Highly accurate differentiation of the exponential map and its tangent operator

Juliano Todesco, Olivier Brüls

October 7, 2023

Department of Aerospace and Mechanical Engineering, University of Liège, Belgium

Abstract

Exponential coordinates are widely used in simulation codes for flexible multibody systems based on a Lie group approach. The accurate and efficient evaluation of the exponential map, the tangent operator and its higher order derivatives is crucial. This paper presents a systematic derivation process based on the matrix series expansion of the exponential map and of the tangent operator. This approach is general as it can be applied to any matrix Lie group. For the Lie groups $SO(3)$ and $SE(3)$, the closed form expression of these operators can also be established and is summarized in the paper. It is shown that the closed form of the operators is affected by round-off errors for small rotation amplitudes, whereas the series form is affected by truncation errors at high rotation amplitudes. The computational efficiency of the two approaches is also discussed. A switching strategy between the closed form and the series form is then proposed to obtain an adjustable compromise between accuracy and computational cost.

Keywords: Derivatives, Exponential map, special orthogonal group, special Euclidean group, numerical methods, Lie group
E-mail addresses: Juliano.Todesco@doct.uliege.be (J. Todesco), O.Bruls@uliege.be (O. Brüls).

- Lie group operators for flexible multibody systems are developed
- The exponential map and its derivatives are given in truncated matrix series form
- The ill-conditioning of exponential map derivatives in closed form is analyzed
- Numerical errors and costs are analyzed for the exponential map and derivatives
- A switching strategy between the closed form and the series form is proposed

1 Introduction

Nowadays, Lie group methods are widely used in computational mechanics and, in particular, in multibody dynamics. For example, they offer a consistent treatment either for rotation variables defined on the special orthogonal group $SO(3)$ or for roto-translation variables (i.e., rigid transformation variables) defined on the special Euclidean group $SE(3)$. Time and space discretization methods on Lie group [1–7] are applicable to systems with rigid bodies, kinematic joints, flexible beams, flexible shells and superelements. The Lie group framework solves a number of numerical difficulties related with the treatment of singularities of parameterizations. These methods still use a parameterization of the motion but only locally, e.g., to describe the incremental motion from one time step to the next or to describe relative positions and orientations within a finite element.

An important ingredient of Lie group methods is the algorithm for the computation of the coordinate map and its derivative [3]. The first derivative of the coordinate map is defined at a current point and in a given direction and it can be conveniently represented by the so-called tangent operator, which is a $k \times k$ matrix where k is the intrinsic dimension of the group. Some advanced numerical formulations also require some higher order directional derivatives and gradients of the tangent operator and its inverse. These diverse operators are involved in finite element discretization processes [6], in implicit time integration procedures [5], in sensitivity analysis [8, 9] and in optimization [10, 11].

Actually, several parameterizations can be selected to define the local coordinates involved in Lie group methods [12–16]. Among these possible choices, exponential coordinates are often considered at first because of their many attractive theoretical properties, which can be attributed to the fact that the exponential map captures the fundamental solution of ordinary differential equations on Lie groups. For these reasons, this paper focuses on exponential coordinates and the associated maps. The inverse of the exponential map is the so-called logarithm map.

In the context of matrix Lie groups, the exponential map takes a matrix of the Lie algebra and computes a matrix on the group based on an infinite matrix series expansion. As pointed out in [17], there is a variety of alternative numerical methods to compute the exponential map of an arbitrary matrix. The evaluation of the tangent operator turns out to be more complicated than the exponential map itself, which calls for dedicated algorithms. The complexity further increases for successive derivatives of the tangent operator and its inverse.

This paper thus addresses the efficient and accurate evaluation of the exponential map, the logarithm map and the tangent operator on matrix Lie groups, such as $SO(3)$ and $SE(3)$, as well as the derivatives of the tangent operator and its inverse. Methods for calculating these operators can be roughly divided into three classes.

The first class relies on the derivative of the analytic or closed form expression of the exponential map, which is available for the Lie groups $SO(3)$ and $SE(3)$. For $SO(3)$, the closed form expression of the exponential map is known as the Rodrigues formula. One issue is that the closed formula suffers from a singularity at the origin. This singularity appears when the rotation is zero and it implies some ill conditioning of the exponential map and its derivatives in the neighborhood of the origin. To overcome the singularity, the closed form of the exponential map and its derivatives can be replaced by its limit value when the rotation amplitude is below a threshold, see [18]. The selection of this threshold has a direct impact on the accuracy, which is rarely investigated in the literature. Instead of using the limit value, a Taylor series expansion with 3 or 4 terms can also be exploited [19].

The second class relies on a scalar series formulation, which replaces the trigonometric functions involved in the closed form for $SO(3)$ and $SE(3)$. These expressions have no singularity issue and do not suffer from ill conditioning at the origin. A family of scalar power series was proposed in [20] to provide compact expressions for the exponential map. A similar approach was followed by [21] who proposed a single series family for the calculation of first and second derivatives of the exponential map on $SO(3)$.

These first two classes require some consequent analytical developments for the formulation of each operator, which may become cumbersome considering the variety of operators needed in general purpose code for flexible multibody systems. In order to obtain a more generic formulation of all operators on an arbitrary matrix Lie group, this article studies a third class of methods which is based on truncated matrix series expansions of the exponential map and of the tangent operator [22, 23]. A similar strategy was exploited by [24] to evaluate the nth order directional derivatives of the Gibbs coordinate map on $SO(3)$. However, we did not find any detailed study regarding the matrix series expansion of the derivatives of the exponential map for general matrix Lie group.

Let us remark that a rotation (resp. a rigid body motion) could alternatively be represented as a unit quaternion (resp. as a unit dual quaternion) which also belong to a Lie group [25]. Exponential coordinates and the associated maps are also available on the quaternion group and on the dual quaternion group. Such representations have been successfully exploited, e.g., in [26–28]. However, these groups are non-matricial ones and are therefore not considered in the scope of the present paper.

The paper is organized as follows. We first review the matrix Lie group framework; the exponential map is presented in its series form and the tangent operator is introduced. Secondly, we develop the derivatives of the tangent operator and its inverse using truncated matrix series. The formulae are rather compact and valid for any matrix Lie group. The derivation procedure turns out to be quite systematic. Thirdly, the operators on $SO(3)$ and $SE(3)$ are reviewed. Closed form expressions are presented for the exponential map, the logarithm map, the tangent operator as well as its derivatives. Fourth, the numerical error and the computational cost is analyzed for the closed form and the truncated series form. Then, numerical experiments are performed to investigate the accuracy of these formulae in a wide range of rotation and motion amplitudes. Based on these observations, a switching strategy between the closed form and the series form on $SO(3)$ and $SE(3)$ is proposed in order to obtain the best compromise between accuracy and computational cost. Finally, some conclusions are given.

2 Matrix Lie group framework

This section presents the fundamental mathematical concepts that are at the core of this paper. We start with the definitions of a matrix Lie group and its Lie algebra, then we continue with their adjoint representations and finally we introduce the exponential map, its tangent operator and its inverse. These concepts, which are well-established in the field, are presented in a rather compact way. The reader may find further information in textbooks or reference papers such as [25, 29–31].

A *group* $(\mathcal{G}, \circ)$ is a set $\mathcal{G}$ of elements q with a group operation $\circ$ which satisfies the four following axioms:

- Closure: If $\forall q_1, q_2 \in \mathcal{G}$, then $q_1 \circ q_2 = q_3 \in \mathcal{G}$;
- Associativity: $q_1 \circ (q_2 \circ q_3) = (q_1 \circ q_2) \circ q_3$;
- Neutral element: There exists an element $e \in \mathcal{G}$ such that $q \circ e = e \circ q = q$;
- Inverse element: $\forall q \in \mathcal{G}$ there exists an element $q^{-1} \in \mathcal{G}$ such that $q \circ q^{-1} = q^{-1} \circ q = e$.

A *Lie group* is a group $\mathcal{G}$ which is a differential manifold and which satisfies the following smoothness requirements for the group operation and the inverse map:

- Smoothness: The maps $(q_1, q_2) \mapsto q_1 \circ q_2$ and $q \mapsto q^{-1}$ are smooth functions.

In this work, we restrict our considerations to *matrix Lie groups* for which the elements are invertible $n \times n$ matrices $\mathbf{Q}$ and the group operation is the usual matrix product written as $\mathbf{Q}_1\mathbf{Q}_2 = \mathbf{Q}_3$. As it is also a manifold, a matrix Lie group has an intrinsic dimension which shall be denoted k and which is generally different from n. The neutral element is (simply) the identity matrix $\mathbf{I}$. Given an element $\mathbf{Y} \in \mathcal{G}$, the group operation leads to the definition of the left and right translation maps.

- *Left translation map*: $L_{\mathbf{Y}}(\mathbf{Q}) : \mathcal{G} \to \mathcal{G}, \quad \mathbf{Q} \longmapsto \mathbf{YQ}$.
- *Right translation map*: $R_{\mathbf{Y}}(\mathbf{Q}) : \mathcal{G} \to \mathcal{G}, \quad \mathbf{Q} \longmapsto \mathbf{QY}$.

Let us differentiate these maps to obtain important relations between the *tangent spaces* of the group. Consider the curve $\mathbf{Q}(a) \in \mathcal{G}$ parametrized by a scalar variable $a \in \mathbb{R}$ and an element $\mathbf{Y}$ which does not depend on this variable a. Let $d_a(\bullet)$ denotes the derivative with respect to a. Then $d_a(\mathbf{Q})$ belongs to $T_{\mathbf{Q}}\mathcal{G}$, the tangent space at element $\mathbf{Q} \in \mathcal{G}$, and we have

$$d_a(L_{\mathbf{Y}}(\mathbf{Q}(a))) = d_a(\mathbf{Y}\mathbf{Q}(a)) = \mathbf{Y}d_a(\mathbf{Q}(a)) \tag{1}$$

$$d_a(R_{\mathbf{Y}}(\mathbf{Q}(a))) = d_a(\mathbf{Q}(a)\mathbf{Y}) = d_a(\mathbf{Q}(a))\mathbf{Y} \tag{2}$$

which motivates the definition of the differentiable and invertible maps

$$L_{\mathbf{Y}*} : T_{\mathbf{Q}}\mathcal{G} \to T_{\mathbf{YQ}}\mathcal{G}, \quad d_a(\mathbf{Q}(a)) \mapsto L_{\mathbf{Y}*}\left(d_a(\mathbf{Q}(a))\right) = \mathbf{Y}d_a(\mathbf{Q}(a)) \tag{3}$$

$$R_{\mathbf{Y}*} : T_{\mathbf{Q}}\mathcal{G} \to T_{\mathbf{QY}}\mathcal{G}, \quad d_a(\mathbf{Q}(a)) \mapsto R_{\mathbf{Y}*}\left(d_a(\mathbf{Q}(a))\right) = d_a(\mathbf{Q}(a))\mathbf{Y} \tag{4}$$

Eq. (3) (resp. Eq. (4)) defines a diffeomorphism between $T_{\mathbf{Y}}\mathcal{G}$ and $T_{\mathbf{YQ}}\mathcal{G}$ (resp. and $T_{\mathbf{QY}}\mathcal{G}$). In the particular case $\mathbf{Y} = \mathbf{Q}^{-1}$, Eqs. (3) and (4) lead to the definition of $\widetilde{\mathbf{a}_L}$ and $\widetilde{\mathbf{a}_R}$

$$\widetilde{\mathbf{a}_L} = \mathbf{Q}^{-1}d_a(\mathbf{Q}(a)) \quad \Leftrightarrow \quad d_a(\mathbf{Q}(a)) = \mathbf{Q}\widetilde{\mathbf{a}_L} \tag{5}$$

$$\widetilde{\mathbf{a}_R} = d_a(\mathbf{Q}(a))\mathbf{Q}^{-1} \quad \Leftrightarrow \quad d_a(\mathbf{Q}(a)) = \widetilde{\mathbf{a}_R}\mathbf{Q} \tag{6}$$

where $\widetilde{\mathbf{a}_L}, \widetilde{\mathbf{a}_R} \in T_{\mathbf{I}}\mathcal{G}$. The maps $L_{\mathbf{Q}^{-1}*} : T_{\mathbf{Q}}\mathcal{G} \to T_{\mathbf{I}}\mathcal{G}$ and $R_{\mathbf{Q}^{-1}*} : T_{\mathbf{Q}}\mathcal{G} \to T_{\mathbf{I}}\mathcal{G}$ are differentiable and invertible maps, which relate the tangent space at any element $\mathbf{Q} \in \mathcal{G}$ to the tangent space at the identity element $\mathbf{I}$. This tangent space at the identity of a matrix Lie group is called the *Lie algebra*, denoted by $T_{\mathbf{I}}\mathcal{G} \equiv \mathfrak{g}$. The Lie algebra has a vector space structure and is isomorphic to $\mathbb{R}^k$ through the invertible linear map:

$$\widetilde{(\bullet)} : \mathbb{R}^k \to \mathfrak{g}, \quad \mathbf{x} \mapsto \tilde{\mathbf{x}} \tag{7}$$

Let us advance the study by analyzing the derivatives of the right and left translation maps, which will result in the definition of the Lie bracket $[\bullet, \bullet]$, of the adjoint representation and its derivative. From Eqs. (5) and (6), we obtain

$$\widetilde{\mathbf{a}_R} = \mathbf{Q}\widetilde{\mathbf{a}_L}\mathbf{Q}^{-1} \tag{8}$$

so that $\mathbf{Q}\widetilde{\mathbf{a}_L}\mathbf{Q}^{-1} \in \mathfrak{g}$ is an element of the Lie algebra. Following the result of Eq. (8), we can define the *adjoint representation of the Lie group* $\mathrm{Ad}_{\mathbf{Q}}$ such that, for any $\mathbf{Q} \in \mathcal{G}$,

$$\mathrm{Ad}_{\mathbf{Q}} : \mathfrak{g} \to \mathfrak{g}, \quad \tilde{\mathbf{a}} \mapsto \mathrm{Ad}_{\mathbf{Q}}(\tilde{\mathbf{a}}) = \mathbf{Q}\tilde{\mathbf{a}}\mathbf{Q}^{-1} \tag{9}$$

$\mathrm{Ad}_{\mathbf{Q}}$ can be interpreted as a linear map in the Lie algebra $\mathfrak{g}$, which is called the *adjoint map*. With a slight abuse of notations, since $\mathbb{R}^k$ is isomorphic to the Lie algebra, the notation $\mathrm{Ad}_{\mathbf{Q}}$ is also used to represent the adjoint map from $\mathbb{R}^k$ to $\mathbb{R}^k$, i.e.,

$$\mathrm{Ad}_{\mathbf{Q}} : \mathbb{R}^k \to \mathbb{R}^k, \quad \mathbf{a} \mapsto \mathrm{Ad}_{\mathbf{Q}}\mathbf{a} \tag{10}$$

In this case, $\mathrm{Ad}_{\mathbf{Q}}$ is interpreted as a $k \times k$ matrix and $\mathrm{Ad}_{\mathbf{Q}}\mathbf{a}$ is the usual matrix–vector product.

Let now $\mathbf{Q}(a, b) \in \mathcal{G}$ be parametrized by two scalar variables $a, b \in \mathbb{R}$. Following the results of the derivative of the left translation map in Eq. (5), the derivatives of $\mathbf{Q}(a, b)$ with respect to a and b are respectively given by $d_a(\mathbf{Q}) = \mathbf{Q}\tilde{\mathbf{a}}$ and $d_b(\mathbf{Q}) = \mathbf{Q}\tilde{\mathbf{b}}$, where $\tilde{\mathbf{a}}$ and $\tilde{\mathbf{b}}$ belong to the Lie algebra. The commutativity of the cross derivatives is expressed as $d_a(d_b(\mathbf{Q})) = d_b(d_a(\mathbf{Q}))$, which implies:

$$d_a(\mathbf{Q}\tilde{\mathbf{b}}) = d_b(\mathbf{Q}\tilde{\mathbf{a}}) \quad \Leftrightarrow \quad \mathbf{Q}\tilde{\mathbf{a}}\tilde{\mathbf{b}} + \mathbf{Q}d_a(\tilde{\mathbf{b}}) = \mathbf{Q}\tilde{\mathbf{b}}\tilde{\mathbf{a}} + \mathbf{Q}d_b(\tilde{\mathbf{a}}) \quad \Leftrightarrow \quad d_b(\tilde{\mathbf{a}}) - d_a(\tilde{\mathbf{b}}) = \tilde{\mathbf{a}}\tilde{\mathbf{b}} - \tilde{\mathbf{b}}\tilde{\mathbf{a}} \tag{11}$$

In general, $d_b(\tilde{\mathbf{a}}) - d_a(\tilde{\mathbf{b}}) \neq \mathbf{0}$, i.e., second derivatives on a Lie group do not commute. These results motivate the definition of the *Lie bracket operator* (or matrix commutator):

$$\mathrm{ad}_{\tilde{\mathbf{a}}}(\tilde{\mathbf{b}}) \equiv [\bullet, \bullet] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}, \quad \tilde{\mathbf{a}}, \tilde{\mathbf{b}} \mapsto [\tilde{\mathbf{a}}, \tilde{\mathbf{b}}] = \tilde{\mathbf{a}}\tilde{\mathbf{b}} - \tilde{\mathbf{b}}\tilde{\mathbf{a}} \tag{12}$$

The Lie bracket satisfies the following axioms [32]:

- Bilinearity: $[a\tilde{\mathbf{x}} + b\tilde{\mathbf{y}}, \tilde{\mathbf{z}}] = a[\tilde{\mathbf{x}}, \tilde{\mathbf{z}}] + b[\tilde{\mathbf{y}}, \tilde{\mathbf{z}}]$ and $[\tilde{\mathbf{z}}, a\tilde{\mathbf{x}} + b\tilde{\mathbf{y}}] = a[\tilde{\mathbf{z}}, \tilde{\mathbf{x}}] + b[\tilde{\mathbf{z}}, \tilde{\mathbf{y}}]$ for all scalars a, b and all elements $\tilde{\mathbf{x}}, \tilde{\mathbf{y}}, \tilde{\mathbf{z}} \in \mathfrak{g}$;
- Skew-symmetry, $[\tilde{\mathbf{x}}, \tilde{\mathbf{y}}] = -[\tilde{\mathbf{y}}, \tilde{\mathbf{x}}]$ for all $\tilde{\mathbf{x}}, \tilde{\mathbf{y}} \in \mathfrak{g}$. $[\bullet, \bullet]$ is skew symmetric;
- Alternativity, $[\tilde{\mathbf{x}}, \tilde{\mathbf{x}}] = \mathbf{0}$ for all $\tilde{\mathbf{x}} \in \mathfrak{g}$;
- Jacobi identity, $[\tilde{\mathbf{x}}, [\tilde{\mathbf{y}}, \tilde{\mathbf{z}}]] + [\tilde{\mathbf{y}}, [\tilde{\mathbf{z}}, \tilde{\mathbf{x}}]] + [\tilde{\mathbf{z}}, [\tilde{\mathbf{x}}, \tilde{\mathbf{y}}]] = \mathbf{0}$ for all $\tilde{\mathbf{x}}, \tilde{\mathbf{y}}, \tilde{\mathbf{z}} \in \mathfrak{g}$.

The map $\mathrm{ad}_{\tilde{\mathbf{a}}}$ is the *adjoint map* or the *adjoint representation of the Lie algebra* [33]. Although $\mathrm{ad}_{\tilde{\mathbf{a}}}(\tilde{\mathbf{b}})$ is equivalent to $[\tilde{\mathbf{a}}, \tilde{\mathbf{b}}]$, the alternative "ad" notation can be useful. For example, the expression $[\tilde{\mathbf{a}}, [\tilde{\mathbf{a}}, [\tilde{\mathbf{a}}, [\tilde{\mathbf{a}}, \tilde{\mathbf{b}}]]]]$ can be replaced by the more compact notation $\mathrm{ad}^4_{\tilde{\mathbf{a}}}(\tilde{\mathbf{b}})$.

Since the commutator is linear with respect to both arguments, with a slight abuse of notations, the Lie bracket can be written in terms of vectors in $\mathbb{R}^k$:

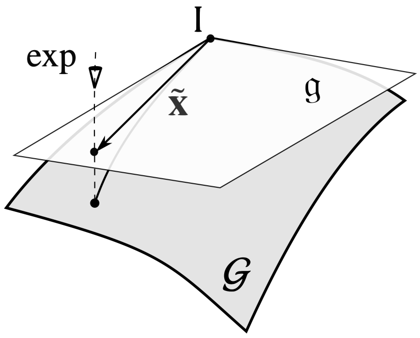
$$\mathrm{ad}_{\mathbf{a}}\mathbf{b} = d_b(\mathbf{a}) - d_a(\mathbf{b}) \tag{13}$$

In this case, $\mathrm{ad}_{\mathbf{a}}$ can be interpreted as a $k \times k$ matrix and $\mathrm{ad}_{\mathbf{a}}\mathbf{b}$ is the usual matrix–vector product. For notational convenience, the "*hat*" operator $\widehat{(\bullet)}$ is introduced as

$$\hat{\mathbf{a}} \equiv \mathrm{ad}_{\mathbf{a}} \tag{14}$$

Since the Lie bracket is bilinear, hat is a linear operator which maps a $k \times 1$ vector $\mathbf{a}$ into a $k \times k$ matrix $\hat{\mathbf{a}}$. The following properties can be observed:

Figure 1: The exponential map.



- $\hat{\mathbf{a}}\mathbf{b} = -\hat{\mathbf{b}}\mathbf{a}$;
- The linear map: $\mathfrak{g} \times \mathfrak{g} \to \mathbb{R}^{k\times k}$, $\mathrm{ad}_{\tilde{\mathbf{a}}}\tilde{\mathbf{b}} \to \mathrm{ad}_{\mathbf{a}}\mathbf{b}$. The useful notation $\mathfrak{g} \times \mathfrak{g} \to \mathbb{R}^{k\times k}$, $\mathrm{ad}^n_{\tilde{\mathbf{a}}}\tilde{\mathbf{b}} \to \mathrm{ad}^n_{\mathbf{a}}\mathbf{b}$;
- The "*check*" $\widecheck{(\bullet)}$ is a linear map such that
$$\hat{\mathbf{a}}^T\mathbf{b} = \check{\mathbf{b}}^T\mathbf{a}, \forall \mathbf{a}, \mathbf{b} \in \mathbb{R}^k, \tag{15}$$
where the superscript T denotes the transpose matrix.

The *exponential operator* maps any element of the Lie algebra $\mathfrak{g}$ to an element of the Lie group $\mathcal{G}$

$$\exp : \mathfrak{g} \to \mathcal{G}, \quad \tilde{\mathbf{x}} \mapsto \mathbf{Q} = \exp(\tilde{\mathbf{x}}) \tag{16}$$

It defines a local diffeomorphism between $\tilde{\mathbf{x}} \in \mathfrak{g}$ and $\mathbf{Q} \in \mathcal{G}$ for $\mathbf{Q}$ sufficiently close to the identity, as illustrated in 1. The exponential map is also defined by the infinite series

$$\exp(\tilde{\mathbf{x}}) = \sum_{i=0}^{\infty} \frac{1}{i!}\tilde{\mathbf{x}}^i \tag{17}$$

where $\tilde{\mathbf{x}}^0$ is the identity element $\mathbf{I}$ and $\tilde{\mathbf{x}}^i$ is the repeated matrix product of $\tilde{\mathbf{x}}$ with itself.

Since the Lie algebra is isomorphic to $\mathbb{R}^k$, the exponential map introduces a local parameterization of the Lie group around the identity $\mathbf{I}$. The exponential map may be seen as a local parameterization in the sense that the argument of the exponential map belongs to a linear space $\mathfrak{g}$ while $\mathcal{G}$ is a non-linear space. In practice, it means that standard vector calculus applies to the argument of the exponential map, such as the multiplication by a scalar or the addition of another k-dimensional vector and that the result of these operations can afterwards be projected onto the group [31].

Eq. (5) can be seen as a first order differential equation on the Lie group $\mathcal{G}$. For a given constant vector field $\widetilde{\mathbf{a}_L} \in \mathfrak{g}$, which does not depend on the parameter $t \in \mathbb{R}$, the initial value problem

$$d_t(\mathbf{Q}) = \mathbf{Q}\widetilde{\mathbf{a}_L}, \quad \mathbf{Q}_0 = \mathbf{Q}(0) \tag{18}$$

admits the solution

$$\mathbf{Q}(t) = \mathbf{Q}_0 \exp(t\widetilde{\mathbf{a}_L}) \tag{19}$$

The exponential map allows one to construct a local parameterization of $\mathcal{G}$ about an arbitrary point $\mathbf{Q}_0 \in \mathcal{G}$ based on the formula

$$\mathbf{Q} = \mathbf{Q}_0 \exp(\tilde{\mathbf{x}}) \tag{20}$$

This diffeomorphism can be written as a coordinate map $\mathbb{R}^k \to \mathcal{G} : \mathbf{x} \mapsto \mathbf{Q} = \mathbf{Q}_0 \exp(\tilde{\mathbf{x}})$.

