Towards Consistent Visual-Inertial Navigation

Guoquan Huang
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
ghuang@csail.mit.edu

Abstract

Visual-inertial navigation systems (VINS) have prevailed in various applications, in part because of the complementary sensing capabilities and decreasing costs as well as sizes. While many of the current VINS algorithms undergo inconsistent estimation, in this paper we introduce a new extended Kalman filter (EKF)-based approach towards consistent estimates. To this end, we impose both state-transition and obervability constraints in computing EKF Jacobians so that the resulting linearized system can best approximate the underlying nonlinear system. Specifically, we enforce the propagation Jacobian to obey the semigroup property, thus being an appropriate state-transition matrix. This is achieved by parametrizing the orientation error state in the global, instead of local, frame of reference, and then evaluating the Jacobian at the propagated, instead of the updated, state estimates. Moreover, the EKF linearized system ensures correct observability by projecting the most-accurate measurement Jacobian onto the observable subspace so that no spurious information is gained. The proposed algorithm is validated by both Monte-Carlo simulation and real-world experimental tests.

1 Introduction

Over the past decades, inertial navigation systems (INS) [1] have been extensively used for estimating the 6 degrees-of-freedom (d.o.f.) poses of sensing platforms (a.k.a. robots) in GPS-denied environments, such as underwater, indoor, in the urban canyon, and on other planets. Most INS rely on an inertial measurement unit (IMU) that measures the 3 d.o.f. rotational velocity and 3 d.o.f. linear acceleration of the sensing platform on which it is rigidly attached. Unfortunately, simple integration of IMU measurements that are corrupted by noise and bias, often results in pose estimates unreliable for long-term navigation. On the other hand, a camera is small, light-weight, inexpensive, and energy efficient while providing rich information. We hence aid an INS with a monocular camera whose measurements are used to provide motion information of the sensor pair, i.e., visual-inertial navigation system (VINS). In this paper, we aim to develop a consistent estimation algorithm for this problem.

Various algorithms are available for VINS problems including visual-inertial simultaneous localization and mapping (SLAM) [2] and visual-inertial odometry (VIO) [3], such as the extended Kalman filter (EKF) [2,4,5], the unscented Kalman filter (UKF) [6], and the batch or incremental smoothers [7,8], among which the EKF-based approach remains arguably the most popular because of its efficiency. However, similar to 2D SLAM [9–11], the standard EKF produces inconsistent estimates when applied to VINS problems, primarily due to the mismatch of observability properties between the EKF linearized VINS and the underlying nonlinear system [3, 12–16]. This significantly limits a long-term deployment of VINS in critical scenarios. As defined in [17], a state estimator is *consistent* if the estimation errors are zero-mean, and the estimated covariance is equal to the true covariance. Consistency is one of the primary criteria for evaluating the performance of any estimator; if an estimator is inconsistent, then the accuracy of the computed state estimates is unknown, which in turn makes the estimator unreliable. In this paper, we also study the VINS problem within the EKF framework, while focusing on improving the filter consistency from the perspective of both state-transition *and* observability properties of the EKF linearized system.

In particular, as shown in [3, 12–16], the standard EKF-based VINS where the propagation and measurement Jacobians are evaluated at the latest state estimates, has different observability properties from the underlying nonlinear system (or the ideal linearized system where Jacobians are computed using the true states). This was shown to be one of main causes for the filter inconsistency. Furthermore, we analytically show for the first time that the propagation Jacobian in the standard EKF-based VINS violates the semigroup property of a state-transition matrix [18]. If such a Jacobian is used as the "state-transition" matrix to represent the underlying dynamical system, the produced state estimates conceivably may drift away from the solutions of the system, and thus become inconsistent or even diverge. To address the aforementioned two (observability and state-transition) issues, in the proposed algorithm, termed state-transition and observability constrained (STOC)-VINS, we first impose correct observability constraints as in [12–15]; and moreover, we *explicitly* enforce the propagation Jacobian to obey the semigroup property. This is achieved by parametrizing the orientation error state in the *global*, instead of *local*, frame of reference, and then directly evaluating the propagation Jacobian at the *propagated*, instead of the *updated*, state estimates. In addition, since in many practical cases the camera-IMU extrinsic calibration is not known perfectly, we include this 6 d.o.f. relative transformation as a part of the state vector and perform *online* calibration, which in effect contributes to improving consistency [16].

2 Related Work

Visual-inertial navigation has recently prevailed in robot localization in 3D (e.g., [2–8,12–16,19–26]), which can be broadly categorized into loosely-coupled and tightly-coupled approaches. The former processes the IMU measurements and/or images separately in a front end, and subsequently fuses them in a back end (e.g., [8, 23]). However, although this type of methods have advantage of computational efficiency, the decoupling results in information loss [16]. The latter seamlessly fuses the visual and inertial measurements by processing them in a single estimation thread (e.g., [3, 5, 12–16, 25, 26]). The approach proposed in this paper falls into the latter category, aiming at consistent VINS.

As system observability plays an important role in the proposed approach, we note that some work has recently studied the VINS observability properties under different scenarios. In particular, in [26, 27], nonlinear observability of IMU-camera extrinsic calibration was analyzed based on Lie derivatives and the conditions under which the IMU-camera transformation is observable were determined. In [25], the VINS observability was studied by examining the system's indistinguishable trajectories [28] under different sensor configurations. Similarly, Martinelli [21] employed the concept of continuous symmetries [28] to show that in VINS, the IMU biases, 3D velocity, and absolute roll and pitch angles are observable.

Recently, similar to robot localization in 2D [9–11], consistency of EKF-based VINS has been investigated in [3, 12–16] from the perspective of observability. Specifically, Li and Mourikis [3, 16] studied the impact of filter inconsistency due to the VINS observability properties, and leveraged the first-estimates-Jacobian methodology [9] to mitigate the inconsistency. In [12–15], following the observability-based methodology proposed in [11, 29], the observability-constrained (OC)-VINS was introduced, which can employ any linearization method to ensure correct observability of the linearized system. While the same observability-based idea is used in the proposed STOC-VINS, we further explicitly enforce the propagation Jacobian to satisfy the semigroup property and thus to be a valid state transition matrix, which results in an alternative way of computing propagation Jacobians to that of the OC-VINS.

3 Visual-Inertial Navigation

In this section, we first describe the IMU propagation and camera measurement models within the EKF framework, which govern the VINS. Then we briefly overview the observability properties of the linearized VINS, which will be useful for the design of our approach. For concise presentation of the analysis, we hereafter consider the case where only a single feature is included in the state vector, while the results can be easily generalized to the case of multiple features.

3.1 IMU propagation model

The EKF uses the IMU (gyroscope and accelerometer) measurements for state propagation, and the state vector consists of the IMU states $\mathbf{x}_I$ and the feature position ${}^G\mathbf{p}_f$:[1]

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_I^T & {}^G\mathbf{p}_f^T \end{bmatrix}^T = \begin{bmatrix} {}^I_G\bar{\mathbf{q}}^T & \mathbf{b}_g^T & {}^G\mathbf{v}^T & \mathbf{b}_a^T & {}^G\mathbf{p}^T & {}^G\mathbf{p}_f^T \end{bmatrix}^T \tag{1}$$

where ${}^I_G\bar{\mathbf{q}}$ is the unit quaternion that represents the rotation from the global frame of reference $\{G\}$ to the IMU frame $\{I\}$ (i.e., different parametrization of the rotation matrix $\mathbf{C}({}^I_G\bar{\mathbf{q}}) =: {}^I_G\mathbf{C}$); ${}^G\mathbf{p}$ and ${}^G\mathbf{v}$ are the IMU position and velocity in the global frame; and $\mathbf{b}_g$ and $\mathbf{b}_a$ denote the gyroscope and accelerometer biases, respectively.

By noting that the feature is static (with trivial dynamics), as well as using the IMU motion dynamics [30], the continuous-time dynamics of the state (1) is given by:

$$\begin{aligned} &{}^I_G\dot{\bar{\mathbf{q}}}(t) = \frac{1}{2}\boldsymbol{\Omega}\left({}^I\boldsymbol{\omega}(t)\right){}^I_G\bar{\mathbf{q}}(t)\ ,\ {}^G\dot{\mathbf{p}}(t) = {}^G\mathbf{v}(t)\ ,\ {}^G\dot{\mathbf{v}}(t) = {}^G\mathbf{a}(t) \\ &\dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg}(t)\ ,\ \dot{\mathbf{b}}_a(t) = \mathbf{n}_{wa}(t)\ ,\quad {}^G\dot{\mathbf{p}}_f(t) = \mathbf{0}_{3\times 1} \end{aligned} \tag{2}$$

where ${}^I\boldsymbol{\omega} = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^T$ is the rotational velocity of the IMU, expressed in $\{I\}$, ${}^G\mathbf{a}$ is the IMU acceleration in $\{G\}$, $\mathbf{n}_{wg}$ and $\mathbf{n}_{wa}$ are the white Gaussian noise processes that drive the IMU biases, and $\boldsymbol{\Omega}(\boldsymbol{\omega})$ is defined by:

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor\boldsymbol{\omega}\times\rfloor & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix}\ ,\ \lfloor\boldsymbol{\omega}\times\rfloor = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$$

A typical IMU provides gyroscope and accelerometer measurements, $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$, both of which are expressed in the IMU local frame $\{I\}$ and given by:

$$\boldsymbol{\omega}_m(t) = {}^I\boldsymbol{\omega}(t) + \mathbf{b}_g(t) + \mathbf{n}_g(t) \tag{3}$$

$$\mathbf{a}_m(t) = \mathbf{C}({}^I_G\bar{\mathbf{q}}(t))\left({}^G\mathbf{a}(t) - {}^G\mathbf{g}\right) + \mathbf{b}_a(t) + \mathbf{n}_a(t) \tag{4}$$

where ${}^G\mathbf{g}$ is the gravitational acceleration expressed in $\{G\}$, and $\mathbf{n}_g$ and $\mathbf{n}_a$ are zero-mean, white Gaussian noise.

