mαcrϕ@ufmg

Universidade Federal de Minas Gerais
Graduate Program in Electrical Engineering
MACRO Research Group - Mechatronics, Control, and Robotics

CONSTRUCTIVE VECTOR FIELDS FOR PATH FOLLOWING IN MATRIX LIE GROUPS

Felipe Bartelt de Assis Pessoa

Belo Horizonte, Brazil
2025

Felipe Bartelt de Assis Pessoa

CONSTRUCTIVE VECTOR FIELDS FOR PATH FOLLOWING IN MATRIX LIE GROUPS

Dissertation submitted to the Graduate Program in Electrical Engineering of Universidade Federal de Minas Gerais, in partial fulfillment of the requirements for the degree of Master in Electrical Engineering.

Advisor: Luciano Cunha de Araújo Pimenta
Co-Advisor: Vinicius Mariano Gonçalves

Belo Horizonte, Brazil
2025

"Wer sich tief weiß, bemüht sich um Klarheit; wer der Menge tief scheinen möchte, bemüht sich um Dunkelheit. Denn die Menge hält alles für tief, dessen Grund sie nicht sehen kann: sie ist so furchtsam und geht so ungern ins Wasser."
(Friedrich Nietzsche)

Agradecimentos

Agradeço primeiramente aos meus pais, Renata e João Bosco, pelo apoio e incentivo durante toda a minha vida. O orgulho de vocês me faz questionar quanto à eficácia dos circuitos de consagração social de Bourdieu. À minha futura esposa, Giovanna, que, além do amor incondicional e companheirismo, mais uma vez adquiriu um novo conhecimento em uma área que não lhe é útil. Após todas as nossas conversas a respeito do meu trabalho, agora você consegue não somente "gastar" termos e conhecimentos de controle e robótica, mas também de matemática abstrata.

Aos amigos do LCR, que já estão quase convencidos a trabalhar com grupos de Lie. Em especial ao Brener, tanto pelas inúmeras pausas e discussões sobre o trabalho, quanto pelo queijo e doce de leite (aqui acredito agradecer também em nome do Luciano), e aos outros companheiros do melhor CBA da história, Aclécio e Daniel, que tornaram a experiência do primeiro congresso, com certeza, única.

Aos amigos do DAicex, especialmente ao Ping e Luiz, pela grande contribuição nas discussões sobre abstrações e na revisão de provas matemáticas. Aos amigos do GEC, que tornaram a experiência deveras conturbada no mestrado mais leve e divertida. Aos amigos de longa data, que, além do apoio, também contribuíram opinando sobre o trabalho, principalmente na legibilidade matemática e na lógica de implementação.

Ao professor Luciano, pela orientação e compreensão, confiando mais na escrita dos artigos e na conclusão da dissertação do que eu mesmo. Ao professor Vinicius, que além da orientação, fomentou minha paixão pela matemática desde a graduação, sem a qual este trabalho certamente não existiria. Ao professor Leo Torres, pela solicitude e que, além de me ajudar a resolver diversos problemas, me presenteou com infindáveis e profícuas conversas sobre os mais diversos assuntos.

Por fim, agradeço aos demais professores, funcionários e técnicos da UFMG, que de forma direta ou indireta participaram e contribuíram para essa jornada de dois (ou oito) anos.

Abstract

This work introduces a novel vector field strategy for controlling systems on connected matrix Lie groups, ensuring both convergence to and circulation around a curve defined on the group. Our approach generalizes the framework presented in Rezende et al. (2022) and reduces to it when applied to the Lie group of translations in Euclidean space. Building upon the key properties from Rezende et al. (2022), such as the orthogonality between convergent and circulating components, we extend these results by exploiting additional Lie group properties. Notably, our method ensures that the control input is non-redundant, matching the dimension of the Lie group rather than the potentially larger dimension of the embedding space. This leads to more practical control inputs in certain scenarios. A significant application of this strategy is in controlling systems on SE(3), where the non-redundant input corresponds to the mechanical twist of the object. This makes the method particularly well-suited for controlling systems with both translational and rotational freedom, such as omnidirectional drones. We also present an efficient algorithm for computing the vector field in this context. Furthermore, the strategy is applied as a high-level kinematic controller in a collaborative manipulation task, where six agents manipulate a large object with unknown parameters in the Lie group $\mathbb{R}^3 \times \mathrm{SO}(3)$. A low-level dynamic adaptive controller guarantees that the velocity tracking error between the system and the kinematic controller output converges to zero, a result supported by theoretical proofs. Simulation results validate the effectiveness of the proposed method in both the kinematic scenario and the integration of kinematic and dynamic controllers.

Keywords: autonomous systems; guidance navigation and control; tracking; asymptotic stabilization; vector fields; Lie groups; adaptive control; collaborative manipulation.

Resumo

Este trabalho apresenta uma nova estratégia de campos vetoriais para controlar sistemas em grupos de Lie matriciais conexos, garantindo tanto a convergência quanto a circulação ao longo de uma curva definida no grupo. Esta abordagem generaliza a estratégia apresentada em Rezende et al. (2022) e se reduz a ela quando aplicada ao grupo de Lie de translações no espaço euclidiano. Baseando-se nas propriedades chave de Rezende et al. (2022), como a ortogonalidade entre os componentes de convergência e de circulação, estenderam-se esses resultados explorando propriedades adicionais dos grupos de Lie. Em especial, este método garante que a entrada de controle seja não redundante, correspondendo à dimensão do grupo de Lie ao invés da dimensão potencialmente maior do espaço em que está mergulhado. Isso resulta em entradas de controle mais práticas em certos cenários. Uma aplicação significativa dessa estratégia é no controle de sistemas em SE(3), no qual a entrada não redundante corresponde ao heligiro do objeto. Isso torna o método particularmente adequado para controlar sistemas com liberdade tanto translacional quanto rotacional, como drones omnidirecionais. Apresenta-se também um algoritmo eficiente para calcular o campo vetorial nesse contexto. Além disso, a estratégia é aplicada como um controlador cinemático de alto nível em uma tarefa de manipulação colaborativa, na qual seis agentes manipulam um objeto grande com parâmetros desconhecidos no grupo de Lie $\mathbb{R}^3 \times \mathrm{SO}(3)$. Um controlador dinâmico adaptativo de baixo nível garante que o erro de rastreamento de velocidade entre o sistema e a saída do controlador cinemático convirja para zero, resultado que é comprovado teoricamente. Os resultados das simulações validam a eficácia do método proposto tanto no cenário cinemático quanto na integração de controladores cinemáticos e dinâmicos.

Palavras-chave: sistemas autônomos; direcionamento, navegação e controle; rastreamento; estabilização assintótica; campos vetoriais; grupos de Lie; controle adaptativo; manipulação colaborativa.

Contents

List of Figures

List of Tables

Acronyms

DoF Degree of Freedom.

EC Element-Curve.

EE Element-Element.

MRAC Model Reference Adaptive Control.

RGB Red, Green, and Blue.

UAV Unmanned Aerial Vehicle.

Notation

G	Lie group.
$\mathrm{GL}(n,\mathbb{R})$, $\mathrm{GL}(n,\mathbb{C})$	General Linear group of order n over the real numbers, General Linear group of order n over the complex numbers.
$\mathrm{SL}(n,\mathbb{R})$, $\mathrm{SL}(n,\mathbb{C})$	Special Linear group of order n over the real numbers, Special Linear group of order n over the complex numbers.
$\mathrm{O}(n)$, $\mathrm{SO}(n)$	Orthogonal group of order n, Special Orthogonal group of order n.
$\mathrm{E}(n)$, $\mathrm{SE}(n)$	Euclidean group of order n, Special Euclidean group of order n.
$\mathrm{T}(n)$	Translation group of order n. Matrix representation of $(\mathbb{R}^n,+)$.
$\mathrm{ISE}(n)$	Independent Special Euclidean group of order n, isomorphic to $\mathbb{R}^n \times \mathrm{SO}(n)$.
$\mathrm{Sp}(2n)$	Real Symplectic group of order $2n$.
$\mathrm{O}(p,q)$, $\mathrm{SO}(p,q)$	Indefinite Orthogonal group of signature (p,q), Indefinite Special Orthogonal group of signature (p,q).
$\mathrm{O}(1,3)$, $\mathrm{SO}(1,3)$, $\mathrm{SO}^+(1,3)$	full Lorentz group, special Lorentz group, proper orthochronous Lorentz group.
$\mathfrak{g}$	Lie algebra of a Lie group G.
$\mathrm{Lie}(G)$	Space of left-invariant, or right-invariant, vector fields on a Lie group G. Isomorphic to $\mathfrak{g}$.
$\mathfrak{gl}(n,\mathbb{R})$, $\mathfrak{gl}(n,\mathbb{C})$	Lie algebras of $\mathrm{GL}(n,\mathbb{R})$ and $\mathrm{GL}(n,\mathbb{C})$ respectively.
$\mathfrak{sl}(n,\mathbb{R})$, $\mathfrak{sl}(n,\mathbb{C})$	Lie algebras of $\mathrm{SL}(n,\mathbb{R})$ and $\mathrm{SL}(n,\mathbb{C})$ respectively.
$\mathfrak{so}(n)$	Lie algebra of $\mathrm{O}(n)$ and $\mathrm{SO}(n)$.

$\mathfrak{se}(n)$	Lie algebra of $\mathrm{E}(n)$ and $\mathrm{SE}(n)$.
$\mathfrak{t}(n)$	Lie algebra of $\mathrm{T}(n)$.
$\mathfrak{ise}(n)$	Lie algebra of $\mathrm{ISE}(n)$.
$\mathfrak{sp}(2n)$	Lie algebra of $\mathrm{Sp}(2n)$.
$\mathfrak{so}(p,q)$	Lie algebra of $\mathrm{O}(p,q)$ and $\mathrm{SO}(p,q)$.
$\mathfrak{so}(1,3)$	Lie algebra of $\mathrm{O}(1,3)$, $\mathrm{SO}(1,3)$ and $\mathrm{SO}^+(1,3)$.
$\mathcal{S}$	Isomorphism that maps $\mathbb{R}^m$ to $\mathfrak{g}$.
$\mathcal{S}^{-1}$	Inverse isomorphism that maps $\mathfrak{g}$ to $\mathbb{R}^m$, such that $\mathcal{S}^{-1}\big(\mathcal{S}(\boldsymbol{\xi})\big) = \boldsymbol{\xi}$.
$\widehat{\mathcal{S}}$	Canonical isomorphism that maps $\mathbb{R}^3$ to $\mathfrak{so}(3)$.
$\mathcal{T}$	Isomorphism that maps $\mathrm{T}(n)$ to $\mathbb{R}^n$.
$\mathbf{e}_1, \ldots, \mathbf{e}_n$	Canonical basis elements of $\mathbb{R}^n$. The columns of an $n \times n$ identity matrix.
$\widehat{\mathbf{e}}_1, \widehat{\mathbf{e}}_2, \widehat{\mathbf{e}}_3$	Canonical basis elements of $\mathbb{R}^3$. The columns of a 3×3 identity matrix.
$\mathbf{E}_1, \ldots, \mathbf{E}_n$	Basis elements of an n-dimensional Lie algebra.
$[\cdot, \cdot]$	Lie bracket.
T_pM	Tangent space of a manifold M at a point p.
ι	Identity element of a Lie group.
G_ι	Identity component of a Lie group G.
$\mathcal{L}_{\mathbf{X}}$	Left translation by $\mathbf{X}$.
$\mathcal{R}_{\mathbf{X}}$	Right translation by $\mathbf{X}$.
$\circ$, $\star$	Group operations.
$\cong$	Isomorphic. If G and H are isomorphic, then $G \cong H$.
$\times$	Direct product of two sets $G \times H$. Also used for the cross product of two vectors.
$\rtimes$	Semidirect product of two groups $G \rtimes H$.
$\oplus$	Direct sum of two vector spaces.
$\to$	Indicates a mapping from the domain of a function to its codomain. The notation $f : A \to B$ means that f maps elements from domain A to codomain B.
$\mapsto$	Indicates the mapping of an element from the domain of a function to its image in the codomain. The notation $a \mapsto b$ means that $f(a) = b$ for some function f.

$\hookrightarrow$	Denotes an inclusion map. The notation $f : A \hookrightarrow B$ implies that A is contained within B and is injectively mapped to B..
$\mathrm{L}[f](\mathbf{Z})$	L operator of a function f with argument $\mathbf{Z} \in G$. Maps a function $f : G \to \mathbb{R}$ into $\mathrm{L}[f] : G \to \mathbb{R}^{1\times m}$.
$\mathrm{L}_{\mathbf{V}}[\cdot](\cdot,\cdot), \mathrm{L}_{\mathbf{W}}[\cdot](\cdot,\cdot)$	L operator for a function of two variables, performing the variations on the first and second variable, respectively.
$\mathcal{Z}(\mathbf{Y})$	Part of the chain rule of the L operator for left-invariant functions.
$\Xi[\mathbf{G}]$	Maps a differentiable function $\mathbf{G} : \mathbb{R} \to G$ into a function $\Xi[\mathbf{G}] : \mathbb{R} \to \mathbb{R}^m$.
$\boldsymbol{\xi}$	Generalized twist.
Ψ	Vector field that guides a system along a curve in a Lie group.
$\boldsymbol{\xi}_T$	Tangent component of the vector field Ψ.
$\boldsymbol{\xi}_N$	Normal component of the vector field Ψ.
k_N, k_T	Gains for the normal and tangent components of the vector field Ψ, respectively. Both are continuous scalar functions.
$\widehat{D}(\mathbf{V}, \mathbf{W})$	Element-Element (EE-) distance between two Lie group elements $\mathbf{V}$ and $\mathbf{W}$.
$D(\mathbf{V})$	Element-Curve (EC-) distance between a Lie group element $\mathbf{V}$ and a curve $\mathcal{C}$.
$\mathcal{P}$	Set of all points that are either non-unique solutions or non-differentiable points of the EE-distance function.
$\mathcal{C}$	A curve in a Lie group.
$\mathbf{H}$	The system state, an element of a Lie group.
$\mathbf{H}_d(s)$	A curve parameterized by s in a Lie group.
$\mathbf{H}_d(s^*)$	The nearest point on a curve $\mathcal{C}$ to a system state $\mathbf{H}$.
$\boldsymbol{\xi}_d(s)$	The curve twist, $\boldsymbol{\xi}_d(s) = \Xi[\mathbf{H}_d](s)$.
$\mathbf{H}_d[i]$	A discretized curve in a Lie group at the i^{th} point.
$\mathbf{H}_d[i^*]$	The nearest point on a discretized curve to a system state $\mathbf{H}$.

$\mathbf{H}'_d[i]$	The derivative of a discretized curve at the i^{th} point.
Φ	A continuous and differentiable path in a Lie group.
$\boldsymbol{\chi}$	Pose of a rigid body represented as a tuple in $\mathbb{R}^3 \times \mathrm{SO}(3)$.
$\underline{x}$	Intermediate variable of x in the Heun's method.
$\dot{x}$	Time derivative of a time-dependent variable x.
$\mathrm{diag}(\cdot)$	Maps a vector into a diagonal matrix.
$\mathrm{blkdiag}(\cdot, \ldots, \cdot)$	Maps a set of matrices into a block-diagonal matrix.
$\mathrm{abs}(\cdot)$	Element-wise absolute value of a vector.
$\mathrm{rot}(\cdot)$	Extracts the rotation matrix from an element of SE(3) or ISE(3).
$\mathrm{tran}(\cdot)$	Extracts the translation vector from an element of SE(3) or ISE(3).
$\mathbf{I}_n, \mathbf{I}$	Identity matrix of dimension n. If the dimension is clear from the context, then $\mathbf{I}$ is used.
$\mathbf{I}_{p,q}$	A block-diagonal matrix containing a $p \times p$ identity matrix, and a $q \times q$ identity matrix multiplied by -1.
$\mathbf{0}_{n\times m}, \mathbf{0}$	Zero matrix, or vector, of dimension $n \times m$. If the dimension is clear from the context, then $\mathbf{0}$ is used.
$\mathbf{A}_i, \{\mathbf{A}\}_i$	The i^{th} column of a matrix $\mathbf{A}$.
$\mathbf{A}_{ij}, \{\mathbf{A}\}_{ij}$	The element at the i^{th} row and j^{th} column of a matrix $\mathbf{A}$.
$\mathbf{v}_i, \{\mathbf{v}\}_i$	The i^{th} element of a vector $\mathbf{v}$.
$.^\top$	Transpose of a matrix or vector.
$\mathrm{Log}(\cdot)$	Principal matrix logarithm.
$\log(\cdot)$	Matrix logarithm and scalar natural logarithm.
$\exp(\cdot)$	Matrix exponential and scalar exponential.
$\mathrm{tr}(\cdot)$	Matrix trace.
$\|\cdot\|$	Euclidean norm.
$\|\cdot\|_F$	Frobenius norm.
$\mathbb{R}$	Set of real numbers.
$\mathbb{R}_+$	Set of non-negative real numbers.

$\mathbb{R}^n$	Euclidean space of dimension n. The elements are n-dimensional column-vectors.
$\mathbb{R}^{n\times n}$	Set of $n \times n$ real matrices.
$\mathbb{R}_+^{n\times n}$	Set of $n \times n$ real matrices with no negative real eigenvalues.
$\mathbb{C}$	Set of complex numbers.
$\mathbb{C}^{n\times n}$	Set of $n \times n$ complex matrices.
$\mathbb{S}^n$	Unit sphere in $\mathbb{R}^{n+1}$.
$\langle \cdot, \cdot \rangle$	Inner product between two vectors.
$o(\cdot)$	Small-o notation.
∇	Gradient operator.
δ_{ij}	Kronecker delta.
$\emptyset$	Empty set.

1 Introduction

Physical motions exhibit a profound connection with Lie groups, appearing in diverse realms such as classical mechanics, quantum mechanics, and special relativity. This connection is not merely a matter of nomenclature but reflects an underlying and fundamental structure. While group operations are inherently interesting, the existence of a manifold equipped with a group structure – a Lie group – elevates this interest to a new level. Although abstract, the study of systems through the lens of Lie theory offers deeper insights into their intrinsic structure, enabling the development of more efficient and elegant control strategies.

The application of Lie groups to control theory aligns closely with geometric control theory (Bullo and Lewis, 2004), which seeks to unify the tools of differential geometry with control problems. Instead of relying on local Euclidean charts, geometric control operates directly within the inherent manifold's structure. Similarly, our objective in this work is to design a control strategy that leverages the Lie group structure of the system, embracing its geometric and algebraic properties.

At the core of this work lies the development of a vector field guidance strategy tailored to systems with an intrinsic matrix Lie group structure. Vector field-based approaches have proven versatile in controlling various robotic systems, providing a unified framework that integrates path planning, trajectory generation, and control (Goncalves et al., 2010; Yao et al., 2021; Rezende et al., 2022; Gao et al., 2022; Nunes et al., 2023; Yao et al., 2023; Chen and Zuo, 2025). This unified framework allows for constructing a path between an initial and final configuration, generating a guidance signal for the system, and ensuring

that the system follows the desired trajectory.

Our work builds upon the results of Rezende et al. (2022), which introduced a vector field guidance strategy in Euclidean space based on the parametric representation of curves. Using this as a foundation, we extend the strategy to systems represented as matrix Lie groups. To establish this connection, we summarize the key properties and results of Rezende et al. (2022), which serve as the groundwork for our generalization. Throughout the text, we use examples to illustrate and clarify the parallels and distinctions between the two approaches, ensuring the continuity and coherence of our contributions.

One particularly compelling application of extending vector fields to matrix Lie groups is the control of systems capable of simultaneous motion in both position and orientation, which can be addressed by considering the group SE(3) within our framework. Omnidirectional unmanned aerial vehicles (UAVs), such as those illustrated in Figure 1.1, exemplify such capabilities, and research in this domain is rapidly advancing (Kamel et al., 2018; Aboudorra et al., 2024; Hamandi et al., 2024). For instance, in Hamandi et al. (2021) a comprehensive review of multirotor designs is provided, identifying critical factors that enable omnidirectional motion. These capabilities are particularly relevant for tasks requiring complex maneuvers, such as tracking paths that involve intricate trajectories. This is especially important when UAVs are equipped with tools like drills, where precise positioning is needed to achieve target poses while avoiding environmental collisions. Our approach facilitates such tasks by controlling the linear and angular velocities of the UAV. Additionally, another notable application of SE(3) is in robotic manipulators, where our method can control the end effector's linear and angular velocities.

Figure 1.1: One application of our strategy is the control of omnidirectional drones, like (left) a Voliro drone (image taken from `https://voliro.com`) and (right) the design from Hamandi et al. (2024). Since they accept arbitrary 6 DoF twists, our methodology can be used to control them to track arbitrary differentiable paths in SE(3).

Although these applications demonstrate the versatility of our approach, it is important to emphasize that our primary focus is not on specific use cases but on the theoretical

foundation itself. As will become evident, our generalization not only enables the use of vector fields on Lie groups but also reveals that the vector field strategy in Euclidean space offers more flexibility than previously recognized. To encapsulate our intent, we draw on the perspective of Francesco Bullo, who states that "the areas of overlap between mechanics and control possess the sort of mathematical elegance that makes them appealing to study, independently of applications." (Bullo and Lewis, 2004).

To illustrate the potential of our generalization, we provide two application scenarios. The first is a kinematic control simulation of a system in SE(3), where the goal is to converge to and circulate along a predefined curve within the group. This simulation offers a concrete example of how our theoretical results can be practically applied and visualized. Within the kinematic scenario, we also simulate a system in $\mathrm{SO}^+(3,1)$. While this does not provide a visual representation of the strategy, it demonstrates its generality.. The second scenario involves a collaborative simulation in which six agents manipulate a cylindrical object with unknown properties to track a target curve. In this case, we work within the group $\mathbb{R}^3 \times \mathrm{SO}(3)$, where translation and rotation are treated as independent motions. Due to the object's unknown parameters, we employ an adaptive control strategy to estimate these properties and achieve control. Here, our vector field strategy serves as a higher-level controller, providing desired velocity inputs to a lower-level dynamic controller responsible for tracking these velocities. This simulation highlights the adaptability of our approach to dynamic scenarios, particularly those requiring parameter estimation and adaptation.

Although not essential to understanding this work, we provide a brief background on its development. Motivated by an interest in nonlinear dynamic control techniques from graduate coursework, we sought to extend previous assignments and explore their broader applications. During this process, we encountered Culbertson et al. (2021), which addressed adaptation in robotic manipulation. This led us to investigate whether their approach could be adapted to a vector field framework based on Rezende et al. (2022). Building on this, we explored the inclusion of orientation, culminating in a formulation in $\mathbb{R}^3 \times \mathrm{SO}(3)$, later published in Pessoa and Pimenta (2024). Observing its properties, we identified a broader generalization for systems in matrix Lie groups, leading to the more general vector field guidance strategy presented here, currently under review at *Automatica*.

The previous disclaimer not only provides insight into how this work evolved but also explains certain modifications made to this text. Here, we focus on the generalized vector field approach for systems represented as matrix Lie groups, treating Pessoa and Pimenta (2024) as an application of this broader framework. Consequently, our presentation adopts a non-chronological order. We begin with the more general case, as it offers a clearer conceptual foundation, and then demonstrate its application to specific scenarios. Some portions of the text from Pessoa and Pimenta (2024) have been rewritten to better fit the

context of this work and may differ from the original. Furthermore, we expand on certain derivations and proofs from both works to provide deeper insights.

1.1 Summary of contributions

The main contributions of this work can be summarized as follows:

- Development of a novel vector field guidance strategy applicable to systems with an inherent matrix Lie group structure;
- Implementation framework for SE(3) systems, providing all necessary tools for practical application of the proposed strategy;
- Validation through kinematic simulations in SE(3) and $\mathrm{SO}^{+}(3,1)$, demonstrating the theoretical results and their practical implications;
- Design of an adaptive control strategy for collaborative simulations in $\mathbb{R}^3 \times \mathrm{SO}(3)$, where the vector field guidance strategy generates reference velocities for dynamic control.

1.2 Publications

In this section we highlight the publications that culminated into the present work

- **Pessoa, F. B. A.**; Pimenta, L. C. A. . Vector Field Based Adaptive Control for Collaborative Manipulation. In: **XXV Congresso Brasileiro de Automática**, 2024, Rio de Janeiro. Anais do XXV Congresso Brasileiro de Automática, 2024
- **Bartelt, F.**; Gonçalves, V. M.; Pimenta, L. C. A. . Constructive Vector Fields for Path Following in Matrix Lie Groups. **Submitted to Automatica**, 2025.

1.3 Structure of the text

This work is organized as follows:

- Chapter 2: Literature Review
  We review the formulation of vector fields and the application of Lie theory across various domains.
- Chapter 3: Theoretical Background
  This chapter revisits the vector field guidance strategy in Euclidean space, provides the necessary background on Lie groups and Lie algebras, and introduces the foundational concepts of adaptive control.

- Chapter 4: Vector Fields in Matrix Lie Groups
  We present the generalization of the vector field guidance strategy to matrix Lie groups, detailing its application to kinematic control in exponential Lie groups and providing the necessary tools for implementation in SE(3).

- Chapter 5: Collaborative Manipulation
  This chapter introduces the system model, the lower-level controller, and the adaptive control strategy for a collaborative manipulation scenario.

- Chapter 6: Results and Discussion
  We present the details and results of simulations for both explored scenarios.

- Chapter 7: Conclusion and Future Work
  The text concludes with a summary of the findings and a discussion of potential directions for future research.

2

Literature Review

This chapter provides a literature review to contextualize our work. Two key topics sustain our research: (1) vector field guidance and (2) applications of Lie theory. We examine the relevant literature in each domain separately.

2.1 Vector field guidance

Vector field-based strategies are widely employed due to their ability to seamlessly integrate path planning, trajectory planning, and control. These steps are typically described as follows: *path planning*, which involves constructing a collision-free path in the configuration space given an initial and final configuration; *trajectory planning*, where a time-dependent curve is generated based on the path to serve as the robot's reference trajectory; and *control*, where a control law is designed to ensure the robot closely tracks the reference trajectory (Rimon and Koditschek, 1992).

In the context of control theory, a vector field assigns a velocity vector to each point in the workspace or configuration space, depending on the method. This vector field is constructed to guide the robot toward its goal. Consequently, the three steps above are inherently embedded in vector field guidance: the velocity vector produced by the vector field serves as the robot's control input, and integrating the robot's dynamics yields a collision-free path. The primary distinction between different vector field guidance approaches lies in the methodology used to construct the vector field. This section reviews key contributions to the field.

Initial work in vector field guidance focused on potential fields, where the vector field is derived as the gradient of a potential function. These methods typically involve: (1) an *attractive potential field* to ensure convergence to the desired configuration; and (2) a *repulsive potential field* to avoid collisions with obstacles. The potential field method was first introduced in Khatib (1985), in which the primary goal was to enable online collision avoidance while solving a regulation problem. However, a significant limitation of this approach is the occurrence of undesired local minima, which can trap the robot away from its goal.

To address this issue, the concept of a *navigation function* was introduced in Rimon and Koditschek (1992). Navigation functions guarantee the existence of a single local minimum and ensure bounded torques for the robot's dynamics. However, this approach requires complete knowledge of the configuration space, and while an algorithm for constructing navigation functions is proposed by the authors, their computation remains challenging. Expanding on this work, in Conn and Kam (1998) collision avoidance in environments with moving obstacles is tackled. By making minimal assumptions, the authors demonstrated that the topology of a star-shaped configuration space remains invariant during motion. This invariance allows the moving obstacle problem to be addressed globally by continuously deforming the stationary solution at a fixed time over the time interval of motion. Nevertheless, like its predecessor, this method necessitates full knowledge of the configuration space. Another approach, presented in (Pimenta et al., 2006), relies on harmonic functions to construct potential fields without local minima. However, this method suffers from scalability issues, as higher dimensions significantly increase the problem's complexity. These potential field-based approaches are considered classical methodologies in vector field guidance, as discussed in foundational texts such as Choset et al. (2005, p. 77) and Spong et al. (2020, p. 299).

In multi-agent scenarios, achieving a specific geometric pattern, or desired formation, is often a primary objective (Chaimowicz et al., 2005; Mong-ying A. Hsieh and Kumar, 2006; Pimenta et al., 2007). For example, the strategy proposed in Chaimowicz et al. (2005) relies on constructing patterns using interpolated radial basis functions, where the resulting vector field was derived as the gradient of this interpolation. Individual agents were assigned repulsive fields to ensure proper distribution and avoid overlap. Building on this, in Mong-ying A. Hsieh and Kumar (2006) this strategy was refined by improving collision avoidance between agents and providing convergence proofs for the formation. In a different approach inspired by fluid dynamics, a method for generating two-dimensional geometric patterns for robot swarms was developed (Pimenta et al., 2007). By modeling agent interactions using principles of hydrodynamics, the vector field was computed through finite element solutions of Laplace's equation, akin to simulating fluids in electrostatic fields. Besides providing a decentralized control of the formation, this method also incorporated static obstacle avoidance.

A more general problem in obstacle-free environments is addressed in Goncalves et al. (2010), which proposes a vector field strategy for robot navigation along time-varying curves in n-dimensional Euclidean space. Although this also applies to formation problems, it encompasses both path and trajectory tracking. The target curve is implicitly defined as the intersection of zero-level surfaces. Specifically, a one-dimensional curve $\mathcal{C}$ embedded in $\mathbb{R}^n$ is described as the intersection of $(n-1)$-dimensional surfaces, with level sets $\alpha_i : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}$, such that $\mathcal{C} = \{[\mathbf{x}^\top,\ t]^\top | \alpha_i(\mathbf{x}, t) = 0 \, \forall \, i \leq n-1\}$. However, computing these surfaces remains a challenging task, and this methodology may still suffer from singularities.

An extension of this work is presented in Yao et al. (2021), which introduces a singularity-free vector field capable of handling self-intersecting curves. These improvements are achieved by projecting the curve into a higher-dimensional space. Despite these advances, challenges related to computing zero-level surfaces persist. Further, in Yao et al. (2023), the method of Goncalves et al. (2010) is generalized to systems embedded in smooth Riemannian manifolds. A notable contribution of this work is the topological analysis of convergence and stability for the vector field methodology in such contexts. It is proven that global convergence in n-dimensional Euclidean space is unattainable, as it inevitably leads to at least one singular point.

Curves are often more intuitive and easier to describe using parametric representations, a concept also explored in vector field guidance strategies. For example, parametric representations of static, possibly self-intersecting curves, alongside contraction theory, were employed in Wu et al. (2018) to develop vector field strategies for systems embedded in manifolds. In Rezende et al. (2022), parametric representations of curves were applied to address time-varying curves in n-dimensional Euclidean space, with the gradient of the Euclidean distance guiding the robot toward the curve. This approach was further extended in Nunes et al. (2022) to incorporate collision avoidance with static and dynamic obstacles.

In this work, the vector field strategy from Rezende et al. (2022) is generalized for application to matrix Lie groups. A detailed review of this approach is presented in Section 3.1 to facilitate connections between our generalization and the original methodology. Key properties of the approach that ensure convergence and circulation around the target curve are outlined, forming the basis for our generalization. Notably, only static curves are considered, thereby excluding the time-varying aspect of the original work. The proposed generalization is also comparable to the work in Yao et al. (2023), which focuses on smooth Riemannian manifolds – structures that encompass matrix Lie groups. This renders their work broader in scope compared to ours. However, this broader approach does not exploit the group structure, which can eliminate redundant control inputs. For example, in the case of the manifold SO(3) of rotation matrices, the control input is expressed as the time derivative of a rotation matrix, resulting in a dimension of 9. In contrast, leveraging

the group structure reduces the control input to a dimension of 3, corresponding to the angular velocity. Thus, our work is more specific and arguably more intuitive. Another advantage of our approach is its use of parametric representations of curves, which are simpler to handle compared to the implicit representations in Yao et al. (2023).

2.2 Lie theory applications

Brief allusions to Lie theory are common in control theory and robotics literature. Yet, most books and papers seldom explore what makes Lie groups and Lie algebras truly remarkable. As a result, readers might easily assume that terms like 'Lie derivatives', 'Lie group', or 'special orthogonal group' are merely naming conventions. To call something a 'Lie group' is to endow it not merely with a name, but with a wealth of profound properties that underpin its remarkable utility. In this section, a spectrum of applications across diverse fields is explored, unveiling the profound elegance and utility of this theory.

In Murray et al. (1994), the use of Lie groups and Lie algebras is extensive, ranging from the definition of kinematic chains to Euler-Lagrange equations. Although more focused on rigid body motions, the text introduces the concepts of Lie theory, and much of the underlying mathematics of robot motion depends on understanding these concepts. Geometric control theory shares connections with Lie theory since the former is based on differential geometry. As such, in Bullo and Lewis (2004), a more in-depth study on Lie groups and Lie algebras is presented, relying heavily on Lie derivatives and Lie brackets to develop the theory of geometric control.

In the scope of control theory, different applications using the language of Lie groups are found. Matrix Lie groups have been used in invariant extended Kalman filters and to prove a strong property of asymptotic convergence around any trajectory of the system (Barrau and Bonnabel, 2017). In McCarthy and Nielsen (2020), global synchronization of a network of agents in a matrix Lie group is achieved. The control law is smooth, distributed, nonlinear, and discrete, and for specific groups, it is shown to guarantee exponential rate synchronization. A hybrid control scheme for trajectory tracking on SO(3) and SE(3) is proposed in Wang and Tayebi (2022). This scheme uses a potential function on an augmented space to guarantee global asymptotic stability. In Duong et al. (2024), the Hamiltonian formulation of robot dynamics is generalized to a port-Hamiltonian formulation on matrix Lie groups. This generalization is then used to design a neural ordinary differential equation model to enable data-driven learning of robot dynamics. Furthermore, a control law is proposed to ensure trajectory tracking based on the learned dynamics.

More focused on the computer science field, Lie theory appears in many works on machine learning and pattern recognition. Lie theory has been used to simplify the real-time three-dimensional visual tracking of complex objects (Drummond and Cipolla,

2002). This methodology has also been showcased to enable visual-servoing of a robot manipulator. In Vemulapalli et al. (2014), a novel representation for human skeletons is proposed using a Lie group of many direct products of SE(3). This representation is shown to perform better than the state of the art for human action recognition. The survey in Lu and Li (2020) highlights the advantages of the innovative learning paradigm of Lie group machine learning, as well as its application to image processing and neuromorphic synergy learning.

Lie theory has found extensive applications across diverse fields of study. For instance, in numerical analysis, the Butcher group – a Lie group – is useful for studying ordinary differential equations via the Runge-Kutta method (Bogfjellmo and Schmeding, 2017). In Hamiltonian mechanics, it has been employed to analyze complex dynamical systems, as demonstrated in Hamburger (2009), with applications including the harmonic oscillator and Kepler's problem. Within quantum physics, Lie groups help explain unusual transport phenomena observed in physical systems (Ilievski et al., 2021). The Lorentz group and the special orthogonal group feature prominently in discussions of quantum gravity (Dreyer, 2003; Kapec et al., 2017).

3

Theoretical Background

In this chapter, we revisit the vector field strategy in Euclidean space based on a curve parametrization (Rezende et al., 2022), as it provides valuable connections between both works. Additionally, we present fundamental concepts of Lie groups and Lie algebras, alongside basic principles of adaptive control, which are essential for developing our extension.

3.1 Vector field in Euclidean space

For clarity, we revisit the vector field strategy presented in Rezende et al. (2022). Since our work extends this previous approach, this review will help establish direct connections between both works and highlight the core aspects of our contributions. The primary goal of the authors in Rezende et al. (2022) is to develop an artificial n-dimensional vector field that guides system trajectories toward a predefined curve and ensures circulation around it. A key element of this formulation is the definition of a distance function with essential properties. As their work focuses solely on Euclidean space, they adopt the Euclidean distance for the vector field computation, which is derived using a parametric representation of the curve.

We summarize the main steps for constructing this vector field, emphasizing the most critical properties. Although the original paper addresses time-varying curves, we limit our discussion to the static portion of the methodology. The authors consider a system

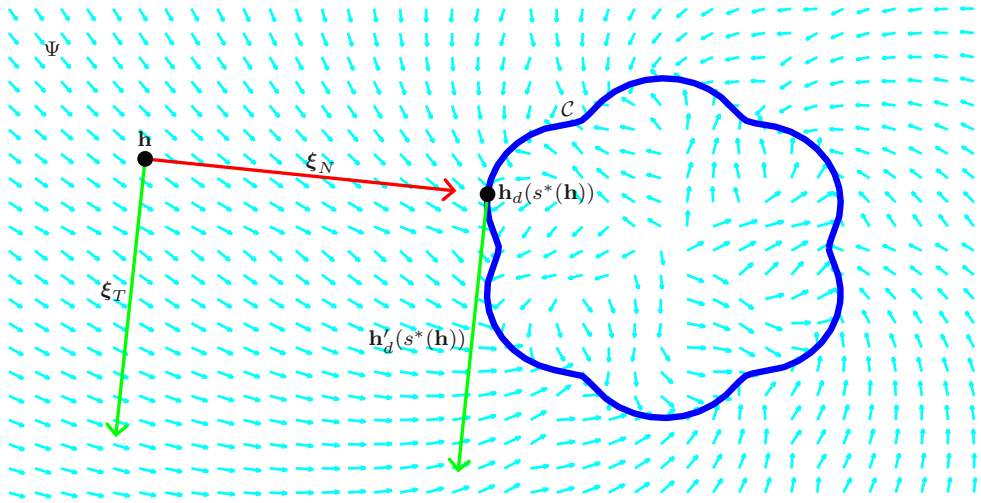


Figure 3.1: Example showing the vector field and the components for a point $\mathbf{h} \in G$ and curve $\mathcal{C}$.

modeled as the following single integrator

$$\dot{\mathbf{h}} = \boldsymbol{\xi}, \tag{3.1}$$

where $\mathbf{h} \in \mathbb{R}^m$ represents the system state, and $\boldsymbol{\xi} \in \mathbb{R}^m$ denotes the system input. The objective is to compute a vector field $\Psi : \mathbb{R}^m \to \mathbb{R}^m$ such that, if the system input is equal to the vector field, the system trajectories converge to and follow the target curve $\mathcal{C}$, for which a parametrization is given by $\mathbf{h}_d(s)$. Despite relying on a parametric representation of the curve, it is important to note that the resulting computations are independent of the specific parametrization chosen.

The authors define their distance function D as the Euclidean distance between the system's current state and the nearest point on the curve, i.e.,

$$D(\mathbf{h}) \triangleq \min_s \widehat{D}(\mathbf{h}, \mathbf{h}_d(s)) = \min_s \|\mathbf{h} - \mathbf{h}_d(s)\|. \tag{3.2}$$

In this context, we denote s^* as the optimal parameter such that $\mathbf{h}_d(s^*(\mathbf{h}))$ is the closest point on the curve to the current state.

Next, the authors introduce two components of their vector field: the *normal* component $\boldsymbol{\xi}_N$, responsible for convergence, and the *tangent* component $\boldsymbol{\xi}_T$, which ensures circulation. The resulting expression for the vector field is

$$\Psi(\mathbf{h}) = k_N(\mathbf{h})\boldsymbol{\xi}_N(\mathbf{h}) + k_T(\mathbf{h})\boldsymbol{\xi}_T(\mathbf{h}), \tag{3.3}$$

where k_N and k_T are gains, dependent on the system state, that balance the predominance of the normal and tangent components. This vector field strategy is illustrated in Figure 3.1.

The normal component, $\boldsymbol{\xi}_N$, is naturally taken as the negative gradient of the distance function, due to the use of Euclidean distance:

$$\boldsymbol{\xi}_N = -(\nabla D)^\top. \tag{3.4}$$

There is a key aspect of the normal component that is crucial for the convergence proof using Lyapunov stability theory: the fact that the time derivative of the distance function can be expressed as $\dot{D} = -\boldsymbol{\xi}_N^\top \boldsymbol{\xi}$. We emphasize the significance of this feature, as it will play an important role in our extension. In our approach, the normal component is similarly constructed by identifying the term that arises when differentiating the distance function.

Next, we address the tangent component. This component is solely related to the target curve and is defined as the tangent vector at the nearest point on the curve, i.e.,

$$\boldsymbol{\xi}_T(\mathbf{h}) = \frac{d}{ds}\mathbf{h}_d(s)|_{s=s^*(\mathbf{h})}. \tag{3.5}$$

A noteworthy property of both components is that they are orthogonal to each other, which is essential in the proof of convergence for this algorithm.

In the Lyapunov stability proof, the final result shows that the time derivative of D is negative semidefinite. The proof is then completed using two other essential properties: the fact that the distance function has no local minima outside the curve, and the fact that the gradient of this function never vanishes. With these features, the authors demonstrate that if the system trajectories follow the vector field, the system will converge to and circulate around a predefined curve. A summary of these key features is as follows:

(a) The time derivative of the distance function is the opposite of the dot product between the so called *normal* component and the system control input;
(b) The *normal* and *tangent* components are orthogonal to each other;
(c) The distance function has no local minima outside the target curve. Furthermore, whenever the distance function is differentiable, its gradient never vanishes.

In our generalization, we will incorporate and build upon these features.

3.2 Lie groups and Lie algebras

This section introduces *Lie groups*, smooth manifolds that are also groups, and *Lie algebras*, vector spaces with additional structure. Along with providing examples of both, we include key concepts relevant to our work. The content integrates ideas from the following texts: Lee (2012); Gallier and Quaintance (2020); Hall (2015); Duistermaat and Kolk (1999), aiming to enhance clarity and accessibility. Additionally, we introduce original concepts

and useful proofs not, to the best of our knowledge, available in the literature. We begin with the general definition of a Lie group.

3.2.1 Lie groups

Definition 3.1 (Lie group)**.** A *Lie group* is a smooth manifold G that satisfies the following properties:

i G is a group (with identity element denoted ι);
ii G is a topological group, meaning the group operation and the inverse map are continuous. Additionally, the group operation $\circ : G \times G \to G$ and the inverse map $\cdot^{-1} : G \to G$ are smooth.

□

Mappings between Lie groups are fundamental and play a significant role in our extension, particularly for relating our approach to the vector field strategy in Euclidean space. These mappings often possess additional properties and are defined as follows:

Definition 3.2. If G and H are Lie groups, a *homomorphism* $\mathcal{H} : G \to H$ is a smooth map (between manifolds G and H) that is also a group homomorphism. Specifically, if $\circ$ and $\star$ are the group operations in G and H, respectively, then

$$\mathcal{H}(\mathbf{X} \circ \mathbf{Y}) = \mathcal{H}(\mathbf{X}) \star \mathcal{H}(\mathbf{Y}) \,\forall\, \mathbf{X}, \mathbf{Y} \in G.$$

If this map is also a diffeomorphism (with the inverse map $\mathcal{H}^{-1}$ being a homomorphism), then $\mathcal{H}$ is called an *isomorphism*. In this case, G and H are said to be *isomorphic*, denoted by $G \cong H$. □

Two important isomorphisms in Lie groups, critical for the concept of Lie algebras, are the translation isomorphisms defined below:

Definition 3.3. Given a Lie group G, the left translation $\mathcal{L}_{\mathbf{X}} : G \to G$ and the right translation $\mathcal{R}_{\mathbf{X}} : G \to G$ are defined for any $\mathbf{X} \in G$ as follows:

$$\mathcal{L}_{\mathbf{X}}(\mathbf{Y}) = \mathbf{X} \circ \mathbf{Y}, \tag{3.6}$$

$$\mathcal{R}_{\mathbf{X}}(\mathbf{Y}) = \mathbf{Y} \circ \mathbf{X}. \tag{3.7}$$

Since the group operation and inverse map are smooth, the left and right translations are diffeomorphisms, making them isomorphisms. □

Matrix Lie Groups

In general, Lie groups that can be represented by matrices are called *matrix Lie groups*. These groups are defined by a set of matrices that satisfy the group properties. The group

operation is matrix multiplication, the identity element is the identity matrix, and the inverse map is the matrix inverse. To define these groups more precisely, we first introduce the *general linear group* and the concept of *convergence* of a sequence of matrices.

