Quaternion kinematics for the error-state KF

Joan Solà

October 7, 2014

Contents

1 Quaternions and rotation matrices

1.1 Definition of quaternion

The quaternion we use is given by the extension of the complex numbers. If we have two complex numbers $A = a + bi$ and $C = c + di$, then constructing $Q = A + Cj$ yields

$$Q = a + bi + cj + dk, \tag{1}$$

where $\{a, b, c, d\} \in \mathbb{R}$ and $\{i, j, k\}$ are defined so that

$$i^2 = j^2 = k^2 = -1, \quad ij = -ji = k. \tag{2}$$

It is noticeable that, while regular complex numbers of unit length $\mathbf{z} = e^{i\theta}$ can encode rotations in the 2D plane (with a product operation $\mathbf{v}' = \mathbf{z}\mathbf{v}$), "extended complex numbers" or quaternions of unit length $\mathbf{q}$ encode half-rotations in the 3D space (with two product operations $\mathbf{v}' = \mathbf{q} \otimes \mathbf{v} \otimes \mathbf{q}^*$, sometimes referred to as the "sandwich product", see (14) later in the document). See (5) just below for the meaning of "half rotations".

BEWARE: Not all quaternion definitions are the same. Some authors write the products as ib instead of bi, and therefore they get the property $k = ji = -ij$. Also, many authors place the real part at the end position, yielding $Q = ia + jb + kc + d$. This has no fundamental implications but makes the whole formulation different in the details.

BEWARE: There are additional conventions that also make the formulation different in details. They concern the "meaning" or "interpretation" we give to the rotation operators, either rotating vectors or rotating reference frames –which, essentially, constitute opposite operations. Refer to Section 1.4 for further explanations and disambiguation.

1.2 Vector representation of the quaternion

The $\{1, i, j, k\}$ notation is not very convenient. We better represent a quaternion Q as a 4-vector

$$Q = q_w + q_x i + q_y j + q_z k \qquad \Leftrightarrow \qquad \mathbf{q} = \begin{bmatrix} q_w \\ \mathbf{q}_\triangledown \end{bmatrix} = \begin{bmatrix} q_w \\ q_x \\ q_y \\ q_z \end{bmatrix}, \tag{3}$$

where q_w is referred to as the *real* or *scalar* part, and $\mathbf{q}_\triangledown$ as the *imaginary* or *vector* part. Our choice for the (q_w, q_x, q_y, q_z) notation is rather related to the fact that we are interested in the geometric properties of the quaternion. Other texts often use alternative notations like *e.g.* (q_0, q_1, q_2, q_3) (for more algebraic interpretations) or (q_1, q_i, q_j, q_k) (for more mathematical interpretations).

1.3 Cross-relations

1.3.1 Quaternion and rotation vector

Given the rotation vector $\mathbf{v} = \phi\mathbf{u}$, representing a rotation of ϕ rad along the axis given by the unit vector $\mathbf{u} = u_x i + u_y j + u_z k$, we have the unit quaternion

$$\mathbf{q} = e^{\mathbf{v}/2} \;, \tag{4}$$

which can be developed using an extension of the Euler formula,

$$\mathbf{q} = \cos\frac{\phi}{2} + (u_x i + u_y j + u_z k)\sin\frac{\phi}{2} = \begin{bmatrix} \cos(\phi/2) \\ \mathbf{u}\sin(\phi/2) \end{bmatrix} \;. \tag{5}$$

We call this the rotation vector to quaternion conversion, denoted by $\mathbf{q} = \mathrm{v2q}(\mathbf{v})$ or $\mathbf{q} = \mathbf{q}\{\mathbf{v}\}$. Conversely,

$$\phi = 2\arctan(\|\mathbf{q}_\triangledown\|, q_w) \tag{6}$$

$$\mathbf{u} = \mathbf{q}_\triangledown / \|\mathbf{q}_\triangledown\| \;. \tag{7}$$

1.3.2 Rotation matrix and rotation vector

The rotation matrix is also defined from the rotation vector $\mathbf{v} = \phi\mathbf{u}$,

$$\mathbf{R} = e^{[\mathbf{v}]_\times} \tag{8}$$

which we denote $\mathbf{R} = \mathbf{R}\{\mathbf{v}\}$. The operator $[\bullet]_\times$ performs

$$[\mathbf{v}]_\times \triangleq \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix} \;, \tag{9}$$

which is a skew-symmetric matrix left-hand-equivalent to the cross product, *i.e.*,

$$[\mathbf{v}]_\times \mathbf{w} = \mathbf{v}\times\mathbf{w}, \quad \forall \mathbf{v}, \mathbf{w} \in \mathbb{R}^3 \;, \tag{10}$$

with the properties

$$[\mathbf{u}]_\times^2 = \mathbf{u}\mathbf{u}^\top - \mathbf{I}, \quad \forall \mathbf{u} \in \mathbb{R}^3 \;\; s.t. \;\; \|\mathbf{u}\| = 1 \tag{11}$$

$$[\mathbf{u}]_\times^3 = -[\mathbf{u}]_\times, \quad \forall \mathbf{u} \in \mathbb{R}^3 \;\; s.t. \;\; \|\mathbf{u}\| = 1. \tag{12}$$

Writing the Taylor expansion of (8) with $\mathbf{v} = \phi\mathbf{u}$, and substituting (11–12) leads to a closed form to obtain the rotation matrix from the rotation vector, the so called Rodrigues formula

$$\mathbf{R} = \mathbf{I} + \sin\phi\,[\mathbf{u}]_\times + (1-\cos\phi)\,[\mathbf{u}]_\times^2 \;. \tag{13}$$

1.3.3 Rotation matrix and quaternion

A rotation applied to a vector $\mathbf{x}$ results in a new vector $\mathbf{x}'$ given by

$$\overline{\mathbf{x}}' = \mathbf{q} \otimes \overline{\mathbf{x}} \otimes \mathbf{q}^* \qquad\qquad \mathbf{x}' = \mathbf{R}\mathbf{x} \ , \tag{14}$$

with

$$\overline{\mathbf{x}} = \begin{bmatrix} 0 \\ \mathbf{x} \end{bmatrix} , \tag{15}$$

and where $\otimes$ represents the quaternion product (*i.e.*, the product of entities of type (1) using the quaternion algebra (2)), and $\mathbf{q}^*$ is the conjugated quaternion,

$$\mathbf{q}^* = \begin{bmatrix} q_w \\ -\mathbf{q}_\triangledown \end{bmatrix} . \tag{16}$$

Here, the bar notation $\overline{\mathbf{x}}$ indicates that the vector $\mathbf{x}$ is expressed in quaternion form, that is, $\overline{\mathbf{x}} = 0 + v_x i + v_y j + v_z k$. However, this circumstance is mostly unambiguous and can be derived from the context, and especially by the presence of the quaternion product $\otimes$. In what is to follow, we are omitting the bar and writing simply,

$$\mathbf{x}' = \mathbf{q} \otimes \mathbf{x} \otimes \mathbf{q}^* \ , \tag{17}$$

which allows us to abuse of the notation and write[1],

$$\mathbf{q} \otimes \mathbf{x} \otimes \mathbf{q}^* = \mathbf{R}\mathbf{x} \ . \tag{18}$$

As both quaternion and rotation matrix expressions (14) are linear in $\mathbf{x}$, an expression of the rotation matrix equivalent to the quaternion is found by developing the former and identifying terms in the latter, yielding

$$\mathbf{R} = \begin{bmatrix} q_w^2 + q_x^2 - q_y^2 - q_z^2 & 2(q_x q_y - q_w q_z) & 2(q_x q_z + q_w q_y) \\ 2(q_x q_y + q_w q_z) & q_w^2 - q_x^2 + q_y^2 - q_z^2 & 2(q_y q_z - q_w q_x) \\ 2(q_x q_z - q_w q_y) & 2(q_y q_z + q_w q_x) & q_w^2 - q_x^2 - q_y^2 + q_z^2 \end{bmatrix} , \tag{19}$$

with the following properties

$$\mathbf{R}\{[1,0,0,0]^\top\} = \mathbf{I} \tag{20}$$
$$\mathbf{R}\{-\mathbf{q}\} = \mathbf{R}\{\mathbf{q}\} \tag{21}$$
$$\mathbf{R}\{\mathbf{q}^*\} = \mathbf{R}\{\mathbf{q}\}^\top \tag{22}$$
$$\mathbf{R}\{\mathbf{q}_1 \otimes \mathbf{q}_2\} = \mathbf{R}\{\mathbf{q}_1\}\mathbf{R}\{\mathbf{q}_2\}. \tag{23}$$

Finally, the inverse of a composed rotation verifies

$$(\mathbf{q}_1 \otimes \mathbf{q}_2)^* = \mathbf{q}_2^* \otimes \mathbf{q}_1^* \qquad\qquad (\mathbf{R}_1\mathbf{R}_2)^\top = \mathbf{R}_2^\top \mathbf{R}_1^\top . \tag{24}$$

[1] The proper expression would be $\mathbf{q} \otimes \begin{bmatrix} 0 \\ \mathbf{x} \end{bmatrix} \otimes \mathbf{q}^* = \begin{bmatrix} 0 \\ \mathbf{R}\mathbf{x} \end{bmatrix}$

1.4 Conventions. My choices.

1.4.1 Meaning of the rotation specification

We have seen how to rotate vectors in 3D. Another way of seeing the effect of $\mathbf{R}$ and $\mathbf{q}$ over a vector $\mathbf{x}$ is to consider that the vector is steady but it is us who have rotated our point of view by an amount specified by $\mathbf{R}$ or $\mathbf{q}$. This is called *frame transformation* and it is governed by the inverse operation (*i.e.*, by using $\mathbf{R}^\top$ or $\mathbf{q}^*$).

The convention we use for interpreting quaternions and rotation matrices is this second one. Consider a global reference frame $\mathcal{G}$ and a local reference frame $\mathcal{L}$. We (re-)define $\mathbf{R}$ to represent the rotation matrix $\mathbf{R}_{\mathcal{G}\mathcal{L}}$ that specifies the orientation of the local frame $\mathcal{L}$ with respect to a global frame $\mathcal{G}$, so that,

$$\mathbf{x}_\mathcal{G} = \mathbf{R}_{\mathcal{G}\mathcal{L}}\mathbf{x}_\mathcal{L} \ . \tag{25}$$

In short, on one side the rotation matrix $\mathbf{R} \equiv \mathbf{R}_{\mathcal{G}\mathcal{L}}$ defines the frame $\mathcal{L}$ in $\mathcal{G}$, on the other side it transforms the vector $\mathbf{x}$ from frame $\mathcal{L}$ to frame $\mathcal{G}$. We see that, as $\mathcal{L}$ is expressed with respect to frame $\mathcal{G}$, we call $\mathcal{L}$ the local frame and $\mathcal{G}$ the global frame, *local* and *global* being always relative one another: $\mathcal{L}$ is local with respect to $\mathcal{G}$, and $\mathcal{G}$ is global with respect to $\mathcal{L}$. We call this convention the *from-frame* convention because $\mathbf{R}$ converts a vector from (a particularly specified) local frame to (an often assumed or implicit) global frame. The equivalent quaternion form is

$$\mathbf{x}_\mathcal{G} = \mathbf{q}_{\mathcal{G}\mathcal{L}} \otimes \mathbf{x}_\mathcal{L} \otimes \mathbf{q}^*_{\mathcal{G}\mathcal{L}}. \tag{26}$$

1.4.2 Quaternion algebra specification

There is quite a bit of ambiguity, confusion, and even controversy in the literature about quaternion conventions. Generally, one can find references to the JPL and the Hamiltonian conventions, representing these two paradigms.

The JPL convention [Trawny and Roumeliotis(2005)] defines $ji = k$ and hence the quaternion algebra becomes,

$$i^2 = j^2 = k^2 = -1 \ , \quad -ij = ji = k \ , \quad -jk = kj = i \ , \quad -ki = ik = j \ ; \tag{27}$$

whereas the Hamilton convention defines $ij = k$ and therefore,

$$i^2 = j^2 = k^2 = -1 \ , \quad ij = -ji = k \ , \quad jk = -kj = i \ , \quad ki = -ik = j \ . \tag{28}$$

The JPL convention is possibly less commonly used. My choice is to take the Hamiltonian convention, which coincides with many software libraries such as Eigen, ROS, Google Ceres, and with a vast amount of literature on Kalman filtering for attitude estimation using IMUs.

On the other hand, JPL is used in JPL literature (obviously) and in key papers by Mourikis, Riumeliotis, and colleagues (see *e.g.* [Trawny and Roumeliotis(2005)]), which are a primary source of inspiration when dealing with visual-inertial odometry and SLAM. You have been warned.

The most common case, however, is that the texts you are to encounter will give you no clue on which convention is used. I recommend therefore that you be very careful when picking up formulas, functions or concepts from differing sources, as the mathematical developments and algorithms that you build can end up being absolutely wrong because of inconsistency of conventions among the sources. You have been re-warned.

Finally, there are also differences in the order of the components, with the scalar part of the quaternion being in first or fourth place. The implications of such change are more obvious and should not represent a big challenge of interpretation.

All the definitions in the previous sections, and all the rest of this document, assume the Hamilton convention for quaternions, with the scalar part in the first position.

1.5 Composition

The composition is also done similarly to rotation matrices

$$\mathbf{q}_{\mathcal{GH}} = \mathbf{q}_{\mathcal{GL}} \otimes \mathbf{q}_{\mathcal{LH}} \qquad\qquad \mathbf{R}_{\mathcal{GH}} = \mathbf{R}_{\mathcal{GL}} \mathbf{R}_{\mathcal{LH}}. \tag{29}$$

The convention adopted here establishes that compositions go from global to local when advancing to the right of the expression of the composition, or from local to global when moving left. This means that both $\mathbf{q}_{\mathcal{LH}}$ and $\mathbf{R}_{\mathcal{LH}}$ are specifications of a frame $\mathcal{H}$ which is local with respect to frame $\mathcal{L}$.

The composition in quaternion form is bi-linear and can be expressed as two equivalent matrix products, namely

$$\boxed{\mathbf{q}_{\mathcal{GH}} = \mathbf{Q}^+_{\mathcal{GL}}\, \mathbf{q}_{\mathcal{LH}} \qquad \text{and} \qquad \mathbf{q}_{\mathcal{GH}} = \mathbf{Q}^-_{\mathcal{LH}}\, \mathbf{q}_{\mathcal{GL}}}\,, \tag{30}$$

with

$$\mathbf{Q}^+ = \begin{bmatrix} q_w & -q_x & -q_y & -q_z \\ q_x & q_w & -q_z & q_y \\ q_y & q_z & q_w & -q_x \\ q_z & -q_y & q_x & q_w \end{bmatrix}, \qquad \mathbf{Q}^- = \begin{bmatrix} q_w & -q_x & -q_y & -q_z \\ q_x & q_w & q_z & -q_y \\ q_y & -q_z & q_w & q_x \\ q_z & q_y & -q_x & q_w \end{bmatrix}, \tag{31}$$

or more concisely

$$\boxed{\mathbf{Q}^+ = q_w\, \mathbf{I} + \begin{bmatrix} 0 & -\mathbf{q}_\triangledown^\top \\ \mathbf{q}_\triangledown & [\mathbf{q}_\triangledown]_\times \end{bmatrix}, \qquad \mathbf{Q}^- = q_w\, \mathbf{I} + \begin{bmatrix} 0 & -\mathbf{q}_\triangledown^\top \\ \mathbf{q}_\triangledown & -[\mathbf{q}_\triangledown]_\times \end{bmatrix}}\,. \tag{32}$$

And since

$$\mathbf{q} \otimes \mathbf{r} \otimes \mathbf{p} = \mathbf{Q}^+\mathbf{R}^+\mathbf{p} = \mathbf{P}^-\mathbf{Q}^+\mathbf{r} \qquad \text{and} \qquad \mathbf{q} \otimes \mathbf{r} \otimes \mathbf{p} = \mathbf{Q}^+\mathbf{P}^-\mathbf{r} \tag{33}$$

we have the relation

$$\mathbf{P}^-\mathbf{Q}^+ = \mathbf{Q}^+\mathbf{P}^-. \tag{34}$$

1.6 Perturbations and time-derivatives

1.6.1 Local perturbations

A perturbed orientation $\tilde{\mathbf{q}}$ may be expressed as the composition of the unperturbed orientation $\mathbf{q}$ with a small perturbation $\Delta\mathbf{q}$ expressed in the local body frame. We give the quaternion and rotation matrix equivalents:

$$\boxed{\tilde{\mathbf{q}} = \mathbf{q} \otimes \Delta\mathbf{q}, \qquad\qquad \tilde{\mathbf{R}} = \mathbf{R}\,\Delta\mathbf{R}}\ . \tag{35}$$

Because of the convention used, the perturbations $\Delta\mathbf{q}$ and $\Delta\mathbf{R}$ are defined in the local frame specified by $\mathbf{q}$ or $\mathbf{R}$.

If the perturbation angle $\Delta\boldsymbol{\theta} \triangleq \Delta\phi\cdot\mathbf{u}$ is small then the perturbation quaternion and rotation matrix can be approximated by the Taylor expansions of (5) and (8) up to the linear terms,

$$\Delta\mathbf{q} \approx \begin{bmatrix} 1 \\ \frac{1}{2}\Delta\boldsymbol{\theta} \end{bmatrix} + O(\|\Delta\boldsymbol{\theta}\|^2), \qquad \Delta\mathbf{R} \approx \mathbf{I} + [\Delta\boldsymbol{\theta}]_\times + O(\|\Delta\boldsymbol{\theta}\|^2). \tag{36}$$

With this we can easily develop expressions for the time-derivatives. Just consider $\mathbf{q} = \mathbf{q}(t)$ as the original state, $\tilde{\mathbf{q}} = \mathbf{q}(t+\Delta t)$ as the perturbed state, and apply the definition of the derivative

$$\frac{df(t)}{dt} \triangleq \lim_{\Delta t \to 0} \frac{f(t+\Delta t) - f(t)}{\Delta t}, \tag{37}$$

to the above, with

$$\boldsymbol{\omega}_L(t) \triangleq \frac{d\boldsymbol{\theta}(t)}{dt}, \tag{38}$$

which, being $\Delta\boldsymbol{\theta}$ a local angular perturbation, corresponds to the angular rate vector in the body frame.

