Acta Astronautica 178 (2021) 700–721

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Research paper

Adaptive extended Kalman filtering strategies for spacecraft formation relative navigation[☆]

Cory T. Fraser [*], Steve Ulrich

Carleton University, Department of Mechanical and Aerospace Engineering, 1125 Colonel By Drive, Ottawa, ON, K1S 5B6, Canada

ARTICLE INFO

MSC:
00-01
99-00

Keywords:
Spacecraft formation flying
Relative navigation
Kalman filter
Adaptive Kalman filter
Maximum likelihood estimation
Fuzzy logic

ABSTRACT

The relative navigation problem for spacecraft formation flying missions in near-Earth orbit is addressed here through the design of two unique adaptive extended Kalman filter algorithms. The adaptive filters are capable of updating the internal noise characteristics of the Kalman filter in real time, and are viable in all orbit scenarios, including elliptical orbits subjected to perturbations. The first adaptive Kalman filter approach uses maximum likelihood estimation techniques to derive analytical adaptations laws, which are then improved through the novel inclusion of an intrinsic smoothing routine. The second approach uses an embedded fuzzy logic system based on a covariance-matching analysis of the filter residuals, where the fuzzy system has been specifically designed for the spacecraft navigation problem at hand. Numerical simulations of two spacecraft formations demonstrate that the proposed adaptive navigation algorithms are appreciably more robust to filter initialization errors, dynamics modelling deficiencies, and measurement noises than the standard Kalman filter.

1. Introduction

The increasing performance capabilities and complexity of formation flying missions continue to drive stringent requirements for precise Guidance, Navigation and Control (GNC) systems on spacecraft. Throughout the past decade, it has further been emphasized that autonomous systems are more than just relevant for future rendezvous, docking, and formation flying applications; they are mandatory for proximity operations, in-orbit servicing, and robust collision avoidance procedures, all of which will become critical as Low Earth Orbit (LEO) becomes more crowded (e.g., see *Kessler Syndrome*). In addition, many measurement systems that require precise baseline separations, particularly synthetic aperture radar [1,2], gravimetry [3], and deep-space observation [4], are significantly enhanced by the flexibility of spacecraft formations and the ease of redundancy they entail. One of the foremost technological challenges to be addressed is therein the development of onboard navigation systems that are accurate and robust for a wide array of formation configurations, operating modes, and orbital environments, while simultaneously adhering to the limited processing capabilities available on spacecraft computers. Knowledge of the relative position and velocity between spacecraft in a formation has direct implications on both the guidance and control subsystems, so the computationally elegant Kalman filter has historically been selected to provide relative navigation solutions. The efficacy of the Kalman filter is demonstrated by advanced GNC systems used in recent missions such as PRISMA [5], CanX-4/5 [6], and AVANTI [7,8], which were able to maintain control accuracies at the sub-metre level and thereby present a new host of possibilities for the safe utilization of co-orbiting, cooperative formations.

The Kalman filter [9] is a recursive algorithm that provides an efficient method to fuse information from a measurement system with predictions made by a dynamical model of the expected system behaviour, thereby estimating the system state more accurately than either the measurements or dynamics alone. Not surprisingly, the measurement system, the dynamics model, and the fusion algorithms chosen within the filter each have a significant impact on the final navigation solution. Furthermore, the standard extended Kalman filter (EKF) assumes fixed uncertainties in stochastic error parameters within the dynamics and measurement models, and these uncertainties are known to propagate through the filter. Typically the effects of the uncertainties can be mitigated to a reasonable degree through the process of tuning the EKF, but this requires a particular amount of human interaction and experience [10], thereby limiting the effectiveness of the filter in the sense of autonomous operation.

The initial suggestion of adapting the Kalman filter to account for unknown *a priori* process and measurement noise statistics was made by Mehra [11,12], who identified four stochastic approaches that could

---

[☆] This work was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under the Alexander Graham Bell Canada Graduate Scholarship.

[*] Corresponding author.
E-mail address: coryt.fraser@carleton.ca (C.T. Fraser).

https://doi.org/10.1016/j.actaastro.2020.10.016
Received 21 July 2020; Accepted 8 October 2020

be applicable for adaptive EKF (AEKF) design. Two of these approaches are promising for real-time implementation, respectively referring to the methods of Maximum Likelihood Estimation (MLE) and Covariance Matching. Using MLE within the Kalman filter is in essence a technique that blends state estimation and system identification; by defining a set of parameters that influence a likelihood function based on the observed measurements and the filter state estimates (*i.e.*, parameters such as the noise covariance matrices, dynamics state transition matrices, or the input mapping matrices), adaptations to these parameters within the Kalman filter can be updated in order to maximize the observed likelihood function. Mehra and Bayard presented an application of an MLE adaptive extended Kalman filter (MLE-AEKF) for spacecraft attitude estimation [13], and subsequent works have applied MLE-AEKFs to INS/GPS navigation problems [14,15]. Busse and How further introduced the MLE technique into the spacecraft formation scenario, where it was demonstrated that improvements in relative position knowledge could be obtained compared to the EKF [16], using GPS single-difference carrier phase measurements. However, their filter employed a linearized model of the relative dynamics.

Contrasting with the MLE-AEKF, another technique commonly used for adapting the Kalman filter is covariance matching, where observations of the filter innovations covariance are compared with the theoretical innovations covariance predicted by the filter. The goal of covariance matching is then to update parameters within the EKF such that the theoretical and observed innovation covariances match, which corresponds to optimal filter performance. In 2011, Jiancheng et al. [17] used such an approach to address in-flight alignment errors developing in aircraft GPS measurements, however the proposed implementation eliminated the dependency of the filter on the measurement noise covariance matrix entirely. In the calculation of the Kalman gain, the theoretical innovations covariance matrix was replaced with the observed innovations covariance matrix, and so this technique restricts the applicability of the adaptation process to situations where only uncertainty in the measurement noise exists. Other authors have however applied covariance matching schemes to adapt both the process noise and measurement noise statistics within the Kalman filter, such as the work in [18], which relates to estimating land vehicle position through combining IMU and GPS data. The idea of covariance matching has given rise to new methods of adapting the EKF in a variety of disciplines, including techniques using optimization [19], forgetting factors [20], neural network inference [21], and fuzzy logic.

Applying fuzzy set theory [22–24] to AEKF design provides a method for encoding human experience, intuition, and reasoning into computer logic decision making processes. Fuzzy systems have appeared in many fields of control theory, state estimation, and Kalman filter development since the inception of fuzzy logic and the associated mathematics, but to the best of the author's knowledge, fuzzy adaptive extended Kalman filter (FAEKF) techniques have not yet been applied to the problem of autonomous spacecraft formation flying. Nevertheless, the use of fuzzy adaptive filters in GPS/INS sensor fusion has been shown to improve navigation accuracy, such as in the work by Da Silva and da Cruz [25], and also Ali [26]. A useful result from these studies showed that by only adapting the diagonal elements, the fuzzy inference system can be treated as multiple single-input single-output systems, thereby reducing computational complexity. Additional mechanisms for adapting the process noise covariances have been explored by Tseng and Lin [27], which demonstrated that suitable adjustments to the process noise covariance could be obtained using scalar metrics of the innovation covariance matrices instead of the full covariance matrices themselves. However, these proposed fuzzy adaptation laws were developed for Earth-borne applications and embedded within a cubature Kalman filter, which is a form of Bayesian filter that is computationally more demanding than the EKF. It can therefore be seen that FAEKF methods are indeed capable of improving the estimation accuracy over the traditional EKF, yet the technology has not yet been applied to relative navigation for cooperative spacecraft.

In this context, the main contributions of this paper are: (1) the design of a standard EKF, which, unlike Busse and How [17] where a linearized dynamics model was assumed, is based on the full nonlinear relative motion dynamics model for spacecraft formation flying. Furthermore, system observability is verified analytically through the use of a Lie derivative criterion; notable past studies including Butcher et al. [28], Huxel et al. [29], and Kaufman et al. [30] performed observability analyses using different dynamics and measurement models than those used here. Thus, a nonlinear observability analysis has not previously been addressed for the system presented herein. (2) the expansion of the EKF to include an online adaptation scheme that harnesses previous filter information in an maximum likelihood approach, which further utilizes an internal smoothing algorithm to improve the MLE solution; (3) the development of an adaptation scheme using fuzzy logic, which consists of a unique set of unevenly distributed output membership functions and FLS gains specific to the spacecraft formation navigation problem at hand. Both of the adaptive EKFs are capable of updating the process and measurement noise covariance matrices within the respective filters, and simulation results of two dual-spacecraft formations are presented to validate the performance of the adaptive EKFs relative to measurement-only and non-adaptive EKF navigation solutions.

2. Relative navigation using an extended Kalman filter

Before discussing the novel adaptive EKF algorithms developed for spacecraft navigation, the following section provides an overview of the relative state estimation framework and the development of a standard EKF that is used as the comparative baseline throughout this paper. The onboard dynamics and measurement models used in this work are also given, followed by a summary of the Kalman filter equations. An analysis of the system observability is presented last, to reinforce the suitability of the EKF to the relative navigation problem for formation flying spacecraft.

2.1. Nonlinear estimation framework

Using a general continuous-time framework to describe the orbital mechanics that govern spacecraft motion, the relative dynamics and discrete measurements of the formation can be treated as a nonlinear system of the form

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, \mathbf{u}, t) + \mathbf{w}(t) \tag{1}$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \tag{2}$$

The spacecraft dynamics models are contained within the non-autonomous vector field $\mathbf{f}(\cdot)$, which is a time-varying function of the state vector $\mathbf{x} \in \mathbb{R}^n$ and known control inputs $\mathbf{u} \in \mathbb{R}^q$. Modelling inaccuracies will inherently be present as well, so a process noise $\mathbf{w} \in \mathbb{R}^p$ is included to account for these deficiencies. For a given set of m sensors, observations of the dynamic system at a discrete time t_k can be collected in the measurement set $\mathbf{z}_k \in \mathbb{R}^m$. The corresponding measurement equations within $\mathbf{h}(\cdot)$ are used to model the sensors and account for the effects of measurement noises through $\mathbf{v}_k \in \mathbb{R}^m$, thereby providing a direct map of the system states into the observation space.

All noise processes considered here are assumed to follow white, Gaussian distributions with zero-mean, such that the dynamics and measurement noises at each discrete time t_k can be classified by normal distributions as $\mathbf{w}_k \sim \mathcal{N}\left(\mathbf{0}, \mathbf{Q}_k\right)$ and $\mathbf{v}_k \sim \mathcal{N}\left(\mathbf{0}, \mathbf{R}_k\right)$, respectively. Taking these noises to be uncorrelated with each other implies $E[\mathbf{v}_k \mathbf{w}_k^T] = \mathbf{0}$, where the notation $E[\cdot]$ designates the expectation operator. From the preceding assumptions, the noise processes can be fully characterized in terms of positive definite symmetric covariance matrices, where $\mathbf{Q}_k = E[\mathbf{w}_k \mathbf{w}_k^T] \in \mathbb{R}^{n \times n}$ defines the process noise covariance, and $\mathbf{R}_k = E[\mathbf{v}_k \mathbf{v}_k^T] \in \mathbb{R}^{m \times m}$ defines the measurement noise covariance.

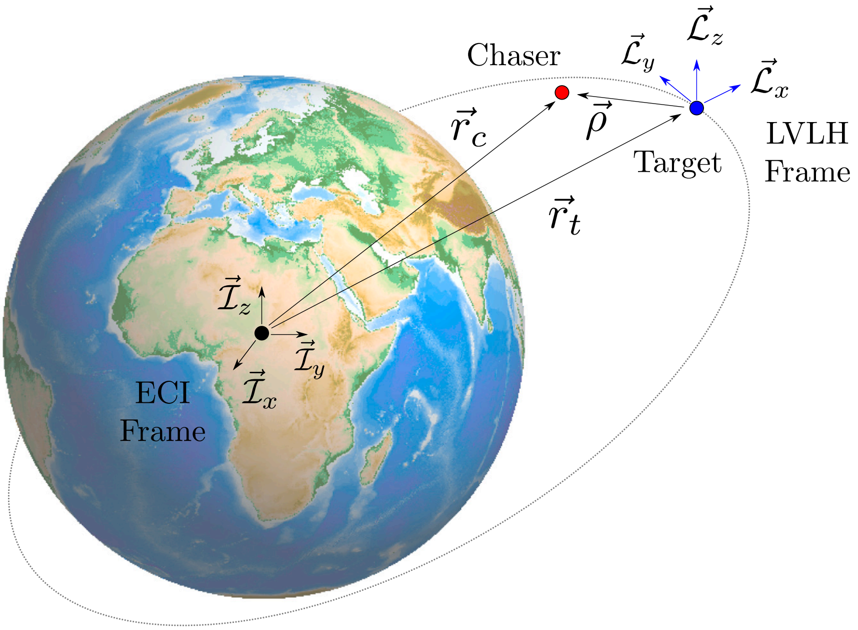


Fig. 1. The target and chaser spacecraft formation, along with the ECI frame $\mathcal{F}_I$ and the LVLH frame $\mathcal{F}_L$.

In order to proceed with the use of standard Kalman filtering techniques, it is assumed that the dynamics models in $\mathbf{f}(\cdot)$ and the measurement models in $\mathbf{h}(\cdot)$ are sufficiently smooth, such that they can be approximated using Taylor series expansions. The EKF utilizes the linear terms of the Taylor expansions to approximate the behaviour of the estimated states, so before exploring the development of the EKF itself, the following two subsections describe the dynamics and measurement models used in the relative navigation problem for spacecraft formations.

2.2. Spacecraft formation dynamics model

The formation configurations analysed in this work consider two spacecraft, one called the target and the other called the chaser, as illustrated in Fig. 1. Relative motion of the chaser with respect to the target is modelled and estimated in the Local-Vertical Local-Horizontal (LVLH) reference frame denoted by $\mathcal{F}_L$, which originates at the target. Developing dynamics equations and visualizing the resulting relative motion in the LVLH frame is intuitive since the frame is coincident with the target spacecraft through the entire orbit. The radial direction $\vec{\mathcal{L}}_x$, the along-track direction $\vec{\mathcal{L}}_y$, and the cross-track direction $\vec{\mathcal{L}}_z$ of the LVLH coordinate system are defined by the orthogonal unit vectors

$$\vec{\mathcal{L}}_x = \frac{\vec{r}_t}{r_t} \qquad \vec{\mathcal{L}}_y = \vec{\mathcal{L}}_z \times \vec{\mathcal{L}}_x \qquad \vec{\mathcal{L}}_z = \frac{\vec{r}_t \times \vec{v}_t}{|\vec{r}_t \times \vec{v}_t|} \tag{3}$$

where $\vec{r}_t$ and $\vec{v}_t$ are the position and velocity vectors of the target in the Earth-Centred Inertial (ECI) frame $\mathcal{F}_I$, which has corresponding unit vectors $\vec{\mathcal{I}}_x$, $\vec{\mathcal{I}}_y$, and $\vec{\mathcal{I}}_z$. In addition, $r_t = |\vec{r}_t|$ is the magnitude of the position vector of the target. Using vectrix notation [31], the relative position vector is

$$\vec{\rho} = \vec{r}_c - \vec{r}_t = \vec{\mathcal{F}}_L^T \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{4}$$

where $\vec{\mathcal{F}}_L$ is the vectrix describing the orientation of the LVLH frame. The Cartesian position terms x, y and z refer to the radial, along-track and cross-track components of the chaser spacecraft, relative to the target. Thus, the components of interest for relative navigation can be summarized as the relative position $\boldsymbol{\rho} = \begin{bmatrix} x & y & z \end{bmatrix}^T$ and the relative velocity $\dot{\boldsymbol{\rho}} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} \end{bmatrix}^T$, which fully describe the three-dimensional motion of the chaser with respect to the target.

The magnitude of the position vector of the chaser is easily identified from the problem geometry as $r_c = |\vec{r}_c| = \sqrt{(r_t + x)^2 + y^2 + z^2}$, and the moving LVLH frame is defined in inertial space through the true anomaly θ and r_t, which establish the position of the target in the orbital frame. Using μ to represent the Earth's gravitational parameter, the nonlinear equations of motion in the LVLH frame for an uncontrolled spacecraft formation are expressed as [31]:

$$\ddot{x} = \dot{\theta}^2 x + 2\dot{\theta}\left(\dot{y} - y\frac{\dot{r}_t}{r_t}\right) + \frac{\mu}{r_t^2} - \frac{\mu}{r_c^3}(r_t + x) \tag{5}$$

$$\ddot{y} = \dot{\theta}^2 y - 2\dot{\theta}\left(\dot{x} - x\frac{\dot{r}_t}{r_t}\right) - \frac{\mu}{r_c^3} y \tag{6}$$

$$\ddot{z} = -\frac{\mu}{r_c^3} z \tag{7}$$

$$\ddot{\theta} = -2\frac{\dot{r}_t}{r_t}\dot{\theta} \tag{8}$$

$$\ddot{r}_t = \dot{\theta}^2 r_t - \frac{\mu}{r_t^2} \tag{9}$$

These nonlinear equations of relative motion consist of five second-order, ordinary differential equations, implying a total of $n = 10$ system states for the relative navigation problem. Casting these equations into the general framework of Eq. (1), the state vector $\mathbf{x} \in \mathbb{R}^{10}$, and the corresponding dynamics model $\dot{\mathbf{x}} \in \mathbb{R}^{10}$, are respectively defined as

$$\mathbf{x} = \begin{bmatrix} x & y & z & \theta & r_t & \dot{x} & \dot{y} & \dot{z} & \dot{\theta} & \dot{r}_t \end{bmatrix}^T \tag{10}$$

$$\mathbf{f}(\mathbf{x}) = \dot{\mathbf{x}} = \begin{bmatrix} \dot{x} & \dot{y} & \dot{z} & \dot{\theta} & \dot{r}_t & \ddot{x} & \ddot{y} & \ddot{z} & \ddot{\theta} & \ddot{r}_t \end{bmatrix}^T \tag{11}$$

2.3. Spacecraft formation measurement model

The selected measurements used in the navigation routines here include direct measurements of the relative position and velocity states of the formation, such as those that could be obtained by differencing Global Positioning System (GPS) observables. In addition, a measurement of the inertial position of the target spacecraft through the true anomaly is provided, which could be generated by ground-based observation systems or on-board inertial measurement systems, for example. These $m = 7$ measurements comprise the measurement vector $\mathbf{h}(\mathbf{x}) \in \mathbb{R}^7$, formally defined by

$$\mathbf{h}(\mathbf{x}) \triangleq \begin{bmatrix} x_m & y_m & z_m & \theta_m & \dot{x}_m & \dot{y}_m & \dot{z}_m \end{bmatrix}^T \tag{12}$$

where the subscript m indicates a measured quantity. For simulation within this research, the measured relative positions $\{x_m, y_m, z_m\}$, relative velocities $\{\dot{x}_m, \dot{y}_m, \dot{z}_m\}$, and true anomaly of the target θ_m are calculated from the output of the real-world orbit propagator, which is explained later in Section 5.

2.4. Extended Kalman filtering

The relative navigation problem is approached using a continuous–discrete EKF, where the objective of the Kalman filter is to provide a state estimate $\hat{\mathbf{x}}_k$ of the true system state $\mathbf{x}_k$, to a higher degree of accuracy than is possible with measurements alone. This accuracy is quantified by the state estimation errors $\mathbf{e}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k$, which are assumed to be uncorrelated with the process and measurement noises, so $E[\mathbf{e}_k\mathbf{w}_k^T] = \mathbf{0}$ and $E[\mathbf{e}_k\mathbf{v}_k^T] = \mathbf{0}$. The derivation of the EKF is based on minimizing the variance of the estimation errors, for which the state error covariance matrix is $\mathbf{P}_k = E[\mathbf{e}_k\mathbf{e}_k^T] \in \mathbb{R}^{n\times n}$. Estimation of the state is completed through two steps: a propagation step that uses the onboard dynamics model, and a correction step that uses the set of available measurements.

