Lie Theory for Robotics

Petter Nilsson

June 13, 2021

Contents

0.1. Next todos . . . 1
0.2. Research questions . . . 1
0.3. Literature . . . 1

I. Theory **2**

1. Introduction **3**
1.1. Numerical integration . . . 3
1.2. Nonlinear control and estimation . . . 3
1.3. Localization . . . 3
1.4. Notation . . . 3

2. Lie Groups **4**
2.1. Fundamentals . . . 4

3. Lie Algebras **6**
3.1. Lie Algebra definition . . . 6
3.2. The Lie bracket . . . 6
3.3. Application: Derive the Laguerre polynomials . . . 8
3.3.1. Hermite polynomials . . . 10

4. The Exponential Map **11**
4.1. One-Parameter Groups . . . 11
4.2. The Exponential Map . . . 11
4.2.1. Modern Definition . . . 12
4.3. The Lie Algebra of a Lie group . . . 13
4.4. The Logarithm . . . 13
4.5. Baker–Campbell–Hausdorff formula . . . 14
4.6. Plus and Minus Operators . . . 14
4.7. Homomorphy of Lie Groups implies Homomorphy of Lie Algebras . . . 14
4.8. The Adjoint . . . 15

5. Derivatives **17**
5.1. Global Derivative . . . 18
5.2. Product rule . . . 19
5.3. Lie Bracket as the Derivative of the Adjoint . . . 20
5.4. Derivatives of the Exponential map . . . 21
5.5. Derivatives of common operations . . . 22
5.6. On Automatic Differentiation . . . 24
5.6.1. Ceres Solver Local Parameterizations . . . 24

6. Dynamical Systems on Lie Groups **26**
6.1. Tangent Space Linearization . . . 27
6.2. Sensitivity Analysis . . . 27
6.2.1. Example . . . 28
6.3. Monotonicity . . . 29
6.4. The Magnus Expansion . . . 30
6.4.1. Example . . . 31

7. Probability Theory **32**
7.1. Gaussian Distributions . . . 32
7.2. The Banana Distribution . . . 32

8. Equivariance **33**

II. Important Matrix Lie Groups for Robotics **34**

9. Classical Lie Groups **35**
9.1. Mathematical Preliminaries . . . 37

10. $\mathbb{SO}(2)$: The 2D Rotation Group **40**
10.1. Formulas . . . 40
10.2. Parameterization via Isomorhphism with $\mathbb{U}(1)$ . . . 41

11. $\mathbb{SO}(3)$: The 3D Rotation Group **44**
11.1. Grab bag . . . 44
11.2. Formulas . . . 47
11.3. Summary . . . 49
11.4. Parameterization via Double Cover by $\mathbb{S}^3$ . . . 50

12. $\mathbb{SE}(2)$: The 2D Rigid Motion Group **51**
12.1. Formulas . . . 52
12.2. Parameterization via Isomorphism with $\mathbb{U}(1) \ltimes \mathbb{R}^2$ . . . 54

13. $\mathbb{SE}(3)$: The 3D Rigid Motion Group 55
13.1. Formulas 55
13.2. Parameterization via Isomorphism with $\mathbb{S}^3 \ltimes \mathbb{R}^3$ 59

III. Applications 60

14. Geometric Numerical Integration 61

15. Application: Geometric Control 62
15.1. A Stabilizing Lie Group Controller 62
15.2. Error Functions 64
15.3. Lyapunov Stability 65
15.4. Direction-driven Attitude Control on SO(3) 66
15.5. Feedback Control 67
15.6. Lyapunov lower bound 67
15.7. Lyapunov derivative 68

16. Application: Model-Predictive Control 70

17. Application: State Estimation 72
17.1. IMU Model 72
17.2. Complementary Filter for Attitude Estimation 72
17.3. TODO 72

18. Nonlinear Least Squares 73
18.1. Solution Sensitivity 73
18.2. Levenberg-Marquardt 74
18.2.1. Trust-Region Problem 75
18.2.2. Solving the Least-Squares Problem 78
18.2.3. Finding the LM Parameter 78

19. Pose Graph Optimization 81
19.1. Maximum Likelihood Estimation as Nonlinear Least Squares 81
19.2. Measurement functions 82
19.2.1. Absolute pose measurement 82
19.2.2. Relative pose measurement 82
19.2.3. Rectified stereo landmark measurement 82

20. Splines on Lie Groups 83
20.1. Bezier Curves 84
20.1.1. Quadratic Bezier curve 85

20.1.2. Cubic Bezier curve 85
20.2. B-Splines 87
20.3. Evaluating Cumulative Splines 88
20.3.1. First order derivative 88
20.3.2. Second order derivative 89
20.3.3. Derivatives w.r.t. Control Points 89
20.4. General Coefficient Recursion 90
20.5. Cardinal Cofficient Recursion 92

21. Advanced control: controllability, lie brackets, and frobenius **93**

22. Advanced: lidar odometry **94**
22.1. Correspondence search 94
22.1.1. Edge matching 94
22.1.2. Plane matching 94
22.2. Optimization 94

23. Advanced: Marginalization of nonlinear least squares **95**
23.1. Lifted information matrix 95
23.2. Marginalization 96
23.3. Algorithm 97
23.4. Correction for singular information matrix 97
23.5. Marginalization factor in local frame 98
23.6. Example 98
23.7. As part of least-squares problem 100

0.1. Next todos

0.2. Research questions

1. How to define semi-simple group products? Can they be implemented using only a definition of group composition? Seems to be the case at least for SE2 and SE3 since the exponential shows up with the derivative of the exponential.

2. Equivariance and control-invariance for reduced-dimensionality asif.

3. Implicit invariance (requires numerical intergration)
    - Magnus expansion with closed-form dexpinv
    - Just overload + in runge kutta scheme

4. Can optimization be improved for functions that are "linear in tangent space"? Should look for conditions where predicted to actual reduction $\rho = 1$, i.e. linearization is exact.

5. Iterative algorithm to fit Bezier splines that are twice differentiable (solve (20.14))

0.3. Literature

- BOOKS:
    - Barfoot: [2]
        - Estimation in robotics
        - Some explicit formulas
    - Chirkijan: [6, 5]
        - Guassians and information theory
    - Agrachev: [1]
- ARTICLES
    - Very basic Lie theory [11], rigorous mathematics.
    - A Micro Lie Theory [17], application-focused.
    - Quadrotor control [13]
    - IMU estimation: [12, 14]

Part I.

Theory

1. Introduction

Summary

- Treat parameterizations as regular groups
- Get rid of $\check{\mathbb{M}}$, define exp and log $\check{\mathfrak{m}} \leftrightarrow \mathbb{M}$
- Get rid of hat and vee for parameterizations
- Overview of notes.
- Advantages of on-manifold tools.
- Applications of Lie theory in robotics.

1.1. Numerical integration

1.2. Nonlinear control and estimation

1.3. Localization

1.4. Notation

	Set notation	Element notation
Group (matrix form)	$\mathbb{M}, \mathbb{N}$	$\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$
Group (param. form)	$\check{\mathbb{M}}, \check{\mathbb{N}}$	$\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$
Algebra (matrix form)	$\mathfrak{m}, \mathfrak{n}$	$\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}$
Algebra (param. form)	$\check{\mathfrak{m}}, \check{\mathfrak{n}}$	$\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c}$
Rotation matrices		$\boldsymbol{R}$
Rotation parameters		$\boldsymbol{q} = [q_w, q_x, q_y, q_z]$
Velocity parameters		$\boldsymbol{v} = [v_x, v_y, v_z]$
Translation parameters		$\boldsymbol{p} = [p_x, p_y, p_z]$
Angular velocity		$\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]$
Vectors $\mathbb{R}^n$		$\boldsymbol{u}$

2. Lie Groups

Summary

- Fundamental definitions and properties.
- Matrix Lie groups that appear in robotics.

2.1. Fundamentals

A Lie group is an object that is both a group and a smooth manifold. As will be illustrated in these notes, inheritence of these two sets of properties places Lie groups at a unique point where theory meets practice.

We recall the definitions of groups and smooth manifolds, respectively.

Definition 2.1 ([8]). *A **group** $(\mathbb{M}, \circ)$ is a set $\mathbb{M}$ closed under a binary operation $(\circ)$ such that*

- ***associativity** holds: $\boldsymbol{X} \circ (\boldsymbol{Y} \circ \boldsymbol{Z}) = (\boldsymbol{X} \circ \boldsymbol{Y}) \circ \boldsymbol{Z}$ for all $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z} \in \mathbb{M}$,*
- *there is an **identity element** $e \in \mathbb{M}$ s.t. $e \circ \boldsymbol{X} = \boldsymbol{X} \circ e$ for all $\boldsymbol{X} \in \mathbb{M}$,*
- *for each element $\boldsymbol{X} \in \mathbb{M}$ there is an **inverse** $\boldsymbol{X}^{-1} \in \mathbb{M}$ s.t. $\boldsymbol{X}^{-1} \circ \boldsymbol{X} = \boldsymbol{X} \circ \boldsymbol{X}^{-1} = e$.*

Definition 2.2 ([4]). *A **smooth manifold** $(\mathbb{M}, \{c_i\})$ of dimension n is a set $\mathbb{M}$ and a family of injective mappings $c_i : U_i \subset \mathbb{R}^n \to \mathbb{M}$ of open sets U_i (called **charts**) such that*

1. *The charts cover the set: $\bigcup_i U_i = \mathbb{M}$,*
2. *For any pair i, j with $c_i(U_i) \cap c_j(U_j) =: W \neq \emptyset$, the sets $c_i^{-1}(W)$ and $c_j^{-1}(W)$ are open in $\mathbb{R}^n$, and the mappings $c_i^{-1} \circ c_j$ are differentiable.*

Figure of chart mappings

The definition of a Lie group is now straightforward.

$$
\begin{tikzcd}
\check{\mathbb{M}} \arrow[r, bend left, "\wedge"] & \mathbb{M} \arrow[l, bend left, "\vee"]
\end{tikzcd}
$$

Figure 2.1.: The $\vee$ (hat) and $\wedge$ (vee) maps map between the matrix and parameter forms of a matrix Lie group.

Definition 2.3. *A* ***Lie group*** *of dimension n is a set $\mathbb{M}$ together with a binary operation $(\circ)$ and a family of injective mappings $c_i : U_i \subset \mathbb{R}^n \to X$ such that*

1. $(\mathbb{M}, \circ)$ *is a group,*
2. $(\mathbb{M}, \{c_i\})$ *is an n-dimensional smooth manifold.*

In the following we use $\mathbb{M}$ to refer both to the Lie group and to its underlying set. A mathematic object that satisfies the properties in Definition 2.3 is the **general linear group** $GL(n, \mathbb{C})$—the set of $n \times n$ invertible complex matrices with matrix multiplication $(\cdot)$ as the group operation. It turns out that many Lie groups of practical interest can be represented as sub-groups of $GL(n, \mathbb{C})$. In these notes we restrict attention to Matrix Lie groups.

Definition 2.4. *A* ***matrix Lie group*** *is a Lie group that is also a sub-group of $GL(n, \mathbb{C})$—the group of invertible matrices with complex coefficients.*

Lie theory is more straightforward to develop for matrix lie groups compared to a more general setting. Matrix Lie groups are however inefficient from a practical point of view since their representation is often redundant. For this first part of the book we focus exclusively on matrix Lie groups. In Part II we use isometries between matrix Lie groups and more concice non-matrix Lie groups to obtain closed-form formulas for the latter.

3. Lie Algebras

Summary

- Fundamental definitions and properties of Lie Algebras.
- The Lie Bracket.
- Hat and vee operators.
- Maybe: connection to Lie Derivative.

3.1. Lie Algebra definition

Definition 3.1. *A **Lie Algebra** is a vector space $\mathfrak{m}$ with a binary relation $[\cdot,\cdot] : \mathfrak{m} \times \mathfrak{m} \to \mathfrak{m}$ called the **Lie bracket** that satisfies*

1. *Bilinearity:* $[\boldsymbol{A}, \beta\boldsymbol{B} + \gamma\boldsymbol{C}] = \beta[\boldsymbol{A},\boldsymbol{B}] + \gamma[\boldsymbol{A},\boldsymbol{C}]$, *and* $[\alpha\boldsymbol{A} + \beta\boldsymbol{B}, \boldsymbol{C}] = \alpha[\boldsymbol{A},\boldsymbol{C}] + \beta[\boldsymbol{B},\boldsymbol{C}]$,
2. $[\boldsymbol{A},\boldsymbol{A}] = 0$,
3. *Jacobi's identity:* $[\boldsymbol{A},[\boldsymbol{B},\boldsymbol{C}]] + [\boldsymbol{C},[\boldsymbol{A},\boldsymbol{B}]] + [\boldsymbol{B},[\boldsymbol{C},\boldsymbol{A}]] = 0$.

3.2. The Lie bracket

Jacobis identity

Clean this up

We have the two flows $\phi^f(t,x)$ and $\phi^g(t,x)$ that are such that

$$\begin{aligned} \phi^f(0,x) &= x, \quad \frac{\partial}{\partial t}\phi^f(t,x) = f(\phi^f(t,x)), \\ \phi^g(0,x) &= x, \quad \frac{\partial}{\partial t}\phi^g(t,x) = g(\phi^g(t,x)). \end{aligned} \tag{3.1}$$

Consequently, we get the second derivative

$$\left[\frac{\partial^2}{\partial t^2}\phi^f(t,x)\right]_i = \frac{\partial\left[f(\phi^f(t,x)\right]_i}{\partial t} = \left.\frac{\mathrm{d}f_i}{\mathrm{d}x_j}\right|_{x_j=[\phi^f(t,x)]_j}\frac{\partial}{\partial t}\left[\phi^f(t,x)\right]_j$$
$$= \left.\frac{\mathrm{d}f_i}{\mathrm{d}x_j}\right|_{x_j=[\phi^f(t,x)]_j}\left[f(\phi^f(t,x))\right]_j. \tag{3.2}$$

Thus we get

$$\frac{\partial^2}{\partial t^2}\phi^f(t,x) = \left.(f\cdot\nabla)\,f\right|_{\phi^f(t,x)}. \tag{3.3}$$

We are interested in the quantity

$$\phi^g(-t,\cdot)\circ\phi^f(-t,\cdot)\circ\phi^g(t,\cdot)\circ\phi^f(t,\cdot)\,[x] \tag{3.4}$$

for small t.

Note that by Taylor expansion,

$$\begin{aligned}\phi^f(t,x) &= \phi^f(0,x) + \left.\frac{\partial}{\partial t}\phi^f(t,x)\right|_{t=0} t + \left.\frac{\partial^2}{\partial t^2}\phi^f(t,x)\right|_{t=0}\frac{t^2}{2} + \mathcal{O}(t^3)\\ &= \phi^f(0,x) + t f(\phi^f(0,x)) + \frac{t^2}{2}\left(f(\phi^f(0,x))\cdot\nabla\right) f(\phi^f(0,x)) + \mathcal{O}(t^3)\\ &= x + t f(x) + \frac{t^2}{2}\left(f(x)\cdot\nabla\right) f(x) + \mathcal{O}(t^3).\end{aligned} \tag{3.5}$$

We also have that

$$g(x+t\alpha) = g(x) + t(\alpha\cdot\nabla)g(x) + \mathcal{O}(t^2). \tag{3.6}$$

Then we get, after omitting the (x) in $f(x)$ and $g(x)$,

$$
\begin{aligned}
&\phi^g(-t,\cdot)\circ\phi^f(-t,\cdot)\circ\phi^g(t,\cdot)\circ\phi^f(t,\cdot)\,[x] = \\
&\quad = \phi^g(-t,\cdot)\circ\phi^f(-t,\cdot)\circ\phi^g(t,\cdot)\left[x + f\,t + \frac{t^2}{2}\,(f\ \cdot\nabla)\,f\ + \mathcal{O}(t^3)\right] \\
&\quad = \phi^g(-t,\cdot)\circ\phi^f(-t,\cdot)\Bigg[x + tf + \frac{t^2}{2}(f\cdot\nabla)\,f + \mathcal{O}(t^3) \\
&\qquad\quad + \underbrace{t\,g\left(x + tf + \frac{t^2}{2}\,(f\cdot\nabla)\,f + \mathcal{O}(t^3)\right)}_{g\ +t(f\cdot\nabla)g+\mathcal{O}(t^2)} \\
&\qquad\quad + \frac{t^2}{2}\nabla g\left(x + tf + \frac{t^2}{2}\,(f\cdot\nabla)\,f + \mathcal{O}(t^3)\right) \\
&\qquad\qquad \cdot g\left(\left(x + tf + \frac{t^2}{2}\,(f\cdot\nabla)\,f + \mathcal{O}(t^3)\right)\right) + \mathcal{O}(t^3)\Bigg] \\
&\quad = \phi^g(-t,\cdot)\circ\phi^f(-t,\cdot)\left[x + t\,\{f + g(x)\} + t^2\left\{\frac{1}{2}\,(f\cdot\nabla)\,f + (f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g\right\} + \mathcal{O}(t^3)\right] \\
&\quad = \phi^g(-t,\cdot)\Bigg[x + t\,\{f + g\} + t^2\left\{\frac{1}{2}\,(f\cdot\nabla)\,f + (f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g\right\} + \mathcal{O}(t^3) \\
&\qquad\quad - t\,f\left(x + t\{f + g\} + t^2\left\{\frac{1}{2}\,(f\cdot\nabla)\,f + (f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g\right\} + \mathcal{O}(t^3)\right) \\
&\qquad\quad + \frac{t^2}{2}\nabla f\left(x + t\,\{f + g\} + t^2\left\{\frac{1}{2}\,(f\cdot\nabla)\,f + (f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g\right\} + \mathcal{O}(t^3)\right) \\
&\qquad\qquad \cdot\, f\left(x + t\,\{f + g\} + t^2\left\{\frac{1}{2}\,(f\cdot\nabla)\,f + (f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g\right\} + \mathcal{O}(t^3)\right) \\
&\quad = \phi^g(-t,\cdot)\left[x + t\,\{g\} + t^2\left\{\frac{1}{2}\,(f\cdot\nabla)\,f + (f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g - \nabla f\cdot(f + g) + \frac{1}{2}\,(f\cdot\nabla)\,f\right\} + \mathcal{O}(t^3)\right] \\
&\quad = \phi^g(-t,\cdot)\left[x + t\,\{g\} + t^2\left\{(f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g - (g\cdot\nabla)\,f\right\} + \mathcal{O}(t^3)\right] \\
&\quad = x + t\,\{g\} + t^2\left\{(f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g - (g\cdot\nabla)\,f\right\} + \mathcal{O}(t^3) \\
&\qquad\quad - t\,g\left(x + t\{g\} + t^2\left\{(f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g - (g\cdot\nabla)\,f\right\} + \mathcal{O}(t^3)\right) \\
&\qquad\quad + \frac{t^2}{2}\nabla g\left(x + t\,\{g\} + t^2\left\{(f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g - (g\cdot\nabla)\,f\right\} + \mathcal{O}(t^3)\right) \\
&\qquad\qquad \cdot\, g\left(x + t\,\{g\} + t^2\left\{(f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g - (g\cdot\nabla)\,f\right\} + \mathcal{O}(t^3)\right) \\
&\quad = x + t\,\{g - g\} + t^2\left\{(f\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g - (g\cdot\nabla)\,f - (g\cdot\nabla)\,g + \frac{1}{2}\,(g\cdot\nabla)\,g\right\} + \mathcal{O}(t^3) \\
&\quad = x + t^2\,\{(f\cdot\nabla)\,g - (g\cdot\nabla)\,f\} + \mathcal{O}(t^3) \\
&\quad = x + t^2[f,g](x) + \mathcal{O}(t^3).
\end{aligned}
$$

3.3. Application: Derive the Laguerre polynomials

This is an exercise from [11].

Consider the equation

$$xy'' + (1-x)y' + ny = 0,$$

we will show via Lie-algebraic concepts that a solution is given by

$$y = e^x \left(\frac{d}{dx}\right)^n e^{-x}x^n.$$

Letting $P = d/dx$ denote derivative and $Q = x$ multiplication by x the equation can be written

$$Ly = (P-I)QPy = -ny.$$

We consider the Lie algebra spanned by P, Q, I with commutator relationships

$$[P,Q]y = PQy - QPy = y + xy' - xy' = Iy, \implies [P,Q] = I$$
$$[P,I] = [P,Q] = 0.$$

We have from the bracket relation that $(P-I)Q = I + Q(P-I)$, consequently

$$\begin{aligned}[Q,(P-I)^n] &= Q(P-I)^n - (P-I)^n Q \\ &= \left(Q(P-I)^{n-1} - (P-I)^{n-1}Q\right)(P-I) - (P-I)^{n-1} \\ &= [Q,(P-I)^{n-1}](P-I) - (P-I)^{n-1}.\end{aligned}$$

From $[Q, P-I] = -I$ it follows by recursion that

$$[Q,(P-I)^n] = -n(P-I)^{n-1}.$$

Let $A_n = (P-I)^n Q^n$, then with the above

$$\begin{aligned}A_{n+1} &= (P-I)^{n+1}Q^{n+1} = (P-I)\left([(P-I)^n, Q] + Q(P-I)^n\right)Q^n \\ &= (P-I)\left\{n(P-I)^{n-1} + Q(P-I)^n\right\}Q^n = (n+A_1)A_n.\end{aligned}$$

Note that we have $L = A_1 P$ and that $PA_1 = P(P-I)Q = (P-I)QP + (P-I) = A_1P + (P-I)$. It follows that

$$[A_1P, A_1] = A_1PA_1 - A_1^2P = A_1(P-I) = L - A_1.$$

Using the bracket relation it follows that

$$L(A_1+n) = (A_1+n)L + [L, A_1+n] = (A_1+n)L + (L+n) - (A_n+n).$$

Proposition: If v_n is an eigenvector of L with eigenvalue $-n$, then $(A_1+n)v_n$ is an eigenvector with eigenvalue $-(n+1)$. **Proof**: We use the relation above to get

$$\begin{aligned}L(A_1+n)v_n &= (A_1+n)Lv_n + (L+n)v_n - (A_n+n)v_n \\ &= -n(A_1+n)v_n - (A_n+n)v_n.\end{aligned}$$

It follows via the relation $A_{n+1} = (A_1+n)A_n$ shown above that if v_0 is an eigenvector with eigenvalue 0, then $A_n v_0$ is an eigenvector with eigenvalue $-n$.

We have solved $Ly = -ny$, a solution is for instance

$$A_n v_0 = (P-I)^n Q^n 1 = \left(e^x \frac{d}{dx} e^{-x}\right)^n x^n = e^x \left(\frac{d}{dx}\right)^n e^{-x}x^n.$$

3.3.1. Hermite polynomials

Consider the equation
$$y'' + xy' - ny = 0.$$
We show that
$$y = e^{-x^2/2}\left(\frac{d}{dx}\right)^n e^{x^2/2}$$
is a solution.

Also the operators $P = d/dx$ and $Q = x + d/dx$ satisfy the same operations, in partiuclar $[P, Q] = I$. We have that
$$Ly = QPy = y'' + xy',$$
so we would like to solve
$$QPv = nv.$$
This is easy: suppose that $QPQ^{n-1}v_0 = (n-1)Q^{n-1}v_0$ which is true for $v_0 = 1$ at $n = 1$. Then,
$$\begin{aligned} QPQ^n v_0 &= QPQQ^{n-1}v_0 = Q([P, Q] + QP)Q^{n-1}v_0 \\ &= Q^n v_0 + Q^2PQ^{n-1}v_0 = Q^n + (n-1)Q^n v_0 = nQ^n v_0. \end{aligned}$$
Thus it follows that the solution is $Q^n 1$, and using that
$$Q = e^{-x^2/2}\frac{d}{dx}e^{x^2/2}$$
the answer is obtained.

4. The Exponential Map

Summary

- The Exponential map and how it connects a Lie group to its Lie algebra.
- The Lie group logarithm, plus and minus operators.
- The structure of the Lie algebras corresponding to common Lie groups.

Need a nice derivation showing how lie algebra properties arise

4.1. One-Parameter Groups

Best way to prove that Lie Groups have Lie Algebras?

- In [11] it is shown that for **matrix** Lie groups the set $\{A \in \mathrm{End} V \ : \ \exp tA \in G \forall t\} = \cap_t t \exp^{-1}(G)$ is a Lie Algebra (i.e. closed under the bracket operation).

Dual viewpoint: solutions $\Phi(x,t)$ of ODEs correspond to one-parameter groups [11]. Connection to linear systems.

4.2. The Exponential Map

Definition 4.1. *The* ***Exponential map*** *of a matrix* $\boldsymbol{A} \in \mathbb{C}^{n \times n}$ *and* $t \in \mathbb{R}$ *is*

$$\mathrm{Exp}(\boldsymbol{A}) = \sum_{n=0}^{\infty} \frac{\boldsymbol{A}^n}{n!} \in \mathbb{C}^{n \times n}. \tag{4.1}$$

Properties of the exponential map

For the exponential map in Definition 4.1 we have

$$\mathrm{Exp}(t\boldsymbol{A})\,\mathrm{Exp}(s\boldsymbol{A}) = \mathrm{Exp}((t+s)\boldsymbol{A}), \tag{4.2a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathrm{Exp}(t\boldsymbol{A}) = \boldsymbol{A}\,\mathrm{Exp}(t\boldsymbol{A}) = \mathrm{Exp}(t\boldsymbol{A})\boldsymbol{A}, \tag{4.2b}$$

$$\det(\mathrm{Exp}(\boldsymbol{A})) = e^{\mathrm{Tr}(\boldsymbol{A})}. \tag{4.2c}$$

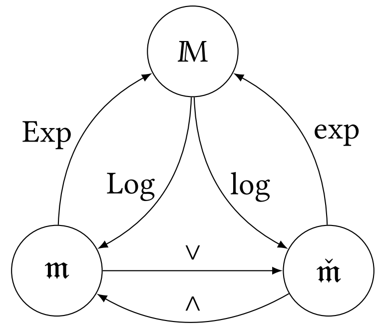


Figure 4.1.: Illustration of how the exponential maps connect a Lie Group $\mathbb{M}$, its Lie Algebra $\mathfrak{m}$, and the Lie Algebra parameterization which is the linear space $\check{\mathfrak{m}} \cong \mathbb{R}^n$.

The first two follow directly from the definition and are analogous to the scalar exponential map. Furthermore, (4.2a) implies that $\{\mathrm{Exp}(t\boldsymbol{A}) : t \in \mathbb{R}\}$ is a one-parameter subgroup of $\mathbb{M}$.

Not however that in general $\exp(\boldsymbol{A} + \boldsymbol{B}) \neq \exp(\boldsymbol{A}) \circ \exp(\boldsymbol{B})$ which is different from the scalar version. Equation (4.2c), known as Jacobi's identity, motivates a short proof:

Proof of (4.2c). It is easy to see that the eigenvalues of $\mathrm{Exp}(\boldsymbol{A})$ are the exponentials of the eigenvalues of $\boldsymbol{A}$. Since the determinant equals the product of the eigenvalues it follows that

$$\det(\mathrm{Exp}\,\boldsymbol{A}) = \prod_{i=1}^{n} \lambda_i(\mathrm{Exp}\,\boldsymbol{A}) = \prod_{i=1}^{n} e^{\lambda_i(\boldsymbol{A})} = e^{\sum_{i=1}^{n} \lambda_i(\boldsymbol{A})} = e^{\mathrm{Tr}(\boldsymbol{A})}. \tag{4.3}$$

□

4.2.1. Modern Definition

We contrast the algebraic Definition 4.1 with a more modern definition usually found in texts on differential geometry.

Definition 4.2. *The* ***exponential*** *of* $\boldsymbol{A} \in \mathfrak{m}$ *is*

$$\mathrm{Exp}\,\boldsymbol{A} := \gamma(1), \tag{4.4}$$

where γ *is a one-parameter subgroup of* $\mathbb{M}$ *such that* $\gamma'(0) = \boldsymbol{A}$.

This definition of course works even in the case that $\mathbb{M}$ is not a matrix Lie group, in fact it the same definition is used in more general differential geometry. We show that it coincides with Definition 4.1 for matrix Lie groups.

Proof. We know that $\gamma(0) = I$, $\gamma'(0) = \boldsymbol{A}$, and $\gamma(s)\gamma(t) = \gamma(s+t)$ by virtue of γ being a one-parameter sub-group.

$$\left(\gamma\left(\frac{h}{2}\right) - \gamma\left(-\frac{h}{2}\right)\right)^n = \sum_{k=0}^{n}(-1)^k\binom{n}{k}\gamma\left(\frac{h}{2}\right)^{n-k}\gamma\left(-\frac{h}{2}\right)^k = \sum_{k=0}^{n}(-1)^k\binom{n}{k}\gamma\left(\left(\frac{n}{2}-k\right)h\right). \tag{4.5}$$

For $h \to 0$ the left-hand side goes to $(h\gamma'(0))^n$, whereas the right-hand side is a finite-difference approximation of $h^n\gamma^{(n)}(0)$. It follows that $\gamma^n(0) = \boldsymbol{A}^n$, and hence a Taylor expansion around 0 gives

$$\gamma(1) = \sum_{k\geq 0} \frac{\gamma^{(n)}(0)(1-0)^n}{n!} = \sum_{k\geq 0} \frac{\boldsymbol{A}^n}{n!}. \tag{4.6}$$

□

A consequence of the proof above is that one-parameter subgroups of Lie groups are uniquely defined by their derivative at zero, and are therefore analogous to geodesics in Riemannian geometry.

4.3. The Lie Algebra of a Lie group

Definition 4.3. *For a matrix Lie group $\mathbb{M}$ the corresponding* ***matrix Lie algebra*** $\mathfrak{m}$ *is*

$$\mathfrak{m} = \{\boldsymbol{A} : \mathrm{Exp}(t\boldsymbol{A}) \in \mathbb{M}\ \forall t \in \mathbb{R}\}. \tag{4.7}$$

Just as for Lie groups, the matrix Lie algebras are typically parameterized by fewer than n^2 coefficients. In order to work with efficient parameterizations we therefore introduce a lower-dimensional parameterization denoted $\check{\mathfrak{m}}$. For this lower-dimensional representation we also define a lowercase exponential that maps from the parameterized lie algebra representation $\check{\mathfrak{m}}$ to the parameterized group representation $\check{\mathbb{M}}$:

$$\exp(\boldsymbol{a}) = \mathrm{Exp}(\boldsymbol{a}^\wedge). \tag{4.8}$$

The relationshop between the exponential maps and the hat and vee maps is shown in Figure 4.1.

Show that Lie algebra defined like this is indeed a Lie algebra(closed under bracket, jacobi, etc). Use property from previous chapter to show that as $t \to 0$ we obtain a tangent that is equal to the bracket. Group property is then enough to conclude.

4.4. The Logarithm

The matrix logarithm $\mathrm{Log} : \mathbb{M} \to \mathfrak{m}$ is defined as the inverse of the matrix exponential, and we also define lowercase $\log : \mathbb{M} \to \check{\mathfrak{m}}$ for mappings between the parameterized representations:

$$\begin{aligned} \mathrm{Log}\, \boldsymbol{X} &= \sum_{k\geq 1} (-1)^{k+1} \frac{(\boldsymbol{X}-I)^k}{k}, \\ \log \boldsymbol{X} &= (\mathrm{Log}\, \boldsymbol{X})^\vee. \end{aligned} \tag{4.9}$$

Show that it's the inverse of the exponential

4.5. Baker–Campbell–Hausdorff formula

Prove BCH formula and provide context

$$\log(\exp \boldsymbol{a} \circ \exp \boldsymbol{b}) = \boldsymbol{a} + \boldsymbol{b} + \frac{1}{2}[\boldsymbol{a}, \boldsymbol{b}] + \frac{1}{12}[\boldsymbol{a}, [\boldsymbol{a}, \boldsymbol{b}]] - \frac{1}{12}[\boldsymbol{b}, [\boldsymbol{a}, \boldsymbol{b}]] + \dots \tag{4.10}$$

4.6. Plus and Minus Operators

Algorithms for optimization and numerical integration require taking small additative steps, but Lie groups are not closed under normal addition and subtraction. We can however define generalized addition and subtraction operators $\oplus, \ominus$ for Lie groups that behave similarly to how $+$ and $-$ operate on regular vector spaces.

The plus operations add an increment $\boldsymbol{a} \in \check{\mathfrak{m}}$ in the parameterized tangent space to an element $\boldsymbol{X} \in \mathbb{M}$ of the group, whereas the minus operators give the difference between two group elements as a vector in the parameterized tangent space.

$$\boldsymbol{X} \oplus_r \boldsymbol{a} = \boldsymbol{X} \circ \exp(\boldsymbol{a}) \in \check{\mathbb{M}}, \tag{right-plus}$$

$$\boldsymbol{Y} \ominus_r \boldsymbol{X} = \log\left(\boldsymbol{X}^{-1} \circ \boldsymbol{Y}\right) \in T_{\boldsymbol{X}}\check{\mathbb{M}} \cong \check{\mathfrak{m}}, \tag{right-minus}$$

$$\boldsymbol{a} \oplus_l \boldsymbol{X} = \exp(\boldsymbol{a}) \circ \boldsymbol{X} \in \check{\mathbb{M}}, \tag{left-plus}$$

$$\boldsymbol{Y} \ominus_l \boldsymbol{X} = \log\left(\boldsymbol{Y} \circ \boldsymbol{X}^{-1}\right) \in T_e\check{\mathbb{M}} \cong \check{\mathfrak{m}}. \tag{left-minus}$$

The plus operators are differentiated by the order: the right-plus has the tangent element at $\boldsymbol{X}$ while left-plus has the reverse order, meaning that the tangent element belongs to the tangent space at e.

Note that the derivates are defined in a way so that

$$\boldsymbol{X} \oplus_r (\boldsymbol{Y} \ominus_r \boldsymbol{X}) = \boldsymbol{X} \circ \exp\log(\boldsymbol{X}^{-1}\boldsymbol{Y}) = \boldsymbol{Y}, \tag{4.11a}$$

$$(\boldsymbol{X} \oplus_r \boldsymbol{a}) \ominus_r \boldsymbol{X} = \log(\boldsymbol{X}^{-1} \circ \boldsymbol{X} \circ \exp \boldsymbol{a}) = \boldsymbol{a}, \tag{4.11b}$$

$$(\boldsymbol{Y} \ominus_l \boldsymbol{X}) \oplus_l \boldsymbol{X} = \exp\log(\boldsymbol{Y} \circ \boldsymbol{X}^{-1}) \circ \boldsymbol{X} = \boldsymbol{Y}, \tag{4.11c}$$

$$(\boldsymbol{a} \oplus_l \boldsymbol{X}) \ominus_l \boldsymbol{X} = \log(\exp \boldsymbol{a} \circ \boldsymbol{X} \circ \boldsymbol{X}^{-1}) = \boldsymbol{a}. \tag{4.11d}$$

4.7. Homomorphy of Lie Groups implies Homomorphy of Lie Algebras

This is important since it implies that $SO(3)$ and S^3 can be treated analogously. A proof is in [11, Corr. 20].

4.8. The Adjoint

We define the **adjoint** $\mathrm{Ad}_X : \mathfrak{m} \to \mathfrak{m}$ of a matrix $\boldsymbol{A} \in \mathfrak{m}$ as

$$\mathrm{Ad}_X \boldsymbol{A} := \boldsymbol{X}\boldsymbol{A}\boldsymbol{X}^{-1}. \tag{4.12}$$

From the definition of the exponential map in (4.1) it can be seen that $\mathrm{Exp}(\mathrm{Ad}_X \boldsymbol{A}) \in \mathbb{M}$ if and only if $\mathrm{Exp}(\boldsymbol{A}) \in \mathbb{M}$, which implies that the lie algebra $\mathfrak{m}$ is closed under action of the adjoint.

The adjoint of a tangent matrix element $\boldsymbol{a} \in \check{\mathfrak{m}}$ is similary defined as a linear mapping $\mathbf{Ad}_X : \check{\mathfrak{m}} \to \check{\mathfrak{m}}$:

$$\mathbf{Ad}_X \boldsymbol{a} := (\mathrm{Ad}_X \hat{\boldsymbol{a}})^\vee = (\boldsymbol{X}\hat{\boldsymbol{a}}\boldsymbol{X}^{-1})^\vee. \tag{4.13}$$

For a given $\boldsymbol{X}$ this is a linear map, so $\mathbf{Ad}_X$ is an $n \times n$ matrix. The adjoint represents a coordinate change from the tangent space $T_X\check{\mathbb{M}}$ to the tangent space $T_e\check{\mathbb{M}} = \check{\mathfrak{m}}$ at the origin.

Remark 4.1. *Since the definition of* Ad *involves matrix multiplication it does not make sense for groups like* $\mathbb{S}\check{\mathbb{O}}(2)$ *and* $\mathbb{S}^3$ *that are not matrix Lie groups. We can however still define the bold-face adjoint* $\boldsymbol{Ad}$ *on* $\check{\mathbb{M}}$ *as*

$$\boldsymbol{Ad}_x := \boldsymbol{Ad}_{\hat{x}}, \tag{4.14}$$

where $\wedge : \check{\mathbb{M}} \to \mathbb{M}$ *is a Lie group homomorphism that maps* $\check{\mathbb{M}}$ *into a matrix Lie group.*

Properties of the adjoint

The adjoints satisfy the following properties:

$$\mathbf{Ad}_X^{-1} = \mathbf{Ad}_{X^{-1}}, \tag{4.15a}$$
$$\mathbf{Ad}_X \mathbf{Ad}_Y = \mathbf{Ad}_{X \circ Y}, \tag{4.15b}$$
$$\exp \ \mathbf{Ad}_X \ \boldsymbol{a} = \boldsymbol{X} \circ \exp \boldsymbol{a} \circ \boldsymbol{X}^{-1}, \tag{4.15c}$$
$$\boldsymbol{X} \oplus_r \boldsymbol{a} = (\mathbf{Ad}_X \boldsymbol{a}) \oplus_l \boldsymbol{X}. \tag{4.15d}$$

The first two properties follow directly from the definition. Equation (4.15c) follows from

$$\exp \mathbf{Ad}_X \boldsymbol{a} = \mathrm{Exp} \ \mathrm{Ad}_X \ \hat{\boldsymbol{a}} = \sum_{k \geq 0} \frac{\left(\boldsymbol{X}\hat{\boldsymbol{a}}\boldsymbol{X}^{-1}\right)^k}{k} = \boldsymbol{X}\left(\sum_{k \geq 0} \frac{\hat{\boldsymbol{a}}^k}{k}\right)\boldsymbol{X}^{-1} = \boldsymbol{X} \exp(\boldsymbol{a})\boldsymbol{X}^{-1}. \tag{4.16}$$

We can then also show (4.15d)

$$\boldsymbol{X} \oplus_r \boldsymbol{a} \overset{\text{(right-plus)}}{=} \boldsymbol{X} \circ \exp(\boldsymbol{a}) = (\boldsymbol{X} \circ \exp(\boldsymbol{a}) \circ \boldsymbol{X}^{-1}) \circ \boldsymbol{X} \overset{\text{(4.15c)}}{=} \exp(\mathbf{Ad}_X \boldsymbol{a}) \circ \boldsymbol{X} \overset{\text{(left-plus)}}{=} (\mathbf{Ad}_X \boldsymbol{a}) \oplus_l \boldsymbol{X}. \tag{4.17}$$

Finally a result regarding the derivative of the adjoint.

Lemma 4.1.

$$\frac{\mathrm{d}}{\mathrm{d}t} \boldsymbol{Ad}_{\exp(\lambda(t)\boldsymbol{a})} = \lambda'(t) \, \mathrm{ad}_{\boldsymbol{a}} \, \boldsymbol{Ad}_{\exp(\lambda(t)\boldsymbol{a})} . \tag{4.18}$$

Proof.

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\,\mathbf{Ad}_{\exp(\lambda(t)\boldsymbol{a})} &\overset{(5.22)}{=} \frac{\mathrm{d}}{\mathrm{d}t}\sum_{k=0}^{\infty}\exp(\mathrm{ad}_{\lambda(t)\boldsymbol{a}}) \overset{(5.20)}{=} \frac{\mathrm{d}}{\mathrm{d}t}\sum_{k=0}^{\infty}\exp(\lambda(t)\,\mathrm{ad}_{\boldsymbol{a}}) \overset{(5.21)}{=} \frac{\mathrm{d}}{\mathrm{d}t}\sum_{k=0}^{\infty}\frac{\lambda(t)^k\,\mathrm{ad}_{\boldsymbol{a}}^k}{k!} \\ &= \lambda'(t)\sum_{k=1}^{\infty}\frac{\lambda(t)^{k-1}\,\mathrm{ad}_{\boldsymbol{a}}^k}{(k-1)!} = \lambda'(t)\,\mathrm{ad}_{\boldsymbol{a}}\sum_{k=1}^{\infty}\frac{\lambda(t)^{k-1}\,\mathrm{ad}_{\boldsymbol{a}}^{k-1}}{(k-1)!} = \lambda'(t)\,\mathrm{ad}_{\boldsymbol{a}}\,\mathbf{Ad}_{\exp(\lambda(t)\boldsymbol{a})}\,.\end{aligned} \tag{4.19}$$

□

5. Derivatives

Summary

- Definition of derivatives on manifolds.
- Differentiation rules.