The derivative of the exponential map can be established as follows. First, the derivative of Eq. (20)

$$d_t(\mathbf{Q}) = \mathbf{Q}_0 D\exp(\tilde{\mathbf{x}}) \cdot d_t(\tilde{\mathbf{x}}) \tag{21}$$

defines a relation between the derivative of the coordinates $\mathbf{x}$ and the derivative of the matrix $\mathbf{Q}$. Considering a functional $\mathbf{F}(\mathbf{y})$ and the vectors $\mathbf{y}$ and $\mathbf{z}$, the directional derivative of $\mathbf{F}$ with respect to $\mathbf{y}$ in the direction $\mathbf{z}$ is denoted as $D_{\mathbf{y}}\mathbf{F}(\mathbf{y})\cdot\mathbf{z}$, the subscript of D can be omitted. In this expression, $D\exp(\tilde{\mathbf{x}}) \cdot d_t(\tilde{\mathbf{x}})$ is the *directional derivative* of the exponential map $\exp(\tilde{\mathbf{x}})$ in the direction of $d_t(\tilde{\mathbf{x}})$. Replacing Eq. (18) in Eq. (20) leads to

$$d_t(\mathbf{Q}) = \mathbf{Q}_0 \exp(\tilde{\mathbf{x}})\widetilde{\mathbf{a}_L} \tag{22}$$

A comparison of Eqs. (21) and (22) leads to a linear relationship between $\mathbf{a}_L$ and $\mathbf{x}$

$$\widetilde{\mathbf{a}_L} = (\exp(\tilde{\mathbf{x}}))^{-1} D\exp(\tilde{\mathbf{x}}) \cdot d_t(\tilde{\mathbf{x}}) \tag{23}$$

In [30], the *left-trivialized differential* (tangent) of the exponential map is defined as a function

$$\mathrm{dexp} : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}, \quad \tilde{\mathbf{x}}, d_t(\tilde{\mathbf{x}}) \mapsto \mathrm{dexp}_{\tilde{\mathbf{x}}}(d_t(\tilde{\mathbf{x}})) \tag{24}$$

such that

$$D\exp(\tilde{\mathbf{x}}(t))\cdot d_t\tilde{\mathbf{x}}(t) = \exp(\tilde{\mathbf{x}}(t))\,\mathrm{dexp}_{-\tilde{\mathbf{x}}(t)}\,(d_t\tilde{\mathbf{x}}(t)) \tag{25}$$

The matrix $\mathrm{dexp}_{\tilde{\mathbf{x}}}$ is an analytic function of the matrix $\mathrm{ad}_{\tilde{\mathbf{x}}}$ which satisfies $\mathrm{dexp}_{\tilde{\mathbf{x}}} = (\exp(\mathrm{ad}_{\tilde{\mathbf{x}}}) - \mathbf{I})/\mathrm{ad}_{\tilde{\mathbf{x}}}$. The operator $\mathrm{dexp}_{-\tilde{\mathbf{x}}}$ can be expressed as a power series in the following manner. Since for a scalar $y \in \mathbb{R}$

$$\frac{1-e^{-y}}{y} = 1 - \frac{1}{2!}y + \frac{1}{3!}y^2 - \frac{1}{4!}y^3 + \cdots + \frac{(-1)^i}{(1+i)!}y^i + \cdots$$

similarly one obtains the left-trivialized differential $\mathrm{dexp}_{-\tilde{\mathbf{x}}}$

$$\mathrm{dexp}_{-\tilde{\mathbf{x}}} = \frac{\mathbf{I} - \exp(-\mathrm{ad}_{\tilde{\mathbf{x}}})}{\mathrm{ad}_{\tilde{\mathbf{x}}}} = \sum_{i=0}^{\infty}\frac{(-1)^i}{(1+i)!}(\mathrm{ad}_{\tilde{\mathbf{x}}})^i \tag{26}$$

where the property $\mathrm{ad}_{-\tilde{\mathbf{x}}} = -\mathrm{ad}_{\tilde{\mathbf{x}}}$ has been used.

Next, the derivative of the exponential map can be formulated in terms of vectors in $\mathbb{R}^k$. From Eqs. (23) and (25), we obtain $\widetilde{\mathbf{a}_L} = \mathrm{dexp}_{-\tilde{\mathbf{x}}(t)}d_t\tilde{\mathbf{x}}(t)$ so that

$$\widetilde{\mathbf{a}_L} = \sum_{i=0}^{\infty}\frac{(-1)^i}{(1+i)!}(\mathrm{ad}_{\tilde{\mathbf{x}}})^i d_t(\tilde{\mathbf{x}}) \tag{27}$$

or

$$\widetilde{\mathbf{a}_L} = d_t(\tilde{\mathbf{x}}) - \frac{1}{2!}[\tilde{\mathbf{x}}, d_t(\tilde{\mathbf{x}})] + \frac{1}{3!}[\tilde{\mathbf{x}}, [\tilde{\mathbf{x}}, d_t(\tilde{\mathbf{x}})]] - \frac{1}{4!}[\tilde{\mathbf{x}}, [\tilde{\mathbf{x}}, [\tilde{\mathbf{x}}, d_t(\tilde{\mathbf{x}})]]] + \cdots \tag{28}$$

This formula can be expressed as a linear relationship between the vectors $\mathbf{a}_L$ and $d_t(\mathbf{x})$ in $\mathbb{R}^k$ as

$$\mathbf{a}_L = \mathbf{I}d_t(\mathbf{x}) - \frac{1}{2!}\hat{\mathbf{x}}d_t(\mathbf{x}) + \frac{1}{3!}\hat{\mathbf{x}}^2 d_t(\mathbf{x}) - \frac{1}{4!}\hat{\mathbf{x}}^3 d_t(\mathbf{x}) + \cdots \tag{29}$$

which can be rewritten as a matrix–vector multiplication

$$\mathbf{a}_L = \mathbf{T}(\mathbf{x})d_t(\mathbf{x}) \tag{30}$$

where $\mathbf{T}$ is the *tangent operator of the exponential map* given by

$$\mathbf{T}(\mathbf{x}) = \sum_{i=0}^{\infty}\frac{(-1)^i}{(1+i)!}\hat{\mathbf{x}}^i \tag{31}$$

Notice that the tangent operator $\mathbf{T}(\mathbf{x})$, Eqs. (31), is the expression of the left-trivialized differential $\mathrm{dexp}_{-\tilde{\mathbf{x}}}$, Eqs. (26), in terms of vectors of $\mathbb{R}^k$. The following relationships also hold

$$d_t(\mathbf{x}) = \mathbf{T}^{-1}(\mathbf{x})\mathbf{a}_L, \qquad (d_t(\mathbf{x}))^T = \mathbf{a}_L^T\mathbf{T}^{-T}(\mathbf{x}), \qquad \mathbf{a}_L^T = d_t^T(\mathbf{x})\mathbf{T}^T(\mathbf{x}) \tag{32}$$

$$\mathbf{T}^{-1}(\mathbf{x}) = \sum_{i=0}^{\infty}\frac{(-1)^i B_i}{i!}\hat{\mathbf{x}}^i, \qquad \mathbf{T}^{-T}(\mathbf{x}) = \sum_{i=0}^{\infty}\frac{(-1)^i B_i}{i!}(\hat{\mathbf{x}}^T)^i, \qquad \mathbf{T}^T(\mathbf{x}) = \sum_{i=0}^{\infty}\frac{(-1)^i}{(1+i)!}(\hat{\mathbf{x}}^T)^i \tag{33}$$

where B_i forms a sequence of rational numbers, called the *Bernoulli numbers* of the first kind.

The inverse of the exponential map is called the *logarithmic map*,

$$\log : \mathcal{G} \to \mathfrak{g}, \quad \mathbf{Q} \mapsto \tilde{\mathbf{x}} = \log(\mathbf{Q}) \tag{34}$$

and is defined such that $\exp(\log(\mathbf{Q})) = \mathbf{Q}$. The logarithm map is defined for any $\mathbf{Q} \in \mathcal{G}$. In some cases, the logarithm map admits the series expansion

$$\log(\mathbf{Q}) = -\sum_{i=1}^{\infty}\frac{(\mathbf{I}-\mathbf{Q})^i}{i} \tag{35}$$

However, this series expansion only converges in some restricted regions of $\mathcal{G}$. If $\mathbf{Q}$ is diagonalizable, the convergence is guaranteed if additionally all eigenvalues λ_i of $\mathbf{Q}$ satisfy $|\lambda_i - 1| < 1$. If these conditions are not satisfied, the logarithm still exists but there is no guarantee that the series expansion (35) converges and can be used for its evaluation. Two different matrices $\tilde{\mathbf{x}}_1, \tilde{\mathbf{x}}_2 \in \mathfrak{g}$ with $\tilde{\mathbf{x}}_1 \neq \tilde{\mathbf{x}}_2$ may have the same exponential $\exp(\tilde{\mathbf{x}}_1) = \exp(\tilde{\mathbf{x}}_2)$. For that reason, the property $\log(\exp(\tilde{\mathbf{x}})) = \tilde{\mathbf{x}}$ is only valid in a restricted region of $\mathfrak{g}$ defined by the condition $\|\tilde{\mathbf{x}}\| < \log 2$ [33].

3 Derivatives

The directional derivatives of the exponential map and of the tangent operator on any matrix Lie group are now developed based on their series expressions. To the best of our knowledge, the resulting matrix series expressions are novel. This section proposes a step-by-step derivation of the formulae so that the reader can reproduce our reasoning. The final results that can be used for a general implementation in a computer code are given in the last 3.3.

3.1 Directional derivative of the tangent operator $D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}$

The development of the first directional derivative of the tangent operator $\mathbf{T}(\mathbf{a})$ in the direction of $\mathbf{b}$, is based on the product rule and on a recursive formulation. Starting from Eq. (31), we obtain

$$D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b} = \sum_{i=0}^{\infty} \frac{(-1)^i}{(1+i)!} D\hat{\mathbf{a}}^i \cdot \mathbf{b} \tag{36}$$

We know that $D\hat{\mathbf{a}} \cdot \mathbf{b} = \hat{\mathbf{b}}$ is valid for all matrix Lie groups since $\widehat{(\bullet)}$ is a linear operator, see Eq. (14). The variable $D\hat{\mathbf{a}}^i \cdot \mathbf{b}$ can be evaluated recursively using the initialization

$$D\hat{\mathbf{a}}^0 \cdot \mathbf{b} = D\mathbf{I} \cdot \mathbf{b} = \mathbf{0} \tag{37}$$

and the recursive step

$$\begin{aligned} D\hat{\mathbf{a}}^i \cdot \mathbf{b} &= D(\hat{\mathbf{a}}\hat{\mathbf{a}}^{i-1}) \cdot \mathbf{b} \\ &= (D\hat{\mathbf{a}} \cdot \mathbf{b})\hat{\mathbf{a}}^{i-1} + \hat{\mathbf{a}} D\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b} \\ &= \hat{\mathbf{b}}\hat{\mathbf{a}}^{i-1} + \hat{\mathbf{a}}\, \underbrace{D\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}}_{\text{Recursive}} \end{aligned} \tag{38}$$

Since $D\hat{\mathbf{a}}^0 \cdot \mathbf{b} = \mathbf{0}$, the series in Eq. (36) could also start at $i = 1$ and be written as $D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b} = \sum_{i=1}^{\infty} \frac{(-1)^i}{(1+i)!} D\hat{\mathbf{a}}^i \cdot \mathbf{b}$.

Similarly, the directional derivative of the inverse operator is obtained as

$$D\mathbf{T}^{-1}(\mathbf{a}) \cdot \mathbf{b} = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} D\hat{\mathbf{a}}^i \cdot \mathbf{b} \tag{39}$$

and the derivative of the transposed operator as

$$D\mathbf{T}^T(\mathbf{a}) \cdot \mathbf{b} = \sum_{i=1}^{\infty} \frac{(-1)^i}{(1+i)!} D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b}, D\mathbf{T}^{-T}(\mathbf{a}) \cdot \mathbf{b} = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b} \tag{40}$$

with the recursive expression

$$D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b} = \hat{\mathbf{b}}^T (\hat{\mathbf{a}}^T)^{i-1} + \hat{\mathbf{a}}^T D(\hat{\mathbf{a}}^T)^{i-1} \cdot \mathbf{b} \tag{41}$$

The second directional derivative of the tangent operator $D_{\mathbf{a}}(D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d}$ is obtained as:

$$D_{\mathbf{a}}(D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d} = \sum_{i=1}^{\infty} \frac{(-1)^i}{(i+1)!} D_{\mathbf{a}}(D\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d} \tag{42}$$

$$= \sum_{i=1}^{\infty} \frac{(-1)^i}{(i+1)!} D_{\mathbf{a}}(\hat{\mathbf{b}}\hat{\mathbf{a}}^{i-1} + \hat{\mathbf{a}} D\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d} \tag{43}$$

$$= \sum_{i=1}^{\infty} \frac{(-1)^i}{(i+1)!} (\, \hat{\mathbf{b}} D\hat{\mathbf{a}}^{i-1} \cdot \mathbf{d} + \hat{\mathbf{d}} D\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b} + \hat{\mathbf{a}}\, \underbrace{D_{\mathbf{a}}(D\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d}}_{\text{Recursive}} \,) \tag{44}$$

In the above formula, one may observe that the term for $i = 1$ vanishes and that, in principle, the series could start at $i = 2$. Clearly, each differentiation step kills one term in the series. In the next developments, we will keep starting the series of higher order derivatives at $i = 1$ and will rely on the convention that $D(\hat{\mathbf{a}})^i \cdot \mathbf{b} = D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b} = \mathbf{0}$ if $i \leq 0$.

In B.1, various derivatives of $\hat{\mathbf{a}}^i$, such as $D_{\mathbf{a}}(D\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}$, are provided. Based on this information, the evaluation of the derivative of $\mathbf{T}$ is straightforwardly obtained by summation using Eq. (42).

Similarly, the second directional derivative of the inverse of tangent operator is :

$$D_{\mathbf{a}}(D\mathbf{T}^{-1}(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d} = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} D_{\mathbf{a}}(D\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d} \tag{45}$$

and the second directional derivatives of the transposed operator are:

$$D_{\mathbf{a}}(D\mathbf{T}^T(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d} = \sum_{i=1}^{\infty} \frac{(-1)^i}{(i+1)!} D_{\mathbf{a}}(D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b}) \cdot \mathbf{d}, D_{\mathbf{a}}(D\mathbf{T}^{-T}(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d} = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} D_{\mathbf{a}}(D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b}) \cdot \mathbf{d} \tag{46}$$

Also, the directional derivatives $\mathbf{T}^{-1}$, $\mathbf{T}^T$ and $\mathbf{T}^{-T}$, are easily evaluated from the derivative of $\hat{\mathbf{a}}^i$ provided in B.1.

3.2 Gradient of the tangent operator multiplied by a constant vector $\nabla_{\mathbf{a}} (\mathbf{T}(\mathbf{a})\ \mathbf{c})$

The gradient $\nabla_{\mathbf{a}}(\mathbf{T}(\mathbf{a})\ \mathbf{c})$ is defined as the operator such that $\nabla_{\mathbf{a}}(\mathbf{T}(\mathbf{a})\ \mathbf{c})\ \mathbf{b} = D_{\mathbf{a}}(\mathbf{T}(\mathbf{a})\ \mathbf{c}) \cdot \mathbf{b}$. It represents the directional derivative of the tangent operator when it is multiplied on the right by a constant vector $\mathbf{c}$, with respect to $\mathbf{a}$ in the direction $\mathbf{b}$. Observing that

$$D_{\mathbf{a}}(\mathbf{T}(\mathbf{a})\ \mathbf{c}) \cdot \mathbf{b} = \sum_{i=1}^{\infty} \frac{(-1)^i}{(1+i)!} D_{\mathbf{a}}(\hat{\mathbf{a}}^i\ \mathbf{c}) \cdot \mathbf{b} \tag{47}$$

forthwith

$$\begin{aligned} D_{\mathbf{a}}(\hat{\mathbf{a}}^i\ \mathbf{c}) \cdot \mathbf{b} &= D_{\mathbf{a}}(\hat{\mathbf{a}}) \cdot \mathbf{b}\ (\hat{\mathbf{a}}^{i-1}\ \mathbf{c}) + \hat{\mathbf{a}} D_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\ \mathbf{c}) \cdot \mathbf{b} \\ &= \hat{\mathbf{b}}(\hat{\mathbf{a}}^{i-1}\ \mathbf{c}) + \hat{\mathbf{a}} D_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\ \mathbf{c}) \cdot \mathbf{b} \\ &= -\widehat{(\hat{\mathbf{a}}^{i-1}\ \mathbf{c})}\mathbf{b} + \hat{\mathbf{a}} D_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\ \mathbf{c}) \cdot \mathbf{b} \end{aligned} \tag{48}$$

using the property Eq. (14) and isolating $\mathbf{b}$ of the Eq. (48), we get

$$\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\ \mathbf{c}) = -\widehat{(\hat{\mathbf{a}}^{i-1}\ \mathbf{c})} + \hat{\mathbf{a}} \underbrace{\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\ \mathbf{c})}_{\text{Recursive}} \tag{49}$$

the computation of $\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\ \mathbf{c})$ can be done recursively with the initialization $\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^0\ \mathbf{c}) = \mathbf{0}$ and the recursive step. we obtain

$$\nabla_{\mathbf{a}}(\mathbf{T}(\mathbf{a})\ \mathbf{c}) = \sum_{i=1}^{\infty} \frac{(-1)^i}{(1+i)!} \nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\ \mathbf{c}) \tag{50}$$

Similarly,

$$\nabla_{\mathbf{a}}(\mathbf{T}^{-1}(\mathbf{a})\ \mathbf{c}) = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} \nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\ \mathbf{c}) \tag{51}$$

Noting that $\nabla_{\mathbf{a}}(\mathbf{T}^T(\mathbf{a})\ \mathbf{c})\,\mathbf{b} = D_{\mathbf{a}}(\mathbf{T}^T(\mathbf{a})\ \mathbf{c}) \cdot \mathbf{b}$, the same technique applied to the transposed tangent operator gives

$$D_{\mathbf{a}}(\mathbf{T}^T(\mathbf{a})\ \mathbf{c}) \cdot \mathbf{b} = \sum_{i=1}^{\infty} \frac{(-1)^i}{(1+i)!} D_{\mathbf{a}}\left((\hat{\mathbf{a}}^T)^i\ \mathbf{c}\right) \cdot \mathbf{b} \tag{52}$$

forthwith

$$\begin{aligned} D_{\mathbf{a}}((\hat{\mathbf{a}}^T)^i\ \mathbf{c}) \cdot \mathbf{b} &= D_{\mathbf{a}}(\hat{\mathbf{a}}^T) \cdot \mathbf{b}\ ((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c}) + \hat{\mathbf{a}}^T D_{\mathbf{a}}((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c}) \cdot \mathbf{b} \\ &= \hat{\mathbf{b}}^T((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c}) + \hat{\mathbf{a}}^T D_{\mathbf{a}}((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c}) \cdot \mathbf{b} \\ &= \widehat{((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c})}\mathbf{b} + \hat{\mathbf{a}}^T D_{\mathbf{a}}((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c}) \cdot \mathbf{b} \end{aligned} \tag{53}$$

using the property Eq. (15) and isolating $\mathbf{b}$ of the Eq. (53), we get

$$\nabla_{\mathbf{a}}((\hat{\mathbf{a}}^T)^i\ \mathbf{c}) = \widehat{((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c})} + \hat{\mathbf{a}}^T \underbrace{\nabla_{\mathbf{a}}((\hat{\mathbf{a}}^T)^{i-1}\ \mathbf{c})}_{\text{Recursive}} \tag{54}$$

Similarly

$$\nabla_{\mathbf{a}}(\mathbf{T}^{-T}(\mathbf{a})\ \mathbf{c}) = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} \nabla_{\mathbf{a}}((\hat{\mathbf{a}}^T)^i\ \mathbf{c}) \tag{55}$$

The same technique is valid for higher order gradient of the tangent operator, some of which are provided in the B.1. For example, the gradient of the derivative of the tangent operator multiplied by a constant vector $\mathbf{c}$ is given by:

$$\nabla_{\mathbf{a}}(D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}\ \mathbf{c}) = \sum_{i=1}^{\infty} \frac{(-1)^i}{(1+i)!} \nabla_{\mathbf{a}}(D\hat{\mathbf{a}}^i \cdot \mathbf{b}\ \mathbf{c}), \qquad \nabla_{\mathbf{a}}(D\mathbf{T}^{-1}(\mathbf{a}) \cdot \mathbf{b}\ \mathbf{c}) = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} \nabla_{\mathbf{a}}(D\hat{\mathbf{a}}^i \cdot \mathbf{b}\ \mathbf{c}) \tag{56}$$

$$\nabla_{\mathbf{a}}(D\mathbf{T}^T(\mathbf{a}) \cdot \mathbf{b}\ \mathbf{c}) = \sum_{i=1}^{\infty} \frac{(-1)^i}{(1+i)!} \nabla_{\mathbf{a}}(D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b}\ \mathbf{c}), \qquad \nabla_{\mathbf{a}}(D\mathbf{T}^{-T}(\mathbf{a}) \cdot \mathbf{b}\ \mathbf{c}) = \sum_{i=1}^{\infty} \frac{(-1)^i B_i}{i!} \nabla_{\mathbf{a}}(D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b}\ \mathbf{c}) \tag{57}$$

where $\nabla_{\mathbf{a}}(D\hat{\mathbf{a}}^i \cdot \mathbf{b}\ \mathbf{c})$ and $\nabla_{\mathbf{a}}(D(\hat{\mathbf{a}}^T)^i \cdot \mathbf{b}\ \mathbf{c})$ are provided in the B.1.

3.3 General expression for high order directional derivatives and gradients

The higher order directional derivatives and gradients of the operator $\mathbf{T}(\mathbf{a})$ are all expressed in terms of the derivatives and gradient of $\hat{\mathbf{a}}^i$. The procedure to obtain these terms turns out to be quite systematic and leads to rather compact and easy to implement expressions, as it can be seen in B.1 in various cases. From these expressions, a general and original formula for directional derivatives of order n in the direction of vectors $\mathbf{b}_j$, $j = 1 \cdots n$, and gradients of order m of the hat operator multiplied by constants vector $\mathbf{c}_k$, $k = 1 \cdots m$ is obtained as

$$\begin{aligned} \nabla^m\left(\left(D^n\left(\left(\hat{\mathbf{a}}^i \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right)\mathbf{c}_1\right) \dots \mathbf{c}_m\right) =& \sum_{j=1}^{n} \widehat{\mathbf{b}_j} \nabla^m\left(\left(D^{n-1}\left(\left(\left(\left(\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_{j-1}\right) \cdot \mathbf{b}_{j+1}\right) \dots \cdot \mathbf{b}_n\right)\mathbf{c}_1\right) \dots \mathbf{c}_m\right) \\ &+ \sum_{k=2}^{m} \widehat{\mathbf{c}_k} \nabla^{m-1}\left(\left(\left(\left(D^n\left(\left(\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right)\mathbf{c}_1\right) \dots \mathbf{c}_{k-1}\right)\mathbf{c}_{k+1}\right) \dots \mathbf{c}_m\right) \\ &- (1 - \delta_{m\,0})\,\widehat{\left(\nabla^{m-1}\left(\left(D^n\left(\left(\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right)\mathbf{c}_1\right) \dots \mathbf{c}_{m-1}\right)\mathbf{c}_m\right)} \\ &+ \hat{\mathbf{a}} \nabla^m\left(\left(D^n\left(\left(\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right)\mathbf{c}_1\right) \dots \mathbf{c}_m\right) \end{aligned} \tag{58}$$

where δ_{ij} is the Kronecker delta.

Likewise, the higher order directional derivatives and gradients of the transposed operator $\mathbf{T}^T(\mathbf{a})$ are obtained as

$$\begin{aligned}\nabla^m \left(\left(D^n \left(\left((\hat{\mathbf{a}}^T)^i \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right) \mathbf{c}_1\right) \dots \mathbf{c}_m\right) = & \sum_{j=1}^{n} \widehat{\mathbf{b}_j}^T \nabla^m \left(\left(D^{n-1} \left(\left(\left(\left((\hat{\mathbf{a}}^T)^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_{j-1}\right) \cdot \mathbf{b}_{j+1}\right) \dots \cdot \mathbf{b}_n\right) \mathbf{c}_1\right) \dots \mathbf{c}_m\right) \\ & + \sum_{k=2}^{m} \widehat{\mathbf{c}_k}^T \nabla^{m-1} \left(\left(\left(\left(D^n \left(\left((\hat{\mathbf{a}}^T)^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right) \mathbf{c}_1\right) \dots \mathbf{c}_{k-1}\right) \mathbf{c}_{k+1}\right) \dots \mathbf{c}_m\right) \\ & + (1 - \delta_{m\,0}) \overline{\left(\nabla^{m-1} \left(\left(D^n \left(\left((\hat{\mathbf{a}}^T)^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right) \mathbf{c}_1\right) \dots \mathbf{c}_{m-1}\right) \mathbf{c}_m\right)}^T \\ & + (\hat{\mathbf{a}}^T) \nabla^m \left(\left(D^n \left(\left((\hat{\mathbf{a}}^T)^{i-1} \cdot \mathbf{b}_1\right) \dots \cdot \mathbf{b}_n\right) \mathbf{c}_1\right) \dots \mathbf{c}_m\right) \end{aligned} \quad (59)$$

In the B.2, we detail a compact algorithm which implements these two general formulae.

4 The $SO(3)$ and $SE(3)$ matrix Lie groups

The formulae obtained in the previous sections are applicable to the Lie groups $SO(3)$ and $SE(3)$ as special cases. For these Lie groups, closed form expressions for the exponential map and its tangent operator are available and can also be considered for the development of the higher order derivatives and gradients. These expressions are detailed below.

4.1 The special orthogonal group $SO(3)$

The set of finite rotation matrices, i.e.,

$$SO(3) :- \left\{\mathbf{R} \in \mathbb{R}^{3\times 3} \mid \mathbf{R}^T\mathbf{R} = \mathbf{I}_{3\times 3},\ \det(\mathbf{R}) = +1\right\} \quad (60)$$

together with the product of a 3×3 real matrix as a composition (group) operation, are a matrix Lie group, which is called the special orthogonal group [34]. It is a group since it satisfies the four axioms. Firstly, the closure axiom is satisfied because the composition of two rotation matrices yields a rotation matrix too. Secondly, the matrix product is associative. Thirdly, the neutral element is the 3×3 identity matrix. Finally, the inverse of a rotation matrix $\mathbf{R}$ always exists and is simply the transposed $\mathbf{R}^T$, which is also a rotation matrix.