The first stage of the Kalman filter relies on the estimation of the state before a measurement is collected. Noting that the dynamics are defined in the continuous domain, numerically integrating Eq. (11) from the past time step t_{k-1} to the current time step t_k gives an *a priori* state estimate, denoted by $\hat{\mathbf{x}}_k^-$. Due to dynamics modelling errors this propagation will introduce error into the state estimate, which must be quantified in the error covariance matrix $\mathbf{P}_k$. Contrasting the nonlinear propagation of the state estimate in the continuous domain, propagation of the error covariance through the time step $\Delta t = t_k - t_{k-1}$ is performed using the discrete state transition matrix $\boldsymbol{\Phi}_{k-1} = \exp\left[\mathbf{F}_{k-1}\Delta t\right] \in \mathbb{R}^{n\times n}$, where $\mathbf{F}_{k-1} \in \mathbb{R}^{n\times n}$ is the linearized dynamics model. The process of linearization involves evaluating the Jacobian of $\mathbf{f}(\cdot)$ at the best state estimate from the previous time step, which is referred to here as the *a posteriori* state estimate $\hat{\mathbf{x}}_{k-1}^+$. Thus, propagation of the state estimate and error covariance is completed through:

$$\hat{\mathbf{x}}_k^- = \hat{\mathbf{x}}_{k-1} + \int_{t_{k-1}}^{t_k} \mathbf{f}(\hat{\mathbf{x}}_{k-1}, t)\ \mathrm{d}t \tag{13}$$

$$\mathbf{P}_k^- = \boldsymbol{\Phi}_{k-1}\mathbf{P}_{k-1}^+\boldsymbol{\Phi}_{k-1}^T + \mathbf{Q}_k \tag{14}$$

The second stage of the EKF uses measurements of the system taken at the current time step to correct the *a priori* estimates. With the measurement model from Eq. (2), estimated measurements are calculated using the *a priori* state estimates with $\hat{\mathbf{z}}_k = \mathbf{h}(\hat{\mathbf{x}}_k^-)$, where the difference between these estimated measurements and the actual measurements are commonly known as the innovations, or the pre-fit residuals. Denoted by $\boldsymbol{\nu}_k = \mathbf{z}_k - \hat{\mathbf{z}}_k$, the residuals result from both inaccuracies in modelling the sensors, and errors within the propagation of the previous state estimate. The linearized measurement model $\mathbf{H}_k \in \mathbb{R}^{m\times n}$ and the Kalman gain $\mathbf{K}_k \in \mathbb{R}^{n\times m}$ are then used to calculate the *a posteriori* state estimate and error covariance matrix as shown below, where $\mathbf{I}$ represents the appropriately sized $n\times n$ identity matrix:

$$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}_k^T\left(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k\right)^{-1} \tag{15}$$

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k\left(\mathbf{z}_k - \hat{\mathbf{z}}_k\right) \tag{16}$$

$$\mathbf{P}_k^+ = \left(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k\right)\mathbf{P}_k^-\left(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k\right)^T + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k^T \tag{17}$$

2.4.1. Linearized dynamic model within the EKF

Since the EKF requires a linear model of the dynamics to propagate the error covariance ahead in time, the nonlinear model in Eq. (10) is linearized by evaluating the partial derivatives of the dynamics model with respect to the states. The Jacobian $\mathbf{F} \in \mathbb{R}^{n\times n}$ for nonlinear relative spacecraft motion expressed in the LVLH frame is given by

$$\mathbf{F} = \frac{\partial\mathbf{f}(\mathbf{x})}{\partial\mathbf{x}} = \begin{bmatrix} \mathbf{0}_{5\times5} & \mathbf{I}_{5\times5} \\ \mathbf{F}_{21} & \mathbf{F}_{22} \end{bmatrix} \tag{18}$$

where the submatrices $\mathbf{F}_{21}$ and $\mathbf{F}_{22}$ are

$$\mathbf{F}_{21} = \begin{bmatrix} \ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} \\ \ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} \\ \ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} \\ 0 & 0 & 0 & 0 & \ddot{\theta}_{r_t} \\ 0 & 0 & 0 & 0 & \ddot{r}_{t_{r_t}} \end{bmatrix} \qquad \mathbf{F}_{22} = \begin{bmatrix} 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\ \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \ddot{\theta}_{\dot{\theta}} & \ddot{\theta}_{\dot{r}_t} \\ 0 & 0 & 0 & \ddot{r}_{t_{\dot{\theta}}} & 0 \end{bmatrix} \tag{19}$$

The various partial derivative terms contained within $\mathbf{F}_{21}$ and $\mathbf{F}_{22}$ are presented in Appendix A, and the shorthand notation applied here uses subscripts to indicate derivatives to simplify the notation. This shorthand expresses that the partial derivative for a general function f with respect to an arbitrary variable x is represented as

$$f_x = \frac{\partial f}{\partial x} \tag{20}$$

2.4.2. Linearized measurement model within the EKF

Similarly to the dynamics linearization, the EKF requires the Jacobian of measurement model in order to correct the filter error covariance. Since the measurements collected here directly correspond to components of the state vector, performing a linearization of the measurement model gives $\mathbf{H} \in \mathbb{R}^{m\times n}$ as

$$\mathbf{H} = \frac{\partial\mathbf{h}}{\partial\mathbf{x}} = \begin{bmatrix} \mathbf{I}_{4\times4} & \mathbf{0}_{4\times1} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times2} \\ \mathbf{0}_{3\times4} & \mathbf{0}_{3\times1} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times2} \end{bmatrix} \tag{21}$$

2.5. Observability analysis for spacecraft formation flying

Before proceeding further, the concept of observability will briefly be discussed as it relates to the Kalman filter. For the general time-varying nonlinear system described by a state vector $\mathbf{x} \in \mathbb{R}^n$ and an output vector $\mathbf{y}(t) \in \mathbb{R}^p$, the system is said to be locally observable over the interval $t \in [0, T]$ if the mapping from the initial state $\mathbf{x}_0$ to the output $\mathbf{y}(t)$ is one-to-one [28]. Stated differently, local observability implies the initial state x_0 can be reconstructed from knowledge contained in the outputs over a given time interval.

Like the observability conditions for linear systems, a check for nonlinear observability involves determining if the observability matrix $\mathcal{O}(\mathbf{x})$ is full rank, such that all its columns are independent, and equivalently, rank $\mathcal{O}(\mathbf{x}) = n$. Construction of the nonlinear observability test matrix relies on the use of Lie derivatives, where the first-order Lie derivative of the output $\mathbf{h}(\mathbf{x})$ along a vector field $\mathbf{f}(\mathbf{x})$ is denoted by $\mathcal{L}_f\mathbf{h}(\mathbf{x}) \in \mathbb{R}^p$ and expressed as

$$\mathcal{L}_f\mathbf{h}(\mathbf{x}) \triangleq \frac{\partial\mathbf{h}(\mathbf{x})}{\partial\mathbf{x}}\mathbf{f}(\mathbf{x}) = \nabla\mathbf{h}(\mathbf{x})\mathbf{f}(\mathbf{x}) \tag{22}$$

with $\nabla(\cdot)$ representing the vector gradient. A recursive scheme for evaluating the next ith-order Lie derivatives is then found through induction to be given for all $i > 0$ by

$$\mathcal{L}_f^i\mathbf{h}(\mathbf{x}) \triangleq \frac{\partial\mathcal{L}_f^{i-1}\mathbf{h}(\mathbf{x})}{\partial\mathbf{x}}\mathbf{f}(\mathbf{x}) = \nabla\mathcal{L}_f^{i-1}\mathbf{h}(\mathbf{x})\mathbf{f}(\mathbf{x}) \tag{23}$$

and the zeroth-order Lie derivative is simply the measurement matrix $\mathbf{h}(\mathbf{x})$, so

$$\mathcal{L}_f^0\mathbf{h}(\mathbf{x}) = \mathbf{h}(\mathbf{x}) \tag{24}$$

The observability matrix $\mathcal{O}(\mathbf{x}) \in \mathbb{R}^{pn\times n}$ for the nonlinear system is then defined as

$$\mathcal{O}(\mathbf{x}) \triangleq \frac{\partial}{\partial\mathbf{x}}\begin{bmatrix} \mathcal{L}_f^0\mathbf{h}(\mathbf{x}) \\ \mathcal{L}_f^1\mathbf{h}(\mathbf{x}) \\ \vdots \\ \mathcal{L}_f^{n-1}\mathbf{h}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \nabla\mathcal{L}_f^0\mathbf{h}(\mathbf{x}) \\ \nabla\mathcal{L}_f^1\mathbf{h}(\mathbf{x}) \\ \vdots \\ \nabla\mathcal{L}_f^{n-1}\mathbf{h}(\mathbf{x}) \end{bmatrix} \tag{25}$$

The size of this nonlinear observability matrix clearly depends on the number of states and measurements in the system, so the computation of the higher-order derivatives can become quite intensive. As such, Butcher et al. [28] highlight that a sufficient (but not necessary) condition for observability can be obtained by only constructing the first N rows of $\mathcal{O}(\mathbf{x})$, or by constructing only a subsequent number of rows necessary to show that the truncated version of $\mathcal{O}(\mathbf{x})$ is full rank. For brevity, only the first $N = 3$ Lie derivatives are presented in the observability matrix $\mathcal{O}_N(\mathbf{x})$ below, demonstrating the full rank condition for the formation navigation problem addressed in this paper.

A thorough development of the observability matrix and the corresponding components is given in Appendix B, along with the definitions of the observability matrix terms $\alpha_{k,j}$ for $k \in 1,\dots,7$ and $j \in 1,\dots,10$. This derivation is notably different from other works in the literature, given the fact that past studies have used different dynamics and measurement models than those that are used here. For example, the observability analysis by Butcher et al. [28] uses nonlinear relative dynamics but angles-only measurements, the work of Huxel et al. [29] uses inertial Keplerian dynamics and range-only measurements, and Kaufman et al. [30] use nonlinear relative dynamics but line-of-sight measurements. Thus, presented here is a unique observability analysis that considers nonlinear relative dynamics with measurements of the relative position, relative velocity, and target true anomaly.

$$\mathcal{O}_N(\mathbf{x}) = \left[\begin{array}{cccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\hdashline
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} & 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\
\ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} & \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\
\ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} & 0 & 0 & 0 & 0 & 0 \\
\hdashline
\ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} & 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\
\ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} & \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\
\ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \ddot{\theta}_{r_t} & 0 & 0 & 0 & \ddot{\theta}_{\dot{\theta}} & \ddot{\theta}_{\dot{r}_t} \\
\alpha_{51} & \alpha_{52} & \alpha_{53} & 0 & \alpha_{55} & \alpha_{56} & \alpha_{57} & \ddot{x}_z & \alpha_{59} & \alpha_{5,10} \\
\alpha_{61} & \alpha_{62} & \alpha_{63} & 0 & \alpha_{65} & \alpha_{66} & \alpha_{67} & \ddot{y}_z & \alpha_{69} & \alpha_{6,10} \\
\alpha_{71} & \alpha_{72} & \alpha_{73} & 0 & \alpha_{75} & \ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t}
\end{array}\right] \tag{26}$$

3. Adaptive Kalman filtering using maximum likelihood estimation

Internal characteristics of the Kalman filter, including the initial conditions, dynamical constants, process noise covariances, and measurement noise covariances, all represent parameters that influence the performance of the EKF which are chosen at the discretion of the filter designer [10]. To improve the performance of the EKF in situations where these parameters are inaccurate, the following section presents a method of adapting the process noise covariance $\mathbf{Q}_k$ and the measurement noise covariance $\mathbf{R}_k$ using maximum likelihood estimation. To preface the derivation, recall the Kalman gain from Eq. (15), and note that the inverted term in parenthesis is defined as the *theoretical* covariance of the residuals $\boldsymbol{\Sigma}_k \in \mathbb{R}^{m\times n}$ from the filter, which gives

$$\boldsymbol{\Sigma}_k = \mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k = \mathbf{H}_k\left(\boldsymbol{\Phi}_{k-1}\mathbf{P}_{k-1}^+\boldsymbol{\Phi}_{k-1}^T + \mathbf{Q}_k\right)\mathbf{H}_k^T + \mathbf{R}_k \tag{27}$$

This filter Residual Covariance Matrix (RCM) contains both the process and measurement noise covariance matrices, thereby providing a mechanism through which the influence of these parameters on the performance of the EKF can be adapted. Denoting a general adaptation parameter at the current time step t_k with α_k, several assumptions are made here in order to obtain closed-form solutions for the adaptation equations; firstly, adaptations of the EKF are implemented through the covariance matrix of the residuals, meaning that for every time step:

$$\boldsymbol{\Sigma}_k = \boldsymbol{\Sigma}_k(\alpha_k) \qquad \frac{\partial \boldsymbol{\Sigma}_k(\alpha_k)}{\partial \alpha_k} \neq 0$$

Secondly, it is assumed that current filter states are insensitive to the adaptation parameters, while both the state transition matrix and the measurement model are independent of α_k and time invariant. This implies:

$$\frac{\partial \mathbf{x}_k}{\partial \alpha_k} = 0 \qquad \frac{\partial \boldsymbol{\Phi}_k}{\partial \alpha_k} = 0 \qquad \frac{\partial \mathbf{H}_k}{\partial \alpha_k} = 0$$

3.1. Statistical preliminaries

The MLE technique for parameter estimation seeks to maximize the likelihood function $L\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right)$ for a general set of n system parameters $\boldsymbol{\Theta} \in \mathbb{R}^n$, based on a set of N data samples $\mathbf{Z}_N \in \mathbb{R}^{m\times N}$, where the discrete random variable Z is defined by $\mathbf{Z}_N = \left\{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_N\right\}$, and it is assumed that the measurement vectors $\mathbf{z}_i \in \mathbb{R}^m\ \forall i = 1,\dots,N$ are Independent and Identically Distributed (IID). In the case of the Kalman filter scenario, these measurement vectors simply correspond to sets of measurements collected at sequential instances in time. The parameter set to be estimated from the EKF is defined as $\boldsymbol{\Theta} \triangleq \{\mathbf{Q}_k, \mathbf{R}_k\}$, and the log-likelihood form of the MLE problem [32] is formulated as

$$\max_{\boldsymbol{\Theta}}\left\{\ln L\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right) = \ln p\left(\mathbf{Z}_N|\boldsymbol{\Theta}\right)\right\} \tag{28}$$

where $p\left(\mathbf{Z}_N|\boldsymbol{\Theta}\right)$ represents the discrete joint Probability Mass Function (PMF) for the observations conditioned on the parameters, and the monotonicity of the logarithmic form is used to later simplify the mathematics. The observations considered here are in fact the residuals of the EKF, because the innovations and the measurements contain the same statistical information. Calling to mind the innovation property of an optimal filter [12,33], the residuals $\boldsymbol{\nu}_k$ of the EKF are assumed to be distributed as zero-mean, white Gaussian noise with covariance $\boldsymbol{\Sigma}_k$, such that $p\left(\boldsymbol{\nu}_k|\boldsymbol{\Theta}\right) = \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_k)$. Thus, the probability function of the measurements can be approximated in terms of the probability functions of the residual sequence [34] using

$$p\left(\mathbf{Z}_N|\boldsymbol{\Theta}\right) \approx \prod_{i=i_0}^{N} p\left(\boldsymbol{\nu}_N|\boldsymbol{\Theta}\right) \tag{29}$$

where the symbol $\prod(\cdot)$ represents the product operator, and i_0 is a starting index. The full PMF would require the entire set of measurements accumulated by the filter (*i.e.*, $i = 1, 2, \dots, k$), but storing the entire set of past EKF data is unrealistic for online estimation in the context of spacecraft formation flying, so instead only a fixed number of past measurements are processed. The onboard likelihood function is therefore evaluated using a window of size N, which contains data from the current point k back to point $i_0 = k - N + 1$. From the above definition, the likelihood function considered in this work is written using the standard matrix probability function for Gaussian distributions [35], which is

$$L\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right) = \prod_{i=i_0}^{N} \frac{1}{\sqrt{(2\pi)^m|\boldsymbol{\Sigma}_i|}} \exp\left[-\frac{\boldsymbol{\nu}_i^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\nu}_i}{2}\right] \tag{30}$$

The term $|\boldsymbol{\Sigma}_i|$ refers to the matrix determinant of $\boldsymbol{\Sigma}_i$, and, as a reminder, $m = 7$ is the number of measurements in each measurement vector. Taking the natural logarithm of the likelihood function eliminates the multiplicity, yielding

$$\ln L\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right) = -\frac{1}{2}\sum_{i=i_0}^{N}\left[\ln|\boldsymbol{\Sigma}_i| + \boldsymbol{\nu}_i^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{\nu}_i + c_i\right] \tag{31}$$

where the constant term $c_i = m\ln(2\pi)$ is independent of the parameters. The goal of maximizing the log-likelihood function is equivalent to minimizing the negative log-likelihood function, so a pseudo-likelihood cost function is defined by negating Eq. (31) and neglecting the constant term and coefficient:

$$J\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right) \triangleq \sum_{i=i_0}^{N}\left[\ln|\boldsymbol{\Sigma}_i| + \boldsymbol{v}_i^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{v}_i\right] \tag{32}$$

Through this reduction process, the MLE solution for the EKF parameters can be approached as a minimization of the pseudo-likelihood cost function $J\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right)$, with positive definite constraints imposed upon $\mathbf{Q}_k$ and $\mathbf{R}_k$. The next section describes one method of solving the MLE problem, following the approach used by Maybeck [32].

3.2. Analytic solution to the MLE problem

The MLE solution to the pseudo-likelihood minimization problem is constructed analytically by taking the derivative of Eq. (32) with respect to the adaptation parameters, and setting the results equal to zero. Assuming that the covariance matrices are diagonal allows this solution to be stated formally with

$$0 = \frac{\partial}{\partial\alpha_k}\left[J\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right)\right] \tag{33}$$

and the resulting partial differential equation that maximizes the likelihood function for the adaptation parameters is

$$0 = \sum_{i=i_0}^{N}\operatorname{tr}\left\{\left[\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1}\boldsymbol{v}_i\boldsymbol{v}_i^T\boldsymbol{\Sigma}_i^{-1}\right]\left[\frac{\partial\mathbf{R}_i}{\partial\alpha_k} + \mathbf{H}_i\frac{\partial\mathbf{Q}_i}{\partial\alpha_k}\mathbf{H}_i^T\right]\right\} \tag{34}$$

A full derivation of Eq. (34) can be found in Appendix C, which presents the necessary partial derivatives of the pseudo-likelihood function. At this point, it is clear that adaptations of both $\mathbf{Q}_k$ and $\mathbf{R}_k$ are possible, stemming from the fact that both the process and measurement noise covariances affect the residual sequence. Unique solutions that provide individual update laws for both the process and measurement noise covariance matrices will be investigated next.

3.3. Adaptations for the process noise covariance

The adaptation equation for the process noise covariance is found by solving Eq. (34) under the conditions that $\mathbf{R}_k$ is known and independent of the adaptation parameters, thus setting $\frac{\partial\mathbf{R}_i}{\partial\alpha_k} = 0$. Each diagonal entry of the process noise covariance is therefore treated as an adaptable term, so at each time step the adaptation parameters $\alpha_n = \mathbf{Q}_{nn}$ correspond to the nth row or column index of the process noise matrix. Eq. (34) then simplifies to

$$0 = \sum_{i=i_0}^{N}\mathbf{H}_i^T\left[\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1}\boldsymbol{v}_i\boldsymbol{v}_i^T\boldsymbol{\Sigma}_i^{-1}\right]\mathbf{H}_i \tag{35}$$

Further assuming that the Kalman gain is optimal, and postulating that any covariance terms must be positive definite symmetric, the estimated process noise covariance $\hat{\mathbf{Q}}_k$ can then be obtained after some manipulation, leading to

$$\hat{\mathbf{Q}}_k = \frac{1}{N}\sum_{i=i_0}^{N}\left[\mathbf{K}_i\boldsymbol{v}_i\boldsymbol{v}_i^T\mathbf{K}_i^T + \left(\mathbf{P}_i^+ - \boldsymbol{\Phi}_{i-1}\mathbf{P}_{i-1}^+\boldsymbol{\Phi}_{i-1}^T\right)\right] \tag{36}$$

The parenthetical term in Eq. (36) contains the change in the error covariances between the current and previous time steps. If the filter has reached steady-state operation, this term is negligible and the Kalman gain becomes constant. This is an attractive simplification in a computational sense, as less information from the previous filter steps need to be maintained in memory. The estimate of the process covariance then reduces to the form used by Mohamed and Schwarz [14]:

$$\hat{\mathbf{Q}}_k = \mathbf{K}_k\left[\frac{1}{N}\sum_{i=i_0}^{N}\boldsymbol{v}_i\boldsymbol{v}_i^T\right]\mathbf{K}_k^T \tag{37}$$

Notice the terms in brackets here contains the residuals of the filter averaged over the N-steps of the fixed window. This term is referred to here as the *observed* residuals covariance matrix, denoted $\hat{\boldsymbol{\Sigma}}_{k|N} \in \mathbb{R}^{m\times n}$, which is defined by

$$\hat{\boldsymbol{\Sigma}}_{k|N} \triangleq \frac{1}{N}\sum_{i=i_0}^{N}\boldsymbol{v}_i\boldsymbol{v}_i^T \tag{38}$$

Using this notion of the observed RCM, the Q-adaptation law for the EKF requires only knowledge of the current Kalman gain and the past N vectors of observed residuals from the filter, with

$$\hat{\mathbf{Q}}_k = \mathbf{K}_k\hat{\boldsymbol{\Sigma}}_{k|N}\mathbf{K}_k^T \tag{39}$$

3.4. Adaptations for the measurement noise covariance

Solving for the measurement noise adaptations uses the same procedure as above, except where $\mathbf{Q}_k$ is designated as the known and adaptation-independent parameter. Setting $\frac{\partial\mathbf{Q}_i}{\partial\alpha_k} = 0$ and choosing the adaptation terms as $\alpha_n = \mathbf{R}_{nn}$, the derivatives in Eq. (34) yield

$$0 = \sum_{i=i_0}^{N}\operatorname{tr}\left\{\boldsymbol{\Sigma}_i^{-1}\left[\boldsymbol{\Sigma}_i - \boldsymbol{v}_i\boldsymbol{v}_i^T\right]\boldsymbol{\Sigma}_i^{-1}\right\} \tag{40}$$

Invariance of the estimation process is assumed for the given memory window, implying that $\boldsymbol{\Sigma}_i^{-1}$ is approximately constant throughout the filtered data in memory [34,36]. This allows an explicit estimate of the measurement noise covariance to be calculated with

$$\hat{\mathbf{R}}_k = \frac{1}{N}\sum_{i=i_0}^{N}\left[\boldsymbol{v}_i\boldsymbol{v}_i^T - \mathbf{H}_i\mathbf{P}_i^-\mathbf{H}_i^T\right] \tag{41}$$

An alternate variation of this R-adaptation law can be derived by using several Kalman filter relations, as implementation of Eq. (41) has been shown to give poor estimates of $\mathbf{R}_k$ that break the positive definite constraint. The necessary substitutions are shown in Appendix C, and give the estimated measurement noise covariance as

$$\hat{\mathbf{R}}_k = \frac{1}{N}\sum_{i=i_0}^{N}\left[\boldsymbol{v}_i^+\boldsymbol{v}_i^{+T} + \mathbf{H}_i\mathbf{P}_i^+\mathbf{H}_i^T\right] \tag{42}$$

Notice that in the equation above, the pre-fit residuals have been replaced with the post-fit residuals $\boldsymbol{v}_i^+$, defined as $\boldsymbol{v}_i^+ \triangleq \mathbf{z}_i - \hat{\mathbf{z}}_i^+$. Post-fit residuals are calculated from corrected measurement estimates $\hat{\mathbf{z}}_i^+$, using the measurement equation evaluated at the corrected state estimate $\hat{\mathbf{x}}_i^+$.

