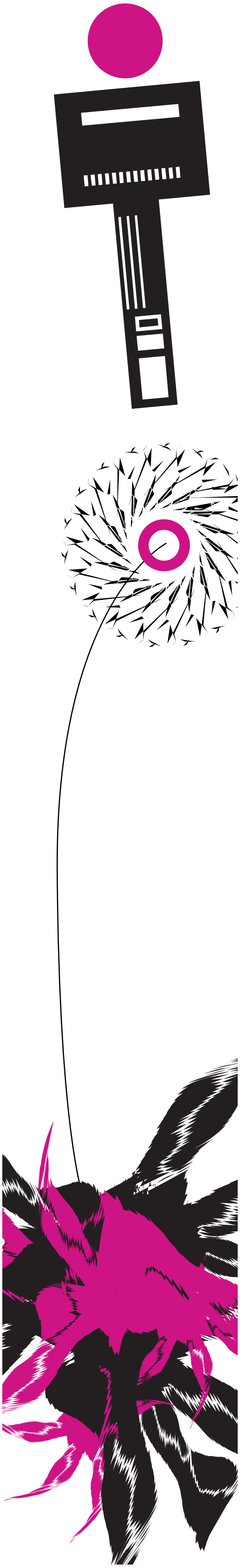


RAM ROBOTICS AND MECHATRONICS

OPTIMAL POTENTIAL SHAPING ON SE(3) VIA NEURAL APPROXIMATORS

Y.P. (Yannik) Wotte

MSC ASSIGNMENT

Committee:
prof. dr. ir. S. Stramigioli
dr. ir. F. Califano
dr. C. Brune

October, 2021

066RaM2021
Robotics and Mechatronics
EEMathCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

UNIVERSITY OF TWENTE. | TECHMED CENTRE
UNIVERSITY OF TWENTE. | DIGITAL SOCIETY INSTITUTE

Optimal Potential Energy Shaping on SE(3) via Neural Approximators

Yannik P. Wotte · Federico Califano · Stefano Massaroli · Michael Poli · Stefano Stramigioli

Abstract This work combines optimal control and energy balancing, passivity based control (EB-PBC) on the Lie group $SE(3)$, which is the configuration space of rigid bodies. Generally, EB-PBC achieves stable interactions with unknown environments by explicitly keeping the energy of a closed-loop system bounded. In the case of rigid bodies on $SE(3)$, this recently allowed deriving impedance control based on a quadratic energy [1]. However, choosing such a quadratic control-law is not connected to any principles from optimal control, which makes it an arbitrary choice. The derivation is phrased as an optimal control problem to extend such geometric impedance control beyond the quadratic case. Neural Nets and the Lie Group structure of $SE(3)$ are used to conveniently solve the arising non-trivial problem of optimization. The final algorithm is validated on a state-regulation task.

Keywords Nonlinear Control · Differential Geometry · Deep Learning · Robotics

1 Introduction

Passivity-based control (PBC) gained popularity in robotics, because passivity guarantees stable interactions with unknown, passive environments [2]. The key-feature is to treat entities as physical systems that exchange a limited amount of energy [3,4]. When a controller is treated as such a physical system with a bounded energy, this guarantees the stability of the closed-loop system when in contact with any other physical system [5,6].

The methodology of energy-balancing passivity-based control (EB-PBC) [3] explicitly guarantees such a bounded energy of a closed-loop system. Within the EB-PBC framework, the use of a quadratic potential on the special Euclidean group $SE(3)$ allowed [1] to derive the geometric impedance control presented by [7]. Yet, this development of passive controllers for rigid bodies is not based on any considerations of optimality.

Within the deep learning community, principles from optimal control are already used to optimize dynamic systems in the framework of neural ordinary differential equations (ODEs) [8,9]. Massaroli et al. also successfully implemented such principles for passive control, using the methodology of optimal potential energy shaping [10]. This allowed the use of a richer class than quadratic potentials, while guaranteeing asymptotic stability with a damping injection, thus effectively learning an optimal Lyapunov function of the dynamic system.

However, the optimization procedure presented by [10] only applies to systems whose configuration is represented on the Euclidean space $\mathbb{R}^n$. This is not the case for configurations represented on the special Euclidean Group $SE(3)$, a matrix Lie Group which has no trivial representation on a Euclidean space. Instead, $SE(3)$ can be split into four regions, each of which may be mapped into $\mathbb{R}^6$.

This work extends the optimal energy shaping methodology of [10] to rigid bodies evolving on $SE(3)$, thus combining optimal control principles with the passive control presented in [1]. The main objectives are to define a general class of EB-PBC controllers on $SE(3)$ and to optimize the rigid body dynamics with a controller

Yannik P. Wotte
RaM, University of Twente
E-mail: y.p.wotte@student.utwente.nl

Federico Califano
RaM, University of Twente
E-mail: f.califano@utwente.nl

Stefano Stramigioli
RaM, University of Twente
E-mail: s.stramigioli@utwente.nl

from this class. Methods from neural ODEs are applied for their proven effectiveness in this optimization context [10].

As related work, [11] applies deep learning on computation graphs involving Lie group transformations $SO(3)$, $SE(3)$ and $Sim(3)$, and show an effective workaround for connected numerical difficulties of automatic differentiation. Optimization of neural ODEs on manifolds is treated by Falorsi et al. [12] and Lou et al. [13], who do not specifically optimize their algorithms for Lie groups. Further, both works merely consider flows expressed on the tangent-bundle of a manifold, whereas the cotangent-bundle is equally essential for the Hamiltonian systems considered in this work. Neither approach is continued here, since specificity of algorithms for the case of $SE(3)$ was judged as more valuable: when a Lie Group is the manifold of interest, many steps have more computationally efficient expressions than for general manifolds.

This article heavily relies on the mathematical framework of Lie groups, in particular $SO(3)$ and $SE(3)$. That background is shown in Section 2. Here, the concept of representing the elements of $SO(3)$ and $SE(3)$ by means of vectors in $\mathbb{R}^n$ is also introduced, which is related to the notion of charts collected in an atlas. A minimal exponential atlas for $SE(3)$ is derived in Section 3, and it is shown how this atlas can be used to efficiently represent functions on $SE(3)$. Such a representation is essential for the definition of a potential energy on $SE(3)$, which is required to construct a control wrench for the rigid body dynamics presented in Section 4. Section 4 also sketches a proof of the stability for the thus constructed dynamics. The optimal control problem is defined in Section 5, which also treats its solution by means of iterative procedures from neural ODEs. The code that was created to solve this problem is presented in Section 6, where examples highlighting its performance on a state-regulation task are presented. Based on these results, Section 7 gives a discussion and Section 8 a conclusion.

2 Background

The special orthogonal group $SO(3)$ and the special euclidean group $SE(3)$ are matrix Lie groups that collect transformations of the Euclidean 3-space $\mathbb{R}^3$. $SO(3)$ can be described as the collection of rotations and $SE(3)$ as the collection of simultaneous rotations and translations. This enables $SE(3)$ to fully describe the pose of rigid bodies in $\mathbb{R}^3$. Define $SO(3)$, and consequently $SE(3)$ as the matrix Lie groups

$$SO(3) := \{R \in \mathbb{R}^{3\times 3} |\, R^T R = I, \det(R) = 1\}, \tag{1}$$

$$SE(3) := \{\begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4} |\, R \in SO(3), p \in \mathbb{R}^3\}, \tag{2}$$

in both cases using matrix composition as the group operation. $H_B^A \in SE(3)$ would represent the pose of a reference frame Ψ_B as seen from Ψ_A, while $H_A^B = {H_B^A}^{-1}$. The Lie algebras of $SO(3)$ and $SE(3)$ are the vector spaces $so(3)$ and $se(3)$, respectively:

$$so(3) := \{\tilde{\omega} \in \mathbb{R}^{3\times 3} |\, \tilde{\omega} = -\tilde{\omega}^T\}, \tag{3}$$

$$se(3) := \{\begin{bmatrix} \tilde{\omega} & v \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{4\times 4} |\, \tilde{\omega} \in so(3), v \in \mathbb{R}^3\}. \tag{4}$$

The vector space isomorphism $\sim: \mathbb{R}^3 \to so(3)$ is defined as

$$\sim \omega = \sim \begin{pmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{pmatrix} := \tilde{\omega} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \tag{5}$$

Here, $\sim$ is overloaded to also be used as $\sim: \mathbb{R}^6 \to se(3)$ for $T = \begin{pmatrix} \omega \\ v \end{pmatrix} \in \mathbb{R}^6$ with $\omega, v \in \mathbb{R}^3$, via

$$\sim T := \tilde{T} = \begin{bmatrix} \tilde{\omega} & v \\ 0 & 0 \end{bmatrix}. \tag{6}$$

Twists $\tilde{T} \in se(3)$ appear as the left and right translated change-rates of $H_B^A \in SE(3)$ by

$$\tilde{T}_B^{B,A} = H_A^B \dot{H}_B^A, \qquad \tilde{T}_B^{A,A} = \dot{H}_B^A H_A^B. \tag{7}$$

Both $\tilde{T}_B^{B,A}$ and $\tilde{T}_B^{A,A}$ represent the change of pose of frame Ψ_B with respect to Ψ_A, but $\tilde{T}_B^{B,A}$ is expressed in Ψ_B while $\tilde{T}_B^{A,A}$ is expressed in Ψ_A. The algebra and group adjoint are defined, respectively, as the 6 by 6 matrices:

$$ad_T = \begin{bmatrix} \tilde{\omega} & 0 \\ \tilde{v} & \tilde{\omega} \end{bmatrix}, \qquad Ad_H = \begin{bmatrix} R & 0 \\ \tilde{p}R & R \end{bmatrix}, \tag{8}$$

with $T = \begin{pmatrix} \omega \\ v \end{pmatrix}$, $H = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} \in SE(3)$. The group adjoint is used to transform twists between frames of reference, e.g. $T_B^{A,A} = Ad_{H_B^A} T_B^{B,A}$, while the algebra adjoint ad_T is the rate of change of the group adjoint Ad_H at the identity, i.e. $ad_T = \dot{Ad}_H$ for $H = I$ and $\dot{H} = \tilde{T}$.