The group $SO(3)$ is a matrix Lie group since the product and the inverse of a matrix are smooth operations. At any point $\mathbf{R}$ on the manifold $SO(3)$, the tangent space is noted $T_{\mathbf{R}}SO(3)$. The Lie algebra $\mathfrak{so}(3)$ is defined as

$$\mathfrak{so}(3) :- \left\{\widetilde{\mathbf{x}_\omega} \in \mathbb{R}^{3\times 3} \mid \widetilde{\mathbf{x}_\omega} + \widetilde{\mathbf{x}_\omega}^T = \mathbf{0}_{3\times 3}\right\} \quad (61)$$

$\mathfrak{so}(3)$ can be identified with $\mathbb{R}^3$ since the three non-trivial components of a skew-symmetric matrix $\widetilde{\mathbf{x}_\omega}$ can be collected in a vector $\mathbf{x}_\omega$ with the following expressions

$$\widetilde{\mathbf{x}_\omega} = \begin{bmatrix} 0 & -x_{\omega 3} & x_{\omega 2} \\ x_{\omega 3} & 0 & -x_{\omega 1} \\ -x_{\omega 2} & x_{\omega 1} & 0 \end{bmatrix}, \quad \mathbf{x}_\omega = \begin{bmatrix} x_{\omega 1} \\ x_{\omega 2} \\ x_{\omega 3} \end{bmatrix} \quad (62)$$

In this way, the tilde operator $\widetilde{(\bullet)}$ maps a vector $\mathbf{x}_\omega \in \mathbb{R}^3$ into a skew-symmetric matrix $\widetilde{\mathbf{x}_\omega} \in \mathfrak{so}(3)$. We also have the relationship between the tilde operator and the cross product between the vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^3$: $\mathbf{a} \times \mathbf{b} = \tilde{\mathbf{a}}\mathbf{b}$. For $SO(3)$, the hat operator $\widehat{(\bullet)}$ defined in Eq. (14) coincides with the tilde operator $\widetilde{(\bullet)}$. Therefore, the check operator $\widecheck{(\bullet)}$, defined in Eq. (15), is equivalent to $\widetilde{(\bullet)}^T$ in this case.

From Eq. (5), the derivative of an element of $SO(3)$ can be represented as

$$d_a(\mathbf{R}) = \mathbf{R}\widetilde{\mathbf{x}_\omega} \quad (63)$$

where $d_a(\mathbf{R}) : SO(3) \to T_{\mathbf{R}}SO(3)$.

The closed form of the exponential map can be obtained by exploiting the property $\hat{\mathbf{x}}_\omega = \tilde{\mathbf{x}}_\omega$ (which only holds for $SO(3)$) and the property of skew-symmetric 3×3 matrices $\tilde{\mathbf{x}}_\omega^3 = -\|\mathbf{x}_\omega\|^2\tilde{\mathbf{x}}_\omega$. In this last expression, $\|\mathbf{x}_\omega\| = \sqrt{x_{\omega 1}^2 + x_{\omega 2}^2 + x_{\omega 3}^2}$ is the Euclidean norm. Applying these properties to Eq. (17), we obtain:

$$\exp_{SO(3)}(\widetilde{\mathbf{x}_\omega}) = \mathbf{I} + \widetilde{\mathbf{x}_\omega} + \frac{\widetilde{\mathbf{x}_\omega}^2}{2!} - \frac{\|\mathbf{x}_\omega\|^2\widetilde{\mathbf{x}_\omega}}{3!} - \frac{\|\mathbf{x}_\omega\|^2\widetilde{\mathbf{x}_\omega}^2}{4!} + \frac{\|\mathbf{x}_\omega\|^4\widetilde{\mathbf{x}_\omega}}{5!} + \frac{\|\mathbf{x}_\omega\|^4\widetilde{\mathbf{x}_\omega}^2}{6!} - \frac{\|\mathbf{x}_\omega\|^6\widetilde{\mathbf{x}_\omega}}{7!} - \frac{\|\mathbf{x}_\omega\|^6\widetilde{\mathbf{x}_\omega}^2}{8!} + \cdots \quad (64)$$

One can then consider the Taylor's series of $\sin(\theta)$ and $\cos(\theta)$ (see [35], chapter 2.3.3)

$$\sin(\theta) = \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \frac{\theta^9}{9!} - \frac{\theta^{11}}{11!} + \frac{\theta^{13}}{13!} - \cdots, \quad \cos(\theta) = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \frac{\theta^8}{8!} - \frac{\theta^{10}}{10!} + \frac{\theta^{12}}{12!} - \cdots \quad (65)$$

In Eq. (64), assuming $\|\mathbf{x}_\omega\| \neq 0$, one can multiply some terms by $\|\mathbf{x}_\omega\|/\|\mathbf{x}_\omega\|$ and other terms by $\|\mathbf{x}_\omega\|^2/\|\mathbf{x}_\omega\|^2$

$$\begin{aligned}\exp_{SO(3)}(\widetilde{\mathbf{x}_\omega}) =& \mathbf{I} + \left(\|\mathbf{x}_\omega\| - \frac{\|\mathbf{x}_\omega\|^3}{3!} + \frac{\|\mathbf{x}_\omega\|^5}{5!} - \frac{\|\mathbf{x}_\omega\|^7}{7!} - \cdots\right)\frac{\widetilde{\mathbf{x}_\omega}}{\|\mathbf{x}_\omega\|} + \left(1 - 1 + \frac{\|\mathbf{x}_\omega\|^2}{2!} - \frac{\|\mathbf{x}_\omega\|^4}{4!} + \frac{\|\mathbf{x}_\omega\|^6}{6!} + \cdots\right)\frac{\widetilde{\mathbf{x}_\omega}^2}{\|\mathbf{x}_\omega\|^2} \\ =& \mathbf{I} + \frac{\sin(\|\mathbf{x}_\omega\|)}{\|\mathbf{x}_\omega\|}\widetilde{\mathbf{x}_\omega} + \frac{1 - \cos(\|\mathbf{x}_\omega\|)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2 \end{aligned} \quad (66)$$

This new expression of the exponential map has a singularity when $\|\mathbf{x}_\omega\| = 0$. This formula, commonly referred to as *Rodrigues'formula*, gives an efficient method for computing $\exp_{SO(3)}(\mathbf{x}_\omega)$. The inverse deduction of the serial formula from the Rodrigues formula was presented in [36, 37]. One can also observe that $\exp_{SO(3)}$ is not injective. Indeed, let $\mathbf{a}^* = (1+2k\pi/\|\mathbf{a}\|)\mathbf{a}$, for all integer k, we have $\exp_{SO(3)}(\widetilde{\mathbf{a}^*}) = \exp_{SO(3)}(\widetilde{\mathbf{a}})$.

The expression of the Rodrigues formula can be adjusted by defining auxiliary quantities:

$$\alpha(\mathbf{x}_\omega) = \frac{\sin(\|\mathbf{x}_\omega\|)}{\|\mathbf{x}_\omega\|}, \quad \beta(\mathbf{x}_\omega) = 2\frac{1-\cos(\|\mathbf{x}_\omega\|)}{\|\mathbf{x}_\omega\|^2}, \quad \gamma(\mathbf{x}_\omega) = \frac{\|\mathbf{x}_\omega\|}{2}\cot(\frac{\|\mathbf{x}_\omega\|}{2}) \tag{67}$$

Notice that $\alpha(\mathbf{0}) = \beta(\mathbf{0}) = \gamma(\mathbf{0}) = 1$. The *closed form of exponential map on* $SO(3)$ is then given by

$$\exp_{SO(3)}(\mathbf{x}_\omega) = \begin{cases} \mathbf{I} + \alpha(\mathbf{x}_\omega)\widetilde{\mathbf{x}_\omega} + \frac{\beta(\mathbf{x}_\omega)}{2}\widetilde{\mathbf{x}_\omega}^2 & \text{if } \|\mathbf{x}_\omega\| \geqslant \epsilon, \\ \mathbf{I} & \text{otherwise} \end{cases} \tag{68}$$

where ϵ is a numerical parameter whose value should be chosen small enough. This parameter defines the switch point between the two equations in the numerical implementation of the exponential map. A more comprehensive analysis will be made later about the influence of this parameter.

The *closed form of the logarithm map on* $SO(3)$ is given by

$$\log_{SO(3)}(\mathbf{R}) = \widetilde{\mathbf{x}_\omega} = \begin{cases} \frac{\theta}{2\sin(\theta)}(\mathbf{R} - \mathbf{R}^T) & \text{if } \mathbf{R} \neq \mathbf{I}_{3\times3}, \\ \mathbf{0} & \text{if } \mathbf{R} = \mathbf{I}_{3\times3} \end{cases} \tag{69}$$

where θ is the angle of rotation [29]. This angle can be calculated in two ways. Firstly, it can be evaluated as $\theta_{\text{acos}} = \text{acos}(0.5\ (\text{trace}(\mathbf{R}) - 1))$, where the trace($\bullet$) function is defined to be the sum of elements on the main diagonal. Secondly, it can be evaluated as $\theta_{\text{asin}} = \text{asin}(\|\text{vect}(\mathbf{R})\|)$, where vect($\bullet$) denotes the vectorial part of a matrix, i.e., $\text{vect}(\mathbf{R}) = 0.5\,[(R_{32} - R_{23})\ (R_{13} - R_{31})\ (R_{21} - R_{12})]^T$. It is important to note that when $\mathbf{R}$ is close to the identity $\mathbf{I}$, θ_{asin} gives a better accuracy than θ_{acos}. This is because the derivative of the cosine is zero when the angle is zero, which induces a singularity in the acos function. For this reason, it is recommended to first calculate θ_{acos} and use this value when the angle is greater than a certain tolerance (1×10^{-7}), otherwise the angle should be recalculated using θ_{asin}. Alternatively, a more straightforward solution can be achieved by: $\theta_{\text{atan2}} = \text{atan2}(\,\|\text{vect}(\mathbf{R})\|,\ 0.5(\text{trace}(\mathbf{R}) - 1)\,)$. A slightly different formula to calculate $\log_{SO(3)}$ is presented in [25], chapter 4.4.1.

The *closed form of the tangent operator on* $SO(3)$ is obtained from Eq. (31) and the property (C.1), following a similar approach as for the exponential map (Eq. (66)):

$$\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) = \begin{cases} \mathbf{I} - \frac{\beta(\mathbf{x}_\omega)}{2}\widetilde{\mathbf{x}_\omega} + \frac{1-\alpha(\mathbf{x}_\omega)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2 & \text{if } \|\mathbf{x}_\omega\| \geqslant \epsilon, \\ \mathbf{I} & \text{otherwise} \end{cases} \tag{70}$$

The directional derivative of the tangent operator can then be evaluated from:

$$D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u = \frac{\partial \mathbf{T}_{SO(3)}(x_{\omega1}, x_{\omega2}, x_{\omega3})}{\partial x_{\omega1}} x_{u1} + \frac{\partial \mathbf{T}_{SO(3)}(x_{\omega1}, x_{\omega2}, x_{\omega3})}{\partial x_{\omega2}} x_{u2} + \frac{\partial \mathbf{T}_{SO(3)}(x_{\omega1}, x_{\omega2}, x_{\omega3})}{\partial x_{\omega3}} x_{u3} \tag{71}$$

where $\frac{\partial}{\partial x_{\omega1}}$ is the partial derivative with respect to the variable $x_{\omega1}$, $\mathbf{x}_u = [x_{u1}\ x_{u2}\ x_{u3}]^T$ is a column vector and $\mathbf{T}_{SO(3)}(x_{\omega1}, x_{\omega2}, x_{\omega3})$ is the closed form (first line of Eq. (70)). Unfortunately, such derivatives in closed form inherit from the singularity at the origin. Again a threshold ϵ should be considered and the formula for the derivative should be replaced by its limit value if the rotation amplitude is below ϵ [18]. The *first directional derivative* $SO(3)$ is obtained as:

$$D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u = \begin{cases} -\frac{\beta}{2}\widetilde{\mathbf{x}_u} - \frac{(\alpha-\beta)\,(\mathbf{x}_\omega^T\mathbf{x}_u)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega} + \frac{1-\alpha}{\|\mathbf{x}_\omega\|^2}\lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil + \frac{(\beta\,\|\mathbf{x}_\omega\|^2 + 6(\alpha-1))\,(\mathbf{x}_\omega^T\mathbf{x}_u)}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2 & \text{if } \|\mathbf{x}_\omega\| \geqslant \epsilon, \\ -\frac{1}{2}\widetilde{\mathbf{x}_u} & \text{otherwise} \end{cases} \tag{72}$$

The evaluation of the gradient yields

$$\nabla_{\mathbf{x}_\omega}(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\mathbf{c}) = \begin{cases} +\frac{\beta}{2}\widetilde{\mathbf{c}} - \frac{(\alpha-\beta)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_\omega^T) + \frac{1-\alpha}{\|\mathbf{x}_\omega\|^2}\lfloor \mathbf{x}_\omega, \mathbf{c} \rfloor + \frac{(\beta\,\|\mathbf{x}_\omega\|^2 + 6(\alpha-1))}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{x}_\omega^T) & \text{if } \|\mathbf{x}_\omega\| \geqslant \epsilon, \\ \frac{1}{2}\widetilde{\mathbf{c}} & \text{otherwise} \end{cases} \tag{73}$$

with $\alpha = \alpha(\mathbf{x}_\omega)$ and $\beta = \beta(\mathbf{x}_\omega)$ from Eq. (67), $\lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil = \widetilde{\mathbf{x}_\omega}\widetilde{\mathbf{x}_u} + \widetilde{\mathbf{x}_u}\widetilde{\mathbf{x}_\omega}$ and $\lfloor \mathbf{x}_\omega, \mathbf{x}_u \rfloor = \widetilde{\mathbf{x}_u}\widetilde{\mathbf{x}_\omega} - 2\widetilde{\mathbf{x}_\omega}\widetilde{\mathbf{x}_u}$.

A few higher order directional derivatives and gradients of the tangent operator on $SO(3)$ are further presented in A.1. One observes that the complexity of the formulae increases significantly at each differentiation step.

4.2 The special Euclidean group $SE(3)$

The set of Euclidean transformations (or rigid transformations) can be represented as the set of 4×4 transformations matrices:

$$SE(3) :- \left\{ \mathbf{H} = \mathcal{H}(\mathbf{R}, \mathbf{x}) = \begin{bmatrix} \mathbf{R} & \mathbf{x} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} : \mathbf{R} \in SO(3),\ \mathbf{x} \in \mathbb{R}^3 \right\} \tag{74}$$

In such a 4×4 matrix, $\mathbf{x}$ represents a translation and $\mathbf{R}$ represents a rotation. $SE(3)$ is a group since, firstly the composition of two elements yields an element of the group, which reads as follows $\mathbf{H}_1\mathbf{H}_2 = \mathcal{H}(\mathbf{R}_1, \mathbf{x}_1)\mathcal{H}(\mathbf{R}_2, \mathbf{x}_2) = \mathcal{H}(\mathbf{R}_1\mathbf{R}_2, \mathbf{x}_1 + \mathbf{R}_1\mathbf{x}_2)$. Secondly, the matrix product is associative. Thirdly, the neutral element is the 4×4 identity matrix, $\mathbf{I}_{4\times4}$. Finally, the inverse element of $\mathbf{H}$, denoted by $\mathbf{H}^{-1}$, is given by $\mathbf{H}^{-1} = \mathcal{H}(\mathbf{R}^T, -\mathbf{R}^T\mathbf{x})$. Also, $SE(3)$ is a matrix Lie group since the product and the inverse of a matrix are smooth operations.

The Lie algebra $\mathfrak{se}(3)$ is the tangent space of 4×4 matrices and is isomorphic to $\mathbb{R}^6$.

$$\mathfrak{se}(3) :- \left\{ \widetilde{\mathbf{h}} = \mathcal{V}(\widetilde{\mathbf{h}_\omega}, \mathbf{h}_u) = \begin{bmatrix} \widetilde{\mathbf{h}_\omega} & \mathbf{h}_u \\ \mathbf{0}_{1\times3} & 0 \end{bmatrix} : \mathbf{h} = \begin{bmatrix} \mathbf{h}_u \\ \mathbf{h}_\omega \end{bmatrix} \in \mathbb{R}^6,\ \widetilde{\mathbf{h}_\omega} \in \mathfrak{so}(3),\ \mathbf{h}_u \in \mathbb{R}^3 \right\} \tag{75}$$

The adjoint representation as defined in Eq. (10) is

$$\mathrm{Ad}_{\mathbf{H}}\,\mathbf{h} = \begin{bmatrix} \mathbf{R} & \widetilde{\mathbf{x}}\mathbf{R} \\ \mathbf{0}_{3\times3} & \mathbf{R} \end{bmatrix} \mathbf{h} \tag{76}$$

For $SE(3)$, the hat operator $\widehat{(\bullet)}$, defined by Eq. (14), and the check operator $\check{(\bullet)}$, defined by Eq. (15), are given by:

$$\widehat{\mathbf{h}} = \begin{bmatrix} \widetilde{\mathbf{h}_\omega} & \widetilde{\mathbf{h}_u} \\ \mathbf{0}_{3\times3} & \widetilde{\mathbf{h}_\omega} \end{bmatrix}, \quad \check{\mathbf{h}} = \begin{bmatrix} \mathbf{0}_{3\times3} & -\widetilde{\mathbf{h}_u} \\ -\widetilde{\mathbf{h}_u} & -\widetilde{\mathbf{h}_\omega} \end{bmatrix} \tag{77}$$

Notice that, for all $\mathbf{h} \in \mathbb{R}^6$, $\check{\mathbf{h}}^T = -\check{\mathbf{h}}$.

The closed form expression of the exponential map on $SE(3)$ can be obtained using a similar approach as for $SO(3)$. The powers of $\widetilde{\mathbf{h}} \in \mathfrak{se}(3)$ satisfy the property

$$\widetilde{\mathbf{h}}^4 = -\|\mathbf{h}_\omega\|^2 \widetilde{\mathbf{h}}^2 \tag{78}$$

which can be combined with Eq. (17) to obtain:

$$\exp_{SE(3)}(\widetilde{\mathbf{h}}) = \mathbf{I} + \widetilde{\mathbf{h}} + \left(\frac{1}{2!}\frac{\|\mathbf{h}_\omega\|^2}{\|\mathbf{h}_\omega\|^2} - \frac{1}{4!}\frac{\|\mathbf{h}_\omega\|^4}{\|\mathbf{h}_\omega\|^2} + \frac{1}{6!}\frac{\|\mathbf{h}_\omega\|^6}{\|\mathbf{h}_\omega\|^2} - \cdots \right) \widetilde{\mathbf{h}}^2 + \left(\frac{1}{3!}\frac{\|\mathbf{h}_\omega\|^3}{\|\mathbf{h}_\omega\|^3} - \frac{1}{5!}\frac{\|\mathbf{h}_\omega\|^5}{\|\mathbf{h}_\omega\|^3} + \frac{1}{7!}\frac{\|\mathbf{h}_\omega\|^7}{\|\mathbf{h}_\omega\|^3} - \cdots \right) \widetilde{\mathbf{h}}^3 \tag{79}$$

$$= \mathbf{I} + \widetilde{\mathbf{h}} + \frac{1}{\|\mathbf{h}_\omega\|^2}\left(1 - \cos(\|\mathbf{h}_\omega\|)\right)\widetilde{\mathbf{h}}^2 + \frac{1}{\|\mathbf{h}_\omega\|^3}\left(\|\mathbf{h}_\omega\| - \sin(\|\mathbf{h}_\omega\|)\right)\widetilde{\mathbf{h}}^3 \tag{80}$$

Using the auxiliary quantities $\alpha(\|\mathbf{h}_\omega\|)$ and $\beta(\|\mathbf{h}_\omega\|)$, we get the closed form:

$$\exp_{SE(3)}(\widetilde{\mathbf{h}}) = \mathbf{I} + \widetilde{\mathbf{h}} + \frac{1}{2}\beta\widetilde{\mathbf{h}}^2 + \left(\frac{1-\alpha}{\|\mathbf{h}_\omega\|^2}\right)\widetilde{\mathbf{h}}^3 \tag{81}$$

This expression can be developed as

$$\exp_{SE(3)}(\mathbf{h}) = \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} + \begin{bmatrix} \widetilde{\mathbf{h}_\omega} & \mathbf{h}_u \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} + \frac{1}{2}\beta \begin{bmatrix} \widetilde{\mathbf{h}_\omega}^2 & \widetilde{\mathbf{h}_\omega}\mathbf{h}_u \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} + \left(\frac{1-\alpha}{\|\mathbf{h}_\omega\|^2}\right) \begin{bmatrix} \widetilde{\mathbf{h}_\omega}^3 & \widetilde{\mathbf{h}_\omega}^2\mathbf{h}_u \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \tag{82}$$

$$= \begin{bmatrix} \mathbf{I}_{3\times3} + \widetilde{\mathbf{h}_\omega} + \frac{1}{2}\beta\widetilde{\mathbf{h}_\omega}^2 - \widetilde{\mathbf{h}_\omega} + \alpha\widetilde{\mathbf{h}_\omega} & (\mathbf{I}_{3\times3} + \frac{1}{2}\beta\widetilde{\mathbf{h}_\omega} + ((1-\alpha/\|\mathbf{h}_\omega\|^2)\widetilde{\mathbf{h}_\omega}^2)\mathbf{h}_u \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \tag{83}$$

where $\mathbf{I}_{3\times3} + \alpha\widetilde{\mathbf{h}_\omega} + \frac{1}{2}\beta\widetilde{\mathbf{h}_\omega}^2$ is equal to $\exp_{SE(3)}(\widetilde{\mathbf{h}_\omega})$ and $\mathbf{I}_{3\times3} + \frac{1}{2}\beta\widetilde{\mathbf{h}_\omega} + ((1-\alpha)/\|\mathbf{h}_\omega\|^2)\widetilde{\mathbf{h}_\omega}^2$ is equal to the transpose of $\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)$ in Eq. (70). Finally, the *closed form of the exponential map on* $SE(3)$ is obtained as:

$$\exp_{SE(3)}(\widetilde{\mathbf{h}}) = \begin{bmatrix} \exp_{SO(3)}(\widetilde{\mathbf{h}_\omega}) & \mathbf{T}^T_{SO(3)}(\mathbf{h}_\omega)\,\mathbf{h}_u \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix} \tag{84}$$

Similarly, the *closed form of the logarithm map on* $SE(3)$ is given by

$$\log_{SE(3)}(\mathcal{H}(\mathbf{R}, \mathbf{h}_u)) = \widetilde{\mathbf{h}} = \begin{bmatrix} \log_{SO(3)}(\mathbf{R}) & \mathbf{T}^{-T}_{SO(3)}(\mathbf{h}_\omega)\,\mathbf{h}_u \\ \mathbf{0}_{1\times3} & 0 \end{bmatrix} \tag{85}$$

in which the variable $\mathbf{h}_\omega$ is calculated using $\widetilde{\mathbf{h}_\omega} = \log_{SO(3)}(\mathbf{R})$. A slightly different formula to calculated $\log_{SE(3)}$ is presented in [25] chapter 4.4.2.

The *closed form of the tangent operator on* $SE(3)$ is obtained from Eq. (31) and C.2, using a similar approach as for the exponential map (Eq. (80)):

$$\mathbf{T}_{SE(3)}(\mathbf{h}) = \begin{bmatrix} \mathbf{T}_{SO(3)}(\mathbf{h}_\omega) & D\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\cdot\mathbf{h}_u \\ \mathbf{0}_{3\times3} & \mathbf{T}_{SO(3)}(\mathbf{h}_\omega) \end{bmatrix} \tag{86}$$

The *first directional derivative of the tangent operator* and the *gradient of the tangent operator multiplied by a constant vector* $\mathbf{k}$ on $SE(3)$ are obtained as:

$$D\mathbf{T}_{SE(3)}(\mathbf{h})\cdot\mathbf{y} = \begin{bmatrix} D\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\cdot\mathbf{y}_\omega & D_{\mathbf{h}}\left(D\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\cdot\mathbf{h}_u\right)\cdot\mathbf{y} \\ \mathbf{0}_{3\times3} & D\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\cdot\mathbf{y}_\omega \end{bmatrix} \tag{87}$$

$$\nabla_{\mathbf{h}}\left(\mathbf{T}_{SE(3)}(\mathbf{h})\,\mathbf{k}\right) = \begin{bmatrix} \nabla_{\mathbf{h}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\,\mathbf{k}_\omega\right) & \nabla_{\mathbf{h}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\,\mathbf{k}_\mathbf{u}\right) + \nabla_{\mathbf{h}_\omega}\left(D\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\cdot\mathbf{h}_u\,\mathbf{k}_\omega\right) \\ \mathbf{0}_{3\times3} & \nabla_{\mathbf{h}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{h}_\omega)\,\mathbf{k}_\omega\right) \end{bmatrix} \tag{88}$$

where $\mathbf{y} = [\mathbf{y}_u^T \, \mathbf{y}_\omega^T]^T$ and $\mathbf{k} = [\mathbf{k}_u^T \, \mathbf{k}_\omega^T]^T$. A few higher order directional derivatives and gradients of the tangent operator on $SE(3)$ are further presented in A.2.

The above operators on $SE(3)$ could alternatively be formulated using dual number representations and the principle of transference. Indeed, this principle states that all mathematical formulae for rotations can be extended to rigid body motions by replacing real variables by their dual counterparts. This idea was for example exploited in [38, 39].

5 Influence of numerical errors

Let us analyze the numerical errors resulting from the sequence in which the aforementioned mathematical operations are performed and stored in the computer's memory. These errors have mainly two sources, the round-off and the truncation errors, which are discussed below.

5.1 Round-off error of the closed form computation

In the floating-point representation of a real number $x \pm M \times 2^e$, the mantissa M and the exponent e are represented in binary format using a limited number of bits. For example, the double-precision floating-point format (float64) relies on 53 bits for the mantissa, which limits the machine precision to $2^{-53} = 1.11 \times 10^{-16}$, 11 bits for the exponent and 1 bit for the sign.