3.5. MLE adaptations with intrinsic smoothing

Typically, the EKF acts as a recursive scheme dependent only on estimates from the previous time step, but incorporating a memory characteristic into the filter can improve estimates of the state and error covariances by harnessing information contained in the entire available data set. Since the MLE adaptation laws already require a fixed-length memory window, a nonlinear variation of the Kalman smoother [34] is appended to the MLE-AEKF algorithm. The smoother runs backwards through a window of size N, which contains the set of data points from $i_0 = k - N + 1$ to the latest point k. Initial smoothed estimates are taken from the filter at the current time step as $\hat{\mathbf{x}}_{N|N} = \hat{\mathbf{x}}_k^+$ and $\mathbf{P}_{N|N} = \mathbf{P}_k^+$, where the smoothed state estimate at time t_k given N past data points is denoted by $\hat{\mathbf{x}}_{k|N}$. Using these initial conditions, the smoothed state and error covariances within the memory window for data from $k = N-1, N-2, \ldots, i_0$ are calculated using

$$\mathbf{G}_{k-1} = \mathbf{P}_{k-1}^+\boldsymbol{\Phi}_{k-1}^T\left(\mathbf{P}_k^-\right)^{-1} \tag{43}$$

$$\hat{\mathbf{x}}_{k-1|N} = \hat{\mathbf{x}}_{k-1}^+ + \mathbf{G}_{k-1}\left(\hat{\mathbf{x}}_{k|N} - \hat{\mathbf{x}}_k^-\right) \tag{44}$$

$$\mathbf{P}_{k-1|N} = \mathbf{P}_{k-1}^+ + \mathbf{G}_{k-1}\left(\mathbf{P}_{k|N} - \mathbf{P}_k^-\right)\mathbf{G}_{k-1}^T \tag{45}$$

where $\mathbf{G}_k \in \mathbb{R}^{n\times m}$ is a smoothing gain derived as the MLE estimator for the current state estimates given the entire set of measurements available [37]. The purpose of the EKF Smoother in the context of

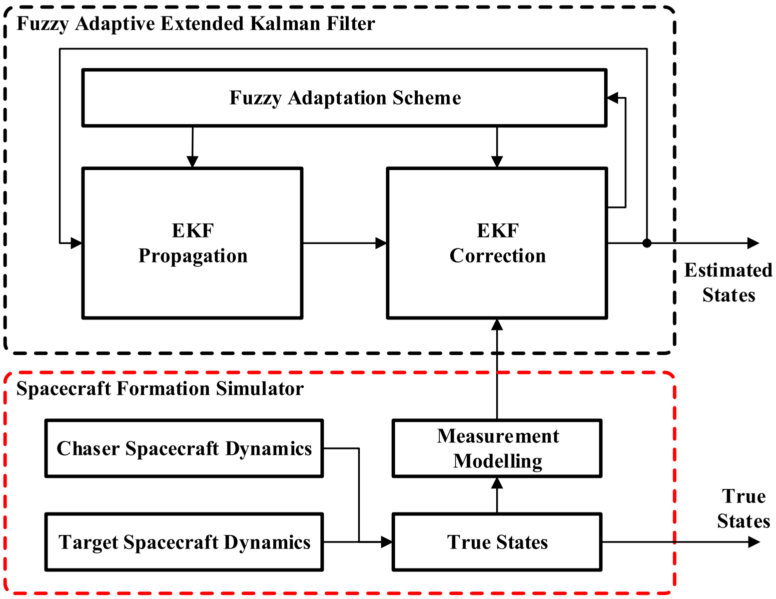


Fig. 2. A block diagram representation of the FAEKF simulation.

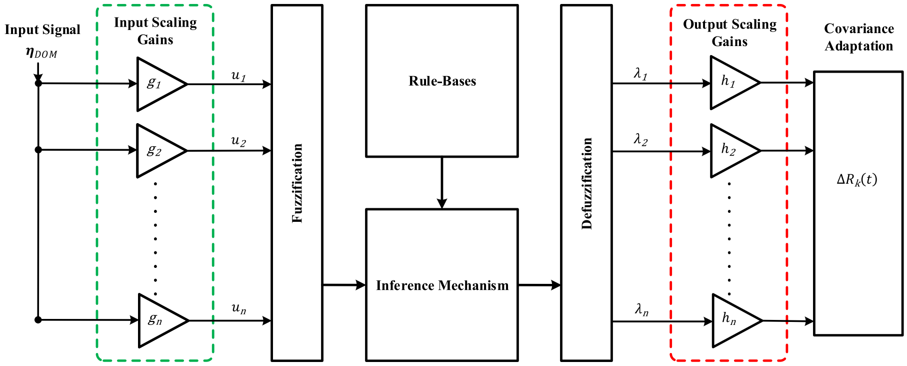


Fig. 3. Block diagram representing the fuzzy adaptation scheme.

this work, is to develop smoothed data within the MLE adaptation scheme that will improve the estimates of the $\mathbf{Q}_k$ or $\mathbf{R}_k$ matrices. For the Q-adaptation equation, the estimate of $\mathbf{Q}_k$ can be improved by using the smoothed state estimates in the calculation of the residuals. Constructing a smoothed estimate of the measurements via $\hat{\mathbf{z}}_{i|N} = \mathbf{H}_{k|N}\hat{\mathbf{x}}_{k|N}$, where the smoothed linearized measurement matrix $\mathbf{H}_{k|N}$ is evaluated using the smoothed state estimates. The final process noise covariance adaptation is then updated from Eq. (37) as

$$\hat{\mathbf{Q}}_k = \mathbf{K}_k \left[\frac{1}{N} \sum_{i=i_0}^{N} \left(\mathbf{z}_i - \hat{\mathbf{z}}_{i|N}\right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_{i|N}\right)^T \right] \mathbf{K}_i^T \tag{46}$$

Again to capitalize on the smoothing process used earlier, the smoothed measurements and covariances are inserted into the R-adaptation scheme as well. Thus, the final adaptation for $\mathbf{R}_k$ is obtained by modifying Eq. (42) accordingly:

$$\hat{\mathbf{R}}_k = \frac{1}{N} \sum_{i=i_0}^{N} \left[\left(\mathbf{z}_i - \hat{\mathbf{z}}_{i|N}\right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_{i|N}\right)^T + \mathbf{H}_{i|N}\mathbf{P}^{+}_{i|N}\mathbf{H}^{T}_{i|N} \right] \tag{47}$$

4. Adaptive Kalman filtering using Fuzzy logic

The design of a fuzzy adaptive extended Kalman filter (FAEKF) is presented in this section, providing a second method for adapting the EKF for relative navigation of spacecraft formations. Adaptations to the process and measurement noise covariances are calculated by a fuzzy logic system (FLS) after the correction phase of the EKF, and the block diagram in Fig. 2 illustrates the FAEKF architecture and the navigation simulation stateflow.

The fuzzy adaptation block shown above in Fig. 2 calculates updates to the noise covariance matrices using the fuzzy adaptation scheme shown in Fig. 3. Each following subsection investigates the components of the designed fuzzy system, particularly the choice of inputs, fuzzification, rule bases, inference techniques, and defuzzification methods.

4.1. Innovation-based covariance matching

Whereas the previous MLE-AEKF sought to obtain closed-form solutions that maximized the likelihood function of the EKF based on the covariance of the residuals, the FAEKF developed here uses a covariance matching method to adapt the process and measurement noise covariances. For this section, the theoretical RCM $\boldsymbol{\Sigma}_k$ and the observed RCM $\hat{\boldsymbol{\Sigma}}_{k|N}$ serve as the basis for developing metrics to adapt the EKF, as the proposed fuzzy adaptive update laws are based on the discrepancies between the actual observed filter covariance and the theoretical covariance bounds suggested by the filter. The first metric selected for adapting the EKF is the Degree of Mismatch (DOM), denoted by $\boldsymbol{\eta}_{DOM} \in \mathbb{R}^{m\times m}$ [25,38]. Representing the element-by-element difference between the predicted filter RCM the experimental RCM sampled from the filter output, the DOM is defined as

$$\boldsymbol{\eta}_{DOM} \triangleq \boldsymbol{\Sigma}_k - \hat{\boldsymbol{\Sigma}}_{k|N} \tag{48}$$

Following the idea of the DOM metric, the second parameter used to detect sub-optimal performance of the filter is the scalar Degree of Divergence (DOD), which will be denoted by $\eta_{DOD} \in \mathbb{R}$ [38,39]. As an indicator of overall divergence of the filter, a large DOD value signifies the filter is providing poor state estimates, where η_{DOD} is defined by

$$\eta_{DOD} \triangleq \mathrm{tr}[\hat{\boldsymbol{\Sigma}}_{k|N}] - \mathrm{tr}[\boldsymbol{\Sigma}_k] = -\,\mathrm{tr}[\boldsymbol{\eta}_{DOM}] \tag{49}$$

4.1.1. Adaptation of the process noise covariance

For situations where noise in the system dynamics is variable or poorly defined, adaptation of $\hat{\mathbf{Q}}_k$ is achieved using the DOD as the input to a single-input single-output (SISO) fuzzy logic system. From the spacecraft dynamics and measurement models, it can be seen that only partial observation of the states is made; more colloquially, 7 measurements of the system are available, but there are 10 states to be estimated. Implicitly this means the dimensions of the process noise covariance and the residuals covariance matrices are not equivalent, and a direct correlation between the two covariances is not apparent. Using the scalar DOD as an input to the FLS however, elements of the process noise covariance are adjusted by applying a single scaling factor $\varepsilon_k^q \in \mathbb{R}$, via the update law

$$\hat{\mathbf{Q}}_{k+1} \triangleq \varepsilon_k^q \hat{\mathbf{Q}}_k \tag{50}$$

where the covariance scale factor is given by

$$\varepsilon_k^q \triangleq 1 + h^q \lambda_k^q \tag{51}$$

Here the scale factor ε_k^q is the product of the FLS output λ_k^q and an output gain h^q. The output gain is an FLS design parameter and should be selected such that $h^q \lambda_k^q > -1 \ \forall k$, to maintain a positive definite symmetric $\hat{\mathbf{Q}}_k$.

4.1.2. Adaptation of the measurement noise covariance

Adaptations of the measurement noise covariance $\mathbf{R}_k$ are made using the degree of mismatch $\boldsymbol{\eta}_{DOM}$ as the input to the FLS. The residual covariance and the measurement noise covariance matrices have the same dimension, such that each element in the DOM matrix directly corresponds to an element in the noise covariance matrix. Taking advantage of this relationship, and assuming that the covariances are diagonal, each element of $\mathbf{R}_k$ is adapted individually using a scale factor $\varepsilon_k^r \in \mathbb{R}$. The update law for the measurement noise covariance matrix is thus defined as

$$\hat{\mathbf{R}}_{k+1}(i,i) \triangleq \varepsilon_k^r(i)\mathbf{R}_k(i,i) \tag{52}$$

where the R-scale factor $\varepsilon_k^r(i)$ corresponds to the ith diagonal element of the noise covariance matrix. Each scale factor is a function of an output gain $h^r(i)$ and the output of the FLS $\lambda_k^r(i) \in \mathbb{R} \ \forall i = 1, \ldots, 7$, through the relation

$$\varepsilon_k^r(i) \triangleq 1 + h^r(i)\lambda_k^r(i) \tag{53}$$

As in the previous case, the FLS design for the measurement noise covariance adaptations must ensure $h^r \lambda_k^r > -1 \ \forall k$, such that $\hat{\mathbf{R}}_k$ is always positive definite. For both of FAEKF process and measurement noise adaptation schemes shown above, it can be seen that the system reduces to that of the standard Kalman filter when the output of the FLS is zero, since the covariance scale factors become unity.

4.2. Fuzzification

The adaptation parameters λ_k^q and λ_k^r are calculated online by the FLS, in order to match the observed performance of the EKF with the predicted optimal performance of the filter. The first process within the FLS is fuzzification, which maps crisp inputs from either the DOD or DOM metrics into corresponding fuzzy sets. The following section specifies the gains used to scale the inputs and outputs of the FLS, the linguistic variables used to describe the inputs and outputs, and the membership functions used to assign the degree of membership to the crisp values. The fuzzy logic rules are also introduced, and the methodology behind their role in the covariance matching technique is given. All operations here take place at the current time step t_k, so the subscript k is implicit and omitted to condense the notation.

4.2.1. For the process noise covariance

The process noise covariance $\mathbf{Q}_k$ is adapted using a single SISO fuzzy system. Time-varying crisp inputs to the FLS are represented by $u_i \in \mathcal{U}_i$, where $\mathcal{U}_i$ defines the controllable universe of discourse for the ith input. The degree of divergence η_{DOD} is used as the input to the FLS here, and the universe of discourse for the DOD is defined as ± 200 m^2. In this work, input scaling gains g_i are selected to normalize the crisp inputs onto $\mathcal{U}_i = [-1 \ \ 1\,]$. Constant *linguistic variables* $\tilde{u}_i$ are used to describe each crisp input u_i, and for the single input of the Q-adaptations, $\tilde{u}_1 =$ *the degree of divergence of the residuals covariance matrix* . The set of linguistic values that can be assigned to the input linguistic variables are defined as

$$\tilde{A}_i = \left\{ \tilde{A}_i^j \ : \ i = n = 1 \ : \ j = 1, \ldots, N_i = 5 \right\} \tag{54}$$

where $\tilde{A}_i^j$ is the jth linguistic value of linguistic variable $\tilde{u}_i$, and the number of input linguistic values is $N_i = 5$. For the outputs, $\tilde{B}_q^p$ is the pth linguistic value of output linguistic variable $\tilde{\lambda}_q$. Since the Q-adaptations only use a single FLS, the number of output linguistic values is $M_q = 1$, and the set of linguistic values is

$$\tilde{B}_q = \left\{ \tilde{B}_q^p \ : \ q = m = 1 \ : \ p = M_q = 1 \right\} \tag{55}$$

The input linguistic variables used for the Q-adaptations are also used subsequently for the R-adaptations, and have been named as: "negative high (NH), negative low (NL), zero (ZERO), positive low (PL)", and "positive high (PH)", and the output linguistic variables are again taken as "negative maximal (NMAX), negative minimal (NMIN), zero (ZERO), positive minimal (PMIN)" and "positive maximal (PMAX)".

4.2.2. For the measurement noise covariance

Adaptations for the measurement noise covariance $\mathbf{R}_k$ are found using seven SISO fuzzy systems, so the FLS input and output indices are $i = 1, 2, \ldots, n = 7$ and $q = 1, 2, \ldots, m = 7$, respectively. The input to the ith FLS is the corresponding diagonal element of the degree of mismatch matrix $\boldsymbol{\eta}_{DOM}$, and the controllable universe of discourse for each of the DOM components are defined to be ± 20 m^2 for the position mismatch, ± 1 m^2/s^2 for the velocity mismatch, and ± 0.1 rad^2 for the true anomaly mismatch. Values outside of these ranges result in a saturated input signal, which means that the output of the FLS is identical for any values outside the controllable universe of discourse.

Constant *linguistic variables* $\tilde{u}_i$ are used to describe the time-varying crisp inputs u_i, and the linguistic variables for the inputs are defined as follows: $\tilde{u}_1, \tilde{u}_2, \tilde{u}_3 =$ *Degree of mismatch for the x, y, z-position components*, $\tilde{u}_4 =$ *Degree of mismatch for the true anomaly θ component*, $\tilde{u}_5, \tilde{u}_6, \tilde{u}_7 =$ *Degree of mismatch for the x, y, z-velocity components*. The set of linguistic values assigned to the input linguistic variables are defined as

$$\tilde{A}_i = \left\{ \tilde{A}_i^j \ : \ i = 1, \ldots, n = 7 \ : \ j = 1, \ldots, N_i = 5 \right\} \tag{56}$$

where the number of input linguistic values is again $N_i = 5$. An equivalent definition can be made for the outputs, where the given number of output linguistic values is $M_q = 7$, and the set of linguistic values is

$$\tilde{B}_q = \left\{ \tilde{B}_q^p \ : \ q = 1, \ldots, m = 7 \ : \ p = 1, \ldots, M_q = 7 \right\} \tag{57}$$

4.2.3. Membership functions

The nonlinear FLS mapping from the crisp inputs u_i to the crisp outputs λ_q requires that the inputs first be transformed into their respective fuzzy sets, denoted by A_i^j. The fuzzy sets characterize how much the particular value of an input meets the criteria defined by each of the relevant linguistic values, and this characterization is defined in

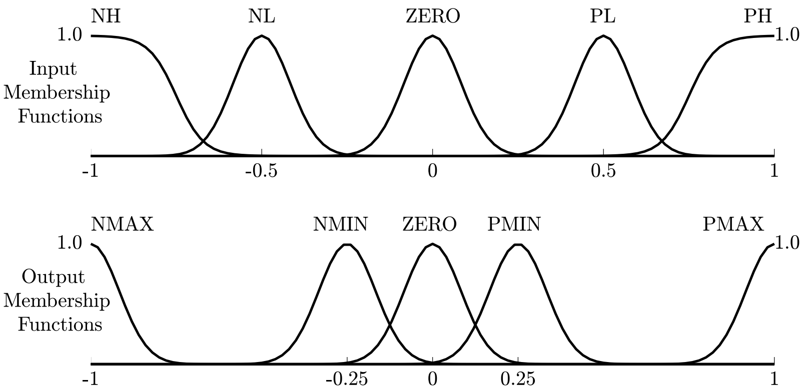


Fig. 4. Membership functions for the normalized input and output variables.

terms of the membership function $\mu_{A_i^j}(u_i)$. The membership function maps the input onto the universe of discourse $\mathcal{U}_i^{\mu} \mapsto [0,1]$, whereby a high the degree of membership for a given linguistic value intuitively indicates that the linguistic value accurately describes the observed input. Mathematically, the fuzzy sets for the input are defined as

$$A_i^j = \left\{ \ \Big(u_i, \mu_{A_i^j}(u_i)\Big) \ : \ u_i \in \mathcal{U}_i \ \right\} \qquad i = 1, \dots, n = 7 \quad j = 1, \dots, N_i = 5 \tag{58}$$

A combination of Sigmoidal and Gaussian curves are used to model the input membership functions, which ensures all fuzzified inputs have non-zero membership over the span of the universe of discourse. The Sigmoid functions are used to saturate the extrema of the universe of discourse, preventing inputs outside the effective control range from yielding net-zero memberships. Thus, the two outermost input membership functions (*i.e.*, $j = 1, 5$) are written as

$$\mu_{A_i^j}(u_i;\ a_{A_i^j}, b_{A_i^j}) = \frac{1}{1 + \exp\left[-a_{A_i^j}(u_i - b_{A_i^j})\right]} \tag{59}$$

for $i = 1, \dots, 7$, where $a_{A_i^j} \in \mathbb{R}$ defines the curvature of the function and $b_{A_i^j} \in \mathbb{R}$ designates the centre of the function (*i.e.*, where the function crosses 0.5). The remaining inputs for $j = 2, 3, 4$ are classified using Gaussian membership functions centred at $c_{A_i^j}$ with widths of $\sigma_{A_i^j}$, so for $i = 1, \dots, 7$, the inner membership functions can be written

$$\mu_{A_i^j}(u_i;\ c_{A_i^j}, \sigma_{A_i^j}) = \exp\left[\frac{-(u_i - c_{A_i^j})^2}{2\sigma^2_{A_i^j}}\right] \tag{60}$$

The output membership functions $\mu_{B_i^j}$ are constructed using equivalent mathematical definitions, all of which are modelled using Gaussian curves. Fig. 4 shows the resulting normalized input and output membership functions, where each curve is defined with 100 points spread equidistant along the respective universe of discourse. The spacecraft formation FAEKF parameters selected for the Sigmoidal membership functions are $a_{A_i^j} = \{-0.75, 0.75\}$ and $b_{A_i^j} = 25$, and the Gaussian membership functions are centred at $c_{A_i^j} = \{-0.5, 0, 0.5\}$ and $c_{B_i^j} = \{-1, -0.25, 0, 0.25, 1\}$, with widths of $\sigma_{A_i^j} = \sigma_{B_i^j} = 1/12$. The clustering of the output membership functions is used to provide more precise control authority to the outputs as the inputs approach zero [40]; this yields larger adaptations when the DOM or DOD parameters are large, and finer adaptations as these parameters approach the ideal value. Both the $\mathbf{Q}_k$ and the $\mathbf{R}_k$ FAEKF adaptation schemes use the membership functions described above.