Twists also appear in the exponential map, which presents an essential, nearly-global diffeomorphism for SO(3) & SE(3) that allows generating $R \in SO(3)$ from

$\tilde{\omega} \in so(3)$ via (9) and $H \in SE(3)$ from $\tilde{T} \in se(3)$ via (10):

$$e^{\tilde{\omega}} = \sum_{i=0}^{\infty} \frac{1}{i!}\tilde{\omega}^i = I + \sin(\theta)\tilde{\hat{\omega}} + (1-\cos(\theta))\tilde{\hat{\omega}}^2\,, \tag{9}$$

$$e^{\tilde{T}} = \sum_{i=0}^{\infty} \frac{1}{i!}\tilde{T}^i = \begin{bmatrix} e^{\tilde{\omega}} & \frac{1}{\theta^2}(I-e^{\tilde{\omega}})\tilde{\omega}v + \omega^T v\omega \\ 0 & 1 \end{bmatrix}\,, \tag{10}$$

with $\theta = \|\omega\|_2 = \sqrt{\omega^T\omega}$ and $\tilde{\hat{\omega}} = \tilde{\omega}/\theta$.

For $\theta < \pi$ their inverses are presented in equations (11) and (12), respectively: the log map for $SO(3)$ is

$$\begin{cases} \log(R) = \cos^{-1}(\frac{1}{2}(\mathrm{Tr}(R)-1))\frac{A}{\|A\|} & R \neq I \\ \log(R) = \log(I) = 0_{3\times 3} & \text{otherwise}\,, \end{cases} \tag{11}$$

with $A = \frac{1}{2}(R - R^T)$ the anti-symmetric part of R, while $\|A\| := \sqrt{-\frac{1}{2}\mathrm{Tr}(A^2)}$.
Denoting $\tilde{\omega} = \log(R)$, the log map for $SE(3)$ is

$$\log(\begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}) = \begin{bmatrix} \tilde{\omega} & Qp \\ 0 & 0 \end{bmatrix}\,, \tag{12}$$

$$Q = I - \frac{1}{2}\tilde{\omega} + \frac{2\sin(\theta) - \theta(1+\cos(\theta))}{2\theta^2\sin(\theta)}\tilde{\omega}^2\,. \tag{13}$$

Since $\lim_{\theta\to 0} Q = I$, a well-defined Q is given by (13) regardless of R, such that the logarithm on $SE(3)$ (12) has the range of validity of the logarithm on $SO(3)$ (11), bounded only by the rotational part.

Last, the concept of an atlas is essential in the remainder of this article. An atlas $\mathcal{A}$ for an n-dimensional manifold $\mathcal{M}$ is a collection of charts (U_i, x_i), where $U_i \subseteq \mathcal{M}$ is an open subset of $\mathcal{M}$ that determines the range of validity for the invertible chart-map $x_i : \mathcal{M} \to \mathbb{R}^n$. The chart map x_i is said to chart U_i into coordinates on $\mathbb{R}^n$. *Open* means, for the level of abstraction needed in this work, that the boundaries of the region U_i are not part of U_i. It is also required that the chart regions collected in the Atlas cover the manifold $\mathcal{M}$, i.e. $\bigcup U_i = \mathcal{M}$.

For example, with $\mathcal{M} = \mathbb{R}^2$ the scenario in Figure 1 would feature open regions $U_1 \subset \mathbb{R}^2$ and $U_2 \subset \mathbb{R}^2$, which are then charted into separate coordinates on $\mathbb{R}^2$ by maps x_1 and x_2. The power of this concept is that the manifold $\mathcal{M}$ could have any of a wide variety of topologies, not necessarily Euclidean, while an atlas would still provide coordinates in a Euclidean space. Concepts like differentiability are also defined in such coordinates, where e.g. a smooth atlas would have smooth transition functions $x_2 \circ x_1$, ref. Figure 1.

Additionally, a particular type of Atlas called a minimal Atlas is used in this article, which is defined as using the minimum number of possible charts such that the manifold $\mathcal{M}$ is still covered by the open sets U_i.

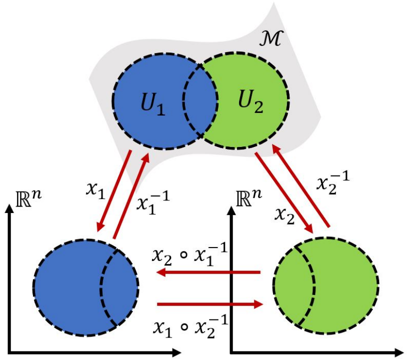


Fig. 1: Example of two charts $(U_1, x_1), (U_2, x_2) \in \mathcal{A}$

3 Defining a Potential on SE(3)

Section 3.1 defines a minimal exponential atlas for $SE(3)$. Section 3.2 shows how transitions between charts work and Section 3.3 shows how general functions can be defined on $SE(3)$ by using an atlas.

3.1 Minimal Atlas

Defining a smooth potential function $V : SE(3) \to \mathbb{R}$ in charts of the Lie-Group $SE(3)$ is an alternative to defining such a function on $\mathbb{R}^{4\times 4}$ and restricting its inputs to $SE(3) \subset \mathbb{R}^{4\times 4}$. It was noted by [11] that the latter definition leads to numerical issues, since the embedding of $SE(3)$ in $\mathbb{R}^{4\times 4}$ is problematic. To keep the definition in charts compact, charts be chosen from a minimal atlas.

Such a minimal atlas $\mathcal{A}_{\min}$ requires four charts in the case of $SO(3)$ and consequentially also in the case of $SE(3)$. Grafarend et al. [14] derive examples of such a minimal atlas using Euler / Cardan angles. They draw on the result that the Lusternik–Schnirelmann category of $SO(3)$, which gives the minimum number of open contractible covers of a manifold, is 4.

Here, chart coordinates are constructed from logarithmic maps (11) and (12), which will prove to be computationally desirable, especially since they feature simple inverse functions and analytic derivatives. Define $R_\omega := e^{\tilde{\omega}}$, then a minimal atlas for $SO(3)$ is given by

$$\mathcal{A}_{\min}^{SO(3)} := \{(U_\omega, x_\omega) \,|\, \omega \in \Omega\}\,, \tag{14}$$

$$U_\omega := \{R_\omega e^{\tilde{\omega}'} \,|\, \omega' \in \mathbb{R}^3, |\omega'| < \pi\}\,, \tag{15}$$

$$x_\omega(R) := \sim^{-1}\log(R_\omega^T R)\,, \tag{16}$$

$$x_\omega^{-1}(\omega') = R_\omega e^{\tilde{\omega}'}\,. \tag{17}$$

Where the admissible ω are collected in the set Ω:

$$\Omega = \left\{ \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \pi \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ \pi \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ \pi \end{pmatrix} \right\}. \tag{18}$$

For brevity, denote these four elements as ω_i with $i \in \{0, 1, 2, 3\}$. The chart maps x_{ω_i} are are chosen such that the zero of each chart corresponds to the orientations given by $R_0 = \mathrm{diag}(1, 1, 1)$, $R_1 = \mathrm{diag}(1, -1, -1)$, $R_2 = \mathrm{diag}(-1, 1, -1)$ and $R_3 = \mathrm{diag}(-1, -1, 1)$. This is also highlighted in Figure 2. The open set U_{ω_i} contains all orientations that are reachable from R_i by a rotation through an angle less than π.

