

Variable Impedance Control using Deep Geometric Potential Fields

Nikhil Potu Surya Prakash* Joochwan Seo*
Koushil Sreenath* Jonguen Choi** Roberto Horowitz*

* *University of California, Berkeley, CA 94720 USA*

** *Yonsei University, Seoul 03722, Republic of Korea*

*e-mails: *{nikhilps,joochwan_seo,koushils,horowitz}@berkeley.edu,**
jongeunchoi@yonsei.ac.kr*

Abstract:

In this paper, we present a novel strategy for developing control mechanisms for fully actuated mechanical systems evolving on manifolds utilizing the representative power of neural networks. This method builds invariant conservative and dissipative potential functions using neural networks that generate a stabilizing nonlinear elastic and damping wrench pair (stable potential field). These elastic and damping forces can be used to replace the proportional and derivative control terms in impedance control to have more representative controllers that can be made to mimic expert demonstrations for kinesthetic teaching or to improve performance using an LQR style formulation. Furthermore, the principle of invariance is instrumental in enhancing the transferability of learning across different scenarios through the invariant potential functions.

Keywords: Impedance Control, Potential Fields, Kinesthetic Teaching, Geometric Control, Deep Learning

1. INTRODUCTION

The development of control laws based on learning for systems operating on smooth manifolds is a challenging yet intriguing issue within the realm of control systems engineering. Learning based approaches can be used to improve performances of traditional control algorithms across various applications, including aerial robots Lee (2012); Kotaru et al. (2020); Kronander and Billard (2016) and robotic manipulators equipped with impedance and admittance control mechanisms Hogan (1985); Seo et al. (2023a,b, 2024). These areas require advanced control strategies capable of navigating the systems' inherent nonlinearities and intricacies. Within this framework, employing potential functions that generate elastic forces to dictate control laws presents a promising method for controller synthesis, providing a reliable means to steer the system's behavior toward targeted states.

In the literature, it has been observed that potential functions formulated on the Special Orthogonal group ($SO(3)$), as discussed in Bullo and Murray (1999); Lee (2012), encounter issues of vanishing gradients when the orientation error reaches π radians. This results in diminutive forces at substantial errors, leading to a tepid response from the system. Previous research efforts, including those by Bullo and Murray (1995); Park (1995); Park et al. (1995) have proposed utilizing a metric on the Lie algebras of $SO(3)$ and $SE(3)$ that maintains a uniformly quadratic nature. However, this approach encounters limitations as the logarithmic map becomes undefined for orientation errors of π radians, given that rotations about any axis by π radians yield identical orientations. While these potential functions allow for the integration of matrix gains to

modulate system responses for positions and orientations, tuning these gains to achieve a desired response is not as straightforward as with linear systems.

These challenges underscore the need for innovative approaches to construct potential functions. One promising direction is the application of neural networks, leveraging their capacity for representation to develop potential functions whose gradients generate elastic spring and damping forces. Nonetheless, it is important to acknowledge that such neural network-based potential functions may only achieve almost global properties, rather than global ones (which are also true for other potential functions). This limitation arises from the impossibility of building a continuous control law that yields a continuous vector field on a compact manifold with a globally asymptotic equilibrium point, as elucidated by Bhat and Bernstein (2000). Consequently, the aim is not to circumnavigate this inherent limitation but to design potential functions in a manner that relocate "problematic" points to regions of the manifold that are of lesser interest, thereby optimizing system performance within the desired operational domain.

An additional significant application of leveraging neural networks for the construction of potential functions is the acquisition of potential fields through expert demonstrations, achieved in an invariant fashion. In our preceding study Seo et al. (2023a), we introduced a neural network-based architecture designed to infer gains as a function of state variables, utilizing potential fields derived from expert demonstrations. The methodology presented in this manuscript extends this concept by offering a generalized framework for the learning of state-dependent gains via

potential functions. This framework not only facilitates the direct derivation of a stabilizing control law during inference but also does so with provable guarantees of stability, thereby enhancing the robustness and applicability of the approach in practical settings.