Linearization of (2) at the current state estimates yields the continuous-time state-estimate propagation model [5]:

$$\begin{aligned} &{}^I_G\dot{\hat{\bar{\mathbf{q}}}}(t) = \frac{1}{2}\boldsymbol{\Omega}\left({}^I\hat{\boldsymbol{\omega}}(t)\right){}^I_G\hat{\bar{\mathbf{q}}}(t)\ ,\ {}^G\dot{\hat{\mathbf{p}}}(t) = {}^G\hat{\mathbf{v}}(t)\ ,\ {}^G\dot{\hat{\mathbf{v}}}(t) = {}^G\hat{\mathbf{a}}(t) \\ &\dot{\hat{\mathbf{b}}}_g(t) = \mathbf{0}_{3\times 1}\ ,\ \dot{\hat{\mathbf{b}}}_a(t) = \mathbf{0}_{3\times 1}\ ,\quad {}^G\dot{\hat{\mathbf{p}}}_f(t) = \mathbf{0}_{3\times 1} \end{aligned} \tag{5}$$

[1] Throughout this paper the subscript $\ell|j$ refers to the estimate of a quantity at time-step ℓ, after all measurements up to time-step j have been processed. $\hat{x}$ is used to denote the estimate of a random variable x, while $\tilde{x} = x - \hat{x}$ is the error in this estimate. $\mathbf{I}_n$ and $\mathbf{0}_n$ are the $n\times n$ identity and zero matrices, respectively. Finally, the left superscript denotes the frame of reference which the vector is expressed with respect to.

where $\hat{\mathbf{a}} = \mathbf{a}_m - \hat{\mathbf{b}}_a$ and $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\mathbf{b}}_g$. The error state of dimension 18×1 is hence defined as follows [see (1)]:

$$\widetilde{\mathbf{x}}(t) = \begin{bmatrix} {}^{I}\widetilde{\boldsymbol{\theta}}^{T}(t) & \widetilde{\mathbf{b}}_g^T(t) & {}^{G}\widetilde{\mathbf{v}}^T(t) & \widetilde{\mathbf{b}}_a^T(t) & {}^{G}\widetilde{\mathbf{p}}^T(t) & {}^{G}\widetilde{\mathbf{p}}_f^T(t) \end{bmatrix}^T \tag{6}$$

where we have employed the multiplicative error model for a quaternion [30]. That is, the error between the quaternion $\bar{\mathbf{q}}$ and its estimate $\hat{\bar{\mathbf{q}}}$ is the 3×1 angle-error vector, ${}^{I}\widetilde{\boldsymbol{\theta}}$, implicitly defined by the error quaternion: $\delta\bar{\mathbf{q}} = \bar{\mathbf{q}} \otimes \hat{\bar{\mathbf{q}}} \simeq \begin{bmatrix} \frac{1}{2}{}^{I}\widetilde{\boldsymbol{\theta}} \\ 1 \end{bmatrix}$, where $\delta\bar{\mathbf{q}}$ describes the small rotation that causes the true and estimated attitude to coincide. The advantage of this parametrization permits a minimal representation, 3×3 covariance matrix $\mathbb{E}\left[{}^{I}\widetilde{\boldsymbol{\theta}}\,{}^{I}\widetilde{\boldsymbol{\theta}}^{T}\right]$, for the attitude uncertainty. It is important to note that the orientation error, ${}^{I}\widetilde{\boldsymbol{\theta}}$, satisfies the following rotation-matrix relation [30]:

$$\mathbf{C}({}^{I}_{G}\bar{\mathbf{q}}) \simeq \left(\mathbf{I}_3 - \lfloor {}^{I}\widetilde{\boldsymbol{\theta}} \times \rfloor\right) \mathbf{C}({}^{I}_{G}\hat{\bar{\mathbf{q}}}) \tag{7}$$

Now the continuous-time error-state propagation is:

$$\dot{\widetilde{\mathbf{x}}}(t) = \mathbf{F}_c(t)\widetilde{\mathbf{x}}(t) + \mathbf{G}_c(t)\mathbf{n}(t) \tag{8}$$

where $\mathbf{n} = \begin{bmatrix} \mathbf{n}_g^T & \mathbf{n}_{wg}^T & \mathbf{n}_a^T & \mathbf{n}_{wa}^T \end{bmatrix}^T$ is the system noise, $\mathbf{F}_c$ is the continuous-time error-state transition matrix, and $\mathbf{G}_c$ is the input noise matrix, which are given by (see [30]):

$$\mathbf{F}_c = \begin{bmatrix} -\lfloor \hat{\boldsymbol{\omega}} \times \rfloor & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -\mathbf{C}^T({}^{I}_{G}\hat{\bar{\mathbf{q}}})\lfloor \hat{\mathbf{a}} \times \rfloor & \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{C}^T({}^{I}_{G}\hat{\bar{\mathbf{q}}}) & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \tag{9}$$

$$\mathbf{G}_c = \begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -\mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -\mathbf{I}_3 & \mathbf{0}_3 & -\mathbf{C}^T({}^{I}_{G}\hat{\bar{\mathbf{q}}}) & \mathbf{0}_3 \\ -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \\ -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \tag{10}$$

The system noise is modelled as zero-mean white Gaussian process with autocorrelation $\mathbb{E}\left[\mathbf{n}(t)\mathbf{n}(\tau)^T\right] = \mathbf{Q}_c\delta(t-\tau)$, which depends on the IMU noise characteristics.

We have presented the continuous-time propagation model using IMU measurements. However, in any practical EKF implementation, the discrete-time state-transition matrix, $\boldsymbol{\Phi}(t_{k+1}, t_k)$, is required in order to propagate the error covariance from time t_k to t_{k+1}. Typically it is found by solving the following matrix differential equation:

$$\dot{\boldsymbol{\Phi}}(t_{k+1}, t_k) = \mathbf{F}_c(t_{k+1})\boldsymbol{\Phi}(t_{k+1}, t_k) \tag{11}$$

with the initial condition $\mathbf{\Phi}(t_k,t_k) = \mathbf{I}_{18}$. And its solution has the following structure:

$$\mathbf{\Phi}_k := \mathbf{\Phi}(t_{k+1},t_k) = \begin{bmatrix} \mathbf{\Phi}_{k,11} & \mathbf{\Phi}_{k,12} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{\Phi}_{k,31} & \mathbf{\Phi}_{k,32} & \mathbf{I}_3 & \mathbf{\Phi}_{k,34} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{\Phi}_{k,51} & \mathbf{\Phi}_{k,52} & \delta t_k \mathbf{I}_3 & \mathbf{\Phi}_{k,54} & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{12}$$

where $\delta t_k = t_{k+1} - t_k$. This matrix (12) can be found either numerically [5, 30] or analytically [3, 13, 16]. Once it is computed, the EKF propagates the error covariance in a standard way [31]:

$$\mathbf{P}_{k+1|k} = \mathbf{\Phi}_k \mathbf{P}_{k|k} \mathbf{\Phi}_k^T + \mathbf{Q}_{d,k} \tag{13}$$

where $\mathbf{Q}_{d,k}$ is the discrete-time system noise covariance matrix computed as follows:

$$\mathbf{Q}_{d,k} = \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1},\tau)\mathbf{G}_c(\tau)\mathbf{Q}_c\mathbf{G}_c^T(\tau)\mathbf{\Phi}^T(t_{k+1},\tau)d\tau \tag{14}$$

3.2 Camera measurement model

The camera observes visual corner features, which are used to concurrently estimate the ego-motion of the sensing platform. Assuming a calibrated perspective camera, the measurement of the feature at time-step k is the perspective projection of the 3D point, ${}^{C_k}\mathbf{p}_f$, expressed in the current camera frame $\{C_k\}$, onto the image plane, i.e.,

$$\mathbf{z}_k = \frac{1}{z_k}\begin{bmatrix} x_k \\ y_k \end{bmatrix} + \mathbf{v}_k \tag{15}$$

$$\begin{bmatrix} x_k \\ y_k \\ z_k \end{bmatrix} = {}^{C_k}\mathbf{p}_f = \mathbf{C}({}^C_I\bar{\mathbf{q}})\mathbf{C}({}^I_G\bar{\mathbf{q}}_k)\left({}^G\mathbf{p}_f - {}^G\mathbf{p}_k\right) + {}^C\mathbf{p}_I \tag{16}$$

where $\mathbf{v}_k$ is the zero-mean, white Gaussian measurement noise with covariance $\mathbf{R}_k$. In (16), $\{{}^C_I\bar{\mathbf{q}}, {}^C\mathbf{p}_I\}$ is the rotation and translation between the camera and the IMU. This transformation can be obtained, for example, by performing camera-IMU extrinsic calibration *offline* [27]. However, in practice when the perfect calibration is unavailable, it is beneficial to VINS consistency if including these calibration parameters in the state vector and concurrently estimating them along with the IMU/camera poses [16]. For this reason, we perform *online* camera-IMU calibration in the proposed STOC-VINS (see Section 4).