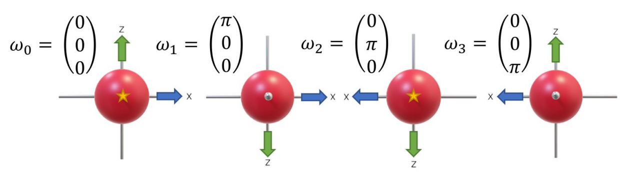


Fig. 2: Orientations corresponding to the zeros of the four charts in $\mathcal{A}_{\min}^{SO(3)}$

Define $H_\omega := \begin{bmatrix} R_\omega & 0 \\ 0 & 1 \end{bmatrix}$, then $\mathcal{A}_{\min}^{SE(3)}$ follows as

$$\mathcal{A}_{\min}^{SE(3)} := \{(\mathcal{U}_a, \mathcal{X}_a) | (U_a, x_a) \in \mathcal{A}_{\min}^{SO(3)}\}, \tag{19}$$

$$\mathcal{U}_\omega := \{ \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix} | R \in U_\omega, p \in \mathbb{R}^3 \}, \tag{20}$$

$$\mathcal{X}_\omega(H) := \sim^{-1} \log(H_\omega^{-1} H), \tag{21}$$

$$\mathcal{X}_\omega^{-1}(T) = H_\omega e^{\tilde{T}}. \tag{22}$$

The intuition for the open regions $\mathcal{U}_i$ of $\mathcal{A}_{\min}^{SE(3)}$ is the same as for $\mathcal{A}_{\min}^{SO(3)}$, and the chart-maps have the same corresponding configurations for their respective zeros as they had for $\mathcal{A}_{\min}^{SO(3)}$, just that translations are also represented in the chart-coordinates. A proof that $\mathcal{A}_{\min}^{SO(3)}$ and $\mathcal{A}_{\min}^{SE(3)}$ indeed constitute atlases can be found in the appendix 9.

Remark 1 The atlas (19) uses twists $q_j = \mathcal{X}_j(H)$ as coordinates for $H \in SE(3)$. Thus, also expressions such as ad_{q_j} are well defined.

3.2 Chart Transitions

Due to strong similarity, chart-transitions are treated for $SE(3)$ only. To translate coordinates $q_j \in \mathbb{R}^6$ for $\mathcal{X}_j^{-1}(q_j) \in \mathcal{U}_j \cap \mathcal{U}_i$ from a chart $(\mathcal{U}_j, \mathcal{X}_j)$ to $(\mathcal{U}_i, \mathcal{X}_i)$, the chart-transition functions are

$$\begin{aligned} \mathcal{X}_{ij}(q_j) &= \mathcal{X}_i \circ \mathcal{X}_j^{-1}(q_j) \\ &= \sim^{-1} \log(\begin{bmatrix} R_{\omega_i}^T R_{\omega_j} & 0 \\ 0 & 1 \end{bmatrix} e^{\tilde{q}_j}). \end{aligned} \tag{23}$$

In order to find the transitions between the rates of change $\dot{q}_j$, the Lie group structure of $SE(3)$ together with the exponential map provides a computationally simple relation. Denote by H_B^0 the pose of a frame Ψ_B seen from a reference Ψ_0, and let $q_j = \mathcal{X}_j(H_B^0)$. Note that this is exactly how the chart-coordinates q_j are usually constructed. Then any $\dot{q}_j$ is related to a twist $T_B^{B,0} = \sim^{-1} H_0^B \dot{H}_B^0$ via the derivative of the exponential map $K(q_j) \in \mathbb{R}^{6 \times 6}$ [15]:

$$T_B^{B,0} = K(q_j) \dot{q}_j. \tag{24}$$

Because (24) holds for all j, the transition between rates of change then follows as:

$$\dot{q}_i = K(q_i)^{-1} K(q_j) \dot{q}_j. \tag{25}$$

This also provides a transition between co-vectors p_j with the inverse transpose of that relation, as follows from their defining property $p_i^T \dot{q}_i = p_j^T \dot{q}_j$:

$$p_i = K(q_i)^T K(q_j)^{-T} p_j. \tag{26}$$

This fact that vectors are pushed forward (25) while co-vectors are pulled back (26) is also highlighted in Figure 3.

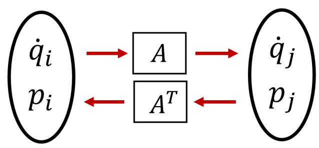


Fig. 3: The transformations of vectors and covectors, with $A = K(q_i)^{-1} K(q_j)$

Generally, the derivative of the exponential map is

$$K(q_i) = \frac{1 - e^{-ad_{q_i}}}{ad_{q_i}} := \sum_k^\infty \frac{(-1)^k}{(k+1)!} \mathrm{ad}_{q_j}^k, \tag{27}$$

but the evaluation of this infinite sum can be avoided by use of the Cayley-Hamilton theorem, which provides an exact matrix function counter-part to any smooth real function, in $(n-1)$ terms for an n-by-n matrix input [16]. Eade E. [17] further simplified this expression for $SE(3)$ and also derived the closed form of the inverse of $K(q_j)$, which strongly decreases the computation cost of such chart-transitions.

It later becomes important that $K(\begin{pmatrix}\omega\\v\end{pmatrix})$ is singular for $\|\omega\|_2 = 2\pi$, but not at the chart-boundaries $\|\omega\|_2 = \pi$:

$$\det(K(\begin{pmatrix}\omega\\v\end{pmatrix})) = 0 \quad \text{iff} \quad \|\omega\|_2 = 2k\pi, k \in \mathbb{N}_+ . \tag{28}$$

A derivation of (28) is shown in the appendix 9.2.

3.3 Function Definition

Denote as $C^\infty(\mathcal{M}, \mathbb{R})$ the collection of all smooth functions $F : \mathcal{M} \to \mathbb{R}$. Given a finite atlas, any $F \in C^\infty(SE(3), \mathbb{R})$ can be represented as the weighted sum of chart-components $F_i \in C^\infty(\mathbb{R}^6, \mathbb{R})$ using a smooth partition of unity with partition functions $\sigma_i \in C^\infty(SE(3), \mathbb{R})$:

$$F(H) = \sum_i \sigma_i(H) F_i(\mathcal{X}_i(H)) . \tag{29}$$

The σ_i satisfy

$$\begin{cases} \sigma_i(H) > 0 & H \in \mathcal{U}_i \\ \sigma_i(H) = 0 & \text{otherwise}, \end{cases} \tag{30}$$

$$\forall H \in SE(3) : \sum_i \sigma_i(H) = 1 , \tag{31}$$

$$\forall H \in SE(3) : \big(\sigma_i(H) = 0 \to \forall n : D^n \sigma_i(H) = 0\big) \tag{32}$$

As shown in [18], such a smooth partition of unity is guaranteed to exist on any smooth manifold. Constructing smooth functions on manifolds as in (29) is presented in much more detail in [19]. One option for the charts in $\mathcal{A}^{SE(3)}_{\text{min}}$ is given by

$$\sigma_i(H) = e^{-s_i(H)^{-1}} / \sum_j e^{-s_j(H)^{-1}} , \tag{33}$$

$$s_i(H) = \text{Tr}(H^T_{\omega_i} H)/4 .$$

Here, the s_i also constitute a smooth partition of unity, that however at most guarantees to reach all functions $V \in C^2(SE(3))$ via the construction in (29), since not all derivatives of the s_i in the i-th chart approach 0 as the chart-boundary is approached.

4 Dynamics & Control

Section 4.1 defines the closed-loop dynamics that are to be optimized, Section 4.2 defines a general class of EB-PBC controllers on $SE(3)$ and Section 4.3 provides an intuitive proof of stability for this class. Section 4.4 highlights the possible use of a potential on $SE(3)$ for the control of general manipulators.

4.1 Dynamics

The trajectory of a rigid body in Euclidean 3D space is fully described by the function $H^0_b : \mathbb{R} \to SE(3)$ that gives the relative position and orientation of a frame Ψ_b attached to the rigid body with respect to another frame Ψ_0, as a function of time. The twist of the body with respect to Ψ_0, expressed in the body frame Ψ_b is

$$T^{b,0}_b = \sim^{-1} (H^b_0 \dot{H}^0_b) . \tag{34}$$

If the charts collected in (19) are used to describe $H^b_0 = \mathcal{X}^{-1}_i(q_i)$ in coordinates q_i, then their change-rate follows directly from (24)

$$\dot{q}_i = K(q_i)^{-1} T^{b,0}_b . \tag{35}$$

The use of (35) already necessitates chart transitions, since $K(q_i)$ inevitably has singularities as given by (28), in general scenario. Without loss of generality, let the body frame diagonalize the body's inertia tensor $\mathcal{I} \in \mathbb{R}^{6\times 6}$. Then the momentum of that rigid body is $P^b = \mathcal{I} T^{b,0}_b$ and a derivation e.g. via Newton's laws shows that the dynamics of a rigid body follow as

$$\dot{P}^b = ad^T_{T^{b,0}_b} P^b + W^b , \tag{36}$$

where W^b is the external wrench exerted on the body, likewise expressed in the body-frame. The combined system of equations given by (35) and (36) is also presented in [15].