In our previous work in Prakash et al. (2024), we developed conservative potential functions and used a linear damping term, whereas in the current work we extend the previous work to generalize the damping potentials. One of the challenges in building these potential functions is that their structure needs to satisfy specific Lyapunov function like properties, such as being zero at the equilibrium and positive everywhere else (or equivalently being lower bound and attaining the minimum at its equilibrium). These properties will be achieved with the use of Input Convex Neural Networks (ICNNs) Amos et al. (2016) and its application to the construction of Lyapunov functions to learn stable dynamical systems in Manek and Kolter (2020). Though these Lyapunov functions were constructed for Euclidean spaces, we will show in the next sections how this approach is also beneficial for systems evolving on smooth manifolds.

The key contributions of this paper are outlined as follows: 1. A recapitulation of errors and variations on manifolds is provided. 2. A Neural Network architecture to construct deep invariant geometric potential functions on smooth manifolds is presented. These potential functions are designed to produce stabilizing elastic (summarized from Prakash et al. (2024)) and dissipative wrench pairs from any random initial network setup. 3. An impedance control law for robotic manipulators and an orientation control law for satellites based on these potential functions, including a training procedure aimed at refining the potential function to enhance convergence of trajectories is presented.

2. PRELIMINARIES

In this section, we will summarize the notion of an error on the manifold and describe the kinematics of a particle moving on the manifold, variations of configurations, and errors in velocities the Special Orthogonal Group (SO(3)) and the Special Euclidean Group (SE(3)) from Bullo and Murray (1999); Lee et al. (2017).

2.1 Special Orthogonal Group (SO(3)):

The Special Orthogonal group (SO(3)) represents the set of all possible rotation matrices R without any reflections. The following is a representation of the group as an embedding in $\mathbb{R}^{3 \times 3}$

$$SO(3) = \{R \in \mathbb{R}^{3 \times 3} : R^T R = \mathcal{I}_3, \det(R) = 1\}. \quad (1)$$

The kinematics of a body that just is restricted to rotate without any translations can be written as

$$\dot{R} = R\hat{\Omega}, \quad (2)$$

where $\Omega \in \mathbb{R}^3$ is the angular velocity expressed in the body-fixed frame and $\hat{\Omega} \in \mathfrak{so}(3)$, the Lie Algebra (tangent space at identity) of SO(3).

The configuration error between a desired configuration R_d and the current configuration R can be defined as

$$R_e \triangleq R_d^T R. \quad (3)$$

This error is called the right error according to Bullo and Murray (1999). Note that the error R_e is also an element of SO(3), and the error becomes \mathcal{I}_3 when $R = R_d$. The body fixed angular velocity error Ω_e according to Bullo and Murray (1999) can be defined using the body fixed angular velocity $\hat{\Omega} = R^T \dot{R}$ and the desired angular velocity $\hat{\Omega}_d = R_d^T \dot{R}_d$ via

$$\dot{R}_e = R_e \hat{\Omega}_e, \quad (4)$$

where

$$\Omega_e \triangleq \Omega - R_e^T \Omega_d. \quad (5)$$

2.2 Special Euclidean Group (SE(3)):

The Special Euclidean group (SE(3)) describes the pose of a rigid body in 3D space via a rotation matrix R and a position p . The following is a representation of the group: $SE(3) = \{(R, p) \in SO(3) \times \mathbb{R}^3 : R^T R = \mathcal{I}_3, \det(R) = 1\}$. The kinematics of a body evolving on SE(3) can be written as

$$\dot{g} = g\Gamma(V^b), \quad (6)$$

where

$$g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}, V^b = \begin{bmatrix} v \\ \Omega \end{bmatrix}, \Gamma(V^b) = \begin{bmatrix} \hat{\Omega} & v \\ 0 & 0 \end{bmatrix}. \quad (7)$$

Here g is called the homogeneous representation of the SE(3) group and $v \in \mathbb{R}^3$ is the translational velocity with $\dot{p} = v$ represented in the body coordinates.