In order to illustrate the influence of round-off errors in our context, let us consider the evaluation of the trigonometric expression $f(\theta) = (1 - \cos\theta)$, which appears in the exponential map, when θ approaches 0. On the one hand, if the closed form expression is used, the cosine is first evaluated and then the subtraction is performed and the result will be noted $f_{closed}(\theta)$. The exact (reference) value of f is found by manipulating the series with an infinite number of terms which leads to $f_{ref}(\theta) = \theta^2/2! - \theta^4/4! + \cdots$. For a practical evaluation, this series should be truncated to k terms, however, k can be chosen so that the omitted terms are maintained in the order of the machine precision of 10^{-16}. The relative and absolute errors is then evaluated as

$$ErrorRel_{closed} = \left| \frac{f_{closed} - f_{ref}}{f_{ref}} \right|, \quad ErrorAbs_{closed} = |f_{closed} - f_{ref}| \tag{89}$$

2a shows the relative and absolute errors of $f_{closed}(\theta)$. Let us first analyze the error when θ is close to zero. When evaluating f_{closed}, we first evaluate $\cos\theta = 1 - \theta^2/2! + \theta^4/4! + \cdots$. For θ less than 10^{-8}, the second term of the series $\theta^2/2!$ is below the machine precision so that, due to round-off errors, the result in float64 format will be 1. Then the subtraction $(1 - \cos\theta)$ will give $f_{closed} = 0$. Therefore, due to the round-off error, f_{closed} is affected by a relative error of order $10^0 = 1$. If the value of θ is progressively increased from 10^{-8} to 10^0, the problem is transferred to the next term of the series and the relative error decreases with order 2 from 10^0 to 10^{-16}.

The same analysis can be done for the function $\beta(\theta) = 2(1 - \cos\theta)/\theta^2$. 2b shows the relative and absolute errors of the closed form of $\beta(\theta)$. The functions $\alpha(\theta)$ and $\gamma(\theta)$ do not suffer from the same problem, however, the problem appears at the level of the successive derivatives of the tangent operator. For example, the operators $D\alpha(\mathbf{x}_\omega) \cdot \mathbf{x}_u = \left(\frac{1-\alpha(\mathbf{x}_\omega)}{\|\mathbf{x}_\omega\|^2} - \frac{\beta(\mathbf{x}_\omega)}{2} \right) \mathbf{x}_\omega^T \mathbf{x}_u$ and $D_{\mathbf{x}_\omega}\gamma(\mathbf{x}_\omega) \cdot \mathbf{x}_u = \left(\frac{\gamma(\mathbf{x}_\omega)(1-\gamma(\mathbf{x}_\omega))}{\|\mathbf{x}_\omega\|^2} - \frac{1}{4} \right) \mathbf{x}_\omega^T \mathbf{x}_u$ respectively involve the functions $(1 - \alpha(\theta)) = (1 - \sin(\theta)/\theta)$ and $(1 - \gamma(\theta)) = (1 - (\theta/2)\cot(\theta/2))$ that exhibit the same kind of numerical problem. The analysis of the relative error of $\beta(\theta)$ is illustrative of the behavior of the relative error observed in the closed form evaluation of the exponential map, the tangent operator, its derivatives and its gradients on $SO(3)$ and $SE(3)$. Let us also mention that alternative expressions of the functions $\beta(\theta)$ and $\gamma(\theta)$ can be found in the literature [22], for which round-off errors would have a different influence. In this paper, the analysis is limited to the particular expressions given in Eq. (67).

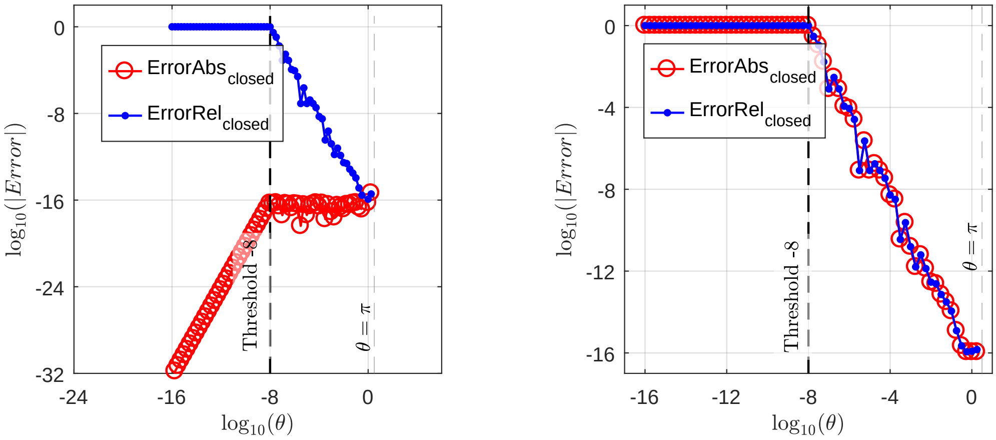
One possible strategy to avoid these difficulties is to enumerate the set of problematic trigonometric functions which are involved in these operators and to replace them by their series expansion. This approach was followed by Ritto-Corrêa and Camotim [21] for a limited number of operators on $SO(3)$. The generalization to all operators on $SO(3)$ and $SE(3)$ was not performed by these authors. Another approach, studied here, is to implement directly all operators in the form of truncated matricial series.

The exponential map on $SO(3)$ (Eq. (66)) involves $\beta(\mathbf{x}_\omega)$ multiplied by $\widetilde{\mathbf{x}_\omega}^2$; hence, the slope of its relative error is expected to be equal to the slope of the relative error on the function β reduced by two. The tangent operator on $SO(3)$ (Eq. (70)) involves $\beta(\mathbf{x}_\omega)$ multiplied by $\widetilde{\mathbf{x}_\omega}$; hence, the slope of its relative error is expected to be equal to the slope of the relative error on the function β reduced by one. The first derivative or gradient of the tangent operator (Eqs. (72) or (73)) involves $\beta(\mathbf{x}_\omega)$ without any multiplication by $\widetilde{\mathbf{x}_\omega}$; hence, the slope of its relative error is expected to be equal to the slope of the relative error of the function β. Considering the successive derivatives or gradients of the tangent operator, at each differentiation step, an additional division by $\|\mathbf{x}_\omega\|$ is performed which increases by one the slope of the relative error, i.e., the effect of round-off errors is amplified at each differentiation step.

The exponential map on $SE(3)$ group (Eq. (81)) also involves $\beta(\mathbf{h}_\omega)$ multiplied by $\widetilde{\mathbf{h}}^2$; hence, the slope of its relative error is equal to the slope of the relative error of the function β reduced by two. The influence of the round-off errors on the tangent operator on $SE(3)$ and its derivative/gradient follows the same behavior as described for $SO(3)$. In the results section, we will perform numerical tests to confirm this behavior of the relative error for $SO(3)$ and $SE(3)$.

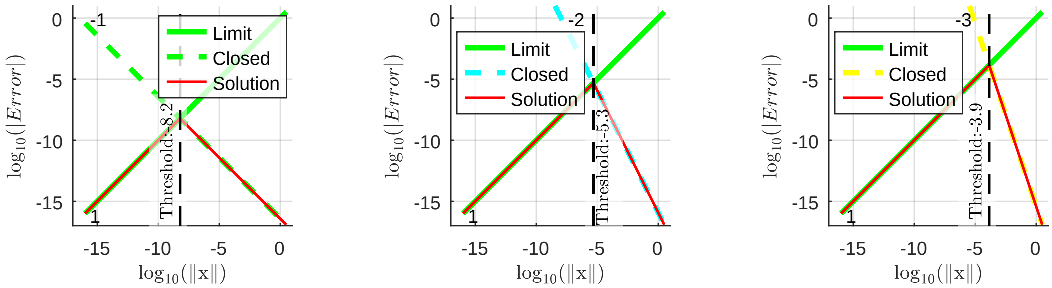
Based on the above analysis and on the numerical results shown below, we propose the following error estimate which captures the influence of round-off errors in the closed form expressions:

$$Error_{closed}(\|\mathbf{x}\|, s) = c_1 \left(\frac{\|\mathbf{x}\|}{\pi} \right)^s \tag{90}$$



(a) $(1-\cos(\theta))$ (b) $\beta = 2(1-\cos(\theta))/\theta^2$

Figure 2: Relative and absolute errors of two analytical expressions.



(a) Tangent operator $\mathbf{T}$. (b) First derivative $D\mathbf{T}$. (c) Second derivative $DD\mathbf{T}$.

Figure 3: Errors on the tangent operator and its derivatives using the closed form (dashed line), the limit value (solid green line) and the combined solution with the best threshold determined according to the proposed procedure (red line).

where $\|\mathbf{x}\|$ is $\|\mathbf{x}_\omega\|$ for $SO(3)$ or $\|\mathbf{h}\|$ for $SE(3)$, and $c_1 = 1.2 \times 10^{-17}$ for the float64 format. The value of the parameter s is 0 for the exponential map, $s = -1$ for the tangent operator, and $s = -1 - d$ for the derivative/gradient of the tangent operator where d is the number of differentiations of the tangent operator. For example, $s = -2$ for $D\mathbf{T}$ and $\nabla\mathbf{T}$, $s = -3$ for $DD\mathbf{T}$, $\nabla D\mathbf{T}$, $D\nabla\mathbf{T}$ and $\nabla\nabla\mathbf{T}$, and so forth.

When implementing the closed form expression, a threshold on $\|\mathbf{x}\|$ should be selected to switch between the closed form and its limit value at the origin. The best choice of this threshold can be made based on the above formula for the relative error. Let us evaluate the accuracy of the limit value. As it boils down to a zero-order truncation of the series expansion, the error estimate for the limit value takes the form $c_2 \|\mathbf{x}\|$. In our numerical experiments, we observe that $c_2 = 1$ provides a good approximation of the error for the tangent operator and its derivatives. Then, the optimal threshold ϵ for the variable $\|\mathbf{x}\|$ is evaluated by solving

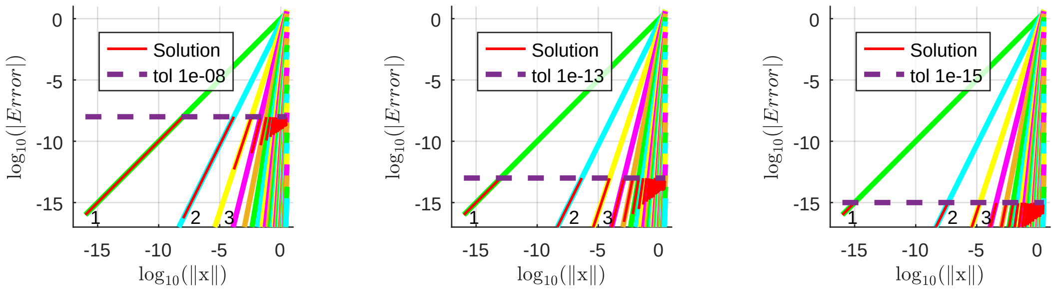
$$c_1 \left(\frac{\epsilon}{\pi}\right)^s = c_2\, \epsilon \tag{91}$$

For the tangent operator ($s = -1$), we obtain $\epsilon = 10^{-8.21}$. For the first order derivative or gradient ($s = -2$), the threshold is $\epsilon = 10^{-5.30}$. For derivatives from order 2 to 6, the thresholds are $10^{-3.86}$, $10^{-2.98}$, $10^{-2.40}$, $10^{-1.99}$ and $10^{-1.68}$, respectively. For all cases, the maximum value of the relative error is equal to $c_2\,\epsilon$. We thus expect rather high inaccuracies for higher order derivatives. For example, 3a, 3b and 3c show the tangent operator $\mathbf{T}$, its first derivative $D\mathbf{T}$ and its second derivative $DD\mathbf{T}$, respectively. Any other choice of threshold value would imply a greater relative error.

For the exponential map ($s = 0$), the closed form is not affected by round-off errors in the same way. We observe that the relative errors remain at the level of machine precision for $\|\mathbf{x}\| \geq 10^{-16}$. Since the limit value of the exponential map is below machine precision when $\|\mathbf{x}\| \leq 10^{-16}$, the threshold is selected as $\epsilon = 10^{-16}$.

5.2 Truncation error of the series computation

In order to evaluate a series with an infinite number of terms, the calculation should usually be approximated by truncation to a finite number of terms. The difference between the approximate solution and the exact solution is called the truncation error, and this error has the order of the first omitted term. Considering truncated series expansions limited to N terms, the exponential



(a) Tolerance of 10^{-8}. (b) Tolerance of 10^{-13}. (c) Tolerance of 10^{-15}.

Figure 4: Evolution of the relative truncation error on $D\mathbf{T}$ as a function $\|\mathbf{x}\|$ when the series is truncated to $N = N_2(tol, \|\mathbf{x}\|)$.

map, $\exp(\widetilde{\mathbf{x}})$ (Eq. (17)), the tangent operator $\mathbf{T}(\mathbf{x})$ (Eq. (31)) and its derivative $D\mathbf{T}(\mathbf{x})$ (Eq. (36)) are:

$$\exp(\widetilde{\mathbf{x}}) = \sum_{i=0}^{N-1} \frac{1}{i!}\tilde{\mathbf{x}}^i \quad + \mathcal{O}\left(\frac{\|\mathbf{x}\|^N}{N!}\right) \tag{92}$$

$$\mathbf{T}(\mathbf{x}) = \sum_{i=0}^{N-1} \frac{(-1)^i}{(1+i)!}\hat{\mathbf{x}}^i \quad + \mathcal{O}\left(\frac{\|\mathbf{x}\|^N}{(1+N)!}\right) \tag{93}$$

$$D\mathbf{T}(\mathbf{x})\cdot\mathbf{y} = \sum_{i=1}^{N} \frac{(-1)^i}{(1+i)!}\left(\hat{\mathbf{y}}\hat{\mathbf{x}}^{i-1} + \hat{\mathbf{x}}D\hat{\mathbf{x}}^{i-1}\cdot\mathbf{y}\right) \quad + \|\mathbf{y}\|\,\mathcal{O}\left(\frac{n_1(N)\,\|\mathbf{x}\|^N}{(2+N)!}\right) \tag{94}$$

$$D^d((\mathbf{T}(\mathbf{x})\cdot\mathbf{y}_1)...\cdot\mathbf{y}_d) = \sum_{i=d}^{N+d-1} \frac{(-1)^i}{(1+i)!}D^d((\hat{\mathbf{x}}^i\cdot\mathbf{y}_1)...\cdot\mathbf{y}_d) \quad + \|\mathbf{y}_1\|\|\mathbf{y}_2\|...\|\mathbf{y}_d\|\,\mathcal{O}\left(\frac{n_d(N)\,\|\mathbf{x}\|^N}{(1+d+N)!}\right) \tag{95}$$

where $n_d(N)$ is the number of terms recursively involved in the evaluation of the expression $D^d((\hat{\mathbf{x}}^N\cdot\mathbf{y}_1)...\cdot\mathbf{y}_d)$. A closer look reveals that $n_d(N) = (d+N)!/N!$ so that an error estimate for the tangent operator and its derivatives can be formulated as

$$Error^d_{series}(\|\mathbf{x}\|, N) = \frac{c_3\,(N+d)!}{N!\,(N+d+1)!}\|\mathbf{x}\|^N = \frac{c_3}{N!\,(N+d+1)}\|\mathbf{x}\|^N \tag{96}$$

where the constant c_3 should be fixed in an appropriate manner. For the exponential map, the error estimate is slightly different:

$$Error_{series}(\|\mathbf{x}\|, N) = \frac{c_4}{N!}\|\mathbf{x}\|^N \tag{97}$$

where the value of the constant c_4 is still to be chosen. In order to simplify the procedure, we propose to use the same error estimator given in Eq. (97) not only for the exponential map but also for the tangent operator and its derivative. Observing that $\|\mathbf{x}\|^N/(N!\,(N+d+1)) \leq \|\mathbf{x}\|^N/N!$, the error estimate $Error_{series}(\|\mathbf{x}\|, N)$ can be made less sharp than $Error^d_{series}(\|\mathbf{x}\|, N)$ provided a suitable choice of c_4, i.e., this simplified error estimate can always be made more conservative. In this error estimate, the numerator $\|\mathbf{x}\|^N$ reflects the convergence of the error with a slope defined by N and the denominator $N!$ induces a downward shift of the curve when N increases. According to Eq. (92), the value $c_4 = 1$ appears as a reasonable choice for the constant c_4, which will be confirmed by numerical experiments. A criterion on the truncation error can then be obtained by comparing this error estimate with a user-defined tolerance tol

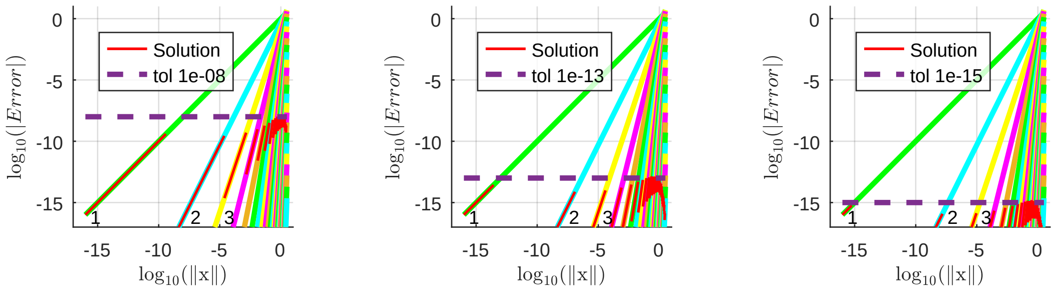
$$Error_{series}(\|\mathbf{x}\|, N) \leq tol \tag{98}$$

Based on this criterion, the series can be evaluated according to three different options. In the first and simplest option, a fixed number of terms $N = N_1(tol)$ is chosen to reach a given tolerance for the whole range of values of $\|\mathbf{x}\|$ from 0 to π. More precisely, $N_1(tol)$ computes the lowest integer such that the criterion in Eq. (98) is satisfied for all $\|\mathbf{x}\|$ in $[0, \pi[$. For the tight tolerance $tol = 2.7\times10^{-16}$, a number of $N = 28$ terms should be kept in the series.

In the second option, the number $N = N_2(tol, \|\mathbf{x}\|)$ is defined as the minimum integer such that the error estimate is below tol for a given $\|\mathbf{x}\|$. More precisely, $N_2(tol, \|\mathbf{x}\|)$ computes the lowest integer such that the criterion in Eq. (98) is satisfied. To reduce the computational cost, it is recommended to precompute the function N_2 before the simulation. For a chosen tolerance, a look-up table is thus established which provides the values of N_2 as a function of $\|\mathbf{x}\|$. 4a, 4b and reffig: Error D method 2 tol 15 show in solid red lines the errors of the derivative of the tangent operator using this approach with tolerances of 10^{-8}, 10^{-13} and 10^{-15}, respectively. The other color lines represent the error estimates for N varying from 1 to 29. Compared to option 1, a much lower number of terms is needed to satisfy the tolerance using option 2, i.e., $N_2(tol, \|\mathbf{x}\|) \leq N_1(tol)$.

In the third option, a simplified expression of the function N_2 is proposed $N_3(tol, \|\mathbf{x}\|) \simeq N_2(tol, \|\mathbf{x}\|)$. Firstly, the criterion in Eq. (98) is reformulated as $N\log_{10}(\|\mathbf{x}\|) - \log_{10}(N!) \leq \log_{10}(tol)$. Then, the logarithm of the factorial $\log_{10}(N!)$ is replaced by the linearized form $a_1N - a_0$, leading to $a_0 - \log_{10}(tol) \leq N\,(a_1 - \log_{10}(\|\mathbf{x}\|))$. In order to satisfy this criterion, the function N_3 is then formulated as

$$N_3(tol, \|\mathbf{x}\|) = ceil\left(\frac{a_0 - \log_{10}(tol)}{a_1 - \log_{10}(\|\mathbf{x}\|)}\right) \tag{99}$$



(a) Tolerance of 10^{-8}. (b) Tolerance of 10^{-13}. (c) Tolerance of 10^{-15}.

Figure 5: Evolution of the relative truncation error on $D\mathbf{T}$ as a function of $\|\mathbf{x}\|$ when the series is truncated to $N = N_3(tol, \|\mathbf{x}\|)$.)

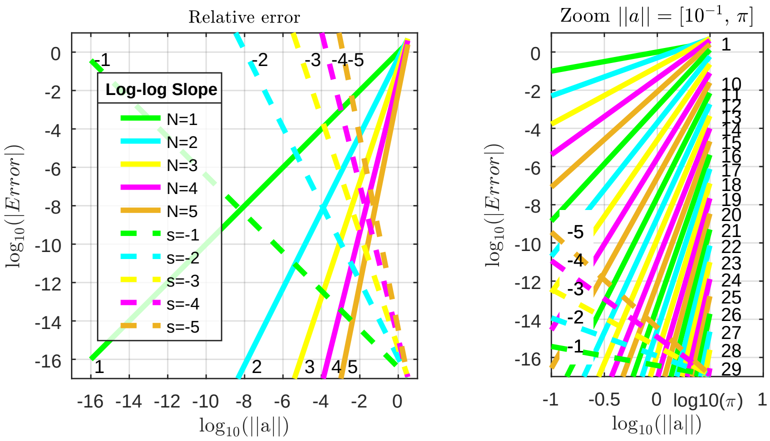


Figure 6: Estimates of the relative error for the tangent operator, its derivatives and gradients for the closed from (dashed lines) and for the truncated series with N terms (solid lines).

where the coefficients are selected as $a_0 = 4.5$ and $a_1 = 1.1$. The function $N_3(tol, \|\mathbf{x}\|)$ allows us to calculate N at a lower computational cost than in the second option. 5a, 5b, 5c show the errors on the truncated series using this third option. As the simplified equation is only an approximation, $N_3(tol, \|\mathbf{x}\|)$ is sometimes larger than $N_2(tol, \|\mathbf{x}\|)$ by one term, which indicates that the approximation is conservative but that the computational cost of the resulting series is only slightly increased compared to option 2. This computational cost of the series is always much smaller compared to option 1 because $N_3(tol, \|\mathbf{x}\|) \leq N_1(tol)$.

6 shows the error estimate of the closed form defined by Eq. (90) and the error estimate of the series form with N terms. The graph on the left shows five dashed lines representing the error estimate of the closed form with negative slopes $s = -1 - d$ and five solid lines representing the error estimate of the truncated series with N terms with positive slopes N. The graph on the right is a zoom including the error estimate of the truncated series with up to 29 terms. In the results section, numerical tests will be performed to compare these theoretical error estimates to the errors observed in numerical experiments for both the closed form and the series form.

6 Computational cost

In order to assess the efficiency of the formulations, the execution time and the number of floating-point operations (FLOP) will be compared in the result section. The execution time cost allows one to estimate the computational cost of a particular case. The number of FLOP provides an indication of the evolution of the computational cost with the key parameters of the method. However, we should keep in mind that those performance indicators are only indicative because the performance of a numerical method is also affected by factors such as the mathematical expression's coding, the chosen programming language, the software library, and the computer hardware being utilized. For the series form, these indicators are evaluated for a recursive implementation of the expressions.

The number of FLOP can also be estimated from the expressions of the operators in the code. The results are presented in 1. For the series form implemented in a recursive manner, a general expression can be obtained for any matrix Lie group. In this case, the number of FLOP depends on five parameters: N is the number of terms kept in the series, n is the size of the $\widetilde{(\bullet)}$ matrix, z_n is the number of zero elements of the $\widetilde{(\bullet)}$ matrix, k is the size of the $\widehat{(\bullet)}$ matrix and z_k is the number of zero elements of the $\widehat{(\bullet)}$ matrix. For example, $n = z_n = k = z_k = 3$ on $SO(3)$ and $n = 4$, $z_n = 4$, $k = 6$ and $z_k = 18$ on $SE(3)$. We consider that $(2k^3 - 2z_k k - k^2)$ FLOP are needed for a sparse matrix–matrix product, $(k^2 - z_k)$ FLOP are needed for the sum of matrices and for the scalar matrix multiplication, and 3 FLOP are needed to calculate the factorial and change the sign. Furthermore, the table presents the number of FLOP for the closed form both for $SO(3)$ and $SE(3)$, according to their implementation in our

Table 1: Number of FLOP needed for the evaluation of the series and closed forms.

Operator	FLOP: series form			FLOP: closed form	
	Matrix Lie groups	$SO(3)$	$SE(3)$	$SO(3)$	$SE(3)$
$\exp(\widetilde{\mathbf{a}})$	$\Big((2n^3 - 2z_n n - n^2) + 2(n^2 - z_n) + 3\Big)N$	$42N$	$107N$	109	220
$\mathbf{T}(\mathbf{a})$	$\Big((2k^3 - 2z_k k - k^2) + 2(k^2 - z_k) + 3\Big)N$	$42N$	$219N$	112	389
$D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}$	$\Big(3(2k^3 - 2z_k k - k^2) + 3(k^2 - z_k) + 3\Big)N$	$102N$	$597N$	293	1140
$\nabla(\mathbf{T}(\mathbf{a})\ \mathbf{c})$	$\Big(3(2k^3 - 2z_k k - k^2) + 4(k^2 - z_k) + 3\Big)N$	$108N$	$615N$	398	1766
$D_{\mathbf{a}}(D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d}$	$\Big(8(2k^3 - 2z_k k - k^2) + 6(k^2 - z_k) + 3\Big)N$	$255N$	$1551N$	755	3127
$\nabla_{\mathbf{a}}(D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}\ \mathbf{c})$	$\Big(8(2k^3 - 2z_k k - k^2) + 8(k^2 - z_k) + 3\Big)N$	$267N$	$11587N$	1167	5729
$D_{\mathbf{a}}(D_{\mathbf{a}}(D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}$	$\Big(20(2k^3 - 2z_k k - k^2) + 14(k^2 - z_k) + 3\Big)N$	$627N$		1941	
$\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}) \cdot \mathbf{d}\ \mathbf{c})$	$\Big(19(2k^3 - 2z_k k - k^2) + 17(k^2 - z_k) + 3\Big)N$	$618N$		3184	

MATLAB code. This evaluation is based on the code for counting the floating point operations shared by Qian on the MATLAB Central File Exchange. This code was also used to verify the number of FLOP of the series form. One observes very different numbers of FLOP for the closed forms on $SO(3)$ and $SE(3)$.