The development of the time-derivative of the quaternion follows (an analogous reasoning is used for the rotation matrix)

$$\begin{aligned}
\dot{\mathbf{q}} &\triangleq \lim_{\Delta t \to 0} \frac{\mathbf{q}(t+\Delta t) - \mathbf{q}(t)}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\mathbf{q} \otimes \Delta\mathbf{q} - \mathbf{q}}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\mathbf{Q}^-(\Delta\mathbf{q})\mathbf{q} - \mathbf{q}}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\left(\mathbf{I} + \frac{1}{2}\begin{bmatrix} 0 & -\Delta\boldsymbol{\theta}^\top \\ \Delta\boldsymbol{\theta} & -[\Delta\boldsymbol{\theta}]_\times \end{bmatrix} + O(\|\Delta\boldsymbol{\theta}\|^2)\right)\mathbf{q} - \mathbf{q}}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\frac{1}{2}\begin{bmatrix} 0 & -\Delta\boldsymbol{\theta}^\top \\ \Delta\boldsymbol{\theta} & -[\Delta\boldsymbol{\theta}]_\times \end{bmatrix}\mathbf{q} + O(\|\Delta\boldsymbol{\theta}\|^2)}{\Delta t} \\
&= \frac{1}{2}\begin{bmatrix} 0 & -\boldsymbol{\omega}_L^\top \\ \boldsymbol{\omega}_L & -[\boldsymbol{\omega}_L]_\times \end{bmatrix}\mathbf{q}\ .
\end{aligned} \tag{39}$$

Defining

$$\boxed{\Omega(\boldsymbol{\omega}_L) \triangleq \begin{bmatrix} 0 & -\boldsymbol{\omega}_L^\top \\ \boldsymbol{\omega}_L & -[\boldsymbol{\omega}_L]_\times \end{bmatrix} = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \equiv \mathbf{Q}^-(\boldsymbol{\omega}_L)}\,, \tag{40}$$

we get from (39) and (30) (we give also its matrix equivalent)

$$\boxed{\dot{\mathbf{q}} = \frac{1}{2}\Omega(\boldsymbol{\omega}_L)\mathbf{q} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega}_L, \qquad \dot{\mathbf{R}} = \mathbf{R}\,[\boldsymbol{\omega}_L]_\times}\,. \tag{41}$$

1.6.2 Global perturbations

It is possible and indeed interesting to consider globally-defined perturbations of the quaternion, and likewise for the related derivatives. Global perturbations are expressed as a product *on the left hand side* of the nominal quaternion,

$$\boxed{\tilde{\mathbf{q}} = \Delta\mathbf{q} \otimes \mathbf{q}\,, \qquad \tilde{\mathbf{R}} = \Delta\mathbf{R}{\cdot}\mathbf{R}}\,. \tag{42}$$

The associated time-derivatives are now,

$$\begin{aligned}
\dot{\mathbf{q}} &\triangleq \lim_{\Delta t \to 0} \frac{\mathbf{q}(t+\Delta t) - \mathbf{q}(t)}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\Delta\mathbf{q} \otimes \mathbf{q} - \mathbf{q}}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\mathbf{Q}^+(\Delta\mathbf{q})\mathbf{q} - \mathbf{q}}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\left(\mathbf{I} + \frac{1}{2}\begin{bmatrix} 0 & -\Delta\boldsymbol{\theta}^\top \\ \Delta\boldsymbol{\theta} & [\Delta\boldsymbol{\theta}]_\times \end{bmatrix} + O(\|\Delta\boldsymbol{\theta}\|^2)\right)\mathbf{q} - \mathbf{q}}{\Delta t} \\
&= \lim_{\Delta t \to 0} \frac{\frac{1}{2}\begin{bmatrix} 0 & -\Delta\boldsymbol{\theta}^\top \\ \Delta\boldsymbol{\theta} & [\Delta\boldsymbol{\theta}]_\times \end{bmatrix}\mathbf{q} + O(\|\Delta\boldsymbol{\theta}\|^2)}{\Delta t} \\
&= \frac{1}{2}\begin{bmatrix} 0 & -\boldsymbol{\omega}_G^\top \\ \boldsymbol{\omega}_G & [\boldsymbol{\omega}_G]_\times \end{bmatrix}\mathbf{q}\,,
\end{aligned} \tag{43}$$

which results in

$$\boxed{\dot{\mathbf{q}} = \frac{1}{2}\,\boldsymbol{\omega}_G \otimes \mathbf{q}\,, \qquad \dot{\mathbf{R}} = [\boldsymbol{\omega}_G]_\times\,\mathbf{R}} \tag{44}$$

where

$$\boldsymbol{\omega}_G(t) \triangleq \frac{d\boldsymbol{\theta}(t)}{dt} \tag{45}$$

is the angular rates vector expressed in the global frame.

1.6.3 Global-to-local perturbations

It is worth noticing the following relation between local and global angular rates,

$$\frac{1}{2}\boldsymbol{\omega}_G \otimes \mathbf{q} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega}_L \tag{46}$$

and therefore, post-multiplying by the conjugate quaternion we have

$$\boldsymbol{\omega}_G = \mathbf{q} \otimes \boldsymbol{\omega}_L \otimes \mathbf{q}^* = \mathbf{R}\,\boldsymbol{\omega}_L \ . \tag{47}$$

Likewise, considering that $\Delta\boldsymbol{\theta} \approx \boldsymbol{\omega}\Delta t$ for small Δt,

$$\Delta\boldsymbol{\theta}_G = \mathbf{q} \otimes \Delta\boldsymbol{\theta}_L \otimes \mathbf{q}^* = \mathbf{R}\,\Delta\boldsymbol{\theta}_L \ , \tag{48}$$

that is, we can transform angular rates vectors $\boldsymbol{\omega}$ and small angular perturbations $\Delta\boldsymbol{\theta}$ via the rotation matrix, as if they were regular vectors.

1.6.4 Other useful expressions with the derivative

We can derive an expression for the local rotation rate

$$\boldsymbol{\omega}_L = 2\mathbf{q}^* \otimes \dot{\mathbf{q}} \qquad\qquad [\boldsymbol{\omega}_L]_\times = \mathbf{R}^\top\dot{\mathbf{R}} \tag{49}$$

and the global rotation rate,

$$\boldsymbol{\omega}_G = 2\dot{\mathbf{q}} \otimes \mathbf{q}^* \qquad\qquad [\boldsymbol{\omega}_G]_\times = \dot{\mathbf{R}}\cdot\mathbf{R}^\top \tag{50}$$

We give also the derivative of the composition. It follows immediately from the fact that the products are bilinear

$$(\mathbf{q}_1 \dot{\otimes}\, \mathbf{q}_2) = \dot{\mathbf{q}}_1 \otimes \mathbf{q}_2 + \mathbf{q}_1 \otimes \dot{\mathbf{q}}_2 \qquad\qquad (\mathbf{R}_1\dot{\mathbf{R}}_2) = \dot{\mathbf{R}}_1\mathbf{R}_2 + \mathbf{R}_1\dot{\mathbf{R}}_2. \tag{51}$$

2 Error-state kinematics

2.1 Motivation

We wish to write the error-estate equations of the kinematics of an inertial system integrating accelerometer and gyrometer readings with bias and noise, using the quaternion to represent the orientation in space or *attitude*. These readings come typically from an Inertial Measurement Unit (IMU). Integrating IMU readings leads to dead-reckoning positioning systems, which drift with time. Avoiding drift is a matter of fusing this information with absolute position readings such as GPS or vision.

The error-state Kalman filter (ESKF) is one of the tools we may use for this purpose. Within the Kalman filtering paradigm, these are the most remarkable assets of the ESKF:

- The error-state is minimal, avoiding issues related to over-parametrization (or redundancy) and the consequent risk of singularity of the involved covariances matrices.
- The error-state system is always operating close to the origin, and therefore far from possible parameter singularities, providing a guarantee that the linearization validity holds at all times.
- The error-state is always small, meaning that all second-order products are negligible. This makes the computation of Jacobians very easy and fast. Some Jacobians may even be constant or equal to available state magnitudes.
- The error dynamics are slow because all the large-signal dynamics have been integrated in the nominal-state.

2.2 The error-state Kalman filter explained

In error-state filter formulations, we speak of true-, nominal- and error-state values, the true-state being expressed as a suitable composition (linear sum, quaternion product or matrix product) of the nominal- and the error-states. The idea is to consider the nominal-state as large-signal (integrable in non-linear fashion) and the error-state as small signal (thus linearly integrable and suitable for linear-Gaussian filtering).

The error-state filter can be explained as follows. On one side, high-frequency IMU data $\mathbf{u}_S$ is integrated into a nominal-state $\mathbf{x}$. This nominal state does not take into account the noise terms $\mathbf{w}$ and other possible model imperfections. As a consequence, it will accumulate errors. These errors are collected in the error-state $\delta\mathbf{x}$ and estimated with the Error-State Kalman Filter (ESKF), this time incorporating all the noise and perturbations. The error-state consists of small-signal magnitudes, and its evolution function is correctly defined by a (time-variant) linear dynamic system, with its dynamic, control and measurement matrices computed from the values of the nominal-state. In parallel with integration of the nominal-state, the ESKF predicts a Gaussian estimate of the error-state. It only predicts, because by now no other measurement is available to correct these estimates. The filter correction is performed at the arrival of information other than IMU (*e.g.* GPS, vision, etc.), which is able to render the errors observable and which happens at a much lower rate than the integration phase. This correction provides a posterior Gaussian estimate of the error-state. After this, the error-state's mean is injected into the nominal-state, then reset to zero, and its covariance conveniently updated to reflect this reset. The system goes on like this forever.

2.3 System kinematics in continuous time

The definition of all the involved variables is summarized in Table 1. Two important decisions regarding conventions are worth mentioning:

Table 1: All variables in the error-state Kalman filter.

Magnitude	True	Nominal	Error	Composition	Sensed	Noise
Full state (*)	$\mathbf{x}_t$	$\mathbf{x}$	$\delta\mathbf{x}$	$\mathbf{x}_t = \mathbf{x} \oplus \delta\mathbf{x}$		
Position	$\mathbf{p}_t$	$\mathbf{p}$	$\delta\mathbf{p}$	$\mathbf{p}_t = \mathbf{p} + \delta\mathbf{p}$		
Velocity	$\mathbf{v}_t$	$\mathbf{v}$	$\delta\mathbf{v}$	$\mathbf{v}_t = \mathbf{v} + \delta\mathbf{v}$		
Quaternion (**)	$\mathbf{q}_t$	$\mathbf{q}$	$\delta\mathbf{q}$	$\mathbf{q}_t = \mathbf{q} \otimes \delta\mathbf{q}$		
Rotation matrix (**)	$\mathbf{R}_t$	$\mathbf{R}$	$\delta\mathbf{R}$	$\mathbf{R}_t = \mathbf{R}\,\delta\mathbf{R}$		
Angles vector			$\delta\boldsymbol{\theta}$	$\delta\mathbf{q} \approx \begin{bmatrix} 1 \\ \delta\boldsymbol{\theta}/2 \end{bmatrix}$ $\delta\mathbf{R} \approx \mathbf{I} + [\delta\boldsymbol{\theta}]_\times$		
Accelerometer bias	$\mathbf{a}_{bt}$	$\mathbf{a}_b$	$\delta\mathbf{a}_b$	$\mathbf{a}_{bt} = \mathbf{a}_b + \delta\mathbf{a}_b$		$\mathbf{a}_w$
Gyrometer bias	$\boldsymbol{\omega}_{bt}$	$\boldsymbol{\omega}_b$	$\delta\boldsymbol{\omega}_b$	$\boldsymbol{\omega}_{bt} = \boldsymbol{\omega}_b + \delta\boldsymbol{\omega}_b$		$\boldsymbol{\omega}_w$
Gravity vector	$\mathbf{g}_t$	$\mathbf{g}$	$\delta\mathbf{g}$	$\mathbf{g}_t = \mathbf{g} + \delta\mathbf{g}$		
Acceleration	$\mathbf{a}_t$		$\mathbf{a}_S$	$\mathbf{a}_n$		
Angular rate	$\boldsymbol{\omega}_t$		$\boldsymbol{\omega}_S$	$\boldsymbol{\omega}_n$		

(*) the $\oplus$ symbol indicates a generic composition

(**) indicates non-minimal representations

- The angular rates are defined *locally* with respect to the nominal quaternion. This allows us to use the gyrometer readings directly, as they provide body-referenced angular rates.

- The angular error is also defined *locally* with respect to the nominal orientation. This is not necessarily the optimal way to proceed, but it corresponds to the choice in most IMU-integration works – what we could call the classical approach. There exists evidence that a globally-defined angular error has better properties. This will be explored too in the present document, but most of the developments, examples and algorithms here are based in this locally-defined angular error.

2.3.1 The true-state kinematics

The true kinematic equations are

$$\begin{aligned}
\dot{\mathbf{p}}_t &= \mathbf{v}_t && (52a)\\
\dot{\mathbf{v}}_t &= \mathbf{a}_t && (52b)\\
\dot{\mathbf{q}}_t &= \frac{1}{2}\mathbf{q}_t \otimes \boldsymbol{\omega}_t && (52c)\\
\dot{\mathbf{a}}_{bt} &= \mathbf{a}_w && (52d)\\
\dot{\boldsymbol{\omega}}_{bt} &= \boldsymbol{\omega}_w && (52e)\\
\dot{\mathbf{g}}_t &= 0 && (52f)
\end{aligned}$$

Here, the true acceleration $\mathbf{a}_t$ and angular rate $\boldsymbol{\omega}_t$ are obtained from an IMU in the form of noisy sensor readings $\mathbf{a}_S$ and $\boldsymbol{\omega}_S$ in body frame, namely

$$\begin{aligned}
\mathbf{a}_S &= \mathbf{R}_t^\top(\mathbf{a}_t - \mathbf{g}_t) + \mathbf{a}_{bt} + \mathbf{a}_n && (53)\\
\boldsymbol{\omega}_S &= \boldsymbol{\omega}_t + \boldsymbol{\omega}_{bt} + \boldsymbol{\omega}_n && (54)
\end{aligned}$$

with $\mathbf{R}_t \triangleq \mathbf{R}(\mathbf{q}_t)$. With that, the true values can be isolated (this means that we have inverted the measurement equations),

$$\begin{aligned}
\mathbf{a}_t &= \mathbf{R}_t(\mathbf{a}_S - \mathbf{a}_{bt} - \mathbf{a}_n) + \mathbf{g}_t && (55)\\
\boldsymbol{\omega}_t &= \boldsymbol{\omega}_S - \boldsymbol{\omega}_{bt} - \boldsymbol{\omega}_n. && (56)
\end{aligned}$$

Substituting above yields the kinematic system

$$\begin{aligned}
\dot{\mathbf{p}}_t &= \mathbf{v}_t && (57a)\\
\dot{\mathbf{v}}_t &= \mathbf{R}_t(\mathbf{a}_S - \mathbf{a}_{bt} - \mathbf{a}_n) + \mathbf{g}_t && (57b)\\
\dot{\mathbf{q}}_t &= \frac{1}{2}\mathbf{q}_t \otimes (\boldsymbol{\omega}_S - \boldsymbol{\omega}_{bt} - \boldsymbol{\omega}_n) && (57c)\\
\dot{\mathbf{a}}_{bt} &= \mathbf{a}_w && (57d)\\
\dot{\boldsymbol{\omega}}_{bt} &= \boldsymbol{\omega}_w && (57e)\\
\dot{\mathbf{g}}_t &= 0 && (57f)
\end{aligned}$$

which we may name $\dot{\mathbf{x}}_t = f_t(\mathbf{x}_t, \mathbf{u}, \mathbf{w})$. This system has state $\mathbf{x}_t$, is governed by IMU noisy readings $\mathbf{u}_S$, and is perturbed by white Gaussian noise $\mathbf{w}$, defined by

$$\mathbf{x}_t = \begin{bmatrix} \mathbf{p}_t \\ \mathbf{v}_t \\ \mathbf{q}_t \\ \mathbf{a}_{bt} \\ \boldsymbol{\omega}_{bt} \\ \mathbf{g}_t \end{bmatrix} \qquad \mathbf{u} = \begin{bmatrix} \mathbf{a}_S - \mathbf{a}_n \\ \boldsymbol{\omega}_S - \boldsymbol{\omega}_n \end{bmatrix} \qquad \mathbf{w} = \begin{bmatrix} \mathbf{a}_w \\ \boldsymbol{\omega}_w \end{bmatrix} . \tag{58}$$

It is to note in the above formulation that the gravity vector $\mathbf{g}_t$ is going to be estimated by the filter. It has a constant evolution equation, (57f), as corresponds to a magnitude that is known to be constant. The system starts at a fixed and arbitrarily known initial orientation $\mathbf{q}_t(t=0) = \mathbf{q}_0$, which, being generally not in the horizontal plane, makes the initial gravity vector generally unknown. For simplicity it is usually taken $\mathbf{q}_0 = (1, 0, 0, 0)$ and thus $\mathbf{R}_0 = \mathbf{R}\{\mathbf{q}_0\} = \mathbf{I}$. We estimate $\mathbf{g}_t$ expressed in frame $\mathbf{q}_0$, and not $\mathbf{q}_t$ expressed in a horizontal frame, so that the initial uncertainty in orientation is transferred to an initial uncertainty on the gravity direction. We do so to improve linearity: indeed, equation (57b) is now linear in $\mathbf{g}$, which carries all the uncertainty, and the initial orientation $\mathbf{q}_0$ is known without uncertainty, so that $\mathbf{q}_0$ starts with no uncertainty. Once the gravity vector is estimated the horizontal plane can be recovered and, if desired, the whole state and recovered motion trajectories can be re-oriented to reflect the estimated horizontal. See [Lupton and Sukkarieh(2009)] for further justification.

2.3.2 The nominal-state kinematics

The nominal-state kinematics corresponds to the modelled system without noises or perturbations,

$$\dot{\mathbf{p}} = \mathbf{v} \tag{59a}$$
$$\dot{\mathbf{v}} = \mathbf{R}(\mathbf{a}_S - \mathbf{a}_b) + \mathbf{g} \tag{59b}$$
$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes (\boldsymbol{\omega}_S - \boldsymbol{\omega}_b) \tag{59c}$$
$$\dot{\mathbf{a}}_b = 0 \tag{59d}$$
$$\dot{\boldsymbol{\omega}}_b = 0 \tag{59e}$$
$$\dot{\mathbf{g}} = 0. \tag{59f}$$

2.3.3 The error-state kinematics

The goal is to determine the linearized dynamics of the error-state. For each state equation, we write its composition (in Table 1), solving for the error state and simplifying all second-order infinitesimals. We give here the full error-state dynamic system and proceed afterwards with comments and proofs.

$$\dot{\delta\mathbf{p}} = \delta\mathbf{v} \tag{60a}$$
$$\dot{\delta\mathbf{v}} = -\mathbf{R}\left[\mathbf{a}_S - \mathbf{a}_b\right]_\times \delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g} - \mathbf{R}\mathbf{a}_n \tag{60b}$$
$$\dot{\delta\boldsymbol{\theta}} = -\left[\boldsymbol{\omega}_S - \boldsymbol{\omega}_b\right]_\times \delta\boldsymbol{\theta} - \delta\boldsymbol{\omega}_b - \boldsymbol{\omega}_n \tag{60c}$$
$$\dot{\delta\mathbf{a}}_b = \mathbf{a}_w \tag{60d}$$
$$\dot{\delta\boldsymbol{\omega}}_b = \boldsymbol{\omega}_w \tag{60e}$$
$$\dot{\delta\mathbf{g}} = 0. \tag{60f}$$

Equations (60a), (60d), (60e) and (60f) are derived from linear equations and their error-state dynamics is trivial. As an example, consider the true and nominal position equations

(57a) and (59a), their composition $\mathbf{p}_t = \mathbf{p} + \delta\mathbf{p}$ from Table 1, and solve for $\dot{\delta\mathbf{p}}$ to obtain (60a).

Equations (60b) and (60c) require some non-trivial manipulations of the non-linear equations (57b) and (57c) to obtain the linearized dynamics of the linear velocity error $\delta\mathbf{v}$ and the angular error $\delta\boldsymbol{\theta}$. Their proofs are developed in the following two sections.