For the use of EKF, linearization of (15) yields the following measurement residual [see (6)]:

$$\widetilde{\mathbf{z}}_k = \mathbf{H}_k\widetilde{\mathbf{x}}_{k|k-1} + \mathbf{v}_k = \mathbf{H}_{\mathbf{I}_k}\widetilde{\mathbf{x}}_{I_{k|k-1}} + \mathbf{H}_{\mathbf{f}_k}{}^G\widetilde{\mathbf{p}}_{f_{k|k-1}} + \mathbf{v}_k \tag{17}$$

where the measurement Jacobian $\mathbf{H}_k$ is computed as:

$$\mathbf{H}_k = \begin{bmatrix}\mathbf{H}_{\mathbf{I}_k} & \mathbf{H}_{\mathbf{f}_k}\end{bmatrix} = \mathbf{H}_{\mathbf{proj}}\mathbf{C}({}^C_I\bar{\mathbf{q}})\begin{bmatrix}\mathbf{H}_{\boldsymbol{\theta}_k} & \mathbf{0}_{3\times 9} & \mathbf{H}_{\mathbf{p}_k} & \mathbf{C}({}^I_G\hat{\bar{\mathbf{q}}}_k)\end{bmatrix} \tag{18}$$

$$\mathbf{H}_{\mathbf{proj}} = \frac{1}{\hat{z}_k^2}\begin{bmatrix}\hat{z}_k & 0 & -\hat{x}_k \\ 0 & \hat{z}_k & -\hat{y}_k\end{bmatrix} \tag{19}$$

$$\mathbf{H}_{\boldsymbol{\theta}_k} = \lfloor \mathbf{C}({}^I_G\hat{\bar{\mathbf{q}}}_k)\left({}^G\hat{\mathbf{p}}_f - {}^G\hat{\mathbf{p}}_k\right)\times\rfloor \,,\; \mathbf{H}_{\mathbf{p}_k} = -\mathbf{C}({}^I_G\hat{\bar{\mathbf{q}}}_k) \tag{20}$$

Once the measurement Jacobian and residual are computed, we can apply the standard EKF update equations to update the state estimates and error covariance [31].

3.3 Observability properties

Observability analysis has recently been performed for both nonlinear and linearized VINS [3, 13], which essentially extends our prior work [9, 10] from 2D to 3D. In particular, the observability matrix for the EKF linearized system over the time interval $[k_o\ k]$ is defined by [31]:

$$\mathbf{M} = \begin{bmatrix} \mathbf{H}_{k_o} \\ \mathbf{H}_{k_o+1}\mathbf{\Phi}_{k_o} \\ \vdots \\ \mathbf{H}_k\mathbf{\Phi}_{k-1}\cdots\mathbf{\Phi}_{k_o} \end{bmatrix} \tag{21}$$

It has been shown in [3, 13] that the nullspace of $\mathbf{M}$ (i.e., unobservable subspace) for the VINS *ideally* spans the following *four* directions:

$$\mathbf{MN} = \mathbf{0} \ \Rightarrow\ \mathbf{N} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{C}({}^I_G\bar{\mathbf{q}}_k)^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & -\lfloor {}^G\mathbf{v}_k \times \rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_k \times \rfloor {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_f \times \rfloor {}^G\mathbf{g} \end{bmatrix} \tag{22}$$

Note that the first block column of $\mathbf{N}$ (22) corresponds to the global translation while the second block column corresponds to the global rotation about the gravity vector ${}^G\mathbf{g}$. When designing a nonlinear estimator for VINS, we would like the system model employed by the estimator to have an unobservable subspace spanned by these directions. However, this is not the case for the standard EKF as shown in [3, 13]. In particular, the standard EKF linearized system, which linearizes system and measurement functions at the current state estimate, has an unobservable subspace of *three*, instead of four, d.o.f. This implies that the filter gains non-existent information from available measurements, which may lead to filter inconsistency.

4 State-Transition-and-Observability Constrained (STOC)-VINS

As discussed in the preceding section, the standard EKF-based VINS where the propagation and measurement Jacobians are evaluated at the latest state estimates, has different observability properties from the underlying nonlinear system or the ideal linearized system where the Jacobians are computed using the true states. This was shown to be *one* of main causes of filter inconsistency [3, 12, 13, 16]. In this section, we further study this issue: We analytically show, for the first time ever, that the propagation Jacobian of the standard EKF-based VINS violates the semigroup property of a state-transition matrix [18]. If such a Jacobian is used as the transition matrix to (approximate) represent the underlying dynamical system, the produced state estimates conceivably may deviate from the solutions of the dynamical system, and thus become inconsistent or even diverge. Moreover, when designing consistent VINS algorithms, besides imposing correct observability constraints in computing the measurement Jacobians, we explicitly enforce the propagation Jacobians to be appropriate state-transition matrices. The resulting method is thus termed as the State-Transition and Observability Constrained (STOC)-VINS.

4.1 Computing propagation Jacobians

We know from control theory that a state-transition matrix must have the following properties [18]:

$$\dot{\mathbf{\Phi}}(t_1,t_0) = \mathbf{F}_c(t_1)\mathbf{\Phi}(t_1,t_0) \tag{23}$$

$$\mathbf{\Phi}(t_0,t_0) = \mathbf{I}_{\dim(\mathbf{x})} \tag{24}$$

$$\mathbf{\Phi}(t_1,t_0) = \mathbf{\Phi}^{-1}(t_0,t_1) \tag{25}$$

$$\mathbf{\Phi}(t_2,t_0) = \mathbf{\Phi}(t_2,t_1)\mathbf{\Phi}(t_1,t_0) \tag{26}$$

which hold for any t_0, t_1 and t_2. Note that in VINS we have derived the analytical state-transition matrix by solving the matrix differential equation (11) with the self-mapping initial condition, which are identical to (23) and (24). Note also that given (26) and (24), the identity of (25) holds. Therefore, we hereafter focus on examining (26) which is so-called the *semigroup* property [18]. In particular, based on the following lemma, the propagation Jacobian of the standard EKF-based VINS is not a valid state-transition matrix:

Lemma 4.1. *The propagation Jacobian* (12) *of the standard EKF-based VINS, which is computed using the current state estimates, violates the semigroup property* (26) *of a state-transition matrix, i.e., for some* t_{k-1}, t_k*, and* t_{k+1}*,*

$$\mathbf{\Phi}(t_{k+1},t_{k-1}) \neq \mathbf{\Phi}(t_{k+1},t_k)\mathbf{\Phi}(t_k,t_{k-1}) \tag{27}$$

Proof. Note first that the standard EKF computes the propagation Jacobian, $\mathbf{\Phi}(t_k,t_\ell)$, using the current state estimates, $\hat{\mathbf{x}}_{k|k-1}$ and $\hat{\mathbf{x}}_{\ell|\ell}$. Based on the analytical form of the IMU propagation Jacobian found in [13], substituting the pertinent current estimates to these expressions, it is not difficult to verify that the standard EKF propagation Jacobians do not satisfy the semigroup property (26). In particular, the multiplication of the two propagation Jacobians, $\mathbf{\Phi}(t_{k+1},t_k)$ and $\mathbf{\Phi}(t_k,t_{k-1})$, does not alter the matrix structure as shown in (28).

