based system identification of spacecraft. Due to a possible loss of communications, or other operational constraints, it may not be possible to apply an arbitrary tumble to a satellite for system identification. In these situations, the conditions required by many existing adaptive control and estimation techniques (see [1], [2], [3], [4], [5], [6], [7], [8], [9], [10] and references therein) to achieve both asymptotic tracking and asymptotic parameter identification may not be satisfied.

An approach to enhance on-board parameter identification via *null motion excitation (NME)* has been first proposed in [11]. In [11], an overactuated spacecraft with an RWA is considered, and it is shown that the spacecraft actuators can be coordinated in such a way that the convergence of estimates of parameters characterizing RWA alignments is enhanced, while the disturbance to the

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_25

nominal spacecraft attitude maneuver is minimized. In other words, it is demonstrated that information about parameters can be gained by adding NME to a nominal forced trajectory relative to the case of the forced trajectory by itself (i.e., passive system identification case). A local gradient approach was used in [11] to optimize the NME.

In this paper, a receding horizon optimization is exploited to generate the NME. At each time instant, the NME sequence is optimized over a finite prediction horizon to minimize a cost functional that penalizes the predicted parameter error covariance and the NME excitation. The first element of the optimized sequence is applied to the spacecraft. The optimization is repeated at the next time instant using the updated error covariance matrix as an initial condition. The proposed approach can be viewed as an on-board Design of Experiments (DoE) procedure, used to enhance persistence of excitation conditions without causing large disturbances. It is related to our earlier work on receding horizon optimization for simultaneous tracking and parameter identification in automotive systems [12,13].

The receding horizon approach of this paper is compared to the local gradient approach of [11], and it is shown that the potential for faster convergence exists at the price of higher computational cost. The differences between the receding horizon approach proposed in this paper and the local gradient method of [11] are in the minimization of a cost function that penalizes the total covariance (i.e., parameter and measurement) matrix over the prediction horizon of N_c steps ahead while the approach in [11] corresponds to minimizing the parameter error estimated only one step ahead and by assuming perfect measurements.

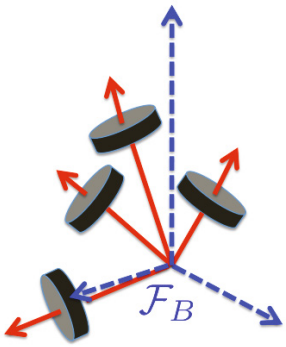


Fig. 1. Four skewed RWA arrangement

2 Spacecraft Dynamics Model

We consider a rigid spacecraft bus actuated by a Reaction Wheel Array (RWA) consisting of four axially symmetric flywheels with negligible friction. See Figure 1. The total angular momentum of the spacecraft relative to its center of

mass with respect to an inertial frame $\mathrm{F_E}$ and resolved in a bus-fixed principal frame $\mathrm{F}_{\mathcal{B}}$ is given by

$$\vec{H}_{\mathrm{sc/c/E}}\Big|_{\mathcal{B}} = J\omega + h, \tag{1}$$

where J is the spacecraft inertia matrix resolved in $\mathrm{F}_{\mathcal{B}}$, ω is the angular velocity of $\mathrm{F}_{\mathcal{B}}$ with respect to $\mathrm{F_E}$, resolved in $\mathrm{F}_{\mathcal{B}}$, and h is the angular momentum of the RWA resolved in $\mathrm{F}_{\mathcal{B}}$. We assume zero external torque, and thus, the total angular momentum is conserved.

The inertial time derivative of (1) yields

$$\vec{H}^{\mathrm{E}\bullet}_{\mathrm{sc/c/E}}\Big|_{\mathcal{B}} = J\dot{\omega} + \omega^{\times} J\omega + \dot{h} + \omega^{\times} h = 0, \tag{2}$$

where $\omega^{\times}$ is the skew-symmetric matrix representing the cross-product, and

$$\dot{h} = \frac{\partial h}{\partial \nu}\dot{\nu} = J_{\alpha}(\theta)\dot{\nu}, \tag{3}$$

where $J_{\alpha}(\theta)$ is the Jacobian matrix, which is a function of actuator alignments parameterized by a vector θ, and $\nu = [\nu_1\ \nu_2\ \nu_3\ \nu_4]^{\mathrm{T}} \in \mathbf{R}^4$ is a column vector of four flywheel rates.

Following [11], we re-parameterize h is terms of components of RWA alignment unit vectors as

$$h = Y_1(\nu)\theta, \tag{4}$$

where

$$Y_1 = I_w \left[\, \nu_1 I_3\ \nu_2 I_3\ \nu_3 I_3\ \nu_4 I_4 \,\right],$$

I_3 denotes the 3×3 identity matrix, and where $\theta \in \mathbf{R}^{12}$ is the parameter vector to be identified.

The NME approach is based on augmenting an excitation signal $n(t) \in \mathbf{R}^4$ to the nominal RWA control signal, $D(t) \in \mathbf{R}^4$, so that

$$\dot{\nu} = D(t) - \Gamma(\hat{\theta}(t))n(t), \tag{5}$$

where I_4 denotes the 4×4 identity matrix, $\Gamma(\hat{\theta}(t)) = \left(I_4 - J_{\alpha}^{\sharp}(\hat{\theta})J_{\alpha}(\hat{\theta})\right)$, and $J_{\alpha}^{\sharp}$ is the pseudo-inverse of J_{α}, $J_{\alpha}(\hat{\theta})J_{\alpha}^{\sharp}(\hat{\theta}) = I_3$. Note that the implementation of (5) is based on estimated alignments, $\hat{\theta}$, with the motivation that if $\hat{\theta} = \theta$, then $\dot{\nu} = D(t)$ and the effects of NME signal are zeroed out. Thus the overactuation capability of a 4 flywheel RWA system can be used to enhance the parameter identifiability. The computation of the excitation signal $n(t)$ is discussed in the next section.

3 Receding Horizon Optimization of the Null Motion Excitation

A discrete-time receding horizon approach is used for the optimization of NME signal $n(t)$ in (5). We use the notation $a(t+k|t)$ to denote the predicted value of

a variable a at the discrete time instant $t+k$ when the prediction is made at the discrete time instant t. Using this notation, and based on (5), the discrete-time update equations for the flywheel rates have the form

$$\nu(t+k+1|t) = \nu(t+k|t) + D(t)\Delta T - \Gamma(\hat{\theta}(t))n(t+k|t)\Delta T, \tag{6}$$

where ΔT is the sampling period, $\hat{\theta}(t) \in \mathbf{R}^{12}$ is the vector of the estimated reaction wheel alignment parameters, and $n \in \mathbf{R}^4$ is the null motion excitation signal that we determine through the receding horizon optimization.

With the motivation of simplifying the optimization problem, and with the justification that the nominal control law and adaptation are sufficiently slow, we do not predict D and the parameter estimate changes over the horizon, thereby assuming $D(t+k|t) = D(t)$ and that $\hat{\theta}(t+k|t) = \hat{\theta}(t)$. Thus, in our approach, the term

$$\Gamma(\hat{\theta}(t)) = \left(I - J_\alpha^\sharp(\hat{\theta})J_\alpha(\hat{\theta})\right),$$

remains constant over the prediction horizon.

The optimization of the NME sequence is performed over a receding horizon of length N_c so that $n(t+k|t),\ k = 0, 1, \cdots, N_c$ minimizes a cost functional of the form

$$J = \sum_{k=0}^{N_c} \left[trace(P(t+k|t)) + \rho \cdot n^{\mathrm{T}}(t+k|t)n(t+k|t)\right]. \tag{7}$$

In (7), P denotes the parameter error covariance matrix and ρ is a weight penalizing the size of NME. Once the sequence is computed, the first element of it, $n(t) = n(t|t)$ is applied as an excitation and the process is repeated at the next time instant, $t+1$.

By combining (1) and (4), one obtains a linear regression model for identifying the parameter vector θ,

$$\begin{aligned} y &= Y_1(\nu)\theta + \epsilon, \\ y &= \vec{H}_{\mathrm{sc/c/E}}\Big|_{\mathcal{B}} - J\omega, \end{aligned} \tag{8}$$

where $\vec{H}_{\mathrm{sc/c/E}}\Big|_{\mathcal{B}} - J\omega$ represents the measurement, with the added measurement noise, ϵ. The assumption of y being a measured signal is reasonable given that ω is measured, J is known, the spacecraft orientation is measured and the total angular momentum vector $\vec{H}_{\mathrm{sc/E}}$ is conserved and is known at the initial time[1]. It should be noted that because both noise processes are assumed to be Gaussian zero mean, the addition of Gaussian variables associated with the RWA encoder and gyro noises is also a Gaussian random variable. It is assumed that all flywheel and gyro *biases* are removed separately from the attitude determination system. It should also be noted that the addition of two measurements

[1] In case $\vec{H}_{\mathrm{sc/c/E}}$ is unknown at the initial time, it can be estimated along with θ using the approach developed in this paper.

does not make use of the measurement covariance optimally in the Kalman filter (i.e., the difference in accuracy of the two different sets of measurements is not exploited). Differencing of the measurements rather than considering separately was done to reduce the computation of the measurement covariance from a 7×7 matrix to a 3×3.

To compute $P(t+k|t)$ in (7), we use Recursive Least Squares (RLS)-based prediction of the parameter error covariance matrix, based on the equations

$$\begin{gathered} K(t+k|t) = \\ P(t+k-1|t)Y_1(t+k|t)^{\mathrm{T}} \times \left(Y_1(t+k|t)P(t+k-1|t)Y_1(t+k|t)^{\mathrm{T}} + R\right)^{-1}, \\ \\ P(t+k|t) = \\ \left(I_{12} - K(t+k|t)Y_1(t+k|t)\right) P(t+k-1|t) \times \left(I_{12} - K(t+k|t)Y_1(t+k|t)\right)^{\mathrm{T}} \\ +K(t+k|t)RK(t+k|t)^{\mathrm{T}}, \end{gathered} \tag{9}$$

where I_{12} denotes the 12×12 identity matrix and $R = E[\epsilon(k)\epsilon(k)^{\mathrm{T}}]$ is the measurement noise covariance matrix. Note that Joseph's form of the a posteriori error covariance matrix update is used in (9) due to its better numerical conditioning properties.

In [11] a gradient type algorithm is used to update the parameter estimates. Here, for consistency with the RLS approach, updates of the form

$$\hat{\theta}(t+1) = \hat{\theta}(t) + K(t+1|t)(y(t+1) - Y_1(\nu(t+1))\hat{\theta}(t)), \tag{10}$$

are employed to extract parameter estimates.

4 Simulation Setup

Simulations are now presented to demonstrate the improved performance of the new receding horizon solution for the NME. The four flywheel RWA to be simulated has a non-orthogonal skew arrangement shown in Figure 1.

The spacecraft and simulated maneuver parameters are

$$\begin{array}{cc} J = diag(10, 20, 60) & [kgm^2], \\ I_w = 0.001 & [kgm^2], \\ \nu(0) = [0\ 0\ 0\ 0]^{\mathrm{T}} & [rad/sec], \\ \omega(0) = [0\ 0\ 0]^{\mathrm{T}} & [rad/sec], \\ D(t) = [0\ \ \sin(0.05t + \frac{\pi}{2})\ \ \sin(0.01t + \frac{\pi}{4})\ \ 0]^T & [rad/sec]. \end{array}$$

Note that based on the initial conditions of the simulation, $\overrightarrow{H}_{\mathrm{sc/c/E}}\Big|_{\mathcal{B}} = 0$. The initial parameter estimates and true RWA alignment parameters are given, respectively, by

$$\hat{\theta}(0) = \begin{bmatrix} 0.7121 \\ 0.0928 \\ 0.6959 \\ 0.0928 \\ 0.7121 \\ 0.6959 \\ 0.6845 \\ 0.7290 \\ 0 \\ 0.9916 \\ 0.1292 \\ 0 \end{bmatrix}, \qquad \theta = \begin{bmatrix} 0.7037 \\ 0.0693 \\ 0.7071 \\ 0.0693 \\ 0.7037 \\ 0.7071 \\ 0.9952 \\ 0.0980 \\ 0 \\ 0 \\ -0.7071 \\ 0.7071 \end{bmatrix}. \tag{11}$$

The initial parameter error covariance matrix is assumed to be of the form,

$$P(0|0) = \frac{1}{3} I_{12},$$

where I_{12} is the 12×12 identity matrix. The covariance of the measurement noise ϵ in (8) has been estimated assuming 0.0005 rad/sec independent error standard deviations in measuring the components of the angular velocity vector ω, and 2 rad/sec independent error standard deviations in measuring the components of ν so that

$$R = 10^{-3} \times diag(0.0290, 0.1040, 0.9040).$$

4.1 Case 1: Baseline Adaptation Algorithm with no NME

The first case to be simulated is the baseline adaptation algorithm of reference [11] which is specified, in continuous-time as,

$$\dot{\hat{\theta}} = \gamma Y_1(\nu)^T [-J\omega - Y_1(\nu)\hat{\theta}], \tag{12}$$

and where we choose $\gamma = 10 I_{12}$. In this case, there is no excitation in the null-space, and $n(t) = 0$.

Results. The angular velocity of the spacecraft and RWA flywheel rates are shown in Figures 2 and 3. The parameter error, shown in Figure 4, does *not* converge to zero. This is because the forced trajectory followed by the spacecraft does not ensure persistency of excitation. The angular momentum error of the spacecraft-RWA array system in Figure 5 asymptotically approaches zero.

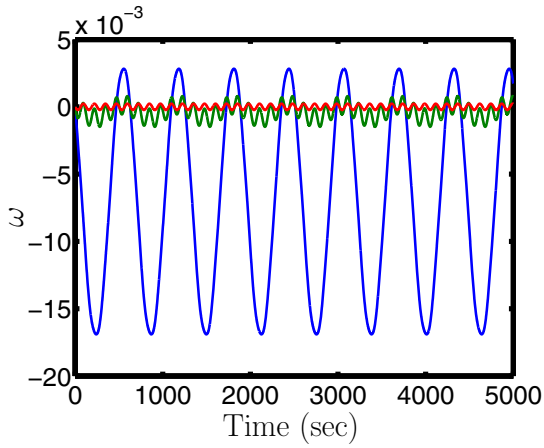


Fig. 2. The time histories of the spacecraft angular velocity components with an adaptation algorithm (12) and no NME

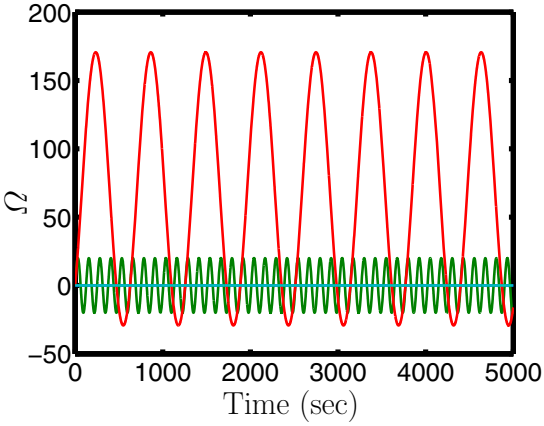


Fig. 3. The time histories of the RWA flywheel rates with an adaptation algorithm (12) and no NME

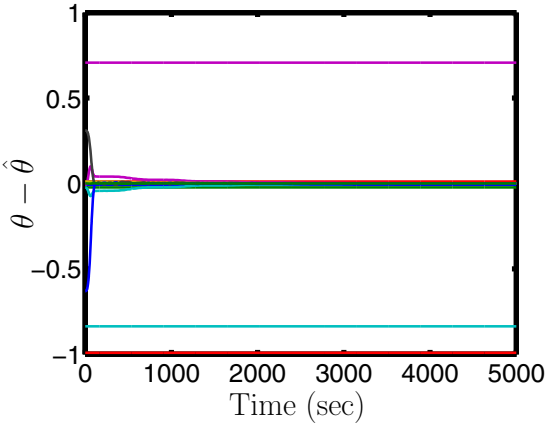


Fig. 4. The time histories of the parameter estimation errors with an adaptation algorithm (12) and no NME

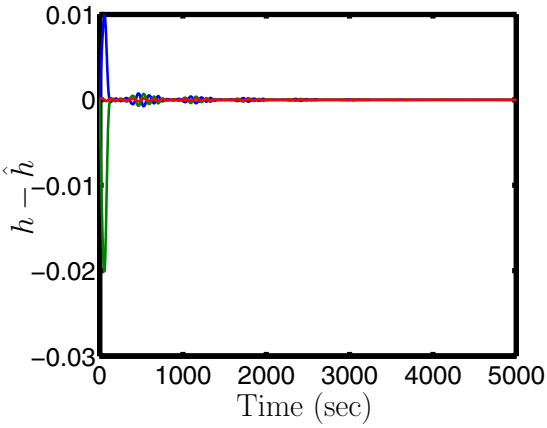


Fig. 5. The time history of the angular momentum error with an adaptation algorithm (12) and no NME

4.2 Case 2: Baseline Adaptation Algorithm and Local Gradient-Based NME Solution

Following [11], the local gradient-based NME signal is now augmented to enhance excitation and facilitate parameter identification. The parameters are estimated by (12), and NME signal, $n(t)$, is generated in the direction of the gradient with respect to $\nu(t)$ of the objective function

$$f = \sum_{i=1}^{3n_a} \sum_{j=1}^{3n_a} q_i^{\mathrm{T}}(t) q_j(t), \tag{13}$$

where $n_a = 4$ is the number of RWA actuators, and $q_i(t)$ and $q_j(t)$ are, respectively, the ith and jth columns of the matrix

$$Q(t) = \left(Y_1^T(t) Y_1(t) + \sum_{i=1}^{N_b - 1} Y_1^T(t-i) Y_1(t-i) \right). \tag{14}$$

Here N_b designates the past time window over which $Q(t)$ is computed, and $Y_1(t)$ depends on $\nu(t)$. The NME from this algorithm adds excitation to the system along the trajectory thereby providing more information and making it possible to identify the parameters of the system without considerably degrading commanded torque tracking performance. The objective function in (13) differs from that of the proposed receding horizon approach. Specifically, (13) does not exploit prediction and minimization with respect to an NME sequence defined over the multi-step prediction horizon. Furthermore, (13) assumes perfect measurements and does not penalize explicitly null space actuation.

Results. The angular velocity of the spacecraft and RWA flywheel rates are shown in Figures 6 and 7. The parameter error using this method is shown in Figure 8. Note that unlike the case without NME, the parameter error converges to zero, however, it has not converged sufficiently over the time interval of 5000 sec. The angular momentum error converges with oscillations at the steady state as shown in Figure 9. The flywheel actuation is shown in Figures 7 and the additional null motion added is shown in Figure 10. More details and discussion of the convergence of the parameters and the trajectory can be found in reference [11].

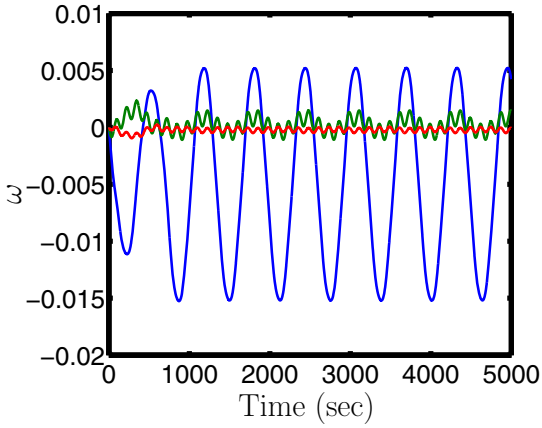


Fig. 6. The time histories of the spacecraft angular velocity components with the baseline NME algorithm and adaptation algorithm given by (12)

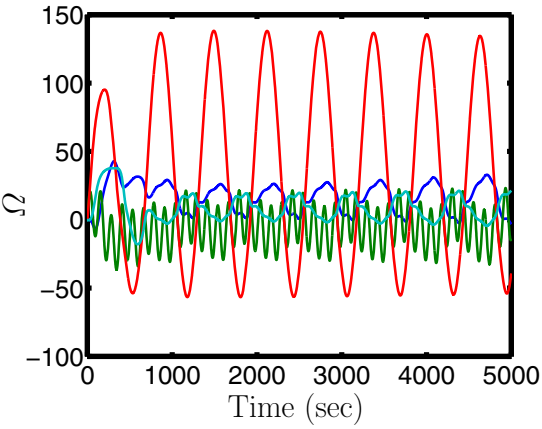


Fig. 7. The time histories of the RWA rates with the baseline NME algorithm and adaptation algorithm given by (12)

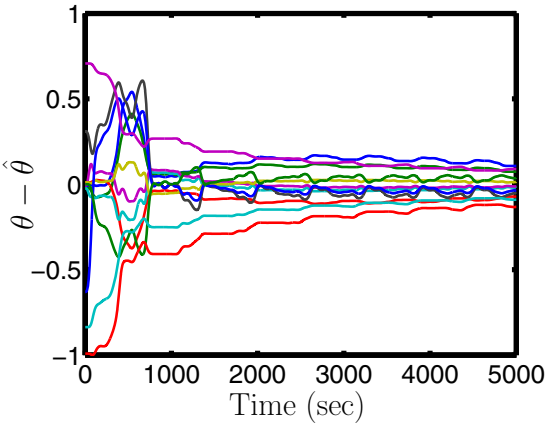


Fig. 8. The time histories of the parameter estimation errors with the baseline NME algorithm and adaptation algorithm given by (12)

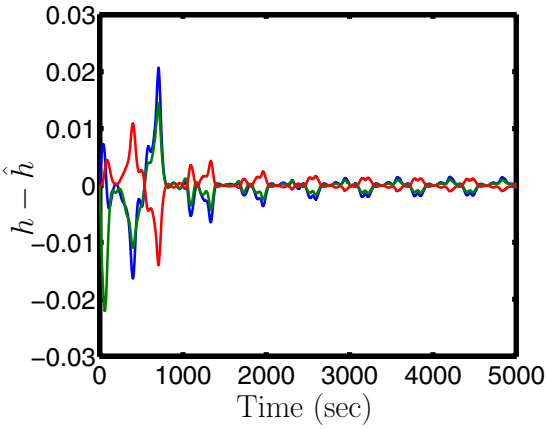


Fig. 9. The time history of the angular momentum error with the baseline NME algorithm and adaptation algorithm given by (12)

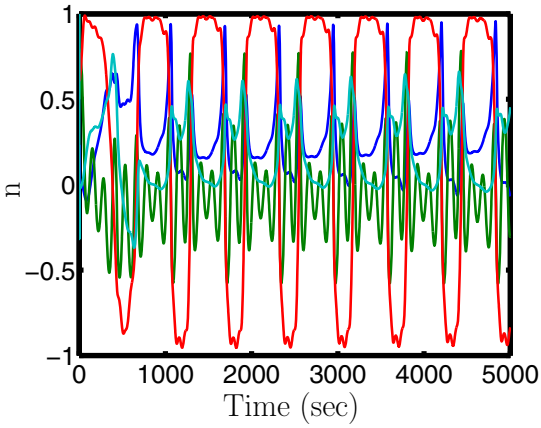


Fig. 10. The time history of the excitation signal n with the baseline NME algorithm and adaptation algorithm given by (12)

4.3 Case 3: RLS Adaptation Algorithm and Receding Horizon NME Solution

The final case in simulation is that of the NME solution found from a receding horizon optimization. Given that the method of choosing NME here is local but based off a larger than a single time-step horizon and that its objective function balances the NME actuation and error covariance, we expect that the receding horizon approach may perform superior to the local-gradient method in Section 4.2. Note also that in the receding horizon case, the noise in the measurements is accounted for in the covariance prediction.