4.2 Control

The external wrench (37) is constructed as a sum of a potential gradient term W^b_V and a damping term W^b_D. That is why it is often referred to as energy shaping with a damping injection.

$$W^b = W^b_V + W^b_D . \tag{37}$$

In this form, a potential gradient term of a function $V \in C^\infty(SE(3), \mathbb{R})$ with gradient calculated in the i-th chart enters (36) as

$$W^b_V = -K(q_i)^{-T} \frac{\partial V}{\partial q_i} = -dV . \tag{38}$$

The reason for the previous expression (38) is that the gradient $\frac{\partial V}{\partial q_i}$ transforms as a covector, hence transforming to the body-frame via the inverse-transpose of the derivative of the exponential map. This is the dual of the transformation in equation (24). While this is a geometrical fact, this transformation rule is also seen from the coordinate invariant change-rate of V given in (39), which shows that if $\dot{q}_i$ transforms as a vector, then $(\frac{\partial V}{\partial q_i})$ must transform as a covector.

$$\dot{V} = (\frac{\partial V}{\partial q_i})^T \dot{q}_i = dV^T T_b^{b,0} \,. \tag{39}$$

Thus, if the potential is expressed as a sum of chart components V_i using equation (29)

$$V(H_b^0) = \sum_i \sigma_i(H_b^0) V_i(\mathcal{X}_i(H_b^0)) \,, \tag{40}$$

then gradients can be calculated component wise in their respective charts and translated back to the body frame, yielding

$$W_V^b = -\sum_i K(q_i)^{-T} \frac{\partial}{\partial q_i}(\sigma_i V_i) = -dV \,. \tag{41}$$

Viscous damping takes the form

$$W_D^b = -B(H_b^0, P^b) P^b \,, \tag{42}$$

with $B(H_b^0, P^b) \in \mathbb{R}^{6\times 6}$ a symmetric and positive definite matrix. The damping matrix can be defined directly in the body-frame, with chart-components $B_i(q_i, P^b)$ likewise summed using (29):

$$B(H_b^0, P) = \sum_i \sigma_i(H_b^0) B_i(\mathcal{X}_i(H_b^0), P^b) \tag{43}$$

Remark 2 In this context, Rashad et al. [1] make the popular yet very particular choice of a constant $B(H, P)$ for the damping injection, while the potential $V(H_b^0)$ was defined directly on $SE(3)$ with a quadratic dependence on translations and a nearly quadratic dependence on rotations. The gradient of their potential was derived via a variational approach.

4.3 Stability

From a geometric perspective, passivity of the system (36) is trivially guaranteed for all sufficiently smooth potential functions V with a finite lower bound, as was proven in [10]. This also applies here, since $SE(3)$ can be embedded in $\mathbb{R}^{12}$ via Whitney's embedding theorem [20]. However, stability can also be seen from the power continuity guaranteed by the energetic structure of the system: the rate of change of the system's kinetic energy $E_{kin} = \frac{1}{2} {P^b}^T \mathcal{I}^{-1} P^b$ is equal to the negative rate of change of potential energy V:

$$\begin{aligned} \dot{E}_{kin} &= (\dot{P}^b)^T \mathcal{I}^{-1} P^b = (\dot{P}^b)^T T_b^{b,0} \\ &= (ad^T_{T_b^{b,0}} P^b - dV)^T T_b^{b,0} \\ &= (-dV)^T T_b^{b,0} = -\dot{V} \,. \end{aligned} \tag{44}$$

Here, the second equality uses that by definition $P^b = \mathcal{I} T_b^{b,0}$. The third equality substitutes (36) with $W^b = -dV$ and dV as in (41). The fourth equality uses the fact that $(ad^T_{T_b^{b,0}} P^b)^T T_b^{b,0} = {P^b}^T (ad_{T_b^{b,0}} T_b^{b,0}) = 0$ since $\forall q \in \mathbb{R}^6 : ad_q q = 0$.

To guarantee asymptotic stability of any strict minimum of V, a positive-definite, symmetric damping matrix is added in the form of non-linear, viscous damping by using W^b as defined in (37):

$$\dot{E}_{kin} = -\dot{V} - (P^b)^T B T_b^{b,0} \leq -\dot{V} \tag{45}$$

Hence, the energy of the system $E = E_{kin} + V$ always has a change-rate $\dot{E} \leq 0$, guaranteeing that an equilibrium $P^b = 0$, $dV = 0$ is approached as long as V is smooth and lower-bounded, and $B \geq 0$. These low requirements to V and B mean that the optimization power of neural nets can be used to find an optimal controller with guaranteed stability, by directly learning a potential function and positive definite damping.

4.4 Extension to general manipulators

Let the configuration manifold of a general manipulator be the smooth, n-dimensional Riemannian manifold $\mathcal{M}$ equipped with metric $\mathcal{I}_p$ given by the inertia tensor of the manipulator and $p \in \mathcal{M}$ corresponding to the coordinate-free configuration of the manipulator. The coordinate-free dynamics read [21]

$$\nabla_{\gamma'} \gamma' + \nabla \mathcal{V} = 0 \,. \tag{46}$$

In (46), the evolution of the trajectory $\gamma : \mathbb{R} \to \mathcal{M}$ is determined by the coordinate-free acceleration $\nabla_{\gamma'}\gamma'$ along the tangent vector field $\gamma' \in T_{\gamma(t)}\mathcal{M}$. Here $\nabla_{\cdot}\cdot$ is the Levi-Civita connection induced by $\mathcal{I}_p$, and ∇V is the coordinate-free gradient of a potential $\mathcal{V} : \mathcal{M} \to \mathbb{R}$.

Given the choice of a chart $\Phi : \mathcal{U} \subset \mathcal{M} \to \mathbb{R}^n$, $d\Phi : T_p\mathcal{U} \to \mathbb{R}^n$, the dynamics (46) may be written into coordinates q, $\dot{q}$ as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + \frac{\partial V}{\partial q} = 0 \,. \tag{47}$$

To use the gradient of a potential on $SE(3)$, define the direct kinematics $f : \mathbb{R}^n \to SE(3)$ relating joint-angles q to a desired frame $H_{\mathrm{D}}^0 \in SE(3)$, for whose task-space a potential should be defined:

$$f(q) = H_{\mathrm{D}}^0 \,. \tag{48}$$

Note that the geometric Jacobian $J(q)$ is the push-forward f_* of this map from the joint-velocities $\dot{q}$ to the twist $T_D^{0,0} \in se(3)$, where $T_D^{D,0} = Ad_{H_0^D} T_D^{0,0}$ is dual to the previously constructed gradient dV of a smooth potential $V(H_D^0)$.

Likewise, the transpose of the geometric Jacobian $J^T(q)$ acts as the pull-back f^* from the co-vectors in $se^*(3)$ to those dual to $\dot{q}$. Given $V : SE(3) \to \mathbb{R}$ and gradient $dV \in se^*(3)$, this gives a natural map of the task-space gradient to joint-torques:

$$\frac{\partial V}{\partial q} = J(q)^T Ad^T_{H_D^0} dV \,. \tag{49}$$

Thus, task-space potentials can also be applied to general manipulators in a well-defined manner. Since $\frac{\partial V}{\partial q}$ in (49) follows from a bounded potential, stability is guaranteed in the same manner as in Section 4.3, and for the same conditions.

Further extensions find themselves in the use of more involved potential functions $V : SE(3) \times ... \times SE(3) \to \mathbb{R}$, or simpler sums of potential gradient terms pulled back from different frames.

If multiple charts are required for the configuration manifold $\mathcal{M}$, then particular care has to be taken in the optimization procedure, with the appropriate transformations of the co-state at chart-transitions, as will become clear in Section 5. The key-step would be to define the direct kinematics (48) as $f : \mathcal{M} \to SE(3)$, to express a globally defined pull-back $f*$ in the appropriate charts. While the detailed definition of such a general scenario goes beyond the scope of this article, it is briefly considered in the Discussion 8, with focus on the numerical difficulties described in 5.1.