7 Results

The numerical experiments aim at comparing the series form with respect to the closed form of the $SO(3)$ and the $SE(3)$ groups in terms of numerical accuracy and computational cost for the exponential map, the tangent operator and their derivatives. The relative error of the series and closed forms is evaluated as a function of the rotation amplitude. The reference solution is obtained using the series form with enough terms to obtain convergence. It is worth noting that the errors depend on the norm $\|\mathbf{x}_\omega\|$ which represents the rotation amplitude, but not on each individual components of the vector $\mathbf{x}_\omega$.

The results were obtained using the following input vectors: $\mathbf{x}_u = [1, 1, 1]^T$, $\mathbf{y}_\omega = [1, 1, 1]^T$, $\mathbf{z}_\omega = [1, 1, 1]^T$, $\mathbf{c} = [1, 1, 1]^T$, $\mathbf{y} = [1, 1, 1, 1, 1, 1]^T$, $\mathbf{z} = [1, 1, 1, 1, 1, 1]^T$ and $\mathbf{k} = [1, 1, 1, 1, 1, 1]^T$. In our experience, the norm and the direction of these vectors have a limited influence on the accuracy, as long as they differ from the null vector. In contrast, the norm of the vector $\mathbf{x}_\omega$ has a direct influence on the accuracy.

7.1 Analysis of operators on the special orthogonal group $SO(3)$

7a and 7b show the relative error of the exponential map and the tangent operator with respect to $\|\mathbf{x}_\omega\|$ for the series form with N terms and the closed form on $SO(3)$. For the series form, N is varying from 1 to 10. For the closed form, according to the developments in 5.1, the threshold to switch to the limit value is fixed at $\|\mathbf{x}_\omega\| = 10^{-16}$ for the exponential map and $\|\mathbf{x}_\omega\| \leq 10^{-8.3}$ for the tangent operator. 6 has been placed in the background to permit a comparison with the theoretical error estimates established in 5. Overall, we observe that for both the exponential map and the tangent operator, the numerical results agree with the theoretical values.

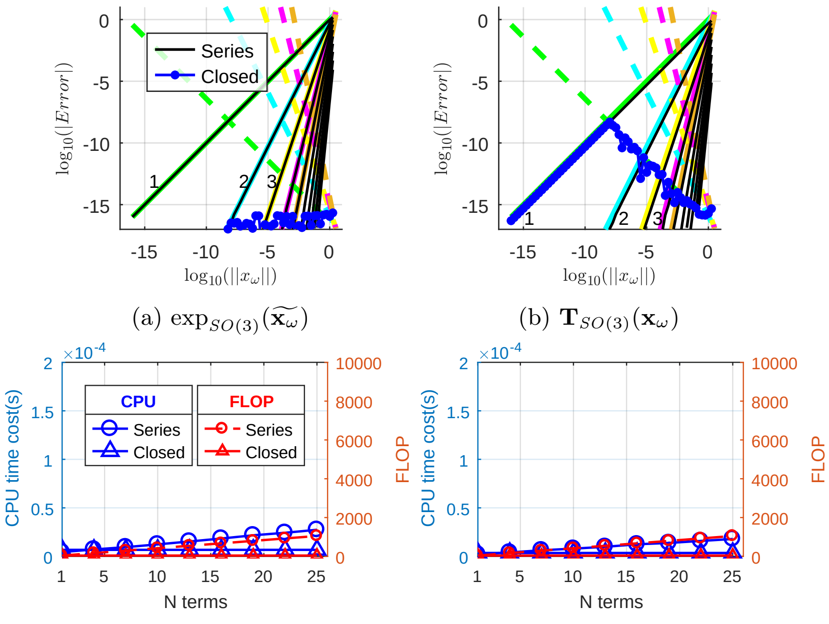
The CPU time is shown in blue in the left axis of 7c and 7d and the number of FLOP is shown in orange in the right axis. For the series form, both indicators increase with the number of terms N.

For the exponential map, these figures show that the closed form provide very accurate results at low computational cost for any rotation amplitude. In contrast, the series form is less competitive. Let us also mention that the truncation error of the series form may imply that the resulting matrix does not exactly satisfy the expected orthonormality property. The truncated series may not provide a matrix which is not exactly on $SO(3)$, which may induce additional numerical difficulties in the simulation code. For these reasons, the closed form should be recommended for the evaluation of the exponential map on $SO(3)$.

For the tangent operator, the error of the closed form is larger and shows a maximum value of the order of $10^{-8.3}$. The closed form may only be used if one is ready to accept such error amplitudes. If a more accurate solution is needed, it can be obtained using the truncated series expansion with an appropriate number of terms.

8a, 8b and 8c show the relative error of the first, second and third derivatives with respect to $\|\mathbf{x}_\omega\|$. 8d, 8e and 8f show the relative error of the gradients $\nabla(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\mathbf{c})$, $\nabla_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u\ \mathbf{c})$ and $\nabla_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega\ \mathbf{c})$. Again, for the series form N is varying from 1 to 10 and the theoretical error estimates of 6 are placed in the background. Overall, a good agreement is observed between the theoretical error estimates and the numerical results. This analysis confirms that, for the closed form, the numerical errors at small rotation amplitudes become more and more influential for higher order derivatives and gradients.

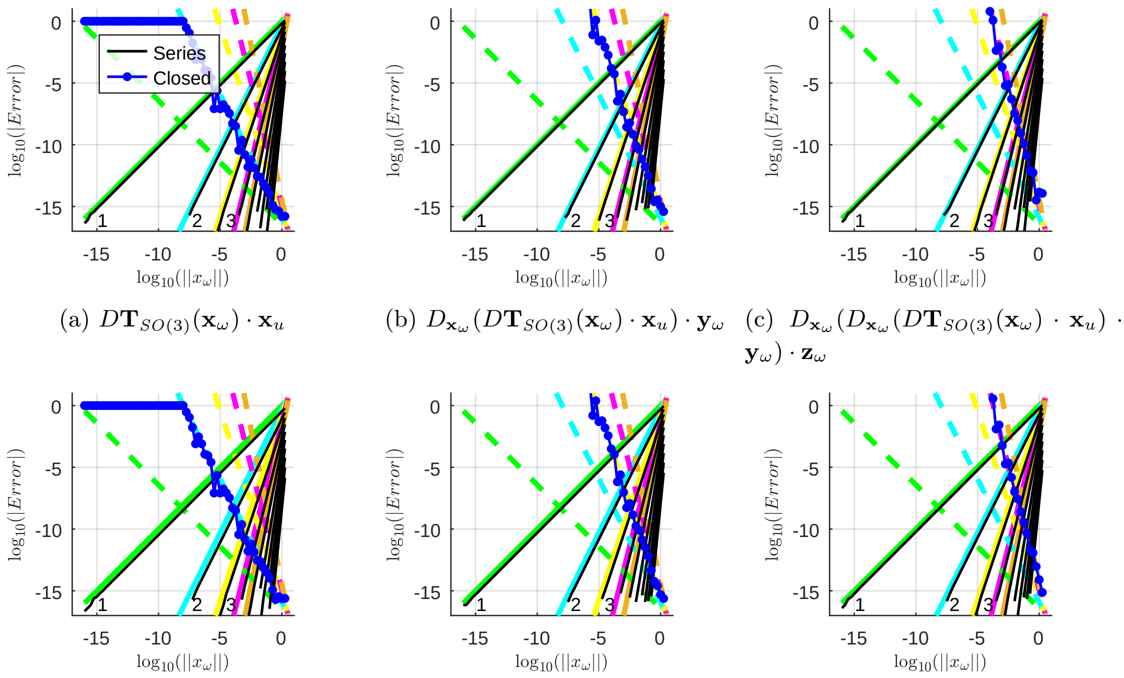
9a–9f show the CPU time cost and the FLOP of the derivatives and gradients of the tangent operator on $SO(3)$. For the series form, the CPU time cost follows a very similar trend as the FLOP number. For the closed form, the evolution of the FLOP and CPU time are still correlated but they do not follow precisely the same trend, which can be attributed to various additional factors that may affect the CPU time.



(a) $\exp_{SO(3)}(\widetilde{\mathbf{x}_\omega})$ (b) $\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)$

(c) $\exp_{SO(3)}(\widetilde{\mathbf{x}_\omega})$ (d) $\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)$

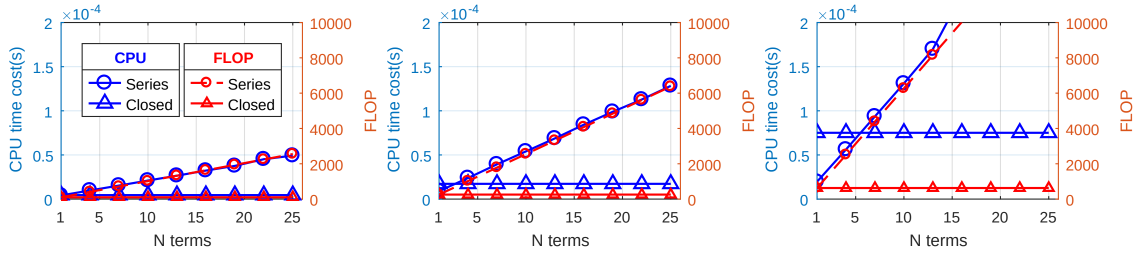
Figure 7: Accuracy and computational cost of the exponential map and its tangent operator on $SO(3)$. (a) and (b): Relative errors. (c) and (d): CPU times and FLOP.



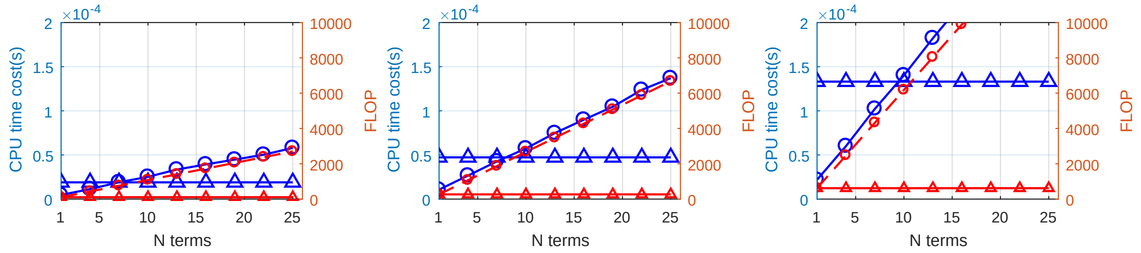
(a) $D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u$ (b) $D_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega$ (c) $D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega) \cdot \mathbf{z}_\omega$

(d) $\nabla(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\, \mathbf{c})$ (e) $\nabla_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u\, \mathbf{c})$ (f) $\nabla_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega\, \mathbf{c})$

Figure 8: Accuracy of the derivatives and gradients of the tangent operator on $SO(3)$.



(a) $D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u$ (b) $D_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega$ (c) $D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega) \cdot \mathbf{z}_\omega$



(d) $\nabla(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\, \mathbf{c})$ (e) $\nabla_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u\, \mathbf{c})$ (f) $\nabla_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega\, \mathbf{c})$

Figure 9: CPU time and number of FLOP of the derivatives and gradients of the tangent operator on $SO(3)$.

7.2 Analysis of operators on the special Euclidean group $SE(3)$

10a–10f show the relative error of the exponential map, the tangent operator, its derivatives and its gradients on $SE(3)$. The theoretical error estimates of 6 (5) are again plotted in the background. Overall, we observe that the numerical results are in agreement with the theoretical error estimates. The errors on the $SE(3)$ operators exhibit a similar behavior as for the $SO(3)$ operators.

11a to 11f show the CPU time and number of FLOP of the exponential map, the tangent operator, its derivatives and its gradients on $SE(3)$. As on $SO(3)$, the CPU time and the number of FLOP follow a similar trend for the series form. For the closed form, a lower correlation is observed. Generally, even though the CPU time of the exponential map for the closed form is lower than for the series form, it grows faster with the differentiation level. This means that the series form becomes more attractive than the closed form for the evaluation of higher order derivatives and gradients.

8 Strategic solution to calculate $SO(3)$ and $SE(3)$ with constraints on tolerance

After the analysis of the accuracy and computational cost, we propose a strategic solution to calculate derivatives of $SO(3)$ and $SE(3)$. The method exploits both the closed form and the series form, with a switching criterion based on a threshold on the rotation amplitude $\|\mathbf{x}\|$. This threshold is determined based on a chosen tolerance on the accuracy, i.e., the threshold is obtained by solving Eq. (90) for $\|Error_{closed}\|$ equal to the chosen tolerance. For values of $\|\mathbf{x}\|$ above the threshold, the closed form is used and for values below the threshold, the series form is used. The number of terms N of the series form can be calculated through either Eq. (97) or Eq. (99), as discussed in 5.2.

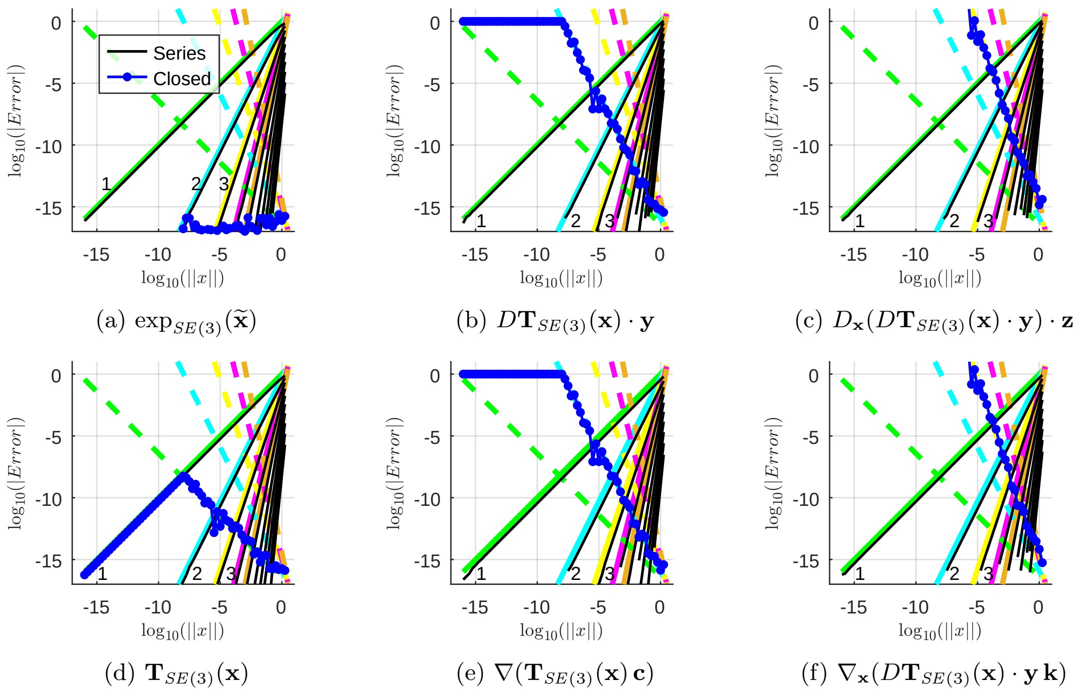
12a illustrates the evaluation of the first derivative of the tangent operator in order to reach a chosen tolerance ($tol = 10^{-13}$) based on the theoretical error estimates. In this case, the threshold on the rotation amplitude to switch between the closed form and the series form is defined by $\log_{10}(\|\mathbf{x}\|) = -1.5$. 12b and 12c show the errors observed on the derivative of the tangent operator on $SO(3)$ and $SE(3)$ using the strategic solution. These figures illustrate how the compromise between computational cost and accuracy is established based on a chosen tolerance.

9 Logarithm and inverse tangent operator

In this section, the analysis is extended to the logarithm, $\log(\mathbf{Q})$, the inverse tangent operator, $\mathbf{T}^{-1}(\mathbf{x})$, and its derivatives.

The logarithm map can be expressed both in closed form and series form. However, the series form does not always converge. Let us refer to the convergence criterion mentioned at the end of 2. On $SO(3)$, since a rotation matrix $\mathbf{R}$ is diagonalizable, the convergence is thus guaranteed if its eigenvalues λ_i satisfy $|\lambda_i - 1| < 1$. One can verify that this condition can be expressed in terms of the rotation amplitude as $\|\mathbf{x}_\omega\| < \pi/3$.

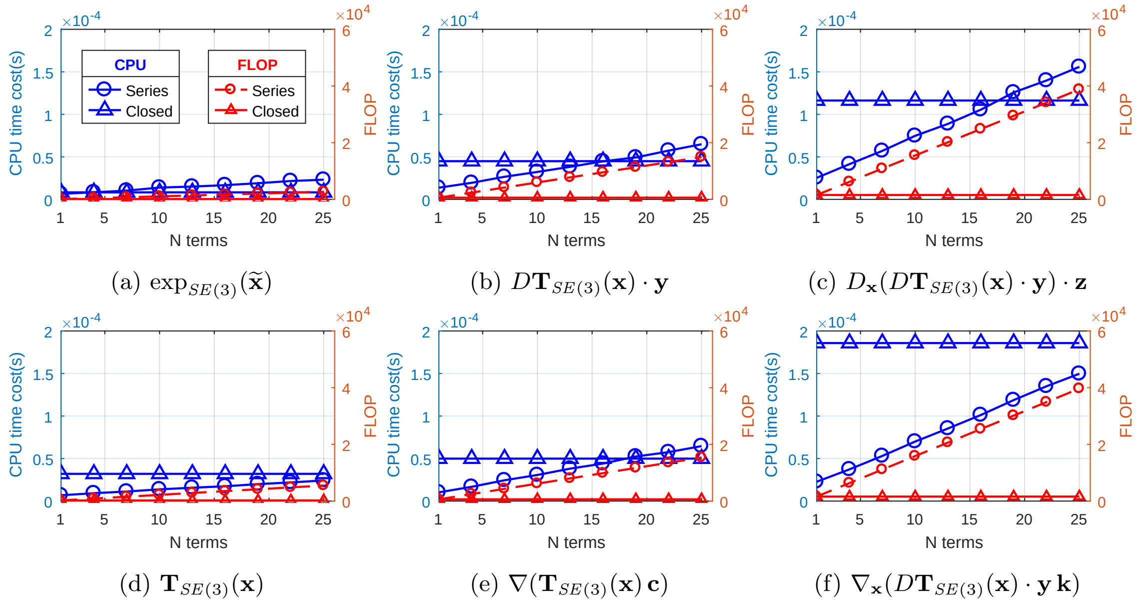
We propose to check the accuracy of the logarithm map by verifying the error on the condition $\exp(\log(\mathbf{Q})) = \mathbf{Q}$. In that expression, the exponential map is evaluated accurately either using the closed form or the series expansion with 100 terms. The errors for the logarithm map on $SO(3)$ and $SE(3)$ are presented in 13a and 13b, respectively. The series form is evaluated with 100 terms, which induces a significant computational cost in comparison to the closed form calculation. It can be observed that,



(a) $\exp_{SE(3)}(\widetilde{\mathbf{x}})$ (b) $D\mathbf{T}_{SE(3)}(\mathbf{x}) \cdot \mathbf{y}$ (c) $D_{\mathbf{x}}(D\mathbf{T}_{SE(3)}(\mathbf{x}) \cdot \mathbf{y}) \cdot \mathbf{z}$

(d) $\mathbf{T}_{SE(3)}(\mathbf{x})$ (e) $\nabla(\mathbf{T}_{SE(3)}(\mathbf{x})\, \mathbf{c})$ (f) $\nabla_{\mathbf{x}}(D\mathbf{T}_{SE(3)}(\mathbf{x}) \cdot \mathbf{y}\, \mathbf{k})$

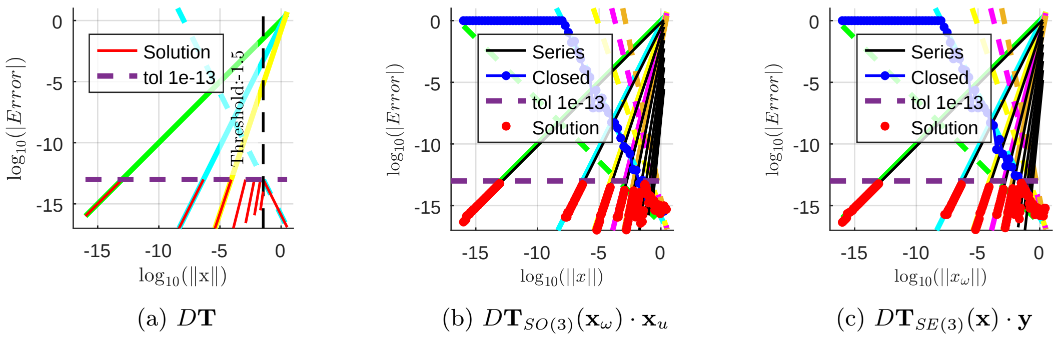
Figure 10: Accuracy of the exponential map, tangent operator, its derivatives and its gradients on $SE(3)$.



(a) $\exp_{SE(3)}(\widetilde{\mathbf{x}})$ (b) $D\mathbf{T}_{SE(3)}(\mathbf{x}) \cdot \mathbf{y}$ (c) $D_{\mathbf{x}}(D\mathbf{T}_{SE(3)}(\mathbf{x}) \cdot \mathbf{y}) \cdot \mathbf{z}$

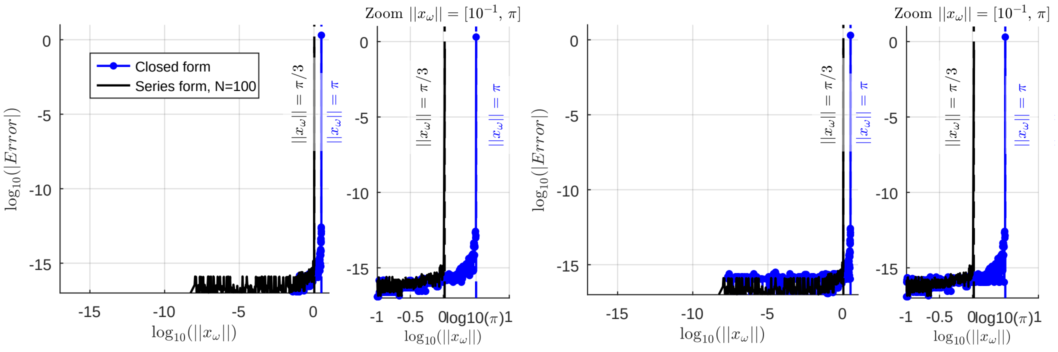
(d) $\mathbf{T}_{SE(3)}(\mathbf{x})$ (e) $\nabla(\mathbf{T}_{SE(3)}(\mathbf{x})\, \mathbf{c})$ (f) $\nabla_{\mathbf{x}}(D\mathbf{T}_{SE(3)}(\mathbf{x}) \cdot \mathbf{y}\, \mathbf{k})$

Figure 11: CPU time and number of FLOP of the exponential map, tangent operator, its derivatives and its gradients on $SE(3)$.



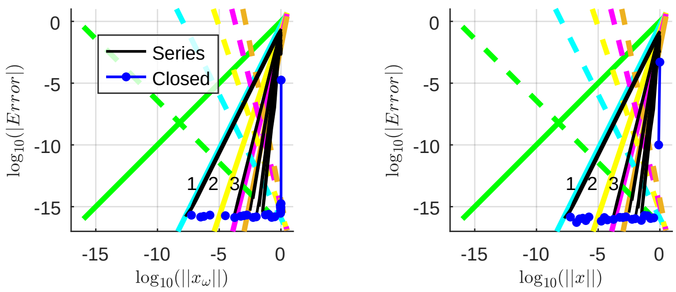
(a) $D\mathbf{T}$ (b) $D\mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \cdot \mathbf{x}_u$ (c) $D\mathbf{T}_{SE(3)}(\mathbf{x}) \cdot \mathbf{y}$

Figure 12: The error of the directional derivative of the tangent operator as a function of $\|\mathbf{x}\|$. (a) Theoretical error estimates, (b) error on $SO(3)$ and (c) error on $SE(3)$. The red line is the strategic solution, the blue line is the closed form and the black line is the series form with N varying from 1 up to 7. The results of 6 are partially plotted in the background.

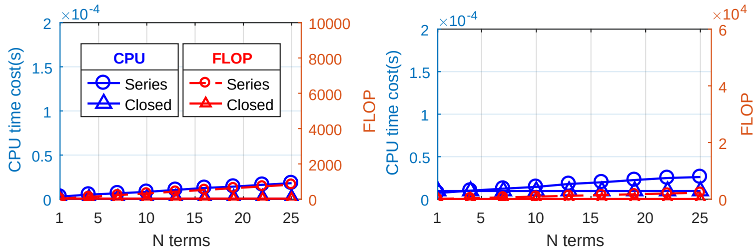


(a) $\mathbf{R}(\mathbf{x}_\omega) - \exp_{SO(3)}(\log_{SO(3)}(\mathbf{R}(\mathbf{x}_\omega)))$ (b) $\mathbf{H}(\mathbf{x}) - \exp_{SE(3)}(\log_{SE(3)}(\mathbf{H}(\mathbf{x})))$

Figure 13: Error of the logarithm map on $SO(3)$ and $SE(3)$.



(a) $\log_{SO(3)}(\|\mathbf{x}_\omega\|)$ (b) $\log_{SE(3)}(\|\mathbf{x}_\omega\|)$



(c) $\log_{SO(3)}(\|\mathbf{x}_\omega\|)$ (d) $\log_{SE(3)}(\|\mathbf{x}_\omega\|)$

Figure 14: Accuracy and computational cost of the logarithm on $SO(3)$ and $SE(3)$. (a), (b): Relative errors. (c), (d): CPU times and FLOP.

for both Lie groups $SO(3)$ and $SE(3)$, the series form is only applicable for amplitudes $\|\mathbf{x}_\omega\| < \pi/3$, as expected, while the closed form can be used in the wider range $[0, \pi[$.