The configuration error between a desired configuration g_d and the current configuration g can be defined as follows with R_e from (3) and $p_e \triangleq p - p_d$ denoting the translational error

$$g_e \triangleq g_d^{-1} g = \begin{bmatrix} R_e & R_d^T p_e \\ 0 & 1 \end{bmatrix}, g_d = \begin{bmatrix} R_d & p_d \\ 0 & 1 \end{bmatrix}. \quad (8)$$

Note that the configuration error g_e is also an element of SE(3) and the error becomes \mathcal{I}_4 when $g = g_d$.

By taking the time derivative of g_e , we can obtain the following

$$\dot{g}_e = g_e \Gamma(e_V). \quad (9)$$

where e_V is the velocity error defined by the following utilizing the desired quantities with subscript d

$$e_V \triangleq \underbrace{\begin{bmatrix} v \\ \Omega \end{bmatrix}}_{V^b} - \underbrace{\begin{bmatrix} R_e^T v_d + R_e^T \hat{\Omega}_d R_d^T (p - p_d) \\ R_e^T \Omega_d \end{bmatrix}}_{V_d^*} = \begin{bmatrix} e_v \\ \Omega_e \end{bmatrix}. \quad (10)$$

Here V^b is the velocity in the body fixed frame and V_d^* is the desired velocity transported to the tangent space of g .

Left Invariance:

It can be seen from the following equations that by transforming the current and desired configurations both from the left arbitrarily by the same translation, we do not get a change in the error.

$$\begin{aligned} (R_l r_d)^T (R_l r) &= r_d^T (R_l^T R_l) r = r_d^T r = r_e \\ (R_l R_d)^T (R_l R) &= R_d^T (R_l^T R_l) R = R_d^T R = R_e \\ (g_l g_d)^{-1} (g_l g) &= g_d^{-1} (g_l^{-1} g_l) g = g_d^{-1} g = g_e \end{aligned} \quad (11)$$

This is an essential property as incorporating this property allows us to transfer trained skills from one scene to

another scene without any new training. We will use these errors to construct potentials in Sec. 3. Since the potentials depend solely on the errors, the potential functions are also left invariant. It is also easy to see that for left error representations, we have right invariance. Seo et al. (2023a) presents a more elaborate explanation of invariance.

3. POTENTIAL FUNCTIONS

In this section, a methodology for designing positive definite conservative potential functions and positive semi-definite dissipative potential functions utilizing Fully Input Convex Neural Networks (FICNNs) and Partially Input Convex Neural Networks (PICNNs) from Amos et al. (2016) and its use in the construction of Lyapunov functions Manek and Kolter (2020) will be presented. The conservative potential functions are summarized from our previous work in Prakash et al. (2024).

For the remainder of the section, quantities defined by W are real matrices, quantities defined by U are non-negative matrices (non-negative entries), and quantities defined by b are real bias vectors, all of appropriate dimensions. The nonlinear activations σ_i must be convex like ReLU but $\tilde{\sigma}_i$ could be arbitrary nonlinear activations. A small positive definite term is added at the end using $\epsilon_1, \epsilon_2 > 0$ to ensure the positive definiteness/semi definiteness of the potential functions. The activation at the final layer σ_{k+1} is a smoothed ReLU (for details about all the activation functions, readers are referred to Manek and Kolter (2020)).

Properties of Conservative Potentials

We consider the following properties to build conservative potential functions Ψ that generate elastic forces f_s which are the nonlinear generalizations of proportional terms in PD control. The potential function Ψ

- (1) is a function of only the configuration error.
- (2) is positive definite.
- (3) obtains its unique global minimum when the current configuration and desired configuration are the same.

The aim is to make the conservative potential function zero when the current configuration and desired configuration coincide and positive everywhere else. Though convexity is a restriction, we can relax this by adding another layer of an invertible residual network Behrmann et al. (2019) layer before FICNN to improve representative power. A caveat here is that though we are using an FICNN to build the conservative potential functions, the manifold is not convex, and hence, the conservative potential function and domain pair together are not convex, but the conservative potential function with a convex domain remains convex.