5 Optimization on SE(3)

This Section deals with optimization of a cost of the type $J(H_b^0, P^b, W_\theta^b) \in \mathbb{R}$, which represents a performance-metric for a given task. The optimization is to be performed with respect to the n parameters collected in θ, where $H_b^0 \in SE(3)$, $P^b, W_\theta^b \in se^*(3)$ are functions of time and W_θ^b additionally depends on θ:

$$\begin{aligned} J(H_b^0, P^b, W_\theta^b) = & E(H_b^0(T), P^b(T), W_\theta^b(T)) \\ & + \int_0^T L(H_b^0(t), P^b(t), W_\theta^b(t)) \mathrm{d}t \,. \end{aligned} \tag{50}$$

Here, E is a terminal cost at time $t = T$ and L is a running cost term. Minimization of (50) occurs over a given distribution of initial values $\mathbb{P}(x_{i,0})$. In addition to the usual dynamic constraint and a constrained forcing term, the problem is also subject to a chart-transition constraints determining the current chart i:

$$\begin{aligned} & \min_\theta \mathbb{E}_{x_{i,0} \sim \mathbb{P}(x_{i,0})} [J(H_b^0, P^b, W_\theta^b)] \,, \\ & \quad \text{s.t.} \\ & \quad \dot{x}_i = f_i(x_i, W_\theta^b) \,, \qquad x_i(0) = x_{i,0} \,, \\ & \quad x_i = \begin{pmatrix} q_i \\ P^b \end{pmatrix} \,, \qquad H_b^0 = \mathcal{X}_i^{-1}(q_i) \,, \\ & \quad W_\theta = -B_\theta(H, P)P - K(q_i)^{-T} \frac{\partial V_\theta}{\partial q_i} \,, \\ & \quad B_\theta = B_\theta^T \geq 0 \,, \\ & \quad i = \arg\max_i (\sigma_i(\mathcal{X}_i^{-1}(q_i))) \,. \end{aligned} \tag{51}$$

The chart i needs to be taken into account in order for (59) to be well-defined for all, as $\frac{\partial \mathcal{H}_i}{\partial q_i}$ requires kinematics (35), which are subject to chart-transitions. Then transformations between charts occur as described in Section 3.2, but note that the momentum P^b is always defined in the same reference frame, it is the dual quantity to $T_b^{b,0} = \mathcal{I}^{-1} P^b$. The function f_i is given by

$$f_i(x_i, W_\theta) = \begin{pmatrix} K(q_i)^{-1} T_b^{b,0} \\ ad^T_{T_b^{b,0}} P^b + W_\theta^b \end{pmatrix} \,. \tag{52}$$

In order to perform the optimization (51), stochastic gradient descent is used [22], where the parameters θ_{k+1} at the $(k+1)$-th iteration follow from the optimal parameters at the k-th iteration by the update rule

$$\theta_{k+1} = \theta_k - \frac{\eta_k}{N} \eta \sum_{i=0}^{N} \frac{\partial}{\partial \theta_k} J(H_{b,i}^0, P_i^b, W_{\theta_k}^b) \,. \tag{53}$$

Here, N gives the batch-size and η_k is the positive scalar learning rate.

To find the sensitivities $\frac{\partial J}{\partial \theta}$ for this scenario, the generalized adjoint sensitivity [9] may be derived via optimal control theory applied [23] in charts. Specifically, Pontryiagin's minimum principle is used [23]. To apply it, define the cost evaluated in chart i as

$$J_i(x_i, W_\theta^b) = J(\mathcal{X}_i^{-1}(q_i), P^b, W_\theta^b) \,. \tag{54}$$

Then define Lagrange multiplier functions $\lambda_i(t) \in \mathbb{R}^1 2$, subsequently called co-states, to define the i-th augmented running cost as

$$\mathcal{L}_i = \lambda_i^T \big(f_i(x_i, W_\theta^b) - \dot{x}_i \big) + L_i(x_i, W_\theta^b) \,, \tag{55}$$

where L_i is given by

$$L_i(x_i, W_\theta^b) = L(\mathcal{X}_i^{-1}(q_i), P^b, W_\theta^b) \,. \tag{56}$$

Given this augmented running cost, the so-called Hamiltonian $\mathcal{H}_i$ follows as $\mathcal{H}_i = \mathcal{L}_i - \dot{y}_i^T \frac{\partial \mathcal{L}_i}{\partial \dot{y}_i}$, with $y_i = \left(\begin{smallmatrix} x_i \\ \theta \end{smallmatrix} \right)$:

$$\mathcal{H}_i = \lambda_i^T f_i(x_i, W_\theta^b) + L_i(q_i, P^b, W_\theta^b) \,. \tag{57}$$

Now, the minimum principle [23] states that the optimal input W_θ^b minimizes the Hamiltonian (57) when the state x_i and co-state λ_i evolve according to (58) and (59), which allows to find the gradient $\frac{\partial J}{\partial \theta}$ with (60):

$$\dot{x}_i = \frac{\partial \mathcal{H}_i}{\partial \lambda} = f_i(x_i, W_\theta^b)\,, \tag{58}$$

$$\dot{\lambda}_i = -\frac{\partial \mathcal{H}_i}{\partial x_i} = -(\frac{\partial f_i}{\partial x_i})^T \lambda_i - \frac{\partial L_i}{\partial x_i}\,, \tag{59}$$

$$\frac{\partial J}{\partial \theta} = \frac{\partial E}{\partial \theta} + \int_0^T \frac{\partial}{\partial \theta}\big(\lambda_i^T f_i(x_i, W_\theta^b) + L_i(x_i, W_\theta^b)\big)\mathrm{d}t\,. \tag{60}$$

Boundary conditions are $x_i(0) = x_{i,0}$, $\lambda_i(T) = \frac{\partial E}{\partial x_i(T)}$, so the dynamics of λ_i (59) need to be calculated backwards in time. For any required chart-transitions in (58), switches also need to be taken into account in (59) and (60). To find the chart transition rules for λ_i, note that λ_i and $\dot{x}_i$ are dual quantities ($\lambda_i^T \dot{x}_i = \lambda_j^T \dot{x}_j$) and thus transform in a dual fashion from chart i to chart j. The transformation rule for the $\dot{x}_i$ is

$$\dot{x}_i = A(x_j)\dot{x}_j\,, \quad A(x_j) = \begin{bmatrix} K(q_i)^{-1}K(q_j) & 0 \\ 0 & I_{6\times 6} \end{bmatrix}\,, \tag{61}$$

such that the co-state λ_i transforms as

$$\lambda_i = A(x_j)^{-T}\lambda_j\,. \tag{62}$$

Summarizing, the optimization (51) can be performed with stochastic gradient descent (53), where the gradient is calculated using (58) to (60), in which chart-transitions occur by means of (61) and (62).

Remark 3 Since $q_j = \mathcal{X}_{ji}(q_i)$, one has that $L_i(x_i, W_\theta) = L_j(x_j, W_\theta)$. Because $\lambda_i^T \dot{x}_i = \lambda_j^T \dot{x}_j$, one also has $\lambda_i^T f_i(x_i, W_\theta) = \lambda_j^T f_j(x_j, W_\theta)$, giving $\mathcal{L}_i = \mathcal{L}_j$ and $\mathcal{H}_i = \mathcal{H}_j$. Hence, a chart-switch is indeed well-defined and the same cost (50) is optimized at any given point, irrespective of the current chart i.

Remark 4 The symmetric, positive-definite condition of B_θ will be implicitly enforced in the optimization, by choosing a diagonal B_θ with positive elements. This is possible without loss of generality, since an optimal B_θ depending on the full state x_i could still cover all cross-terms that would otherwise be present. [10]

5.1 Numerical Considerations

Teed et al. [11] point out that Pytorch's automatic differentiation `Autograd` is not suited for evaluating derivatives of the exponential and logarithmic maps on $SE(3)$. `Autograd` likewise makes large numerical errors when differentiating the derivative $K(q)$ of the exponential map, which becomes necessary in (59). The work-around was to manually construct this second derivative and parts of (59) where the exponential map is differentiated (i.e. in evaluating $\frac{\partial B}{\partial q}$. This is worthwhile, since $K(q)$ plays an essential role in the optimization of ODE's on $SE(3)$ and since the manual differentiation of other terms also always has the same structure.

Differentiating the $B(H,P)$ with respect to the q_i is strongly analogous to the method in (41), so this is only shown in the appendix 9.3. To avoid clutter, the expanded form of the adjoint dynamics (59) is likewise only in the appendix 9.3. In the following, only the differentiation of $K(q)$ is shown.

Denoting $q = (\omega, v)^T$ and $\|\omega_i\|_2 = \theta$, the derivative $K(q)$ of the exponential map is found via the Caley-Hamilton theorem [16]:

$$K(q) = \sum_{i=0}^{5} a_i(\theta) ad_q^i\,. \tag{63}$$

This gives the derivative

$$\dot{K}(q) = \sum_{i=0}^{5} \dot{a}_i(\theta)(ad_q)^i + a_i(\theta)\frac{d}{dt}(ad_q)^i\,, \tag{64}$$

and the derivative of the inverse is found as

$$\dot{K}^{-1}(q) = K^{-1}(q)\dot{K}(q)K^{-1}(q). \tag{65}$$

The functions $a_i(\theta)$ and their derivatives are given in the appendix 9.3. Note also, that the i-th partial derivative of $K(q)$ is found by setting $\dot{q}_i$ equal to the $i-th$ unit vector when computing $\dot{K}(q)$. Further, since the transpose commutes with the time-derivative, the derivatives of $K(q)^T$ are not treated separately.

6 Training

In Section 6.1 an overview of the code and its dependencies is given and in Section 6.2 the training of a quadratic controller alike that of [1] is set up to be optimized in the presented framework, for a state-regulation task. The results of this training are shown in Section 6.3.

6.1 Code Infrastructure

This code is available at `github.com/ShapingSE3`. It is conceptually divided into higher and lower level functionality.