In the interval $[0, \pi/3[$, the series form can be considered as a reference solution provided that a sufficient number of terms is kept. 14a and 14b show the influence of the number of terms N on the relative error of the logarithm map measured with respect to this reference solution. The evolution of the error for $\|\mathbf{x}_\omega\|$ approaching $\pi/3$ is irrelevant because the reference solution becomes invalid. 14c anf 14d show the CPU time cost and the FLOP of the logarithm map. In both cases, the closed form is more efficient. Based on the results of our analysis, it can be concluded that the closed form is more efficient and offers higher precision in a wider range compared to the series form for the evaluation of the logarithm map.

Let us consider the evaluation of the inverse tangent operator (Eq. (33)) and its derivatives (Eq. (39)). Following a similar analysis as in 5.2, an error estimate of the series truncated to N terms is given by

$$Error_{series;\mathbf{T}^{-1}}(\|\mathbf{x}\|, d, N) = \frac{c_5\, B_{d+N} \|\mathbf{x}\|^N}{N!} \tag{100}$$

where the constant c_5 can be chosen equal to 10. On the basis of this error estimate, we can follow the same three options as developed in 5.2. The first option for reaching a given tolerance is by selecting a fixed number of $N = N_1(tol)$ that covers the entire range of $\mathbf{x}$ from 0 to π. To achieve a tight tolerance $tol = 2.7 \times 10^{-16}$, it is recommended to keep a total of $N = 52$ terms in the series. For the second option, N is obtained by solving Eq. (100) to find $N = N_{2,\mathbf{T}^{-1}}(tol, \|\mathbf{x}\|, d)$. This number is defined as the minimum integer such that the error estimate is below the tol for a given $\|\mathbf{x}\|$. For the third option, a simplified expression is proposed for the solution of Eq. (100). The simplification is based on a linearization of the term $N!/B_{d+N} \simeq a_1 N - a_0 - b_1 d$. We obtain

$$N_{3;\mathbf{T}^{-1}}(tol, \|\mathbf{x}\|) = \text{ceil}\left(\frac{a_0 + b_1 d - \log_{10}(tol)}{a_1 - \log_{10}(\|\mathbf{x}\|)}\right) + 1 \tag{101}$$

with $a_1 = 0.8$, $a_0 = 0.53$, and $b_1 = 1$.

10 Conclusion

This paper presents and analyzes closed forms and series forms of various Lie group operators which are needed for the simulation of flexible multibody systems. In particular, the algorithms for the exponential map, the tangent operator and its derivatives on $SO(3)$ and $SE(3)$ are investigated. The analysis reveals that both closed forms and series forms can be combined to guarantee the numerical accuracy and the computational efficiency of the code.

The closed form, which is quite popular in the literature, relies on an analytic expression of the exponential map and of the tangent operator but it suffers from a singularity at the origin. Consequently, the evaluation is affected by increasing round-off errors when the rotation amplitude decreases. In a practical implementation, the analytical formula should be replaced by its limit value if the rotation or motion amplitude is below a given threshold. In this way, an accurate evaluation of the exponential map is obtained in the whole range of amplitudes with relative errors of the order of 10^{-16}. Regarding the tangent operator, for the particular closed form studied in this paper, the relative error exhibits a maximum value of $10^{-8.3}$ at the threshold. Two difficulties then appear when differentiating the tangent operator. Firstly, the analytical expressions become particularly complex, making their implementation in a code rather tedious. Secondly, the influence of the round-off errors for low amplitudes increases significantly at each differentiation step.

The series form relies on a truncated series expansion of the exponential map and tangent operator and does not suffer from any singularity. Furthermore, exploiting the Lie group theory, the same general expressions can be obtained for any matrix Lie group such as $SO(3)$ and $SE(3)$. Also, a unique recursive formula can be obtained for the evaluation of the high order derivatives and gradients of the tangent operator, its transpose, its inverse and its inverse transpose. This formula is one original contribution of the paper. Its implementation in a code is thereby much simpler than for the closed form. However, the result of the series form is affected by truncation errors whose influence increases with the rotation or motion amplitude. A criterion has been proposed to evaluate the number of terms to be kept in order to reach a given tolerance at a given rotation amplitude. This means that highly accurate expressions can be obtained even for large rotation amplitudes, but at a high computational cost.

An analysis of the number of FLOP and CPU time reveals that the closed form of the exponential map is generally less expensive than the series form. However, the computational cost of the two methods increases with the differentiation level due to the increased complexity of the operations in the code. Depending on the differentiated operator under study, the series form is often cheaper than the closed form if truncated to a small number of terms, but more expensive otherwise. In addition, the relation between the number of FLOP and the CPU time is rather complex and potentially influenced by many external factors. Generally, the cost of the operators remains in a similar range for the closed form and the series form.

Based on this analysis, we recommend to use the closed form for the evaluation of the exponential map as it provides a highly accurate expression for a low computational cost. The tangent operator may also be reasonably evaluated using the closed form if the tolerance on the numerical error is larger than $10^{-8.3}$. If a tighter tolerance is needed, then it will be necessary to switch to the series expansion whenever the condition in Eq. (90)($tol \leq c_1\pi/\|\mathbf{x}\|$) is not fulfilled. Regarding the derivatives of the tangent operator, the strategy may depend on the availability of the closed form in the code. If the closed form is available, then the strategic solution which combines the series form and the closed form can be considered. Otherwise, the series form with a number of terms evaluated using Eq. (97) or Eq. (99) can always reach a desired accuracy.

A similar analysis can be made for the series form and the closed form of the logarithm map and the inverse of the tangent operator. For the logarithm map, the convergence of the series is restricted to rotation amplitudes in the interval $[0, \pi/3[$ rad. In addition, a high number of terms is needed to reach convergence, which implies a high computational cost. In contrast, the closed form provides an accurate solution in the whole rotation range, $[0, \pi[$, with relative errors of the order of 10^{-16}. Therefore, it is recommended to compute the logarithm map according to the closed form as it provides a highly accurate expression for a low computational cost. The recommendation regarding the inverse tangent operator and its derivatives is similar to the recommendation made for the tangent operator and its inverse, even though the error estimate takes a different expression.

This work provides the basis for the accurate linearization of the equations of motion of a flexible multibody system, e.g., for the linearization of a geometrically exact finite element formulation on Lie group. This accurate linearization may improve the convergence of the Newton iterations in implicit time integration schemes. It is also essential for the sensitivity analysis and the gradient-based optimization of flexible multibody systems.

Acknowledgment: This work has received financial support from the EI-OPT and ORFI projects funded by the Walloon Region (Pôle Skywin) and from the Robotix Academy project funded by the Interreg Greater Region program, which are gratefully acknowledged. It has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 860124. The present paper only reflects the author's view. The European Commission and its Research Executive Agency (REA) are not responsible for any use that may be made of the information it contains.

A Derivatives of the tangent operator in closed form

A.1 Closed form of tangent operator on SO(3):

Let us anticipate here the expression of some directional derivatives of terms which shall appear in the tangent operator:

$$
\begin{array}{lll}
D_{\mathbf{x}_\omega}\|\mathbf{x}_\omega\|\cdot\mathbf{x}_u=\frac{1}{\|\mathbf{x}_\omega\|}\mathbf{x}_\omega^T\mathbf{x}_u & D_{\mathbf{x}_\omega}\frac{1}{\|\mathbf{x}_\omega\|}\cdot\mathbf{x}_u=-\frac{1}{\|\mathbf{x}_\omega\|^3}\mathbf{x}_\omega^T\mathbf{x}_u & D_{\mathbf{x}_\omega}\frac{1}{\|\mathbf{x}_\omega\|^2}\cdot\mathbf{x}_u=-\frac{2}{\|\mathbf{x}_\omega\|^4}\mathbf{x}_\omega^T\mathbf{x}_u \\
D_{\mathbf{x}_\omega}\frac{1}{\|\mathbf{x}_\omega\|^4}\cdot\mathbf{x}_u=-\frac{4}{\|\mathbf{x}_\omega\|^6}\mathbf{x}_\omega^T\mathbf{x}_u & D_{\mathbf{x}_\omega}\frac{1}{\|\mathbf{x}_\omega\|^6}\cdot\mathbf{x}_u=-\frac{6}{\|\mathbf{x}_\omega\|^8}\mathbf{x}_\omega^T\mathbf{x}_u & D_{\mathbf{x}_\omega}\frac{1}{\|\mathbf{x}_\omega\|^8}\cdot\mathbf{x}_u=-\frac{8}{\|\mathbf{x}_\omega\|^{10}}\mathbf{x}_\omega^T\mathbf{x}_u \\
D_{\mathbf{x}_\omega}\|\mathbf{x}_\omega\|^2\cdot\mathbf{x}_u=2\mathbf{x}_\omega^T\mathbf{x}_u & D_{\mathbf{x}_\omega}\|\mathbf{x}_\omega\|^3\cdot\mathbf{x}_u=3\|\mathbf{x}_\omega\|\mathbf{x}_\omega^T\mathbf{x}_u & D_{\mathbf{x}_\omega}\|\mathbf{x}_\omega\|^4\cdot\mathbf{x}_u=4\|\mathbf{x}_\omega\|^2\mathbf{x}_\omega^T\mathbf{x}_u \\
D_{\mathbf{x}_\omega}\widetilde{\mathbf{x}_\omega}\cdot\mathbf{x}_u=\widetilde{\mathbf{x}_u} & D_{\mathbf{x}_\omega}\widetilde{\mathbf{x}_\omega}^2\cdot\mathbf{x}_u=\widetilde{\mathbf{x}_\omega}\widetilde{\mathbf{x}_u}+\widetilde{\mathbf{x}_u}\widetilde{\mathbf{x}_\omega}=\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil & D_{\mathbf{x}_\omega}\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil\cdot\mathbf{y}_\omega=\lceil\mathbf{y}_\omega,\mathbf{x}_u\rceil \\
\nabla_{\mathbf{x}_\omega}(\widetilde{\mathbf{x}_\omega}\,\mathbf{c})=-\widetilde{\mathbf{c}} & \nabla_{\mathbf{x}_\omega}(\widetilde{\mathbf{x}_\omega}^2\,\mathbf{c})=\widetilde{\mathbf{c}}\,\widetilde{\mathbf{x}_\omega}-2\widetilde{\mathbf{x}_\omega}\,\widetilde{\mathbf{c}}=\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor & D_{\mathbf{x}_\omega}\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor\cdot\mathbf{y}_\omega=\lfloor\mathbf{y}_\omega,\mathbf{c}\rfloor
\end{array}
$$

where $\mathbf{x}_\omega=[x_{\omega1}\ x_{\omega2}\ x_{\omega3}]^T$, $\mathbf{x}_u=[x_{u1}\ x_{u2}\ x_{u3}]^T$, $\mathbf{y}_\omega=[y_{\omega1}\ y_{\omega2}\ y_{\omega3}]^T$, $\mathbf{z}_\omega=[z_{\omega1}\ z_{\omega2}\ z_{\omega3}]^T$, $\mathbf{c}=[c_1\ c_2\ c_3]^T$ and $\|\mathbf{x}_\omega\|=\sqrt{x_{\omega1}^2+x_{\omega2}^2+x_{\omega3}^2}$.

The directional derivative of the auxiliary quantities is given by:

$$\alpha(\mathbf{x}_\omega)=\frac{\sin(\|\mathbf{x}_\omega\|)}{\|\mathbf{x}_\omega\|},\qquad D_{\mathbf{x}_\omega}\alpha(\mathbf{x}_\omega)\cdot\mathbf{x}_u=\left(\frac{1-\alpha(\mathbf{x}_\omega)}{\|\mathbf{x}_\omega\|^2}-\frac{\beta(\mathbf{x}_\omega)}{2}\right)\mathbf{x}_\omega^T\mathbf{x}_u \tag{102}$$

$$\beta(\mathbf{x}_\omega)=2\frac{1-\cos(\|\mathbf{x}_\omega\|)}{\|\mathbf{x}_\omega\|^2},\qquad D_{\mathbf{x}_\omega}\beta(\mathbf{x}_\omega)\cdot\mathbf{x}_u=\frac{2}{\|\mathbf{x}_\omega\|^2}\left(\alpha(\mathbf{x}_\omega)-\beta(\mathbf{x}_\omega)\right)\mathbf{x}_\omega^T\mathbf{x}_u \tag{103}$$

$$\gamma(\mathbf{x}_\omega)=\frac{\|\mathbf{x}_\omega\|}{2}\cot(\frac{\|\mathbf{x}_\omega\|}{2}),\qquad D_{\mathbf{x}_\omega}\gamma(\mathbf{x}_\omega)\cdot\mathbf{x}_u=\left(\frac{\gamma(\mathbf{x}_\omega)(1-\gamma(\mathbf{x}_\omega))}{\|\mathbf{x}_\omega\|^2}-\frac{1}{4}\right)\mathbf{x}_\omega^T\mathbf{x}_u \tag{104}$$

$$D_{\mathbf{x}_\omega}\alpha(\mathbf{x}_\omega)^2\cdot\mathbf{x}_u=2\alpha(\mathbf{x}_\omega)\,D_{\mathbf{x}_\omega}\alpha(\mathbf{x}_\omega)\cdot\mathbf{x}_u,\qquad D_{\mathbf{x}_\omega}\alpha(\mathbf{x}_\omega)^3\cdot\mathbf{x}_u=3\alpha(\mathbf{x}_\omega)^2\,D_{\mathbf{x}_\omega}\alpha(\mathbf{x}_\omega)\cdot\mathbf{x}_u \tag{105}$$

The tangent operator and its transpose on $SO(3)$:

$$\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)=\mathbf{I}_{3\times3}\text{“}-\text{”}\frac{\beta(\mathbf{x}_\omega)}{2}\widetilde{\mathbf{x}_\omega}+\frac{1-\alpha(\mathbf{x}_\omega)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2\quad\text{and}\quad\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)=\mathbf{I}_{3\times3}\text{“}+\text{”}\frac{\beta(\mathbf{x}_\omega)}{2}\widetilde{\mathbf{x}_\omega}+\frac{1-\alpha(\mathbf{x}_\omega)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2 \tag{106}$$

only differ by the sign of one term, (“ − ” / “ + ”). For the sake of conciseness, only the derivatives of the operator will be presented. The sign change is indicated in the derivatives of the tangent operator by the symbols $\pm$ and $\mp$, where the upper sign is the sign of the operator, $\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)$, and the lower sign is the sign of the transpose, $\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)$. The auxiliary quantities $\alpha(\mathbf{x}_\omega)$, $\beta(\mathbf{x}_\omega)$ and $\gamma(\mathbf{x}_\omega)$ will be presented hereafter respectively by α, β and γ.

$$D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u=\mp\frac{\beta}{2}\widetilde{\mathbf{x}_u}\mp\frac{(\alpha-\beta)\,(\mathbf{x}_\omega^T\mathbf{x}_u)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}+\frac{1-\alpha}{\|\mathbf{x}_\omega\|^2}\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil+\frac{(\beta\,\|\mathbf{x}_\omega\|^2-6(1-\alpha))\,(\mathbf{x}_\omega^T\mathbf{x}_u)}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2 \tag{107}$$

$$\begin{aligned}
D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega&=\mp\frac{(\alpha-\beta)}{\|\mathbf{x}_\omega\|^2}\left((\mathbf{x}_\omega^T\mathbf{x}_u)\,\widetilde{\mathbf{y}_\omega}+(\mathbf{x}_u^T\mathbf{y}_\omega)\,\widetilde{\mathbf{x}_\omega}+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,\widetilde{\mathbf{x}_u}\right)\\
&\mp\frac{(8\,\beta-\beta\,\|\mathbf{x}_\omega\|^2-10\,\alpha+2)\,(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}+\frac{(1-\alpha)}{\|\mathbf{x}_\omega\|^2}\lceil\mathbf{y}_\omega,\mathbf{x}_u\rceil\\
&+\frac{(\beta\,\|\mathbf{x}_\omega\|^2+6\,(\alpha-1))}{2\,\|\mathbf{x}_\omega\|^4}\left((\mathbf{x}_u^T\mathbf{y}_\omega)\,\widetilde{\mathbf{x}_\omega}^2+(\mathbf{x}_\omega^T\mathbf{x}_u)\,\lceil\mathbf{x}_\omega,\mathbf{y}_\omega\rceil+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil\right)\\
&-\frac{(30\,(\alpha-1)+(7\,\beta-2\,\alpha)\,\|\mathbf{x}_\omega\|^2)(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)}{2\,\|\mathbf{x}_\omega\|^6}\widetilde{\mathbf{x}_\omega}^2
\end{aligned} \tag{108}$$

$$\begin{aligned}
D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega)\cdot\mathbf{z}_\omega&=\mp\frac{(\alpha-\beta)}{\|\mathbf{x}_\omega\|^2}\left((\mathbf{x}_u^T\mathbf{y}_\omega)\,\widetilde{\mathbf{z}_\omega}+(\mathbf{x}_u^T\mathbf{z}_\omega)\,\widetilde{\mathbf{y}_\omega}+(\mathbf{y}_\omega^T\mathbf{z}_\omega)\,\widetilde{\mathbf{x}_u}\right)\\
&\mp\frac{(66\,(\alpha-1)-48\,(\beta-1)-2\,\alpha\,\|\mathbf{x}_\omega\|^2+9\,\beta\,\|\mathbf{x}_\omega\|^2)(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)}{2\,\|\mathbf{x}_\omega\|^6}\widetilde{\mathbf{x}_\omega}\\
&\mp\frac{(10\,(1-\alpha)-\beta\,\|\mathbf{x}_\omega\|^2-8\,(1-\beta))}{2\,\|\mathbf{x}_\omega\|^4}\Bigg((\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,\widetilde{\mathbf{z}_\omega}+(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)\,\widetilde{\mathbf{y}_\omega}+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)\,\widetilde{\mathbf{x}_u}\\
&+\left((\mathbf{y}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)+(\mathbf{x}_u^T\mathbf{z}_\omega)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)+(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{y}_\omega^T\mathbf{z}_\omega)\right)\widetilde{\mathbf{x}_\omega}\Bigg)\\
&+\frac{(210\,(\alpha-1)-24\,\alpha\,\|\mathbf{x}_\omega\|^2+57\,\beta\,\|\mathbf{x}_\omega\|^2-\beta\,\|\mathbf{x}_\omega\|^4+2\,\|\mathbf{x}_\omega\|^2)(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)}{2\,\|\mathbf{x}_\omega\|^8}\widetilde{\mathbf{x}_\omega}^2\\
&-\frac{(30\,(\alpha-1)-2\,\alpha\,\|\mathbf{x}_\omega\|^2+7\,\beta\,\|\mathbf{x}_\omega\|^2)}{2\,\|\mathbf{x}_\omega\|^6}\Bigg(\left((\mathbf{x}_u^T\mathbf{z}_\omega)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)+(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{y}_\omega^T\mathbf{z}_\omega)+(\mathbf{y}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)\right)\widetilde{\mathbf{x}_\omega}^2\\
&+(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,\lceil\mathbf{x}_\omega,\mathbf{z}_\omega\rceil+(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)\,\lceil\mathbf{x}_\omega,\mathbf{y}_\omega\rceil+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,(\mathbf{x}_\omega^T\mathbf{z}_\omega)\,\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil\Bigg)\\
&+\frac{(\beta\,\|\mathbf{x}_\omega\|^2+6\,(\alpha-1))}{2\,\|\mathbf{x}_\omega\|^4}\Bigg((\mathbf{x}_\omega^T\mathbf{x}_u)\,\lceil\mathbf{y}_\omega,\mathbf{z}_\omega\rceil+(\mathbf{x}_\omega^T\mathbf{z}_\omega)\,\lceil\mathbf{x}_u,\mathbf{y}_\omega\rceil+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,\lceil\mathbf{x}_u,\mathbf{z}_\omega\rceil+(\mathbf{x}_u^T\,\mathbf{z}_\omega)\,\lceil\mathbf{x}_\omega,\mathbf{y}_\omega\rceil\\
&+(\mathbf{y}_\omega^T\mathbf{x}_u)\,\lceil\mathbf{x}_\omega,\mathbf{z}_\omega\rceil+(\mathbf{y}_\omega^T\mathbf{z}_\omega)\,\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil\Bigg)
\end{aligned} \tag{109}$$

The gradient of the tangent operator on $SO(3)$ when multiplied by a constant vector is obtained from the directional derivative after some manipulations.

$$
\begin{aligned}
D_{\mathbf{x}_\omega}(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\mathbf{c})\cdot\mathbf{b} =& D_{\mathbf{x}_\omega}\left(\mathbf{I}_{3\times3}\,\mathbf{c}\mp\frac{\beta}{2}\widetilde{\mathbf{x}_\omega}\,\mathbf{c}+\frac{1-\alpha}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2\,\mathbf{c}\right)\cdot\mathbf{b}\\
=&\mp\frac{\beta}{2}\widetilde{\mathbf{b}}\,\mathbf{c}\mp\frac{(\alpha-\beta)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}\,\mathbf{c}(\mathbf{x}_\omega^T\mathbf{b})+\frac{1-\alpha}{\|\mathbf{x}_\omega\|^2}(\widetilde{\mathbf{b}}\,\widetilde{\mathbf{x}_\omega}\,\mathbf{c}+\widetilde{\mathbf{x}_\omega}\,\widetilde{\mathbf{b}}\,\mathbf{c})+\frac{\beta\,\|\mathbf{x}_\omega\|^2+6(\alpha-1)}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2\,\mathbf{c}(\mathbf{x}_\omega^T\mathbf{b})\\
=&\left(\pm\frac{\beta}{2}\widetilde{\mathbf{c}}\mp\frac{(\alpha-\beta)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_\omega^T)+\frac{1-\alpha}{\|\mathbf{x}_\omega\|^2}\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor+\frac{\beta\,\|\mathbf{x}_\omega\|^2+6(\alpha-1)}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{x}_\omega^T)\right)\mathbf{b}
\end{aligned}\tag{110}
$$

Making use of the properties $\widetilde{\mathbf{b}}\,\mathbf{c}=-\widetilde{\mathbf{c}}\,\mathbf{b}$ and $\widetilde{(\widetilde{\mathbf{x}_\omega}\mathbf{c})}=\widetilde{\mathbf{x}_\omega}\,\widetilde{\mathbf{c}}-\widetilde{\mathbf{c}}\,\widetilde{\mathbf{x}_\omega}=[\widetilde{\mathbf{x}_\omega},\widetilde{\mathbf{c}}]$ on $SO(3)$, then $(\widetilde{\mathbf{b}}\,\widetilde{\mathbf{x}_\omega}\,\mathbf{c}+\widetilde{\mathbf{x}_\omega}\,\widetilde{\mathbf{b}}\,\mathbf{c})=(\widetilde{\mathbf{c}}\,\widetilde{\mathbf{x}_\omega}-2\widetilde{\mathbf{x}_\omega}\,\widetilde{\mathbf{c}})\mathbf{b}=\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor\,\mathbf{b}$. We get the gradient since $D_{\mathbf{x}_\omega}(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\mathbf{c})\cdot\mathbf{b}=\nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}\right)\,\mathbf{b}$.