$$\mathbf{\Phi}(t_{k+1},t_k)\mathbf{\Phi}(t_k,t_{k-1}) = \tag{28}$$

$$\begin{bmatrix}
\mathbf{\Phi}_{k,11}\mathbf{\Phi}_{k-1,11} & \mathbf{\Phi}_{k,11}\mathbf{\Phi}_{k-1,12}+\mathbf{\Phi}_{k,12} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\
\mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\
\mathbf{\Phi}_{k,31}\mathbf{\Phi}_{k-1,11}+\mathbf{\Phi}_{k-1,31} & \mathbf{\Phi}_{k,31}\mathbf{\Phi}_{k-1,12}+\mathbf{\Phi}_{k,32}+\mathbf{\Phi}_{k-1,32} & \mathbf{I}_3 & \mathbf{\Phi}_{k,34}+\mathbf{\Phi}_{k-1,34} & \mathbf{0}_3 & \mathbf{0}_3 \\
\mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\
\mathbf{\Phi}_{k,51}\mathbf{\Phi}_{k-1,11}+\delta t_k\mathbf{\Phi}_{k-1,31}+\mathbf{\Phi}_{k-1,51} & \mathbf{\Phi}_{k,51}\mathbf{\Phi}_{k-1,12}+\mathbf{\Phi}_{k,52}+\delta t_k\mathbf{\Phi}_{k-1,32}+\mathbf{\Phi}_{k-1,52} & (\delta t_k+\delta t_{k-1})\mathbf{I}_3 & \delta t_k\mathbf{\Phi}_{k-1,34}+\mathbf{\Phi}_{k,54}+\mathbf{\Phi}_{k-1,54} & \mathbf{I}_3 & \mathbf{0}_3 \\
\mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3
\end{bmatrix}$$

Now let us verify $(1,1)$ entry of the above multiplication. Substitution of the current state estimates into the analytical expressions of $\mathbf{\Phi}(t_{k+1},t_k)$ and $\mathbf{\Phi}(t_k,t_{k-1})$ [13] yields:

$$\mathbf{\Phi}_{k,11}\mathbf{\Phi}_{k-1,11} = \mathbf{C}\left({}^{I(k+1|k)}_{I(k|k)}\hat{\bar{\mathbf{q}}}\right)\mathbf{C}^T\left({}^{I(k-1|k-1)}_{I(k|k-1)}\hat{\bar{\mathbf{q}}}\right) \neq \mathbf{C}\left({}^{I(k+1|k)}_{I(k-1|k-1)}\hat{\bar{\mathbf{q}}}\right) = \mathbf{\Phi}_{11}(t_{k+1},t_{k-1}) \tag{29}$$

where we have employed the fact that the propagated estimate of the IMU orientation generally is different from its updated estimate, i.e., $\mathbf{C}\left({}^I_G\hat{\bar{\mathbf{q}}}_{k|k-1}\right) \neq \mathbf{C}\left({}^I_G\hat{\bar{\mathbf{q}}}_{k|k}\right)$. The above inequality (29) immediately completes the proof.

□

As a state-transition matrix is used to construct the general solution of the corresponding linear dynamical systems, an invalid state-transition matrix can result in an erroneous solution. Therefore, using the propagation Jacobian as the incorrect "transition" matrix for the EKF linearized VINS system conceivably may cause the filter producing inaccurate, or even inconsistent, estimates. To address this issue, we aim to construct the propagation Jacobian in such a way that enforces this Jacobian to be a valid state-transition matrix for the EKF linearized system, and in particular, to possess the semigroup property (26). The key idea of our approach is that we parametrize the IMU orientation error in the *global*, instead of *local* (as commonly used in the regular VINS formulation [5, 12, 13]), frame of reference; and then *analytically* compute the propagation Jacobian using the *propagated*, instead of *updated*, state estimates.

In particular, we first notice that [in contrast to (7)]:

$$\mathbf{C}\left({}^{I}_{G}\bar{\mathbf{q}}\right) \simeq \mathbf{C}\left({}^{I}_{G}\hat{\bar{\mathbf{q}}}\right)\left(\mathbf{I}_3 - \lfloor {}^{G}\widetilde{\boldsymbol{\theta}} \times \rfloor\right) \tag{30}$$

which results in the global orientation error state, ${}^{G}\widetilde{\boldsymbol{\theta}} = \mathbf{C}^T\left({}^{I}_{G}\hat{\bar{\mathbf{q}}}\right){}^{I}\widetilde{\boldsymbol{\theta}}$. With this parametrization, the error states except the biases are all in the global frame, which will be useful for our ensuing derivations [see (6)]:

$$\widetilde{\mathbf{x}}' := \begin{bmatrix} {}^{G}\widetilde{\boldsymbol{\theta}} \\ \widetilde{\mathbf{b}}_g \\ {}^{G}\widetilde{\mathbf{v}} \\ \widetilde{\mathbf{b}}_a \\ {}^{G}\widetilde{\mathbf{p}} \\ {}^{G}\widetilde{\mathbf{p}}_f \end{bmatrix} = \mathbf{Diag}\left(\mathbf{C}^T({}^{I}_{G}\hat{\bar{\mathbf{q}}}), \mathbf{I}_{15}\right) \begin{bmatrix} {}^{I}\widetilde{\boldsymbol{\theta}} \\ \widetilde{\mathbf{b}}_g \\ {}^{G}\widetilde{\mathbf{v}} \\ \widetilde{\mathbf{b}}_a \\ {}^{G}\widetilde{\mathbf{p}} \\ {}^{G}\widetilde{\mathbf{p}}_f \end{bmatrix} =: \boldsymbol{\Lambda}^T \widetilde{\mathbf{x}} \tag{31}$$

Now the new error-state propagation can be written as:

$$\widetilde{\mathbf{x}}'_{k+1|k} = \boldsymbol{\Lambda}^T_{k+1}\widetilde{\mathbf{x}}_{k+1|k} = \boldsymbol{\Lambda}^T_{k+1}\boldsymbol{\Phi}_k\boldsymbol{\Lambda}_k\widetilde{\mathbf{x}}'_{k|k} =: \boldsymbol{\Phi}'_k\widetilde{\mathbf{x}}'_{k|k} \tag{32}$$

where we have used the fact that $\boldsymbol{\Lambda}^{-1} = \boldsymbol{\Lambda}^T$ [see (31)]. Note that $\boldsymbol{\Phi}'_k := \boldsymbol{\Lambda}^T_{k+1}\boldsymbol{\Phi}_k\boldsymbol{\Lambda}_k$ is the propagation Jacobian for the new parametrization, and can be computed analytically based on the analytical expression of $\boldsymbol{\Phi}_k$ (see (12) and [13]):

$$\boldsymbol{\Phi}'_k := \boldsymbol{\Phi}'(t_{k+1}, t_k) = \begin{bmatrix} \mathbf{C}^T({}^{I}_{G}\hat{\bar{\mathbf{q}}}_{k+1})\boldsymbol{\Phi}_{k,11}\mathbf{C}({}^{I}_{G}\hat{\bar{\mathbf{q}}}_k) & \mathbf{C}^T({}^{I}_{G}\hat{\bar{\mathbf{q}}}_{k+1})\boldsymbol{\Phi}_{k,12} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}_{k,31}\mathbf{C}({}^{I}_{G}\hat{\bar{\mathbf{q}}}_k) & \boldsymbol{\Phi}_{k,32} & \mathbf{I}_3 & \boldsymbol{\Phi}_{k,34} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}_{k,51}\mathbf{C}({}^{I}_{G}\hat{\bar{\mathbf{q}}}_k) & \boldsymbol{\Phi}_{k,52} & \delta t_k\mathbf{I}_3 & \boldsymbol{\Phi}_{k,54} & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{33}$$

We now show that the propagation Jacobian, $\boldsymbol{\Phi}'(t_{k+1}, t_k)$, can be constructed analytically so as to satisfy the semigroup property (26) for a valid state-transition matrix.

Lemma 4.2. *If the propagation Jacobian $\boldsymbol{\Phi}'(t_{\ell+1}, t_\ell)$ is evaluated at the propagated state estimates, $\hat{\mathbf{x}}_{\ell+1|\ell}$ and $\hat{\mathbf{x}}_{\ell|\ell-1}$, then it satisfies the semigroup property* (26)*, i.e.,*

$$\boldsymbol{\Phi}'(t_{k+1}, t_{k-1}) = \boldsymbol{\Phi}'(t_{k+1}, t_k)\boldsymbol{\Phi}'(t_k, t_{k-1}) \tag{34}$$

Proof. First of all, by using the propagated state estimates in computing the $\boldsymbol{\Phi}'(t_{k+1}, t_k)$ for all k, we ensure that the same value (linearization point) $\hat{\mathbf{x}}_{k|k-1}$ is always used for the state variable, $\mathbf{x}_k$. As compared to the *ideal* propagation Jacobian that is evaluated at the true states (which, in fact, guarantees a same value always used for the same state), we simply use a different (less accurate) value in computing this propagation Jacobian while having the exactly same structure. Therefore, directly using the analytical expression of $\boldsymbol{\Phi}(t_{k+1}, t_k)$ [13], we obtain the analytical expression of $\boldsymbol{\Phi}'(t_{k+1}, t_k)$ [see (33)]. For simplicity of notations, we hereafter denote the propagated state estimate, $\hat{\mathbf{x}}_{k|k-1}$, simply by $\hat{\mathbf{x}}_k$.