Equation (60b): The linear velocity. We wish to determine $\dot{\delta\mathbf{v}}$, the dynamics of the velocity errors. We start with the following relations

$$\mathbf{R}_t = \mathbf{R}(\mathbf{I} + [\delta\boldsymbol{\theta}]_\times) + O(\|\delta\boldsymbol{\theta}\|^2) \tag{61}$$

$$\dot{\mathbf{v}} = \mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{g}, \tag{62}$$

where (61) is the small-signal approximation of $\mathbf{R}_t$, and in (62) we rewrote (59b) but introducing $\mathbf{a}_\mathcal{B}$ and $\delta\mathbf{a}_\mathcal{B}$, defined as the large- and small-signal accelerations in body frame,

$$\mathbf{a}_\mathcal{B} \triangleq \mathbf{a}_S - \mathbf{a}_b \tag{63}$$

$$\delta\mathbf{a}_\mathcal{B} \triangleq -\delta\mathbf{a}_b - \mathbf{a}_n \tag{64}$$

so that we can write the true acceleration in inertial frame as a composition of large- and small-signal terms,

$$\mathbf{a}_t = \mathbf{R}_t(\mathbf{a}_\mathcal{B} + \delta\mathbf{a}_\mathcal{B}) + \mathbf{g} + \delta\mathbf{g}. \tag{65}$$

We proceed by writing the expression (57b) of $\dot{\mathbf{v}}_t$ in two different forms (left and right developments), where the terms $O(\|\delta\boldsymbol{\theta}\|^2)$ have been ignored,

$$\begin{aligned}
\dot{\mathbf{v}} + \dot{\delta\mathbf{v}} = \boxed{\;\dot{\mathbf{v}}_t\;} &= \mathbf{R}(\mathbf{I} + [\delta\boldsymbol{\theta}]_\times)(\mathbf{a}_\mathcal{B} + \delta\mathbf{a}_\mathcal{B}) + \mathbf{g} + \delta\mathbf{g} \\
\mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{g} + \dot{\delta\mathbf{v}} = \quad\quad &= \mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{R}\delta\mathbf{a}_\mathcal{B} + \mathbf{R}\,[\delta\boldsymbol{\theta}]_\times\,\mathbf{a}_\mathcal{B} + \mathbf{R}\,[\delta\boldsymbol{\theta}]_\times\,\delta\mathbf{a}_\mathcal{B} + \mathbf{g} + \delta\mathbf{g}
\end{aligned}$$

This leads after removing $\mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{g}$ from left and right to

$$\dot{\delta\mathbf{v}} = \mathbf{R}(\delta\mathbf{a}_\mathcal{B} + [\delta\boldsymbol{\theta}]_\times\,\mathbf{a}_\mathcal{B}) + \mathbf{R}\,[\delta\boldsymbol{\theta}]_\times\,\delta\mathbf{a}_\mathcal{B} + \delta\mathbf{g} \tag{66}$$

Eliminating the second order terms and reorganizing some cross-products (with $[\mathbf{a}]_\times\,\mathbf{b} = -\,[\mathbf{b}]_\times\,\mathbf{a}$), we get

$$\dot{\delta\mathbf{v}} = \mathbf{R}(\delta\mathbf{a}_\mathcal{B} - [\mathbf{a}_\mathcal{B}]_\times\,\delta\boldsymbol{\theta}) + \delta\mathbf{g}, \tag{67}$$

then, recalling (63) and (64),

$$\dot{\delta\mathbf{v}} = \mathbf{R}(-\,[\mathbf{a}_S - \mathbf{a}_b]_\times\,\delta\boldsymbol{\theta} - \delta\mathbf{a}_b - \mathbf{a}_n) + \delta\mathbf{g} \tag{68}$$

which after proper rearranging leads to the dynamics of the linear velocity error,

$$\boxed{\dot{\delta\mathbf{v}} = -\mathbf{R}\,[\mathbf{a}_S - \mathbf{a}_b]_\times\,\delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g} - \mathbf{R}\mathbf{a}_n}\;. \tag{69}$$

To further clean up this expression, we can often times assume that the accelerometer noise is white, uncorrelated and isotropic[2],

$$\mathbb{E}[\mathbf{a}_n] = 0 \qquad \mathbb{E}[\mathbf{a}_n\mathbf{a}_n^\top] = \sigma_a^2\mathbf{I}, \tag{70}$$

that is, the covariance ellipsoid is a sphere centered at the origin, which means that its mean and covariances matrix are invariant upon rotations (*Proof:* $\mathbb{E}[\mathbf{R}\mathbf{a}_n] = \mathbf{R}\mathbb{E}[\mathbf{a}_n] = 0$ and $\mathbf{E}[(\mathbf{R}\mathbf{a}_n)(\mathbf{R}\mathbf{a}_n)^\top] = \mathbf{R}\mathbb{E}[\mathbf{a}_n\mathbf{a}_n^\top]\mathbf{R}^\top = \mathbf{R}\sigma_a^2\mathbf{I}\mathbf{R}^\top = \sigma_a^2\mathbf{I}$). Then we can redefine the accelerometer noise vector, with absolutely no consequences, according to

$$\mathbf{a}_n \leftarrow \mathbf{R}\mathbf{a}_n \tag{71}$$

which gives

$$\boxed{\dot{\delta\mathbf{v}} = -\mathbf{R}\left[\mathbf{a}_S - \mathbf{a}_b\right]_\times \delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g} - \mathbf{a}_n} \ . \tag{72}$$

Equation (60c): The orientation error. We wish to determine $\dot{\delta\boldsymbol{\theta}}$, the dynamics of the angular errors. We start with the following relations

$$\dot{\mathbf{q}}_t = \frac{1}{2}\mathbf{q}_t \otimes \boldsymbol{\omega}_t \tag{73}$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega}, \tag{74}$$

which are the true- and nominal- definitions of the quaternion derivatives.

As we did with the acceleration, we group large- and small-signal terms in the angular rate for clarity,

$$\boldsymbol{\omega} \triangleq \boldsymbol{\omega}_S - \boldsymbol{\omega}_b \tag{75}$$

$$\delta\boldsymbol{\omega} \triangleq -\delta\boldsymbol{\omega}_b - \boldsymbol{\omega}_n, \tag{76}$$

so that $\boldsymbol{\omega}_t$ can be written with a nominal part and an error part,

$$\boldsymbol{\omega}_t = \boldsymbol{\omega} + \delta\boldsymbol{\omega}. \tag{77}$$

We proceed by computing $\dot{\mathbf{q}}_t$ by two different means (left and right developments)

$$\begin{aligned}
\frac{1}{2}\mathbf{q}_t \otimes \boldsymbol{\omega}_t = \boxed{\dot{\mathbf{q}}_t} &= (\mathbf{q} \dot{\otimes} \delta\mathbf{q}) \\
\frac{1}{2}\mathbf{q} \otimes \delta\mathbf{q} \otimes \boldsymbol{\omega}_t = \quad &= \dot{\mathbf{q}} \otimes \delta\mathbf{q} + \mathbf{q} \otimes \dot{\delta\mathbf{q}} \\
&= \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega} \otimes \delta\mathbf{q} + \mathbf{q} \otimes \dot{\delta\mathbf{q}}
\end{aligned}$$

[2] This assumption cannot be made in cases where the three XYZ accelerometers are not identical.

simplifying the leading $\mathbf{q}$ and isolating $\dot{\delta\mathbf{q}}$ we obtain

$$\begin{aligned}
\begin{bmatrix} 0 \\ \dot{\delta\boldsymbol{\theta}} \end{bmatrix} = \boxed{2\dot{\delta\mathbf{q}}} &= \delta\mathbf{q} \otimes \boldsymbol{\omega}_t - \boldsymbol{\omega} \otimes \delta\mathbf{q} \\
&= \Omega(\boldsymbol{\omega}_t)\delta\mathbf{q} - \mathbf{Q}^+(\boldsymbol{\omega})\delta\mathbf{q} \\
&= \begin{bmatrix} 0 & -(\boldsymbol{\omega}_t - \boldsymbol{\omega})^\top \\ (\boldsymbol{\omega}_t - \boldsymbol{\omega}) & -\left[\boldsymbol{\omega}_t + \boldsymbol{\omega}\right]_\times \end{bmatrix} \begin{bmatrix} 1 \\ \delta\boldsymbol{\theta}/2 \end{bmatrix} + O(\|\delta\boldsymbol{\theta}\|^2) \\
&= \begin{bmatrix} 0 & -\delta\boldsymbol{\omega}^\top \\ \delta\boldsymbol{\omega} & -\left[2\boldsymbol{\omega} + \delta\boldsymbol{\omega}\right]_\times \end{bmatrix} \begin{bmatrix} 1 \\ \delta\boldsymbol{\theta}/2 \end{bmatrix} + O(\|\delta\boldsymbol{\theta}\|^2)
\end{aligned} \tag{78}$$

which results in one scalar- and one vector- equalities

$$0 = \delta\boldsymbol{\omega}^\top \delta\boldsymbol{\theta} + O(|\delta\boldsymbol{\theta}|^2) \tag{79}$$

$$\dot{\delta\boldsymbol{\theta}} = \delta\boldsymbol{\omega} - \left[\boldsymbol{\omega}\right]_\times \delta\boldsymbol{\theta} - \frac{1}{2}\left[\delta\boldsymbol{\omega}\right]_\times \delta\boldsymbol{\theta} + O(\|\delta\boldsymbol{\theta}\|^2). \tag{80}$$

The first equation leads to $\delta\boldsymbol{\omega}^\top\delta\boldsymbol{\theta} = O(\|\delta\boldsymbol{\theta}\|^2)$, which is formed by second-order infinitesimals, not very useful. The second equation yields, after neglecting all second-order terms,

$$\dot{\delta\boldsymbol{\theta}} = -\left[\boldsymbol{\omega}\right]_\times \delta\boldsymbol{\theta} + \delta\boldsymbol{\omega} \tag{81}$$

and finally, recalling (75) and (76), we get the linearized dynamics of the angular error,

$$\boxed{\dot{\delta\boldsymbol{\theta}} = -\left[\boldsymbol{\omega}_S - \boldsymbol{\omega}_b\right]_\times \delta\boldsymbol{\theta} - \delta\boldsymbol{\omega}_b - \boldsymbol{\omega}_n} \,. \tag{82}$$

2.4 The differences equations

The differential equations above need to be integrated into differences equations to account for discrete time intervals $\Delta t > 0$. The integration methods may vary. In some cases, one will be able to use exact closed-form solutions. In other cases, numerical integration of varying degree of accuracy may be employed. Please refer to the Appendices for pertinent details on integration methods.

Integration needs to be done for the following sub-systems:

1. The nominal state.
2. The error-state.
    (a) The deterministic part: state dynamics and control.
    (b) The stochastic part: noise and perturbations.

2.4.1 The nominal state

We can write the differences equations of the nominal-state as

$$\mathbf{p} \leftarrow \mathbf{p} + \mathbf{v}\,\Delta t + \frac{1}{2}(\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b) + \mathbf{g})\,\Delta t^2 \tag{83a}$$
$$\mathbf{v} \leftarrow \mathbf{v} + (\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b) + \mathbf{g})\,\Delta t \tag{83b}$$
$$\mathbf{q} \leftarrow \mathbf{q} \otimes \mathbf{q}\{(\boldsymbol{\omega}_S - \boldsymbol{\omega}_b)\,\Delta t\} \tag{83c}$$
$$\mathbf{a}_b \leftarrow \mathbf{a}_b \tag{83d}$$
$$\boldsymbol{\omega}_b \leftarrow \boldsymbol{\omega}_b \tag{83e}$$
$$\mathbf{g} \leftarrow \mathbf{g}\ , \tag{83f}$$

where $x \leftarrow f(x, \bullet)$ stands for a time update of the type $x_{k+1} = f(x_k, \bullet_k)$, $\mathbf{R} \triangleq \mathbf{R}\{\mathbf{q}\}$ is the rotation matrix associated to the current nominal orientation $\mathbf{q}$, and $\mathbf{q}\{v\}$ is the quaternion associated to the rotation v, according to (5).

We can also use more precise integration, please see the Appendices for more information.

2.4.2 The error-state

The deterministic part is integrated normally (in this case we follow the methods in App. C.2), and the integration of the stochastic part results in random impulses (see App. E), thus,

$$\delta\mathbf{p} \leftarrow \delta\mathbf{p} + \delta\mathbf{v}\,\Delta t \tag{84a}$$
$$\delta\mathbf{v} \leftarrow \delta\mathbf{v} + (-\mathbf{R}\,[\mathbf{a}_S - \mathbf{a}_b]_\times\,\delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g})\Delta t + \mathbf{v_i} \tag{84b}$$
$$\delta\boldsymbol{\theta} \leftarrow \mathbf{R}^\top\{(\boldsymbol{\omega}_S - \boldsymbol{\omega}_b)\Delta t\}\delta\boldsymbol{\theta} - \delta\boldsymbol{\omega}_b\Delta t + \boldsymbol{\theta}_\mathbf{i} \tag{84c}$$
$$\delta\mathbf{a}_b \leftarrow \delta\mathbf{a}_b + \mathbf{a_i} \tag{84d}$$
$$\delta\boldsymbol{\omega}_b \leftarrow \delta\boldsymbol{\omega}_b + \boldsymbol{\omega}_\mathbf{i} \tag{84e}$$
$$\delta\mathbf{g} \leftarrow \delta\mathbf{g}\ . \tag{84f}$$

Here, $\mathbf{v_i}$, $\boldsymbol{\theta}_\mathbf{i}$, $\mathbf{a_i}$ and $\boldsymbol{\omega}_\mathbf{i}$ are the random impulses applied to the velocity, orientation and bias estimates, modeled by white Gaussian processes. Their mean is zero, and their covariances matrices are obtained by integrating the covariances of $\mathbf{a}_n$, $\boldsymbol{\omega}_n$, $\mathbf{a}_w$ and $\boldsymbol{\omega}_w$ over the step time Δt (see App. E),

$$\mathbf{V_i} = \sigma^2_{\tilde{\mathbf{a}}_n}\Delta t^2\mathbf{I} \qquad [m^2/s^2] \tag{85}$$
$$\Theta_\mathbf{i} = \sigma^2_{\tilde{\boldsymbol{\omega}}_n}\Delta t^2\mathbf{I} \qquad [rad^2] \tag{86}$$
$$\mathbf{A_i} = \sigma^2_{\mathbf{a}_w}\Delta t\mathbf{I} \qquad [m^2/s^4] \tag{87}$$
$$\Omega_\mathbf{i} = \sigma^2_{\boldsymbol{\omega}_w}\Delta t\mathbf{I} \qquad [rad^2/s^2] \tag{88}$$

where $\sigma_{\tilde{\mathbf{a}}_n}[m/s^2]$, $\sigma_{\tilde{\boldsymbol{\omega}}_n}[rad/s]$, $\sigma_{\mathbf{a}_w}[m/s^2\sqrt{s}]$ and $\sigma_{\boldsymbol{\omega}_w}[rad/s\sqrt{s}]$ are to be determined from the information in the IMU datasheet, or from experimental measurements.

2.4.3 The error-state Jacobian and perturbation matrices

The Jacobians are obtained by simple inspection of the error-state differences equations in the previous section.

To write these equations in compact form, we consider the nominal state vector $\mathbf{x}$, the error state vector $\delta\mathbf{x}$, the input vector $\mathbf{u}_S$, and the perturbation impulses vector $\mathbf{i}$, as follows (see App. E.1 for details and justifications),

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \\ \mathbf{a}_b \\ \boldsymbol{\omega}_b \\ \mathbf{g} \end{bmatrix} \quad , \quad \delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} \\ \delta\mathbf{v} \\ \delta\boldsymbol{\theta} \\ \delta\mathbf{a}_b \\ \delta\boldsymbol{\omega}_b \\ \delta\mathbf{g} \end{bmatrix} \quad , \quad \mathbf{u}_S = \begin{bmatrix} \mathbf{a}_S \\ \boldsymbol{\omega}_S \end{bmatrix} \quad , \quad \mathbf{i} = \begin{bmatrix} \mathbf{v_i} \\ \boldsymbol{\theta}_\mathbf{i} \\ \mathbf{a_i} \\ \boldsymbol{\omega}_\mathbf{i} \end{bmatrix} \tag{89}$$

The error-state system is now

$$\delta\mathbf{x} \leftarrow f(\mathbf{x}, \delta\mathbf{x}, \mathbf{u}_S, \mathbf{i}) = \mathbf{F_x}(\mathbf{x}, \mathbf{u}_S)\cdot\delta\mathbf{x} + \mathbf{F_i}\cdot\mathbf{i}, \tag{90}$$

The ESKF prediction equations are written:

$$\hat{\delta\mathbf{x}} \leftarrow \mathbf{F_x}(\mathbf{x}, \mathbf{u}_S)\cdot\hat{\delta\mathbf{x}} \tag{91}$$

$$\mathbf{P} \leftarrow \mathbf{F_x}\mathbf{P}\mathbf{F_x}^\top + \mathbf{F_i}\mathbf{Q_i}\mathbf{F_i}^\top \;, \tag{92}$$

where $\delta\mathbf{x} \sim \mathcal{N}\{\hat{\delta\mathbf{x}}, \mathbf{P}\}$[3]; $\mathbf{F_x}$ and $\mathbf{F_i}$ are the Jacobians of $f()$ with respect to the error and perturbation vectors; and $\mathbf{Q_i}$ is the covariances matrix of the perturbation impulses.

The expressions of the Jacobian and covariances matrices above are detailed below. All state-related values appearing herein are extracted directly from the nominal state.

$$\mathbf{F_x} = \left.\frac{\partial f}{\partial \delta\mathbf{x}}\right|_{\mathbf{x},\mathbf{u}_S} = \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & 0 & 0 & 0 & 0 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}_S - \mathbf{a}_b\right]_\times \Delta t & -\mathbf{R}\Delta t & 0 & \mathbf{I}\Delta t \\ 0 & 0 & \mathbf{R}^\top\{(\boldsymbol{\omega}_S - \boldsymbol{\omega}_b)\Delta t\} & 0 & -\mathbf{I}\Delta t & 0 \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} \end{bmatrix} \tag{93}$$

$$\mathbf{F_i} = \left.\frac{\partial f}{\partial \mathbf{i}}\right|_{\mathbf{x},\mathbf{u}_S} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I} \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad , \quad \mathbf{Q_i} = \begin{bmatrix} \mathbf{V_i} & 0 & 0 & 0 \\ 0 & \Theta_\mathbf{i} & 0 & 0 \\ 0 & 0 & \mathbf{A_i} & 0 \\ 0 & 0 & 0 & \Omega_\mathbf{i} \end{bmatrix} \;. \tag{94}$$

[3] $x \sim \mathcal{N}\{\mu, \Sigma\}$ means that x is a Gaussian random variable with mean and covariances matrix specified by μ and Σ.

Please note particularly that $\mathbf{F_x}$ is the system's transition matrix, which can be approximated to different levels of precision in a number of ways. We showed here one of its simplest forms (the Euler form). Se Appendices B to D for further reference.

Please note also that, being the mean of the error $\delta\mathbf{x}$ initialized to zero, the linear equation (91) always returns zero. You should of course skip line (91) in your code. I recommend that you write it, though, but that you comment it out so that you are sure you did not forget anything.

And please note, finally, that you should NOT skip the covariance prediction (92)!! In effect, the term $\mathbf{F_i}\mathbf{Q_i}\mathbf{F_i}^\top$ is not null and therefore this covariance grows continuously – as it must be in any prediction step.

3 Fusing IMU with complementary sensory data

At the arrival of other kind of information than IMU, such as GPS or vision, we proceed to correct the ESKF. In a well-designed system, this should render the IMU biases observable and allow the ESKF to correctly estimate them. There are a myriad of possibilities, the most popular ones being GPS + IMU, monocular vision + IMU, and stereo vision + IMU. In recent years, the combination of visual sensors with IMU has attracted a lot of attention, and thus generated a lot of scientific activity. These vision + IMU setups are very interesting for use in GPS-denied environments, and can be implemented on mobile devices (typically smart phones), but also on UAVs and other small, agile platforms.

While the IMU information has served so far to make predictions to the ESKF, this other information is used to correct the filter. The correction consists of three steps:

1. observation of the error-state via filter correction,
2. injection of the observed errors into the nominal state, and
3. reset of the error-state.

These steps are developed in the following sections.

3.1 Observation of the error state via filter correction

Suppose as usual that we have a sensor that delivers information that depends on the state, such as

$$\mathbf{y} = h(\mathbf{x}_t) + v \ , \tag{95}$$

where $h()$ is a general nonlinear function of the system state (the true state), and v is a white Gaussian noise with covariance $\mathbf{V}$,

$$v \sim \mathcal{N}\{0, \mathbf{V}\} \ . \tag{96}$$

Our filter is estimating the error state, and therefore the filter correction equations[4],

$$\mathbf{K} = \mathbf{P}\mathbf{H}^\top(\mathbf{H}\mathbf{P}\mathbf{H}^\top + \mathbf{V})^{-1} \tag{97}$$
$$\hat{\delta\mathbf{x}} \leftarrow \mathbf{K}(\mathbf{y} - h(\hat{\mathbf{x}}_t)) \tag{98}$$
$$\mathbf{P} \leftarrow (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P} \tag{99}$$

require the Jacobian matrix $\mathbf{H}$ to be defined with respect to the error state $\delta\mathbf{x}$, and evaluated at the best true-state estimate $\hat{\mathbf{x}}_t = \mathbf{x} \oplus \hat{\delta\mathbf{x}}$. As the error state mean is zero at this stage, we have $\hat{\mathbf{x}}_t = \mathbf{x}$ and we can use the nominal error $\mathbf{x}$ as the evaluation point, leading to

$$\mathbf{H} \equiv \left.\frac{\partial h}{\partial \delta\mathbf{x}}\right|_{\mathbf{x}} . \tag{100}$$

3.1.1 Jacobian computation for the filter correction

The Jacobian above might be computed in a number of ways. The most illustrative one is by making use of the chain rule,

$$\mathbf{H} \triangleq \left.\frac{\partial h}{\partial \delta\mathbf{x}}\right|_{\mathbf{x}} = \left.\frac{\partial h}{\partial \mathbf{x}_t}\right|_{\mathbf{x}} \left.\frac{\partial \mathbf{x}_t}{\partial \delta\mathbf{x}}\right|_{\mathbf{x}} = \mathbf{H}_{\mathbf{x}}\ \mathbf{X}_{\delta\mathbf{x}} . \tag{101}$$

Here, $\mathbf{H}_{\mathbf{x}} \triangleq \left.\frac{\partial h}{\partial \mathbf{x}_t}\right|_{\mathbf{x}}$ is the standard Jacobian of $h()$ with respect to its own argument (*i.e.*, the Jacobian one would use in a regular EKF). This first part of the chain rule depends on the sensor function, and is not presented here.