Results. To ensure that the excitation is maintained over time, the weight ρ in (7) is made time-varying and decreased at a linear rate to a constant value. See Figure 11. See also reference [13] for additional remarks. We set $\Delta T = 1$ sec and we use the horizon $N_c = 10$ in (7). While shorter horizons can reduce the computational time and effort, for the assumed levels of measurement noise shorter horizons produce slower parameter error convergence.

The angular velocity of the spacecraft and RWA flywheel rates are shown in Figures 12 and 13. The parameter error for the receding horizon method of NME exhibits faster parameter error convergence over 5000 sec time interval versus the local gradient approach, compare Figures 14 and 8. In addition, the angular momentum error is smaller than with the local-gradient method, compare Figures 15 and 9. Finally, the additional actuation is an order of magnitude less with the receding horizon approach than with the local gradient method, which is evident by comparing the null motion added in Figure 16 and Figure 10. Figure 17 demonstrates that the error covariance matrix is decreasing through plotting of the maximum 1-σ bounds on the covariance matrix. We emphasize that these results are not due to the difference between the parameter update laws (10)

and (12); For instance, by setting $n(t) = 0$ and executing (10), the parameter estimates do not converge to zero, as shown in Figure 18.

Even though the results are dependent on the choices of each algorithm parameters, they do indicate that the receding horizon approach has a potential to induce null motion excitation that facilitates fast parameter adaptation and smaller perturbations to the spacecraft albeit at a higher computational cost.

The analysis of observability has been left out in this paper. For a detailed treatment of the observability of redundant/over-actuated systems see reference [14].

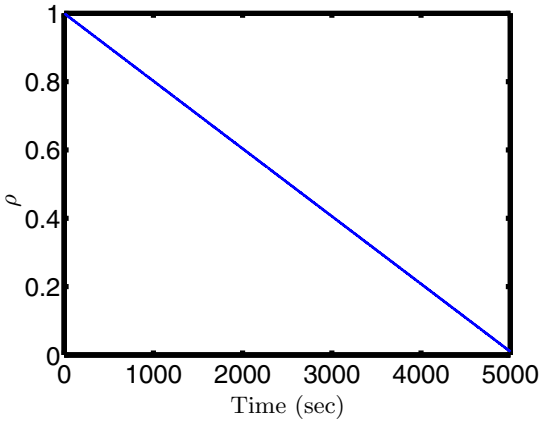


Fig. 11. The time history of the weight ρ

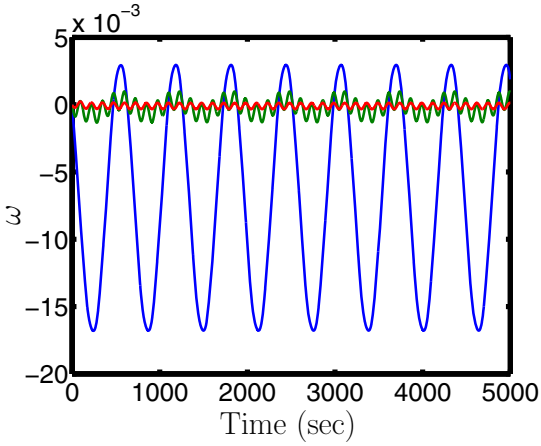


Fig. 12. The time histories of the spacecraft angular velocity components with the receding horizon NME algorithm

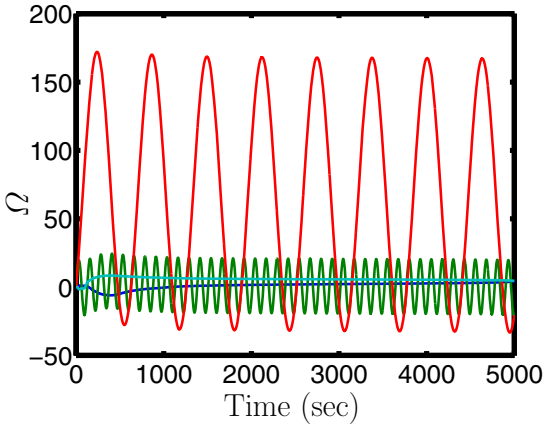


Fig. 13. The time histories of the RWA rates with the receding horizon NME algorithm

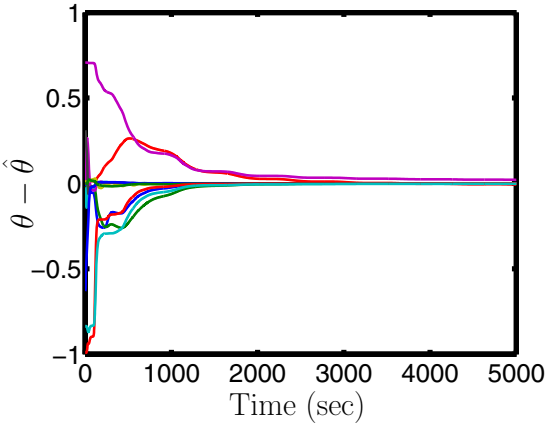


Fig. 14. The time histories of the parameter estimation errors with the receding horizon NME algorithm

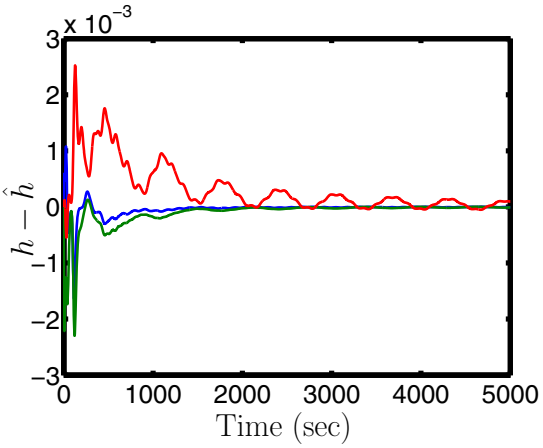


Fig. 15. The time history of the angular momentum error with the receding horizon NME algorithm

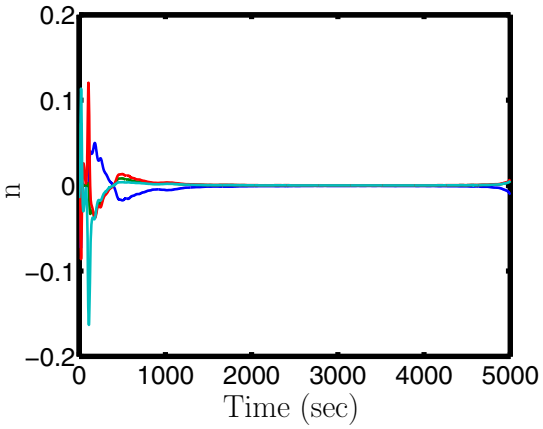


Fig. 16. The time history of the signal n with the receding horizon NME algorithm

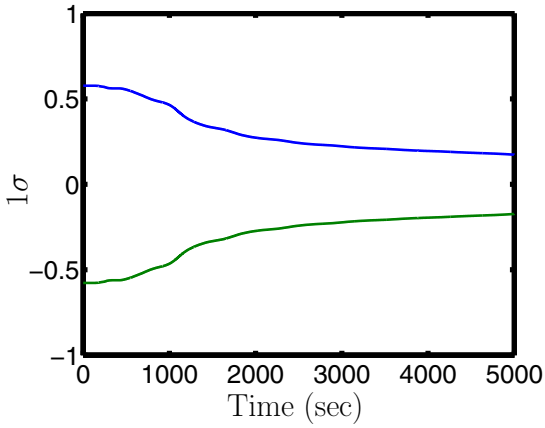


Fig. 17. The time history of the maximum 1σ standard deviation with the receding horizon NME algorithm

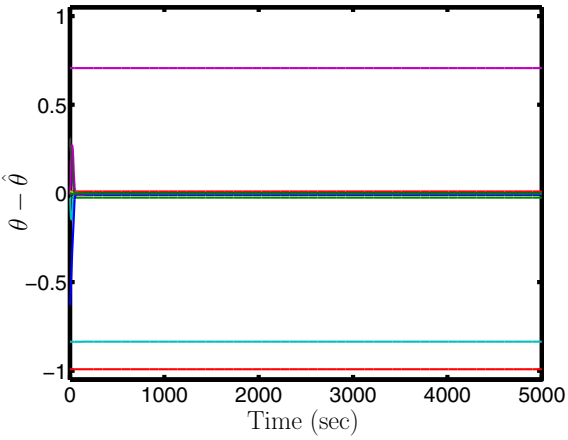


Fig. 18. The time histories of the parameter estimation errors without NME algorithm

5 Conclusion

The Null Motion Excitation (NME) takes advantage of the spacecraft actuation redundancy (over-actuation) to provide excitation for parameter estimation while minimizing the disturbance to the nominal spacecraft maneuver. A specific approach to NME is proposed in this paper. This approach is based on the receding horizon optimization of the excitation input to minimize the predicted estimation error covariance. Simulation results for the case of identifying alignments in the Reaction Wheel Assembly demonstrate that the receding horizon approach ensures faster parameter convergence versus zero excitation case, and that it has a potential to outperform a previously proposed algorithm in [11], albeit at a higher on-board computational cost. We note that the proposed approach may be viewed a variant of Design of Experiments (DoE) technique wherein nominal control signals are augmented with bounded excitation signals that improve parameter identifiability while satisfying the imposed constraints and minimizing the impact on the nominal spacecraft motion. The receding horizon framework is beneficial as it facilitates re-optimizing the excitation trajectory every time the error covariance matrix estimate and parameter estimates are updated from the actual measurements. Our simulation results for the case of a spacecraft actuated by a reaction wheel array demonstrate clearly that the approach is effective even though the current parameter estimates are used in determining the null space and in minimizing the spacecraft disturbance. While we consider the application of this null motion excitation strategy over finite intervals of time only, we note that more general receding horizon controllers can be applied over an infinite time interval and can incorporate the penalty on tracking error in addition to the estimation error in the cost function. The analysis of closed-loop properties of such 'dual adaptive' controllers is beyond the scope of the present paper and is left to future research.

Several assumptions were made to simplify the treatment of the problem, that will be relaxed in future publications. Various enhancements will be pursued. In particular, an approach which uses a Gaussian sum in place of the predicted covariance for an objective function will be investigated. This is needed for accommodating certain sensors such as electro-optical sensors for which the measurement noise has a Poisson rather than a Gaussian distribution. Other developments will be concerned with applying this type of solution that incorporates NME to a slew of different problems (e.g., CMG gimbal axis alignments). In addition, future work will compare the energy, power, and time associated with separate identification maneuvers versus that of this approach.

Acknowledgments. The authors gratefully acknowledge the contribution from the Air Force Office of Scientific Research under the lab task (LRIR11RV15COR) to support this research.

References

1. MacKunis, W., Dupree, K., Fitz-Coy, N., Dixon, W.: Adaptive satellite attitude control in the presence of inertia and CMG gimbal friction uncertainties. Journal of the Astronautical Sciences 56(1), 121–134 (2008)
2. Dixon, W.: Nonlinear control of engineering systems: A Lyapunov-based approach. Birkhauser (2003)
3. Mercker, T., Akella, M.: Onboard adaptive compensation for large-scale misalignments in responsive space systems. In: AAS Guidance and Control Conference, Breckenridge, Co. (2010)
4. Ahmed, J., Coppola, V., Bernstein, D.: Asymptotic tracking of spacecraft attitude motion with inertia matrix identification. In: Proceedings of the 36th IEEE Conference on Decision and Control, vol. 3, pp. 2471–2476. IEEE (2002)
5. Chaturvedi, N., Bernstein, D., Ahmed, J., Bacconi, F., McClamroch, N.: Globally convergent adaptive tracking of angular velocity and inertia identification for a 3-DOF rigid body. IEEE Transactions on Control Systems Technology 14(5), 841–853 (2006)
6. Ma, O., Dang, H., Pham, K.: On-Orbit Identification of Inertia Properties of Spacecraft Using a Robotic Arm. Journal of Guidance, Control, and Dynamics 31(6) (2008)
7. Chakrabortty, A., Arcak, M., Tsiotras, P.: Robust design of a spacecraft attitude tracking control system with actuator uncertainties. In: 47th IEEE Conference on Decision and Control, CDC 2008, pp. 1587–1592. IEEE (2009)
8. Costic, B., Dawson, D., De Queiroz, M., Kapila, V.: A quaternion-based adaptive attitude tracking controller without velocity measurements. In: Proceedings of the 39th IEEE Conference on Decision and Control, vol. 3, pp. 2424–2429. IEEE (2002)
9. Astrom, K., Wittenmark, B.: Adaptive control. Addison-Wesley Longman Publishing Co., Inc., Boston (1994)
10. Sastry, S., Bodson, M.: Adaptive control: stability, convergence, and robustness (1989)
11. Leve, F., Jah, M.: Spacecraft actuator alignment determination through null motion excitation. In: Proceedings of 62nd International Astronautical Congress (2011)
12. Kolmanovsky, I., Winstead, V.: A receding horizon optimal control approach to active state and parameter estimation in automotive systems. In: Proceedings of 2006 IEEE Conference on Control Applications, pp. 2796–2801. IEEE (2006)
13. Kolmanovsky, I., Filev, D.P.: Optimal finite and receding horizon control for identification in automotive systems. In: Alberer, D., Hjalmarsson, H., del Re, L. (eds.) Identification for Automotive Systems. LNCIS, vol. 418, pp. 327–348. Springer, Heidelberg (2012)
14. Pittelkau, M.E.: Calibration and attitude determination with redundant inertial measurement units. Journal of Guidance, Control, and Dynamics 28(4), 743–752 (2005)

Spacecraft Constrained Maneuver Planning Using Positively Invariant Constraint Admissible Sets

Avishai Weiss[1], Christopher Petersen[1], Morgan Baldwin[2], Richard Scott Erwin[2], and Ilya V. Kolmanovsky[1]

[1] University of Michigan, Ann Arbor, MI, USA
{avishai,cdpete,ilya}@umich.edu
[2] Space Vehicles Directorate, Kirtland Air Force Base, NM, USA
AFRL.RVSV@kirtland.af.mil

Abstract. The paper considers spacecraft motion planning based on the use of safe positively invariant sets. In this approach, a connectivity graph is constructed between a set of forced equilibria, forming a virtual net that is centered around a nominal orbital position. The connectivity between two equilibria is determined based on safe positively invariant sets in order to guarantee that transitions between equilibria can be effected while spacecraft actuator limits are adhered to and debris collisions are avoided. A graph search algorithm is implemented to find the shortest path around the debris. Simulation results are presented that illustrate this approach.

1 Introduction

Orbital debris is a growing problem, with about 40% of ground-trackable objects originating from explosions that now number approximately 5 per year [1]. Spacecraft maneuver planning procedures thus have to address debris avoidance requirements. While obstacle avoidance is a standard problem in robotics [2,3], the related spacecraft problems have several unique features. In particular, the space environment is relatively uncluttered, thus permitting for a variety of maneuvers. Spacecraft dynamics are quite different from those of typical robots. Maneuver efficiency with respect to time and fuel consumption is a critical consideration. The states of the spacecraft and the debris can only be estimated, often with a significant estimation error. Finally, computational algorithms must be fast and optimized given moving objects and the limited computing power onboard most spacecraft. These unique features of spacecraft maneuver planning problems provide the motivation for the development of specialized algorithms.

In [4], we have introduced an on-board maneuver planning approach based on the use of constraint-admissible positively invariant sets to determine connectivity between a set of forced and unforced spacecraft equilibria forming a *virtual net* in the vicinity of the spacecraft. Two equilibria are connected if a choice of a Linear Quadratic (LQ) feedback gain can be made that results in a transition between the equilibria which avoids the debris collision and satisfies the

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_26

limits on thrust. The connectivity graph for all the equilibria in the net is constructed and real-time graph search algorithms are used to optimize maneuver time or fuel consumption while avoiding debris collisions. While safe positively invariant sets around both forced and unforced equilibria are used in maneuver construction, we note that the spacecraft does not spend any finite amount of time at selected forced equilibria, and that the switch to the next set-point in the sequence occurs immediately once appropriate conditions are satisfied. We also note that fuel consumption or maneuver time optimization is performed at the graph search stage.

Unlike the open-loop trajectory optimization approaches, we do not rely on precise assignment of spacecraft position to the time instants along the trajectory, but instead switch to the next set-point and controller gain once appropriate conditions are satisfied. While this approach is conservative, it facilitates fault-tolerant and disturbance-tolerant execution of the maneuvers. Furthermore, by using disturbance-invariant sets [5] in the construction, we can assure robustness to unmeasured (but set-bounded) disturbances and uncertainties. This extension to handling unmeasured disturbances and uncertainties using techniques of [5] is not pursued here but will be reported elsewhere.

To facilitate the on-board computations of the connectivity graph, a fast growth distance computation procedure between two ellipsoidal sets has been proposed in [4]. In this approach, using the Karush-Kuhn-Tucker conditions, the growth distance computations are reduced to a root finding problem for the scalar value of the Lagrange multiplier. Then a predictor-corrector dynamic Newton-Raphson algorithm is used to update the Lagrange multiplier thereby rapidly estimating the growth distance from different equilibria in the virtual net to the debris.

In this paper, we incorporate limited thrust requirements into the computation of thrust limit on the growth distance, and we simulate maneuvers that adhere to the limited thrust constraints. Even though the computation of thrust limits on the growth distance can be performed offline for the nominal operating conditions, fast computational procedures are beneficial in case of thruster failures, degradations, and restrictions on thrust directions (e.g., caused by the presence of other spacecraft nearby), all of which can lead to changing constraints on thrust during spacecraft missions. We show that in the case of polyhedral norm bounds on thrust the problem of finding thrust limit on growth distance is easily and explicitly solvable.

The related literature on spacecraft trajectory optimization with obstacle/debris avoidance is surveyed in [4]. Previous research addresses topics in spacecraft trajectory optimization [6], collision avoidance strategies based on risk assessment [7], the use of artificial potential functions [8,9], and the use of conventional and mixed integer linear programming techniques [10,11,12,13].

The paper is organized as follows. In Sections 2-6, we develop and introduce the relative motion model, the virtual net, an LQ controller with gain switching, positively invariant sets, and debris representation. In Section 7, we build upon these concepts with growth distances and connectivity graphs in order to develop

the static debris avoidance approach. Section 8 addresses cost matrices. Finally, Section 9 presents simulations that illustrate these approaches for avoiding both static and moving debris. The moving debris case is handled by covering its path in the relative frame by a union of a finite number of static debris. Concluding remarks are made in Section 10.

2 Relative Motion Model

The spacecraft relative motion model presents spacecraft dynamics in the (non-inertial) Hill's frame, where the origin is a target location on a nominal circular orbit.

2.1 Nonlinear Equations of Motion

The relative position vector of the spacecraft with respect to a target location on an orbit is expressed as

$$\delta\vec{r} = x\hat{i} + y\hat{j} + z\hat{k},$$

where x, y and z are the components of the position vector of the spacecraft relative to the target location and $\hat{i}$, $\hat{j}$, $\hat{k}$ are the unit vectors of the Hill's frame. The Hill's frame has its x-axis along the orbital radius, y-axis orthogonal to the x-axis and in the orbital plane, and z-axis orthogonal to orbital plane.