Properties of Dissipative Potentials

We consider the following properties to build dissipative potential functions \mathcal{R} (Rayleigh dissipation potentials) that generate damping forces which are the nonlinear generalizations of derivative terms in PD control. The potential function \mathcal{R}

- (1) is a non negative function of the configuration error (first argument) and the velocity error (second argument).
- (2) is positive definite in its second argument.
- (3) is zero when the velocity error is zero.

$$0 = \mathcal{R}(\cdot, 0) \leq \mathcal{R}(\cdot, \cdot) \quad (12)$$

- (4) is convex in its second argument i.e., velocity error. This property is necessary for proving asymptotic stability.

3.1 Potential Functions on $SE(3)$

Conservative Potentials: We can define $\Psi : SE(3) \mapsto \mathbb{R}^+$ such that $\Psi(\mathcal{I}_3) = 0$ as follows. Here $\bar{g}_e = \begin{bmatrix} \bar{R}_e \\ R_d^T p_e \end{bmatrix}$, where \bar{R}_e is the 9×1 vector reshaping of R_e , i.e., $\bar{R}_e = \text{vec}(R_e)$.

$$\begin{aligned} z_1 &= \sigma_0(W_0 \bar{g}_e + b_0) \\ z_{i+1} &= \sigma_i(U_i z_i + W_i \bar{g}_e + b_i), \\ \Phi(g_e) &\equiv z_k \\ \Psi(g_e) &= \sigma_{k+1}(\Phi(g_e) - \Phi(\mathcal{I}_4)) + \epsilon_1 \|\mathcal{I}_4 - g_e\|_F^2. \end{aligned} \quad (13)$$

Dissipative Potentials: We can define the dissipative potential function $\mathcal{R} : SE(3) \times \mathbb{R}^6 \mapsto \mathbb{R}^+$ such that $\mathcal{R}(g_e, 0_{6 \times 1}) = 0$ as follows:

$$\begin{aligned} u_0 &= \bar{g}_e \\ u_{i+1} &= \tilde{\sigma}_i(\tilde{W}_i u_i + \tilde{b}_i) \\ z_{i+1} &= \sigma_i(U_i^z(z_i \circ [W_i^{zu} u_i + b_i^z]_+)) \\ &\quad + W_i^y(y \circ [W_i^{yu} u_i + b_i^y]) + W_i^u u_i + b_i \\ \xi(g_e, e_V) &\equiv z_k \\ \mathcal{R}(g_e, e_V) &= \sigma_{k+1}(\xi(g_e, e_V) - \xi(g_e, 0_{6 \times 1})) \\ &\quad + \epsilon_2 \|e_V\|_2^2, \end{aligned} \quad (14)$$

where \circ represent the Hadamard product or an element-wise multiplication. Here $i = 1, \dots, k-1$ and z_i refers to the output of the i^{th} layer with $z_0 = e_V$. By making $p_e = 0$, the potential function in (13) reduces to the potential on $SO(3)$. Similarly, making $p_e = 0, v_e = v - v_d = 0$ and just using R_e, Ω_e in (14) also reduces the dissipative potential to that of $SO(3) \times \mathbb{R}^3$. It is worth noting here that the potential functions defined in (13) and (14) always produce a stabilizing elastic force and damping force pair irrespective of the initialization of the weights as long as U matrices are non-negative. This constraint on U can be enforced by first initializing them randomly and then using the softplus activation function to make them all positive. The subscript $+$ in dissipation potentials follows the convention in Amos et al. (2016) and makes its corresponding arguments positive using softplus activation function.

4. ELASTIC AND DAMPING WRENCHES

In this section, we briefly show, without extensive details, the various elastic wrenches from conservative potentials and also define the damping forces as gradients of dissipative potentials. Definitions, detailed derivations of elastic forces and stability analyses can be found in our previous work Prakash et al. (2024) but the definitions of damping terms are novel.