The second part, $\mathbf{X}_{\delta\mathbf{x}} \triangleq \left.\frac{\partial \mathbf{x}_t}{\partial \delta\mathbf{x}}\right|_{\mathbf{x}}$, is the Jacobian of the true state with respect to the error state. This part can be derived here as it only depends on the ESKF composition of states. We have the derivatives,

$$\mathbf{X}_{\delta\mathbf{x}} = \begin{bmatrix} \frac{\partial(\mathbf{p}+\delta\mathbf{p})}{\partial\delta\mathbf{p}} & & & & & 0 \\ & \frac{\partial(\mathbf{v}+\delta\mathbf{v})}{\partial\delta\mathbf{v}} & & & & \\ & & \frac{\partial(\mathbf{q}\otimes\delta\mathbf{q})}{\partial\delta\boldsymbol{\theta}} & & & \\ & & & \frac{\partial(\mathbf{a}_b+\delta\mathbf{a}_b)}{\partial\delta\mathbf{a}_b} & & \\ & 0 & & & \frac{\partial(\boldsymbol{\omega}_b+\delta\boldsymbol{\omega}_b)}{\partial\delta\boldsymbol{\omega}_b} & \\ & & & & & \frac{\partial(\mathbf{g}+\delta\mathbf{g})}{\partial\delta\mathbf{g}} \end{bmatrix} \tag{102}$$

which results in all identity 3×3 blocks (for example, $\frac{\partial(\mathbf{p}+\delta\mathbf{p})}{\partial\delta\mathbf{p}} = \mathbf{I}_3$) except for the 4×3 quaternion term $\mathbf{Q}_{\delta\boldsymbol{\theta}} = \frac{\partial(\mathbf{q}\otimes\delta\mathbf{q})}{\partial\delta\boldsymbol{\theta}}$. Therefore we have the form,

$$\mathbf{X}_{\delta\mathbf{x}} \triangleq \left.\frac{\partial \mathbf{x}_t}{\partial \delta\mathbf{x}}\right|_{\mathbf{x}} = \begin{bmatrix} \mathbf{I}_6 & 0 & 0 \\ 0 & \mathbf{Q}_{\delta\boldsymbol{\theta}} & 0 \\ 0 & 0 & \mathbf{I}_9 \end{bmatrix} \tag{103}$$

[4]We give the simplest form of the covariance update, $\mathbf{P} \leftarrow (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}$. This form is known to have poor numerical stability, as its outcome is not guaranteed to be symmetric nor positive definite. The reader is free to use more stable forms such as 1) the symmetric form $\mathbf{P} \leftarrow \mathbf{P} - \mathbf{K}(\mathbf{H}\mathbf{P}\mathbf{H}^\top + \mathbf{V})\mathbf{K}^\top$ and 2) the symmetric and positive *Joseph* form $\mathbf{P} \leftarrow (\mathbf{I} - \mathbf{K}\mathbf{H})\mathbf{P}(\mathbf{I} - \mathbf{K}\mathbf{H})^\top + \mathbf{K}\mathbf{V}\mathbf{K}^\top$.

Using (30–31) and the first-order approximation $\delta\mathbf{q} \to \begin{bmatrix} 1 \\ \frac{1}{2}\delta\boldsymbol{\theta} \end{bmatrix}$, the quaternion term $\mathbf{Q}_{\delta\boldsymbol{\theta}}$ may be derived as follows,

$$\mathbf{Q}_{\delta\boldsymbol{\theta}} \triangleq \left.\frac{\partial(\mathbf{q}\otimes\delta\mathbf{q})}{\partial\delta\boldsymbol{\theta}}\right|_{\mathbf{q}} = \left.\frac{\partial(\mathbf{q}\otimes\delta\mathbf{q})}{\partial\delta\mathbf{q}}\right|_{\mathbf{q}} \left.\frac{\partial\delta\mathbf{q}}{\partial\delta\boldsymbol{\theta}}\right|_{\hat{\delta\boldsymbol{\theta}}=0} \tag{104a}$$

$$= \mathbf{Q}^{+}(\mathbf{q})\,\frac{1}{2}\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{104b}$$

$$= \frac{1}{2}\begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & -q_z & q_y \\ q_z & q_w & -q_x \\ -q_y & q_x & q_w \end{bmatrix}. \tag{104c}$$

3.2 Injection of the observed error into the nominal state

After the ESKF update, the nominal state gets updated with the observed error state using the appropriate compositions (sums or quaternion products, see Table 1),

$$\mathbf{x} \leftarrow \mathbf{x} \oplus \hat{\delta\mathbf{x}}\,, \tag{105}$$

that is,

$$\mathbf{p} \leftarrow \mathbf{p} + \hat{\delta\mathbf{p}} \tag{106a}$$
$$\mathbf{v} \leftarrow \mathbf{v} + \hat{\delta\mathbf{v}} \tag{106b}$$
$$\mathbf{q} \leftarrow \mathbf{q} \otimes \mathbf{q}\{\hat{\delta\boldsymbol{\theta}}\} \tag{106c}$$
$$\mathbf{a}_b \leftarrow \mathbf{a}_b + \hat{\delta\mathbf{a}}_b \tag{106d}$$
$$\boldsymbol{\omega}_b \leftarrow \boldsymbol{\omega}_b + \hat{\delta\boldsymbol{\omega}}_b \tag{106e}$$
$$\mathbf{g} \leftarrow \mathbf{g} + \hat{\delta\mathbf{g}} \tag{106f}$$

3.3 ESKF reset

After error injection into the nominal state, the error state mean $\hat{\delta\mathbf{x}}$ gets reset. This is especially relevant for the orientation part, as the new orientation error will be expressed locally with respect to the orientation frame of the new nominal state. To make the ESKF update complete, the covariance of the error needs to be updated according to this modification.

Let us call the error reset function $g()$. It is written as follows,

$$\delta\mathbf{x} \leftarrow g(\delta\mathbf{x}) = \delta\mathbf{x} \ominus \hat{\delta\mathbf{x}}\,, \tag{107}$$

where $\ominus$ stands for the composition inverse of $\oplus$. The ESKF error reset operation is thus,

$$\begin{aligned} \hat{\delta\mathbf{x}} &\leftarrow 0 & (108)\\ \mathbf{P} &\leftarrow \mathbf{G}\,\mathbf{P}\,\mathbf{G}^\top\ . & (109)\end{aligned}$$

where $\mathbf{G}$ is the Jacobian matrix defined by,

$$\mathbf{G} \triangleq \left.\frac{\partial g}{\partial \delta\mathbf{x}}\right|_{\hat{\delta\mathbf{x}}}\ . \tag{110}$$

Similarly to what happened with the update Jacobian above, this Jacobian is the identity on all diagonal blocks except in the orientation error. We give here the full expression and proceed in the following section with the derivation of the orientation error block, $\mathbf{I} - \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times$.

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_6 & 0 & 0 \\ 0 & \mathbf{I} - \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times & 0 \\ 0 & 0 & \mathbf{I}_9 \end{bmatrix}\ . \tag{111}$$

3.3.1 Jacobian of the reset operation with respect to the orientation error

We want to obtain the expression of the new angular error with respect to the old error. Consider these facts:

- The true orientation does not change on error reset, *i.e.*, $\mathbf{q}_t^+ \equiv \mathbf{q}_t$. This gives:

$$\mathbf{q}^+ \otimes \delta\mathbf{q}^+ = \mathbf{q} \otimes \delta\mathbf{q}\ . \tag{112}$$

- The observed error mean has been injected into the nominal state (see (106c) and (29)):

$$\mathbf{q}^+ = \mathbf{q} \otimes \hat{\delta\mathbf{q}}\ . \tag{113}$$

Combining both identities we obtain an expression of the new orientation error with respect to the old one and the observed error,

$$\delta\mathbf{q}^+ = \hat{\delta\mathbf{q}}^* \otimes \delta\mathbf{q} = \mathbf{Q}^+(\hat{\delta\mathbf{q}}^*) \cdot \delta\mathbf{q}\ . \tag{114}$$

Considering that $\hat{\delta\mathbf{q}}^* \approx \begin{bmatrix} 1 \\ -\frac{1}{2}\hat{\delta\boldsymbol{\theta}} \end{bmatrix}$, the identity above can be expanded as

$$\begin{bmatrix} 1 \\ \frac{1}{2}\delta\boldsymbol{\theta}^+ \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{2}\hat{\delta\boldsymbol{\theta}}^\top \\ -\frac{1}{2}\hat{\delta\boldsymbol{\theta}} & \mathbf{I} - \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times \end{bmatrix} \cdot \begin{bmatrix} 1 \\ \frac{1}{2}\delta\boldsymbol{\theta} \end{bmatrix} + \mathcal{O}(\|\delta\boldsymbol{\theta}\|^2) \tag{115}$$

which gives one scalar- and one vector- equations,

$$\frac{1}{4}\hat{\delta\boldsymbol{\theta}}^\top \delta\boldsymbol{\theta} = \mathcal{O}(\|\delta\boldsymbol{\theta}\|^2) \tag{116}$$

$$\delta\boldsymbol{\theta}^+ = -\hat{\delta\boldsymbol{\theta}} + \left(\mathbf{I} - \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times\right)\delta\boldsymbol{\theta} + \mathcal{O}(\|\delta\boldsymbol{\theta}\|^2) \tag{117}$$

among which the first one is not very informative in that it is only a relation of infinitesimals. One can show from the second equation that $\hat{\delta\boldsymbol{\theta}}^+ = 0$, which is what we expect from the reset operation. The Jacobian is obtained by simple inspection,

$$\boxed{\frac{\partial \delta\boldsymbol{\theta}^+}{\partial \delta\boldsymbol{\theta}} = \mathbf{I} - \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times} \,. \tag{118}$$

4 Implementation using global angular errors

We explore in this section the implications of having the angular error defined in the global reference, as opposed to the local definition we have used so far. We retrace the development of sections 2 and 3, and particularize the subsections that present changes with respect to the new definition.

A global definition of the angular error $\delta\boldsymbol{\theta}$ implies a composition *on the left hand side*, *i.e.*,

$$\mathbf{q}_t = \delta\mathbf{q} \;\otimes \mathbf{q} = \mathbf{q}\{\delta\boldsymbol{\theta}\} \otimes \mathbf{q}\ .$$

We remark for the sake of completeness that the angular rates vector $\boldsymbol{\omega}$ is locally defined, *i.e.*, $\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega}$ in continuous time, and therefore $\mathbf{q} \leftarrow \mathbf{q} \otimes \mathbf{q}\{\boldsymbol{\omega}\Delta t\}$ in discrete time, regardless of the angular error being defined globally. This is so for convenience, as the measure of the angular rates provided by the gyrometers is in body frame, that is, local.

4.1 The error-state kinematics

We start by writing the equations of the error-state kinematics, and proceed afterwards with comments and proofs.

$$\dot{\delta\mathbf{p}} = \delta\mathbf{v} \tag{119a}$$

$$\dot{\delta\mathbf{v}} = -\left[\mathbf{R}(\mathbf{a}_S - b\mathbf{a}_b)\right]_\times \delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g} - \mathbf{R}\mathbf{a}_n \tag{119b}$$

$$\dot{\delta\boldsymbol{\theta}} = -\mathbf{R}\delta\boldsymbol{\omega}_b - \mathbf{R}\boldsymbol{\omega}_n \tag{119c}$$

$$\dot{\delta\mathbf{a}}_b = \mathbf{a}_w \tag{119d}$$

$$\dot{\delta\boldsymbol{\omega}}_b = \boldsymbol{\omega}_w \tag{119e}$$

$$\dot{\delta\mathbf{g}} = 0\ , \tag{119f}$$

where, again, all equations except those of $\dot{\delta\mathbf{v}}$ and $\dot{\delta\boldsymbol{\theta}}$ are trivial. The non-trivial expressions are developed below.

Equation (119b): The linear velocity error. We wish to determine $\dot{\delta\mathbf{v}}$, the dynamics of the velocity errors. We start with the following relations

$$\mathbf{R}_t = (\mathbf{I} + [\delta\boldsymbol{\theta}]_\times)\mathbf{R} + O(\|\delta\boldsymbol{\theta}\|^2) \tag{120}$$

$$\dot{\mathbf{v}} = \mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{g} \ , \tag{121}$$

where (120) is the small-signal approximation of $\mathbf{R}_t$ using a globally defined error, and in (121) we introduced $\mathbf{a}_\mathcal{B}$ and $\delta\mathbf{a}_\mathcal{B}$ as the large- and small- signal accelerations in body frame, defined in(63) and (64), as we did for the locally-defined case.

We proceed by writing the expression (57b) of $\dot{\mathbf{v}}_t$ in two different forms (left and right developments), where the terms $O(\|\delta\boldsymbol{\theta}\|^2)$ have been ignored,

$$\begin{aligned}
\dot{\mathbf{v}} + \dot{\delta\mathbf{v}} = \boxed{\dot{\mathbf{v}}_t} &= (\mathbf{I} + [\delta\boldsymbol{\theta}]_\times)\mathbf{R}(\mathbf{a}_\mathcal{B} + \delta\mathbf{a}_\mathcal{B}) + \mathbf{g} + \delta\mathbf{g} \\
\mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{g} + \dot{\delta\mathbf{v}} = \quad &= \mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{R}\delta\mathbf{a}_\mathcal{B} + [\delta\boldsymbol{\theta}]_\times \mathbf{R}\mathbf{a}_\mathcal{B} + [\delta\boldsymbol{\theta}]_\times \mathbf{R}\delta\mathbf{a}_\mathcal{B} + \mathbf{g} + \delta\mathbf{g}
\end{aligned}$$

This leads after removing $\mathbf{R}\mathbf{a}_\mathcal{B} + \mathbf{g}$ from left and right to

$$\dot{\delta\mathbf{v}} = \mathbf{R}\delta\mathbf{a}_\mathcal{B} + [\delta\boldsymbol{\theta}]_\times \mathbf{R}(\mathbf{a}_\mathcal{B} + \delta\mathbf{a}_\mathcal{B}) + \delta\mathbf{g} \tag{122}$$

Eliminating the second order terms and reorganizing some cross-products (with $[\mathbf{a}]_\times \mathbf{b} = -[\mathbf{b}]_\times \mathbf{a}$), we get

$$\dot{\delta\mathbf{v}} = \mathbf{R}\delta\mathbf{a}_\mathcal{B} - [\mathbf{R}\mathbf{a}_\mathcal{B}]_\times \delta\boldsymbol{\theta} + \delta\mathbf{g} \ , \tag{123}$$

and finally, recalling (63) and (64) and rearranging, we obtain the expression of the derivative of the velocity error,

$$\boxed{\dot{\delta\mathbf{v}} = -[\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b)]_\times \delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g} - \mathbf{R}\mathbf{a}_n} \tag{124}$$

Equation (119c): The orientation error. We wish to determine $\dot{\delta\boldsymbol{\theta}}$, the dynamics of the angular errors. We start with the following relations

$$\dot{\mathbf{q}}_t = \frac{1}{2}\mathbf{q}_t \otimes \boldsymbol{\omega}_t \tag{125}$$

$$\dot{\mathbf{q}} = \frac{1}{2}\mathbf{q} \otimes \boldsymbol{\omega} \ , \tag{126}$$

which are the true- and nominal- definitions of the quaternion derivatives. We remind that we are using a globally-defined angular error, *i.e.*,

$$\mathbf{q}_t = \delta\mathbf{q} \otimes \mathbf{q} \ . \tag{127}$$

As we did for the locally-defined error case, we also group large- and small-signal angular rates (75–76). We proceed by computing $\dot{\mathbf{q}}_t$ by two different means (left and right developments),

$$\begin{aligned}
\frac{1}{2}\mathbf{q}_t \otimes \boldsymbol{\omega}_t = \boxed{\dot{\mathbf{q}}_t} &= (\dot{\overline{\delta\mathbf{q} \otimes \mathbf{q}}}) \\
\frac{1}{2}\delta\mathbf{q} \otimes \mathbf{q} \otimes \boldsymbol{\omega}_t = \qquad &= \dot{\delta\mathbf{q}} \otimes \mathbf{q} + \delta\mathbf{q} \otimes \dot{\mathbf{q}} \\
&= \dot{\delta\mathbf{q}} \otimes \mathbf{q} + \frac{1}{2}\delta\mathbf{q} \otimes \mathbf{q} \otimes \boldsymbol{\omega} \ .
\end{aligned}$$

Having $\boldsymbol{\omega}_t - \boldsymbol{\omega} = \delta\boldsymbol{\omega}$, we can write

$$\dot{\delta\mathbf{q}} \otimes \mathbf{q} = \frac{1}{2}\delta\mathbf{q} \otimes \mathbf{q} \otimes \delta\boldsymbol{\omega} \ . \tag{128}$$

Multiplying left and right terms by $\mathbf{q}^*$, and recalling that $\mathbf{q} \otimes \delta\boldsymbol{\omega} \otimes \mathbf{q}^* \equiv \mathbf{R}\delta\boldsymbol{\omega}$, we can further develop as follows,

$$\begin{aligned}
\dot{\delta\mathbf{q}} &= \frac{1}{2}\delta\mathbf{q} \otimes \mathbf{q} \otimes \delta\boldsymbol{\omega} \otimes \mathbf{q}^* \\
&= \frac{1}{2}\delta\mathbf{q} \otimes (\mathbf{R}\delta\boldsymbol{\omega}) \\
&= \frac{1}{2}\delta\mathbf{q} \otimes \delta\boldsymbol{\omega}_G \ ,
\end{aligned} \tag{129}$$

with $\delta\boldsymbol{\omega}_G \triangleq \mathbf{R}\delta\boldsymbol{\omega}$ the small-signal angular rate expressed in the global frame. Then,

$$\begin{aligned}
\begin{bmatrix} 0 \\ \dot{\delta\boldsymbol{\theta}} \end{bmatrix} = \boxed{2\dot{\delta\mathbf{q}}} &= \delta\mathbf{q} \otimes \delta\boldsymbol{\omega}_G \\
&= \Omega(\delta\boldsymbol{\omega}_G)\,\delta\mathbf{q} \\
&= \begin{bmatrix} 0 & -\delta\boldsymbol{\omega}_G^\top \\ \delta\boldsymbol{\omega}_G & -[\delta\boldsymbol{\omega}_G]_\times \end{bmatrix} \begin{bmatrix} 1 \\ \delta\boldsymbol{\theta}/2 \end{bmatrix} + O(\|\delta\boldsymbol{\theta}\|^2)
\end{aligned} \tag{130}$$

which results in one scalar- and one vector- equalities

$$0 = \delta\boldsymbol{\omega}_G^\top \delta\boldsymbol{\theta} + O(|\delta\boldsymbol{\theta}|^2) \tag{131}$$

$$\dot{\delta\boldsymbol{\theta}} = \delta\boldsymbol{\omega}_G - \frac{1}{2}\left[\delta\boldsymbol{\omega}_G\right]_\times \delta\boldsymbol{\theta} + O(\|\delta\boldsymbol{\theta}\|^2). \tag{132}$$

The first equation leads to $\delta\boldsymbol{\omega}_G^\top \delta\boldsymbol{\theta} = O(\|\delta\boldsymbol{\theta}\|^2)$, which is formed by second-order infinitesimals, not very useful. The second equation yields, after neglecting all second-order terms,

$$\dot{\delta\boldsymbol{\theta}} = \delta\boldsymbol{\omega}_G = \mathbf{R}\delta\boldsymbol{\omega} \ . \tag{133}$$

Finally, recalling (75) and (76), we obtain the linearized dynamics of the global angular error,

$$\boxed{\dot{\delta\boldsymbol{\theta}} = -\mathbf{R}\delta\boldsymbol{\omega}_b - \mathbf{R}\boldsymbol{\omega}_n} \ . \tag{134}$$

4.2 The differences equations

4.2.1 The nominal state

The nominal state equations do not involve errors and are therefore the same as in the case where the orientation error is defined locally.