Definition 3.4. The *general linear group* over the real numbers, denoted $\mathrm{GL}(n, \mathbb{R})$, is the set of all invertible $n \times n$ matrices with real entries. The general linear group over the complex numbers, denoted $\mathrm{GL}(n, \mathbb{C})$, is the group of all invertible $n \times n$ matrices with complex entries. □

Definition 3.5. Let $\mathbf{X}_k$ be a sequence of complex matrices in $\mathbb{C}^{n\times n}$. We say that $\mathbf{X}_k$ *converges* to a matrix $\mathbf{Y}$ if each entry of $\mathbf{X}_k$ converges to the corresponding entry of $\mathbf{Y}$ as $k \to \infty$. Specifically, for each $i, j \in [1, n]$, the sequence $\{\mathbf{X}_k\}_{ij}$ converges to $\mathbf{Y}_{ij}$. □

With these definitions in place, we can now formally define a matrix Lie group.

Definition 3.6. A *matrix Lie group* is a subgroup G of $\mathrm{GL}(n, \mathbb{C})$ with the property that if $\mathbf{X}_k$ is a sequence of matrices in G, and $\mathbf{X}_k$ converges to some matrix $\mathbf{Y}$, then either $\mathbf{Y}$ is in G or $\mathbf{Y}$ is not invertible. This property is equivalent to saying that G is a *closed subgroup* of $\mathrm{GL}(n, \mathbb{C})$. □

Furthermore, every matrix Lie group is an *embedded submanifold* of $\mathbb{C}^{n\times n} \cong \mathbb{R}^{2n^2}$ (Hall, 2015, p. 70).

Although the concepts of *compactness* and *connectedness* apply to any Lie group, their definitions are easier to state for matrix Lie groups, and knowledge of these properties can reduce complexity in their study.

Definition 3.7. A matrix Lie group $G \subset \mathrm{GL}(n, \mathbb{C})$ is said to be *compact* if and only if:

i whenever a sequence $\mathbf{X}_k$ is in G and $\mathbf{X}_k$ converges to $\mathbf{Y}$, then $\mathbf{Y}$ is in G;
ii there exists a constant $C > 0$ such that for all $\mathbf{X} \in G$, $\|\mathbf{X}_{ij}\| \leq C$ for all $i, j \in [1, n]$.

□

Definition 3.8. A matrix Lie group G is termed *connected* if for all $\mathbf{X}, \mathbf{Y} \in G$, there exists a continuous path $\mathbf{G}(\sigma) \in G$, $a \leq \sigma \leq b$ such that $\mathbf{G}(a) = \mathbf{X}$ and $\mathbf{G}(b) = \mathbf{Y}$. The *identity component* of G, denoted G_ι, is the set of elements $\mathbf{X}$ in G for which there exists a continuous path $\mathbf{G}(\sigma) \in G$, $a \leq \sigma \leq b$ such that $\mathbf{G}(a) = \iota = \mathbf{I}$ and $\mathbf{G}(b) = \mathbf{X}$. □

Examples of Lie groups

In this section, we present some of the most common examples of Lie groups, which also serve to introduce the notation and concepts used in our extension.

Example 3.9. The *special linear group* $\mathrm{SL}(n, \mathbb{R})$ (resp. $\mathrm{SL}(n, \mathbb{C})$) is a subgroup of $\mathrm{GL}(n, \mathbb{R})$ (resp. $\mathrm{GL}(n, \mathbb{C})$) consisting of invertible matrices with determinant equal to 1.

While none of these groups are compact, $\mathrm{GL}(n, \mathbb{C})$, $\mathrm{SL}(n, \mathbb{R})$ and $\mathrm{SL}(n, \mathbb{C})$ are connected.

□

Example 3.10. The group formed by the direct product of Lie groups, $G = G_1 \times G_2 \times \cdots \times G_k$, is itself a Lie group (Lee, 2012, p. 152). Additionally, the semidirect product of Lie groups, $G = G_1 \rtimes G_2$, is also a Lie group (Lee, 2012, p. 168). □

Example 3.11. The *orthogonal group* $\mathrm{O}(n)$ is the Lie group of distance-preserving transformations, which includes rotations and reflections. It consists of all $n \times n$ orthogonal matrices:

$$\mathrm{O}(n) = \left\{ \mathbf{X} \in \mathbb{R}^{n \times n} \,|\, \mathbf{X}^\top \mathbf{X} = \mathbf{I}_n \right\}. \tag{3.8}$$

Its subgroup $\mathrm{SO}(n)$, the *special orthogonal group*, consists of all orthogonal matrices with determinant equal to 1. This group represents rotations and is formed by the set of rotation matrices:

$$\mathrm{SO}(n) = \left\{ \mathbf{X} \in \mathbb{R}^{n \times n} \,|\, \mathbf{X}^\top \mathbf{X} = \mathbf{I}_n,\, \det(\mathbf{X}) = 1 \right\}. \tag{3.9}$$

Both $\mathrm{O}(n)$ and $\mathrm{SO}(n)$ are compact; however, only $\mathrm{SO}(n)$ is connected. □

Example 3.12. The *Euclidean group* $\mathrm{E}(n)$ is a Lie group of isometries in Euclidean space, formed by the semidirect product $\mathbb{R}^n \rtimes \mathrm{O}(n)$:

$$\mathrm{E}(n) = \left\{ \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)} \,\middle|\, \mathbf{R} \in \mathrm{O}(n),\, \mathbf{p} \in \mathbb{R}^n \right\}. \tag{3.10}$$

Its subgroup $\mathrm{SE}(n)$, the *special Euclidean group*, consists of all matrices in the Euclidean group with determinant equal to 1, formed by the semidirect product $\mathbb{R}^n \rtimes \mathrm{SO}(n)$. This group represents rigid transformations and is formed by the set of homogeneous transformation matrices:

$$\mathrm{SE}(n) = \left\{ \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)} \,\middle|\, \mathbf{R} \in \mathrm{SO}(n),\, \mathbf{p} \in \mathbb{R}^n \right\}. \tag{3.11}$$

While neither of these groups are compact, $\mathrm{SE}(n)$ is connected. □

Example 3.13. The group $(\mathbb{R}^n, +)$, also denoted $\mathbb{R}^n$, is a Lie group where the group operation is vector addition, the identity element is the zero vector, and the inverse map is the negation of the vector. This group is typically represented by an inclusion map $\mathbb{R}^n \hookrightarrow \mathrm{SE}(n)$, and is denoted the *translation group* $\mathrm{T}(n)$:

$$\mathrm{T}(n) = \left\{ \begin{bmatrix} \mathbf{I}_n & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \in \mathrm{SE}(n) \,\middle|\, \mathbf{p} \in \mathbb{R}^n \right\}. \tag{3.12}$$

This group is not compact, but it is connected. □

Example 3.14. Although not present in the literature, we introduce the group of indepen-

dent translations and rotations (*Independent Special Euclidean group*) $\mathrm{ISE}(n) \cong \mathbb{R}^n \times \mathrm{SO}(n)$, defined as:

$$\begin{aligned}\mathrm{ISE}(n) &= \left\{ \begin{bmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{P} \end{bmatrix} \in \mathbb{R}^{(2n+1)\times(2n+1)} \,\middle|\, \mathbf{R} \in \mathrm{SO}(n),\, \mathbf{P} \in \mathrm{T}(n) \right\} \\ &= \left\{ \begin{bmatrix} \mathbf{R} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{p} \\ \mathbf{0} & \mathbf{0} & 1 \end{bmatrix} \in \mathbb{R}^{(2n+1)\times(2n+1)} \,\middle|\, \mathbf{R} \in \mathrm{SO}(n),\, \mathbf{p} \in \mathbb{R}^n \right\}. \end{aligned} \tag{3.13}$$

While these two representations are isomorphic, we use $\mathbb{R}^n \times \mathrm{SO}(n)$ to denote tuples of translations and rotations $(\mathbf{p}, \mathbf{R})$, and $\mathrm{ISE}(n)$ to denote elements represented as the matrices in (3.13). Both representations are interchangeable, as position and orientation can be easily extracted from the matrix representation via matrix multiplication.

This group is not compact, but it is connected. □

Example 3.15. The *real symplectic group* $\mathrm{Sp}(2n)$ is the group of all $2n \times 2n$ real matrices that preserve a non-degenerate skew-symmetric bilinear form ω. This group plays an important role in classical mechanics, especially in the study of Hamiltonian systems. The group is defined as

$$\mathrm{Sp}(2n) = \left\{ \mathbf{X} \in \mathbb{R}^{2n\times 2n} \,|\, \mathbf{X}^\top \mathbf{\Omega} \mathbf{X} = \mathbf{\Omega} \right\} \tag{3.14}$$

$$= \left\{ \mathbf{X} \in \mathbb{R}^{2n\times 2n} \,|\, -\mathbf{\Omega} \mathbf{X}^\top \mathbf{\Omega} = \mathbf{X}^{-1} \right\}, \tag{3.15}$$

where

$$\mathbf{\Omega} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_n \\ -\mathbf{I}_n & \mathbf{0} \end{bmatrix}. \tag{3.16}$$

The skew-symmetric bilinear form can be characterized by $\omega(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{\Omega} \mathbf{y} \rangle \; \forall \, \mathbf{x}, \mathbf{y} \in \mathbb{R}^{2n}$. We can also define the *complex symplectic group* $\mathrm{Sp}(2n, \mathbb{C})$ as the group of all $2n \times 2n$ complex matrices that preserve the same non-degenerate skew-symmetric bilinear form. The group definition remains the same, but with complex matrices.

Both $\mathrm{Sp}(2n)$ and $\mathrm{Sp}(2n, \mathbb{C})$ are non-compact, but they are connected. □

Example 3.16. The *indefinite orthogonal group* (or generalized orthogonal group) $\mathrm{O}(p, q)$ is the group of all $n \times n$ real matrices

$$\mathrm{O}(p, q) = \left\{ \mathbf{X} \in \mathbb{R}^{n\times n} \,|\, \mathbf{X}^\top \mathbf{I}_{p,q} \mathbf{X} = \mathbf{I}_{p,q} \right\}, \tag{3.17}$$

where $n = p + q$, and

$$\mathbf{I}_{p,q} = \begin{bmatrix} \mathbf{I}_p & \mathbf{0} \\ \mathbf{0} & -\mathbf{I}_q \end{bmatrix}. \tag{3.18}$$

Similarly, the *indefinite special orthogonal group* $\mathrm{SO}(p, q)$ is defined as:

$$\mathrm{SO}(p, q) = \{\mathbf{X} \in \mathrm{O}(p, q) \mid \det(\mathbf{X}) = 1\}. \tag{3.19}$$

The group $\mathrm{O}(1, 3)$, of particular interest in the study of special relativity, is called the *Lorentz group*. It preserves the Lorentz metric:

$$(t, x, y, z) \mapsto t^2 - x^2 - y^2 - z^2. \tag{3.20}$$

The Lorentz group is also represented as $\mathrm{O}(3, 1)$, which would preserve the metric $x^2 + y^2 + z^2 - t^2$. If we restrict this group to transformations that preserve orientation, we obtain the *special Lorentz group* $\mathrm{SO}(1, 3)$. Furthermore, the subgroup of all Lorentz transformations that preserve orientation and the time direction is called the *proper orthochronous Lorentz group* $\mathrm{SO}^+(1, 3)$, which is the identity component G_ι of $\mathrm{SO}(1, 3)$.

Neither $\mathrm{O}(p, q)$ nor $\mathrm{SO}(p, q)$ are compact or connected. However, $\mathrm{SO}^+(1, 3)$ is connected. □

Example 3.17. An example of a Lie group that is not a matrix Lie group is given in Hall (2015, p. 25) and reproduced here:

The group $G = \mathbb{R} \times \mathbb{R} \times \mathbb{S}^1 = \{(x, y, \theta) | x \in \mathbb{R}, y \in \mathbb{R}, \theta \in \mathbb{S}^1 \subset \mathbb{C}\}$, equipped with the group operation

$$(x_1, y_1, \theta_1) \circ (x_2, y_2, \theta_2) = (x_1 + x_2, y_1 + y_2, e^{ix_1y_2}\theta_1\theta_2) \tag{3.21}$$

is a Lie group that does not have a matrix representation. The identity element is $\iota = (0, 0, 1)$, and the inverse map is $(x, y, \theta)^{-1} = (-x, -y, e^{ixy}\theta^{-1})$. □

3.2.2 Lie algebras

Lie groups are intimately connected to a mathematical structure called the *Lie algebra*, which provides a linear algebraic framework for studying the properties of Lie groups. The Lie algebra of a Lie group G, denoted $\mathrm{Lie}(G)$, is defined as the set of all smooth *left-invariant* vector fields on G. These are vector fields that remain unchanged under the "shifting" induced by the group operations. To formally define left-invariant vector fields, it is necessary first to understand the concept of differentials in the context of Lie groups.

Consider a homomorphism of Lie groups $\mathcal{H} : G \to H$. If V is a vector field on G, the differential of $\mathcal{H}$ at a point $\mathbf{X} \in G$, denoted $d(\mathcal{H})_\mathbf{X}$, maps the vector $V(\mathbf{X})$ in the

tangent space of G at $\mathbf{X}$ to a vector in the tangent space of H at $\mathcal{H}(\mathbf{X})$. Formally, $d(\mathcal{H})_{\mathbf{X}} : T_{\mathbf{X}}G \to T_{\mathcal{H}(\mathbf{X})}H$.

If there exists a vector field W on H such that $d(\mathcal{H})_{\mathbf{X}}\Big(V(\mathbf{X})\Big) = W\Big(\mathcal{H}(\mathbf{X})\Big)$ for all $\mathbf{X} \in G$, then V and W are said to be $\mathcal{H}$-related. This concept is particularly significant because, if $\mathcal{H}$ is a diffeomorphism (a Lie group isomorphism in this context), there exists a unique vector field W on H that is $\mathcal{H}$-related to V (Lee, 2012, p. 183).

The concept of left-invariant (resp. right-invariant) vector fields on Lie groups can now be introduced using the differential mappings of the translation operations:

Definition 3.18. A vector field V on a Lie group G is said to be *left-invariant* (resp. *right-invariant*) if it is invariant under all left (resp. right) translations:

$$d(\mathcal{L}_{\mathbf{X}})_{\mathbf{Y}}(V(\mathbf{Y})) = V(\mathbf{X} \circ \mathbf{Y}) \,\forall\, \mathbf{X}, \mathbf{Y} \in G,$$
$$\text{(resp.)}\ d(\mathcal{R}_{\mathbf{X}})_{\mathbf{Y}}(V(\mathbf{Y})) = V(\mathbf{Y} \circ \mathbf{X}) \,\forall\, \mathbf{X}, \mathbf{Y} \in G.$$

□

Since the translation operations are isomorphisms, every left-invariant (resp. right-invariant) vector field on a Lie group can be fully determined by its value at the identity element. In other words, if V is a left-invariant (resp. right-invariant) vector field and $V_\iota \triangleq V(\iota) \in T_\iota G$ is its value at the identity element ι, then V and V_ι are $\mathcal{L}_{\mathbf{X}}$-related (resp. $\mathcal{R}_{\mathbf{X}}$-related) for all $\mathbf{X} \in G$. This relationship implies that $V(\mathbf{X}) = d(\mathcal{L}_{\mathbf{X}})_\iota V_\iota$ (resp. $V(\mathbf{X}) = d(\mathcal{R}_{\mathbf{X}})_\iota V_\iota$).

From this observation, the Lie algebra can be understood as a vector space that encapsulates the local behavior of the group near the identity element. For this reason, the Lie algebra of G is often defined as the tangent space at the identity element, $T_\iota G$, and is commonly denoted by $\mathfrak{g}$. More generally, a Lie algebra is defined as follows:

Definition 3.19. A *real Lie algebra* is a real vector space $\mathfrak{g}$ endowed with a map $[\cdot,\cdot] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g}$, called the *Lie bracket* on $\mathfrak{g}$, that satisfies the following properties for all $\mathbf{A}, \mathbf{B}, \mathbf{C} \in \mathfrak{g}$:

i *Bilinearity*: For all $a, b \in \mathbb{R}$,

$$[a\mathbf{A} + b\mathbf{B}, \mathbf{C}] = a[\mathbf{A}, \mathbf{C}] + b[\mathbf{B}, \mathbf{C}],$$
$$[\mathbf{C}, a\mathbf{A} + b\mathbf{B}] = a[\mathbf{C}, \mathbf{A}] + b[\mathbf{C}, \mathbf{B}];$$

ii *Skew-symmetry*:

$$[\mathbf{A}, \mathbf{B}] = -[\mathbf{B}, \mathbf{A}];$$

iii *Jacobi identity*:

$$[\mathbf{A}, [\mathbf{B}, \mathbf{C}]] + [\mathbf{B}, [\mathbf{C}, \mathbf{A}]] + [\mathbf{C}, [\mathbf{A}, \mathbf{B}]] = 0.$$

□

As a direct consequence of skew-symmetry (Property 3.19.ii), it follows that $[\mathbf{A}, \mathbf{A}] = 0 \, \forall \, \mathbf{A} \in \mathfrak{g}$.

Mappings between Lie algebras, particularly homomorphisms and isomorphisms, play a crucial role in understanding their structure. A homomorphism of Lie algebras is defined as follows:

Definition 3.20. Let $\mathfrak{g}$ and $\mathfrak{h}$ be Lie algebras with Lie brackets $[\cdot, \cdot]_\mathfrak{g}$ and $[\cdot, \cdot]_\mathfrak{h}$, respectively. A linear map $\mathscr{H} : \mathfrak{g} \to \mathfrak{h}$ is called a *homomorphism* if it preserves the Lie bracket:

$$\mathscr{H}([\mathbf{A}, \mathbf{B}]_\mathfrak{g}) = [\mathscr{H}(\mathbf{A}), \mathscr{H}(\mathbf{B})]_\mathfrak{h} \, \forall \, \mathbf{A}, \mathbf{B} \in \mathfrak{g}.$$

If $\mathscr{H}$ is bijective, then $\mathscr{H}^{-1}$ is also a homomorphism, and $\mathscr{H}$ is called an *isomorphism*. In this case, $\mathfrak{g}$ and $\mathfrak{h}$ are said to be *isomorphic*, denoted $\mathfrak{g} \cong \mathfrak{h}$. □

Lie algebra homomorphisms can be induced by Lie group homomorphisms. Specifically, if $\mathcal{H}$ is a Lie group homomorphism between G and H, then the differential of $\mathcal{H}$ at the identity element, $d\mathcal{H}_\iota = \mathscr{H}$, is a Lie algebra homomorphism between $\mathfrak{g} = T_\iota G$ and $\mathfrak{h} = T_\iota H$ (Duistermaat and Kolk, 1999, p. 41). Moreover, if $\mathcal{H}$ is a Lie group isomorphism, then $d\mathcal{H}_\iota$ is a Lie algebra isomorphism.

The most important Lie algebra homomorphism is the differential of the translation operations. As discussed earlier, every left-invariant (resp. right-invariant) vector field can be associated with the tangent space at the identity element. Since the translation operations are isomorphisms, the space of left-invariant (resp. right-invariant) vector fields, $\mathrm{Lie}(G)$, is isomorphic to the tangent space at the identity element, $\mathfrak{g}$, i.e., $\mathrm{Lie}(G) \cong \mathfrak{g}$. Both definitions of the Lie algebra of a Lie group are equally significant and are used interchangeably depending on the context, as each definition may simplify comprehension or computations in specific scenarios. Homomorphisms serve as a bridge between these definitions, with their utility stemming from the ability to identify the tangent space of G at any point $\mathbf{X}$, $T_\mathbf{X} G$, with the tangent space at the identity $\mathfrak{g}$.

Additionally, left-invariant and right-invariant vector fields form a basis for the tangent space of G at every point $\mathbf{X} \in G$. Consequently, any element of $T_\mathbf{X} G$ can be expressed as a linear combination of a finite collection of linearly independent left-invariant or right-invariant vector fields. In differential geometry terms, this implies that left-invariant and right-invariant vector fields form a *global frame* for G (Lee, 2012, p. 192).

We now introduce another isomorphism essential for our generalization. Since the Lie algebra is a vector space, we can define a basis $\{\mathbf{E}_k\} \in \mathfrak{g}, k \in [1, m]$ for an m-dimensional

Lie algebra $\mathfrak{g}$. For any $\mathbf{A} \in \mathfrak{g}$, there exist scalars $\{\zeta_k\}$, $k \in [1, m]$, such that $\mathbf{A} = \sum_{k=1}^{m} \zeta_k \mathbf{E}_k$. Furthermore, for each choice of basis for the Lie algebra, we can uniquely define a linear operator $\mathcal{S} : \mathbb{R}^m \to \mathfrak{g}$ as follows:

Definition 3.21 (S map)**.** Let $\mathbf{E}_1, \ldots, \mathbf{E}_m$ be a basis for an m-dimensional Lie algebra $\mathfrak{g}$, and let $\boldsymbol{\zeta}$ be an m-dimensional vector. The *S map* is the unique isomorphism $\mathcal{S} : \mathbb{R}^m \to \mathfrak{g}$, defined by

$$\mathcal{S}(\boldsymbol{\zeta}) \triangleq \sum_{k=1}^{m} \zeta_k \mathbf{E}_k. \tag{3.22}$$

□

Since $\mathcal{S}$ is an isomorphism, the *inverse map*, which maps elements of the Lie algebra $\mathfrak{g}$ back to a vector in $\mathbb{R}^m$, can also be defined:

Definition 3.22 (Inverse S map)**.** Let $\mathfrak{g}$ be an m-dimensional Lie algebra. The *inverse S map* is the isomorphism $\mathcal{S}^{-1} : \mathfrak{g} \to \mathbb{R}^m$, such that $\mathcal{S}^{-1}(\mathcal{S}(\boldsymbol{\zeta})) = \boldsymbol{\zeta}$. □

The intuition behind the S maps is that they allow elements of the Lie algebra $\mathfrak{g}$ to be represented as tangent vectors to $\mathbb{R}^m$. In cases where the Lie group has physical significance, this enables terms like "velocity" to describe elements of the tangent space of G – the Lie algebra $\mathfrak{g}$ – itself. Clearly, this approach requires the choice of a basis for $\mathfrak{g}$ that aligns with the underlying physical interpretation.

Examples of Lie algebras

This section presents the Lie algebras corresponding to the Lie groups introduced previously, along with several Lie algebra isomorphisms.

Example 3.23. The Lie algebra of the general linear group $\mathrm{GL}(n, \mathbb{R})$ is the space of all $n \times n$ matrices, denoted $\mathfrak{gl}(n, \mathbb{R})$, with its Lie bracket defined by the commutator $[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA}$. Similarly, the Lie algebra of the general linear group $\mathrm{GL}(n, \mathbb{C})$, denoted $\mathfrak{gl}(n, \mathbb{C})$, is the set of all $n \times n$ complex matrices with the same Lie bracket. Furthermore, the Lie algebra of the special linear group $\mathrm{SL}(n, \mathbb{R})$ (resp. $\mathrm{SL}(n, \mathbb{C})$) is the space of all $n \times n$ real (resp. complex) matrices with null trace, denoted $\mathfrak{sl}(n, \mathbb{R})$ (resp. $\mathfrak{sl}(n, \mathbb{C})$). □

Example 3.24. The lie algebra for the direct product of Lie groups $G = G_1 \times G_2 \times \cdots \times G_k$ is the direct sum of the Lie algebras of the individual groups, i.e. $\mathfrak{g}_1 \oplus \mathfrak{g}_2 \oplus \cdots \oplus \mathfrak{g}_k$. □

Example 3.25. The Lie algebra of the orthogonal group $\mathrm{O}(n)$ is identical to that of the special orthogonal group $\mathrm{SO}(n)$, denoted $\mathfrak{so}(n)$. This Lie algebra consists of all $n \times n$ skew-symmetric matrices:

$$\mathfrak{so}(n) = \left\{ \mathbf{A} \in \mathbb{R}^{n \times n} \,|\, \mathbf{A}^\top = -\mathbf{A} \right\}.$$

In the special case of SO(3), the Lie algebra $\mathfrak{so}(3)$is commonly represented using the canonical basis

$$\mathbf{E}_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{E}_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}, \quad \mathbf{E}_3 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

Additionally, a mapping $\mathcal{S}$ is often employed to represent elements of $\mathfrak{so}(3)$ by mapping a vector $\boldsymbol{\zeta} = [\zeta_1 \ \zeta_2 \ \zeta_3]^\top$ to a skew-symmetric matrix:

$$\mathcal{S}(\boldsymbol{\zeta}) = \zeta_1\mathbf{E}_1 + \zeta_2\mathbf{E}_2 + \zeta_3\mathbf{E}_3 = \begin{bmatrix} 0 & -\zeta_3 & \zeta_2 \\ \zeta_3 & 0 & -\zeta_1 \\ -\zeta_2 & \zeta_1 & 0 \end{bmatrix}.$$

This mapping $\mathcal{S}$ establishes a Lie algebra isomorphism between the Lie algebra $\mathbb{R}^3$ (under the cross product as its Lie bracket) and $\mathfrak{so}(3)$. Specifically, given two vectors $\mathbf{a} = [a_1 \ a_2 \ a_3]^\top$ and $\mathbf{b} = [b_1 \ b_2 \ b_3]^\top$, then

$$\mathcal{S}(\mathbf{a} \times \mathbf{b}) = \begin{bmatrix} 0 & a_2b_1 - a_1b_2 & a_3b_1 - a_1b_3 \\ a_1b_2 - a_2b_1 & 0 & a_3b_2 - a_2b_3 \\ a_1b_3 - a_3b_1 & a_2b_3 - a_3b_2 & 0 \end{bmatrix},$$

now computing $\mathcal{S}(\mathbf{a})\,\mathcal{S}(\mathbf{b})$ and $\mathcal{S}(\mathbf{b})\,\mathcal{S}(\mathbf{a})$ gives

$$\mathcal{S}(\mathbf{a})\,\mathcal{S}(\mathbf{b}) = \begin{bmatrix} -a_3b_3 - a_2b_2 & a_2b_1 & a_3b_1 \\ a_1b_2 & -a_3b_3 - a_1b_1 & a_3b_2 \\ a_1b_3 & a_2b_3 & -a_2b_2 - a_1b_1 \end{bmatrix},$$

$$\mathcal{S}(\mathbf{b})\,\mathcal{S}(\mathbf{a}) = \begin{bmatrix} -a_3b_3 - a_2b_2 & a_1b_2 & a_1b_3 \\ a_2b_1 & -a_3b_3 - a_1b_1 & a_2b_3 \\ a_3b_1 & a_3b_2 & -a_2b_2 - a_1b_1 \end{bmatrix}.$$

Clearly, $\mathcal{S}(\mathbf{a} \times \mathbf{b}) = [\mathcal{S}(\mathbf{a}), \mathcal{S}(\mathbf{b})] = \mathcal{S}(\mathbf{a})\,\mathcal{S}(\mathbf{b}) - \mathcal{S}(\mathbf{b})\,\mathcal{S}(\mathbf{a})$. This demonstrates that $\mathcal{S}$ is indeed a Lie algebra isomorphism. □

Example 3.26. The Lie algebra of the Euclidean group $\mathrm{E}(n)$ and the special Euclidean group $\mathrm{SE}(n)$, denoted $\mathfrak{se}(n)$, is the space

$$\mathfrak{se}(n) = \left\{ \begin{bmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{0} & 0 \end{bmatrix} \in \mathbb{R}^{(n+1)\times(n+1)} \,\middle|\, \mathbf{A} \in \mathfrak{so}(n),\ \mathbf{b} \in \mathbb{R}^n \right\}.$$

For the case of $\mathfrak{se}(3)$, elements of this Lie algebra are frequently represented as six-dimensional vectors, called twists, $\boldsymbol{\zeta} = [\zeta_1 \ \zeta_2 \ \zeta_3 \ \zeta_4 \ \zeta_5 \ \zeta_6]^\top$. The canonical isomorphism $\mathcal{S}$

is given by

$$\mathcal{S}(\boldsymbol{\zeta}) = \begin{bmatrix} 0 & -\zeta_6 & \zeta_5 & \zeta_1 \\ \zeta_6 & 0 & -\zeta_4 & \zeta_2 \\ -\zeta_5 & \zeta_4 & 0 & \zeta_3 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

where $[\zeta_1 \ \zeta_2 \ \zeta_3]^\top$ represents the linear velocity $\mathbf{v}$ and $[\zeta_4 \ \zeta_5 \ \zeta_6]^\top$ represents the angular velocity $\boldsymbol{\omega}$. □

Example 3.27. The Lie algebra of $\mathrm{T}(n)$ is a subset of $\mathfrak{se}(n)$, explicitly given by

$$\mathfrak{t}(n) = \left\{ \begin{bmatrix} \mathbf{0} & \mathbf{a} \\ \mathbf{0} & 0 \end{bmatrix} \in \mathfrak{se}(n) \,\middle|\, \mathbf{a} \in \mathbb{R}^n \right\}.$$

□

Example 3.28. The Lie algebra of the independent special Euclidean group $\mathrm{ISE}(n)$ is given by

$$\begin{aligned} \mathfrak{ise}(n) &= \left\{ \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{0} & \mathbf{B} \end{bmatrix} \in \mathbb{R}^{(2n+1)\times(2n+1)} \,\middle|\, \mathbf{A} \in \mathfrak{so}(n),\, \mathbf{B} \in \mathfrak{t}(n) \right\} \\ &= \left\{ \begin{bmatrix} \mathbf{A} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{b} \\ \mathbf{0} & \mathbf{0} & 0 \end{bmatrix} \in \mathbb{R}^{(2n+1)\times(2n+1)} \,\middle|\, \mathbf{A} \in \mathfrak{so}(n),\, \mathbf{b} \in \mathbb{R}^n \right\}. \end{aligned}$$

This implies that $\mathfrak{ise}(n) = \mathfrak{so}(n) \oplus \mathfrak{t}(n)$, consistent with the definition of the Lie algebra of a direct product of Lie groups. □

Example 3.29. The symplectic Lie algebra $\mathfrak{sp}(2n)$ is defined as

$$\begin{aligned} \mathfrak{sp}(2n) &= \left\{ \mathbf{A} \in \mathbb{R}^{2n\times 2n} \mid \mathbf{A}^\top \boldsymbol{\Omega} + \boldsymbol{\Omega}\mathbf{A} = 0 \right\} \\ &= \left\{ \begin{bmatrix} \mathbf{B} & \mathbf{C} \\ \mathbf{D} & -\mathbf{B}^\top \end{bmatrix} \in \mathbb{R}^{2n\times 2n} \,\middle|\, \mathbf{C} = \mathbf{C}^\top,\, \mathbf{D} = \mathbf{D}^\top \right\}, \end{aligned}$$

where $\boldsymbol{\Omega}$ is the same matrix as defined in Example 3.15. The dimension of $\mathfrak{sp}(2n)$ is

$n(2n+1)$. For a vector $\boldsymbol{\zeta} = [\zeta_1 \ \dots \ \zeta_{2n^2+n}]^\top$, an isomorphism $\mathcal{S}$ is given by

$$\mathcal{S}(\boldsymbol{\zeta}) = \begin{bmatrix} \zeta_1 & \cdots & \zeta_n & \zeta_{n^2+1} & \cdots & \zeta_{n^2+n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \zeta_{n^2-n+1} & \cdots & \zeta_{n^2} & \zeta_{n^2+n} & \cdots & \zeta_{\frac{3n^2+n}{2}} \\ \zeta_{\frac{3n^2+n}{2}+1} & \cdots & \zeta_{\frac{3n^2+3n}{2}} & -\zeta_1 & \cdots & -\zeta_{n^2-n+1} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \zeta_{\frac{3n^2+3n}{2}} & \cdots & \zeta_{2n^2+n} & -\zeta_n & \cdots & -\zeta_{n^2} \end{bmatrix}.$$

□

Example 3.30. The Lie algebra of the indefinite orthogonal group $\mathrm{O}(p,q)$ and the indefinite special orthogonal group $\mathrm{SO}(p,q)$ is the same and is denoted $\mathfrak{so}(p,q)$. This Lie algebra is defined as

$$\mathfrak{so}(p,q) = \left\{ \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix} \in \mathbb{R}^{(p+q)\times(p+q)} \,\middle|\, \mathbf{A}^\top = -\mathbf{A},\, \mathbf{C}^\top = -\mathbf{C},\, \mathbf{B} \in \mathbb{R}^{p\times q} \right\}.$$

For the Lorentz group $\mathrm{O}(3,1)$, the special Lorentz group $\mathrm{SO}(3,1)$, and the proper orthochronous Lorentz group $\mathrm{SO}^+(3,1)$, the shared Lie algebra $\mathfrak{so}(3,1)$ has an isomorphism for a vector $\boldsymbol{\zeta} = [\zeta_1 \ \zeta_2 \ \zeta_3 \ \zeta_4 \ \zeta_5 \ \zeta_6]^\top$ defined as:

$$\mathcal{S}(\boldsymbol{\zeta}) = \begin{bmatrix} 0 & -\zeta_3 & \zeta_2 & \zeta_4 \\ \zeta_3 & 0 & -\zeta_1 & \zeta_5 \\ -\zeta_2 & \zeta_1 & 0 & \zeta_6 \\ \zeta_4 & \zeta_5 & \zeta_6 & 0 \end{bmatrix}.$$

□

3.2.3 Exponential map

The *exponential map* is a fundamental concept in the study of Lie groups and Lie algebras, providing a means to map elements of a Lie algebra to elements of its corresponding Lie group. While the exponential map can be defined for any Lie group, we focus on matrix Lie groups, where the definition is more straightforward and directly relevant to our work. For matrix Lie groups, the exponential map is given by the matrix exponential:

$$\exp : \mathfrak{g} \to G, \tag{3.23}$$

defined via the power series:

$$\exp(\mathbf{A}) = \sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!},\ \mathbf{A} \in \mathfrak{g}. \tag{3.24}$$

The exponential map is also instrumental in determining the Lie algebra of a matrix Lie group, as shown in the following classical theorem:

Theorem 3.31 (Von Neumann and Cartan, 1927)**.** *Let $G \subset GL(n, \mathbb{R})$ be a matrix Lie group. The set $\mathfrak{g}$ defined as*

$$\mathfrak{g} = \left\{ \mathbf{A} \in \mathbb{R}^{n \times n} \mid \exp(\sigma \mathbf{A}) \in G \, \forall \, \sigma \in \mathbb{R} \right\} \tag{3.25}$$

forms a vector space equal to the tangent space $T_\iota G$ at the identity element ι. Furthermore, $\mathfrak{g}$ is closed under the Lie bracket $[\mathbf{A}, \mathbf{B}] = \mathbf{AB} - \mathbf{BA} \, \forall \, \mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times n}$. □

Although the exponential map maps elements of the Lie algebra to the Lie group, it is not necessarily surjective. The image of the exponential map lies within the identity component G_ι of the Lie group and, more specifically, forms a neighborhood of the identity element (Hall, 2015, p. 56). While sufficient conditions for surjectivity exist – such as the Lie group being connected and compact (Hall, 2015, p. 316) – no general necessary conditions are currently known.

Lie groups for which the exponential map is surjective are called *exponential Lie groups* (Đoković and Thắńg, 1995). Examples of exponential Lie groups include the special orthogonal group $\mathrm{SO}(n)$ (Gallier and Quaintance, 2020, p. 28), the special Euclidean group $\mathrm{SE}(n)$ (Gallier and Quaintance, 2020, p. 42), and the proper orthochronous Lorentz group $\mathrm{SO}^+(3, 1)$ (Gallier and Quaintance, 2020, p. 197). Additional nontrivial examples can be found in Đoković and Thắńg (1995). Conversely, an example of a non-exponential Lie group is the special linear group $\mathrm{SL}(n)$ (Gallier and Quaintance, 2020, p. 28).

3.2.4 Integral curves, flows and derivatives

The contents of this section will be utilized to define our kinematic model for the vector field strategy. We begin by introducing the concept of integral curves, which correspond to the solutions of the differential equation defined by a vector field. More formally, their definition is as follows:

Definition 3.32. If V is a vector field on a Lie group G, and $I \subseteq \mathbb{R}$ is an open interval containing 0, an *integral curve* of V is a differentiable curve $\gamma : I \to G$ such that its velocity at each point $\mathbf{X} \in G$ equals the value of $V(\mathbf{X})$ at that point:

$$\frac{d}{d\sigma}\gamma(\sigma) = V\Big(\gamma(\sigma)\Big) \, \forall \, \sigma \in I.$$

The point $\gamma(0)$ is called the *initial condition* of the integral curve. Note that it is always possible to redefine the interval I to include the point 0 by the translation lemma (Lee, 2012, p. 208). □

This definition implies that, at every point $\mathbf{X} = \gamma(\sigma)$, the velocity $\frac{d}{d\sigma}\gamma(\sigma)$ matches the value of the vector field $V(\mathbf{X})$ at that point.

A flow, in this context, is a collection of integral curves of a vector field V, i.e., a family of solutions to the differential equation defined by V. Its formal definition is:

Definition 3.33. Let V be a vector field on a Lie group G, and let $\mathbf{X}$ be a point in G. A *local flow* of V is a map

$$\rho : I \times G \to G,$$

where $I \subseteq \mathbb{R}$ is an open interval containing 0. It follows that, for every $\mathbf{X} \in G$, the curve $\rho(\sigma, \mathbf{X})$ is an integral curve of V with initial condition $\mathbf{X}$. Furthermore, if $I = \mathbb{R}$, then ρ is a *global flow*. Since a flow is directly dependent on the vector field it is often denoted by ρ_V. However, for conciseness, we will omit this explicit dependency. □

In control theory, the concept of a flow aligns with the notion of a trajectory. For example, stating that a trajectory $x(t)$ has the initial condition $x(0) = x_0$ is equivalent to defining a flow $\rho(t, x_0)$ that maps a point to the integral curve with the given initial condition at time t.

It is known that every smooth global flow gives rise to a smooth vector field whose integral curves coincide with the curves defined by the flow (Lee, 2012, p. 211). However, the converse is not always true: not every smooth vector field generates a smooth global flow. This limitation does not apply to left-invariant or right-invariant vector fields on Lie groups, as these are *complete* (Gallier and Quaintance, 2020, p. 570), meaning that they generate global flows (Lee, 2012, p. 215). Moreover, the integral curves of these flows are unique (Duistermaat and Kolk, 1999, p. 18) and are commonly referred to as *maximal integral curves* (Gallier and Quaintance, 2020, p. 570).

With these concepts in place, we can now focus on an important aspect of our work: the trajectories and velocities of a system on a matrix Lie group. More specifically, we aim to study the global flows of vector fields defined on matrix Lie groups. To that end, we present the following lemma, adapted from Gallier and Quaintance (2020, p. 570):

Lemma 3.34. *Let G be a Lie group, V a right-invariant (resp. left-invariant) vector field, $\rho : \mathbb{R} \times G \to G$ its global flow, and $\gamma(\sigma) = \rho(\sigma, \mathbf{X})$ the associated maximal integral curve with initial condition $\mathbf{X} \in G$. Then*

$$\gamma(\sigma) = \rho(\sigma, \mathbf{X}) = \rho(\sigma, \iota) \circ \mathbf{X} = \mathcal{R}_{\mathbf{X}}\Big(\rho(\sigma, \iota)\Big),$$
$$(\textit{resp.})\ \rho(\sigma, \mathbf{X}) = \mathbf{X} \circ \rho(\sigma, \iota) = \mathcal{L}_{\mathbf{X}}\Big(\rho(\sigma, \iota)\Big).$$

Proof. Let $\bar{\gamma}(\sigma) = \mathcal{R}_{\mathbf{X}}\Big(\rho(\sigma, \iota)\Big)$, where $\bar{\gamma}(0) = \mathbf{X}$. Applying the chain rule:

$$\frac{d}{d\sigma}\bar{\gamma}(\sigma) = d(\mathcal{R}_{\mathbf{X}})_{\rho(\sigma,\iota)}\Big(V\Big(\rho(\sigma,\iota)\Big)\Big) = V\Big(\mathcal{R}_{\mathbf{X}}\Big(\rho(\sigma,\iota)\Big)\Big) = V\Big(\bar{\gamma}(\sigma)\Big), \tag{3.26}$$

since V is right-invariant. By the uniqueness of maximal integral curves, $\bar{\gamma}(\sigma) = \rho(\sigma, \mathbf{X})\,\forall\,\sigma$. Hence $\rho(\sigma, \mathbf{X}) = \rho(\sigma, \iota) \circ \mathbf{X}$. The proof for the left-invariant case is analogous. □

This lemma indicates that following a curve with an initial condition $\mathbf{X}$ and velocity v is equivalent to following a curve starting at the identity with the same velocity and subsequently translating the result by $\mathbf{X}$.

We now examine the behavior of trajectories and their velocities in matrix Lie groups. The following lemma will play a central role in this analysis:

Lemma 3.35. *Let G be a matrix Lie group, and let $\mathbf{G} : \mathbb{R} \to G$ be a differentiable function. Then, there exists a function $\mathbf{A} : \mathbb{R} \to \mathfrak{g}$ such that*

$$\frac{d}{d\sigma}\mathbf{G}(\sigma) = \mathbf{A}(\sigma)\mathbf{G}(\sigma). \tag{3.27}$$

Proof. As defined, $\mathbf{G}(\sigma)$ is an integral curve with some initial condition $\mathbf{X} \in G$. Hence, its derivative is given by the value of a vector field V at that point, i.e., $\frac{d}{d\sigma}\mathbf{G}(\sigma) = V\Big(\mathbf{G}(\sigma)\Big)$. Since right-invariant vector fields form a basis for the tangent space at any point in a Lie group, we may assume, without loss of generality, that V is a right-invariant vector field.

Let $\rho : \mathbb{R} \times G \to G$ denote the global flow of V, thus $\mathbf{G}(\sigma) = \rho(\sigma, \mathbf{X})$. By Lemma 3.34, we can express

$$\frac{d}{d\sigma}\mathbf{G}(\sigma) = V\Big(\rho(\sigma, \mathbf{X})\Big) = V\Big(\mathcal{R}_{\rho(\sigma,\mathbf{X})}(\iota)\Big) = d(\mathcal{R}_{\rho(\sigma,\mathbf{X})})_{\iota}\Big(V(\iota)\Big) = d(\mathcal{R}_{\mathbf{G}(\sigma)})_{\iota}\Big(V(\iota)\Big). \tag{3.28}$$

Furthermore, since V is right-invariant, it belongs to $\mathrm{Lie}(G)$. Given the isomorphism between $\mathrm{Lie}(G)$ and $\mathfrak{g}$, we can identify $V\Big(\rho(\sigma, \mathbf{X})\Big)$ with an element in the tangent space at the identity, $V(\iota) \in T_{\iota}G$, for every σ.