4.1 Special Orthogonal Group $SO(3)$:

A conservative potential on $SO(3)$ can be represented as $\Psi(R_e)$ by letting $p_e = 0$ in (13) and a dissipative potential on $SO(3) \times T_R SO(3)$ can be represented as $\mathcal{R}(R_e, \Omega_e)$ by letting $p_e = 0, v_e = 0$. The elastic and damping forces can be found as

$$\begin{aligned} f_s &= -D_{R_e} \Psi = -(R_e^T \Psi'(R_e) - \Psi'(R_e)^T R_e)^\vee \\ f_d &= -\partial_{\Omega_e} \mathcal{R}(R_e, \Omega_e) \end{aligned} \quad (15)$$

where $\Psi' = \frac{\partial}{\partial M} \Psi(M)|_{M=R_e}$.

4.2 Special Euclidean Group $SE(3)$:

The variation of the conservative potential function on $SE(3)$ defined in (13) and the gradient of dissipative potential function defined in (14) generate the following elastic and damping forces. This yields the elastic force as

$$\begin{aligned} f_s &= \begin{bmatrix} -R_e^T \partial_2 \Psi \\ -(R_e^T \partial_1 \Psi - \partial_1 \Psi^T R_e)^\vee \end{bmatrix} \\ f_d &= -\partial_{e_V} \mathcal{R}(g_e, e_V) \end{aligned} \quad (16)$$

where $\partial_1 \Psi = \frac{\partial}{\partial M} \Psi(M, z)|_{M=R_e, z=R_d^T p_e}$ and $\partial_2 \Psi = \frac{\partial}{\partial z} \Psi(M, z)|_{M=R_e, z=R_d^T p_e}$.

Theorem 1. The dissipative potential functions defined in (14) satisfy $\Omega_e^T \partial_{\Omega_e} \mathcal{R}(R_e, \Omega_e) \geq 0$ and $e_V^T \partial_{e_V} \mathcal{R}(g_e, e_V) \geq 0$.

Proof. For a smooth convex function $h(\cdot)$, we have the following first-order necessary and sufficient condition in terms of its gradient $\forall x, y \in \text{dom}(h)$

$$h(y) \geq h(x) + \partial_x h(x)^T (y - x) \quad (17)$$

By interchanging x and y , we get

$$\begin{aligned} h(x) &\geq h(y) + \partial_y h(y)^T (x - y) \\ \implies \partial_x h(x)^T (y - x) &\leq h(y) - h(x) \leq \partial_y h(y)^T (y - x) \\ \implies (\partial_y h(y) - \partial_x h(x))^T (y - x) &\geq 0 \end{aligned} \quad (18)$$

Since $\mathcal{R}(\cdot, \cdot)$ is convex in its second argument, we have

$$(\partial_y \mathcal{R}(\cdot, y) - \partial_x \mathcal{R}(\cdot, x))^T (y - x) \geq 0, \quad (19)$$

By letting $x = 0$, we get

$$\partial_y \mathcal{R}(\cdot, y)^T y = y^T \partial_y \mathcal{R}(\cdot, y) \geq 0. \quad (20)$$

This is equivalent to

$$f_d(\cdot, y)^T y \leq 0. \quad (21)$$

By letting $y = e_v, \Omega_e$ or e_V , the theorem can be proved for all manifolds.

5. DYNAMIC CONTROL

In this section, we consider two interesting and practical problems on $SO(3)$ and $SE(3)$ manifolds namely orientation control of a satellite and Impedance control of a robotic manipulator respectively. The dynamics of both the systems and stabilizing control laws using the constructed potential functions will be presented.

5.1 Control of a Satellite on $SO(3)$:

A simple model of a rigid body rotating without translating can be used to describe the orientation control problem

of a satellite. The orientation of the satellite is described through rotation matrices $R \in SO(3)$. The control is achieved through momentum wheels attached to three perpendicular axes of the satellite. Again for simplicity, we will ignore the dynamics of the reaction wheels and assume the availability of three independent torque components along its three perpendicular axes. The dynamics can be written as follows with \mathbb{J} as the symmetric positive definite inertia matrix, $\Omega \in \mathbb{R}^3$ as the angular velocity represented in the body-fixed frame and $\tau \in \mathbb{R}^3$ as the torque.