Define derivatives w.r.t. matrix elements only, motivate that we can disregard parameterized expressions.

Definition 5.1. *The **right derivative** of* $f : \mathbb{M} \to \mathbb{N}$ *at* $\boldsymbol{X} \in \check{\mathbb{M}}$ *is a linear mapping* $\mathrm{d}^r f_X : T\mathbb{M}_X \to T\mathbb{N}_{f(X)}$ *such that:*

$$\mathrm{d}^r f_X := \lim_{\boldsymbol{a}\to 0} \frac{f(\boldsymbol{X} \oplus_r \boldsymbol{a}) \ominus_r f(\boldsymbol{X})}{\boldsymbol{a}} = \lim_{\boldsymbol{a}\to 0} \frac{\log\left(f(\boldsymbol{X})^{-1} \circ f(\boldsymbol{X} \circ \exp(\boldsymbol{a}))\right)}{\boldsymbol{a}}, \tag{5.1}$$

where $\boldsymbol{a} \in T_X\check{\mathbb{M}}$ *is a member of the parameterized Lie algebra and the division is component-wise.*

*Similarly, the **left derivative** is a linear mapping* $\mathrm{d}^r f_X : T\mathbb{M}_e \to T\mathbb{N}_e$ *such that*

$$\mathrm{d}^l f_X := \lim_{\boldsymbol{a}\to 0} \frac{f(\boldsymbol{X} \oplus_l \boldsymbol{a}) \ominus_l f(\boldsymbol{X})}{\boldsymbol{a}} = \lim_{\boldsymbol{a}\to 0} \frac{\log\left(f(\exp(\boldsymbol{a}) \circ \boldsymbol{X}) \circ f(\boldsymbol{X})^{-1}\right)}{\boldsymbol{a}}, \tag{5.2}$$

From the definition it can be seen that for small $\boldsymbol{a}$ it approximately holds that

$$f(\boldsymbol{X} \oplus_r \boldsymbol{a}) = f(\boldsymbol{X}) \oplus_r \left(\mathrm{d}^r f_X \boldsymbol{a} + \mathcal{O}(\|\boldsymbol{a}\|^2)\right), \tag{5.3}$$

and for left-plus:

$$f(\boldsymbol{a} \oplus_l \boldsymbol{X}) = \left(\mathrm{d}^l f_X \boldsymbol{a} + \mathcal{O}(\|\boldsymbol{a}\|^2)\right) \oplus_l f(\boldsymbol{X}). \tag{5.4}$$

From (5.3) and (5.4) we have that for small $\boldsymbol{a}$,

$$f(\boldsymbol{X}) \oplus_r (\mathrm{d}^r f_X \boldsymbol{a}) \overset{(5.3)}{=} f(\boldsymbol{X} \oplus_r \boldsymbol{a}) \overset{(4.15d)}{=} f\left(\mathbf{Ad}_X \boldsymbol{a} \oplus_l \boldsymbol{X}\right) \overset{(5.4)}{=} (\mathrm{d}^l f_X \,\mathbf{Ad}_X \boldsymbol{a}) \oplus_l f(\boldsymbol{X}). \tag{5.5}$$

Consequently,

$$\exp(\mathrm{d}^l f_X \,\mathbf{Ad}_X \boldsymbol{a}) = f(\boldsymbol{X}) \circ \exp(\mathrm{d}^r f_X \boldsymbol{a}) \circ f(\boldsymbol{X})^{-1} = \mathbf{Ad}_{f(X)} \exp(\mathrm{d}^r f_X \boldsymbol{a}), \tag{5.6}$$

and due to (4.15c) it follows that left and right derivatives are related through the adjoints via

$$\mathrm{d}^l f_X = \mathbf{Ad}_{f(X)}\ \mathrm{d}^r f_X\ \mathbf{Ad}_X^{-1}. \tag{5.7}$$

With the interpretation of the adjoints as coordinate changes this formula can be seen as follows: the derivative of f with respect to a tangent vector ${}^e\boldsymbol{a}$ at e can be obtained by

1. Convert ${}^e\boldsymbol{a}$ to a tangent vector at $\boldsymbol{X}$: ${}^X\boldsymbol{a} = \mathbf{Ad}_X^{-1}\,{}^e\boldsymbol{a} \in T_X\check{\mathbb{M}}$,
2. Map the tangent vector through the derivative: ${}^X\boldsymbol{b} = \mathrm{d}^r f_X\,{}^X\boldsymbol{a} \in T_{f(X)}\check{\mathbb{M}}$,
3. Convert the result back to a tangent vector at e: ${}^e\boldsymbol{b} = \mathbf{Ad}_{f(X)}\,{}^X\boldsymbol{b} \in T_e\check{\mathbb{M}}$.

Jacobians on Lie Groups satisfy the chain rule. Indeed, if $f(\boldsymbol{X}) = g \circ h(\boldsymbol{X})$ for some $g : \mathbb{M}' \to \mathbb{M}''$ and $h : \mathbb{M} \to \mathbb{M}'$ we have with $\boldsymbol{Z} := h(\boldsymbol{X})$

$$\begin{aligned}\mathrm{d}^r(g \circ h)_X &= \lim_{\boldsymbol{a}\to 0} \frac{g\left(h(\boldsymbol{X} \oplus_r \boldsymbol{a})\right) \ominus_r g(h(\boldsymbol{X}))}{\boldsymbol{a}} \overset{(5.3)}{=} \lim_{\boldsymbol{a}\to 0} \frac{g\left(h(\boldsymbol{X}) \oplus_r \left(\mathrm{d}^r h_X \boldsymbol{a} + \mathcal{O}(\|\boldsymbol{a}\|^2)\right)\right) \ominus_r g(h(\boldsymbol{X}))}{\boldsymbol{a}} \\ &\overset{(5.3)}{=} \lim_{\boldsymbol{a}\to 0} \frac{\left(g(\boldsymbol{Z}) \oplus_r \left(\mathrm{d}^r g_Z\, \mathrm{d}^r h_X \boldsymbol{a} + \mathcal{O}(\|\boldsymbol{a}\|^2)\right)\right) \ominus_r g(h(\boldsymbol{X}))}{\boldsymbol{a}} \overset{(4.11b)}{=} \mathrm{d}^r g_Z\, \mathrm{d}^r h_X.\end{aligned} \tag{5.8}$$

An analogous left chain rule can be developed in the same manner via (5.4) in lieu of (5.3).

Important formulas for Lie group derivatives

- Right derivative: $\mathrm{d}^r f_X := \lim_{\boldsymbol{a}\to 0} \frac{\log\left(f(X)^{-1} \circ f(X \circ \exp(\boldsymbol{a}))\right)}{\boldsymbol{a}} \in T_X\mathbb{M}$,
- Left derivative: $\mathrm{d}^l f_X := \lim_{\boldsymbol{a}\to 0} \frac{\log\left(f(\exp(\boldsymbol{a}) \circ X) \circ f(X)^{-1}\right)}{\boldsymbol{a}} \in T_e\mathbb{M}$,
- Conversion between left and right jacobians: $\mathrm{d}^l f_X = \mathbf{Ad}_{f(X)}\ \mathrm{d}^r f_X\ \mathbf{Ad}_X^{-1}$,
- Right chain rule: $\mathrm{d}^r(g(h(\boldsymbol{X})))_X = \mathrm{d}^r g_{h(X)}\ \mathrm{d}^r h_X$,
- Left chain rule: $\mathrm{d}^l(g(h(\boldsymbol{X})))_X = \mathrm{d}^l g_{h(X)}\ \mathrm{d}^l h_X$.

5.1. Global Derivative

For a mapping $f : \mathbb{M} \to \mathbb{N}$ between two manifolds the classical way to define a derivative $\mathrm{D} f_X$ is as a mapping $T_X\mathbb{M} \to T_{f(X)}\mathbb{N}$ defined as

$$\mathrm{D} f_X \boldsymbol{B} := \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} f(\gamma(t)), \qquad \begin{cases}\gamma(0) = \boldsymbol{X}, \\ \gamma'(0) = \boldsymbol{B}.\end{cases} \tag{5.9}$$

for $\boldsymbol{B} \in T_X\mathbb{M}$. Note that this definition wouldn't make sense for an arbitrary matrix $\boldsymbol{B}$; for γ to take values in $\mathbb{M}$ the derivative at zero must be on the form $\boldsymbol{B} = \boldsymbol{X}\hat{\boldsymbol{a}}$. Being in global matrix coordinates, $\mathrm{D} f_X \boldsymbol{B}$ typically does not exhibit the structure of the tangent space at $T_{f(X)}\mathbb{N}$. However, it can be mapped to the tangent space via group action, which yields an alternative way of defining the right and left derivatives.

$$
\begin{aligned}
\mathrm{d}^r f_X \boldsymbol{a} &\coloneqq \left(f(X)^{-1}\,(\mathrm{D}f_X\ X\hat{\boldsymbol{a}})\right)^{\vee} = \left(f(X)^{-1}\left(\frac{\mathrm{d}}{\mathrm{d}t}\Big|_{t=0} f(\gamma(t))\right)\right)^{\vee}, \qquad &&\begin{cases}\gamma(0) = X,\\ \gamma'(0) = X\hat{\boldsymbol{a}},\end{cases}\\
\mathrm{d}^l f_X \boldsymbol{a} &\coloneqq \left((\mathrm{D}f_X\ \hat{\boldsymbol{a}}X)\, f(X)^{-1}\right)^{\vee} = \left(\left(\frac{\mathrm{d}}{\mathrm{d}t}\Big|_{t=0} f(\gamma(t))\right) f(X)^{-1}\right)^{\vee}, \qquad &&\begin{cases}\gamma(0) = X,\\ \gamma'(0) = \hat{\boldsymbol{a}}X.\end{cases}
\end{aligned}
\tag{5.10}
$$

Show that these definitions agree with those above

5.2. Product rule

Consider a function $f(X) = g(X) \circ h(X)$, we utilize (5.3) to obtain

$$
\begin{aligned}
f(X \oplus \boldsymbol{a}) &= \left(g(X) \oplus \left(\mathrm{d}^r g_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2)\right)\right) \circ \left(h(X) \oplus \left(\mathrm{d}^r h_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2)\right)\right)\\
&= g(X) \circ \exp(\mathrm{d}^r g_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2)) \circ h(X) \circ \exp(\mathrm{d}^r h_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2))\\
&= g(X) \circ h(X) \circ \left(\mathrm{Ad}_{h(X)^{-1}} \exp\left(\mathrm{d}^r g_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2)\right)\right) \circ \exp(\mathrm{d}^r h_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2))\\
&\overset{(4.15c)}{=} g(X) \circ h(X) \circ \left(\exp \mathbf{Ad}_{h(X)^{-1}} \left(\mathrm{d}^r g_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2)\right)\right) \circ \exp(\mathrm{d}^r h_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2))\\
&\overset{(4.10)}{=} g(X) \circ h(X) \circ \exp\left(\mathbf{Ad}_{h(X)^{-1}}\, \mathrm{d}^r g_X \boldsymbol{a} + \mathrm{d}^r h_X \boldsymbol{a} + \mathcal{O}(\boldsymbol{a}^2))\right).
\end{aligned}
\tag{5.11}
$$

From here we can conclude that

$$
\mathrm{d}^r (g \circ h)_X = \mathbf{Ad}_{h(X)^{-1}}\, \mathrm{d}^r g_X + \mathrm{d}^r h_X \tag{5.12}
$$

which is the product rule for Lie group derivatives.

Remark 5.1. *There is no Lie group equivalent of the rule of total derivative. Consider*

$$
\begin{aligned}
f(g(X \oplus \boldsymbol{a}), h(X \oplus \boldsymbol{a})) &\approx f\left(g(X) \oplus \mathrm{d}^r g_X\ \boldsymbol{a}, h(X) \oplus \mathrm{d}^r h_X\ \boldsymbol{a}\right)\\
&\approx f\left(g(X), h(X) \oplus \mathrm{d}h_X \boldsymbol{a}\right) \oplus \mathrm{d}^r f_g\ \mathrm{d}^r g_X \boldsymbol{a}\\
&\approx \left[f\left(g(X), h(X)\right) \oplus \mathrm{d}^r f_h\ \mathrm{d}^r h_X \boldsymbol{a}\right] \oplus \mathrm{d}^r f_g\ \mathrm{d}^r g_X \boldsymbol{a}\\
&= f\left(g(X), h(X)\right) \circ \left[\exp\left(\mathrm{d}^r f_h\ \mathrm{d}^r h_X \boldsymbol{a}\right) \circ \exp\left(\mathrm{d}^r f_g\ \mathrm{d}^r g_X \boldsymbol{a}\right)\right].
\end{aligned}
\tag{5.13}
$$

That is, if $f(X) = f(g(X), h(X))$ we typically have that

$$
\mathrm{d}^r (f(g(X), h(X)))_X \neq \mathrm{d}^r f_{g(X)}\ \mathrm{d}^r g_X + \mathrm{d}^r f_{h(X)}\ \mathrm{d}^r h_X. \tag{5.14}
$$

However, from (4.10) it can be seen that if

$$
\left[\mathrm{d}^r f_{h(X)}\ \mathrm{d}^r h_X \boldsymbol{a}, \mathrm{d}^r f_{g(X)}\ \mathrm{d}^r g_X \boldsymbol{a}\right] = 0, \qquad \forall \boldsymbol{a}, \tag{5.15}
$$

then the rule of total derivatives applies. One important case when this holds is when f takes values in $\mathbb{E}(n)$ since matrix multiplication on $\mathbb{E}(n)$ corresponds to vector addition on $\mathbb{R}^n$ and hence all brackets are zero.

5.3. Lie Bracket as the Derivative of the Adjoint

The Lie bracket between two tangent elements can be defined as the global derivative of the adjoint operator at identity. Consider the mapping $f(\boldsymbol{X}) \coloneqq \mathbf{Ad}_X\, \boldsymbol{b} = \boldsymbol{X}\hat{\boldsymbol{b}}\boldsymbol{X}^{-1}$ and take a curve $\gamma(t) \in \mathbb{M}$ such that $\gamma(0) = \boldsymbol{X}$ and $\gamma'(0) = \hat{\boldsymbol{a}}$. From $\frac{\mathrm{d}}{\mathrm{d}t}\gamma(t)\gamma(t)^{-1} = 0$ it follows that $\frac{\mathrm{d}}{\mathrm{d}t}\gamma(t)^{-1} = -\gamma(t)^{-1}\gamma'(t)\gamma(t)^{-1}$, hence

$$\left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} f(\gamma(t)) = \gamma'(0)\hat{\boldsymbol{b}}\gamma(0)^{-1} - \gamma(0)\hat{\boldsymbol{b}}\gamma(0)^{-1}\gamma'(0)\gamma(0)^{-1} = \hat{\boldsymbol{a}}\hat{\boldsymbol{b}}\boldsymbol{X}^{-1} - \boldsymbol{X}\hat{\boldsymbol{b}}\boldsymbol{X}^{-1}\hat{\boldsymbol{a}}\boldsymbol{X}^{-1}. \tag{5.16}$$

The derivatives of $\mathbf{Ad}_X$ with respect to $\boldsymbol{X}$ are

$$\mathrm{D}(\mathbf{Ad}_X\, \boldsymbol{b})_X\hat{\boldsymbol{a}} = \hat{\boldsymbol{a}}\hat{\boldsymbol{b}}\boldsymbol{X}^{-1} - \boldsymbol{X}\hat{\boldsymbol{b}}\boldsymbol{X}^{-1}\hat{\boldsymbol{a}}X^{-1}, \tag{5.17a}$$

$$\mathrm{d}^r(\mathbf{Ad}_X\, \boldsymbol{b})_X\boldsymbol{a} = \left(\mathrm{D}\left(\mathbf{Ad}_X\, \boldsymbol{b}\right)_X \boldsymbol{X}\hat{\boldsymbol{a}}\right)^\vee = \mathbf{Ad}_X\left[\boldsymbol{a}, \boldsymbol{b}\right] = \left[\mathbf{Ad}_X\, \boldsymbol{a}, \mathbf{Ad}_X\, \boldsymbol{b}\right], \tag{5.17b}$$

$$\mathrm{d}^l(\mathbf{Ad}_X\, \boldsymbol{b})_X\boldsymbol{a} = \left(\mathrm{D}\left(\mathbf{Ad}_X\, \boldsymbol{b}\right)_X \hat{\boldsymbol{a}}\boldsymbol{X}\right)^\vee = \left[\boldsymbol{a}, \mathbf{Ad}_X\, \boldsymbol{b}\right], \tag{5.17c}$$

whereas at $\boldsymbol{X} = e$ they simplify to

$$(\mathrm{D}(\mathbf{Ad}_X\, \boldsymbol{b})_{X=e}\hat{\boldsymbol{a}})^\vee = \mathrm{d}^r(\mathbf{Ad}_X\, \boldsymbol{b})_{X=e}\boldsymbol{a} = \mathrm{d}^l\left(\mathbf{Ad}_X\, \boldsymbol{b}\right)_{X=e}\boldsymbol{a} = \left[\boldsymbol{a}, \boldsymbol{b}\right]. \tag{5.18}$$

The lower-case adjoint is defined as

$$\begin{aligned}
\mathrm{ad}_{\boldsymbol{a}}^0\, \boldsymbol{b} &:= \boldsymbol{b} \\
\mathrm{ad}_{\boldsymbol{a}}^1\, \boldsymbol{b} &:= \mathrm{ad}_{\boldsymbol{a}}\, \boldsymbol{b} = [\boldsymbol{a}, \boldsymbol{b}] \\
\mathrm{ad}_{\boldsymbol{a}}^2\, \boldsymbol{b} &:= [\boldsymbol{a}, \mathrm{ad}_{\boldsymbol{a}}\, \boldsymbol{b}] = \underbrace{[\boldsymbol{a}, [\boldsymbol{a}, \boldsymbol{b}]]}_{2-\text{times}} \\
&\vdots \\
\mathrm{ad}_{\boldsymbol{a}}^k\, \boldsymbol{b} &:= [\boldsymbol{a}, \mathrm{ad}_{\boldsymbol{a}}^{k-1}\, \boldsymbol{b}] = \underbrace{[\boldsymbol{a}, [\boldsymbol{a}, \dots, [\boldsymbol{a}, \boldsymbol{b}]]]}_{k\text{ times}}, \qquad k \geq 1.
\end{aligned} \tag{5.19}$$

From this definition it can be seen that for a scalar s,

$$\mathrm{ad}_{s\boldsymbol{a}}^k = s^k\, \mathrm{ad}_{\boldsymbol{a}}, \quad s \in \mathbb{R} \tag{5.20}$$

If we formally define the exponential of the adjoint as

$$\exp \mathrm{ad}_{\boldsymbol{a}} \coloneqq \sum_{k=0}^{\infty} \frac{\mathrm{ad}_{\boldsymbol{a}}^k}{k!} \tag{5.21}$$

we can also show that the adjoint of the exponential equals the exponential of the adjoint.

$$\mathbf{Ad}_{\exp \boldsymbol{a}} = \exp \mathrm{ad}_{\boldsymbol{a}}\,. \tag{5.22}$$

Proof of (5.22). By expanding the left-hand side in (5.22) and letting $\boldsymbol{A} = \hat{\boldsymbol{a}}, \boldsymbol{B} = \hat{\boldsymbol{b}}$ we obtain

$$(\mathbf{Ad}_{\exp \boldsymbol{a}} \boldsymbol{b})^\wedge = \operatorname{Exp}(\boldsymbol{A})\boldsymbol{B}\operatorname{Exp}(-\boldsymbol{A}) = \sum_{k=0}^{\infty}\sum_{i=0}^{k} \frac{\boldsymbol{A}^i\boldsymbol{B}(-\boldsymbol{A})^{k-i}}{i!(k-i)!}. \tag{5.23}$$

We next show by induction that the summands in (5.23) and (5.21) are equal for each value of k. Equality evidently holds for the base case $k = 0$. Assume that it holds for $k-1$, i.e. that

$$\left(\frac{\operatorname{ad}_{\boldsymbol{a}}^{k-1} \boldsymbol{b}}{(k-1)!}\right)^\wedge = \sum_{i=0}^{k-1} \frac{\boldsymbol{A}^i\boldsymbol{B}(-\boldsymbol{A})^{k-1-i}}{i!(k-1-i)!}. \tag{5.24}$$

Then we have that

$$\begin{aligned}\left(\frac{\operatorname{ad}_{\boldsymbol{a}}^{k} \boldsymbol{b}}{k!}\right)^\wedge &= \frac{1}{k}\left[\boldsymbol{A}\frac{(\operatorname{ad}_{\boldsymbol{A}}^{k-1}\boldsymbol{B})}{(k-1)!} - \frac{(\operatorname{ad}_{\boldsymbol{A}}^{k-1}\boldsymbol{B})}{(k-1)!}\boldsymbol{A}\right] = \frac{1}{k}\left[\sum_{i=0}^{k-1}\frac{\boldsymbol{A}^{i+1}\boldsymbol{B}(-\boldsymbol{A})^{k-1-i}}{i!(k-1-i)!} + \sum_{i=0}^{k-1}\frac{\boldsymbol{A}^{i}\boldsymbol{B}(-\boldsymbol{A})^{k-i}}{i!(k-1-i)!}\right]\\ &= \frac{1}{k}\left[\sum_{i=0}^{k-1}\frac{\boldsymbol{A}^{i}\boldsymbol{B}(-\boldsymbol{A})^{k-i}}{i!(k-1-i)!} + \sum_{i=1}^{k}\frac{\boldsymbol{A}^{i}\boldsymbol{B}(-\boldsymbol{A})^{k-i}}{(i-1)!(k-i)!}\right] = \frac{\boldsymbol{B}(-\boldsymbol{A})^k}{k!} + \sum_{i=1}^{k-1} c_i\boldsymbol{A}^i\boldsymbol{B}(-\boldsymbol{A})^{k-i} + \frac{\boldsymbol{A}^k\boldsymbol{B}}{k!},\end{aligned}$$

where $c_i = \frac{1}{k}\left(\frac{1}{i!(k-1-i)!} + \frac{1}{(i-1)!(k-i)!}\right)$ and it can be verified that $c_i = \frac{1}{i!(k-i)!}$ as required. □

As a consequence,

$$\mathbf{Ad}_{\lambda \exp(\boldsymbol{a})} \boldsymbol{a} = \exp \operatorname{ad}_{\lambda \boldsymbol{a}} \boldsymbol{a} = \boldsymbol{a}. \tag{5.25}$$

Another useful identity is the following.

$$\mathbf{Ad}_X [\boldsymbol{a}, \boldsymbol{b}] = [\mathbf{Ad}_X \boldsymbol{a}, \mathbf{Ad}_X \boldsymbol{b}]. \tag{5.26}$$

5.4. Derivatives of the Exponential map

The derivatives of the exponential map is a fundamental expression that often shows up when manipulating derivatives on Lie groups. From (5.10) we have that

$$\mathrm{d}^r \exp_{\boldsymbol{a}} \boldsymbol{b} = \left(\exp(\boldsymbol{\gamma}(0))^{-1} \left.\frac{\mathrm{d}}{\mathrm{d}t}\right|_{t=0} \exp(\boldsymbol{\gamma}(t))\right)^\vee, \quad \boldsymbol{\gamma}(0) = \boldsymbol{a},\ \boldsymbol{\gamma}'(0) = \boldsymbol{b}. \tag{5.27}$$

To calculate this derivative consider a curve $\boldsymbol{\gamma}(t) \in \check{\mathfrak{m}}$ and the expression

$$\Gamma(\sigma, t) = \exp(\sigma\boldsymbol{\gamma}(t))^{-1}\frac{\partial}{\partial t}\exp(\sigma\boldsymbol{\gamma}(t)) = \operatorname{Exp}\left(-\sigma\hat{\boldsymbol{\gamma}}(t)\right)\frac{\partial}{\partial t}\operatorname{Exp}(\sigma\hat{\boldsymbol{\gamma}}(t)). \tag{5.28}$$

Take the derivative with respect to σ:

$$\begin{aligned}\frac{\partial}{\partial\sigma}\Gamma(\sigma,t) &= -\operatorname{Exp}\left(-\sigma\hat{\boldsymbol{\gamma}}(t)\right)\hat{\boldsymbol{\gamma}}(t)\frac{\partial}{\partial t}\operatorname{Exp}\left(\sigma\hat{\boldsymbol{\gamma}}(t)\right) + \operatorname{Exp}(-\sigma\hat{\boldsymbol{\gamma}}(t))\frac{\partial}{\partial t}\left[\hat{\boldsymbol{\gamma}}(t)\operatorname{Exp}(\sigma\hat{\boldsymbol{\gamma}}(t))\right]\\ &= \operatorname{Exp}(-\sigma\hat{\boldsymbol{\gamma}}(t))\hat{\boldsymbol{\gamma}}'(t)\operatorname{Exp}(\sigma\hat{\boldsymbol{\gamma}}(t)) = \operatorname{Ad}_{\operatorname{Exp}(-\sigma\hat{\boldsymbol{\gamma}}(t))}\hat{\boldsymbol{\gamma}}'(t) = \left(\mathbf{Ad}_{\exp(-\sigma\hat{\boldsymbol{\gamma}}(t))}\boldsymbol{\gamma}'(t)\right)^\wedge\\ &\overset{(5.22)}{=} \left(\exp\operatorname{ad}_{-\sigma\boldsymbol{\gamma}(t)}\boldsymbol{\gamma}'(t)\right)^\wedge = \left(\sum_{k=0}^{\infty}\frac{\operatorname{ad}_{-\sigma\boldsymbol{\gamma}(t)}^k}{k!}\boldsymbol{\gamma}'(t)\right)^\wedge = \left(\sum_{k=0}^{\infty}\sigma^k\frac{\operatorname{ad}_{-\boldsymbol{\gamma}(t)}^k}{k!}\boldsymbol{\gamma}'(t)\right)^\wedge.\end{aligned} \tag{5.29}$$

Integrating from 0 to 1 with respect to σ and setting $t = 0$ then yields

$$\Gamma(1,0)^\vee = \int_0^1 \frac{\partial}{\partial\sigma}\Gamma(\sigma,0)^\vee \mathrm{d}\sigma = \sum_{k=0}^{\infty} \frac{\operatorname{ad}^k_{-\boldsymbol{\gamma}(0)}}{(k+1)!}\boldsymbol{\gamma}'(0). \tag{5.30}$$

From (5.28) we can see that $\Gamma(1,0)$ is equal to the right derivative of exp at $\gamma(0)$ in the direction $\gamma'(0)$.

The right- and left derivatives of the exponential map are

$$\begin{aligned} \mathrm{d}^r \exp_{\boldsymbol{a}} &= \frac{I - \exp(-\operatorname{ad}_{\boldsymbol{a}})}{\operatorname{ad}_{\boldsymbol{a}}} := \sum_{k=0}^{\infty} \frac{(-1)^k}{(k+1)!} \operatorname{ad}^k_{\boldsymbol{a}}, \\ \mathrm{d}^l \exp_{\boldsymbol{a}} &= \frac{\exp \operatorname{ad}_a - I}{\operatorname{ad}_a} := \sum_{k=0}^{\infty} \frac{1}{(k+1)!} \operatorname{ad}^k_{\boldsymbol{a}}. \end{aligned} \tag{5.31}$$

Through the Bernoulli numbers $B_0 = 0, B_1 = -1/2, B_2 = 1/6, \ldots$ that are defined as

$$\frac{t}{e^t - 1} = \sum_{n=0}^{\infty} \frac{B_n}{n!} t^n. \tag{5.32}$$

we can also write the formal inverses of the derivatives of the exponential

$$\begin{aligned} \left(\mathrm{d}^r \exp_{\boldsymbol{a}}\right)^{-1} &= \frac{\operatorname{ad}_{\boldsymbol{a}}}{I - \exp(-\operatorname{ad}_{\boldsymbol{a}})} := \sum_{n=0}^{\infty} B_n \frac{(-1)^n}{n!} \operatorname{ad}^n_{\boldsymbol{a}}, \\ \left(\mathrm{d}^l \exp_{\boldsymbol{a}}\right)^{-1} &= \frac{\operatorname{ad}_{\boldsymbol{a}}}{\exp \operatorname{ad}_{\boldsymbol{a}} - I} := \sum_{n=0}^{\infty} B_n \frac{1}{n!} \operatorname{ad}^n_{\boldsymbol{a}}. \end{aligned} \tag{5.33}$$

Notably, at $\boldsymbol{a} = \mathbf{0}$ these are all equal to the identity matrix. Since $B_1 = -1/2$ and $B_n = 0$ for odd $n > 1$ it follows that

$$\left(\mathrm{d}^l \exp_{\boldsymbol{a}}\right)^{-1} = -\operatorname{ad}_{\boldsymbol{a}} + \left(\mathrm{d}^r \exp_{\boldsymbol{a}}\right)^{-1}. \tag{5.34}$$

From these definitions it follows that

$$[\boldsymbol{a}, \boldsymbol{b}] = 0 \quad \Longrightarrow \quad \mathrm{d}^r \exp_{\boldsymbol{b}} \boldsymbol{a} = \mathrm{d}^l \exp_{\boldsymbol{b}} = \left(\mathrm{d}^r \exp_{\boldsymbol{b}}\right)^{-1} = \left(\mathrm{d}^l \exp_{\boldsymbol{b}}\right)^{-1} = \boldsymbol{a}, \tag{5.35}$$

which in particular holds for the case $\boldsymbol{b} = \lambda \boldsymbol{a}$.

While the derivatives (5.31) - (5.33) could be evaluated to arbitrary precision by adding enough terms, this is not a practical solution. Fortunately closed-form expressions can be obtained for all groups of interest, but we postpone those calculations to the next part.

5.5. Derivatives of common operations

Group composition We calculate the right derivatives using (5.1) and the left derivatives via (5.7).

$$\begin{aligned}\mathrm{d}^r(X \circ Y)_X &\overset{(5.1)}{=} \lim_{a \to 0} \frac{\log\left((X \circ Y)^{-1} \circ X \circ \exp(a) \circ Y\right)}{a} = \lim_{a \to 0} \frac{\log\left(Y^{-1} \circ \exp(a) \circ Y\right)}{a} \\ &\overset{(4.15c)}{=} \lim_{a \to 0} \frac{\log \exp \mathbf{Ad}_{Y^{-1}}\, a}{a} = \mathbf{Ad}_{Y^{-1}},\end{aligned} \tag{5.36}$$

$$\mathrm{d}^l(X \circ Y)_X \overset{(5.7)}{=} \mathbf{Ad}_{X \circ Y}\, \mathbf{Ad}_{Y^{-1}}\, \mathbf{Ad}_X^{-1} \overset{(4.15)}{=} I_n, \tag{5.37}$$

$$\mathrm{d}^r(X \circ Y)_Y \overset{(5.1)}{=} \lim_{a \to 0} \frac{\log\left((X \circ Y)^{-1} \circ X \circ Y \circ \exp(a)\right)}{a} = I_n, \tag{5.38}$$

$$\mathrm{d}^l(X \circ Y)_Y \overset{(5.7)}{=} \mathbf{Ad}_{X \circ Y}\, I_n\, \mathbf{Ad}_Y^{-1} \overset{(4.15)}{=} \mathbf{Ad}_X\,. \tag{5.39}$$

Group inverse

$$\begin{aligned}\mathrm{d}^r(X^{-1})_X &\overset{(5.1)}{=} \lim_{a \to 0} \frac{\log\left(X \circ (X \circ \exp(a))^{-1}\right)}{a} = \lim_{a \to 0} \frac{\log\left(X \circ \exp(-a) \circ X^{-1}\right)}{a} \\ &\overset{(4.15c)}{=} \frac{\log \exp \mathbf{Ad}_X -a}{a} = -\mathbf{Ad}_X\,.\end{aligned} \tag{5.40}$$

$$\mathrm{d}^l(X^{-1})_X \overset{(5.7)}{=} -\mathbf{Ad}_{X^{-1}}\, \mathbf{Ad}_X\, \mathbf{Ad}_{X^{-1}} = -\mathbf{Ad}_{X^{-1}}\,. \tag{5.41}$$

Logarithm From differentiating $a = \log \exp a$ using the chain rule we get $I = \mathrm{d}^r \log_{\exp a}\ \mathrm{d}^r \exp_a$, which implies that

$$\mathrm{d}^r \log_X = \left[\mathrm{d}^r \exp_{\log X}\right]^{-1}, \tag{5.42}$$

$$\mathrm{d}^l \log_X = \left[\mathrm{d}^l \exp_{\log X}\right]^{-1}. \tag{5.43}$$

$$\tag{5.44}$$

Plus and minus From the chain rule and the above we can also deduce the derivatives of the plus and minus maps

$$\mathrm{d}^r(X \oplus_r a)_X = \mathrm{d}^r(X \circ \exp(a))_X \overset{(5.36)}{=} \mathbf{Ad}_{\exp(a)}^{-1}, \tag{5.45}$$

$$\mathrm{d}^r(X \oplus_r a)_a = \mathrm{d}^r(X \circ \exp(a))_a \overset{(5.8)}{=} \mathrm{d}^r(X \circ \exp(a))_{\exp a}\ \mathrm{d}^r \exp_a \overset{(5.38)}{=} \mathrm{d}^r \exp_a, \tag{5.46}$$

$$\mathrm{d}^r(Y \ominus_r X)_Y = \mathrm{d}^r\left(\log X^{-1} \circ Y\right)_Y \overset{(5.8)}{=} \mathrm{d}^r \log_{X^{-1} \circ Y}\ \mathrm{d}^r(X^{-1} \circ Y)_Y \overset{(5.38)}{=} \left[\mathrm{d}^r \exp_{Y \ominus_r X}\right]^{-1}, \tag{5.47}$$

$$\begin{aligned}\mathrm{d}^r(Y \ominus_r X)_X &= \mathrm{d}^r\left(\log X^{-1} \circ Y\right)_X \overset{(5.8)}{=} \mathrm{d}^r \log_{X^{-1} \circ Y}\ \mathrm{d}^r(X^{-1} \circ Y)_{X^{-1}}\ \mathrm{d}^r(X^{-1})_X \\ &\overset{(5.42),(5.36),(5.40)}{=} \left[\mathrm{d}^r \exp_{Y \ominus_r X}\right]^{-1} \mathbf{Ad}_{Y^{-1}}(-\mathbf{Ad}_X) \overset{(4.15b)}{=} -\left[\mathrm{d}^r \exp_{Y \ominus_r X}\right]^{-1} \mathbf{Ad}_{Y^{-1} \circ X} \\ &= -\left[\mathrm{d}^r \exp_{Y \ominus_r X}\right]^{-1} \mathbf{Ad}_{\exp Y \ominus_r X}^{-1} \overset{(5.7)}{=} -\left[\mathrm{d}^l \exp_{Y \ominus_r X}\right]^{-1}.\end{aligned} \tag{5.48}$$

5.6. On Automatic Differentiation

Consider a function $f : \mathbb{M} \to \mathbb{N}$ whose derivative we are interested in. Since autodiff tools are not aware of manifolds we can not directly obtain e.g. $\mathrm{d}^r f_X$; here we discuss how to obtain on-manifold derivatives by only differentiating Euclidean functions. Since $\mathrm{d}^r(f(X \oplus_r \boldsymbol{a}))_{\boldsymbol{a}=0} = \mathrm{d}^r f_X \mathrm{d}\exp_0 = \mathrm{d}^r f_X$ we can write

$$\mathrm{d}^r f_X \boldsymbol{b} = \mathrm{d}^r(f(X \oplus_r \boldsymbol{a}))_{\boldsymbol{a}=0}\boldsymbol{b} = \left(f(X \oplus_r \boldsymbol{a})^{-1} \mathrm{D}(f(X \oplus_r \boldsymbol{a}))_{\boldsymbol{a}} \hat{\boldsymbol{b}}\right)^{\vee}\Big|_{\boldsymbol{a}=0} = \left(f(X)^{-1} \frac{\mathrm{d}}{\mathrm{d}t}\Big|_{t=0} f(X \oplus (t\boldsymbol{b}))\right)^{\vee}. \tag{5.49}$$

Here the function $t \mapsto f(X \oplus (t\boldsymbol{b}))$ maps a scalar to a matrix and can therefore be differentiated using regular tools, after which the expression can be evaluated to obtain $\mathrm{d}^r f_X \boldsymbol{b}$. Naturally, if the complete derivative $\mathrm{d}^r f_X$ is desired it can be obtained by repeating this procedure n times for each basis unit vector.

If f maps to a Euclidean space (i.e. $\mathbb{N} = \mathbb{R}^k$) this further simplifies to

$$\mathrm{d}^r f_X b = \frac{\mathrm{d}}{\mathrm{d}t}\Big|_{t=0} f(X \oplus (t\boldsymbol{b})), \quad f : \mathbb{M} \to \mathbb{R}^n, \tag{5.50}$$

which with some abuse of notation can be written as

$$\mathrm{d}^r f_X = \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{b}}\Big|_{\boldsymbol{b}=0} f(X \oplus \boldsymbol{b}). \tag{5.51}$$

Pretty sure this will immediately yield the correct derivative

$$\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{b}}\Big|_{\boldsymbol{b}=0} \log\left(f(X)^{-1} f(X \oplus \boldsymbol{b})\right) \tag{5.52}$$

5.6.1. Ceres Solver Local Parameterizations

A special case of when numerical derivatives are used is in the nonlinear optimizer Ceres. Being unaware of Lie groups, Ceres considers cost functions that are functions of some parameters, $f(\boldsymbol{x})$, and uses automatic differentiation of $f : \mathbb{R}^p \to \mathbb{R}^k$ with respect to $\boldsymbol{x} \in \mathbb{R}^p$ to figure out in what direction to move in order to minimize f. However, if $\boldsymbol{x}$ represents the coordinates of a manifold chart, i.e. $\hat{\boldsymbol{x}} = X$ for $X \in \mathbb{M}$, it is not desirable to directly apply an update in the direction of the gradient since this may lead the resulting point no longer being on the manifold.

Being unaware of the manifold structure, automatic differentiation can only evaluate $\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{x}} f(\hat{\boldsymbol{x}})$, where we are using the hat and vee maps to denote conversions between elements of a Lie group $\mathbb{M}$ and its parameterization $\check{\mathbb{M}}$ as was done in Chapter 2. Ceres provides an interface for specifying cusom *local parameterizations* that enable on-manifold optimization. In the following we specify how a local parameterization for Lie Group optimization can be constructed.

According to (5.51) we can write a tangent space derivative for a Euclidean-valued function

$$\mathrm{d}^r f_X = \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{b}}\Big|_{\boldsymbol{b}=0} f(X \oplus \boldsymbol{b}) = \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{b}}\Big|_{\boldsymbol{b}=0} f\left(\left((\hat{\boldsymbol{x}} \oplus \boldsymbol{b})^{\vee}\right)^{\wedge}\right) = \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{y}}\Big|_{\boldsymbol{y}=\boldsymbol{x}} f(\hat{\boldsymbol{y}}) \times \frac{\mathrm{d}}{\mathrm{d}\boldsymbol{b}}\Big|_{\boldsymbol{b}=0} (\hat{\boldsymbol{x}} \oplus \boldsymbol{b})^{\vee}. \tag{5.53}$$

Thus, if $\frac{\mathrm{d}}{\mathrm{d}y}\Big|_{y=x} f(\hat{y})$ is obtained through automatic differentiation it needs to be right-multiplied by a state-dependent matrix in order to obtain the tangent-space derivative. In Ceres parlance these matrices are called

- Local derivative: $\mathrm{d}^r f_X$, a $k \times n$ matrix,
- Global derivative: $\frac{\mathrm{d}}{\mathrm{d}y}\Big|_{y=x} f(\hat{y})$, a $k \times p$ matrix,
- Jacobian: $\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{b}}\Big|_{\boldsymbol{b}=0} (\hat{\boldsymbol{x}} \oplus \boldsymbol{b})^{\vee}$, a $p \times n$ matrix

and it holds that (local derivative) = (global derivative) × (jacobian). The local parameterization for a Lie group can be specified as follows:

- Plus operation: $\boldsymbol{x} \boxplus \boldsymbol{b} \coloneqq (\hat{\boldsymbol{x}} \oplus \boldsymbol{b})^{\vee}$,
- Local dimension: $n = \|T\mathbb{M}\|$ tangent space dimension,
- Global dimension: $p = \|\check{\mathbb{M}}\|$ group parameterization dimension,
- Jacobian: $\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{b}}\Big|_{\boldsymbol{b}=0} (\hat{\boldsymbol{x}} \oplus \boldsymbol{b})^{\vee}$.

6. Dynamical Systems on Lie Groups

Having defined Lie group derivatives a logical next step is to consider differential equations on Lie groups, i.e. solutions $\boldsymbol{x}(t)$ to

$$\begin{aligned} \mathrm{d}^r \boldsymbol{x}_t &= f(t, \boldsymbol{x}(t)), \\ \boldsymbol{x}(0) &= \boldsymbol{x}_0. \end{aligned} \tag{6.1}$$

In this chapter we study various properties of this system: its linearization, sensitivity with respect to initial conditions, monotonicity, and finally a method to analyze it via a system on $\mathbb{R}^n$.