4.2.2 The error state

Using Euler integration, we obtain the following set of differences equations,

$$\delta\mathbf{p} \leftarrow \delta\mathbf{p} + \delta\mathbf{v}\,\Delta t \tag{135a}$$
$$\delta\mathbf{v} \leftarrow \delta\mathbf{v} + (-\left[\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b)\right]_\times \delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b + \delta\mathbf{g})\Delta t + \mathbf{v_i} \tag{135b}$$
$$\delta\boldsymbol{\theta} \leftarrow \delta\boldsymbol{\theta} - \mathbf{R}\delta\boldsymbol{\omega}_b\Delta t + \boldsymbol{\theta}_\mathbf{i} \tag{135c}$$
$$\delta\mathbf{a}_b \leftarrow \delta\mathbf{a}_b + \mathbf{a_i} \tag{135d}$$
$$\delta\boldsymbol{\omega}_b \leftarrow \delta\boldsymbol{\omega}_b + \boldsymbol{\omega}_\mathbf{i} \tag{135e}$$
$$\delta\mathbf{g} \leftarrow \delta\mathbf{g}. \tag{135f}$$

4.2.3 The error state Jacobian and perturbation matrices

The Transition matrix is obtained by simple inspection of the equations above,

$$\mathbf{F_x} = \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & 0 & 0 & 0 & 0 \\ 0 & \mathbf{I} & \boxed{-\left[\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b)\right]_\times \Delta t} & -\mathbf{R}\Delta t & 0 & \mathbf{I}\Delta t \\ 0 & 0 & \boxed{\mathbf{I}} & 0 & \boxed{-\mathbf{R}\Delta t} & 0 \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} \end{bmatrix}. \tag{136}$$

We observe three changes with respect to the case with a locally-defined angular error (compare the boxed terms in the Jacobian above to the ones in (93)); these changes are summarized in Table 2.

The perturbation Jacobian and the perturbation matrix are unchanged after considering isotropic noises and the developments of App. E,

$$\mathbf{F_i} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I} \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad , \quad \mathbf{Q_i} = \begin{bmatrix} \mathbf{V_i} & 0 & 0 & 0 \\ 0 & \Theta_\mathbf{i} & 0 & 0 \\ 0 & 0 & \mathbf{A_i} & 0 \\ 0 & 0 & 0 & \Omega_\mathbf{i} \end{bmatrix}. \tag{137}$$

4.3 Fusing with complementary sensory data

The fusing equations involving the ESKF machinery vary only slightly. We revise these variations in the error state observation via ESKF correction, the injection of the error into the nominal state, and the reset step.

4.3.1 Error state observation

The only difference with respect to the local error definition is in the Jacobian block of the observation function that relates the orientation to the angular error. This new block is developed below.

Using (30–31) and the first-order approximation $\delta\mathbf{q} \to \begin{bmatrix} 1 \\ \frac{1}{2}\delta\boldsymbol{\theta} \end{bmatrix}$, the quaternion term $\mathbf{Q}_{\delta\boldsymbol{\theta}}$ may be derived as follows,

$$\mathbf{Q}_{\delta\boldsymbol{\theta}} \triangleq \left.\frac{\partial(\delta\mathbf{q}\otimes\mathbf{q})}{\partial\delta\boldsymbol{\theta}}\right|_{\mathbf{q}} = \left.\frac{\partial(\delta\mathbf{q}\otimes\mathbf{q})}{\partial\delta\mathbf{q}}\right|_{\mathbf{q}} \left.\frac{\partial\delta\mathbf{q}}{\partial\delta\boldsymbol{\theta}}\right|_{\hat{\delta\boldsymbol{\theta}}=0} \tag{138a}$$

$$= \mathbf{Q}^{-}(\mathbf{q})\,\frac{1}{2}\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{138b}$$

$$= \frac{1}{2}\begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & q_z & -q_y \\ -q_z & q_w & q_x \\ q_y & -q_x & q_w \end{bmatrix}\,. \tag{138c}$$

4.3.2 Injection of the observed error into the nominal state

The composition $\mathbf{x} \leftarrow \mathbf{x} \oplus \hat{\delta\mathbf{x}}$ of the nominal and error states is depicted as follows,

$$\mathbf{p} \leftarrow \mathbf{p} + \delta\mathbf{p} \tag{139a}$$

$$\mathbf{v} \leftarrow \mathbf{v} + \delta\mathbf{v} \tag{139b}$$

$$\mathbf{q} \leftarrow \mathbf{q}\{\hat{\delta\boldsymbol{\theta}}\} \otimes \mathbf{q} \tag{139c}$$

$$\mathbf{a}_b \leftarrow \mathbf{a}_b + \delta\mathbf{a}_b \tag{139d}$$

$$\boldsymbol{\omega}_b \leftarrow \boldsymbol{\omega}_b + \delta\boldsymbol{\omega}_b \tag{139e}$$

$$\mathbf{g} \leftarrow \mathbf{g} + \delta\mathbf{g}\,. \tag{139f}$$

where only the equation for the quaternion update has been affected. This is summarized in Table 2.

4.3.3 ESKF reset

The ESKF error mean is reset, and the covariance updated, according to,

$$\hat{\delta\mathbf{x}} \leftarrow 0 \tag{140}$$
$$\mathbf{P} \leftarrow \mathbf{G}\mathbf{P}\mathbf{G}^\top \tag{141}$$

with the Jacobian

$$\mathbf{G} = \begin{bmatrix} \mathbf{I}_6 & 0 & 0 \\ 0 & \mathbf{I} + \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times & 0 \\ 0 & 0 & \mathbf{I}_9 \end{bmatrix} \tag{142}$$

whose non-trivial term is developed as follows. Our goal is to obtain the expression of the new angular error $\delta\boldsymbol{\theta}^+$ with respect to the old error $\delta\boldsymbol{\theta}$. We consider these facts:

- The true orientation does not change on error reset, *i.e.*, $\mathbf{q}_t^+ \equiv \mathbf{q}_t$. This gives:

$$\delta\mathbf{q}^+ \otimes \mathbf{q}^+ = \delta\mathbf{q} \otimes \mathbf{q}\ . \tag{143}$$

- The observed error mean has been injected into the nominal state (see (106c) and (29)):

$$\mathbf{q}^+ = \hat{\delta\mathbf{q}} \otimes \mathbf{q}\ . \tag{144}$$

Combining both identities we obtain an expression of the new orientation error with respect to the old one and the observed error $\hat{\delta\mathbf{q}}$,

$$\delta\mathbf{q}^+ = \delta\mathbf{q} \otimes \hat{\delta\mathbf{q}}^* = \mathbf{Q}^-(\hat{\delta\mathbf{q}}^*) \cdot \delta\mathbf{q}\ . \tag{145}$$

Considering that $\hat{\delta\mathbf{q}}^* \approx \begin{bmatrix} 1 \\ -\frac{1}{2}\hat{\delta\boldsymbol{\theta}} \end{bmatrix}$, the identity above can be expanded as

$$\begin{bmatrix} 1 \\ \frac{1}{2}\delta\boldsymbol{\theta}^+ \end{bmatrix} = \begin{bmatrix} 1 & \frac{1}{2}\hat{\delta\boldsymbol{\theta}}^\top \\ -\frac{1}{2}\hat{\delta\boldsymbol{\theta}} & \mathbf{I} + \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times \end{bmatrix} \cdot \begin{bmatrix} 1 \\ \frac{1}{2}\delta\boldsymbol{\theta} \end{bmatrix} + \mathcal{O}(\|\delta\boldsymbol{\theta}\|^2) \tag{146}$$

which gives one scalar- and one vector- equations,

$$\frac{1}{4}\hat{\delta\boldsymbol{\theta}}^\top\delta\boldsymbol{\theta} = \mathcal{O}(\|\delta\boldsymbol{\theta}\|^2) \tag{147}$$
$$\delta\boldsymbol{\theta}^+ = -\hat{\delta\boldsymbol{\theta}} + \left(\mathbf{I} + \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times\right)\delta\boldsymbol{\theta} + \mathcal{O}(\|\delta\boldsymbol{\theta}\|^2) \tag{148}$$

among which the first one is not very informative in that it is only a relation of infinitesimals. One can show from the second equation that $\hat{\delta\boldsymbol{\theta}}^+ = 0$, which is what we expect from the reset operation. The Jacobian is obtained by simple inspection,

$$\boxed{\frac{\partial\delta\boldsymbol{\theta}^+}{\partial\delta\boldsymbol{\theta}} = \mathbf{I} + \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times}\ . \tag{149}$$

The difference with respect to the local error case is summarized in Table 2.

Table 2: Algorithm modifications related to the definition of the orientation errors.

Context	Item	local angular error	global angular error
Error integration	$\partial\delta\mathbf{v}_{k+1} \,/\, \partial\delta\boldsymbol{\theta}_k$	$-\mathbf{R}\left[\mathbf{a}_S - \mathbf{a}_b\right]_\times \Delta t$	$-\left[\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b)\right]_\times \Delta t$
	$\partial\delta\boldsymbol{\theta}_{k+1} \,/\, \partial\delta\boldsymbol{\theta}_k$	$\mathbf{R}^\top\{(\boldsymbol{\omega}_S - \boldsymbol{\omega}_b)\Delta t\}$	$\mathbf{I}$
	$\partial\delta\theta_{k+1} \,/\, \partial\delta\boldsymbol{\omega}_{b,k}$	$-\mathbf{I}\Delta t$	$-\mathbf{R}\Delta t$
Error observation	$\partial(\delta\mathbf{q} \otimes \mathbf{q}) \,/\, \partial\delta\boldsymbol{\theta}$	$\frac{1}{2}\begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & -q_z & q_y \\ q_z & q_w & -q_x \\ -q_y & q_x & q_w \end{bmatrix}$	$\frac{1}{2}\begin{bmatrix} -q_x & -q_y & -q_z \\ q_w & q_z & -q_y \\ -q_z & q_w & q_x \\ q_y & -q_x & q_w \end{bmatrix}$
Error injection	$\mathbf{q}$	$\mathbf{q} \leftarrow \mathbf{q} \otimes \mathbf{q}\{\hat{\delta\boldsymbol{\theta}}\}$	$\mathbf{q} \leftarrow \mathbf{q}\{\hat{\delta\boldsymbol{\theta}}\} \otimes \mathbf{q}$
Error reset	$\partial\delta\boldsymbol{\theta}^+ \,/\, \partial\delta\boldsymbol{\theta}$	$\mathbf{I} - \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times$	$\mathbf{I} + \left[\frac{1}{2}\hat{\delta\boldsymbol{\theta}}\right]_\times$

A Runge-Kutta numerical integration methods

We aim at integrating nonlinear differential equations of the form

$$\dot{\mathbf{x}} = f(t, \mathbf{x}) \tag{150}$$

over a limited time interval Δt, in order to convert them to a differences equation, *i.e.*,

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(t) + \int_t^{t+\Delta t} f(\tau, \mathbf{x}(\tau)) d\tau \ , \tag{151}$$

or equivalently, if we assume that $t_n = n\Delta t$ and $\mathbf{x}_n \triangleq \mathbf{x}(t_n)$,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \int_{n\Delta t}^{(n+1)\Delta t} f(\tau, \mathbf{x}(\tau)) d\tau \ . \tag{152}$$

One of the most utilized family of methods is the Runge-Kutta methods (from now on, RK). These methods use several iterations to estimate the derivative over the interval, and then use this derivative to integrate over the step Δt.

In the sections that follow, several RK methods are presented, from the simplest one to the most general one, and are named according to their most common name.

NOTE: All the material here is taken from the *Runge-Kutta method* entry in the English Wikipedia.

A.1 The Euler method

The Euler method assumes that the derivative $f(\cdot)$ is constant over the interval, and therefore

$$\boxed{\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \cdot f(t_n, \mathbf{x}_n) \ .} \tag{153}$$

Put as a general RK method, this corresponds to a single-stage method, which can be depicted as follows. Compute the derivative at the initial point,

$$k_1 = f(t_n, \mathbf{x}_n) \ , \tag{154}$$

and use it to compute the integrated value at the end point,

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \cdot k_1 \ . \tag{155}$$

A.2 The midpoint method

The midpoint method assumes that the derivative is the one at the midpoint of the interval, and makes one iteration to compute the value of $\mathbf{x}$ at this midpoint, *i.e.*,

$$\boxed{\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \cdot f\Big(t_n + \frac{1}{2}\Delta t \ , \ \mathbf{x}_n + \frac{1}{2}\Delta t \cdot f(t_n, \mathbf{x}_n)\Big)} \ . \tag{156}$$

The midpoint method can be explained as a two-step method as follows. First, use the Euler method to integrate until the midpoint, using k_1 as defined previously,

$$k_1 = f(t_n, \mathbf{x}_n) \tag{157}$$

$$\mathbf{x}(t_n + \tfrac{1}{2}\Delta t) = \mathbf{x}_n + \frac{1}{2}\Delta t{\cdot}k_1 \ . \tag{158}$$

Then use this value to evaluate the derivative at the midpoint, k_2, leading to the integration

$$k_2 = f(t_n + \tfrac{1}{2}\Delta t \ , \ \mathbf{x}(t_n + \tfrac{1}{2}\Delta t)) \tag{159}$$

$$\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t{\cdot}k_2 \ . \tag{160}$$

A.3 The RK4 method

This is usually referred to as simply the Runge-Kutta method. It assumes evaluation values for $f()$ at the start, midpoint and end of the interval. And it uses four stages or iterations to compute the integral, with four derivatives, $k_1 \ldots k_4$, that are obtained sequentially. These derivatives, or *slopes*, are then weight-averaged to obtain the 4th-order estimate of the derivative in the interval.

The RK4 method is better specified as a small algorithm than a one-step formula like the two methods above. The RK4 integration step is,

$$\boxed{\mathbf{x}_{n+1} = \mathbf{x}_n + \frac{\Delta t}{6}\Big(k_1 + 2k_2 + 2k_3 + k_4\Big)} \ , \tag{161}$$

that is, the increment is computed by assuming a slope which is the weighted average of the slopes k_1, k_2, k_3, k_4, with

$$k_1 = f(t_n, \mathbf{x}_n) \tag{162}$$

$$k_2 = f\Big(t_n + \frac{1}{2}\Delta t \ , \ \mathbf{x}_n + \frac{\Delta t}{2}k_1\Big) \tag{163}$$

$$k_3 = f\Big(t_n + \frac{1}{2}\Delta t \ , \ \mathbf{x}_n + \frac{\Delta t}{2}k_2\Big) \tag{164}$$

$$k_4 = f\Big(t_n + \Delta t \ , \ \mathbf{x}_n + \Delta t{\cdot}k_3\Big) \ . \tag{165}$$

The different slopes have the following interpretation:

- k_1 is the slope at the beginning of the interval, using $\mathbf{x}_n$, (Euler's method);
- k_2 is the slope at the midpoint of the interval, using $\mathbf{x}_n + \frac{1}{2}\Delta t{\cdot}k_1$, (midpoint method);
- k_3 is again the slope at the midpoint, but now using $\mathbf{x}_n + \frac{1}{2}\Delta t{\cdot}k_2$;
- k_4 is the slope at the end of the interval, using $\mathbf{x}_n + \Delta t{\cdot}k_3$.

A.4 General Runge-Kutta method

More elaborated RK methods are possible. They aim at either reduce the error and/or increase stability. They take the general form

$$\boxed{\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta t \sum_{i=1}^{s} b_i k_i} \ , \tag{166}$$

where

$$k_i = f\Big(t_n + \Delta t \cdot c_i, \mathbf{x}_n + \Delta t \sum_{j=1}^{s} a_{ij} k_j\Big) \ , \tag{167}$$

that is, the number of iterations (the order of the method) is s, the averaging weights are defined by b_i, the evaluation time instants by c_i, and the slopes k_i are determined using the values a_{ij}. Depending on the structure of the terms a_{ij}, one can have *explicit* or *implicit* RK methods.

- In explicit methods, all k_i are computed sequentially, *i.e.*, using only previously computed values. This implies that the matrix $[a_{ij}]$ is lower triangular with zero diagonal entries (*i.e.*, $a_{ij} = 0$ for $j \geq i$). Euler, midpoint and RK4 methods are explicit.
- Implicit methods have a full $[a_{ij}]$ matrix and require the solution of a linear set of equations to determine all k_i. They are therefore costlier to compute, but they are able to improve on accuracy and stability with respect to explicit methods.

Please refer to specialized documentation for more detailed information.

B Closed-form integration methods

In many cases it is possible to arrive to a closed-form expression for the integration step. We consider now the case of a first-order linear differential equation,

$$\dot{\mathbf{x}}(t) = \mathbf{A} \cdot \mathbf{x}(t) \ , \tag{168}$$

that is, the relation is linear and constant over the interval. In such cases, the integration over the interval $[t_n, t_n + \Delta t]$ results in

$$\mathbf{x}_{n+1} = e^{\mathbf{A} \Delta t} \mathbf{x}_n = \Phi \mathbf{x}_n \ , \tag{169}$$

where Φ is known as the transition matrix. The Taylor expansion of this transition matrix is

$$\Phi = e^{\mathbf{A}\Delta t} = \mathbf{I} + \mathbf{A}\Delta t + \frac{1}{2}\mathbf{A}^2 \Delta t^2 + \frac{1}{3!}\mathbf{A}^3 \Delta t^3 + \cdots = \sum_{k=0}^{\infty} \frac{1}{k!} \mathbf{A}^k \Delta t^k \ . \tag{170}$$

When writing this series for known instances of $\mathbf{A}$, it is sometimes possible to identify known series in the result. This allows writing the resulting integration in closed form. A few examples follow.

B.1 Integration of the angular error

For example, consider the angular error dynamics without bias and noise (a cleaned version of Eq. (60c)),

$$\dot{\delta\boldsymbol{\theta}} = -\left[\boldsymbol{\omega}\right]_\times \delta\boldsymbol{\theta} \tag{171}$$

Its transition matrix can be written as a Taylor series,

$$\begin{aligned}\Phi &= e^{-[\boldsymbol{\omega}]_\times \Delta t} && (172)\\ &= \mathbf{I} - \left[\boldsymbol{\omega}\right]_\times \Delta t + \frac{1}{2}\left[\boldsymbol{\omega}\right]_\times^2 \Delta t^2 - \frac{1}{3!}\left[\boldsymbol{\omega}\right]_\times^3 \Delta t^3 + \frac{1}{4!}\left[\boldsymbol{\omega}\right]_\times^4 \Delta t^4 - \ldots && (173)\end{aligned}$$

Now defining $\boldsymbol{\omega}\Delta t \triangleq \mathbf{u}\Delta\theta$, the unitary axis of rotation and the rotated angle, and applying (12), we can group terms and get

$$\begin{aligned}\Phi &= \mathbf{I} - \left[\mathbf{u}\right]_\times \Delta\theta + \frac{1}{2}\left[\mathbf{u}\right]_\times^2 \Delta\theta^2 - \frac{1}{3!}\left[\mathbf{u}\right]_\times^3 \Delta\theta^3 + \frac{1}{4!}\left[\mathbf{y}\right]_\times^4 \Delta\theta^4 - \ldots\\ &= \mathbf{I} - \left[\mathbf{u}\right]_\times \left(\Delta\theta - \frac{\Delta\theta^3}{3!} + \frac{\Delta\theta^5}{5!} - \cdots\right) + \left[\mathbf{u}\right]_\times^2 \left(\frac{\Delta\theta^2}{2!} - \frac{\Delta\theta^4}{4!} + \frac{\Delta\theta^6}{6!} - \cdots\right)\\ &= \mathbf{I} - \left[\mathbf{u}\right]_\times \sin\Delta\theta + \left[\mathbf{u}\right]_\times^2 (1 - \cos\Delta\theta)\ , && (174)\end{aligned}$$

which is a closed-form solution.