4.3. Logic rule bases

The fuzzy logic framework can be used to provide highly specialized control and adaptation laws specific to an application, since the designer is effectively able to code human knowledge and reasoning into rule bases for the FLS. Using the linguistic variables introduced previously, governing IF-THEN rules for the FLS are written in terms of a premise and a consequent. For this work, with the inputs corresponding to the premises and the outputs corresponding to the consequents, the rules are written as:

IF $\tilde{u}_1$ is PL **THEN** λ_1 is PMIN
IF $\tilde{u}_4$ is ZERO **THEN** λ_4 is ZERO
IF $\tilde{u}_7$ is NMAX **THEN** λ_7 is NH

and so forth. These rules are a large design consideration when customizing the fuzzy system, and have been chosen to align with the innovation-based covariance matching methodology. As an example, if the degree of mismatch for a particular covariance element is large, this implies the theoretical RCM is larger than the observed RCM, and the theoretical noise covariance should be decreased. With this in mind, the linguistic rules are defined such that the output of the FLS will adapt the noise covariance matrix accordingly, and Table 1 summarizes the rule bases used for the fuzzy system. There are $n_r = 5$ rules for the table shown below, so the rule base index can be defined as $i_r = 1, \dots, 5$.

Table 1
Rule table for the SISO Fuzzy logic system.

Inputs $\{\tilde{u}_i\}$	**NH**	**NL**	**ZERO**	**PL**	**PH**
Outputs $\{\tilde{\lambda}_q\}$	NMAX	NMIN	ZERO	PMIN	PMAX

4.4. The inference mechanism

After successfully mapping the inputs into their respective fuzzy sets, and defining rule bases that contain the logical relations to be maintained by the FLS, the process of inference is used to determine how likely the input corresponds to a particular output. Since each of the rules applied here involves a single premise only, and assuming singleton fuzzification, the degree of certainty to which the i_r^{th} rule applies for the ith input is simply given by the membership function of the input evaluated for that rule. Hence, $\mu_{i_r}(u_i) = \mu_{A_i^j}(u_i)$ for $i_r = 1, \dots, 5$ for each input. The inference stage is then completed using the MIN operator, whereby the output membership functions $\mu_{B_q^i}(\lambda_q)$ of the implied fuzzy sets B_q^i are evaluated over the range of inputs, outputs, and logic rules, as

$$\mu_{B_q^{i_r}}(\lambda_q) = \min\left[\mu_{i_r}(u_i),\ \mu_{B_q^p}(\lambda_q)\right] \tag{61}$$

Much like the input membership functions, the implied fuzzy set B_q^i quantifies the level of membership the output has for a specific crisp output, albeit only taking into account the i_r^{th} rule. The MIN operator identifies the intersection of the two fuzzy membership functions within the argument, and therefore ensures that uncertainties in the premises (through $\mu_{A_i^j}$) limit the confidence of the conclusions from the inference process. Implied fuzzy sets are implemented here, as aggregated implied fuzzy sets were found to be computationally burdensome without improving performance.

4.5. *Defuzzification*

The defuzzification process calculates crisp values for the outputs $\lambda_q \in \Lambda_q$ from within their appropriate universe of discourse Λ_q, and the centre of gravity (COG) method for implied fuzzy sets [41] is used here. COG defuzzification gives the crisp outputs for $q = 1, \dots, m$ as

$$\lambda_q = \frac{\sum_{i_r=1}^{n_r} b_{i_r}^q \int_{\Lambda_q} \mu_{B_q^{i_r}}(\lambda_q)\,\mathrm{d}\lambda_q}{\sum_{i_r=1}^{n_r} \int_{\Lambda_q} \mu_{B_q^{i_r}}(\lambda_q)\,\mathrm{d}\lambda_q} \tag{62}$$

where n_r is the number of linguistic rules for the adaptations, and $b_{i_r}^q \in \mathbb{R}$ is the centre of area of the membership function of the output fuzzy set B_q^p, which corresponds to the implied fuzzy set $B_q^{i_r}$ for the i_r^{th} rule. The crisp normalized outputs are lastly re-dimensionalized through multiplication with the appropriate scaling gains. The universe of discourse for the normalized outputs λ_q spans $\Lambda = \begin{bmatrix} -1 & 1 \end{bmatrix}$, and output scaling gains h_i transform the normalized outputs back into dimensional space. Once the crisp fuzzy output has been un-normalized, the scale factors for $\mathbf{Q}_k$ and $\mathbf{R}_k$ are calculated with Eqs. (51) and (53), and the final adaptations to the noise covariance matrices are made through Eqs. (50) and (52).

5. Simulation and results

Numerical simulations of two near-Earth spacecraft are used to assess the formation navigation accuracy of the proposed MLE-AEKF and FAEKF algorithms. The following section provides an overview of the simulation environment and introduces the spacecraft formations being modelled, after which the numerical results from the simulations are presented. Relative position and velocity estimates obtained from the adaptive EKFs are compared with the measurement-only and classical EKF relative navigation solutions, followed by a discussion of the results.

5.1. *The formation flying navigation simulator*

The true states of both the target and the chaser spacecraft are simulated using a real-world orbit propagator, which models the dynamics of each spacecraft individually in the inertial frame. The propagator outputs the ECI position states $\mathbf{r}_t, \mathbf{r}_c \in \mathbb{R}^3$ and velocity states $\mathbf{v}_t, \mathbf{v}_c \in \mathbb{R}^3$ for the target and chaser respectively, and the relative motion between the spacecraft is then evaluated by differencing these absolute states. A coordinate transformation based on the LVLH frame definition is then applied to map the relative states from the ECI frame into the LVLH frame fixed to the target, using standard kinematics relations [42].

For the following scenarios, the inertial force models use the standard two-body equations of motion [31], expanded to include perturbations due to the oblateness of the Earth (J_2), atmospheric drag, solar radiation pressure, and luni-solar third body gravity. Details about the mathematical modelling of these perturbations can be found in the text by Montenbruck [43], and all simulations are completed within the MATLAB-Simulink software environment. Numerical integration for the orbit propagator is performed using Simulink integrator blocks configured with the `ODE4` fixed time-step solver with a step size of 1 s, with absolute and relative error tolerances set at 10^{-6}.

Whereas the states output by the orbit propagator are used to represent the true state of the formation, the EKF only receives information about the spacecraft through measurements. Hence, simulated measurements of the spacecraft formation are modelled by first corrupting the inertial target and chaser states with zero-mean, white Gaussian noise, via

$$\mathbf{r}_t^m = \mathbf{r}_t + \mathcal{N}\left(\mathbf{0}, \boldsymbol{\sigma}_r\right) \qquad \mathbf{r}_c^m = \mathbf{r}_c + \mathcal{N}\left(\mathbf{0}, \boldsymbol{\sigma}_r\right) \tag{63}$$

$$\mathbf{v}_t^m = \mathbf{v}_t + \mathcal{N}\left(\mathbf{0}, \boldsymbol{\sigma}_v\right) \qquad \mathbf{v}_c^m = \mathbf{v}_c + \mathcal{N}\left(\mathbf{0}, \boldsymbol{\sigma}_v\right) \tag{64}$$

where the superscript m denotes measured quantities. For all scenarios tested in this research, the standard deviation of the position and velocity noises, $\boldsymbol{\sigma}_r \in \mathbb{R}^3$ and $\boldsymbol{\sigma}_v \in \mathbb{R}^3$ respectively, are treated as equal and spatially-independent noise processes that corrupt each component of the measurement, so $\boldsymbol{\sigma}_r = \begin{bmatrix} \sigma_r & \sigma_r & \sigma_r \end{bmatrix}^T$ and $\boldsymbol{\sigma}_v = \begin{bmatrix} \sigma_v & \sigma_v & \sigma_v \end{bmatrix}^T$. These resulting noisy ECI states are then used to calculate the ECI-to-LVLH rotation matrix, and the noise-corrupted relative position and velocity components are transformed into the LVLH frame. The resulting LVLH components, with the embedded random noise, are finally passed to the EKF through the measurement vector defined previously in Eq. (12).

5.2. *Quantifying performance of the navigation filters*

Comparisons between the various filter algorithms are based on the resulting estimation errors from the simulations, which define the difference between the true state of the formation and the state estimated by the filter. Recall that the estimation error $\mathbf{e}_k$ is calculated as $\mathbf{e}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k$, and note that the error covariance bounds of the Kalman filter are also useful for demonstrating that the errors stay within the covariance bounds predicted by the filter covariance $\mathbf{P}_k$. The standard deviation σ_k of the ith state is calculated from the corresponding ith diagonal element of the state error covariance matrix of the filter, so at a given time t_k, the standard deviation is

$$\sigma_k(i) = \sqrt{\mathbf{P}_k(i,i)} \tag{65}$$

Both the positive and negative 3σ-bounds of the filter are plotted when considering the estimation errors, thereby illustrating the range of errors that the filter estimates should fall between with 99% certainty. In addition, the three-dimensional Root-Mean-Square (3D-RMS) error is utilized as a scalar metric to quantify the overall filter accuracy. The position and velocity 3D-RMS errors are calculated over the past N data points from the filter output using the equations below, where it is important to note that x, y and z used here indicate the specific LVLH states:

$$\text{Position 3D-RMS} \quad e_{\rho|N}^{RMS} = \sqrt{\frac{1}{N}\sum_{j=j_0}^{n}\left[\left(x_j - \hat{x}_j\right)^2 + \left(y_j - \hat{y}_j\right)^2 + \left(z_j - \hat{z}_j\right)^2\right]} \tag{66}$$

$$\text{Velocity 3D-RMS} \quad e_{\dot{\rho}|N}^{RMS} = \sqrt{\frac{1}{N}\sum_{j=j_0}^{n}\left[\left(\dot{x}_j - \hat{\dot{x}}_j\right)^2 + \left(\dot{y}_j - \hat{\dot{y}}_j\right)^2 + \left(\dot{z}_j - \hat{\dot{z}}_j\right)^2\right]} \tag{67}$$

The number of data points used in the calculation of the RMS error is $N = n - j_0 + 1$, where n represents the total number of data points collected in the simulation, and j_0 represents the starting point for the RMS calculation. Since the initial portion of the EKF estimation requires a period of convergence to the true states, the starting point of the RMS calculation for the analyses here consider only the data from the second orbit of the two-orbit simulations; this implies that the EKF has converged to the true state trajectory after one orbital period.

The following subsections contain the analyses of two spacecraft formation scenarios. First, the PRISMA formation will be considered, followed by a generic projected elliptical orbit (PEO) formation. Each section presents the relevant orbital parameters, spacecraft characteristics, and Kalman filter settings, along with the resulting data analysis. As a performance benchmark, past works have identified that the relative position accuracy should be at least two orders of magnitude lower than the separation of the spacecraft, based on current GNC system performances [44].

Table 2
Initial orbital elements of the PRISMA formation.

Parameter	Target (TANGO)	Chaser (MANGO)
True Anomaly θ	358.903 490 28°	0.00°
Semi-Major Axis a	7087.297 556 km	7087.297 677 km
Eccentricity e	0.001 454 43	0.001 459 08
Inclination i	98.185 286 13°	98.184 666 76°
RAAN Ω	189.891 384 5°	189.890 860 2°
Argument of Perigee ω	1.097 451 382°	0.00°

Table 3
Spacecraft properties for the PRISMA formation.

Parameter	Target (TANGO)	Chaser (MANGO)
Mass, m	42.5 kg	154.4 kg
Drag Area, A_d	0.38 m^2	2.75 m^2
Drag Coefficient, C_d	2.25	2.5
SRP Area, A_r	0.55 m^2	2.5 m^2
SRP Coefficient, C_r	1.2 m^2	1.32

Table 4
Initial EKF settings for the PRISMA formation.

Initial States $\mathbf{x}_0$	$x_0 = -54.72$ m	$\dot{x}_0 = 210.73$ mm/s
	$y_0 = -86.82$ m	$\dot{y}_0 = 71.70$ mm/s
	$z_0 = 45.99$ m	$\dot{z}_0 = 83.19$ mm/s
	$\theta_0 = 358.9$ deg	$\dot{\theta}_0 = 0.0608$ deg/s
	$r_{t_0} = 7077.04$ km	$\dot{r}_{t_0} = -0.21$ m/s
Initial State Error Covariance $\mathbf{P}_0$	$\sigma_x^2 = 100$ m^2	$\sigma_{\dot{x}}^2 = 1$ m^2/s^2
	$\sigma_y^2 = 100$ m^2	$\sigma_{\dot{y}}^2 = 1$ m^2/s^2
	$\sigma_z^2 = 100$ m^2	$\sigma_{\dot{z}}^2 = 1$ m^2/s^2
	$\sigma_\theta^2 = 1$ deg^2	$\sigma_{\dot{\theta}}^2 = 0.01$ deg^2/s^2
	$\sigma_{r_t}^2 = 10\ 000$ m^2	$\sigma_{\dot{r}_t}^2 = 100$ m^2/s^2
Initial Process Noise Covariance $\mathbf{Q}_0$	$q_x = 0.2$ m^2	$q_{\dot{x}} = 5 \times 10^{-3}$ m^2/s^2
	$q_y = 0.2$ m^2	$q_{\dot{y}} = 5 \times 10^{-3}$ m^2/s^2
	$q_z = 0.2$ m^2	$q_{\dot{z}} = 5 \times 10^{-3}$ m^2/s^2
	$q_\theta = 1 \times 10^{-3}$ deg^2	$q_{\dot{\theta}} = 1 \times 10^{-6}$ deg^2/s^2
	$q_{r_t} = 5 \times 10^{-3}$ m^2	$q_{\dot{r}_t} = 5 \times 10^{-5}$ m^2/s^2
Initial Measurement Noise Covariance $\mathbf{R}_0$	$r_x = 20$ m^2	$r_{\dot{x}} = 0.5$ m^2/s^2
	$r_y = 20$ m^2	$r_{\dot{y}} = 0.5$ m^2/s^2
	$r_z = 20$ m^2	$r_{\dot{z}} = 0.5$ m^2/s^2
	$r_\theta = 0.01$ deg^2	

Table 5
3D-RMS errors from PRISMA simulation.

Method	Position (cm) [% Error][a]	Velocity (cm/s) [% Error][a]	Relative Runtime
Measurements	299.10 [2.67%]	7.42 [32.43%]	N/A
Standard EKF	86.20 [0.77%]	3.76 [16.44%]	1.00
Q-MLE-AEKF	23.89 [0.21%]	0.10 [0.44%]	2.29
R-MLE-AEKF	105.11 [0.94%]	6.91 [30.21%]	2.47
QR-MLE-AEKF	48.60 [0.43%]	0.63 [2.75%]	2.96
Q-FAEKF	26.46 [0.24%]	0.10 [0.44%]	1.25
R-FAEKF	86.85 [0.77%]	3.12 [13.63%]	1.29
QR-FAEKF	61.58 [0.55%]	1.76 [7.70%]	1.30

[a] The position and velocity percent errors are calculated relative to the minimum distance and speed between the spacecraft, respectively.

5.3. Relative navigation analysis for the PRISMA formation

The PRISMA formation consists of two spacecraft, the target (named TANGO), and the chaser (named MANGO). The formation operates in low-Earth, Sun-synchronous orbit, and details of the orbital characteristics [45] and spacecraft properties [46] can be found in Tables 2 and 3, respectively. The orbits of both spacecraft have small eccentricities, with an operational altitude band ranging from 668 km to 749 km, and a relatively short orbital period of 99 min. Due to the LEO environment, both spacecraft are subjected to the effects of atmospheric drag and solar radiation pressure (SRP); this is suitable for demonstrating the perturbation models that are used within the real-world orbit propagator and the adaptability of the AEKF methods, because the onboard dynamics model is of lower fidelity than the real-world model. Fig. 5 shows the relative motion resulting from the orbit configuration of PRISMA, where the chaser spacecraft follows a bounded relative orbit about the target that is offset in the radial direction. The separation between spacecraft is on the order of 120 m, which means acceptable 1% position errors are on the level of one metre.

5.3.1. PRISMA simulation conditions

The simulation duration is twice the orbital period of the target spacecraft, and a sampling frequency of 1 Hz is used for processing the measurements. The random zero-mean Gaussian noise added to the measurements is based on accuracy specifications of the NovAtel FlexPak6 GNSS Receiver,[1] where the standard deviations for the position and velocity noise processes are $\sigma_r = 1.2$ m and $\sigma_v = 0.03$ m/s, respectively. It is critical to reiterate these noises are added to measurements in the inertial frame, not the LVLH frame; consequently the noise characteristics of the filter cannot be expected to directly match the added noise, due to the transformation between the ECI and LVLH frames. Initial conditions for the EKF states and covariances are listed on the following page in Table 4.

Within the filter, the relative position states of the formation are initialized to values that constitute initial state errors of approximately 3% of the amplitude of the respective state. For the MLE-AEKF, the moving-window size was selected as $N = 30$ to provide reasonable smoothing performance without excessively straining the computational requirements. Similarly in the FAEKF, the averaging of residuals uses the past $N = 30$ observations. This value of N was selected empirically, based on knowledge that N must be large enough to smooth the data but small enough capture the dynamical environment throughout the spacecraft orbit (*i.e.*, if a large value of N encompasses an entire orbit, the periodic nature of the relative dynamics would be averaged to roughly zero, while a very small N provides negligible smoothing). Selecting the same value of N for both the MLE-AEKF and FAEKF also provides reasonable grounds for comparing the computational costs of each method. The input scaling gain for the Q-adaptations using the DOD is selected as $g^q = -5 \times 10^{-3}$, and the output scaling gain is $h^q = 10^{-3}$. Corresponding gains for the R-adaptations are $g_i^r = \{0.05, 0.05, 0.05, 10, 1, 1, 1\}$ and $h_i^r = \{10^{-4}, 10^{-4}, 10^{-4}, 10^{-4}, 10^{-3}, 10^{-3}, 10^{-3}\}\ \forall i = 1, \ldots, 7$, for the inputs and outputs, respectively.

5.3.2. PRISMA simulation results

The resulting 3D-RMS errors for the PRISMA simulations are presented in Table 5, along with the relative runtime for each variant of the EKF. The relative runtimes are calculated with respect to the non-adaptive EKF, and the costs of the adaptation schemes agree with the expected trends: the MLE-AEKF is more complex than the FAEKF and the standard EKF by over a factor of 2, due to the storing and processing required by the smoothing routine and update laws within the MLE algorithm. The R-MLE adaptations in Eq. (47) have a larger number of calculations than the Q-MLE adaptations in Eq. (46), which is reflected in the larger runtime for the R-MLE scenario. Not surprisingly, attempting to estimate both noise covariances at the same time leads to the QR-MLE-AEKF displaying the highest level of computational complexity. The FAEKF is less computationally demanding than the MLE-AEKF, with the R-FAEKF being more costly than the Q-FAEKF. Since the R-FAEKF involves processing seven inputs through the FLS and the Q-FAEKF requires only one, the R-FAEKF understandably requires a longer runtime.