In particular, as compared to $\boldsymbol{\Phi}(t_{k+1}, t_k)$, the only blocks changed are $\boldsymbol{\Phi}'_{k,11}$, $\boldsymbol{\Phi}'_{k,12}$, $\boldsymbol{\Phi}'_{k,31}$, and $\boldsymbol{\Phi}'_{k,51}$, which we analytically compute as follows:

$$\boldsymbol{\Phi}'_{k,11} = \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}_{k+1})\mathbf{C}\left({}^{I(k+1)}_{I(k)}\hat{\bar{\mathbf{q}}}\right)\mathbf{C}({}^I_G\hat{\bar{\mathbf{q}}}_k) = \mathbf{I}_3 \tag{35}$$

$$\boldsymbol{\Phi}'_{k,12} = -\int_{t_k}^{t_{k+1}} \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}(t_\tau))d\tau \tag{36}$$

$$\boldsymbol{\Phi}'_{k,31} = -\lfloor\left({}^G\hat{\mathbf{v}}_{k+1} - {}^G\hat{\mathbf{v}}_k - {}^G\mathbf{g}\delta t_k\right)\times\rfloor \tag{37}$$

$$\boldsymbol{\Phi}'_{k,51} = -\left\lfloor\left({}^G\hat{\mathbf{p}}_{k+1} - {}^G\hat{\mathbf{p}}_k - {}^G\hat{\mathbf{v}}_k\delta t_k - \frac{1}{2}{}^G\mathbf{g}\delta t_k^2\right)\times\right\rfloor \tag{38}$$

Using these new entries as well as the other ones whose analytical expressions are given in [13], we have $\boldsymbol{\Phi}'(t_{k+1}, t_k)$ in the following simplified form:

$$\boldsymbol{\Phi}'(t_{k+1}, t_k) = \begin{bmatrix} \mathbf{I}_3 & \boldsymbol{\Phi}'_{k,12} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,31} & \boldsymbol{\Phi}_{k,32} & \mathbf{I}_3 & \boldsymbol{\Phi}_{k,34} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,51} & \boldsymbol{\Phi}_{k,52} & \delta t_k\mathbf{I}_3 & \boldsymbol{\Phi}_{k,54} & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{39}$$

In what follows, we begin with the case of no biases which plays the most important role in design of the proposed approach (by noting that the biases are modelled as random walk processes), and subsequently examine the case with biases.

Case I: Without biases We first examine the case of *no* bias. In this case, we have $\boldsymbol{\Phi}'(t_{k+1}, t_k)$ as follows (by removing the entries involving biases, $\mathbf{b}_g$ and $\mathbf{b}_a$):

$$\boldsymbol{\Phi}'(t_{k+1}, t_k) = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,31} & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,51} & \delta t_k\mathbf{I}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{40}$$

The produce of the two propagation Jacobians is:

$$\boldsymbol{\Gamma} := \boldsymbol{\Phi}'(t_{k+1}, t_k)\boldsymbol{\Phi}'(t_k, t_{k-1}) = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,31} & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,51} & \delta t_k\mathbf{I}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k-1,31} & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k-1,51} & \delta t_{k-1}\mathbf{I}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{41}$$

$$= \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,31} + \boldsymbol{\Phi}'_{k-1,31} & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}'_{k,51} + \delta t_k\boldsymbol{\Phi}'_{k-1,31} + \boldsymbol{\Phi}'_{k-1,51} & (\delta t_{k-1} + \delta t_k)\mathbf{I}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{42}$$

where

$$
\begin{aligned}
\boldsymbol{\Gamma}_{31} &= \boldsymbol{\Phi}'_{k,31} + \boldsymbol{\Phi}'_{k-1,31} && (43)\\
&= -\lfloor \left({}^G\hat{\mathbf{v}}_{k+1} - {}^G\hat{\mathbf{v}}_{k-1} - {}^G\mathbf{g}(\delta t_{k-1} + \delta t_k) \right) \times \rfloor && (44)\\
&= \boldsymbol{\Phi}'_{31}(t_{k+1}, t_{k-1}) && (45)\\
\boldsymbol{\Gamma}_{51} &= \boldsymbol{\Phi}'_{k,51} + \delta t_k \boldsymbol{\Phi}'_{k-1,31} + \boldsymbol{\Phi}'_{k-1,51} && (46)\\
&= -\left\lfloor \left({}^G\hat{\mathbf{p}}_{k+1} - {}^G\hat{\mathbf{p}}_{k} - {}^G\hat{\mathbf{v}}_{k}\delta t_k - \frac{1}{2}{}^G\mathbf{g}\delta t_k^2 \right) \times \right\rfloor - \delta t_k \lfloor \left({}^G\hat{\mathbf{v}}_{k} - {}^G\hat{\mathbf{v}}_{k-1} - {}^G\mathbf{g}\delta t_{k-1} \right) \times \rfloor && (47)\\
&\quad - \left\lfloor \left({}^G\hat{\mathbf{p}}_{k} - {}^G\hat{\mathbf{p}}_{k-1} - {}^G\hat{\mathbf{v}}_{k-1}\delta t_{k-1} - \frac{1}{2}{}^G\mathbf{g}\delta t_{k-1}^2 \right) \times \right\rfloor\\
&= -\left\lfloor \left({}^G\hat{\mathbf{p}}_{k+1} - {}^G\hat{\mathbf{p}}_{k-1} - {}^G\hat{\mathbf{v}}_{k-1}(\delta t_k + \delta t_{k-1}) - \frac{1}{2}{}^G\mathbf{g}(\delta t_k + \delta t_{k-1})^2 \right) \times \right\rfloor && (48)\\
&= \boldsymbol{\Phi}'_{51}(t_{k+1}, t_{k-1}) && (49)
\end{aligned}
$$

Note that the other (trivial) entries are easy to verify. Thus, this completes the proof for the case of no biases, i.e., the propagation Jacobian evaluated at the propagated state estimates satisfies the semigroup property for a valid state transition matrix. Now let us consider the case with biases.

Case II: With biases Similarly, the product of the two propagation Jacobians is given by [see (28)]:

$$
\boldsymbol{\Gamma} := \boldsymbol{\Phi}'(t_{k+1}, t_k)\boldsymbol{\Phi}'(t_k, t_{k-1}) = \tag{50}
$$

$$
\begin{bmatrix}
\mathbf{I}_3 & \boldsymbol{\Phi}'_{k-1,12} + \boldsymbol{\Phi}'_{k,12} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\
\mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\
\boldsymbol{\Phi}'_{k,31} + \boldsymbol{\Phi}'_{k-1,31} & \boldsymbol{\Phi}'_{k,31}\boldsymbol{\Phi}'_{k-1,12} + \boldsymbol{\Phi}'_{k,32} + \boldsymbol{\Phi}'_{k-1,32} & \mathbf{I}_3 & \boldsymbol{\Phi}'_{k,34} + \boldsymbol{\Phi}'_{k-1,34} & \mathbf{0}_3 & \mathbf{0}_3\\
\mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3\\
\boldsymbol{\Phi}'_{k,51} + \delta t_k \boldsymbol{\Phi}'_{k-1,31} + \boldsymbol{\Phi}'_{k-1,51} & \boldsymbol{\Phi}'_{k,51}\boldsymbol{\Phi}'_{k-1,12} + \boldsymbol{\Phi}'_{k,52} + \delta t_k \boldsymbol{\Phi}'_{k-1,32} + \boldsymbol{\Phi}'_{k-1,52} & (\delta t_k + \delta t_{k-1})\mathbf{I}_3 & \delta t_k \boldsymbol{\Phi}'_{k-1,34} + \boldsymbol{\Phi}'_{k,54} + \boldsymbol{\Phi}'_{k-1,54} & \mathbf{I}_3 & \mathbf{0}_3\\
\mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3
\end{bmatrix}
$$

Similarly, we examine the non-trivial blocks (by using the analytical expression of $\boldsymbol{\Phi}_k$ [13] as well as the ones we found above). Note first that here $\boldsymbol{\Gamma}_{31}$ and $\boldsymbol{\Gamma}_{51}$ are exactly same as in the case of no biases [see (45) and (49)].