$$\begin{aligned} \dot{R} &= R \hat{\Omega} \\ \mathbb{J} \dot{\Omega} + \hat{\Omega} \mathbb{J} \Omega &= \tau \end{aligned} \quad (22)$$

Theorem 2. The following control almost globally asymptotically tracks $R_d(t)$ for a dynamical system described by (22) with the elastic force f_s and damping force f_d described by (15)

$$\tau = \hat{\Omega} \mathbb{J} \Omega - \mathbb{J} \hat{\Omega}_e R_e^T \Omega_d + \mathbb{J} R_e^T \dot{\Omega}_d + \mathbb{J} (f_s + f_d). \quad (23)$$

Proof. This control law achieves the following error dynamics

$$\begin{aligned} \dot{\Omega}_e + \partial_{\Omega_e} \mathcal{R}(R_e, \Omega_e) + D_{R_e} \Psi(R_e) &= 0 \\ \text{or } \dot{\Omega}_e - f_d(R_e, \Omega_e) - f_s(R_e) &= 0 \end{aligned} \quad (24)$$

We will consider the following positive definite Lyapunov candidate function

$$\begin{aligned} W &= \Psi(R_e) + \frac{1}{2} \Omega_e^T \Omega_e \\ \implies \dot{W} &= \dot{\Psi}(R_e) + \Omega_e^T \dot{\Omega}_e = \Omega_e^T (\dot{\Omega}_e - f_s(R_e)) \\ &= \Omega_e^T f_d(R_e, \Omega_e) \leq 0 \text{ from (21)}. \end{aligned}$$

Using Lasalle's Invariance principle, we can also conclude that the equilibrium $R_e = \mathcal{I}_3$ of the error dynamics in (24) is almost globally asymptotically stable as the largest invariant set where $\dot{W} = 0$ only when $R_e = \mathcal{I}_3$ (removing the other unstable equilibria). It may firstly be a bit non-intuitive, but it should be noted that Ω_e can be expressed in terms of R_e and \dot{R}_e which makes the entire error equation a function of just R_e and its time derivatives. The expression is omitted for compactness.

5.2 Control of a Robotic Manipulator on $SE(3)$:

Another problem where potential functions play an important role is in the control of robotic manipulators. We will demonstrate an application to Impedance control of a robotic manipulator Seo et al. (2023b). The manipulator equations in joint space can be written as follows with $q \in \mathbb{R}^n$ as the vector of generalized coordinates of the manipulator.

$$M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = \tau + J_b(q)^T T_e, \quad (25)$$

where $M(q) \in \mathbb{R}^{n \times n}$ is the symmetric positive definite inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ is a Coriolis matrix, $G(q) \in \mathbb{R}^n$ represents the gravitational terms, $\tau \in \mathbb{R}^n$ is the joint torque, and $T_e \in \mathbb{R}^6$ is an external wrench at the end-effector from contacts, human inputs etc. The Coriolis matrix satisfies the property that $M - 2C$ is skew-symmetric.

In the field of impedance control combined with operational space formulation, it is well known from Khatib (1987) that the robot dynamics (25) can be rewritten as

$$\tilde{M}\dot{V}^b + \tilde{C}V^b + \tilde{G} = \tilde{\tau} + \tilde{\tau}_e, \quad (26)$$

where the definitions of \tilde{M} , \tilde{C} , \tilde{G} , \tilde{T} , and \tilde{T}_e can be referred to Seo et al. (2023b).