The position vector of the spacecraft with respect to the center of the Earth is given by $\vec{R} = \vec{R}_0 + \delta\vec{r}$, where $\vec{R}_0$ is the nominal orbital position vector. The nonlinear equation of motion for the spacecraft (relative to an inertial frame) is given by

$$\ddot{\vec{R}} = -\mu\frac{\vec{R}}{R^3} + \frac{1}{m_c}\vec{F}, \tag{1}$$

where $\vec{F}$ is the vector of external forces applied to the spacecraft, $R = |\vec{R}|$, m_c is the mass of the spacecraft, and μ is the gravitational constant.

2.2 Linearized CWH Equations

For $\delta r \ll R$, the linearized Clohessy-Wiltshire-Hill (CWH) equations [14] approximate the relative motion of the spacecraft on a circular orbit as

$$\begin{aligned} \ddot{x} - 3n^2x - 2n\dot{y} &= \frac{F_x}{m_c}, \\ \ddot{y} + 2n\dot{x} &= \frac{F_y}{m_c}, \\ \ddot{z} + n^2z &= \frac{F_z}{m_c}, \end{aligned} \tag{2}$$

where F_x, F_y, F_z are components of the external force vector (excluding gravity) acting on the spacecraft, $n = \sqrt{\frac{\mu}{R_0^3}}$ denotes the mean motion of the nominal

orbit, and R_0 is the nominal orbital radius. The linearized dynamics account for differences in gravity between the spacecraft and nominal orbital location, and for relative motion effects.

The spacecraft relative motion dynamics in the orbital plane (x and y) and in the out-of-orbital plane (z) are decoupled. The in-plane dynamics are Lyapunov unstable (2 eigenvalues at the origin and 2 eigenvalues on the imaginary axis at $\pm nj$), while the out-of-plane dynamics are Lyapunov stable (2 eigenvalues on the imaginary axis at $\pm nj$). The in-plane dynamics are completely controllable from F_y input but are not controllable from F_x input. The out-of-plane dynamics are controllable from F_z input. These dynamics are clearly different from typical ground robots.

Assuming a sampling period of ΔT sec, we can convert the model (2) to a discrete-time form

$$X(t+1) = AX(t) + BU(t), \tag{3}$$

where $X(t) = [x(t),\ y(t),\ z(t),\ \dot{x}(t),\ \dot{y}(t), \dot{z}(t)]^{\mathrm{T}}$ is the state vector at the time instant $t \in Z^+$, $U(t) = [F_x(t),\ F_y(t),\ F_z(t)]^{\mathrm{T}}$ is the control vector of thrust forces at the time instant $t \in Z^+$, and $A = \exp(A_c \Delta T)$, $B = \int_0^{\Delta T} \exp(A_c(\Delta T - \tau))d\tau B_c$ are the discretized matrices obtained based on the continuous-time system realization (A_c, B_c) in (2).

3 Virtual Net

The virtual net comprises a finite set of equilibria, $X_e(r)$, corresponding to a finite set of prescribed spacecraft relative positions $r \in \mathcal{N} = \{r_1, r_2, \ldots, r_n\} \subset R^3$ and constant control inputs,

$$X_e(r_k) = \begin{bmatrix} r_k & 0 \end{bmatrix}^{\mathrm{T}} = \begin{bmatrix} r_{x,k} & r_{y,k} & r_{z,k} & 0 & 0 & 0 \end{bmatrix}^{\mathrm{T}}, \quad k = 1, \cdots, n. \tag{4}$$

Note that for the equilibria in the virtual net velocity states are zero, and n is the number of equilibria in the virtual net. See Figure 1. We assume that for all $r \in \mathcal{N}$, the corresponding values of control necessary to support the specified equilibria in steady-state satisfy the imposed thrust limits.

4 LQ Controller with Gain Switching

A conventional Linear-Quadratic (LQ) feedback

$$U = K(X - X_e(r)) + \Gamma r = KX + H(K)r, \tag{5}$$

can control the spacecraft thrust to arrive at a commanded equilibrium (4), where

$$\Gamma = \begin{bmatrix} -3n^2 m_c & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & n^2 m_c \end{bmatrix},$$

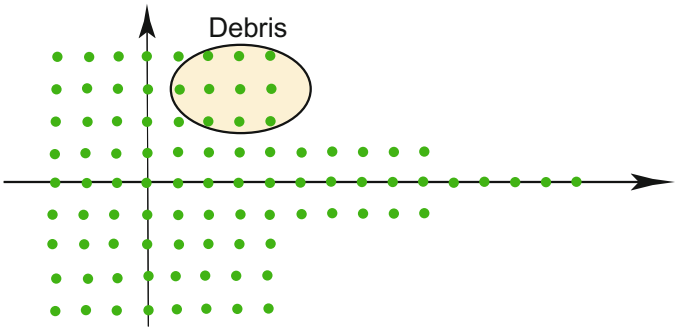


Fig. 1. The virtual net for debris avoidance. Dots correspond to positions $r_k, k = 1, \ldots, n$, at equilibria, $X_e(r)$, on a virtual net. The ellipsoid represents the debris position and uncertainty.

$$H(K) = \Gamma - K \begin{bmatrix} I_3 \\ 0_3 \end{bmatrix},$$

and where I_3 denotes the 3×3 identity matrix while 0_3 denotes the 3×3 zero matrix. The LQ controller provides an asymptotically stable closed-loop system but does not enforce debris avoidance constraints.

To provide greater flexibility in handling constraints, a multimode controller architecture is employed [15]. Specifically, we assume that a finite set of LQ gains $K \in \mathcal{K} = \{K_1, \cdots, K_m\}$ is available to control the spacecraft. By using a large control weight in the LQ cost functional, motions with low fuel consumption yet large excursions can be generated; using a large control weight in the LQ cost, motions with short transition time can be generated [16]. We assume that a preference ordering has been defined and the gains are arranged in the order of descending preference, from K_1 being the highest preference gain to K_m being the lowest preference gain.

5 Positively Invariant Sets

The ellipsoidal set

$$\bar{C}(r, K) = \{X \in R^6 : \frac{1}{2}(X - X_e(r))^T P(K)(X - X_e(r)) \leq 1\} \subset R^6, \quad (6)$$

where

$$\bar{A}(K)^{\mathrm{T}} P \bar{A}(K) - P < 0, \quad (7)$$

$\bar{A}(K) = (A + BK)$, and $P = P(K) > 0$ is positively invariant for the closed-loop dynamics. Positive invariance implies that any trajectory of the closed-loop system that starts in $\bar{C}(r, K)$ is guaranteed to stay in $\bar{C}(r, K)$ as long as the same LQ gain K is used and the set-point command r is maintained. To achieve the positive invariance, the matrix P can be obtained as the solution of the

discrete-time Riccati equation in the LQ problem or as the solution of the above Lyapunov equation for the closed-loop asymptotically stable system. We note that, because the system is linear, the positive invariance of $\bar{C}(r, K)$ implies the positive invariance of the scaled set

$$C(r, K, \rho) = \{X \in R^6 : \frac{1}{2}(X - X_e(r))^T P(K)(X - X_e(r)) \leq \rho^2\}, \quad \rho \geq 0.$$

Geometrically, the set $C(r, K, \rho)$ corresponds to an ellipsoid scaled by the value of ρ and centered around $X_e(r)$, $r \in \mathcal{N}$.

6 Debris Representation

We use a set, $O(z, Q)$, centered around the position $z \in R^3$, to over-bound the position of the debris, i.e.,

$$O(z, Q) = \{X \in R^6 : (SX - z)^T Q(SX - z) \leq 1\}, \tag{8}$$

where $Q = Q^{\mathrm{T}} > 0$ and

$$S = \begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0 \end{bmatrix}. \tag{9}$$

The set $O(z, Q)$ can account for the debris and spacecraft physical sizes and also for the uncertainties in the estimation of the debris/spacecraft position. Note that the set $O(z, Q)$ has an ellipsoidal shape in the position directions and it is unbounded in the velocity directions. Ellipsoidal sets, rather than polyhedral sets, are used here to over-bound the debris, since ellipsoidal bounds are typically produced by position estimation algorithms, such as the Extended Kalman Filter (EKF).

7 Debris Avoidance Approach

Consider now $r_i \in \mathcal{N}$, representing a possible position on the net that the spacecraft can move to as a part of the debris avoidance maneuver. Suppose that the current state of the spacecraft is $X(t_0)$ at the time instant $t_0 \in Z^+$. If there exists a $\rho \geq 0$ and $K_j \in \mathcal{K}$ such that

$$X(t_0) \in C(r_i, K_j, \rho) \text{ and } O(z, Q) \cap C(r_i, K_j, \rho) = \emptyset, \tag{10}$$

the spacecraft can move to the position $r_i \in \mathcal{N}$ by engaging the control law with $r(t) = r_i$ and $K(t) = K_j$, $t \geq t_0$, and without hitting the debris confined to $O(z, Q)$. This idea underlies our subsequent approach to debris avoidance, where we maintain the spacecraft within a tube formed by positively invariant sets that do not intersect with debris.

To avoid a non-stationary debris, its path can be covered by a union of a finite number of ellipsoidal sets,

$$\mathcal{D} = \bigcup_{l=1}^{l=n_d} O(z_l, Q_l), \tag{11}$$

where the center of the lth set is denoted by $z_l \in R^3$, and the lth set shape is defined by $Q_l = Q_l^{\mathrm{T}} > 0$. Then, the debris avoidance condition for the closed-loop trajectory that emanates from $X(0)$ with the set-point r_i and gain K_j is given by

$$X(0) \in C(r_i, K_j, \rho) \text{ and } O(z_l, Q_l) \cap C(r_i, K_j, \rho) = \emptyset, \quad \text{for all } l = 1, \cdots, n_d. \tag{12}$$

The same approach, with larger n_d, can be used to handle *multiple* non-stationary debris. Note, however, that this approach is conservative as it does not account for the debris progressions along their paths versus time. It is appropriate for debris moving at high relative velocity. The case of debris moving at a low relative velocity will be considered in other publications.

7.1 Growth Distances

The minimum value of $\rho \geq 0$ for which $O(z, Q) \bigcap C(r, K, \rho) \neq \emptyset$ is referred to as the *growth distance* [17]. This growth distance can also be viewed as the least upper bound on the values of ρ for which $O(z, Q)$ and $C(r, K, \rho)$ do not intersect. See Figure 2. We use the notation $\rho_g(r, K, Q, z)$ to reflect the dependence of the growth distance on the set-point $r \in \mathcal{N}$, the control gain $K \in \mathcal{K}$ and the obstacle parameters Q and z.

Note that the growth distance depends on the position of the debris which may be unknown in advance. Consequently, growth distance computations have to be performed online.

Since spacecraft have limited thrust, we additionally define a maximum value of $\rho = \rho_u(r, K)$ for which $X \in C(r, K, \rho_u(r, K))$ implies that the thrust $U = KX + H(K)r$ satisfies the imposed thrust limits. We refer to ρ_u as the *thrust limit on growth distance*. Unlike ρ_g, the value of ρ_u does not depend on the position or shape of the debris and can be pre-computed off-line.

Finally, we define *the thrust limited growth distance*

$$\rho^*(r, K, Q, z) = \min\{\rho_g(r, K, Q, z), \rho_u(r, K)\}. \tag{13}$$

Note that $X(t_0) \in C(r_i, K_j, \rho^*(r_i, K_j, Q, z))$ implies that the ensuing closed-loop spacecraft trajectory under the control (5), where $r(t) = r_i$ and $K(t) = K_j$ for $t \geq t_0$, satisfies the thrust limits and avoids collisions with a debris confined to $O(z, Q)$.

The above definitions were given for the case of a single stationary debris, $O(z, Q)$. In the case of multiple debris, the growth distance is replaced by the *multi-growth distance*, which is the minimum growth distance to each of $O(z_l, Q_l)$, $l = 1, \cdots, n_d$.

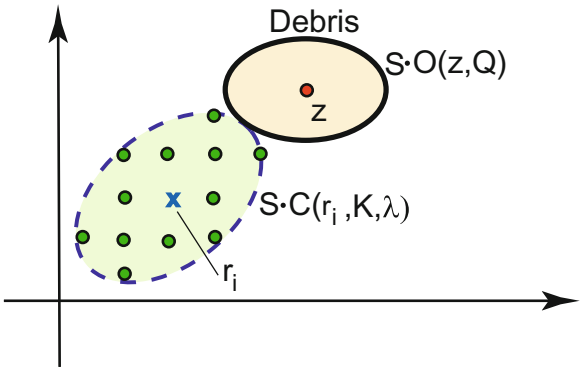


Fig. 2. The positively invariant set is grown till touching the debris. The spacecraft can move from any of the equilibria on the virtual net inside the positively invariant set $C(r, K, \rho)$ to $X_e(r_i)$ marked by 'x' without colliding with the debris.

7.2 Growth Distance Computations

Define $\bar{X} = X - X_e(r)$ and $\alpha = 2\rho^2$. The problem of determining the growth distance $\rho_g(r, K, Q, z)$, reduces to the constrained optimization problem

$$\begin{array}{ll} \min\limits_{\alpha,\bar{X}} & \alpha \\ \text{subject to} & \bar{X}^{\mathrm{T}} P \bar{X} \leq \alpha \\ & ((S(\bar{X} + X_e(r)) - z)^{\mathrm{T}} Q((S(\bar{X} + X_e(r)) - z) \leq 1, \end{array} \tag{14}$$

where we find the minimum size invariant ellipsoid that shares a common point with the debris. To solve this optimization problem, we use the Karush-Kuhn-Tucker (KKT) conditions [18,19]. Note that standard linear independence constraint qualification conditions hold given that $P > 0$. We define

$$\mathcal{L} = \alpha + \lambda_1(\bar{X}^{\mathrm{T}} P \bar{X} - \alpha) + \lambda_2((S(\bar{X} + X_e(r)) - z)^{\mathrm{T}} Q(S(\bar{X} + X_e(r)) - z) - 1),$$

where λ_1 and λ_2 are Lagrange multipliers. The stationarity of the Lagrangian (setting partial derivative equal to zero) with respect to α yields $\lambda_1 = 1$. The stationarity of the Lagrangian with respect to $\bar{X}$ yields

$$\bar{X} = \bar{X}(\lambda_2, r, z) = -(P + \lambda_2 S^{\mathrm{T}} Q S)^{-1} S^{\mathrm{T}} Q(S X_e(r) - z)\lambda_2, \tag{15}$$

where the scalar $\lambda_2 \geq 0$ is to be determined. Note that $P > 0$, $S^{\mathrm{T}} Q S \geq 0$, $\lambda_2 \geq 0$ (as the Lagrange multiplier corresponding to an inequality constraint) imply that $(P + \lambda_2 S^{\mathrm{T}} Q S)$ is invertible. The problem reduces to finding a nonnegative scalar λ_2, which is the root of

$$F(\lambda_2, r, z) = (SX - z)^{\mathrm{T}} Q(SX - z) - 1 = 0, \tag{16}$$

where

$$X = \bar{X}(\lambda_2, r, z) + X_e(r).$$

The scalar root finding problem (16) has to be solved online multiple times for different $r \in \mathcal{N}$, and in the case of avoiding a predicted debris path also for different z's. To solve this problem fast, while reusing previously found solutions as approximations, a dynamic Newton-Raphson's algorithm is used [20,19,21]. This algorithm uses predictor-corrector updates to track the root as a function of z and r, and is given by

$$\lambda_2^{k+1,+} = \lambda_2^k + \{\frac{\partial F}{\partial \lambda_2}(\lambda_2^k, z^k, r^k)\}^{-1}\{-F(\lambda_2^k, z^k, r^k) - \frac{\partial F}{\partial z}(\lambda_2^k, z^k, r^k)(z^{k+1} - z^k) - \frac{\partial F}{\partial r}(\lambda_2^k, z^k, r^k)(r^{k+1} - r^k)\},$$
$$\lambda_2^{k+1} = max\{0, \lambda_2^{k+1,+}\}.$$

To implement the algorithm, we take advantage of the known functional form for F and explicitly compute the partial derivatives,

$$\begin{aligned}
\frac{\partial \bar{X}}{\partial \lambda_2} &= (P + \lambda_2 S^{\mathrm{T}} Q S)^{-1} \left\{-S^{\mathrm{T}} Q (S X_e(r) - z) - S^{\mathrm{T}} Q S \bar{X}\right\}, \\
\frac{\partial F}{\partial \lambda_2} &= 2(SX - z)^{\mathrm{T}} Q (S \frac{\partial \bar{X}}{\partial \lambda_2}), \\
\frac{\partial \bar{X}}{\partial r} &= (P + \lambda_2 S^{\mathrm{T}} Q S)^{-1} \left\{-S^{\mathrm{T}} Q S \Omega\right\} \lambda_2, \\
\frac{\partial F}{\partial r} &= 2(S\bar{X} - z + r)^{\mathrm{T}} Q (S \frac{\partial \bar{X}}{\partial r} + I_3), \\
\frac{\partial \bar{X}}{\partial z} &= (P + \lambda_2 S^{\mathrm{T}} Q S)^{-1} S^{\mathrm{T}} Q S \Omega \lambda_2, \\
\frac{\partial F}{\partial z} &= 2(S\bar{X} - z + r)^{\mathrm{T}} Q (S \frac{\partial \bar{X}}{\partial z} - I_3),
\end{aligned} \tag{17}$$

where, $X_e(r) = \Omega r$,

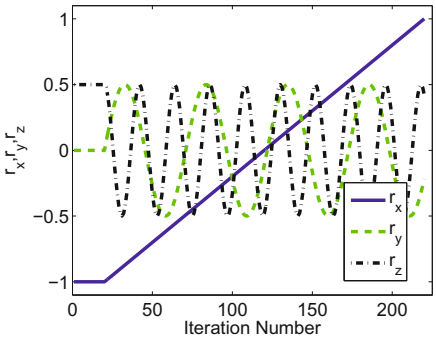
$$\Omega = \begin{bmatrix} I_3 \\ 0 \end{bmatrix},$$

and I_3 denotes the 3×3 identity matrix. Note that $S\Omega = I_3$.

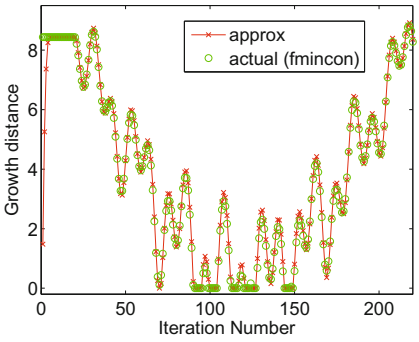
Figure 3 illustrates growth distance tracking. For the first 20 iterations, r^k is held constant to enable initial convergence of the algorithm. Then, r^k varies through the virtual net. One iteration of the Newton-Raphson algorithm per value of r^k is used to update the root, λ_2^{k+1}. Figure 3b demonstrates that the growth distance tracking is accurate. The growth distance is occasionally zero indicating an overlap between several r^k and the debris. Figure 3c illustrates the trajectory of r^k in three dimensions.

7.3 Thrust Limit on Growth Distance Computations

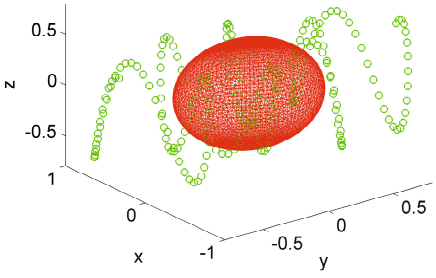
Suppose that the thrust limits are expressed in the form $||LU|| \leq 1$ for an appropriately defined matrix L and norm $||\cdot||$. The computational procedures



(a)



(b)



(c)

Fig. 3. (a) Components of r, r_x, r_y and r_z varying versus the iteration number. (b) Growth distance versus iteration number computed by dynamic Newton-Raphson algorithm. (c) The trajectory of r and the debris.

to determine $\rho_u(r, K)$ involve solving a bilevel optimization problem where $||L(KX + H(K)r)||$ is maximized subject to the constraint $X \in C(r, K, \alpha)$, and bisections are performed on the value of α so that the maximum value is

driven to 1. As we demonstrate in this section, in special cases this computation can be greatly simplified.

Suppose that the thrust constraints are prescribed in terms of polyhedral norm bounds, specifically

$$e_i^{\mathrm{T}}(KX + Hr) \leq u_{\max}, \quad i = 1, 2, \cdots, m, \tag{18}$$

where e_i are the vertices of the unit norm polytope, and $u_{\max}$ is the norm bound. The infinity norm, for instance, has $m = 6$, and

$$\begin{aligned} e_1 &= \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad e_2 = \begin{bmatrix} -1 \\ 0 \\ 0 \end{bmatrix} \quad e_3 = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \\ e_4 &= \begin{bmatrix} 0 \\ -1 \\ 0 \end{bmatrix} \quad e_5 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad e_6 = \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}. \end{aligned} \tag{19}$$

In the case of non-polyhedral norm bounds, such as the 2-norm, an approximation by a polyhedral norm bound may be employed.

The thrust limit on the growth distance is then determined based on solving, for $i = 1, \cdots, n$, the optimization problems

$$\begin{aligned} &\text{maximize } e_i^{\mathrm{T}}(KX + Hr) \\ &\text{subject to } \tfrac{1}{2}(X - X_e(r))^{\mathrm{T}} P (X - X_e(r)) \leq c. \end{aligned} \tag{20}$$

If the value of c is found for which the solutions X_i^* of (20) satisfy $\max_i\{e_i^{\mathrm{T}}(KX_i^* + Hr)\} = u_{\max}$, then $\rho_u(r, K) = \sqrt{c}$.