$$
\begin{aligned}
\boldsymbol{\Gamma}_{12} &= \boldsymbol{\Phi}'_{k-1,12} + \boldsymbol{\Phi}'_{k,12} && (51)\\
&= -\int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}(t_\tau))d\tau - \int_{t_k}^{t_{k+1}} \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}(t_\tau))d\tau && (52)\\
&= -\int_{t_{k-1}}^{t_{k+1}} \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}(t_\tau))d\tau && (53)\\
&= \boldsymbol{\Phi}'_{12}(t_{k+1}, t_{k-1}) && (54)
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\Gamma}_{34} &= \boldsymbol{\Phi}'_{k,34} + \boldsymbol{\Phi}'_{k-1,34} && (55)\\
&= -\int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}(t_\tau))d\tau - \int_{t_k}^{t_{k+1}} \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}(t_\tau))d\tau && (56)\\
&= -\int_{t_{k-1}}^{t_{k+1}} \mathbf{C}^T({}^I_G\hat{\bar{\mathbf{q}}}(t_\tau))d\tau && (57)\\
&= \boldsymbol{\Phi}'_{34}(t_{k+1}, t_{k-1}) && (58)
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\Gamma}_{54} = & \ \delta t_k \boldsymbol{\Phi}'_{k-1,34} + \boldsymbol{\Phi}'_{k,54} + \boldsymbol{\Phi}'_{k-1,54} && (59)\\
= & -\delta t_k \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau - \int_{t_k}^{t_{k+1}} \int_{t_k}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds - \int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds && (60)\\
= & -\int_{t_k}^{t_{k+1}} \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds - \int_{t_k}^{t_{k+1}} \int_{t_k}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds - \int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds && (61)\\
= & -\int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds - \int_{t_k}^{t_{k+1}} \left(\int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau + \int_{t_k}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau \right) ds && (62)\\
= & -\int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds - \int_{t_k}^{t_{k+1}} \left(\int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau \right) ds && (63)\\
= & -\int_{t_{k-1}}^{t_{k+1}} \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau ds && (64)\\
= & \ \boldsymbol{\Phi}'_{54}(t_{k+1}, t_{k-1}) && (65)
\end{aligned}
$$

$$
\begin{aligned}
\boldsymbol{\Gamma}_{32} =\ & \boldsymbol{\Phi}'_{k,31}\boldsymbol{\Phi}'_{k-1,12} + \boldsymbol{\Phi}'_{k,32} + \boldsymbol{\Phi}'_{k-1,32} && (66)\\
=\ & \lfloor \left({}^{G}\hat{\mathbf{v}}_{k+1} - {}^{G}\hat{\mathbf{v}}_{k} - {}^{G}\mathbf{g}\delta t_k\right) \times \rfloor \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}\right) d\tau && (67)\\
& + \int_{t_k}^{t_{k+1}} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
& + \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
=\ & \delta t_k \lfloor ({}^{G}\mathbf{a} - {}^{G}\mathbf{g}) \times \rfloor \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}\right) d\tau && (68)\\
& + \int_{t_k}^{t_{k+1}} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
& + \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
=\ & \int_{t_k}^{t_{k+1}} \lfloor ({}^{G}\mathbf{a} - {}^{G}\mathbf{g}) \times \rfloor ds \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \mathbf{C}^T\left({}^{I(s)}_{I(\tau)}\hat{\bar{\mathbf{q}}}\right) d\tau && (69)\\
& + \int_{t_k}^{t_{k+1}} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
& + \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
=\ & \int_{t_k}^{t_{k+1}} \lfloor ({}^{G}\mathbf{a} - {}^{G}\mathbf{g}) \times \rfloor \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) ds \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{I(\tau)}\hat{\bar{\mathbf{q}}}\right) d\tau && (70)\\
& + \int_{t_k}^{t_{k+1}} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
& + \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
=\ & \int_{t_k}^{t_{k+1}} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{I(\tau)}\hat{\bar{\mathbf{q}}}\right) d\tau ds && (71)\\
& + \int_{t_k}^{t_{k+1}} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
& + \int_{t_{k-1}}^{t_k} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds \\
=\ & \int_{t_{k-1}}^{t_{k+1}} \mathbf{C}^T\left({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}\right) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T\left({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}\right) d\tau ds && (72)\\
=\ & \boldsymbol{\Phi}'_{32}(t_{k+1}, t_{k-1}) && (73)
\end{aligned}
$$

where in the third equality we have used that ${}^{G}\hat{\mathbf{v}}_{k+1} = {}^{G}\hat{\mathbf{v}}_{k} + {}^{G}\mathbf{a}\delta t_k$, and in the sixth equality we have employed the identity $\lfloor \mathbf{C}\mathbf{a}\times \rfloor = \mathbf{C}\lfloor \mathbf{a}\times \rfloor \mathbf{C}^T$.

$$
\begin{aligned}
\boldsymbol{\Gamma}_{52} = \; & \boldsymbol{\Phi}'_{k,51}\boldsymbol{\Phi}'_{k-1,12} + \delta t_k \boldsymbol{\Phi}'_{k-1,32} + \boldsymbol{\Phi}'_{k,52} + \boldsymbol{\Phi}'_{k-1,52} && (74)\\
= \; & \left\lfloor \left({}^G\hat{\mathbf{p}}_{k+1} - {}^G\hat{\mathbf{p}}_k - {}^G\hat{\mathbf{v}}_k \delta t_k - \frac{1}{2}{}^G\mathbf{g}\delta t_k^2 \right) \times \right\rfloor \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau && (75)\\
& + \delta t_k \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds \\
& + \int_{t_k}^{t_{k+1}} \int_{t_k}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
& + \int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
= \; & \frac{1}{2}\delta t_k^2 \lfloor ({}^G\mathbf{a} - {}^G\mathbf{g}) \times \rfloor \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(\tau)}_{G}\hat{\bar{\mathbf{q}}}) d\tau && (76)\\
& + \delta t_k \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds \\
& + \int_{t_k}^{t_{k+1}} \int_{t_k}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
& + \int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
= \; & \int_{t_k}^{t_{k+1}} \int_{t_k}^{\iota} ds d\iota \lfloor ({}^G\mathbf{a} - {}^G\mathbf{g}) \times \rfloor \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau && (77)\\
& + \int_{t_k}^{t_{k+1}} d\iota \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds \\
& + \int_{t_k}^{t_{k+1}} \int_{t_k}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
& + \int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
= \; & \int_{t_k}^{t_{k+1}} \int_{t_k}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \left\lfloor {}^{I(s)}\mathbf{a}\times \right\rfloor \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota && (78)\\
& + \int_{t_k}^{t_{k+1}} \int_{t_{k-1}}^{t_k} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
& + \int_{t_k}^{t_{k+1}} \int_{t_k}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_k}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
& + \int_{t_{k-1}}^{t_k} \int_{t_{k-1}}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota \\
= \; & \int_{t_{k-1}}^{t_{k+1}} \int_{t_{k-1}}^{\iota} \mathbf{C}^T({}^{I(s)}_{G}\hat{\bar{\mathbf{q}}}) \lfloor {}^{I(s)}\mathbf{a}\times \rfloor \int_{t_{k-1}}^{s} \mathbf{C}^T({}^{I(\tau)}_{I(s)}\hat{\bar{\mathbf{q}}}) d\tau ds d\iota && (79)\\
= \; & \boldsymbol{\Phi}'_{52}(t_{k+1}, t_{k-1}) && (80)
\end{aligned}
$$

where in the third equality we have used that ${}^G\hat{\mathbf{p}}_{k+1} = {}^G\hat{\mathbf{p}}_k + {}^G\hat{\mathbf{v}}_k\delta t_k + \frac{1}{2}{}^G\mathbf{a}\delta t_k^2$. and in the fourth equality we have employed the identity $\lfloor \mathbf{C}\mathbf{a}\times \rfloor = \mathbf{C}\lfloor \mathbf{a}\times \rfloor \mathbf{C}^T$. With this we complete the proof. □

4.2 Computing measurement Jacobians

The measurement Jacobian with respect to the new error state (31) is calculated as follows [see (15) and (18)-(20)]:

$$\mathbf{H}'_k = \begin{bmatrix}\mathbf{H}'_{\mathbf{I}_k} & \mathbf{H}'_{\mathbf{f}_k}\end{bmatrix} = \mathbf{H}_{\mathbf{proj}}\mathbf{C}({}^C_I\bar{\mathbf{q}})\begin{bmatrix}\mathbf{H}'_{\boldsymbol{\theta}_k} & \mathbf{0}_{3\times 9} & \mathbf{H}'_{\mathbf{p}_k} & \mathbf{C}({}^I_G\hat{\bar{\mathbf{q}}}_k)\end{bmatrix} \tag{81}$$

$$\mathbf{H}'_{\boldsymbol{\theta}_k} = \lfloor\left({}^G\hat{\mathbf{p}}_f - {}^G\hat{\mathbf{p}}_k\right)\times\rfloor\ ,\ \ \mathbf{H}'_{\mathbf{p}_k} = -\mathbf{I}_3 \tag{82}$$

We notice first that by performing observability analysis similar to [13], the *ideal* linearized error-state system (i.e., Jacobians are computed using the true states) with the global orientation-error parametrization has the following unobservable subspace of 4 d.o.f. [see (22)]:

$$\mathbf{N}' = \begin{bmatrix}\mathbf{0}_3 & {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & -\lfloor {}^G\mathbf{v}_k\times\rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_k\times\rfloor {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_f\times\rfloor {}^G\mathbf{g}\end{bmatrix} \tag{83}$$

However, in analogy to the case of standard EKF-based VINS (see Sectioin 3.3), it is not difficult to show that if we compute the measurement Jacobian using the current best state estimates as for the standard EKF, while computing the propagation Jacobian using the propagated state estimates as devised in the previous section, the resulting linearized error-state system has an unobservable subspace of only 3 (instead of 4) d.o.f. which may still result in inconsistent estimates.