For a given initial condition $\boldsymbol{x}_0$ the solution of (6.1) at time $t \geq 0$ can be denoted $\phi(t; \boldsymbol{x}_0)$ where the *flow* operator $\phi : \mathbb{R} \times \mathbb{M} \to \mathbb{M}$ is s.t.

$$\begin{aligned} \phi(0; \boldsymbol{x}_0) &= \boldsymbol{x}_0, \\ \mathrm{d}^r \phi(t; \boldsymbol{x}_0)_t &= f\left(t, \phi(t; \boldsymbol{x}_0)\right). \end{aligned} \tag{6.2}$$

Remark 6.1. *Parameters and initial conditions are equivalent. A parameter-dependent system*

$$\begin{aligned} \mathrm{d}^r \boldsymbol{x}_t &= f(t, \boldsymbol{x}; p_0), \\ \boldsymbol{x}(t_0) &= \boldsymbol{x}_0, \end{aligned} \tag{6.3}$$

is equivalent to the parameter-free system on $\mathbb{M} \times \mathbb{R}^n$

$$\begin{aligned} &\mathrm{d}^r(\boldsymbol{x}, p)_t = \left(\mathrm{g}(t, \boldsymbol{x}, p), 0\right), \qquad \mathrm{g}(t, \boldsymbol{x}, p) := f(t, \boldsymbol{x}; p), \\ &(\boldsymbol{x}, p)(t_0) = (\boldsymbol{x}_0, p_0), \end{aligned} \tag{6.4}$$

where $\mathrm{g}(t, \boldsymbol{x}, p) = f(t, \boldsymbol{x}; p)$*. Conversely, the system*

$$\begin{aligned} \mathrm{d}^r \boldsymbol{x}_t &= f(t, \boldsymbol{x}), \\ \boldsymbol{x}(t_0) &= \boldsymbol{x}_0, \end{aligned} \tag{6.5}$$

with a non-trivial initial condition is (locally) equivalent to the parameter-dependent system

$$\begin{aligned} &\mathrm{d}^r \boldsymbol{a}_t = \mathrm{g}(t, \boldsymbol{a}; t_0, \boldsymbol{x}_0), \qquad \mathrm{g}(t, \boldsymbol{a}; t_0, \boldsymbol{x}_0) := \left[\mathrm{d}^r \exp_{\boldsymbol{a}}\right]^{-1} f(t_0 + t, \boldsymbol{x}_0 \oplus_r \boldsymbol{a}), \\ &\boldsymbol{a}(0) = 0, \end{aligned} \tag{6.6}$$

with trivial initial conditions, in the sense that $\Phi^{\boldsymbol{x}}(t_0 + t; t_0, \boldsymbol{x}_0) = \boldsymbol{x}_0 \oplus_r \Phi^{\boldsymbol{a}}(t; t_0, \boldsymbol{x}_0)$*.*

6.1. Tangent Space Linearization

We start by considering the dynamics around a nominal trajectory $(X_l(t), u_l(t))$. Consider the difference

$$\boldsymbol{a}_e = X(t) \ominus_r X_l(t) \tag{6.7}$$

between the state of (6.1) and the nominal trajectory. Since $\boldsymbol{a}_e$ takes values in $T_{X_l(t)}\mathbb{M} \cong \mathbb{R}^n$ the rule of total derivatives in Remark 5.1 applies and the time derivative of $\boldsymbol{a}_e$ is

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{a}_e}{\mathrm{d}t} &= \mathrm{d}^r(\boldsymbol{a}_e)_t = \mathrm{d}^r(X \ominus_r X_l)_X \mathrm{d}^r X_t + \mathrm{d}^r(X \ominus_r X_l)_{X_l}\, \mathrm{d}^r(X_l)_t \\ &\overset{(5.47),(5.48)}{=} \left[\mathrm{d}^r \exp_{\boldsymbol{a}_e}\right]^{-1} f\left(X_l \oplus_r \boldsymbol{a}_e, u_l + u_e\right) - \left[\mathrm{d}^l \exp_{\boldsymbol{a}_e}\right]^{-1}\, \mathrm{d}^r(X_l)_t, \end{aligned} \tag{6.8}$$

Setting $(\boldsymbol{a}_e, u_e) = (0, 0)$ yields the linear time-varying system

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{a}_e = A(t)\boldsymbol{a}_e + B(t)u_e + E(t), \tag{6.9}$$

where, since $\mathrm{d}^r \exp_0 = \mathrm{d}^l \exp_0 = I$,

$$A(t) \coloneqq \left.\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}_e}\right|_{\boldsymbol{a}_e=0} \left[\mathrm{d}^r \exp_{\boldsymbol{a}_e}\right]^{-1} f\left(X_l(t) \oplus_r \boldsymbol{a}_e, u_l(t)\right), \tag{6.10}$$

$$B(t) \coloneqq \left.\frac{\mathrm{d}}{\mathrm{d}u_e}\right|_{u_e=0} f\left(X_l(t), u_l(t) + u_e\right), \tag{6.11}$$

$$E(t) \coloneqq f\left(X_l(t), u_l(t)\right) - \mathrm{d}^r(X_l)_t. \tag{6.12}$$

May be able to simplify derivative of dexpinv further

6.2. Sensitivity Analysis

Next we study the sensitivity of $\phi(t; \boldsymbol{x}_0)$ with respect to the initial conditon. Due to the equivalence between parameters and initial conditions as discussed in Remark 6.1, sensitivity with respect to the initial conditons can also be used to figure the sensitivity with respect to paramters.

Global derivative on matrix form $\Phi = \hat{\phi}$.

$$\dot{\Phi}(t; \boldsymbol{x}_0) = \Phi(t; \boldsymbol{x}_0)\left(\mathrm{d}^r\Phi(t; \boldsymbol{x}_0)_t\right)^{\wedge} = \Phi(t; \boldsymbol{x}_0)\hat{f}\left(t, \Phi(t; \boldsymbol{x}_0)\right). \tag{6.13}$$

Derivative of inverse

$$\begin{aligned} 0 = \frac{\mathrm{d}}{\mathrm{d}t}\Phi(t; \boldsymbol{x}_0) \circ \Phi(t; \boldsymbol{x}_0)^{-1} &= \dot{\Phi}(t; \boldsymbol{x}_0)\Phi(t; \boldsymbol{x}_0)^{-1} + \Phi(t; \boldsymbol{x}_0)\frac{\mathrm{d}}{\mathrm{d}t}\Phi(t; \boldsymbol{x}_0)^{-1} \\ \implies \quad & \frac{\mathrm{d}}{\mathrm{d}t}\phi(t; \boldsymbol{x}_0)^{-1} = \Phi(t; \boldsymbol{x}_0)^{-1}\dot{\Phi}(t; \boldsymbol{x}_0)\Phi(t; \boldsymbol{x}_0)^{-1}. \end{aligned} \tag{6.14}$$

We can then evaluate how $\mathrm{d}^r\Phi(t;\boldsymbol{x}_0)_t$ depends on t by moving to global derivatives and changing the order of integration.

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathrm{d}^r\Phi(t;\boldsymbol{x}_0)_{\boldsymbol{x}_0}\boldsymbol{a}\right) &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\Phi(t;\boldsymbol{x}_0)^{-1}\left.\frac{\mathrm{d}}{\mathrm{d}\tau}\right|_{\tau=0}\Phi(t;\boldsymbol{x}_0\oplus\tau\boldsymbol{a})\right)^\vee \\
&= \left(\left(-\Phi(t;\boldsymbol{x}_0)^{-1}\dot{\Phi}(t;\boldsymbol{x}_0)\Phi(t;\boldsymbol{x}_0)\right)^{-1}\frac{\mathrm{d}}{\mathrm{d}\tau}\Phi(t;\boldsymbol{x}_0\oplus\tau\boldsymbol{a}) + \Phi(t;\boldsymbol{x}_0)^{-1}\left.\frac{\mathrm{d}}{\mathrm{d}\tau}\right|_{\tau=0}\dot{\Phi}(t;\boldsymbol{x}_0\oplus\tau\boldsymbol{a})\right)^\vee \\
&= \left(-\hat{f}\left(t;\Phi(t;\boldsymbol{x}_0)\right)\Phi(t;\boldsymbol{x}_0)^{-1}\left.\frac{\mathrm{d}}{\mathrm{d}\tau}\right|_{\tau=0}\Phi(t;\boldsymbol{x}_0\oplus\tau\boldsymbol{a})\right)^\vee \\
&+ \left(\Phi(t;\boldsymbol{x}_0)^{-1}\left.\frac{\mathrm{d}}{\mathrm{d}\tau}\right|_{\tau=0}\Phi(t;\boldsymbol{x}_0\oplus\tau\boldsymbol{a})\hat{f}(t;\Phi(t;\boldsymbol{x}_0\oplus\tau\boldsymbol{a}))\right)^\vee \\
&= -\left(\hat{f}(t;\Phi(t;\boldsymbol{x}_0))\left(\mathrm{d}^r\Phi(t;\boldsymbol{x}_0)_{\boldsymbol{x}_0}\boldsymbol{a}\right)^\wedge + \left(\mathrm{d}^r\Phi(t;\boldsymbol{x}_0)_{\boldsymbol{x}_0}\boldsymbol{a}\right)^\wedge\hat{f}(t;\Phi(t;\boldsymbol{x}_0)) + \left.\frac{\mathrm{d}}{\mathrm{d}\tau}\right|_{\tau=0}\hat{f}(t;\Phi(t;\boldsymbol{x}_0\oplus\tau\boldsymbol{a}))\right)^\vee \\
&= -\left[f(t;\Phi(t;\boldsymbol{x}_0)),\mathrm{d}^r\Phi(t;\boldsymbol{x}_0)_{\boldsymbol{x}_0}\boldsymbol{a}\right] + \mathrm{d}^r f(t,\Phi(t;\boldsymbol{x}_0))_{\boldsymbol{x}_0}\boldsymbol{a} \\
&= -\operatorname{ad}_{f(t;\Phi(t;\boldsymbol{x}_0))}\mathrm{d}^r\Phi(t;\boldsymbol{x}_0)_{\boldsymbol{x}_0}\boldsymbol{a} + \mathrm{d}^r f(t,\boldsymbol{x})_{\boldsymbol{x}=\Phi(t;\boldsymbol{x}_0)}\mathrm{d}^r\Phi(t;\boldsymbol{x}_0))_{\boldsymbol{x}_0}\boldsymbol{a}.
\end{aligned}
$$

The sensitivity $S(t) \coloneqq \mathrm{d}^r\Phi(t;\boldsymbol{x}_0)_{\boldsymbol{x}_0}$ satisfies the matrix-valued ODE

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}S(t) &= \left(-\operatorname{ad}_{f(t,\Phi(t;\boldsymbol{x}_0))} + \mathrm{d}^r f_{\boldsymbol{x}}\big|_{\boldsymbol{x}=\Phi(t;\boldsymbol{x}_0)}\right)S(t), \\
S(0) &= I.
\end{aligned} \tag{6.15}
$$

6.2.1. Example

Example 6.1

If $\mathrm{d}^r\boldsymbol{x}_t = f(\boldsymbol{x}) \equiv \boldsymbol{a}$, then $\boldsymbol{x}(t) = \boldsymbol{x}_0\exp(t\boldsymbol{a})$ and we get

$$\mathrm{d}^r(\boldsymbol{x}(t))_{\boldsymbol{x}_0} \overset{(5.36)}{=} \mathbf{Ad}_{\exp(-t\boldsymbol{a})}\,. \tag{6.16}$$

We furthermore know from Lemma 4.1 that

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{Ad}_{\exp(-t\boldsymbol{a})} = -\operatorname{ad}_{\boldsymbol{a}}\mathbf{Ad}_{\exp(-t\boldsymbol{a})}, \tag{6.17}$$

i.e. the sensitivity equations are

$$\frac{\mathrm{d}}{\mathrm{d}t}S(t) = -\operatorname{ad}_{\boldsymbol{a}}S(t), \tag{6.18}$$

which was expected from (6.15) since f is constant.

6.3. Monotonicity

Very much a work in progress

Monotonicity is a useful property of dynamical systems that can be leveraged in order to bound the envelope of possible behaviors by a small number of extremal trajectories. For instance, a forward-traveling vehicle that is trying to stop is always better of the less it accelerates, which means that it is sufficient to analyze it's minimal acceleration in order to determine whether it can stop in time.

Monotonicity on $\mathbb{R}^n$ Monotonicity is usually defined with respect to a *cone*—a set with the property that $0 \in K$ and $\boldsymbol{x} \in K \implies \alpha K \in K$ for $\alpha \geq 0$. For a cone we can define an ordering $\preceq_K$ such that

$$x \preceq_K y \quad \iff \quad y - x \in K. \tag{6.19}$$

Monotonicity of a function f can then be defined as the following property:

$$x \preceq_K y \quad \implies \quad f(x) \preceq_K f(y). \tag{6.20}$$

Monotonicity on Lie groups The usual notion of monotonicity only applies for *ordered spaces*, which is a property that is not present in the usual Lie groups used in robotics. Indeed, for a circle ordering makes little sense. However, the tangent space of a Lie group is monotone which makes it possible to define a notion of *local monotonicity* in a way that is analogous to the Euclidean case.

Definition 6.1. *A function $f : \mathbb{M} \to \mathbb{N}$ is locally monotone around $\boldsymbol{Z} \in \mathbb{M}$ with respect to a cone $K \subset \mathfrak{m} \cong \mathbb{R}^n$ if for all $\boldsymbol{a}, \boldsymbol{b}$ that are sufficiently small it holds that*

$$\boldsymbol{a} \preceq_K \boldsymbol{b} \quad \implies \quad f(\boldsymbol{Z} \oplus_r \boldsymbol{a}) \ominus_r f(\boldsymbol{Z}) \preceq_K f(\boldsymbol{Z} \oplus_r \boldsymbol{b}) \ominus f(\boldsymbol{Z}). \tag{6.21}$$

When $\mathbb{M}$ and $\mathbb{N}$ are Euclidean spaces $\boldsymbol{Z}$ can be set to zero to retrieve the original definition.

- Define mixed monotonicity corresponding to Def (6.1).
- Derive a jacobian condition on f for mixed monotonicity that is analogous to sign-stability?
- Create a dynamical system that over-approximates reach sets of one of these forms:
  - MID-DOWN-UP: $\mathrm{A}(\boldsymbol{X}, l, u) = \{\boldsymbol{Y} : l \preceq_K \boldsymbol{Y} \ominus_r \boldsymbol{X} \preceq_K u\}$
  - MID-SINGLE: $\mathrm{A}(\boldsymbol{X}, k) = \{\boldsymbol{Y} : -k \preceq_K \boldsymbol{Y} \ominus_r \boldsymbol{X} \preceq_K k\}$
  - MID-RADIUS: $\mathrm{A}(\boldsymbol{X}, r) = \{\boldsymbol{Y} : \|\boldsymbol{Y} \ominus_r \boldsymbol{X}\| < r\}$
  - The values l, u need to be twisted as part of the mapping

The derivative of the mapping $\boldsymbol{a} \mapsto f(\boldsymbol{Z} \oplus_r \boldsymbol{a}) \ominus f(\boldsymbol{Z})$ from $\mathbb{R}^n$ to $\mathbb{R}^m$ is

$$\mathrm{d}^r \left(f(\boldsymbol{Z} \oplus_r \boldsymbol{a}) \ominus f(\boldsymbol{Z}) \right)_{\boldsymbol{a}} = \left[\mathrm{d}^r \exp_{f(\boldsymbol{Z} \oplus_r \boldsymbol{a}) \ominus f(\boldsymbol{Z})} \right]^{-1} \mathrm{d}^r f_{\boldsymbol{Z} \oplus \boldsymbol{a}} \, \mathrm{d}^r \exp_{\boldsymbol{a}} \tag{6.22}$$

It follows that this is what we need to make sign-stable, so the decomposition should depend on it. Challenge is that the decomposition may have to depend on both $\boldsymbol{a}$ and on $\boldsymbol{Z}$.

Reach mapping: Set $\{X : \underline{\boldsymbol{a}} \preceq_K X \ominus_r Z \preceq_K \overline{\boldsymbol{a}}\}$
Decomposition function g s.t. $f(Z \oplus \boldsymbol{a}) \ominus f(Z) = g_Z(\boldsymbol{a}, \boldsymbol{a})$
Mapped set:

$$\left\{X : g_Z(\underline{\boldsymbol{a}}, \overline{\boldsymbol{a}}) \preceq_K X \ominus f(Z) \preceq_K g_Z(\overline{\boldsymbol{a}}, \underline{\boldsymbol{a}})\right\}. \tag{6.23}$$

- How to go from monotonicity of $f : \mathbb{M} \to \mathbb{R}^m$ to monotonicity of the flow $\phi : \mathbb{M} \to \mathbb{M}$?
- How is decomposition function done in practice? Like in Necmiyes paper?

6.4. The Magnus Expansion

For the case when the right-hand side in (6.1) only depends on t,

$$\mathrm{d}^r \boldsymbol{x}_t = \boldsymbol{a}(t), \tag{6.24}$$

we can posit that the solution be on the form

$$\boldsymbol{x}(t) = \exp(\Omega(t)), \quad \Omega(t) \in \mathbb{M}. \tag{6.25}$$

From the differentiation rules it follows that

$$\boldsymbol{a}(t) = \mathrm{d}^r \boldsymbol{x}_t = \mathrm{d}^r \exp_{\Omega(t)} \frac{\mathrm{d}}{\mathrm{d}t} \Omega(t), \tag{6.26}$$

which yields an ODE for $\Omega(t)$.

Theorem 6.1. *The solution of the time-varying ODE (6.24) is given by*

$$\boldsymbol{x}(t) = \exp(\Omega(t)) \boldsymbol{x}_0, \tag{6.27}$$

where $\Omega(t)$ satisfies the initial-value problem

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} \Omega(t) &= \left(\mathrm{d}^r \exp_{\Omega(t)}\right)^{-1} \boldsymbol{a}(t), \\ \Omega(0) &= 0. \end{aligned} \tag{6.28}$$

The initial value problem for Ω may still be challenging to solve in case an expression for $\left(\mathrm{d}^r \exp_{\boldsymbol{a}}\right)^{-1}$ is not available. The **Magnus expansion** is obtained by setting $\boldsymbol{a} = \epsilon \tilde{\boldsymbol{a}}$ and expressing Ω as a series

$$\Omega(t) = \sum_{k \geq 1} \epsilon^k \Omega_k(t). \tag{6.29}$$

Inserting this in (6.28) and comparing powers of ϵ yields

$$\begin{aligned} \Omega_1(t) &= \int_0^t \boldsymbol{a}(s_1) \mathrm{d}s_1, \\ \Omega_2(t) &= -\frac{1}{2} \int_0^t [\Omega_1(s_1), \boldsymbol{a}(s_1)] \, \mathrm{d}s_1 = \frac{1}{2} \int_0^t \int_0^{s_2} [\boldsymbol{a}(s_1), \boldsymbol{a}(s_2)] \, \mathrm{d}s_2 \mathrm{d}s_1, \\ \Omega_3(t) &= \frac{1}{6} \int_0^t \int_0^{s_1} \int_0^{s_2} [\boldsymbol{a}(s_1), [\boldsymbol{a}(s_2), \boldsymbol{a}(s_3)]] + [[\boldsymbol{a}(s_1), \boldsymbol{a}(s_2)], \boldsymbol{a}(s_3)] \, \mathrm{d}s_3 \mathrm{d}s_2 \mathrm{d}s_1, \end{aligned} \tag{6.30}$$

and so on for higher powers of k.

6.4.1. Example

Consider the initial value problem on $\mathbb{SE}(2)$:

$$\dot{\boldsymbol{X}}(t) = \boldsymbol{A}(t)\boldsymbol{X}(t), \quad \boldsymbol{X}(0) = \boldsymbol{X}_0, \quad \boldsymbol{X} \in \mathbb{SE}(2), \quad \boldsymbol{A} \in \mathfrak{se}(2). \tag{6.31}$$

We assume that $\boldsymbol{A}(t) = \hat{\boldsymbol{a}}(t)$ is a known curve, i.e.

$$\boldsymbol{A}(t) = \begin{bmatrix} 0 & -\theta(t) & u(t) \\ \theta(t) & 0 & v(t) \\ 0 & 0 & 0 \end{bmatrix} \tag{6.32}$$

According to Theorem 6.1 the solution is then

$$\boldsymbol{x}(t) = \operatorname{Exp}(\Omega(t))\boldsymbol{x}_0 = \operatorname{Exp}\left(\sum_{k\geq 0} \Omega_k(t)\right)\boldsymbol{x}_0. \tag{6.33}$$

The Lie algebra $\mathfrak{se}(2)$ is not nilpotent, so the exact solution requires the full Magnus expansion. Below we develop an approximate solution corresponding to the first two terms

$$\boldsymbol{x}(t) \approx \operatorname{Exp}\left(\int_0^t A(t)\mathrm{d}t + \frac{1}{2}\int_0^t \left(\int_0^{t_1} [A(t_1), A(t_2)]\,\mathrm{d}t_2\right)\mathrm{d}t_1\right)\boldsymbol{x}_0. \tag{6.34}$$

To find Ω_2 we consider the commutator of matrices in $\mathfrak{se}(2)$ on the form (6.32):

$$\begin{aligned}
[A(t_1), A(t_2)] &= \begin{bmatrix} 0 & -\theta(t_1) & u(t_1) \\ \theta(t_1) & 0 & v(t_1) \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & -\theta(t_2) & u(t_2) \\ \theta(t_2) & 0 & v(t_2) \\ 0 & 0 & 0 \end{bmatrix} \\
&- \begin{bmatrix} 0 & -\theta(t_2) & u(t_2) \\ \theta(t_2) & 0 & v(t_2) \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & -\theta(t_1) & u(t_1) \\ \theta(t_1) & 0 & v(t_1) \\ 0 & 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} -\theta(t_1)\theta(t_2) & 0 & -\theta(t_1)v(t_2) \\ 0 & -\theta(t_1)\theta(t_2) & \theta(t_1)u(t_2) \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} -\theta(t_1)\theta(t_2) & 0 & -\theta(t_2)v(t_1) \\ 0 & -\theta(t_1)\theta(t_2) & \theta(t_2)u(t_1) \\ 0 & 0 & 0 \end{bmatrix} \\
&= \begin{bmatrix} 0 & 0 & -\theta(t_1)v(t_2) + \theta(t_2)v(t_1) \\ 0 & 0 & \theta(t_1)u(t_2) - \theta(t_2)u(t_1) \\ 0 & 0 & 0 \end{bmatrix}.
\end{aligned}$$

In the affine case where $\theta(t) = \theta_0 + a_\theta t$, and similarly for u and v, we get after evaluating the integrals:

$$\begin{bmatrix} a(t) \\ b(t) \\ x(t) \\ y(t) \end{bmatrix} \approx \exp\left(\begin{bmatrix} \theta_0 t + a_\theta t\frac{t^2}{2} \\ u_0 t + a_u\frac{t^2}{2} + \theta_0 a_v\frac{t^3}{12} - a_\theta v_0\frac{t^3}{12} \\ v_0 t + a_v\frac{t^2}{2} - \theta_0 a_u\frac{t^3}{12} + a_\theta u_0\frac{t^3}{12} \end{bmatrix}\right).$$

7. Probability Theory

7.1. Gaussian Distributions

Map tangent-space uncertainty through the group

7.2. The Banana Distribution

8. Equivariance

Write about equivariant systems

- Left-invariant, right-invariant, equivariant dynamical systems
- Grizzle: [10]
- Filtering: [9]

Part II.

Important Matrix Lie Groups for Robotics

9. Classical Lie Groups

Having gone through the foundational theory in the first part, we now turn our attention to specific groups and derive closed-form formulas for many of the concepts.

We now introduce some classical matrix Lie groups. Any subset of invertible square matrices that is closed under matrix multiplication is a Lie group. To find the structure of the corresponding Lie algebra it is useful to consider a trajectory

$$\boldsymbol{X}(t) = \operatorname{Exp}(t\boldsymbol{A}) \in \mathbb{M} \tag{9.1}$$

that satisfies $\boldsymbol{X}(0) = I$ and $\boldsymbol{X}'(0) = \boldsymbol{A}$.

The trajectory $\boldsymbol{X}(t)$ must satisfy a certain group constraint, which translates into a condition on $\boldsymbol{A}$. Since the Lie algebra of a group consists of all matrices $\boldsymbol{A}$ such that $\exp \boldsymbol{A} \in \mathbb{M}$ this yields the structure of the Lie algebra.

General Linear Group $\mathbb{GL}(n, F)$ The general linear group over a field F (here F is either the real numbers $\mathbb{R}$ or the complex numbers $\mathbb{C}$) is the most general Matrix lie group and contains all other groups as subgroups.

$$\mathbb{GL}(n, F) := \{\boldsymbol{A} \in F^{n\times n} \mid \det \boldsymbol{A} \neq 0\}. \tag{9.2}$$

The exponential map always produces invertible matrices, so the corresponding lie algebra is the space of all $n \times n$ matrices.

$$\mathfrak{gl}(n, F) = F^{n\times n}. \tag{9.3}$$

Any subset of $\mathbb{GL}(n, F)$ that is closed under matrix multiplication is also a matrix Lie group.

Translation Group $\mathbb{T}(n)$ The usual Euclidean vector space $\mathbb{R}^n$ can be embedded in matrices on the form $\begin{bmatrix} I_n & \boldsymbol{p} \\ 0_{1\times n} & 1 \end{bmatrix}$ for $\boldsymbol{p} \in \mathbb{R}^n$, so that matrix multiplication corresponds to addition in $\mathbb{R}^n$. Being a closed subset of $\mathbb{GL}(n, \mathbb{R})$ those matrices form a matrix Lie group.

To find the corresponding Lie algebra consider a trajectory

$$\boldsymbol{X}(t) = \begin{bmatrix} I_n & \boldsymbol{p}(t) \\ 0_{1\times n} & 1 \end{bmatrix} = \operatorname{Exp}(t\boldsymbol{A}) \in \mathbb{T}(n), \tag{9.4}$$

differentiating with respect to t then shows that

$$\begin{bmatrix} \boldsymbol{0}_{n\times n} & \boldsymbol{p}'(t) \\ \boldsymbol{0}_{1\times n} & 0 \end{bmatrix} \overset{!}{=} \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{X}(t)\bigg|_{t=0} = \boldsymbol{A}. \tag{9.5}$$

From here it follows that the Lie algebra $\mathfrak{e}(n)$ of $\mathbb{T}(n)$ consists of matrices where only the top n coefficients in the right-most column are non-zero.

The translation groups $\mathbb{T}(n)$ and corresponding Lie algebras $\mathfrak{t}(n)$:

$$\mathbb{T}(n) = \left\{ \begin{bmatrix} I_n & \boldsymbol{p} \\ 0 & 1 \end{bmatrix} \in \mathbb{GL}(n+1, \mathbb{R}) \mid \boldsymbol{p} \in \mathbb{R}^n \right\}, \tag{9.6a}$$

$$\mathfrak{t}(n) = \left\{ \begin{bmatrix} \boldsymbol{0}_{n\times n} & \boldsymbol{v} \\ \boldsymbol{0}_{1\times n} & 0 \end{bmatrix}, \boldsymbol{v} \in \mathbb{R}^n \right\}. \tag{9.6b}$$

Orthogonal Groups $\mathbb{O}(n)$ and $\mathbb{SO}(n)$ The orthogonal matrices $\mathbb{O}(n)$ are real matrices $\boldsymbol{X}$ s.t. the inverse is equal to the transpose, i.e. $\boldsymbol{X}\boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{X}\boldsymbol{X}^T = I_n$. The special orthogonal matrices in addition have a determinant equal to 1. In robotics $\mathbb{SO}(n)$ is particularly useful in the $n = 2, 3$ cases since those correspond to rotation matrices in two and three dimensions.

Take a one-parameter subgroup $\boldsymbol{X}(t) := \operatorname{Exp}(t\boldsymbol{A})$ and differentiate the group constraint $I_n = \boldsymbol{X}(t)^T\boldsymbol{X}(t)$:

$$0 \overset{!}{=} \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{X}(t)^T\boldsymbol{X}(t)\Big|_{t=0} = \boldsymbol{X}'(0)^T\boldsymbol{X}(0) + \boldsymbol{X}(0)^T\boldsymbol{X}'(0) = \boldsymbol{A}^T + \boldsymbol{A}. \tag{9.7}$$

It follows that the Lie algebra $\mathfrak{so}(n)$ corresponding to $\mathbb{SO}(n)$ consists of **skew-symmetric matrices**.

Orthogonal groups and corresponding Lie algebras:

$$\mathbb{O}(n) = \left\{ \boldsymbol{X} \in \mathbb{GL}(n, \mathbb{R}) \mid \boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{X}\boldsymbol{X}^T = I_n \right\}, \tag{9.8a}$$

$$\mathbb{SO}(n) = \left\{ \boldsymbol{X} \in \mathbb{GL}(n, \mathbb{R}) \mid \boldsymbol{X}^T\boldsymbol{X} = \boldsymbol{X}\boldsymbol{X}^T = I_n, \det \boldsymbol{X} = 1 \right\}, \tag{9.8b}$$

$$\mathfrak{o}(n) = \mathfrak{so}(n) = \left\{ \boldsymbol{A} \in \mathbb{R}^{n\times n} \;:\; \boldsymbol{A}^T + \boldsymbol{A} = 0 \right\}. \tag{9.8c}$$

Unitary Groups $\mathbb{U}(n)$ and $\mathbb{SU}(n)$ Unitary matrices $\boldsymbol{X}$ are characterized by the inverse being equal to the Hermitian transpose, i.e. $\boldsymbol{X}^*\boldsymbol{X} = \boldsymbol{X}\boldsymbol{X}^* = I$[1]. In the case of $\mathbb{SU}(n)$ the determinant is also required to be equal to 1.

For a one-parameter subgroup $\boldsymbol{X}(t) = \operatorname{Exp}(t\boldsymbol{A})$ constraint differentiation yields

$$0 = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{X}(t)^*\boldsymbol{X}(t)\Big|_{t=0} = \boldsymbol{X}'(0)^*\boldsymbol{X}(0) + \boldsymbol{X}(0)^*\boldsymbol{X}'(0) = \boldsymbol{A}^* + \boldsymbol{A}. \tag{9.9}$$

This shows that $\mathfrak{u}(n)$ consists of **skew-Hermitian matrices**. In addition, due to the Jacobi identity (4.2c) $\det \operatorname{Exp}(t\boldsymbol{A}) = 1$ implies that $\operatorname{Tr}\boldsymbol{A} = 0$ i.e. $\mathfrak{su}(n)$ consists of **skew-Hermitian matrices with vanishing trace**.

[1] The Hermitian transpose (also known as *conjugate transpose*) of $\boldsymbol{X}_{ij}$ is $\boldsymbol{X}^*_{ij} = \bar{\boldsymbol{X}}_{ji}$.

Unitary groups and corresponding Lie algebras:

$$\mathbb{U}(n) = \{\boldsymbol{X} \in \mathbb{GL}(n, \mathbb{C}) \mid \boldsymbol{X}^*\boldsymbol{X} = \boldsymbol{X}\boldsymbol{X}^* = I_n\} \tag{9.10a}$$

$$\mathbb{SU}(n) = \{\boldsymbol{X} \in \mathbb{GL}(n, \mathbb{C}) \mid \boldsymbol{X}^*\boldsymbol{X} = \boldsymbol{X}\boldsymbol{X}^* = I_n, \det \boldsymbol{X} = 1\}, \tag{9.10b}$$

$$\mathfrak{u}(n) = \{\boldsymbol{A} \in \mathbb{C}^{n\times n} \mid \boldsymbol{A}^* + \boldsymbol{A} = 0\}, \tag{9.10c}$$

$$\mathfrak{su}(n) = \{\boldsymbol{A} \in \mathbb{C}^{n\times n} \mid \boldsymbol{A}^* + \boldsymbol{A} = 0, \operatorname{Tr}\boldsymbol{A} = 0\}. \tag{9.10d}$$

Euclidean Groups $\mathbb{E}(n)$ and $\mathbb{SE}(n)$ A "semi-simple product" between $\mathbb{O}(n)$ (resp. $\mathbb{SO}(n)$) and $\mathbb{T}(n)$ obtained by replacing the identity matrix in $\mathbb{T}(n)$ by a member of $\mathbb{O}(n)$ (resp. $\mathbb{SO}(n)$). For example, $\mathbb{SE}(n)$ consists of matrices on the form $\begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0_{1\times n} & 1 \end{bmatrix}$ for $\boldsymbol{R} \in \mathbb{SO}(n)$. For a one-parameter subgroup $\boldsymbol{X}(t) = \begin{bmatrix} \boldsymbol{R}(t) & \boldsymbol{p}(t) \\ 0 & 1 \end{bmatrix} = \operatorname{Exp}(t\boldsymbol{A})$ differentiating shows that

$$\begin{bmatrix} \boldsymbol{R}'(t) & \boldsymbol{p}'(t) \\ 0 & 0 \end{bmatrix} = \boldsymbol{A}. \tag{9.11}$$

Since we already know that structure of $\boldsymbol{R}'(t)$ from (9.8c) the form of $\boldsymbol{A}$ can be determined.

Euclidean groups and corresponding Lie algebras:

$$\mathbb{E}(n) = \left\{\begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix} \in \mathbb{GL}(n+1, \mathbb{R}) \mid \boldsymbol{R} \in \mathbb{O}(n), \boldsymbol{p} \in \mathbb{R}^n\right\}, \tag{9.12a}$$

$$\mathbb{SE}(n) = \left\{\begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix} \in \mathbb{GL}(n+1, \mathbb{R}) \mid \boldsymbol{R} \in \mathbb{SO}(n), \boldsymbol{p} \in \mathbb{R}^n\right\}, \tag{9.12b}$$

$$\mathfrak{e}(n) = \mathfrak{se}(n) = \left\{\begin{bmatrix} \boldsymbol{B} & \boldsymbol{v} \\ 0 & 0 \end{bmatrix} \mid \boldsymbol{B}^T + \boldsymbol{B} = 0_{n\times n}, \boldsymbol{v} \in \mathbb{R}^n\right\}. \tag{9.12c}$$

Symplectic Groups $\mathbb{S}\mathbb{p}(n)$

9.1. Mathematical Preliminaries

Trigonometric Sums Starting with the familiar Taylor expansions of cosine and sine

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} = \cos x, \tag{9.13}$$

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = \sin x, \tag{9.14}$$

some higher-order formulas can be derived for $x \neq 0$ by dividing by a factor of $\boldsymbol{x}$ and subtracting the first summation terms.

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n} = \frac{1}{x} \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = \frac{\sin x}{x}, \tag{9.15}$$

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+2)!} x^{2n} = -\frac{1}{x^2} \sum_{n=1}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} = \frac{1-\cos x}{x^2}, \tag{9.16}$$

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+3)!} x^{2n} = -\frac{1}{x^3} \sum_{n=1}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = \frac{x-\sin x}{x^3}, \tag{9.17}$$

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+4)!} x^{2n} = \frac{1}{x^4} \sum_{n=2}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} = \frac{\cos x - 1 + \frac{x^2}{2}}{x^4}, \tag{9.18}$$

$$\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+5)!} x^{2n} = \frac{1}{x^5} \sum_{n=2}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = \frac{\sin x - x + \frac{x^3}{6}}{x^5}. \tag{9.19}$$

Bernoulli Numbers Finally a sum involving the Bernoulli numbers will be useful for some groups of interest.

Proposition 9.1.

$$\sum_{n=1}^{\infty} \frac{B_{2n}(-1)^n x^{2n}}{(2n)!} = \frac{x}{2} \cot\left(\frac{x}{2}\right). \tag{9.20}$$

Proof. By setting $x = iy$ and observing that $B_n = 0$ for odd $n > 1$ we get

$$\begin{aligned}\sum_{n=0}^{\infty} \frac{B_{2n}(-1)^n x^{2n}}{(2n)!} &= \sum_{n=0}^{\infty} \frac{B_{2n}(-1)^n y^{2n}(-1)^n}{(2n)!} = \sum_{n=0}^{\infty} \frac{B_n y^n}{n!} - B_1 y \overset{(5.32)}{=} \frac{y}{e^y - 1} + \frac{y}{2} = \frac{y}{2}\frac{e^y+1}{e^y-1} \\ &= \frac{ix}{2}\frac{1+e^{-iy}}{1-e^{-iy}} - 1 = \frac{ix}{2}\frac{e^{iy/2}+e^{-ix/2}}{e^{ix/2}-e^{-ix/2}} = \frac{ix}{2}\frac{\cos(x/2)}{i\sin(x/2)} = \frac{x}{2}\cot\left(\frac{x}{2}\right).\end{aligned} \tag{9.21}$$

□

Exponential of Structured Matrices We derive an identity that will be useful to construct the exponential maps for semi-simple groups.

Lemma 9.1. *Consider two matrices $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{n\times n}$ such that $\boldsymbol{B}^2 = \boldsymbol{B}\boldsymbol{A} = 0$. Then we have that*

$$\operatorname{Exp}(\boldsymbol{A}+\boldsymbol{B}) = \operatorname{Exp}(\boldsymbol{A}) + \sum_{k=0}^{\infty} \frac{\boldsymbol{A}^k}{(k+1)!}\boldsymbol{B}. \tag{9.22}$$

Proof. When we expand $(\boldsymbol{A}+\boldsymbol{B})^k$ all terms that contain a $\boldsymbol{B}$ before an $\boldsymbol{A}$, or multiple $\boldsymbol{B}$ in a row, vanish. As a result,

$$\operatorname{Exp}(\boldsymbol{A}+\boldsymbol{B}) = I_n + \sum_{k=1}^{\infty} \frac{(\boldsymbol{A}+\boldsymbol{B})^k}{k!} = I_n + \sum_{k=1}^{\infty} \frac{\boldsymbol{A}^k + \boldsymbol{A}^{k-1}\boldsymbol{B}}{k!} = \operatorname{Exp}(\boldsymbol{A}) + \sum_{k=1}^{\infty} \frac{\boldsymbol{A}^{k-1}}{k!}\boldsymbol{B}. \tag{9.23}$$

□

10. $\mathbb{SO}(2)$: The 2D Rotation Group

Let $\boldsymbol{R}$ denote an element of $\mathbb{SO}(2)$; we know that $\boldsymbol{R}\boldsymbol{R}^T = I_2$.

Action on $\mathbb{R}^2$: In robotics applications it is convenient to define a rotational action on vectors in $\mathbb{R}^2$. For $\boldsymbol{R} \in \mathbb{SO}(2)$ and $\boldsymbol{u} \in \mathbb{R}^2$ the action is matrix multiplication

$$\langle \boldsymbol{R}, \boldsymbol{u} \rangle_{\mathbb{SO}(2)} = \boldsymbol{R} \cdot \boldsymbol{u}. \tag{10.1}$$

Lie Algebra The 2×2 skew-symmetric matrices have only one degree of freedom, let this single parameter of $\check{\mathfrak{so}}(2)$ be denoted ω_z so that

$$\mathfrak{so}(2) = \left\{ \begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix} \mid \omega_z \in \mathbb{R} \right\}, \tag{10.2}$$

and the Lie algebra hat and vee maps become

$$\begin{tikzcd} \mathbb{R}^1 \ni \left[\omega_z\right] \arrow[rr, bend left, "\wedge"] & & \begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix} \in \mathfrak{so}(2) \arrow[ll, bend left, "\vee"] \end{tikzcd}$$

10.1. Formulas

Adjoint From the definition,

$$\mathbf{Ad}_{\boldsymbol{R}} \left[\omega_z\right] = \left(\boldsymbol{R}\omega_z^\wedge \boldsymbol{R}^{-1}\right)^\vee = \left(\boldsymbol{R} \begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix} \boldsymbol{R}^T\right)^\vee = \begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix}^\vee = \left[\omega_z\right], \tag{10.3}$$

so it follows that $\mathbf{Ad}_{\boldsymbol{R}} = \left[1\right]$.

Exponential and Logarithm Take an element $\left[\omega_z\right]$; The exponential is calculated by noting that $(\omega_z^\wedge)^{2k} = (-1)^k \omega_z^{2k} I_2$:

$$\begin{aligned} \operatorname{Exp} \omega_z^\wedge &= \sum_{k=0}^{\infty} \frac{(\omega_z^\wedge)^k}{k!} = \sum_{k=0}^{\infty} \frac{\left(\omega_z^\wedge\right)^{2k}}{(2k)!} + \frac{\left(\omega_z^\wedge\right)^{2k+1}}{(2k+1)!} = \sum_{k=0}^{\infty} \frac{(-1)^k \omega_z^{2k}}{(2k)!} I_2 + \frac{(-1)^k \omega_z^{2k}}{(2k+1)!} \omega_z^\wedge \\ &\overset{(9.13),(9.15)}{=} \cos\omega_z I_2 + \frac{\sin\omega_z}{\omega_z} \omega_z^\wedge = \begin{bmatrix} \cos\omega_z & -\sin\omega_z \\ \sin\omega_z & \cos\omega_z \end{bmatrix}. \end{aligned} \tag{10.4}$$

Derivatives of the Exponential Consider algebra elements $\omega_z, \bar{\omega}_z \in \mathfrak{so}(2)$. The bracket on $\mathfrak{so}(2)$ is zero since

$$[\omega_z, \bar{\omega}_z] = \left(\begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix} \begin{bmatrix} 0 & -\bar{\omega}_z \\ \bar{\omega}_z & 0 \end{bmatrix} - \begin{bmatrix} 0 & -\bar{\omega}_z \\ \bar{\omega}_z & 0 \end{bmatrix} \begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix} \right)^{\vee} = 0. \tag{10.5}$$

It follows that all terms in (5.31) and (5.33) vanish except for $n = 0$, so the derivatives of the exponential are equal to $I_1 = \begin{bmatrix} 1 \end{bmatrix}$.