---

[1] https://www.novatel.com/assets/Documents/Papers/FlexPak6.pdf

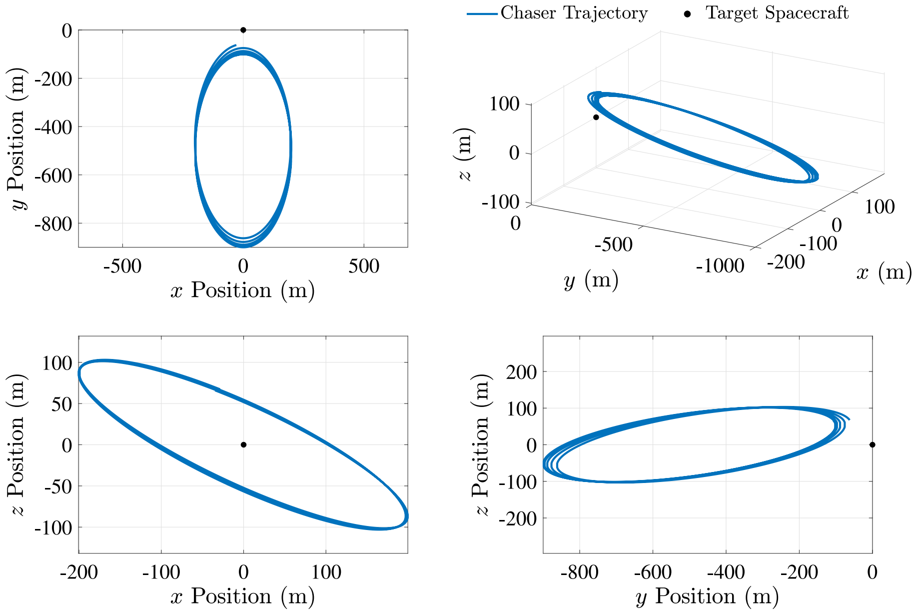


Fig. 5. Relative orbit of the PRISMA mission over five orbital periods of the target spacecraft, under the influence of perturbations from solar radiation pressure, atmospheric drag, third-body gravity and J_2.

More interestingly however, are the estimation errors of the filtering algorithms. First note that the standard EKF is entirely capable of providing better position and velocity estimates than the measurements alone. Furthermore, adaptations of the process noise matrix by both the MLE-AEKF and the FAEKF improve the estimates of the EKF, yielding position estimation errors less than 30% those of the EKF, and velocity estimation errors less than 1% those of the EKF. Fig. 6 presents the state estimates and estimation errors of the proposed filtering algorithms, for both the relative position and relative velocity estimates. For brevity, only plots of the EKF, the Q-MLE-AEKF, and the Q-FAEKF are shown, since Q-adaptations provided the best overall performance in this case study, and the EKF provides a baseline for comparison.

Fig. 7(a) provides a set of histograms comparing the position estimation error distributions for the PRISMA scenario. The first row compares the EKF and the GPS measurements, and shows that the estimation errors of the EKF are more frequently distributed at smaller values than those of the measurements (*i.e.*, the EKF improves the position estimation accuracy relative to the GPS measurements). The second row compares the Q-MLE-AEKF and the EKF, where the adaptive EKF markedly outperforms the standard EKF. A comparison of the Q-FAEKF and the Q-MLE-AEKF is given in the third row, where it is more difficult to determine visually which method provides the best estimation accuracy. It was established from the RMS errors that the Q-MLE-AEKF does indeed provide estimation errors that are 2.57 mm less than those of the Q-FAEKF, so the third row here is effectually demonstrating that both methods provide a comparable level of estimation accuracy. Similar results are obtained for the velocity estimation error distributions, which are illustrated in Fig. 7(b). As a final note, see that the range of the error distributions (the horizontal axes) decreases along the first, second, and third rows, indicating the estimation accuracy increases along the rows.

5.3.3. PRISMA simulation conclusions

The array of figures shown for the simulations of the PRISMA formation have demonstrated that both the Q-MLE-AEKF and the Q-FAEKF provide the best estimation performance out of the proposed algorithms, for this particular formation configuration. The 3D-RMS values further supported these results, where both Q-MLE-AEKF and the Q-FAEKF methods outperform the standard EKF. Although both adaptive schemes provide comparable accuracy in the relative velocity estimates, the position estimates using the MLE adaptations are marginally more accurate than the estimates from the FAEKF adaptations. This additional accuracy comes at the cost of additional computational time however, with the Q-MLE-AEKF requiring 1.8 times more processing time than the Q-FAEKF. Lastly, the position estimation errors obtained in this scenario are well within the predefined one-metre accuracy objective.

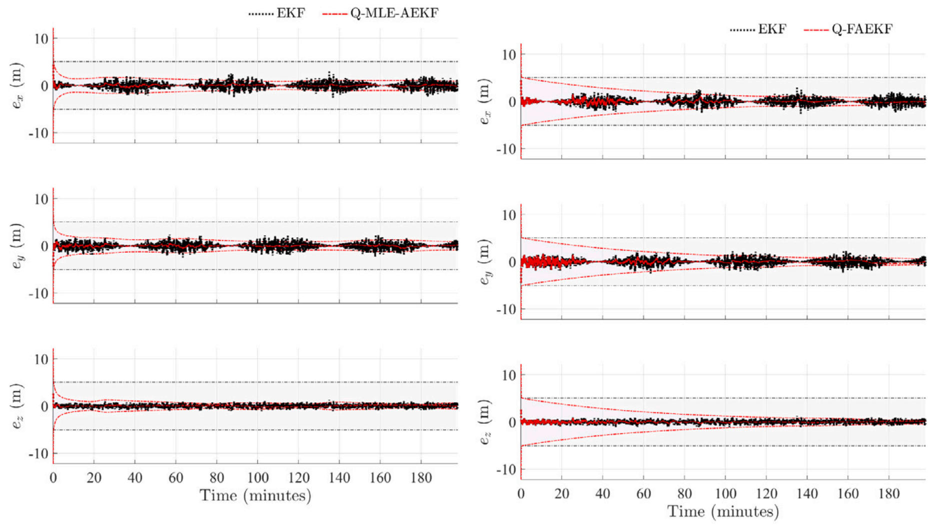
To clarify why the Q-adaptation routines provide the best results in this scenario, consider the dynamical and measurement environments of the system tested here: in Low Earth Orbit, the nonlinear effects of perturbations are significant and dominate the errors due to the measurements. The altitude range spanned by the formation means that the spacecraft are subjected to varying degrees of non-conservative forces from atmospheric drag, and similarly the effects of J_2 will be prevalent at these altitudes. These effects are unmodelled by the EKF's dynamics model, so by adapting the process noise covariance matrix of the EKF, a more appropriate weighting between the dynamics and the measurements of the filter can be obtained using Q-adaptations.

5.4. Relative navigation analysis for the projected elliptical orbit formation

The second formation configuration considered in this analysis yields a projected elliptical orbit between the chaser and the target. Physical spacecraft characteristics similar to the PRISMA formation are used here for continuity, but the defining orbital elements and resulting relative motion have been modified. In this case, both the target and the chaser orbits have an eccentricity near 0.1, two orders of magnitude larger than the PRISMA mission. The relative orbit of the chaser is also centred on the target, whereas the PRISMA formation featured an offset relative orbit. Table 6 lists the orbital elements used to define the PEO formation, and due to the non-zero eccentricity in this LEO scenario, the orbital environment varies in altitude from between 372 km to 1 862 km. The separation between spacecraft oscillates between 370 m and 1 500 m, and the trajectory of the chaser with respect to the target is shown in Fig. 8.

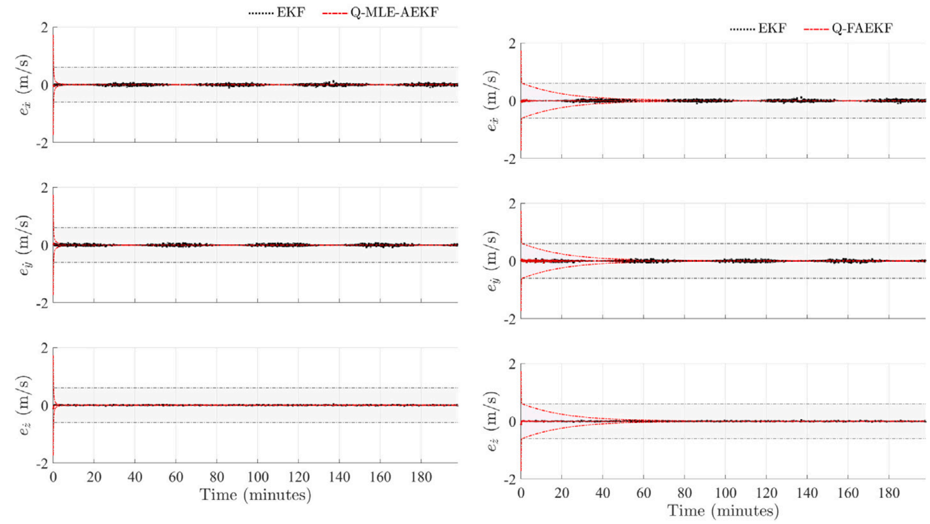
5.4.1. PEO simulation conditions

To demonstrate the performance of the filtering algorithms in the presence of larger measurement noises, the standard deviations of the



(a) Position errors for the EKF and Q-MLE-AEKF

(b) Position errors for the EKF and Q-FAEKF



(c) Velocity errors for the EKF and Q-MLE-AEKF

(d) Velocity errors for the EKF and Q-FAEKF

Fig. 6. Comparison of estimation errors for the PRISMA formation, between the EKF, Q-MLE-AEKF and Q-FAEKF. The improvement of the AEKF estimates are shown along with the decreasing 3σ covariance bounds.

Table 6
Initial orbital elements of the projected elliptical orbit formation.

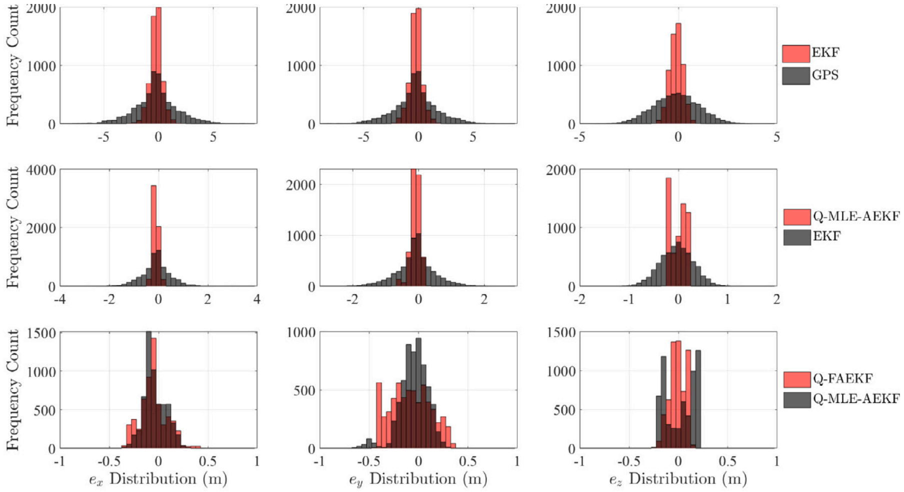
Parameter	Target	Chaser
Semi-Major Axis, a	7500 km	7500 km
Eccentricity, e	0.100 00	0.100 05
Inclination, i	98.188°	98.178°
RAAN, Ω	189.891°	189.891°
Argument of Perigee, ω	1.094°	1.094°
True Anomaly, θ	0°	0°

noise processes have been increased by an order of magnitude from the previous case, now taking on values of $\sigma_r = 12$ m and $\sigma_v = 0.3$ m/s. The initial measurement noise covariance matrix has been increased in light of the additional noise, and the initial conditions of the EKF are also modified to account for the different configuration of the formation. Table 7 contains the initial settings for the EKF, and this case study provides insight into the performance of the filter adaptation schemes in situations where the measurement noises are improperly defined or unknown *a priori*.

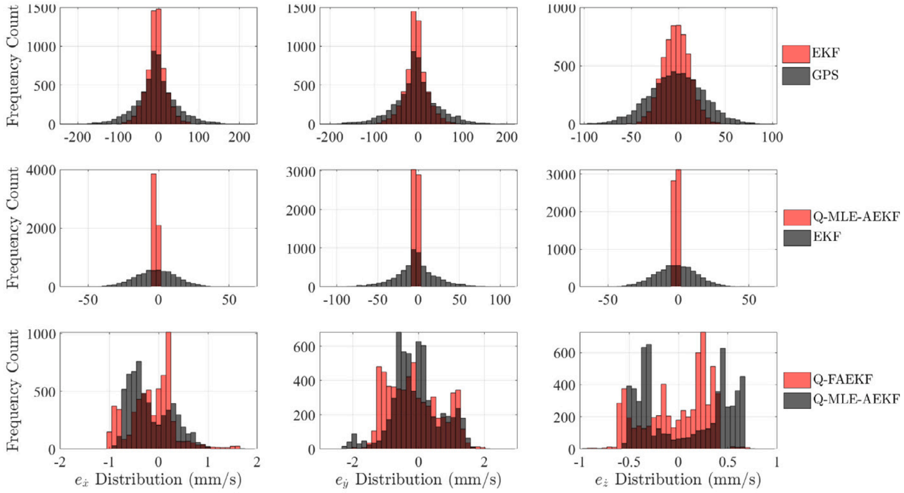
The initial relative position and velocity states of the formation are offset from the true values by 20 m and 20 mm/s, respectively, and the same moving window size of $N = 30$ is used for both the MLE-AEKF and the FAEKF. Here, values of $g^q = -5 \times 10^{-3}$ and $h^q = 10^{-2}$ are selected for the input scaling gain for the Q-adaptations, and $g_i^r = \{0.05, 0.05, 0.05, 10, 1, 1, 1\}$ and $h_i^r = \{10^{-4}, 10^{-4}, 10^{-4}, 10^{-4}, 10^{-3}, 10^{-3}, 10^{-3}\}$ $\forall i = 1, \ldots, 7$ are chosen for the R-adaptation input and output gains.

5.4.2. PEO simulation results

Despite the increased measurement noises used in this case study, all filters are successfully able to converge to the true state trajectory and provide relative position and velocity estimates that are more accurate than the measurements themselves. Table 8 is a compilation of the 3D-RMS errors and the relative runtimes observed in the PEO simulations. The MLE-AEKF schemes require more processing time than the FAEKF



(a) Relative position error distributions.



(b) Relative velocity error distributions.

Fig. 7. Estimation error distributions for the PRISMA formation, comparing the measurement-only, EKF, Q-adaptive MLE-AEKF, and Q-adaptive FAEKF.

Table 7
Initial EKF settings for the PEO in LEO formation.

Initial States $\mathbf{x}_0$	$x_0 = -395.00$ m	$\dot{x}_0 = 2.00$ mm/s
	$y_0 = 20.00$ m	$\dot{y}_0 = 852.70$ mm/s
	$z_0 = -42.49$ m	$\dot{z}_0 = -1404.48$ mm/s
	$\theta_0 = 0$ deg	$\dot{\theta}_0 = 0.0684$ deg/s
	$r_{t_0} = 6750.05$ km	$\dot{r}_{t_0} = 0.00$ m/s
Initial State Error Covariance $\mathbf{P}_0$	$\sigma_x^2 = 100$ m^2	$\sigma_{\dot{x}}^2 = 1$ m^2/s^2
	$\sigma_y^2 = 100$ m^2	$\sigma_{\dot{y}}^2 = 1$ m^2/s^2
	$\sigma_z^2 = 100$ m^2	$\sigma_{\dot{z}}^2 = 1$ m^2/s^2
	$\sigma_\theta^2 = 1$ deg^2	$\sigma_{\dot{\theta}}^2 = 0.01$ deg^2/s^2
	$\sigma_{r_t}^2 = 10\ 000$ m^2	$\sigma_{\dot{r}_t}^2 = 100$ m^2/s^2
Initial Process Noise Covariance $\mathbf{Q}_0$	$q_x = 0.2$ m^2	$q_{\dot{x}} = 5 \times 10^{-3}$ m^2/s^2
	$q_y = 0.2$ m^2	$q_{\dot{y}} = 5 \times 10^{-3}$ m^2/s^2
	$q_z = 0.2$ m^2	$q_{\dot{z}} = 5 \times 10^{-3}$ m^2/s^2
	$q_\theta = 1 \times 10^{-3}$ deg^2	$q_{\dot{\theta}} = 1 \times 10^{-6}$ deg^2/s^2
	$q_{r_t} = 5 \times 10^{-3}$ m^2	$q_{\dot{r}_t} = 5 \times 10^{-5}$ m^2/s^2
Initial Measurement Noise Covariance $\mathbf{R}_0$	$r_x = 100$ m^2	$r_{\dot{x}} = 2.5$ m^2/s^2
	$r_y = 100$ m^2	$r_{\dot{y}} = 2.5$ m^2/s^2
	$r_z = 100$ m^2	$r_{\dot{z}} = 2.5$ m^2/s^2
	$r_\theta = 0.05$ deg^2	

schemes as discussed previously, and similarly the R-adaptations are more costly than the Q-adaptations. These computational considerations match the results from the previous case study of the PRISMA formation.

With the additional measurement noises applied in this study, adapting the measurement noise covariance matrix is presumably the ideal way to improve filter performance, and the results of this case study verify that prediction. In both the MLE-AEKF and the FAEKF tests, the R-adaptation schemes provide the best position estimation results, with RMS errors of 444 cm and 488 cm, respectively. In contrast, both the Q-MLE-AEKF and Q-FAEKF provide position estimates that are more erroneous than the non-adaptive EKF. This recapitulates the fact that identifying and selecting the proper adaptation method for a particular scenario is indeed a design consideration in and of itself.

The adaptive EKF algorithms are able to obtain position estimation accuracies between 1% – 2% of the minimum spacecraft separation distance, an impressive result considering the magnitude of the noises added to the measurements in this scenario, and the varying range of separation between the spacecraft. Further support for the numerical results of the PEO in LEO simulations are provided the position and velocity estimation error plots in Fig. 9, along with an analysis of

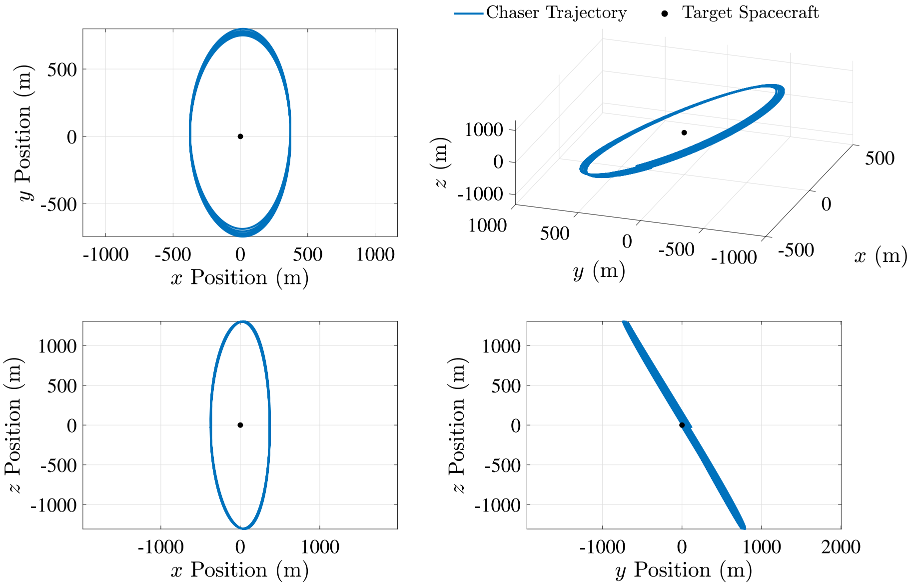


Fig. 8. A projected elliptical orbit in LEO. From its namesake, viewed from above the projection of the relative motion onto the x–y plane produces an ellipse centred on the target.

Table 8
3D-RMS errors from PEO simulation.

Method	Position (cm) [% Error]	Velocity (cm/s) [% Error]	Relative Runtime
Measurements	2974.18 [7.98%]	74.11 [204.28%]	N/A
Standard EKF	719.32 [1.93%]	29.68 [81.89%]	1.00
Q-MLE-AEKF	1697.27 [4.56%]	0.42 [1.16%]	2.32
R-MLE-AEKF	443.61 [1.19%]	20.41 [56.31%]	2.56
QR-MLE-AEKF	457.24 [1.23%]	5.88 [16.22%]	2.58
Q-FAEKF	2375.70 [6.38%]	59.27 [163.53%]	1.17
R-FAEKF	487.72 [1.31%]	19.01 [52.45%]	1.28
QR-FAEKF	2415.14 [6.48%]	74.10 [204.45%]	1.30

the error distributions in Fig. 10. For this scenario with increased measurement noises, figures relating to the EKF, the R-MLE-AEKF, and the R-FAEKF are shown.

The position estimation errors distributions for the PEO case study are contained in Fig. 10(a), and use the same format as the previous histograms: the first row compares the EKF and the GPS measurements, the second row compares the R-MLE-AEKF and the EKF, and the third row compares the R-FAEKF and the R-MLE-AEKF. The magnitude of the measurement noise in this scenario leads to a widely distributed error plot for the measurements, spanning ±50 m. However, the baseline EKF is able to greatly reduce this distribution to within ±25 m, and from the RMS values shown previously, the position errors from the EKF are 25% as large as the measurement errors. Considering the second row, errors from the R-MLE-AEKF appear more tightly-grouped around zero than errors from the EKF, signifying that errors from the adaptive filter are more frequently at a lower magnitude than errors from the non-adaptive filter. This further substantiates that adapting the measurement noise covariance is a suitable tactic for dealing with the large, unknown noises in the measurements. In brief, the position estimation errors of the R-MLE-AEKF and R-FAEKF are similar, but the MLE-AEKF yields errors that are more frequently smaller than the errors from the FAEKF.