To address this issue, we impose appropriate observability constraints when computing measurement Jacobians, by following our prior observability-constrained methodology for designing consistent SLAM estimators [11, 29], which was also exploited in [12, 13]. Specifically, when computing the measurement Jacobian, we enforce that each block row of the observability matrix (21) has the same nullspace, i.e.,

$$\min_{\mathbf{H}'_k} \ ||\mathbf{H}'_k - \mathbf{H}_k||_F^2 \tag{84}$$

$$\text{subject to } \ \mathbf{H}'_k\boldsymbol{\Phi}'_{k-1}\cdots\boldsymbol{\Phi}'_{k_o}\mathbf{N}' = \mathbf{0} \tag{85}$$

where $||\cdot||_F$ denotes the Frobenius norm. Ideally, $\mathbf{H}_k$ in (84) is the measurement Jacobian computed using the true states, which, however, is not realizable in practice. Hence, we employ the latest, and thus the best, state estimates to compute this Jacobian as for the standard EKF, i.e., $\mathbf{H}_k = \mathbf{H}_k(\hat{\mathbf{x}}_{k|k-1})$. On the other hand, $\mathbf{N}'$ in (85) essentially is a design choice that defines the desired nullspace. Although we would like to have the same one as in (83) computed using the true states, we select a nullspace that has the same structure as in (83) while computing it with the first available state estimates, i.e., $\mathbf{N}' = \mathbf{N}'(\hat{\mathbf{x}}_{k_o|k_o})$.

Once the choice of the nullspace $\mathbf{N}'$ is made, we find the optimal solution to the above problem (84)-(85) in *closed form* based on the following lemma:

Lemma 4.3. *The optimal solution to the constrained minimization problem* (84)-(85) *is given by:*

$$\mathbf{H}'_k = \mathbf{H}_k\left(\mathbf{I}_{\dim(\mathbf{x})} - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\right) \tag{86}$$

where $\mathbf{U} = \boldsymbol{\Phi}'_{k-1}\cdots\boldsymbol{\Phi}'_{k_o}\mathbf{N}'$.

Proof. The constraint (85) states that the rows of $\mathbf{H}_k$ lie in the left nullspace of the matrix $\mathbf{U}$. Therefore, if $\mathbf{L}$ is a matrix whose rows span this nullspace, then $\mathbf{H}_k$ can be written as:

$$\mathbf{H}'_k = \boldsymbol{\Theta}\mathbf{L} \tag{87}$$

where $\boldsymbol{\Theta}$ is the unknown matrix we seek to find. We note that there are several possible ways of computing an appropriate matrix $\mathbf{L}$, whose rows lie in the nullspace of $\mathbf{U}$. For instance, such a matrix is given in closed form by:

$$\mathbf{L} = \begin{bmatrix} \mathbf{I}_m & \mathbf{0}_{m\times(n-m)} \end{bmatrix} \left(\mathbf{I}_n - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T \right) =: \boldsymbol{\Gamma}\boldsymbol{\Pi} \tag{88}$$

where m is the dimension of the measurement, and n is the dimension of the state vector. It is not difficult to see that $\boldsymbol{\Pi} := \mathbf{I}_n - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T$ is an orthogonal projection matrix (i.e., $\boldsymbol{\Pi}^2 = \boldsymbol{\Pi}$ and $\boldsymbol{\Pi}^T = \boldsymbol{\Pi}$) and hence has the eigenvalues of either 1 or 0, whose reduced SVD is given by $\boldsymbol{\Pi} = \mathbf{Q}\mathbf{Q}^T$. Using this result, $\mathbf{L}^T$ immediately can be written as $\mathbf{L}^T = \mathbf{Q}\mathbf{Q}^T\boldsymbol{\Gamma}^T$. By substituting this identity into the cost function, we have:

$$\begin{aligned} &\min\ ||\mathbf{H}'_k - \mathbf{H}_k||_F^2 = ||\mathbf{Q}^T\boldsymbol{\Gamma}^T\boldsymbol{\Theta}^T - \mathbf{Q}^T\mathbf{H}_k{}^T||_F^2 \\ &\Rightarrow\ \ \boldsymbol{\Theta} = \mathbf{H}_k\mathbf{Q}\left(\boldsymbol{\Gamma}\mathbf{Q}\right)^{-1} \end{aligned} \tag{89}$$

Therefore, substitution of the above equation in (87) yields:

$$\begin{aligned} \mathbf{H}'_k &= \mathbf{H}_k\mathbf{Q}\left(\boldsymbol{\Gamma}\mathbf{Q}\right)^{-1}\boldsymbol{\Gamma}\mathbf{Q}\mathbf{Q}^T = \mathbf{H}_k\mathbf{Q}\mathbf{Q}^T = \mathbf{H}_k\boldsymbol{\Pi} \\ &= \mathbf{H}_k\left(\mathbf{I}_n - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\right) \end{aligned} \tag{90}$$

This completes the proof. □

It is interesting to note that $\mathbf{U}$ in the above lemma is the propagated unobservable subspace (nullspace) at time-step k, and $\left(\mathbf{I}_{\dim(\mathbf{x})} - \mathbf{U}(\mathbf{U}^T\mathbf{U})^{-1}\mathbf{U}^T\right)$ is the subspace orthogonal to $\mathbf{U}$, i.e., the observable subspace. Hence, as seen from (86), the measurement Jacobian of the proposed STOC-VINS is the projection of the most accurate measurement Jacobian onto the observable subspace.

4.3 Application to MSCKF

The multi-state constraint Kalman filter (MSCKF) [5, 32] is a well-known VINS algorithm that performs tightly-coupled VIO over a sliding window of m poses, and has complexity only linear in the number of observed features. The MSCKF utilizes all feature observations available within the sliding window to impose probabilistic constraints between poses, without building a map. In what follows, we apply the proposed STOC-VINS to the MSCKF framework to address the VIO problem, while our methodology is flexible enough for many other VINS problems including SLAM. Note that this is a straightforward extension of the OC-VINS applied to the MSCKF [12].

The MSCKF state vector at time-step k augments the current IMU state by the past m poses where the images were taken (i.e., stochastic cloning [33]):

$$\mathbf{x}_{A_k} = \begin{bmatrix} \mathbf{x}_{I_k}^T & \mathbf{y}_{k-1}^T & \cdots & \mathbf{y}_{k-m}^T \end{bmatrix}^T \tag{91}$$

where $\mathbf{y}_\ell^T = \begin{bmatrix} {}^{I}_{G}\bar{\mathbf{q}}_\ell^T & {}^{G}\mathbf{p}_\ell^T \end{bmatrix}$ is the IMU pose (quaternion and position) where the image is recorded at time-step ℓ. Since the nullspace, $\mathbf{N}'$, is required for computing the STOC-VINS measurement Jacobian [see (86)], we accordingly augment the nullspace with the ones corresponding to the cloning states as follows:

$$\mathbf{N}'_A = \begin{bmatrix} \mathbf{N}' \\ \mathbf{N}'_{\text{clone},1} \\ \vdots \\ \mathbf{N}'_{\text{clone},m} \end{bmatrix} = \begin{bmatrix} & \mathbf{N}' \\ \mathbf{0}_3 & {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{p}}_{k-1|k-2} \times \rfloor {}^G\mathbf{g} \\ & \vdots \\ \mathbf{0}_3 & {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{p}}_{k-m|k-m-1} \times \rfloor {}^G\mathbf{g} \end{bmatrix} \tag{92}$$

During the MSCKF propagation, the current state estimates evolve forward in time by integrating (2), while the cloning-state estimates remain static. On the other hand, the augmented covariance is propagated as follows [see (13)]:

$$\mathbf{P}_{A_{k+1|k}} = \mathbf{Diag}\left(\mathbf{\Phi}'_k, \mathbf{I}_{6m}\right) \mathbf{P}_{A_{k|k}} \mathbf{Diag}\left(\mathbf{\Phi}'^{T}_k, \mathbf{I}_{6m}\right) + \mathbf{Diag}\left(\mathbf{Q}_{d,k}, \mathbf{0}_{6m}\right) \tag{93}$$

where the propagation Jacobian $\mathbf{\Phi}'_k$ is computed using the propagated state estimates as devised in Section 4.1.