This solution corresponds to a rotation matrix, $\Phi = \mathbf{R}\{-\mathbf{u}\Delta\theta\} = \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top$, according to (13), a result that could be obtained by direct inspection of (172) and recalling (8). Let us therefore write this as the final closed-form result,

$$\boxed{\Phi = \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top}\ . \tag{175}$$

B.2 Simplified IMU example

Consider the simplified, IMU driven system with error-state dynamics governed by,

$$\begin{aligned}\dot{\delta\mathbf{p}} &= \delta\mathbf{v} && (176a)\\ \dot{\delta\mathbf{v}} &= -\mathbf{R}\left[\mathbf{a}\right]_\times \delta\boldsymbol{\theta} && (176b)\\ \dot{\delta\boldsymbol{\theta}} &= -\left[\boldsymbol{\omega}\right]_\times \delta\boldsymbol{\theta}\ , && (176c)\end{aligned}$$

where $(\mathbf{a}, \boldsymbol{\omega})$ are the IMU readings, and we have obviated gravity and sensor biases. This system is defined by the state vector and the dynamic matrix,

$$\mathbf{x} = \begin{bmatrix}\mathbf{p}\\ \mathbf{v}\\ \boldsymbol{\theta}\end{bmatrix} \qquad \mathbf{A} = \begin{bmatrix}0 & \mathbf{P_v} & 0\\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\\ 0 & 0 & \Theta_{\boldsymbol{\theta}}\end{bmatrix}\ . \tag{177}$$

with

$$
\begin{aligned}
\mathbf{P_v} &= \mathbf{I} && (178)\\
\mathbf{V}_{\boldsymbol{\theta}} &= -\mathbf{R}\,[\mathbf{a}]_\times && (179)\\
\Theta_{\boldsymbol{\theta}} &= -\,[\boldsymbol{\omega}]_\times && (180)
\end{aligned}
$$

Its integration with a step time Δt is $\mathbf{x}_{n+1} = e^{(\mathbf{A}\Delta t)}\cdot\mathbf{x}_n = \Phi\cdot\mathbf{x}_n$. The transition matrix Φ admits a Taylor development (170), in increasing powers of $\mathbf{A}\Delta t$.

We can write a few powers of $\mathbf{A}$ to get an illustration of their general form,

$$
\mathbf{A} = \begin{bmatrix} 0 & \mathbf{P_v} & 0 \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}} \\ 0 & 0 & \Theta_{\boldsymbol{\theta}} \end{bmatrix}, \mathbf{A}^2 = \begin{bmatrix} 0 & 0 & \mathbf{P}_v\mathbf{V}_{\boldsymbol{\theta}} \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}} \\ 0 & 0 & \Theta_{\boldsymbol{\theta}}^2 \end{bmatrix}, \mathbf{A}^3 = \begin{bmatrix} 0 & 0 & \mathbf{P}_v\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}} \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^2 \\ 0 & 0 & \Theta_{\boldsymbol{\theta}}^3 \end{bmatrix}, \mathbf{A}^4 = \begin{bmatrix} 0 & 0 & \mathbf{P}_v\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^2 \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^3 \\ 0 & 0 & \Theta_{\boldsymbol{\theta}}^4 \end{bmatrix}, \tag{181}
$$

from which it is now visible that, for $k > 1$,

$$
\mathbf{A}^{k>1} = \begin{bmatrix} 0 & 0 & \mathbf{P}_v\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^{k-2} \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^{k-1} \\ 0 & 0 & \Theta_{\boldsymbol{\theta}}^{k} \end{bmatrix} \tag{182}
$$

We can observe that the terms in the increasing powers of $\mathbf{A}$ have a fixed part and an increasing power of $\Theta_{\boldsymbol{\theta}}$. These powers can lead to closed form solutions, as in the previous section.

Let us partition the matrix Φ as follows,

$$
\Phi = \begin{bmatrix} \mathbf{I} & \Phi(\mathbf{p},\mathbf{v}) & \Phi(\mathbf{p},\boldsymbol{\theta}) \\ 0 & \mathbf{I} & \Phi(\mathbf{v},\boldsymbol{\theta}) \\ 0 & 0 & \Phi(\boldsymbol{\theta},\boldsymbol{\theta}) \end{bmatrix} , \tag{183}
$$

and let us advance step by step, exploring all the non-zero blocks of Φ one by one.

Trivial diagonal terms Starting by the two upper terms in the diagonal, they are the identity as shown.

Rotational diagonal term Next is the rotational diagonal term $\Phi(\boldsymbol{\theta},\boldsymbol{\theta})$, relating the new angular error to the old angular error. Writing the full Taylor series for this term leads to

$$
\Phi(\boldsymbol{\theta},\boldsymbol{\theta}) = \sum_{k=0}^{\infty}\frac{1}{k!}\Theta_{\boldsymbol{\theta}}^k\Delta t^k = \sum_{k=0}^{\infty}\frac{1}{k!}\,[-\boldsymbol{\omega}]_\times^k\,\Delta t^k\ , \tag{184}
$$

which corresponds, as we have seen in the previous section, to our well-known rotation matrix,

$$
\boxed{\Phi(\boldsymbol{\theta},\boldsymbol{\theta}) = \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top}\ . \tag{185}
$$

Position-vs-velocity term The simplest off-diagonal term is $\Phi(\mathbf{p}, \mathbf{v})$, which is

$$\boxed{\Phi(\mathbf{p}, \mathbf{v}) = \mathbf{P_v}\Delta t = \mathbf{I}\Delta t} \ . \tag{186}$$

Velocity-vs-angle term Let us now move to the term $\Phi(\mathbf{v}, \boldsymbol{\theta})$, by writing its series,

$$\Phi(\mathbf{v}, \boldsymbol{\theta}) = \mathbf{V}_{\boldsymbol{\theta}}\Delta t + \frac{1}{2}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}\Delta t^2 + \frac{1}{3!}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^2\Delta t^3 + \cdots \tag{187}$$

At this point we have two options. We can truncate the series at the first significant term, obtaining $\Phi(\mathbf{v}, \boldsymbol{\theta}) = \mathbf{V}_{\boldsymbol{\theta}}\Delta t$, but this would not be a closed-form. See next section for results using this simplified method.

Alternatively, let us factor $\mathbf{V}_{\boldsymbol{\theta}}$ out and write

$$\Phi(\mathbf{v}, \boldsymbol{\theta}) = \mathbf{V}_{\boldsymbol{\theta}}\Sigma_1 \tag{188}$$

with

$$\Sigma_1 = \mathbf{I}\Delta t + \frac{1}{2}\Theta_{\boldsymbol{\theta}}\Delta t^2 + \frac{1}{3!}\Theta_{\boldsymbol{\theta}}^2\Delta t^3 + \cdots \ . \tag{189}$$

The series Σ_1 ressembles the series we wrote for $\Phi(\boldsymbol{\theta}, \boldsymbol{\theta})$, (184), with two exceptions:

- The powers of $\Theta_{\boldsymbol{\theta}}$ in Σ_1 do not match with the rational coefficients $\frac{1}{k!}$ and with the powers of Δt. In fact, we remark here that the subindex "1" in Σ_1 denotes the fact that one power of $\Theta_{\boldsymbol{\theta}}$ is missing in each of the members.
- Some terms at the start of the series are missing. Again, the subindex "1" indicates that one such term is missing.

The first issue may be solved by applying (12) to (180), which yields the identity

$$\Theta_{\boldsymbol{\theta}} = \frac{[\boldsymbol{\omega}]_\times^3}{\|\boldsymbol{\omega}\|^2} = \frac{-\Theta_{\boldsymbol{\theta}}^3}{\|\boldsymbol{\omega}\|^2} \ . \tag{190}$$

This expression allows us to increase the exponents of $\Theta_{\boldsymbol{\theta}}$ in the series by two, and write, if $\boldsymbol{\omega} \neq 0$,

$$\Sigma_1 = \mathbf{I}\Delta t - \frac{\Theta_{\boldsymbol{\theta}}}{\|\boldsymbol{\omega}\|^2}\left(\frac{1}{2}\Theta_{\boldsymbol{\theta}}^2\Delta t^2 + \frac{1}{3!}\Theta_{\boldsymbol{\theta}}^3\Delta t^3 + \ldots\right) \ , \tag{191}$$

and $\Sigma_1 = \mathbf{I}\Delta t$ otherwise. All the powers in the new series match with the correct coefficients. Of course, and as indicated before, some terms are missing. This second issue can be solved by adding and substracting the missing terms, and substituting the full series by its closed form. We obtain

$$\Sigma_1 = \mathbf{I}\Delta t - \frac{\Theta_{\boldsymbol{\theta}}}{\|\boldsymbol{\omega}\|^2}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \mathbf{I} - \Theta_{\boldsymbol{\theta}}\Delta t\right) \ , \tag{192}$$

which is a closed-form solution valid if $\boldsymbol{\omega} \neq 0$. Therefore we can finally write

$$\boxed{\Phi(\mathbf{v},\boldsymbol{\theta}) = \begin{cases} -\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t & \boldsymbol{\omega} \to 0 \\ -\mathbf{R}\left[\mathbf{a}\right]_\times \left(\mathbf{I}\Delta t + \frac{\left[\boldsymbol{\omega}\right]_\times}{\|\boldsymbol{\omega}\|^2}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \mathbf{I} + \left[\boldsymbol{\omega}\right]_\times \Delta t\right)\right) & \boldsymbol{\omega} \neq 0 \end{cases}} \tag{193a, 193b}$$

Position-vs-angle term Let us finally board the term $\Phi(\mathbf{p},\boldsymbol{\theta})$. Its Taylor series is,

$$\Phi(\mathbf{p},\boldsymbol{\theta}) = \frac{1}{2}\mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Delta t^2 + \frac{1}{3!}\mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}\Delta t^3 + \frac{1}{4!}\mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^2\Delta t^4 + \cdots \tag{194}$$

We factor out the constant terms and get,

$$\Phi(\mathbf{p},\boldsymbol{\theta}) = \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\ \Sigma_2\ , \tag{195}$$

with

$$\Sigma_2 = \frac{1}{2}\mathbf{I}\Delta t^2 + \frac{1}{3!}\Theta_{\boldsymbol{\theta}}\Delta t^3 + \frac{1}{4!}\Theta_{\boldsymbol{\theta}}^2\Delta t^4 + \cdots\ . \tag{196}$$

where we note the subindex "2" in Σ_2 admits the following interpretation:

- Two powers of $\Theta_{\boldsymbol{\theta}}$ are missing in each term of the series,
- The first two terms of the series are missing.

Again, we use (190) to increase the exponents of $\Theta_{\boldsymbol{\theta}}$, yielding

$$\Sigma_2 = \frac{1}{2}\mathbf{I}\Delta t^2 - \frac{1}{\|\boldsymbol{\omega}\|^2}\left(\frac{1}{3!}\Theta_{\boldsymbol{\theta}}^3\Delta t^3 + \frac{1}{4!}\Theta_{\boldsymbol{\theta}}^4\Delta t^4 + \cdots\right)\ . \tag{197}$$

We substitute the incomplete series by its closed form,

$$\Sigma_2 = \frac{1}{2}\mathbf{I}\Delta t^2 - \frac{1}{\|\boldsymbol{\omega}\|^2}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \mathbf{I} - \Theta_{\boldsymbol{\theta}}\Delta t - \frac{1}{2}\Theta_{\boldsymbol{\theta}}^2\Delta t^2\right)\ , \tag{198}$$

which leads to the final result

$$\boxed{\Phi(\mathbf{p},\boldsymbol{\theta}) = \begin{cases} -\mathbf{R}\left[\mathbf{a}\right]_\times \frac{\Delta t^2}{2} & \boldsymbol{\omega} \to 0 \\ -\mathbf{R}\left[\mathbf{a}\right]_\times \left(\frac{1}{2}\mathbf{I}\Delta t^2 - \frac{1}{\|\boldsymbol{\omega}\|^2}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \sum_{k=0}^{2}\frac{(-\left[\boldsymbol{\omega}\right]_\times \Delta t)^k}{k!}\right)\right) & \omega \neq 0 \end{cases}}$$

(199a)

(199b)

B.3 Full IMU example

In order to give means to generalize the methods exposed in the simplified IMU example, we need to examine the full IMU case from a little closer.

Consider the full IMU system (60), which can be posed as

$$\dot{\delta\mathbf{x}} = \mathbf{A}\delta\mathbf{x} + \mathbf{B}\mathbf{w} \ , \tag{200}$$

whose discrete-time integration requires the transition matrix

$$\Phi = \sum_{k=0}^{\infty} \frac{1}{k!}\mathbf{A}^k \Delta t^k = \mathbf{I} + \mathbf{A}\Delta t + \frac{1}{2}\mathbf{A}^2\Delta t^2 + \dots \ , \tag{201}$$

which we wish to compute. The dynamic matrix $\mathbf{A}$ is block-sparse, and its blocks can be easily determined by examining the original equations (60),

$$\mathbf{A} = \begin{bmatrix} 0 & \mathbf{P_v} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}} & \mathbf{V_a} & 0 & \mathbf{V_g} \\ 0 & 0 & \Theta_{\boldsymbol{\theta}} & 0 & \Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} . \tag{202}$$

As we did before, let us write a few powers of $\mathbf{A}$,

$$\mathbf{A}^2 = \begin{bmatrix} 0 & 0 & \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}} & \mathbf{P_v}\mathbf{V_a} & 0 & \mathbf{P_v}\mathbf{V_g} \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}} & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & \Theta_{\boldsymbol{\theta}}^2 & 0 & \Theta_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A}^3 = \begin{bmatrix} 0 & 0 & \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}} & 0 & \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^2 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & \Theta_{\boldsymbol{\theta}}^3 & 0 & \Theta_{\boldsymbol{\theta}}^2\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A}^4 = \begin{bmatrix} 0 & 0 & \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^2 & 0 & \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^3 & 0 & \mathbf{V}_{\boldsymbol{\theta}}\Theta_{\boldsymbol{\theta}}^2\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & \Theta_{\boldsymbol{\theta}}^4 & 0 & \Theta_{\boldsymbol{\theta}}^3\Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} .$$

Basically, we observe the following,

- The only term in the diagonal of $\mathbf{A}$, the rotational term $\Theta_{\boldsymbol{\theta}}$, propagates right and up in the sequence of powers $\mathbf{A}^k$. All terms not affected by this propagation vanish. This propagation afects the structure of the sequence $\{\mathbf{A}^k\}$ in the three following aspects:

- The sparsity of the powers of $\mathbf{A}$ is stabilized after the 3rd power. That is to say, there are no more non-zero blocks appearing or vanishing for powers of $\mathbf{A}$ higher than 3.

- The upper-left 3×3 block, corresponding to the simplified IMU model in the previous example, has not changed with respect to that example. Therefore, its closed-form solution developed before holds.

- The terms related to the gyrometer bias error (those of the fifth column) introduce a similar series of powers of $\Theta_{\boldsymbol{\theta}}$, which can be solved with the same techniques we used in the simplified example.

We are interested at this point in finding a generalised method to board the construction of the closed-form elements of the transition matrix Φ. Let us recall the remarks we made about the series Σ_1 and Σ_2,

- The subindex coincides with the lacking powers of $\Theta_{\boldsymbol{\theta}}$ in each of the members of the series.

- The subindex coincides with the number of terms missing at the beginning of the series.

Taking care of these properties, let us introduce the series $\Sigma_n(\mathbf{X}, y)$, defined by[5]

$$\Sigma_n(\mathbf{X}, y) \triangleq \sum_{k=n}^{\infty} \frac{1}{k!}\mathbf{X}^{k-n}y^k \tag{203}$$

in which the sum starts at term n and the terms lack n powers of the matrix $\mathbf{X}$. It follows immediately that Σ_1 and Σ_2 respond to

$$\Sigma_n = \Sigma_n(\Theta_{\boldsymbol{\theta}}, \Delta t) \ , \tag{204}$$

and that $\Sigma_0 = \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top$. We can now write the transition matrix (201) as a function of these series,

$$\Phi = \begin{bmatrix} \mathbf{I} & \mathbf{P_v}\Delta t & \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Sigma_2 & \frac{1}{2}\mathbf{P_v}\mathbf{V_a}\Delta t^2 & \mathbf{P_v}\mathbf{V}_{\boldsymbol{\theta}}\Sigma_3\boldsymbol{\theta}_{\boldsymbol{\omega}} & \frac{1}{2}\mathbf{P_v}\mathbf{V_g}\Delta t^2 \\ 0 & \mathbf{I} & \mathbf{V}_{\boldsymbol{\theta}}\Sigma_1 & \mathbf{V_a}\Delta t & \mathbf{V}_{\boldsymbol{\theta}}\Sigma_2\boldsymbol{\theta}_{\boldsymbol{\omega}} & \mathbf{V_g}\Delta t \\ 0 & 0 & \Sigma_0 & 0 & \Sigma_1\boldsymbol{\theta}_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} \end{bmatrix} . \tag{205}$$

[5] Note that, being $\mathbf{X}$ a square matrix that is not necessarily invertible (as it is the case for $\mathbf{X} = \Theta_{\boldsymbol{\theta}}$), we are not allowed to rearrange the definition of Σ_n with $\Sigma_n = \mathbf{X}^{-n}\sum_{k=n}^{\infty}\frac{1}{k!}(y\mathbf{X})^k$.

Our problem has now derived to the problem of finding a general, closed-form expression for Σ_n. Let us observe the closed-form results we have obtained so far,

$$\Sigma_0 = \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top \tag{206}$$

$$\Sigma_1 = \mathbf{I}\Delta t - \frac{\Theta_{\boldsymbol{\theta}}}{\|\boldsymbol{\omega}\|^2}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \mathbf{I} - \Theta_{\boldsymbol{\theta}}\Delta t\right) \tag{207}$$

$$\Sigma_2 = \frac{1}{2}\mathbf{I}\Delta t^2 - \frac{1}{\|\boldsymbol{\omega}\|^2}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \mathbf{I} - \Theta_{\boldsymbol{\theta}}\Delta t - \frac{1}{2}\Theta_{\boldsymbol{\theta}}^2\Delta t^2\right) \ . \tag{208}$$

In order to develop Σ_3, we need to apply the identity (190) twice (because we lack three powers, and each application of (190) increases this number by only two), getting

$$\Sigma_3 = \frac{1}{3!}\mathbf{I}\Delta t^3 + \frac{\Theta_{\boldsymbol{\theta}}}{\|\boldsymbol{\omega}\|^4}\left(\frac{1}{4!}\Theta_{\boldsymbol{\theta}}^4\Delta t^4 + \frac{1}{5!}\Theta_{\boldsymbol{\theta}}^5\Delta t^5 + \dots\right) \ , \tag{209}$$

which leads to

$$\Sigma_3 = \frac{1}{3!}\mathbf{I}\Delta t^3 + \frac{\Theta_{\boldsymbol{\theta}}}{\|\boldsymbol{\omega}\|^4}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \mathbf{I} - \Theta_{\boldsymbol{\theta}}\Delta t - \frac{1}{2}\Theta_{\boldsymbol{\theta}}^2\Delta t^2 - \frac{1}{3!}\Theta_{\boldsymbol{\theta}}^3\Delta t^3\right) \ . \tag{210}$$

By careful inspection of the series $\Sigma_0 \dots \Sigma_3$, we can now derive a general, closed-form expression for Σ_n, as follows,

$$\boxed{\Sigma_n = \begin{cases} \dfrac{1}{n!}\mathbf{I}\Delta t^n & \boldsymbol{\omega} \to 0 \quad (211a) \\ \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top & n = 0 \quad (211b) \\ \dfrac{1}{n!}\mathbf{I}\Delta t^n - \dfrac{(-1)^{\frac{n+1}{2}}\,[\boldsymbol{\omega}]_\times}{\|\boldsymbol{\omega}\|^{n+1}}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \displaystyle\sum_{k=0}^{n}\frac{(-[\boldsymbol{\omega}]_\times\Delta t)^k}{k!}\right) & n \text{ odd} \quad (211c) \\ \dfrac{1}{n!}\mathbf{I}\Delta t^n + \dfrac{(-1)^{\frac{n}{2}}}{\|\boldsymbol{\omega}\|^{n}}\left(\mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top - \displaystyle\sum_{k=0}^{n}\frac{(-[\boldsymbol{\omega}]_\times\Delta t)^k}{k!}\right) & n \text{ even} \quad (211d) \end{cases}}$$

The final result for the transition matrix Φ follows immediately by substituting the appropriate values of $\Sigma_n,\ n \in \{0, 1, 2, 3\}$, in the corresponding positions of (205).