Additionally, as the Lie algebra $\mathfrak{g}$ is defined as the tangent space at the identity, we can write $V(\iota) = \mathbf{A} \in \mathfrak{g}$. However, V is not an arbitrary vector field but one that generates the integral curve $\mathbf{G}(\sigma)$. Thus, to make the dependence on σ explicit, we write $V(\iota) = \mathbf{A}(\sigma) \in \mathfrak{g}$. Substituting this, the derivative becomes

$$\frac{d}{d\sigma}\mathbf{G}(\sigma) = d(\mathcal{R}_{\mathbf{G}(\sigma)})_{\iota}\Big(\mathbf{A}(\sigma)\Big). \tag{3.29}$$

Since G is a matrix Lie group, the right translation $\mathcal{R}_{\mathbf{X}}(\mathbf{Y})$ is a linear map $\mathbf{Y} \mapsto \mathbf{YX}$. This implies that the differential of this map is given by $d(\mathcal{R}_{\mathbf{X}})_{\mathbf{Y}}(W(\mathbf{Y})) = W(\mathbf{Y})\mathbf{X}$ (Lee,

2012, p. 194) for any vector field W on G. Consequently, we can write

$$\frac{d}{d\sigma}\mathbf{G}(\sigma) = d(\mathcal{R}_{\mathbf{G}(\sigma)})_{\iota}\Big(\mathbf{A}(\sigma)\Big) = \mathbf{A}(\sigma)\mathbf{G}(\sigma). \tag{3.30}$$

□

From an engineering perspective, Lemma 3.35 is of limited utility, as it is uncommon to control systems directly through their Lie algebra. For instance, in the case of SO(3), control is typically applied via angular velocity, which belongs to $\mathbb{R}^3$, rather than through the Lie algebra $\mathfrak{so}(3)$. However, the previously defined S map (Definition 3.21) enables precisely this intuition. Thus, we restate Lemma 3.35 in terms of the S map as follows:

Lemma 3.36. *Given a differentiable function* $\mathbf{G} : \mathbb{R} \to G$*, there exists a function* $\boldsymbol{\zeta} : \mathbb{R} \to \mathbb{R}^m$*, such that*

$$\frac{d}{d\sigma}\mathbf{G}(\sigma) = \mathcal{S}\Big(\boldsymbol{\zeta}(\sigma)\Big)\mathbf{G}(\sigma). \tag{3.31}$$

Proof. This is a direct consequence of Lemma 3.35 and Definition 3.21. □

This lemma implies that the velocity of a curve in G – or trajectories in G – can be expressed as a single integrator model. As such, this result will be used to formulate a kinematic model for our strategy.

3.3 Adaptive control

Adaptive control emerged in the 1950s as a response to the challenges of designing aircraft autopilots. Aircraft operate across a wide range of speeds and altitudes, leading to significant variations in their parameters. This inspired the central idea of adaptive control: the online estimation of parameters based on measured system signals (Slotine and Li, 1991). Similar needs arise in other systems: robot manipulators may handle objects of unknown mass; chemical processes often involve unmodeled dynamics; and variations in fuel consumption can challenge conventional vehicle controllers. In this section, we briefly present the main concepts of adaptive control, drawing on the foundational works of Slotine and Li (1991), Krstic et al. (1995) and Ioannou and Sun (2012).

An adaptive controller combines a control law, designed for a nominal model, with an adaptation law that estimates the unknown parameters of the plant in real time. Adaptive control can be broadly categorized into two main approaches based on how the parameter estimation is performed: *direct* and *indirect* adaptive control. In indirect adaptive control, the plant parameters are estimated online and subsequently used to compute the controller parameters. In contrast, the direct approach parameterizes the plant model in terms of the controller parameters, which are then estimated online.

In the indirect approach, the plant model $P(\theta)$ is parameterized with respect to an unknown parameter vector θ. For instance, using the Euler-Lagrange formulation for dynamic systems, the plant can be represented by a regressor matrix $\mathbf{Y}$ and a parameter vector θ, which contains the unknowns. The adaptation law performs online estimation of the parameters, yielding an estimate $\widehat{\theta}(t)$, which is then used to construct an estimated plant $\widehat{P}(\widehat{\theta})$. This estimated plant is treated as if it were the true model and serves as the basis for designing the controller. The controller parameters $\theta_c(t)$ are determined by solving equations that depend on the estimated plant. A block diagram of this approach is shown in Figure 3.2.

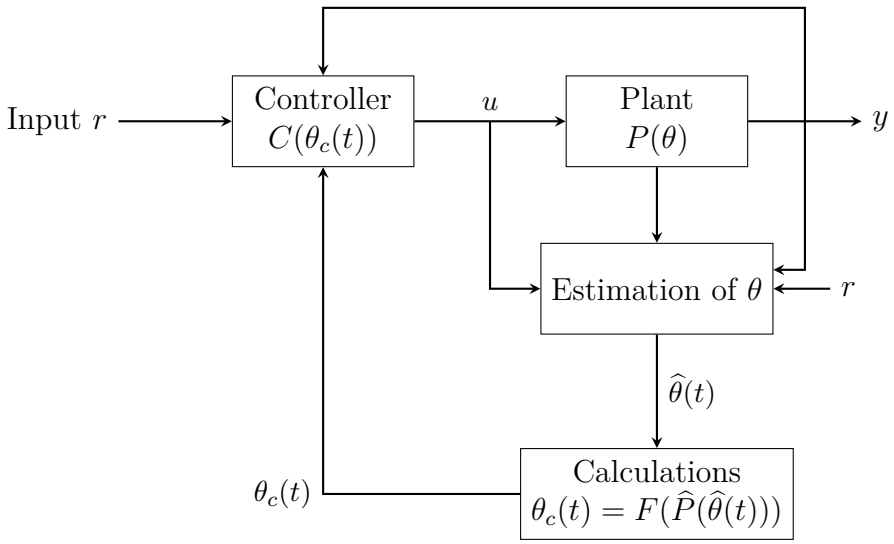


Figure 3.2: Block diagram of an indirect adaptive control system.

In the direct adaptive control approach, the plant model $P(\theta)$ is parameterized directly in terms of the unknown controller parameter vector θ_c. This parameterization ensures that when the true controller parameters are used, denoted by $P_c(\theta_c)$, the results are equivalent to the behavior of the true plant. The adaptation law estimates the controller parameters $\widehat{\theta}_c(t)$, which are then used to compute the control signal. A block diagram of this approach is shown in Figure 3.3.

A common application of the indirect approach is in Model Reference Adaptive Control (MRAC). In MRAC, the plant's behavior is compared to a reference model, and the controller is designed to make the plant emulate the reference model. This reference model is chosen to represent the desired relationship between the plant's input and output. The adaptive controller then adjusts the plant dynamics to align with those of the reference model. This approach will be employed throughout this text.

A frequent challenge in adaptive control is the issue of *parameter drift*. Parameter drift occurs when the input signal is not persistently exciting, meaning it lacks sufficient information to accurately estimate the parameters. As a result, the estimated parameters can drift away from their true values, potentially causing the control system to become

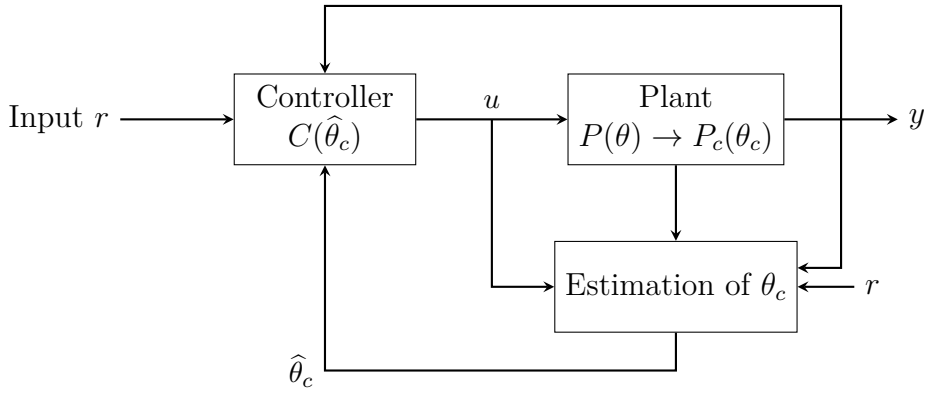


Figure 3.3: Block diagram of a direct adaptive control system.

unstable, even if it initially appears stable over some interval. This problem frequently arises in practice, and various methods have been proposed to mitigate it.

One of the simplest and most widely used techniques is the *dead-zone* or *deadband* method. This approach involves setting a threshold for parameter adaptation: if the tracking error falls below this threshold, the adaptation dynamics are disabled. For example, if γ is a positive constant, $\mathbf{Y}$ is a matrix related to a plant parametrization, and $\mathbf{e}$ is the tracking error, an adaptation law is expressed as $\dot{\widehat{\boldsymbol{\theta}}} = -\gamma \mathbf{Y}\mathbf{e}$. The dead-zone modification results in:

$$\dot{\widehat{\boldsymbol{\theta}}} = \begin{cases} -\gamma \mathbf{Y}\mathbf{e} & \|\mathbf{e}\| > \epsilon, \\ \mathbf{0} & \|\mathbf{e}\| \leq \epsilon, \end{cases}$$

where ϵ is the size of the dead-zone or deadband.

Another category of techniques designed to address parameter drift falls under the umbrella of *leakage modifications*, which are often based on Lyapunov stability analysis. These techniques modify the adaptation law to include a leakage term, resulting in the form:

$$\dot{\widehat{\boldsymbol{\theta}}} = -\gamma \mathbf{Y}\mathbf{e} - \gamma w \widehat{\boldsymbol{\theta}}, \tag{3.32}$$

where $w(t)$ is the *leakage term*.

One of the most common leakage modifications is the *σ-modification*, which introduces a small positive constant σ as the leakage term. When combined with the dead-zone technique, this modification leads to the *switching σ-modification*, where the leakage term $w(t)$ is defined as:

$$w(t) = \begin{cases} \sigma & \|\widehat{\boldsymbol{\theta}}\| \geq M, \\ 0 & \|\widehat{\boldsymbol{\theta}}\| < M, \end{cases} \tag{3.33}$$

Another related method is the ϵ-*modification*, where the leakage term depends on the norm of the tracking error, specifically

$$w(t) = \nu_0 \|\mathbf{e}\|, \tag{3.34}$$

where ν_0 is a positive constant.

In addition to these techniques, other methods exist to enhance robustness, as discussed in Ioannou and Sun (2012, ch. 8). For instance, *parameter projection* constrains the parameter estimates within predefined bounds, ensuring they do not grow unbounded. Similarly, *normalization* mitigates large parameter variations by normalizing the adaptation dynamics.

By employing these techniques and selecting an appropriate adaptation law, it is possible to design an adaptive controller that is both robust and suitable for practical applications.

4

Vector Fields in Matrix Lie Groups

Our main objective is to develop a vector field strategy that guides a system to follow a curve $\mathcal{C}$ in a Lie group. In this chapter, we build upon the concepts introduced in Section 3.2 to develop the tools necessary for generalizing the results from Rezende et al. (2022) to any connected matrix Lie group. We start by defining some useful operators that will aid in the development of our results. Next, we introduce two distance functions: one that measures the distance between arbitrary elements in the group and another that measures the distance between an element and a curve. These distance functions are fundamental to defining the vector field components. As we will show, the properties of orthogonality and the absence of local minima are directly related to the properties of the distance function. With all the concepts in place, we present our main theorem, which guarantees the convergence and circulation of the target curve.

In this chapter, we adopt G as a connected matrix Lie group of dimension m (the degrees of freedom of the group), whose representation is an $n \times n$ matrix. Given Lemma 3.36, we will *assume* that our control system is given by

$$\dot{\mathbf{H}}(t) = \mathcal{S}\Big(\boldsymbol{\xi}(t)\Big)\mathbf{H}(t), \tag{4.1}$$

in which $\mathbf{H} \in G$ is the state variable and $\boldsymbol{\xi} \in \mathbb{R}^m$ is the control input.

For a practical example, let $G = \mathrm{SE}(3)$ (thus $n = 4$, $m = 6$), with $\mathbf{H}$ representing the pose of an omnidirectional UAV in a fixed frame. By properly selecting the basis $\mathbf{E}_k$, $\boldsymbol{\xi}$ corresponds to a twist in the world frame, representing the UAV's linear and angular

velocities. This control system setup assumes arbitrary control over the UAV's 6-DoF twist, a reasonable model for an omnidirectional UAV. Analogously, we henceforth refer to $\boldsymbol{\xi}$ as the *generalized twist*.

Similar to the single integrator system model in Rezende et al. (2022), the model in (4.1) assumes maximal control freedom within the constraints of the Lie group. This means each component of $\boldsymbol{\xi}$ can be independently controlled, enabling arbitrary motion of an element $\mathbf{H} \in G$ while ensuring that $\mathbf{H}$ remains within the group. Additionally, both systems exhibit first-order dynamics.

Remark 4.1. The results presented in this chapter also apply to systems of the form $\dot{\mathbf{H}} = \mathbf{H}\mathcal{S}'(\boldsymbol{\xi}')$, where $\mathcal{S}'$ is an appropriate isomorphism mapping $\mathbb{R}^m$ to $\mathfrak{g}$. For instance, considering the UAV example, this system could model a UAV controlled via the twist in the *body frame* rather than the *fixed frame*. This adaptation can be achieved by rewriting (4.1) as $\dot{\mathbf{H}} = \mathbf{H}(\mathbf{H}^{-1}\mathcal{S}(\boldsymbol{\xi})\,\mathbf{H})$. It is well known in the context of Lie groups (see Gallier and Quaintance (2020, p. 86)) that for any $\mathbf{A} \in \mathfrak{g}$ and $\mathbf{H} \in G$, the term $\mathbf{H}^{-1}\mathbf{A}\mathbf{H}$ also belongs to $\mathfrak{g}$. Therefore, there exists a unique $\boldsymbol{\xi}' \in \mathfrak{g}$ such that $\mathcal{S}'(\boldsymbol{\xi}') = \mathbf{H}^{-1}\mathcal{S}(\boldsymbol{\xi})\,\mathbf{H}$, since $\mathcal{S}'$ is a bijection. This correspondence enables the calculation of a controller for the modified system based on the controller designed for the original system. □

We begin with the definition of the operators that will be used to extract information from the group elements.

4.1 Operators and derivatives

According to Lemma 3.36, for each differentiable function $\mathbf{G} : \mathbb{R} \to G$ there exists a respective function $\boldsymbol{\zeta} \in \mathbb{R}^m$ according to equation (3.31). Thus, we will define the following operator that extracts this $\boldsymbol{\zeta}(\sigma)$ from $\mathbf{G}(\sigma)$:

Definition 4.2 (Ξ operator)**.** Let G be an m-dimensional Lie group. Given a choice of S map $\mathcal{S} : \mathbb{R}^m \to \mathfrak{g}$, the respective Ξ operator maps a differentiable function $\mathbf{G} : \mathbb{R} \to G$ into a function $\Xi[\mathbf{G}] : \mathbb{R} \to \mathbb{R}^m$ as $\Xi[\mathbf{G}](\sigma) = \mathcal{S}^{-1}\Big(\frac{d\mathbf{G}}{d\sigma}(\sigma)\mathbf{G}(\sigma)^{-1}\Big)$. □

In our development, it will be necessary to take derivatives along the manifold G. This is related to the concept of *Lie derivatives*. For this purpose, the following definition will be useful.

Definition 4.3 (L operator)**.** Let G be an m-dimensional Lie group. Given a choice of S map and a differentiable function $f : G \to \mathbb{R}$, we define the L operator such that the function $\mathrm{L}[f] : G \to \mathbb{R}^{1\times m}$ satisfies:

$$\lim_{\varepsilon\to 0}\frac{1}{\varepsilon}\left(f\Big(\exp\big(\mathcal{S}(\boldsymbol{\zeta})\,\varepsilon\big)\mathbf{G}\Big) - f\Big(\mathbf{G}\Big)\right) = \mathrm{L}[f](\mathbf{G})\boldsymbol{\zeta}\ \forall\,\boldsymbol{\zeta} \in \mathbb{R}^m. \tag{4.2}$$

Explicitly, the j^{th} entry of the row vector $\mathrm{L}[f](\mathbf{G})$ can be constructed as the left-hand side of (4.2) when $\boldsymbol{\zeta} = \mathbf{e}_j$. The expression (4.2) is also representable with derivatives:

$$\frac{d}{d\varepsilon}\left(f\Big(\exp\big(\mathcal{S}(\boldsymbol{\zeta})\,\varepsilon\big)\mathbf{G}\Big)\right)\bigg|_{\varepsilon=0} = \mathrm{L}[f](\mathbf{G})\boldsymbol{\zeta}\ \forall\,\boldsymbol{\zeta}\in\mathbb{R}^m. \tag{4.3}$$

In addition, if $f : G \times G \to \mathbb{R}$ is a function of two variables, $f(\mathbf{V}, \mathbf{W})$, we define the *partial* L operators $\mathrm{L}_{\mathbf{V}}$ and $\mathrm{L}_{\mathbf{W}}$ analogously as in (4.2) but making the variation only in the first or in the second variable, respectively. □

The following version of the chain rule using the L operator can be established.

Lemma 4.4. *Let G be an m-dimensional Lie group. Let $\mathbf{G} : \mathbb{R} \to G$ and $f : G \to \mathbb{R}$ be differentiable functions. Then:*

$$\frac{d}{d\sigma} f\Big(\mathbf{G}(\sigma)\Big) = \mathrm{L}[f]\Big(\mathbf{G}(\sigma)\Big)\Xi[\mathbf{G}](\sigma). \tag{4.4}$$

Proof. Let $\boldsymbol{\zeta}(\sigma) = \Xi[\mathbf{G}](\sigma)$, according to Lemma 3.36 and Definition 4.2, we can write that for a small ε, $\mathbf{G}(\sigma+\varepsilon) \approx \exp(\mathcal{S}(\boldsymbol{\zeta}(\sigma))\,\varepsilon)\mathbf{G}(\sigma)$. Applying the definition of the traditional derivative:

$$\begin{aligned}\frac{d}{d\sigma} f\Big(\mathbf{G}(\sigma)\Big) &= \lim_{\varepsilon\to 0}\frac{f\Big(\mathbf{G}(\sigma+\varepsilon)\Big) - f\Big(\mathbf{G}(\sigma)\Big)}{\varepsilon} \\ &= \lim_{\varepsilon\to 0}\frac{f\Big(\exp(\mathcal{S}(\boldsymbol{\zeta}(\sigma))\,\varepsilon)\mathbf{G}(\sigma)\Big) - f\Big(\mathbf{G}(\sigma)\Big)}{\varepsilon} = \mathrm{L}[f](\mathbf{G})\boldsymbol{\zeta}(\sigma),\end{aligned} \tag{4.5}$$

in which the defining property of $\mathrm{L}[f]$ in equation (4.2) was applied. This concludes the proof. □

As a consequence of Lemma 4.4, we can establish the following corollary.

Corollary 4.5. *If we have a function $f : G \times G \to \mathbb{R}$ instead of a function of a single variable, and two differentiable functions $\mathbf{V}, \mathbf{W} : \mathbb{R} \to G$, then:*

$$\frac{d}{d\sigma} f(\mathbf{V}, \mathbf{W}) = \mathrm{L}_{\mathbf{V}}[f]\Xi[\mathbf{V}] + \mathrm{L}_{\mathbf{W}}[f]\Xi[\mathbf{W}], \tag{4.6}$$

in which the dependency on $\mathbf{V}, \mathbf{W}$ was omitted on the right-hand side.

Further properties of the L operator are investigated in Appendix A.

4.2 Vector field formulation

Following the same steps as in Section 3.1, we propose a vector field strategy that ensures both convergence to and circulation around a curve $\mathcal{C}$ defined in a Lie group G, adopting the

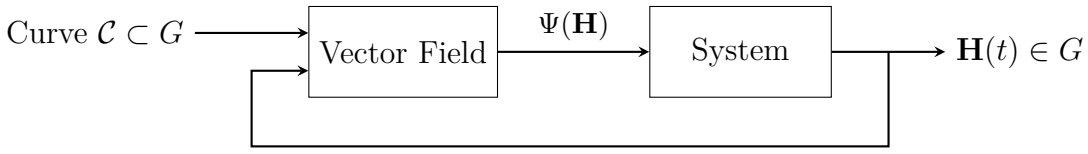


Figure 4.1: Block diagram of the kinematic control using the vector field guidance.

system in (4.1). We assume that this curve is differentiable and without self-intersections. Thus, we aim to synthesize a state feedback control law $\boldsymbol{\xi} = \Psi(\mathbf{H})$ to achieve this. Let $\mathbf{H}_d : [0, 1] \to G$ be a differentiable parametrization for the target curve $\mathcal{C}$. The proposed vector field is then expressed as:

$$\Psi(\mathbf{H}) \triangleq k_N(\mathbf{H})\boldsymbol{\xi}_N(\mathbf{H}) + k_T(\mathbf{H})\boldsymbol{\xi}_T(\mathbf{H}), \tag{4.7}$$

where the *normal* component $\boldsymbol{\xi}_N$ ensures convergence, and the *tangent* component $\boldsymbol{\xi}_T$ governs circulation. These components will be formally defined later. In this case, $k_N : G \to \mathbb{R}$ and $k_T : G \to \mathbb{R}$ are continuous functions in which $k_T(\mathbf{H})$ is positive and $k_N(\mathbf{H}) = 0$ when $\mathbf{H} \in \mathcal{C}$ and $k_N(\mathbf{H}) > 0$ otherwise. The block diagram of this control strategy is shown in Figure 4.1.

Remark 4.6. The proposed results generalize the vector field approach from Rezende et al. (2022), as outlined in Section 3.1. Throughout this section, we will specify the choices required to reduce the proposed approach to that of Rezende et al. (2022). To align with their results, the group G in our framework should be taken as the *m-dimensional translation group*, denoted by $\mathrm{T}(m)$ (see Example 3.13). Notably, Rezende et al. (2022) does not use the Lie group formalism, instead working with vectors – a feasible approach because each element of $\mathrm{T}(m)$ corresponds uniquely to a vector in $\mathbb{R}^m$. To facilitate the connection between both works, we will also adopt this vector representation, using the isomorphism $\mathcal{T} : \mathrm{T}(m) \to \mathbb{R}^m$, where $\mathcal{T}(\mathbf{H})$ is obtained by extracting the first m elements of the last column of $\mathbf{H}$. Henceforth, we take as a basis of the Lie algebra of $\mathrm{T}(m)$ the matrices $\mathbf{E}_k$, $k \in [1, m]$, where $\mathbf{E}_k$ has all entries 0 except for the k^{th} entry of the last column, which is 1. With this choice, the system (4.1) reduces to the single integrator model used in Rezende et al. (2022), as $\frac{d}{dt}\mathcal{T}(\mathbf{H}) = \boldsymbol{\xi}$. □

For the vector field computation, we need to measure the distance between an element and a curve within the Lie group. Thus, we first define a distance function $\widehat{D}$ between arbitrary elements $\mathbf{V}$ and $\mathbf{W}$ in the group, as follows.

Definition 4.7 (EE-distance function)**.** Let G be a Lie group. We call $\widehat{D} : G \times G \to \mathbb{R}_+$ an *Element-to-Element (EE-)distance function*, a function that measures the distance between elements $\mathbf{V}, \mathbf{W} \in G$ with the following properties:

i (Positive Definiteness) $\widehat{D}(\mathbf{V}, \mathbf{W}) \geq 0$ and $\widehat{D}(\mathbf{V}, \mathbf{W}) = 0 \iff \mathbf{V} = \mathbf{W}$;

ii (Differentiability) $\widehat{D}$ is at least once differentiable in both arguments almost ev-

erywhere, that is, the limit in (4.2) should exist. In addition, there should exist $D_{\min,\mathcal{C}} > 0$ such that the derivative exists when $0 < \widehat{D} < D_{\min,\mathcal{C}}$. Finally, where the derivative does not exist, every directional limit should exist (although they do not need to be equal) and be bounded.

□

Example 4.8. To obtain the results in Rezende et al. (2022), $\widehat{D}$ should be taken as the Euclidean distance between the respective position vectors $\widehat{D}(\mathbf{V}, \mathbf{W}) = \|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\|$. □

By allowing the function to be non-differentiable in certain cases, we can incorporate important distance functions, such as the Euclidean distance $\widehat{D}(\mathbf{V}, \mathbf{W}) = \|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\|$ when $G = \mathrm{T}(m)$. This is not differentiable at $\mathbf{V} = \mathbf{W}$, but all directional limits of the derivatives exist and are bounded. Furthermore, although it is not differentiable when $\widehat{D} = 0$, it is differentiable everywhere else, so $D_{\min,\mathcal{C}} = \infty$ can be taken. Overall, the (possible) non-differentiability when $\mathbf{V} = \mathbf{W}$ for a generic $\widehat{D}$ will not be an issue since it will be canceled in the final controller, as will be clear soon.

Now, a distance between an element $\mathbf{H}$ to the curve $\mathcal{C}$ is defined as the minimum distance, as measured by $\widehat{D}$, between $\mathbf{H}$ and any $\mathbf{Y}$ in the curve.

Definition 4.9 (EC-Distance Function)**.** Given an EE-distance function as in Definition 4.7, an *Element-to-Curve (EC-)*distance function $D : G \to \mathbb{R}_+$ measures the distance between an element $\mathbf{H}$ and a curve $\mathcal{C}$ parameterized by $\mathbf{H}_d(s)$. It is defined as:

$$D(\mathbf{H}) \triangleq \min_{\mathbf{Y}\in\mathcal{C}} \widehat{D}(\mathbf{H}, \mathbf{Y}) = \min_{s\in[0,1]} \widehat{D}\Big(\mathbf{H}, \mathbf{H}_d(s)\Big). \tag{4.8}$$

Furthermore, let $\mathcal{P}_1 \subset G$ be the set of $\mathbf{H}$ such that the optimization problem in (4.8) does not have a unique solution. In addition, we define $s^* : G \to [0, 1]$ so, when $\mathbf{H} \notin \mathcal{P}_1$, $s^*(\mathbf{H})$ is the unique minimizer on s for the given $\mathbf{H}$. We also define $\mathcal{P}_2$ as the set of $\mathbf{H} \notin (\mathcal{P}_1 \cup \mathcal{C})$ such that $\mathbf{V} = \mathbf{H}$, $\mathbf{W} = \mathbf{H}_d\Big(s^*(\mathbf{H})\Big)$ is a non-differentiability point of $\widehat{D}(\mathbf{V}, \mathbf{W})$. Finally, we define $\mathcal{P} \triangleq \mathcal{P}_1 \cup \mathcal{P}_2$. □

Note that, from the definition of $\mathcal{P}$, $\mathcal{C} \cap \mathcal{P} = \emptyset$. Furthermore, the fact that $D_{\min,\mathcal{C}} > 0$ in Definition 4.7 means that there is a non-zero separation distance between them. For the sake of notational simplicity, we will often omit the dependency of s^* on $\mathbf{H}$ and write simply s^* instead of $s^*(\mathbf{H})$.

Before we define the vector field components, we will state a lemma that will be extensively used to prove many other lemmas and propositions throughout this paper. For that, it will be useful to define $\boldsymbol{\xi}_d$, the necessary *generalized twist* $\boldsymbol{\xi}$ in (4.1) to follow the desired curve at a given $\mathbf{H} = \mathbf{H}_d(s)$.

Definition 4.10 (Curve twist)**.** Let $\boldsymbol{\xi}_d(s) \triangleq \Xi[\mathbf{H}_d](s)$ (see Definition 4.2). Furthermore, we call the parametrization $\mathbf{H}_d(s)$ *proper* if $\boldsymbol{\xi}_d(s) \neq \mathbf{0}$ for all $s \in [0, 1]$. □

Having a proper parametrization is a purely geometric feature of the curve $\mathcal{C}$. The non-self-intersection property of the curve $\mathcal{C}$ implies that $\mathbf{H}_d : [0,1] \to \mathcal{C}$ is bijective and thus any non-proper parametrization can be transformed into a proper one through reparametrization (e.g., arc length parametrization). With this definition, we can proceed with the following useful lemma.

Lemma 4.11. *When $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$, the first order optimality condition of (4.8) implies that*

$$\mathrm{L}_{\mathbf{W}}[\widehat{D}]\Big(\mathbf{H}, \mathbf{H}_d(s^*)\Big)\boldsymbol{\xi}_d(s^*) = 0. \tag{4.9}$$

Proof. Let $\mathbf{H} \in G$, $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$ be an arbitrary element, $\mathbf{H}_d(s) \in G$ the parametrization of a curve, and D an EC-distance defined as in Definition 4.9. Since D is the minimum value of an EE-distance function $\widehat{D}$ (see Definition 4.7), by the first order optimality condition of (4.8) we know that the optimal parameter s^* will render $\frac{d}{ds}\widehat{D}(\mathbf{H}, \mathbf{H}_d(s))|_{s=s^*} = 0$. Taking the derivative at $s = s^*$, using Corollary 4.5, Definition 4.10 and setting it to zero, we obtain the desired result. Furthermore, since $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$, the function $\widehat{D}$ is guaranteed to be differentiable. □

4.3 Normal component

Following the same steps as in Section 3.1, after defining our EC-distance function, we state our vector field components. To define our *normal* component, we build upon Feature (a) (in page 13). By differentiating D, we denote the opposite of the term that multiplies the generalized twist $\boldsymbol{\xi}$ as the normal component. We begin with the following lemma.

Lemma 4.12. *When $\mathbf{H}(t) \notin \mathcal{C} \cup \mathcal{P}$, the time derivative of an EC-distance $D(\mathbf{H}(t))$, under the dynamics in (4.1), is given by*

$$\frac{d}{dt}D = \mathrm{L}[D](\mathbf{H})\boldsymbol{\xi} = \mathrm{L}_{\mathbf{V}}[\widehat{D}]\Big(\mathbf{H}, \mathbf{H}_d(s^*)\Big)\boldsymbol{\xi}. \tag{4.10}$$

Proof. The first part of the equation comes from the chain rule in Lemma 4.4 and (4.1). For the second equality, use the fact that $D(\mathbf{H}) = \widehat{D}\Big(\mathbf{H}, \mathbf{H}_d(s^*(\mathbf{H}))\Big)$ and differentiate applying the chain rule using Corollary 4.5:

$$\dot{D} = \mathrm{L}_{\mathbf{V}}[\widehat{D}]\boldsymbol{\xi} + \Big(\mathrm{L}_{\mathbf{W}}[\widehat{D}]\boldsymbol{\xi}_d\Big)\frac{ds^*}{dt} \tag{4.11}$$

in which the dependencies of $\mathrm{L}_{\mathbf{V}}$ and $\mathrm{L}_{\mathbf{W}}$ on $\mathbf{H}$ and $\mathbf{H}_d(s^*)$ were omitted. By Lemma 4.11, the term within parenthesis in (4.11) vanishes and we obtain the desired result. Note that (4.10) shows that the derivative exists and is continuous whenever the remaining term is continuous; that is, when $\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{H}, \mathbf{H}_d(s^*))$ is continuous, i.e., when $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$. □

Now, (4.10) in Lemma 4.12 allows us to define the normal component.

Definition 4.13 (Normal component)**.** When $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$, the *normal* component of the vector field $\boldsymbol{\xi}_N : G \to \mathbb{R}^m$ is defined as the negative transpose of the term that multiplies $\boldsymbol{\xi}$ in (4.10), i.e.,

$$\boldsymbol{\xi}_N(\mathbf{H}) \triangleq -\mathrm{L}_{\mathbf{V}}[\widehat{D}]\Big(\mathbf{H}, \mathbf{H}_d(s^*)\Big)^\top. \tag{4.12}$$

□

When $\mathbf{H} \in \mathcal{C} \cup \mathcal{P}$, $\boldsymbol{\xi}_N(\mathbf{H})$ is left undefined. As we will see in Section 4.7, this lack of definition at $\mathcal{C}$ is not an issue, as it will not be necessary to define it in that context.

Finally, note that the previous definition allows us to express, as in Feature (a), that $\dot{D} = -\boldsymbol{\xi}_N(\mathbf{H})^\top \boldsymbol{\xi}$, provided that $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$. In Rezende et al. (2022), it was possible to write $\dot{D} = (\nabla D)^\top \boldsymbol{\xi}$, where the gradient is taken with respect to the state, as the system state in that case lies in $\mathbb{R}^m$, which lacks nonlinear manifold constraints. However, our case is more general since the state $\mathbf{H}$ lies on an m-dimensional (non-linear, in general) manifold embedded in a space with higher dimension n^2. Thus, by comparing $\dot{D} = (\nabla D)^\top \boldsymbol{\xi}$ with $\dot{D} = -\boldsymbol{\xi}_N(\mathbf{H})^\top \boldsymbol{\xi}$, we can see that $-\boldsymbol{\xi}_N(\mathbf{H})$ serves as the "gradient of the distance function" in this constrained setting.

Example 4.14. Applying Definition 4.13, the normal component in Rezende et al. (2022) will be given by

$$\boldsymbol{\xi}_N(\mathbf{H}) = \frac{\mathcal{T}\Big(\mathbf{H}_d(s^*)\Big) - \mathcal{T}(\mathbf{H})}{\Big\|\mathcal{T}\Big(\mathbf{H}_d(s^*)\Big) - \mathcal{T}(\mathbf{H})\Big\|},$$

i.e., the normalized vector that points from the current point to the nearest point on the curve. □

4.4 Tangent component

As in Section 3.1, the *tangent* component of our vector field is associated solely with the curve. It can be easily defined using Definition 4.10.

Definition 4.15 (Tangent component)**.** For $\mathbf{H} \notin \mathcal{P}$, the *tangent* component of the vector field, $\boldsymbol{\xi}_T : G \to \mathbb{R}^m$ is defined as $\boldsymbol{\xi}_T(\mathbf{H}) \triangleq \boldsymbol{\xi}_d(s^*(\mathbf{H}))$. □

Example 4.16. In Rezende et al. (2022), the tangent component is precisely the tangent vector of the curve at the nearest point, so $\boldsymbol{\xi}_T = \frac{d}{ds}\mathcal{T}(\mathbf{H}_d(s))\Big|_{s=s^*}$. However, this is not necessarily true in our more general case. Here, the tangent component represents the generalized twist required at the nearest point $\mathbf{H}_d(s^*)$ on the curve for the system, under the dynamics of (4.1), to move along the curve. □

4.5 Orthogonality of components

According to Feature (b), it is necessary that the normal and tangent components of the vector field be orthogonal to each other. This fact is related only to the EC-distance function D, consequently to the EE-distance function $\widehat{D}$, and can be achieved through a property of *left-invariance*. We will first define a *left-invariant* distance function and then provide a proposition for the orthogonality condition.

Definition 4.17 (Left-invariant distance)**.** An EE-distance function (Definition 4.7) $\widehat{D}: G \times G \to \mathbb{R}_+$ is said to be *left-invariant* if it also satisfies $\widehat{D}(\mathbf{XV}, \mathbf{XW}) = \widehat{D}(\mathbf{V}, \mathbf{W})$ for all $\mathbf{V}, \mathbf{W}, \mathbf{X} \in G$. □

Given this definition, we can state the following:

Proposition 4.18. *Let $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$. If $\widehat{D}$ is a left-invariant distance function (Definition 4.17), then the vector field components (Definitions 4.13 and 4.15) will be orthogonal to each other, i.e., $\boldsymbol{\xi}_N(\mathbf{H})^\top \boldsymbol{\xi}_T(\mathbf{H}) = 0$.*

Proof. Since $\widehat{D}$ is left-invariant, then $\widehat{D}(\mathbf{XV}, \mathbf{XW}) = \widehat{D}(\mathbf{V}, \mathbf{W})\ \forall\ \mathbf{V}, \mathbf{W}, \mathbf{X} \in G$. Since this holds for any value, take $\mathbf{V} = \mathbf{H}$, $\mathbf{W} = \mathbf{H}_d(s^*)$, and $\mathbf{X} = \exp\Big(\tau\, \mathcal{S}(\boldsymbol{\xi}_T)\Big)$. Now, due to the left-invariance,

$$\widehat{D}\Big(\exp\Big(\tau\, \mathcal{S}(\boldsymbol{\xi}_T)\Big)\mathbf{H},\ \exp\Big(\tau\, \mathcal{S}(\boldsymbol{\xi}_T)\Big)\mathbf{H}_d(s^*)\Big) = \widehat{D}\Big(\mathbf{H}, \mathbf{H}_d(s^*)\Big)\ \forall\ \mathbf{H} \in G,\ \tau \in \mathbb{R}. \tag{4.13}$$

Differentiating both sides of this equation with respect to τ, using Corollary 4.5, and evaluating at $\tau = 0$ gives

$$\mathrm{L}_{\mathbf{V}}[\widehat{D}]\boldsymbol{\xi}_T + \mathrm{L}_{\mathbf{W}}[\widehat{D}]\boldsymbol{\xi}_T = 0, \tag{4.14}$$

in which the dependencies of $\mathrm{L}_{\mathbf{V}}[\widehat{D}], \mathrm{L}_{\mathbf{W}}[\widehat{D}]$ on $\mathbf{H}$ and $\mathbf{H}_d(s^*)$ were omitted. Noting that, by Definition 4.15, $\boldsymbol{\xi}_T = \boldsymbol{\xi}_d(s^*)$, and invoking Lemma 4.11, implies that $\mathrm{L}_{\mathbf{W}}[\widehat{D}]\boldsymbol{\xi}_T = 0$. Finally, using Definition 4.13, we prove the orthogonality property: $\mathrm{L}_{\mathbf{V}}[\widehat{D}]\boldsymbol{\xi}_T = -\boldsymbol{\xi}_N^\top \boldsymbol{\xi}_T = 0$ □

Example 4.19. In Rezende et al. (2022), the EE-distance function $\widehat{D}(\mathbf{V}, \mathbf{W}) = \|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\|$ (Example 4.8) is left-invariant. Note that $\mathcal{T}(\mathbf{XY}) = \mathcal{T}(\mathbf{X}) + \mathcal{T}(\mathbf{Y})\ \forall\ \mathbf{X}, \mathbf{Y} \in \mathrm{T}(m)$. Let $\mathbf{V}, \mathbf{W}, \mathbf{X} \in \mathrm{T}(m)$, consequently, $\widehat{D}(\mathbf{XV}, \mathbf{XW}) = \|\mathcal{T}(\mathbf{XV}) - \mathcal{T}(\mathbf{XW})\| = \|\mathcal{T}(\mathbf{X}) + \mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{X}) - \mathcal{T}(\mathbf{W})\| = \widehat{D}(\mathbf{V}, \mathbf{W})$. □

4.6 Local minima and gradients in distance function

Feature (c) is the only feature remaining to be present in our formulation. It consists of two parts: the absence of local minima outside the curve, and the fact that the gradient, here represented by its general form $-\boldsymbol{\xi}_N(\mathbf{H})$, never vanishes (whenever it exists).

In order for the EC-distance function to lack local minima outside the curve, we introduce the concept of a *chainable* distance function and then prove that the *chainability*

property leads to a distance function without local minima outside the curve. This property requires defining a *path* between elements in a Lie group.

Definition 4.20 (Path)**.** In a Lie group G, a *path* $\Phi : [0,1] \times G \times G \to G$ connecting an element $\mathbf{V}$ to an element $\mathbf{W}$ satisfies the following properties for all $\mathbf{V}, \mathbf{W} \in G$, and $\sigma \in [0,1]$:

i $\Phi(\sigma, \mathbf{V}, \mathbf{W})$ is differentiable in σ;
ii $\Phi(0, \mathbf{V}, \mathbf{W}) = \mathbf{V}$, $\Phi(1, \mathbf{V}, \mathbf{W}) = \mathbf{W}$.

□

With the definition of a path, we can now define a chainable distance function.

Definition 4.21 (Chainable distance)**.** A function $\widehat{D} : G \times G \to \mathbb{R}_+$ is called a *chainable* distance if it meets the criteria of an EE-distance function (Definition 4.7) and satisfies the following property. Specifically, there exists a path Φ (Definition 4.20), such that for any points $\mathbf{V}, \mathbf{W} \in G$ and any $\sigma \in [0,1]$:

$$\widehat{D}(\mathbf{V}, \mathbf{W}) = \widehat{D}\Big(\mathbf{V}, \Phi(\sigma, \mathbf{V}, \mathbf{W})\Big) + \widehat{D}\Big(\Phi(\sigma, \mathbf{V}, \mathbf{W}), \mathbf{W}\Big).$$

□

A chainable distance between two elements can be thought as a chain, such that it can be broken into pieces within the specific path Φ and render the same result.

Example 4.22. The EE-distance function $\widehat{D}(\mathbf{V}, \mathbf{W}) = \|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\|$ from Rezende et al. (2022) (see Example 4.8) is also chainable. To demonstrate this, we first define an appropriate path in accordance with Definition 4.20. We use $\Phi(\sigma, \mathbf{V}, \mathbf{W})$ so $\mathcal{T}(\Phi(\sigma, \mathbf{V}, \mathbf{W}))$ $= (1-\sigma)\mathcal{T}(\mathbf{V}) + \sigma\mathcal{T}(\mathbf{W})$ $\forall\, \mathbf{V}, \mathbf{W} \in \mathrm{T}(m)$ (i.e., a linear path).

Given the defined path, we show that the EE-distance function $\widehat{D}$ is chainable as follows:

$$\widehat{D}(\mathbf{V}, \Phi_\sigma) = \|\mathcal{T}(\mathbf{V}) - (1-\sigma)\mathcal{T}(\mathbf{V}) - \sigma\mathcal{T}(\mathbf{W})\| = \sigma\|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\|, \tag{4.15}$$

$$\widehat{D}(\Phi_\sigma, \mathbf{W}) = \|(1-\sigma)\mathcal{T}(\mathbf{V}) + \sigma\mathcal{T}(\mathbf{W}) - \mathcal{T}(\mathbf{W})\| = (1-\sigma)\|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\|, \tag{4.16}$$

in which $\Phi_\sigma = \Phi(\sigma, \mathbf{V}, \mathbf{W})$ and the fact that $0 \leq \sigma \leq 1$ was used. Summing both terms and comparing them, the chainability property is evident. □

It will now be proved that the chainability property in Definition 4.21 implies the absence of local minima outside the curve in the EC-distance function. This fact will become useful when we prove the convergence to the curve.

Proposition 4.23. *If $\widehat{D}$ is a chainable distance function (Definition 4.21), then it does not have local minima outside the curve $\mathcal{C}$.*

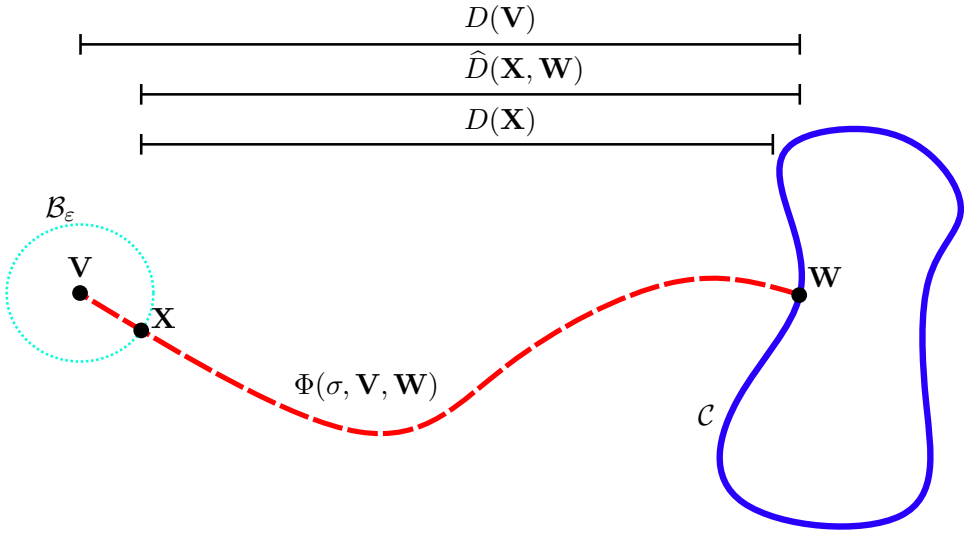


Figure 4.2: Depiction of the proof in Proposition 4.23

Proof. The proof proceeds by contradiction and is depicted in Figure 4.2. Take D as an EC-distance function (Definition 4.9) with $\widehat{D}$ being a chainable EE-distance function. Assume that $\mathbf{V} \notin \mathcal{C}$ is a local minimum of D outside the curve, and let $\mathbf{W} \in G$ be (one of) the nearest point(s) on $\mathcal{C}$ to $\mathbf{V}$, i.e., $\mathbf{W} \in \arg\min_{\mathbf{Y}\in\mathcal{C}} \widehat{D}(\mathbf{V},\mathbf{Y})$.