Theorem 3. The following control law almost globally asymptotically tracks $g_d(t)$ for a dynamical system described by (26) with the elastic and damping forces described by (16) when $T_e = 0$

$$\tilde{\tau} = \tilde{M}\dot{V}_d^* + \tilde{C}V_d^* + \tilde{G} + \tilde{M}(f_d + f_s). \quad (27)$$

Proof. When $T_e = 0$, this control law achieves the following error dynamics

$$\begin{aligned} \dot{e}_V + \partial_{e_V}\mathcal{R}(g_e, e_V) + D_{g_e}\Psi(g_e) &= 0 \\ \text{or } \dot{e}_V - f_d(g_e, e_V) - f_s(g_e) &= 0 \end{aligned} \quad (28)$$

We will consider the following positive definite Lyapunov candidate function

$$\begin{aligned} W &= \Psi + \frac{1}{2}e_V^T e_V \\ \implies \dot{W} &= \dot{\Psi} + e_V^T \dot{e}_V = e_V^T (\dot{e}_V - f_s(g_e)) \\ &= e_V^T f_d(g_e, e_V) \leq 0 \text{ from (21)}. \end{aligned} \quad (29)$$

Using LaSalle's, we can conclude again that the desired equilibrium is almost globally asymptotically stable.

This control law is left invariant and has good learning and transferability properties similar to Seo et al. (2023a).

6. TRAINING THE NEURAL NETWORK

Once the structure of the potential function is finalized (by fixing the number of layers and their sizes), an objective according to the needs of the user can be specified which can be posed as an optimization problem of minimizing a loss function by gradient descent. For improving the performance, we can consider an LQR-style problem where we have a running cost along the trajectory that needs to be minimized. A sample loss function for an error trajectory is shown in (30) with a positive weight λ . We could also add weighting matrices like in the LQR problem instead of the scalar λ .

$$L_1 = \int_0^T (\|\mathcal{I}_3 - R_e(t)\|_F^2 + \lambda \|\Omega_e(t)\|_2^2) dt \quad (30)$$

The procedure to shape the potential function to minimize this loss is shown in Fig. 1. We first start by forming a set of initial conditions around which the system is expected to start. The set of parameters defining the neural network potential functions will be denoted by $\theta = [\theta_c, \theta_d]$ with θ_c denoting the parameters of the conservative potential function and θ_d denoting the parameters of the dissipative potential function. Since any potential with random initialization (of course with some non-negative weights which can be taken care of by softplus function in PyTorch) becomes a valid potential function, we can obtain the corresponding stabilizing elastic and damping forces and integrate the system forward to obtain the error trajectories for each of the initial conditions for a user-defined fixed time T without the fear of trajectories blowing up. Here we have shown it for SO(3) error dynamics in (24), but any of the error dynamics can be used here depending on the system and situation of interest. A mean loss is

computed by taking the average of individual losses corresponding to the trajectory for each initial condition. Next, standard back-propagation algorithms with the choice of gradient descent, such as Stochastic Gradient Descent (SGD), RMSprop, and ADAM, can be used to update the parameters of the potential functions. The function $\alpha(\cdot)$ is used to represent the choice of our optimizer. These updated potential functions generate the updated elastic and damping forces and the system is integrated forward for all the initial conditions again. This cycle is repeated till convergence or any other user-specified termination criterion. Note that since the dynamics evolve on manifolds, a variational integrator like Kobilarov and Marsden (2011); Kobilarov et al. (2009); Prakash (2022) could be a better choice to integrate the system forward as they preserve the geometry of the manifold. In Fig.1, the dynamical

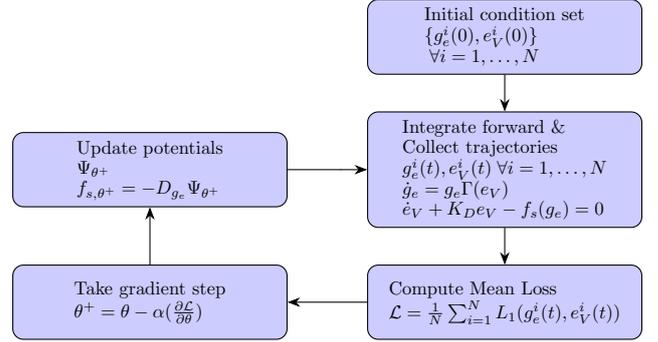


Fig. 1. Flow chart showing the training process for potential functions for a satellite orientation control problem.

equations in the integration block can be replaced by (28) and the loss function can be updated appropriately to account for SE(3) trajectories.