The problem (20) can be solved by diagonalizing P, using an orthogonal matrix, V,

$$P = V^{\mathrm{T}} \Lambda V, \ \Lambda = \mathrm{diag}[\lambda_1^2, \cdots, \lambda_6^2], \lambda_i > 0.$$

By defining, $z = X - X_e(r)$, and ζ so that

$$z = V^{\mathrm{T}} \Lambda^{-\frac{1}{2}} \zeta,$$

it follows that

$$\begin{aligned} z^{\mathrm{T}} P z &= \zeta^{\mathrm{T}} \Lambda^{-\frac{1}{2}} V P V^{\mathrm{T}} \Lambda^{-\frac{1}{2}} \zeta \\ &= \zeta^{\mathrm{T}} \zeta. \end{aligned}$$

The problem (20) can now be re-written as

$$\begin{aligned} &\text{maximize } h_i^{\mathrm{T}} \zeta + e_i^{\mathrm{T}} \Gamma r \\ &\text{subject to } \tfrac{1}{2} \zeta^{\mathrm{T}} \zeta \leq c, \end{aligned} \tag{21}$$

where

$$h_i^{\mathrm{T}} = e_i^{\mathrm{T}} K V^{\mathrm{T}} \Lambda^{-\frac{1}{2}}.$$

The solution to the constrained maximization problem (21) of maximizing the inner-product of two vectors over a unit 2-norm ball is given by

$$\zeta_i = \frac{h_i}{||h_i||}\sqrt{2c}, \tag{22}$$

where $||\cdot||$ denotes the vector 2-norm. The maximum value of the objective function in (20) is given by

$$||h_i||\sqrt{2c} + e_i^{\mathrm{T}} \Gamma r.$$

Consequently, to satisfy (18), we let

$$c = \begin{cases} 0, & \text{if } \exists\ i : u_{\max} \leq e_i^{\mathrm{T}} \Gamma r, \\ \min\limits_i \dfrac{1}{2}\left(\dfrac{u_{\max} - e_i^{\mathrm{T}} \Gamma r}{||h_i||}\right)^2, & \text{otherwise.} \end{cases} \tag{23}$$

Thus, the problem of finding the thrust limit on the growth distance for polyhedral norm bounds has an explicit solution given by (23). Even though the computation of thrust limits on the growth distance can be performed offline for the nominal operating conditions, fast computational procedures are beneficial in case of thruster failures, degradations, and restrictions on thrust directions (e.g., caused by the presence of other spacecraft nearby), all of which can lead to changing constraints on thrust during spacecraft missions.

We note that the condition $u_{\max} \geq \max_i\{e_i^{\mathrm{T}} \Gamma r\}$ is satisfied if the available thrust can maintain the equilibrium $X_e(r)$ in steady-state. We also note, that, based on the form of Γ, c is independent of r_y, the in-track component of the equilibrium in the virtual net. Hence the computations of $\rho_u(r, K)$ need only be performed with $r_y = 0$.

When a spacecraft does not have independent thrusters in x, y and z directions, a 2-norm thrust limit is more practical. Unfortunately, (20) is, in general, a non-convex problem. In this case, the 2-norm bound can be approximated by a polyhedral norm bound (18), with the vertices e_i selected on the unit 2-norm ball in R^3. We note that higher accuracy of this approximation requires a higher number of vertices in (18), which thus, complicates (23).

7.4 Connectivity Graph and Graph Search

We now introduce a notion of *connectivity* between two vertices of the virtual net, $r_i \in \mathcal{N}$ and $r_j \in \mathcal{N}$. The vertex r_i is connected to the vertex r_j if there exists a gain $K \in \mathcal{K}$ such that

$$X_e(r_i) \in int\, C(r_j, K, \rho^*(r_j, K, Q, z)), \tag{24}$$

where *int* denotes the interior of a set. The connectivity implies that a spacecraft located close to an equilibrium corresponding to r_i can transition to an equilibrium $X_e(r_j)$ by using limited thrust and avoiding collision with the debris. We

note that if r_i is connected to r_j this does not imply that, in turn, r_j is connected to r_i. We also note that connectivity depends on the existence of an appropriate control gain from the set of gains $\mathcal{K}$ but the condition (24) does not need to hold for all gains.

The on-line maneuver is computed according to the following procedure (for simplicity, described here for the case of a single debris):

Step 1: Determine the debris location and shape (i.e., z and Q).

Step 2: By using fast growth distance computations, determine the thrust limited growth distance based on (13), with ρ_g computed online and ρ_u precomputed offline.

Step 3: Construct a graph connectivity matrix between all $r_i, r_j \in \mathcal{N}$. In the graph connectivity matrix, if two vertices are not connected, the corresponding matrix element is zero; if they are connected the corresponding matrix element is 1. In parallel, build the control gain selectivity matrix, which identifies the index of the highest preference gain K for which r_i and r_j are connected. This gain will be applied if the edge connecting r_i and r_j is traversed.

Step 4: Perform graph search (using any standard graph search algorithm) to determine a sequence of connected vertices $r[k] \in \mathcal{N}$ and control gains $K[k] \in \mathcal{K}$, $k = 1, \cdots, l_p$, such that $r[1]$ satisfies the initial constraints, $r[l_p]$ satisfies the final constraints, and a cost function (see Section 8) is minimized.

After the path has been determined as a sequence of the set-points and the corresponding control gains, the execution of the path proceeds by checking if the current state, $X(t)$ is in the safe positively invariant set corresponding to the next reference r^+ and next control gain K^+ in the sequence; if it is, then the controller switches to this reference and control gain:

$$X(t) \in C(r^+, K^+, \rho^*(r^+, K^+, Q, z)) \rightarrow r(t) = r^+, \; K(t) = K^+. \qquad (25)$$

We note that the time and effort to execute the above procedure can be reduced systematically by employing various heuristics.

8 Cost Matrices

As described in the previous section, the connectivity graph matrix is comprised of ones and zeros, and thus, graph search that uses it as a cost matrix results in a minimum length path between desired $r_i, r_j \in \mathcal{N}$.

In order to produce time efficient and thrust efficient paths, offline we simulate transitions between all $r_i, r_j \in \mathcal{N}$ for each $K \in \mathcal{K}$ and record the time and fuel consumption to reach a box with side of 1m around the target vertex. The results are merged into time and fuel matrices that store the respective minimum value, while in parallel, the control selectivity matrix identifies which gain K produced said minimum.

Step 3 in the motion planning procedure is augmented so that the graph connectivity matrix is multiplied elementwise with a desired cost matrix. Vertices that are not connected retain a corresponding matrix element of zero, while vertices that are connected now contain a matrix element of time or fuel cost.

9 Simulation Results

Simulations are now provided that illustrate the debris avoidance approach. We consider a nominal circular orbit of 850 km and discretize the HCW equations with a sampling period, ΔT, of 30 seconds. We construct an approximately 2 km cubed virtual net. We let $\mathcal{K} = \{K_1, K_2, K_3\}$, where K_1, K_2, K_3 are the LQ gains associated with state and control weight matrices $Q = \text{diag}(100, 100, 100, 10^7, 10^7, 10^7)$, and $R_1 = 2 \times 10^5 I_3$, $R_2 = 2 \times 10^7 I_3$, and $R_3 = 2 \times 10^9 I_3$. These gains are chosen to represent preferences for fuel considerations, maneuver time considerations, and a compromise between them. We impose a maximum thrust constraint of 10 N in each axis. In all simulations, Dijkstra's algorithm is used to find the shortest cost path from initial node to final node.

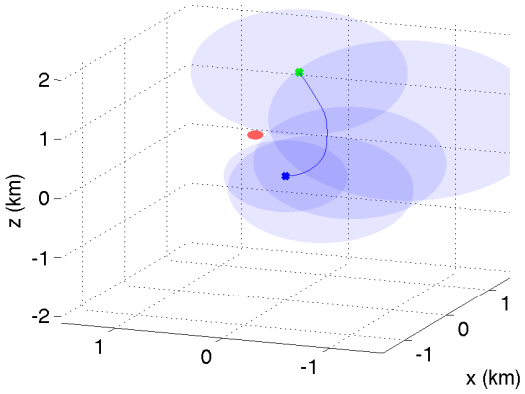
We consider an ellipsoidal set $O(z_1, Q_1)$ over-bounding a debris centered at $z_1 = [0.3\ 0.4\ 0.5]^{\mathrm{T}}$ km, where $Q_1 = 100 I_3$. The spacecraft's initial condition is $X(0) = X_e(r_0)$, where $r_0 = [0.32\ 0\ 1.61]^{\mathrm{T}}$ km. The target equilibrium node is $X_e(0)$.

Figure 4 shows the path the spacecraft takes under closed-loop control in order to avoid the debris and the time history of thrust magnitude. The spacecraft is able to complete the desired maneuver well within maximum thrust constraints while successfully avoiding the debris. In Figure 5 we rerun the simulation for a grid of initial conditions. The figure clearly demonstrates the initial conditions for which the maneuver path is perturbed from that which the spacecraft would have taken had there been no debris.

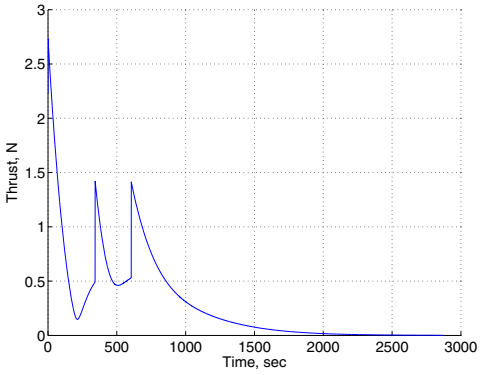
Next, we add a second debris $O(z_2, Q_2)$ centered at $z_2 = [0.3\ \ -0.4\ \ 0.5]^{\mathrm{T}}$, where $Q_2 = 100 I_3$. In calculating the growth distance, we take the minimum distance to each of $O(z_i, Q_i)$, $i = 1, 2$. Figure 6 shows the path the spacecraft takes under closed-loop control in order to avoid both debris and the time history of thrust magnitude.

Finally, we consider the case of a non-stationary debris where we cover its path by the union of static debris (11). A union of ellipsoidal sets over-bounds the debris' motion, where the debris positions z_i are generated by sampling the relative motion of the debris with the initial condition $[0\ 0.5\ 0\ 0\ 0.0006\ 0]^{\mathrm{T}}$, and where $Q_i = 200 I_3, i = 1 \ldots n_d$. The spacecraft's initial condition is $X(0) = X_e(r_0)$, where $r_0 = [0\ 1\ 0]^{\mathrm{T}}$ km. The target equilibrium node is $X_e(r_d)$, where $r_d = [0\ \ -1\ 0]^{\mathrm{T}}$ km. We use the single gain K_2 and do not include fuel or time cost matrices in the simulation, searching for a minimum length path. Figure 7 demonstrates that the spacecraft is able to avoid the closed debris path by 'hopping' under it.

In Figure 8, we repeat the simulation for time efficient and thrust efficient paths and allow all $K \in \mathcal{K}$. Table 1 summarizes the total time, thrust and nodes



(a) Debris Avoidance Path.



(b) Time history of thrust magnitude.

Fig. 4. (a) Debris avoidance path for a single debris. The green x marks the initial node. The blue x marks the final node. The red ellipsoid represents the debris. The blue line is the path the spacecraft takes in order to avoid the debris. The blue ellipsoids represent the safe positively invariant sets along the path. (b) The time history of thrust magnitude.

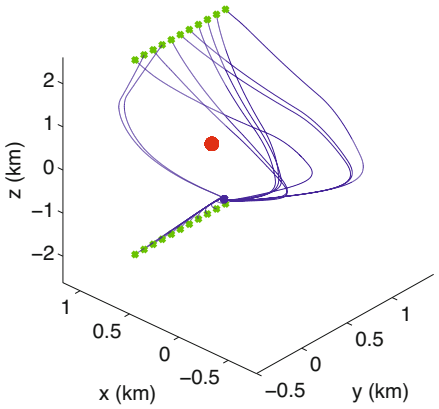
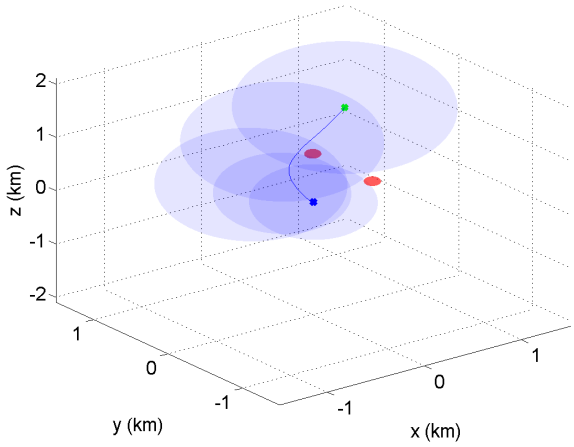


Fig. 5. Debris avoidance paths for many initial conditions. Each green x marks an intial condition. The blue x marks the final node. The red ellipsoid represents the debris. The blue lines are the paths that the spacecraft takes from each initial condition in order to avoid the debris. We do not show the invariant set ellipsoids for visual clarity.

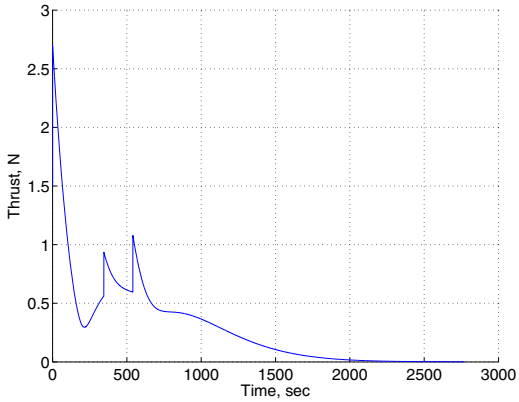
traversed for the three paths. Note that the minimum length path now 'hops' over the debris path instead of under it, as now that it has access to K_1 it finds a shorter path. Also note that the time efficient path takes longer to complete than the minimum length path. While the cost matrices described in Section 8 calculate time and thrust to travel between all vertices in the virtual net, the execution of the path does not require the spacecraft to reach intermediate vertices, rather, switching to the next reference once the current state enters the next reference's invariant set (25). As such, the cost matrices only provide a heuristic for selecting efficient paths. In Figure 9 we require the paths to travel through intermediate vertices to show that, in this case, the cost matrices accurately determine efficient paths. The results are summarized in Table 2.

Table 1. Total Time, Thrust, and Nodes Traversed for all Maneuver Paths for a Union of Static Debris

	Total Time	Total Thrust	Total # of Nodes	Gains used
Minimum Length Path	2611.5 s	1472.85 N·s	6	K_1
Time Efficient Path	2841 s	1264.95 N·s	6	K_1, K_2
Thrust Efficient Path	9177 s	671.297 N·s	11	K_2, K_3

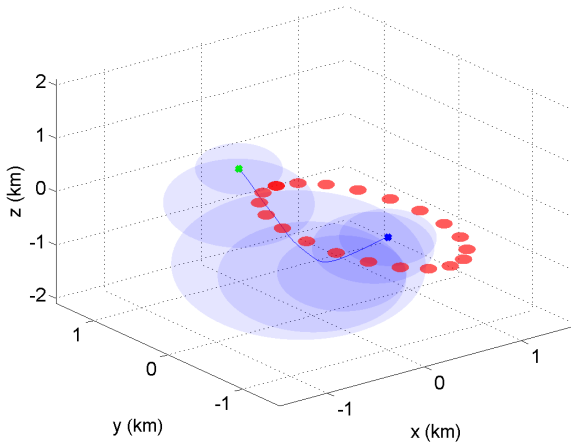


(a) Debris Avoidance Path.

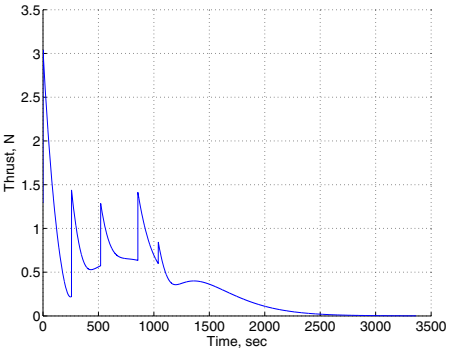


(b) Time history of thrust magnitude.

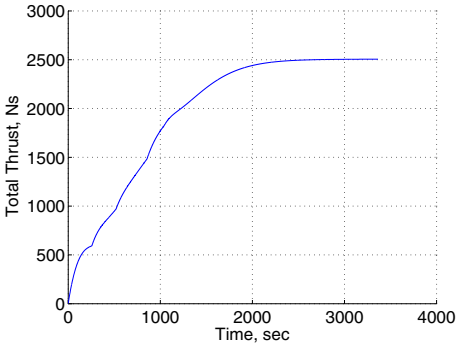
Fig. 6. (a) Debris avoidance path for 2 pieces of debris. The green x marks the initial node. The blue x marks the final node. The red ellipsoids represents the 2 pieces of debris. The blue line is the path the spacecraft takes in order to avoid the debris. The blue ellipsoids represent the invariant sets along the path. (b) The time history of thrust magnitude.



(a) Debris Avoidance Path.

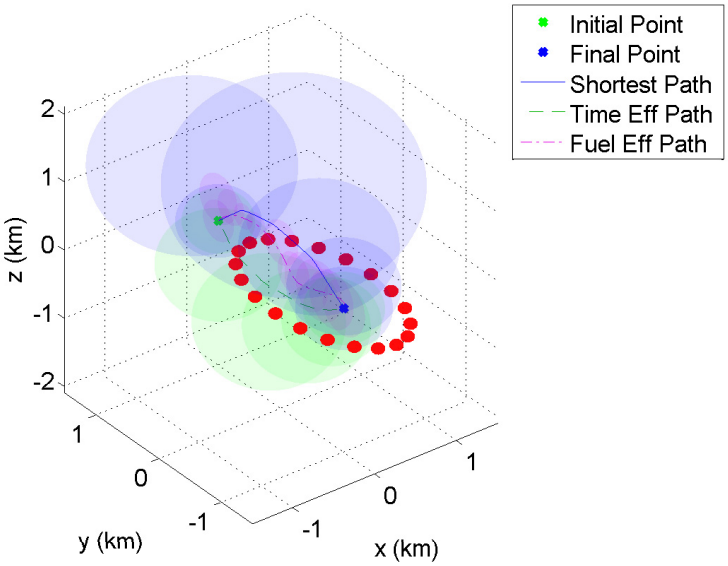


(b) Time history of thrust magnitude.

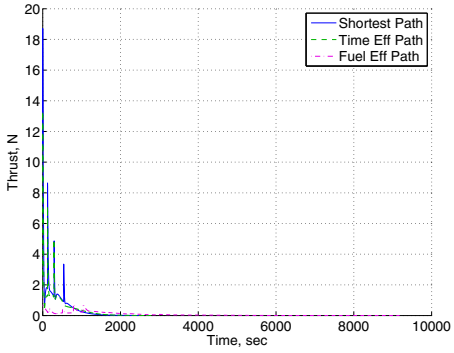


(c) Cumulative Thrust.

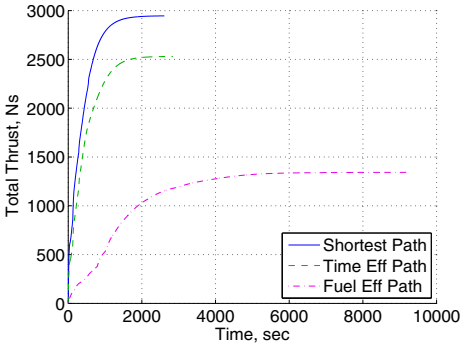
Fig. 7. (a) Debris avoidance path for a non-stationary debris using the union method. The green x marks the initial node. The blue x marks the final node. The red ellipsoids represent the debris path. The blue line is the path the spacecraft takes in order to avoid the debris. The blue ellipsoids represent the maximally grown invariant sets, C, along the path. (b) The time history of thrust magnitude. (c) Cumulative thrust vs time.



(a) Debris Avoidance Path.

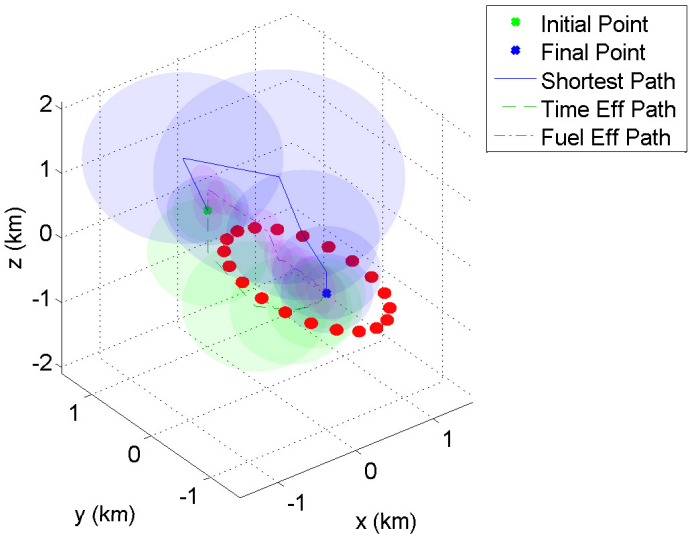


(b) Time history of thrust magnitude.

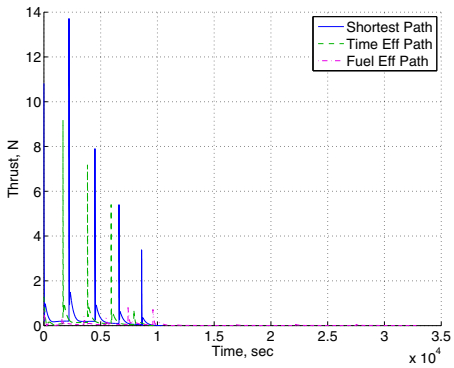


(c) Cumulative Thrust.