$\mathbb{SO}(2)$ formula sheet

Consists of 2×2 rotation matrices $\boldsymbol{R}$ that act on $\mathbb{R}^2$ via $\boldsymbol{v} \mapsto \boldsymbol{R}\boldsymbol{v}$.

Algebra Parameterization

$$\{\omega_w \mid \omega_w \in [-\pi, \pi]\}, \qquad \omega_w^{\wedge} = \begin{bmatrix} 0 & -\omega_w \\ \omega_w & 0 \end{bmatrix} \in \mathfrak{so}(2). \tag{10.6}$$

Adjoint

$$\mathbf{Ad}_{\boldsymbol{R}} = \begin{bmatrix} 1 \end{bmatrix}. \tag{10.7}$$

Exponential and Logarithm

$$\exp(\omega_w) = \begin{bmatrix} \cos \omega_w & -\sin \omega_w \\ \sin \omega_w & \cos \omega_w \end{bmatrix}, \tag{10.8a}$$

$$\log(\boldsymbol{R}) = \arccos \left(\frac{\boldsymbol{R}^T + \boldsymbol{R}}{2} \right). \tag{10.8b}$$

Bracket and Lowercase adjoint

$$[\omega_z, \omega_z'] = 0 \tag{10.9a}$$

$$\mathrm{ad}_{\omega_z} = 0 \tag{10.9b}$$

Derivatives of the Exponential

$$\mathrm{d}^r \exp_{\omega_z} = \mathrm{d}^l \exp_{\omega_z} = \left(\mathrm{d}^r \exp_{\omega_z} \right)^{-1} = \left(\mathrm{d}^l \exp_{\omega_z} \right)^{-1} = \begin{bmatrix} 1 \end{bmatrix}. \tag{10.10}$$

10.2. Parameterization via Isomorhphism with $\mathbb{U}(1)$

We use the isomorphism $\mathbb{SO}(2) \cong \mathbb{U}(1)$, where $\mathbb{U}(1)$ is the unitary group consisting of complex elements $c = q_w + iq_z$ with unit length, to parameterize elements of $\mathbb{SO}(2)$.

$$\mathbb{U}(1) = \left\{ (q_w, q_z) \in \mathbb{R}^2 \mid q_w^2 + q_z^2 = 1 \right\}. \tag{10.11}$$

The hat and vee maps between the parameterization and matrix forms are thus as follows:

$$\begin{tikzcd} \mathbb{U}(1) \ni (q_w, q_z) \arrow[rr, bend left, "\wedge"] & & \boldsymbol{R} = \begin{bmatrix} q_w & -q_z \\ q_z & q_w \end{bmatrix} \in \mathbb{SO}(2) \arrow[ll, bend left, "\vee"] \end{tikzcd}$$

From the isomorphism to complex numbers it follows that the identity element is $(1, 0)$ and the inverse is $(q_w, q_z)^{-1} = (q_w, -q_z)$. A formula for group composition can either be obtained by expanding the complex product $(q_w + iq_z)(q'_w + iq_z)$ and identifying the coefficients, or by going via the matrix form:

$$(q_w, q_z) \circ (q'_w, q'_z) = ((q_w, q_z)^\wedge \cdot (q'_w, q'_z)^\wedge)^\vee = \left(\begin{bmatrix} q_w & -q_z \\ q_z & q_w \end{bmatrix} \cdot \begin{bmatrix} q'_w & -q'_z \\ q'_z & q'_w \end{bmatrix} \right)^\vee = (q_w q'_w - q_z q'_z, q_z q'_w + q_w q'_z).$$

It also follows from isomorphism that

$$\exp(\omega_z) = (\cos \omega_z, \sin \omega_z),$$

consequently $\log(q_w, q_z) = \operatorname{arctan2}(q_z, q_w)$, where $\operatorname{arctan2}(y, x)$ is the angle between the ray passing through $(1, 0)$ and the ray passing through (x, y).

$\mathbb{U}(1)$ as a parameterization of $\mathbb{SO}(2)$

Group Parameterization

$$\mathbb{U}(1) = \{c \in \mathbb{C} \mid |c| = 1\} \tag{10.12}$$

Group Operations

- Identity element: $1 \in \mathbb{C}$,
- Inverse: $c^{-1} = \bar{c}$, where bar denotes complex conjugate,
- Composition: $c \circ c' = \mathrm{Re}(c)\mathrm{Re}(c') - \mathrm{Im}(c)\mathrm{Im}(c') + i\,(\mathrm{Re}(c)\mathrm{Im}(c') + \mathrm{Im}(c)\mathrm{Re}(c'))$.

$\mathbb{U}(1)$ is isomorphic to $\mathbb{SO}(2)$ via $\wedge : \mathbb{U}(1) \to \mathbb{SO}(2)$

$$c^\wedge = \begin{bmatrix} \mathrm{Re}(c) & -\mathrm{Im}(c) \\ \mathrm{Im}(c) & \mathrm{Re}(c) \end{bmatrix} \tag{10.13}$$

and therefore inherits Lie algebra properties from $\mathbb{SO}(2)$.

Rotation Action on $\boldsymbol{u} \in \mathbb{R}^2$

$$\langle c, \boldsymbol{u} \rangle = \begin{bmatrix} \mathrm{Re}(c) & -\mathrm{Im}(c) \\ \mathrm{Im}(c) & \mathrm{Re}(c) \end{bmatrix} \boldsymbol{u}. \tag{10.14}$$

Exponential and Logarithm

$$\exp(\omega_w) = \cos\omega_w + i\sin\omega_w, \tag{10.15a}$$

$$\log(c) = \operatorname{arctan2}\left(\operatorname{Im}(c), \operatorname{Re}(c)\right). \tag{10.15b}$$

11. $\mathbb{SO}(3)$: The 3D Rotation Group

Needs to be cleaned up

As opposed to the 2D case where $\mathbb{SO}(2)$ as defined above is the canonical way to represent rotations, the situation is more complicated in three dimensions. While $\mathbb{SO}(2)$ generalizes to $\mathbb{SO}(3)$ that consists of orthogonal 3×3 matrices with determinant 1, it is no longer as easy to construct a lower-dimensional representation. The usual choice is the unit quaternions, which are isomorphic to the matrix Lie group $\mathbb{SU}(2)$. We begin by defining the matrix group $\mathbb{SO}(3)$.

11.1. Grab bag

We proceed with $\mathbb{SO}(3)$: skew-symmetric matrices of size 3×3 are parameterized by three parameters $\boldsymbol{\omega} := (\omega_x, \omega_y, \omega_z)$ so that $\mathfrak{so}(3)$ consists of elements on the form

$$\hat{\boldsymbol{\omega}} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}. \tag{11.1}$$

Such a matrix has several interesting properties. First of all, for $\boldsymbol{u} \in \mathbb{R}^3$ left matrix multiplication of $\hat{\boldsymbol{\omega}}$ is equivalent to taking the vector cross product: $\hat{\boldsymbol{\omega}}\boldsymbol{u} = \boldsymbol{\omega} \times \boldsymbol{u}$. As a result many properties of the cross product are inherited by the embedding $\mathbb{R}^3 \overset{\wedge}{\mapsto} \mathbb{R}^{3\times 3}$.

Properties of $\wedge$ on $\mathfrak{so}(3)$

For $\boldsymbol{a}, \boldsymbol{b}, \boldsymbol{c} \in \mathbb{R}^3$:

$$\hat{\boldsymbol{a}}\hat{\boldsymbol{b}}\hat{\boldsymbol{a}} = -(\boldsymbol{a} \cdot \boldsymbol{b})\hat{\boldsymbol{a}}, \tag{11.2a}$$

$$\hat{\boldsymbol{a}}\boldsymbol{b} = -\hat{\boldsymbol{b}}\boldsymbol{a}, \tag{11.2b}$$

$$\boldsymbol{a} \cdot (\hat{\boldsymbol{b}}\boldsymbol{c}) = \boldsymbol{b} \cdot (\hat{\boldsymbol{c}}\boldsymbol{a}), \tag{11.2c}$$

$$A\hat{\boldsymbol{b}} = \operatorname{Tr}(A)\hat{\boldsymbol{b}} - (A\boldsymbol{b})^\wedge - \hat{\boldsymbol{b}}A, \quad A \text{ symmetric } 3 \times 3 \text{ matrix}, \tag{11.2d}$$

$$\boldsymbol{a} \cdot \boldsymbol{b} = \frac{1}{2}\left\langle \hat{\boldsymbol{a}}, \hat{\boldsymbol{b}} \right\rangle_F = -\frac{1}{2}\operatorname{Tr}\left(\hat{\boldsymbol{a}}, \hat{\boldsymbol{b}}\right). \tag{11.2e}$$

Proof of (11.2a). Consider $\hat{\boldsymbol{a}}\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}\boldsymbol{c} = \boldsymbol{a} \times (\boldsymbol{b} \times (\boldsymbol{a} \times \boldsymbol{c}))$. Expanding with the vector triple product gives

$$\hat{\boldsymbol{a}}\hat{\boldsymbol{b}}\hat{\boldsymbol{a}}\boldsymbol{c} = \boldsymbol{a} \times ((\boldsymbol{b} \cdot \boldsymbol{c})\boldsymbol{a} - (\boldsymbol{a} \cdot \boldsymbol{b})\boldsymbol{c}) = -(\boldsymbol{a} \cdot \boldsymbol{b})\boldsymbol{a} \times \boldsymbol{c} = -(\boldsymbol{a} \cdot \boldsymbol{b})\hat{\boldsymbol{a}}\boldsymbol{c}. \tag{11.3}$$

□

We can use (11.2a) to obtain the exponential map on $\mathfrak{so}(3)$:

$$\begin{aligned}
\operatorname{Exp}\ \hat{\boldsymbol{\omega}} = \sum_{k\geq 0} \frac{(\hat{\boldsymbol{\omega}})^k}{k!} &= I + \hat{\boldsymbol{\omega}} + \frac{\hat{\boldsymbol{\omega}}^2}{2!} - \|\boldsymbol{\omega}\|^2 \left(\frac{\hat{\boldsymbol{\omega}}}{3!} + \frac{\hat{\boldsymbol{\omega}}^2}{4!}\right) + \|\boldsymbol{\omega}\|^4 \left(\frac{\hat{\boldsymbol{\omega}}}{5!} + \frac{\hat{\boldsymbol{\omega}}^2}{6!}\right) + \ldots \\
&= I + \left(1 - \frac{\|\boldsymbol{\omega}\|^2}{3!} + \frac{\|\boldsymbol{\omega}\|^4}{5!} - \ldots\right)\hat{\boldsymbol{\omega}} + \left(\frac{1}{2!} - \frac{\|\boldsymbol{\omega}\|^2}{4!} + \frac{\|\boldsymbol{\omega}\|^4}{6!} - \ldots\right)\hat{\boldsymbol{\omega}}^2 \\
&= I + \frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}} + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}^2.
\end{aligned} \tag{11.4}$$

To obtain the logarithm the expression

$$\boldsymbol{R} = I_3 + \frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}} + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}^2 \tag{11.5}$$

should be inverted. First note that due to $\hat{\boldsymbol{\omega}}$ being skew-symmetric:

$$\boldsymbol{R} - \boldsymbol{R}^T = \frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}} + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}^2 - \frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}}^T - \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}(\hat{\boldsymbol{\omega}}^T)^2 = 2\frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}}. \tag{11.6}$$

Secondly,

$$\operatorname{Tr}(\boldsymbol{R}) = 3 + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\operatorname{Tr}(\hat{\boldsymbol{\omega}}^2) = 3 - 2\frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\|\boldsymbol{\omega}\|^2 = 1 + 2\cos\|\boldsymbol{\omega}\|, \tag{11.7}$$

which makes it possible to write down an expression for the logarithm.

The exponential and logarithm on $\mathbb{SO}(3)$ are

$$\exp_{\mathbb{SO}(3)} \boldsymbol{\omega} = \operatorname{Exp}_{\mathbb{SO}(3)} \hat{\boldsymbol{\omega}} = I + \frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}} + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}^2, \tag{11.8a}$$

$$\operatorname{Log}_{\mathbb{SO}(3)} \boldsymbol{R} = \frac{\alpha}{\sin\alpha}\frac{\boldsymbol{R}-\boldsymbol{R}^T}{2}, \quad \alpha = \arccos\left(\frac{\operatorname{Tr}(\boldsymbol{R})-1}{2}\right). \tag{11.8b}$$

The lower-dimensional representation of $\mathbb{SO}(3)$ is $\mathbb{S}^3$, but as shown previously the $\wedge$ and $\vee$ mappings are not straightforward. In the next section we obtain the exponential and logarithm on $\mathbb{S}^3$ through its relation to $\mathbb{SU}(2)$.

$\mathbb{SO}(3)$: three-dimensional rotations

$\mathbb{SO}(3)$ is a matrix Lie group that consists of 3×3 orthogonal matrices with determinant equal to one:

$$\mathbb{SO}(3) = \left\{\boldsymbol{R} \in \mathbb{GL}(3) \mid \boldsymbol{R}^T\boldsymbol{R} = I, \det(\boldsymbol{R}) = 1\right\}. \tag{11.9}$$

These matrices are usually referred to as **rotation matrices**.

There is no trivial low-dimensional parameterizations of this set, however, it is isometric to another group $\mathbb{SU}(2)$ that is in turn isometric to the unit quaternions $\mathbb{S}^3$ which can be used as a lower-dimensional representation of $\mathbb{SO}(3)$.

$$\mathbb{S}^3 = \left\{(q_w, q_x, q_y, q_z) : q_w^2 + q_x^2 + q_y^2 + q_z^2 = 1\right\}. \tag{11.10}$$

However, the mapping is not 1-to-1, since both $\boldsymbol{q} \coloneqq (q_w, q_x, q_y, q_z)$ and $-\boldsymbol{q}$ correspond to the same rotation matrix.

Action on $\mathbb{R}^3$ The action of $\boldsymbol{R} \in \mathbb{SO}(3)$ on $\boldsymbol{u} \in \mathbb{R}^3$ is rotation:

$$\langle \boldsymbol{R}, \boldsymbol{u} \rangle = \boldsymbol{R} \cdot \boldsymbol{u}. \tag{11.11}$$

$\mathbb{SU}(2)$ and its relation to the quaternion group $\mathbb{S}^3$

We can associate a quaternion $\boldsymbol{q} = q_w + q_x \boldsymbol{i} + q_y \boldsymbol{j} + q_z \boldsymbol{k}$ with the unitary matrix

$$\mathbb{SU}(2) = \left\{ \begin{bmatrix} q_w + iq_z & -q_x - iq_y \\ q_x - iq_y & q_w - iq_z \end{bmatrix} \mid q_w^2 + q_x^2 + q_y^2 + q_z^2 = 1 \right\} \tag{11.12}$$

for which it holds that $A_{\boldsymbol{q}_1 * \boldsymbol{q}_2} = A_{\boldsymbol{q}_1} A_{\boldsymbol{q}_2}$. Thus the unit quaternions $\mathbb{S}^3$ are isomorphic to $\mathbb{SU}(2)$ and can therefore be viewed as a matrix Lie group.

By muliplying two elements in $\mathbb{SU}$ we retrieve quaternion multiplication:

$$\begin{bmatrix} q_w + iq_z & -q_x - iq_y \\ q_x - iq_y & q_w - iq_z \end{bmatrix} \begin{bmatrix} q'_w + iq'_z & -q'_x - iq'_y \\ q'_x - iq'_y & q'_w - iq'_z \end{bmatrix} = \begin{bmatrix} q''_w + iq''_z & -q''_x - iq''_y \\ q''_x - iq''_y & q''_w - iq''_z \end{bmatrix} \tag{11.13}$$

where

$$\begin{aligned} q''_w &= q_w q'_w - q_x q'_x - q_y q'_y - q_z q'_z, \\ q''_x &= q_x q'_w + q_w q'_x + q_y q'_z - q_z q'_y, \\ q''_y &= q_y q'_w + q_w q'_y + q_z q'_x - q_x q'_z, \\ q''_z &= q_z q'_w + q_w q'_z + q_x q'_y - q_y q'_x, \end{aligned} \tag{11.14}$$

which is exactly what is obtained by carrying out the usual quaternion multiplication

$$(q_w + q_x \boldsymbol{i} + q_y \boldsymbol{j} + q_z \boldsymbol{k}) * (q'_w + q'_x \boldsymbol{i} + q'_y \boldsymbol{j} + q'_z \boldsymbol{k}) \tag{11.15}$$

with the quaternion rules $\boldsymbol{i}\boldsymbol{j} = \boldsymbol{k}$, $\boldsymbol{j}\boldsymbol{k} = \boldsymbol{i}, \boldsymbol{k}\boldsymbol{i} = \boldsymbol{j}$ and $\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = -1$.

Action on $\mathbb{R}^3$ A quaternion $\boldsymbol{q} = q_w + q_x \boldsymbol{i} + q_y \boldsymbol{j} + q_z \boldsymbol{k}$ acts on $\boldsymbol{u} := \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \in \mathbb{R}^3$ as quaternion rotation $\boldsymbol{q} * \boldsymbol{u} * \bar{\boldsymbol{q}}$ where $\boldsymbol{u}$ is associated with the quaternion $u_x \boldsymbol{i} + u_y \boldsymbol{j} + u_z \boldsymbol{k}$.

In terms of matrix multiplication operation can be written

$$\begin{aligned} &\begin{bmatrix} q_w + iq_z & -q_x - iq_y \\ q_x - iq_y & q_w - iq_z \end{bmatrix} \begin{bmatrix} iu_z & -u_x - iu_y \\ u_x - iu_y & iu_z \end{bmatrix} \begin{bmatrix} q_w - iq_z & q_x + iq_y \\ -q_x + iq_y & q_w + iq_z \end{bmatrix} \\ &\qquad = \begin{bmatrix} iu'_z & -u'_x - iu'_y \\ u'_x - iu'_y & iu'_z \end{bmatrix} \end{aligned} \tag{11.16}$$

for

$$\begin{aligned} u'_x &= (1 - 2(q_y^2 + q_z^2))u_x + 2(q_x q_y - q_w q_z)u_y + 2(q_x q_z + q_w q_y)u_z \\ u'_y &= 2(q_x q_y + q_w q_z)u_x + (1 - 2(q_x^2 + q_z^2))u_y + 2(q_y q_z - q_w q_x)u_z \\ u'_z &= 2(q_x q_z - q_w q_y)u_x + 2(q_w q_x + q_y q_z)u_y + (1 - 2(q_x^2 + q_y^2))u_z \end{aligned} \tag{11.17}$$

Since this is a linear mapping in u_x, u_y, u_z we can identify $\boldsymbol{q}$ with a matrix $\boldsymbol{R}(\boldsymbol{q})$ with coefficients

$$\boldsymbol{R}(\boldsymbol{q}) = \begin{bmatrix} (1-2(q_y^2+q_z^2)) & 2(q_xq_y - q_wq_z) & 2(q_xq_z + q_wq_y) \\ 2(q_xq_y + q_wq_z) & (1-2(q_x^2+q_z^2)) & 2(q_yq_z - q_wq_x) \\ 2(q_xq_z - q_wq_y) & 2(q_wq_x + q_yq_z) & (1-2(q_x^2+q_y^2)) \end{bmatrix}. \tag{11.18}$$

Thus we can utilize the quaternion group $\mathbb{S}^3$ as the lower-dimensional representation of $\mathbb{SO}(3)$.

Useful quaternion identities

Axis-angle to quaternion The quaternion $\boldsymbol{q}$ representing the rotation about a unit axis $\boldsymbol{\beta} = (\beta_x, \beta_y, \beta_z)$ for an angle α is

$$\boldsymbol{q} = \cos\left(\frac{\alpha}{2}\right) + \sin\left(\frac{\alpha}{2}\right)\left(\beta_x\boldsymbol{i} + \beta_y\boldsymbol{j} + \beta_z\boldsymbol{k}\right). \tag{11.19}$$

Two vectors to quaternion A quaternion $\boldsymbol{q}$ such that $\boldsymbol{qu} = \boldsymbol{v}$ for unit vectors $\boldsymbol{u}, \boldsymbol{v}$.

$$\boldsymbol{q} = \sqrt{\frac{1+s}{2}} + \sqrt{\frac{1-s}{2}}\left(\beta_x\boldsymbol{i} + \beta_y\boldsymbol{j} + \beta_z\boldsymbol{k}\right), \quad s = \boldsymbol{u}\cdot\boldsymbol{v}, \;\; \boldsymbol{\beta} = \boldsymbol{u}\times\boldsymbol{v}. \tag{11.20}$$

Hopf fibration The quaternions can be parameterized as the product of a rotation $\boldsymbol{q}_\theta$ around the z axis and a quaternion that rotates $\boldsymbol{e}_z$ to $\boldsymbol{\beta} := \left[\beta_x, \beta_y, \beta_z\right] \in \mathbb{S}^2$ as

$$\boldsymbol{q} = \boldsymbol{q}_{\boldsymbol{\beta}} * \boldsymbol{q}_\theta, \quad \boldsymbol{q}_{\boldsymbol{\beta}} = \frac{1}{\sqrt{2(1+\beta_z)}}\left(1 + \beta_z - \boldsymbol{i}\beta_x + \boldsymbol{j}\beta_y\right), \;\; \boldsymbol{q}_\theta = \cos\left(\frac{\theta}{2}\right) + \boldsymbol{k}\sin\left(\frac{\theta}{2}\right). \tag{11.21}$$

The special case when $\beta_z = -1$ is a singularity and must be handled separately, for example by setting $\boldsymbol{q}_{\left[0,\,0,\,-1\right]} = \boldsymbol{i}$. The Hopf parameterization is a manifestation of the fact that $\mathbb{S}^3$ locally is a product of the spaces $\mathbb{S}^2$ and $\mathbb{S}^1$.

Proof of (11.20). From properties of the dot and cross products the sought-after rotation is about the axis $\boldsymbol{\beta} = \boldsymbol{u}\times\boldsymbol{v}$ for the angle α such that $s := \boldsymbol{u}\cdot\boldsymbol{v} = \cos(\alpha)$. The half-angle formulas then give that $\cos(\alpha/2) = \sqrt{(1+s)/2}$, and similarly for the sine part in (11.19). □

11.2. Formulas

Adjoint

Exponential and Logarithm The Lie algebra $\mathfrak{su}(2)$ is parameterized by three elements $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$ that correspond to the skew-Hermitian matrix $\hat{\boldsymbol{\omega}} =:= \frac{1}{2}\begin{bmatrix} i\omega_z & -\omega_x - i\omega_y \\ \omega_x - i\omega_y & -i\omega_z \end{bmatrix}$, where the factor $1/2$ is added for reasons that will become clear below. A simple calculation reveals that

$\hat{\boldsymbol{\omega}}^2 = -\frac{\|\boldsymbol{\omega}\|}{4} I_2$ which can be used to evaluate the exponential.

$$
\begin{aligned}
\operatorname{Exp} \hat{\boldsymbol{\omega}} &= \sum_{k \geq 0} \frac{\hat{\boldsymbol{\omega}}^k}{k!} = \sum_{k \geq 0} \frac{1}{k!} \left(-\frac{\|\boldsymbol{\omega}\|}{2} \right)^{2 * \lfloor \frac{k}{2} \rfloor} \hat{\boldsymbol{\omega}}^{(k \mod 2)} \\
&= \left(1 - \frac{(\|\boldsymbol{\omega}\|/2)^2}{2!} + \frac{(\|\boldsymbol{\omega}\|/2)^4}{4!} - \dots \right) I + \left(1 - \frac{(\|\boldsymbol{\omega}\|/2)^2}{3!} + \frac{(\|\boldsymbol{\omega}\|/2)^4}{5!} + \dots \right) \hat{\boldsymbol{\omega}} \\
&= \cos(\|\boldsymbol{\omega}\|/2) I + 2 \frac{\sin \|\boldsymbol{\omega}\|/2}{\|\boldsymbol{\omega}\|} \hat{\boldsymbol{\omega}} = \begin{bmatrix} \cos(\|\boldsymbol{\omega}\|/2) + i \frac{\omega_z \sin(\|\boldsymbol{\omega}\|/2)}{\|\boldsymbol{\omega}\|} & (-\omega_x - i\omega_y) \frac{\sin \|\boldsymbol{\omega}\|/2}{\|\boldsymbol{\omega}\|} \\ (\omega_x - i\omega_y) \frac{\sin \|\boldsymbol{\omega}\|/2}{\|\boldsymbol{\omega}\|} & \cos(\|\boldsymbol{\omega}\|/2) - i \frac{\omega_z \sin(\|\boldsymbol{\omega}\|/2)}{\|\boldsymbol{\omega}\|} \end{bmatrix}.
\end{aligned}
$$

Since the mappings $\wedge : \mathbb{S}^3 \to \mathbb{SU}(2)$ and $\vee : \mathbb{SU}(2) \to \mathbb{S}^3$ are straightforward, we can also write down the exponentail and logarithm on $\mathbb{S}^3$. Since $\mathbb{S}^3$ is not itself a matrix Lie group, the uppercase exponential and logarithm do not have a meaning.

The exponential and logarithm on $\mathbb{S}^3$ are

$$
\exp(\omega_x, \omega_y, \omega_z) = \left(\cos(\|\boldsymbol{\omega}\|/2), \frac{\omega_x}{\|\boldsymbol{\omega}\|} \sin(\|\boldsymbol{\omega}\|/2), \frac{\omega_y}{\|\boldsymbol{\omega}\|} \sin(\|\boldsymbol{\omega}\|/2), \frac{\omega_z}{\|\boldsymbol{\omega}\|} \sin(\|\boldsymbol{\omega}\|/2) \right), \tag{11.22a}
$$

$$
\log(q_w, q_x, q_y, q_z) = \left(2 \frac{\operatorname{arctan2}\left(\sqrt{q_x^2 + q_y^2 + q_z^2}, q_w \right)}{\sqrt{q_x^2 + q_y^2 + q_z^2}} \right) \times (q_x, q_y, q_z). \tag{11.22b}
$$

From (11.22) the reason to divide the expression for $\hat{\boldsymbol{\omega}}$ by a factor 2 becomes apparent—$\|\boldsymbol{\omega}\|$ represents the rotation angle in radians. We provide a quick proof for the logarithm expression.

Proof of (11.22b). Let $(q_w, q_x, q_y, q_z) = \exp(\omega_x, \omega_y, \omega_z)$. From (11.22a) we have that

$$
\sqrt{q_x^2 + q_y^2 + q_z^2} = \sqrt{\left(\frac{\omega_x}{\|\boldsymbol{\omega}\|} \sin(\|\boldsymbol{\omega}\|/2) \right)^2 + \left(\frac{\omega_x}{\|\boldsymbol{\omega}\|} \sin(\|\boldsymbol{\omega}\|/2) \right)^2 + \left(\frac{\omega_x}{\|\boldsymbol{\omega}\|} \sin(\|\boldsymbol{\omega}\|/2) \right)^2} = \sin(\|\boldsymbol{\omega}\|/2). \tag{11.23}
$$

It also follows from the same equation that

$$
\begin{aligned}
\boldsymbol{\omega} &= \frac{\|\boldsymbol{\omega}\|}{\sin(\|\boldsymbol{\omega}\|/2)} (q_x, q_y, q_z) = \frac{\operatorname{arctan2}(\sin \|\boldsymbol{\omega}\|, \cos \|\boldsymbol{\omega}\|)}{\sqrt{q_x^2 + q_y^2 + q_z^2}} \\
&= \frac{2 \operatorname{arctan2}(\sin(\|\boldsymbol{\omega}\|/2), \cos(\|\boldsymbol{\omega}\|/2))}{\sqrt{q_x^2 + q_y^2 + q_z^2}} = \frac{2 \operatorname{arctan2}\left(\sqrt{q_x^2 + q_y^2 + q_z^2}, q_w \right)}{\sqrt{q_x^2 + q_y^2 + q_z^2}}.
\end{aligned} \tag{11.24}
$$

□

Derivatives of the Exponential We know that $\mathrm{ad}_{\boldsymbol{\omega}} = \hat{\boldsymbol{\omega}}$ and from (11.2a) that $\hat{\boldsymbol{\omega}}^3 = -\|\boldsymbol{\omega}\|^2 \hat{\boldsymbol{\omega}}$. Thus $\mathrm{ad}^3_{\boldsymbol{\omega}} = -\|\boldsymbol{\omega}\|^2 \,\mathrm{ad}_{\boldsymbol{\omega}}$ and we get,

$$\begin{aligned}
\sum_{n=0}^{\infty} \frac{B_n(-1)^n}{n!} \mathrm{ad}^n_{\boldsymbol{\omega}} &= \sum_{n=0}^{\infty} \frac{B_n(-1)^n}{n!} \hat{\boldsymbol{\omega}}^n = I_3 + \frac{\mathrm{ad}_{\boldsymbol{\omega}}}{2} + \sum_{n\geq 2} \frac{B_n(-1)^n}{n!} \mathrm{ad}^n_{\boldsymbol{\omega}} \\
&= I_3 + \frac{\mathrm{ad}_{\boldsymbol{\omega}}}{2} + \left(\frac{B_2}{2!} \mathrm{ad}^2_{\boldsymbol{\omega}} - \frac{B_4\|\boldsymbol{\omega}\|^2}{4!} \mathrm{ad}^2_{\boldsymbol{\omega}} + \frac{B_6\|\boldsymbol{\omega}\|^4}{6!} \mathrm{ad}^2_{\boldsymbol{\omega}} - \dots \right) = I_3 + \frac{\mathrm{ad}_{\boldsymbol{\omega}}}{2} - \frac{1}{\|\boldsymbol{\omega}\|^2} \sum_{n\geq 1} \frac{B_{2n}(-1)^n \|\boldsymbol{\omega}\|^{2n}}{(2n)!} \mathrm{ad}^2_{\boldsymbol{\omega}} \\
&\overset{(9.20)}{=} I_3 + \frac{\mathrm{ad}_{\boldsymbol{\omega}}}{2} - \frac{1}{\|\boldsymbol{\omega}\|^2} \left(\frac{\|\boldsymbol{\omega}\|}{2} \cot\left(\frac{\|\boldsymbol{\omega}\|}{2} \right) - 1 \right) \mathrm{ad}^2_{\boldsymbol{\omega}} = I_3 + \frac{\mathrm{ad}_{\boldsymbol{\omega}}}{2} + \left(\frac{1}{\|\boldsymbol{\omega}\|^2} - \frac{1+\cos\|\boldsymbol{\omega}\|}{2\|\boldsymbol{\omega}\|\sin\|\boldsymbol{\omega}\|} \right) \mathrm{ad}^2_{\boldsymbol{\omega}},
\end{aligned}$$

where the half-angle formula $\cot x = (1+\cos x)/\sin x$ has been used. The left jacobian $\mathrm{d}^l \exp_{\boldsymbol{\omega}}$ was already calculated in (13.2) and since $\left(\mathrm{d}^r \exp_{\boldsymbol{\omega}}\right)^{-1} = \left[\left(\mathrm{d}^l \exp_{\boldsymbol{\omega}}\right)^{-1}\right]^T$. Due to the anti-symmetry of $\mathrm{ad}_{\boldsymbol{\omega}}$ it follows that also $\mathrm{d}^r \exp_{\boldsymbol{\omega}} = \left[\mathrm{d}^l \exp_{\boldsymbol{\omega}}\right]^T$ must hold.

11.3. Summary

$\mathbb{SO}(3)$

Group definition

$$\mathbb{SO}(3) = \left\{ \boldsymbol{R} \in \mathbb{GL}(3,\mathbb{R}) \mid \boldsymbol{R}\boldsymbol{R}^T = \boldsymbol{R}^T\boldsymbol{R} = I_3, \det \boldsymbol{R} = 1 \right\}. \tag{11.25}$$

Algebra parameterization

$$\left\{ \boldsymbol{\omega} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \mid \omega_x, \omega_y, \omega_z \in [-\pi, \pi] \right\}, \quad \hat{\boldsymbol{\omega}} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \in \mathfrak{so}(3). \tag{11.26}$$

Adjoint

$$\mathrm{Ad}_{\boldsymbol{R}} = \boldsymbol{R} \tag{11.27}$$

Exponential

$$\exp(\boldsymbol{\omega}) = I + \frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|} \hat{\boldsymbol{\omega}} + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2} \hat{\boldsymbol{\omega}}^2 \tag{11.28}$$

Logarithm

$$\log \boldsymbol{R} = \left(\frac{\alpha}{\sin\alpha} \frac{\boldsymbol{R} - \boldsymbol{R}^T}{2} \right)^{\vee}, \quad \alpha = \arccos\left(\frac{\mathrm{Tr}(\boldsymbol{R}) - 1}{2} \right). \tag{11.29}$$

Lowercase adjoint

$$\mathrm{ad}_{\boldsymbol{\omega}} = \hat{\boldsymbol{\omega}} \tag{11.30}$$

Derivatives of the Exponential

$$\mathrm{d}^r \exp_{\boldsymbol{\omega}} = I_3 - \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}} + \frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^3}\hat{\boldsymbol{\omega}}^2, \tag{11.31a}$$

$$\mathrm{d}^l \exp_{\boldsymbol{\omega}} = I_3 + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}} + \frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^3}\hat{\boldsymbol{\omega}}^2, \tag{11.31b}$$

$$\left(\mathrm{d}^r \exp_{\boldsymbol{\omega}}\right)^{-1} = I_3 + \frac{\hat{\boldsymbol{\omega}}}{2} + \left(\frac{1}{\|\boldsymbol{\omega}\|^2} - \frac{1+\cos\|\boldsymbol{\omega}\|}{2\|\boldsymbol{\omega}\|\sin\|\boldsymbol{\omega}\|}\right)\hat{\boldsymbol{\omega}}^2, \tag{11.31c}$$

$$\left(\mathrm{d}^l \exp_{\boldsymbol{\omega}}\right)^{-1} = I_3 - \frac{\hat{\boldsymbol{\omega}}}{2} + \left(\frac{1}{\|\boldsymbol{\omega}\|^2} - \frac{1+\cos\|\boldsymbol{\omega}\|}{2\|\boldsymbol{\omega}\|\sin\|\boldsymbol{\omega}\|}\right)\hat{\boldsymbol{\omega}}^2. \tag{11.31d}$$

11.4. Parameterization via Double Cover by $\mathbb{S}^3$

The unit quaternion group $\mathbb{S}^3$

Group definition

$$\mathbb{S}^3 = \left\{\boldsymbol{q} = (q_w, q_x, q_y, q_z) \,:\, q_w^2 + q_x^2 + q_z^2 = 1\right\} \tag{11.32}$$

- Identity element: $(1, 0, 0, 0)$
- Inverse: $(q_w, q_x, q_y, q_z)^{-1} = (q_w, -q_x, -q_y, -q_z)$
- Composition via (11.14)

$\mathbb{S}^3$ forms a double cover of $\mathbb{SO}(3)$ via (11.18) and therefore inherits Lie algebra properties from $\mathbb{SO}(3)$.

Exponential

$$\exp(\boldsymbol{\omega}) = \left(\cos(\|\boldsymbol{\omega}\|/2), \frac{\omega_x}{\|\boldsymbol{\omega}\|}\sin(\|\boldsymbol{\omega}\|/2), \frac{\omega_y}{\|\boldsymbol{\omega}\|}\sin(\|\boldsymbol{\omega}\|/2), \frac{\omega_z}{\|\boldsymbol{\omega}\|}\sin(\|\boldsymbol{\omega}\|/2)\right), \tag{11.33}$$

Logarithm

$$\log(\boldsymbol{q}) = \left(2\frac{\operatorname{arctan2}\left(\sqrt{q_x^2 + q_y^2 + q_z^2}, q_w\right)}{\sqrt{q_x^2 + q_y^2 + q_z^2}}\right) \times \begin{bmatrix} q_x \\ q_y \\ q_z \end{bmatrix}. \tag{11.34}$$

12. $\mathbb{SE}(2)$: The 2D Rigid Motion Group

We now combine $\mathbb{E}(2)$ and $\mathbb{SO}(2)$ into a group that simultaneously represents translation and rotation in two dimensions. The result is $\mathbb{SE}(2)$—the special euclidean group in two dimensions.

In matrix form $\mathbb{SE}(2)$ consists of matrices on the form

$$\mathbb{SO}(2) = \left\{ \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \boldsymbol{0}_{1\times 2} & 1 \end{bmatrix} \mid R \in \mathbb{SO}(2) \right\}, \tag{12.1}$$

It follows that both $\mathbb{SO}(2)$ and $\mathbb{E}(2)$ are sub-groups of $\mathbb{SE}(2)$. In addition, $\mathbb{E}(2)$ is a normal subgroup[1] which implies that $\mathbb{SE}(2)$ is a *semi-direct product* denoted $\mathbb{SE}(2) \cong \mathbb{SO}(2) \ltimes \mathbb{E}(2)$. Group products (direct and semi-direct) are discussed further below.

Algebra

Lower-dimensional representation: Four parameters are required, two for each subgroup

$$\left\{ ((q_w, q_z), (p_x, p_y)) \right\}. \tag{12.2}$$

Identity: The identity element is inherited:

Inverse: From matrix inverse it follows that $(\boldsymbol{R}, \boldsymbol{p})^{-1} = (\boldsymbol{R}^T, -\boldsymbol{R}^T \boldsymbol{p})$.

Composition: Matrix multiplication shows that composition in the lower-dimensional representation is

$$(\boldsymbol{R}, \boldsymbol{p}) \circ (\boldsymbol{R}', \boldsymbol{p}') = (\boldsymbol{R}\boldsymbol{R}', \boldsymbol{R}\boldsymbol{p}' + \boldsymbol{p}). \tag{12.3}$$

Action on R^2: This group has a natural action on two-dimensional vectors that consists of rotation and translation. For $X \coloneqq (\boldsymbol{R}, \boldsymbol{p}) \in \mathbb{S}\check{\mathbb{E}}(2)$ the action is

$$\langle X, \boldsymbol{u} \rangle_{\mathbb{SE}(2)} = \langle \boldsymbol{R}, \boldsymbol{u} \rangle_{\mathbb{SO}(2)} + \boldsymbol{p} = \boldsymbol{R}\boldsymbol{u} + \boldsymbol{p}. \tag{12.4}$$

That is, the vector $\boldsymbol{u}$ is first rotated through the action of the $\mathbb{SO}(2)$ part of the state, and then subjected to a translation. This action can be written as a matrix multiplication if we associate $\boldsymbol{u}$ with its homogeneous counterpart $\boldsymbol{u}^H$:

$$\left\langle X, \boldsymbol{u}^H \right\rangle = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \boldsymbol{0}_{1\times 2} & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{u} \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{R}\boldsymbol{u} + \boldsymbol{p} \\ 1 \end{bmatrix}. \tag{12.5}$$

[1] If $X \in \mathbb{E}^2$ and $Y \in \mathbb{SO}(2)$, then $YXY^{-1} \in \mathbb{E}^2$.

The action has a natural interpretation as a change of coordinates: if $\begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \boldsymbol{0}_{1\times 2} & 1 \end{bmatrix} \in \mathbb{SE}(2)$, then $\langle \boldsymbol{X}, \boldsymbol{u} \rangle$ represents the transformation from a coordinate frame attached at $\boldsymbol{p}$ with unit vectors the columns of $\boldsymbol{R}$, to the global coordinate frame.

Action on $\mathbb{R}^2$:

Lie Algebra

12.1. Formulas

Adjoint Take $\boldsymbol{X} = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix}$ and recall the hat formula $\hat{\omega}_z = \begin{bmatrix} 0 & -\omega_z \\ \omega_z & 0 \end{bmatrix}$ and adjoint $(\mathbf{Ad}_{\boldsymbol{R}}\, \omega_z)^\wedge = \boldsymbol{R}\hat{\omega}_z\boldsymbol{R}^T = \hat{\omega}_z$ from $\mathbb{SO}(2)$. Evaluating (4.13) gives

$$\begin{aligned}
\mathrm{Ad}_X \begin{bmatrix} \boldsymbol{v} \\ \omega_z \end{bmatrix} &= \left(\begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \hat{\omega}_z & \boldsymbol{v} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix}^{-1} \right)^\vee = \left(\begin{bmatrix} \boldsymbol{R}\hat{\omega}_z & \boldsymbol{R}\boldsymbol{v} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \boldsymbol{R}^T & -\boldsymbol{R}^T\boldsymbol{p} \\ 0 & 1 \end{bmatrix} \right)^\vee \\
&= \begin{bmatrix} \boldsymbol{R}\hat{\omega}_z\boldsymbol{R}^T & \boldsymbol{R}\boldsymbol{v} - \boldsymbol{R}\hat{\omega}_z\boldsymbol{R}^T\boldsymbol{p} \\ 0 & 1 \end{bmatrix}^\vee = \begin{bmatrix} \boldsymbol{R}\boldsymbol{v} - \hat{\omega}_z\boldsymbol{p} \\ \omega_z \end{bmatrix} = \begin{bmatrix} \boldsymbol{R} & -\hat{1}\,\boldsymbol{p} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \boldsymbol{v} \\ \omega_z \end{bmatrix},
\end{aligned} \tag{12.6}$$

which exposes the adjoint as the matrix $\begin{bmatrix} \boldsymbol{R} & -\hat{1}\,\boldsymbol{p} \\ 0 & 1 \end{bmatrix}$.