Since $\mathbf{V} \notin \mathcal{C}$ is a local minimum of D, there exists a ball $\mathcal{B}_\varepsilon$, with respect to the topology induced by the distance function, of radius $\varepsilon > 0$, small enough to not touch the curve, centered at $\mathbf{V}$ such that $D(\mathbf{Y}) \geq D(\mathbf{V})\ \forall\ \mathbf{Y} \in \mathcal{B}_\varepsilon$. Given that a path Φ between $\mathbf{V}$ and $\mathbf{W}$ is continuous (see Definition 4.20), there must exist $\sigma_0 \in (0,1)$ such that a point $\mathbf{X} = \Phi(\sigma_0,\mathbf{V},\mathbf{W})$ lies in the intersection of the boundary of the ball $\partial\mathcal{B}_\varepsilon$ and the path. According to our assumption, $D(\mathbf{V}) \leq D(\mathbf{X})$, where $D(\mathbf{X}) = \min_{\mathbf{Y}\in\mathcal{C}} \widehat{D}(\mathbf{X},\mathbf{Y})$. Now, since $D(\mathbf{X})$ is the minimum EE-distance between $\mathbf{X}$ and the curve, it must be true that this distance is less than or equal to the EE-distance between $\mathbf{X}$ and $\mathbf{W}$, i.e., $D(\mathbf{X}) \leq \widehat{D}(\mathbf{X},\mathbf{W})$. The results until now allows us to obtain the following:

$$D(\mathbf{V}) \leq D(\mathbf{X}) \leq \widehat{D}(\mathbf{X},\mathbf{W}). \tag{4.17}$$

In addition, by the chainability property (see Definition 4.21), we also have $D(\mathbf{V}) \geq \widehat{D}(\Phi(\sigma,\mathbf{V},\mathbf{W}),\mathbf{W})\ \forall\ \sigma \in [0,1]$. Since this holds for any value of σ, take $\sigma = \sigma_0$, which results in

$$D(\mathbf{V}) \geq \widehat{D}(\mathbf{X},\mathbf{W}). \tag{4.18}$$

We now force a contradiction. Since conditions (4.17) and (4.18) must hold simultaneously, it follows that $D(\mathbf{V}) = \widehat{D}(\mathbf{X},\mathbf{W})$. But, from the chainability property (see Definition 4.21), we know that $D(\mathbf{V}) = \widehat{D}(\mathbf{V},\mathbf{W}) = \widehat{D}(\mathbf{V},\mathbf{X}) + \widehat{D}(\mathbf{X},\mathbf{W})$. This implies that $\widehat{D}(\mathbf{V},\mathbf{X}) = 0$, therefore, since $\widehat{D}$ is positive definite, it follows that $\mathbf{V} = \mathbf{X}$, contradicting the existence

of such a ball $\mathcal{B}_\varepsilon$. □

We now establish that, under certain mild conditions, $-\boldsymbol{\xi}_N(\mathbf{H})$, which plays the role of the gradient of the distance in our case, never vanishes wherever it is defined. This result is evident in Rezende et al. (2022), as demonstrated in Example 4.14. The necessary "mild" condition for establishing this is as follows.

Definition 4.24 (Locally linear)**.** A chainable EE-distance $\widehat{D}$ is said to be *locally linear* if, for any $\mathbf{V}, \mathbf{W} \in G$, $\mathbf{V} \neq \mathbf{W}$:

$$\lim_{\sigma \to 0^+} \frac{1}{\sigma} \widehat{D}\Big(\mathbf{V}, \Phi(\sigma, \mathbf{V}, \mathbf{W})\Big) > 0. \tag{4.19}$$

□

The name arises from the fact that, for any $\mathbf{V} \neq \mathbf{W}$, $\widehat{D}\Big(\mathbf{V}, \Phi(\sigma, \mathbf{V}, \mathbf{W})\Big) \approx o(\sigma)$ (i.e., it is approximately linear in the small "o" notation).

Example 4.25. Let $\widehat{D}$ be the EE-distance as in Example 4.8 and Φ the path in Example 4.22. From (4.15), we know that $\widehat{D}(\mathbf{V}, \Phi(\sigma, \mathbf{V}, \mathbf{W})) = \sigma \|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\|$, thus $\lim_{\sigma \to 0^+} \frac{1}{\sigma} \widehat{D}\Big(\mathbf{V}, \Phi(\sigma, \mathbf{V}, \mathbf{W})\Big) = \lim_{\sigma \to 0^+} \frac{1}{\sigma} \sigma \|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\| = \|\mathcal{T}(\mathbf{V}) - \mathcal{T}(\mathbf{W})\| > 0$ for $\mathbf{V} \neq \mathbf{W}$, and thus $\widehat{D}$ is locally linear. □

Lemma 4.26. *If $\widehat{D}$ is chainable (Definition 4.21) and locally linear (Definition 4.24), for any $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$, $\boldsymbol{\xi}_N(\mathbf{H}) \neq \mathbf{0}$.*

Proof. Let $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$. Choosing $\mathbf{V} = \mathbf{H}$, $\mathbf{W} = \mathbf{H}_d(s^*) = \mathbf{H}^*$, and using the property of chainable functions:

$$D(\mathbf{H}) = \widehat{D}(\mathbf{H}, \mathbf{H}^*) = \widehat{D}\Big(\mathbf{H}, \Phi(\sigma, \mathbf{H}, \mathbf{H}^*)\Big) + \widehat{D}\Big(\Phi(\sigma, \mathbf{H}, \mathbf{H}^*), \mathbf{H}^*\Big). \tag{4.20}$$

Now, $\widehat{D}\Big(\Phi(\sigma, \mathbf{H}, \mathbf{H}^*), \mathbf{H}^*\Big) \geq D\Big(\Phi(\sigma, \mathbf{H}, \mathbf{H}^*)\Big)$. Thus:

$$D(\mathbf{H}) - D\Big(\Phi(\sigma, \mathbf{H}, \mathbf{H}^*)\Big) \geq \widehat{D}\Big(\mathbf{H}, \Phi(\sigma, \mathbf{H}, \mathbf{H}^*)\Big). \tag{4.21}$$

Divide both sides by $\sigma > 0$ and take the limit as $\sigma \to 0^+$. Since $\mathbf{H} \notin \mathcal{C}$, $\mathbf{H} \neq \mathbf{H}^*$, and, using Definition 4.24:

$$\lim_{\sigma \to 0^+} \frac{D(\mathbf{H}) - D\Big(\Phi(\sigma, \mathbf{H}, \mathbf{H}^*)\Big)}{\sigma} > 0. \tag{4.22}$$

Let $\boldsymbol{\xi}_\Phi(\sigma, \mathbf{H}) \triangleq \Xi[\Phi](\sigma, \mathbf{H})$. Then, from Lemma 3.36, it is true that $\Phi(\sigma, \mathbf{H}, \mathbf{H}^*) \approx \exp(\mathcal{S}(\boldsymbol{\xi}_\Phi)\, \sigma)\mathbf{H}$ for $\sigma \approx 0$. Consequently, the left-hand side of (4.22) can be written as:

$$-\lim_{\sigma \to 0^+} \left(\frac{D\Big(\exp(\mathcal{S}(\boldsymbol{\xi}_\Phi)\, \sigma)\mathbf{H}\Big) - D(\mathbf{H})}{\sigma} \right). \tag{4.23}$$

Using Definition 4.3, this last limit, whenever it exists, is exactly $-\mathrm{L}[D](\mathbf{H})\boldsymbol{\xi}_\Phi$. This limit exists when $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$. But, from Lemma 4.12 and Definition 4.13, it can be seen that this limit is also $\boldsymbol{\xi}_N(\mathbf{H})^\top \boldsymbol{\xi}_\Phi$. Thus, from (4.22), $\boldsymbol{\xi}_N(\mathbf{H})^\top \boldsymbol{\xi}_\Phi > 0$, which implies that $\boldsymbol{\xi}_N(\mathbf{H}) \neq \mathbf{0}$. □

Intuitively, this lemma means that from any point $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$ we can always move in the direction of the closest point $\mathbf{H}^*$ along the path Φ by applying the twist $\boldsymbol{\xi}_\Phi$ in (4.1). This motion decreases D sufficiently to ensure that the derivative does not vanish, which means that $\boldsymbol{\xi}_N(\mathbf{H})$ cannot be zero, since $\dot{D} = -\boldsymbol{\xi}_N(\mathbf{H})^\top \boldsymbol{\xi}$ for any arbitrary twist $\boldsymbol{\xi}$ (Lemma 4.12).

4.7 Convergence results

With the established definitions, lemmas, and propositions, we can now prove the main result of this text. To do so, we first require the following lemma.

Lemma 4.27. *If $\widehat{D}$ is an EE-distance function (Definition 4.7), then*

$$\lim_{\mathbf{H}\to\mathcal{C}} k_N(\mathbf{H})\boldsymbol{\xi}_N(\mathbf{H}) = \mathbf{0}. \tag{4.24}$$

Proof. From Definition 4.13, when $\mathbf{H} \to \mathcal{C}$, the quantity $\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{V}, \mathbf{W})$ is evaluated when $\mathbf{V} = \mathbf{W}$ (i.e., $\mathbf{H} = \mathbf{H}_d(s^*)$). According to Definition 4.7, this derivative does not necessarily exist, but all directional limit should exist and be bounded. Since $k_N(\mathbf{H}) = 0$ when $\mathbf{H} \to \mathcal{C}$, this concludes the result. □

This lemma shows that the vector field in (4.7) remains well-defined when $\mathbf{H} \in \mathcal{C}$, even though $\boldsymbol{\xi}_N(\mathbf{H})$ is undefined at these points. Now, we present our theorem:

Theorem 4.28. *Let $\widehat{D}$ be a* left-invariant *(Definition 4.17) ,* chainable *(Definition 4.21) and* locally linear *(Definition 4.24) EE-distance function, and $\mathbf{H}_d(s)$ a proper (Definition 4.10) parametrization for $\mathcal{C}$. Then, the closed loop autonomous dynamical system in (4.1), with the input given by (4.7), is such that*

(i) the system's state converges either to $\mathcal{C}$ or $\mathcal{P}$;

(ii) the set $\mathcal{P}$ is "escapable": there exists a policy of choosing arbitrarily small $\boldsymbol{\xi}$ every time $\mathbf{H} \in \mathcal{P}$ such that there exists a finite t' in which $\mathbf{H}(t) \notin \mathcal{P}$ for all $t \geq t'$;

(iii) if the system's state converges to $\mathcal{C}$, $\mathbf{H}$ circulates the target curve.

Proof. **Statement (i):** Consider D as in Definition 4.9 as a Lyapunov function candidate. According to Lemma 4.12 and Definition 4.13, its derivative is given by $\dot{D} = -\boldsymbol{\xi}_N^\top \boldsymbol{\xi}$ for any

$\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$. Substituting $\boldsymbol{\xi} = \Psi = k_N \boldsymbol{\xi}_N + k_T \boldsymbol{\xi}_T$ and using Proposition 4.18, we obtain

$$\frac{d}{dt} D(\mathbf{H}) = -k_N(\mathbf{H}) \|\boldsymbol{\xi}_N(\mathbf{H})\|^2 \; \forall \; \mathbf{H} \notin \mathcal{C} \cup \mathcal{P} \tag{4.25}$$

Note, however, that Lemma 4.27 guarantees that $\dot{D}$ will also vanish when $\mathbf{H} \in \mathcal{C}$. This fact, along with (4.25) and Lemma 4.26, shows that $\dot{D} < 0$ for all $\mathbf{H} \notin \mathcal{C} \cup \mathcal{P}$, $\dot{D} = 0$ for $\mathbf{H} \in \mathcal{C}$, and D is non-differentiable when $\mathbf{H} \in \mathcal{P}$. This shows that the system either converges to $\mathcal{C}$ or $\mathcal{P}$.

Statement (ii): Proposition 4.23 shows that D will not have any local minima outside $\mathcal{C}$. Thus, for each $\mathbf{H} \in \mathcal{P}$, there exists an arbitrarily small perturbation twist $\boldsymbol{\xi}_P(\mathbf{H})$ [1] such that the perturbed state $\mathbf{H}'$ satisfies $D(\mathbf{H}') < D(\mathbf{H})$. Furthermore, there is a non-zero minimum decrease δ that can be obtained at all steps. Let t_k be the time at which $\mathbf{H}(t)$ enters $\mathcal{P}$ for the k^{th} time, under the application of the controller and the corresponding perturbation policy. Then, we have $D(\mathbf{H}(t_{k+1})) < D(\mathbf{H}(t_k))$, with a decrease of at least δ at each step. Let $D_{\min,\mathcal{P}} \triangleq \min_{\mathbf{H} \in \mathcal{P}} D(\mathbf{H})$, which is positive since $\mathcal{C} \cap \mathcal{P} = \emptyset$ and $D_{\min,\mathcal{C}}$ in Definition 4.7 is strictly positive. The decreasing sequence $D(t_k)$ must eventually fall below $D_{\min,\mathcal{P}}$ for some finite k. From that point onward, since $\dot{D} \leq 0$, $\mathcal{P}$ will not be re-entered.

Statement (iii): Circulation comes from the fact that, once in $\mathcal{C}$, the term $k_N \boldsymbol{\xi}_N$ vanishes (Lemma 4.27), while $k_T \boldsymbol{\xi}_T$ – the necessary twist to track the curve in a given sense (clockwise or counter-clockwise) – remains non-zero. This non-zero value, due to k_T being positive and that $\mathbf{H}_d$ being a proper parametrization (see Definition 4.10), enforces the circulation of the curve. □

Note that the sense of circulation (clockwise or counterclockwise) is determined by the choice of parametrization $\mathbf{H}_d(s)$. For instance, using the reparametrization $\mathbf{H}_{d,\text{new}}(s) = \mathbf{H}_d(1-s)$ results in circulation in the opposite direction.

Result (ii) in Theorem 4.28 implies that if the system enters the "problematic set" $\mathcal{P}$, there always exists an arbitrarily small sequence of maneuvers that enables the system to eventually escape this set in finite time and never return. Furthermore, as a corollary of Theorem 4.28, we can interpret that the closed loop system is asymptotically stable to the desired curve if $D(\mathbf{H}(0)) < D_{\min,\mathcal{P}}$. In other words, starting sufficiently close to the target curve ensures that $\mathcal{P}$ is never reached.

4.8 Explicit construction for exponential Lie groups

While our results are valid for any connected matrix Lie group, we now provide explicit computations for exponential Lie groups (see Section 3.2.3), leveraging the surjectivity of

[1] To be more precise, Proposition 4.23 shows that one such twist is one that moves $\mathbf{H}$ in the direction of any of the possible $\mathbf{H}_d(s^*)$ through the path induced by Φ.

the exponential map. In order to define the path and the EE-distance function, we first introduce some properties of the logarithm.

For exponential Lie groups, while the exponential map is surjective, it is not necessarily bijective. However, we can define an inverse function in the following manner. Let $\mathbb{R}_+^{n\times n}$ be the set of real matrices with no negative real eigenvalues. Additionally, let $\mathcal{L}^n$ represent the set of $n \times n$ real matrices whose eigenvalues λ lie within the strip $\{\lambda : -\pi < \mathrm{Im}(\lambda) < \pi\}$. The function $\exp : \mathcal{L}^n \to \mathbb{R}_+^{n\times n}$ is bijective, ensuring the existence of an inverse function $\mathrm{Log} : \mathbb{R}_+^{n\times n} \to \mathcal{L}^n$, referred to as the *principal logarithm* (Gallier and Quaintance, 2020, p. 319). Furthermore, let $\mathcal{L}_{\mathfrak{g}}^n \subseteq \mathfrak{g}$ be the subset of $n \times n$ real matrices – elements of the Lie algebra– whose eigenvalues λ lie in the strip $\{\lambda : -\pi \leq \mathrm{Im}(\lambda) \leq \pi\}$. We define the logarithm function $\log : G \to \mathcal{L}_{\mathfrak{g}}^n$ such that $\log(\mathbf{Z})$ coincides with $\mathrm{Log}(\mathbf{Z})$ when $\mathbf{Z} \in \mathbb{R}_+^{n\times n}$, and otherwise corresponds to any matrix $\mathbf{Y} \in \mathcal{L}_{\mathfrak{g}}^n$ that satisfies $\exp(\mathbf{Y}) = \mathbf{Z}$. This is feasible because we assume the group is exponential. Moreover, this choice should be predefined and deterministic.

With this, we can state the following important lemma.

Lemma 4.29. *For all $\mathbf{Z} \in G$ and for all $r \in [0, 1]$, the following property holds:*

$$\log\Big(\exp\big(r \log(\mathbf{Z})\big)\Big) = r \log(\mathbf{Z})$$

Proof. The proof will be divided in three cases.

Case 1: When $\mathbf{Z} \in \mathbb{R}_+^{n\times n}$ and $r \in [0, 1]$, $\log(\mathbf{Z}) = \mathrm{Log}(\mathbf{Z})$ and therefore $\log(\mathbf{Z})$ will lie on $\mathcal{L}^n$. Since $0 \leq r \leq 1$, then[2] it holds that $r \log(\mathbf{Z})$ will also lie on $\mathcal{L}^n$. Moreover, since in this case, the logarithm is equal to the principal logarithm, which is invertible within its domain, it holds by definition that $\log\Big(\exp\big(r \log(\mathbf{Z})\big)\Big) = r \log(\mathbf{Z})$.

Case 2: Now, let $\mathbf{Z} \notin \mathbb{R}_+^{n\times n}$ and $r \in [0, 1)$, then $\log(\mathbf{Z}) \in \mathcal{L}_{\mathfrak{g}}^n$. Thus $r \log(\mathbf{Z}) \in \mathcal{L}^n$, where the logarithm is bijective. Therefore, the expression also holds.

Case 3: Now, let $\mathbf{Z} \notin \mathbb{R}_+^{n\times n}$ and $r = 1$. In this case, the expression reduces to $\log(\exp(\log(\mathbf{Z}))) = \log(\mathbf{Z})$. By the definition of log, it follows that $\exp(\log(\mathbf{Z})) = \mathbf{Z}$, which ensures that the equality holds since the log function is predefined and deterministic. □

For the exponential Lie groups, a path $\Phi_\sigma \triangleq \Phi(\sigma, \mathbf{V}, \mathbf{W})$ can be defined as:

$$\Phi(\sigma, \mathbf{V}, \mathbf{W}) = \mathbf{V} \exp\Big(\log\Big(\mathbf{V}^{-1}\mathbf{W}\Big)\sigma\Big), \tag{4.26}$$

which is in accordance with Definition 4.20. Then, an EE-distance function is defined as:

$$\widehat{D}(\mathbf{V}, \mathbf{W}) = \| \log\left(\mathbf{V}^{-1}\mathbf{W}\right)\|_F. \tag{4.27}$$

[2]Note that if λ is an eigenvalue of $\mathbf{X}$, then $r\lambda$ is an eigenvalue of $r\mathbf{X}$ for any scalar r.

Remark 4.30. The path (4.26) and EE-distance (4.27) reduce to the ones in Example 4.22 and Example 4.8, respectively, when applied to the particular case $G = \mathrm{T}(m)$.

Let $\mathcal{T}(\mathbf{V}) = \mathbf{v} \in \mathbb{R}^m$, $\mathcal{T}(\mathbf{W}) = \mathbf{w} \in \mathbb{R}^m$. Using the series expansion of log and exp, we find that

$$\mathbf{V} \exp\Big(\log\left(\mathbf{V}^{-1}\mathbf{W}\right)\sigma\Big) = \begin{bmatrix} \mathbf{I} & (1-\sigma)\mathbf{v} + \sigma\mathbf{w} \\ \mathbf{0} & 1 \end{bmatrix}. \tag{4.28}$$

Note that $\mathcal{T}\Big((1-\sigma)\mathbf{V} + \sigma\mathbf{W}\Big) = (1-\sigma)\mathcal{T}(\mathbf{V}) + \sigma\mathcal{T}(\mathbf{W})$, the path in Example 4.22.

Using the series expansion of log again, $\|\log\left(\mathbf{V}^{-1}\mathbf{W}\right)\|_F = \|\mathbf{V}^{-1}\mathbf{W} - \mathbf{I}\|_F$. Note that $\mathbf{V}^{-1}\mathbf{W} - \mathbf{I}$ is a matrix whose only non-zero column is the last one, equal to $[\,(\mathbf{w}-\mathbf{v})^\top \quad 0\,]^\top$, this implies that $\|\mathbf{V}^{-1}\mathbf{W} - \mathbf{I}\|_F = \|\mathbf{w} - \mathbf{v}\|$, which is clearly equal to the EE-distance in Example 4.8. □

In order to invoke Theorem 4.28, function $\widehat{D}$ in (4.27) needs to be an EE-distance (see Definition 4.7) that is left-invariant (see Definition 4.17), chainable (see Definition 4.21) and locally linear (see Definition 4.24). Thus, we prove all of these properties in the following proposition.

Proposition 4.31. *Adopting the path Φ in (4.26), the function $\widehat{D}$ in (4.27) is a left-invariant, chainable, and locally linear EE-distance.*

Proof. We prove each property separately.

EE-distance: Positive definiteness and differentiability are immediate upon inspection, and thus $\widehat{D}$ is an EE-distance.

Left-invariant: The distance function is left-invariant since, for all $\mathbf{X} \in G$, we have $\widehat{D}(\mathbf{XV}, \mathbf{XW}) = \|\log\left(\mathbf{V}^{-1}\mathbf{X}^{-1}\mathbf{XW}\right)\|_F$, which is clearly equal to $\widehat{D}(\mathbf{V}, \mathbf{W})$.

Chainable: To prove the chainability property, we first substitute Φ by its expression (4.26) in (4.27), which results in

$$\widehat{D}(\mathbf{V}, \Phi_\sigma) = \left\|\log\Big(\mathbf{V}^{-1}\mathbf{V}\exp\Big(\log(\mathbf{V}^{-1}\mathbf{W})\sigma\Big)\Big)\right\|_F = \sigma\left\|\log\left(\mathbf{V}^{-1}\mathbf{W}\right)\right\|_F, \tag{4.29}$$

using Lemma 4.29 with $\mathbf{Z} = \mathbf{V}^{-1}\mathbf{W}$ and $r = \sigma$, and the fact that $\sigma \geq 0$. Now, using the fact that, by definition, $\mathbf{V}^{-1}\mathbf{W} = \exp(\log(\mathbf{V}^{-1}\mathbf{W}))$, we can express the following:

$$\widehat{D}(\Phi_\sigma, \mathbf{W}) = \left\|\log\Big(\exp\Big(-\log(\mathbf{V}^{-1}\mathbf{W})\sigma\Big)\exp\Big(\log(\mathbf{V}^{-1}\mathbf{W})\Big)\Big)\right\|_F \tag{4.30}$$

Note that $\log(\mathbf{V}^{-1}\mathbf{W})$ commutes with $-\sigma\log(\mathbf{V}^{-1}\mathbf{W})$, and thus we can express the product of exponentials as the exponential of the sum of the arguments:

$$\widehat{D}(\Phi_\sigma, \mathbf{W}) = \left\|\log\Big(\exp\Big((1-\sigma)\log(\mathbf{V}^{-1}\mathbf{W})\Big)\Big)\right\|_F. \tag{4.31}$$

Invoking Lemma 4.29 with $\mathbf{Z} = \mathbf{V}^{-1}\mathbf{W}$ and $r = 1 - \sigma$, and using the fact that $0 \leq \sigma \leq 1$, the previous expression reduces to

$$\widehat{D}(\Phi_\sigma, \mathbf{W}) = (1-\sigma)\| \log\left(\mathbf{V}^{-1}\mathbf{W}\right)\|_F. \tag{4.32}$$

Clearly, $\widehat{D}(\mathbf{V}, \Phi_\sigma) + \widehat{D}(\Phi_\sigma, \mathbf{W}) = \| \log(\mathbf{V}^{-1}\mathbf{W})\|_F = \widehat{D}(\mathbf{V}, \mathbf{W})$.

Locally linear: to prove that $\widehat{D}$ is locally linear, first note that, using Lemma 4.29 and the fact that σ is non-negative, $\widehat{D}(\mathbf{V}, \Phi_\sigma) = \sigma\| \log\left(\mathbf{V}^{-1}\mathbf{W}\right)\|_F$, thus we have

$$\lim_{\sigma \to 0^+} \frac{1}{\sigma}\widehat{D}(\mathbf{V}, \Phi_\sigma) = \lim_{\sigma \to 0^+} \frac{\sigma}{\sigma}\| \log\left(\mathbf{V}^{-1}\mathbf{W}\right)\|_F = \| \log\left(\mathbf{V}^{-1}\mathbf{W}\right)\|_F > 0 \tag{4.33}$$

as long as $\mathbf{V} \neq \mathbf{W}$. □

4.8.1 The particular case of SE(3)

As mentioned, the group SE(3) is exponential, allowing us to use the construction from Section 4.8. However, instead of computing $\widehat{D}(\mathbf{V}, \mathbf{W}) = \| \log(\mathbf{V}^{-1}\mathbf{W})\|_F$ through a generic algorithm to compute the matrix logarithm followed by applying the Frobenius norm, the structure of the group SE(3) allows a more efficient and simpler approach. The algorithm for computing $\widehat{D}(\mathbf{V}, \mathbf{W})$ is as follows:

Let $\mathbf{R}_v$, $\mathbf{R}_w$, $\mathbf{p}_v$, and $\mathbf{p}_w$ denote the rotation matrices and positions of $\mathbf{V}$ and $\mathbf{W}$, respectively. Let $\mathbf{V}^{-1}\mathbf{W} = \mathbf{Z}$, and express it as

$$\mathbf{V}^{-1}\mathbf{W} = \mathbf{Z} = \begin{bmatrix} \mathbf{R}_v^\top \mathbf{R}_w & \mathbf{R}_v^\top(\mathbf{p}_w - \mathbf{p}_v) \\ \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{Q} & \mathbf{u} \\ \mathbf{0} & 1 \end{bmatrix}.$$

From Gallier and Quaintance (2020, p. 41), the logarithm of a homogeneous transformation matrix $\mathbf{Z}$ can be expressed as

$$\log \mathbf{Z} = \begin{bmatrix} \log \mathbf{Q} & \mathbf{X}\mathbf{u} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{4.34}$$

where $\mathbf{X} = (\mathbf{I} - \mathbf{Q})^{-1}\log(\mathbf{Q})$ with some abuse of notation. With this, we can express the EE-distance as

$$\begin{aligned} \widehat{D}(\mathbf{V}, \mathbf{W}) = \| \log \mathbf{Z}\|_F &= \operatorname{tr}\left(\begin{bmatrix} (\log \mathbf{Q})^\top \log \mathbf{Q} & (\log \mathbf{Q})^\top \mathbf{X}\mathbf{u} \\ \mathbf{u}^\top \mathbf{X}^\top \log \mathbf{Q} & \mathbf{u}^\top \mathbf{X}^\top \mathbf{X}\mathbf{u} \end{bmatrix}\right)^{\frac{1}{2}} \\ &= \sqrt{\| \log \mathbf{Q}\|_F^2 + \mathbf{u}^\top \mathbf{X}^\top \mathbf{X}\mathbf{u}} = \sqrt{\| \log \mathbf{Q}\|_F^2 + \|\bar{\mathbf{u}}\|^2}, \end{aligned} \tag{4.35}$$

where $\|\bar{\mathbf{u}}\|^2 = \mathbf{u}^\top \bar{\mathbf{X}}\mathbf{u}$ and $\bar{\mathbf{X}} = \mathbf{X}^\top \mathbf{X}$.

The term $\| \log \mathbf{Q}\|_F^2$ can be obtained by the following steps, using the properties derived

in Appendix B.2:

1. Compute $u \triangleq \frac{1}{2}\big(\mathrm{tr}(\mathbf{Q})-1\big)$ and $v \triangleq \frac{1}{2\sqrt{2}}\|\mathbf{Q}-\mathbf{Q}^\top\|_F$, where $u=\cos(\theta)$ and $v=\sin(\theta)$, in which $\theta \in [0,\pi]$ is the rotation angle related to $\mathbf{Q}$;
2. Compute $\theta = \mathrm{atan2}(v,u)$;
3. Then, $\|\log \mathbf{Q}\|_F^2 = 2\theta^2$.

It is also possible to derive a simple expression for $\|\bar{\mathbf{u}}\|^2$. First, we express $\bar{\mathbf{X}}$ as a function $\Phi(\mathbf{Q})$ using the Cayley-Hamilton theorem (Chen, 2009, p. 63). The derivation of this function is presented in Appendix B.3 and the resulting expression is:

$$\Phi(\mathbf{Q}) = \bar{\mathbf{X}} = (1-2\beta_0)\mathbf{I} + \beta_0\big(\mathbf{Q}+\mathbf{Q}^\top\big), \tag{4.36}$$

where $\beta_0 = \frac{2-2\cos\theta-\theta^2}{4(1-\cos\theta)^2}$. Thus $\|\bar{\mathbf{u}}\|^2$ can be easily obtained from the already computed angle θ and the translation vector $\mathbf{u}$. The EE-distance function reduces to

$$\widehat{D}(\mathbf{V},\mathbf{W}) = \sqrt{2\theta^2 + \mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}}, \tag{4.37}$$

where $\bar{\mathbf{X}}$ is explicitly computed by (4.36). More succinctly, the algorithm for computing $\widehat{D}(\mathbf{V},\mathbf{W})$ is shown in Algorithm 4.1.

Algorithm 4.1 Computation of $\widehat{D}(\mathbf{V},\mathbf{W})$ in SE(3)

Input: Matrices $\mathbf{V},\mathbf{W}$
Output: Distance $\widehat{D}$

Compute $\mathbf{Z} \leftarrow \mathbf{V}^{-1}\mathbf{W}$
Extract $\mathbf{Q} \leftarrow$ rotation part of $\mathbf{Z}$ and $\mathbf{u} \leftarrow$ translation part of $\mathbf{Z}$
$u \leftarrow \frac{1}{2}(\mathrm{tr}(\mathbf{Q})-1)$
$v \leftarrow \frac{1}{2\sqrt{2}}\|\mathbf{Q}-\mathbf{Q}^\top\|_F$
$\theta \leftarrow \mathrm{atan2}(v,u)$
$\beta_0 \leftarrow \frac{2-2u-\theta^2}{4(1-u)^2}$
$\bar{\mathbf{X}} \leftarrow \mathbf{I}(1-2\beta_0) + (\mathbf{Q}+\mathbf{Q}^\top)\beta_0$
$\widehat{D} \leftarrow \sqrt{2\theta^2 + \mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}}$

It can be shown that the result of $\|\log(\mathbf{V}^{-1}\mathbf{W})\|_F$ is independent of the choice of $\log(\mathbf{V}^{-1}\mathbf{W})$ in the edge cases where $\mathbf{V}^{-1}\mathbf{W}$ has negative eigenvalues (see the discussion in Section 4.8). This is evident in the fact that Algorithm 4.1 does not include any components that require a choice to be made.

Note that β_0 is well-defined for all $\theta \in (0,\pi]$. When $\theta = 0$, we just need to take the limit to obtain $\beta_0 = -1/12$ (see Appendix B.3.1). To identify the points of non-differentiability of $\widehat{D}$, it suffices to analyze the derivatives with the respect to the variables $\mathbf{Q}$ and $\mathbf{u}$. The analysis reveals that the only sources of non-differentiability occur when (type i) $\mathbf{Q} = \mathbf{Q}^\top$,

$\mathbf{Q} \neq \mathbf{I}$ (i.e., at rotations of π radians) or when (type ii) $\widehat{D} = 0$. However, in both cases, the directional derivatives exist. Furthermore, $D_{\min,\mathcal{C}}$, as defined in Definition 4.7, can be taken as $\widehat{D}$ when $\theta = \pi$ and $\mathbf{u} = \mathbf{0}$, which gives $D_{\min,\mathcal{C}} = \sqrt{2}\pi$. Thus, when $\widehat{D} < D_{\min,\mathcal{C}}$, it necessarily follows that $\theta < \pi$, avoiding the non-differentiable points of type i. Additionally, when $\widehat{D} > 0$, the non-differentiable points of type ii are also avoided. Therefore, the condition $0 < \widehat{D} < \sqrt{2}\pi$ guarantees that $\widehat{D}$ is differentiable, as required in Definition 4.7.

Components computation

Although it may not be trivial, the components of the vector field can be computed explicitly. First, observe that the tangent component $\boldsymbol{\xi}_T$ depends on the nearest point on the curve and the derivative of the curve at this point. Since the curve is parametrized, we already have the equation that describes it, implying that the derivative is also known. Therefore, the tangent component $\boldsymbol{\xi}_T = \mathcal{S}^{-1}\left(\frac{d\mathbf{H}_d(s^*)}{ds}\mathbf{H}_d(s^*)^{-1}\right)$ can be computed explicitly.

To compute the normal component $\boldsymbol{\xi}_N$, we need to evaluate $\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{H}, \mathbf{H}_d(s^*))$. Two approaches can be used for this computation. The first is a numerical approach, where we evaluate the left-hand side of (4.2) for $\boldsymbol{\zeta} = \mathbf{e}_i$ with a small ε. The second approach involves computing the explicit L operator of $\widehat{D}$ with respect to $\mathbf{H}$, which we will now derive.

First, note that the EE-distance is symmetric, i.e., $\left\|\log(\mathbf{V}^{-1}\mathbf{W})\right\|_F = \left\|\log(\mathbf{W}^{-1}\mathbf{V})\right\|_F$. Let $\mathbf{Z} = \mathbf{H}_d(s^*)^{-1}\mathbf{H}$, then using expression (4.37), we define the following equivalent function:

$$\widehat{E}(\mathbf{Z}) \triangleq \widehat{D}(\mathbf{H}, \mathbf{H}_d(s^*)) = \sqrt{2\theta^2 + \mathbf{u}^\top \bar{\mathbf{X}}\mathbf{u}}, \tag{4.38}$$

for which the L operator can be computed as

$$\mathrm{L}[\widehat{E}](\mathbf{Z}) = \frac{1}{2\sqrt{2\theta^2 + \mathbf{u}^\top \bar{\mathbf{X}}\mathbf{u}}}\mathrm{L}\left[2\theta^2 + \mathbf{u}^\top \bar{\mathbf{X}}\mathbf{u}\right](\mathbf{Z}), \tag{4.39}$$

which in turn can be expressed by

$$\mathrm{L}[\widehat{E}](\mathbf{Z}) = \frac{4\theta\mathrm{L}[\theta] + \mathrm{L}\left[\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbf{u}_i\mathbf{u}_j\bar{\mathbf{X}}_{ij}\right](\mathbf{Z})}{2\widehat{E}(\mathbf{Z})}. \tag{4.40}$$

The computation of the two L operators in (4.40) is rather lengthy and is provided in Appendix B.4. However, note that these computations only rely on previously computed terms and the elements of $\mathbf{u}$ and $\mathbf{Q}$. For completeness, we present the expressions for these terms without the detailed derivation:

$$\mathrm{L}[\theta](\mathbf{Z}) = \frac{-\cos\theta}{8\sin\theta}\mathbf{f} - \frac{\sin\theta}{2}\mathbf{g}, \tag{4.41}$$

where

$$\mathbf{f} = 2\begin{bmatrix}\mathbf{0} & \{\mathbf{Q}^2\}_{23} - \{\mathbf{Q}^2\}_{32} & \{\mathbf{Q}^2\}_{31} - \{\mathbf{Q}^2\}_{13} & \{\mathbf{Q}^2\}_{12} - \{\mathbf{Q}^2\}_{21}\end{bmatrix}, \tag{4.42}$$

$$\mathbf{g} = \begin{bmatrix}\mathbf{0} & \mathbf{Q}_{23} - \mathbf{Q}_{32} & \mathbf{Q}_{31} - \mathbf{Q}_{13} & \mathbf{Q}_{12} - \mathbf{Q}_{21}\end{bmatrix}. \tag{4.43}$$

The remaining L operator is expressed as

$$\mathrm{L}\Big[\sum_{i=1}^{3}\sum_{j=1}^{3}\mathbf{u}_i\mathbf{u}_j\bar{\mathbf{X}}_{ij}\Big](\mathbf{Z}) = \sum_{i=1}^{3}\sum_{j=1}^{3}\Big(2\,\mathrm{L}[\mathbf{u}_i](\mathbf{Z})\mathbf{u}_j\bar{\mathbf{X}}_{ij} + \mathbf{u}_i\mathbf{u}_j\,\mathrm{L}\Big[\bar{\mathbf{X}}_{ij}\Big](\mathbf{Z})\Big), \tag{4.44}$$

where

$$\mathrm{L}[\mathbf{u}_i](\mathbf{Z}) = \begin{bmatrix}\delta_{1i} & \delta_{2i} & \delta_{3i} & \{\widehat{\mathcal{S}}(\widehat{\mathbf{e}}_1)\mathbf{u}\}_i & \{\widehat{\mathcal{S}}(\widehat{\mathbf{e}}_2)\mathbf{u}\}_i & \{\widehat{\mathcal{S}}(\widehat{\mathbf{e}}_3)\mathbf{u}\}_i\end{bmatrix}, \tag{4.45}$$

$$\mathrm{L}[\bar{\mathbf{X}}_{ij}](\mathbf{Z}) = \mathrm{L}[\beta_0](\mathbf{Z})\Big(-2\delta_{ij} + \mathbf{Q}_{ij} + \mathbf{Q}_{ji}\Big) + \beta_0\Big(\mathrm{L}[\mathbf{Q}_{ij}](\mathbf{Z}) + \mathrm{L}[\mathbf{Q}_{ji}](\mathbf{Z})\Big) \tag{4.46}$$

in which δ_{ij} is the Kronecker delta, $\widehat{\mathcal{S}} : \mathbb{R}^3 \to \mathfrak{so}(3)$ is a skew-symmetric matrix, $\widehat{\mathbf{e}}_i$ is the i^{th} canonical basis vector in $\mathbb{R}^3$, and

$$\mathrm{L}[\beta_0](\mathbf{Z}) = \frac{\theta^2\sin(\theta) - \theta - \sin(\theta) + \Big(\theta + \sin(\theta)\Big)\cos(\theta)}{2(1-\cos(\theta))^3}\,\mathrm{L}[\theta](\mathbf{Z}), \tag{4.47}$$

$$\mathrm{L}[\mathbf{Q}_{ij}](\mathbf{Z}) = \begin{bmatrix}\mathbf{0} & \Big\{\widehat{\mathcal{S}}(\widehat{\mathbf{e}}_1)\mathbf{Q}\Big\}_{ij} & \Big\{\widehat{\mathcal{S}}(\widehat{\mathbf{e}}_2)\mathbf{Q}\Big\}_{ij} & \Big\{\widehat{\mathcal{S}}(\widehat{\mathbf{e}}_3)\mathbf{Q}\Big\}_{ij}\end{bmatrix}. \tag{4.48}$$

To compute $\boldsymbol{\xi}_N$, we need to evaluate $\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{H}, \mathbf{H}_d(s^*))$. However, by applying the chain rule, we can compute it via $\mathrm{L}[\widehat{E}](\mathbf{Z})$ as follows (see Appendix A.3.1):

$$\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{H}, \mathbf{H}_d(s^*)) = \mathrm{L}[\widehat{E}](\mathbf{Z})\mathcal{Z}\Big(\mathbf{H}_d(s^*)\Big), \tag{4.49}$$

where $\mathcal{Z}\Big(\mathbf{H}_d(s^*)\Big)$ is computed as

$$\mathcal{Z}\Big(\mathbf{H}_d(s^*)\Big) = \begin{bmatrix}\mathbf{R}_d^\top(s^*) & -\mathbf{R}_d^\top(s^*)\widehat{\mathcal{S}}\Big(\mathbf{p}_d(s^*)\Big) \\ \mathbf{0} & \mathbf{R}_d^\top(s^*)\end{bmatrix}, \tag{4.50}$$

in which $\mathbf{R}_d(s^*)$ and $\mathbf{p}_d(s^*)$ are the rotation matrix and position of $\mathbf{H}_d(s^*)$, respectively. The computation of $\mathcal{Z}\Big(\mathbf{H}_d(s^*)\Big)$ is straightforward, and thus $\boldsymbol{\xi}_N$ can be computed explicitly. Algorithmically, the computation of $\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{Z})$ is shown in Algorithm 4.2.

Algorithm 4.2 Computation of $\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{Z})$ in SE(3)

Input: $\mathbf{H}_d(s^*)$, $\mathbf{H}$

Output: $\mathrm{L}_{\mathbf{V}}[\widehat{D}](\mathbf{Z})$

$\mathbf{Z} \leftarrow \mathbf{H}_d(s^*)^{-1}\mathbf{H}$

Extract rotation matrix $\mathbf{Q}$ and translation $\mathbf{u}$ from $\mathbf{Z}$

$u \leftarrow \frac{1}{2}(\mathrm{tr}(\mathbf{Q}) - 1)$

$v \leftarrow \frac{1}{2\sqrt{2}}\|\mathbf{Q} - \mathbf{Q}^\top\|_F$

$\theta \leftarrow \mathrm{atan2}(v, u)$

$\beta_0 \leftarrow \frac{2-2u-\theta^2}{4(1-u)^2}$

$\bar{\mathbf{X}} \leftarrow \mathbf{I}(1 - 2\beta_0) + (\mathbf{Q} + \mathbf{Q}^\top)\beta_0$

$\widehat{E}(\mathbf{Z}) \leftarrow \sqrt{2\theta^2 + \mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}}$

$\mathbf{f} \leftarrow 2\begin{bmatrix}\mathbf{0} & \{\mathbf{Q}^2\}_{23} - \{\mathbf{Q}^2\}_{32} & \{\mathbf{Q}^2\}_{31} - \{\mathbf{Q}^2\}_{13} & \{\mathbf{Q}^2\}_{12} - \{\mathbf{Q}^2\}_{21}\end{bmatrix}$

$\mathbf{g} \leftarrow \begin{bmatrix}\mathbf{0} & \mathbf{Q}_{23} - \mathbf{Q}_{32} & \mathbf{Q}_{31} - \mathbf{Q}_{13} & \mathbf{Q}_{12} - \mathbf{Q}_{21}\end{bmatrix}$

$\mathrm{L}[\theta] \leftarrow \frac{-\cos\theta}{8\sin\theta}\mathbf{f} - \frac{\sin\theta}{2}\mathbf{g}$

$\mathrm{L}[\beta_0] \leftarrow \frac{\theta^2\sin(\theta) - \theta - \sin(\theta) + \Big(\theta + \sin(\theta)\Big)\cos(\theta)}{2(1-\cos(\theta))^3}\mathrm{L}[\theta]$

Initialize $\mathrm{L}[\mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}] \leftarrow 0$

for $i \leftarrow 1$ to 3 **do**

  $\mathrm{L}[\mathbf{u}_i] \leftarrow \begin{bmatrix}\delta_{1i} & \delta_{2i} & \delta_{3i} & \{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_1)\mathbf{u}\}_i & \{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_2)\mathbf{u}\}_i & \{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_3)\mathbf{u}\}_i\end{bmatrix}$

  for $j \leftarrow 1$ to 3 **do**

    $\mathrm{L}[\mathbf{Q}_{ij}] \leftarrow \begin{bmatrix}\mathbf{0} & \Big\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_1)\mathbf{Q}\Big\}_{ij} & \Big\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_2)\mathbf{Q}\Big\}_{ij} & \Big\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_3)\mathbf{Q}\Big\}_{ij}\end{bmatrix}$

    $\mathrm{L}[\mathbf{Q}_{ji}] \leftarrow \begin{bmatrix}\mathbf{0} & \Big\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_1)\mathbf{Q}\Big\}_{ji} & \Big\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_2)\mathbf{Q}\Big\}_{ji} & \Big\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_3)\mathbf{Q}\Big\}_{ji}\end{bmatrix}$

    $\mathrm{L}[\bar{\mathbf{X}}_{ij}] \leftarrow \mathrm{L}[\beta_0]\Big(-2\delta_{ij} + \mathbf{Q}_{ij} + \mathbf{Q}_{ji}\Big) + \beta_0\Big(\mathrm{L}[\mathbf{Q}_{ij}] + \mathrm{L}[\mathbf{Q}_{ji}]\Big)$

    $\mathrm{L}[\mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}] \leftarrow \mathrm{L}[\mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}] + 2\,\mathrm{L}[\mathbf{u}_i]\mathbf{u}_j\bar{\mathbf{X}}_{ij} + \mathbf{u}_i\mathbf{u}_j\,\mathrm{L}[\bar{\mathbf{X}}_{ij}]$

  end for

end for

$\mathrm{L}[\widehat{E}] \leftarrow \frac{4\theta\,\mathrm{L}[\theta] + \mathrm{L}[\mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}]}{2\widehat{E}(\mathbf{Z})}$

$\mathcal{Z}\Big(\mathbf{H}_d(s^*)\Big) \leftarrow$(4.50)

$\mathrm{L}_{\mathbf{V}}[\widehat{D}] \leftarrow \mathrm{L}[\widehat{E}]\mathcal{Z}\Big(\mathbf{H}_d(s^*)\Big)$

5

Collaborative Manipulation

Our objective in this chapter is developing an adaptive control for a collaborative task of manipulating a large object with unknown parameters. As the task involves convergence and circulation of the object along a predefined curve, we use the control strategy developed in Chapter 4 as a reference for velocity tracking. Our final control system can be thought as a two-stage controller: the first one is kinematic and outputs the necessary velocity to guarantee curve tracking; the second controller is a dynamic one, responsible for ensuring that the object's velocity is equal to the kinematic controller output. We begin by presenting the system modelling, then we move to the vector field guidance employed in the kinematic controller, and finally, we present the adaptive control strategy.