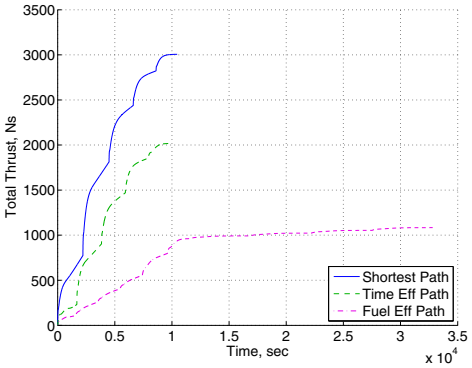
Fig. 8. (a) Multiple debris avoidance paths for a non-stationary debris using the union method. (b) The time history of thrust magnitude. (c) Cumulative thrust vs time.



(a) Debris Avoidance Path.



(b) Time history of thrust magnitude.



(c) Cumulative Thrust.

Fig. 9. (a) Multiple debris avoidance paths that travel through intermediate nodes for a non-stationary debris using the union method. (b) The time history of thrust magnitude. (c) Cumulative thrust vs time.

Table 2. Total Time, Thrust, and Nodes Traversed for all Maneuver Paths that Travel Through Intermediate Nodes for a Union of Static Debris

	Total Time	Total Thrust	Total # of Nodes	Gains used
Minimum Length Path	10457.5 s	3006.13 N·s	6	K_1
Time Efficient Path	9862 s	2017.11 N·s	6	K_1, K_2
Thrust Efficient Path	32812.5 s	1083.58 N·s	11	K_2, K_3

10 Conclusion

We described a technique for spacecraft relative motion maneuvering that uses positively-invariant sets in order to avoid collisions with debris, obstacles or moving debris paths, while adhering to specified thrust limits. The approach is based on maintaining the spacecraft trajectory within a tube formed by safe positively-invariant sets around forced and unforced equilibria in an appropriately constructed virtual net. The maneuver construction procedure exploits fast growth distance computations to determine equilibria connectivity; graph search based on optimal cost matrices is then employed to obtain a sequence of equilibria (essentially waypoints) to be commanded to the spacecraft. The switch to the next commanded equilibrium occurs immediately once appropriate conditions are satisfied and without waiting to settle to an equilibrium. For the case where thrust limits can be specified as polyhedral norm bounds, we have shown that the thrust limit on the growth distance can be easily computed; it is, in fact, feasible to perform these computations onboard a spacecraft in order to account for thruster failure or degradation.

References

1. Committee for the Assessment of NASA's Orbital Debris Programs; National Research Council, editor, Limiting Future Collision Risk to Spacecraft: An Assessment of NASA's Meteoroid and Orbital Debris Programs. The National Academies Press (2011)
2. LaValle, S.M.: Planning Algorithms. Cambridge University Press (1996)
3. Maia, M.H., Galvao, R.K.H.: On the use of mixed-integer linear predictive control with avoidance constraints. International Journal of Robust and Nonlinear Control 19, 822–828 (2008)
4. Baldwin, M., Weiss, A., Kolmanovsky, I., Erwin, R.S.: Spacecraft Debris Avoidance using Constraint Admissible Positively Invariant Sets. In: Proceedings of AAS Space Flight Mechanics Meeting, Charleston, South Carolina, pp. Paper AAS 12–250 (2012)
5. Kolmanovsky, I., Gilbert, E.: Theory and computation of disturbance invariant sets for discrete-time linear systems. Mathematical Problems in Engineering 4(4), 317–367 (1998)
6. Epenoy, R.: Fuel optimization for continuous-thrust orbital rendezvous with collision avoidance constraint. Journal of Guidance, Control, and Dynamics 34(2) (2011)
7. Sanchez-Ortiz, N., Bello-Mora, M., Klinkrad, H.: Collision avoidance manoeuvres during spacecraft mission lifetime: Risk reduction and required Δ V. Advances in Space Research 38 (2006)
8. Martinson, N., Munoz, J., Wiens, G.: A new method of guidance control for autonomous rendezvous in a cluttered space environment. In: Proceedings of AIAA Guidance, Navigation, and Control Conference, Hilton Head, SC (August 2007)
9. Martinson, N.: Obstacle avoidance guidance and control algorithms for spacecraft maneuvers. In: Proceedings of AIAA Guidance, Navigation, and Control Conference, Chicago, IL (August 2009)

10. Mueller, J.B., Griesemer, P.R., Thomas, S.: Avoidance maneuver planning incorporating station-keeping constraints and automatic relaxation. In: AIAA Infotech@Aerospace 2010, Atlanta, GA (April 2010)
11. Ulybyshev, Y.: Trajectory optimization for spacecraft proximity operations with constraints. In: AIAA Guidance, Navigation, and Control Conference, Portland, OR (August 2011)
12. Richards, A., Schouwenaars, T., How, J., Feron, E.: Spacecraft trajectory planning with avoidance constraints using mixed-integer linear programming. AIAA Journal of Guidance, Control, and Dynamics 25(4) (2002)
13. Mueller, J.B.: Onboard planning of collision avoidance maneuvers using robust optimization. In: AIAA Infotech@ Aerospace Conference, Seattle, AIAA, Washington, DC, Seattle, WA (April 2009)
14. Clohessy, W., Wiltshire, R.: Terminal guidance system for satellite rendezvous. J. Aerospace Sci. 27(9), 653–658 (1960)
15. Kolmanovsky, I., Gilbert, E.: Multimode regulators for systems with state and control constraints and disturbance inputs. In: Stephen Morse, A. (ed.) Control Using Logic-Based Switching. LNCIS, vol. 222, pp. 104–117. Springer, Heidelberg (1997)
16. Park, H., Di Cairano, S., Kolmanovsky, I.: Linear Quadratic Model Predictive Control approach to spacecraft rendezvous and docking. In: Proceedings of 21st AAS/AIAA Space Flight Mechanics Meeting, New Orleans, Louisiana, USA, pp. Paper AAS–142 (2011)
17. Ong, C.J., Gilbert, E.G.: Growth distances: New measures for object separation and penetration. IEEE Transactions on Robotics and Automation 12(6), 888–903 (1996)
18. Luenberger, D.G., Ye, Y.: Linear and nonlinear programming, vol. 116. Springer (2008)
19. Bazaraa, M.S., Sherali, H.D., Shetty, C.M.: Nonlinear programming: theory and algorithms. Wiley-interscience (2006)
20. Guddat, J., Vazquez, F.G., Jongen, H.T.: Parametric optimization: singularities, pathfollowing and jumps. Teubner (1990)
21. Quoc, T.D., Savorgnan, C., Diehl, M.: Adjoint-based predictor-corrector sequential convex programming for parametric nonlinear optimization. SIAM Journal on Optimization 22(4), 1258–1284 (2012)

A Comparison of Nonlinear PI and PID Inertia-Free Spacecraft Attitude Control Laws

Marc Camblor[2], Avishai Weiss[1], Gerardo Cruz[1],
Yousaf Rahman[1], Sergio Esteban[2],
Ilya V. Kolmanovsky[1], and Dennis S. Bernstein[1]

[1] Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109-2140, USA
{avishai,gecruz,yousafrm,ilya,dsbaero}@umich.com
[2] Department of Aerospace Engineering, Universidad de Sevilla, 41092 Sevilla, Spain
marcamchi@alumn.us.es, sesteban@us.es

Abstract. We compare four spacecraft attitude control laws that require no prior modeling of the spacecraft mass distribution. All four control laws are based on rotation matrices, which provide a singularity-free attitude representation and unwinding-free operation without discontinuous switching. We apply these control laws to motion-to-rest and motion-to-spin maneuvers. Simulation results are given to illustrate the robustness of the control laws to uncertainty in the spacecraft inertia. For motion-to-rest maneuvers about a principal axis with bounded torque, we compare the settling time of the inertia-free control laws with the time-optimal bang-bang control law operating under known inertia. We also investigate closed-loop performance in the presence of attitude-dependent torque disturbances, actuator nonlinearities, sensor noise, and actuator bias.

Keywords: Attitude control, unmodeled inertia, rotation matrix, SO(3).

1 Introduction

The development of a spacecraft attitude control system is often a labor-intensive process due to the need for an accurate model of the spacecraft inertia. Determining and predicting the mass properties of a spacecraft may be difficult due to fuel usage, deployment, structural articulation, and docking. To alleviate this need, this paper focuses on spacecraft attitude control laws that require no modeling of the spacecraft's mass distribution. An adaptive inertia-free attitude control law is given in [1] for minimum-time maneuvers. Inertia-free control laws for motion-to-rest and tracking are given in [2–4].

Attitude control laws can use various parameterizations of the rotation group SO(3). Euler angles are conceptually the simplest, but cannot represent all angular velocities due to singularities corresponding to gimbal lock. A related obstacle arises in the use of Rodrigues parameters and modified Rodrigues parameters,

D. Choukroun et al. (eds.), *Advances in Estimation, Navigation, and Spacecraft Control*,
DOI: 10.1007/978-3-662-44785-7_27

which have singularities at 180-deg and 360-deg rotation angles, respectively. The most common attitude representation is based on quaternions, which can represent all attitudes and all angular velocities, but provide a double cover of SO(3), that is, each physical attitude is represented by two elements of the 4-dimensional sphere S^3. A continuous controller designed on the set of quaternions can thus inadvertently command the spacecraft to needlessly rotate 360 degrees to reach the commanded attitude. This is the unwinding problem [5]. The inertia-free, quaternion-based control laws in [3, 6, 7] exhibit unwinding.

There are several approaches to avoiding unwinding. The traditional approach is to implement a logic statement that confines the quaternions to a hemisphere of S^3 [8]. This approach introduces a discontinuous control law, which can lead to chattering in the presence of noise. This issue and associated complications are addressed in [9].

In the present paper we avoid unwinding by representing attitude in terms of rotation matrices, which constitute a one-to-one representation of physical attitude without attitude or angular-velocity singularities [10]. Attitude control on SO(3) thus provides the ability to implement continuous control laws that do not exhibit unwinding [11–13]. Inertia-free control laws on SO(3) are developed in [14, 15]. Lie groups are used for control in [16].

Since SO(3) is a compact manifold, every continuous vector field on it necessarily possesses more than one equilibrium, in fact, at least four. This means that global convergence on SO(3) under continuous, time-invariant control is impossible. Consequently, the objective of [12, 13, 15] is almost global stabilization, where the spurious equilibria are saddle points. Although the spurious equilibria can slow the rate of convergence, this approach avoids the complications of discontinuous control laws.

Although the derivation of the inertia-free controller in [15] and the present paper is based on rotation matrices, the attitude error given by the S-parameter defined by (7) can be computed from any attitude parameterization, such as quaternions or modified Rodrigues parameters, and thus these results are not confined to rotation matrices per se.

The goal of this paper is to compare four continuous, inertia-free attitude control laws based on rotation matrices. These control laws are called SO(3)/0, SO(3)/3, SO(3)/6, and SO(3)/9, where the last number represents the number of integrators in the control law. These control laws take the form of nonlinear PD/PID control laws tailored to the nonlinear characteristics of spacecraft dynamics. Since linearized rigid-body dynamics comprise a double integrator about each principal axis, we expect (as in the case of linear systems) that asymptotic tracking of attitude ramp commands (that is, spin commands) about each principal axis is possible without integral action. The primary role of integral control in spacecraft attitude dynamics is thus to reject constant disturbances.

In the simplest case of PD control, the inertia-free SO(3)/0 control law is given in [14]. In contrast, the SO(3)/9 control given in [15] is also inertia free but employs three integrators inside the feedback loop as well as six integrators for inertia estimation. The control laws SO(3)/3 and SO(3)/6 are ad hoc

simplifications of SO(3)/9. The goal of this paper is to numerically investigate and compare the closed-loop performance of these control laws under various command-following and disturbance-rejection scenarios as well as under various off-nominal conditions involving rate-sensor noise and unmodeled actuator nonlinearities, such as saturation, on-off, and deadzone.

We consider two basic scenarios, namely, motion-to-rest (M2R) maneuvers and motion-to-spin (M2S) maneuvers, where "rest" and "spin" refer to motion relative to an inertial frame. If the M2R and M2S maneuvers begin from zero angular velocity, then we use the terminology rest-to-rest (R2R) and rest-to-spin (R2S), respectively. A M2S maneuver aims to bring the spacecraft from an arbitrary initial angular velocity and attitude to a specified constant angular velocity relative to an inertial frame. In other words, the goal is to have the spacecraft rotate at a constant rate about a body-fixed axis whose inertial direction is fixed.

Although the spacecraft inertia is unknown, and thus the directions of the principal axes of inertia are unknown, we consider commanded spins about both principal and non-principal axes (without knowing whether the commanded axis of rotation is principal or non-principal) in order to demonstrate how these control laws perform in various scenarios. For example, a commanded spin about a principal axis has the advantage that, once the spacecraft reaches the commanded spin, no additional torque is needed in the absence of disturbances except possibly to stabilize a spin about the minimum and intermediate axes, where the latter is naturally unstable and the former is unstable due to energy dissipation, although we do not model this effect. Furthermore, as shown in [17, p. 377], a spin about a non-principal axis with constant torque and for which all components of the angular velocity are nonzero is unstable and thus stabilization is required. Finally, a commanded spin about a non-principal axis requires constant, nonzero torques and thus is more sensitive to torque saturation than a commanded spin about a principal axis. In summary, a commanded spin about a non-principal axis places significantly higher demands on the control law in terms of stabilization and control authority.

Throughout this paper, all control torques are assumed to be provided by thrusters or gas jets without onboard stored momentum.

2 Spacecraft Model

The spacecraft equations of motion are given by Euler's and Poisson's equations

$$J\dot{\omega} = (J\omega) \times \omega + Bu + z_{\text{dist}}, \tag{1}$$

$$\dot{R} = R\omega^{\times}, \tag{2}$$

where $\omega \in \mathbb{R}^3$ is the angular velocity of the spacecraft frame relative to the inertial frame resolved in the spacecraft frame, $\omega^{\times}$ is the cross-product matrix of ω, $J \in \mathbb{R}^{3\times3}$ is the inertia matrix of the spacecraft, the components of the vector $u \in \mathbb{R}^3$ represent three independent torque inputs, and the nonsingular matrix $B \in \mathbb{R}^{3\times3}$ determines the applied torque about each axis of the spacecraft

frame due to u. The rotation matrix $R = \mathcal{O}_{\mathrm{In/SC}} \in \mathbb{R}^{3\times 3}$ is the physical rotation matrix that transforms the inertial frame into the spacecraft frame resolved in the spacecraft frame, and where $\mathcal{O}_{\mathrm{In/SC}}$ is the orientation (direction cosine) matrix that transforms components of a vector resolved in the spacecraft frame into the components of the same vector resolved in the inertial frame. The vector $z_{\mathrm{dist}} \in \mathbb{R}^3$ represents disturbance torques, such as the gravity gradient torques modeled by (25) below.

The objective of the attitude control problem is to determine control inputs such that the spacecraft attitude given by R follows a commanded attitude trajectory given by the possibly time-varying C^1 rotation matrix $R_{\mathrm{d}}(t)$. For $t \geq 0$, $R_{\mathrm{d}}(t)$ is given by

$$\dot{R}_{\mathrm{d}}(t) = R_{\mathrm{d}}(t)\omega_{\mathrm{d}}(t)^{\times}, \quad R_{\mathrm{d}}(0) = R_{\mathrm{d}0}, \tag{3}$$

where ω_{d} is the commanded possibly time-varying angular velocity vector resolved in the desired body frame specified by $R_{\mathrm{d}}(t)$. The error between $R(t)$ and $R_{\mathrm{d}}(t)$ is given by the attitude-error rotation matrix $\tilde{R} \triangleq R_{\mathrm{d}}^{\mathrm{T}} R$, which satisfies the differential equation $\dot{\tilde{R}} = \tilde{R}\tilde{\omega}^{\times}$, where the angular-velocity error $\tilde{\omega}$ is defined by $\tilde{\omega} \triangleq \omega - \tilde{R}^{\mathrm{T}}\omega_{\mathrm{d}}$. We rewrite (1) in terms of $\tilde{\omega}$ as

$$J\dot{\tilde{\omega}} = J(\tilde{\omega} + \tilde{R}^{\mathrm{T}}\omega_{\mathrm{d}}) \times (\tilde{\omega} + \tilde{R}^{\mathrm{T}}\omega_{\mathrm{d}}) + J(\tilde{\omega} \times \tilde{R}^{\mathrm{T}}\omega_{\mathrm{d}} - \tilde{R}^{\mathrm{T}}\dot{\omega}_{\mathrm{d}}) + Bu + z_{\mathrm{dist}}. \tag{4}$$

A scalar measure of attitude error is given by the eigenaxis error

$$e(t) \triangleq \cos^{-1}(\tfrac{1}{2}[\mathrm{tr}\,\tilde{R}(t) - 1]).$$

3 Control Laws

3.1 SO(3)/9

To estimate the spacecraft inertia, we introduce the notation $J\omega = L(\omega)\gamma$, where $\gamma \in \mathbb{R}^6$ is defined by

$$\gamma \triangleq \begin{bmatrix} J_{11} & J_{22} & J_{33} & J_{23} & J_{13} & J_{12} \end{bmatrix}^{\mathrm{T}} \tag{5}$$

and

$$L(\omega) \triangleq \begin{bmatrix} \omega_1 & 0 & 0 & 0 & \omega_3 & \omega_2 \\ 0 & \omega_2 & 0 & \omega_3 & 0 & \omega_1 \\ 0 & 0 & \omega_3 & \omega_2 & \omega_1 & 0 \end{bmatrix}.$$

Next, let $\hat{J} \in \mathbb{R}^{3\times 3}$ denote an estimate of J, and define the inertia-estimation error $\tilde{J} \triangleq J - \hat{J}$. Letting $\hat{\gamma}, \tilde{\gamma} \in \mathbb{R}^6$ represent $\hat{J}, \tilde{J}$, respectively, as in (5), it follows that $\tilde{\gamma} = \gamma - \hat{\gamma}$. Likewise, let $\hat{z}_{\mathrm{dist}} \in \mathbb{R}^3$ denote an estimate of z_{dist}, and define the disturbance-estimation error $\tilde{z}_{\mathrm{dist}} \triangleq z_{\mathrm{dist}} - \hat{z}_{\mathrm{dist}}$.

Assuming that the disturbance is harmonic, z_{dist} can be modeled by

$$\dot{d} = A_{\text{dist}} d, \quad z_{\text{dist}} = C_{\text{dist}} d, \tag{6}$$

where $A_{\text{dist}} \in \mathbb{R}^{n_d \times n_d}$ and $C_{\text{dist}} \in \mathbb{R}^{3 \times n_d}$ are known matrices. In this model, $d(0)$ is unknown, which is equivalent to the assumption that the amplitude and phase of all harmonic components in the disturbance are unknown; however, the spectrum of d is assumed to be known. To provide asymptotic rejection of harmonic disturbances, the matrix A_{dist} is chosen to include eigenvalues of all frequency components that may be present in z_{dist}, where the zero eigenvalue corresponds to a constant disturbance. Since z_{dist} is harmonic, A_{dist} is chosen to be skew symmetric. Let $\hat{d} \in \mathbb{R}^{n_d}$ denote an estimate of d, and define the disturbance-state estimation error $\tilde{d} \triangleq d - \hat{d}$.

The role of $\text{tr}(A - A\tilde{R})$ in the stability analysis below is explained by the following result.

Lemma 1. [15] Let $A \in \mathbb{R}^{3\times3}$ be a diagonal positive-definite matrix, and let R be a rotation matrix. Then, the following statements hold:

i) For all $i, j = 1, 2, 3$, $R_{ij} \in [-1, 1]$.
ii) $\text{tr}(A - AR) \geq 0$.
iii) $\text{tr}(A - AR) = 0$ if and only if $R = I$.

The attitude error S is defined by [11, 13–15]

$$S \triangleq \sum_{i=1}^{3} a_i (\tilde{R}^{\text{T}} e_i) \times e_i, \tag{7}$$

where a_1, a_2, a_3 are distinct positive numbers and $e_1, e_2, e_3 \in \mathbb{R}^3$ are the standard basis vectors.

Theorem 1. [15] Let K_{p} be a positive number, let $K_1 \in \mathbb{R}^{3\times3}$, $Q \in \mathbb{R}^{6\times6}$, and $D \in \mathbb{R}^{n_d \times n_d}$ be positive definite, let $A = \text{diag}(a_1, a_2, a_3)$ be a diagonal positive-definite matrix, and define the attitude error S by (7). Then the Lyapunov candidate

$$V(\tilde{\omega}, \tilde{R}, \tilde{\gamma}, \tilde{d}) \triangleq \tfrac{1}{2}(\tilde{\omega} + K_1 S)^{\text{T}} J (\tilde{\omega} + K_1 S) + K_{\text{p}} \text{tr}\,(A - A\tilde{R}) + \tfrac{1}{2}\tilde{\gamma}^{\text{T}} Q \tilde{\gamma} + \tfrac{1}{2}\tilde{d}^{\text{T}} D \tilde{d}$$

is positive definite, that is, V is nonnegative, and $V = 0$ if and only if $\tilde{\omega} = 0$, $\tilde{R} = I$, $\tilde{\gamma} = 0$, and $\tilde{d} = 0$.