Exponential and Logarithm We use Lemma 9.1 to derive the exponential map. The Lie algebra elements have structure

$$\boldsymbol{A} = \boldsymbol{B} + \boldsymbol{C}, \quad \boldsymbol{B} = \begin{bmatrix} \hat{\omega}_z & \boldsymbol{0} \\ \boldsymbol{0} & 0 \end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{v} \\ \boldsymbol{0} & 0 \end{bmatrix}. \tag{12.7}$$

Thus, it suffices to compute $S(\omega) := \sum_{k=0}^{\infty} \frac{\hat{\boldsymbol{\omega}}}{(k+1)!}$ to obtain the exponential map for the semi-simple groups.

For $\mathfrak{se}(2)$, disregarding the trivial case $\omega_z = 0$, we obtain

$$\begin{aligned}
S(\omega_z) &= \sum_{k=0}^{\infty} \frac{\hat{\omega}_z^k}{(k+1)!} = (\hat{\omega}_z)^{-1}\,(\mathrm{Exp}\,\hat{\omega}_z - I) \\
&= \frac{1}{\omega_z^2} \begin{bmatrix} 0 & \omega_z \\ -\omega_z & 0 \end{bmatrix} \begin{bmatrix} \cos\omega_z - 1 & -\sin\omega_z \\ \sin\omega_z & \cos\omega_z - 1 \end{bmatrix} = \frac{1}{\omega_z} \begin{bmatrix} \sin\omega_z & \cos\omega_z - 1 \\ 1 - \cos\omega_z & \sin\omega_z \end{bmatrix}.
\end{aligned} \tag{12.8}$$

Lemma 9.1 now gives the exponential.

$$\exp(\boldsymbol{A}) = \exp(\boldsymbol{B} + \boldsymbol{C}) = \exp(\boldsymbol{B}) + \begin{bmatrix} S(\omega_z) & 0 \\ 0 & 1 \end{bmatrix} C = \begin{bmatrix} \exp_{\mathbb{SO}(2)} \omega_z & S(\omega_z)\boldsymbol{v} \\ 0 & 1 \end{bmatrix} \tag{12.9}$$

Derivatives of the Exponential We first calculate an expression for the bracket.

$$\begin{aligned}\left[\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}, \begin{bmatrix} \bar{v}_x \\ \bar{v}_y \\ \bar{\omega}_z \end{bmatrix}\right] &= \left(\begin{bmatrix} 0 & -\omega_z & v_x \\ \omega_z & 0 & v_y \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & -\bar{\omega}_z & \bar{v}_x \\ \bar{\omega}_z & 0 & \bar{v}_y \\ 0 & 0 & 0 \end{bmatrix} - \begin{bmatrix} 0 & -\bar{\omega}_z & \bar{v}_x \\ \bar{\omega}_z & 0 & \bar{v}_y \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} 0 & -\omega_z & v_x \\ \omega_z & 0 & v_y \\ 0 & 0 & 0 \end{bmatrix}\right)^\vee \\ &= \begin{bmatrix} 0 & 0 & -\omega_z\bar{v}_y + \bar{\omega}_z v_y \\ 0 & 0 & \omega_z\bar{v}_x - \bar{\omega}_z v_x \\ 0 & 0 & 0 \end{bmatrix}^\vee = \begin{bmatrix} -\omega_z\bar{v}_y + \bar{\omega}_z v_y \\ \omega_z\bar{v}_x - \bar{\omega}_z v_x \\ 0 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & -\omega_z & v_y \\ \omega_z & 0 & -v_x \\ 0 & 0 & 0 \end{bmatrix}}_{\mathrm{ad}_{\boldsymbol{a}}}\begin{bmatrix} \bar{v}_x \\ \bar{v}_y \\ \bar{\omega}_z \end{bmatrix}.\end{aligned} \tag{12.10}$$

A quick calculation reveals that $\mathrm{ad}_{\boldsymbol{a}}^3 = -\omega_z^2\,\mathrm{ad}_{\boldsymbol{a}}$, which is exactly the relation we used for $\mathbb{SO}(3)$ above. Consequently the inverse derivatives must have the same form as on $\mathbb{SO}(3)$.

$\mathbb{SE}(2)$ formula sheet

Consists of 3×3 matrices $\begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ 0 & 1 \end{bmatrix}$ that act on $\mathbb{R}^2$ via $\boldsymbol{v} \mapsto \boldsymbol{R}\boldsymbol{v} + \boldsymbol{p}$.

Algebra Parameterization

$$\left\{\begin{bmatrix} \boldsymbol{v} \\ \omega_z \end{bmatrix} \mid \boldsymbol{v} \in \mathbb{R}^2, \omega_z \in [-\pi, \pi]\right\}, \quad \begin{bmatrix} \boldsymbol{v} \\ \omega_z \end{bmatrix}^\wedge = \begin{bmatrix} \hat{\omega}_z & \boldsymbol{v} \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(2). \tag{12.11}$$

Adjoint

$$\mathrm{Ad}_X = \begin{bmatrix} \boldsymbol{R} & \begin{bmatrix} p_y \\ -p_x \end{bmatrix} \\ 0 & 1 \end{bmatrix} \tag{12.12}$$

Exponential and Logarithm

$$\exp_{\mathbb{SE}(2)}\left(\begin{bmatrix} \boldsymbol{v} \\ \omega_z \end{bmatrix}\right) = \begin{bmatrix} \exp_{\mathbb{SO}(2)}(\omega_z) & S(\omega_z)\boldsymbol{v} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix}, \tag{12.13}$$

$$\log_{\mathbb{SE}(2)}\begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \mathbf{0}_{1\times 2} & 1 \end{bmatrix} = \begin{bmatrix} (S(\omega_z))^{-1}\,\boldsymbol{p} \\ \alpha \end{bmatrix}. \tag{12.14}$$

Bracket and Lowercase Adjoint

$$\begin{aligned}\left[\begin{bmatrix} v_x \\ v_y \\ \omega_z \end{bmatrix}, \begin{bmatrix} v'_x \\ v'_y \\ \omega'_z \end{bmatrix}\right] &= \begin{bmatrix} -\omega_z v'_y + \omega'_z v_y \\ \omega_z v'_x - \omega'_z v_x \\ 0 \end{bmatrix}, \\ \mathrm{ad}_{\boldsymbol{a}} &= \begin{bmatrix} 0 & -\omega_z & v_y \\ \omega_z & 0 & -v_x \\ 0 & 0 & 0 \end{bmatrix}.\end{aligned} \tag{12.15}$$

Derivatives of the Exponential Let $\boldsymbol{a} = \begin{bmatrix} v_x & v_y & \omega_z \end{bmatrix}^T$. Then,

$$\mathrm{d}^r \exp_{\boldsymbol{a}} = I_3 - \frac{1 - \cos\omega_z}{\omega_z^2}\mathrm{ad}_{\boldsymbol{a}} + \frac{\omega_z - \sin\omega_z}{\omega_z^3}\mathrm{ad}_{\boldsymbol{a}}{}^2, \tag{12.16}$$

$$\mathrm{d}^l \exp_{\boldsymbol{a}} = I_3 + \frac{1 - \cos\omega_z}{\omega_z^2}\mathrm{ad}_{\boldsymbol{a}} + \frac{\omega_z - \sin\omega_z}{\omega_z^3}\mathrm{ad}_{\boldsymbol{a}}{}^2, \tag{12.17}$$

$$\left(\mathrm{d}^r \exp_{\boldsymbol{a}}\right)^{-1} = I_3 + \frac{\mathrm{ad}_{\boldsymbol{a}}}{2} + \left(\frac{1}{\omega_z^2} - \frac{1 + \cos\omega_z}{2\omega_z \sin\omega_z}\right)\mathrm{ad}_{\boldsymbol{a}}^2, \tag{12.18}$$

$$\left(\mathrm{d}^l \exp_{\boldsymbol{a}}\right)^{-1} = I_3 - \frac{\mathrm{ad}_{\boldsymbol{a}}}{2} + \left(\frac{1}{\omega_z^2} - \frac{1 + \cos\omega_z}{2\omega_z \sin\omega_z}\right)\mathrm{ad}_{\boldsymbol{a}}^2. \tag{12.19}$$

12.2. Parameterization via Isomorphism with $\mathbb{U}(1) \ltimes \mathbb{R}^2$

13. $\$\mathbb{E}(3)$: The 3D Rigid Motion Group

Parameterization

Action on $\mathbb{R}^3$:

Lie Algebra

13.1. Formulas

Adjoint

Exponential and Logarithm Continuing with $\$\mathbb{E}(3)$ we utilize (11.2a) to calculate

$$\sum_{k=0}^{\infty} \frac{\hat{\boldsymbol{\omega}}^k}{(k+1)!} = I_3 - \frac{1}{\|\boldsymbol{\omega}\|^2}\hat{}\sum_{k=2}^{\infty} \frac{\hat{\boldsymbol{\omega}}^k}{k!} = I_3 - \frac{1}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}\left(\mathrm{Exp}_{\$\mathbb{O}(3)}(\hat{\boldsymbol{\omega}}) - I_3 - \hat{\boldsymbol{\omega}}\right). \tag{13.1}$$

From (11.28) we then have that

$$\begin{aligned} \mathrm{d}^l\left(\exp_{\$\mathbb{O}(3)}\right)_{\boldsymbol{\omega}} &:= \sum_{k=0}^{\infty} \frac{\hat{\boldsymbol{\omega}}^k}{(k+1)!} = I_3 - \frac{1}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}\left(\left(I_3 + \frac{\sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|}\hat{\boldsymbol{\omega}} + \frac{(1-\cos\|\boldsymbol{\omega}\|)}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}^2\right) - I_3 - \hat{\boldsymbol{\omega}}\right) \\ &= I_3 - \frac{(\sin\|\boldsymbol{\omega}\| - \|\boldsymbol{\omega}\|)}{\|\boldsymbol{\omega}\|^3}\hat{\boldsymbol{\omega}}^2 + \frac{\|\boldsymbol{\omega}\|^2(1-\cos\|\boldsymbol{\omega}\|)}{\|\boldsymbol{\omega}\|^4}\hat{\boldsymbol{\omega}} \\ &= I_3 + \frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^3}\hat{\boldsymbol{\omega}}^2 + \frac{1-\cos\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^2}\hat{\boldsymbol{\omega}}. \end{aligned} \tag{13.2}$$

Applying Lemma 9.1 then gives the exponential.

The matrix exponential on $\mathfrak{se}(3)$ is

$$\exp_{\mathbb{SE}(3)}(\boldsymbol{\omega}, \boldsymbol{v}) = \operatorname{Exp}_{\mathbb{SE}(3)} \begin{bmatrix} \hat{\boldsymbol{\omega}} & \boldsymbol{v} \\ \mathbf{0}_{1\times 3} & 0 \end{bmatrix} = \begin{bmatrix} \operatorname{Exp}_{\mathbb{SO}(3)}(\hat{\boldsymbol{\omega}}) & \mathrm{d}^l \left(\exp_{\mathbb{SO}(3)}\right)_{\boldsymbol{\omega}} \boldsymbol{v} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \tag{13.3a}$$

$$\operatorname{Log}_{\mathbb{SE}(3)} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{\alpha}} & \left(\mathrm{d}^l \left(\exp_{\mathbb{SO}(3)}\right)_{\boldsymbol{\alpha}}\right)^{-1} \boldsymbol{p} \\ \mathbf{0}_{1\times 3} & 0 \end{bmatrix}, \tag{13.3b}$$

$$\log_{\mathbb{SE}(3)} \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix} = \left(\boldsymbol{\alpha}, \left(\mathrm{d}^l \left(\exp_{\mathbb{SO}(3)}\right)_{\boldsymbol{\alpha}}\right)^{-1} \boldsymbol{p}\right), \tag{13.3c}$$

where $\boldsymbol{\alpha} = \log_{\mathbb{SO}(3)} R$.

Derivatives of the Exponential First we derive an expression for $\mathrm{ad}_{\boldsymbol{a}}$ utilizing that for the hat operator on $\mathbb{SO}(3)$, $\hat{\boldsymbol{a}}\boldsymbol{b} = -\hat{\boldsymbol{b}}\boldsymbol{a}$

$$\begin{aligned} \left[\begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{bmatrix}, \begin{bmatrix} \bar{\boldsymbol{v}} \\ \bar{\boldsymbol{\omega}} \end{bmatrix}\right]_{\mathbb{SE}(3)} &= \left(\begin{bmatrix} \hat{\boldsymbol{\omega}} & \boldsymbol{v} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \hat{\bar{\boldsymbol{\omega}}} & \bar{\boldsymbol{v}} \\ 0 & 0 \end{bmatrix} - \begin{bmatrix} \hat{\bar{\boldsymbol{\omega}}} & \bar{\boldsymbol{v}} \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \hat{\boldsymbol{\omega}} & \boldsymbol{v} \\ 0 & 0 \end{bmatrix}\right)^{\vee} = \begin{bmatrix} [\boldsymbol{\omega}, \bar{\boldsymbol{\omega}}]^{\wedge}_{\mathbb{SO}(3)} & \hat{\boldsymbol{\omega}}\bar{\boldsymbol{v}} - \hat{\bar{\boldsymbol{\omega}}}\boldsymbol{v} \\ 0 & 0 \end{bmatrix}^{\vee} \\ &= \begin{bmatrix} \hat{\boldsymbol{\omega}}\bar{\boldsymbol{v}} - \hat{\bar{\boldsymbol{\omega}}}\boldsymbol{v} \\ [\boldsymbol{\omega}, \bar{\boldsymbol{\omega}}]_{\mathbb{SO}(3)} \end{bmatrix} = \underbrace{\begin{bmatrix} \hat{\boldsymbol{\omega}} & \hat{\boldsymbol{v}} \\ 0 & \hat{\boldsymbol{\omega}} \end{bmatrix}}_{\mathrm{ad}_{\boldsymbol{a}}} \begin{bmatrix} \bar{\boldsymbol{v}} \\ \bar{\boldsymbol{\omega}} \end{bmatrix}. \end{aligned} \tag{13.4}$$

We are interested in the powers $\mathrm{ad}^k_{\boldsymbol{a}}$ in order to evaluate the exponential derivatives. For $k \geq 1$

$$\mathrm{ad}^k_{\boldsymbol{a}} = \begin{bmatrix} \hat{\boldsymbol{\omega}} & \hat{\boldsymbol{v}} \\ 0 & \hat{\boldsymbol{\omega}} \end{bmatrix}^k = \begin{bmatrix} \hat{\boldsymbol{\omega}}^k & \sum_{i=0}^{k-1} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k-1-i} \\ 0 & \hat{\boldsymbol{\omega}}^k \end{bmatrix}. \tag{13.5}$$

Thus the left derivative of the exponential can be written

$$\mathrm{d}^l \exp_{\boldsymbol{a}} = \sum_{k=0}^{\infty} \frac{\mathrm{ad}^k_{\boldsymbol{a}}}{(k+1)!} = I + \sum_{k=1}^{\infty} \frac{1}{(k+1)!} \begin{bmatrix} \hat{\boldsymbol{\omega}}^k & \sum_{i=0}^{k-1} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k-1-i} \\ 0 & \hat{\boldsymbol{\omega}}^k \end{bmatrix} = \begin{bmatrix} \mathrm{d}^l \left(\exp_{\mathbb{SO}(3)}\right)_{\boldsymbol{\omega}} & Q^l(\boldsymbol{v}, \boldsymbol{\omega}) \\ 0 & \mathrm{d}^l \left(\exp_{\mathbb{SO}(3)}\right)_{\boldsymbol{\omega}} \end{bmatrix},$$

where a closed-form expression for $Q^l(\boldsymbol{v}, \boldsymbol{\omega})$ can be painstakingly obtained through a series of sum manipulations. We first convert the formula to a form that is symmetric in i and k.

$$\begin{aligned} Q^l(\boldsymbol{v}, \boldsymbol{\omega}) &:= \sum_{k=1}^{\infty} \frac{1}{(k+1)!} \sum_{i=0}^{k-1} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k-1-i} = \sum_{k=0}^{\infty} \sum_{i=0}^{k} \frac{1}{(k+2)!} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k-i} \\ &= \sum_{i=0}^{\infty} \sum_{k=i}^{\infty} \frac{1}{(k+2)!} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k-i} = \sum_{i=0}^{\infty} \sum_{k=0}^{\infty} \frac{1}{(k+i+2)!} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k}. \end{aligned}$$

With the same steps the right derivative can be shown to be

$$Q^r(\boldsymbol{v}, \boldsymbol{\omega}) := \sum_{k=1}^{\infty} \frac{(-1)^k}{(k+1)!} \sum_{i=0}^{k-1} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k-1-i} = \ldots = -\sum_{i=0}^{\infty} \sum_{k=0}^{\infty} \frac{(-1)^{k+i}}{(k+i+2)!} \hat{\boldsymbol{\omega}}^i \hat{\boldsymbol{v}} \hat{\boldsymbol{\omega}}^{k} \tag{13.6}$$

and we can see that

$$Q^r(\boldsymbol{v},\boldsymbol{\omega}) = Q^l(-\boldsymbol{v},-\boldsymbol{\omega}) \tag{13.7}$$

which is convenient to know since calculating one of them is tedious enough.

In the following calculation the sum $\sum_{k,i\geq 0}$ is first split into parts ($k = i = 0$), ($k = 0, i \geq 1$), ($k \geq 1, i = 0$) and ($k, i \geq 1$), and then the resulting single sums are split into two sums $i = 0, 2, \ldots$ and $i = 1, 3, \ldots$. Also using that

$$\hat{\boldsymbol{\omega}}^{2k+1} = (-1)^k \|\boldsymbol{\omega}\|^{2k}\hat{\boldsymbol{\omega}}, \qquad \hat{\boldsymbol{\omega}}^{2k+2} = (-1)^k \|\boldsymbol{\omega}\|^{2k}\hat{\boldsymbol{\omega}}^2, \tag{13.8}$$

which follows from (11.2a), we get

$$\begin{aligned}
Q^l(\boldsymbol{v},\boldsymbol{\omega}) &= \frac{1}{2}\hat{\boldsymbol{v}} + \sum_{i=1}^{\infty}\frac{\hat{\boldsymbol{\omega}}^i\hat{\boldsymbol{v}}}{(i+2)!} + \sum_{k=1}^{\infty}\frac{\hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^k}{(k+2)!} + \sum_{i=1}^{\infty}\sum_{k=1}^{\infty}\frac{1}{(i+k+2)!}\hat{\boldsymbol{\omega}}^i\hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^k \\
&= \frac{1}{2}\hat{\boldsymbol{v}} + \sum_{i=0}^{\infty}\frac{\hat{\boldsymbol{\omega}}^{i+1}\hat{\boldsymbol{v}} + \hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^{i+1}}{(i+3)!} + \sum_{i=0}^{\infty}\sum_{k=0}^{\infty}\frac{1}{(i+k+4)!}\hat{\boldsymbol{\omega}}^{i+1}\hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^{k+1} \\
&= \frac{1}{2}\hat{\boldsymbol{v}} + \sum_{i=0}^{\infty}\frac{\hat{\boldsymbol{\omega}}^{2i+1}\hat{\boldsymbol{v}} + \hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^{2i+1}}{(2i+3)!} + \sum_{i=0}^{\infty}\frac{\hat{\boldsymbol{\omega}}^{2i+2}\hat{\boldsymbol{v}} + \hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^{2i+2}}{(2i+4)!} + \sum_{i=0}^{\infty}\sum_{k=0}^{\infty}\frac{1}{(i+k+4)!}\hat{\boldsymbol{\omega}}^{i+1}\hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^{k+1} \\
&= \frac{1}{2}\hat{\boldsymbol{v}} + \sum_{i=0}^{\infty}\frac{(-1)^i\|\boldsymbol{\omega}\|^{2i}}{(2i+3)!}\left(\hat{\boldsymbol{\omega}}\hat{\boldsymbol{v}} + \hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}\right) + \sum_{i=0}^{\infty}\frac{(-1)^i\|\boldsymbol{\omega}\|^{2i}}{(2i+4)!}\left(\hat{\boldsymbol{\omega}}^2\hat{\boldsymbol{v}} + \hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^2\right) + \sum_{i=0}^{\infty}\sum_{k=0}^{\infty}\frac{1}{(i+k+4)!}\hat{\boldsymbol{\omega}}^{i+1}\hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^{k+1}.
\end{aligned}$$

The first two sums can now be evaluated in a fairly straightforward manner:

$$\sum_{i=0}^{\infty}\frac{(-1)^i}{(2i+3)!}\|\boldsymbol{\omega}\|^{2i} = -\frac{1}{\|\boldsymbol{\omega}\|^3}\sum_{i=0}^{\infty}\frac{(-1)^{i+1}}{(2(i+1)+1)!}\|\boldsymbol{\omega}\|^{2(i+1)+1} = \frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^3}, \tag{13.9a}$$

$$\sum_{i=0}^{\infty}\frac{(-1)^i}{(2i+4)!}\|\boldsymbol{\omega}\|^{2i} = \frac{1}{\|\boldsymbol{\omega}\|^4}\sum_{i=0}^{\infty}\frac{(-1)^{i+2}}{(2(i+2))!}\|\boldsymbol{\omega}\|^{2(i+2)} = \frac{\cos\|\boldsymbol{\omega}\| - 1 + \frac{\|\boldsymbol{\omega}\|^2}{2}}{\|\boldsymbol{\omega}\|^4}. \tag{13.9b}$$

The double sum requries additional work. Using $\hat{\boldsymbol{\omega}}\hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}} = (-\boldsymbol{\omega}\cdot\boldsymbol{v})\hat{\boldsymbol{\omega}}$ from (11.2a) yields

$$\begin{aligned}
\sum_{i=0}^{\infty}\sum_{k=0}^{\infty}\frac{1}{(k+i+4)!}\hat{\boldsymbol{\omega}}^{i+1}\hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^{k+1} &= (-\boldsymbol{\omega}\cdot\boldsymbol{v})\sum_{i=0}^{\infty}\sum_{k=0}^{\infty}\frac{1}{(k+i+4)!}\hat{\boldsymbol{\omega}}^{k+i+1} \overset{j=k+i}{=} (-\boldsymbol{\omega}\cdot\boldsymbol{v})\sum_{j=0}^{\infty}\sum_{k=0}^{j}\frac{1}{(j+4)!}\hat{\boldsymbol{\omega}}^{j+1} \\
&= -(\boldsymbol{\omega}\cdot\boldsymbol{v})\sum_{j=0}^{\infty}\frac{j+1}{(j+4)!}\hat{\boldsymbol{\omega}}^{j+1} = -(\boldsymbol{\omega}\cdot\boldsymbol{v})\sum_{j=0}^{\infty}\left(\frac{1}{(j+3)!} - \frac{3}{(j+4)!}\right)\hat{\boldsymbol{\omega}}^{j+1} \\
&= -(\boldsymbol{\omega}\cdot\boldsymbol{v})\left(\sum_{j=0}^{\infty}\left(\frac{1}{(2j+3)!} - \frac{3}{(2j+4)!}\right)\hat{\boldsymbol{\omega}}^{2j+1} + \sum_{j=0}^{\infty}\left(\frac{1}{(2j+4)!} - \frac{3}{(2j+5)!}\right)\hat{\boldsymbol{\omega}}^{2j+2}\right) \\
&\overset{(13.8)}{=} (\boldsymbol{\omega}\cdot\boldsymbol{v})\left(-\sum_{j=0}^{\infty}\left(\frac{(-1)^j}{(2j+3)!}\|\boldsymbol{\omega}\|^{2j} + 3\frac{(-1)^j}{(2j+4)!}\|\boldsymbol{\omega}\|^{2j}\right)\hat{\boldsymbol{\omega}} + \sum_{j=0}^{\infty}\left(-\frac{(-1)^j}{(2j+4)!}\|\boldsymbol{\omega}\|^{2j} + 3\frac{(-1)^j}{(2j+5)!}\|\boldsymbol{\omega}\|^{2j}\right)\hat{\boldsymbol{\omega}}^2\right).
\end{aligned}$$

The sums (13.9a) and (13.9b) appear again and can be re-used. The remaining sum with denominator $(2j+5)!$ evaluates to a higher-order sine expression as follows

$$\sum_{j=0}^{\infty} \frac{(-1)^j}{(2j+5)!} \|\boldsymbol{\omega}\|^{2j} = \frac{1}{\|\boldsymbol{\omega}\|^5} \sum_{j=0}^{\infty} \frac{(-1)^{j+2}}{((2j+2)+1)} \|\boldsymbol{\omega}\|^{2(j+2)+1} = \frac{\sin\|\boldsymbol{\omega}\| - \|\boldsymbol{\omega}\| + \frac{\|\boldsymbol{\omega}\|}{6}}{\|\boldsymbol{\omega}\|^5}. \tag{13.10}$$

After collecting the various expressions the closed-form expression for Q^l can be written down

$$\begin{aligned} Q^l(\boldsymbol{v}, \boldsymbol{\omega}) = {} & \frac{1}{2}\hat{\boldsymbol{v}} + \frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\|}{\|\boldsymbol{\omega}\|^3} \left(\hat{\boldsymbol{\omega}}\hat{\boldsymbol{v}} + \hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}} - (\boldsymbol{\omega}\cdot\boldsymbol{v})\hat{\boldsymbol{\omega}}\right) \\ & + \frac{\cos\|\boldsymbol{\omega}\| - 1 + \frac{\|\boldsymbol{\omega}\|^2}{2}}{\|\boldsymbol{\omega}\|^4} \left(\hat{\boldsymbol{\omega}}^2\hat{\boldsymbol{v}} + \hat{\boldsymbol{v}}\hat{\boldsymbol{\omega}}^2 + (\boldsymbol{\omega}\cdot\boldsymbol{v})(3\hat{\boldsymbol{\omega}} - \hat{\boldsymbol{\omega}}^2)\right) \\ & - 3(\boldsymbol{\omega}\cdot\boldsymbol{v}) \left(\frac{\|\boldsymbol{\omega}\| - \sin\|\boldsymbol{\omega}\| - \frac{\|\boldsymbol{\omega}\|^3}{6}}{\|\boldsymbol{\omega}\|^5} \right) \hat{\boldsymbol{\omega}}^2. \end{aligned} \tag{13.11}$$

The Q matrix allows us to write down a closed-form expression for $\mathrm{d}^l \exp_{\boldsymbol{a}}$ on $\mathbb{SE}(3)$, and $\left(\mathrm{d}^l \exp_{\boldsymbol{a}}\right)^{-1}$ follows from noting that $\begin{bmatrix} A & B \\ 0 & A \end{bmatrix}^{-1} = \begin{bmatrix} A^{-1} & -A^{-1}BA^{-1} \\ 0 & A^{-1} \end{bmatrix}$ for A invertible.

Lowercase adjoint and exponential derivatives on $\mathbb{SE}(3)$

Let $\boldsymbol{a} = \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{bmatrix}$ and Q^l as in (13.11). Then

$$\mathrm{ad}_{\boldsymbol{a}} = \begin{bmatrix} \hat{\boldsymbol{\omega}} & \hat{\boldsymbol{v}} \\ 0 & \hat{\boldsymbol{\omega}} \end{bmatrix}, \tag{13.12}$$

$$\mathrm{d}^r \exp_{\boldsymbol{a}} = \begin{bmatrix} J^r_{\mathbb{SO}(3)} & Q^l(-\boldsymbol{v}, -\boldsymbol{\omega}) \\ 0 & J^r_{\mathbb{SO}(3)} \end{bmatrix}, \tag{13.13}$$

$$\mathrm{d}^l \exp_{\boldsymbol{a}} = \begin{bmatrix} J^l_{\mathbb{SO}(3)} & Q^l(\boldsymbol{v}, \boldsymbol{\omega}) \\ 0 & J^l_{\mathbb{SO}(3)} \end{bmatrix} \tag{13.14}$$

$$\left(\mathrm{d}^r \exp_{\boldsymbol{a}}\right)^{-1} = \begin{bmatrix} \left(J^r_{\mathbb{SO}(3)}\right)^{-1} & -\left(J^r_{\mathbb{SO}(3)}\right)^{-1} Q^l(-\boldsymbol{v}, -\boldsymbol{\omega}) \left(J^r_{\mathbb{SO}(3)}\right)^{-1} \\ 0 & \left(J^r_{\mathbb{SO}(3)}\right)^{-1} \end{bmatrix}, \tag{13.15}$$

$$\left(\mathrm{d}^l \exp_{\boldsymbol{a}}\right)^{-1} = \begin{bmatrix} \left(J^l_{\mathbb{SO}(3)}\right)^{-1} & -\left(J^l_{\mathbb{SO}(3)}\right)^{-1} Q^l(\boldsymbol{v}, \boldsymbol{\omega}) \left(J^l_{\mathbb{SO}(3)}\right)^{-1} \\ 0 & \left(J^l_{\mathbb{SO}(3)}\right)^{-1} \end{bmatrix}. \tag{13.16}$$

where $J^{l/r}_{\mathbb{SO}(3)} = \left(\mathrm{d}^{l/r} \exp_{\mathbb{SO}(3)}\right)_{\boldsymbol{\omega}}$ and $\left(J^{l/r}_{\mathbb{SO}(3)}\right)^{-1} = \left(\left(\mathrm{d}^{l/r} \exp_{\mathbb{SO}(3)}\right)_{\boldsymbol{\omega}}\right)^{-1}$. Note that in these formulas $\hat{\boldsymbol{\omega}}$ and $\hat{\boldsymbol{v}}$ denote the hat operator on $\mathbb{SO}(3)$.

Formulas for $\mathbb{SE}(3)$

Group Parameterization

(13.17)

Algebra Parameterization

(13.18)

Group Operations

- Identity element:
- Inverse:
- Composition:

Adjoint

(13.19)

Exponential

(13.20)

Logarithm

(13.21)

Lowercase adjoint

(13.22)

Derivatives of the Exponential

(13.23)

13.2. Parameterization via Isomorphism with $\mathbb{S}^3 \ltimes \mathbb{R}^3$

Part III.

Applications

14. Geometric Numerical Integration

- Two methods:
  - use $\oplus_r$, $\ominus_r$ for regular RK schemes
  - magnus method schemes like in [3]

15. Application: Geometric Control

Summary

- Extend PD theory to Lie Groups.
- Model-predictive control

15.1. A Stabilizing Lie Group Controller

Consider the system

$$\begin{aligned} \mathrm{d}^r X_t &= v \\ \mathrm{d}^r v_t &= u \end{aligned} \tag{15.1}$$

where u is a control input, and the objective of tracking a twice differentiable trajectory $X_d(t)$ with first and second right-derivatives v_d and a_d. Consider the error

$$e_X := X_d \ominus_r X, \tag{15.2}$$

with derivative

$$\mathrm{d}^r(e_X)_t \overset{(5.47),(5.48)}{=} \left(\mathrm{d}^r \exp_{e_X}\right)^{-1} \mathrm{d}^r X_d - \left(\mathrm{d}^l \exp_{e_X}\right)^{-1} \mathrm{d}^r X = \left(\mathrm{d}^l \exp_{e_X}\right)^{-1} (\mathbf{Ad}_{\exp(e_X)} v_d - v). \tag{15.3}$$

Note that $\mathbf{Ad}_{\exp(e_X)} = \exp \mathrm{ad}_e = \sum_{k\geq 0} \frac{\mathrm{ad}_e}{k!}$ can typically be found on closed form via the usual expansion tricks. Let $e_v := \mathbf{Ad}_{\exp(e_X)} v_d - v$ be the velocity error in the body frame; we then have the double intergrator-like error system

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} e_X &= \left(\mathrm{d}^l \exp_{e_X}\right)^{-1} e_v, \\ \frac{\mathrm{d}}{\mathrm{d}t} e_v &= \frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{Ad}_{\exp(e_X)} v_d\right) - u, \end{aligned} \tag{15.4}$$

Where we can further simplify

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{Ad}_{\exp(e_X)} v_d\right) \overset{(5.17c)}{=} \left[\mathrm{d}^l \exp_{e_X} \dot{e}_X, \mathbf{Ad}_{\exp(e_X)} v_d\right] + \mathbf{Ad}_{\exp(e_X)} \dot{v}_d = \left[e_v, \mathbf{Ad}_{\exp(e_X)} v_d\right] + \mathbf{Ad}_{\exp(e_X)} \dot{v}_d. \tag{15.5}$$

If we further consider an input on the form $u = \left[e_v, \mathbf{Ad}_{\exp(e_X)} v_d\right] + \mathbf{Ad}_{\exp(e_X)} \dot{v}_d + k_p \left(\mathrm{d}^l \exp_{e_X}\right)^{-T} e_X + k_d e_v$ that cancels out the contribution from v_d and adds PD feedback terms the closed-loop dynamics become

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t} e_X &= \left(\mathrm{d}^l \exp_{e_X}\right)^{-1} e_v, \\ \frac{\mathrm{d}}{\mathrm{d}t} e_v &= -k_p \left(\mathrm{d}^l \exp_{e_X}\right)^{-T} e_X - k_d e_v. \end{aligned} \tag{15.6}$$

Now consider a Lyapunov candidate function on the form

$$V = \frac{k_p}{2}\|e_X\|^2 + \frac{1}{2}\|e_v\|^2 + c\langle e_v, e_X\rangle \geq \frac{1}{2}\begin{bmatrix}\|e_X\| \\ \|e_v\|\end{bmatrix}^T \begin{bmatrix}k_p & -c \\ -c & 1\end{bmatrix}\begin{bmatrix}\|e_X\| \\ \|e_v\|\end{bmatrix}, \tag{15.7}$$

where c is s.t. $k_p - c^2 \geq 0$ so that the matrix is positive definite. Its derivative evaluates to

$$\begin{aligned}
\dot{V} &= k_p\left\langle e_X, \left(\mathrm{d}^l \exp_{e_X}\right)^{-1} e_v\right\rangle - k_p\left\langle e_v, \left(\mathrm{d}^l \exp_{e_X}\right)^{-T} e_X\right\rangle - k_d\|e_v\|^2 + c\langle \dot{e}_v, e_X\rangle + c\langle e_v, \dot{e}_X\rangle \\
&= -k_d\|e_v\|^2 - c\left\langle k_p\left(\mathrm{d}^l \exp_{e_X}\right)^{-T} e_X + k_d e_v, e_X\right\rangle + c\left\langle e_v, \left(\mathrm{d}^l \exp_{e_X}\right)^{-1} e_v\right\rangle \\
&= -k_d\|e_v\|^2 - ck_p\|e_X\|^2 - ck_d\langle e_v, e_X\rangle + c\left\langle e_v, \left(\mathrm{d}^l \exp_{e_X}\right)^{-1} e_v\right\rangle - ck_p\left\langle ((\mathrm{d}^l \exp_{e_X})^{-T} - I)e_X, e_X\right\rangle \\
&\leq -k_d\|e_v\|^2 - ck_p\|e_X\|^2 + ck_d\|e_v\|\|e_X\| + c\lambda_{\max}\left(\left(\mathrm{d}^l \exp_{e_X}\right)^{-1}\right)\|e_v\|^2 + ck_p\lambda_{\max}\left(\left(\mathrm{d}^l \exp_{e_X}\right)^{-1} - I\right)\|e_X\|^2.
\end{aligned}$$

- Eigenvalues of $(\mathrm{d}^l \exp_{e_X})^{-1}$ can be shown to be on the form $\frac{\lambda}{e^\lambda - 1} = \sum_{k=0}^{\infty}\frac{B_n}{n!}\lambda^n$, where λ is an eigenvalue of ad_e.
- Zero is always an eigenvalue of ad_e since $\mathrm{ad}_e\, e = 0$ due to it being a commutator (the corresponding eigenvalue of $(\mathrm{d}^l \exp_e)^{-1}$ is 1
- Often, eigenvalues of ad_e are purely imaginary. The corresponding eigenvalues of $(\mathrm{d}^l \exp_{e_X})^{-1}$ are

$$\frac{i\lambda}{e^{i\lambda} - 1} = \frac{i\lambda e^{-i\lambda/2}}{e^{i\lambda/2} - e^{-i\lambda/2}} = \frac{i\lambda e^{-i\lambda/2}}{2i\sin\lambda/2} = \lambda\frac{\cos\lambda/2 - i\sin\lambda/2}{2\sin\lambda/2} = \frac{\lambda}{2}\cot\frac{\lambda}{2} - i\frac{\lambda}{2}. \tag{15.8}$$

  That is, the real part is equal to $\frac{\lambda}{2}\cot\frac{\lambda}{2}$.
- For angular groups we should throttle the angular part of $\|e_X\|$ at $\pm\pi/2$ in order to avoid the region where the eigenvalues approach zero which otherwise would lead to sluggish convergence

The maximal real part for $\lambda \in [-\pi, \pi]$ is attained at $\lambda = 0$ and is equal to 1, as shown in Figure 15.1. Thus, for lie groups s.t. $\mathrm{ad}_{\boldsymbol{a}}$ has purely imaginary eigenvalues in the range $[-\pi, \pi]$ for all $\boldsymbol{a}$, it holds that $\left(\mathrm{d}^l \exp_{e_X}\right)^{-1}$ has no eigenvalue with real absolute magnitude larger than 1.

Let $\epsilon = \lambda_{\max}\left(\left(\mathrm{d}^l \exp_{e_X}\right)^{-1} - I\right)$; then we have

$$\dot{V} \leq -\begin{bmatrix}\|e_X\| \\ \|e_v\|\end{bmatrix}^T \begin{bmatrix}ck_p(1-\epsilon) & -\frac{ck_d}{2} \\ -\frac{ck_d}{2} & k_d - c\end{bmatrix}\begin{bmatrix}\|e_X\| \\ \|e_v\|\end{bmatrix} \tag{15.9}$$

Therefore, if

$$\begin{aligned}
ck_p(1-\epsilon) + k_d - c &\geq 0 \\
ck_p(1-\epsilon) - \frac{c^2k_d^2}{4} &\geq 0
\end{aligned} \tag{15.10}$$

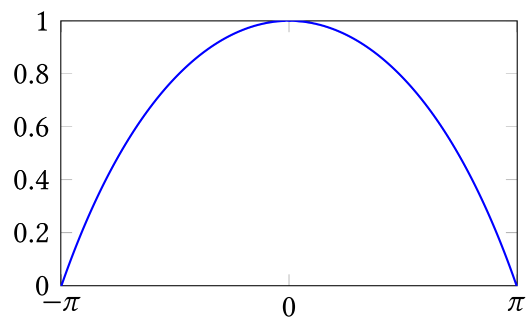


Figure 15.1.: Function $x \mapsto \frac{x}{2} \cot \frac{x}{2}$.

In the following we let $M = SO(3)$ be the Lie group consisting of rotation matrices with matrix multiplication being the group action. We also write $e = I_3$ for the identity element of the group.

Consider the rigid body dynamics

$$\dot{R} = R\,{}^R\hat{\boldsymbol{\omega}}, \tag{15.11a}$$

$$J\,{}^R\dot{\boldsymbol{\omega}} = -{}^R\hat{\boldsymbol{\omega}} J {}^R\boldsymbol{\omega} + u, \tag{15.11b}$$

where J is the moment of inertia, ${}^R\hat{\boldsymbol{\omega}} \in TM_R$ is the angular velocity in the body frame, and $R \in SO(3)$ is the attitude. We can see that the angular velocity ${}^e\hat{\boldsymbol{\omega}} \in TM_e$ in the inertial frame can be obtained as

$$^e\boldsymbol{\omega} = \mathrm{Ad}_R\left({}^R\boldsymbol{\omega}\right) = R^R\boldsymbol{\omega}. \tag{15.12}$$

We also see that ${}^e\hat{\boldsymbol{\omega}} = \widehat{\mathrm{Ad}_R^R\,\boldsymbol{\omega}} = R\,{}^R\boldsymbol{\omega}\,R^T$, so it follows that (15.11a) can be written as

$$\dot{R} = {}^e\hat{\boldsymbol{\omega}}R. \tag{15.13}$$

We assume that a smooth trajectory in the inertial frame is given by R_d and ${}^e\boldsymbol{\omega}_d$ satisfying the dynamics

$$\dot{R}_d = {}^e\hat{\boldsymbol{\omega}}_d R_d, \tag{15.14}$$

and the goal is to control u in (15.11) so that R and ${}^e\boldsymbol{\omega}$ are close to R_d and ${}^e\boldsymbol{\omega}_d$.