5.1 Modelling and problem statement

The problem addressed in this section revolves around designing controller to guide a manipulated object along a predefined curve $\mathcal{C} \subset \mathbb{R}^3 \times \mathrm{SO}(3)$. The object's dynamics include uncertain parameters like mass and geometric properties. With N agents involved in the manipulation process, a decentralized control law is devised for each agent. The agents are unaware of their precise positioning relative to a measurement point. It is assumed that each agent possesses the capability to measure the pose and velocity of the object's measurement point.

We consider the cooperative manipulation of a rigid body by a team of N autonomous agents, as illustrated in Figure 5.1. The body undergoes both translations and rotations.

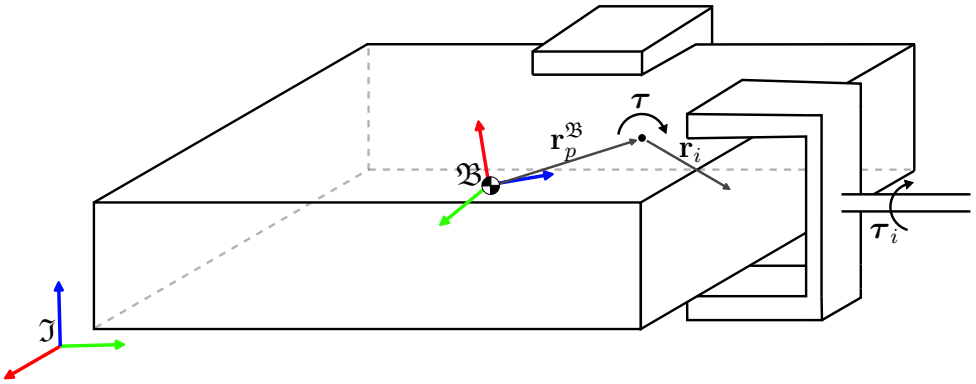


Figure 5.1: Collaborative task

We establish two reference frames: the world-fixed inertial frame denoted $\mathfrak{I}$, and the body-fixed frame denoted $\mathfrak{B}$, centered at the body's center of mass, $\mathbf{b} \triangleq \mathbf{b}^{\mathfrak{I}}$. Additionally, a body-fixed measurement point, $\mathbf{p}^{\mathfrak{B}}$, is situated at a distance $\mathbf{r}_p^{\mathfrak{B}}$ from the body's center of mass.

The body possesses a mass m and a constant inertia tensor $\mathbb{I}_{\text{cm}}^{\mathfrak{B}}$ about its center of mass. We assume each agent is rigidly attached to the body at $\mathbf{r}_i$ from $\mathbf{p}^{\mathfrak{B}}$. While $\mathbf{r}_i$ ideally should be expressed relative to a frame at $\mathbf{p}^{\mathfrak{B}}$, for simplicity, we interchangeably consider $\mathbf{r}_i = \mathbf{r}_i^{\mathfrak{B}}$. This assumption is made under the understanding that this information solely pertains to torque computations, and the frame associated with the measurement point shares the same orientation as that of the center of mass. Each agent exerts a wrench $\boldsymbol{\tau}_i$ onto the body.

Let $\mathbf{p} \triangleq \mathbf{p}^{\mathfrak{I}}(t)$ be the position of the measurement point in $\mathfrak{I}$ and let $\mathbf{R} \triangleq \mathbf{R}_{\mathfrak{B}}^{\mathfrak{I}}(t)$ be the rotation matrix from frame $\mathfrak{B}$ to frame $\mathfrak{I}$. Let the linear velocity of the point $\mathbf{p}$ expressed in frame $\mathfrak{I}$ be represented by $\dot{\mathbf{p}}$, and let $\boldsymbol{\omega} \triangleq \boldsymbol{\omega}^{\mathfrak{I}}(t)$ denote the angular velocity of frame $\mathfrak{B}$ with respect to $\mathfrak{I}$, expressed in $\mathfrak{I}$. Additionally, let $\ddot{\mathbf{p}}$ and $\dot{\boldsymbol{\omega}}$ represent the linear and angular accelerations of the body, respectively. Then, we can express the linear velocity at $\mathbf{p}$ as

$$\dot{\mathbf{p}} = \frac{d}{dt}\left(\mathbf{b} + \mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) = \dot{\mathbf{b}} + \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}} + \mathbf{R}\dot{\mathbf{r}}_p^{\mathfrak{B}} = \dot{\mathbf{b}} + \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}, \tag{5.1}$$

where the fact that $\mathbf{r}_p^{\mathfrak{B}}$ is constant in frame $\mathfrak{B}$ was used. The linear acceleration is given by

$$\ddot{\mathbf{p}} = \frac{d}{dt}\left(\dot{\mathbf{b}} + \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) = \ddot{\mathbf{b}} + \mathcal{S}(\dot{\boldsymbol{\omega}})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}} + \mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}. \tag{5.2}$$

Thus, the resulting force acting on the body is given by Newton's second law as

$$\mathbf{F}^{\mathfrak{I}} = m\ddot{\mathbf{b}} = m\left(\ddot{\mathbf{p}} - \mathcal{S}(\dot{\boldsymbol{\omega}})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}} - \mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right). \tag{5.3}$$

The total torque at point $\mathbf{p}$ expressed in the inertial frame is given by

$$\mathbf{T}_p^{\mathfrak{I}} = \mathbf{R}\mathbf{T}_p^{\mathfrak{B}} = \mathbf{R}\mathbb{I}_{\text{cm}}^{\mathfrak{B}}\mathbf{R}^\top\dot{\boldsymbol{\omega}} + \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbb{I}_{\text{cm}}^{\mathfrak{B}}\mathbf{R}^\top\boldsymbol{\omega} - \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathbf{F}^{\mathfrak{I}}, \tag{5.4}$$

expanding the last term, we have

$$-\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathbf{F}^{\mathfrak{I}} = -\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\ddot{\mathbf{p}} - \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\boldsymbol{\omega}} + \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}. \tag{5.5}$$

From Steiner's theorem (Lemos, 2018), we know that $\mathbb{I}_p^{\mathfrak{B}} = \mathbb{I}_{\text{cm}}^{\mathfrak{B}} - m\mathcal{S}\Big(\mathbf{r}_p^{\mathfrak{B}}\Big)^2$, where $\mathbb{I}_p^{\mathfrak{B}}$ is the inertia tensor about point $\mathbf{p}$. Thus, we can obtain the following expression for the second term in (5.5)

$$-\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\boldsymbol{\omega}} = -\mathbf{R}\mathcal{S}\Big(\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathbf{R}^\top\mathbf{R}\mathcal{S}\Big(\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathbf{R}^\top\dot{\boldsymbol{\omega}} = \mathbf{R}\Big(\mathbb{I}_p^{\mathfrak{B}} - \mathbb{I}_{\text{cm}}^{\mathfrak{B}}\Big)\mathbf{R}^\top\dot{\boldsymbol{\omega}}. \tag{5.6}$$

Expanding the last term in (5.5), we have

$$\begin{aligned}\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}} &= \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}(\boldsymbol{\omega})\,(\dot{\mathbf{p}} - \dot{\mathbf{b}}) \\ &= \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}(\boldsymbol{\omega})\,\dot{\mathbf{p}} - \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}(\boldsymbol{\omega})\,\dot{\mathbf{b}}.\end{aligned} \tag{5.7}$$

Applying Jacobi's identity, we obtain

$$\begin{aligned}&m\mathcal{S}\Big(\mathcal{S}(\dot{\mathbf{p}})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega} + m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega}\Big)\,\dot{\mathbf{p}} - m\mathcal{S}\Big(\mathcal{S}\Big(\dot{\mathbf{b}}\Big)\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega} - m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega}\Big)\,\dot{\mathbf{b}} \\ &= -m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}\Big)\,\boldsymbol{\omega} - m\mathcal{S}\Big(\dot{\mathbf{p}} - \dot{\mathbf{b}}\Big)\,\dot{\mathbf{p}} - m\mathcal{S}\Big(\mathcal{S}\Big(\dot{\mathbf{b}}\Big)\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega} + m\mathcal{S}\Big(\dot{\mathbf{p}} - \dot{\mathbf{b}}\Big)\,\dot{\mathbf{b}} \\ &= -m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}\Big)\,\boldsymbol{\omega} + m\mathcal{S}\Big(\dot{\mathbf{b}}\Big)\,\dot{\mathbf{p}} - m\mathcal{S}\Big(\mathcal{S}\Big(\dot{\mathbf{b}}\Big)\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega} + m\mathcal{S}(\dot{\mathbf{p}})\,\dot{\mathbf{b}} \\ &= -m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}\Big)\,\boldsymbol{\omega} - m\mathcal{S}\Big(\mathcal{S}\Big(\dot{\mathbf{p}} - \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega} \\ &= -m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}\Big)\,\boldsymbol{\omega} - m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}} - m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\boldsymbol{\omega} \\ &= -m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}\Big)\,\boldsymbol{\omega} - m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}} - m\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\Big(\mathbb{I}_p^{\mathfrak{B}} - \mathbb{I}_{\text{cm}}^{\mathfrak{B}}\Big)\mathbf{R}^\top\boldsymbol{\omega}.\end{aligned} \tag{5.8}$$

Finally, using (5.4), (5.5), (5.6), and (5.8), the resulting torque is expressed as

$$\begin{aligned}\mathbf{T}_p^{\mathfrak{I}} =& \mathbf{R}\mathbb{I}_{\text{cm}}^{\mathfrak{B}}\mathbf{R}^\top\dot{\boldsymbol{\omega}} + \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbb{I}_{\text{cm}}^{\mathfrak{B}}\mathbf{R}^\top\boldsymbol{\omega} - \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\ddot{\mathbf{p}} + \mathbf{R}\Big(\mathbb{I}_p^{\mathfrak{B}} - \mathbb{I}_{\text{cm}}^{\mathfrak{B}}\Big)\mathbf{R}^\top\dot{\boldsymbol{\omega}} \\ &- m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}\Big)\,\boldsymbol{\omega} - m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}} - m\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\Big(\mathbb{I}_p^{\mathfrak{B}} - \mathbb{I}_{\text{cm}}^{\mathfrak{B}}\Big)\mathbf{R}^\top\boldsymbol{\omega}\end{aligned} \tag{5.9}$$

$$\begin{aligned}=& \mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top\dot{\boldsymbol{\omega}} + \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top\boldsymbol{\omega} - \mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\ddot{\mathbf{p}} - m\mathcal{S}\Big(\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}\Big)\,\boldsymbol{\omega} \\ &- m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\Big(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\Big)\,\dot{\mathbf{p}}.\end{aligned} \tag{5.10}$$

Define the pose $\boldsymbol{\chi} = (\mathbf{p}, \mathbf{R}) \in \mathbb{R}^3 \times \text{SO}(3)$, and, abusing notation, let $\mathbb{R}^6 \ni \dot{\boldsymbol{\chi}} = \Big[\dot{\mathbf{p}}^\top, \boldsymbol{\omega}^\top\Big]^\top$ represent the system's linear and angular velocities. Additionally, let $\mathbb{R}^6 \ni \ddot{\boldsymbol{\chi}} = \Big[\ddot{\mathbf{p}}^\top, \dot{\boldsymbol{\omega}}^\top\Big]^\top$ represent the system's linear and angular accelerations. With this notation, the

dynamics can be succinctly expressed as

$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{\chi})\ddot{\boldsymbol{\chi}} + \mathbf{C}(\boldsymbol{\chi}, \dot{\boldsymbol{\chi}})\dot{\boldsymbol{\chi}} + \mathbf{g}, \tag{5.11}$$

where $\boldsymbol{\tau} \triangleq [\mathbf{F}^\top \ \mathbf{T}_p^\top]^\top$ denotes the total wrench applied to the body about the point $\mathbf{p}$ expressed in frame $\mathfrak{I}$, vector $\mathbf{g}$ is the gravity vector, matrix $\mathbf{M}$ is the system's Cartesian space mass matrix, represented by

$$\mathbf{M}(\boldsymbol{\chi}) = \begin{bmatrix} m\mathbf{I} & m\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) \\ -m\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) & \mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top \end{bmatrix}, \tag{5.12}$$

and the matrix $\mathbf{C}(\boldsymbol{\chi}, \dot{\boldsymbol{\chi}})$ is the system's Coriolis matrix expressed by:

$$\mathbf{C}(\boldsymbol{\chi}, \dot{\boldsymbol{\chi}}) = \begin{bmatrix} \mathbf{0} & m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) \\ -m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) & \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top - m\mathcal{S}\left(\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\dot{\mathbf{p}}\right) \end{bmatrix}. \tag{5.13}$$

The mass and Coriolis matrices share the same properties as the ones obtained through Euler-Lagrange approach. For completeness, we provide the same proofs presented in Culbertson et al. (2021):

Lemma 5.1. *Mass matrix* $\mathbf{M}(\boldsymbol{\chi})$ *as in* (5.12) *is symmetric and positive definite.*

Proof. The symmetry property is immediate, since $\mathcal{S}(\mathbf{a})^\top = -\mathcal{S}(\mathbf{a})$. To show that the matrix is positive definite, we use Schur's complement. Thus, $\mathbf{M} > 0 \iff m\mathbf{I} > 0$, which is obvious, and the Schur complement $\mathbf{S} > 0$. Matrix $\mathbf{S}$ is given by

$$\begin{aligned} \mathbf{S} &= \mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top - \left(-m\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\left(\frac{1}{m}\mathbf{I}\right)m\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\right) \\ &= \mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top + m\mathbf{R}\mathcal{S}\left(\mathbf{r}_p^{\mathfrak{B}}\right)^2\mathbf{R}^\top \\ &= \mathbf{R}\left(\mathbb{I}_p^{\mathfrak{B}} + m\mathcal{S}\left(\mathbf{r}_p^{\mathfrak{B}}\right)^2\right)\mathbf{R}^\top \\ &= \mathbf{R}\mathbb{I}_{\mathrm{cm}}^{\mathfrak{B}}\mathbf{R}^\top, \end{aligned} \tag{5.14}$$

from Steiner's theorem. Since $\mathbb{I}_{\mathrm{cm}}^{\mathfrak{B}}$ is positive definite, $\mathbf{S}$ is also positive definite, and thus $\mathbf{M}$ is positive definite. □

Lemma 5.2. *The matrix* $\dot{\mathbf{M}}(\boldsymbol{\chi}) - 2\mathbf{C}(\boldsymbol{\chi}, \dot{\boldsymbol{\chi}})$*, as in* (5.12) *and* (5.13)*, is skew-symmetric.*

Proof. To express the time derivative of the mass matrix, first note that $m\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) = m\mathbf{R}\mathcal{S}\left(\mathbf{r}_p^{\mathfrak{B}}\right)\mathbf{R}^\top$. Now, since $\frac{d}{dt}(\mathbf{R}\mathbf{R}^\top) = \frac{d}{dt}\mathbf{I} = \mathbf{0}$, this implies that $\dot{\mathbf{R}}^\top = -\mathbf{R}^\top\mathcal{S}(\boldsymbol{\omega})$. Thus the time derivative of the second term in $\mathbf{M}$ is given by

$$\frac{d}{dt}\left(m\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\right) = m\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathcal{S}\left(\mathbf{r}_p^{\mathfrak{B}}\right)\mathbf{R}^\top - m\mathbf{R}\mathcal{S}\left(\mathbf{r}_p^{\mathfrak{B}}\right)\mathbf{R}^\top\mathcal{S}(\boldsymbol{\omega}) \tag{5.15}$$

$$= m\mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right) - m\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\mathcal{S}(\boldsymbol{\omega}), \tag{5.16}$$

since $\mathbf{r}_p^{\mathfrak{B}}$ is constant. Note also that if we let $\mathbf{A} = m\,\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\,\mathcal{S}\left(\boldsymbol{r}_p^{\mathfrak{B}}\right)\mathbf{R}^\top$, then we can express the derivative as $\frac{d}{dt}\left(m\,\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\right) = \mathbf{A} - \mathbf{A}^\top$. As for the last term in the mass matrix, we have

$$\frac{d}{dt}\left(\mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top\right) = \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top - \mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top\mathcal{S}(\boldsymbol{\omega})\,. \tag{5.17}$$

Thus, we have the following result

$$\dot{\mathbf{M}} - 2\mathbf{C} = \begin{bmatrix} \mathbf{0} & -\mathbf{A} - \mathbf{A}^\top \\ \mathbf{A} + \mathbf{A}^\top & -\mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top - \mathbf{R}\mathbb{I}_p^{\mathfrak{B}}\mathbf{R}^\top\mathcal{S}(\boldsymbol{\omega}) - m\,\mathcal{S}\left(\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\dot{\mathbf{p}}\right) \end{bmatrix}. \tag{5.18}$$

The last term on this matrix can be written as $\mathbf{X} - \mathbf{X}^\top - m\,\mathcal{S}\left(\mathcal{S}\left(\mathbf{R}\mathbf{r}_p^{\mathfrak{B}}\right)\dot{\mathbf{p}}\right)$. Since $\mathbf{X} - \mathbf{X}^\top$ is skew-symmetric, and $\mathcal{S}(\mathbf{a})$ is skew-symmetric for any vector $\mathbf{a}$, the last term is skew-symmetric. Thus, since the off-diagonal blocks are the negative of each other, the entire matrix is skew-symmetric. □

The dynamics described in (5.11) govern the motion of the body under the influence of the total wrench $\boldsymbol{\tau}$ applied about $\mathbf{p}$. This quantity needs to be expressed as a function of the wrenches $\boldsymbol{\tau}_i$ applied by each agent about their respective displacement $\mathbf{r}_i$ from $\mathbf{p}$. We represent $\boldsymbol{\tau}$ as the sum:

$$\boldsymbol{\tau} = \sum_{i=1}^{N} \mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i)\boldsymbol{\tau}_i, \tag{5.19}$$

where $\mathbf{G}$ denotes the grasp matrix, defined as:

$$\mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathcal{S}(\mathbf{R}\mathbf{r}_i) & \mathbf{I} \end{bmatrix}. \tag{5.20}$$

The grasp matrix exhibits interesting properties. Its inverse is given by a simple relation:

$$\mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i)^{-1} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathcal{S}(\mathbf{R}\mathbf{r}_i) & \mathbf{I} \end{bmatrix} = \mathbf{G}(\boldsymbol{\chi}, -\mathbf{r}_i). \tag{5.21}$$

The multiplication of two grasp matrices is also straightforward:

$$\mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i)\mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_j) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathcal{S}(\mathbf{R}(\mathbf{r}_i + \mathbf{r}_j)) & \mathbf{I} \end{bmatrix} = \mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i) + \mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_j) - \mathbf{I}. \tag{5.22}$$

As previously stated, some system parameters will be unknown. For simplicity we list here the parameters that must be adapted:

- Mass properties m and $\mathbb{I}_{\text{cm}}$;
- geometric properties $\mathbf{r}_p$ and $\mathbf{r}_i$.

5.2 Vector field control

The manipulated object must follow a curve of poses $\mathcal{C} \subset \mathbb{R}^3 \times \mathrm{SO}(3)$, which can be interpreted as a curve in Euclidean space with an orientation frame attached to each of its points, as shown in Figure 5.2. This objective aligns with our vector field formulation.

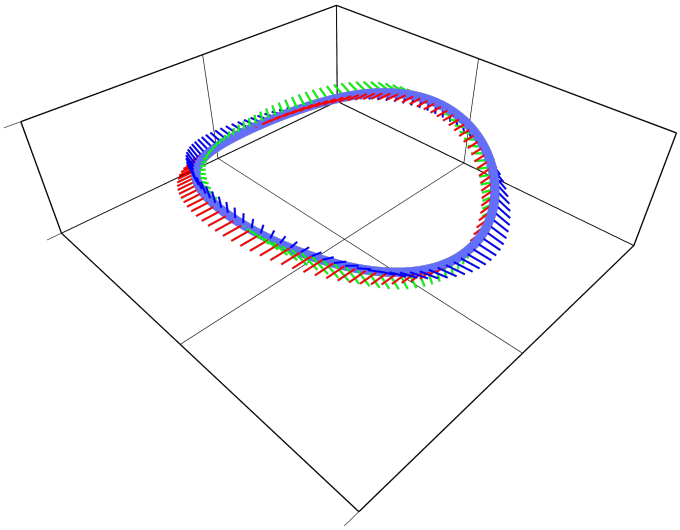


Figure 5.2: Curve $\mathcal{C}$ in $\mathbb{R}^3 \times \mathrm{SO}(3)$. Orientation frames are depicted using RGB axes.

The task can be achieved determining a control input $\boldsymbol{\xi}$ such that the body follows the curve $\mathcal{C}$ under a kinematic model. Although the system model was developed using the tuple representation of $\mathbb{R}^3 \times \mathrm{SO}(3)$, in this section we adopt the independent representation of ISE(3), as it simplifies the control design. Note that this identification is always possible, since for any $\mathbf{H} \in \mathrm{ISE}(3)$, we can identify $\boldsymbol{\chi} = (\mathbf{p}, \mathbf{R})$ using the following map

$$\begin{aligned} &\mathbf{H} \mapsto (\mathbf{BHc},\ \mathbf{AHA}^\top) \in \mathbb{R}^3 \times \mathrm{SO}(3), \\ &\mathbf{A} = \begin{bmatrix} \mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 4} \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{0}_{3\times 3} & \mathbf{I}_{3\times 3} & \mathbf{0} \end{bmatrix}, \quad \mathbf{c} = \begin{bmatrix} \mathbf{0}^\top & 1 \end{bmatrix}^\top. \end{aligned} \tag{5.23}$$

Now, let the system state be represented by an element $\mathbf{H}(t) \in \mathrm{ISE}(3)$, with a generalized twist $\boldsymbol{\xi} \in \mathbb{R}^6$. Given a curve parametrization $\mathbf{H}_d(s) \in \mathrm{ISE}(3)$, we want to compute a vector field $\Psi : \mathrm{ISE}(3) \to \mathbb{R}^6$ such that if $\boldsymbol{\xi} = \Psi(\mathbf{H})$, the system follows the curve $\mathcal{C}$. The system can be described through the following single integrator model:

$$\dot{\mathbf{H}} = \mathcal{S}(\boldsymbol{\xi})\, \mathbf{H}. \tag{5.24}$$

In order to use our vector field methodology, we define an EE-distance function $\widehat{D} : \mathrm{ISE}(3) \times \mathrm{ISE}(3) \to \mathbb{R}_+$ between two elements $\mathbf{V}, \mathbf{W}$ in the group ISE(3) as follows:

$$\widehat{D}(\mathbf{V}, \mathbf{W}) = \| \log(\mathbf{V}^{-1}\mathbf{W}) \|_F. \tag{5.25}$$

Note that this is exactly the same distance as defined in Section 4.8, and so it possesses the same properties.

Remark 5.3. In our original approach, we treat our system directly in $\mathbb{R}^3 \times \mathrm{SO}(3)$, and use the following distance function between two tuples:

$$\widehat{D}(\mathbf{p}_1, \mathbf{R}_1, \mathbf{p}_2, \mathbf{R}_2) \triangleq \frac{1}{2}\left(\|\mathbf{p}_1 - \mathbf{p}_2\|^2 + \beta \left\|\mathbf{I} - \mathbf{R}_2^T \mathbf{R}_1\right\|_F^2\right). \tag{5.26}$$

Although they are not equivalent, the change in the distance function will maintain consistency with our generalization to matrix Lie groups.

Note that the distance in (5.26) is left-invariant, and although it lacks a chainability proof, we can still show that the vector field it renders has a measure zero set of singularities. This change of distance is beneficial, as it aligns with all the properties and proofs given. □

We will assume that the target curve $\mathcal{C}$ is twice differentiable, closed, and without self-intersections. Thus, for an element $\mathbf{H} \in \mathrm{ISE}(3)$ and a curve parametrization $\mathbf{H}_d(s) \in \mathrm{ISE}(3)$, the vector field has the same expression

$$\Psi(\mathbf{H}) = k_N(\mathbf{H})\boldsymbol{\xi}_N(\mathbf{H}) + k_T(\mathbf{H})\boldsymbol{\xi}_T(\mathbf{H}). \tag{5.27}$$

By following this vector field, the system (5.24) will converge to and follow the target curve.

5.2.1 Computation of the EE-distance

It is possible to use a simple algorithm to compute the defined EE-distance (5.25). First, note that for any elements $\mathbf{V}, \mathbf{W} \in \mathrm{ISE}(3)$, we can write

$$\mathbf{V}^{-1}\mathbf{W} = \mathbf{Z} = \begin{bmatrix} \mathbf{R}_v^\top \mathbf{R}_w & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{p}_w - \mathbf{p}_v \\ \mathbf{0} & \mathbf{0} & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{Q} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{u} \\ \mathbf{0} & \mathbf{0} & 1 \end{bmatrix}, \tag{5.28}$$

where $\mathbf{R}_v$, $\mathbf{R}_w$, $\mathbf{p}_v$, and $\mathbf{p}_w$ are the rotation matrices and positions of $\mathbf{V}$ and $\mathbf{W}$, respectively. Using Cayley-Hamilton's theorem and properties of block triangular matrices, it is possible to write

$$\log(\mathbf{Z}) = \begin{bmatrix} \log \mathbf{Q} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{u} \\ \mathbf{0} & \mathbf{0} & 0 \end{bmatrix}. \tag{5.29}$$

Now, applying the Frobenius norm, we have

$$\|\log(\mathbf{Z})\|_F^2 = \operatorname{tr}\Big(\log(\mathbf{Z})^\top \log(\mathbf{Z})\Big)^{\frac{1}{2}} = \operatorname{tr}\left(\begin{bmatrix} (\log \mathbf{Q})^\top \log \mathbf{Q} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{u}^\top \mathbf{u} \end{bmatrix}\right)^{\frac{1}{2}} \tag{5.30}$$

$$= \sqrt{\|\log \mathbf{Q}\|_F^2 + \|\mathbf{u}\|^2} = \sqrt{2\theta^2 + \|\mathbf{u}\|^2}, \tag{5.31}$$

where θ is the rotation angle as in Appendix B. Thus, in this case the algorithm to compute the EE-distance is a simplified version of Algorithm 4.1, as shown in Algorithm 5.1.

Algorithm 5.1 Computation of $\widehat{D}(\mathbf{V}, \mathbf{W})$ in ISE(3)

Input: Matrices $\mathbf{V}, \mathbf{W}$
Output: Distance $\widehat{D}$
1: Compute $\mathbf{Z} \leftarrow \mathbf{V}^{-1}\mathbf{W}$
2: Extract $\mathbf{Q} \leftarrow$ rotation part of $\mathbf{Z}$ and $\mathbf{u} \leftarrow$ translation part of $\mathbf{Z}$
3: $u \leftarrow \frac{1}{2}(\operatorname{tr}(\mathbf{Q}) - 1)$
4: $v \leftarrow \frac{1}{2\sqrt{2}}\|\mathbf{Q} - \mathbf{Q}^\top\|_F$
5: $\theta \leftarrow \operatorname{atan2}(v, u)$
6: $\widehat{D} \leftarrow \sqrt{2\theta^2 + \mathbf{u}^\top \mathbf{u}}$

5.3 Adaptive control

In this section, we develop the adaptive control strategy to guide the manipulated object along the predefined curve $\mathcal{C} \subset \mathbb{R}^3 \times \mathrm{SO}(3)$. The goal is to provide the necessary total wrench $\boldsymbol{\tau}$ to ensure the system's velocity converges to the vector field Ψ, thereby enabling the object to follow the curve $\mathcal{C}$.

To develop this dynamic controller, we modify the adaptive control strategy proposed in Culbertson et al. (2021). In their approach, an error vector is established between the system velocity and a velocity reference, resulting in a non-autonomous system. In contrast, our method defines the velocity error $\boldsymbol{\zeta}$ with respect to the vector field, thereby establishing an autonomous system. This error vector is expressed as:

$$\boldsymbol{\zeta} = \dot{\boldsymbol{\chi}} - \Psi, \tag{5.32}$$

If the dynamic controller guarantees that $\boldsymbol{\zeta}$ goes to zero, the system follows the integrator model (5.24), with the vector field Ψ acting as its input. Therefore, as shown in Chapter 4, we anticipate that the system will exhibit the desired behavior. To develop the adaptive control strategy, we first define our reference model.

Given that the system is overactuated (meaning multiple sets of control inputs $\boldsymbol{\tau}_i$ produce identical object dynamics), each agent can contribute a portion of the control

effort. Thus, we define a set of N positive constants α_i, ensuring $\sum_{i=1}^{N} \alpha_i = 1$. Furthermore, analogously to the Euler-Lagrange formulation, the reference model can be linearly parametrized. Thus, the reference model can be expressed as

$$\alpha_i \left(\mathbf{M}(\boldsymbol{\chi})\dot{\Psi} + \mathbf{C}(\boldsymbol{\chi}, \dot{\boldsymbol{\chi}})\Psi + \mathbf{g} \right) = \mathbf{Y}_o(\boldsymbol{\chi}, \dot{\boldsymbol{\chi}}, \Psi, \dot{\Psi})\mathbf{o}_i, \tag{5.33}$$

where $\mathbf{Y}_o$ is the regressor matrix given by

$$\mathbf{Y}_o = \begin{bmatrix} \ddot{\mathbf{p}}_d & -\mathcal{S}(\dot{\boldsymbol{\omega}}_d)\,\mathbf{R} - \mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}(\boldsymbol{\omega}_d)\,\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathcal{S}(\ddot{\mathbf{p}}_d)\,\mathbf{R} + \mathcal{S}(\boldsymbol{\omega})\,\mathcal{S}(\dot{\mathbf{p}}_d)\,\mathbf{R} - \mathcal{S}(\boldsymbol{\omega}_d)\,\mathcal{S}(\dot{\mathbf{p}})\,\mathbf{R} & \mathbf{R}\mathcal{J}\left(\mathbf{R}\dot{\boldsymbol{\omega}}_d\right) + \mathcal{S}(\boldsymbol{\omega})\,\mathbf{R}\mathcal{J}\left(\mathbf{R}^\top\boldsymbol{\omega}_d\right) \end{bmatrix}, \tag{5.34}$$

where $\mathcal{J} : \mathbb{R}^3 \to \mathbb{R}^{3\times 6}$ is the following map

$$\mathcal{J}\left([a_1\ a_2\ a_3]^\top\right) = \begin{bmatrix} a_1 & a_2 & a_3 & 0 & 0 & 0 \\ 0 & a_1 & 0 & a_2 & a_3 & 0 \\ 0 & 0 & a_1 & 0 & a_2 & a_3 \end{bmatrix}. \tag{5.35}$$

The term $\mathbf{o}_i$ is the parameter vector given by

$$\mathbf{o}_i = \alpha_i \begin{bmatrix} m & m\mathbf{r}_p & \{\mathbb{I}_p\}_{11} & \{\mathbb{I}_p\}_{12} & \{\mathbb{I}_p\}_{22} & \{\mathbb{I}_p\}_{23} & \{\mathbb{I}_p\}_{33} \end{bmatrix}^\top, \tag{5.36}$$

where $\{\mathbb{I}_p\}_{ij}$ are the elements of the inertia matrix $\mathbb{I}_p^{\mathfrak{B}}$ at the i^{th} row and j^{th} column. Note that, since the inertia tensor is symmetric, we only need to consider the upper triangular part of the matrix. Also, note that the sum $\sum_{i=1}^{N} \mathbf{Y}_o\mathbf{o}_i = \mathbf{Y}_o\mathbf{o}$ results in the total wrench $\boldsymbol{\tau}$ applied to the object. Moreover, since the agents estimate the scaled vector $\mathbf{o}_i$, the constants α_i need not be known a priori.

Next, we propose the following control law

$$\boldsymbol{\tau}_i = \mathbf{G}(\boldsymbol{\chi}, \hat{\mathbf{r}}_i)^{-1}\boldsymbol{\eta}_i, \tag{5.37}$$

where $\hat{\mathbf{r}}_i$ represents the estimate of the i^{th} agent's displacement with respect to the measurement point, and

$$\boldsymbol{\eta}_i = \mathbf{Y}_o\hat{\mathbf{o}}_i - \mathbf{K}_D\boldsymbol{\zeta}, \tag{5.38}$$

where $\mathbf{K}_D$ is a positive definite matrix, and $\hat{\mathbf{o}}_i$ is the i^{th} agent's estimate of the object's parameters.

If we let $\tilde{\mathbf{r}}_i = \hat{\mathbf{r}}_i - \mathbf{r}_i$, we can achieve the following linear parametrization:

$$-\mathbf{G}(\boldsymbol{\chi}, \tilde{\mathbf{r}}_i)\boldsymbol{\eta}_i = \mathbf{Y}_r\left(\boldsymbol{\eta}_i, \boldsymbol{\chi}\right)\tilde{\mathbf{r}}_i. \tag{5.39}$$

With this parametrization and the other expressions, we propose the following adaptation

laws:

$$\dot{\hat{\mathbf{o}}}_i = -\boldsymbol{\Gamma}_o \mathbf{Y}_o^\top \left(\boldsymbol{\chi}, \dot{\boldsymbol{\chi}}, \Psi, \dot{\Psi}\right) \boldsymbol{\zeta}, \tag{5.40}$$

$$\dot{\hat{\mathbf{r}}}_i = -\boldsymbol{\Gamma}_r \mathbf{Y}_r^\top \left(\boldsymbol{\eta}_i, \boldsymbol{\chi}\right) \boldsymbol{\zeta}, \tag{5.41}$$

where $\boldsymbol{\Gamma}_o$ and $\boldsymbol{\Gamma}_r$ are definite positive matrices corresponding to the adaptation gains. With the proposed control law (5.37) and adaptation laws (5.40) and (5.41), we can guarantee that the system follows the curve $\mathcal{C}$ as proven in the following theorem.

Theorem 5.4. *Under the adaptive control laws* (5.37) *and* (5.38)*, and adaptation laws* (5.40) *and* (5.41)*, the system converges to and follows the curve* $\mathcal{C}$*.*

Proof. Consider the following Lyapunov function candidate

$$V = \frac{1}{2}\left(\boldsymbol{\zeta}^\top \mathbf{M} \boldsymbol{\zeta} + \sum_{i=1}^{N} \tilde{\mathbf{o}}_i^\top \boldsymbol{\Gamma}_o^{-1} \tilde{\mathbf{o}}_i + \tilde{\mathbf{r}}_i^\top \boldsymbol{\Gamma}_r^{-1} \tilde{\mathbf{r}}_i\right), \tag{5.42}$$

where $\tilde{\mathbf{o}}_i = \hat{\mathbf{o}}_i - \mathbf{o}_i$ represents the parameter estimation error for the i^{th} agent.

Since the true parameters do not vary in time, i.e. $\dot{\tilde{\mathbf{o}}}_i = \dot{\hat{\mathbf{o}}}_i$ and $\dot{\tilde{\mathbf{r}}}_i = \dot{\hat{\mathbf{r}}}_i$, the time derivative of the Lyapunov candidate yields

$$\dot{V} = \boldsymbol{\zeta}^\top \mathbf{M} \dot{\boldsymbol{\zeta}} + \frac{1}{2} \boldsymbol{\zeta}^\top \dot{\mathbf{M}} \boldsymbol{\zeta} + \sum_{i=1}^{N} \tilde{\mathbf{o}}_i^\top \boldsymbol{\Gamma}_o^{-1} \dot{\hat{\mathbf{o}}}_i + \tilde{\mathbf{r}}_i^\top \boldsymbol{\Gamma}_r^{-1} \dot{\hat{\mathbf{r}}}_i. \tag{5.43}$$

Using the fact that $\dot{\boldsymbol{\zeta}} = \ddot{\boldsymbol{\chi}} - \dot{\Psi}$ and $\dot{\boldsymbol{\chi}} = \boldsymbol{\zeta} + \Psi$, along with the system dynamics equation (5.11) and expression (5.19), and (5.33), we can express the first term as

$$\begin{aligned}
\boldsymbol{\zeta}^\top \mathbf{M} \dot{\boldsymbol{\zeta}} &= \boldsymbol{\zeta}^\top \mathbf{M}\left(\ddot{\boldsymbol{\chi}} - \dot{\Psi}\right) = \boldsymbol{\zeta}^\top \left(\boldsymbol{\tau} - \mathbf{C}\dot{\boldsymbol{\chi}} - \mathbf{g} - \mathbf{M}\dot{\Psi}\right) \\
&= \boldsymbol{\zeta}^\top \left(\left(\sum_{i=1}^{N} \mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i)\boldsymbol{\tau}_i\right) - \mathbf{C}\Psi - \mathbf{C}\boldsymbol{\zeta} - \mathbf{g} - \left(-\mathbf{C}\Psi - \mathbf{g} + \sum_{i=1}^{N} \mathbf{Y}_o \mathbf{o}_i\right)\right) \\
&= \boldsymbol{\zeta}^\top \left(-\mathbf{C}\boldsymbol{\zeta} + \sum_{i=1}^{N} \mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i)\boldsymbol{\tau}_i - \mathbf{Y}_o \mathbf{o}_i\right).
\end{aligned} \tag{5.44}$$

Now, using equations (5.37), (5.38) and (5.39), the first term in (5.44) can be rewritten as

$$\boldsymbol{\zeta}^\top \mathbf{M}\dot{\boldsymbol{\zeta}} = \boldsymbol{\zeta}^\top \left(-\mathbf{C}\boldsymbol{\zeta} + \sum_{i=1}^{N} \mathbf{G}(\boldsymbol{\chi}, \mathbf{r}_i)\mathbf{G}(\boldsymbol{\chi}, \hat{\mathbf{r}}_i)^{-1}\boldsymbol{\eta}_i - \mathbf{Y}_o \mathbf{o}_i \right) \tag{5.45}$$

$$= \boldsymbol{\zeta}^\top \left(-\mathbf{C}\boldsymbol{\zeta} + \sum_{i=1}^{N} (\mathbf{I} - \mathbf{G}(\boldsymbol{\chi}, \tilde{\mathbf{r}}_i))\boldsymbol{\eta}_i - \mathbf{Y}_o \mathbf{o}_i \right) \tag{5.46}$$

$$= \boldsymbol{\zeta}^\top \left(-\mathbf{C}\boldsymbol{\zeta} + \sum_{i=1}^{N} \left(\mathbf{Y}_o \hat{\mathbf{o}}_i + \mathbf{Y}_r \tilde{\mathbf{r}}_i - \mathbf{Y}_o \mathbf{o}_i - \mathbf{K}_D \boldsymbol{\zeta} \right) \right) \tag{5.47}$$

$$= \boldsymbol{\zeta}^\top \left(-\mathbf{C}\boldsymbol{\zeta} + \sum_{i=1}^{N} \left(\mathbf{Y}_o \tilde{\mathbf{o}}_i + \mathbf{Y}_r \tilde{\mathbf{r}}_i - \mathbf{K}_D \boldsymbol{\zeta} \right) \right). \tag{5.48}$$

Finally, substituting (5.48) into (5.43) renders the following result

$$\dot{V} = \frac{1}{2}\boldsymbol{\zeta}^\top \left(\dot{\mathbf{M}} - 2\mathbf{C} \right) \boldsymbol{\zeta} + \sum_{i=1}^{N} \tilde{\mathbf{o}}_i^\top \boldsymbol{\Gamma}_o^{-1} \dot{\hat{\mathbf{o}}}_i + \tilde{\mathbf{r}}_i^\top \boldsymbol{\Gamma}_r^{-1} \dot{\hat{\mathbf{r}}}_i + \boldsymbol{\zeta}^\top \left(\mathbf{Y}_o \tilde{\mathbf{o}}_i + \mathbf{Y}_r \tilde{\mathbf{r}}_i - \mathbf{K}_D \boldsymbol{\zeta} \right). \tag{5.49}$$

Substituting adaptation laws (5.40) and (5.41), and using the fact that $\dot{\mathbf{M}} - 2\mathbf{C}$ is skew-symmetric, we have

$$\dot{V} = \sum_{i=1}^{N} -\tilde{\mathbf{o}}_i^\top \mathbf{Y}_o^\top \boldsymbol{\zeta} - \tilde{\mathbf{r}}_i^\top \mathbf{Y}_r^\top \boldsymbol{\zeta} + \boldsymbol{\zeta}^\top \mathbf{Y}_o \tilde{\mathbf{o}}_i + \boldsymbol{\zeta}^\top \mathbf{Y}_r \tilde{\mathbf{r}}_i - \boldsymbol{\zeta}^\top \mathbf{K}_D \boldsymbol{\zeta}. \tag{5.50}$$

Since all terms are scalars, we can transpose the first two terms and achieve the following result

$$\dot{V} = \sum_{i=1}^{N} -\boldsymbol{\zeta}^\top \mathbf{K}_D \boldsymbol{\zeta} = -N\boldsymbol{\zeta}^\top \mathbf{K}_D \boldsymbol{\zeta} \leq 0. \tag{5.51}$$

Since N is positive and $\mathbf{K}_D$ is definite positive, $\boldsymbol{\zeta}$ converges to zero as t goes to ∞. This implies that the velocity error is zero and the system behaves as the kinematic model (5.24). Thus we can invoke Theorem 4.28 to conclude that the system converges to and follows the curve $\mathcal{C}$. □

The connection of both controllers is shown in the block diagram of Figure 5.3.

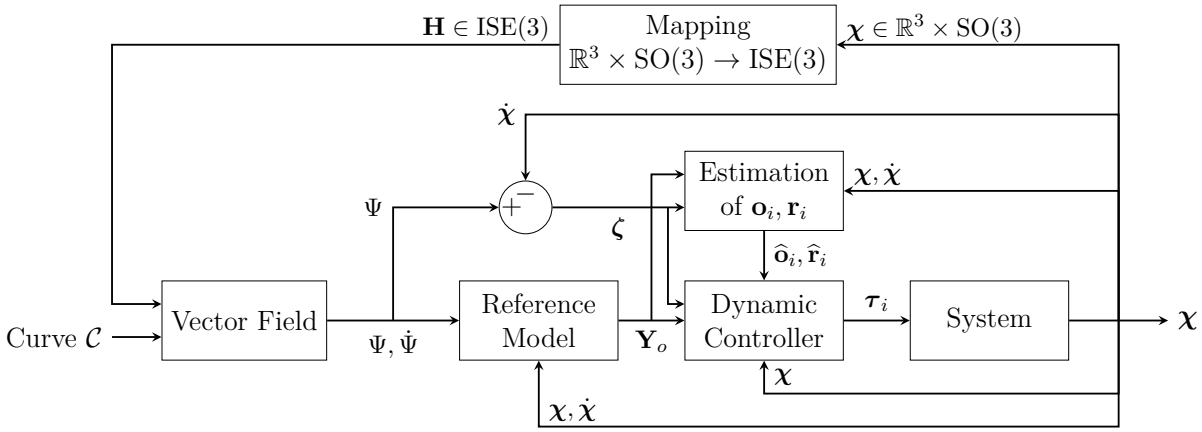


Figure 5.3: Block diagram of the collaborative manipulation control framework.