On a high level, the libraries `pytorch`, `pytorch-lightning` and `wandb` are used for streamlined optimization and tracking of training progress. On a lower level, the library `torchdyn` [24] adds functionality for training of neural ODEs, and custom changes were made to the `sensitivity`, `odeint_hybrid` and `ODE_problem`, which were readily included with `torchdyn` to allow training of neural ODEs when chart-switches occur. At the lowest level, all required functions for charting, chart-switches and computation of gradients and functions on $SE(3)$, as well as the partition of unity were written for single inputs in the custom library `potential_shaping_SE3` using exclusively `pytorch` functions. Functions in `potential_shaping_SE3` were written to allow use of `functorch/vmap` to efficiently batch inputs, which facilitates efficient training with arbitrary batch-sizes.

6.2 Experimental Setup

The optimization performed is largely equivalent to that given in Section 5. A different control law for the wrench W^b is used, namely the quadratic potential controller of [1] in a setting of motion control.

Let the dynamics be given by (36), with the pose H^0_b of the body's center-of-mass-frame Ψ_b as seen from an inertial reference Ψ_0, and the inertia tensor $\mathcal{I} = \mathrm{diag}(0.01, 0.01, 0.01, 1, 1, 1)$. Here, the body's center of mass is controlled to reach the inertial position $H^0_d = I$. The control law of [1] in this setting of motion control is:

$$W^b_\theta = W^b_V + W^b_D \,, \tag{66}$$

$$W^b_V = \begin{pmatrix} \tau^b_V \\ f^b_V \end{pmatrix} , \tag{67}$$

$$\tilde{\tau}^b_V = -2\mathrm{sk}(G_0 R^d_b) - sk(G_t R^b_d \tilde{p}^d_b \tilde{p}^d_b R^d_b) \,, \tag{68}$$

$$\tilde{f}^b_V = -R^b_d \mathrm{sk}(G_t \tilde{p}^d_b) R^d_b - \mathrm{sk}(G_t R^b_d \tilde{p}^d_b R^d_b) \,, \tag{69}$$

$$W^b_D = -K_D T^{B,0}_B \,. \tag{70}$$

Here, $\mathrm{sk}(A) = 1/2(A - A^T)$ denotes the skew symmetric part of a matrix A. The matrices $G_0, G_t \in \mathbb{R}^{3\times 3}$ and $K_D \in \mathbb{R}^{6\times 6}$ are taken diagonal, where each diagonal element is determined by e^θ_i. The 12 parameters θ_i are then learned during the optimization described by

$$\begin{aligned}
&\min_\theta \mathbb{E}_{x_{i,0}\sim\mathbb{P}(x_{i,0})}[J(H^0_b, P^b, W^b_\theta)] \,, \\
&\quad \text{s.t.} \\
&\quad \dot{x}_i = f_i(x_i, W^b_\theta) \,, \qquad x_i(0) = x_{i,0} \,, \\
&\quad x_i = \begin{pmatrix} q_i \\ P^b \end{pmatrix} , \qquad H^0_b = \mathcal{X}_i^{-1}(q_i) \,, \\
&\quad K_d = K_d^T \geq 0 \,, \quad G_0 = G_0^T \geq 0 \,, \quad G_t = G_t^T \geq 0 \,, \\
&\quad i = \arg\max_i (\sigma_i(\mathcal{X}_i^{-1}(q_i))) \,.
\end{aligned}$$

Here, the distribution $\mathbb{P}(x_{i,0})$ uniformly samples

$$\theta \sim [0, \pi] \,, \tag{71}$$

$$d \sim [0, 1] \,, \tag{72}$$

$$\theta_p \sim [0, 0.03] \,, \tag{73}$$

$$d_p \sim [0, 1] \,. \tag{74}$$

Then, ω, v, ω_p, v_p are sampled from the normal distribution $\mathcal{N}(\mu, \sigma^2)$ with standard deviation $\mu = (0, 0, 0)^T$ and variance $\sigma^2 = I_{3\times 3}$. The initial x_i is constructed as

$$x_i = \begin{pmatrix} q_i \\ P^b \end{pmatrix} \tag{75}$$

$$q = \begin{pmatrix} \theta\omega/\|\omega\|_2 \\ dv/\|v|_2 \end{pmatrix} , \tag{76}$$

$$i = \arg\max_i (\sigma_i(\exp(\tilde{q})) \,, \tag{77}$$

$$q_i = \mathcal{X}_i(\exp(\tilde{q})) \,, \tag{78}$$

$$P^b = \begin{pmatrix} \theta_p\omega_p/\|\omega_p\|_2 \\ d_p v_p/\|v_p|_2 \end{pmatrix} , \tag{79}$$

where σ_i are chosen as in (33) and the $\mathcal{X}_i$ are taken from $\mathcal{A}^{SE(3)}_{\min}$ (19). The choice of i and the large initial distribution for θ guarantees that training indeed occurs over all four charts. The cost $J(H^0_b, P^b, W^b_\theta)$ is constructed as

$$\begin{aligned}
J(H^0_b, P^b, W^b_\theta) = {} & E(H^0_b(T), P^b(T)) \\
& + \int_0^T \|(W^b_\theta(t)\|_2 \mathrm{d}t \,.
\end{aligned} \tag{80}$$

Here, the final cost $E(H^0_b(T), P^b(T))$ is given by the negative log of the probability of a target density constructed as in (75), but with sampling of θ, d, θ_p, d_p from normal distributions:

$$\theta \sim \mathcal{N}(0, 1e-2) \,, \tag{81}$$

$$d \sim \mathcal{N}(0, 1e-2) \,, \tag{82}$$

$$\theta_p \sim \mathcal{N}(0, 1e-5) \,, \tag{83}$$

$$d_p \sim \mathcal{N}(0, 1e-3) \,. \tag{84}$$

Remark 5 The sampling of θ in the target distribution glances over the fact that a distribution for θ should have finite support, but this can be ignored for normal distributions with sufficiently small variance σ^2.

6.3 Results

A small horizon optimization is performed with final time $T = 0.1$, batch-size of 2048 and learning rate $1e-3$ using the ADAM optimizer. Training was performed on a GTX-1080ti GPU over 40 hours. For the ODE solver, `DormandPrince45` was used with absolute and relative tolerances of $1e-8$. The results of this optimization are shown in Figure 4.

Clear training progress can be seen by the decrease of the total and running loss in Figures 4a and 4b. This decreased loss is achieved by damping out motion as quickly as possible, as becomes clear from Figures 4c and 4d that show the initial and final momentum during training, respectively for the angular and linear momentum. Given the presented training specifications, reducing such excess momentum appears to be strongly prioritized over reducing the deviation from the goal pose, as becomes clear from Figures 4e and 4f. These figures show the initial and final deviation from the goal pose via the angle and distance from the goal pose, respectively. Initial and final angle /distance show a strong overlap because they barely change over the controlled time. In fact, they change less as the controller training progresses.

7 Discussion

The presented optimization strategy in multiple charts of a minimal atlas clearly works. However, there is currently considerable numerical instability in the sensitivity calculation. This prevented showing optimization results for a longer time horizon, as well optimization of the class of controllers derived in this work. It should, however, be entirely possible to prevent such numerical instability given the time for an alternative implementation of the sensitivity calculation. Currently, the forward pass only saves the final conditions. The backward pass then uses these final conditions in the sensitivity calculation to calculate backwards in time both the dynamics of the state (58) and the dynamics of the co-state (59). It is these backwards dynamics of the state that are unstable, and this issue can be prevented by saving a sufficient number of points from the forward dynamics for multiple shooting in the backward pass, which is much more robust than the current single shooting.

The use of the Lie group structure of $SE(3)$ in the form of a minimal exponential atlas for the definition of a general class of controllers lead to a natural definition of a control structure via canonical mappings defined for $SE(3)$. By these canonical mappings, in particular the derivative of the exponential map is meant. Implementation of this class of controllers in a state of the art framework for deep learning was rather friction-less, and allowed reusing large chunks of existing structure regardless of the more involved topology of $SE(3)$ compared to usual spaces. However, numerical issues also appear in computing partial derivatives and gradients of the defined functions, which required extra care to be taken in such steps. The analytic definition of some derivatives lead to slow functions. While computationally efficient implementations of these are entirely possible, they likewise demand extra time for implementation.

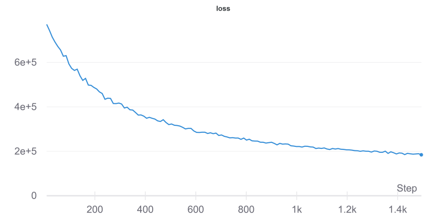
In the context of general manipulators such numerical issues are not expected to become more involved when differentiating the geometric Jacobian in the sensitivity calculations. This is because the geometric Jacobian should be problem-free when being differentiated by autodifferentation, since the geometric Jacobian largely consists of functions such as sines and cosines.