During the MSCKF update, we stack together all the feature measurements within the sliding window, and linearize them with respected to the augmented IMU states as well as the feature position [see (17)]:

$$\begin{aligned} \begin{bmatrix} \widetilde{\mathbf{z}}_k \\ \vdots \\ \widetilde{\mathbf{z}}_{k-m} \end{bmatrix} &= \begin{bmatrix} \mathbf{H}'_k \\ \vdots \\ \mathbf{H}'_{k-m} \end{bmatrix} \begin{bmatrix} \widetilde{\mathbf{x}}_{A_{k|k-1}} \\ {}^G\widetilde{\mathbf{p}}_f \end{bmatrix} + \begin{bmatrix} \mathbf{v}_k \\ \vdots \\ \mathbf{v}_{k-m} \end{bmatrix} \\ &=: \mathbf{H}'_{\mathbf{x}} \widetilde{\mathbf{x}}_{A_{k|k-1}} + \mathbf{H}'_{\mathbf{f}} {}^G\widetilde{\mathbf{p}}_f + \mathbf{v} \end{aligned} \tag{94}$$

where the measurement Jacobian $\mathbf{H}'_k$ is computed as (86). Note that the feature position is not included in the MSCKF state vector (91), while we want to utilize the information contained in its measurements, we hence project (94) onto the left nullspace of $\mathbf{H}'_{\mathbf{f}}$ (i.e., $\mathbf{W}^T\mathbf{H}'_{\mathbf{f}} = \mathbf{0}$) and have:

$$\mathbf{W}^T\widetilde{\mathbf{z}} = \mathbf{W}^T\mathbf{H}'_{\mathbf{x}}\widetilde{\mathbf{x}}_{A_{k|k-1}} + \mathbf{W}^T\mathbf{v} \;\; \Leftrightarrow \;\; \widetilde{\mathbf{z}}'' = \mathbf{H}''_{\mathbf{x}}\widetilde{\mathbf{x}}_{A_{k|k-1}} + \mathbf{v}'' \tag{95}$$

The EKF uses the above residual equation to update the state estimates and covariance [32].

As mentioned before, we include the camera-IMU extrinsic calibration parameters, $\{{}^C_I\bar{\mathbf{q}}, {}^C\mathbf{p}_I\}$, in the state vector so as to perform this calibration *online*. As shown in [16], it is often unrealistic to assume the 6 d.o.f. camera-IMU transformation perfectly known, while using imperfect (known with finite precision) calibration as if it was underestimates the uncertainty and thus harms the filter consistency. Interestingly, the inclusion of the calibration parameters in the state vector incurs minimal modifications to the MSCKF [16]: Since this transformation is static, it is easy to propagate over time (in a similar way as for the static feature); Linearization of the stacked measurements renders new Jacobian terms with respect to these parameters [see (94) and (81)], which can be easily used in the standard EKF update equations.

5 Conclusions and Future Work

In this paper, we have introduced a new EKF-based VINS algorithm, termed STOC-VINS, which ensures appropriate state-transition and observability properties of the linearized system so as to improve consistency and accuracy. In particular, we use the global, instead of local, parametrization for the orientation error state, which enables the direct analytical computation of the propagation Jacobian that fulfils the semigroup property of an appropriate state-transition matrix. Moreover, by adopting the observability-constrained methodology, we project the best-available measurement Jacobian – computed using the latest, and thus best, state estimates – onto the observable subspace so that no spurious information is gained by the filter. As a result, the proposed STOC-VINS was shown to outperform the state-of-the-art VINS algorithms, in terms of consistency and accuracy. In the future, we will continue to investigate different ways to improve VINS performance (including accuracy, consistency, and efficiency), e.g., how to integrate with loop closure to enable long-term operation while attaining bounded errors.

References

[1] D. Titterton and J. Weston, *Strapdown Inertial Navigation Technology*, 2nd ed. The Institution of Engineering and Technology, 2005.

[2] J. Kim and S. Sukkarieh, "Real-time implementation of airborne inertial-SLAM," *Robotics and Autonomous Systems*, vol. 55, no. 1, pp. 62–71, Jan. 2007.

[3] M. Li and A. I. Mourikis, "Improving the accuracy of EKF-based visual-inertial odometry," in *Proc. of the IEEE International Conference on Robotics and Automation*, Minneapolis, MN, May 2012, pp. 828–835.

[4] M. Bryson and S. Sukkarieh, "Observability analysis and active control for airborne SLAM," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 261–280, Jan. 2008.

[5] A. Mourikis, N. Trawny, S. Roumeliotis, A. Johnson, A. Ansar, and L. Matthies, "Vision-aided inertial navigation for spacecraft entry, descent, and landing," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 264 –280, Apr. 2009.

[6] S. Ebcin and M. Veth, "Tightly-coupled image-aided inertial navigation using the unscented Kalman filter," Air Force Institute of Technology, Dayton, OH, Tech. Rep., 2007.

[7] D. Strelow, "Motion estimation from image and inertial measurements," Ph.D. dissertation, Carnegie Mellon University, 2004.

[8] V. Indelman, S. Williams, M. Kaess, and F. Dellaert, "Information fusion in navigation systems via factor graph based incremental smoothing," *Robotics and Autonomous Systems*, vol. 61, no. 8, pp. 721–738, Aug. 2013.

[9] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Analysis and improvement of the consistency of extended Kalman filter-based SLAM," in *Proc. of the IEEE International Conference on Robotics and Automation*, Pasadena, CA, May 19–23, 2008, pp. 473–479.

[10] ——, "Observability-based rules for designing consistent EKF SLAM estimators," *International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, Apr. 2010.

[11] G. Huang, "Improving the consistency of nonlinear estimators: Analysis, algorithms, and applications," Ph.D. dissertation, University of Minnesota, Dept. of Computer Science and Engineering, 2012.

[12] D. G. Kottas, J. A. Hesch, S. L. Bowman, and S. I. Roumeliotis, "On the consistency of vision-aided inertial navigation," in *Proc. of the 13th International Symposium on Experimental Robotics*, Quebec City, Canada, Jun. 17–20, 2012.

[13] J. Hesch, D. Kottas, S. Bowman, and S. Roumeliotis, "Towards consistent vision-aided inertial navigation," in *Algorithmic Foundations of Robotics X*, ser. Springer Tracts in Advanced Robotics, E. Frazzoli, T. Lozano-Perez, N. Roy, and D. Rus, Eds. Springer Berlin Heidelberg, 2013, vol. 86, pp. 559–574.

[14] ——, "Consistency analysis and improvement of vision-aided inertial navigation," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 158–176, Feb. 2013.

[15] ——, "Camera-IMU-based localization: Observability analysis and consistency improvement," *International Journal of Robotics Research*, vol. 33, pp. 182–201, 2014.

[16] M. Li and A. I. Mourikis, "High-precision, consistent EKF-based visual-inertial odometry," *International Journal of Robotics Research*, vol. 32, no. 6, pp. 690–711, Jun. 2013.

[17] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation*. New York: Wiley, 2001.

[18] W. L. Brogan, *Modern Control Theory*. Upper Saddle River, NJ: Prentice Hall, 1991.

[19] P. Corke, J. Lobo, and J. Dias, "An introduction to inertial and visual sensing," *International Journal of Robotics Research*, vol. 26, no. 6, pp. 519–535, Jun. 2007.

[20] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 61 –76, Feb. 2012.

[21] A. Martinelli, "Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 44–60, 2012.

[22] S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar, "Vision-based state estimation for autonomous rotorcraft mavs in complex environments," in *Proc. of the IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, May 6–10, 2013, pp. 1750–1756.

[23] S. Weiss and R. Siegwart, "Real-time metric state estimation for modular vision-inertial systems," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2011, pp. 4531–4537.

[24] D. Kottas and S. Roumeliotis, "Exploiting urban scenes for vision-aided inertial navigation," in *Proc. of the Robotics: Science and Systems Conference*, Berlin, Germany, Jun. 24–28, 2013.

[25] E. S. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *International Journal of Robotics Research*, vol. 30, no. 4, pp. 407–430, Apr. 2011.

[26] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, Jan. 2011.

[27] F. M. Mirzaei and S. I. Roumeliotis, "A Kalman filter-based algorithm for IMU-camera calibration: Observability analysis and performance evaluation," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1143–1156, Oct. 2008.

[28] A. Isidori, *Nonlinear Control Systems*. Springer, 1995.

[29] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, “A quadratic-complexity observability-constrained unscented Kalman filter for SLAM,” *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1226–1243, Oct. 2013.

[30] N. Trawny and S. I. Roumeliotis, “Indirect Kalman filter for 3D attitude estimation,” University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep., Mar. 2005.

[31] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, ser. Mathematics in Science and Engineering. London: Academic Press, 1979, vol. 141-1.

[32] A. I. Mourikis and S. I. Roumeliotis, “A multi-state constraint Kalman filter for vision-aided inertial navigation,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy, Apr. 10–14, 2007, pp. 3565–3572.

[33] S. I. Roumeliotis and J. W. Burdick, “Stochastic cloning: A generalized framework for processing relative state measurements,” in *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington, DC, May 11-15 2002, pp. 1788–1795.