Theorem 2. [15] Let K_{p} be a positive number, let $K_{\text{v}} \in \mathbb{R}^{3\times3}$, $K_1 \in \mathbb{R}^{3\times3}$, $Q \in \mathbb{R}^{6\times6}$, and $D \in \mathbb{R}^{n_d \times n_d}$ be positive definite, assume that $A_{\text{dist}}^{\text{T}} D + D A_{\text{dist}}$ is negative semidefinite, let $A = \text{diag}(a_1, a_2, a_3)$ be a diagonal positive-definite matrix with distinct diagonal entries, define S and V as in Theorem 1, and let $\hat{\gamma}$ and $\hat{d}$ satisfy

$$\dot{\hat{\gamma}} = Q^{-1}[L^{\text{T}}(\omega)\omega^{\times} + L^{\text{T}}(K_1 \dot{S} + \tilde{\omega} \times \omega - \tilde{R}^{\text{T}} \dot{\omega}_{\text{d}})](\tilde{\omega} + K_1 S), \tag{8}$$

where

$$\dot{S} = \sum_{i=1}^{3} a_i[(\tilde{R}^{\mathrm{T}} e_i) \times \tilde{\omega}] \times e_i, \tag{9}$$

and

$$\dot{\hat{d}} = A_{\mathrm{dist}}\hat{d} + D^{-1}C_{\mathrm{dist}}^{\mathrm{T}}(\tilde{\omega} + K_1 S), \quad \hat{z}_{\mathrm{dist}} = C_{\mathrm{dist}}\hat{d}. \tag{10}$$

Furthermore, let

$$u = B^{-1}(v_1 + v_2 + v_3), \tag{11}$$

where

$$v_1 \stackrel{\triangle}{=} -(\hat{J}\omega) \times \omega - \hat{J}(K_1\dot{S} + \tilde{\omega} \times \omega - \tilde{R}^{\mathrm{T}}\dot{\omega}_{\mathrm{d}}), \tag{12}$$

$$v_2 \stackrel{\triangle}{=} -\hat{z}_{\mathrm{dist}}, \quad v_3 \stackrel{\triangle}{=} -K_{\mathrm{p}}S - K_{\mathrm{v}}(\tilde{\omega} + K_1 S). \tag{13}$$

Then,

$$\begin{aligned}\dot{V}(\tilde{\omega}, \tilde{R}, \tilde{\gamma}, \tilde{d}) = &- (\tilde{\omega} + K_1 S)^{\mathrm{T}} K_{\mathrm{v}}(\tilde{\omega} + K_1 S) - K_{\mathrm{p}} S^{\mathrm{T}} K_1 S \\ &+ \tfrac{1}{2}\tilde{d}^{\mathrm{T}}(A_{\mathrm{dist}}^{\mathrm{T}} D + D A_{\mathrm{dist}})\tilde{d}\end{aligned} \tag{14}$$

is negative semidefinite. Furthermore, the equilibrium manifold $(\tilde{\omega}, \tilde{R}, (\tilde{\gamma}, \tilde{d})) = (0, I, \mathcal{Q}_0)$ of the closed-loop system given by (4) and (8)-(13) is locally asymptotically stable, and the remaining equilibrium manifolds given by $(0, \mathcal{R}_i, \mathcal{Q}_i)$, for $i \in \{1, 2, 3\}$ are unstable. Finally, the set of all initial conditions converging to these equilibrium manifolds forms a lower dimensional submanifold of $\mathbb{R}^3 \times \mathrm{SO}(3) \times \mathbb{R}^6 \times \mathbb{R}^3$.

Saturation techniques for SO(3)/9 are discussed in [18].

3.2 SO(3)/6

The control law SO(3)/6 is a simplification of the SO(3)/9 control law (11)–(13) with v_2 and thus (10) omitted. In particular, this control law has the form

$$u = -B^{-1}[(K_{\mathrm{p}}I + K_{\mathrm{v}}K_1)S + K_{\mathrm{v}}\tilde{\omega} + \hat{J}K_1\dot{S} + (\hat{J}\omega) \times \omega + \hat{J}(\tilde{\omega} \times \omega) - \hat{J}\tilde{R}^{\mathrm{T}}\dot{\omega}_{\mathrm{d}}]. \tag{15}$$

3.3 SO(3)/3

The control law SO(3)/3 is a simplification of the SO(3)/9 control law (11)–(13) with the inertia estimate (8) omitted and with $A_{\mathrm{dist}} = 0$. In particular, this control law has the form

$$u = -B^{-1}\left[(K_{\mathrm{p}}I + K_{\mathrm{v}}K_1)S + K_{\mathrm{i}}C_{\mathrm{dist}}D^{-1}C_{\mathrm{dist}}^{\mathrm{T}}\int_0^t [\tilde{\omega}(s) + K_1 S(s)]\,\mathrm{d}s + K_{\mathrm{v}}\tilde{\omega}\right], \tag{16}$$

where the integral gain K_{i} is a positive number.

3.4 SO(3)/0

The SO(3)/0 control law for almost global stabilization [14, 15] is given by

$$u = -B^{-1}(K_{\mathrm{p}}S + K_{\mathrm{v}}\tilde{\omega}), \tag{17}$$

where the positive number K_{p} and the positive-definite matrix $K_{\mathrm{v}} \in \mathbb{R}^{3\times 3}$ are proportional (attitude) and derivative (angular velocity) gains, respectively.

Note that the control law (17) is inertia-free. The stabilizing effect of this control law on the attitude of a rigid spacecraft follows from the Lyapunov function

$$V(\omega, \tilde{R}) \triangleq \tfrac{1}{2}\omega^{\mathrm{T}}J\omega + K_{\mathrm{p}}\mathrm{tr}(A - A\tilde{R}), \tag{18}$$

where $A \triangleq \mathrm{diag}(a_1, a_2, a_3)$ and for which $\dot{V}(\omega, \tilde{R}) = -\omega^{\mathrm{T}}K_{\mathrm{v}}\omega$. The invariant set theorem is used in [14] to ensure almost global asymptotic stability.

By choosing K_{v} to be a function of ω, the control law (17) satisfies the following saturation bounds [15, 18].

Proposition 1. Let α and β be positive numbers, let $A = \mathrm{diag}(a_1, a_2, a_3)$ have distinct positive diagonal entries, and let K_{p} and $K_{\mathrm{v}}(\omega)$ be given by

$$K_{\mathrm{p}} = \frac{\alpha}{\mathrm{tr}\, A} \tag{19}$$

and

$$K_{\mathrm{v}}(\omega) = \beta \begin{bmatrix} \frac{1}{1+|\omega_1|} & 0 & 0 \\ 0 & \frac{1}{1+|\omega_2|} & 0 \\ 0 & 0 & \frac{1}{1+|\omega_3|} \end{bmatrix}. \tag{20}$$

Then, for all $t \geq 0$, the control torque given by (17) satisfies

$$\|u(t)\|_\infty \leq \frac{\alpha + \beta}{\sigma_{\min}(B)}. \tag{21}$$

For the remainder of the paper, K_{p} and K_{v} are assumed to be given by (19) and (20). Alternative forms of the gain $K_{\mathrm{v}}(\omega)$ are given in [19].

4 Modeling Inertia Variations

If the inertia tensor is resolved in a non-principal body-fixed frame, then the diagonal entries of the resulting inertia matrix are the moments of inertia and the off-diagonal entries are the products of inertia. The off-diagonal entries of the inertia matrix are thus a consequence of an unknown rotation between a principal body-fixed frame and an arbitrarily chosen body-fixed frame.

Figure 1 shows the triangular region of feasible principal moments of inertia of a rigid body. There are five cases that are highlighted for the principal moments

of inertia $\lambda_1 \geq \lambda_2 \geq \lambda_3 > 0$, where $\lambda_1, \lambda_2, \lambda_3$ satisfy the triangle inequality $\lambda_1 < \lambda_2 + \lambda_3$. Let m denote the mass of the rigid body. The point $\lambda_1 = \lambda_2 = \lambda_3$ corresponds to a sphere of radius $\hat{r} = \sqrt{\frac{5\lambda_1}{2m}}$; the point $\lambda_1 = \lambda_2 = 2\lambda_3$ corresponds to a cylinder of length l and radius r, where $l = 3r$ and $r = \sqrt{\frac{2\lambda_1}{m}}$; and the point $\lambda_1 = \frac{6}{5}\lambda_2 = 2\lambda_3$ is a brick whose side lengths are l_1, l_2, l_2 and whose inertia is located at the centroid of the triangular region. The remaining two cases in Figure 1 are limiting cases. In particular, the thin disk is a cylinder with zero length, positive radius, and infinite density, while the thin cylinder is a cylinder with positive length, zero radius, and infinite density. Note that the inertia matrix of the thin disk is positive definite, whereas the inertia matrix of the thin cylinder is positive semidefinite but not positive definite. Table 1 summarizes the parameters and densities for each of these rigid bodies.

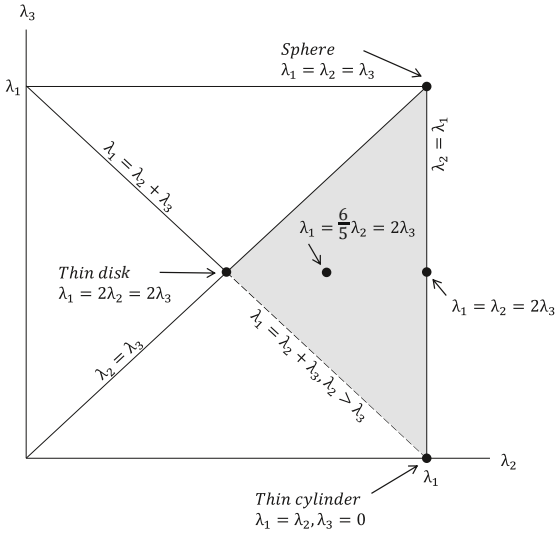


Fig. 1. Feasible region of the principal moments of inertia $\lambda_1, \lambda_2, \lambda_3$ of a rigid body satisfying $0 < \lambda_3 \leq \lambda_2 \leq \lambda_1$, where $\lambda_1 < \lambda_2 + \lambda_3$. The shaded region shows all feasible values of λ_2 and λ_3 in terms of the largest principal moment of inertia λ_1. The open dots and dashed line segment indicate nonphysical, limiting cases.

5 M2R Examples

For all of the examples in this section, we assume that the nominal body-fixed frame is a principal body-fixed frame. However, the body-fixed frame is not a principal-axis frame for the off-nominal cases considered below. The nominal spacecraft shape is chosen to be a brick corresponding to the centroid of the triangular region in Figure 1. For all cases considered, we choose $\lambda_1 = 10$ kg-m^2, which for the centroidal brick yields the inertia $J_3 = \text{diag}(10, 25/3, 5)$. Consequently, the inertias J_1, J_2, J_4, and J_5 of the sphere, cylinder, thin disk, and thin

Table 1. Parameters and densities for the inertia matrices considered in Figure 1. Note that the parameters for all shapes are set by the ratio λ_1/m, where λ_1 is the principal moment of inertia and m is the mass.

Shape	Parameters	Density
Sphere	$\hat{r} = \sqrt{\frac{5\lambda_1}{2m}}$	$\rho = \frac{4}{3}\pi m^{2/5}(\frac{5\lambda_1}{2})^{2/5}$
Cylinder	$l = 3r$, $r = \sqrt{\frac{2\lambda_1}{m}}$	$\rho = 3\pi m^{2/5}(2\lambda_1)^{2/5}$
Brick	$l_1 = \sqrt{\frac{2\lambda_1}{m}}$, $l_2 = \sqrt{\frac{4\lambda_1}{m}}$, $l_3 = \sqrt{\frac{8\lambda_1}{m}}$	$\rho = 8m^{2/5}\lambda_1^{2/5}$
Thin disk	$l = 0$, $r = \sqrt{\frac{2\lambda_1}{m}}$	$\rho = \infty$
Thin cylinder	$r = 0$, $l = \sqrt{\frac{12\lambda_1}{m}}$	$\rho = \infty$

cylinder are given, respectively, by $J_1 = \mathrm{diag}(10, 10, 10)$, $J_2 = \mathrm{diag}(10, 10, 5)$, $J_4 = \mathrm{diag}(10, 5, 5)$, and $J_5 = \mathrm{diag}(10, 10, 0.1)$, where all units are kg-m^2. The inertia matrix J_3 corresponding to the centroid of the inertia region serves as the nominal inertia matrix, while a perturbation $J(\alpha)$ of J_i in the direction of J_j has the form $J(\alpha) = (1-\alpha)J_i + \alpha J_j$, where $\alpha \in [0, 1]$ is the perturbation parameter. To facilitate numerical integration, J_5 is chosen to be a nonsingular approximation of the inertia of a thin cylinder.

For all examples in the remainder of the paper, let $\alpha = \beta = 1$, $K_1 = I_3$, $K_\mathrm{i} = 0.015$, $A = \mathrm{diag}(1, 2, 3)$, $\mathrm{B} = \mathrm{I}_3$, $\mathrm{C_{dist}} = \mathrm{I}_3$, $\mathrm{D} = \mathrm{I}_3$, and $\mathrm{Q} = \mathrm{I}_6$. Furthermore, K_p and K_v are defined in (19) and (20), respectively. To evaluate the performance for R2R examples, we use the settling-time metric

$$k_0 = \min_{k>100}\{k : \text{ for all } i \in \{1, \ldots, 100\}, e((k-i)T_\mathrm{s}) < 0.05\,\mathrm{rad}\}, \tag{22}$$

where k is the simulation step, T_s is the integration step size, and $e(kT_\mathrm{s})$ is the eigenaxis error at the kth simulation step. This metric is thus the minimal time such that the eigenaxis error in the 100 most recent simulation steps is less than 0.05 rad.

5.1 M2R Examples without Disturbances

To illustrate the inertia-free property of the control laws, the inertia of the spacecraft is varied using

$$J_{ij}(\alpha) = (1-\alpha)J_i + \alpha J_j, \tag{23}$$

where $\alpha \in [0, 1]$ for $i, j \in \{(1, 5), (3, 1), (3, 5), (3, 4)\}$.

Next, we examine the robustness of the thrusters to misalignment relative to the principal axes. To model this misalignment, the inertia matrix is rotated by an angle θ about either the x-axis, y-axis, or z-axis. For each rotation, J_3 is transformed by

$$J_3' = \mathcal{O}(\theta)J_3\mathcal{O}(\theta)^\mathrm{T}, \tag{24}$$

where $\mathcal{O}(\theta)$ is a direction cosine matrix.

Figure 2 shows how the thruster misalignment angle θ affects the settling time, where θ is varied from -180 deg to 180 deg. Figure 2 also shows how the R2R settling time depends on α. Both inertia robustness studies are shown for SO(3)/3 and SO(3)/9.

5.2 M2R Examples with Disturbances

Figure 3 illustrates how the control laws handle body-constant disturbance torques about the minor axis. Note that SO(3)/0 and SO(3)/6 are not able to reject constant torque disturbances.

Next, we consider a gravity gradient disturbance torque τ_{g} modeled by [17, pp. 386–390]

$$\tau_g = 3n^2(\mathcal{O}_{\mathrm{SC/L}}e_3)^{\times} J(\mathcal{O}_{\mathrm{SC/L}}e_3), \tag{25}$$

where $n \triangleq \sqrt{\mu/r^3}$ is the orbital mean motion, μ is the gravitational parameter, r is the orbit radius, e_3 is the third column of the 3×3 identity matrix, and $\mathcal{O}_{\mathrm{SC/L}} \in \mathbb{R}^{3\times 3}$ is the orientation matrix of the spacecraft frame $\mathrm{F_{SC}}$ relative to the local-vertical-local-horizontal frame $\mathrm{F_L}$. The satellite orbit is circular with an altitude of 300 km.

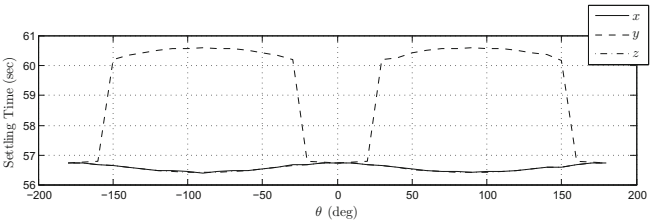
Figure 4 shows that SO(3)/3 and SO(3)/9 can reject gravity gradient disturbances for a R2R maneuver. Furthermore, Figure 5 shows the closed-loop performance of SO(3)/3 and SO(3)/9 for a commanded inertial attitude in the presence of a gravity gradient disturbance.

Next, we consider an inertially constant disturbance torque. Figure 6 shows the performance of all four controllers as the disturbance magnitude is increased. The settling time is computed for the control laws that can reject the inertially constant disturbance, whereas the steady-state error is computed for those that bring the spacecraft to rest with an incorrect attitude.

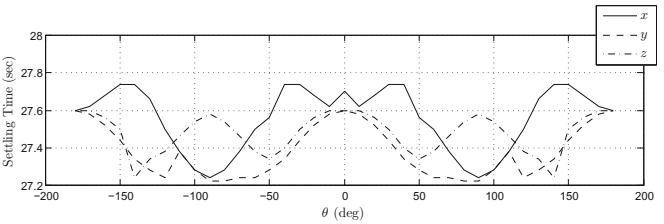
5.3 M2R Examples with Input Nonlinearities

Next, we consider the effect of three nonlinearities, namely, torque cut-off saturation, control-torque deadzone, and thrusters operating in on-off mode with the input torque given by the on-off control law $u(t) = u_{\max}\mathrm{sign}(v(t))$, where $v(t)$ is the torque commanded by the SO(3) control law. Figure 7 shows the effect of increasingly restrictive saturation levels for all of the control laws. Figure 8 shows how each controller performs the same M2R maneuver using on-off actuation. Figure 9 illustrates how the settling time changes as a function of the width of the unknown control-torque deadzone.

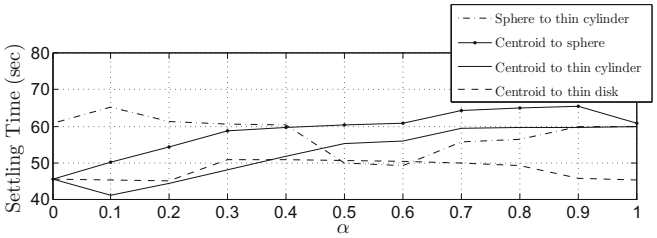
We also consider the effect of sensor noise corrupting the angular-velocity measurement. Two types of noise are considered, namely, gyro bias, that is, a constant error in the measurement of ω, as well as zero-mean white gyro noise with a signal-to-noise ratio of 20. Figure 10 shows the performance of all four controllers when either gyro bias or stochastic gyro noise is present. SO(3)/3, SO(3)/6, and SO(3)/9 are able to reduce the attitude error below 0.05 rad.



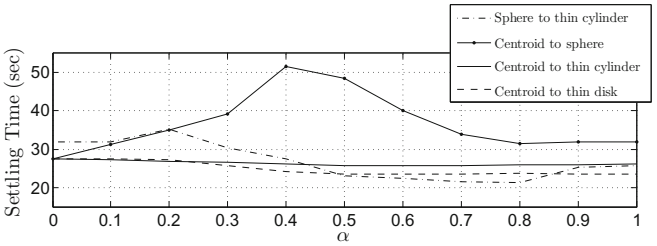
(a) SO(3)/3. Variations in the settling time are within 7%.



(b) SO(3)/9. Variations in the settling time are within 2%.

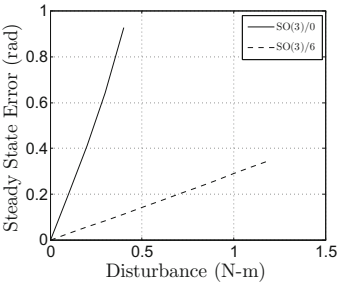


(c) SO(3)/3. Variations in the settling time are within 30%.

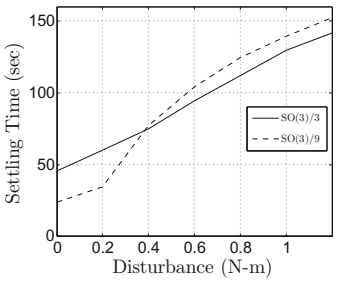


(d) SO(3)/9. Variations in the settling time are within 44%.

Fig. 2. R2R settling time with no disturbance for SO(3)/3 and SO(3)/9 as a function of the principal-frame/body-frame rotation angle θ for misalignments about each of the three principal axes of J_3 (a), (b), and the perturbation parameter α for various combinations of inertia matrices (c), (d). The commanded maneuver is a 40-deg rotation about the body-fixed direction $[1\ 1\ 1]^{\mathrm{T}}$. Each controller is implemented with a single tuning for all inertia cases. Convergence is achieved for all four cases.



(a) R2R steady-state error for SO(3)/0 and SO(3)/6.



(b) R2R settling time for SO(3)/3 and SO(3)/9.

Fig. 3. Body-constant torque-disturbance rejection about the minor axis. The commanded maneuver is a 40-deg rotation about the body-fixed direction $[1\ 1\ 1]^{\mathrm{T}}$. Note that the performance of SO(3)/6 is substantially better than the performance of SO(3)/0, and the performance of SO(3)/3 improves relative to SO(3)/9 as the magnitude of the torque disturbance increases.

However, SO(3)/0 is not able to achieve the commanded attitude in the presence of either gyro bias or zero-mean white measurement noise.