15.2. Error Functions

In general we would like to pick for $\tilde{e}_r = R_d \ominus R$ the error function $\frac{1}{2}\|\tilde{e}_r\|^2$ with derivative $\langle \tilde{e}_r, \tilde{e}_\omega \rangle$ for

$$\tilde{e}_\omega = \dot{\tilde{e}}_r = J_{R_d}^{R_d \ominus R}\,{}^{R_d}\boldsymbol{\omega}_d + J_R^{R_d \ominus R}\,{}^R\boldsymbol{\omega}. \tag{15.15}$$

This is general for any Lie group, and we can pick u to stabilize a double integrator system in the tangent space. However, the derivative of $\tilde{e}_\omega$ is cumbersome to evaluate and it is possible to arrive at a simpler formulation in $SO(3)$. Consider the error functions

$$\Psi(R, R_d) = 1 - \cos(\theta) = \frac{1 - \mathrm{Tr}(RR_d^T)}{2} = -\frac{1 - \langle R_d, R\rangle_F}{2}, \tag{15.16a}$$

$$e_r = \frac{1}{2}(R_d^T R - R^T R_d)^\vee, \tag{15.16b}$$

$$e_\omega = \boldsymbol{\omega} - R^T \boldsymbol{\omega}^d \in TSO(3)_R. \tag{15.16c}$$

It can be seen by (11.29) that e_r is a rescaling of $\tilde{e}_r$. The derivative of Ψ is $\langle e_r, e_\omega\rangle$ as above, indeed

$$\begin{aligned}
\dot{\Psi} = &-\frac{1}{2}\left(\left\langle R_d, \dot{R}\right\rangle_F + \left\langle \dot{R}_d, R\right\rangle_F\right) = -\frac{1}{2}\left(\langle R_d, R\hat{\boldsymbol{\omega}}\rangle_F + \langle \hat{\boldsymbol{\omega}}_d R_d, R\rangle_F\right) = \\
&= -\frac{1}{2}\left(\left\langle R^T R_d, \hat{\boldsymbol{\omega}}\right\rangle_F - \left\langle \hat{\boldsymbol{\omega}}_d^T R_d, R\right\rangle_F\right) = -\frac{1}{2}\left(\left\langle R^T R_d, \hat{\boldsymbol{\omega}}\right\rangle_F - \langle R_d, \hat{\boldsymbol{\omega}}_d R\rangle_F\right) \\
&= -\frac{1}{2}\left(\left\langle R^T R_d, \hat{\boldsymbol{\omega}}\right\rangle_F - \left\langle R_d, R\widehat{R^T \boldsymbol{\omega}_d}\right\rangle_F\right) = -\frac{1}{2}\left\langle R^T R_d, \hat{e}_\omega\right\rangle_F \\
&= \frac{1}{4}\left\langle R_d^T R - R^T R_d, \hat{e}_\omega\right\rangle_F = e_r \cdot e_\omega,
\end{aligned} \tag{15.17}$$

where we have used the property that the Frobenius product $\langle A, B\rangle_F = -\left\langle A^T, B\right\rangle_F$ for B skew-symmetric.

Do derivatives via jacobians instead

15.3. Lyapunov Stability

We let the input be

$$u = -k_r e_r - k_\omega e_\omega + \widehat{R^T \boldsymbol{\omega}_d} J R^T \boldsymbol{\omega}_d + J R^T \dot{\boldsymbol{\omega}}_d. \tag{15.18}$$

and consider a Lyapunov candidate on the form

$$V = \frac{1}{2} e_\omega \cdot J e_\omega + k_r \Psi + c e_r \cdot J e_\omega \tag{15.19}$$

The derivative of the Lyapunov candidate then

Proposition 15.1. *It holds that*

$$J\dot{e}_\omega = -k_r e_r - k_\omega e_\omega + \left(J e_\omega + \left(2 J R^T \boldsymbol{\omega}_d - \mathrm{trace}(J) I\right) R^T \boldsymbol{\omega}_d\right) \times e_\omega. \tag{15.20}$$

Proof.

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t} J e_\omega &\overset{(15.28b)}{=} J\dot{\omega} - J\dot{R}^T\omega_d - JR^T\dot{\omega}_d \overset{(15.11)}{=} u - \hat{\omega}J\omega - J\left(R\hat{\omega}\right)^T\omega_d - JR^T\dot{\omega}_d \\
&\overset{(15.29)}{=} -k_r e_r - k_\omega e_\omega + \widehat{R^T\,\omega_d} JR^T\,\omega_d - \hat{\omega}J\omega - J\hat{\omega}^T R^T\omega_d \\
&\overset{(15.28b)}{=} -k_r e_r - k_\omega e_\omega + \widehat{R^T\,\omega_d} JR^T\,\omega_d - (\hat{e}_\omega + \widehat{R^T\omega_d})J(e_\omega + R^T\omega_d) \\
&\quad + J\left(\hat{e}_\omega + \cancelto{0}{\widehat{R^T\omega_d}}\right)R^T\omega_d \\
&= -k_r e_r - k_\omega e_\omega + \left(\widehat{Je_\omega} + \widehat{JR^T\omega_d} - \widehat{R^T\omega_d}J - J\widehat{R^T\omega_d}\right)e_\omega \\
&= -k_r e_r - k_\omega e_\omega + \left(Je_\omega + \left(2JR^T\omega_d - \operatorname{trace}(J)I\right)R^T\omega_d\right)^\wedge e_\omega.
\end{aligned}
$$

□

We then get

$$\dot{V} = -k_\omega\|e_\omega\|^2 + c\dot{e}_r\cdot Je_\omega + ce_r\cdot J\dot{e}_\omega \tag{15.21}$$

It remains to bound the terms involving c. We have that $\|\omega\|_2^2 = \frac{1}{2}\|\hat{\omega}\|_F^2$. We also have

$$\frac{\mathrm{d}}{\mathrm{d}t}R_d^T R = R_d^T R\hat{\omega} + R_d^T\hat{\omega}_d^T R = R_d^T R\hat{\omega} - R_d^T\hat{\omega}_d R = R_d^T R\hat{\omega} - R_d^T R\,\widehat{R^T\omega_d} = R_d^T R\hat{e}_\omega. \tag{15.22}$$

and therefore we get that $\|\dot{\hat{e}}_r\|_F = \left\|\frac{1}{2}\left(R_d^T R\hat{e}_\omega + \hat{e}_\omega R^T R_d\right)\right\|_F \le \|\hat{e}_\omega\|_F$, so it follows that

$$\|\dot{e}_r\|_2 \le \|e_\omega\|_2 \qquad \Longrightarrow\quad \dot{e}_r\cdot Je_\omega \le \lambda_M(J)\|e_\omega\|_2^2. \tag{15.23}$$

Finally, using that $\|e_r\| \le 1$,

$$
\begin{aligned}
J\dot{e}_\omega\cdot e_r &\overset{(15.20)}{=} \left(-k_r e_r - k_\omega e_\omega + \left(Je_\omega + \left(2JR^T\omega_d - \operatorname{trace}(J)I\right)R^T\omega_d\right)\times e_\omega\right)\cdot e_r \\
&\le -k_r\|e_r\|^2 + k_\omega\|e_r\|\|e_\omega\| + \lambda_M(J)\|e_\omega\|^2 + B\|e_\omega\|\|e_r\|.
\end{aligned} \tag{15.24}
$$

We can now bound the derivative as follows:

$$\dot{V} \le -\begin{bmatrix}\|e_r\|\\ \|e_\omega\|\end{bmatrix}^T\begin{bmatrix} ck_r & -c(k_\omega+B)/2 \\ -c(k_\omega+B)/2 & k_\omega - 2c\lambda_M(J)\end{bmatrix}\begin{bmatrix}\|e_r\|\\ \|e_\omega\|\end{bmatrix}, \tag{15.25}$$

and it follows that if we choose c small enough then the matrix is positive definite and thus V decreases along trajectories of the closed-loop system.

15.4. Direction-driven Attitude Control on SO(3)

We pick two orthogonal unit-length directions b_1 and b_2 and define the following error function:

$$\Psi_i(R) = \frac{1}{2}\left\|Rb_i - R_d b_i\right\|^2 = 1 - (Rb_i)\cdot(R_d b_i). \tag{15.26}$$

The derivative of $\Psi_i(R)$ becomes

$$\begin{aligned}
\dot{\Psi}_i(R) &= -\dot{R}b_i \cdot R_d b_i - Rb_i \cdot \dot{R}_d b_i \overset{(15.11a),(15.14)}{=} -R^R\hat{\boldsymbol{\omega}}b_i \cdot R_d b_i - Rb_i \cdot {}^e\hat{\boldsymbol{\omega}}_d R_d b_i \\
&= -{}^R\hat{\boldsymbol{\omega}}b_i \cdot R^T R_d b_i - b_i \cdot R^T\, {}^e\hat{\boldsymbol{\omega}}_d R_d b_i = -{}^R\hat{\boldsymbol{\omega}}b_i \cdot R^T R_d b_i - b_i \cdot \widehat{R^T\, {}^e\boldsymbol{\omega}_d} R^T R_d b_i \\
&= -{}^R\boldsymbol{\omega} \cdot \left(\widehat{b_i}\, R^T R_d b_i\right) - R^T\, {}^e\boldsymbol{\omega}_d \cdot \widehat{R^T R_d b_i} b_i = \underbrace{\left({}^R\boldsymbol{\omega} - R^T\, {}^e\boldsymbol{\omega}_d\right)}_{e_\omega} \cdot \underbrace{\widehat{R^T R_d b_i} b_i}_{e_{r_i}},
\end{aligned} \tag{15.27}$$

where we have defined two error functions

$$e_{r_i} = \widehat{R^T R_d b_i} b_i, \tag{15.28a}$$

$$e_\omega = {}^R\boldsymbol{\omega} - R^T\, {}^e\boldsymbol{\omega}_d, \tag{15.28b}$$

that are small when $R \approx R_d$ and when $\mathrm{Ad}_R\, {}^R\hat{\boldsymbol{\omega}} = R^R\hat{\boldsymbol{\omega}} \approx {}^e\boldsymbol{\omega}_d$, respectively.

15.5. Feedback Control

Given these error functions we consider the feedback control

$$u = -e_r - k_\omega e_\omega + \widehat{R^T\, {}^e\boldsymbol{\omega}_d} J R^T\, {}^e\boldsymbol{\omega}_d + J R^{T\,e}\dot{\boldsymbol{\omega}}_d, \tag{15.29}$$

where

$$e_r = k_1 e_{r_1} + k_2 e_{r_2}, \tag{15.30}$$

and k_1, k_2, k_ω are positive gains. Take the candidate Lyapunov function

$$V = \frac{1}{2} e_\omega \cdot J e_\omega + k_1 \Psi_1(R) + k_2 \Psi_2(R) + c J e_\omega \cdot e_r. \tag{15.31}$$

In the following we drop the upper left superscripts and write $\boldsymbol{\omega} = {}^R\boldsymbol{\omega}$ and $\boldsymbol{\omega}_d = {}^e\boldsymbol{\omega}_d$.

15.6. Lyapunov lower bound

We would like to show that $V = 0$ implies that $\|e_r\|$ and $\|e_\omega\|$ are zero. The main challenge lies in bounding the terms containing Ψ_i. Note that

$$\|e_{r_i}\| = \|\widehat{R^T R_d b_i} b_i\| = \|R^T R_d b_i \times b_i\| = \sin\theta_i, \tag{15.32}$$

where θ_i is the angle between $R^T R_d b_i$ and b_i. Note that θ_i is always in the range $[0, \pi]$. Similarly,

$$\Psi_i(R) = 1 - R^T R_d b_i \cdot b_i = 1 - \cos\theta_i. \tag{15.33}$$

Utilizing this and $(a+b)^2 \leq 2(a^2+b^2)$ we get:

$$\begin{aligned}
\|e_r\|^2 &\overset{(15.30)}{=} \|k_1 e_{r_1} + k_2 e_{r_2}\|^2 \leq (k_1\|e_{r_1}\| + k_2\|e_{r_2}\|)^2 \overset{(15.32)}{=} (k_1 \sin\theta_1 + k_2 \sin\theta_2)^2 \\
&= \left(k_1\sqrt{1-\cos^2\theta_1} + k_2\sqrt{1-\cos^2\theta_2}\right)^2 \leq \left(k_1\sqrt{2(1-\cos\theta_1)} + k_2\sqrt{2(1-\cos\theta_2)}\right)^2 \\
&\overset{(15.33)}{=} 2\left(k_1\sqrt{\Psi_1(R)} + k_2\sqrt{\Psi_2(R)}\right)^2 \leq 4\min(k_1,k_2)\,(k_1\Psi_1(R) + k_2\Psi_2(R))\,.
\end{aligned}$$

We therefore get

$$V \geq \frac{1}{2}\begin{bmatrix}\|e_r\| \\ \|e_\omega\|\end{bmatrix}^T \begin{bmatrix}\frac{1}{2\min(k_1,k_2)} & -c\lambda_M(J) \\ -c\lambda_M(J) & \lambda_m(J)\end{bmatrix}\begin{bmatrix}\|e_r\| \\ \|e_\omega\|\end{bmatrix} \tag{15.34}$$

where the matrix is positive definite for small enough c.

15.7. Lyapunov derivative

We start with an intermediate result

Proposition 15.2. *It holds that*

$$J\dot{e}_\omega = -e_r - k_\omega e_\omega + \left(Je_\omega + \left(2JR^T\omega_d - \operatorname{trace}(J)I\right)R^T\omega_d\right) \times e_\omega. \tag{15.35}$$

Proof.

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}Je_\omega &\overset{(15.28b)}{=} J\dot{\omega} - J\dot{R}^T\omega_d - JR^T\dot{\omega}_d \overset{(15.11)}{=} u - \hat{\omega}J\omega - J\left(R\hat{\omega}\right)^T\omega_d - JR^T\dot{\omega}_d \\
&\overset{(15.29)}{=} -e_r - k_\omega e_\omega + \widehat{R^T\omega_d}JR^T\omega_d - \hat{\omega}J\omega - J\hat{\omega}^T R^T\omega_d \\
&\overset{(15.28b)}{=} -e_r - k_\omega e_\omega + \widehat{R^T\omega_d}JR^T\omega_d - (\hat{e}_\omega + \widehat{R^T\omega_d})J(e_\omega + R^T\omega_d) \\
&\quad + J\left(\hat{e}_\omega + \cancelto{0}{\widehat{R^T\omega_d}}\right)R^T\omega_d \\
&= -e_r - k_\omega e_\omega + \left(\widehat{Je_\omega} + \widehat{JR^T\omega_d} - \widehat{R^T\omega_d}J - J\widehat{R^T\omega_d}\right)e_\omega \\
&= -e_r - k_\omega e_\omega + \left(Je_\omega + \left(2JR^T\omega_d - \operatorname{trace}(J)I\right)R^T\omega_d\right)^\wedge e_\omega.
\end{aligned}$$

□

Thus the derivative of V is

$$\dot{V} \overset{(15.27)}{=} e_\omega \cdot J\dot{e}_\omega + e_r \cdot e_\omega + cJ\dot{e}_\omega \cdot e_r + cJe_\omega \cdot \dot{e}_r \overset{(15.35)}{=} -k_\omega\|e_\omega\|^2 + cJ\dot{e}_\omega \cdot e_r + cJe_\omega \cdot \dot{e}_r, \tag{15.36}$$

so we would like to bound $J\dot{e}_\omega \cdot e_r$ and $Je_\omega \cdot \dot{e}_r$ in terms of $\|e_\omega\|$ and $\|e_r\|$. First we have

$$\frac{\mathrm{d}}{\mathrm{d}t}R_d^T R = R_d^T R\hat{\omega} + R_d^T\hat{\omega}_d^T R = R_d^T R\hat{\omega} - R_d^T\hat{\omega}_d R = R_d^T R\hat{\omega} - R_d^T R\,\widehat{R^T\omega_d} = R_d^T R\hat{e}_\omega. \tag{15.37}$$

Now, $e_{r_i} = \widehat{R^T R_d b_i} b_i$, so by linearity of the hat mapping and that $\|\hat{b}_i\| = \|b_i\| = 1$ it follows that

$$\dot{e}_{r_i} = \widehat{R_d^T R \hat{e}_\omega b_i}\; b_i = -\widehat{R_d^T R \hat{b}_i e_\omega}\; b_i, \quad \implies \|\dot{e}_{r_i}\| \leq \|R_d^T R\| \|\hat{b}_i\| \|e_\omega\| \|b_i\| = \|e_\omega\|. \tag{15.38}$$

Thus, for $\lambda_M(J)$ the maximal eigenvalue of J,

$$\|J e_\omega \cdot \dot{e}_r\| \leq \lambda_M(J)(k_1 + k_2)\|e_\omega\|^2. \tag{15.39}$$

Finally, we bound the last term, utilizing that $\|e_r\| \leq k_1 + k_2$:

$$\begin{aligned} J\dot{e}_\omega \cdot e_r &\overset{(15.35)}{=} \left(-e_r - k_\omega e_\omega + \left(J e_\omega + \left(2JR^T \boldsymbol{\omega}_d - \operatorname{trace}(J) I\right) R^T \boldsymbol{\omega}_d\right) \times e_\omega\right) \cdot e_r \\ &\leq -\|e_r\|^2 + k_\omega \|e_r\| \|e_\omega\| + \lambda_M(J)(k_1 + k_2)\|e_\omega\|^2 + B\|e_\omega\| \|e_r\|, \end{aligned} \tag{15.40}$$

where B is some number that upper bounds $\|\left(2JR^T \boldsymbol{\omega}_d - \operatorname{trace}(J) I\right) R^T \boldsymbol{\omega}_d\|$.

We can now bound the derivative as follows:

$$\dot{V} \leq -\begin{bmatrix} \|e_r\| \\ \|e_\omega\| \end{bmatrix}^T \begin{bmatrix} c & -c(k_\omega + B)/2 \\ -c(k_\omega + B)/2 & k_\omega - 2c\lambda_M(J)(k_1 + k_2) \end{bmatrix} \begin{bmatrix} \|e_r\| \\ \|e_\omega\| \end{bmatrix}, \tag{15.41}$$

and it follows that if we choose c small enough then the matrix is positive definite and thus V decreases along trajectories of the closed-loop system.

Remaining steps:

- Show that undesired equilibria are unstable

16. Application: Model-Predictive Control

Consider a system $\boldsymbol{X}(t)$ evolving on a Matrix Lie group

$$\mathrm{d}^r \boldsymbol{X}_t = f(\boldsymbol{X}, u), \quad \boldsymbol{X} \in \mathbb{M}, \qquad f : \mathbb{M} \times U \to T\mathbb{M}. \tag{16.1}$$

We are interested in finding an approximate solution to the optimal control problem

$$\begin{cases} \min & \int_0^T \left\| \sqrt{Q(\tau)}(\boldsymbol{X}(\tau) \ominus_r \boldsymbol{X}_d(\tau)) \right\|_2^2 + \left\| \sqrt{R(\tau)}(u(\tau) - u_d(\tau)) \right\| \mathrm{d}\tau + \left\| \sqrt{Q(T)}(\boldsymbol{X}(T) \ominus_r x_d(T)) \right\|_2^2 \\ \text{s.t.} & (16.1) \\ & \boldsymbol{X}(0) = \boldsymbol{X}_0 \end{cases}, \tag{16.2}$$

for positive semi-definite matrices Q and R.

We start by considering the dynamics around a nominal trajectory $(\boldsymbol{X}_l(t), u_l(t))$. Consider the error $\boldsymbol{a}_e = \boldsymbol{X}(t) \ominus_r \boldsymbol{X}_l(t)$. Since the error takes values in $T_{\boldsymbol{X}_l(t)}\mathbb{M} \cong \mathbb{R}^n$ the rule of total derivatives in Remark 5.1 applies and the error dynamics become

$$\begin{aligned} \frac{\mathrm{d}\boldsymbol{a}_e}{\mathrm{d}t} &= \mathrm{d}^r(\boldsymbol{a}_e)_t = \mathrm{d}^r(\boldsymbol{X} \ominus_r \boldsymbol{X}_l)_{\boldsymbol{X}} \mathrm{d}^r \boldsymbol{X}_t + \mathrm{d}^r(\boldsymbol{X} \ominus_r \boldsymbol{X}_l)_{\boldsymbol{X}_l} \, \mathrm{d}^r(\boldsymbol{X}_l)_t \\ &\overset{(5.47),(5.48)}{=} \left[\mathrm{d}^r \exp_{\boldsymbol{a}_e}\right]^{-1} f\left(\boldsymbol{X}_l \oplus_r \boldsymbol{a}_e, u_l + u_e\right) - \left[\mathrm{d}^l \exp_{\boldsymbol{a}_e}\right]^{-1} \, \mathrm{d}^r(\boldsymbol{X}_l)_t, \end{aligned} \tag{16.3}$$

Thus we can change coordinates and rewrite (16.2) as

$$\begin{cases} \min & \int_0^T \left\| \sqrt{Q(\tau)}\left((\boldsymbol{X}_l(\tau) \oplus_r \boldsymbol{a}_e(\tau)) \ominus_r \boldsymbol{X}_d(\tau)\right) \right\|_2^2 + \left\| \sqrt{R(\tau)}(u_l(\tau) + u_e(\tau) - u_d(\tau)) \right\| \mathrm{d}\tau, \\ \text{s.t.} & (16.3), \\ & \boldsymbol{a}_e(0) = \boldsymbol{X}_0 \ominus \boldsymbol{X}_l(0). \end{cases}, \tag{16.4}$$

This is now a regular optimal control problem and we can proceed by linearizing around $(\boldsymbol{a}_e, u_e) = (0, 0)$ to obtain the linear time-varying system:

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{a}_e = A(t)\boldsymbol{a}_e + B(t)u_e + E(t), \tag{16.5}$$

where, since $\mathrm{d}^r \exp_0 = \mathrm{d}^l \exp_0 = I$,

$$A(t) := \left.\frac{\mathrm{d}}{\mathrm{d}\boldsymbol{a}_e}\right|_{\boldsymbol{a}_e=0} \left[\mathrm{d}^r \exp_{\boldsymbol{a}_e}\right]^{-1} f\left(\boldsymbol{X}_l(t) \oplus_r \boldsymbol{a}_e, u_l(t)\right), \tag{16.6}$$

$$B(t) := \left.\frac{\mathrm{d}}{\mathrm{d}u_e}\right|_{u_e=0} f\left(\boldsymbol{X}_l(t), u_l(t) + u_e\right), \tag{16.7}$$

$$E(t) := f\left(\boldsymbol{X}_l(t), u_l(t)\right) - \mathrm{d}^r(\boldsymbol{X}_l)_t. \tag{16.8}$$

To facilitate evaluating the cost function we note that

$$(\boldsymbol{X}_l \oplus_r \boldsymbol{a}_e) \ominus_r \boldsymbol{X}_d = \log\left(\boldsymbol{X}_d^{-1} \circ \boldsymbol{X}_l \circ \exp(\boldsymbol{a}_e)\right) = \log(\exp(\boldsymbol{X}_l \ominus_r \boldsymbol{X}_d) \circ \exp(\boldsymbol{a}_e)) \approx \boldsymbol{X}_l \ominus_r \boldsymbol{X}_d + \boldsymbol{a}_e(t), \quad (16.9)$$

where the last approximate step follows from the Baker-Campbell-Hausdorff formula (4.10).

We can thus write it on the form

$$\left\|\sqrt{Q}\left((\boldsymbol{X}_l \oplus_r \boldsymbol{a}_e) \ominus_r \boldsymbol{X}_d\right)\right\|_2^2 \approx (\boldsymbol{X}_l \ominus_r \boldsymbol{X}_d + \boldsymbol{a}_e)^T Q\,(\boldsymbol{X}_l \ominus_r \boldsymbol{X}_d + \boldsymbol{a}_e) = \boldsymbol{a}_e^T Q \boldsymbol{a}_e + 2\,(\boldsymbol{X}_l \ominus_r \boldsymbol{X}_d)^T Q \boldsymbol{a}_e. \quad (16.10)$$

17. Application: State Estimation

17.1. IMU Model

An imu typically consists of a gyro measuring angular velocity, an accelerometer, and a magnetometer that estimates the orientation with respect to the earth magnetic field. Consider a body moving in the world described by the IMU frame to world frame transform $P_{WI} = (\boldsymbol{q}_{WI}, \boldsymbol{p}_{WI}) \in \mathbb{SE}(3)$.

The gyro and accelerometer measurements of the IMU are then well modeled by the following:

$$\tilde{\boldsymbol{\omega}} = \mathrm{d}^r(\boldsymbol{q}_{WI})_t + \boldsymbol{b}_{\boldsymbol{\omega}} + \eta_{\boldsymbol{\omega}}, \tag{17.1a}$$

$$\tilde{\boldsymbol{a}} = \left[\mathrm{d}^{2r} P_{WI}\right]_{3:6} + \boldsymbol{q}_{WI}^{-1} \boldsymbol{g}_W + \boldsymbol{b}_{\boldsymbol{a}} + \eta_{\boldsymbol{a}}. \tag{17.1b}$$

In (17.1a), the first term is the actual body angular velocity, $\boldsymbol{b}_{\boldsymbol{\omega}}$ is a gyro bias, and $\eta_{\boldsymbol{\omega}}$ is white noise.

For the accelerometer model (17.1b), $\left[\mathrm{d}^{2r}(P_{WI})_t\right]_{3:6}$ denotes the linear acceleration in the body frame (the last three components of the second derivative), $\boldsymbol{g}_W$ is the gravity in the world frame, and $\boldsymbol{b}_{\boldsymbol{a}}$ and $\eta_{\boldsymbol{a}}$ are bias and noise as above.

17.2. Complementary Filter for Attitude Estimation

Filter in [14] is of form

$$\dot{\hat{R}} = (\hat{R}\Omega + k_P \hat{R}\omega)\hat{R}, \quad \omega = vex\left(\frac{1}{2}\left(\tilde{R} - \tilde{R}^T\right)\right), \quad \tilde{R} = \hat{R}R_y.$$

That is, the natural dynamics are amended with a term $k_P \hat{R}\omega$ that induces stability of the observer. The quantitiy ω is a rotation quantity in body coordinates that corresponds to the anti-symmetric part of the empirical rotation error. It can be shown that for $R_y = R$ we have

$$\frac{d}{dt}\frac{1}{2}\mathrm{Tr}\left(I - \hat{R}^T R\right) = -\frac{k_P}{2}|\omega|^2.$$

Remark: Filter above is the *passive* form, if $R_y\Omega$ is used instead of $\hat{R}\Omega$ it is called *direct*.

This can be amended with an integrator that estimates a bias term in the gyro estimates.

17.3. TODO

1. Convert the attitude estimation filter from $SO(3)$ to S^3.

18. Nonlinear Least Squares

Like how Lie groups thread the line between linear and nonlinear manifolds, the same can be said for the role of nonlinear least squares in optimization, which is a type of optimization problem is rich enough for to model a wide variety of situations, yet structured enough to be amenable to practical algorithms.

A non-linear least squares problem has the general form

$$\min_{\boldsymbol{x}\in\mathbb{M}} \frac{1}{2}\sum_{i=1}^{N} \|r_i(\boldsymbol{x})\|^2, \quad r_i : \mathscr{M} \to \mathbb{R}^{n_{r_i}}. \tag{18.1}$$

The manifold $\mathbb{M}$ can be a Lie group or a Lie group product $\boldsymbol{x} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_k) \in \mathbb{M}_1 \times \dots \times \mathbb{M}_k$. For the latter case, typically not every residual depends on each member of the bundle, i.e. $r_i(\boldsymbol{x}) = r_i\left(\{\boldsymbol{x}_j\}_{j\in I_i}\right)$ where $I_i \subset \{1, \dots, k\}$ is a subset of variables.

Remark 18.1. *An equivalent problem with a single residual is* $\min_{\boldsymbol{x}\in\mathbb{M}} \frac{1}{2}\|r(\boldsymbol{x})\|^2$ *for*

$$r(\boldsymbol{x}) = \begin{bmatrix} r_1(\boldsymbol{x}) \\ \vdots \\ r_k(\boldsymbol{x}) \end{bmatrix}. \tag{18.2}$$

Although the single residual formulation simplifies notation somewhat, in practice it is for large problems important to leverage the sparsity structure which is better exposed in (18.1).

18.1. Solution Sensitivity

In many applications the residuals $r_i(\boldsymbol{x})$ are obtained from data and are therefore associated with uncertainty. In this situation it is natural to ask how sensitive the optimal solution of the nonlinear least squares problem is to noise in the data. Assume that the noise associated with each residual is Gaussian and independent of other residuals, i.e. that

$$r_i(\boldsymbol{x}) \sim \mathcal{N}\left(\bar{r}_i(\boldsymbol{x}), I\right), \tag{18.3}$$

and consider a point $\bar{\boldsymbol{x}}$. We expand the objective using a Taylor approximation as

$$\min_{\bar{\boldsymbol{x}}\in\mathbb{M}} \frac{1}{2}\sum_{i=1}^{N} \|r_i(\bar{\boldsymbol{x}})\|^2 \approx \min_{\boldsymbol{a}\in T_{\bar{\boldsymbol{x}}}\mathbb{M}} \frac{1}{2}\sum_{i=1}^{N} \|r_i(\bar{\boldsymbol{x}}) + \mathrm{d}^r(r_i)_{\bar{\boldsymbol{x}}}\boldsymbol{a}\|^2. \tag{18.4}$$

The optimal solution $\boldsymbol{x}^*$ of the left problem can be approximately retrieved from $\boldsymbol{a}^*$ as $\boldsymbol{x}^* = \bar{\boldsymbol{x}} \oplus_r \boldsymbol{a}^*$ assuming that $\boldsymbol{a}^*$ is small.

Letting $r_i := r_i(\bar{x})$ and $J_i := \mathrm{d}^r(r_i)_{\bar{x}}$ expanding the square and ignoring the constant term yields

$$\min_{\boldsymbol{a}\in T_{\bar{x}}\mathbb{M}} \sum_{i=1}^{N} \frac{1}{2}\boldsymbol{a}^T J_i^T J_i \boldsymbol{a} + \boldsymbol{a}^T J_i^T r_i = \min_{\boldsymbol{a}\in T_{\bar{x}}\mathbb{M}} \frac{1}{2}\boldsymbol{a}^T \left(\sum_{i=1}^{N} J_i^T J_i\right) \boldsymbol{a} + \boldsymbol{a}^T \sum_{i=1}^{N} J_i^T r_i. \tag{18.5}$$

The optimal solution of this problem can be obtained by setting the gradient w.r.t. $\boldsymbol{a}$ to zero and is

$$\boldsymbol{a}^* = -\left(\sum_{i=1}^{N} J_i^T J_i\right)^{\dagger} \left(\sum_{i=1}^{N} J_i^T r_i\right). \tag{18.6}$$

From this we can infer the sensitivity of $\boldsymbol{a}^*$ to noise in r_i: recalling that $\mathrm{Var}(Ax + By) = A\mathrm{Var}(x)A^T + B\mathrm{Var}(y)B^T$ we get that

$$\boldsymbol{a}^* \sim \mathcal{N}\left(-\left(\sum_{i=1}^{N} J_i^T J_i\right)^{\dagger} \left(\sum_{i=1}^{N} J_i^T \bar{r}_i\right), \left(\sum_{i=1}^{N} J_i^T J_i\right)^{\dagger} \left(\sum_{i=1}^{N} J_i^T \Sigma_i J_i\right)\left(\sum_{i=1}^{N} J_i^T J_i\right)^{\dagger}\right). \tag{18.7}$$

For the special case when all r_i's have unit covariance, i.e. $\Sigma_i = I$, the expression simplifies to

$$\boldsymbol{a}^* \sim \mathcal{N}\left(-\left(\sum_{i=1}^{N} J_i^T J_i\right)^{\dagger} \left(\sum_{i=1}^{N} J_i^T \bar{r}_i\right), \left(\sum_{i=1}^{N} J_i^T J_i\right)^{\dagger}\right). \tag{18.8}$$

Given a residual $r(x) \sim \mathcal{N}(\bar{r}(x), \Sigma)$ a residual with unit covariance can be obtained by left-multiplying with the square root information matrix $\sqrt{I} := \Sigma^{-1/2}$:

$$\sqrt{I}r(x) \sim \mathcal{N}\left(\sqrt{I}\bar{r}(x), I\right). \tag{18.9}$$

Scaling with $\sqrt{I}$ makes sense in many applications since it in effect scales the residual by the inverse noise magnitude.

For the unit covariance case $\Sigma_i = I$ the tangent space covariance of the optimal solution x^* is

$$\left(\sum_{i=1}^{N} \left(\mathrm{d}^r(r_i)_{x^*}\right)^T \mathrm{d}^r(r_i)_{x^*}\right)^{\dagger}. \tag{18.10}$$

18.2. Levenberg-Marquardt

Resources

- Original MINPACK manual: `https://www.netlib.org/minpack/`

LM Implementations

- Original MINPACK in fortran
- cminpack (ported from fortran) https://devernay.github.io/cminpack/
- Eigen unsupported (ported from cminpack)

Consider a Lie group optimization problem

$$\min_{\boldsymbol{x}} \frac{1}{2} \|f(\boldsymbol{x})\|^2, \quad f : \mathbb{M} \to \mathbb{R}^m. \tag{18.11}$$

We are interested in devising an iterative algorithm for minimizing this function.

Given a point $\boldsymbol{x}$ we can solve a local optimization problem to find step $\boldsymbol{a} \in T\mathbb{M}_{\boldsymbol{x}}$ that leads to an improved estimate $\boldsymbol{x} \oplus_r \boldsymbol{a}$. The optimization problem can be re-formulated in terms of $\boldsymbol{a}$ as

$$\arg\min_{\boldsymbol{a}} \|f(\boldsymbol{x} \oplus_r \boldsymbol{a})\|^2. \tag{18.12}$$

Since the problem is nonlinear we resort to linearization. To avoid stepping outside the region where the linearization is accurate we also limit the stepsize and obtain the new problem

The diagonal scaling matrix $D = \mathrm{Diag}(d_1, \ldots, d_n)$ is typically chosen so that a component d_i is inversely proportional to the magnutide of the gradient in that direction, which has the effect of allowing larger steps in directions with low gradient. Common choices include $D = \sqrt{\mathrm{Diag}\left(\mathrm{diag}(J^T J)\right)}$ and $d_i = \|[\mathrm{d}^r f_{\boldsymbol{x}}]_{\cdot,i}\|$—the norm of the i:th column of the jacobian.

A complete Levenberg-Marquardt procedure is shown in Algorithm 1. The crucial step occurs on line 3 and is discussed further below.

Calculation of the actual to predicted reduction ratio can be rewritten as

$$\rho = \frac{1 - \left(\frac{\|f(\boldsymbol{x} \oplus \boldsymbol{a}_{\mathrm{LM}})\|}{\|r\|}\right)^2}{\left(\frac{\|J\boldsymbol{a}\|}{\|r\|}\right)^2 + 2\left(\frac{\sqrt{\lambda}\|D\boldsymbol{a}\|}{\|r\|}\right)^2} \tag{18.13}$$

where we have used that $\|r\|^2 - \|r + J\boldsymbol{a}\|^2 = -2\boldsymbol{a}^T Jr - \boldsymbol{a}^T J^T J\boldsymbol{a} = \|J\boldsymbol{a}\|^2 + 2\lambda\|D\boldsymbol{a}\|^2$ which is a consequence of (18.16b). This formulation has the benefit of avoiding subtraction of numbers of large magnitude which may cause floating point roundoff errors.

18.2.1. Trust-Region Problem

We discuss how to solve a trust-region problem on the form

$$\arg\min_{\boldsymbol{a} : \|D\boldsymbol{a}\| \leq \Delta} \|J\boldsymbol{a} + r\|^2, \tag{18.14}$$

where D is a diagonal scaling matrix and Δ is a maximal step size. This constrained problem can be transformed into an unconstrained problem.

Algorithm 1: One iteration of the Levenberg-Marquardt algorithm.

Data: Iteration variables: point $\boldsymbol{x}^k$, trust region Δ^k, scaling parameters d_i^k as diagonal matrix D^k

Result: Updated iteration variables $\boldsymbol{x}^{k+1}$, Δ^{k+1}, d_i^{k+1}

$r = f(\boldsymbol{x})$

$J = \mathrm{d}^r f_{\boldsymbol{x}}$

$\boldsymbol{a}_{\mathrm{LM}} = \arg\min_{\boldsymbol{a}\,:\,\|D^k \boldsymbol{a}\| \leq \Delta^k} \|r + J\boldsymbol{a}\|^2$ `// calculate increment step`

$\rho = \frac{\|r\|^2 - \|f(\boldsymbol{x} \oplus_r \boldsymbol{a}_{\mathrm{LM}})\|^2}{\|r\|^2 - \|r + J\boldsymbol{a}_{\mathrm{LM}}\|^2}$ `// actual to predicted reduction ratio`

if $\rho \leq 0.25$ **then**

  $\Delta^{k+1} = \Delta^k / 2$ `// decrease trust region`

else if $\rho \geq 0.75$ **then**

  $\Delta^{k+1} = 2\Delta^k$ `// increase trust region`

end

if $\rho \leq 0.0001$ **then**

  $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k$ `// reject step`

else

  $\boldsymbol{x}^{k+1} = \boldsymbol{x}^k \oplus_r \boldsymbol{a}_{\mathrm{LM}}$ `// accept step`

  $d_i^{k+1} = \max\left(d_i^k, \left\|\left[\mathrm{d}^r f_{\boldsymbol{x}^{k+1}}\right]_{\cdot,i}\right\|\right)$ `// update scaling parameters`

end

Theorem 18.1. *A vector $\boldsymbol{a}^*$ is a global minimizer of*

$$\underset{\|D\boldsymbol{a}\|\leq\Delta}{\arg\min}\ \frac{1}{2}\|J\boldsymbol{a}+r\|^2. \tag{18.15}$$

if and only if there exists $\lambda \geq 0$ such that

$$J^TJ+\lambda D^TD \succeq 0, \tag{18.16a}$$

$$(J^TJ+\lambda D^TD)\boldsymbol{a} = -J^Tr, \tag{18.16b}$$

$$\lambda\left(\|D\boldsymbol{a}\|-\Delta\right) = 0. \tag{18.16c}$$

We provide an argument based on duality to support this fact, see e.g. [16, Theorem 4.1] for a more rigorous proof.

Proof. Let the lagrangian of the problem be

$$L(\boldsymbol{a},\lambda) = \frac{1}{2}\|J\boldsymbol{a}+r\|^2 + \frac{\lambda}{2}\left(\|D\boldsymbol{a}\|^2-\Delta^2\right), \tag{18.17}$$

so that the optimization problem (18.14) equivalently can be written $\inf_{\boldsymbol{a}}\sup_{\lambda\geq 0} L(\boldsymbol{a},\lambda)$, since the value of the inner problem is $+\infty$ when the constraint $\|D\boldsymbol{a}\|\leq\Delta$ is not satisfied.

Assuming that strong duality holds, the dual problem $\sup_{\lambda\geq 0}\inf_{\boldsymbol{a}} L(\boldsymbol{a},\lambda)$ has the same optimal value. The inner infimum of the dual problem can be re-written as

$$\inf_{\boldsymbol{a}} L(\boldsymbol{a},\lambda) = \frac{1}{2}\boldsymbol{a}^T(J^TJ+\lambda D^TD)\boldsymbol{a} + r^TJ\boldsymbol{a} - \lambda\frac{\Delta^2}{2}. \tag{18.18}$$

This inner problem has value $-\infty$ unless $J^TJ+\lambda D^TD\succeq 0$, so the outer supremum restricts λ to values that imply positive semi-definiteness. In this case the finite optimal value is attained for $\boldsymbol{a}$ such that $\left(J^TJ+\lambda D^TD\right)\boldsymbol{a} = -J^Tr$ which reduces the dual problem to

$$\sup_{\lambda\geq 0} -\frac{1}{2}\boldsymbol{a}^T\left(J^TJ+\lambda D^TD\right)\boldsymbol{a} - \lambda\frac{\Delta^2}{2}. \tag{18.19}$$

Also this problem has a closed-form solution: either the optimal solution is attained at the boundary, i.e. $\lambda = 0$, or it is attained at zero derivative w.r.t. λ which necessitates $\boldsymbol{a}^TD^TD\boldsymbol{a}+\Delta^2 = 0$. These two latter conditions imply that the complementarity condition $\lambda(\|D\boldsymbol{a}\|-\Delta) = 0$ holds. □

Equation (18.16b) represents the normal equations for the least-squares problem

$$\underset{\boldsymbol{a}}{\arg\min}\ \frac{1}{2}\boldsymbol{a}^T\left(J^TJ+\lambda D^TD\right)\boldsymbol{a} + r^TJ\boldsymbol{a}. \tag{18.20}$$

Equivalently, it can be written on the standard form

$$\underset{\boldsymbol{a}}{\arg\min}\left\|\begin{bmatrix} J \\ \sqrt{\lambda}D \end{bmatrix}\boldsymbol{a} + \begin{bmatrix} r \\ 0 \end{bmatrix}\right\|^2. \tag{18.21}$$

For numerical stability it is preferable to solve a least-squares problem instead of directly solving the normal equations.

Theorem 18.1 suggests that the trust-region problem can be recast as a least squares problem, but doing so requires knowledge of the parameter λ. In the following we show how the least squares problem can be efficiently solved assuming knowledge of λ, and then discuss an algorithm for finding λ.