$$
\nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}\right)=\pm\frac{\beta}{2}\widetilde{\mathbf{c}}\mp\frac{(\alpha-\beta)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_\omega^T)+\frac{1-\alpha}{\|\mathbf{x}_\omega\|^2}\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor+\frac{\beta\,\|\mathbf{x}_\omega\|^2+6(\alpha-1)}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{x}_\omega^T)\tag{111}
$$

$$
\begin{aligned}
\nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}\right)=&\mp\frac{(\alpha-\beta)}{\|\mathbf{x}_\omega\|^2}\,\left(\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_u^T)-(\mathbf{x}_\omega^T\mathbf{x}_u)\,\widetilde{\mathbf{c}}+\widetilde{\mathbf{x}_u}\,(\mathbf{c}\,\mathbf{x}_\omega^T)\right)\\
&\mp\frac{8\,\beta-\beta\,\|\mathbf{x}_\omega\|^2-10\,\alpha+2}{2\,\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_\omega^T)\,(\mathbf{x}_u\mathbf{x}_\omega^T)-\frac{30\,(\alpha-1)-2\,\alpha\,\|\mathbf{x}_\omega\|^2+7\,\beta\,\|\mathbf{x}_\omega\|^2}{2\,\|\mathbf{x}_\omega\|^6}\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{x}_\omega^T)(\mathbf{x}_u\mathbf{x}_\omega^T)\\
&-\frac{(\alpha-1)}{\|\mathbf{x}_\omega\|^2}\lfloor\mathbf{x}_u,\mathbf{c}\rfloor+\frac{\beta\,\|\mathbf{x}_\omega\|^2+6\,(\alpha-1)}{2\,\|\mathbf{x}_\omega\|^4}\left(\widetilde{\mathbf{x}_\omega}^2(\mathbf{c}\,\mathbf{x}_u^T)+\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil\,(\mathbf{c}\,\mathbf{x}_\omega^T)+(\mathbf{x}_\omega^T\mathbf{x}_u)\,\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor\right)
\end{aligned}\tag{112}
$$

$$
\begin{aligned}
\nabla_{\mathbf{x}_\omega}&\left(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega\,\mathbf{c}\right)=\mp\frac{(\beta-\alpha)}{\|\mathbf{x}_\omega\|^2}\,\left((\mathbf{y}_\omega^T\mathbf{x}_u)\,\widetilde{\mathbf{c}}-\widetilde{\mathbf{y}_\omega}\,(\mathbf{c}\,\mathbf{x}_u^T)-\widetilde{\mathbf{x}_u}\,(\mathbf{c}\,\mathbf{y}_\omega^T)\right)\\
&\mp\frac{\left(66\,\alpha-2\,\alpha\,\|\mathbf{x}_\omega\|^2-48\,\beta+9\,\beta\,\|\mathbf{x}_\omega\|^2-18\right)\,(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)}{2\,\|\mathbf{x}_\omega\|^6}\,\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_\omega^T)\\
&\mp\frac{\left(8\,\beta-\beta\,\|\mathbf{x}_\omega\|^2-10\,\alpha+2\right)}{2\,\|\mathbf{x}_\omega\|^4}\Bigg((\mathbf{x}_\omega^T\mathbf{x}_u)\widetilde{\mathbf{y}_\omega}\,(\mathbf{c}\,\mathbf{x}_\omega^T)+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,\widetilde{\mathbf{x}_u}(\mathbf{c}\,\mathbf{x}_\omega^T)+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_u^T)\\
&+(\mathbf{x}_\omega^T\mathbf{x}_u)\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{y}_\omega^T)+(\mathbf{y}_\omega^T\,\mathbf{x}_u)\widetilde{\mathbf{x}_\omega}\,(\mathbf{c}\,\mathbf{x}_\omega^T)-(\mathbf{x}_\omega^T\mathbf{x}_u)\,(\mathbf{x}_\omega^T\mathbf{y}_\omega)\,\widetilde{\mathbf{c}}\Bigg)+\frac{(\beta\|\mathbf{x}_\omega\|^2+6(\alpha-1))}{2\|\mathbf{x}_\omega\|^4}\Bigg(\lceil\mathbf{x}_\omega,\mathbf{y}_\omega\rceil(\mathbf{c}\,\mathbf{x}_u^T)\\
&+\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil(\mathbf{c}\,\mathbf{y}_\omega^T)+\lceil\mathbf{x}_u,\mathbf{y}_\omega\rceil(\mathbf{c}\,\mathbf{x}_\omega^T)+(\mathbf{y}_\omega^T\mathbf{x}_u)\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor+(\mathbf{x}_\omega^T\mathbf{x}_u)\lfloor\mathbf{y}_\omega,\mathbf{c}\rfloor+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\lfloor\mathbf{x}_u,\mathbf{c}\rfloor\Bigg)\\
&-\frac{\left(30\,\alpha-2\,\alpha\,\|\mathbf{x}_\omega\|^2+7\,\beta\,\|\mathbf{x}_\omega\|^2-30\right)}{2\,\|\mathbf{x}_\omega\|^6}\Bigg((\mathbf{x}_\omega^T\mathbf{x}_u)\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{y}_\omega^T)+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{x}_u^T)\\
&+(\mathbf{y}_\omega^T\mathbf{x}_u)\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{x}_\omega^T)+(\mathbf{x}_\omega^T\mathbf{x}_u)\lceil\mathbf{x}_\omega,\mathbf{y}_\omega\rceil\,(\mathbf{c}\,\mathbf{x}_\omega^T)+(\mathbf{x}_\omega^T\mathbf{y}_\omega)\lceil\mathbf{x}_\omega,\mathbf{x}_u\rceil\,(\mathbf{c}\,\mathbf{x}_\omega^T)+(\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)\lfloor\mathbf{x}_\omega,\mathbf{c}\rfloor\Bigg)\\
&+\frac{\left(210\,\alpha-24\,\alpha\,\|\mathbf{x}_\omega\|^2+57\,\beta\,\|\mathbf{x}_\omega\|^2-\beta\,\|\mathbf{x}_\omega\|^4+2\,\|\mathbf{x}_\omega\|^2-210\right)\,(\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)}{2\,\|\mathbf{x}_\omega\|^8}\widetilde{\mathbf{x}_\omega}^2\,(\mathbf{c}\,\mathbf{x}_\omega^T)
\end{aligned}\tag{113}
$$

Notice that when $\mathbf{x}_\omega$ is a zero vector, one has

$$\mathbf{T}_{SO(3)}(\mathbf{0})=\mathbf{I}_{3\times3}\qquad \|\mathbf{x}_\omega\|<10^{-8.00}\tag{114}$$

$$D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{0})\cdot\mathbf{x}_u=\mp\frac{1}{2}\widetilde{\mathbf{x}_u}\qquad \|\mathbf{x}_\omega\|<10^{-5.30}\tag{115}$$

$$D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{0})\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega=+\frac{1}{6}\lceil\mathbf{y}_\omega,\mathbf{x}_u\rceil\qquad \|\mathbf{x}_\omega\|<10^{-3.86}\tag{116}$$

$$D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{0})\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega)\cdot\mathbf{z}_\omega=\pm\frac{1}{12}\left((\mathbf{x}_u^T\mathbf{y}_\omega)\widetilde{\mathbf{z}_\omega}+(\mathbf{x}_u^T\mathbf{z}_\omega)\widetilde{\mathbf{y}_\omega}+(\mathbf{y}_\omega^T\mathbf{z}_\omega)\widetilde{\mathbf{x}_u}\right)\qquad \|\mathbf{x}_\omega\|<10^{-2.98}\tag{117}$$

$$\nabla_{\mathbf{x}_\omega}(\mathbf{T}_{SO(3)}(\mathbf{0})\,\mathbf{c})=\pm\frac{1}{2}\widetilde{\mathbf{c}}\qquad \|\mathbf{x}_\omega\|<10^{-5.20}\tag{118}$$

$$\nabla_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{0})\cdot\mathbf{x}_u\,\mathbf{c})=\frac{1}{6}\lfloor\mathbf{x}_u,\mathbf{c}\rfloor\qquad \|\mathbf{x}_\omega\|<10^{-3.78}\tag{119}$$

$$\nabla_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{0})\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega\,\mathbf{c})=\mp\frac{1}{12}\left((\mathbf{y}_\omega^T\mathbf{x}_u)\,\widetilde{\mathbf{c}}-\widetilde{\mathbf{y}_\omega}\,(\mathbf{c}\,\mathbf{x}_u^T)-\widetilde{\mathbf{x}_u}\,(\mathbf{c}\,\mathbf{y}_\omega^T)\right)\qquad \|\mathbf{x}_\omega\|<10^{-2.88}\tag{120}$$

where $\|\mathbf{x}_\omega\|<\epsilon$ is the best threshold.

The inverse of the tangent operator and its transpose on $SO(3)$:

$$\mathbf{T}_{SO(3)}^{-1}(\mathbf{x}_\omega)=\mathbf{I}_{3\times3}\text{``}+\text{''}\frac{1}{2}\widetilde{\mathbf{x}_\omega}+\frac{(1-\gamma)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2,\qquad \mathbf{T}_{SO(3)}^{-T}(\mathbf{x}_\omega)=\mathbf{I}_{3\times3}\text{``}-\text{''}\frac{1}{2}\widetilde{\mathbf{x}_\omega}+\frac{(1-\gamma)}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2\tag{121}$$

only differ by the sign of one term, (“ + ” / “ − ”). For the sake of conciseness, only the derivatives of the inverse operator will be presented.

$$
D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{x}_\omega) \cdot \mathbf{x}_u = \pm\frac{1}{2}\widetilde{\mathbf{x}_u} + \frac{1-\gamma}{\|\mathbf{x}_\omega\|^2}\lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil + \frac{(1/\beta+\gamma-2)(\mathbf{x}_\omega^T\mathbf{x}_u)}{\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2 \tag{122}
$$

$$
\begin{aligned}
D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{x}_\omega) \cdot \mathbf{x}_u\right) \cdot \mathbf{y}_\omega = &\left(\frac{(\gamma+2)(\gamma-1)}{\|\mathbf{x}_\omega\|^4} + \frac{1}{4\|\mathbf{x}_\omega\|^2}\right)(\mathbf{x}_\omega^T\mathbf{y}_\omega)\lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil + \frac{1-\gamma}{\|\mathbf{x}_\omega\|^2}\lceil \mathbf{y}_\omega, \mathbf{x}_u \rceil \\
&+ \frac{\gamma+1/\beta-2}{\|\mathbf{x}_\omega\|^4}\left((\mathbf{x}_u^T\,\mathbf{y}_\omega)\widetilde{\mathbf{x}_\omega}^2 + (\mathbf{x}_\omega^T\mathbf{x}_u)\lceil \mathbf{x}_\omega, \mathbf{y}_\omega \rceil\right) - \frac{(4\gamma^2+12\gamma+\|\mathbf{x}_\omega\|^2-32+8(\beta+\alpha)/\beta^2)(\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)}{4\|\mathbf{x}_\omega\|^6}\widetilde{\mathbf{x}_\omega}^2
\end{aligned} \tag{123}
$$

$$
\begin{aligned}
D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega \cdot \mathbf{z}_\omega = &+\frac{(\gamma+1/\beta-2)}{\|\mathbf{x}_\omega\|^4}\left((\mathbf{x}_\omega^T\mathbf{x}_u)\lceil \mathbf{y}_\omega, \mathbf{z}_\omega \rceil + (\mathbf{y}_\omega^T\mathbf{x}_u)\lceil \mathbf{x}_\omega, \mathbf{z}_\omega \rceil + (\mathbf{x}_u^T\mathbf{z}_\omega)\lceil \mathbf{x}_\omega, \mathbf{y}_\omega \rceil\right) \\
&+ \frac{(4\gamma^2+4\gamma+\|\mathbf{x}_\omega\|^2-8)}{4\|\mathbf{x}_\omega\|^4}\left((\mathbf{x}_\omega^T\mathbf{y}_\omega)\lceil \mathbf{x}_u, \mathbf{z}_\omega \rceil + (\mathbf{x}_\omega^T\mathbf{z}_\omega)\lceil \mathbf{y}_\omega, \mathbf{x}_u \rceil + (\mathbf{y}_\omega^T\mathbf{z}_\omega)\lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil\right) \\
&- \frac{(4\gamma^2+12\gamma-32+\|\mathbf{x}_\omega\|^2+8(\alpha+\beta)/\beta^2)}{4\|\mathbf{x}_\omega\|^6}\left((\mathbf{x}_\omega^T\mathbf{y}_\omega)(\mathbf{x}_u^T\mathbf{z}_\omega) + (\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{y}_\omega^T\mathbf{z}_\omega) + (\mathbf{y}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{z}_\omega)\right)\widetilde{\mathbf{x}_\omega}^2 \\
&+ \Bigg(\frac{(8\gamma^3+28\gamma^2+2\gamma\|\mathbf{x}_\omega\|^2+60\gamma-192+7\|\mathbf{x}_\omega\|^2+(32+4\|\mathbf{x}_\omega\|^2)/\beta+(40\alpha-8)/\beta^2+32\alpha^2/\beta^3)}{4\|\mathbf{x}_\omega\|^8} \\
&(\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)(\mathbf{x}_\omega^T\mathbf{z}_\omega)\widetilde{\mathbf{x}_\omega}^2\Bigg) - \frac{(8\gamma^3+12\gamma^2+2\gamma\|\mathbf{x}_\omega\|^2+12\gamma+3\|\mathbf{x}_\omega\|^2-32)}{4\|\mathbf{x}_\omega\|^6}\Bigg((\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)\lceil \mathbf{x}_\omega, \mathbf{z}_\omega \rceil \\
&+ (\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{z}_\omega)\lceil \mathbf{x}_\omega, \mathbf{y}_\omega \rceil + (\mathbf{x}_\omega^T\mathbf{y}_\omega)(\mathbf{x}_\omega^T\mathbf{z}_\omega)\lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil\Bigg)
\end{aligned} \tag{124}
$$

$$
\nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}^{-1}(\mathbf{x}_\omega)\,\mathbf{c}\right) = \mp\frac{1}{2}\widetilde{\mathbf{c}} + \frac{1-\gamma}{\|\mathbf{x}_\omega\|^2}\lfloor \mathbf{x}_\omega, \mathbf{c} \rfloor + \frac{((\gamma-1)(\gamma+2)+\|\mathbf{x}_\omega\|^2/4)}{\|\mathbf{x}_\omega\|^4}\widetilde{\mathbf{x}_\omega}^2(\mathbf{c}\,\mathbf{x}_\omega^T) \tag{125}
$$

$$
\begin{aligned}
\nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{x}_\omega) \cdot \mathbf{x}_u\,\mathbf{c}\right) = &+\frac{(1-\gamma)}{\|\mathbf{x}_\omega\|^2}\lfloor \mathbf{x}_u, \mathbf{c} \rfloor + \frac{(2\gamma+1)(4\gamma(1-\gamma)-\|\mathbf{x}_\omega\|^2)+2\|\mathbf{x}_\omega\|^2}{4\|\mathbf{x}_\omega\|^6}\widetilde{\mathbf{c}}^2(\mathbf{c}\,\mathbf{x}_\omega^T)(\mathbf{x}_u\mathbf{w}_\omega^T) \\
&+ \frac{4(\gamma+2)\ (\gamma-1)+\|\mathbf{x}_\omega\|^2}{4\|\mathbf{x}_\omega\|^4}\left(\lfloor \mathbf{x}_\omega, \mathbf{c} \rfloor(\mathbf{x}_u\mathbf{x}_\omega^T) + (\mathbf{x}_\omega^T\mathbf{x}_u)\lfloor \mathbf{x}_\omega, \mathbf{c} \rfloor + \widetilde{\mathbf{c}}^2(\mathbf{c}\,\mathbf{x}_u^T) - \frac{4}{\|\mathbf{x}_\omega\|^2}\widetilde{\mathbf{x}_\omega}^2(\mathbf{c}\,\mathbf{x}_\omega^T)(\mathbf{x}_u\mathbf{x}_\omega^T)\right)
\end{aligned} \tag{126}
$$

$$
\begin{aligned}
\nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{x}_\omega) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega\,\mathbf{c}\right) = &+\frac{24(\gamma+1/\beta-2)}{\|\mathbf{x}_\omega\|^8}(\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)\widetilde{\mathbf{x}_\omega}^2(\mathbf{c}\,\mathbf{x}_\omega^T) \\
&+ \frac{(\gamma+1/\beta-2)}{\|\mathbf{x}_\omega\|^4}\left((\mathbf{x}_\omega^T\mathbf{x}_u)\lfloor \mathbf{y}_\omega, \mathbf{c} \rfloor + (\mathbf{y}_\omega^T\mathbf{x}_u)\lfloor \mathbf{x}_\omega, \mathbf{c} \rfloor + \lceil \mathbf{x}_\omega, \mathbf{y}_\omega \rceil(\mathbf{c}\,\mathbf{x}_u^T)\right) \\
&- \frac{(8\gamma^3+12\gamma^2+2\gamma\|\mathbf{x}_\omega\|^2+12\gamma+3\|\mathbf{x}_\omega\|^2-32)}{4\|\mathbf{x}_\omega\|^6}\lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil(\mathbf{x}_\omega^T\mathbf{y}_\omega)(\mathbf{c}\,\mathbf{x}_\omega^T) \\
&+ \frac{(4\gamma^2+4\gamma+\|\mathbf{x}_\omega\|^2-8)}{4\|\mathbf{x}_\omega\|^4}\left(\lceil \mathbf{y}_\omega, \mathbf{x}_u \rceil(\mathbf{c}\,\mathbf{x}_\omega^T) + \lceil \mathbf{x}_\omega, \mathbf{x}_u \rceil(\mathbf{c}\,\mathbf{y}_\omega^T) + (\mathbf{x}_\omega^T\mathbf{y}_\omega)\lfloor \mathbf{x}_u, \mathbf{c} \rfloor\right) \\
&+ \Bigg(\frac{(\beta^3\ (8\gamma^3+28\gamma^2+2\gamma\|\mathbf{x}_\omega\|^2-36\gamma+7\|\mathbf{x}_\omega\|^2)+\beta^2\ (4\|\mathbf{x}_\omega\|^2-64)+\beta\ (40\alpha-8)+32\alpha^2)}{4\beta^3\|\mathbf{x}_\omega\|^8} \\
&(\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)\widetilde{\mathbf{x}_\omega}^2(\mathbf{c}\,\mathbf{x}_\omega^T)\Bigg) - \frac{4\gamma^2+12\gamma+\|\mathbf{x}_\omega\|^2-32+8(\beta+\alpha)/\beta^2}{4\|\mathbf{x}_\omega\|^6}\Bigg((\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{x}_\omega^T\mathbf{y}_\omega)\lfloor \mathbf{x}_\omega, \mathbf{c} \rfloor \\
&+ (\mathbf{x}_\omega^T\mathbf{x}_u)\lceil \mathbf{x}_\omega, \mathbf{y}_\omega \rceil(\mathbf{c}\,\mathbf{x}_\omega^T) + \widetilde{\mathbf{x}_\omega}^2\left((\mathbf{x}_\omega^T\mathbf{y}_\omega)(\mathbf{c}\,\mathbf{x}_u^T) + (\mathbf{x}_\omega^T\mathbf{x}_u)(\mathbf{c}\,\mathbf{y}_\omega^T) + (\mathbf{y}_\omega^T\mathbf{x}_u)(\mathbf{c}\,\mathbf{x}_\omega^T)\right)\Bigg)
\end{aligned} \tag{127}
$$

Notice that when the $\mathbf{x}_\omega$ is a zero vector, one has

$$
\mathbf{T}_{SO(3)}^{-1}(\mathbf{0}) = \mathbf{I}_{3\times 3} \qquad \|\mathbf{x}_\omega\| < 10^{-16.0} \tag{128}
$$

$$
D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{0}) \cdot \mathbf{x}_u = \pm\frac{1}{2}\widetilde{\mathbf{x}_u} \qquad \|\mathbf{x}_\omega\| < 10^{-3.84} \tag{129}
$$

$$
D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{0}) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega = \frac{1}{12}\lceil \mathbf{y}_\omega, \mathbf{x}_u \rceil \qquad \|\mathbf{x}_\omega\| < 10^{-2.32} \tag{130}
$$

$$
D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{0}) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega) \cdot \mathbf{z}_\omega = \mathbf{0}_{3\times 3} \qquad \|\mathbf{x}_\omega\| < 10^{-1.67} \tag{131}
$$

$$
\nabla_{\mathbf{x}_\omega}(\mathbf{T}_{SO(3)}^{-1}(\mathbf{0})\,\mathbf{c}) = \mp\frac{1}{2}\widetilde{\mathbf{c}} \qquad \|\mathbf{x}_\omega\| < 10^{-7.61} \tag{132}
$$

$$
\nabla_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{0}) \cdot \mathbf{x}_u\,\mathbf{c}) = \frac{1}{12}\lfloor \mathbf{x}_u, \mathbf{c} \rfloor \qquad \|\mathbf{x}_\omega\| < 10^{-3.36} \tag{133}
$$

$$
\nabla_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega} \mathbf{T}_{SO(3)}^{-1}(\mathbf{0}) \cdot \mathbf{x}_u) \cdot \mathbf{y}_\omega\,\mathbf{c}) = \mathbf{0}_{3\times 3} \qquad \|\mathbf{x}_\omega\| < 10^{-1.63} \tag{134}
$$

where $\|\mathbf{x}_\omega\| < \epsilon$ is the best threshold.

A.2 Closed form of tangent operator on $SE(3)$:

The tangent operator and its inverse on $SE(3)$:

$$\mathbf{T}_{SE(3)}(\mathbf{x}) = \begin{bmatrix} \mathbf{T}_{SO(3)}(\mathbf{x}_\omega) & D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u \\ \mathbf{0}_{3\times3} & \mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \end{bmatrix} \quad \text{and} \quad \mathbf{T}^{-1}_{SE(3)}(\mathbf{x}) = \begin{bmatrix} \mathbf{T}^{-1}_{SO(3)}(\mathbf{x}_\omega) & D_{\mathbf{x}_\omega}\mathbf{T}^{-1}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u \\ \mathbf{0}_{3\times3} & \mathbf{T}^{-1}_{SO(3)}(\mathbf{x}_\omega) \end{bmatrix} \tag{135}$$

have the same matrix structure, differing only that the tangent operators on $SO(3)$ are the inverse ones in the second case. For the sake of conciseness, only the derivatives of the operator will be presented. The vectors are $\mathbf{x} = [\mathbf{x}_u^T\ \mathbf{x}_\omega^T]^T$, $\mathbf{y} = [\mathbf{y}_u^T\ \mathbf{y}_\omega^T]^T$, $\mathbf{z} = [\mathbf{x}_u^T\ \mathbf{z}_\omega^T]^T$ and $\mathbf{c} = [\mathbf{c}_u^T\ \mathbf{c}_\omega^T]^T$.

$$D_{\mathbf{x}}\mathbf{T}_{SE(3)}(\mathbf{x})\cdot\mathbf{y} = \begin{bmatrix} D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega & D_{\mathbf{x}}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y} \\ \mathbf{0}_{3\times3} & D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega \end{bmatrix} \tag{136}$$

$$D_{\mathbf{x}}(D_{\mathbf{x}}\mathbf{T}_{SE(3)}(\mathbf{x})\cdot\mathbf{y})\cdot\mathbf{z} = \begin{bmatrix} D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\right)\cdot\mathbf{z}_\omega & D_{\mathbf{x}}\left(D_{\mathbf{x}}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y}\right)\cdot\mathbf{z} \\ \mathbf{0}_{3\times3} & D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\right)\cdot\mathbf{z}_\omega \end{bmatrix} \tag{137}$$

where $D_{\mathbf{x}}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}$ and $D_{\mathbf{x}}\left(D_{\mathbf{x}}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y}\right)\cdot\mathbf{z}$ are explicitly given by:

$$\begin{aligned} D_{\mathbf{x}}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y} =& D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega + D_{\mathbf{x}_u}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_u \\ =& D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega + D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_u \end{aligned} \tag{138}$$

$$\begin{aligned} D_{\mathbf{x}}\left(D_{\mathbf{x}}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y}\right)\cdot\mathbf{z} =& D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_\omega)\cdot\mathbf{z}_\omega + D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_u)\cdot\mathbf{z}_\omega \\ &+ D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega)\cdot\mathbf{z}_u \end{aligned} \tag{139}$$

where the expression $D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_u$ is equal to the expression $D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u$ (Eq. (107)) provided that $\mathbf{x}_u$ is replaced by $\mathbf{y}_u$. Similarly, $D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_u)\cdot\mathbf{z}_\omega$ and $D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega)\cdot\mathbf{z}_u$ is equal to $D_{\mathbf{x}_\omega}(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u)\cdot\mathbf{y}_w$ (Eq. (108)) with appropriate replacements.

Higher order directional derivatives of the tangent operator in $SE(3)$ follow the same logic as the derivatives above. The gradient of the tangent operator on $SE(3)$ when multiplied by a constant vector is obtained from the directional derivative after some manipulations.

$$\begin{aligned} D_{\mathbf{x}}\left(\mathbf{T}_{SE(3)}(\mathbf{x})\ \mathbf{c}\right)\cdot\mathbf{b} =& D_{\mathbf{x}}\left(\begin{bmatrix} \mathbf{T}_{SO(3)}(\mathbf{x}_\omega) & D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u \\ \mathbf{0}_{3\times3} & \mathbf{T}_{SO(3)}(\mathbf{x}_\omega) \end{bmatrix}\begin{bmatrix}\mathbf{c}_u \\ \mathbf{c}_\omega\end{bmatrix}\right)\cdot\begin{bmatrix}\mathbf{b}_u \\ \mathbf{b}_\omega\end{bmatrix} \\ =& \begin{bmatrix} D_{\mathbf{x}}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_u\right)\cdot\mathbf{b} + D_{\mathbf{x}}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}_\omega\right)\cdot\mathbf{b} \\ D_{\mathbf{x}}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right)\cdot\mathbf{b} \end{bmatrix} \\ =& \begin{bmatrix} D_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_u\right)\cdot\mathbf{b}_\omega + D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}_\omega\right)\cdot\mathbf{b}_\omega + D_{\mathbf{x}_u}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}_\omega\right)\cdot\mathbf{b}_u \\ D_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right)\cdot\mathbf{b}_\omega \end{bmatrix} \\ =& \begin{bmatrix} \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_u\right)\ \mathbf{b}_\omega + \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}_\omega\right)\ \mathbf{b}_\omega + \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right)\ \mathbf{b}_u \\ \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right)\ \mathbf{b}_\omega \end{bmatrix} \end{aligned} \tag{140}$$

Manipulating Eq.(140), we can write it as a matrix product

$$\begin{aligned} D_{\mathbf{x}}\left(\mathbf{T}_{SE(3)}(\mathbf{x})\ \mathbf{c}\right)\cdot\mathbf{b} =& \begin{bmatrix} \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right) & \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_u\right) + \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}_\omega\right) \\ \mathbf{0}_{3\times3} & \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right) \end{bmatrix}\begin{bmatrix}\mathbf{b}_u \\ \mathbf{b}_\omega\end{bmatrix} \\ =& \nabla_{\mathbf{x}}(\mathbf{T}_{SE(3)}(\mathbf{x})\ \mathbf{c})\ \mathbf{b} \end{aligned} \tag{141}$$

We obtain the gradient from Eq.(141). In the same way, other gradients can be obtained.

$$\nabla_{\mathbf{x}}(\mathbf{T}_{SE(3)}(\mathbf{x})\ \mathbf{c}) = \begin{bmatrix} \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right) & \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_u\right) + \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}_\omega\right) \\ \mathbf{0}_{3\times3} & \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right) \end{bmatrix} \tag{142}$$

$$\nabla_{\mathbf{x}}(D_{\mathbf{x}}\mathbf{T}_{SE(3)}(\mathbf{x})\cdot\mathbf{y}\ \mathbf{c}) = \begin{bmatrix} \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\,\mathbf{c}_\omega\right) & * \\ \mathbf{0}_{3\times3} & \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\,\mathbf{c}_\omega\right) \end{bmatrix} \tag{143}$$

where $* \equiv \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_u\,\mathbf{c}_\omega\right) + \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\,\mathbf{c}_u\right) + \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y}_\omega\,\mathbf{c}_\omega\right)$.