6

Results and Discussion

6.1 Kinematic control

In this section, we present the results of two kinematic control simulation to demonstrate the effectiveness of the proposed vector field guidance strategy. The first simulation involves a system following a curve $\mathcal{C}$ in SE(3), while the second simulation regards $\mathcal{C}$ in $\mathrm{SO}^+(3,1)$. As the control is kinematic, the system follows a single integrator model

$$\dot{\mathbf{H}}(t) = \mathcal{S}(\boldsymbol{\xi}(t))\,\mathbf{H}(t), \tag{6.1}$$

where our control input is the generalized twist $\boldsymbol{\xi}$.

6.1.1 Special Euclidean group

The target curve $\mathcal{C} \subset \mathrm{SE}(3)$ was based on a hyperbolic paraboloid rotating about the x-axis in the world frame. The parametrization of the curve $\mathbf{H}_d$ was defined as:

$$\mathbf{H}_d(s) = \begin{bmatrix} 1 & 0 & 0 & r\cos(2\pi s) \\ 0 & \cos(2\pi s) & \sin(2\pi s) & r\sin(2\pi s) \\ 0 & -\sin(2\pi s) & \cos(2\pi s) & b + dr^2\Big(\cos^2(2\pi s) - \sin^2(2\pi s)\Big) \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{6.2}$$

where $b = 1\,\text{m}$, $d = 0.2\,\text{m}$, and $r = 1\,\text{m}$. This curve was discretized into $P = 5000$ points by sampling s into 5000 evenly spaced points within the interval $[0, 1]$. To distinguish the continuous and discrete parametrizations, we use the notations $\mathbf{H}_d[i]$ for the i-th sample of the discretized curve and $\mathbf{H}_d(s)$ for the continuous curve.

Choice of S map

For the simulation, we chose the canonical isomorphism $\mathcal{S}$ between $\mathbb{R}^6$ and $\mathfrak{se}(3)$. Specifically, let $\boldsymbol{\xi} = [\xi_1\ \xi_2\ \xi_3\ \xi_4\ \xi_5\ \xi_6]^\top$, the used $\mathcal{S}$ is:

$$\mathcal{S}(\boldsymbol{\xi}) = \begin{bmatrix} 0 & -\xi_6 & \xi_5 & \xi_1 \\ \xi_6 & 0 & -\xi_4 & \xi_2 \\ -\xi_5 & \xi_4 & 0 & \xi_3 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \tag{6.3}$$

The basis $\mathbf{E}_1, ..., \mathbf{E}_6$ of the Lie algebra $\mathfrak{se}(3)$ can be obtained by setting $\mathbf{E}_k = \mathcal{S}(\mathbf{e}_k)$, where $\mathbf{e}_k$ are the columns of the 6×6 identity matrix. Geometrically, the interpretation of this basis choice is that $\boldsymbol{\xi}$ is the classical *twist* in mechanics. More precisely, $\boldsymbol{\omega} \triangleq [\xi_4\ \xi_5\ \xi_6]^\top$ represents the x, y and z-axes components of the angular velocity measured in the inertial frame, whereas $\mathbf{v} \triangleq [\xi_1\ \xi_2\ \xi_3]^\top$ represents the linear velocities, along the x, y and z-axes, of a virtual point at the origin of the inertial frame, measured in this inertial frame. This is related to the object's linear velocity $\dot{\mathbf{p}}$ by $\dot{\mathbf{p}} = \boldsymbol{\omega} \times \mathbf{p} + \mathbf{v}$.

Employed numerical methods

To solve the optimization problem in (4.8) and compute the distance $D(\mathbf{H})$, we used the discretized curve and computed the nearest point $\mathbf{H}_d[i^*]$ through a brute-force approach. Although it would be possible to compute the curve derivative explicitly, we opted for finite differences. Thus, we approximated the derivative $\mathbf{H}'_d[i]$ as

$$\mathbf{H}'_d[i] \approx \begin{cases} \frac{\mathbf{H}_d[i+1]-\mathbf{H}_d[i]}{\Delta s}, & i < P \\ \frac{\mathbf{H}_d[1]-\mathbf{H}_d[i]}{\Delta s}, & i = P, \end{cases} \tag{6.4}$$

adopting $\Delta s = 1 \cdot 10^{-3}$. This implies that our tangent component $\boldsymbol{\xi}_T$ is also approximate, since

$$\boldsymbol{\xi}_T \approx \mathcal{S}^{-1}\Big(\mathbf{H}'_d[i^*]\mathbf{H}_d[i^*]^{-1}\Big). \tag{6.5}$$

For the computation of the normal component $\boldsymbol{\xi}_N$, we adopted an approximation by evaluating the right-hand side of (4.2) for each $\mathbf{e}_i$. This gives the i^{th} normal component

element $\{\boldsymbol{\xi}_N\}_i$ as:

$$\{\boldsymbol{\xi}_N\}_i \approx \frac{\widehat{D}\Big(\exp\big(\mathcal{S}(\mathbf{e}_i)\big)\mathbf{H}(t), \mathbf{H}_d[s^*]\Big) - \widehat{D}\Big(\mathbf{H}(t), \mathbf{H}_d[s^*]\Big)}{\varepsilon}, \tag{6.6}$$

in which $\varepsilon = 1 \cdot 10^{-3}$ was used.

The system was simulated adopting the following approximation:

$$\mathbf{H}(t + \Delta t) \approx \exp\Big(\mathcal{S}\big(\Psi(\mathbf{H})\big)\Delta t\Big)\mathbf{H}(t), \tag{6.7}$$

using a time step of $\Delta t = 1 \cdot 10^{-2}\,\mathrm{s}$.

Vector field gains and initial condition

The vector field gains were selected to complement each other, ensuring that the normal component would dominate when the system was far from the curve, and the tangent component would dominate when the system was near the curve. Based on this reasoning, the following gains were chosen:

$$k_N(\mathbf{H}) = \tanh\Big(20D(\mathbf{H})\Big), \tag{6.8}$$

$$k_T(\mathbf{H}) = 1 - 0.5\tanh\Big(D(\mathbf{H})\Big). \tag{6.9}$$

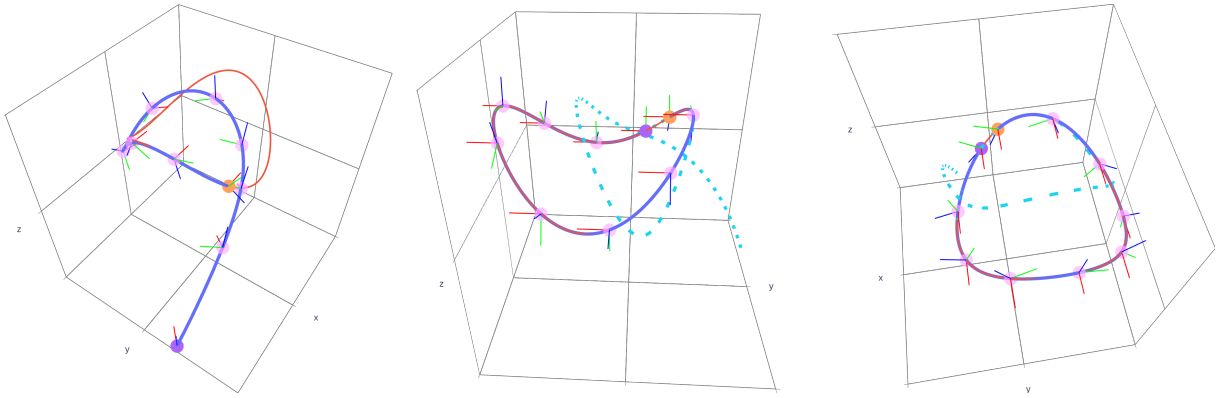
The initial condition $\mathbf{H}(0)$ was set as

$$\mathbf{H}(0) = \begin{bmatrix} \frac{\sqrt{2}}{2} & -\frac{\sqrt{2}}{2} & 0 & -2 \\ \frac{\sqrt{2}}{2} & \frac{\sqrt{2}}{2} & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{6.10}$$

Simulation

The simulation was conducted over a period of 15 s and implemented in C++. On average, the computation of the vector field took (60 ± 5) ms per iteration on a single core of an Intel i5-10300H @ 4.5GHz CPU, with 8 GB of RAM. On average, 99.5% of this time was spent solving the optimization problem in (4.8). Since the optimization process is highly parallelizable – allowing simultaneous computation of $\widehat{D}(\mathbf{H}, \mathbf{H}_d(s))$ for different discretized values of s – the computational effort could be significantly reduced by leveraging parallel architectures such as GPUs, SIMD, or multi-threading.

The system's trajectory is illustrated in Figure 6.1, where the blue line represents the system trajectory, the red line represents the target curve, and the system state is depicted by colored spheres. The values of the distance function D are shown in Figure 6.2. Additionally, Figure 6.2 provides a more intuitive representation of the position error (in



(a) $t \in [0, 5]s$ (b) $t \in [5, 9.7]s$ (c) $t \in [9.7, 14.5]s$

Figure 6.1: The solid blue line depicts the system trajectory, while the solid red line indicates the target curve. The dashed light blue line represents the past trajectory. The initial and final positions are marked by purple and orange spheres, respectively. Translucent pink spheres denote intermediate positions, with orientation frames shown by RGB axes.

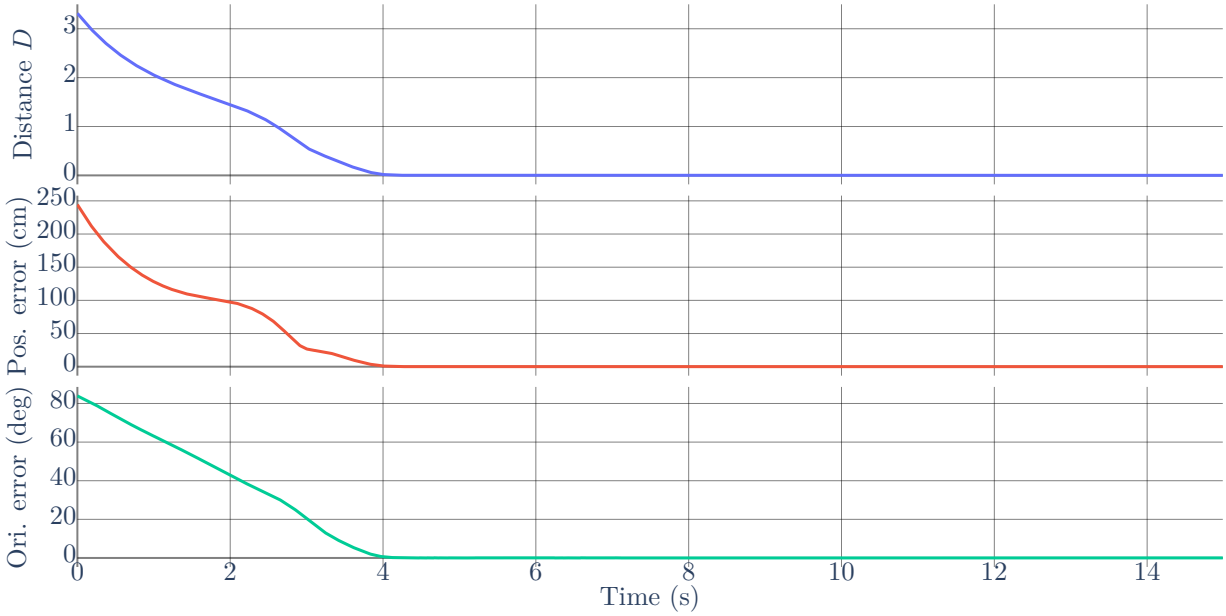


Figure 6.2: Value of EC-distance D, position error in centimeters and orientation error in degrees over time.

centimeters) and the orientation error (in degrees). These results confirm that the system successfully converges to the desired curve and circulates around it as expected. Once the system reaches the curve, it remains there without deviation.

Although our implementation was carried out in C++, we provide a less efficient sample code in Python for clarity and accessibility, available at `https://github.com/fbartelt/SE3-example/blob/main/SE3example.ipynb`.

6.1.2 Proper orthochronous Lorentz group

Before describing the simulation conducted in $\mathrm{SO}^+(3,1)$, we first introduce key concepts from special relativity. Relativistic motion takes place within Minkowski space, a four-dimensional representation of spacetime. The elements of this space, known as *events*, are represented by vectors $\mathbf{p} = [x\ y\ z\ t]^\top \in \mathbb{R}^4$, where x, y, and z are spatial coordinates and t is the time coordinate. Since effects such as time dilation and length contraction depend on the relative motion of observers, a transformation of reference frames is necessary to relate the coordinates of events between different observers. These transformations are known as *Lorentz transformations*.

Lorentz transformations form a group called the *Lorentz group*, encompassing spatial rotations, reflections, and *boosts* (Carroll, 2019, p. 12) – the latter being transformations in Minkowski space without rotation, which are analogous to linear velocities in Minkowski space. Specifically, the proper orthochronous Lorentz group, $\mathrm{SO}^+(3,1)$, is a subgroup of the Lorentz group that includes only rotations and boosts preserving the direction of time. The elements of $\mathrm{SO}^+(3,1)$ are 4×4 matrices that preserve the *Minkowski metric*, defined by:

$$\nu^2 = x^2 + y^2 + z^2 - c^2t^2, \tag{6.11}$$

where ν is the *spacetime interval*, and c is the speed of light.

If an observer O_2 moves near the speed of light, they will perceive an event with coordinates $\mathbf{p}' = [x'\ y'\ z'\ t']^\top$ that differ from the coordinates $\mathbf{p}$ observed by an inertial observer O_1. These coordinates are related by the transformation $\mathbf{p}' = \mathbf{\Lambda p}$, where $\mathbf{\Lambda}$ is a Lorentz transformation. For instance, if O_2 moves along the x-axis with velocity v, the Lorentz transformation is given by

$$\begin{aligned} x' &= \gamma(x - vt), \\ y' &= y, \\ z' &= z, \\ t' &= \gamma\left(t - \frac{vx}{c^2}\right), \end{aligned} \tag{6.12}$$

where $\gamma = \frac{1}{\sqrt{1-v^2/c^2}}$ is the *Lorentz factor*.

As shown in (6.12), observer O_2 experiences time dilation when traveling near the speed of light. It is common to interpret the differing "times" as if each reference frame has an attached clock that ticks at a distinct rate. The time measured by a clock in the inertial frame O_1 is referred to as the *coordinate time*, while the time measured by a clock in the relativistic frame O_2 is called the *proper time*. The proper time τ is related to the coordinate time t via $\frac{dt}{d\tau} = \gamma$ (Steane, 2012, p. 32).

Although there exists a physical meaning for this group, we only use it as a mathematical structure in our simulation. As such, we do not provide any interpretation for the curve following behavior. The equations shown are valid for non-varying boosts along one axis, however, the general case is more complex and would require a more thorough analaysis. While the curve was constructed in a simple way, the motion generated by the vector field might induce both spatial rotations and boosts along any direction.

With these concepts established, we now turn to the simulation conducted in $\mathrm{SO}^+(3,1)$. For simplicity, we adopt $c = 1$, as is customary in special relativity, implying all quantities are dimensionless.

Target curve

The target curve $\mathcal{C} \subset \mathrm{SO}^+(3,1)$ was chosen as a periodic boost along the x-axis. The parametrization of the curve, denoted by $\mathbf{H}_d$, is defined as:

$$\mathbf{H}_d(s) = \begin{bmatrix} \gamma(s) & 0 & 0 & -\gamma(s)v(s) \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\gamma(s)v(s) & 0 & 0 & \gamma(s) \end{bmatrix}, \tag{6.13}$$

where

$$v(s) = 0.9 + \frac{0.09}{2}(\cos(2\pi s) + 1), \tag{6.14}$$

$$\gamma(s) = \frac{1}{\sqrt{1 - v(s)^2}}. \tag{6.15}$$

This curve was discretized into $P = 30\,000$ points by sampling s into 30 000 evenly spaced points within the interval $[0, 1]$. To distinguish the continuous and discrete parametrizations, we again adopt the notations $\mathbf{H}_d[i]$ for the i-th sample of the discretized curve and $\mathbf{H}_d(s)$ for the continuous curve.

Choice of S map

The selected S map is the same as the one presented in Example 3.30. Thus, for a vector $\boldsymbol{\xi} = [\xi_1\ \xi_2\ \xi_3\ \xi_4\ \xi_5\ \xi_6]^\top$, the corresponding isomorphism $\mathcal{S} : \mathbb{R}^6 \to \mathfrak{so}(3,1)$ is given by:

$$\mathcal{S}(\boldsymbol{\xi}) = \begin{bmatrix} 0 & -\xi_3 & \xi_2 & \xi_4 \\ \xi_3 & 0 & -\xi_1 & \xi_5 \\ -\xi_2 & \xi_1 & 0 & \xi_6 \\ \xi_4 & \xi_5 & \xi_6 & 0 \end{bmatrix}. \tag{6.16}$$

Employed numerical methods

The optimization problem in (4.8) was solved using the discretized curve, with the nearest point $\mathbf{H}_d[i^*]$ computed through a brute-force approach. The derivative $\mathbf{H}'_d[i]$ was obtained as the explicit derivative of the discrete version of (6.13). The normal component was approximated following the same method as in the previous simulation. Specifically, the i^{th} normal component element $\{\boldsymbol{\xi}_N\}_i$ as:

$$\{\boldsymbol{\xi}_N\}_i \approx \frac{\widehat{D}\Big(\exp\big(\mathcal{S}(\mathbf{e}_i)\big)\mathbf{H}(t), \mathbf{H}_d[s^*]\Big) - \widehat{D}\Big(\mathbf{H}(t), \mathbf{H}_d[s^*]\Big)}{\varepsilon}, \tag{6.17}$$

in which $\varepsilon = 1 \cdot 10^{-5}$ was used.

The system was also simulated adopting the same approximation

$$\mathbf{H}(t + \Delta t) \approx \exp\Big(\mathcal{S}\big(\Psi(\mathbf{H})\big)\Delta t\Big)\mathbf{H}(t), \tag{6.18}$$

using a time step of $\Delta t = 1 \cdot 10^{-3}\,\text{s}$.

Vector field gains and initial condition

The gains were selected following the same reasoning as in the previous simulation, ensuring that the normal component would dominate when the system was far from the curve, while the tangent component would dominate when the system was near the curve. The chosen gains were:

$$k_N(\mathbf{H}) = \tanh\Big(1000 D(\mathbf{H})\Big) \tag{6.19}$$

$$k_T(\mathbf{H}) = 0.5\Big(1 - \tanh\Big(100 D(\mathbf{H})\Big)\Big). \tag{6.20}$$

The initial condition $\mathbf{H}(0)$ was set as a boost along the x-axis in the opposite direction:

$$\mathbf{H}(0) = \exp\Big(\mathcal{S}(\boldsymbol{\xi}_0)\Big), \tag{6.21}$$

where $\boldsymbol{\xi}_0 = [0\ 0\ 0\ 0.7\ 0\ 0]^\top$.

Simulation

As it is unfeasible to plot the behavior of the system in $\mathrm{SO}^+(3,1)$ directly, we can only provide the value of the distance function along the simulated time, which is shown in Figure 6.3. The system successfully converges to the desired curve, achieving a final distance of approximately 0.02, which was deemed acceptable but could potentially be reduced by adjusting ε in the approximation of the normal component. Although slowly,

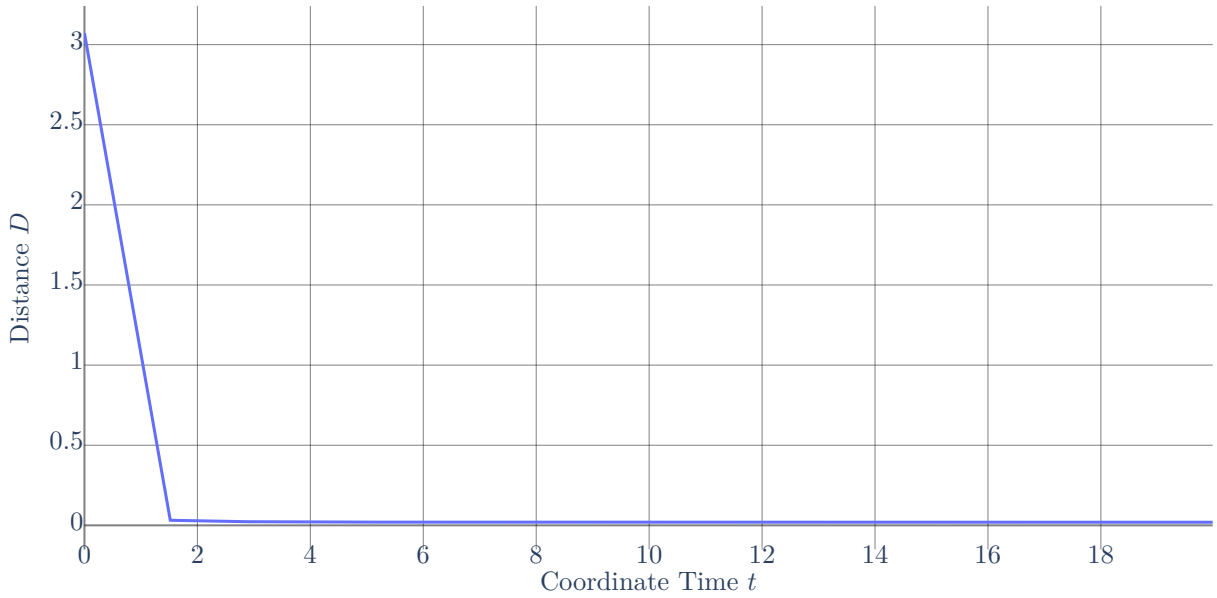


Figure 6.3: Value of EC-distance D in $\mathrm{SO}^+(3,1)$ over time.

the magnitude of the distance is decreasing along the interval shown.

6.2 Collaborative Manipulation

For the adaptive control simulation described in Chapter 5, we considered the manipulation of a large cylindrical body by six agents. The goal was to converge to and circulate a curve $\mathcal{C} \subset \mathbb{R}^3 \times \mathrm{SO}(3)$. The cylindrical body had a radius r of 0.25 m, a height h of 1 m, and a mass m of $1.28 \cdot 10^3$ kg. The measurement point $\mathbf{r}_p$ was taken as the position of the first agent, $\mathbf{r}_1 = [0\ 0\ 1.5]^\top$. The positions of the other agents, along with the components of the inertia tensor, are provided in Table 6.1.

Table 6.1: Adaptive simulation parameters

Parameter	Value	Parameter	Value
r	0.25 m	$\mathbf{r}_1$	$[0\ 0\ 1.5]^\top$ m
h	1 m	$\mathbf{r}_2$	$[0\ 0\ -1.5]^\top$ m
m	$1.28 \cdot 10^3$ kg	$\mathbf{r}_3$	$[0.5\ 0\ 0]^\top$ m
$\mathbb{I}_{\mathrm{cm},xx}$	$1.56 \cdot 10^2$ kg m^2	$\mathbf{r}_4$	$[-0.5\ 0\ 0]^\top$ m
$\mathbb{I}_{\mathrm{cm},yy}$	$1.56 \cdot 10^2$ kg m^2	$\mathbf{r}_5$	$[0\ 0.5\ 0]^\top$ m
$\mathbb{I}_{\mathrm{cm},zz}$	$4.93 \cdot 10^1$ kg m^2	$\mathbf{r}_6$	$[0\ -0.5\ 0]^\top$ m
$\mathbf{r}_p$	$[0\ 0\ 1.5]^\top$ m		

Note: The inertia tensor $\mathbb{I}_{\mathrm{cm}}$ is diagonal, with the components $\mathbb{I}_{\mathrm{cm},xx}, \mathbb{I}_{\mathrm{cm},yy}$ and $\mathbb{I}_{\mathrm{cm},zz}$ representing the diagonal elements in order.

6.2.1 Curve parametrization and S map

In this simulation, we employed a more complex curve in the group ISE(3), defined as:

$$\mathbf{H}_d(s) = \begin{bmatrix} \mathbf{R}_d(s) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{p}_d(s) \\ \mathbf{0} & \mathbf{0} & 1 \end{bmatrix}, \tag{6.22}$$

where

$$\mathbf{p}_d(s) = \begin{bmatrix} 0.7(\sin(2\pi s) + 2\sin(4\pi s)) \\ 0.7(\cos(2\pi s) - 2\cos(4\pi s)) \\ 0.4 - 0.7\sin(6\pi s) \end{bmatrix}, \tag{6.23}$$

$$\mathbf{R}_d(s) = \begin{bmatrix} \cos(2\pi s) & -\sin(2\pi s) & 0 \\ \sin(2\pi s) & \cos(2\pi s) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(4\pi s) & -\sin(4\pi s) \\ 0 & \sin(4\pi s) & \cos(4\pi s) \end{bmatrix}. \tag{6.24}$$

Instead of using the continuous curve, we discretized it into $P = 2000$ points by sampling the parameter s into 2000 evenly spaced values within the interval $[0, 1]$. To distinguish between the discrete and continuous representations, we henceforth use the notations $\mathbf{H}_d[i]$, $\mathbf{p}_d[i]$ and $\mathbf{R}_d[i]$ to refer to the i^{th} sample of the curve, position and orientation, respectively.

We selected the following S map for a vector $\boldsymbol{\xi} = [\xi_1\ \xi_2\ \xi_3\ \xi_4\ \xi_5\ \xi_6]^\top = [\mathbf{v}^\top\ \boldsymbol{\omega}^\top]^\top$, where $\mathbf{v}, \boldsymbol{\omega} \in \mathbb{R}^3$:

$$\mathcal{S}(\boldsymbol{\xi}) = \begin{bmatrix} \widehat{\mathcal{S}}(\boldsymbol{\omega}) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{v} \\ \mathbf{0} & \mathbf{0} & 0 \end{bmatrix}, \tag{6.25}$$

where $\widehat{\mathcal{S}}$ is the canonical isomorphism for $\mathfrak{so}(3)$ (see Example 3.25).

6.2.2 Employed numerical methods

Although the curve is represented in ISE(3), our adaptive control formulation used the tuple representation of $\mathbb{R}^3 \times \mathrm{SO}(3)$. To avoid confusion, we use $\mathbf{H}$ to represent the element of ISE(3) and $\boldsymbol{\chi} = (\mathbf{R}, \mathbf{p})$ to represent the same entity as an element of $\mathbb{R}^3 \times \mathrm{SO}(3)$. This equivalence implies that $\boldsymbol{\xi} = \dot{\boldsymbol{\chi}}$ and $\dot{\boldsymbol{\xi}} = \ddot{\boldsymbol{\chi}}$, thus we adopt the notation using $\boldsymbol{\chi}$ for twists and accelerations to maintain consistency with the dynamics outlined in Section 5.1.

The reference model, as defined in (5.33), depends on the acceleration, which is the derivative of the vector field. This acceleration was approximated using the symmetric

difference quotient:

$$\dot{\Psi}\Big(\mathbf{H}(t)\Big) \approx \frac{\Psi\left(\exp\left(\mathcal{S}\Big(\Delta t \Psi\Big(\mathbf{H}(t)\Big)\Big)\right)\mathbf{H}(t)\right) - \Psi\left(\exp\left(\mathcal{S}\Big(-\Delta t \Psi\Big(\mathbf{H}(t)\Big)\Big)\right)\mathbf{H}(t)\right)}{2\Delta t}. \tag{6.26}$$

We then discuss the integration of the system using *Heun's method* (Campos Filho, 2007, p. 330), also known as the *improved Euler method*, which is a two-stage Runge-Kutta method. To describe how this method was applied in the context of Lie groups, we use the notation $\underline{\cdot}$ for the intermediate variables in Heun's method. Let $\mathbf{H}_{\text{ref}}$ denote the reference state, i.e., the state of the system if it were following the vector field perfectly. The update for the system's acceleration $\ddot{\boldsymbol{\chi}}$, the derivatives of the parameters $\dot{\widehat{\mathbf{o}}}_i$ and $\dot{\widehat{\mathbf{r}}}_i$, and the respective intermediate variables were taken as follows:

$$\begin{aligned}
\ddot{\underline{\boldsymbol{\chi}}}(t+\Delta t) &= \mathbf{M}\Big(\underline{\boldsymbol{\chi}}(t)\Big)^{\dagger(\epsilon)}\left(-\mathbf{C}\Big(\underline{\boldsymbol{\chi}}(t), \dot{\underline{\boldsymbol{\chi}}}(t)\Big)\dot{\underline{\boldsymbol{\chi}}}(t) + \sum_{i=1}^{6}\mathbf{G}\Big(\underline{\boldsymbol{\chi}}(t), \underline{\widehat{\mathbf{r}}}_i(t)\Big)^{-1}\underline{\boldsymbol{\eta}}_i(t)\right),\\
\ddot{\boldsymbol{\chi}}(t+\Delta t) &= \mathbf{M}\Big(\boldsymbol{\chi}(t)\Big)^{\dagger(\epsilon)}\left(-\mathbf{C}\Big(\boldsymbol{\chi}(t), \dot{\boldsymbol{\chi}}(t)\Big)\dot{\boldsymbol{\chi}}(t) + \sum_{i=1}^{6}\mathbf{G}\Big(\boldsymbol{\chi}(t), \widehat{\mathbf{r}}_i(t)\Big)^{-1}\boldsymbol{\eta}_i(t)\right),\\
\dot{\underline{\widehat{\mathbf{o}}}}_i(t+\Delta t) &= -\boldsymbol{\Gamma}_o\mathbf{Y}_o\Big(\underline{\boldsymbol{\chi}}(t), \dot{\underline{\boldsymbol{\chi}}}(t), \Psi\Big(\underline{\mathbf{H}}_{\text{ref}}(t)\Big), \dot{\Psi}\Big(\underline{\mathbf{H}}_{\text{ref}}(t)\Big)\Big)\underline{\boldsymbol{\zeta}},\\
\dot{\widehat{\mathbf{o}}}_i(t+\Delta t) &= -\boldsymbol{\Gamma}_o\mathbf{Y}_o\Big(\boldsymbol{\chi}(t), \dot{\boldsymbol{\chi}}(t), \Psi\Big(\mathbf{H}_{\text{ref}}(t)\Big), \dot{\Psi}\Big(\mathbf{H}_{\text{ref}}(t)\Big)\Big)\boldsymbol{\zeta},\\
\dot{\underline{\widehat{\mathbf{r}}}}_i(t+\Delta t) &= -\boldsymbol{\Gamma}_r\mathbf{Y}_r\Big(\underline{\boldsymbol{\eta}}_i(t), \underline{\boldsymbol{\chi}}(t)\Big)\underline{\boldsymbol{\zeta}},\\
\dot{\underline{\widehat{\mathbf{r}}}}_i(t+\Delta t) &= -\boldsymbol{\Gamma}_r\mathbf{Y}_r\Big(\boldsymbol{\eta}_i(t), \boldsymbol{\chi}(t)\Big)\boldsymbol{\zeta},
\end{aligned} \tag{6.27}$$

where $\underline{\boldsymbol{\eta}}_i(t) = \mathbf{Y}_o\Big(\underline{\boldsymbol{\chi}}(t), \dot{\underline{\boldsymbol{\chi}}}(t), \Psi\Big(\underline{\mathbf{H}}_{\text{ref}}(t)\Big), \dot{\Psi}\Big(\underline{\mathbf{H}}_{\text{ref}}(t)\Big)\Big)\underline{\widehat{\mathbf{o}}}_i(t) - \mathbf{K}_D\underline{\boldsymbol{\zeta}}$, the intermediate velocity error $\underline{\boldsymbol{\zeta}}$ is given by $\underline{\boldsymbol{\zeta}} = \dot{\underline{\boldsymbol{\chi}}}(t) - \Psi\Big(\underline{\mathbf{H}}_{\text{ref}}(t)\Big)$, and $\mathbf{M}^{\dagger(\epsilon)}$ is the damped pseudo-inverse $\mathbf{M}^{\dagger(\epsilon)} = (\mathbf{M}^\top\mathbf{M} + \epsilon\mathbf{I})^{-1}\mathbf{M}^\top$, with $\epsilon = 0.01$. The update for the system's twist and parameters estimates is given by:

$$\begin{aligned}
\dot{\underline{\boldsymbol{\chi}}}(t+\Delta t) &= \dot{\boldsymbol{\chi}}(t) + \Delta t\ddot{\underline{\boldsymbol{\chi}}}(t),\\
\dot{\boldsymbol{\chi}}(t+\Delta t) &= \dot{\boldsymbol{\chi}}(t) + \frac{\Delta t}{2}\Big(\ddot{\boldsymbol{\chi}}(t) + \ddot{\underline{\boldsymbol{\chi}}}(t)\Big),\\
\widehat{\mathbf{o}}_i(t+\Delta t) &= \widehat{\mathbf{o}}_i(t) + \frac{\Delta t}{2}\Big(\dot{\widehat{\mathbf{o}}}_i(t) + \dot{\underline{\widehat{\mathbf{o}}}}_i(t)\Big),\\
\widehat{\mathbf{r}}_i(t+\Delta t) &= \widehat{\mathbf{r}}_i(t) + \frac{\Delta t}{2}\Big(\dot{\widehat{\mathbf{r}}}_i(t) + \dot{\underline{\widehat{\mathbf{r}}}}_i(t)\Big).
\end{aligned} \tag{6.28}$$

Finally, the update of the system's state and reference model was given by

$$\begin{aligned}
\underline{\mathbf{H}}(t+\Delta t) &= \exp\Big(\Delta t\,\mathcal{S}\big(\dot{\boldsymbol{\chi}}(t)\big)\Big)\mathbf{H}(t),\\
\mathbf{H}(t+\Delta t) &= \exp\left(\frac{\Delta t}{2}\,\mathcal{S}\big(\dot{\boldsymbol{\chi}}(t)+\dot{\underline{\boldsymbol{\chi}}}(t)\big)\right)\mathbf{H}(t),\\
\underline{\mathbf{H}}_{\text{ref}}(t+\Delta t) &= \exp\Big(\Delta t\,\mathcal{S}\Big(\Psi\big(\mathbf{H}(t)\big)\Big)\Big)\mathbf{H}(t),\\
\mathbf{H}_{\text{ref}}(t+\Delta t) &= \mathbf{H}(t+\Delta t).
\end{aligned} \tag{6.29}$$

As defined in Section 5.2, the normal component is discontinuous. To avoid chattering and noise propagation caused by numerical differentiation, we employed a smoothing strategy for the distance function D based on the approach of Gonçalves et al. (2024). The smoothed distance $\bar{D}$ is defined in terms of the discretized curve as

$$\bar{D}(\mathbf{H}) = D(\mathbf{H}) - \delta \log\left(\frac{\sum_{i=1}^{P} w_i}{P}\right), \tag{6.30}$$

where $\delta = 0.05$ is a smoothing factor, $P = 2000$ is the total number of points in the curve, and w_i are the distance weights defined as

$$w_i = \frac{\exp\big(D(\mathbf{H}) - \widehat{D}(\mathbf{H}, \mathbf{H}_d[i])\big)}{\delta}. \tag{6.31}$$

With this, the L operator of the smoothed distance is given by

$$\mathrm{L}[\bar{D}](\mathbf{H}) = \frac{\sum_{i=1}^{P} w_i \mathrm{L}_{\mathbf{V}}[\widehat{D}]\big(\mathbf{H}, \mathbf{H}_d[i]\big)}{\sum_{i=1}^{P} w_i}, \tag{6.32}$$

which implies that the j^{th} entry of the normal component $\{\boldsymbol{\xi}_N\}_j$ can be approximated as

$$\{\boldsymbol{\xi}_N\}_j(\mathbf{H}) = -\frac{\sum_{i=1}^{P} \frac{w_i}{\varepsilon}\left(\widehat{D}\Big(\exp\big(\varepsilon\,\mathcal{S}(\mathbf{e}_j)\big)\mathbf{H}, \mathbf{H}_d[i]\Big) - \widehat{D}\big(\mathbf{H}, \mathbf{H}_d[i]\big)\right)}{\sum_{i=1}^{P} w_i}, \tag{6.33}$$

with $\varepsilon = 0.001$.

As the curve parametrization has an explicit equation (6.22), the derivative $\frac{d}{ds}\mathbf{H}_d(s)$ was computed analytically. Thus, the derivative of the i^{th} point in the discretized curve is given explicitly by $\mathbf{H}'_d[i]$. The smoothed tangent component $\boldsymbol{\xi}_T$ is then expressed as

$$\boldsymbol{\xi}_T(\mathbf{H}) = \frac{\sum_{i=1}^{P} w_i\, \mathcal{S}^{-1}\big(\mathbf{H}'_d[i]\mathbf{H}_d[i]^{-1}\big)}{\sum_{i=1}^{P} w_i}. \tag{6.34}$$

This smoothing process behaves as an interpolation to determine the nearest point, thereby deforming the original curve to an interpolated one. The parameter δ controls

the extent of this deformation. While this strategy prevents us from invoking previously proven lemmas and theorems, we claim that all signals will remain bounded and the system will converge to a neighborhood of the curve that is sufficiently close for the task.

The real distance $D(\mathbf{H})$ was computed using a brute force approach, with a slight modification of the expression derived in Section 5.2.1. To avoid numerical errors, the distance was computed as

$$D(\mathbf{H}) = \sqrt{2\theta^2 + \|\mathbf{u}\|^2 + 0.01^2} - 0.01. \tag{6.35}$$

6.2.3 Controller parameters

The vector field strategy, employed as a reference for adaptive control, requires the specification of appropriate vector field gains. The gains $k_N(\mathbf{H})$ and $k_T(\mathbf{H})$ were defined as follows:

$$k_N(\mathbf{H}) = \tanh\Big(10\Big(\bar{D}(\mathbf{H}) - \gamma\Big)\Big), \tag{6.36}$$

$$k_T(\mathbf{H}) = 0.2\Big(1 - \tanh\Big(\bar{D}(\mathbf{H}) - \gamma\Big)\Big), \tag{6.37}$$

where $\gamma = 0.27$ is an offset determined by evaluating the difference between the smoothed distance $\bar{D}$ and the actual distance D in steady state.

For the adaptive control, the adaptation gains were set as:

$$\mathbf{\Gamma}_o = \frac{1}{30}\operatorname{diag}\Big(\operatorname{abs}(\mathbf{o}_i) + 1 \cdot 10^{-2}\Big), \tag{6.38}$$

$$\mathbf{\Gamma}_r = \frac{1}{3000}\mathbf{I}_{3\times 3}, \tag{6.39}$$

where $\operatorname{diag}(\cdot)$ maps a vector into a diagonal matrix, and $\operatorname{abs}(\cdot)$ represents the element-wise absolute value of a vector. The control gain was defined as

$$\mathbf{K}_D = 3.5\,\operatorname{blkdiag}\Big(20 \cdot 10^3\mathbf{I}_{3\times 3},\, 25 \cdot 10^3\mathbf{I}_{3\times 3}\Big), \tag{6.40}$$

where $\operatorname{blkdiag}(\cdot, \ldots, \cdot)$ constructs a block diagonal matrix from a list of matrices.

6.2.4 Simulation

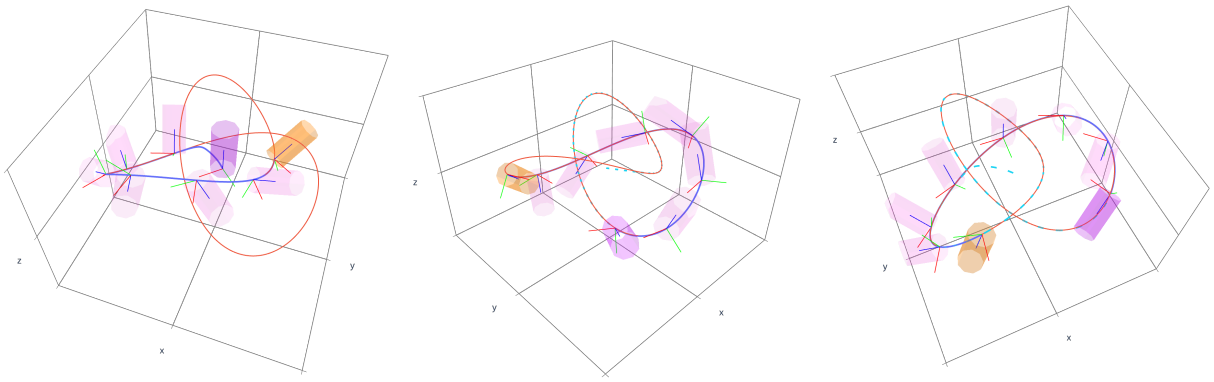
The simulation was conducted for 40 s with a time step of $\Delta t = 1 \cdot 10^{-3}$ s for the numerical integration described in Section 6.2.2. The dead zone strategy (see Section 3.3) was

employed adopting a deadband of $\|\boldsymbol{\zeta}\| \leq 0.01$. The initial conditions were set as follows:

$$\mathbf{p}(0) = \begin{bmatrix} -0.1 & 0 & 0.2 \end{bmatrix}^\top, \tag{6.41}$$

$$\mathbf{R}(0) = \begin{bmatrix} \cos\left(\frac{\pi}{4}\right) & -\sin\left(\frac{\pi}{4}\right) & 0 \\ \sin\left(\frac{\pi}{4}\right) & \cos\left(\frac{\pi}{4}\right) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{6.42}$$

The initial estimates $\hat{\mathbf{o}}_i$ and $\hat{\mathbf{r}}_i$ were randomly initialized following a normal distribution with a mean of zero and standard deviations of 1 and 2, respectively.



(a) $t \in [0, 13]s$ (b) $t \in [13, 26]s$ (c) $t \in [26, 39]s$

Figure 6.4: Trajectory of the manipulated cylinder. The solid blue line depicts the system trajectory, while the solid red line indicates the target curve. The dashed light blue line represents the past trajectory. The initial and final positions are marked by purple and orange cylinders, respectively. Pink cylinders denote intermediate positions, with orientation frames shown by RGB axes.

The trajectory of the manipulated object is shown in Figure 6.4. The figure provides three different perspectives to facilitate visualization of the target curve, depicted in red. The system's trajectory, shown in blue, is divided into three time segments, $t \in [0,\ 13]s$, $t \in [13,\ 26]s$ and $t \in [26,\ 39]s$. In each segment, eight cylinders represent the object's movement during the simulation, along with its orientation frames. As shown, the object rapidly converges to the target curve and follows it closely.

The error vector $\boldsymbol{\zeta}$ rapidly decreased to zero; however, after approximately 0.8 s experienced a slight increase, as illustrated in Figure 6.5. This behavior is expected due to the adaptation process. After approximately 1.2 s, $\boldsymbol{\zeta}$ converged to 0.01, indicating that the agents successfully tracked the desired velocity given by the vector field. The norm of this error exhibited minor oscillations for the remainder of the simulation but remained close to zero. These oscillations can be attributed to the dead-zone strategy and the approximations made during computations.