It might be worth noticing that the series now appearing in these new expressions of Σ_n have a finite number of terms, and thus that they can be effectively computed. That is to say, the expression of Σ_n is a closed form as long as $n < \infty$, which is always the case. For the current example, we have $n \leq 3$ as can be observed in (205).

C Approximate methods using truncated series

In the previous section, we have devised closed-form expressions for the transition matrix of complex, IMU-driven dynamic systems written in their linearized, error-state form $\dot{\delta\mathbf{x}} =$

$\mathbf{A}\delta\mathbf{x}$. Closed form expressions may always be of interest, but it is unclear up to which point we should be worried about high order errors and their impact on the performance of real algorithms. This remark is particularly relevant in systems where IMU integration errors are observed (and thus compensated for) at relatively high rates, such as visual-inertial or GPS-inertial fusion schemes.

In this section we devise methods for approximating the transition matrix. They start from the same assumption that the transition matrix can be expressed as a Taylor series, and then truncate these series at the most significant terms. This truncation can be done system-wise, or block-wise.

C.1 System-wise truncation

C.1.1 First-order truncation: the finite differences method

A typical, widely used integration method for systems of the type

$$\dot{\mathbf{x}} = f(t, \mathbf{x})$$

is based on the finite-differences method for the computation of the derivative, *i.e.*,

$$\dot{\mathbf{x}} \triangleq \lim_{\delta t \to 0} \frac{\mathbf{x}(t+\delta t) - \mathbf{x}(t)}{\delta t} \approx \frac{\mathbf{x}_{n+1} - \mathbf{x}_n}{\Delta t}. \tag{212}$$

This leads immediately to

$$\mathbf{x}_{n+1} \approx \mathbf{x}_n + \Delta t\, f(t_n, \mathbf{x}_n) \ , \tag{213}$$

which is precisely the Euler method. Linearization of the function $f()$ at the beginning of the integration interval leads to

$$\mathbf{x}_{n+1} \approx \mathbf{x}_n + \Delta t\, \mathbf{A}\, \mathbf{x}_n \ , \tag{214a}$$

where $\mathbf{A} \triangleq \frac{\partial f}{\partial \mathbf{x}}(t_n, \mathbf{x}_n)$ is a Jacobian matrix. This is strictly equivalent to writing the exponential solution to the linearized differential equation and truncating the series at the linear term (*i.e.*, the following relation is identical to the previous one),

$$\mathbf{x}_{n+1} = e^{\mathbf{A}\Delta t}\mathbf{x}_n \approx (\mathbf{I} + \Delta t\, \mathbf{A})\, \mathbf{x}_n \ . \tag{214b}$$

This means that the Euler method (App. A.1), the finite-differences method, and the first-order system-wise Taylor truncation method, are all the same. We get the approximate transition matrix,

$$\boxed{\Phi \approx \mathbf{I} + \Delta t \mathbf{A}} \ . \tag{215}$$

For the simplified IMU example of Section B.2, the finite-differences method results in the approximated transition matrix

$$\Phi \approx \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & 0 \\ 0 & \mathbf{I} & -\mathbf{R}\,[\mathbf{a}]_\times \Delta t \\ 0 & 0 & \mathbf{I} - [\boldsymbol{\omega}\Delta t]_\times \end{bmatrix} \ . \tag{216}$$

However, we already know from Section B.1 that the rotational term has a compact, closed-form solution, $\Phi(\boldsymbol{\theta}, \boldsymbol{\theta}) = \mathbf{R}(\boldsymbol{\omega}\Delta t)^\top$. It is convenient to re-write the transition matrix according to it,

$$\Phi \approx \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & 0 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t \\ 0 & 0 & \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top \end{bmatrix} . \tag{217}$$

C.1.2 N-order truncation

Truncating at higher orders will increase the precision of the approximated transition matrix. A particularly interesting order of truncation is that which exploits the sparsity of the result to its maximum. In other words, the order after which no new non-zero terms appear.

For the simplified IMU example of Section B.2, this order is 2, resulting in

$$\mathbf{\Phi} \approx \mathbf{I} + \mathbf{A}\Delta t + \frac{1}{2}\mathbf{A}^2\Delta t^2 = \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & -\frac{1}{2}\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t^2 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}\right]_\times (\mathbf{I} - \frac{1}{2}\left[\boldsymbol{\omega}\right]_\times \Delta t)\Delta t \\ 0 & 0 & \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top \end{bmatrix} . \tag{218}$$

In the full IMU example of Section B.3, the is order 3, resulting in

$$\mathbf{\Phi} \approx \mathbf{I} + \mathbf{A}\Delta t + \frac{1}{2}\mathbf{A}^2\Delta t^2 + \frac{1}{6}\mathbf{A}^3\Delta t^3 , \tag{219}$$

whose full form is not given here for space reasons. The reader may consult the expressions of $\mathbf{A}$, $\mathbf{A}^2$ and $\mathbf{A}^3$ in Section B.3.

C.2 Block-wise truncation

A fairly good approximation to the closed forms previously explained results from truncating the Taylor series of each block of the transition matrix at the first significant term. That is, instead of truncating the series in full powers of $\mathbf{A}$, as we have just made above, we regard each block individually. Therefore, truncation needs to be analyzed in a per-block basis. We explore it with two examples.

For the simplified IMU example of Section B.2, we had series Σ_1 and Σ_2, which we can truncate as follows

$$\Sigma_1 = \mathbf{I}\Delta t + \frac{1}{2}\Theta_{\boldsymbol{\theta}}\Delta t^2 + \cdots \approx \mathbf{I}\Delta t \tag{220}$$

$$\Sigma_2 = \frac{1}{2}\mathbf{I}\Delta t^2 + \frac{1}{3!}\Theta_{\boldsymbol{\theta}}\Delta t^3 + \cdots \approx \frac{1}{2}\mathbf{I}\Delta t^2 . \tag{221}$$

This leads to the approximate transition matrix

$$\Phi \approx \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & -\frac{1}{2}\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t^2 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t \\ 0 & 0 & \mathbf{R}(\boldsymbol{\omega}\Delta t)^\top \end{bmatrix} , \tag{222}$$

which is more accurate than the one in the system-wide first-order truncation above (because of the upper-right term which has now appeared), yet it remains easy to obtain and compute, especially when compared to the closed forms developed in Section B. Again, observe that we have taken the closed-form for the lowest term, *i.e.*, $\Phi(\boldsymbol{\theta}, \boldsymbol{\theta}) = \mathbf{R}(\boldsymbol{\omega}\Delta t)^\top$.

In the general case, it suffices to approximate each Σ_n except Σ_0 by the first term of its series, *i.e.*,

$$\boxed{\Sigma_0 = \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top \ , \qquad \Sigma_{n>0} \approx \frac{1}{n!}\mathbf{I}\Delta t^n} \ . \tag{223}$$

For the full IMU example, feeding the previous Σ_n into (205) yields the approximated transition matrix,

$$\Phi \approx \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & -\frac{1}{2}\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t^2 & -\frac{1}{2}\mathbf{R}\Delta t^2 & \frac{1}{3!}\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t^3 & \frac{1}{2}\mathbf{I}\Delta t^2 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t & -\mathbf{R}\Delta t & \frac{1}{2}\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t^2 & \mathbf{I}\Delta t \\ 0 & 0 & \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top & 0 & -\mathbf{I}\Delta t & 0 \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} \end{bmatrix} \tag{224}$$

with (see (60))

$$\mathbf{a} = \mathbf{a}_\mathcal{S} - \mathbf{a}_b \ , \quad \boldsymbol{\omega} = \boldsymbol{\omega}_\mathcal{S} - \boldsymbol{\omega}_b \ , \quad \mathbf{R} = \mathbf{R}(\mathbf{q}) \ ,$$

and where we have substituted the matrix blocks by their appropriate values (see also (60)),

$$\mathbf{P_v} = \mathbf{I} \ , \quad \mathbf{V}_{\boldsymbol{\theta}} = -\mathbf{R}\left[\mathbf{a}\right]_\times \ , \quad \mathbf{V_a} = -\mathbf{R} \ , \quad \mathbf{V_g} = \mathbf{I} \ , \quad \Theta_{\boldsymbol{\theta}} = -\left[\boldsymbol{\omega}\right]_\times \ , \quad \Theta_{\boldsymbol{\omega}} = -\mathbf{I}$$

A slight simplification of this method is to limit each block in the matrix to a certain maximum order n. For $n = 1$ we have,

$$\Phi \approx \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & 0 & 0 & 0 & 0 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t & -\mathbf{R}\Delta t & 0 & \mathbf{I}\Delta t \\ 0 & 0 & \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top & 0 & -\mathbf{I}\Delta t & 0 \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} \end{bmatrix} , \tag{225}$$

which is the Euler method, whereas for $n = 2$,

$$\Phi \approx \begin{bmatrix} \mathbf{I} & \mathbf{I}\Delta t & -\frac{1}{2}\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t^2 & -\frac{1}{2}\mathbf{R}\Delta t^2 & 0 & \frac{1}{2}\mathbf{I}\Delta t^2 \\ 0 & \mathbf{I} & -\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t & -\mathbf{R}\Delta t & \frac{1}{2}\mathbf{R}\left[\mathbf{a}\right]_\times \Delta t^2 & \mathbf{I}\Delta t \\ 0 & 0 & \mathbf{R}\{\boldsymbol{\omega}\Delta t\}^\top & 0 & -\mathbf{I}\Delta t & 0 \\ 0 & 0 & 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{I} \end{bmatrix} . \tag{226}$$

For $n \geq 3$ we have the full form (224).

D The transition matrix via Runge-Kutta integration

Still another way to approximate the transition matrix is to use Runge-Kutta integration. This might be necessary in cases where the dynamic matrix $\mathbf{A}$ cannot be considered constant along the integration interval, *i.e.*,

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) \; . \tag{227}$$

Let us rewrite the following two relations defining the same system in continuous- and discrete-time. They involve the dynamic matrix $\mathbf{A}$ and the transition matrix Φ,

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}(t)\cdot\mathbf{x}(t) &(228)\\ \mathbf{x}(t_n+\tau) &= \Phi(t_n+\tau|t_n)\cdot\mathbf{x}(t_n) \; . &(229) \end{aligned}$$

These equations allow us to develop $\dot{\mathbf{x}}(t_n+\tau)$ in two ways as follows (left and right developments, please note the tiny dots indicating the time-derivatives),

$$\begin{array}{rcl} \overset{\cdot}{\left(\Phi(t_n+\tau|t_n)\mathbf{x}(t_n)\right)} = & \boxed{\dot{\mathbf{x}}(t_n+\tau)} & = \mathbf{A}(t_n+\tau)\mathbf{x}(t_n+\tau) \\ \dot{\Phi}(t_n+\tau|t_n)\mathbf{x}(t_n) + \Phi(t_n+\tau|t_n)\dot{\mathbf{x}}(t_n) = & & = \mathbf{A}(t_n+\tau)\Phi(t_n+\tau|t_n)\mathbf{x}(t_n) \\ \dot{\Phi}(t_n+\tau|t_n)\mathbf{x}(t_n) = & & \end{array}$$

Here, (230) comes from noticing that $\dot{\mathbf{x}}(t_n) = \dot{\mathbf{x}}_n = 0$, because it is a sampled value. Then,

$$\dot{\Phi}(t_n+\tau|t_n) = \mathbf{A}(t_n+\tau)\Phi(t_n+\tau|t_n) \tag{230}$$

which is the same ODE as (228), now applied to the transition matrix instead of the state vector.

We know that, in the cases where $\mathbf{A}$ is considered constant over the integration interval (*i.e.*, $\mathbf{A}(t) = \mathbf{A}(t_n) \triangleq \mathbf{A}$ for $t_n < t < t_n + \Delta t$), the transition matrix can be computed in closed form, $\Phi = e^{\mathbf{A}\Delta t}$, or approximated by truncation of its Taylor series. If $\mathbf{A}(t)$ does not want to be considered constant, then one can attempt numerical integration of (230) using any high-order runge-Kutta method. Mind that, because of the identity $\mathbf{x}(t_n) = \Phi_{t_n|t_n}\mathbf{x}(t_n)$, the transition matrix at the beginning of the interval, $t = t_n$, is always the identity,

$$\Phi_{t_n|t_n} = \mathbf{I} \; . \tag{231}$$

Using RK4 with $f(t, \Phi(t)) = \mathbf{A}(t)\Phi(t)$, we have

$$\Phi \triangleq \Phi(t_n+\Delta t|t_n) = \mathbf{I} + \frac{\Delta t}{6}(\mathbf{K}_1 + 2\mathbf{K}_2 + 2\mathbf{K}_3 + \mathbf{K}_4) \tag{232}$$

with

$$\mathbf{K}_1 = \mathbf{A}(t_n) \tag{233}$$

$$\mathbf{K}_2 = \mathbf{A}\Big(t_n + \frac{1}{2}\Delta t\Big)\Big(\mathbf{I} + \frac{\Delta t}{2}\mathbf{K}_1\Big) \tag{234}$$

$$\mathbf{K}_3 = \mathbf{A}\Big(t_n + \frac{1}{2}\Delta t\Big)\Big(\mathbf{I} + \frac{\Delta t}{2}\mathbf{K}_2\Big) \tag{235}$$

$$\mathbf{K}_4 = \mathbf{A}\Big(t_n + \Delta t\Big)\Big(\mathbf{I} + \Delta t{\cdot}\mathbf{K}_3\Big) \ . \tag{236}$$

D.1 Error-state example

Let us consider the error-state Kalman filter for the non-linear, time-varying system

$$\dot{\mathbf{x}}_t(t) = f(t, \mathbf{x}_t(t), \mathbf{u}(t)) \tag{237}$$

where $\mathbf{x}_t$ denotes the true state, and $\mathbf{u}$ is a control input. This true state is a composition, denoted by $\oplus$, of a nominal state $\mathbf{x}$ and the error state $\delta\mathbf{x}$,

$$\mathbf{x}_t(t) = \mathbf{x}(t) \oplus \delta\mathbf{x}(t) \tag{238}$$

where the error-state dynamics admits a linear form which is time-varying depending on the nominal state and control, *i.e.*,

$$\dot{\delta\mathbf{x}} = \mathbf{A}(\mathbf{x}(t), \mathbf{u}(t)){\cdot}\delta\mathbf{x} \tag{239}$$

that is, the error-state dynamic matrix in (227) has the form $\mathbf{A}(t) = \mathbf{A}(\mathbf{x}(t), \mathbf{u}(t))$. The dynamics of the error-state transition matrix can be written,

$$\dot{\Phi}(t_n + \tau|t_n) = \mathbf{A}(\mathbf{x}(t), \mathbf{u}(t)){\cdot}\Phi(t_n + \tau|t_n) \ . \tag{240}$$

In order to RK-integrate this equation, we need the values of $\mathbf{x}(t)$ and $\mathbf{u}(t)$ at the RK evaluation points, which for RK4 are $\{t_n, t_n + \Delta t/2, t_n + \Delta t\}$. Starting by the easy ones, the control inputs $\mathbf{u}(t)$ at the evaluation points can be obtained by linear interpolation of the current and last measurements,

$$\mathbf{u}(t_n) = \mathbf{u}_n \tag{241}$$

$$\mathbf{u}(t_n + \Delta t/2) = \frac{\mathbf{u}_n + \mathbf{u}_{n+1}}{2} \tag{242}$$

$$\mathbf{u}(t_n + \Delta t) = \mathbf{u}_{n+1} \tag{243}$$

The nominal state dynamics should be integrated using the best integration practicable. For example, using RK4 integration we have,

$$\begin{aligned}
\mathbf{k}_1 &= f(\mathbf{x}_n, \mathbf{u}_n) \\
\mathbf{k}_2 &= f(\mathbf{x}_n + \frac{\Delta t}{2}\mathbf{k}_1, \frac{\mathbf{u}_n + \mathbf{u}_{n+1}}{2}) \\
\mathbf{k}_3 &= f(\mathbf{x}_n + \frac{\Delta t}{2}\mathbf{k}_2, \frac{\mathbf{u}_n + \mathbf{u}_{n+1}}{2}) \\
\mathbf{k}_4 &= f(\mathbf{x}_n + \Delta t\mathbf{k}_3, \mathbf{u}_{n+1}) \\
\mathbf{k} &= (\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4)/6 \ ,
\end{aligned}$$

which gives us the estimates at the evaluation points,

$$\mathbf{x}(t_n) = \mathbf{x}_n \tag{244}$$

$$\mathbf{x}(t_n + \Delta t/2) = \mathbf{x}_n + \frac{\Delta t}{2}\mathbf{k} \tag{245}$$

$$\mathbf{x}(t_n + \Delta t) = \mathbf{x}_n + \Delta t\mathbf{k}\ , \tag{246}$$

where we notice that $\mathbf{x}(t_n + \Delta t/2) = \frac{\mathbf{x}_n + \mathbf{x}_{n+1}}{2}$, the same linear interpolation we used for the control. This should not be surprising given the linear nature of the RK update.

Whichever the way we obtained the nominal state values, we can now compute the RK4 matrices for the integration of the transition matrix,

$$\begin{aligned}
\mathbf{K}_1 &= \mathbf{A}(\mathbf{x}_n, \mathbf{u}_n) \\
\mathbf{K}_2 &= \mathbf{A}\Big(\mathbf{x}_n + \frac{\Delta t}{2}\mathbf{k}, \frac{\mathbf{u}_n + \mathbf{u}_{n+1}}{2}\Big)\Big(\mathbf{I} + \frac{\Delta t}{2}\mathbf{K}_1\Big) \\
\mathbf{K}_3 &= \mathbf{A}\Big(\mathbf{x}_n + \frac{\Delta t}{2}\mathbf{k}, \frac{\mathbf{u}_n + \mathbf{u}_{n+1}}{2}\Big)\Big(\mathbf{I} + \frac{\Delta t}{2}\mathbf{K}_2\Big) \\
\mathbf{K}_4 &= \mathbf{A}\Big(\mathbf{x}_n + \Delta t\mathbf{k}, \mathbf{u}_{n+1}\Big)\Big(\mathbf{I} + \Delta t\mathbf{K}_3\Big) \\
\mathbf{K} &= (\mathbf{K}_1 + 2\mathbf{K}_2 + 2\mathbf{K}_3 + \mathbf{K}_4)/6
\end{aligned}$$

which finally lead to,

$$\boxed{\Phi \triangleq \Phi_{t_n+\Delta t|t_n} = \mathbf{I} + \Delta t\mathbf{K}} \tag{247}$$

E Integration of random noise and perturbations

We aim now at giving appropriate methods for the integration of random variables within dynamic systems. Of course, we cannot integrate unknown values, but we can integrate their variances and covariances for the sake of uncertainty propagation. This is needed in order to establish the covariances matrices in estimators for systems that are of continuous nature (and specified in continuous time) but estimated in a discrete manner.