The tangent operator transposed and its inverse on $SE(3)$:

$$\mathbf{T}^T_{SE(3)}(\mathbf{x}) = \begin{bmatrix} \mathbf{T}^T_{SO(3)}(\mathbf{x}_\omega) & \mathbf{0}_{3\times3} \\ D_{\mathbf{x}_\omega}\mathbf{T}^T_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u & \mathbf{T}^T_{SO(3)}(\mathbf{x}_\omega) \end{bmatrix}, \quad \mathbf{T}^{-T}_{SE(3)}(\mathbf{x}) = \begin{bmatrix} \mathbf{T}^{-T}_{SO(3)}(\mathbf{x}_\omega) & \mathbf{0}_{3\times3} \\ D_{\mathbf{x}_\omega}\mathbf{T}^{-T}_{SO(3)}(\mathbf{x}_\omega)\cdot\mathbf{x}_u & \mathbf{T}^{-T}_{SO(3)}(\mathbf{x}_\omega) \end{bmatrix} \tag{144}$$

have the same matrix structure, differing only that the transposed tangent operators on $SO(3)$, $\mathbf{T}^T_{SO(3)}$, are the inverse ones in the second case, $\mathbf{T}_{SO(3)}^{-T}$. For the sake of conciseness, only the derivatives of the transposed operator will be presented.

$$D_{\mathbf{x}}\mathbf{T}_{SE(3)}^T(\mathbf{x})\cdot\mathbf{y} = \begin{bmatrix} D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega & \mathbf{0}_{3\times3} \\ D_{\mathbf{x}}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y} & D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega \end{bmatrix} \quad (145)$$

$$D_{\mathbf{x}}(D_{\mathbf{x}}\mathbf{T}_{SE(3)}^T(\mathbf{x})\cdot\mathbf{y})\cdot\mathbf{z} = \begin{bmatrix} D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\right)\cdot\mathbf{z}_\omega & \mathbf{0}_{3\times3} \\ D_{\mathbf{x}}\left(D_{\mathbf{x}}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y}\right)\cdot\mathbf{z} & D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\right)\cdot\mathbf{z}_\omega \end{bmatrix} \quad (146)$$

$$\nabla_{\mathbf{x}}(\mathbf{T}_{SE(3)}^T(\mathbf{x})\,\mathbf{c}) = \begin{bmatrix} \mathbf{0}_{3\times3} & \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\,\mathbf{c}_u\right) \\ \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\,\mathbf{c}_u\right) & \nabla_{\mathbf{x}_\omega}\left(\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\,\mathbf{c}_\omega\right)+\nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{x}_u\,\mathbf{c}_u\right) \end{bmatrix} \quad (147)$$

$$\nabla_{\mathbf{x}}(D_{\mathbf{x}}\mathbf{T}_{SE(3)}^T(\mathbf{x})\cdot\mathbf{y}\,\mathbf{c}) = \begin{bmatrix} \mathbf{0}_{3\times3} & \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\,\mathbf{c}_u\right) \\ \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\,\mathbf{c}_u\right) & ** \end{bmatrix} \quad (148)$$

where $** \equiv \nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_u\,\mathbf{c}_u\right)+\nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{y}_\omega\,\mathbf{c}_\omega\right)+\nabla_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\left(D_{\mathbf{x}_\omega}\mathbf{T}_{SO(3)}^T(\mathbf{x}_\omega)\cdot\mathbf{x}_u\right)\cdot\mathbf{y}_\omega\,\mathbf{c}_u\right)$.

B Derivatives of the tangent operator in series form

B.1 Derivatives of the hat operator $\widehat{(\bullet)}$

Let us consider the linear operator "*hat*", $\hat{\mathbf{a}}$, introduced in Section 2. The directional derivative of the matrix product $\hat{\mathbf{a}}^i = \underbrace{\hat{\mathbf{a}}\hat{\mathbf{a}}\cdots\hat{\mathbf{a}}}_{i\text{ times}}$ in the direction $\mathbf{b}$ is defined as a linear application in $\mathbf{b}$.

Directional derivatives only:

$$D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b} = \hat{\mathbf{b}}\hat{\mathbf{a}}^{i-1}+\hat{\mathbf{a}}D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b} \quad (149)$$

$$D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{d} = \hat{\mathbf{b}}D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{d}+\hat{\mathbf{d}}D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b}+\hat{\mathbf{a}}D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b})\cdot\mathbf{d} \quad (150)$$

$$\begin{aligned} D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}(\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{d})\cdot\mathbf{e} &= \hat{\mathbf{b}}D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{d})\cdot\mathbf{e}+\hat{\mathbf{d}}D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b})\cdot\mathbf{e}+\hat{\mathbf{e}}D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b})\cdot\mathbf{d} \\ &\quad +\hat{\mathbf{a}}D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b})\cdot\mathbf{d})\cdot\mathbf{e} \end{aligned} \quad (151)$$

Gradient only:

$$\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c}) = -\widehat{(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})}+\hat{\mathbf{a}}\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c}) \quad (152)$$

$$\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c})\,\mathbf{k}) = \hat{\mathbf{k}}\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})-\widehat{(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})\,\mathbf{k})}+\hat{\mathbf{a}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})\,\mathbf{k}) \quad (153)$$

$$\begin{aligned} \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) &= \hat{\mathbf{k}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})\,\mathbf{q})+\hat{\mathbf{q}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})\,\mathbf{k})-\widehat{(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q})} \\ &\quad +\hat{\mathbf{a}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) \end{aligned} \quad (154)$$

Second order gradient and directional derivative:

$$\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b}\,\mathbf{c}) = \hat{\mathbf{b}}\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})-\widehat{(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b}\,\mathbf{c})}+\hat{\mathbf{a}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b}\,\mathbf{c}) \quad (155)$$

$$\nabla_{\mathbf{b}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b}\,\mathbf{c}) = \nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c}) \quad (156)$$

$$D_{\mathbf{c}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c}))\cdot\mathbf{b} = \nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{b}) \quad (157)$$

$$\nabla_{\mathbf{c}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c})\,\mathbf{k}) = D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{k} \quad (158)$$

Third order gradient and directional derivative:

$$\begin{aligned} \nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{d}\,\mathbf{c}) &= \hat{\mathbf{b}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{d}\,\mathbf{c})+\hat{\mathbf{d}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b}\,\mathbf{c}) \\ &\quad -\widehat{(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b})\cdot\mathbf{d}\,\mathbf{c})}+\hat{\mathbf{a}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b})\cdot\mathbf{d}\,\mathbf{c}) \end{aligned} \quad (159)$$

$$\nabla_{\mathbf{b}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{d}\,\mathbf{c}) = \nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{d}\,\mathbf{c}) \quad (160)$$

$$\nabla_{\mathbf{a}}(D_{\mathbf{b}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{d}\,\mathbf{c}) = \nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{d}\,\mathbf{c}) \quad (161)$$

$$\nabla_{\mathbf{d}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{d}\,\mathbf{c}) = \nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b}\,\mathbf{c}) \quad (162)$$

$$\nabla_{\mathbf{d}}(D_{\mathbf{b}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{d}\,\mathbf{c}) = \nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c}) \quad (163)$$

$$\begin{aligned} \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b}\,\mathbf{c})\,\mathbf{k}) &= \hat{\mathbf{b}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^{i-1}\,\mathbf{c})\,\mathbf{k})+\hat{\mathbf{k}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b}\,\mathbf{c}) \\ &\quad -\widehat{(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b}\,\mathbf{c})\,\mathbf{k})}+\hat{\mathbf{a}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1}\cdot\mathbf{b}\,\mathbf{c})\,\mathbf{k}) \end{aligned} \quad (164)$$

$$\nabla_{\mathbf{b}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b}\,\mathbf{c})\,\mathbf{k}) = \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c})\,\mathbf{k}) \quad (165)$$

$$\nabla_{\mathbf{c}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b}\,\mathbf{c})\,\mathbf{k}) = D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b})\cdot\mathbf{k} \quad (166)$$

$$\nabla_{\mathbf{c}}(\nabla_{\mathbf{b}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{b}\,\mathbf{c})\,\mathbf{k}) = D_{\mathbf{a}}\hat{\mathbf{a}}^i\cdot\mathbf{k} \quad (167)$$

Fourth order gradient and directional derivative:

$$\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) = \hat{\mathbf{b}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) + \hat{\mathbf{d}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{e}\,\mathbf{c}) + \hat{\mathbf{e}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c}) - \widehat{(D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c})} + \hat{\mathbf{a}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) \tag{168}$$

$$\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) = \hat{\mathbf{b}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) + \hat{\mathbf{d}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k}) + \hat{\mathbf{k}}\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c}) - \widehat{(\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k})} + \hat{\mathbf{a}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) \tag{169}$$

$$\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) = \hat{\mathbf{b}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) + \hat{\mathbf{k}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{q}) + \hat{\mathbf{q}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k}) - \widehat{(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q})} + \hat{\mathbf{a}}\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^{i-1} \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) \tag{170}$$

$$\nabla_{\mathbf{b}}(D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) = \nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) \tag{171}$$

$$\nabla_{\mathbf{d}}(D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) = \nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{e}\,\mathbf{c}) \tag{172}$$

$$\nabla_{\mathbf{a}}(D_{\mathbf{b}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) = \nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) \tag{173}$$

$$\nabla_{\mathbf{d}}(D_{\mathbf{b}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e}\,\mathbf{c}) = \nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{e}\,\mathbf{c}) \tag{174}$$

$$\nabla_{\mathbf{b}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) = \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) \tag{175}$$

$$\nabla_{\mathbf{d}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) = \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) \tag{176}$$

$$\nabla_{\mathbf{c}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) = D_{\mathbf{a}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{k}) \tag{177}$$

$$\nabla_{\mathbf{d}}(\nabla_{\mathbf{b}}(D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}\,\mathbf{c})\,\mathbf{k}) = \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c})\,\mathbf{k}) \tag{178}$$

$$\nabla_{\mathbf{b}}(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) = \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) \tag{179}$$

$$\nabla_{\mathbf{c}}(\nabla_{\mathbf{b}}(\nabla_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) = \nabla_{\mathbf{a}}(\nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{k})\,\mathbf{q}) \tag{180}$$

$$\nabla_{\mathbf{k}}(\nabla_{\mathbf{c}}(\nabla_{\mathbf{b}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}\,\mathbf{c})\,\mathbf{k})\,\mathbf{q}) = \nabla_{\mathbf{a}}(\hat{\mathbf{a}}^i\,\mathbf{q}) \tag{181}$$

Let us note certain properties of the directional derivative. The commutativity of the derivative of the "*hat*" operator $\widehat{(\bullet)}$ takes the form $D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d} = D_{\mathbf{a}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{d}) \cdot \mathbf{b}$. The linearity of the derivative with respect to $\mathbf{b}$ results in, e.g. $D_{\mathbf{b}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d} = D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{d}$. Also due to the linearity of $\mathbf{b}$, a second derivative with respect to $\mathbf{b}$ is equal to zero $D_{\mathbf{b}}(D_{\mathbf{b}}(D_{\mathbf{a}}\hat{\mathbf{a}}^i \cdot \mathbf{b}) \cdot \mathbf{d}) \cdot \mathbf{e} = \mathbf{0}$.

B.2 Algorithm

In the following, we detail an algorithm for computing the series-form with N terms of both the tangent operator $\mathbf{T}(\mathbf{a})$ and the n order of directional derivative in the direction of the column vectors $\mathbf{b}_1, \mathbf{b}_2, \dots \mathbf{b}_n$ and m order of the gradient of tangent operator multiplied by the column vectors $\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_m$, $\nabla^m((D^n((\mathbf{T}(\mathbf{a}) \cdot \mathbf{b}_1) \dots \cdot \mathbf{b}_n)\mathbf{c}_1) \dots \mathbf{c}_m)$. It is worth noting that this tangent operator of the exponential map is valid in any matrix Lie group. The inputs are: $\mathbf{a}$ is a k-by-1 column vector. $\mathbf{b} = [\mathbf{b}_n \dots \mathbf{b}_2\,\mathbf{b}_1]$ is a k-by-n matrix. $\mathbf{c} = [\mathbf{c}_m \dots \mathbf{c}_2\,\mathbf{c}_1]$ is a k-by-m matrix. $N > 1$ is the truncation number (5.2). z_n is the length of the $\widehat{(\bullet)}$ matrix of the group. `@CheckT` function is the check operator transposed function. If `opInvers` $== 1$, the algorithm computes the inverse of the tangent operator, otherwise `opInvers` $== 0$. If `opTransp` $== 1$, the algorithm computes the transposed of the tangent operator and, in this case, `@Hat` function is the hat operator transposed function, otherwise (`opTransp` $== 0$) `@Hat` function is the hat operator function.

Algorithm $[GDT]$ = Gradient_Derivative_Tangent_Operator($\mathbf{a}, \mathbf{b}, \mathbf{c}, N$, @Hat, @CheckT, z_k, opTransp, opInvers)

```
bc := [b, c]
nm := columnSize(bc)
GDT := zerosMatrix(z_k, z_k)
GDa := zerosMatrix(z_k, z_k, 2^nm)
GDa(:, :, 1) := identityMatrix(z_k, z_k)
fact := 1
N_0 := 1
if nm == 0
    N_0 := 0
end if
mi := 0
if columnSize(c) > 1
    mi := columnSize(c) − 1
end if
for i = N_0 to N + nm − 1 do
    for e = nm to 1 do
        forf = (nm choose e) to 1 do
            v := subSet(nm, e, f)
            n := 0
            m := 0
            for g = 1 to e do
                if v(g) <= columnSize(b) + mi do
                    n += 1
                else
                    m += 1
                end if
            end for
            idx := 1 + sumPower2(nm − v)
            GDa(:, :, idx) := Hat(a) ∗ GDa(:, :, idx)
            for j = 1 to n and n + 2 to n + m do
                idx2 := 1 + sumPower2(nm − v([1:j − 1, j + 1:e]))
                GDa(:, :, idx) += Hat( bc(:, v(j) ) ∗ GDa(:, :, idx2)
            end do
            if m > 0  do
                idx2 := 1 + sumPower2(nm − v([1:n, n + 2:e]))
                if opTransp == 1  do
                    GDa(:, :, idx) += CheckT( GDa(:, :, idx2) ∗ bc(:, v(n + 1)) )
                else
                    GDa(:, :, idx) −= Hat( GDa(:, :, idx2) ∗ bc(:, v(n + 1)) )
                end if
            end if
        end for
    end for
    GDa(:, :, 1) := Hat(a) ∗ GDa(:, :, 1)
    if opInvers == 1  do
        GDT += (−1)^i ∗ GDa(:, :, end) ∗ B(i)/fact
        fact := fact ∗ (i + 1)
    else
        fact := fact ∗ (i + 1)
        GDT += (−1)^i ∗ GDa(:, :, end)/fact
    end if
end for
return GDT
```

where `columnSize(•)` function evaluates the column size of the input. The `zerosMatrix`(i, j, k) function creates an i-by-j-by-k array of zeros. The `identityMatrix`(i) function creates an i-by-i identity matrix. The `subSet`(n, e, f) generates a vector containing the f subset of $\{1, 2, \dots, n\}$ with e terms, for example, all distinct combinations (subsets) of the set $\{1, 2, 3\}$, $n = 3$, including the empty set are $\{\{\}; \{1\}; \{2\}; \{3\}; \{2, 3\}; \{1, 3\}; \{1, 2\}; \{1, 2, 3\}\}$, combinations with two terms $e = 2$ are $\{\{1, 2\}; \{1, 3\}; \{2, 3\}\}$, the first combination in this case $f = 1$ is $\{1, 2\}$, the second $f = 2$ is $\{1, 3\}$ and the third $f = 3$ is $\{2, 3\}$ (e.g. the matlab's nchoosek function). The `sumPower2`$(\mathbf{z})$ function computes the sum of the powers of two of each element of the array $\mathbf{z}$ as the exponent. The binomial distribution $\binom{n}{k} = \frac{n!}{k!(n-k)!}$. The $B(i)$ is a sequence of rational numbers, called the *Bernoulli numbers* of the first kind. It can be calculated using the recursive algorithm: $(i + 1)B(i) = -\sum_{k=0}^{i-1} \binom{i+1}{k} B(k)$ where $B(0) = 1$. We emphasize that for the computational cost study performed in 7, specifically optimized implementations for each derivative were used.

C Properties:

C.1 $SO(3)$ group

$$\widehat{\mathbf{x}_\omega}^i = \begin{cases} (-1)^{\frac{i+3}{2}} \|\mathbf{x}_\omega\|^{i-1}\widehat{\mathbf{x}_\omega} & \text{if } i \text{ is odd,} \\ (-1)^{\frac{i+2}{2}} \|\mathbf{x}_\omega\|^{i-2}\widehat{\mathbf{x}_\omega}^{\,2} & \text{otherwise.} \end{cases} \tag{182}$$

$$\widehat{\mathbf{x}_\omega}^3 = -\|\mathbf{x}_\omega\|^2\widehat{\mathbf{x}_\omega} \tag{183}$$

$$\widehat{\mathbf{x}_\omega}^4 = -\|\mathbf{x}_\omega\|^2\widehat{\mathbf{x}_\omega}^2 \tag{184}$$

C.2 $SE(3)$ group

$$\hat{\mathbf{x}}^i = \begin{cases} (-1)^{\frac{i+3}{2}} \|\mathbf{x}_\omega\|^{i-1}\widehat{\mathbf{x}} \;+ (-1)^{\frac{i+3}{2}}(i-1)\|\mathbf{x}_\omega\|^{i-3}(\mathbf{x}_\omega^T\mathbf{x}_u)\begin{bmatrix}\mathbf{0} & \widetilde{\mathbf{x}_\omega} \\ \mathbf{0} & \mathbf{0}\end{bmatrix} & \text{if } i \text{ is odd,} \\ (-1)^{\frac{i+2}{2}} \|\mathbf{x}_\omega\|^{i-2}\widehat{\mathbf{x}}^{\,2} + (-1)^{\frac{i+2}{2}}(i-2)\|\mathbf{x}_\omega\|^{i-4}(\mathbf{x}_\omega^T\mathbf{x}_u)\begin{bmatrix}\mathbf{0} & \widetilde{\mathbf{x}_\omega}^2 \\ \mathbf{0} & \mathbf{0}\end{bmatrix} & \text{otherwise.} \end{cases} \tag{185}$$

$$\hat{\mathbf{x}}^3 = -\|\mathbf{x}_\omega\|^2\widehat{\mathbf{x}} \;- 2(\mathbf{x}_\omega^T\mathbf{x}_u)\begin{bmatrix}\mathbf{0} & \widetilde{\mathbf{x}_\omega} \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \tag{186}$$

$$\hat{\mathbf{x}}^4 = -\|\mathbf{x}_\omega\|^2\widehat{\mathbf{x}}^2 - 2(\mathbf{x}_\omega^T\mathbf{x}_u)\begin{bmatrix}\mathbf{0} & \widetilde{\mathbf{x}_\omega}^2 \\ \mathbf{0} & \mathbf{0}\end{bmatrix} \tag{187}$$

References

[1] Hans Munthe-Kaas. Runge-Kutta methods on Lie groups. *BIT Numerical Mathematics*, 38(1):92–111, 1998.

[2] Marco Borri, Carlo Luigi Bottasso, and Lorenzo Trainelli. Integration of elastic multibody systems by invariant conserving/dissipating algorithms. i. Formulation. *Computer Methods in Applied Mechanics and Engineering*, 190(29–30):3669–3699, 2001.

[3] Elena Celledoni and Brynjulf Owren. Lie group methods for rigid body dynamics and time integration on manifolds. *Computer Methods in Applied Mechanics and Engineering*, 192(3–4):421–438, 2003.

[4] Olivier Brüls and Alberto Cardona. On the use of Lie group time integrators in multibody dynamics. *ASME Journal of Computational and Nonlinear Dynamics*, 5(3), 2010.

[5] Olivier Brüls, Alberto Cardona, and Martin Arnold. Lie group generalized-α time integration of constrained flexible multibody systems. *Mechanism and Machine Theory*, 48:121–137, 2012.

[6] Valentin Sonneville, Alberto Cardona, and O Brüls. Geometrically exact beam finite element formulated on the special Euclidean group SE(3). *Computer Methods in Applied Mechanics and Engineering*, 268:451–474, 01 2014.

[7] Bruno A Roccia, Alejandro Cosimo, Sergio Preidikman, and Olivier Brüls. Numerical models for the static analysis of cable structures used in airborne wind turbines. In *International Symposium on Multibody Systems and Mechatronics*, pages 140–147. Springer, 2020.

[8] Olivier Brüls and Peter Eberhard. Sensitivity analysis for dynamic mechanical systems with finite rotations. *International Journal for Numerical Methods in Engineering*, 74(13):1897–1927, 2008.

[9] Valentin Sonneville and Olivier Brüls. Sensitivity analysis for multibody systems formulated on a Lie group. *Multibody System Dynamics*, 31(1):47–67, 2014.

[10] Arthur Lismonde, Valentin Sonneville, and Olivier Brüls. A geometric optimization method for the trajectory planning of flexible manipulators. *Multibody System Dynamics*, 47:347–362, 2019.

[11] Emmanuel Tromme, Alexander Held, Pierre Duysinx, and Olivier Brüls. System-based approaches for structural optimization of flexible mechanisms. *Archives of Computational Methods in Engineering*, 25(3):817–844, 2018.

[12] John Stuelpnagel. On the parametrization of the three-dimensional rotation group. *SIAM review*, 6(4):422–430, 1964.

[13] A Cardona. *An integrated approach to mechanism analysis.* PhD thesis, Faculté des Sciences Appliquées Université de Liège, 1989.

[14] Olivier Andre Bauchau and Lorenzo Trainelli. The vectorial parameterization of rotation. *Nonlinear Dynamics*, 32(1):71–92, 2003.

[15] Olivier Andre Bauchau. *Flexible Multibody Dynamics, vol. 176.* Springer, 2011.

[16] Andreas Müller. Screw and Lie group theory in multibody kinematics: Motion representation and recursive kinematics of tree-topology systems. *Multibody System Dynamics*, 43(1):37–70, 2018.

[17] Cleve Moler and Charles Van Loan. Nineteen dubious ways to compute the exponential of a matrix, twenty-five years later. *SIAM review*, 45(1):3–49, 2003.

[18] Alberto Cardona and Michel Géradin. A beam finite element non-linear theory with finite rotations. *Int. J. Numer. Methods Eng.*, 26(11):2403–2438, 1988.

[19] Jili Rong, Zhipei Wu, Cheng Liu, and Olivier Brüls. Geometrically exact thin-walled beam including warping formulated on the special euclidean group SE(3). *Computer Methods in Applied Mechanics and Engineering*, 369, 2020.

[20] Felix Pfister. Bernoulli numbers and rotational kinematics. *ASME J. Appl. Mech.*, 65(3):758–763, 1998.

[21] Manuel Ritto-Corrêa and Dinar Camotim. On the differentiation of the rodrigues formula and its significance for the vector-like parameterization of Reissner-Simo beam theory. *International Journal for Numerical Methods in Engineering*, 55(9):1005–1032, 2002.

[22] Andreas Müller. Review of the exponential and cayley map on SE(3) as relevant for Lie group integration of the generalized Poisson equation and flexible multibody systems. *Proc. R. Soc. A*, 477(2253), 2021.

[23] Eduardo Alberto de Souza Neto. The exact derivative of the exponential of an unsymmetric tensor. *Comput. Methods Appl. Mech. Eng.*, 190(18–19):2377–2383, 2001.

[24] Mayank Chadha and Michael Douglas Todd. On the derivatives of curvature of framed space curve and their time-updating scheme. *Applied Mathematics Letters*, 99, 2020.

[25] J.M. Selig. *Geometric Fundamentals of Robotics, Monographs in Computer Science.* Springer, 2005.

[26] Michel Géradin and Alberto Cardona. Kinematics and dynamics of rigid and flexible mechanisms using finite elements and quaternion algebra. *Computational Mechanics*, 4:115–135, 1988.

[27] Eva Zupan, Miran Saje, and Dejan Zupan. The quaternion-based three-dimensional beam theory. *Computer Methods in Applied Mechanics and Engineering*, 198(49):3944–3956, 2009.

[28] Thomas Leitz, Rodrigo Takuro Sato Martín de Almagro, and Sigrid Leyendecker. Multisymplectic Galerkin Lie group variational integrators for geometrically exact beam dynamics based on unit dual quaternion interpolation-no shear locking. *Computer Methods in Applied Mechanics and Engineering*, 374, 2021.

[29] Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A Mathematical Introduction To Robotic Manipulation.* CRC Press, first ed. edition, 1994.

[30] Arieh Iserles, Hans Z. Munthe-Kaas, Syvert Paul Nørsett, and Antonella Zanna. Lie-group methods. *Acta numerica*, 9:215–365, 2000.

[31] Valentin Sonneville, Alberto Cardona, and Olivier Brüls. Geometric interpretation of a non-linear beam finite element on the Lie group SE(3). *Archive of Mechanical Engineering*, 61(2):305–329, 2014.

[32] Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric Numerical Integration: Structure-Preserving Algorithms for Ordinary Differential Equations.* Springer, 2006.

[33] Brian C. Hall. *Lie Groups, Lie Algebras, and Representations an Elementary Introduction.* Graduate Texts in Mathematics, 222. Springer International Publishing, 2nd ed. 2015. edition, 2015.

[34] Valentin Sonneville. *A geometric local frame approach for flexible multibody systems.* PhD thesis, Université de Liège, Liège, Belgique, 2015.

[35] Jorge Angeles. *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms.* Springer-Verlag, 2003.

[36] John H. Argyris and Sp. Symeonidis. Nonlinear finite element analysis of elastic systems under nonconservative loading-natural formulation. part i. Quasistatic problems. *Computer Methods in Applied Mechanics and Engineering*, 26(1):75–123, 1981.

[37] John Argyris. An excursion into large rotations. *Comput. Methods Appl. Mech. Eng.*, 32(1–3):85–155, 1982.

[38] Daniel Condurache and Adrian Burlacu. Dual tensors based solutions for rigid body motion parameterization. *Mechanism and Machine Theory*, 74:390–412, 2014.

[39] Olivier Andre Bauchau and Valentin Sonneville. Formulation of shell elements based on the motion formalism. *Applied Mechanics*, 2(4):1009–1036, 2021.