Examining the behavior of the distance function D in Figure 6.6, it can be observed that the object approached the target curve asymptotically at approximately 1.2 s, consistent

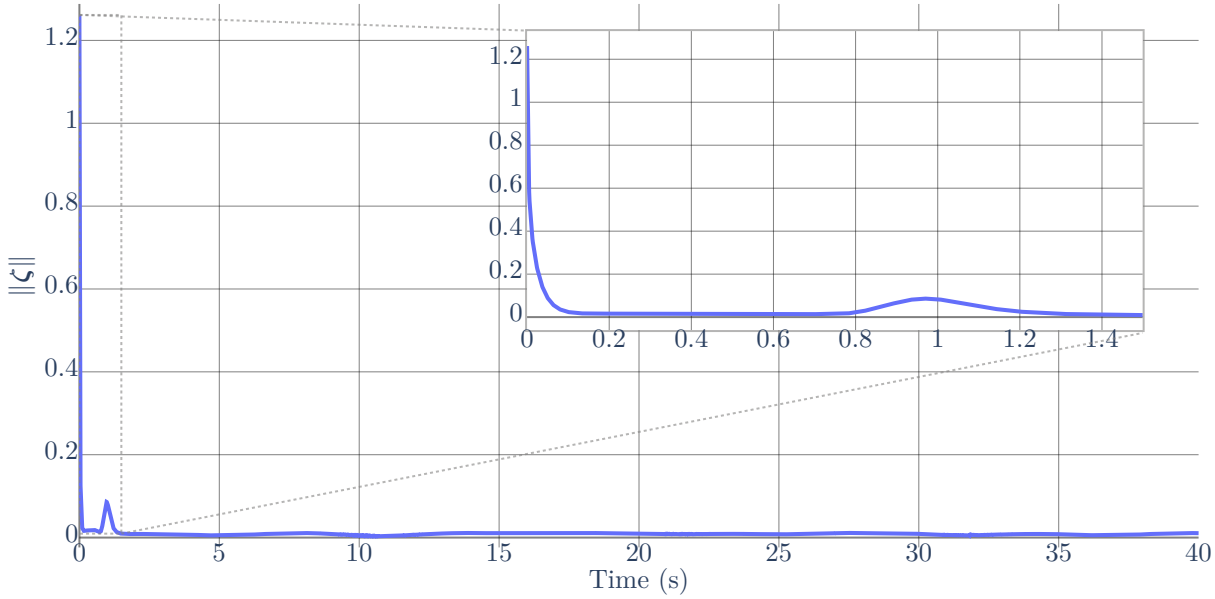


Figure 6.5: Norm of error vector $\boldsymbol{\zeta}$ during the adaptive control simulation over time.

with the behavior of $\boldsymbol{\zeta}$. The final observed value of D was 0.004, corresponding to a position error of 0.6 cm and an orientation error of 0.32°.

The average forces and torques applied by each agent are shown in Table 6.2 and Table 6.3, respectively. The average force and standard deviation were similar for all agents, with values of approximately (88 ± 111) N. The maximum and minimum forces were also comparable, approximately 7411 N and 31 N, respectively. In contrast, the torques applied by each agent exhibited greater variation, with the highest average observed for agent 5 at (392.079 ± 573.948) N m and the lowest for agent 3 at (109.546 ± 138.170) N m. The maximum torque, 46 835.582 N m, was applied by agent 5, while the minimum torque, 3.731 N m, was applied by agent 2. These high values were expected, given the large mass and dimensions of the cylinder and the complexity of the manipulation task, which was performed within a minimal time frame. The maximum linear speed experienced by the object was 1.09 m/s, and the maximum angular speed was 1.04 rad/s, both of which are demanding values.

Table 6.2: Statistics of the forces applied by each agent

Agent	Average Force	Min. Force	Max. Force
1	(88.093 ± 111.561) N	30.833 N	7411.424 N
2	(88.159 ± 111.621) N	31.208 N	7411.544 N
3	(88.562 ± 111.718) N	31.150 N	7411.903 N
4	(88.015 ± 111.579) N	31.151 N	7410.960 N
5	(88.395 ± 111.689) N	31.204 N	7411.755 N
6	(87.703 ± 111.502) N	31.017 N	7411.278 N

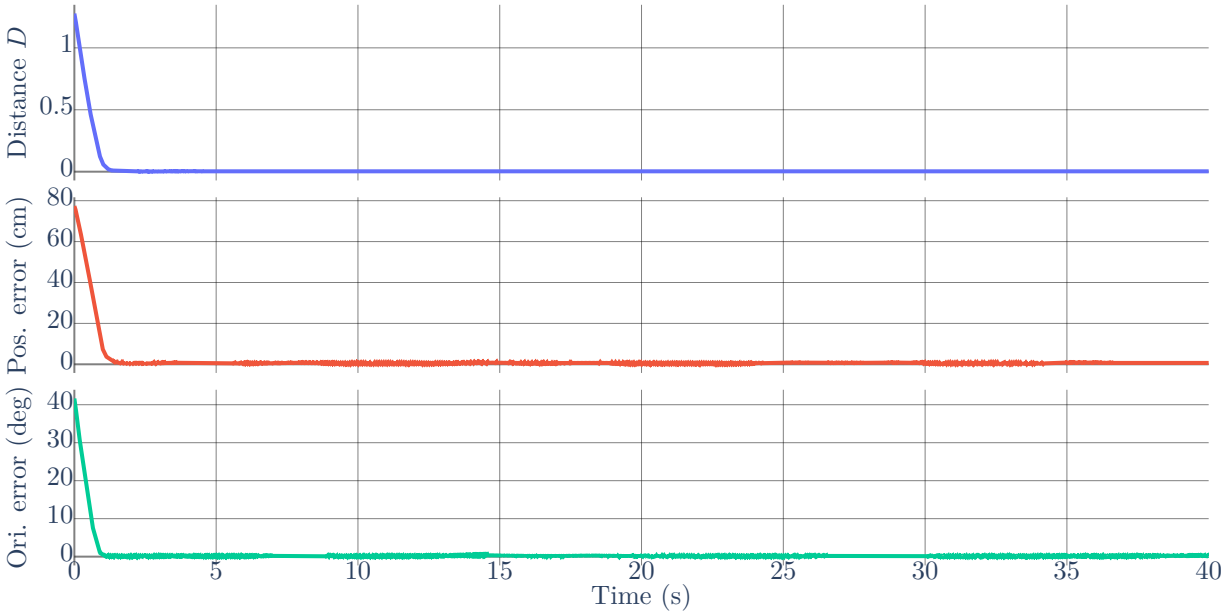


Figure 6.6: Distance D during adaptive control simulation, alongside the position error in centimeters and orientation error in degrees.

Table 6.3: Statistics of the torques applied by each agent

Agent	Average Torque	Min. Torque	Max. Torque
1	(148.854 ± 181.826) N m	54.256 N m	8737.108 N m
2	(114.282 ± 139.403) N m	3.731 N m	4838.576 N m
3	(109.546 ± 138.178) N m	6.095 N m	8539.131 N m
4	(253.599 ± 288.916) N m	79.889 N m	15 571.921 N m
5	(392.079 ± 573.948) N m	48.958 N m	46 835.582 N m
6	(145.025 ± 207.934) N m	42.274 N m	12 251.425 N m

7

Conclusion and Future Work

In this work, we presented a strategy for generating vector fields on connected matrix Lie groups that enforce convergence to and circulation around curves defined within the same group. This approach generalizes our previous work, which was limited to parametric curve representations in Euclidean space. To achieve this broader applicability, we examined key properties of distance functions that maintain the desired features of the vector field approach. The three essential properties – left-invariance, chainability, and local linearity – not only support this generalization to matrix Lie groups but also permit flexibility in the choice of metric for Euclidean space applications. Notably, since we use an isomorphism between the Euclidean space and the Lie algebra, the proposed strategy also employs a non-redundant control input, meaning that it acts on the group's degrees of freedom directly, eliminating the need to specify elements of the tangent space as inputs.

We validated our strategy within the special Euclidean group SE(3) under a kinematic control approach, and provided an efficient algorithm for computing the vector field in this context. The kinematic control results validate the convergence and circulation proofs for the vector field, and demonstrate the ability to generate complex motions in 3D space. We also demonstrated the generality of the strategy by applying kinematic control on the group $\mathrm{SO}^+(3,1)$, showcasing its capability to control Lorentz transformation matrices.

Additionally, we showed that the vector field can serve as a high-level controller, acting as a velocity reference for a lower-level dynamic controller. In particular, we demonstrated that using the vector field guidance in an adaptive control scenario enables an autonomous system to track the target curve with high precision, even in the presence of unknown

parameters. The dynamic control scenario also involves the use of the group ISE(3) of independent rigid motions, highlighting the generality of the proposed strategy.

For these simulations, we employed two methods for computing the tangent component – one numerical and one analytical – and showed that both methods yield satisfactory results. Although not used in the simulations, we also introduced a method for explicitly computing the normal component, which could improve the tracking performance in future work, as the approximation approach used here may introduce numerical errors, making the explicit normal component a valuable tool.

Future work will focus on extending this strategy to time-varying curves and investigating simpler distance functions with the necessary properties, as outlined in Theorem 4.28. Given that the computation of the distance is the primary bottleneck in the proposed strategy, exploring optimization algorithms for minimizing functions on matrix Lie groups will be an area of interest. Additionally, validating this strategy on a real omnidirectional UAV is a promising avenue, as is the consideration of self-intersecting curves, which would enable more complex motions, such as lemniscate-like trajectories. The use of smoothed distances will also be examined to determine the conditions under which they ensure the desired properties. Addressing nonholonomic constraints is also of interest, as it would allow for the control of UAVs that do not have omnidirectional capabilities. Lastly, the adaptive control strategy could be further refined by incorporating agent uncertainties and dynamics.

A

Properties of the L Operator

In this chapter, we explore additional properties of the L operator, which is defined in Definition 4.3. First, we demonstrate that the L operator shares all the properties of derivatives. Let G be an m-dimensional matrix Lie group and $\mathbf{X} \in G$ be a point in the group. Let $\mathcal{S} : \mathbb{R}^m \to \mathfrak{g}$ be an isomorphism between m-dimensional vectors and the Lie algebra $\mathfrak{g}$ of G. For some function $f : G \to \mathbb{R}$, the L operator can be expressed as

$$\mathrm{L}[f](\mathbf{X})\boldsymbol{\zeta} = \frac{d}{d\varepsilon}\left(f\left(\exp\left(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\right)\mathbf{X}\right)\right)\Big|_{\varepsilon=0} \quad \forall\, \boldsymbol{\zeta} \in \mathbb{R}^m. \tag{A.1}$$

A.1 Linearity

The L operator is linear. Specifically, for any functions $f, g : G \to \mathbb{R}$ and scalars $a, b \in \mathbb{R}$, we have

$$\begin{aligned}
\mathrm{L}[af + bg](\mathbf{X})\boldsymbol{\zeta} &= \frac{d}{d\varepsilon}\left(af\left(\exp\left(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\right)\mathbf{X}\right) + bg\left(\exp\left(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\right)\mathbf{X}\right)\right)\Big|_{\varepsilon=0} \\
&= \frac{d}{d\varepsilon}\left(af\left(\exp\left(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\right)\mathbf{X}\right)\right)\Big|_{\varepsilon=0} + \frac{d}{d\varepsilon}\left(bg\left(\exp\left(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\right)\mathbf{X}\right)\right)\Big|_{\varepsilon=0} \\
&= a\,\mathrm{L}[f](\mathbf{X})\boldsymbol{\zeta} + b\,\mathrm{L}[g](\mathbf{X})\boldsymbol{\zeta} \\
&= \left(a\,\mathrm{L}[f](\mathbf{X}) + b\,\mathrm{L}[g](\mathbf{X})\right)\boldsymbol{\zeta},
\end{aligned} \tag{A.2}$$

which shows that $\mathrm{L}[af + bg] = a\,\mathrm{L}[f] + b\,\mathrm{L}[g]$, which confirms that the L operator is linear.

A.2 Product and quotient rule

The L operator satisfies both the product and quotient rules. Let $f, g : G \to \mathbb{R}$ be scalar functions. By definition, we have

$$\begin{aligned}
\mathrm{L}[fg](\mathbf{X})\boldsymbol{\zeta} =& \frac{d}{d\varepsilon}\left(f\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\right)\Big|_{\varepsilon=0} \\
=& \frac{d}{d\varepsilon}\left(f\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\right)\Big|_{\varepsilon=0} g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\Big|_{\varepsilon=0} \\
&+ f\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\Big|_{\varepsilon=0} \frac{d}{d\varepsilon}\left(g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\right)\Big|_{\varepsilon=0} \\
=& \frac{d}{d\varepsilon}\left(f\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\right)\Big|_{\varepsilon=0} g(\mathbf{X}) + f(\mathbf{X})\left(g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\right)\Big|_{\varepsilon=0} \\
=& \mathrm{L}[f](\mathbf{X})\boldsymbol{\zeta} g(\mathbf{X}) + f(\mathbf{X})\,\mathrm{L}[g](\mathbf{X})\boldsymbol{\zeta} \\
=& \Big(\mathrm{L}[f](\mathbf{X})g(\mathbf{X}) + f(\mathbf{X})\,\mathrm{L}[g](\mathbf{X})\Big)\boldsymbol{\zeta},
\end{aligned} \tag{A.3}$$

since f and g are scalar functions. This shows that $\mathrm{L}[fg] = \mathrm{L}[f]g + f\,\mathrm{L}[g]$. Next, we demonstrate the quotient rule. For that, it suffices to show that $\mathrm{L}[1/g] = -\,\mathrm{L}[g]/g^2$. In this case, we have

$$\begin{aligned}
\mathrm{L}\left[\frac{1}{g}\right](\mathbf{X})\boldsymbol{\zeta} &= \frac{d}{d\varepsilon}\left(\frac{1}{g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)}\right)\Big|_{\varepsilon=0} \\
&= -\,\frac{\dfrac{d}{d\varepsilon}\left(g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\right)\Big|_{\varepsilon=0}}{\left(g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)g\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{X}\Big)\right)\Big|_{\varepsilon=0}} \\
&= -\,\frac{\mathrm{L}[g]}{g(\mathbf{X})^2}\boldsymbol{\zeta},
\end{aligned} \tag{A.4}$$

which implies

$$\mathrm{L}\left[\frac{f}{g}\right] = \mathrm{L}[f]\frac{1}{g} - f\frac{\mathrm{L}[g]}{g^2} = \frac{\mathrm{L}[f]g - f\,\mathrm{L}[g]}{g^2}. \tag{A.5}$$

Thus, the L operator satisfies the quotient rule.

A.3 Chain rules

A chain rule has already been shown in Lemma 4.4, which applies to expressions of the form $\frac{d}{d\sigma} f\Big(\mathbf{X}(\sigma)\Big)$. In this section, we present two additional chain rules. The first pertains to the L operator applied to function composition, while the second concerns the partial

$\mathrm{L}_{\mathbf{V}}$ operator for left-invariant functions.

Let $f : \mathbb{R} \to \mathbb{R}$ and $g : G \to \mathbb{R}$ be functions, and $\mathbf{X}$ be a point in G. The chain rule for the L operator of function composition is given by:

$$\mathrm{L}\Big[f\Big(g(\mathbf{X})\Big)\Big](\mathbf{X})\boldsymbol{\zeta} = \frac{d}{d\varepsilon}\left(f\left(g\Big(\exp\Big(\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{X}\Big)\right)\right)\bigg|_{\varepsilon=0} \tag{A.6}$$

$$= \frac{df}{dg}\Big(g(\mathbf{X})\Big)\left(\frac{d}{d\varepsilon}g\Big(\exp\Big(\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{X}\Big)\right)\bigg|_{\varepsilon=0} \tag{A.7}$$

$$= \frac{df}{dg}\Big(g(\mathbf{X})\Big)\,\mathrm{L}[g](\mathbf{X})\boldsymbol{\zeta}, \tag{A.8}$$

which implies

$$\mathrm{L}\Big[f\Big(g(\mathbf{X})\Big)\Big](\mathbf{X}) = \frac{df}{dg}\Big(g(\mathbf{X})\Big)\,\mathrm{L}[g](\mathbf{X}). \tag{A.9}$$

We now derive a chain rule specifically related to the $\mathrm{L}_{\mathbf{V}}$ operator for left-invariant functions. Let $\mathbf{X}, \mathbf{Y} \in G$, and $f : G \times G \to \mathbb{R}$ be a left-invariant function. Define $\mathbf{Z} = \mathbf{Y}^{-1}\mathbf{X}$, so that $f(\mathbf{X}, \mathbf{Y}) = g(\mathbf{Z})$ for some $g : G \to \mathbb{R}$. Our goal is to determine the L operator of g with respect to $\mathbf{X}$.

First, consider a variation of $\mathbf{X}$ using the right-hand side of (A.1) for g:

$$\frac{d}{d\varepsilon}\left(g\Big(\mathbf{Y}^{-1}\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{X}\Big)\right)\bigg|_{\varepsilon=0} = \frac{d}{d\varepsilon}\left(g\Big(\mathbf{Y}^{-1}\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{Y}\mathbf{Y}^{-1}\mathbf{X}\Big)\right)\bigg|_{\varepsilon=0} \tag{A.10}$$

$$= \frac{d}{d\varepsilon}\left(g\Big(\exp\Big(\varepsilon\mathbf{Y}^{-1}\,\mathcal{S}(\boldsymbol{\zeta})\,\mathbf{Y}\Big)\mathbf{Y}^{-1}\mathbf{X}\Big)\right)\bigg|_{\varepsilon=0} \tag{A.11}$$

$$= \frac{d}{d\varepsilon}\left(g\Big(\exp\Big(\varepsilon\mathcal{S}'(\boldsymbol{\zeta})\Big)\mathbf{Y}^{-1}\mathbf{X}\Big)\right)\bigg|_{\varepsilon=0}, \tag{A.12}$$

where $\mathcal{S}'(\boldsymbol{\zeta}) = \mathbf{Y}^{-1}\,\mathcal{S}(\boldsymbol{\zeta})\,\mathbf{Y}$. Since $\mathbf{Y}^{-1}\,\mathcal{S}(\boldsymbol{\zeta})\,\mathbf{Y}$ lies in $\mathfrak{g}$, there exists a vector $\boldsymbol{\zeta}'$ such that $\mathcal{S}(\boldsymbol{\zeta}') = \mathbf{Y}^{-1}\,\mathcal{S}(\boldsymbol{\zeta})\,\mathbf{Y}$. This implies that $\boldsymbol{\zeta}'$ is a function of $\boldsymbol{\zeta}$ and $\mathbf{Y}$, i.e., $\boldsymbol{\zeta}' \triangleq \boldsymbol{\zeta}'(\boldsymbol{\zeta}, \mathbf{Y})$. Therefore, we can write

$$\frac{d}{d\varepsilon}\left(g\Big(\mathbf{Y}^{-1}\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{X}\Big)\right)\bigg|_{\varepsilon=0} = \frac{d}{d\varepsilon}\left(g\Big(\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta}')\Big)\mathbf{Y}^{-1}\mathbf{X}\Big)\right)\bigg|_{\varepsilon=0} \tag{A.13}$$

$$= \mathrm{L}[g](\mathbf{Z})\boldsymbol{\zeta}'(\boldsymbol{\zeta}, \mathbf{Y}), \tag{A.14}$$

by definition. Now, since f is left-invariant, we can express

$$\mathrm{L}_{\mathbf{V}}[f](\mathbf{X}, \mathbf{Y})\boldsymbol{\zeta} = \mathrm{L}_{\mathbf{V}}[f](\mathbf{Y}^{-1}\mathbf{X}, \mathbf{I})\boldsymbol{\zeta} = \frac{d}{d\varepsilon}\left(f\Big(\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{Y}^{-1}\mathbf{X}, \mathbf{I}\Big)\right)\bigg|_{\varepsilon=0}, \tag{A.15}$$

which leads to

$$\mathrm{L}_{\mathbf{V}}[f](\mathbf{X},\mathbf{Y})\boldsymbol{\zeta} = \mathrm{L}[g](\mathbf{Z})\boldsymbol{\zeta}'(\boldsymbol{\zeta},\mathbf{Y}), \tag{A.16}$$

which is the expression for the chain rule in the context of left-invariant functions. Clearly, $\boldsymbol{\zeta}'$ is expressed as:

$$\begin{aligned}
\boldsymbol{\zeta}'(\boldsymbol{\zeta},\mathbf{Y}) &= \mathcal{S}^{-1}\Big(\mathbf{Y}^{-1}\,\mathcal{S}(\boldsymbol{\zeta})\,\mathbf{Y}\Big) && \text{(A.17)}\\
&= \mathcal{S}^{-1}\left(\sum_{i=1}^{m}\mathbf{Y}^{-1}\,\mathcal{S}(\mathbf{e}_i)\,\mathbf{Y}\zeta_i\right) && \text{(A.18)}\\
&= \sum_{i=1}^{m}\mathcal{S}^{-1}\Big(\mathbf{Y}^{-1}\,\mathcal{S}(\mathbf{e}_i)\,\mathbf{Y}\Big)\,\zeta_i && \text{(A.19)}\\
&= \Big[\mathcal{S}^{-1}(\mathbf{Y}^{-1}\,\mathcal{S}(\mathbf{e}_1)\,\mathbf{Y}) \quad \cdots \quad \mathcal{S}^{-1}(\mathbf{Y}^{-1}\,\mathcal{S}(\mathbf{e}_m)\,\mathbf{Y})\Big]\,\boldsymbol{\zeta} && \text{(A.20)}\\
&= \mathcal{Z}(\mathbf{Y})\boldsymbol{\zeta}, && \text{(A.21)}
\end{aligned}$$

where the linearity properties of $\mathcal{S}$ and $\mathcal{S}^{-1}$ have been used. This implies that the chain rule for the $\mathrm{L}_{\mathbf{V}}$ operator can be expressed as:

$$\mathrm{L}_{\mathbf{V}}[f](\mathbf{X},\mathbf{Y})\boldsymbol{\zeta} = \mathrm{L}[g](\mathbf{Z})\mathcal{Z}(\mathbf{Y})\boldsymbol{\zeta}. \tag{A.22}$$

A.3.1 The case of SE(3)

In the case of SE(3), we can show that $\boldsymbol{\zeta}'(\boldsymbol{\zeta},\mathbf{Y})$ is given by a simple matrix multiplication. Let $\mathcal{S}:\mathbb{R}^6 \to \mathfrak{se}(3)$ be the canonical isomorphism between $\mathbb{R}^6$ and $\mathfrak{se}(3)$, so that $\mathbf{Y}$ can be written as

$$\mathbf{Y} = \begin{bmatrix}\mathbf{R}_y & \mathbf{p}_y\\ \mathbf{0} & 1\end{bmatrix},\ \mathbf{R}_y \in \mathrm{SO}(3), \mathbf{p}_y \in \mathbb{R}^3. \tag{A.23}$$

Next, define $\widehat{\mathcal{S}}:\mathbb{R}^3 \to \mathfrak{so}(3)$ as the canonical isomorphism between $\mathbb{R}^3$ and $\mathfrak{so}(3)$, and let $\boldsymbol{\zeta} = [\mathbf{v}^\top\ \boldsymbol{\omega}^\top]^\top$, where $\mathbf{v},\boldsymbol{\omega} \in \mathbb{R}^3$. Then, we have the following expression:

$$\begin{aligned}
\mathbf{Y}^{-1}\,\mathcal{S}(\boldsymbol{\zeta})\,\mathbf{Y} &= \begin{bmatrix}\mathbf{R}_y^\top & -\mathbf{R}_y^\top\mathbf{p}_y\\ \mathbf{0} & 1\end{bmatrix}\begin{bmatrix}\widehat{\mathcal{S}}(\boldsymbol{\omega}) & \mathbf{v}\\ \mathbf{0} & 0\end{bmatrix}\begin{bmatrix}\mathbf{R}_y & \mathbf{p}_y\\ \mathbf{0} & 1\end{bmatrix}\\
&= \begin{bmatrix}\mathbf{R}_y^\top\widehat{\mathcal{S}}(\boldsymbol{\omega})\mathbf{R}_y & \mathbf{R}_y^\top\widehat{\mathcal{S}}(\boldsymbol{\omega})\mathbf{p}_y + \mathbf{R}_y^\top\mathbf{v}\\ \mathbf{0} & 0\end{bmatrix} && \text{(A.24)}\\
&= \mathcal{S}(\boldsymbol{\zeta}')\,.
\end{aligned}$$

Using the properties of $\widehat{\mathcal{S}}$, we have $\mathbf{R}_y^\top \widehat{\mathcal{S}}(\boldsymbol{\omega})\mathbf{R}_y = \widehat{S}(\mathbf{R}_y^\top \boldsymbol{\omega})$, thus we can express $\boldsymbol{\zeta}'$ as

$$\boldsymbol{\zeta}'(\boldsymbol{\zeta}, \mathbf{Y}) = \begin{bmatrix} \mathbf{R}_y^\top \widehat{\mathcal{S}}(\boldsymbol{\omega})\mathbf{p}_y + \mathbf{R}_y^\top \mathbf{v} \\ \mathbf{R}_y^\top \boldsymbol{\omega} \end{bmatrix}. \tag{A.25}$$

Note that $\widehat{\mathcal{S}}(\boldsymbol{\omega})\mathbf{p}_y = -\widehat{\mathcal{S}}(\mathbf{p}_y)\boldsymbol{\omega}$, so the following holds

$$\boldsymbol{\zeta}'(\boldsymbol{\zeta}, \mathbf{Y}) = \begin{bmatrix} \mathbf{R}_y^\top & -\mathbf{R}_y^\top \widehat{\mathcal{S}}(\mathbf{p}_y) \\ \mathbf{0} & \mathbf{R}_y^\top \end{bmatrix} \begin{bmatrix} \mathbf{v} \\ \boldsymbol{\omega} \end{bmatrix} = \mathcal{Z}(\mathbf{Y})\boldsymbol{\zeta}. \tag{A.26}$$

Thus, achieving the expression (A.22):

$$\mathrm{L}_{\mathbf{V}}[f](\mathbf{X}, \mathbf{Y}) = \mathrm{L}[g](\mathbf{Z})\mathcal{Z}(\mathbf{Y}). \tag{A.27}$$

A.4 Connection to the Lie derivative

Lie derivatives are directional derivatives adapted to the structure of a manifold. While standard directional derivatives rely on the properties of Euclidean space and are thus only locally valid for manifolds, Lie derivatives are defined in a way that ensures their validity globally on the manifold. Although the L operator pertains to functions, we begin by recalling the definition of the Lie derivative for vector fields for the sake of completeness.

In the context of vector fields, the Lie derivative provides a measure of the change of one vector field along the flow generated by another. Let G be a Lie group, and let $V \in T_{\mathbf{X}}G$ and $W \in T_{\mathbf{Y}}G$ be vector fields on G. Since these vector fields belong to different tangent spaces, it is necessary to transport points using the flow. Let $\rho : \mathbb{R} \times G$ be the flow of V. Following the integral curve of ρ with initial condition $\mathbf{X}$, we can evaluate $W(\rho(\sigma, \mathbf{X}))$, which lies in the tangent space of G at $\rho(\sigma, \mathbf{X})$. To compare both vector fields, it is necessary to transport $W\big(\rho(\sigma, \mathbf{X})\big)$ back to $T_{\mathbf{X}}G$. The Lie derivative of W with respect to V, denoted by $\mathscr{L}_V W$, at $\mathbf{X}$ is defined as follows (Gallier and Quaintance, 2020, p. 316):

$$(\mathscr{L}_V W)_{\mathbf{X}} = \frac{d}{d\sigma}\Big(d(\rho(-\sigma, \mathbf{X}))_{\rho(\sigma,\mathbf{X})}\Big)\Big(W(\rho(\sigma, \mathbf{X}))\Big)\Big|_{\sigma=0}. \tag{A.28}$$

This operation is often expressed more concisely using the *pull-back* of W by ρ, denoted $\rho(\sigma, \mathbf{X})^* W$, leading to the simplified expression (Gallier and Quaintance, 2020, p. 317)

$$(\mathscr{L}_V W)_{\mathbf{X}} = \frac{d}{d\sigma}\Big(\rho(\sigma, \mathbf{X})^* W\Big)\Big|_{\sigma=0}. \tag{A.29}$$

The Lie derivative of a function $f : G \to \mathbb{R}$ at a point $\mathbf{X} \in G$ with respect to a vector

field V with flow ρ is expressed as (Lang, 2012, p. 121):

$$(\mathscr{L}_V f)_{\mathbf{X}} = \lim_{\sigma \to 0} \frac{f\Big(\rho(\sigma, \mathbf{X})\Big) - f(\mathbf{X})}{\sigma} = \frac{d}{d\sigma}\Big(f\Big(\rho(\sigma, \mathbf{X})\Big)\Big)\Big|_{\sigma=0}. \tag{A.30}$$

For a matrix Lie group G, let $\mathbf{G}(\sigma)$ be a curve in G with tangent vector V at $\mathbf{G}(\sigma)$. The curve derivative is then given by

$$\frac{d}{d\sigma}\mathbf{G}(\sigma) = V(\mathbf{G}(\sigma)), \tag{A.31}$$

which, as shown in Lemma 3.36, can be expressed as

$$\frac{d}{d\sigma}\mathbf{G}(\sigma) = \mathcal{S}\Big(\boldsymbol{\xi}(\sigma)\Big)\mathbf{G}(\sigma), \tag{A.32}$$

for some isomorphism $\mathcal{S} : \mathbb{R}^m \to \mathfrak{g}$ and function $\boldsymbol{\xi} : \mathbb{R} \to \mathbb{R}^m$.

Let $\mathbf{G} = \mathbf{G}(\sigma)$ for some fixed σ. A local flow $\rho(\varepsilon, \mathbf{G})$ of V can be expressed as $\exp\Big(\varepsilon\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{G}$ for small ε and some vector $\boldsymbol{\zeta} \in \mathbb{R}^m$. The Lie derivative of f with respect to V at $\mathbf{G}$ is therefore

$$(\mathscr{L}_V f)_{\mathbf{G}} = \lim_{\varepsilon \to 0} \frac{f\Big(\exp\Big(\varepsilon\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{G}\Big) - f(\mathbf{G})}{\varepsilon}, \tag{A.33}$$

which, by definition, is exactly equal to $\mathrm{L}[f](\mathbf{G})\boldsymbol{\zeta}$, demonstrating that the L operator represents a component of the Lie derivative.

B

In-depth Derivations for the EE-Distance in SE(3)

In this chapter, we provide all the derivations for the EE-distance in SE(3) and its properties. We begin by deriving Rodrigues' formula for rotation matrices in SO(3) and showing some important properties of rotation matrices. This serves as a self-contained explanation. Although the EE-distance is not directly treated here, we include only the key properties used to derive it. The reader can easily make the necessary connections to understand the derivations in Section 4.8.1.

B.1 Deriving Rodrigues' rotation formula

In this section, we derive Rodrigues' formula using a Lie group approach. This derivation is useful for obtaining important properties of rotation matrices. Let $\mathbf{R} \in \mathrm{SO}(3)$ be a rotation matrix. Since SO(3) is an exponential group, it follows that $\mathbf{R}$ is the exponential of some Lie algebra element $\mathcal{S}(\boldsymbol{\omega}) \in \mathfrak{so}(3)$, i.e.

$$\mathbf{R} = \exp\Big(\mathcal{S}(\boldsymbol{\omega})\Big). \tag{B.1}$$

We begin by exploring powers of the Lie algebra element. Let $\boldsymbol{\omega} = [\omega_1 \ \omega_2 \ \omega_3]^\top$, so we

can express the squared Lie algebra element as

$$\mathcal{S}(\boldsymbol{\omega})^2 = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}^2 = \begin{bmatrix} -\omega_3^2 - \omega_2^2 & \omega_2\omega_1 & \omega_3\omega_1 \\ \omega_1\omega_2 & -\omega_3^2 - \omega_1^2 & \omega_3\omega_2 \\ \omega_1\omega_3 & \omega_2\omega_3 & -\omega_2^2 - \omega_1^2 \end{bmatrix}. \tag{B.2}$$

Next, let $\theta = \sqrt{\omega_1^2 + \omega_2^2 + \omega_3^2}$, and consider the matrix

$$\mathbf{B} = \begin{bmatrix} \omega_1^2 & \omega_2\omega_1 & \omega_3\omega_1 \\ \omega_1\omega_2 & \omega_2^2 & \omega_3\omega_2 \\ \omega_1\omega_3 & \omega_2\omega_3 & \omega_3^2 \end{bmatrix}. \tag{B.3}$$

It is now clear that we can express $\mathcal{S}(\boldsymbol{\omega})^2$ as

$$\mathcal{S}(\boldsymbol{\omega})^2 = -\theta^2\mathbf{I} + \mathbf{B}. \tag{B.4}$$

From this, we can also express

$$\mathcal{S}(\boldsymbol{\omega})^3 = \mathcal{S}(\boldsymbol{\omega})^2\,\mathcal{S}(\boldsymbol{\omega}) = -\theta^2\,\mathcal{S}(\boldsymbol{\omega}) + \mathbf{B}\,\mathcal{S}(\boldsymbol{\omega})\,. \tag{B.5}$$

Note that

$$\mathcal{S}(\boldsymbol{\omega})\,\mathbf{B} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \begin{bmatrix} \omega_1^2 & \omega_2\omega_1 & \omega_3\omega_1 \\ \omega_1\omega_2 & \omega_2^2 & \omega_3\omega_2 \\ \omega_1\omega_3 & \omega_2\omega_3 & \omega_3^2 \end{bmatrix} \tag{B.6}$$

$$= \begin{bmatrix} -\omega_3\omega_1\omega_2 + \omega_2\omega_1\omega_3 & -\omega_3\omega_2^2 + \omega_2^2\omega_3 & -\omega_3^2\omega_2 + \omega_2\omega_3^2 \\ \omega_3\omega_1^2 - \omega_1^2\omega_3 & \omega_3\omega_2\omega_1 - \omega_1\omega_2\omega_3 & \omega_3^2\omega_1 - \omega_1\omega_3^2 \\ -\omega_2\omega_1^2 + \omega_1^2\omega_2 & -\omega_2\omega_1 + \omega_1\omega_2^2 & -\omega_2\omega_3\omega_1 + \omega_1\omega_2\omega_3 \end{bmatrix} \tag{B.7}$$

$$= \mathbf{0}. \tag{B.8}$$

This property also holds for $\mathbf{B}\,\mathcal{S}(\boldsymbol{\omega})$, since $(\mathbf{B}\,\mathcal{S}(\boldsymbol{\omega}))^\top = -\,\mathcal{S}(\boldsymbol{\omega})\,\mathbf{B}$. Finally, we can express

$$\mathcal{S}(\boldsymbol{\omega})^3 = -\theta^2\,\mathcal{S}(\boldsymbol{\omega})\,, \tag{B.9}$$

$$\mathcal{S}(\boldsymbol{\omega})^4 = \mathcal{S}(\boldsymbol{\omega})^3\,\mathcal{S}(\boldsymbol{\omega}) = \theta^4\mathbf{I} - \theta^2\mathbf{B}. \tag{B.10}$$

Thus, by induction, we can express that for any natural number $k > 0$,

$$\mathcal{S}(\boldsymbol{\omega})^{4k+1} = \theta^{4k}\mathcal{S}(\boldsymbol{\omega}), \tag{B.11}$$

$$\mathcal{S}(\boldsymbol{\omega})^{4k+2} = -\theta^{4k+2}\mathbf{I} + \theta^{4k}\mathbf{B}, \tag{B.12}$$

$$\mathcal{S}(\boldsymbol{\omega})^{4k+3} = -\theta^{4k+2}\mathcal{S}(\boldsymbol{\omega}), \tag{B.13}$$

$$\mathcal{S}(\boldsymbol{\omega})^{4k+4} = \theta^{4k+4}\mathbf{I} - \theta^{4k+2}\mathbf{B}. \tag{B.14}$$

Using the power series of the matrix exponential, the expression for the rotation matrix $\mathbf{R}$ is

$$\begin{aligned}
\exp\Big(\mathcal{S}(\boldsymbol{\omega})\Big) &= \sum_{k=0}^{\infty} \frac{\mathcal{S}(\boldsymbol{\omega})^k}{k!} = \mathbf{I} + \mathcal{S}(\boldsymbol{\omega}) + \frac{\mathcal{S}(\boldsymbol{\omega})^2}{2!} + \frac{\mathcal{S}(\boldsymbol{\omega})^3}{3!} + \frac{\mathcal{S}(\boldsymbol{\omega})^4}{4!} + \dots \\
&= \mathbf{I} + \mathcal{S}(\boldsymbol{\omega}) + \frac{-\theta^2\mathbf{I} + \mathbf{B}}{2!} - \frac{\theta^2\mathcal{S}(\boldsymbol{\omega})}{3!} + \frac{\theta^4\mathbf{I} - \theta^2\mathbf{B}}{4!} + \frac{\theta^4\mathcal{S}(\boldsymbol{\omega})}{5!} + \frac{-\theta^6\mathbf{I} + \theta^4\mathbf{B}}{6!} + \dots \\
&= \left(1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} + \dots\right)\mathbf{I} + \left(1 - \frac{\theta^2}{3!} + \frac{\theta^4}{5!} + \dots\right)\mathcal{S}(\boldsymbol{\omega}) + \left(\frac{1}{2!} - \frac{\theta^2}{4!} + \frac{\theta^4}{6!} + \dots\right)\mathbf{B}.
\end{aligned} \tag{B.15}$$

Manipulating this expression and utilizing the power series for sine and cosine functions, we obtain the rotation matrix as

$$\mathbf{R} = \left(\sum_{i=0}^{\infty}(-1)^k\frac{\theta^{2k}}{(2k)!}\right)\mathbf{I} + \frac{1}{\theta}\left(\sum_{i=0}^{\infty}(-1)^k\frac{\theta^{2k+1}}{(2k+1)!}\right)\mathcal{S}(\boldsymbol{\omega}) + \frac{1}{\theta^2}\left(1 - \sum_{i=0}^{\infty}(-1)^k\frac{\theta^{2k}}{(2k)!}\right)\mathbf{B} \tag{B.16}$$

$$= \cos(\theta)\mathbf{I} + \frac{\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega}) + \frac{1-\cos\theta}{\theta^2}\mathbf{B}. \tag{B.17}$$

Finally, using the fact that $\mathcal{S}(\boldsymbol{\omega})^2 = -\theta^2\mathbf{I} + \mathbf{B}$, we arrive at the most common form of Rodrigues' formula:

$$\mathbf{R} = \mathbf{I} + \frac{\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega}) + \frac{1-\cos\theta}{\theta^2}\mathcal{S}(\boldsymbol{\omega})^2. \tag{B.18}$$

B.2 Rotation matrix properties

The Rodrigues' formula allows us to derive important properties of rotation matrices.

B.2.1 Cosine of the angle

For instance, we can show that the trace of a rotation matrix is equal to $1 + 2\cos\theta$. This can be demonstrated by taking the trace of the Rodrigues formula:

$$\begin{aligned}
\operatorname{tr}(\mathbf{R}) &= \operatorname{tr}\left(\cos(\theta)\mathbf{I} + \frac{\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega}) + \frac{1-\cos\theta}{\theta^2}\mathbf{B}\right) && \text{(B.19)}\\
&= 3\cos\theta + \frac{1-\cos\theta}{\theta^2}\operatorname{tr}(\mathbf{B}) && \text{(B.20)}\\
&= 3\cos\theta + 1 - \cos\theta && \text{(B.21)}\\
&= 1 + 2\cos\theta. && \text{(B.22)}
\end{aligned}$$

B.2.2 Sine of the angle

It is also possible to derive an expression for $\sin\theta$ by computing $\|\mathbf{R} - \mathbf{R}^\top\|_F$. First, we compute the subtraction using the Rodrigues' formula:

$$\begin{aligned}
\mathbf{R} - \mathbf{R}^\top &= \cos(\theta)\mathbf{I} + \frac{\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega}) + \frac{1-\cos\theta}{\theta^2}\mathbf{B} - \cos(\theta)\mathbf{I} + \frac{\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega}) - \frac{1-\cos\theta}{\theta^2}\mathbf{B}\\
&= \frac{2\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega})\,. && \text{(B.23)}
\end{aligned}$$

Then, we compute the Frobenius norm of the subtraction:

$$\begin{aligned}
\|\mathbf{R} - \mathbf{R}^\top\|_F &= \sqrt{\operatorname{tr}\left(\left(\frac{2\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega})\right)^\top\left(\frac{2\sin\theta}{\theta}\mathcal{S}(\boldsymbol{\omega})\right)\right)} && \text{(B.24)}\\
&= \sqrt{\operatorname{tr}\left(-\frac{4(\sin\theta)^2}{\theta^2}\Big(-\theta^2\mathbf{I} + \mathbf{B}\Big)\right)} && \text{(B.25)}\\
&= \sqrt{12(\sin\theta)^2 - 4(\sin\theta)^2} \implies \sin\theta = \frac{1}{2\sqrt{2}}\|\mathbf{R} - \mathbf{R}^\top\|_F. && \text{(B.26)}
\end{aligned}$$

Note that, without loss of generality, we can assume that $\theta \in [0, \pi]$. Thus, $\sin\theta$ is always positive.

B.2.3 Eigenvalues of the rotation matrix

The eigenvalues of a rotation matrix $\mathbf{R}$ can be determined by noting that the eigenvalues of $\exp\Big(\mathcal{S}(\boldsymbol{\omega})\Big)$ are the exponentials of the eigenvalues of $\mathcal{S}(\boldsymbol{\omega})$. Thus, to find the eigenvalues

λ of $\mathcal{S}(\boldsymbol{\omega})$, we compute the characteristic polynomial:

$$\det\Big(\lambda\mathbf{I} - \mathcal{S}(\boldsymbol{\omega})\Big) = \begin{vmatrix} \lambda & \omega_3 & -\omega_2 \\ -\omega_3 & \lambda & \omega_1 \\ \omega_2 & -\omega_1 & \lambda \end{vmatrix} = \lambda^3 + \lambda(\omega_1^2 + \omega_2^2 + \omega_3^2) = \lambda(\lambda^2 + \theta^2) = 0 \quad \text{(B.27)}$$

$$\implies \lambda \in \{0, i\theta, -i\theta\}. \quad \text{(B.28)}$$

Thus, the eigenvalues of $\mathbf{R}$ are $e^0 = 1$, $e^{i\theta}$, and $e^{-i\theta}$.

B.2.4 Frobenius norm of the logarithm of a rotation matrix

The Frobenius norm of the logarithm of a rotation matrix can be computed by noting that, since $\mathbf{R} = \exp\Big(\mathcal{S}(\boldsymbol{\omega})\Big)$, it follows that $\mathcal{S}(\boldsymbol{\omega}) = \log(\mathbf{R})$. Therefore, the Frobenius norm of the logarithm of a rotation matrix is:

$$\|\log(\mathbf{R})\|_F^2 = \|\mathcal{S}(\boldsymbol{\omega})\|_F^2 = \operatorname{tr}\Big(\mathcal{S}(\boldsymbol{\omega})^\top \mathcal{S}(\boldsymbol{\omega})\Big) = \operatorname{tr}\Big(-\mathcal{S}(\boldsymbol{\omega})^2\Big) = \operatorname{tr}\Big(\theta^2\mathbf{I} - \mathbf{B}\Big) = 2\theta^2. \quad \text{(B.29)}$$

B.3 Translation component of the SE(3) EE-distance

Our objective in this section is to derive the function that allows us to compute the translation component of the EE-distance in SE(3) (see Section 4.8.1). Let $\mathbf{X} = (\mathbf{I} - \mathbf{Q})^{-1}\log(\mathbf{Q})$, where $\mathbf{Q} \in \mathrm{SO}(3)$ is a rotation matrix. Furthermore, let $\bar{\mathbf{X}} = \mathbf{X}^\top\mathbf{X}$. The derivations here will then allow a simple expression for $\mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}$ for any $\mathbf{u} \in \mathbb{R}^3$.