18.2.2. Solving the Least-Squares Problem

The least-squares problem

$$\arg\min_{\boldsymbol{a}} \left\| \begin{bmatrix} J \\ \sqrt{\lambda} D^k \end{bmatrix} \boldsymbol{a} + \begin{bmatrix} r \\ 0 \end{bmatrix} \right\|^2 \tag{18.22}$$

has structure which can be exploited to find a solution. Consider a QR decomposition with column pivoting of J s.t. $JP = QR$, where $P \in \mathbb{R}^{n\times n}$ is a permutation matrix, $Q \in \mathbb{R}^{n\times n}$ is orthogonal, and $R \in \mathbb{R}^{n\times n}$ is upper-diagonal. If $\boldsymbol{a}$ is a minimizer of (18.22) it is also a minimizer of

$$\arg\min_{\boldsymbol{a}} \left\| \begin{bmatrix} Q^T JP \\ \sqrt{\lambda} P^T D^k P \end{bmatrix} P^T \boldsymbol{a} + \begin{bmatrix} Q^T r \\ 0 \end{bmatrix} \right\|^2 = \arg\min_{\boldsymbol{a}} \left\| \begin{bmatrix} R \\ \sqrt{\lambda} P^T D^k P \end{bmatrix} P^T \boldsymbol{a} + \begin{bmatrix} Q^T r \\ 0 \end{bmatrix} \right\|^2 . \tag{18.23}$$

Consider a second QR decomposition s.t.

$$\begin{bmatrix} R \\ \sqrt{\lambda} P^T D^k P \end{bmatrix} = \tilde{Q} \begin{bmatrix} \tilde{R} \\ 0 \end{bmatrix} \tag{18.24}$$

where $\tilde{Q} = \begin{bmatrix} \tilde{Q}_{11} & \tilde{Q}_{12} \\ \tilde{Q}_{21} & \tilde{Q}_{22} \end{bmatrix} \in \mathbb{R}^{2n\times 2n}$ is orthogonal and $\tilde{R} \in \mathbb{R}^{n\times n}$ is upper-diagonal and has rank n. Since there are only n non-zero variables in the lower triangular part this step can be efficiently computed via $n(n+1)/2$ Givens rotations. In these variables the least-squares problem takes the form

$$\arg\min_{\boldsymbol{a}} \left\| \tilde{Q} \begin{bmatrix} \tilde{R} \\ 0 \end{bmatrix} P^T \boldsymbol{a} + \begin{bmatrix} Q^T r \\ 0 \end{bmatrix} \right\|^2 = \arg\min_{\boldsymbol{a}} \left\| \begin{bmatrix} \tilde{R} \\ 0 \end{bmatrix} P^T \boldsymbol{a} + \tilde{Q}^T \begin{bmatrix} Q^T r \\ 0 \end{bmatrix} \right\|^2 = \arg\min_{\boldsymbol{a}} \left\| \tilde{R} P^T \boldsymbol{a} + \tilde{Q}_{11}^T Q^T r \right\|^2 ,$$

and it is now apparent that the optimal solution is

$$\boldsymbol{a}_{\mathrm{LM}} = -P \tilde{R}^{-1} \tilde{Q}_{11}^T Q^T r. \tag{18.25}$$

When solving (18.22) repeatedly for different values of λ only the second QR decomposition needs to be re-computed.

18.2.3. Finding the LM Parameter

To search for a parameter that satisfies the relations in Theorem 18.1 consider the function

$$\phi(\alpha) = \left\| D \left(J^T J + \alpha D^T D \right)^{-1} J^T r \right\| - \Delta. \tag{18.26}$$

Note that $\phi(\alpha)$ can be evaluated by solving the a structured least-squares problem as discussed in Section 18.2.2. With those variables we have

$$\phi(\alpha) = \|DP\tilde{R}^{-1}\tilde{Q}_{11}^T Q^T r\| - \Delta \tag{18.27}$$

where $\tilde{R}$ and $\tilde{Q}$ depend on α due to (18.24).

We are interested in finding a value of $\alpha > 0$ s.t. $\phi(\alpha) \approx 0$. If $\phi(0) \leq 0$ it must hold that $\lambda = 0$ in Theorem 18.1, so we disregard this case. We follow [15] to construct an algorithm for the case $\phi(0) > 0$.

The function ϕ is strictly decreasing in α, and approximately of the form $\phi(\alpha) \approx \tilde{\phi}(\alpha) = \frac{a}{b+\alpha} - \Delta$. Setting $\tilde{\phi}(\alpha)$ to zero then gives $\alpha = -b + a/\Delta$, and fitting a and b s.t. $\phi(\alpha_k) = \tilde{\phi}(\alpha_k)$ and $\phi'(\alpha_k) = \tilde{\phi}'(\alpha_k)$ gives the update rule

$$\alpha_{k+1} = \alpha_k - \frac{\phi(\alpha_k) + \Delta}{\Delta} \frac{\phi(\alpha_k)}{\phi'(\alpha_k)}. \tag{18.28}$$

To implement this algorithm not only ϕ needs to be calculated, but also its derivative.

Derivative of ϕ Introduce $q(\alpha) = D(J^T J + \alpha D^T D)^{-1} J^T r$ so that $\phi(\alpha) = \sqrt{q(\alpha)^T q(\alpha)} - \Delta$. The derivative of q becomes

$$q'(\alpha) = -D(J^T J + \alpha D^T D)^{-1} D^T D (J^T J + \alpha D^T D)^{-1} J^T r = -D(J^T J + \alpha D^T D) D^T q(\alpha). \tag{18.29}$$

where we have utilized that the derivative of $(A + \alpha B)^{-1}(A + \alpha B)$ is zero which gives

$$\frac{\mathrm{d}}{\mathrm{d}\alpha}(A + \alpha B)^{-1} = -(A + \alpha B)^{-1}) B (A + \alpha B)^{-1}. \tag{18.30}$$

Therefore the derivative of ϕ becomes

$$\phi'(\alpha) = \frac{q(\alpha)^T q'(\alpha)}{\|q(\alpha)\|} = -\frac{D^T q(\alpha))^T (J^T J + \alpha D^T D)^{-1} (D^T q(\alpha))}{\|q(\alpha)\|}. \tag{18.31}$$

When utilizing the method in Section 18.2.2 the same QR decompositions $PJ = QR$ and (18.24) can be leveraged to evaluate the expression efficiently.

$$\begin{aligned} J^T J + \alpha D^T D &= (QRP^T)^T (QRP^T) + \alpha D^T D = PR^T RP^T + \alpha D^T D \\ &= P \begin{bmatrix} R^T & \sqrt{\alpha}(P^T D P)^T \end{bmatrix} \begin{bmatrix} R \\ \sqrt{\alpha} P^T D P \end{bmatrix} P^T \overset{(18.24)}{=} P \tilde{R}^T \tilde{R} P^T. \end{aligned} \tag{18.32}$$

Therefore

$$\phi'(\alpha) = -\frac{\left\|\tilde{R}^{-T} P^T D^T q(\alpha)\right\|^2}{\|q(\alpha)\|} = -\|q(\alpha)\| \left\| \tilde{R}^{-T} \frac{P^T D^T q(\alpha)}{\|q(\alpha)\|} \right\|^2 \tag{18.33}$$

We conclude by writing down the algorithm from [15] for finding λ.

Remark 18.2. *Ceres just uses $\lambda = 1/\Delta$.*

Sparse strategy: as ceres use $\lambda = 1/\Delta$ and use a sparse eigen solver for the step.

Algorithm 2: LM parameter algorithm

Data: Matrices J, D, vector r, scalar Δ

Result: λ s.t. $\phi(\lambda) \leq 0.1$

Calculate QR decomposition $JP = QR$

$l_0 = \begin{cases} -\phi(0)/\phi'(0) & \text{if } J \text{ nonsingular} \\ 0 & \text{otherwise} \end{cases}$

$u_0 = \frac{\|(JD^{-1})^T r\|}{\Delta}$

$\alpha_0 = \sqrt{l_0 u_0}$

repeat

- If $\alpha_k \notin [l_k, u_k]$, set $\alpha_k = \max\left(0.001 u_k, \sqrt{l_k u_k}\right)$
- Calculate QR decomposition $\begin{bmatrix} R \\ \sqrt{\lambda} P^T D^k P \end{bmatrix} = \tilde{Q} \begin{bmatrix} \tilde{R} \\ 0 \end{bmatrix}$
- $z = -P\tilde{R}^{-1}\tilde{Q}_{11}^T Q^T r$
- $\phi = \|Dz\| - \Delta$
- $\phi' = -\|Dz\| \left\| \tilde{R}^{-T} \frac{P^T D^T Dz}{\|Dz\|} \right\|^2$
- $l_{k+1} = \max\left(l_k, \alpha_k - \frac{\phi}{\phi'}\right)$
- $u_{k+1} = \begin{cases} \alpha_k & \text{if } \phi < 0 \\ u_k & \text{otherwise} \end{cases}$
- $\alpha_{k+1} = \alpha_k - \frac{\phi + \Delta}{\Delta} \frac{\phi}{\phi'}$

until $|\phi| \leq 0.1\Delta$

19. Pose Graph Optimization

19.1. Maximum Likelihood Estimation as Nonlinear Least Squares

Notation :

1. $X = \{x_i\}$ set of variables
2. $\hat{y}_j \in \mathbb{R}^{d_j}$
3. h_j measurement function s.t. $y_j \sim \mathcal{N}(h_j(X_j), \Sigma_j)$ for a (often small) subset of variables $X_j \subset X$.

Given a collection of measurements $\hat{y}_j$ we are interested in finding the **maximum-likelihood estimate** of the variables X. In the Gaussian setting this becomes

$$
\begin{aligned}
X^* = \arg\max_x \prod_j p_j(\hat{y}_j \mid X_j) &= \arg\max_x \prod_j \frac{1}{\sqrt{(2\pi)^{d_j}|\Sigma_j|}} \exp\left(-\frac{(\hat{y}_j - h_j(X_j))^T \Sigma_j^{-1}(\hat{y}_j - h_j(X_j))}{2}\right) \\
&= \arg\max_x \prod_j \exp\left(-\frac{(\hat{y}_j - h_j(X_j))^T \Sigma_j^{-1}(\hat{y}_j - h_j(X_j))}{2}\right)
\end{aligned}
$$

Maximizing $f(x)$ is equivalent to minimizing $2\log(-f(x))$. Taking the negative log and multiplying by two yields

$$
X^* = \arg\min_x \sum_j (\hat{y}_j - h_j(X_j))^T \Sigma_j^{-1}(\hat{y}_j - h_j(X_j)) = \arg\min_x \sum_j \left\| \sqrt{I}_j \left(\hat{y}_j - h_j(X_j)\right) \right\|^2,
$$

where $\sqrt{I}_j := \Sigma_j^{-1/2}$ is the **square root information matrix**. We have thus converted the maximum-likelihood estimation problem into a **least squares problem**. When the functions h_j involve 3D geometry the least squares problem is typically **nonlinear**.

The least-squares problem can be viewed as a **bipartite factor graph** where variables and measurements (factors) are nodes. By exploiting the graph structure updates can be made locally in the graph, but this requires sophisticated data structures and solvers [7].

19.2. Measurement functions

19.2.1. Absolute pose measurement

Let the measurement be $\hat{y}_j = \log(\hat{P})$ where $\log : SE(3) \to \mathfrak{se}(3)$ is the logarithm on $SE(3)$ and $\hat{P}$ a pose measurement, then

$$h(P) = \log(P) \in \mathbb{R}^6.$$

Since $SE(3) = SO(3) \times \mathbb{R}^3$ we can use the same formula for individual measurements of orientation ($SO(3)$) or position ($\mathbb{R}^3$).

19.2.2. Relative pose measurement

Let the measurement be $\hat{y}_j = \log(\hat{P}_{12})$ where $\log : SE(3) \to \mathfrak{se}(3)$ is the logarithm on $SE(3)$ and $\hat{P}_{12}$ an estimate of the relative pose. Then

$$h(P_1, P_2) = \log(P_1^{-1} P_2) \in \mathbb{R}^6.$$

19.2.3. Rectified stereo landmark measurement

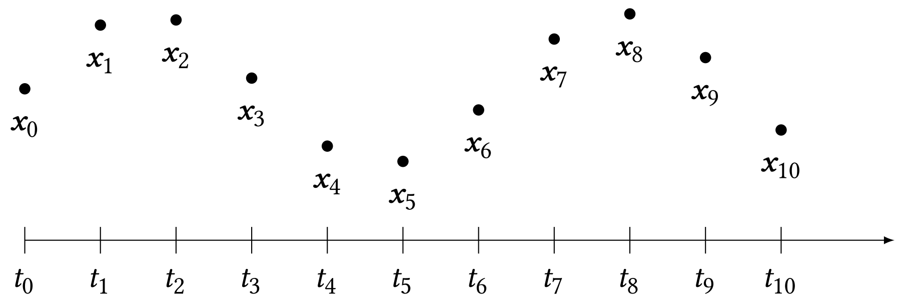
* $P \in SE(3)$ is the pose of the left camera (variable) * $l \in \mathbb{R}^3$ world location of a landmark (variable) * $P_{rl} \in SE(3)$ the pose of the right camera w.r.t. the left camera (known) * CM_l, CM_r camera projection matrices

The landmark is projected to the left and right image pixel planes as

$$\lambda_l \tilde{\mathbf{x}}_l = CM_l P^{-1} l, \quad \lambda_r \tilde{\mathbf{x}}_r = CM_r P_{rl} P^{-1} l.$$

In a rectified system we have $y_l = y_r$, so we can let the 3-dimensional measurement be the pixel locations $\hat{x}_l, \hat{x}_r, \hat{y}_l$. The measurement function $h(P, l)$ is described by the equations above.

20. Splines on Lie Groups



It is often convenient to be able to interpolate sparsely defined data. Examples in robotics include representing trajectories as a sequence of $(t_i, \boldsymbol{x}_i)$ pairs, and interpolating sensor data for calibration.

A general form for a spline $p(x)$ constructed from *control points* $\boldsymbol{x}_i$ is

$$p(t) = \sum_{i=0}^{n} B_i(t)\boldsymbol{x}_i, \tag{20.1}$$

where $\boldsymbol{B}_i$ are some form of *basis functions* that form a partition of unity, i.e. $\sum_i B_i(t) = 1$ for all t. The weighted sum does not have an immediate analogue on Lie groups, but (20.1) can be re-arranged on *cumulative form* as

$$p(t) = \boldsymbol{x}_0 + \sum_{i=1}^{n} \tilde{B}_i(t)(\boldsymbol{x}_i - \boldsymbol{x}_{i-1}), \tag{20.2}$$

where $\tilde{B}_i(t) = \sum_{j=i}^{n} B_i(t)$ are *cumulative basis functions.* From this formulation we can write down a Lie group generalization as follows:

$$p(t) = \boldsymbol{x}_0 \circ \exp\left(\tilde{B}_1(t)(\boldsymbol{x}_1 \ominus \boldsymbol{x}_0)\right) \circ \exp\left(\tilde{B}_2(t)(\boldsymbol{x}_2 \ominus \boldsymbol{x}_1)\right) \circ \ldots \circ \exp\left(\tilde{B}_n(t)(\boldsymbol{x}_n \ominus \boldsymbol{x}_{n-1})\right) \tag{20.3}$$

We first discuss two choices for basis functions—Bezier curves and B-Splines—and then show how the value and derivatives of such splines can be calculated.

Below we will need the derivative with respect to t of (20.3). For this purpose it will be convenient to introduce the variables

$$\begin{aligned} \boldsymbol{v}_i &= \boldsymbol{x}_i \ominus \boldsymbol{x}_{i-1}, \\ \boldsymbol{s}_i &= \tilde{B}_i(t)\boldsymbol{v}_i. \end{aligned} \tag{20.4}$$

With this notation, $\mathrm{d}^r(\exp \boldsymbol{s}_i)_t = \tilde{B}_i'(t)\mathrm{d}^r \exp_{\boldsymbol{s}_i} \boldsymbol{v}_i = \tilde{B}_i'(t)\boldsymbol{v}_i$. We can now write down the derivative of (20.3) by utilizing (5.36) and (5.38):

$$\begin{aligned}\mathrm{d}^r p_t &= \mathbf{Ad}_{\exp(-\boldsymbol{s}_n)} \mathrm{d}^r (\boldsymbol{x}_0 \exp(\boldsymbol{s}_1)\dots\exp(\boldsymbol{s}_{n-1}))_t + \mathrm{d}^r \exp(\boldsymbol{s}_n)_t \\ &= \mathbf{Ad}_{\exp(-\boldsymbol{s}_n)\exp(-\boldsymbol{s}_{n-1})} \mathrm{d}^r (\boldsymbol{x}_0 \exp(\boldsymbol{s}_1)\dots\exp(\boldsymbol{s}_{n-2}))_t + \mathbf{Ad}_{\exp(-\boldsymbol{s}_n)} \mathrm{d}^r \exp(\boldsymbol{s}_{n-1})_t + \mathrm{d}^r \exp(\boldsymbol{s}_n)_t \\ &= \sum_{i=1}^{n} \mathbf{Ad}_{\exp(-\boldsymbol{s}_n)\dots\exp(-\boldsymbol{s}_{i+1})} \mathrm{d}^r \exp(\boldsymbol{s}_i)_t = \sum_{i=1}^{n} \tilde{B}_i'(t)\,\mathbf{Ad}_{\exp(-\boldsymbol{s}_n)\dots\exp(-\boldsymbol{s}_{i+1})} \boldsymbol{v}_i.\end{aligned} \tag{20.5}$$

20.1. Bezier Curves

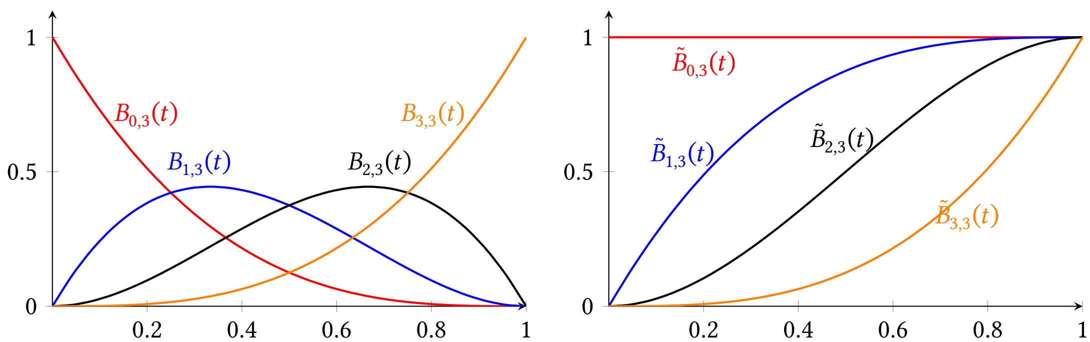


Figure 20.1.: Bernstein cubic (order 3) basis functions (left) and cumulative basis functions (right).

A Bezier curve of order k is defined on the interval $[0, 1]$ via the *Bernstein basis functions* defined as

$$B_{i,k}(t) = \binom{k}{i} t^i (1-t)^{k-i}, \qquad i = 0, \dots, k. \tag{20.6}$$

The binomial formula $1 = (t + 1 - t)^n = \sum_{i=0}^{n} B_{i,k}(t)$ shows that the Berstein basis is indeed a partition of unity. A Bezier spline is a curve that consists of Bezier curve pieces.

The Bernstein basis functions have the following properties

- They satisfy the recurrence relation

$$B_{i,k}(t) = tB_{i-1,k-1}(t) + (1-t)B_{i,k-1}(t). \tag{20.7}$$

- They are symmetric in the sense that $B_{i,k}(t) = B_{k-i,k}(1-t)$.
- The derivative can be expressed in terms of lower-order polynomials

$$\begin{aligned}B_{i,k}'(t) &= i\binom{k}{i} t^{i-1}(1-t)^{k-i} - (k-i)\binom{k}{i} t^i (1-t)^{k-i-1} \\ &= \frac{k!}{(i-1)!(k-i)!} t^{i-1}(1-t)^{k-i} + \frac{k!}{i!(k-i-1)!} t^i (1-t)^{k-i+1} = k\left(B_{i-1,k-1}(t) - B_{i,k-1}(t)\right).\end{aligned}$$

- At $t = 0$ only the first spline is nonzero: $B_{0,k}(0) = 1$, and $B_{1,k}(0) = \ldots = B_{k,k}(0) = 0$. It follows that the non-zero derivatives at $t = 0$ are $B'_{0,k}(0) = -k$ and $B'_{1,k}(0) = k$. The converse holds for $t = 1$ due to symmetry.
- At $t = 0$ only the $i = 0$ cumulative spline is nonzero: $\tilde{B}_{0,k}(0) = 1$, and only the $i = 1$ cumulative derivative is non-zero: $\tilde{B}'_{1,k}(0) = k$.
- At $t = 1$ only the $i = k$ cumulative spline is nonzero: $\tilde{B}_{k,k}(1) = 1$, and only the $i = k$ cumulative derivative is non-zero: $\tilde{B}'_{k,k}(0) = k$.

Utilizing these fact and that $\mathrm{d}^r \exp_{\boldsymbol{a}} \boldsymbol{a} = \boldsymbol{a}$ we can evaluate (20.3) and (20.5)

$$p(0) = \boldsymbol{x}_0, \quad p(1) = \boldsymbol{x}_n, \quad \mathrm{d}^r p_t|_{t=0} = \tilde{B}'_{1,k}(0)\boldsymbol{v}_1 = k\boldsymbol{v}_1, \quad \mathrm{d}^r p_t|_{t=1} = \tilde{B}'_{k,k}(0)\boldsymbol{v}_k = k\boldsymbol{v}_k. \tag{20.8}$$

These formulas can be used to fit low-degree splines to given boundary conditions.

20.1.1. Quadratic Bezier curve

A $k = 2$ spline $\boldsymbol{x}(t) = \boldsymbol{x}_0 \exp\left(\tilde{B}_{1,2}(t)\boldsymbol{v}_1\right) \exp\left(\tilde{B}_{2,2}(t)\boldsymbol{v}_2\right)$ such that $\boldsymbol{x}(0) = \boldsymbol{x}_a$, $\boldsymbol{x}(1) = \boldsymbol{x}_b$, and $\mathrm{d}^r \boldsymbol{x}_t|_{t=0} = \boldsymbol{a}_a$ is defined by the coefficients

$$\boldsymbol{x}_0 = \boldsymbol{x}_a, \quad \boldsymbol{v}_1 = \frac{\boldsymbol{a}_a}{2}, \quad \boldsymbol{v}_2 = \log\left(\exp\left(-\frac{\boldsymbol{a}_a}{2}\right)\boldsymbol{x}_a^{-1}\boldsymbol{x}_b\right), \tag{20.9}$$

and has the property that $\mathrm{d}^r \boldsymbol{x}_t|_{t=1} = 2\boldsymbol{v}_2$.

20.1.2. Cubic Bezier curve

For the cubic case $k = 3$ a Bezier curve takes the form

$$\boldsymbol{x}(t) = \boldsymbol{x}_0 \exp\left(\tilde{B}_{1,3}(t)\boldsymbol{v}_1\right) \exp\left(\tilde{B}_{2,3}(t)\boldsymbol{v}_2\right) \exp\left(\tilde{B}_{3,3}(t)\boldsymbol{v}_3\right) \tag{20.10}$$

and has first and second derivatives w.r.t. time

$$\begin{aligned}
\mathrm{d}^r \boldsymbol{x}_t &= \tilde{B}'_{1,3}(t)\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_3)}\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_2)}\, \boldsymbol{v}_1 \\
&\quad + \tilde{B}'_{2,3}(t)\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_3)}\, \boldsymbol{v}_2 \\
&\quad + \tilde{B}'_{3,3}(t)\boldsymbol{v}_3 \\
\mathrm{d}^{2r} \boldsymbol{x}_{tt} &= \tilde{B}''_{1,3}(t)\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_3)}\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_2)}\, \boldsymbol{v}_1 \\
&\qquad - \tilde{B}'_{1,3}(t)\tilde{B}'_{3,3}(t)\left[\boldsymbol{v}_3, \mathbf{Ad}_{\exp(-\boldsymbol{s}_3)}\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_2)}\, \boldsymbol{v}_1\right] \\
&\qquad - \tilde{B}'_{1,3}(t)\tilde{B}'_{2,3}(t)\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_3)}\left[\boldsymbol{v}_2, \mathbf{Ad}_{\exp(-\boldsymbol{s}_2)}\, \boldsymbol{v}_1\right] \\
&\quad + \tilde{B}''_{2,3}(t)\, \mathbf{Ad}_{\exp(-\boldsymbol{s}_3)}\, \boldsymbol{v}_2 - \tilde{B}'_{2,3}(t)\tilde{B}'_{3,3}(t)\left[\boldsymbol{v}_3, \mathbf{Ad}_{\exp(-\boldsymbol{s}_3)}\, \boldsymbol{v}_2\right] \\
&\quad + \tilde{B}''_{3,3}(t)\boldsymbol{v}_3.
\end{aligned} \tag{20.11}$$

The cumulative basis functions of order three are $\tilde{B}_{1,3}(t) = 3t - 3t^2 + t^3$, $\tilde{B}_{2,3}(t) = 3x^2 - 2x^3$, and $\tilde{B}_{3,3}(t) = t^3$. With the derivative formulas above it therefore follows that

$$\begin{aligned}
\mathrm{d}^r \boldsymbol{x}_t|_{t=0} &= 3\boldsymbol{v}_1, \quad \mathrm{d}^r \boldsymbol{x}_t|_{t=1} = 3\boldsymbol{v}_3, \\
\mathrm{d}^{2r} \boldsymbol{x}_{tt}|_{t=0} &= 6(\boldsymbol{v}_2 - \boldsymbol{v}_1), \quad \mathrm{d}^{2r} \boldsymbol{x}_{tt}|_{t=1} = 6\left(\boldsymbol{v}_3 - \mathbf{Ad}_{\exp(-\boldsymbol{v}_3)}\, \boldsymbol{v}_2\right)
\end{aligned} \tag{20.12}$$

This information can be used in two ways.

Bezier curve with given endpoint velocities First, if the values $\boldsymbol{x}_a, \boldsymbol{x}_b$ and first-order derivatives $\boldsymbol{a}_a, \boldsymbol{a}_b$ at the end points are given there is a unique Bezier curve $\boldsymbol{x}(t)$ such that $\boldsymbol{x}(0) = \boldsymbol{x}_a$, $\boldsymbol{x}(1) = \boldsymbol{x}_b$, $\mathrm{d}^r \boldsymbol{x}_t|_{t=0} = \boldsymbol{a}_a$, and $\mathrm{d}^r \boldsymbol{x}_t|_{t=1} = \boldsymbol{a}_b$. It is defined by the coefficients

$$\boldsymbol{x}_0 = \boldsymbol{x}_a, \quad \boldsymbol{v}_1 = \frac{\boldsymbol{a}_a}{3}, \quad \boldsymbol{v}_3 = \frac{\boldsymbol{a}_b}{3}, \quad \boldsymbol{v}_2 = \log\left(\exp\left(-\frac{\boldsymbol{a}_a}{3}\right)\boldsymbol{x}_a^{-1}\boldsymbol{x}_b \exp\left(-\frac{\boldsymbol{a}_b}{3}\right)\right). \tag{20.13}$$

Cubic interpolating Bezier spline Secondly, an interpolating spline for a dataset $\{(t_i, \boldsymbol{x}_i)\}_{i=0}^n$ can be defined as a collection of n Bezier curves

$$\boldsymbol{x}_i(u) = \boldsymbol{x}_{a,i} \exp(\tilde{B}_{1,3}(u)\boldsymbol{v}_{1,i}) \exp(\tilde{B}_{2,3}(u)\boldsymbol{v}_{2,i}) \exp(\tilde{B}_{3,3}(u)\boldsymbol{v}_{3,i})$$

where $u = (t - t_i)/(t_{i+1} - t_i) \in [0, 1]$. To ensure a smooth interpolation consider the following constraints (for simplicity we assume that $t_{i+1} - t_i = 1$ for all i; if not the derivative constraints have to be rescaled):

- Interpolation: $\boldsymbol{x}_{a,i} = \boldsymbol{x}_i$ and $\boldsymbol{x}_{a,i} \exp(\boldsymbol{v}_1) \exp(\boldsymbol{v}_2) \exp(\boldsymbol{v}_3) = \boldsymbol{x}_{i+1}$, for $i = 0, \dots, n-1$,
- First derivative continuity: $\boldsymbol{v}_{3,i} = \boldsymbol{v}_{1,i+1}$ for $i = 0, \dots, n-2$,
- Second derivative continuity: $\mathbf{Ad}_{\exp(-\boldsymbol{v}_{3,i})} \boldsymbol{v}_{2,i} - \boldsymbol{v}_{3,i} = \boldsymbol{v}_{2,i+1} - \boldsymbol{v}_{1,i+1}$ for $i = 0, \dots, n-2$,
- Zero second derivative at start: $\boldsymbol{v}_{2,0} - \boldsymbol{v}_{1,0} = 0$, and at end: $\mathbf{Ad}_{\exp(-\boldsymbol{v}_{3,n-1})} \boldsymbol{v}_{2,n-1} - \boldsymbol{v}_{3,n-1} = 0$.

The variables $\boldsymbol{x}_{a,i}, \boldsymbol{x}_{b,i}$ can be eliminated to end up with the following system of equations for $\boldsymbol{v}_{j,i}$:

$$\begin{aligned}
&\boldsymbol{v}_{2,0} = \boldsymbol{v}_{1,0}, \\
&\mathbf{Ad}_{\exp(-\boldsymbol{v}_{3,n-1})} \boldsymbol{v}_{2,n-1} = \boldsymbol{v}_{3,n-1}, \\
&\exp(\boldsymbol{v}_{1,i}) \exp(\boldsymbol{v}_{2,i}) \exp(\boldsymbol{v}_{3,i}) = \boldsymbol{x}_i^{-1}\boldsymbol{x}_{i+1}, && i = 0, \dots, n-1, \\
&\boldsymbol{v}_{3,i} = \boldsymbol{v}_{1,i+1}, && i = 0, \dots, n-2, \\
&\mathbf{Ad}_{\exp(-\boldsymbol{v}_{3,i})} \boldsymbol{v}_{2,i} - \boldsymbol{v}_{3,i} = \boldsymbol{v}_{2,i+1} - \boldsymbol{v}_{1,i+1}, && i = 0, \dots, n-2.
\end{aligned} \tag{20.14}$$

For Euclidean space this is a linear system of equations and is therefore straightforward to solve, but on Lie groups this is no longer possible due to nonlinearities. To find an interpolating spline we therefore either have to solve a nonlinear system of equations, or give up the strict requirement on continuity of the second-order derivatives. The following simple algorithm is of the latter kind:

1. Solve the following linearized version of (20.14):

$$\begin{aligned}
&\boldsymbol{v}_{2,0} = \boldsymbol{v}_{1,0}, \\
&\boldsymbol{v}_{2,n-1} = \boldsymbol{v}_{3,n-1}, \\
&\boldsymbol{v}_{1,i} + \boldsymbol{v}_{2,i} + \boldsymbol{v}_{3,i} = \log\left(\boldsymbol{x}_i^{-1}\boldsymbol{x}_{i+1}\right), && i = 0, \dots, n-1, \\
&\boldsymbol{v}_{3,i} = \boldsymbol{v}_{1,i+1}, && i = 0, \dots, n-2, \\
&\boldsymbol{v}_{2,i} - \boldsymbol{v}_{3,i} = \boldsymbol{v}_{2,i+1} - \boldsymbol{v}_{1,i+1}, && i = 0, \dots, n-2.
\end{aligned} \tag{20.15}$$

2. Set $\boldsymbol{v}_{2,i} = \log\left(\exp(-\boldsymbol{v}_{1,i}) x_i^{-1} x_{i+1} \exp(-\boldsymbol{v}_{3,i})\right)$ for $i = 0, \dots, n-1$.

The first step finds values for all coefficients that do not necessarily produce a continuous spline. This is however rectified in the second step. The resulting spline is guaranteed to be continuous and have a continuous first-order derivative, but in general does not have a continuous second-order derivative.

20.2. B-Splines

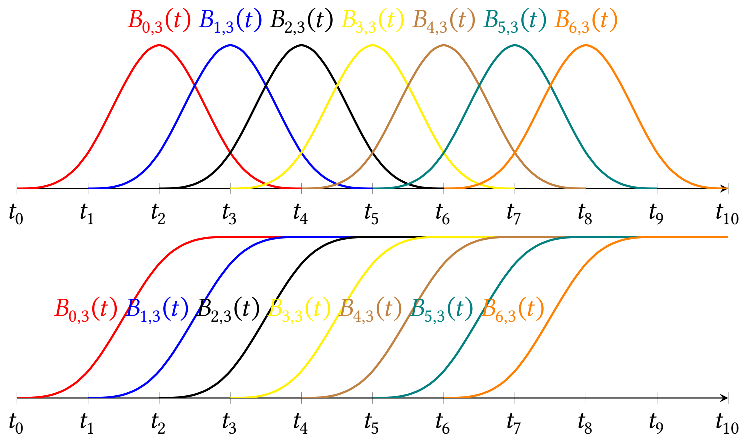


Figure 20.2.: B-spline basis function and cumulative basis functions. For $k = 3$ and $t \in [t_4, t_5)$ the non-zero basis functions are $B_{1,4}$, $B_{2,4}$, $B_{3,4}$ and $B_{4,4}$.

A B-spline interpolation of order k is a function $\boldsymbol{x}(t) = \sum_{i=0}^{N} B_{i,k}(t)\boldsymbol{x}_{\nu(i)}$ where $\boldsymbol{x}_{\nu(i)} \in \mathbb{R}^n$ are **control points** for **knots** t_i, and $B_{i,k}(t)$ are **basis functions** recursively defined as

$$\begin{aligned} B_{i,0}(t) &= \begin{cases} 1 & t_i \leq t < t_{i+1}, \\ 0 & \text{otherwise.} \end{cases} \\ B_{i,k}(t) &= \frac{t - t_i}{t_{i+k} - t_i} B_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} B_{i+1,k-1}(t). \end{aligned} \tag{20.16}$$

The following are some well-known properties of B-splines:

- $B_{i,k}(t)$ has finite support and is zero outside the interval $[t_i, t_{i+k+1})$,
- Inside this interval it is a piecewise polynomial of degree k,
- It is centered in the middle of that interval, it therefore makes sense to select k odd and $\nu(i) = i + (k+1)/2$ so that $x_{\nu(i)}$ coincides with the maximum of $B_{i,k}(t)$ (c.f. Figure 20.2),
- $\sum_i B_{i,k}(t) = 1$ for all t,

We are interested in an expression for the coefficients of the polynomial $B_{i,k}(t)$. We pose that for a **fixed interval** $t \in [t_{i^*}, t_{i^*+1})$ we have scalar cofficients $\alpha^i_{i,k}$ such that

$$B_{i,k}(t) = \sum_{l=0}^{k} \alpha^l_{i,k}\, u^l(t), \quad u(t) = \frac{t - t_{i^*}}{t_{i^*+1} - t_{i^*}} \tag{20.17}$$

where $i \in \{i^* - k, i^* - k + 1, \ldots, i^*\}$ are the indices for which $B_{i,k}(t)$ is non-zero on $[t_{i^*}, t_{i^*+1})$ (c.f. Figure 20.2). If we also introduce

$$\begin{aligned} N_{i,k} &:= \begin{bmatrix} \alpha_{i,k}^0 & \alpha_{i,k}^1 & \cdots & \alpha_{i,k}^k \end{bmatrix}^T \in \mathbb{R}^{k+1} \\ M_{i^*,k} &:= \begin{bmatrix} N_{i^*-k,k} & N_{i^*-k+1,k} & \cdots & N_{i^*,k} \end{bmatrix} \in \mathbb{R}^{k+1,k+1} \end{aligned} \tag{20.18}$$

we can write the value of a spline $x(t)$ for $t \in [t_{i^*}, t_{i^*+1}]$ as

$$\boldsymbol{x}(t) = \sum_{j=0}^{n} B_{j,k}(t) \boldsymbol{x}_{\nu(j)} = \sum_{j=i^*-k}^{i^*} B_{j,k}(t) \boldsymbol{x}_{\nu(j)} = \sum_{j=i^*-k}^{i^*} \sum_{l=0}^{k} \alpha_{j,k}^l u^l \boldsymbol{x}_{\nu(j)} = \begin{bmatrix} 1 & u & \cdots & u^k \end{bmatrix} M_{i^*,k} \begin{bmatrix} \boldsymbol{x}_{\nu(i^*-k)} \\ \vdots \\ \boldsymbol{x}_{\nu(i^*)} \end{bmatrix}. \tag{20.19}$$

20.3. Evaluating Cumulative Splines

Since for $t \in [t_{i^*}, t_{i^*+1})$ it holds that $\tilde{B}_{i,k}(t) = 1$ for $i \leq i^* - k$ and $\tilde{B}_{i,k}(t) = 0$ for $i \geq i^* + 1$ we can simplify (20.2) into

$$\boxed{\boldsymbol{x}(t) = \boldsymbol{x}_{i^*-k} \circ \prod_{j=i^*-k+1}^{i^*} \exp\left[\tilde{B}_{j,k}(t) \boldsymbol{v}_j\right], \qquad \boldsymbol{v}_j := \boldsymbol{x}_j \ominus_r \boldsymbol{x}_{j-1} = \log\left(\boldsymbol{x}_{j-1}^{-1} \circ \boldsymbol{x}_j\right).} \tag{20.20}$$

Given the $N_{j,k}$:s we can evaluate $\tilde{B}_{j,k}(t)$ as

$$\begin{bmatrix} \tilde{B}_{i^*-k,k}(t) & \tilde{B}_{i^*-k+1,k}(t) & \ldots & \tilde{B}_{i^*,k}(t) \end{bmatrix} = \begin{bmatrix} 1 & u & \ldots & u^k \end{bmatrix} \tilde{M}_{i^*,k}, \tag{20.21}$$

where $\tilde{M}_{i^*,k} \in \mathbb{R}^{k+1,k+1}$ is the column-wise reverse cumulative sum of $M_{i^*,k}$:

$$\tilde{M}_{i^*,k} = \begin{bmatrix} \sum_{j=i^*-k}^{i^*} N_{j,k} & \sum_{j=i^*-k+1}^{i^*} N_{j,k} & \ldots & N_{i^*,k} \end{bmatrix} \tag{20.22}$$

20.3.1. First order derivative

To evaluate the derivative of a spline consider the formula

$$\boldsymbol{x}(t) = \boldsymbol{y}(t) \circ \boldsymbol{z}(t), \quad \boldsymbol{z}(t) := \exp\left(\lambda(t) \boldsymbol{v}\right). \tag{20.23}$$

Since $t \in \mathbb{R}$ we can, as discussed in Remark 5.1, evaluate the derivative of x w.r.t. t as

$$\mathrm{d}^r \boldsymbol{x}_t = \mathrm{d}^r (\boldsymbol{y} \circ \boldsymbol{z})_y \, \mathrm{d}^r \boldsymbol{y}_t + \mathrm{d}^r (\boldsymbol{y} \circ \boldsymbol{z})_z \, \mathrm{d}^r \boldsymbol{z}_t. \tag{20.24}$$

From the right-jacobian derivative rules (5.36), (5.38) we know that $\mathrm{d}^r (\boldsymbol{y} \circ \boldsymbol{z})_y = \mathbf{Ad}_{z^{-1}}$ and $\mathrm{d}^r (\boldsymbol{y} \circ \boldsymbol{z})_z = I$. It therefore follows that

$$\mathrm{d}^r \boldsymbol{x}_t = \mathbf{Ad}_{\exp(-\lambda(t)\boldsymbol{v})} \, \mathrm{d}^r \boldsymbol{y}_t + \lambda'(t) \boldsymbol{v}. \tag{20.25}$$

This gives us a recursive procedure to calculate the derivative of a form (20.20) where we instead of matrix elements consider tangent elements $\boldsymbol{w}_i, \boldsymbol{v}_j \in \mathbb{R}^n$:

$$\begin{aligned}\boldsymbol{w}_{i^*-k} &= \mathbf{0},\\ \boldsymbol{w}_j &= \mathbf{Ad}_{\exp(-\tilde{B}_{j,k}(t)\boldsymbol{v}_j)}\,\boldsymbol{w}_{j-1} + \tilde{B}'_{j,k}(t)\boldsymbol{v}_j, \quad j = i^*+1,\dots,i^*+k,\\ \mathrm{d}^r\boldsymbol{x}_t &= \boldsymbol{w}_{i^*}.\end{aligned} \tag{20.26}$$

Due to using right jacobians this will result in a body velocity along the spline. If the world velocity is instead desired it can be obtained using

$$\mathrm{d}^l\boldsymbol{x}_t = \mathbf{Ad}_{x(t)}\,\mathrm{d}^r\boldsymbol{x}_t. \tag{20.27}$$

20.3.2. Second order derivative

The recursion in (20.26) can be differentiated a second time with respect to t to obtain the second order derivative. We use some properties of the adjoint to show

$$\begin{aligned}\mathbf{Ad}_{\exp(\lambda(t)\boldsymbol{u}_1)}\,\boldsymbol{u}_2 &\overset{(5.22)}{=} \exp\left(\mathrm{ad}_{\lambda(t)\boldsymbol{u}_1}\right)\boldsymbol{u}_2 \overset{(5.20)}{=} \exp\left(\lambda(t)\,\mathrm{ad}_{\boldsymbol{u}_1}\right)\boldsymbol{u}_2 \overset{(5.21)}{=} \sum_{k=0}^{\infty}\frac{\lambda(t)^k\,\mathrm{ad}_{\boldsymbol{u}_1}^k}{k!}\boldsymbol{u}_2\\ \implies \frac{\mathrm{d}}{\mathrm{d}t}\mathbf{Ad}_{\exp(\lambda(t)\boldsymbol{u}_1)}\,\boldsymbol{u}_2 &= \lambda'(t)\sum_{k=1}^{\infty}\frac{\lambda(t)^{k-1}\,\mathrm{ad}_{\boldsymbol{u}_1}^k}{(k-1)!}\boldsymbol{u}_2 = \lambda'(t)\,\mathrm{ad}_{\boldsymbol{u}_1}\sum_{k=1}^{\infty}\frac{\lambda(t)^{k-1}\,\mathrm{ad}_{\boldsymbol{u}_1}^{k-1}}{(k-1)!}\boldsymbol{u}_2\\ &= \lambda'(t)\left[\boldsymbol{u}_1, \sum_{k=0}^{\infty}\frac{\lambda(t)^k\,\mathrm{ad}_{\boldsymbol{u}_1}^k}{k!}\boldsymbol{u}_2\right] = \lambda'(t)\left[\boldsymbol{u}_1, \mathbf{Ad}_{\exp(\lambda(t)\boldsymbol{u}_1)}\,\boldsymbol{u}_2\right].\end{aligned} \tag{20.28}$$

With $\lambda(t) \to -\tilde{B}_{j,k}(t), \boldsymbol{u}_1 \to \boldsymbol{v}_j, \boldsymbol{u}_2 \to \boldsymbol{w}_{j-1}$ we get $\mathbf{Ad}_{\exp(-\tilde{B}_{j,k}(t)\boldsymbol{v}_j)}\,\boldsymbol{w}_{j-1} \overset{(20.26)}{=} \boldsymbol{w}_j - \tilde{B}'_{j,k}(t)\boldsymbol{v}_j$, and therefore by introducing $\boldsymbol{q}_j \coloneqq \frac{\mathrm{d}\boldsymbol{w}_j}{\mathrm{d}t}$:

$$\begin{aligned}\boldsymbol{q}_{i^*-k} &= \mathbf{0},\\ \boldsymbol{q}_j &= \tilde{B}'_{j,k}(t)\left[\boldsymbol{w}_j, \boldsymbol{v}_j\right] + \mathbf{Ad}_{\exp(-\tilde{B}_{j,k}(t)\boldsymbol{v}_j)}\,\boldsymbol{q}_{j-1} + \tilde{B}^{(2)}_{j,k}(t)\boldsymbol{v}_j, \qquad j = i^*+1,\dots,i^*+k,\\ \frac{\mathrm{d}}{\mathrm{d}t}\mathrm{d}^r\boldsymbol{x}_t &= \boldsymbol{q}_{i^*+k}.\end{aligned} \tag{20.29}$$

20.3.3. Derivatives w.r.t. Control Points

Finally it can be useful to express the derivative of $\boldsymbol{x}(t)$ with respect to the control point values $\boldsymbol{x}_j$. Recall that

$$\boldsymbol{x}(t) = \boldsymbol{x}_{i^*-k} \circ \prod_{j=i^*-k+1}^{i^*} \exp\left[\tilde{B}_{j,k}(t)\boldsymbol{v}_j\right], \tag{20.30}$$

so it is again just a matter of differentiating.