8 Conclusion

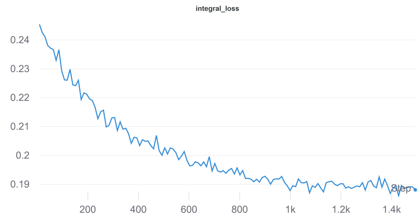
In this work the control paradigm of optimal energy shaping was extended to rigid bodies with configuration space on $SE(3)$. The core principle of this approach was to consider the structure of the Lie group $SE(3)$ from a design stage on, heavily using the structure provided by $SE(3)$ in the ensuing definition and optimization of a general class of energy balancing, passivity based controllers. As opposed to previously available controllers of this type, optimization plays a crucial role and allows the learning of *tasks* in a robot's work-space, rather than merely achieving stabilization or trajectory tracking. Future work prior to actually handing in this article to a journal will aim at increasing the numerical side of these algorithms and to alleviate stability issues in simulating the dynamics backward from a final condition, as is done in the sensitivity update. The optimization will also be performed for the presented general class of potentials and damping injections on $SE(3)$.

References

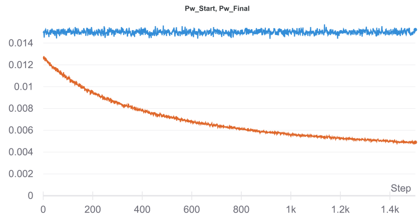
1. R. Rashad, F. Califano, and S. Stramigioli, "Port-Hamiltonian Passivity-Based Control on SE(3) of a Fully Actuated UAV for Aerial Physical Interaction



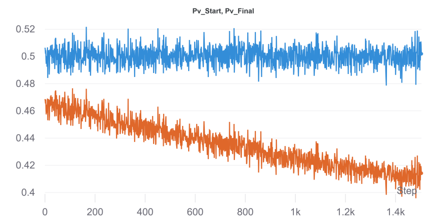
(a) Total Loss



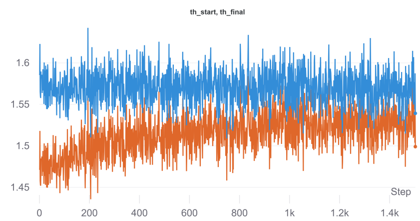
(b) Integral Loss



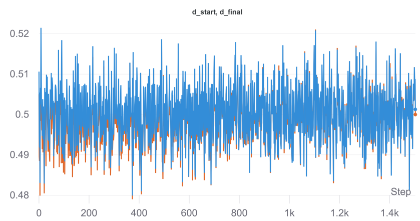
(c) Initial and final angular momentum



(d) Initial and final linear momentum



(e) Initial and final angular goal pose deviation



(f) Initial and final linear goal pose deviation

Fig. 4: Data from training the quadratic potential in the small horizon, plotted against training epoch

Near-Hovering," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4378–4385, 2019.
2. S. Stramigioli, *Energy-Aware robotics*, pp. 37–50. Lecture Notes in Control and Information Sciences, Springer, Dec. 2015.
3. R. Ortega and I. Mareels, "Energy-balancing passivity-based control," *Proceedings of the American Control Conference*, vol. 2, pp. 1265–1270, 2000.
4. A. van der Schaft, *L2-Gain and Passivity Techniques in Nonlinear Control. Lecture Notes in Control and Information Sciences 218*. Netherlands: Springer, 1996.
5. G. A. Folkertsma and S. Stramigioli, "Energy in Robotics," *Foundations and Trends in Robotics*, vol. 6, no. 3, pp. 140–210, 2017.
6. S. Groothuis, G. Folkertsma, and S. Stramigioli, "A general approach to achieving stability and safe behavior in distributed robotic architectures," *Frontiers in robotics and AI*, vol. 5, Oct. 2018.
7. S. Stramigioli, *Modeling and IPC Control of Interactive Mechanical Systems: A Coordinate-free Approach*. No. 266 in Lecture Notes in Control and Information Sciences, Netherlands: Springer, 2001.
8. R. T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, "Neural ordinary differential equations," 2019.
9. S. Massaroli, M. Poli, J. Park, A. Yamashita, and H. Asama, "Dissecting neural ODEs," *Advances in Neural Information Processing Systems*, vol. 2020-Decem, no. NeurIPS, 2020.
10. S. Massaroli, M. Poli, F. Califano, J. Park, A. Yamashita, and H. Asama, "Optimal Energy Shaping via Neural Approximators," pp. 1–12, 2021.
11. Z. Teed and J. Deng, "Tangent Space Backpropagation for 3D Transformation Groups," mar 2021.
12. L. Falorsi and P. Forr, "Neural Ordinary Differential Equations on Manifolds," no. Icml, 2020.
13. A. Lou, D. Lim, I. Katsman, L. Huang, Q. Jiang, S.-N. Lim, and C. De Sa, "Neural Manifold Ordinary Differential Equations," *Advances in Neural Information Processing Systems*, vol. 2020-Decem, jun 2020.
14. E. W. Grafarend and W. Kühnel, "A minimal atlas for the rotation group SO(3)," *GEM - International Journal on Geomathematics 2011 2:1*, vol. 2, pp. 113–122, may 2011.
15. V. Duindam, A. Macchelli, S. Stramigioli, and H. Bruyninckx, *Modeling and Control of Complex Physical Systems*. 2009.
16. M. Visser, S. Stramigioli, and C. Heemskerk, "Cayley-Hamilton for roboticists," *IEEE International Conference on Intelligent Robots and Systems*, vol. 1, pp. 4187–4192, 2006.
17. E. Eade, "Derivative of the Exponential Map," no. 6, 2018.
18. J. Lloyd, "Smooth Partitions of Unity on Manifolds," *Transactions of the American Mathematical Society*, vol. 187, p. 249, jan 1974.
19. M. R. Hossain, M. Raknuzzaman, M. M. Rahman, and M. J. Hossain, "Global Theory of Smooth Functions of Manifolds," *Annals of Pure and Applied Mathematics*, vol. 12, pp. 221–232, nov 2016.
20. A. Skopenkov, "Embedding and knotting of manifolds in Euclidean spaces," *Surveys in Contemporary Mathematics*, pp. 248–342, apr 2006.
21. F. Bullo and R. M. Murray, "Tracking for fully actuated mechanical systems: a geometric framework," *Automatica*, vol. 35, pp. 17–34, jan 1999.
22. H. Robbins and S. Monro, "A Stochastic Approximation Method," *The Annals of Mathematical Statistics*, vol. 22, no. 3, pp. 400 – 407, 1951.
23. H. Kaufman, "The mathematical theory of optimal processes, by l. s. pontryagin, v. g. boltyanskii, r. v. gamkrelidze, and e. f. mishchenko. authorized translation from the russian. translator: K. n. trirogoff, editor: L.. w. neustadt. interscience publishers (division of john wiley and sons, inc. , new york) 1962. viii 360 pages.," *Canadian Mathematical Bulletin*, vol. 7, no. 3, p. 500–500, 1964.
24. M. Poli, S. Massaroli, A. Yamashita, H. Asama, and J. Park, "TorchDyn: A Neural Differential Equations Library," sep 2020.

9 Appendix

9.1 Completeness of Minimal Atlas

The completeness of $\mathcal{A}_{\text{min}}^{SE(3)}$ follows from completeness of $\mathcal{A}_{\text{min}}^{SO(3)}$, so only the latter is shown here. For the i-th chart in $\mathcal{A}_{\text{min}}^{SO(3)}$, denote

$$\Omega_i := SO(3) \setminus U_{\omega_i} \,. \tag{85}$$

This section will show that $\bigcup_i \Omega_i = \emptyset$. Let ω_i' be the coordinates of a matrix $R = R_{\omega_i} R_{\omega_i'}$ in chart i. For $\|\omega_i'\|_2 = \pi$, a matrix R is retrieved that is not part of the chart-region corresponding to the i-th chart. Via equations (11) and (16), $\|\omega_i'\|_2 = \pi$ corresponds to

$$\cos^{-1}\left(\frac{1}{2}(\text{Tr}(R_{\omega_i}^T R) - 1)\right) = \pi \,, \tag{86}$$

from which follows the condition for matrices unreachable for the chart associated with R_{ω_i}

$$\text{Tr}(R_{\omega_i}^T R) = -1 \,. \tag{87}$$

So the set Ω_i of unreachable matrices for a chart $(U_i, x_i) \in \mathcal{A}_{\text{min}}^{SO(3)}$ are

$$\Omega_i = \{R \mid \text{Tr}(R_{\omega_i}^T R) = -1\} \,. \tag{88}$$

Denoting $c_x := \cos(\|x\|_2)$, $s_x := \sin(\|x\|_2)$, $\hat{\omega} = \frac{\omega}{\|\omega\|_2} = (\hat{\omega}_1, \hat{\omega}_2, \hat{\omega}_3)^T$ this condition means that the set $\bigcup_i \Omega_i$ is found by the solutions ω' to equations (89) to (92), using $R = R_{\omega'}$.

$$\text{Tr}(R_{\omega_0}^T R) = \text{Tr}(R) = 1 + 2c_\omega' = -1 \,, \tag{89}$$
$$\text{Tr}(R_{\omega_1}^T R) = -1 + 2(1 - c_\omega')\hat{\omega}_1^2 = -1 \,, \tag{90}$$
$$\text{Tr}(R_{\omega_2}^T R) = -1 + 2(1 - c_\omega)\hat{\omega}_2^2 = -1 \,, \tag{91}$$
$$\text{Tr}(R_{\omega_3}^T R) = -1 + 2(1 - c_\omega)\hat{\omega}_3^2 = -1 \,. \tag{92}$$

While it can be readily shown that this has no valid solution and $\mathcal{A}_{\text{min}}$ must cover all $SO(3)$, this is better seen by recognizing that the functions $p_i(R) = (\text{Tr}(R_{\omega_i}^T R) + 1)/4$ constitute a partition of unity: they always range between 0 and 1, they are per definition only 0 outside the i-th chart region, and their sum is always 1:

$$\begin{aligned}&\sum_i (\text{Tr}(R_{\omega_i}^T R) + 1)/4 \\ &\quad = (1 + 2c_\omega - 3 + 2(1 - c_\omega))/4 + 1 = 1 \,.\end{aligned} \tag{93}$$

Hence, there is always a valid chart given by $p_i(R) > 0$ and $\mathcal{A}_{\text{min}}$ is indeed a minimal Atlas.