6 M2S Examples

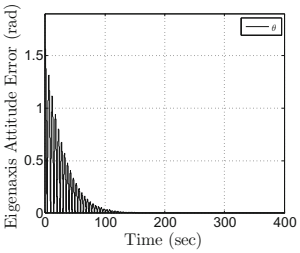
6.1 M2S Examples without Disturbances

Next, we consider M2S maneuvers. For spins about a principal axis, Euler's equation becomes a linear second-order system, and thus integrators in the controller are not required to stabilize spin commands. As shown in Figure 11, SO(3)/0 can stabilize spins about a principal axis. Figure 12 shows, however, that SO(3)/0 cannot follow spin commands about a non-principal axis.

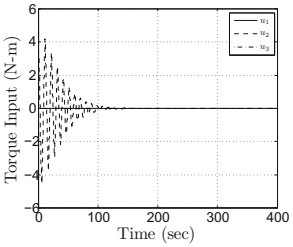
6.2 M2S Examples with Disturbances

Figure 13 shows that SO(3)/3 and SO(3)/9 can achieve spins about a non-principal axis in the presence of constant torque disturbances.

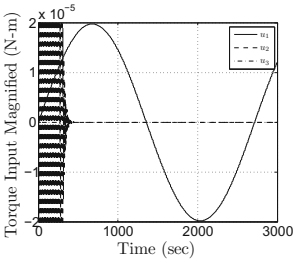
Figure 14 shows that SO(3)/6 is able to follow spin commands about a non-principal axis, albeit with large settling times. In the presence of a torque disturbance, SO(3)/6 cannot follow spin commands, and the resulting spin is about an incorrect axis. Consistent with [17, pp. 377], Figure 14 also confirms that non-principal-axis spins are unstable, since the spacecraft attitude diverges when the input torque is switched off.



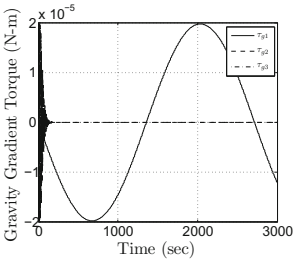
(a) SO(3)/3. Eigenaxis attitude error.



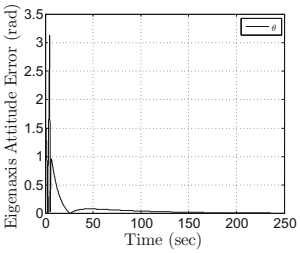
(b) SO(3)/3. Torque input



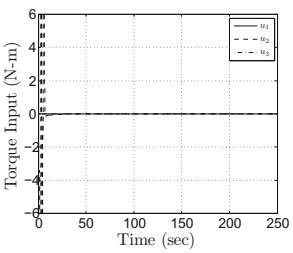
(c) SO(3)/3. Torque input zoomed.



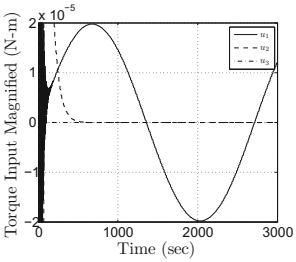
(d) SO(3)/3. Gravity gradient disturbance torque.



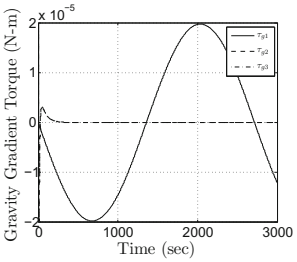
(e) SO(3)/9. Eigenaxis error.



(f) SO(3)/9. Torque input.

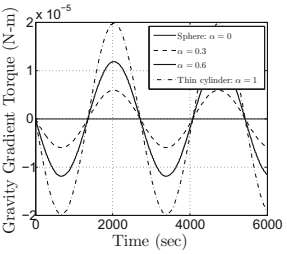


(g) SO(3)/9. Torque input zoomed.

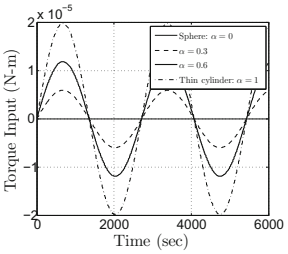


(h) SO(3)/9. Gravity gradient disturbance torque.

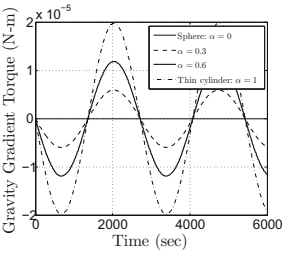
Fig. 4. Gravity gradient disturbance rejection for SO(3)/3 and SO(3)/9. The commanded maneuver is a 90-deg rotation about the body-fixed direction $[0\ 1\ 0]^{\mathrm{T}}$. The spacecraft is stabilized, and the disturbance torque is rejected. Note that the control input is the mirror image of the disturbance torque once that the commanded attitude is achieved.



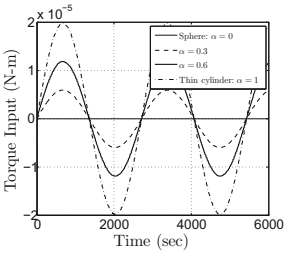
(a) SO(3)/3. Gravity gradient disturbance torque.



(b) SO(3)/3. Torque input.

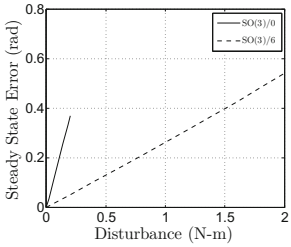


(c) SO(3)/9. Gravity gradient disturbance torque.

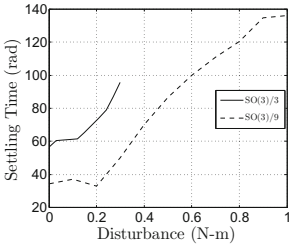


(d) SO(3)/9. Torque input.

Fig. 5. Gravity-gradient disturbance rejection for SO(3)/3 and SO(3)/9. The commanded motion is along a circular orbit with $R(0) = I$ and the commanded attitude $R_{\mathrm{d}} = I$. Note that the inertially constant pointing command is achieved despite the presence of an attitude-dependent sinusoidal disturbance due to gravity gradients. Note that the control input is the mirror image of the disturbance torque.

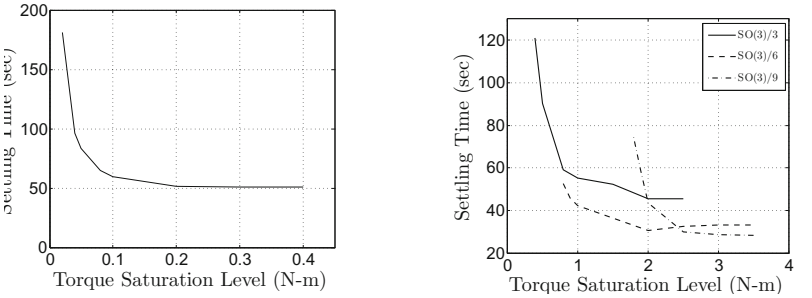


(a) R2R steady-state error for SO(3)/0 and SO(3)/6.



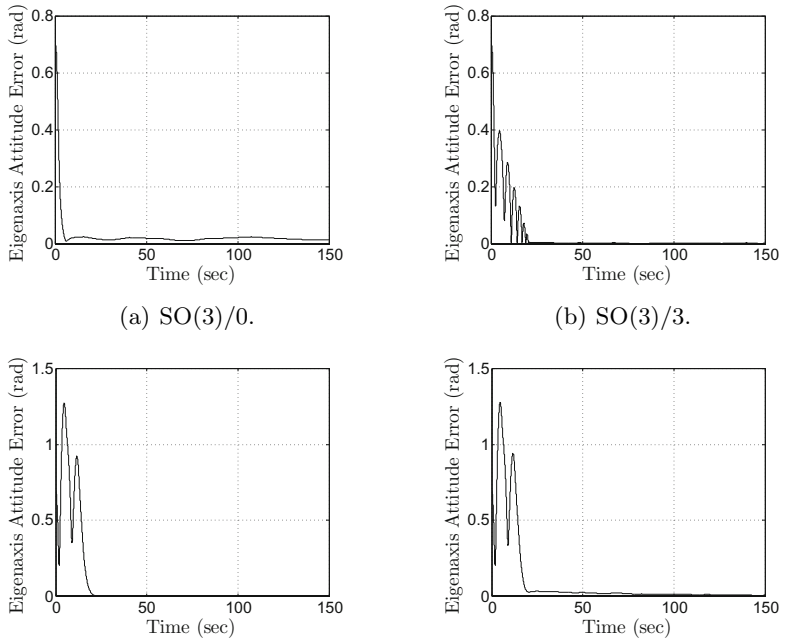
(b) R2R settling time for SO(3)/3 and SO(3)/9.

Fig. 6. Inertially constant disturbance-torque rejection about the inertially fixed direction $[0\ 0\ 1]^{\mathrm{T}}$. The maneuver is a 40-deg rotation about the body-fixed direction $[1\ 1\ 1]^{\mathrm{T}}$. The control laws SO(3)/0 and SO(3)/6 bring the spacecraft to rest with an attitude offset, whereas SO(3)/3 and SO(3)/9 bring the spacecraft to rest with the commanded attitude. Note that SO(3)/6 and SO(3)/9 perform substantially better than SO(3)/0 and SO(3)/3.



(a) SO(3)/0. (b) SO(3)/3, SO(3)/6, and SO(3)/9.

Fig. 7. R2R settling time as a function of the control-torque saturation level on all three axes. The maneuver is a 40-deg rotation about the body-fixed direction $[1\ 1\ 1]^{\mathrm{T}}$. Note that, at low saturation levels, SO(3)/0 stabilizes the spacecraft, whereas SO(3)/3, SO(3)/6, and SO(3)/9 fail. Saturation does not affect the performance of SO(3)/0 for saturation levels greater than 0.3 N-m.



(a) SO(3)/0. (b) SO(3)/3.

(c) SO(3)/6. (d) SO(3)/9.

Fig. 8. Performance comparison using on-off thrusters. The maneuver is a 40-deg rotation about the body-fixed direction $[1\ 1\ 1]^{\mathrm{T}}$. The tuning parameters and control-torque magnitude are the same in all four cases.

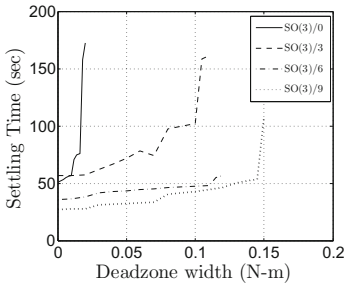


Fig. 9. R2R settling time for all four control laws as a function of the width of an unknown control-torque deadzone. The maneuver is a 40-deg rotation about the body-fixed direction $[1\ 1\ 1]^{\mathrm{T}}$. Note that, for deadzones of small width, SO(3)/0 fails to stabilize the spacecraft, whereas SO(3)/3, SO(3)/6, and SO(3)/9 can stabilize at much higher control-torque deadzones.

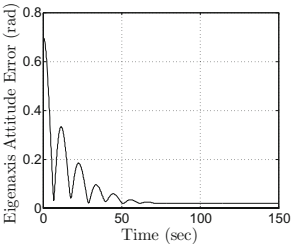
7 Comparison to Classical Optimal Control

Classical optimal control laws have been applied extensively to spacecraft rotational maneuvers [20–22]. These control laws are based on the minimum principle, and, unlike the inertia-free SO(3) control laws considered in this paper, they assume exact knowledge of the inertia properties of the spacecraft. Nevertheless, it is useful to compare the performance of these control laws to the SO(3) control laws in order to assess the effect of inertia uncertainty modeling information.

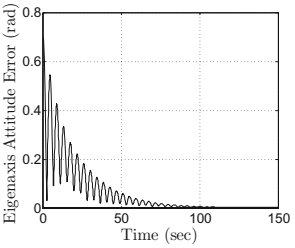
For 3-axis maneuvers, the control laws given in [20–22] involve complicated switching strategies. For simplicity, we therefore assume that the commanded maneuver is about the major axis only, giving double integrator dynamics, and with the direction and moment of inertia of the major axis assumed to be known. Using the classical time-optimal control law

$$u = -u_{\max}\mathrm{sign}\left(\theta - \frac{J}{2u_{\max}}\dot{\theta}|\dot{\theta}|\right), \tag{26}$$

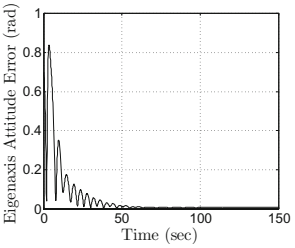
where $u_{\max} > 0$ is the control-torque magnitude and θ is the rotation angle about the major axis, we simulate the closed-loop system and compare the settling time to that of the SO(3) control laws operating in on-off mode with the same magnitude as (26). Note that the inertia in (26) is the true spacecraft inertia. To determine the performance of (26) under imperfect modeling information, we then introduce uncertainty about both the major moment of inertia and the direction of the major axis. Comparisons with the SO(3) control laws provide a baseline tradeoff between settling time and modeling accuracy. The results of these comparisons are shown in Figures 15 and 16.



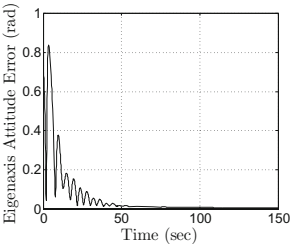
(a) SO(3)/0. Gyro bias.



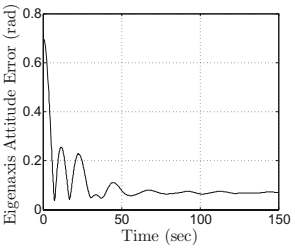
(b) SO(3)/3. Gyro bias.



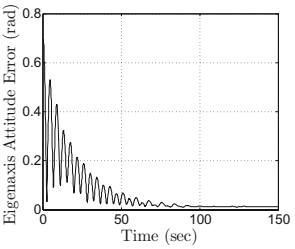
(c) SO(3)/6. Gyro bias.



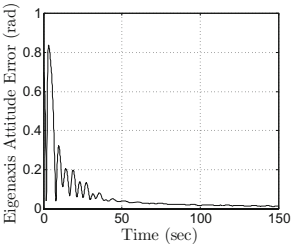
(d) SO(3)/9. Gyro bias.



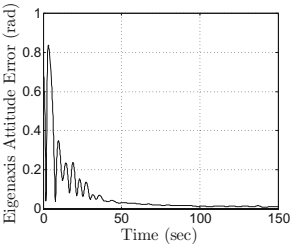
(e) SO(3)/0. Zero-mean white gyro measurement noise.



(f) SO(3)/3. Zero-mean white gyro measurement noise.

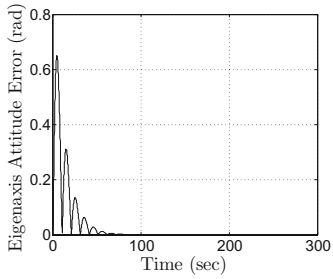


(g) SO(3)/6. Zero-mean white gyro measurement noise.

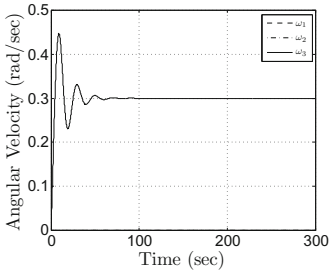


(h) SO(3)/9. Zero-mean white gyro measurement noise.

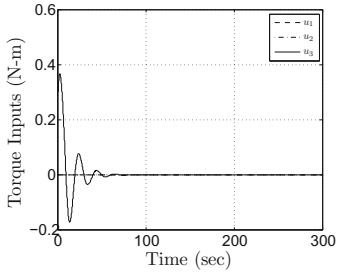
Fig. 10. (a)–(d) Controller performance in the presence of a gyro bias of $[0\ 0\ 0.01]^{\mathrm{T}}$ rad/sec. (e)–(h) Comparison of controller sensitivity to zero-mean white gyro measurement noise with a signal-to-noise ratio of 20. All four controllers are able to bring the spacecraft to rest. However, SO(3)/0 is not able to stabilize to the correct attitude with either gyro bias or white noise. In all simulations, the maneuver is a 40-deg rotation about the body-fixed direction $[1\ 1\ 1]^{\mathrm{T}}$. The tuning parameters and signal-to-noise ratio are kept the same.



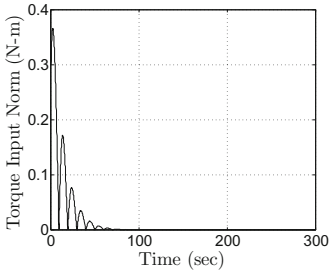
(a) Eigenaxis error.



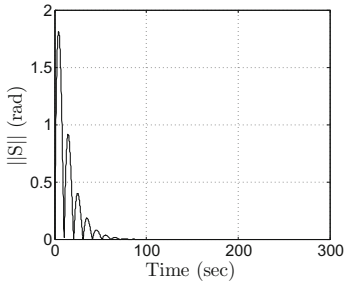
(b) Angular velocity components.



(c) Torque inputs.

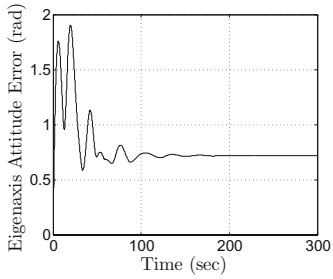


(d) Torque input 2-norm.

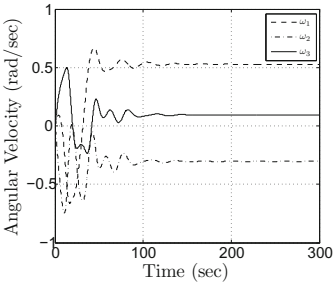


(e) 2-norm of S.

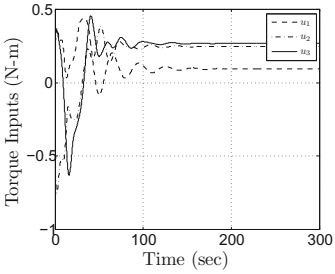
Fig. 11. R2S maneuver for SO(3)/0 with $\omega_{\mathrm{d}} = [0\ 0\ 0.3]^{\mathrm{T}}$ rad/sec. The spacecraft is initially at rest with $R = I$ and $R_{\mathrm{d}}(0) = I$. The controller is able to follow the spin command, which is about a principal axis.



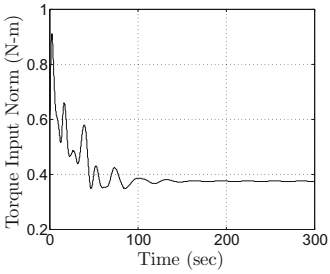
(a) Eigenaxis error.



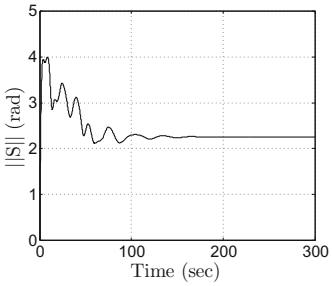
(b) Angular velocity components.



(c) Torque inputs.

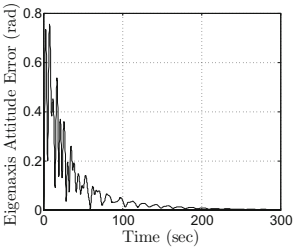


(d) Torque input 2-norm.

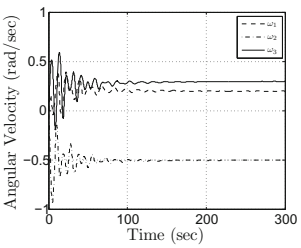


(e) 2-norm of S.

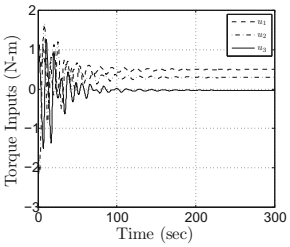
Fig. 12. R2S maneuver for SO(3)/0 with $\omega_{\mathrm{d}} = [0.2\ -0.5\ \ 0.3]^{\mathrm{T}}$ rad/sec. The spacecraft is initially at rest with $R = I$ and $R_{\mathrm{d}}(0) = I$. The controller spins the spacecraft with the commanded angular rate but about an incorrect axis, as shown by the attitude error. Thus SO(3)/0 cannot follow the spin command, which is about a non-principal axis.



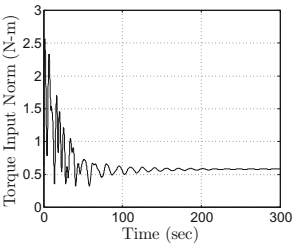
(a) SO(3)/3. Eigenaxis error.



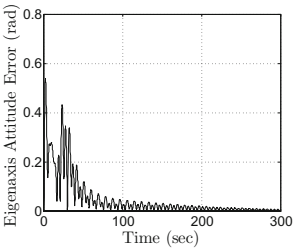
(b) SO(3)/3. Angular velocity components.



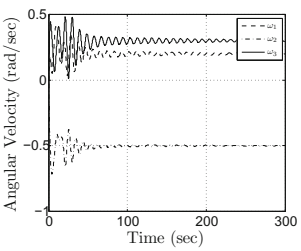
(c) SO(3)/3. Torque inputs.



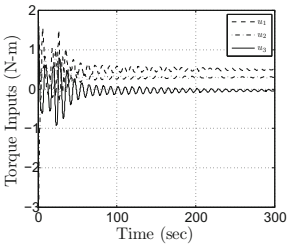
(d) SO(3)/3. Torque input 2-norm.