Let $\boldsymbol{s}_j = \tilde{B}_{j,k}(t)\boldsymbol{v}_j$, we then have that $\boldsymbol{x}(t) = x_{i^*-k} \circ \prod_{j=i^*-k+1}^{i^*} \exp(s_j)$. Derivatives with respect to the terms are

$$\mathrm{d}^r(\boldsymbol{s}_j)_{x_j} \overset{(5.47)}{=} \tilde{B}_{j,k}(t)\left[\mathrm{d}^r \exp_{\boldsymbol{v}_j}\right]^{-1} \implies \mathrm{d}^r\left(\exp(\boldsymbol{s}_j)\right)_{\boldsymbol{x}_j} = \tilde{B}_{j,k}(t)\left[\mathrm{d}^r \exp_{\boldsymbol{s}_j}\right]\left[\mathrm{d}^r \exp_{\boldsymbol{v}_j}\right]^{-1}, \tag{20.31a}$$

$$\mathrm{d}^r(\boldsymbol{s}_j)_{x_{j-1}} \overset{(5.48)}{=} -\tilde{B}_{j,k}(t)\left[\mathrm{d}^l \exp_{\boldsymbol{v}_j}\right]^{-1} \implies \mathrm{d}^r\left(\exp(\boldsymbol{s}_j)\right)_{\boldsymbol{x}_{j-1}} = -\tilde{B}_{j,k}(t)\left[\mathrm{d}^r \exp_{\boldsymbol{s}_j}\right]\left[\mathrm{d}^l \exp_{\boldsymbol{v}_j}\right]^{-1}. \tag{20.31b}$$

Thus the derivatives $\boldsymbol{r}_j := \mathrm{d}^r\boldsymbol{x}(t)_{\boldsymbol{x}_j}$ of $\boldsymbol{x}$ become (where the $\bar{z}$'s are constant w.r.t. the differentiation variable)

$$\begin{aligned}
\boldsymbol{r}_{i^*} &= \mathrm{d}^r\left(\bar{\boldsymbol{z}} \circ \exp\left[\boldsymbol{s}_{i^*}\right]\right)_{\boldsymbol{x}_{i^*}} \overset{(5.12)}{=} \mathrm{d}^r\exp\left(\boldsymbol{s}_{i^*}\right)_{\boldsymbol{x}_{i^*}} \overset{(20.31a)}{=} \tilde{B}_{i^*,k}\mathrm{d}^r\exp_{\boldsymbol{s}_{i^*}}\left[\mathrm{d}^r\exp_{\boldsymbol{v}_{i^*}}\right]^{-1}, \\
\boldsymbol{r}_j &= \mathrm{d}^r\left(\bar{z}_1 \circ \exp\left[\boldsymbol{s}_j\right] \circ \exp\left[\boldsymbol{s}_{j+1}\right] \circ \bar{z}_2\right)_{\boldsymbol{x}_j} \overset{(5.12)}{=} \mathbf{Ad}_{\bar{z}_2^{-1}}\,\mathrm{d}^r\left(\bar{z}_1 \circ \exp\left[\boldsymbol{s}_j\right] \circ \exp\left[\boldsymbol{s}_{j+1}\right]\right)_{\boldsymbol{x}_j} \\
&\overset{(5.12)}{=} \mathbf{Ad}_{\bar{z}_2^{-1}}\left(\mathbf{Ad}_{\exp(-\boldsymbol{s}_{j+1})}\,\mathrm{d}^r(\bar{z}_1 \circ \exp(\boldsymbol{s}_j))_{\boldsymbol{x}_j} + \mathrm{d}^r(\exp(\boldsymbol{s}_{j+1}))_{\boldsymbol{x}^j}\right) \\
&\overset{(5.12)}{=} \mathbf{Ad}_{\bar{z}_2^{-1}}\left(\mathbf{Ad}_{\exp(-\boldsymbol{s}_{j+1})}\,\mathrm{d}^r(\exp(\boldsymbol{s}_j))_{\boldsymbol{x}_j} + \mathrm{d}^r(\exp(\boldsymbol{s}_{j+1}))_{\boldsymbol{x}^j}\right) \\
&\overset{(20.31)}{=} \mathbf{Ad}_{\bar{z}_2^{-1}}\left(\tilde{B}_{j,k}(t)\,\mathbf{Ad}_{\exp(-s_{j+1})}\left[\mathrm{d}^r\exp_{\boldsymbol{s}_j}\right]\left[\mathrm{d}^r\exp_{\boldsymbol{v}_j}\right]^{-1} - \tilde{B}_{j+1,k}(t)\left[\mathrm{d}^r\exp_{\boldsymbol{s}_{j+1}}\right]\left[\mathrm{d}^l\exp_{\boldsymbol{v}_{j+1}}\right]^{-1}\right), \\
\boldsymbol{r}_{i^*-k} &= \mathrm{d}^r\left(\boldsymbol{x}_{i^*-k} \circ \exp\left[\boldsymbol{s}_{i^*-k+1}\right] \circ \bar{z}_2^{-1}\right) = \\
&= \mathbf{Ad}_{\bar{z}_2^{-1}}\left(\mathbf{Ad}_{\exp(-\boldsymbol{s}_{i^*-k+1})} - \tilde{B}_{i^*-k+1}(t)\mathrm{d}^r\exp_{\boldsymbol{s}_{i^*-k+1}}\left[\mathrm{d}^r\exp_{\boldsymbol{v}_{i^*-k+1}}\right]^{-1}\right).
\end{aligned}$$

20.4. General Coefficient Recursion

We seek an expression for $M_{i^*,k}$ which via (20.22) immediately gives $\tilde{M}_{i^*,k}$ that allows easy evaluation of the basis functions $\tilde{B}_{i,j}$. Inserting the basis expansion (20.17) into the recursive definition (20.16) yields

$$
\begin{aligned}
&\sum_{j=0}^{k} \alpha_{i,k}^{j} u^j =: B_{i,k}(t) = \frac{t - t_i}{t_{i+k} - t_i} B_{i,k-1}(t) + \frac{t_{i+k+1} - t}{t_{i+k+1} - t_{i+1}} B_{i+1,k-1}(t) \\
&= \left[\frac{t_{i^*} - t_i}{t_{i+k} - t_i} + \frac{t_{i^*+1} - t_{i^*}}{t_{i+k} - t_i} u \right] B_{i,k-1}(t) + \left[\frac{t_{i+k+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} - \frac{t_{i^*+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} u \right] B_{i+1,k-1}(t) \\
&= \frac{t_{i^*} - t_i}{t_{i+k} - t_i} \sum_{j=0}^{k-1} \alpha_{i,k-1}^{j} u^j + \frac{t_{i^*+1} - t_{i^*}}{t_{i+k} - t_i} \sum_{j=0}^{k-1} \alpha_{i,k-1}^{j} u^{j+1} \\
&+ \frac{t_{i+k+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} \sum_{j=0}^{k-1} \alpha_{i+1,k-1}^{j} u^j - \frac{t_{i^*+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} \sum_{j=0}^{k-1} \alpha_{i+1,k-1}^{j} u^{j+1} \\
&= \frac{t_{i^*} - t_i}{t_{i+k} - t_i} \sum_{j=0}^{k-1} \alpha_{i,k-1}^{j} u^j + \frac{t_{i^*+1} - t_{i^*}}{t_{i+k} - t_i} \sum_{j=1}^{k} \alpha_{i,k-1}^{j-1} u^j + \\
&\frac{t_{i+k+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} \sum_{j=0}^{k-1} \alpha_{i+1,k-1}^{j} u^j - \frac{t_{i^*+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} \sum_{j=1}^{k} \alpha_{i+1,k-1}^{j-1} u^j \\
&= \sum_{j=0}^{k} \left[\frac{t_{i^*} - t_i}{t_{i+k} - t_i} \alpha_{i,k-1}^{j} + \frac{t_{i^*+1} - t_{i^*}}{t_{i+k} - t_i} \alpha_{i,k-1}^{j-1} + \frac{t_{i+k+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} \alpha_{i+1,k-1}^{j} - \frac{t_{i^*+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}} \alpha_{i+1,k-1}^{j-1} \right] u^j,
\end{aligned}
$$

with the convention that $\alpha_{i,k}^{j} = 0$ for $j < 0$ and for $j > k$. By matching coefficients we therefore have that

$$
\alpha_{i,k}^{j} = \underbrace{\frac{t_{i^*} - t_i}{t_{i+k} - t_i}}_{=:\tilde{\beta}_{i,i^*,k}} \alpha_{i,k-1}^{j} + \underbrace{\frac{t_{i^*+1} - t_{i^*}}{t_{i+k} - t_i}}_{=:\tilde{\gamma}_{i,i^*,k}} \alpha_{i,k-1}^{j-1} + \underbrace{\frac{t_{i+k+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}}}_{1-\tilde{\beta}_{i+1,i^*,k}} \alpha_{i+1,k-1}^{j} - \underbrace{\frac{t_{i^*+1} - t_{i^*}}{t_{i+k+1} - t_{i+1}}}_{\tilde{\gamma}_{i+1,iz*,k}} \alpha_{i+1,k-1}^{j-1}, \tag{20.32}
$$

or equivalently that for $N_{i,k}$ as in (20.18),

$$
\begin{aligned}
N_{i,k} &= \tilde{\beta}_i \begin{bmatrix} N_{i,k-1} \\ 0 \end{bmatrix} + \tilde{\gamma}_i \begin{bmatrix} 0 \\ N_{i,k-1} \end{bmatrix} + (1 - \tilde{\beta}_{i+1,i^*,k}) \begin{bmatrix} N_{i+1,k-1} \\ 0 \end{bmatrix} - \tilde{\gamma}_{i+1,i^*,k} \begin{bmatrix} 0 \\ N_{i+1,k-1} \end{bmatrix} \\
&= \begin{bmatrix} N_{ik-1} & N_{i+1,k-1} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{\beta}_i \\ 1 - \tilde{\beta}_{i+1} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ N_{i,k-1} & N_{i+1,k-1} \end{bmatrix} \begin{bmatrix} \tilde{\gamma}_{i,i^*,k} \\ -\tilde{\gamma}_{i+1,i^*,k} \end{bmatrix},
\end{aligned} \tag{20.33}
$$

For convenience we re-define β and γ as

$$
\beta_{j,i^*,k} := \tilde{\beta}_{i^*-j,i^*,k} = \frac{t_{i^*} - t_{i^*-j}}{t_{i^*-j+k} - t_{i^*-j}}, \quad \gamma_{j,i^*,k} := \tilde{\gamma}_{i^*-j,i^*,k} = \frac{t_{i^*+1} - t_{i^*}}{t_{i^*-j+k} - t_{i^*-j}} \tag{20.34}
$$

Now we can write down a recursive formula for $M_{i^*,k}$ as given in (20.18):

$$
\begin{aligned}
M_{i^*,0} &= \begin{bmatrix} 1 \end{bmatrix}, \\
M_{i^*,k} &= \begin{bmatrix} M_{i^*,k-1} \\ 0 \end{bmatrix} \begin{bmatrix} 1-\beta_{k-1,i^*,k} & \beta_{k-1,i^*,k} & 0 & \cdots & 0 \\ 0 & 1-\beta_{k-2,i^*,k} & \beta_{k-2,i^*,k} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 1-\beta_{0,i^*,k} & \beta_{0,i^*,k} \end{bmatrix} \\
&+ \begin{bmatrix} 0 \\ M_{i^*,k-1} \end{bmatrix} \begin{bmatrix} -\gamma_{k-1,i^*,k} & \gamma_{k-1,i^*,k} & 0 & \cdots & 0 \\ 0 & -\gamma_{k-2,i^*,k} & \gamma_{k-2,i^*,k} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -\gamma_{0,i^*,k} & \gamma_{0,i^*,k} \end{bmatrix}.
\end{aligned}
\tag{20.35}
$$

However, close to the endpoints $\beta_{j,i^*,k}$ and $\gamma_{j,i^*,k}$ can no longer be evaluated. We can introduce artificial boundary knot points—$k-1$ to the left and $k-2$ on the right—to ensure that all splines have full support. Then β and γ can be computed using the expressions

$$
\beta_{j,i^*,k} = \frac{t_{i^*} - t_{\max(i^*-j,0)}}{t_{\min(i^*-j+k,n)} - t_{\max(i^*-j,0)}}, \quad \gamma_{j,i^*,k} = \frac{t_{i^*+1} - t_{i^*}}{t_{\min(i^*-j+k,n)} - t_{\max(i^*-j,0)}}
\tag{20.36}
$$

that are valid for all indices $0 \le i^* < n$.

20.5. Cardinal Cofficient Recursion

When all control points with indices $i^*-k+1,\dots,i^*$ are equally spaced such that $t_{i+1} - t_i = \Delta t$ for all i the expression can be simplified and $M_{i^*,k}$ no longer depends on i^* for interior points. In this case we have that

$$
\beta_{j,i^*,k} = \frac{j\Delta t}{k\Delta t} = \frac{j}{k}, \quad \gamma_{j,i^*,k} = \frac{1}{k}.
\tag{20.37}
$$

We can use this to retrieve the first couple of matrices:

$$
M_{i^*,0} = \begin{bmatrix} 1 \end{bmatrix}, \tag{20.38a}
$$

$$
M_{i^*,1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1-\beta_{i^*,1} & \beta_{i^*,1} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \begin{bmatrix} -\gamma_1 & \gamma_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \end{bmatrix}, \tag{20.38b}
$$

$$
M_{i^*,2} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 1-\beta_{i^*-1,2} & \beta_{i^*-1,2} & 0 \\ 0 & 1-\beta_{i^*,2} & \beta_{i^*,2} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} -\gamma_2 & \gamma_2 & 0 \\ 0 & -\gamma_2 & \gamma_2 \end{bmatrix} = \frac{1}{2!} \begin{bmatrix} 1 & 1 & 0 \\ -2 & 2 & 0 \\ 1 & -2 & 1 \end{bmatrix} \tag{20.38c}
$$

$$
M_{i^*,3} = \dots = \frac{1}{3!} \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}. \tag{20.38d}
$$

Close to the boundary the formulas (20.36) should instead be applied.

21. Advanced control: controllability, lie brackets, and frobenius

- Difference is that vector fields are non-constant

22. Advanced: lidar odometry

22.1. Correspondence search

Assume that for the two clouds Q_{k-1} and Q_k edge and plane features $\mathscr{E}_{k-1}, \mathscr{H}_K, \mathscr{E}_k$, and $\mathscr{H}_k$ have been extracted.

Edge points in $\mathscr{E}_k$ should be matched with an edge in $\mathscr{E}_{k-1}$.

22.1.1. Edge matching

For a point $x_e \in \mathscr{E}_k$ select the n closest points S from $\mathscr{E}_{k-1}$, calculate the x-y-z covariance matrix and ensure that one eigenvalue is much larger than the other two. The fitting residual errors are

$$r_e(x_e) = (x_e - x_0) - ((x_e - x_0) \cdot \hat{d})\hat{d} = (I - \hat{d}\hat{d}^T)(x_e - x_0) \in \mathbb{R}^3$$

where x_0 is the centroid of S (assumed to be on the edge) and $\hat{d}$ is the normalized direction of the edge (eigenvector corresponding to largest eigenvalue of covariance matrix).

22.1.2. Plane matching

For a point $x_h \in \mathscr{H}_k$ select the n closest points from $\mathscr{H}_{k-1}$, calculate the x-y-z covariance matrix and ensure that one eigenvalue is much smaller than the other two. The fitting residual error is

$$r_h(x_h) = ((x_h - x_0) \cdot \hat{n})\,\hat{n} = (\hat{n}\hat{n}^T)\,(x_h - x_0) \in \mathbb{R}^3$$

where x_0 is the centroid of S (assumed to be in the plane) and $\hat{n}$ is the normalized normal of the plane (eigenvector corresponding to smallest eigenvalue of covariance matrix).

22.2. Optimization

For each detected correspondence we consider the transformed residuals

$$r_e(\exp(\omega\tau)x_e),$$

and

$$r_h(\exp(\omega\tau)x_h),$$

respectively, where $\omega \in \mathfrak{se}(3)$ represents the velocity during the sweep that is to be estimated, and τ is the time that has elapsed since the start of the sweep. That is, $\exp(\omega\tau)$ represents the relative pose between time t_k (start of the sweep), and time $t_k + \tau$ (time of point cloud collection).

23. Advanced: Marginalization of nonlinear least squares

Objective is to remove a variable from the problem in a way so that
 * The optimal solution is not effected * The first derivative at the optimal solution remains the same
 For a nonlinear problem this is not possible, so we do it around a linearization point.

23.1. Lifted information matrix

Consider a set of variables $X = \{x_1, \dots, x_k\}$ where $x_i \in M_i$ and a square form

$$S = \frac{1}{2} \sum_j \left(h_j(X_j) - y_j\right)^T I_j \left(h_j(X_j) - y_j\right),$$

where $X_j = \{x_{j_1}, \dots x_{j_{n_j}}\} \subset X$ is a set of variables for the j:th measurement, and $h_j : X_j \mapsto h_j(X_j) \in \mathbb{R}^{p_j}$ are nonlinear measurement functions. Also let $I_j = \{j_1, \dots, j_{n_j}\}$ be the variable indices for measurement number j.

We are interested in marginalizing the expression S around a point $\{x_k = \mu_k\}$. Via Taylor expansion we obtain with $\mu_j = \begin{bmatrix} \mu_{j_1} & \cdots & \mu_{j_{n_j}} \end{bmatrix}$ being the measurement mean:

$$2S \approx \sum_j \left(h_j(\mu_j) + \sum_{i \in I_j} [\mathrm{d}_i h_j]_{\mu_j} e_i - y_j \right)^T I_j \left(h_j(\mu_j) + \sum_{i \in I_j} [\mathrm{d}_i h_j]_{\mu_j} e_i - y_j \right).$$

Here $[\mathrm{d}_i h_j]_{\mu_j} : TM_i \to \mathbb{R}^{p_j}$ is the differential of the measurement $h_j : \prod_{i \in I_j} M_i \to \mathbb{R}^{p_j}$ with respect to x_i evaluated at μ_j. The error differentials e_i are such that

$$x_i = \mu_i \oplus e_i = \mu_i \mathrm{Exp}_i(e_i) \quad \Longleftrightarrow \quad e_i = x_i \ominus \mu_i = \mathrm{Log}_i(\mu_i^{-1} x_i),$$

where $\mathrm{Exp} : \mathbb{R}^{n_i} \to M_i$ maps from coordinates in the tangent space to the manifold, and Log is the inverse mapping.

We now expand the sum

$$\begin{aligned} 2S \approx & \sum_j \left(h_j(\mu_j) - y_j\right)^T I_j \left(h_j(\mu_j) - y_j\right) + \sum_j \left(\sum_{i \in I_j} [\mathrm{d}_i h_j]_{\mu_j} e_i \right)^T I_j \left(\sum_{i \in I_j} [\mathrm{d}_i h_j]_{\mu_j} e_i \right) \\ & + 2 \sum_j \left(h_j(\mu_j) - y_j\right)^T I_j \left(\sum_{i \in I_j} [\mathrm{d}_i h_j]_{\mu_j} e_i \right) \end{aligned}$$

and consider the term quadratic in e. We can augment the middle matrix with the differentials and

$$\sum_j \left(\sum_{i \in I_j} [\mathrm{d}_i h_j]_{\mu_j} e_i \right)^T I_j \left(\sum_{i \in I_j} [\mathrm{d}_i h_j]_{\mu_j} e_i \right)$$
$$= \sum_j \begin{bmatrix} e_{j_1} & \cdots & e_{j_{n_j}} \end{bmatrix} \begin{bmatrix} [\mathrm{d}_{j_1} h_j]_{\mu_j}^T \\ \vdots \\ [\mathrm{d}_{j_{n_j}} h_j]_{\mu_j}^T \end{bmatrix} I_j \begin{bmatrix} [\mathrm{d}_{j_1} h_j]_{\mu_j} & \cdots & [\mathrm{d}_{j_{n_j}} h_j]_{\mu_j} \end{bmatrix} \begin{bmatrix} e_{j_1} \\ \vdots \\ e_{j_{n_j}} \end{bmatrix}$$
$$= \sum_j \begin{bmatrix} e_{j_1} & \cdots & e_{j_{n_j}} \end{bmatrix} \begin{bmatrix} [\mathrm{d}_{j_1} h_j]_{\mu_j}^T I_j [\mathrm{d}_{j_1} h_j]_{\mu_j} & \cdots & [\mathrm{d}_{j_1} h_j]_{\mu_j}^T I_j [\mathrm{d}_{j_{n_j}} h_j]_{\mu_j} \\ \vdots & \ddots & \vdots \\ [\mathrm{d}_{j_{n_j}} h_j]_{\mu_j}^T I_j [\mathrm{d}_{j_1} h_j]_{\mu_j} & \cdots & [\mathrm{d}_{j_{n_j}} h_j]_{\mu_j}^T I_j [\mathrm{d}_{j_{n_j}} h_j]_{\mu_j} \end{bmatrix} \begin{bmatrix} e_{j_1} \\ \vdots \\ e_{j_{n_j}} \end{bmatrix}$$
$$= \sum_{i_1, i_2} e_{i_1} \left[\sum_{j : i_1, i_2 \in I_j} [\mathrm{d}_{i_1} h_j]_{\mu_j}^T I_j [\mathrm{d}_{i_2} h_j]_{\mu_j} \right] e_{i_2} = \begin{bmatrix} e_1 & \cdots & e_k \end{bmatrix} \Lambda \begin{bmatrix} e_1 \\ \vdots \\ e_k \end{bmatrix},$$

where Λ is the sum of **block-lifted information matrices** obtained by placing the blocks from cost functions at the appropriate places.

It follows that we can write S on the information form

$$S \sim \eta^T \mathbf{e} + \frac{1}{2} \mathbf{e}^T \Lambda \mathbf{e}$$

with Λ as above and η a similarly block-lifted column vector such that

$$\eta^T \mathbf{e} = \sum_{i=1}^k \left[\sum_{j : i \in I_j} (h_j(\mu_j) - y_j)^T I_j [\mathrm{d}_i h_j]_{\mu_j} \right] e_i.$$

23.2. Marginalization

We group the variables into e_α and e_β, where e_β is the variable to be removed, and write S on the general information form

$$S \sim \begin{bmatrix} \eta_\alpha^T & \eta_\beta^T \end{bmatrix} \begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix} + \frac{1}{2} \begin{bmatrix} e_\alpha^T & e_\beta^T \end{bmatrix} \begin{bmatrix} \Lambda_{\alpha\alpha} & \Lambda_{\alpha\beta} \\ \Lambda_{\beta\alpha} & \Lambda_{\beta\beta} \end{bmatrix} \begin{bmatrix} e_\alpha \\ e_\beta \end{bmatrix}$$

We can then expand the information matrix in the same way as in Joplin normal distribution to obtain

$$S \sim \begin{bmatrix} \eta_\alpha^T - \eta_\beta^T \Lambda_{\beta\beta}^{-1} \Lambda_{\beta\alpha} & \eta_\beta^T \end{bmatrix} \begin{bmatrix} e_\alpha \\ e_\beta + \Lambda_{\beta\beta}^{-1} \Lambda_{\beta\alpha} e_\alpha \end{bmatrix} + \frac{1}{2} \begin{bmatrix} e_\alpha \\ e_\beta + \Lambda_{\beta\beta}^{-1} \Lambda_{\beta\alpha} e_\alpha \end{bmatrix}^T \begin{bmatrix} \Lambda / \Lambda_{\beta\beta} & 0 \\ 0 & \Lambda_{\beta\beta} \end{bmatrix} \begin{bmatrix} e_\alpha \\ e_\beta + \Lambda_{\beta\beta}^{-1} \Lambda_{\beta\alpha} e_\alpha \end{bmatrix}.$$

That is, we have separated the expression into two quadratic expressions. A coordinate change

$$\kappa = e_\beta + \Lambda_{\beta\beta}^{-1} \Lambda_{\beta\alpha} e_\alpha$$

reveals that they can be solved independently. The sub-problem for e_α reads

$$\left(\eta_\alpha^T - \eta_\beta^T \Lambda_{\beta\beta}^{-1} \Lambda_{\beta\alpha}\right) e_a + \frac{1}{2} e_\alpha^T (\Lambda/\Lambda_{\beta\beta}) e_\alpha.$$

By completing the square we can write this as

$$\sim \frac{1}{2}\left(e_\alpha + (\Lambda/\Lambda_{\beta\beta})^{-1}\left(\eta_\alpha - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\eta_\beta\right)\right)^T (\Lambda/\Lambda_{\beta\beta})\left(e_\alpha + (\Lambda/\Lambda_{\beta\beta})^{-1}\left(\eta_\alpha - \Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\eta_\beta\right)\right)$$

which is the marginalized form of the problem.

23.3. Algorithm

Suppose we want to remove a variable x_k. Consider the set of factors F such that $k \in I_f$ for all $f \in F$, and the resulting blanket set of variables $X_F = \bigcup_{f \in F} \bigcup_{j \in f_j} X_j$.

Ceres provides evaluations $E_j = \sqrt{I_j}(h_j(\mu_j) - y_j)$ and gradient blocks $G_{ji} = \sqrt{I_j}[\mathrm{d}_i h_j]_{\mu_j}$. **It's fine if residuals are defined as the negative since the signs will cancel in multiplication**.

1. Find the information matrix Λ by summing over all factors in F. Sum blocks in Λ are of the form $[\mathrm{d}_{i_1} h_j]_{\mu_j}^T I_j [\mathrm{d}_{i_2} h_j]_{\mu_j} = G_{ji_1}^T G_{ji_2}$. 2. Find the mean vector η by summing over all factors in F. Sum segments in η are of the form $(h_j(\mu_j) - y_j)^T I_j [\mathrm{d}_i h_j]_{\mu_j} = E_j^T G_{ji}$. 3. Partition Λ and η as

$$\Lambda = \begin{bmatrix} \Lambda_{\neg k \neg k} & \Lambda_{\neg k k} \\ \Lambda_{k \neg k} & \Lambda_{kk} \end{bmatrix}, \quad \eta = \begin{bmatrix} \eta_{\neg k} \\ \eta_l \end{bmatrix}.$$

3. Calculate $\tilde{\Lambda} = \Lambda/\Lambda_k$ and $\gamma = -(\Lambda/\Lambda_k)^{-1}\left(\eta_{\neg k} - \Lambda_{\neg k k}\Lambda_{kk}^{-1}\eta_k\right)$ 4. Remove factors F and instead insert a new factor with cost function

$$(e_{\neg k} - \gamma)^T \tilde{\Lambda} (e_{\neg k} - \gamma) = \begin{bmatrix} e_1 - \gamma_1 \\ \vdots \\ e_n - \gamma_n \end{bmatrix}^T \tilde{\Lambda} \begin{bmatrix} e_1 - \gamma_1 \\ \vdots \\ e_n - \gamma_n \end{bmatrix} = \begin{bmatrix} \mathrm{Log}(\mu_1^{-1} x_1) - \gamma_1 \\ \vdots \\ \mathrm{Log}(\mu_n^{-1} x_n) - \gamma_n \end{bmatrix}^T \tilde{\Lambda} \begin{bmatrix} \mathrm{Log}(\mu_1^{-1} x_1) - \gamma_1 \\ \vdots \\ \mathrm{Log}(\mu_n^{-1} x_n) - \gamma_n \end{bmatrix}$$

23.4. Correction for singular information matrix

In the event that $\tilde{\Lambda}$ is singular we can not calculate γ. Instead consider the decomposition $\tilde{\Lambda} = UDU^T$, where D is a square diagonal matrix with only non-zero diagonal entries. We can then let

$$\gamma = -UD^{-1}U^T\left(\eta_{\neg k} - \eta_k \Lambda_{kk}^{-1} \Lambda_{k \neg k}\right)$$

and consider the cost

$$\left\| \sqrt{D} U^T \begin{bmatrix} \mathrm{Log}(\mu_1^{-1} x_1) - \gamma_1 \\ \vdots \\ \mathrm{Log}(\mu_n^{-1} x_n) - \gamma_n \end{bmatrix} \right\|^2$$

which has a non-zero information matrix.

23.5. Marginalization factor in local frame

The above linearizes around a world point $\{\mu\}$. This makes sense if marginalizing a node that has an absolute factor, but perhaps not when it is only connected via relative factors and there may be a lot of drift. We can transform the measurements into a local frame by instead introducing the cost

$$\left\| \sqrt{D} U^T \begin{bmatrix} \mathrm{Log}(\mu_{01}^{-1} x_0^{-1} x_1) - \gamma_1 \\ \vdots \\ \mathrm{Log}(\mu_{0n}^{-1} x_0^{-1} x_n) - \gamma_n \end{bmatrix} \right\|^2$$

where $\mu_{0i} = \mu_0^{-1} \mu_i$ are linearization points transformed into the local frame of x_0, which should be selected as a pose in the vicinity of the removed node.

* Only works if the measurements h_j are invariant to rigid transformations. This property holds for relative measurements such as relative poses and landmark triangulations.

* For a node with absolute factors the measurement h would have to be adjusted before building the *gamma* vector.

23.6. Example

Consider the least-squares problem

$$S = (x_1 - x_3 + 1)^2 + (x_2 - x_3 + 1)^2$$

where $h_1(\mathbf{x}) = x_1 - x_3$, $y_1 = -1$, $h_2(\mathbf{x}) = x_2 - x_3$, and $y_2 = -1$.

We expand in a Taylor form as above

$$\begin{aligned} S &\approx \left(h_1(\mathbf{x}_0) + \mathrm{d}h_1 \cdot (\mathbf{x} - \mathbf{x}_0) - y_1\right)^2 + \left(h_2(\mathbf{x}_0) + \mathrm{d}h_2 \cdot (\mathbf{x} - \mathbf{x}_0) - y_2\right)^2 \\ &= (\mathrm{d}h_1 \cdot (\mathbf{x} - \mathbf{x}_0))^2 + (\mathrm{d}h_2 \cdot (\mathbf{x} - \mathbf{x}_0))^2 \\ &\quad + 2\left[(h_1(\mathbf{x}_0) - y_1)\mathrm{d}h_1 \cdot (\mathbf{x} - \mathbf{x}_0) + (h_2(\mathbf{x}_0) - y_2)\mathrm{d}h_2 \cdot (\mathbf{x} - \mathbf{x}_0)\right] + C \end{aligned}$$

where C is a constant. Since both h are linear this expression is exact for any $\mathbf{x}_0$ (can be verified).

We let $e_i = x_i - x_0^i$ and get

$$
\begin{aligned}
S &= \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}^T \left(\mathrm{d}h_1^T \mathrm{d}h_1 + \mathrm{d}h_2^T \mathrm{d}h_2\right) \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} + 2\left[(h_1(\mathbf{x}_0) - y_1)\mathrm{d}h_1 + (h_2(\mathbf{x}_0) - y_2)\mathrm{d}h_2\right] \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \\
&= \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}^T \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \\
&+ 2\left[(x_1^0 - x_3^0 + 1)\begin{bmatrix} 1 & 0 & -1 \end{bmatrix} + (x_2^0 - x_3^0 + 1)\begin{bmatrix} 0 & 1 & -1 \end{bmatrix}\right] \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \\
&= \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}^T \begin{bmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \\
&+ 2\begin{bmatrix} x_1^0 - x_3^0 + 1 & x_2^0 - x_3^0 + 1 & -x_1^0 - x_2^0 + 2x_3^0 - 2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix}
\end{aligned}
$$

We now marginalize out x_3 and identify the marginalized covariance

$$
\tilde{\Lambda} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} -1 \\ -1 \end{bmatrix} \begin{bmatrix} 2 \end{bmatrix}^{-1} \begin{bmatrix} -1 & -1 \end{bmatrix} = \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 1/2 \end{bmatrix}
$$

and the marginalized mean

$$
\begin{aligned}
\tilde{\eta}^T &= \eta_\alpha^T - \eta_\beta^T \Lambda_{\beta\beta}^{-1} \Lambda_{\beta\alpha} \\
&= \begin{bmatrix} x_1^0 - x_3^0 + 1 & x_2^0 - x_3^0 + 1 \end{bmatrix} - \begin{bmatrix} -x_1^0 - x_2^0 + 2x_3^0 - 2 \end{bmatrix} \begin{bmatrix} 2 \end{bmatrix}^{-1} \begin{bmatrix} -1 & -1 \end{bmatrix} \\
&= \frac{1}{2} \begin{bmatrix} x_1^0 - x_2^0 & -x_1^0 + x_2^0 \end{bmatrix}.
\end{aligned}
$$

The marginalized problem is now

$$
\tilde{S} = \begin{bmatrix} x_1 - x_1^0 \\ x_2 - x_2^0 \end{bmatrix}^T \begin{bmatrix} 1/2 & -1/2 \\ -1/2 & 1/2 \end{bmatrix} \begin{bmatrix} x_1 - x_1^0 \\ x_2 - x_2^0 \end{bmatrix} + 2\begin{bmatrix} \frac{x_1^0 - x_2^0}{2} & \frac{-x_1^0 + x_2^0}{2} \end{bmatrix} \begin{bmatrix} x_1 - x_1^0 \\ x_2 - x_2^0 \end{bmatrix}.
$$

After expanding and removing constant terms this is equal to

$$
\tilde{S} = \frac{1}{2} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^T \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \frac{1}{2}(x_1 - x_2)^2.
$$

That is, the problem is independent of the linearization point, as expected.

23.7. As part of least-squares problem

Now, for the least-squares problem that has this as a factor:

$$x_1^2 + (x_1 - x_2 + 1)^2 + (x_1 - x_3 + 1)^2 + (x_2 - x_3 + 1)^2$$

that the above as terms, marginalizing x_3 replaces the last two terms with the above, so the marginalized problem becomes

$$x_1^2 + (x_1 - x_2 + 1)^2 + \frac{1}{2}(x_1 - x_2)^2.$$

This has the same optimial solution $x_1 = 0, x_2 = 2/3$ and optimzal value $1/3$ as the full problem above.

Bibliography

[1] Andrei A Agrachev and Yuri Sachkov. *Control Theory from the Geometric Viewpoint.* Springer, 2004. ISBN: 978-3-540-21019-1. DOI: 10/dw8x.

[2] Timothy D. Barfoot. *State Estimation for Robotics.* Cambridge University Press, 2017. DOI: 10/ggmw5j.

[3] Sergio Blanes and Fernando Casas. *A Concise Introduction to Geometric Numerical Integration.* Monographs and Research Notes in Mathematics. Chapman and Hall/CRC, 2016. DOI: 10.1201/b21563.

[4] Manfredo do Carmo. *Riemannian Geometry.* 1992.

[5] Gregory S. Chirikjian. *Stochastic Models, Information Theory, and Lie Groups, Volume 1.* Birkhäuser Boston, 2009. DOI: 10/bsjnxd.

[6] Gregory S. Chirikjian. *Stochastic Models, Information Theory, and Lie Groups, Volume 2.* Birkhäuser, 2012. DOI: 10/ctqkdh.

[7] Frank Dellaert and Michael Kaess. "Factor Graphs for Robot Perception". In: *Foundations and Trends in Robotics* 6.1-2 (2017). DOI: 10.1561/2300000043.

[8] John B. Fraleigh. *A first course in abstract algebra.* Pearson, 2014. ISBN: 978-0-3211-7340-9.

[9] Pieter van Goor, Tarek Hamel, and Robert Mahony. "Equivariant Filter (EqF)". In: *arXiv:2010.14666 [cs, eess]* (Oct. 27, 2020). arXiv: 2010.14666. URL: http://arxiv.org/abs/2010.14666 (visited on 10/29/2020).

[10] JW Grizzle and SI Marcus. "The structure of nonlinear control systems possessing symmetries". In: *IEEE Transactions on Automatic Control* 30.3 (1985), pp. 248–258. ISSN: 15582523. DOI: 10/c7mnk4. URL: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1103927.

[11] Roger Howe. "Very Basic Lie Theory". In: *The American Mathematical Monthly* 90.9 (1983), pp. 600–623. DOI: 10/dm9qv9.

[12] Minh Duc Hua et al. "Implementation of a nonlinear attitude estimator for aerial robotic vehicles". In: *IEEE Transactions on Control Systems Technology* 22.1 (2014), pp. 201–213. ISSN: 10636536. DOI: 10/f5ndx3.

[13] Taeyoung Lee. "Global Exponential Attitude Tracking Controls on SO(3)". In: *IEEE Transactions on Automatic Control* 60.10 (2015), pp. 2837–2842. DOI: 10.1109/TAC.2015.2407452.

[14] Robert Mahony, Tarek Hamel, and Jean-Michel Pflimlin. “Nonlinear Complementary Filters on the Special Orthogonal Group”. In: *IEEE Transactions on Automatic Control* 53.5 (2008), pp. 1203–1218. DOI: 10/bt4xd4.

[15] Jorge J. Moré. “The Levenberg-Marquardt algorithm: Implementation and theory”. en. In: *Numerical Analysis*. Ed. by G. A. Watson. Vol. 630. Series Title: Lecture Notes in Mathematics. Berlin, Heidelberg: Springer Berlin Heidelberg, 1978, pp. 105–116. ISBN: 978-3-540-08538-6 978-3-540-35972-2. DOI: 10.1007/BFb0067700. (Visited on 04/30/2021).

[16] Jorge Nocedal and Stephen J. Wright. *Numerical optimization*. 2nd ed. Springer, 2006.

[17] Joan Solà, Jeremie Deray, and Dinesh Atchuthan. *A micro Lie theory for state estimation in robotics*. 2020. arXiv: 1812.01537 [cs.RO].