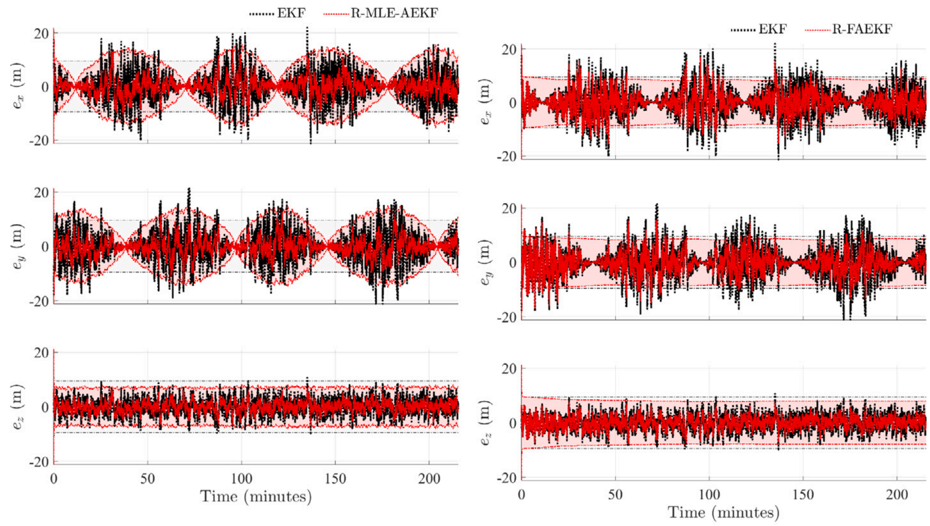
Fig. 10(b) presents the velocity estimation error distributions for the PEO formation, and the first row clearly demonstrates that the velocity estimates from the EKF are more accurate than the measurements provided to the EKF, a result that is not unexpected. Interestingly, the second row exposes that the span of the velocity estimation errors from the R-MLE-AEKF is nearly as large as the EKF. Although the width of both these symmetric distributions is similar, the sharpness of the R-MLE-AEKF is greater than that of the EKF. Thus, more estimation errors from the MLE-AEKF are smaller than the errors from the EKF, which corroborates with the smaller RMS errors achieved by the MLE-AEKF compared to the EKF. The third row confirms that the velocity estimates obtained by the R-FAEKF are consistently more accurate than the estimates by the R-MLE-AEKF. The RMS error for the FAEKF is only 14 mm/s lower than that of the MLE-AEKF, but the histograms shown below provide an astute representation of the filter performances.

5.4.3. PEO simulation conclusions

The EKF, MLE-AEKF, and FAEKF demonstrated improved relative position and velocity estimation throughout this study of a spacecraft PEO formation. Increasing the measurement noises led to poor performance from the Q-adaptation routines, a foreseeable result given that the formulations of both the MLE-AEKF and the FAEKF do not distinguish between noises due to measurement errors and noises due to modelling mismatch. As such, selecting an appropriate adaptation scheme for a particular scenario is a consideration that must be made before implementing these adaptation filters in a practical setting. The processing requirements for the Kalman filter variants in these simulations matched those from the PRISMA case, where QR-adaptation was the most expensive, followed by R-adaptation and then Q-adaptation. The FAEKF required 30% more computational time than the EKF, and the MLE-AEKF took twice as long as the FAEKF. From the PRISMA test case, recall that the 3D-RMS position errors of the standard EKF were reduced by 72% through the use of the MLE-AEKF for resolving errors in the dynamical models, and here use of the MLE-AEKF in dealing with measurement noises gave a 3D-RMS position error reduction upwards of 38%.

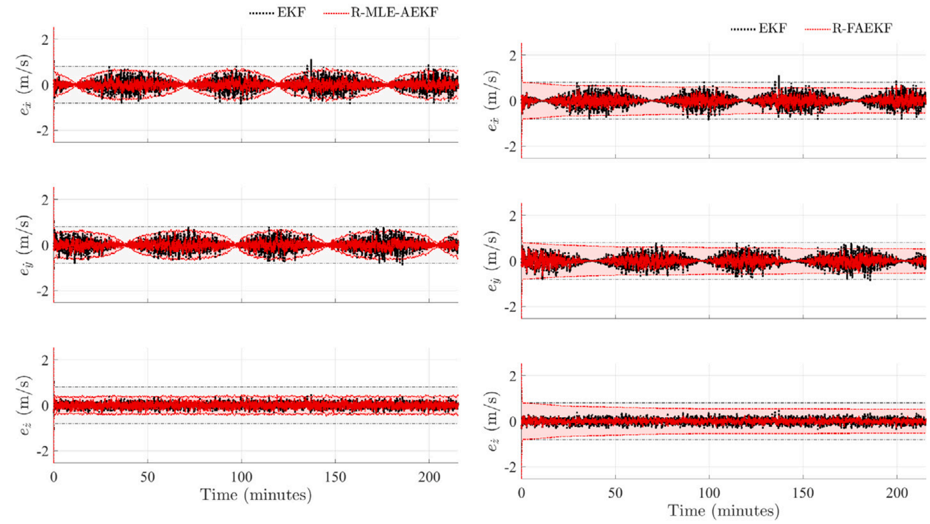
6. Conclusions

This paper presented the methodology, design and numerical validation of two adaptive extended Kalman filtering algorithms. The performance of these filter was analysed through the simulation of two unique spacecraft formation scenarios, the first of which considered the PRISMA formation in a low-eccentricity, low-Earth orbit. In these



(a) Position errors from the EKF and R-MLE-AEKF

(b) Position errors from the EKF and R-FAEKF



(c) Velocity errors from the EKF and R-MLE-AEKF

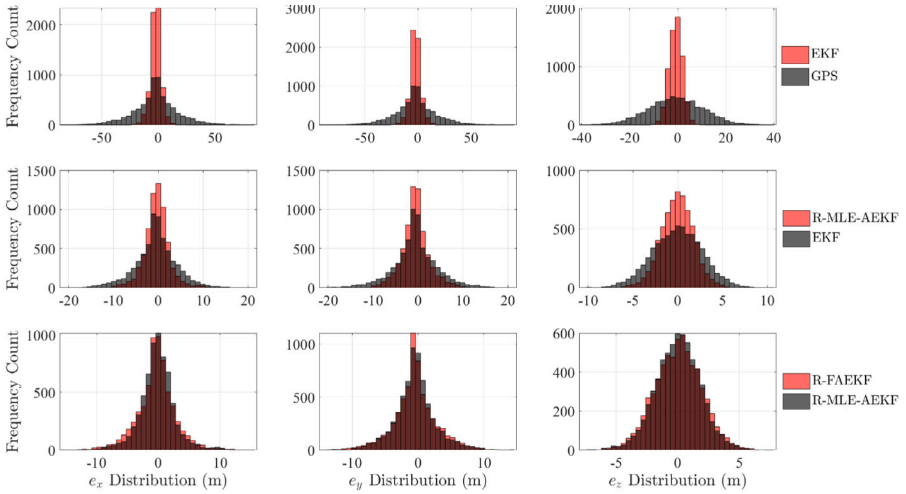
(d) Velocity errors from the EKF and R-FAEKF

Fig. 9. Comparing the performance of the navigation routines for the PEO formation, showing estimation errors and 3σ covariance bounds from the EKF, the R-MLE-AEKF, and the R-FAEKF.

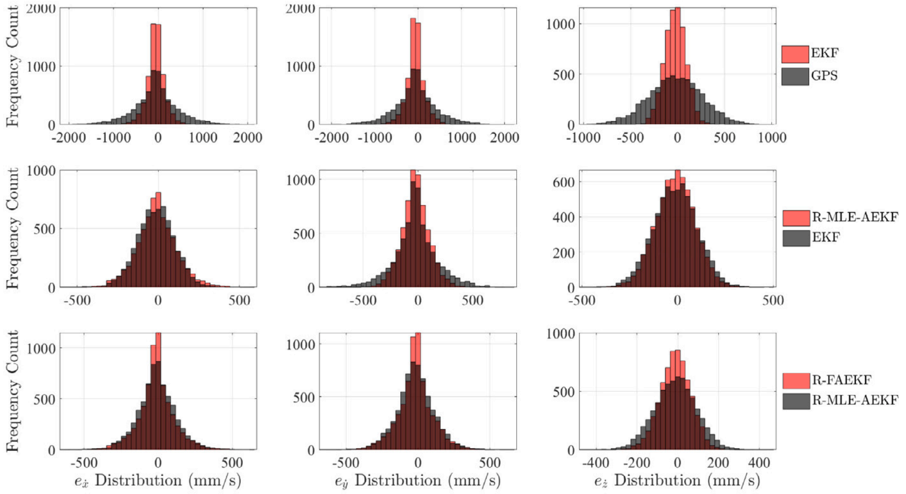
conditions, unmodelled orbital perturbations were expected to dominate the state estimation errors of the EKF, and it was shown that the proposed Q-adaptation methods were able to drastically improve the accuracy of the state estimates. The second scenario simulated a representative spacecraft formation in a LEO-based projected-elliptical orbit with mild eccentricity, and featured an increase in the measurement noises. The performance of the R-adaptive Kalman filter schemes accordingly provided better state estimates in this scenario than the non-adaptive EKF. In both case studies, the performance of the Maximum Likelihood Estimation EKF was superior to that of the Fuzzy Adaptive EKF. The analytic MLE adaptation laws displayed rapid responsiveness, while the FAEKF provided slower adaptations based on the current tuning of the FLS. Improved estimation accuracy by the MLE-AEKF comes at the cost of the additional processing power needed to complete the smoothing and adaptation calculations within the MLE routine, as the MLE-AEKF required twice the computational time of the FAEKF. While the MLE-AEKF estimation errors were lower than those of the FAEKF in these case studies, they were not commensurate; this means careful consideration of the processing budget and desired navigation accuracy would need to be considered before implementing one of these methods for onboard operation. Above all, by intelligently selecting to adapt either the process noise or the measurement noise covariances within the EKF, the adaptation schemes presented in this chapter successfully demonstrate two methods for improving the estimation accuracy of the standard EKF in the presence of modelling errors, measurement noise, and poorly-defined initial noise statistics within the filter itself.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.



(a) Relative position error distributions.



(b) Relative velocity error distributions.

Fig. 10. Estimation error distributions for the PEO formation, comparing the measurement-only, EKF, R-adaptive MLE-AEKF, and R-adaptive FAEKF.

Appendix A. Linearization of the nonlinear dynamics models

This appendix provides the partial derivatives terms used to construct the Jacobian of the nonlinear dynamics model. In the context of spacecraft formation navigation presented in this paper, the Jacobian is necessary to construct the discrete time state transition matrix of the linear dynamics, which is then used to propagate the EKF error covariance ahead one step in time. The full Jacobian matrix is defined in Eq. (18), and the non-zero partial derivatives of the nonlinear dynamics are given here. The derivatives of the radial position dynamics with respect to the states are:

$$\ddot{x}_x = \frac{\partial \ddot{x}}{\partial x} = \frac{\mu}{r_c^3}\left[3\left(\frac{r_t+x}{r_c}\right)^2 - 1\right] + \dot{\theta}^2 \tag{A.1}$$

$$\ddot{x}_{\dot{y}} = \frac{\partial \ddot{x}}{\partial \dot{y}} = 2\dot{\theta} \tag{A.2}$$

$$\ddot{x}_y = \frac{\partial \ddot{x}}{\partial y} = 3\mu\left(\frac{r_t+x}{r_c^5}\right)y - 2\dot{\theta}\frac{\dot{r}_t}{r_t} \tag{A.3}$$

$$\ddot{x}_{\dot{r}_t} = \frac{\partial \ddot{x}}{\partial \dot{r}_t} = -2\frac{\dot{\theta}}{r_t}y \tag{A.4}$$

$$\ddot{x}_z = \frac{\partial \ddot{x}}{\partial z} = 3\mu\left(\frac{r_t+x}{r_c^5}\right)z \tag{A.5}$$

$$\ddot{x}_{\dot{\theta}} = \frac{\partial \ddot{x}}{\partial \dot{\theta}} = 2\left(\dot{\theta}x + \dot{y} - \frac{\dot{r}_t}{r_t}y\right) \tag{A.6}$$

$$\ddot{x}_{r_t} = \frac{\partial \ddot{x}}{\partial r_t} = 2\dot{\theta}\frac{\dot{r}_t}{r_t^2}y - 2\frac{\mu}{r_t^3} + \frac{\mu}{r_c^3}\left[3\left(\frac{r_t+x}{r_c}\right)^2 - 1\right] \tag{A.7}$$

and the linearized components for the along-track position dynamics are:

$$\ddot{y}_x = \frac{\partial \ddot{y}}{\partial x} = 2\dot{\theta}\frac{\dot{r}_t}{r_t} + 3\mu\left(\frac{r_t+x}{r_c^5}\right)y \tag{A.8}$$

$$\ddot{y}_{\dot{x}} = \frac{\partial \ddot{y}}{\partial \dot{x}} = -2\dot{\theta} \tag{A.9}$$

$$\ddot{y}_y = \frac{\partial \ddot{y}}{\partial y} = \frac{\mu}{r_c^3}\left[3\left(\frac{y}{r_c}\right)^2 - 1\right] + \dot{\theta}^2 \tag{A.10}$$

$$\ddot{y}_{\dot{r}_t} = \frac{\partial \ddot{y}}{\partial \dot{r}_t} = 2\frac{\dot{\theta}}{r_t}x \tag{A.11}$$

$$\ddot{y}_z = \frac{\partial \ddot{y}}{\partial z} = 3\mu\left(\frac{y}{r_c^5}\right)z \tag{A.12}$$

$$\ddot{y}_{\dot{\theta}} = \frac{\partial \ddot{y}}{\partial \dot{\theta}} = 2\left(\dot{\theta}y - \dot{x} + \frac{\dot{r}_t}{r_t}x\right) \tag{A.13}$$

$$\ddot{y}_{r_t} = \frac{\partial \ddot{y}}{\partial r_t} = 3\mu\left(\frac{r_t + x}{r_c^5}\right)y - 2\dot{\theta}\frac{\dot{r}_t}{r_t^2}x \tag{A.14}$$

Partial derivatives of the cross-track position are given by:

$$\ddot{z}_x = \frac{\partial \ddot{z}}{\partial x} = 3\mu\left(\frac{r_t + x}{r_c^5}\right)z \tag{A.15}$$

$$\ddot{z}_y = \frac{\partial \ddot{z}}{\partial y} = 3\mu\left(\frac{y}{r_c^5}\right)z \tag{A.16}$$

$$\ddot{z}_z = \frac{\partial \ddot{z}}{\partial z} = \frac{\mu}{r_c^3}\left[3\left(\frac{z}{r_c}\right)^2 - 1\right] \tag{A.17}$$

$$\ddot{z}_{r_t} = \frac{\partial \ddot{z}}{\partial r_t} = 3\mu\left(\frac{r_t + x}{r_c^5}\right)z \tag{A.18}$$

and linearizing the true anomaly dynamics equation gives:

$$\ddot{\theta}_{r_t} = \frac{\partial \ddot{\theta}}{\partial r_t} = 2\frac{\dot{r}_t}{r_t^2}\dot{\theta} \tag{A.19}$$

$$\ddot{\theta}_{\dot{\theta}} = \frac{\partial \ddot{\theta}}{\partial \dot{\theta}} = -2\frac{\dot{r}_t}{r_t} \tag{A.20}$$

$$\ddot{\theta}_{\dot{r}_t} = \frac{\partial \ddot{\theta}}{\partial \dot{r}_t} = -2\frac{\dot{\theta}}{r_t} \tag{A.21}$$

Lastly, the linearized target radius components are:

$$\ddot{r}_{t_{r_t}} = \frac{\partial \ddot{r}_t}{\partial r_t} = \dot{\theta}^2 + 2\frac{\mu}{r_t^3} \tag{A.22}$$

$$\ddot{r}_{t_{\dot{\theta}}} = \frac{\partial \ddot{r}_t}{\partial \dot{\theta}} = 2\dot{\theta}r_t \tag{A.23}$$

All components of the states and their respective dynamical equations shown above take their usually definitions from the formation flying relative motion descriptions described in Section 2. Within the EKF, the state transition matrix is calculated at every time step by evaluating each of these partial derivative components at the current best estimate of the state (*i.e.*, at the *a posteriori* state estimate from the previous time step).

Appendix B. Observability for relative spacecraft motion

Continuing from the in-text discussion on observability within the EKF, this appendix outlines the construction of the observability matrix $\mathcal{O}(\mathbf{x})$, with the objective of showing that the observability matrix is full rank. Recalling that the zeroth-order Lie derivative is simply the measurement model of the system, it is known that

$$\mathcal{L}_f^0\mathbf{h}(\mathbf{x}) = \mathbf{h}(\mathbf{x}) \tag{B.1}$$

where $\mathbf{h}(\mathbf{x})$ is given by Eq. (12). The gradient of the zeroth-order Lie Derivative of $\mathbf{f}(\mathbf{x})$ along $\mathbf{h}(\mathbf{x})$ can then be established, and corresponds to the linearized measurement model used within the EKF. As such, this yields

$$\nabla\mathcal{L}_f^0\mathbf{h}(\mathbf{x}) = \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} = \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{B.2}$$

The first-order Lie Derivative requires more work to derive, and is seen to be

$$\mathcal{L}_f^1\mathbf{h}(\mathbf{x}) = \mathbf{H}\mathbf{f}(\mathbf{x}) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{r}_t \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{\theta} \\ \ddot{r}_t \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} \tag{B.3}$$

Taking the gradient of the first-order Lie Derivative leads to the following result:

$$\nabla\mathcal{L}_f^1\mathbf{h}(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} & 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\ \ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} & \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\ \ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{B.4}$$

As a reminder, the symbol $\ddot{x}_x$ is shorthand notation used to indicate the partial derivative of $\ddot{x}$ with respect to x, and so on for the other dynamical states. The second-order Lie Derivative in the spacecraft formation scenario is

$$\mathcal{L}_f^2\mathbf{h}(\mathbf{x}) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ \ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} & 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\ \ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} & \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\ \ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{\theta} \\ \dot{r}_t \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{\theta} \\ \ddot{r}_t \end{bmatrix} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \\ \ddot{\theta} \\ L_1 \\ L_2 \\ L_3 \end{bmatrix} \tag{B.5}$$

where L_1, L_2 and L_3 represent the following functions

$$L_1 = \dot{x}\ddot{x}_x + \dot{y}\ddot{x}_y + \dot{z}\ddot{x}_z + \dot{r}_t\ddot{x}_{r_t} + \ddot{y}\ddot{x}_{\dot{y}} + \ddot{\theta}\ddot{x}_{\dot{\theta}} + \ddot{r}_t\ddot{x}_{\dot{r}_t} \tag{B.6}$$

$$L_2 = \dot{x}\ddot{y}_x + \dot{y}\ddot{y}_y + \dot{z}\ddot{y}_z + \dot{r}_t\ddot{y}_{r_t} + \ddot{x}\ddot{y}_{\dot{x}} + \ddot{\theta}\ddot{y}_{\dot{\theta}} + \ddot{r}_t\ddot{y}_{\dot{r}_t} \tag{B.7}$$

$$L_3 = \dot{x}\ddot{z}_x + \dot{y}\ddot{z}_y + \dot{z}\ddot{z}_z + \dot{r}_t\ddot{z}_{r_t} \tag{B.8}$$