9.2 Invertibility of K

For $q = (\omega, v)^T \in \mathbb{R}^6$ The eigenvalues of $K(q)$ can be expressed as a function of ω by first writing ad_q in its Jordan normal form

$$ad_\phi = PJP^{-1} \,, \tag{94}$$

where J is the block-diagonal Jordan normal form and P is an invertible matrix. Note that with the scalar function $f(x) = \frac{1-e^{-x}}{x}$, $K(q) = f(ad_q) = Pf(J)P^{-1}$ has the determinant

$$\det K(\phi) = \det(f(J)) \,. \tag{95}$$

Here, J is given as

$$J = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & i\|\omega\|_2 & s_v & 0 & 0 \\ 0 & 0 & 0 & i\|\omega\|_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & -i\|\omega\|_2 & s_v \\ 0 & 0 & 0 & 0 & 0 & -i\|\omega\|_2 \end{bmatrix} , \tag{96}$$

where $s_v = \text{sign}(\|v\|_2)$ is 0 if $|v| = 0$ and 1 otherwise. This results in

$$f(J) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & f(i\|\omega\|_2) & s_v f'(i\|\omega\|_2) & 0 & 0 \\ 0 & 0 & 0 & f(i\|\omega\|_2) & 0 & 0 \\ 0 & 0 & 0 & 0 & f(-i\|\omega\|_2) & s_v f'(-i\|\omega\|_2) \\ 0 & 0 & 0 & 0 & 0 & f(-i\|\omega\|_2) \end{bmatrix} . \tag{97}$$

Such that

$$\det f(J) = f(i\|\omega\|_2)^2 f(-i\|\omega\|_2)^2 \,. \tag{98}$$

Note that

$$f(i\alpha\pi) = (\sin(\alpha\pi) + i(\cos(\alpha\pi) - 1))/(\alpha\pi) \,, \tag{99}$$

such that $K(q)$ becomes singular for $\alpha = 2$ but not $\alpha = 1$, i.e. $\|\omega\|_2 = 2\pi$ but not $\|\omega\|_2 = \pi$. This also shows that $K(q)$ becomes singular whenever $\|\omega\|_2$ is an integer multiple of 2π.

9.3 Differentiating K

Let $\mathrm{sinc}(\theta) := \sin(\theta)/\theta$, then the $a_i(\theta)$ are given by

$$\begin{aligned}
a_0 &= 1\,,\\
a_1 &= -\frac{1}{2}\,,\\
a_2 &= \frac{1}{4\theta^2}(8 + 2\cos(\theta) - 10\,\mathrm{sinc}(\theta))\,,\\
a_3 &= \frac{1}{4\theta^3}(-4\theta + \frac{12 - 12\cos(\theta)}{\theta} - 2\sin(\theta))\,,\\
a_4 &= \frac{1}{4\theta^4}(4 + 2\cos(\theta) - 6\,\mathrm{sinc}(\theta))\,,\\
a_5 &= \frac{1}{4\theta^5}(-2\theta + \frac{8 - 8\cos(\theta)}{\theta} - 2\sin(\theta))\,.
\end{aligned}$$

Then the derivatives of the a_i follow as $\frac{d}{dt}a_i = \dot{\theta}\frac{d}{d\theta}a_i$ with $\dot{\theta} = \frac{\omega^T\dot{\omega}}{\theta}$ and $\frac{d}{d\theta}a_i$ given by

$$\begin{aligned}
\frac{d}{d\theta}a_0 &= 0\,,\\
\frac{d}{d\theta}a_1 &= 0\,,\\
\frac{d}{d\theta}a_2 &= \frac{1}{2\theta^3}(15\,\mathrm{sinc}(\theta) - 8 - 7\cos(\theta))\,,\\
\frac{d}{d\theta}a_3 &= \frac{1}{2\theta^3}(\frac{24\cos(\theta) - 24}{\theta^2} - \cos(\theta) + 9\,\mathrm{sinc}(\theta) + 4)\,,\\
\frac{d}{d\theta}a_4 &= \frac{1}{2\theta^5}(15\,\mathrm{sinc}(\theta) - 8 - 7\cos(\theta))\,,\\
\frac{d}{d\theta}a_5 &= \frac{1}{2\theta^5}(\frac{24\cos(\theta) - 24}{\theta^2} - \cos(\theta) + 9\,\mathrm{sinc}(\theta) + 4)\,.
\end{aligned}$$

9.4 Full Adjoint Dynamics

Einstein notation is used for clarity in this Section, since the derivatives in (59) involve the differentiation of matrices with respect to vectors. The chart is denoted as a preceding superscript, i.e. the j-th element of λ_i from Section 5 would read ${}^{i}\lambda_j$. For readability, this preceding superscript is largely left out. The change-rate of the i-th element of the co-state in any chart is then

$$\dot{\lambda}_i = -\frac{\partial}{\partial x^i}(\lambda_k f^k + L)\,. \tag{100}$$

Here, the first six elements of f^i are given by

$$f^i = (K^{-1})^i_j(\mathcal{I}^{-1})^{j,k}P_k\,, \tag{101}$$

and the final six are given by

$$f_i = (\mathrm{ad}^T_{(\mathcal{I}^{-1})^{j,l}P_l})^k_i P_k - B^k_i P_k - dV_i\,. \tag{102}$$

In this, the B^k_i and dV are constructed as described in Section 4.1:

$$dV = \sum_n ({}^{n}K^{-T})^j_i \frac{\partial}{\partial q_j}({}^{n}_{\sigma}V)\,, \tag{103}$$

$$B^k_i = \sum_n {}^{n}_{\sigma}B^k_i\,. \tag{104}$$

The partial derivatives of (101) are

$$\frac{\partial}{\partial q^i}(\lambda_k f^k) = \lambda_k(\frac{\partial}{\partial q^i}K^{-1})^k_l(\mathcal{I}^{-1})^{l,m}P_m\,, \tag{105}$$

$$\frac{\partial}{\partial P_i}(\lambda_k f^k) = \lambda_k(K^{-1})^k_l(\mathcal{I}^{-1})^{l,m}\delta_{m,i}\,. \tag{106}$$

The partial derivatives of (102) are

$$\begin{aligned}
\frac{\partial}{\partial q^i}(\lambda^k f_k) &= \lambda^k\frac{\partial}{\partial q^i}(B^k_i P_k + dV_k) \\
&= \lambda^k \sum_n {}^{n}A^j_i\Bigg(\frac{\partial\, {}^{n}_{\sigma}B^k_i}{\partial q^j}P_k \\
&\quad + \frac{\partial({}^{n}K^{-T})^l_k}{\partial({}^{n}q^j)}\frac{\partial\, {}^{n}_{\sigma}V}{\partial({}^{n}q^l)} \\
&\quad + ({}^{n}K^{-T})^l_k\frac{\partial^2\, {}^{n}_{\sigma}V}{\partial({}^{n}q^j)\partial({}^{n}q^l)}\Bigg)\,.
\end{aligned} \tag{107}$$

$$\begin{aligned}
\frac{\partial}{\partial P_i}(\lambda^k f_k) &= \lambda^k\big((\mathrm{ad}^T_{(\mathcal{I}^{-1})^{j,l}\delta^i_l})^l_k P_l \\
&\quad + (\mathrm{ad}^T_{(\mathcal{I}^{-1})^{j,l}P_l})^l_k\delta^i_l - \frac{\partial B^l_k}{\partial P_i}P_l - B^l_k\delta^i_l\big)\,.
\end{aligned} \tag{108}$$

To compute derivatives of K refer to Section 5.1, the ${}^{n}A^j_i$ are given by

$${}^{n}A^j_i = (K({}^{m}q_i)^T K({}^{n}q)^{-T})^j_i\,, \tag{109}$$

and m is the chart of λ_i.