Consider the continuous-time dynamic system,

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{w}) \tag{248}$$

where $\mathbf{x}$ is the state vector, $\mathbf{u}$ is a vector of control signals containing noise $\tilde{\mathbf{u}}$, so that the control measurements are $\mathbf{u}_S = \mathbf{u} + \tilde{\mathbf{u}}$, and $\mathbf{w}$ is a vector of random perturbations. Both noise and perturbations are assumed white Gaussian processes, specified by,

$$\tilde{\mathbf{u}} \sim \mathcal{N}\{0, \mathbf{U}^c\} \quad , \quad \mathbf{w}^c \sim \mathcal{N}\{0, \mathbf{W}^c\}\ , \tag{249}$$

where the super-index $\bullet^c$ indicates a continuous-time uncertainty specification.

There exists an important difference between the natures of the noise levels in the control signals, $\tilde{\mathbf{u}}$, and the random perturbations, $\mathbf{w}$:

- On discretization, the control signals are sampled at the time instants $n\Delta t$, having $\mathbf{u}_{S,n} \triangleq \mathbf{u}_S(n\Delta t) = \mathbf{u}(n\Delta t) + \tilde{\mathbf{u}}(n\Delta t)$. The measured part is obviously considered constant over the integration interval, *i.e.*, $\mathbf{u}_S(t) = \mathbf{u}_{S,n}$, and therefore the noise level at the sampling time $n\Delta t$ is also held constant,

$$\tilde{\mathbf{u}}(t) = \tilde{\mathbf{u}}(n\Delta t) = \tilde{\mathbf{u}}_n, \quad n\Delta t < t < (n+1)\Delta t \ . \tag{250}$$

- The perturbations $\mathbf{w}$ are never sampled.
- As a consequence, the integration over Δt of these two stochastic processes differs.

The continuous-time error-state dynamics can be linearized to

$$\dot{\delta\mathbf{x}} = \mathbf{A}\delta\mathbf{x} + \mathbf{B}\tilde{\mathbf{u}} + \mathbf{C}\mathbf{w} \ , \tag{251}$$

with

$$\mathbf{A} \triangleq \left.\frac{\partial f}{\partial \delta\mathbf{x}}\right|_{\mathbf{x},\mathbf{u}_S} \quad , \quad \mathbf{B} \triangleq \left.\frac{\partial f}{\partial \tilde{\mathbf{u}}}\right|_{\mathbf{x},\mathbf{u}_S} \quad , \quad \mathbf{C} \triangleq \left.\frac{\partial f}{\partial \mathbf{w}}\right|_{\mathbf{x},\mathbf{u}_S} \quad , \tag{252}$$

and integrated over the sampling period Δt, giving,

$$\begin{aligned} \delta\mathbf{x}_{n+1} &= \delta\mathbf{x}_n + \int_{n\Delta t}^{(n+1)\Delta t} \left(\mathbf{A}\delta\mathbf{x}(\tau) + \mathbf{B}\tilde{\mathbf{u}}(\tau) + \mathbf{C}\mathbf{w}^c(\tau)\right) d\tau & (253) \\ &= \delta\mathbf{x}_n + \int_{n\Delta t}^{(n+1)\Delta t} \mathbf{A}\delta\mathbf{x}(\tau) d\tau + \int_{n\Delta t}^{(n+1)\Delta t} \mathbf{B}\tilde{\mathbf{u}}(\tau) d\tau + \int_{n\Delta t}^{(n+1)\Delta t} \mathbf{C}\mathbf{w}^c(\tau) d\tau & (254) \end{aligned}$$

which has three terms of very different nature. They can be integrated as follows:

1. From App. B we know that the dynamic part is integrated giving the transition matrix,

$$\delta\mathbf{x}_n + \int_{n\Delta t}^{(n+1)\Delta t} \mathbf{A}\delta\mathbf{x}(\tau) d\tau = \Phi\cdot\delta\mathbf{x}_n \tag{255}$$

   where $\Phi = e^{\mathbf{A}\Delta t}$ can be computed in closed-form or approximated at different levels of accuracy.

2. From (250) we have

$$\int_{n\Delta t}^{(n+1)\Delta t} \mathbf{B}\tilde{\mathbf{u}}(\tau) d\tau = \mathbf{B}\Delta t\tilde{\mathbf{u}}_n \tag{256}$$

   which means that the measurement noise, once sampled, is integrated in a deterministic manner because its behavior inside the integration interval is known.

3. From Probability Theory we know that the integration of continuous white Gaussian noise over a period Δt produces a discrete white Gaussian impulse $\mathbf{w}_n$ described by

$$\mathbf{w}_n \triangleq \int_{n\Delta t}^{(n+1)\Delta t} \mathbf{w}(\tau) d\tau \quad , \quad \mathbf{w}_n \sim \mathcal{N}\{0, \mathbf{W}\} \quad , \quad \text{with } \mathbf{W} = \mathbf{W}^c\Delta t \tag{257}$$

Table 3: Effect of integration on system and covariances matrices.

Description	Continuous time t	Discrete time $n\Delta t$
state	$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{w})$	$\mathbf{x}_{n+1} = g(\mathbf{x}_n, \mathbf{u}_n, \mathbf{w}_n)$
error-state	$\dot{\delta\mathbf{x}} = \mathbf{A}\delta\mathbf{x} + \mathbf{B}\tilde{\mathbf{u}} + \mathbf{C}\mathbf{w}$	$\delta\mathbf{x}_{n+1} = \mathbf{F}_\mathbf{x}\delta\mathbf{x}_n + \mathbf{F}_\mathbf{u}\tilde{\mathbf{u}}_n + \mathbf{F}_\mathbf{w}\mathbf{w}_n$
system matrix	$\mathbf{A}$	$\mathbf{F}_\mathbf{x} = \Phi = e^{\mathbf{A}\Delta t}$
control matrix	$\mathbf{B}$	$\mathbf{F}_\mathbf{u} = \mathbf{B}\Delta t$
perturbation matrix	$\mathbf{C}$	$\mathbf{F}_\mathbf{w} = \mathbf{C}$
control covariance	$\mathbf{U}^c$	$\mathbf{U} = \mathbf{U}^c$
perturbation covariance	$\mathbf{W}^c$	$\mathbf{W} = \mathbf{W}^c\Delta t$

> We obsereve that, contrary to the measurement noise just above, the perturbation does not have a deterministic behavior inside the integration interval, and hence it must be integrated stochastically.

Therefore, the discrete-time, error-state dynamic system can be written as

$$\delta\mathbf{x}_{n+1} = \mathbf{F}_\mathbf{x}\delta\mathbf{x}_n + \mathbf{F}_\mathbf{u}\tilde{\mathbf{u}}_n + \mathbf{F}_\mathbf{w}\mathbf{w}_n \tag{258}$$

with transition, control and perturbation matrices given by

$$\mathbf{F}_\mathbf{x} = \Phi = e^{\mathbf{A}\Delta t} \quad , \quad \mathbf{F}_\mathbf{u} = \mathbf{B}\Delta t \quad , \quad \mathbf{F}_\mathbf{w} = \mathbf{C} \quad , \tag{259}$$

where it is illustrative to appreciate the different contributions of the step time Δt in the integrated values. The noise and perturbation levels are defined by

$$\tilde{\mathbf{u}}_n \sim \mathcal{N}\{0, \mathbf{U}\} \quad , \quad \mathbf{w}_n \sim \mathcal{N}\{0, \mathbf{W}\} \tag{260}$$

with

$$\mathbf{U} = \mathbf{U}^c \quad , \quad \mathbf{W} = \mathbf{W}^c\Delta t \ . \tag{261}$$

These results are summarized in Table 3. The prediction stage of an EKF would propagate the error state's mean and covariances matrix according to

$$\hat{\delta\mathbf{x}}_{n+1} = \mathbf{F}_\mathbf{x}\hat{\delta\mathbf{x}}_n \tag{262}$$

$$\begin{aligned} \mathbf{P}_{n+1} &= \mathbf{F}_\mathbf{x}\mathbf{P}_n\mathbf{F}_\mathbf{x}^\top + \mathbf{F}_\mathbf{u}\mathbf{U}\mathbf{F}_\mathbf{u}^\top + \mathbf{F}_\mathbf{w}\mathbf{W}\mathbf{F}_\mathbf{w}^\top \\ &= e^{\mathbf{A}\Delta t}\mathbf{P}_n(e^{\mathbf{A}\Delta t})^\top + \Delta t^2\mathbf{B}\mathbf{U}^c\mathbf{B}^\top + \Delta t\mathbf{C}\mathbf{W}^c\mathbf{C}^\top \end{aligned} \tag{263}$$

It is important here to observe, again, the different effects of the integration time, Δt, on the three terms of the covariance update (263).

E.1 Noise and perturbation impulses

One is oftentimes confronted (for example when reusing existing code or when interpreting other authors' documents) with EKF prediction equations of a simpler form than those that we used here, namely,

$$\mathbf{P}_{n+1} = \mathbf{F_x}\mathbf{P}_n\mathbf{F_x}^\top + \mathbf{Q} \ . \tag{264}$$

This corresponds to the general discrete-time dynamic system,

$$\delta\mathbf{x}_{n+1} = \mathbf{F_x}\delta\mathbf{x}_n + \mathbf{i} \tag{265}$$

where

$$\mathbf{i} \sim \mathcal{N}\{0, \mathbf{Q}\} \tag{266}$$

is a vector of random (white, Gaussian) impulses that are directly added to the state vector at time t_{n+1}. The matrix $\mathbf{Q}$ is simply considered the impulses covariances matrix. From what we have seen, we should compute this covariances matrix as follows,

$$\mathbf{Q} = \Delta t^2\mathbf{B}\mathbf{U}^c\mathbf{B}^\top + \Delta t\mathbf{C}\mathbf{W}^c\mathbf{C}^\top \ . \tag{267}$$

In the case where the impulses do not affect the full state, as it is often the case, the matrix $\mathbf{Q}$ is not full-diagonal and may contain a significant amount of zeros. One can then write the equivalent form

$$\delta\mathbf{x}_{n+1} = \mathbf{F_x}\cdot\delta\mathbf{x}_n + \mathbf{F_i}\cdot\mathbf{i} \tag{268}$$

with

$$\mathbf{i} \sim \mathcal{N}\{0, \mathbf{Q_i}\} \ , \tag{269}$$

where the matrix $\mathbf{F_i}$ simply maps each individual impulse to the part of the state vector it affects to. The associated covariance $\mathbf{Q_i}$ is then smaller and full-diagonal. Please refer to the next section for an example. In such case the ESKF time-update becomes

$$\begin{aligned} \hat{\delta\mathbf{x}}_{n+1} &= \mathbf{F_x}\cdot\hat{\delta\mathbf{x}}_n & (270)\\ \mathbf{P}_{n+1} &= \mathbf{F_x}\mathbf{P}_n\mathbf{F_x}^\top + \mathbf{F_i}\mathbf{Q_i}\mathbf{F_i}^\top \ . & (271) \end{aligned}$$

Obviously, all these forms are equivalent, as it can be seen in the following double identity for the general perturbation $\mathbf{Q}$,

$$\mathbf{F_i}\mathbf{Q_i}\mathbf{F_i}^\top = \boxed{\mathbf{Q}} = \Delta t^2\mathbf{B}\mathbf{U}^c\mathbf{B}^\top + \Delta t\mathbf{C}\mathbf{W}^c\mathbf{C}^\top \ . \tag{272}$$

E.2 Full IMU example

We study the construction of an error-state Kalman filter for an IMU. The error-state system is defined in (60) and involves a nominal state $\mathbf{x}$, an error-state $\delta\mathbf{x}$, a noisy control signal $\mathbf{u}_S = \mathbf{u} + \tilde{\mathbf{u}}$ and a perturbation $\mathbf{w}$, specified by,

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \\ \mathbf{a}_b \\ \boldsymbol{\omega}_b \\ \mathbf{g} \end{bmatrix} \quad , \quad \delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} \\ \delta\mathbf{v} \\ \delta\boldsymbol{\theta} \\ \delta\mathbf{a}_b \\ \delta\boldsymbol{\omega}_b \\ \delta\mathbf{g} \end{bmatrix} \quad , \quad \mathbf{u}_S = \begin{bmatrix} \mathbf{a}_S \\ \boldsymbol{\omega}_S \end{bmatrix} \quad , \quad \tilde{\mathbf{u}} = \begin{bmatrix} \tilde{\mathbf{a}} \\ \tilde{\boldsymbol{\omega}} \end{bmatrix} \quad , \quad \mathbf{w} = \begin{bmatrix} \mathbf{a}_w \\ \boldsymbol{\omega}_w \end{bmatrix} \tag{273}$$

In a model of an IMU like the one we are considering throughout this document, the control noise corresponds to the additive noise in the IMU measurements. The perturbations affect the biases, thus producing their random-walk behavior. The dynamic, control and perturbation matrices are (see (251), (202) and (60)),

$$\mathbf{A} = \begin{bmatrix} 0 & \mathbf{P_v} & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{V}_{\boldsymbol{\theta}} & \mathbf{V_a} & 0 & \mathbf{V_g} \\ 0 & 0 & \Theta_{\boldsymbol{\theta}} & 0 & \Theta_{\boldsymbol{\omega}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad , \quad \mathbf{B} = \begin{bmatrix} 0 & 0 \\ -\mathbf{R} & 0 \\ 0 & -\mathbf{I} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad , \quad \mathbf{C} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \mathbf{I} & 0 \\ 0 & \mathbf{I} \\ 0 & 0 \end{bmatrix} \tag{274}$$

In the regular case of IMUs with accelerometer and gyrometer triplets of the same kind on the three axes, noise and perturbations are isotropic. Their standard deviations are specified as scalars as follows

$$\sigma_{\tilde{\mathbf{a}}}[m/s^2] \quad , \quad \sigma_{\tilde{\boldsymbol{\omega}}}[rad/s] \quad , \quad \sigma_{\mathbf{a}_w}[m/s^2\sqrt{s}] \quad , \quad \sigma_{\boldsymbol{\omega}_w}[rad/s\sqrt{s}] \tag{275}$$

and their covariances matrices are purely diagonal, giving

$$\mathbf{U}^c = \begin{bmatrix} \sigma_{\tilde{\mathbf{a}}}^2\mathbf{I} & 0 \\ 0 & \sigma_{\tilde{\boldsymbol{\omega}}}^2\mathbf{I} \end{bmatrix} \quad , \quad \mathbf{W}^c = \begin{bmatrix} \sigma_{\mathbf{a}_w}^2\mathbf{I} & 0 \\ 0 & \sigma_{\boldsymbol{\omega}_w}^2\mathbf{I} \end{bmatrix} . \tag{276}$$

The system evolves with sampled measures at intervals Δt, following (258–261), where the transition matrix $\mathbf{F_x} = \Phi$ can be computed in a number of ways – see previous appendices.

E.2.1 Noise and perturbation impulses

TODO: *Describe first the general case, and then the isotropic case*

In the case of a perturbation specification in the form of impulses $\mathbf{i}$, we can re-define our system as follows,

$$\delta\mathbf{x}_{n+1} = \mathbf{F_x}(\mathbf{x}_n, \mathbf{u}_S)\cdot\delta\mathbf{x}_n + \mathbf{F_i}\cdot\mathbf{i} \tag{277}$$

with the nominal-state, error-state, control, and impulses vectors defined by,

$$\mathbf{x} = \begin{bmatrix} \mathbf{p} \\ \mathbf{v} \\ \mathbf{q} \\ \mathbf{a}_b \\ \boldsymbol{\omega}_b \\ \mathbf{g} \end{bmatrix} \quad , \quad \delta\mathbf{x} = \begin{bmatrix} \delta\mathbf{p} \\ \delta\mathbf{v} \\ \delta\boldsymbol{\theta} \\ \delta\mathbf{a}_b \\ \delta\boldsymbol{\omega}_b \\ \delta\mathbf{g} \end{bmatrix} \quad , \quad \mathbf{u}_S = \begin{bmatrix} \mathbf{a}_S \\ \boldsymbol{\omega}_S \end{bmatrix} \quad , \quad \mathbf{i} = \begin{bmatrix} \mathbf{v_i} \\ \boldsymbol{\theta}_\mathbf{i} \\ \mathbf{a_i} \\ \boldsymbol{\omega}_\mathbf{i} \end{bmatrix} \quad , \tag{278}$$

the transition and perturbations matrices defined by,

$$\mathbf{F_x} = \Phi = e^{\mathbf{A}\Delta t} \qquad , \qquad \mathbf{F_i} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \mathbf{I} & 0 & 0 & 0 \\ 0 & \mathbf{I} & 0 & 0 \\ 0 & 0 & \mathbf{I} & 0 \\ 0 & 0 & 0 & \mathbf{I} \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad , \tag{279}$$

and the impulses variances specified by

$$\mathbf{i} \sim \mathcal{N}\{0, \mathbf{Q_i}\} \quad , \quad \mathbf{Q_i} = \begin{bmatrix} \sigma_{\tilde{\mathbf{a}}}^2 \Delta t^2 \mathbf{I} & & & 0 \\ & \sigma_{\tilde{\boldsymbol{\omega}}}^2 \Delta t^2 \mathbf{I} & & \\ & & \sigma_{\mathbf{a}_w}^2 \Delta t \mathbf{I} & \\ 0 & & & \sigma_{\boldsymbol{\omega}_w}^2 \Delta t \mathbf{I} \end{bmatrix} . \tag{280}$$

The trivial specification of $\mathbf{F_i}$ may appear surprising given especially that of $\mathbf{B}$ in (274). What happens is that the errors are defined isotropic in $\mathbf{Q_i}$, and therefore $-\mathbf{R}\sigma^2\mathbf{I}(-\mathbf{R})^\top = \sigma^2\mathbf{I}$ and $-\mathbf{I}\sigma^2\mathbf{I}(-\mathbf{I})^\top = \sigma^2\mathbf{I}$, leading to the expression given for $\mathbf{F_i}$. This is not possible when considering non-isotropic IMUs, where a proper Jacobian $\mathbf{F_i} = \begin{bmatrix} \mathbf{B} & \mathbf{C} \end{bmatrix}$ should be used together with a proper specification of $\mathbf{Q_i}$.

We can of course use full-state perturbation impulses,

$$\delta\mathbf{x}_{n+1} = \mathbf{F_x}(\mathbf{x}_n, \mathbf{u}_S) \cdot \delta\mathbf{x}_n + \mathbf{i} \tag{281}$$

with

$$\mathbf{i} = \begin{bmatrix} 0 \\ \mathbf{v_i} \\ \boldsymbol{\theta}_\mathbf{i} \\ \mathbf{a_i} \\ \boldsymbol{\omega}_\mathbf{i} \\ 0 \end{bmatrix} \quad , \quad \mathbf{i} \sim \mathcal{N}\{0, \mathbf{Q}\} \quad , \quad \mathbf{Q} = \begin{bmatrix} 0 & & & & & \\ & \sigma_{\tilde{\mathbf{a}}}^2 \Delta t^2 \mathbf{I} & & & 0 & \\ & & \sigma_{\tilde{\boldsymbol{\omega}}}^2 \Delta t^2 \mathbf{I} & & & \\ & & & \sigma_{\mathbf{a}_w}^2 \Delta t \mathbf{I} & & \\ & 0 & & & \sigma_{\boldsymbol{\omega}_w}^2 \Delta t \mathbf{I} & \\ & & & & & 0 \end{bmatrix} . \tag{282}$$

Bye bye.

References

[Lupton and Sukkarieh(2009)] Lupton T, Sukkarieh S (2009) Efficient integration of inertial observations into visual slam without initialization. In: IEEE/RSJ Int. Conf. on Intelligent Robots and Systems

[Trawny and Roumeliotis(2005)] Trawny N, Roumeliotis SI (2005) Indirect Kalman filter for 3D attitude estimation. Tech. Rep. 2005-002, University of Minnesota, Dept. of Comp. Sci. & Eng., URL http://www-users.cs.umn.edu/~trawny/Publications/Quaternions_Techreport.htm