The gradient of the second-order Lie Derivative can be written as

$$\nabla\mathcal{L}_f^2\mathbf{h}(\mathbf{x}) = \begin{bmatrix} \ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} & 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\ \ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} & \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\ \ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \ddot{\theta}_{r_t} & 0 & 0 & 0 & \ddot{\theta}_{\dot{\theta}} & \ddot{\theta}_{\dot{r}_t} \\ \alpha_{51} & \alpha_{52} & \alpha_{53} & 0 & \alpha_{55} & \alpha_{56} & \alpha_{57} & \ddot{x}_z & \alpha_{59} & \alpha_{5,10} \\ \alpha_{61} & \alpha_{62} & \alpha_{63} & 0 & \alpha_{65} & \alpha_{66} & \alpha_{67} & \ddot{y}_z & \alpha_{69} & \alpha_{6,10} \\ \alpha_{71} & \alpha_{72} & \alpha_{73} & 0 & \alpha_{75} & \ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} \end{bmatrix} \tag{B.9}$$

where the various $\alpha_{k,j}$ matrix components are shown on the following page, for $k \in 1, \ldots, 7$ and $j \in 1, \ldots, 10$. The second-order form of the partial derivative shorthand notation is needed within the $\alpha_{k,j}$ terms, where for example $\ddot{x}_{zx}$ represents

$$\ddot{x}_{zx} = \frac{\partial \ddot{x}_z}{\partial x} = \frac{\partial}{\partial x}\frac{\partial \ddot{x}}{\partial z} = \frac{\partial^2 \ddot{x}}{\partial x \partial z} \tag{B.10}$$

$$\alpha_{51} = \frac{\partial L_1}{\partial x} = \dot{x}\ddot{x}_{xx} + \dot{y}\ddot{x}_{yx} + \dot{z}\ddot{x}_{zx} + \dot{r}\ddot{x}_{rx} + \ddot{y}_x\ddot{x}_{\dot{y}} + \ddot{\theta}\ddot{x}_{\dot{\theta}x} \tag{B.11}$$

$$\alpha_{52} = \frac{\partial L_1}{\partial y} = \dot{x}\ddot{x}_{xy} + \dot{y}\ddot{x}_{yy} + \dot{z}\ddot{x}_{zy} + \dot{r}\ddot{x}_{ry} + \ddot{y}_y\ddot{x}_{\dot{y}} + \ddot{\theta}\ddot{x}_{\dot{\theta}y} + \ddot{r}\ddot{x}_{\dot{r}y} \tag{B.12}$$

$$\alpha_{53} = \frac{\partial L_1}{\partial z} = \dot{x}\ddot{x}_{xz} + \dot{y}\ddot{x}_{yz} + \dot{z}\ddot{x}_{zz} + \dot{r}\ddot{x}_{rz} + \ddot{y}_z\ddot{x}_{\dot{y}} \tag{B.13}$$

$$\alpha_{55} = \frac{\partial L_1}{\partial r_t} = \dot{x}\ddot{x}_{xr} + \dot{y}\ddot{x}_{yr} + \dot{z}\ddot{x}_{zr} + \dot{r}\ddot{x}_{rr} + \ddot{y}_r\ddot{x}_{\dot{y}} + \ddot{\theta}_r\ddot{x}_{\dot{\theta}r} + \ddot{r}_r\ddot{x}_{\dot{r}} + \ddot{r}\ddot{x}_{\dot{r}r} \tag{B.14}$$

$$\alpha_{56} = \frac{\partial L_1}{\partial \dot{x}} = \ddot{x}_x + \ddot{y}_{\dot{x}}\ddot{x}_{\dot{y}} \tag{B.15}$$

$$\alpha_{57} = \frac{\partial L_1}{\partial \dot{y}} = \ddot{x}_y + \ddot{\theta}\ddot{x}_{\dot{\theta}\dot{y}} \tag{B.16}$$

$$\alpha_{58} = \frac{\partial L_1}{\partial \dot{z}} = \ddot{x}_z \tag{B.17}$$

$$\alpha_{59} = \frac{\partial L_1}{\partial \dot{\theta}} = \dot{x}\ddot{x}_{x\dot{\theta}} + \dot{y}\ddot{x}_{y\dot{\theta}} + \dot{r}\ddot{x}_{r\dot{\theta}} + \ddot{y}_{\dot{\theta}}\ddot{x}_{\dot{y}} + \ddot{y}\ddot{x}_{\dot{y}\dot{\theta}} + \ddot{\theta}_{\dot{\theta}}\ddot{x}_{\dot{\theta}} + \ddot{\theta}\ddot{x}_{\dot{\theta}\dot{\theta}} + \ddot{r}_{\dot{\theta}}\ddot{x}_{\dot{r}} + \ddot{r}\ddot{x}_{\dot{r}\dot{\theta}} \tag{B.18}$$

$$\alpha_{5,10} = \frac{\partial L_1}{\partial \dot{r}_t} = \dot{y}\ddot{x}_{y\dot{r}} + \ddot{x}_r + \ddot{y}_{\dot{r}}\ddot{x}_{\dot{y}} + \ddot{\theta}_{\dot{r}}\ddot{x}_{\dot{\theta}} + \ddot{\theta}\ddot{x}_{\dot{\theta}\dot{r}} \tag{B.19}$$

The second-order partial derivative terms in these matrices are not presented here, as they are quite lengthy and cumbersome. They are easily derivable using software that features an analytic solver, such as Maple or the MATLAB Symbolic Toolbox. Thus, for the purposes of confirming observability for the proposed extended Kalman filter algorithms, it is sufficient to demonstrate that these terms are non-zero.

$$\alpha_{61} = \frac{\partial L_2}{\partial x} = \dot{x}\ddot{y}_{xx} + \dot{y}\ddot{y}_{yx} + \dot{z}\ddot{y}_{zx} + \dot{r}\ddot{y}_{rx} + \ddot{x}_x\ddot{y}_{\dot{x}} + \ddot{\theta}\ddot{y}_{\dot{\theta}x} + \ddot{r}\ddot{y}_{\dot{r}x} \tag{B.20}$$

$$\alpha_{62} = \frac{\partial L_2}{\partial y} = \dot{x}\ddot{y}_{xy} + \dot{y}\ddot{y}_{yy} + \dot{z}\ddot{y}_{zy} + \dot{r}\ddot{y}_{ry} + \ddot{x}_y\ddot{y}_{\dot{x}} + \ddot{\theta}\ddot{y}_{\dot{\theta}y} \tag{B.21}$$

$$\alpha_{63} = \frac{\partial L_2}{\partial z} = \dot{x}\ddot{y}_{xz} + \dot{y}\ddot{y}_{yz} + \dot{z}\ddot{y}_{zz} + \dot{r}\ddot{y}_{rz} + \ddot{x}_z\ddot{y}_{\dot{x}} \tag{B.22}$$

$$\alpha_{65} = \frac{\partial L_2}{\partial r_t} = \dot{x}\ddot{y}_{xr} + \dot{y}\ddot{y}_{yr} + \dot{z}\ddot{y}_{zr} + \dot{r}\ddot{y}_{rr} + \ddot{x}_r\ddot{y}_{\dot{x}} + \ddot{\theta}_r\ddot{y}_{\dot{\theta}} + \ddot{\theta}\ddot{y}_{\dot{\theta}r} + \ddot{r}_r\ddot{y}_{\dot{r}} + \ddot{r}\ddot{y}_{\dot{r}r} \tag{B.23}$$

$$\alpha_{66} = \frac{\partial L_2}{\partial \dot{x}} = \ddot{y}_x + \ddot{\theta}\ddot{y}_{\dot{\theta}\dot{x}} \tag{B.24}$$

$$\alpha_{67} = \frac{\partial L_2}{\partial \dot{y}} = \ddot{y}_y + \ddot{x}_{\dot{y}}\ddot{y}_{\dot{x}} \tag{B.25}$$

$$\alpha_{68} = \frac{\partial L_2}{\partial \dot{z}} = \ddot{y}_z \tag{B.26}$$

$$\alpha_{69} = \frac{\partial L_2}{\partial \dot{\theta}} = \dot{x}\ddot{y}_{x\dot{\theta}} + \dot{y}\ddot{y}_{y\dot{\theta}} + \dot{r}\ddot{y}_{r\dot{\theta}} + \ddot{x}_{\dot{\theta}}\ddot{y}_{\dot{x}} + \ddot{x}\ddot{y}_{\dot{x}\dot{\theta}} + \ddot{\theta}_{\dot{\theta}}\ddot{y}_{\dot{\theta}} + \ddot{\theta}\ddot{y}_{\dot{\theta}\dot{\theta}} + \ddot{r}_{\dot{\theta}}\ddot{y}_{\dot{r}} + \ddot{r}\ddot{y}_{\dot{r}\dot{\theta}} \tag{B.27}$$

$$\alpha_{6,10} = \frac{\partial L_2}{\partial \dot{r}_t} = \dot{x}\ddot{y}_{x\dot{r}} + \ddot{y}_r + \dot{r}\ddot{y}_{r\dot{r}} + \ddot{x}_{\dot{r}}\ddot{y}_{\dot{x}} + \ddot{\theta}_{\dot{r}}\ddot{y}_{\dot{\theta}} + \ddot{\theta}\ddot{y}_{\dot{\theta}\dot{r}} \tag{B.28}$$

$$\alpha_{71} = \frac{\partial L_3}{\partial x} = \dot{x}\ddot{z}_{xx} + \dot{y}\ddot{z}_{yx} + \dot{z}\ddot{z}_{zx} + \dot{r}\ddot{z}_{rx} \tag{B.29}$$

$$\alpha_{72} = \frac{\partial L_3}{\partial y} = \dot{x}\ddot{z}_{xy} + \dot{y}\ddot{z}_{yy} + \dot{z}\ddot{z}_{zy} + \dot{r}\ddot{z}_{ry} \tag{B.30}$$

$$\alpha_{73} = \frac{\partial L_3}{\partial z} = \dot{x}\ddot{z}_{xz} + \dot{y}\ddot{z}_{yz} + \dot{z}\ddot{z}_{zz} + \dot{r}\ddot{z}_{rz} \tag{B.31}$$

$$\alpha_{75} = \frac{\partial L_3}{\partial r_t} = \dot{x}\ddot{z}_{xr} + \dot{y}\ddot{z}_{yr} + \dot{z}\ddot{z}_{zr} + \dot{r}\ddot{z}_{rr} \tag{B.32}$$

For the spacecraft formation dynamics and measurement models used in this research, the first three Lie derivatives are sufficient to show that observability is ensured when the relative position, relative velocity, and target true anomaly are available as measurements. The first $N = 3$ Lie derivatives are presented in the observability matrix below, demonstrating the full rank condition, and the partition lines indicate the zeroth, first, and second order Lie derivative portions of the matrix. The resulting truncated observability matrix $\mathcal{O}_N(\mathbf{x})$ can be summarized as:

$$\mathcal{O}_N(\mathbf{x}) = \left[\begin{array}{cccccccccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
\hdashline
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} & 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\
\ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} & \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\
\ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} & 0 & 0 & 0 & 0 & 0 \\
\hdashline
\ddot{x}_x & \ddot{x}_y & \ddot{x}_z & 0 & \ddot{x}_{r_t} & 0 & \ddot{x}_{\dot{y}} & 0 & \ddot{x}_{\dot{\theta}} & \ddot{x}_{\dot{r}_t} \\
\ddot{y}_x & \ddot{y}_y & \ddot{y}_z & 0 & \ddot{y}_{r_t} & \ddot{y}_{\dot{x}} & 0 & 0 & \ddot{y}_{\dot{\theta}} & \ddot{y}_{\dot{r}_t} \\
\ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \ddot{\theta}_{r_t} & 0 & 0 & 0 & \ddot{\theta}_{\dot{\theta}} & \ddot{\theta}_{\dot{r}_t} \\
\alpha_{51} & \alpha_{52} & \alpha_{53} & 0 & \alpha_{55} & \alpha_{56} & \alpha_{57} & \ddot{x}_z & \alpha_{59} & \alpha_{5,10} \\
\alpha_{61} & \alpha_{62} & \alpha_{63} & 0 & \alpha_{65} & \alpha_{66} & \alpha_{67} & \ddot{y}_z & \alpha_{69} & \alpha_{6,10} \\
\alpha_{71} & \alpha_{72} & \alpha_{73} & 0 & \alpha_{75} & \ddot{z}_x & \ddot{z}_y & \ddot{z}_z & 0 & \ddot{z}_{r_t} \\
\alpha_{51} & \alpha_{52} & \alpha_{53} & 0 & \alpha_{55} & \alpha_{56} & \alpha_{57} & \ddot{x}_z & \alpha_{59} & \alpha_{5,10}
\end{array}\right] \tag{B.33}$$

Appendix C. MLE adaptation equation derivations

The proof of the MLE adaptation equations for $\mathbf{Q}_k$ and $\mathbf{R}_k$ is explained here for completeness, to clarify and consolidate past references that have described different aspects of the derivation [14,16,32,36]. To begin, the goal of maximizing the log-likelihood function is equivalent to minimizing the negative log-likelihood function, and a pseudo-likelihood cost function was reduced to the form in Eq. (32), which is rewritten here for convenience as

$$J\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right) \triangleq \sum_{i=i_0}^{N}\left[\ln|\boldsymbol{\Sigma}_i| + \boldsymbol{v}_i^T\boldsymbol{\Sigma}_i^{-1}\boldsymbol{v}_i\right] \tag{C.1}$$

The MLE solution to the pseudo-likelihood minimization problem can then be constructed analytically by taking the derivative of the cost function with respect to the adaptation parameter α_k, and setting the results equal to zero:

$$0 = \frac{\partial}{\partial\alpha_k}\left[J\left(\boldsymbol{\Theta}|\mathbf{Z}_N\right)\right] \tag{C.2}$$

To evaluate this partial derivative, a primary assumption is that the filter states are insensitive to the adaptation parameters, implying that $\frac{\partial\mathbf{x}_k}{\partial\alpha_k} = 0$. Two additional matrix identities are then required, where the first specifies the partial derivative of the logarithm of a matrix determinant, and the second specifies the derivative of a matrix inverse [32]. For a generic square matrix $\mathbf{A}$ that is dependent on some scalar term x, the two necessary identities are

$$\frac{\partial\ln|\mathbf{A}|}{\partial x} = \frac{\partial\ln|\mathbf{A}|}{\partial|\mathbf{A}|}\frac{\partial|\mathbf{A}|}{\partial x} = \frac{1}{|\mathbf{A}|}\frac{\partial|\mathbf{A}|}{\partial x} = \operatorname{tr}\left\{\mathbf{A}^{-1}\frac{\partial\mathbf{A}}{\partial x}\right\} \tag{C.3}$$

$$\frac{\partial\mathbf{A}^{-1}}{\partial x} = -\mathbf{A}^{-1}\frac{\partial\mathbf{A}}{\partial x}\mathbf{A}^{-1} \tag{C.4}$$

Using Eqs. (C.3) and (C.4), the partial derivatives of the pseudo-likelihood cost function in Eq. (C.1) can be rewritten as

$$0 = \sum_{i=i_0}^{N}\left[\operatorname{tr}\left\{\boldsymbol{\Sigma}_i^{-1}\frac{\partial\boldsymbol{\Sigma}_i}{\partial\alpha_k}\right\} - \boldsymbol{v}_i^T\boldsymbol{\Sigma}_i^{-1}\frac{\partial\boldsymbol{\Sigma}_i}{\partial\alpha_k}\boldsymbol{\Sigma}_i^{-1}\boldsymbol{v}_i\right] \tag{C.5}$$

At this point, the relationship between the residuals covariance matrix and filter process and measurement noise covariances can be used, where the partial derivative of $\boldsymbol{\Sigma}_i = \mathbf{H}_i\mathbf{P}_i^{-}\mathbf{H}_i^T + \mathbf{R}_i$ with respect to the adaptation parameter can be inserted into the preceding equation. With the assumption that the measurement model is independent of α_k and time invariant, such that $\frac{\partial \mathbf{H}_i}{\partial \alpha_k} = 0$, evaluating the partial derivative of the residuals covariance gives

$$\frac{\partial \boldsymbol{\Sigma}_i}{\partial \alpha_k} = \frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{P}_i^-}{\partial \alpha_k} \mathbf{H}_i^T \tag{C.6}$$

Further assuming that the state transition matrix of the dynamical process is independent of the adaptation parameters, take $\frac{\partial \boldsymbol{\Phi}_i}{\partial \alpha_k} \approx 0$, and obtain the sensitivity of the *a priori* state error covariance $\mathbf{P}_i^- = \boldsymbol{\Phi}_{i-1} \mathbf{P}_{i-1}^+ \boldsymbol{\Phi}_{i-1}^T + \mathbf{Q}_i$ from Eq. (14) as

$$\frac{\partial \mathbf{P}_i^-}{\partial \alpha_k} = \boldsymbol{\Phi}_{i-1} \frac{\partial \mathbf{P}_{i-1}^+}{\partial \alpha_k} \boldsymbol{\Phi}_{i-1}^T + \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \tag{C.7}$$

If the convergence of the Kalman filter is assumed to be slower than the rate of adaptation of the parameters (*i.e.*, if the processes within the estimation window are assumed to be at steady-state), than the term $\frac{\partial \mathbf{P}_{i-1}^+}{\partial \alpha_k} \approx 0$. Thus sensitivity of $\mathbf{P}_i^-$ reduces to a dependency only on $\frac{\partial \mathbf{Q}_i}{\partial \alpha_k}$, and Eq. (C.6) reduces to

$$\frac{\partial \boldsymbol{\Sigma}_i}{\partial \alpha_k} = \frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \tag{C.8}$$

Next, substituting Eq. (C.8) into (C.5) gives

$$0 = \sum_{i=i_0}^{N} \left[\operatorname{tr}\left\{ \boldsymbol{\Sigma}_i^{-1} \left(\frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \right) \right\} - \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \left(\frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \right) \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \right] \tag{C.9}$$

Using another identity of the trace operator [32], for two arbitrary square matrices **A** and **B**, the following relations hold:

$$\mathbf{A}^T \mathbf{B} = \operatorname{tr}\left\{ \mathbf{A}\mathbf{B}^T \right\} = \operatorname{tr}\left\{ \mathbf{B}\mathbf{A}^T \right\} \tag{C.10}$$

and applying the identity from Eq. (C.10) to the right-hand term in the summation of Eq. (C.9) leads to

$$0 = \sum_{i=i_0}^{N} \left[\operatorname{tr}\left\{ \boldsymbol{\Sigma}_i^{-1} \left(\frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \right) \right\} - \operatorname{tr}\left\{ \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \left(\frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \right) \right\} \right] \tag{C.11}$$

Bringing both terms of Eq. (C.11) inside the trace gives

$$0 = \sum_{i=i_0}^{N} \operatorname{tr}\left\{ \boldsymbol{\Sigma}_i^{-1} \left(\frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \right) - \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \left(\frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \right) \right\} \tag{C.12}$$

and factoring similar terms shows that the partial differential equation that maximizes the likelihood function for the adaptation parameters is

$$0 = \sum_{i=i_0}^{N} \operatorname{tr}\left\{ \left[\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \right] \left[\frac{\partial \mathbf{R}_i}{\partial \alpha_k} + \mathbf{H}_i \frac{\partial \mathbf{Q}_i}{\partial \alpha_k} \mathbf{H}_i^T \right] \right\} \tag{C.13}$$

This equation represents the MLE solution for estimating either to process noise covariance or the measurement noise covariance, and independent analytic adaptation laws can be derived for both.

C.1. Adaptations for the process noise covariance

By setting $\frac{\partial \mathbf{R}_i}{\partial \alpha_k} = 0$, the adaptation equation for the process noise covariance is found by solving Eq. (C.13). Each diagonal entry of the process noise covariance is treated as an adaptable term, so at each time step the adaptation parameters $\alpha_n = \mathbf{Q}_{nn}$ correspond to the nth row or column index of the process noise matrix. By assuming that the noise covariance is diagonal, $\frac{\partial \mathbf{Q}_i}{\partial \alpha_k} = \mathbf{I}$, and Eq. (C.13) simplifies to

$$0 = \sum_{i=i_0}^{N} \operatorname{tr}\left\{ \left[\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \right] \left[\mathbf{0} + \mathbf{H}_i \mathbf{I} \mathbf{H}_i^T \right] \right\} \tag{C.14}$$

$$= \sum_{i=i_0}^{N} \operatorname{tr}\left\{ \left[\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \right] \mathbf{H}_i \mathbf{H}_i^T \right\} \tag{C.15}$$

Recalling the trace identity from Eq. (C.10), the result from Eq. (C.15) can be manipulated to

$$0 = \sum_{i=i_0}^{N} \mathbf{H}_i^T \left[\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \right] \mathbf{H}_i \tag{C.16}$$

$$= \sum_{i=i_0}^{N} \left[\mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1} \mathbf{H}_i - \mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \mathbf{H}_i \right] \tag{C.17}$$

Now, consider that the Kalman gain equation $\mathbf{K}_i = \mathbf{P}_i^- \mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1}$ can be rearranged and transposed to give

$$\boldsymbol{\Sigma}_i^{-1} \mathbf{H}_i = \mathbf{K}_i^T [\mathbf{P}_i^-]^{-1} \tag{C.18}$$

$$\mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1} = [\mathbf{P}_i^-]^{-1} \mathbf{K}_i \tag{C.19}$$

and then substitute these two terms into Eq. (C.17) to find

$$0 = \sum_{i=i_0}^{N} \left[[\mathbf{P}_i^-]^{-1} \mathbf{K}_i \mathbf{H}_i - [\mathbf{P}_i^-]^{-1} \mathbf{K}_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \mathbf{K}_i^T [\mathbf{P}_i^-]^{-1} \right] \tag{C.20}$$

$$= \sum_{i=i_0}^{N} \left[[\mathbf{P}_i^-]^{-1} \left(\mathbf{K}_i \mathbf{H}_i \mathbf{P}_i^- - \mathbf{K}_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \mathbf{K}_i^T \right) [\mathbf{P}_i^-]^{-1} \right] \tag{C.21}$$

It can further be realized that the covariance terms must be positive definite symmetric by definition, so the only way to solve Eq. (C.21) is to zero the inner term and proceed with

$$0 = \sum_{i=i_0}^{N} \left[\mathbf{K}_i \mathbf{H}_i \mathbf{P}_i^- - \mathbf{K}_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \mathbf{K}_i^T \right] \tag{C.22}$$

To continue simplifying the equation, expand the optimal *a posteriori* covariance equation [47] and rearrange, as follows:

$$\mathbf{P}_i^+ = \left(\mathbf{I} - \mathbf{K}_i \mathbf{H}_i \right) \mathbf{P}_i^- = \mathbf{P}_i^- - \mathbf{K}_i \mathbf{H}_i \mathbf{P}_i^- \tag{C.23}$$

$$\mathbf{P}_i^- - \mathbf{P}_i^+ = \mathbf{K}_i \mathbf{H}_i \mathbf{P}_i^- \tag{C.24}$$