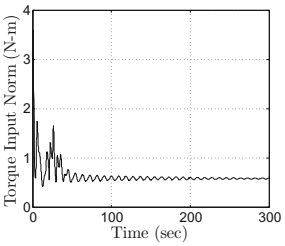
(e) SO(3)/9. Eigenaxis error.



(f) SO(3)/9. Angular velocity components.

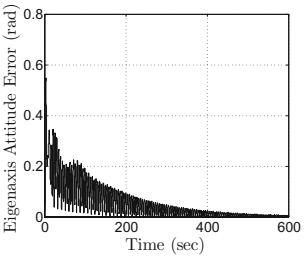


(g) SO(3)/9. Torque inputs.

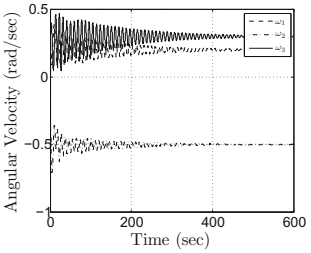


(h) SO(3)/9. Torque input 2-norm.

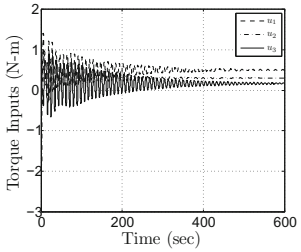
Fig. 13. R2S maneuver for $\omega_{\mathrm{d}} = [0.2\ -0.5\ \ 0.3]^{\mathrm{T}}$ rad/sec with gravity gradient disturbance and the body-constant disturbance torque $d = [0\ 0\ 0.2]^{\mathrm{T}}$ N-m for SO(3)/3 and SO(3)/9. The spacecraft is initially at rest with $R = I$ and $R_{\mathrm{d}}(0) = I$. The controller rejects the disturbances and follows the spin command.



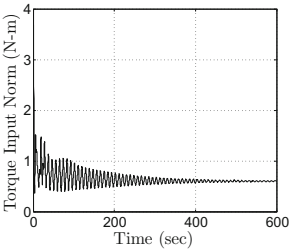
(a) SO(3)/6. Eigenaxis error.



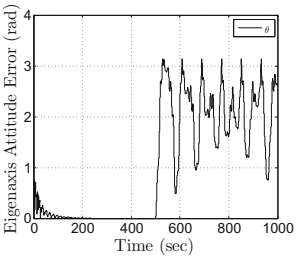
(b) SO(3)/6. Angular velocity.



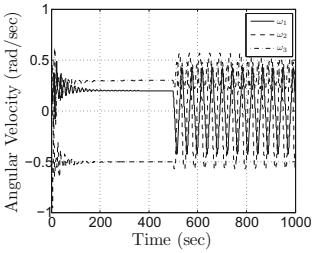
(c) SO(3)/6. Torque inputs.



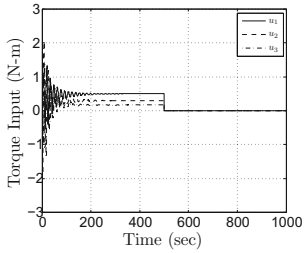
(d) SO(3)/6. Torque input norm.



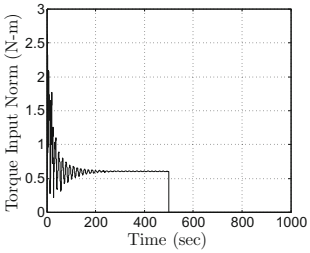
(e) SO(3)/3. Eigenaxis error.



(f) SO(3)/3. Angular velocity.

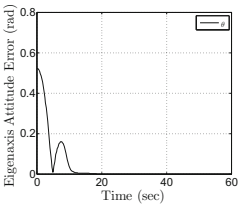


(g) SO(3)/3. Torque inputs.

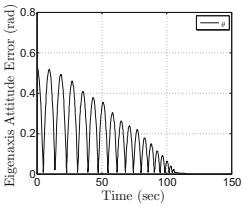


(h) SO(3)/3. Torque input norm.

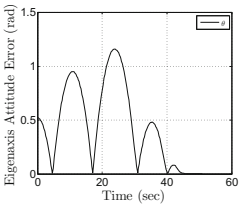
Fig. 14. R2S maneuver for SO(3)/6 and SO(3)/3 with $\omega_{\rm d} = [0.2 \ -0.5 \ \ 0.3]^{\rm T}$ rad/sec. The spacecraft is initially at rest with $R = I$ and $R_{\rm d}(0) = I$. The controllers follow the spin command. In the case of SO(3)/3, the torque input is switched off at time $t = 500$ sec to show the instability of the commanded spin about a non-principal axis.



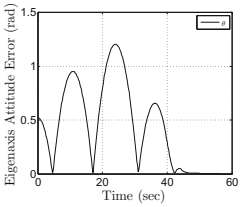
(a) SO(3)/0 control law.



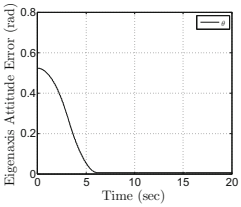
(b) SO(3)/3 control law.



(c) SO(3)/6 control law.

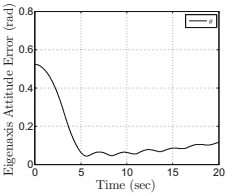


(d) SO(3)/9 control law.

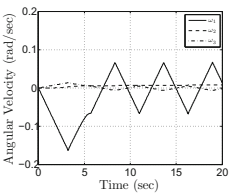


(e) Bang-bang control law.

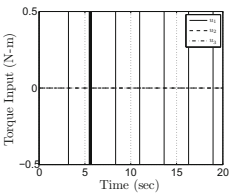
Fig. 15. Convergence-time comparison for the SO(3) control laws using on-off thrusters and the classical optimal bang-bang control law. The maneuver is a 30-deg rotation about the body-fixed principal-axis direction $[1\ 0\ 0]^{\mathrm{T}}$. The torque on-level $u_{\max} = 0.5$ N-m is the same for all controllers. The inertia used in the optimal bang-bang control law is the true spacecraft inertia J.



(a) Eigenaxis error.



(b) Angular velocity.



(c) Torque input.

Fig. 16. Performance of the classical optimal bang-bang control law in the presence of an inertia error. The actual bang-bang inertia matrix is rotated relative to the true spacecraft inertia matrix. This inertia misalignment is a 5-deg rotation about the body-fixed direction $[0\ 1\ 0]^{\mathrm{T}}$. The commanded maneuver is a 30-deg rotation about the body-fixed direction $[1\ 0\ 0]^{\mathrm{T}}$. Note that a slight inertia misalignment leads to failure of the bang-bang control law.

Table 2. Summary of controller capabilities. Motion to rest (M2R), gravity gradient disturbances (GG), constant torque disturbance (in body frame) (BD), constant torque disturbance (in inertial frame) (ID), torque saturation level (SAT), unknown control-torque deadzone (DZ), and on-off thruster actuation (ON/OFF). The notation used is, Y, N: Yes or no; L, S: based on Lyapunov theory or based on simulation.

Maneuver	SO(3)/0	SO(3)/3	SO(3)/6	SO(3)/9
M2R	Y/L	Y/S	Y/S	Y/L
M2R + ID	N/S	Y/S	N/S	Y/L
M2R + GG + BD	N/S	Y/S	N/S	Y/L
M2R + SAT	Y/S	Y/S	Y/S	Y/S
M2R + DZ	Y/S	Y/S	Y/S	Y/S
M2R + ON/OFF	Y/S	Y/S	Y/S	Y/S
M2R + GYRO BIAS	N/S	Y/S	Y/S	Y/S
M2R + WHITE NOISE	N/S	Y/S	Y/S	Y/S
M2S	N/S	Y/S	Y/S	Y/L
M2S + BD	N/S	Y/S	N/S	Y/L
M2S + SAT	N/S	Y/S	Y/S	Y/S

8 Conclusions

We compared four inertia-free PID-type spacecraft attitude control laws (SO(3)/0, SO(3)/3, SO(3)/6, SO(3)/9) under M2R and M2S command scenarios with various types of disturbances. All four controllers are able to achieve M2R and M2S around a principal axis in the absence of disturbances. In addition, SO(3)/3 and SO(3)/9 can achieve M2R and M2S in the presence of inertially constant and body-constant disturbances, and M2R in the presence of inertially time-varying disturbances around both principal and non-principal axes. Note that SO(3)/3 needs six fewer integrators than SO(3)/9, although SO(3)/9 achieves the commanded motion in less time.

Furthermore, all four controllers achieve M2R in the presence of torque saturation. For this objective, SO(3)/0 can stabilize the spacecraft with a significantly lower level of saturation than SO(3)/3, SO(3)/6, and SO(3)/9. For M2S maneuvers, SO(3)/0 is not effective, although SO(3)/3, SO(3)/6, and SO(3)/9 are effective for arbitrary spin axes. Table 2 summarizes the cases for which each SO(3) control law is able or not able to achieve the commanded maneuver.

For an unknown control-torque deadzone nonlinearity, we found that SO(3)/6 and SO(3)/9 are less sensitive to this nonlinearity. For on-off control torques, SO(3)/3 is the most accommodating. We also compared performance when sensor noise is present. In this case, SO(3)/3, SO(3)/6, and SO(3)/9 can stabilize M2R maneuvers despite gyro bias or white noise.

Future research will focus on a Lyapunov foundation for SO(3)/3 as well as extensions to spacecraft with wheels. Preliminary results are given in [19, 23].

References

1. Junkins, J.L., Akella, M.R., Robinett, R.D.: Nonlinear Adaptive Control of Spacecraft Maneuvers. AIAA J. Guid. Contr. Dyn. 20, 1104–1110 (1997)
2. Egeland, O., Godhavn, J.M.: Passivity-Based Adaptive Attitude Control of a Rigid Spacecraft. IEEE Transactions on Automatic Control 39, 842–846 (1994)
3. Ahmed, J., Coppola, V.T., Bernstein, D.S.: Asymptotic Tracking of Spacecraft Attitude Motion with Inertia Identification. AIAA J. Guid. Contr. Dyn. 21, 684–691 (1998)
4. Wen, J.T.Y., Kreutz-Delgado, K.: The Attitude Control Problem. IEEE Transactions on Automatic Control 36, 1148–1162 (1991)
5. Bhat, S.P., Bernstein, D.S.: A Topological Obstruction to Continuous Global Stabilization of Rotational Motion and the Unwinding Phenomenon. Sys. Contr. Lett. 39, 63–70 (2000)
6. Wie, B., Barba, P.M.: Quaternion Feedback for Spacecraft Large Angle Maneuvers. AIAA J. Guid. Contr. Dyn. 8, 360–365 (1985)
7. Joshi, S.M., Kelkar, A.G., Wen, J.T.: Robust Attitude Stabilization Using Nonlinear Quaternion Feedback. IEEE Trans. Autom. Contr. 40, 1148–1161 (1995)
8. Crassidis, J.L., Vadali, S.R., Markley, F.L.: Optimal Variable-Structure Control Tracking of Spacecraft Maneuvers. AIAA J. Guid. Contr. Dyn. 23, 564–566 (2000)
9. Mayhew, C.G., Sanfelice, R.G., Teel, A.R.: Quaternion-Based Hybrid Control for Robust Global Attitude Tracking. IEEE Trans. Autom. Contr. 56, 2555–2566 (2011)
10. Chaturvedi, N., Sanyal, A.K., McClamroch, N.H.: Rigid Body Attitude Control: Using Rotation Matrices for Continuous, Singularity-free Control Laws. IEEE Contr. Sys. Mag. 31(3), 30–51 (2011)
11. Koditschek, D.E.: The Application of Total Energy as a Lyapunov Function for Mechanical Control Systems. In: Proc. AMS-IMS-SIAM Joint Summer Research Conference, AMS Dynamics and Control of Multibody Systems, pp. 131–157 (1988)
12. Chaturvedi, N.A., McClamroch, N.H., Bernstein, D.S.: Asymptotic Smooth Stabilization of the Inverted 3D Pendulum. IEEE Trans. Automa. Contr. 54, 1204–1215 (2009)
13. Chaturvedi, N.A., McClamroch, N.H., Bernstein, D.S.: Stabilization of a 3D Axially Symmetric Pendulum. Automatica 44, 2258–2265 (2008)
14. Chaturvedi, N.A.: Global Dynamics and Stabilization of Rigid Body Attitude Systems. PhD thesis, University of Michigan, Ann Arbor, MI (2007)
15. Sanyal, A., Fosbury, A., Chaturvedi, N., Bernstein, D.S.: Inertia-Free Spacecraft Attitude Tracking with Disturbance Rejection and Almost Global Stabilization. AIAA J. Guid. Contr. Dyn. 32, 1167–1178 (2009)
16. Bullo, F., Lewis, A.D.: Geometric Control of Mechanical Systems. Springer (2005)
17. Wie, B.: Space Vehicle Dynamics and Control, vol. 2. AIAA (2008)
18. Cruz, G., Yang, X., Weiss, A., Kolmanovsky, I., Bernstein, D.S.: Torque-saturated, Inertia-free Spacecraft Attitude Control. AIAA–2011–6507 (August 2011)
19. Weiss, A., Kolmanovsky, I., Bernstein, D.S., Sanyal, A.: Inertia-Free Spacecraft Attitude Control Using Reaction Wheels. AIAA J. Guid. Contr. Dyn. 36, 1425–1439 (2013)
20. Junkins, J.L., Turner, J.D.: Optimal Spacecraft Rotational Maneuvers. Elsevier, Amsterdam (1986)
21. Bai, K.D., Wie, B.: Time-Optimal Three-Axis Reorientation of a Rigid Spacecraft. AIAA J. Guid. Contr. Dyn. 16, 446–452 (1993)

22. Bai, X., Junkins, J.L.: New Results for Time-Optimal Three-Axis Reorientation of a Rigid Spacecraft. AIAA J. Guid. Cont. Dyn. 32, 1071–1076 (2009)
23. Agarwal, K., Weiss, A., Kolmanovsky, I., Bernstein, D.S.: Inertia-Free Spacecraft Attitude Control with Control-Moment-Gyro Actuation. In: Proc. AIAA Guid. Nav. Contr. Conf. AIAA-2012-5003-282 (August 2012)

Editors Biography

Daniel Choukroun received the B.Sc. (summa cum laude), M.Sc., and Ph.D. degrees in 1997, 2000, and 2003, respectively, from Technion – Israel Institute of Technology, Faculty of Aerospace Engineering. He also received the title Engineer under Instruction from the Ecole Nationale de l'Aviation Civile, France, in 1994, and a Master in Space Systems Engineering from Delft University of Technology, Netherlands, in 2011. From 1998 to 2003, he was a teaching and research assistant in the field of automatic control at Technion—Israel Institute of Technology. From 2003 to 2006, he has been a postdoctoral fellow and a lecturer at the University of California at Los Angeles in the Department of Mechanical and Aerospace Engineering. Since 2006, he has been with the Department of Mechanical Engineering at Ben-Gurion University of the Negev, Israel. Between 2010 and 2014 he was with the Faculty of Aerospace Engineering at Delft University of Technology, The Netherlands, on a leave from Ben-Gurion University. He was Invited Lecturer at MAST - Master in Aerospace Science and Technology, Universitat Politecnica de Catalunya, Barcelona, Spain (2009), Visiting Assistant Professor at Université Paul Sabatier and LAAS/CNRS – Laboratoire d'Architecture et d'Analyse de Systèmes, Toulouse, France (2010), and Invited Lecturer at the European Space Agency graduate students courses for the European Student Earth Orbiter mission (2013).

Dr. Choukroun received the Miriam and Aaron Gutwirth Special Excellency Award for achievement in research from Technion—Israel Institute of Technology. His is co-author on the best paper awarded at the 2^{nd} CEAS Specialist Conference on Guidance, Navigation, and Control Conference in the graduate student competition.

He has authored and co-authored more than 60 papers in archival journals, international conference proceedings, and collective volumes, and has edited two collective volumes. His research has been supported by the Israel Science Foundation (ISF), the European Space Agency (ESA Education Office), and the German

Space Center (DLR/Oberpfaffenhofen). He is a member of the AIAA Technical Committee on Guidance Navigation and Control, of the CEAS-Council of European Aerospace Societies Technical Committee on Guidance, Navigation and Control, and of the EUCASS-European Conference of AeroSpace Sciences Board. Dr. Choukroun serves as technical editor in Guidance and Control for the IEEE Transactions on Aerospace and Electronic Systems, and as Guidance and Control field editor for the CEAS Space Journal.

Julie Thienel received her BS in Aerospace Engineering from the University of Arizona in 1986. She received an MS in Applied Physics from the Johns Hopkins University in 1989, and a PhD in Aerospace Engineering from the University of Maryland in 2004.

Dr. Thienel started with the NASA Goddard Space Flight Center in 1986 and worked in spacecraft guidance, navigation and control. From 2006-2010 she was on the faculty of the Aerospace Engineering Department of the US Naval Academy. In 2010 she returned to NASA Goddard as a proposal manager, leading new business efforts for missions and instruments.

Dr. Thienel is an Associate Fellow of the AIAA, and has served on the AIAA Guidance, Navigation, and Control technical committee. Dr. Thienel served as the Technical Chair of the AIAA GNC conference in 2009 and as the General Chair in 2012. She is the recipient of numerous NASA Goddard awards and holds a US Patent on a GPS/Magnetometer navigation system.

Moshe Idan received his B.Sc. and M.Sc. degrees from the Faculty of Aerospace Engineering at the Technion in 1983 and 1986, respectively. He earned his Ph.D. from the Department of Aerospace Engineering at Stanford University in 1990. Since 1991, Dr. Idan has been with the Faculty of Aerospace Engineering at the Technion. In 2000-2001 he was a visiting research scholar at the School of Aerospace Engineering at Georgia Tech, and in 2007-2009 he spent a two year sabbatical at the Department of Mechanical and Aerospace Engineering in UCLA. Dr. Idan is an Associate Fellow of the AIAA, serves as an international member the Guidance Navigation and Control technical committee of the AIAA, and is an Associate Editor for Guidance and Control for the IEEE Transactions on Aerospace and Electronic Systems. His current research interests include robust and adaptive flight control system design techniques and applications, innovative future air traffic control concepts, and control and estimation of systems with non-Gaussian noises.

Yaakov Oshman received his B.Sc. (*summa cum laude*) and D.Sc. degrees, both in Aeronautical Engineering, from the Technion—Israel Institute of Technology, Haifa, Israel, in 1975 and 1986, respectively.

From 1975 to 1981 he was with the Israeli Air Force, where he worked in the areas of structural dynamics and flutter analysis and flight testing. In 1987 he was a Research Associate at the Department of Mechanical and Aerospace Engineering of the State University of New York at Buffalo, where he was, in 1988, a Visiting Professor. Since 1989 he has been with the Department of Aerospace Engineering at the Technion—Israel Institute of Technology, where he is presently a Professor and Dean, and incumbent of the Louis and Helen Rogow Chair in Aeronautical Engineering.

Dr. Oshman is a member of the Technion's Asher Space Research Institute. He headed the Technion's Philadelphia Flight Control Laboratory (1993 – 1996). During the 1996/7 and 1997/8 academic years he spent a sabbatical with the Guidance, Navigation and Control Center of NASA's Goddard Space Flight Center, where he worked in research and development of spacecraft attitude estimation algorithms. He has consulted to RADA Electronic Industries Ltd., RAFAEL Advanced Defense Systems Ltd., Israeli Ministry of Defense, and Israel Aerospace Industries (MBT Missile Division). His research has been supported by the Israeli Aircraft Industries, the Israeli Ministry of Defense, the U.S. Air Force Office of Scientific Research (AFOSR), RAFAEL Advanced Defense Systems Ltd., and the Israeli Science Foundation (ISF), as well as by various Technion research grants.

Dr. Oshman served as President of the Israeli Association for Automatic Control, a national member organization of the International Federation of Automatic Control (IFAC), between 2003—2008. He was a member of the national board of the Israeli Society of Aeronautics and Astronautics between 2004—2009. He was a member of the AIAA Guidance, Navigation, and Control Technical Committee between 2002—2008, an International Advisor (member of the editorial board) of the *AIAA Journal of Guidance, Control and Dynamics* between 2002—2009, and a (founding) Technical Editor for Guidance and Control Systems for the *IEEE Transactions on Aerospace and Electronic Systems* between 2005—2011. He was a member of the Board of Governors of the IEEE Aerospace and Electronic Systems Society (AESS) between 2008—2010, and serves as Chair of the Israeli Chapter of IEEE/AESS since 2010.

Dr. Oshman's research interests are in advanced estimation, tracking, information fusion and control methods and their applications in aerospace guidance, navigation, and control systems, including structural estimation and control, flow control, and health monitoring/fault detection and isolation (FDI) systems. He has published over 150 journal and conference papers and book chapters, and numerous technical reports in these areas. He is a co-author of the paper that was awarded the Best Paper Award of the 2002 AIAA *Astrodynamics Specialist Conference*, and a co-author and advisor of the paper that was awarded the Best Paper Award of the 2004 AIAA Guid-

ance, Navigation and Control Conference. He received the Technion's *Raymond and Miriam Klein Research Prize* for his research on enhanced air-to-air missile tracking using target orientation observations (2002), the Technion's *Meir Hanin Research Prize* for his work on spacecraft angular velocity estimation (2004), which has been put to use in the Israeli AMOS-2 communication satellite, and multiple Technion awards for outstanding teaching. He has been on the program committees of over a dozen international conferences.

Dr. Oshman is a *Fellow* of the American Institute of Aeronautics and Astronautics (AIAA), and a *Fellow* of the Institute of Electrical and Electronics Engineers (IEEE).

Author Index