We start by expressing $\bar{\mathbf{X}}$ as as a function $\Phi(\mathbf{Q})$:

$$\begin{aligned}
\bar{\mathbf{X}} = \mathbf{X}^\top\mathbf{X} &= (\log\mathbf{Q})^\top\Big((\mathbf{I}-\mathbf{Q})^{-1}\Big)^\top(\mathbf{I}-\mathbf{Q})^{-1}\log\mathbf{Q} && \text{(B.30)}\\
&= -\log\mathbf{Q}(\mathbf{I}-\mathbf{Q}^\top)^{-1}(\mathbf{I}-\mathbf{Q})^{-1}\log\mathbf{Q} && \text{(B.31)}\\
&= -\log\mathbf{Q}\Big((\mathbf{I}-\mathbf{Q})(\mathbf{I}-\mathbf{Q}^\top)\Big)^{-1}\log\mathbf{Q} && \text{(B.32)}\\
&= -\log\mathbf{Q}(2\mathbf{I}-(\mathbf{Q}+\mathbf{Q}^\top))^{-1}\log\mathbf{Q} && \text{(B.33)}\\
&= \Phi(\mathbf{Q}), && \text{(B.34)}
\end{aligned}$$

where we used the fact[1] that $(\log\mathbf{Q})^\top = -\log\mathbf{Q}$. Now, from the Cayley-Hamilton theorem (Chen, 2009, p. 63), this function can be expressed by the characteristic polynomial of $\mathbf{Q}$:

$$\Phi(\mathbf{Q}) = \beta_0\mathbf{Q}^{-1} + \beta_1\mathbf{I} + \beta_2\mathbf{Q}, \quad \text{(B.35)}$$

for which the coefficients β_i can be found by means of the eigenvalues of $\mathbf{Q}$. These

[1]The fact that $(\log\mathbf{Q})^\top = \log(\mathbf{Q}^\top) = -\log\mathbf{Q}$ can be easily seen through the power series of the matrix logarithm or from the fact that the logarithm of a rotation matrix is a skew-symmetric matrix.

coefficients can be found by evaluating the function Φ at each eigenvalue and comparing to (B.33). Since the eigenvalues λ_i of $\mathbf{Q}$ are $\{1, e^{i\theta}, e^{-i\theta}\}, \theta \in [0, \pi]$, we must solve the following equation for each λ_i:

$$\beta_0\lambda_i^{-1} + \beta_1 + \beta_2\lambda_i = -\log\lambda_i(2 - (\lambda_i + \lambda_i^{-1}))^{-1}\log\lambda_i. \tag{B.36}$$

Thus, for each eigenvalue, we have:

$$\beta_0 + \beta_1 + \beta_2 = \frac{-\log(1)^2}{2 - (1+1)} = 1, \qquad \text{for } \lambda = 1, \tag{B.37}$$

$$\beta_0 e^{-i\theta} + \beta_1 + \beta_2 e^{i\theta} = \frac{\theta^2}{2 - (e^{i\theta} + e^{-i\theta})}, \qquad \text{for } \lambda = e^{i\theta}, \tag{B.38}$$

$$\beta_0 e^{i\theta} + \beta_1 + \beta_2 e^{-i\theta} = \frac{\theta^2}{2 - (e^{-i\theta} + e^{i\theta})}, \qquad \text{for } \lambda = e^{i\theta}. \tag{B.39}$$

Subtracting (B.39) from (B.38) and multiplying by $\frac{1}{2i}$ results in

$$\frac{1}{2i}\beta_2(e^{i\theta} - e^{-i\theta}) + \frac{1}{2i}\beta_0(e^{-i\theta} - e^{i\theta}) = \left(\frac{\theta^2}{2 - (e^{i\theta} + e^{-i\theta})} - \frac{\theta^2}{2 - (e^{-i\theta} + e^{i\theta})}\right)\frac{1}{2i} \tag{B.40}$$

$$\beta_2 \sin\theta - \beta_0 \sin\theta = 0 \tag{B.41}$$

$$\implies \beta_0 = \beta_2. \tag{B.42}$$

Now, summing (B.38) and (B.39), multiplying by $\frac{1}{2}$, and using $\beta_0 = \beta_2$, results in:

$$\beta_1 + \frac{1}{2}\beta_2(e^{i\theta} + e^{-i\theta}) + \frac{1}{2}\beta_0(e^{-i\theta} + e^{i\theta}) = \left(\frac{\theta^2}{2 - (e^{i\theta} + e^{-i\theta})} + \frac{\theta^2}{2 - (e^{-i\theta} + e^{i\theta})}\right)\frac{1}{2} \tag{B.43}$$

$$\beta_1 + 2\beta_0\cos\theta = \frac{\theta^2}{2 - 2\cos\theta} \tag{B.44}$$

$$\implies \beta_1 = \frac{\theta^2}{2 - 2\cos\theta} - 2\beta_0\cos\theta. \tag{B.45}$$

Finally, substituting (B.42) and (B.45) into (B.37), we obtain:

$$1 = \frac{\theta^2}{2 - 2\cos\theta} - 2\beta_0\cos\theta + 2\beta_0 \tag{B.46}$$

$$\implies \beta_0 = \frac{2 - 2\cos\theta - \theta^2}{4(1 - \cos\theta)^2}. \tag{B.47}$$

Thus, we can express

$$\Phi(\mathbf{Q}) = \bar{\mathbf{X}} = (1 - 2\beta_0)\mathbf{I} + \beta_0\left(\mathbf{Q} + \mathbf{Q}^\top\right), \tag{B.48}$$

since $\beta_1 = 1 - \beta_0 - \beta_1 = 1 - 2\beta_0$. Therefore, since we can easily obtain the angle θ from the rotation matrix, the expression $\mathbf{u}^\top\bar{\mathbf{X}}\mathbf{u}$ does not need to involve the computation of

matrix logarithms.

B.3.1 Discontinuity analysis

The expression in (B.48) depends on the coefficient β_0, which in turn has a discontinuity when $\theta = 0$, since the denominator would be zero. Our objective in this section is to show that the limit of β_0 as $\theta \to 0$ exists and can be used when $\theta \approx 0$. First, note that when $\theta \to 0$, β_0 is indeterminate, so we apply L'Hôpital's rule:

$$\lim_{\theta\to 0} \beta_0 = \lim_{\theta\to 0} \frac{2 - 2\cos\theta - \theta^2}{4(1-\cos\theta)^2} = \lim_{\theta\to 0} \frac{2\sin\theta - 2\theta}{8(1-\cos\theta)\sin\theta}, \tag{B.49}$$

which is again indeterminate. Applying L'Hôpital's rule again, we find

$$\lim_{\theta\to 0} \frac{2\sin\theta - 2\theta}{8(1-\cos\theta)\sin\theta} = \lim_{\theta\to 0} \frac{2\cos\theta - 2}{8(1-\cos\theta)\cos\theta + 8(\sin\theta)^2}, \tag{B.50}$$

which is still indeterminate. Applying L'Hôpital's rule one more time, we find

$$\lim_{\theta\to 0} \frac{2\cos\theta - 2}{8(1-\cos\theta)\cos\theta + 8(\sin\theta)^2} = \lim_{\theta\to 0} \frac{-2\sin\theta}{-8(1-\cos\theta)\sin\theta + 24\cos\theta\sin\theta}, \tag{B.51}$$

which remains indeterminate. Applying L'Hôpital's rule one last time, we find

$$\lim_{\theta\to 0} \beta_0 = \lim_{\theta\to 0} \frac{-2\sin\theta}{-8(1-\cos\theta)\sin\theta + 24\cos\theta\sin\theta} \tag{B.52}$$

$$= \lim_{\theta\to 0} \frac{-2\cos\theta}{-8(1-\cos\theta)\cos\theta + 24(\cos\theta)^2 - 32(\sin\theta)^2} = -\frac{1}{12}. \tag{B.53}$$

Thus, when $\theta \approx 0$, we can use $\beta_0 = -\frac{1}{12}$ in (B.48).

B.4 Explicit gradient of the EE-Distance in SE(3)

In this section, we derive the explicit expression for the L operator, which behaves as a gradient, of the EE-distance $\widehat{D}$ in SE(3), as discussed in Section 4.8.1. This will allow us to explicitly compute the normal vector $\boldsymbol{\xi}_N$, as defined in Definition 4.13. This section will rely on the properties derived in Appendix A.

We begin by introducing two operators that will be used throughout this derivation. Let $\mathrm{rot}(\cdot)$ denote the operator that extracts the rotation matrix from a group element, such that $\mathrm{rot}(\mathbf{Z}) = \mathbf{Q} \in \mathrm{SO}(3)$ for some $\mathbf{Z} \in \mathrm{SE}(3)$. Similarly, let $\mathrm{tran}(\cdot)$ be the operator that extracts the translation vector from a group element, such that for $\mathbf{Z} \in \mathrm{SE}(3)$, $\mathrm{tran}(\mathbf{Z}) = \mathbf{u} \in \mathbb{R}^3$.

Thus, the EE-distance can be written as:

$$\widehat{E}(\mathbf{Z}) = \sqrt{2\theta^2 + \operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})}, \tag{B.54}$$

where θ depends on the rotation part of $\mathbf{Z}$.

Next, we will distinguish between the canonical isomorphisms for the Lie algebras $\mathfrak{se}(3)$ and $\mathfrak{so}(3)$. Let $\widehat{\mathcal{S}} : \mathbb{R}^3 \to \mathfrak{so}(3)$ be the canonical isomorphism for $\mathfrak{so}(3)$, and $\mathcal{S} : \mathbb{R}^6 \to \mathfrak{se}(3)$ be the canonical isomorphism for $\mathfrak{se}(3)$. Additionally, we will use the notation $\widehat{\mathbf{e}}_i$ for the canonical basis of $\mathbb{R}^3$ and $\mathbf{e}_i$ for the canonical basis of $\mathbb{R}^6$.

The L operator of the EE-distance with respect to $\mathbf{Z}$ can be expressed as:

$$\begin{aligned} \mathrm{L}[\widehat{E}](\mathbf{Z}) &= \frac{1}{2\sqrt{2\theta^2 + \operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})}} \mathrm{L}\Big[2\theta^2 + \operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})\Big](\mathbf{Z}) \\ &= \frac{1}{2\sqrt{2\theta^2 + \operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})}} \Big(\mathrm{L}\Big[2\theta^2\Big](\mathbf{Z}) + \mathrm{L}\Big[\operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})\Big](\mathbf{Z})\Big), \end{aligned} \tag{B.55}$$

where $\theta, \mathbf{u}$ and $\bar{\mathbf{X}}$ are all functions of $\mathbf{Z}$. We will now analyze the two terms in the operator expression separately. Starting with the first term, we have

$$\mathrm{L}\Big[2\theta^2\Big](\mathbf{Z}) = 4\theta \mathrm{L}[\theta](\mathbf{Z}). \tag{B.56}$$

As shown in Section 4.8.1, θ is expressed as $\theta = \operatorname{atan2}(\sin\theta, \cos\theta)$, however both derivatives of atan2 and $\tan^{-1}$ coincide. The L operator of θ is then expressed as

$$\mathrm{L}[\theta](\mathbf{Z}) = \frac{1}{\left(\frac{\sin\theta}{\cos\theta}\right)^2 + 1} \mathrm{L}\left[\frac{\sin\theta}{\cos\theta}\right](\mathbf{Z}) \tag{B.57}$$

$$= \cos^2\theta \left(\frac{\cos\theta\, \mathrm{L}[\sin\theta](\mathbf{Z}) - \sin\theta\, \mathrm{L}[\cos\theta](\mathbf{Z})}{\cos^2\theta}\right) \tag{B.58}$$

$$= \cos\theta\, \mathrm{L}[\sin\theta](\mathbf{Z}) - \sin\theta\, \mathrm{L}[\cos\theta](\mathbf{Z}), \tag{B.59}$$

where the quotient rule has been used (see Appendix A.2). The expressions for $\sin\theta$ and $\cos\theta$ are exactly the ones in Appendix B.2, and we will use the cosine and sine notations for simplicity.

Focusing for now only on L operator of $\cos\theta$, let $\mathbb{R}^6 \ni \boldsymbol{\zeta} = [\mathbf{v}^\top\ \boldsymbol{\omega}^\top]^\top$ with $\mathbf{v}, \boldsymbol{\omega} \in \mathbb{R}^3$. Then, clearly

$$\operatorname{rot}\Big(\exp\Big(\varepsilon\, \mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{Z}\Big) = \exp\Big(\varepsilon\, \widehat{\mathcal{S}}(\boldsymbol{\omega})\Big) \operatorname{rot}(\mathbf{Z}). \tag{B.60}$$

Using this result and the implicit definition of L through derivatives, we can express

$$\begin{aligned}
\mathrm{L}[\cos\theta](\mathbf{Z})\boldsymbol{\zeta} &= \mathrm{L}\left[\frac{1}{2}\Big(\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})\Big)-1\Big)\right](\mathbf{Z})\boldsymbol{\zeta} = \frac{1}{2}\frac{d}{d\varepsilon}\left(\mathrm{tr}\left(\mathrm{rot}\Big(\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{Z}\Big)\right)\right)\Bigg|_{\varepsilon=0} \\
&= \frac{1}{2}\frac{d}{d\varepsilon}\left(\mathrm{tr}\left(\exp\Big(\varepsilon\widehat{\mathcal{S}}(\boldsymbol{\omega})\Big)\,\mathrm{rot}(\mathbf{Z})\right)\right)\Bigg|_{\varepsilon=0} = \frac{1}{2}\,\mathrm{tr}\left(\frac{d}{d\varepsilon}\left(\exp\Big(\varepsilon\widehat{\mathcal{S}}(\boldsymbol{\omega})\Big)\,\mathrm{rot}(\mathbf{Z})\right)\Bigg|_{\varepsilon=0}\right) \\
&= \frac{1}{2}\,\mathrm{tr}\left(\left(\widehat{\mathcal{S}}(\boldsymbol{\omega})\exp\Big(\varepsilon\widehat{\mathcal{S}}(\boldsymbol{\omega})\Big)\,\mathrm{rot}(\mathbf{Z})\right)\Bigg|_{\varepsilon=0}\right) = \frac{1}{2}\,\mathrm{tr}\Big(\widehat{\mathcal{S}}(\boldsymbol{\omega})\,\mathrm{rot}(\mathbf{Z})\Big). \qquad \text{(B.61)}
\end{aligned}$$

This implies that:

$$\mathrm{L}[\cos\theta](\mathbf{Z}) = \frac{1}{2}\begin{bmatrix}\mathbf{0} & \mathrm{tr}\Big(\widehat{\mathcal{S}}(\hat{\mathbf{e}}_1)\,\mathrm{rot}(\mathbf{Z})\Big) & \mathrm{tr}\Big(\widehat{\mathcal{S}}(\hat{\mathbf{e}}_2)\,\mathrm{rot}(\mathbf{Z})\Big) & \mathrm{tr}\Big(\widehat{\mathcal{S}}(\hat{\mathbf{e}}_3)\,\mathrm{rot}(\mathbf{Z})\Big)\end{bmatrix}. \qquad \text{(B.62)}$$

If we denote $\mathrm{rot}(\mathbf{Z})_{ij}$ as the element in the i^{th} row and j^{th} column of $\mathrm{rot}(\mathbf{Z})$, this expression can also be written as:

$$\mathrm{L}[\cos\theta](\mathbf{Z}) = \frac{1}{2}\underbrace{\begin{bmatrix}\mathbf{0} & \mathrm{rot}(\mathbf{Z})_{23}-\mathrm{rot}(\mathbf{Z})_{32} & \mathrm{rot}(\mathbf{Z})_{31}-\mathrm{rot}(\mathbf{Z})_{13} & \mathrm{rot}(\mathbf{Z})_{12}-\mathrm{rot}(\mathbf{Z})_{21}\end{bmatrix}}_{\mathbf{g}}. \qquad \text{(B.63)}$$

Switching our attention to the other L operator of (B.59), i.e., $\mathrm{L}[\sin\theta]$, we note the following:

$$\begin{aligned}
\Big\|\mathrm{rot}(\mathbf{Z})-\mathrm{rot}(\mathbf{Z})^\top\Big\|_F &= \mathrm{tr}\Big(\Big(\mathrm{rot}(\mathbf{Z})-\mathrm{rot}(\mathbf{Z})^\top\Big)^\top\Big(\mathrm{rot}(\mathbf{Z})-\mathrm{rot}(\mathbf{Z})^\top\Big)\Big)^{\frac{1}{2}} && \text{(B.64)} \\
&= \mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})^\top\mathrm{rot}(\mathbf{Z})-\mathrm{rot}(\mathbf{Z})^\top\mathrm{rot}(\mathbf{Z})^\top+\mathrm{rot}(\mathbf{Z})\,\mathrm{rot}(\mathbf{Z})^\top-\mathrm{rot}(\mathbf{Z})\,\mathrm{rot}(\mathbf{Z})\Big)^{\frac{1}{2}} && \text{(B.65)} \\
&= \sqrt{2\,\mathrm{tr}(\mathbf{I})-2\,\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})\,\mathrm{rot}(\mathbf{Z})\Big)} = \sqrt{6-2\,\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})\,\mathrm{rot}(\mathbf{Z})\Big)}, && \text{(B.66)}
\end{aligned}$$

using properties of the trace. With this, we can express the L operator of $\sin\theta$ as:

$$\begin{aligned}
\mathrm{L}[\sin\theta](\mathbf{Z}) &= \mathrm{L}\left[\frac{1}{2\sqrt{2}}\Big\|\mathrm{rot}(\mathbf{Z})-\mathrm{rot}(\mathbf{Z})^\top\Big\|_F\right](\mathbf{Z}) = \mathrm{L}\left[\frac{1}{2\sqrt{2}}\sqrt{6-2\,\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})^2\Big)}\right](\mathbf{Z}) && \text{(B.67)} \\
&= \frac{-2}{4\sqrt{2}\sqrt{6-2\,\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})^2\Big)}}\,\mathrm{L}\Big[\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})^2\Big)\Big](\mathbf{Z}) && \text{(B.68)} \\
&= \frac{-1}{4\sqrt{3-\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})^2\Big)}}\,\mathrm{L}\Big[\mathrm{tr}\Big(\mathrm{rot}(\mathbf{Z})^2\Big)\Big](\mathbf{Z}). && \text{(B.69)}
\end{aligned}$$

Using the implicit definition of L:

$$\mathrm{L}\Big[\mathrm{tr}\big(\mathrm{rot}(\mathbf{Z})^2\big)\Big](\mathbf{Z})\boldsymbol{\zeta} = \frac{d}{d\varepsilon}\left(\mathrm{tr}\left(\mathrm{rot}\Big(\exp\big(\varepsilon\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{Z}\Big)\,\mathrm{rot}\Big(\exp\big(\varepsilon\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{Z}\Big)\right)\right)\Bigg|_{\varepsilon=0} \tag{B.70}$$

$$= \mathrm{tr}\left(\frac{d}{d\varepsilon}\Big(\exp\big(\varepsilon\widehat{\mathcal{S}}(\boldsymbol{\omega})\big)\,\mathrm{rot}(\mathbf{Z})\exp\big(\varepsilon\widehat{\mathcal{S}}(\boldsymbol{\omega})\big)\,\mathrm{rot}(\mathbf{Z})\Big)\Big|_{\varepsilon=0}\right) \tag{B.71}$$

$$= \mathrm{tr}\Big(\widehat{\mathcal{S}}(\boldsymbol{\omega})\,\mathrm{rot}(\mathbf{Z})\,\mathrm{rot}(\mathbf{Z}) + \mathrm{rot}(\mathbf{Z})\widehat{\mathcal{S}}(\boldsymbol{\omega})\,\mathrm{rot}(\mathbf{Z})\Big) \tag{B.72}$$

$$= 2\,\mathrm{tr}\big(\widehat{\mathcal{S}}(\boldsymbol{\omega})\,\mathrm{rot}(\mathbf{Z})^2\big). \tag{B.73}$$

Note that this final result is in the same format as (B.61), and thus has a similar expression to $\mathrm{L}[\cos\theta](\mathbf{Z})$. Therefore, we can express $\mathrm{L}[\sin\theta]$ as:

$$\mathrm{L}[\sin\theta](\mathbf{Z}) = \frac{-1}{4\sqrt{3-\mathrm{tr}\big(\mathrm{rot}(\mathbf{Z})^2\big)}}\mathbf{f}, \tag{B.74}$$

where

$$\mathbf{f} = 2\begin{bmatrix}\mathbf{0} & \{\mathrm{rot}(\mathbf{Z})^2\}_{23} - \{\mathrm{rot}(\mathbf{Z})^2\}_{32} & \{\mathrm{rot}(\mathbf{Z})^2\}_{31} - \{\mathrm{rot}(\mathbf{Z})^2\}_{13} & \{\mathrm{rot}(\mathbf{Z})^2\}_{12} - \{\mathrm{rot}(\mathbf{Z})^2\}_{21}\end{bmatrix}.$$

It is possible to simplify the expression for $\mathrm{L}[\sin\theta]$ even further by expanding $\mathrm{tr}(\mathrm{rot}(\mathbf{Z})^2)$ using the Rodrigues' formula (B.17):

$$\begin{aligned}\mathrm{tr}\big(\mathrm{rot}(\mathbf{Z})^2\big) =& (\cos^2\theta)\,\mathrm{tr}(\mathbf{I}) + \frac{2(\sin\theta - \sin\theta\cos\theta)}{\theta^3}\,\mathrm{tr}\big(\mathcal{S}(\boldsymbol{\omega})\,\mathbf{B}\big) + \frac{2(\cos\theta - \cos^2\theta)}{\theta^2}\,\mathrm{tr}\,(\mathbf{B}) \\ &+ \frac{\sin^2\theta}{\theta^2}\,\mathrm{tr}\big(\mathcal{S}(\boldsymbol{\omega})^2\big) + \frac{2\cos\theta\sin\theta}{\theta}\,\mathrm{tr}\big(\mathcal{S}(\boldsymbol{\omega})\big) + \frac{(1-\cos\theta)^2}{\theta^4}\,\mathrm{tr}(\mathbf{B}^2) \\ =& 3\cos^2\theta + \frac{2(\cos\theta - \cos^2\theta)}{\theta^2}\,\mathrm{tr}\,(\mathbf{B}) + \frac{\sin^2\theta}{\theta^2}\,\mathrm{tr}\big(\mathcal{S}(\boldsymbol{\omega})^2\big) + \frac{(1-\cos\theta)^2}{\theta^4}\,\mathrm{tr}(\mathbf{B}^2) \\ =& 3\cos^2\theta + 2\cos\theta - 2\cos^2\theta - 2\sin^2\theta + \frac{(1-\cos\theta)^2}{\theta^4}\,\mathrm{tr}(\mathbf{B}^2),\end{aligned} \tag{B.75}$$

where $\mathrm{tr}(\mathbf{B}^2)$ is given by

$$\mathrm{tr}(\mathbf{B}^2) = \mathrm{tr}\Big(\big(\mathcal{S}(\boldsymbol{\omega})^2 + \theta^2\mathbf{I}\big)\Big) = \theta^4\,\mathrm{tr}(\mathbf{I}) + 2\theta^2\,\mathrm{tr}\big(\mathcal{S}(\boldsymbol{\omega})^2\big) + \mathrm{tr}\big(\mathcal{S}(\boldsymbol{\omega})^4\big) \tag{B.76}$$

$$= 2\theta^4\,\mathrm{tr}(\mathbf{I}) + 2\theta^2\,\mathrm{tr}\big(\mathcal{S}(\boldsymbol{\omega})^2\big) - \theta^2\,\mathrm{tr}(\mathbf{B}) \tag{B.77}$$

$$= 6\theta^4 - 4\theta^4 - \theta^4 = \theta^4, \tag{B.78}$$

thus

$$\operatorname{tr}\Big(\operatorname{rot}(\mathbf{Z})^2\Big) = 3\cos^2\theta + 2\cos\theta - 2\cos^2\theta - 2\sin^2\theta + 1 - 2\cos\theta + \cos^2\theta \tag{B.79}$$

$$= 2\cos^2\theta - 2\sin^2\theta + 1 = 3 - 4\sin^2\theta. \tag{B.80}$$

Finally, $\mathrm{L}[\sin\theta]$ reduces the the expression:

$$\mathrm{L}[\sin\theta](\mathbf{Z}) = \frac{-1}{4\sqrt{3 - \operatorname{tr}\Big(\operatorname{rot}(\mathbf{Z})^2\Big)}}\mathbf{f} = \frac{-1}{8\sin\theta}\mathbf{f}. \tag{B.81}$$

Now, using (B.63) and (B.81), we can express $\mathrm{L}[\theta]$ as:

$$\mathrm{L}[\theta](\mathbf{Z}) = \frac{-\cos\theta}{8\sin\theta}\mathbf{f} - \frac{\sin\theta}{2}\mathbf{g}. \tag{B.82}$$

To finally determine $\mathrm{L}[\widehat{E}]$, we still need to compute $\mathrm{L}\Big[\operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})\Big]$. Since L is defined for scalar functions, we begin by expressing $\operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})$ element-wise. This results in:

$$\operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z}) = \sum_{i=1}^{3}\sum_{j=1}^{3} \operatorname{tran}(\mathbf{Z})_i \operatorname{tran}(\mathbf{Z})_j \bar{\mathbf{X}}_{ij}. \tag{B.83}$$

The L operator of this expression is then given by:

$$\begin{aligned}\mathrm{L}\Big[\operatorname{tran}(\mathbf{Z})^\top \bar{\mathbf{X}} \operatorname{tran}(\mathbf{Z})\Big](\mathbf{Z}) = \sum_{i=1}^{3}\sum_{j=1}^{3}\Big(&2\,\mathrm{L}\Big[\operatorname{tran}(\mathbf{Z})_i\Big](\mathbf{Z}) \operatorname{tran}(\mathbf{Z})_j \bar{\mathbf{X}}_{ij} \\ &+ \operatorname{tran}(\mathbf{Z})_i \operatorname{tran}(\mathbf{Z})_j \,\mathrm{L}\Big[\bar{\mathbf{X}}_{ij}\Big](\mathbf{Z})\Big).\end{aligned} \tag{B.84}$$

This holds because $\bar{\mathbf{X}}$ is symmetric.

Next, we compute the L operator of $\operatorname{tran}(\mathbf{Z})_i$, which is expressed implicitly as:

$$\begin{aligned}\mathrm{L}\Big[\operatorname{tran}(\mathbf{Z})_i\Big](\mathbf{Z})\boldsymbol{\zeta} &= \frac{d}{d\varepsilon}\left(\operatorname{tran}\Big(\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{Z}\Big)_i\right)\Big|_{\varepsilon=0} = \operatorname{tran}\left(\frac{d}{d\varepsilon}\Big\{\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{Z}\Big\}_i\Big|_{\varepsilon=0}\right) \\ &= \operatorname{tran}\left(\Big\{\mathcal{S}(\boldsymbol{\zeta})\exp\Big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\Big)\mathbf{Z}\Big\}_i\Big|_{\varepsilon=0}\right) = \operatorname{tran}\Big(\big\{\mathcal{S}(\boldsymbol{\zeta})\,\mathbf{Z}\big\}_i\Big) \\ &= \Big\{\widehat{\mathcal{S}}(\boldsymbol{\omega})\operatorname{tran}(\mathbf{Z}) + \mathbf{v}\Big\}_i.\end{aligned} \tag{B.85}$$

This in turn implies

$$\mathrm{L}\Big[\operatorname{tran}(\mathbf{Z})_i\Big](\mathbf{Z}) = \Big[\delta_{1i} \quad \delta_{2i} \quad \delta_{3i} \quad \{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_1)\operatorname{tran}(\mathbf{Z})\}_i \quad \{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_2)\operatorname{tran}(\mathbf{Z})\}_i \quad \{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_3)\operatorname{tran}(\mathbf{Z})\}_i\Big], \tag{B.86}$$

where δ_{ij} is the Kronecker delta, i.e., $\delta_{ij} = 1 \iff i = j$ and $\delta_{ij} = 0$ otherwise.

Next, we consider $\mathrm{L}[\bar{\mathbf{X}}_{ij}]$, which is given by:

$$\mathrm{L}[\bar{\mathbf{X}}_{ij}](\mathbf{Z}) = \mathrm{L}[\beta_0](\mathbf{Z})\Big(-2\delta_{ij} + \mathrm{rot}(\mathbf{Z})_{ij} + \mathrm{rot}(\mathbf{Z})_{ji}\Big) + \beta_0\Big(\mathrm{L}[\mathrm{rot}(\mathbf{Z})_{ij}](\mathbf{Z}) + \mathrm{L}[\mathrm{rot}(\mathbf{Z})_{ij}^{\top}](\mathbf{Z})\Big). \tag{B.87}$$

By expressing the L operator for each element in the i^{th} row and j^{th} column of $\mathrm{rot}(\mathbf{Z})$, we obtain

$$\begin{aligned}
\mathrm{L}[\mathrm{rot}(\mathbf{Z})_{ij}](\mathbf{Z})\boldsymbol{\zeta} &= \frac{d}{d\varepsilon}\left(\left\{\mathrm{rot}\Big(\exp\big(\varepsilon\,\mathcal{S}(\boldsymbol{\zeta})\big)\mathbf{Z}\Big)\right\}_{ij}\right)\Bigg|_{\varepsilon=0} && \text{(B.88)}\\
&= \frac{d}{d\varepsilon}\left(\left\{\exp\big(\varepsilon\widehat{\mathcal{S}}(\boldsymbol{\omega})\big)\,\mathrm{rot}(\mathbf{Z})\right\}_{ij}\right)\Bigg|_{\varepsilon=0} && \text{(B.89)}\\
&= \left\{\widehat{\mathcal{S}}(\boldsymbol{\omega})\,\mathrm{rot}(\mathbf{Z})\right\}_{ij}, && \text{(B.90)}
\end{aligned}$$

which implies

$$\mathrm{L}[\mathrm{rot}(\mathbf{Z})_{ij}](\mathbf{Z}) = \begin{bmatrix}\mathbf{0} & \left\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_1)\,\mathrm{rot}(\mathbf{Z})\right\}_{ij} & \left\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_2)\,\mathrm{rot}(\mathbf{Z})\right\}_{ij} & \left\{\widehat{\mathcal{S}}(\hat{\mathbf{e}}_3)\,\mathrm{rot}(\mathbf{Z})\right\}_{ij}\end{bmatrix}. \tag{B.91}$$

It is clear that the L operator of each element in the i^{th} row and j^{th} column of $\mathrm{rot}(\mathbf{Z})^{\top}$ is directly given by $\mathrm{L}[\mathrm{rot}(\mathbf{Z})_{ij}](\mathbf{Z})$:

$$\mathrm{L}[\mathrm{rot}(\mathbf{Z})_{ij}^{\top}](\mathbf{Z}) = \mathrm{L}[\mathrm{rot}(\mathbf{Z})_{ji}](\mathbf{Z}). \tag{B.92}$$

The last term remaining is $\mathrm{L}[\beta_0](\mathbf{Z})$, which is given by

$$\begin{aligned}
\mathrm{L}[\beta_0](\mathbf{Z}) &= \frac{d\beta_0}{d\theta}\,\mathrm{L}[\theta](\mathbf{Z}) && \text{(B.93)}\\
&= \frac{\theta^2\sin(\theta) - \theta - \sin(\theta) + \big(\theta + \sin(\theta)\big)\cos(\theta)}{2(1-\cos(\theta))^3}\,\mathrm{L}[\theta](\mathbf{Z}). && \text{(B.94)}
\end{aligned}$$

Now, with all the necessary ingredients, we can compute:

$$\mathrm{L}[\widehat{E}](\mathbf{Z}) = \frac{1}{2\widehat{E}(\mathbf{Z})}\Big(4\theta\,\mathrm{L}[\theta](\mathbf{Z}) + \mathrm{L}\Big[\mathrm{tran}(\mathbf{Z})^{\top}\bar{\mathbf{X}}\,\mathrm{tran}(\mathbf{Z})\Big](\mathbf{Z})\Big), \tag{B.95}$$

using the expression in (B.82) for $\mathrm{L}[\theta]$ and expression (B.84) for $\mathrm{L}\Big[\mathrm{tran}(\mathbf{Z})^{\top}\bar{\mathbf{X}}\,\mathrm{tran}(\mathbf{Z})\Big]$. This concludes the derivation of the L operator of the EE-distance in SE(3).

Bibliography

Y. Aboudorra, C. Gabellieri, R. Brantjes, Q. Sablé, and A. Franchi. Modelling, Analysis, and Control of OmniMorph: an Omnidirectional Morphing Multi-rotor UAV. *Journal of Intelligent & Robotic Systems*, 110(1):21, mar 2024. doi: 10.1007/s10846-024-02054-x.

A. Barrau and S. Bonnabel. The invariant extended Kalman filter as a stable observer. *IEEE Transactions on Automatic Control*, 62(4):1797–1812, apr 2017. doi: 10.1109/TAC.2016.2594085.

G. Bogfjellmo and A. Schmeding. The Lie Group Structure of the Butcher Group. *Foundations of Computational Mathematics*, 17(1):127–159, feb 2017. doi: 10.1007/s10208-015-9285-5.

F. Bullo and A. D. Lewis. *Geometric Control of Mechanical Systems - Modeling, Analysis, and Design for Simple Mechanical Control Systems*. Springer, New York, 2004. ISBN 978-0-387-22195-3.

F. Campos Filho. *Algoritmos numéricos*. LTC, Rio de Janeiro, 2 edition, 2007. ISBN 9788521615378.

S. M. Carroll. *Spacetime and Geometry: An Introduction to General Relativity*. Cambridge University Press, Cambridge, 2019.

L. Chaimowicz, N. Michael, and V. Kumar. Controlling Swarms of Robots Using Interpolated Implicit Functions. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 2487–2492. IEEE, 2005. doi: 10.1109/ROBOT.2005.1570486.

C.-T. Chen. *Linear System Theory and Design*. Oxford University Press, New York, 3 edition, 2009. ISBN 978-0-195-39207-4.

Z. Chen and Z. Zuo. Non-singular cooperative guiding vector field under a homotopy equivalence transformation. *Automatica*, 171:111962, jan 2025. doi: 10.1016/j.automatica.2024.111962.

H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki, and S. Thrun. *Principles of Robot Motion - Theory, Algorithms, and Implementations*. MIT Press, Cambridge, 2005. ISBN 978-0-262-03327-5.

R. Conn and M. Kam. Robot motion planning on N-dimensional star worlds among moving obstacles. *IEEE Transactions on Robotics and Automation*, 14(2):320–325, apr 1998. doi: 10.1109/70.681250.

P. Culbertson, J.-J. Slotine, and M. Schwager. Decentralized Adaptive Control for Collaborative Manipulation of Rigid Bodies. *IEEE Transactions on Robotics*, 37(6): 1906–1920, dec 2021. doi: 10.1109/TRO.2021.3072021.

O. Dreyer. Quasinormal Modes, the Area Spectrum, and Black Hole Entropy. *Physical Review Letters*, 90(8):081301, feb 2003. doi: 10.1103/PhysRevLett.90.081301.

T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):932–946, jul 2002. doi: 10.1109/TPAMI.2002.1017620.

J. J. Duistermaat and J. A. C. Kolk. *Lie Groups*. Springer, Berlin, Heidelberg, 1999. ISBN 978-3-540-15293-4.

T. Duong, A. Altawaitan, J. Stanley, and N. Atanasov. Port-Hamiltonian Neural ODE Networks on Lie Groups for Robot Dynamics Learning and Control. *IEEE Transactions on Robotics*, 40:3695–3715, 2024. doi: 10.1109/TRO.2024.3428433.

J. Gallier and J. Quaintance. *Differential Geometry and Lie Groups*. Geometry and Computing. Springer, Cham, 1 edition, 2020. ISBN 978-3-030-46040-2.

Y. Gao, C. Bai, L. Zhang, and Q. Quan. Multi-UAV cooperative target encirclement within an annular virtual tube. *Aerospace Science and Technology*, 128:107800, sep 2022. doi: 10.1016/J.AST.2022.107800.

V. M. Goncalves, L. C. A. Pimenta, C. A. Maia, B. C. O. Dutra, and G. A. S. Pereira. Vector Fields for Robot Navigation Along Time-Varying Curves in n-Dimensions. *IEEE Transactions on Robotics*, 26(4):647–659, aug 2010. doi: 10.1109/TRO.2010.2053077.

V. M. Gonçalves, A. Tzes, F. Khorrami, and P. Fraisse. Smooth Distances for Second-Order Kinematic Robot Control. *IEEE Transactions on Robotics*, 40:2950–2966, 2024. doi: 10.1109/TRO.2024.3400924.

B. C. Hall. *Lie Groups, Lie Algebras, and Representations*, volume 222 of *Graduate Texts in Mathematics*. Springer International Publishing, Cham, 2 edition, 2015. ISBN 978-3-319-13466-6. doi: 10.1007/978-3-319-13467-3.

M. Hamandi, F. Usai, Q. Sablé, N. Staub, M. Tognon, and A. Franchi. Design of multirotor aerial vehicles: A taxonomy based on input allocation. *The International Journal of Robotics Research*, 40(8-9):1015–1044, aug 2021. doi: 10.1177/02783649211025998.

M. Hamandi, A. M. Ali, N. Evangeliou, D. Chaikalis, A. Tzes, K. Kyriakopoulos, and F. Khorrami. Mechatronic Design of an Omnidirectional Octorotor UAV. In *2024 10th International Conference on Automation, Robotics and Applications (ICARA)*, pages 300–304. Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/ICARA60736.2024.10553043.

C. Hamburger. Vektorfeldalgebren, Gruppenaktionen und das Keplerproblem. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 89(2):132–159, feb 2009. doi: 10.1002/zamm.200700169.

E. Ilievski, J. De Nardis, S. Gopalakrishnan, R. Vasseur, and B. Ware. Superuniversality of Superdiffusion. *Physical Review X*, 11(3):031023, jul 2021. doi: 10.1103/PhysRevX.11.031023.

P. A. Ioannou and J. Sun. *Robust Adaptive Control*. Courier Corporation, New York, 2012. ISBN 978-0-486-32072-4.

M. Kamel, S. Verling, O. Elkhatib, C. Sprecher, P. Wulkop, Z. Taylor, R. Siegwart, and I. Gilitschenski. The Voliro Omniorientational Hexacopter: An Agile and Maneuverable Tiltable-Rotor Aerial Vehicle. *IEEE Robotics & Automation Magazine*, 25(4):34–44, dec 2018. doi: 10.1109/MRA.2018.2866758.

D. Kapec, P. Mitra, A.-M. Raclariu, and A. Strominger. 2D Stress Tensor for 4D Gravity. *Physical Review Letters*, 119(12):121601, sep 2017. doi: 10.1103/PhysRevLett.119.121601.

O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. In *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, volume 2, pages 500–505. Institute of Electrical and Electronics Engineers, 1985. doi: 10.1109/ROBOT.1985.1087247.

M. Krstic, I. Kanellakopoulos, and P. V. Kokotovic. *Nonlinear and Adaptive Control Design*. Wiley, New York, 1995. ISBN 978-0-471-12732-1.

S. Lang. *Differential and Riemannian Manifolds*. Springer, New York, NY, 3 edition, 2012. ISBN 978-1-461-24182-9.

J. M. Lee. *Introduction to Smooth Manifolds*. Graduate Texts in Mathematics. Springer, New York, NY, 2 edition, 2012. ISBN 978-1-4419-9981-8.

N. A. Lemos. *Analytical Mechanics*. Cambridge University Press, Cambridge, 2018. ISBN 9781108416580.

M. Lu and F. Li. Survey on lie group machine learning. *Big Data Mining and Analytics*, 3 (4):235–258, dec 2020. doi: 10.26599/BDMA.2020.9020011.

P. J. McCarthy and C. Nielsen. Global Synchronization of Sampled-Data Invariant Systems on Exponential Lie Groups. *IEEE Transactions on Control of Network Systems*, 7(3): 1080–1089, sep 2020. doi: 10.1109/TCNS.2019.2963019.

Mong-ying A. Hsieh and V. Kumar. Pattern generation with multiple robots. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, volume 2006, pages 2442–2447. IEEE, 2006. doi: 10.1109/ROBOT.2006.1642068.

R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, Fla, 1994. ISBN 978-1-351-46978-4.

A. H. D. Nunes, A. M. C. Rezende, G. P. Cruz, G. M. Freitas, V. M. Goncalves, and L. C. A. Pimenta. Vector field for curve tracking with obstacle avoidance. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, volume 2022-Decem, pages 2031–2038. IEEE, dec 2022. doi: 10.1109/CDC51059.2022.9992435.

A. H. D. Nunes, G. V. Raffo, and L. C. A. Pimenta. Integrated vector field and backstepping control for quadcopters. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, volume 2023-May, pages 1256–1262. IEEE, may 2023. doi: 10.1109/ICRA48891.2023.10160824.

F. B. A. Pessoa and L. C. A. Pimenta. Vector Field Based Adaptive Control for Collaborative Manipulation. In *XXV Congresso Brasileiro de Automática*, 2024.

L. Pimenta, A. Fonseca, G. Pereira, R. Mesquita, E. Silva, W. Caminhas, and M. Campos. Robot navigation based on electrostatic field computation. *IEEE Transactions on Magnetics*, 42(4):1459–1462, 2006. doi: 10.1109/TMAG.2006.870931.

L. C. A. Pimenta, M. L. Mendes, R. C. Mesquita, and G. A. S. Pereira. Fluids in Electrostatic Fields: An Analogy for Multirobot Control. *IEEE Transactions on Magnetics*, 43(4):1765–1768, apr 2007. doi: 10.1109/TMAG.2007.892514.

A. M. C. Rezende, V. M. Goncalves, and L. C. A. Pimenta. Constructive Time-Varying Vector Fields for Robot Navigation. *IEEE Transactions on Robotics*, 38(2):852–867, apr 2022. doi: 10.1109/TRO.2021.3093674.

E. Rimon and D. E. Koditschek. Exact Robot Navigation using Artificial Potential Functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, 1992. doi: 10.1109/70.163777.

J.-J. E. Slotine and W. Li. *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, N.J, 1991. ISBN 0-13-040890-5.

M. W. Spong, S. Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. John Wiley & Sons, New York, 2 edition, 2020. ISBN 978-1-119-52399-4.

A. M. Steane. *Relativity Made Relatively Easy*. Oxford University Press, Oxford, 2012. ISBN 978-0-19-966286-9.

R. Vemulapalli, F. Arrate, and R. Chellappa. Human Action Recognition by Representing 3D Skeletons as Points in a Lie Group. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 588–595. IEEE, jun 2014. doi: 10.1109/CVPR.2014.82.

M. Wang and A. Tayebi. Hybrid Feedback for Global Tracking on Matrix Lie Groups SO(3) and SE(3). *IEEE Transactions on Automatic Control*, 67(6):2930–2945, jun 2022. doi: 10.1109/TAC.2021.3097704.

C. Wu, J. Chen, D. Jeltsema, and C. Dai. Guidance Vector Field Encoding based on Contraction Analysis. In *2018 European Control Conference (ECC)*, pages 282–287. IEEE, jun 2018. doi: 10.23919/ECC.2018.8550301.

W. Yao, H. G. H. G. de Marina, B. Lin, and M. Cao. Singularity-free guiding vector field for robot navigation. *IEEE Transactions on Robotics*, 37(4):1206–1221, aug 2021. doi: 10.1109/TRO.2020.3043690.

W. Yao, B. Lin, B. D. O. Anderson, and M. Cao. Topological Analysis of Vector-Field Guided Path Following on Manifolds. *IEEE Transactions on Automatic Control*, 68(3): 1353–1368, mar 2023. doi: 10.1109/TAC.2022.3151236.

D. Ž. Đoković and N. Q. Thắńg. On the Exponential Map of Almost Simple Real Algebraic Groups. *Journal of Lie Theory*, 5(2):275–291, 1995.