Now, Eq. (C.24) is inserted into the Eq. (C.22) and yields

$$0 = \sum_{i=i_0}^{N} \left[\mathbf{P}_i^- - \mathbf{P}_i^+ - \mathbf{K}_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \mathbf{K}_i^T \right] \tag{C.25}$$

and substituting with the definition of the a priori error covariance $\mathbf{P}_i^-$ from Eq. (14) leads to

$$0 = \sum_{i=i_0}^{N} \left[\left(\boldsymbol{\Phi}_{i-1} \mathbf{P}_{i-1}^+ \boldsymbol{\Phi}_{i-1}^T + \mathbf{Q}_i \right) - \mathbf{P}_i^+ - \mathbf{K}_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \mathbf{K}_i^T \right] \tag{C.26}$$

Finally, by extracting the process noise covariance as the average of the cumulative adaptation parameter throughout the estimation window, the estimated process noise covariance $\hat{\mathbf{Q}}_k$ can be obtained as

$$\hat{\mathbf{Q}}_k = \frac{1}{N} \sum_{i=i_0}^{N} \left[\mathbf{K}_i \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \mathbf{K}_i^T + \left(\mathbf{P}_i^+ - \boldsymbol{\Phi}_{i-1} \mathbf{P}_{i-1}^+ \boldsymbol{\Phi}_{i-1}^T \right) \right] \tag{C.27}$$

Note that the parenthetical term in Eq. (C.27) contains the change in the error covariances between the current and previous time steps. If the filter has reached steady-state operation, as was assumed earlier in the derivation, then the term in parentheses is approximately zero, while the Kalman gain is roughly constant and can be taken outside of the averaged summation. The estimate of the process covariance then reduces to:

$$\hat{\mathbf{Q}}_k = \mathbf{K}_k \left[\frac{1}{N} \sum_{i=i_0}^{N} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \right] \mathbf{K}_k^T \tag{C.28}$$

The bracketed terms contain the residuals of the filter averaged over the N-steps of the fixed window, which has already been referred to as the observed residuals covariance matrix and is defined as

$$\hat{\boldsymbol{\Sigma}}_{k|N} \triangleq \frac{1}{N} \sum_{i=i_0}^{N} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \tag{C.29}$$

Thus, the final Q-adaptation law for the EKF can be re-written in the form

$$\hat{\mathbf{Q}}_k = \mathbf{K}_k \hat{\boldsymbol{\Sigma}}_{k|N} \mathbf{K}_k^T \tag{C.30}$$

C.2. Adaptations for the measurement noise covariance

Solving for the measurement noise adaptations uses the same procedure as above, except where $\mathbf{Q}_k$ is designated as the known and adaptation-independent parameter. Setting all $\frac{\partial \mathbf{Q}_i}{\partial \alpha_k} = 0$ and choosing the adaptation terms as $\alpha_n = \mathbf{R}_{nn}$, the derivatives in Eq. (C.13) yield

$$0 = \sum_{i=i_0}^{N} \operatorname{tr}\left\{ \boldsymbol{\Sigma}_i^{-1} \left[\boldsymbol{\Sigma}_i - \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \right] \boldsymbol{\Sigma}_i^{-1} \left[\mathbf{I} + \mathbf{0} \right] \right\} \tag{C.31}$$

Several trivial manipulations are needed next, but are shown here to be explicit. First, the trace identity from Eq. (C.10) (*i.e.*, $\mathbf{A}^T\mathbf{B} = \operatorname{tr}\left\{\mathbf{B}\mathbf{A}^T\right\}$), is used again to remove the trace operator from the equation:

$$0 = \sum_{i=i_0}^{N} \left[\mathbf{I}^T \right] \boldsymbol{\Sigma}_i^{-1} \left[\boldsymbol{\Sigma}_i - \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \right] \boldsymbol{\Sigma}_i^{-1} \tag{C.32}$$

Clearly $\mathbf{I}^T = \mathbf{I}$ can also be removed from the equation now, and inserting the definition of $\boldsymbol{\Sigma}_i$ from Eq. (27) then gives the new form of Eq. (C.32) as

$$0 = \sum_{i=i_0}^{N} \boldsymbol{\Sigma}_i^{-1} \left[\left(\mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T + \mathbf{R}_i \right) - \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \right] \boldsymbol{\Sigma}_i^{-1} \tag{C.33}$$

Invariance of the estimation process is assumed for the given memory window, implying that $\boldsymbol{\Sigma}_i^{-1}$ is approximately constant throughout the filtered data in memory [32,34,36]. Furthermore, the covariance $\boldsymbol{\Sigma}_i$ will also be positive definite, hence the solution for the R-adaptation equation focuses on the inner terms of Eq. (C.33) only. Consider then

$$0 = \sum_{i=i_0}^{N} \left[\left(\mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T + \mathbf{R}_i \right) - \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \right] \tag{C.34}$$

from which an explicit estimate of the measurement noise covariance is extracted to be

$$\hat{\mathbf{R}}_k = \frac{1}{N} \sum_{i=i_0}^{N} \left[\boldsymbol{\nu}_i \boldsymbol{\nu}_i^T - \mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T \right] \tag{C.35}$$

However, implementation of Eq. (C.35) has been shown to give poor estimates of $\mathbf{R}_k$ that break the positive definite constraints, so an alternate variation of this R-adaptation law is derived by using several other Kalman filter relations. Continuing from Eq. (C.33) and multiplying the inverse residual covariances through, it can be seen that

$$0 = \sum_{i=i_0}^{N} \left[\boldsymbol{\Sigma}_i^{-1} \left(\mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T + \mathbf{R}_i \right) \boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} \right] \tag{C.36}$$

Next, the identity relating the post-fit measurement estimates $\hat{\mathbf{z}}_i^+$ and the residuals is needed [32], which is given by

$$\boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i = \boldsymbol{\Sigma}_i^{-1} \left[\mathbf{z}_i - \hat{\mathbf{z}}_i^- \right] = \mathbf{R}_i^{-1} \left[\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right] \tag{C.37}$$

The residuals covariance $\boldsymbol{\Sigma}_i$ is positive definite symmetric, therefore its inverse will also be positive definite symmetric, and it is possible to construct

$$\boldsymbol{\Sigma}_i^{-1} \boldsymbol{\nu}_i \boldsymbol{\nu}_i^T \boldsymbol{\Sigma}_i^{-1} = \mathbf{R}_k^{-1} \left[\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right] \left[\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right]^T \mathbf{R}_i^{-1} \tag{C.38}$$

Replacing the right-hand term in Eq. (C.36) with the result from Eq. (C.38), and expanding the left-hand terms, the problem reduces to

$$0 = \sum_{i=i_0}^{N} \left[\left(\boldsymbol{\Sigma}_i^{-1} \mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1} + \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \right) - \mathbf{R}_i^{-1} \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T \mathbf{R}_i^{-1} \right] \tag{C.39}$$

and then multiplying both terms by $\mathbf{R}_i$ on the left and right, obtaining

$$0 = \sum_{i=i_0}^{N} \left[\mathbf{R}_i \left(\boldsymbol{\Sigma}_i^{-1} \mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1} + \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \right) \mathbf{R}_i - \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T \right] \tag{C.40}$$

A number of other relations are needed before proceeding further; first, an equivalent form of the Kalman Gain matrix can be found using the Matrix Inversion Lemma [47], which incorporates $\mathbf{R}_i$ through

$$\mathbf{K}_i = \mathbf{P}_i^- \mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1} = \mathbf{P}_i^+ \mathbf{H}_i^T \mathbf{R}_i^{-1} \tag{C.41}$$

Looking at the two terms that do not involve the Kalman gain, left-multiplying both sides of the equation by the measurement matrix $\mathbf{H}_i$ gives

$$\mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T \boldsymbol{\Sigma}_i^{-1} = \mathbf{H}_i \mathbf{P}_i^+ \mathbf{H}_i^T \mathbf{R}_i^{-1} \tag{C.42}$$

and by rearranging Eq. (27) to isolate the covariance of the residuals $\boldsymbol{\Sigma}_i$, it can be written that

$$\mathbf{H}_i \mathbf{P}_i^- \mathbf{H}_i^T = \boldsymbol{\Sigma}_i - \mathbf{R}_i \tag{C.43}$$

Now inserting Eq. (C.43) into the left side of Eq. (C.42), the result is

$$\left(\boldsymbol{\Sigma}_i - \mathbf{R}_i \right) \boldsymbol{\Sigma}_i^{-1} = \mathbf{I} - \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} = \mathbf{H}_i \mathbf{P}_i^+ \mathbf{H}_i^T \mathbf{R}_i^{-1} \tag{C.44}$$

With the necessary relations defined, proceed with the derivation by substituting Eq. (C.41) into Eq. (C.40), which changes the dependency from the *a priori* error covariance to the *a posteriori* state covariance:

$$0 = \sum_{i=i_0}^{N} \left[\mathbf{R}_k \left(\boldsymbol{\Sigma}_i^{-1} \mathbf{H}_i \mathbf{P}_i^+ \mathbf{H}_i^T \mathbf{R}_i^{-1} + \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \right) \mathbf{R}_i - \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T \right] \tag{C.45}$$

Further substituting Eq. (C.44) into the left-most term in parenthesis within Eq. (C.45) yields:

$$0 = \sum_{i=i_0}^{N} \left[\mathbf{R}_i \left(\boldsymbol{\Sigma}_i^{-1} \left(\mathbf{I} - \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \right) + \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \right) \mathbf{R}_i - \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T \right] \tag{C.46}$$

$$= \sum_{i=i_0}^{N} \left[\mathbf{R}_i \left(\boldsymbol{\Sigma}_i^{-1} - \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} + \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \right) \mathbf{R}_i - \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T \right] \tag{C.47}$$

$$= \sum_{i=i_0}^{N} \left[\mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i - \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T \right] \tag{C.48}$$

Quickly recall and rearrange Eq. (C.44), and then right-multiply it with $\mathbf{R}_i$ to obtain

$$\mathbf{R}_i \boldsymbol{\Sigma}_i^{-1} \mathbf{R}_i = \mathbf{R}_i - \mathbf{H}_i \mathbf{P}_i^+ \mathbf{H}_i^T \tag{C.49}$$

which can be substituted into Eq. (C.48), resulting in

$$0 = \sum_{i=i_0}^{N} \left[\mathbf{R}_i - \mathbf{H}_i \mathbf{P}_i^+ \mathbf{H}_i^T - \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T \right] \tag{C.50}$$

Finally, the measurement noise covariance can be estimated as the average of the cumulative adaptation parameter throughout the estimation window, and extracting $\mathbf{R}_k$ from the preceding equation gives

$$\hat{\mathbf{R}}_k = \frac{1}{N} \sum_{i=i_0}^{N} \left[\left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right) \left(\mathbf{z}_i - \hat{\mathbf{z}}_i^+ \right)^T + \mathbf{H}_i \mathbf{P}_i^+ \mathbf{H}_i^T \right] \tag{C.51}$$

Notice that in the equation above, the *a priori* covariance $\mathbf{P}_i^-$ has been replaced by the *a posteriori* covariance $\mathbf{P}_i^+$, and the pre-fit residuals $\boldsymbol{\nu}_i$ have been replaced with the post-fit residuals $\boldsymbol{\nu}_i^+$. defined by

$$\boldsymbol{\nu}_i^+ \triangleq \mathbf{z}_i - \hat{\mathbf{z}}_i^+ = \mathbf{z}_i - \mathbf{H}_i \hat{\mathbf{x}}_i^+ \tag{C.52}$$

Thus, the final form of the R-adaptation equation can be written as

$$\hat{\mathbf{R}}_k = \frac{1}{N}\sum_{i=i_0}^{N}\left[\boldsymbol{\nu}_i^{+}\boldsymbol{\nu}_i^{+T} + \mathbf{H}_i\mathbf{P}_i^{+}\mathbf{H}_i^{T}\right] \tag{C.53}$$

References

[1] A. Moccia, A. Renga, Distributed Space Missions for Earth System Monitoring, Springer Science & Business Media, 2013, pp. 3–59.

[2] G. Krieger, I. Hajnsek, K.P. Papathanassiou, M. Younis, A. Moreira, Interferometric synthetic aperture radar (SAR) missions employing formation flying, Proc. IEEE 98 (5) (2010) 816–843.

[3] B. Tapley, J. Ries, S. Bettadpur, D. Chambers, M. Cheng, F. Condi, B. Gunter, Z. Kang, P. Nagel, R. Pastor, T. Pekker, S. Poole, F. Wang, GGM02-an improved earth gravity field model from GRACE, J. Geod. 79 (8) (2005) 1–11.

[4] C.S. Cockell, T. Herbst, A. Léger, O. Absil, C. Beichman, W. Benz, A. Brack, B. Chazelas, A. Chelli, H. Cottin, Darwin - an experimental astronomy mission to search for extrasolar planets, Exp. Astron. 23 (1) (2009) 435–461.

[5] S. D'Amico, J. Ardaens, R. Larsson, Spaceborne autonomous formation-flying experiment on the PRISMA mission, J. Guid. Control Dyn. 35 (3) (2012) 834–850.

[6] G. Bonin, N. Roth, S. Armitage, J. Newman, B. Risi, R.E. Zee, CanX–4 and canX–5 precision formation flight: Mission accomplished!, in: Proceedings of the 29th Annual AIAA/USU Conference on Small Satellites, in: Technical Sessions I: All Systems Go!, Utah State University, 2015, pp. 1–15.

[7] G. Gaias, J.-S. Ardaens, Flight demonstration of autonomous noncooperative rendezvous in low earth orbit, J. Guidance Control Dyn. 41 (6) (2018) 1337–1354.

[8] G. Gaias, J. Ardaens, C. Schultz, The AVANTI experiment: Flight results, in: 10th International ESA Conference on Guidance, Navigaiton and Control System, ESA, 2017, pp. 1–15.

[9] R.E. Kalman, A new approach to linear filtering and prediction problems, Trans. ASME 82 (1) (1960) 35–45.

[10] B.P. Gibbs, Advanced Kalman Filtering, Least-Squares and Modeling: A Practical Handbook, Wiley, 2011, pp. 467–480.

[11] R. Mehra, On the identification of variances and adaptive Kalman filtering, IEEE Trans. Autom. Control 15 (2) (1970) 175–184.

[12] R. Mehra, Approaches to adaptive filtering, IEEE Trans. Autom. Control 17 (5) (1972) 693–698.

[13] R. Mehra, S. Seereeram, D. Bayard, F. Hadaegh, Adaptive Kalman filtering, failure detection and identification for spacecraft attitude estimation, in: Proceedings of the 4th IEEE Conference on Control Applications, IEEE, 1995, pp. 176–181.

[14] A.H. Mohamed, K.P. Schwarz, Adaptive Kalman filtering for INS/GPS, J. Geodesy 73 (4) (1999) 193–203.

[15] C. Hide, T. Moore, M. Smith, Adaptive Kalman filtering for low-cost INS/GPS, J. Navig. 56 (1) (2003) 143–152.

[16] F.D. Busse, J.P. How, J. Simpson, Demonstration of adaptive extended Kalman filter for low-earth-orbit formation estimation using CDGPS, Navig.: J. Inst. Navig. 50 (2) (2003) 79–93.

[17] F. Jiancheng, Y. Sheng, Study on innovation adaptive EKF for in-flight alignment of airborne POS, IEEE Trans. Instrum. Measur. 60 (4) (2011) 1378–1388.

[18] F.A. Ghaleb, A. Zainal, M.A. Rassam, A. Abraham, Improved vehicle positioning algorithm using enhanced innovation-based adaptive Kalman filter, Pervasive Mob. Comput. 40 (2017) 139–155.

[19] W. Li, D. Gong, M. Liu, J. Chen, D. Duan, Adaptive robust Kalman filter for relative navigation using global positioning system, IET Radar Sonar Navig. 7 (5) (2013) 471–479.

[20] C. Hu, W. Chen, Y. Chen, D. Liu, Adaptive Kalman filtering for vehicle navigation, J. Glob. Positioning Syst. 2 (1) (2003) 42–47.

[21] A. Chatterjee, F. Matsuno, A neuro-fuzzy assisted extended Kalman filter-based approach for simultaneous localization and mapping (SLAM) problems, IEEE Trans. Fuzzy Syst. 15 (5) (2007) 984–997.

[22] L.A. Zadeh, Fuzzy sets, Inf. Control 8 (3) (1965) 338–353.

[23] L.A. Zadeh, Fuzzy algorithms, Inf. Control 12 (2) (1968) 94–102.

[24] L.A. Zadeh, Outline of a new approach to the analysis of complex systems and decision processes, IEEE Trans. Syst. Man Cybern. 3 (1) (1973) 28–44.

[25] A.L. da Silva, J.J. da Cruz, Fuzzy adaptive extended Kalman filter for UAV INS/GPS data fusion, J. Braz. Soc. Mech. Sci. Eng. 38 (6) (2016) 1671–1688.

[26] J. Ali, Strapdown inertial navigation system/astronavigation system data synthesis using innovation-based fuzzy adaptive Kalman filtering, IET Sci. Measur. Technol. 4 (5) (2010) 246–255.

[27] C.-H. Tseng, S.-F. Lin, D.-J. Jwo, Fuzzy adaptive Cubature Kalman filter for integrated navigation systems, Sensors 16 (8) (2016) 1167–1189.

[28] E.A. Butcher, J. Wang, T.A. Lovell, On Kalman filtering and observability in nonlinear sequential relative orbit estimation, J. Guid. Control Dyn. 40 (9) (2017) 2167–2182.

[29] P.J. Huxel, R.H. Bishop, Navigation algorithms and observability analysis for formation flying missions, J. Guidance Control Dyn. 32 (4) (2009) 1218–1231.

[30] E. Kaufmann, T. Lovell, T. Lee, Nonlinear observability for relative satellite orbits with angles-only measurements, J. Astronaut. Sci. 63 (1) (2016) 60–80.

[31] A. de Ruiter, C. Damaren, J.R. Forbes, Spacecraft Dynamics and Control, John Wiley & Sons, Chichester, 2013, pp. 189–207, 459–497.

[32] P.S. Maybeck, Stochastic Models, Estimation and Control, Vol. 2, Academic Press, 1982, pp. 68–144.

[33] T. Kailath, An innovations approach to least-squares estimation–part I: Linear filtering in additive white noise, IEEE Trans. Autom. Control 13 (6) (1968) 646–655.

[34] V.A. Bavdekar, A.P. Deshpande, S.C. Patwardhan, Identification of process and measurement noise covariance for state and parameter estimation using extended Kalman filter, J. Process Control 21 (4) (2011) 585–601.

[35] A. Leon-Garcia, Probability, Statistics, and Random Processes for Electrical Engineering, third ed., Pearson/Prentice Hall, 2008, pp. 303–346.

[36] K.A. Myers, B.D. Tapley, Adaptive sequential estimation with unknown noise statistics, IEEE Trans. Autom. Control 21 (4) (1976) 520–523.

[37] H.E. Rauch, F. Tung, C.T. Striebel, Maximum likelihood estimates of linear dynamic systems, AIAA J. 3 (8) (1965) 1445–1450.

[38] D.J. Jwo, F.C. Chung, T.P. Weng, Adaptive Kalman Filter for Navigation Sensor Fusion, Sensor Fusion and its Applications, Intech, 2010, pp. 65–90.

[39] D.-J. Jwo, T.-S. Cho, A practical note on evaluating Kalman filter performance optimality and degradation, Appl. Math. Comput. 193 (2) (2007) 482–505.

[40] K. Passino, S. Yurkovich, Fuzzy Control, Addison-Wesley, 1998, pp. 51–98.

[41] S. Ulrich, J.Z. Sasiadek, Direct fuzzy adaptive control of a manipulator with elastic joints, J. Guidance Control Dyn. 36 (1) (2012) 311–319.

[42] T. Alfriend, S. Vadali, P. Gurfil, J. How, L. Breger, Spacecraft Formation Flying: Dynamics, Control and Navigation, Elsevier, 2010, pp. 59–82.

[43] O. Montenbruck, E. Gill, Satellite Orbits: Models, Methods, and Applications, Springer, 2001, pp. 53–104.

[44] G. Di Mauro, M. Lawn, R. Bevilacqua, Survey on guidance navigation and control requirements for spacecraft formation-flying missions, J. Guidance Control Dyn. (2017) 1–22.

[45] O. Montenbruck, M. Delpech, J. Ardaens, N. Delong, S. D'Amico, Cross-validation of GPS- and FFRF-based relative navigation for the PRISMA mission, in: 4th ESA Workshop on Satellite Navigation User Equipment Technologies NAVITEC2008, 2008, pp. 1–10.

[46] S. D'Amico, Autonomous Formation Flying in Low Earth Orbit (Ph.D. thesis), Technical University of Delft, 2010.

[47] A. Gelb (Ed.), Applied Optimal State Estimation, MIT Press, 1974, pp. 317–320.