7. SIMULATION RESULTS

In this section, we consider the problem of regulating the error g_e to \mathcal{I}_4 of a rigid body evolving on SE(3) according to the error dynamics (28). A neural network with 3 hidden layers of 7 neurons each, using Kaiming initialization within bounds $(-\sqrt{3}, \sqrt{3})$, was trained on 50 initial conditions. Initial conditions were set with rotation errors normally distributed around π radians from the identity matrix, position errors normally distributed around $[1, 1, 1]$ with unit variance and velocity errors with zero mean and unit variance. The training utilized the Adam optimizer at a learning rate of 0.01 to minimize the loss function in (30) with $\lambda = 1$. Results in Fig.2 illustrates the evolution of configuration error trajectories for neural network potential functions for a random initialization without any training. Fig. 3 shows improved trajectories after the training process. We can see that the trajectories remain stable despite random initializations even in this case. In all the plots, we will use $\Psi_F(g_e) = \frac{1}{2} \|\mathcal{I}_4 - g_e\|_F^2$ to evaluate the configuration error.

Fig. 4 shows the trajectories with the trained potential functions but with large initial errors. The benchmark potential functions used are $\Psi_b = \frac{1}{2} \|\mathcal{I}_4 - g_e\|_F^2$ and $\mathcal{R}_b = \frac{1}{2} \|e_V\|_2^2$.

REFERENCES

- Amos, B., Xu, L., and Kolter, J.Z. (2016). Input convex neural networks. In *International Conference on Machine Learning*.
- Behrmann, J., Grathwohl, W., Chen, R.T.Q., Duvenaud, D., and Jacobsen, J.H. (2019). Invertible residual networks.
- Bhat, S.P. and Bernstein, D.S. (2000). A topological obstruction to continuous global stabilization of rotational motion and the unwinding phenomenon.
- Bullo, F. and Murray, R.M. (1995). Proportional derivative (PD) control on the euclidean group.
- Bullo, F. and Murray, R.M. (1999). Tracking for fully actuated mechanical systems: a geometric framework. *Automatica*, 35(1), 17–34.
- Hogan, N. (1985). Impedance control: An approach to manipulation: Part ii—implementation.
- Khatib, O. (1987). A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1), 43–53.
- Kobilarov, M., Crane, K., and Desbrun, M. (2009). Lie group integrators for animation and control of vehicles. *ACM Trans. Graph.*, 28(2).
- Kobilarov, M.B. and Marsden, J.E. (2011). Discrete geometric optimal control on lie groups. *IEEE Transactions on Robotics*, 27(4), 641–655.
- Kotaru, P., Edmonson, R., and Sreenath, K. (2020). Geometric l1 adaptive attitude control for a quadrotor unmanned aerial vehicle.
- Kronander, K. and Billard, A. (2016). Stability considerations for variable impedance control. *IEEE Transactions on Robotics*, 32(5), 1298–1305.
- Lee, T. (2012). Exponential stability of an attitude tracking control system on so(3) for large-angle rotational maneuvers. *Systems & Control Letters*, 61(1), 231–237.
- Lee, T., Leok, M., and McClamroch, H.N. (2017). Global formulations of lagrangian and hamiltonian dynamics on manifolds. 13.
- Manek, G. and Kolter, J.Z. (2020). Learning stable deep dynamics models. *CoRR*, abs/2001.06116.
- Park, F.C. (1995). Distance metrics on the rigid-body motions with applications to mechanism design.
- Park, F.C., Bobrow, J.E., and Ploen, S.R. (1995). A lie group formulation of robot dynamics. *The International journal of robotics research*, 14(6), 609–618.
- Prakash, N.P.S. (2022). A computational approach for variational integration of attitude dynamics on so(3).
- Prakash, N.P.S., Seo, J., Sreenath, K., Choi, J., and Horowitz, R. (2024). Deep geometric potential functions for tracking on manifolds.
- Seo, J., Prakash, N.P.S., Choi, J., and Horowitz, R. (2024). A comparison between lie group- and lie algebra- based potential functions for geometric impedance control.
- Seo, J., Prakash, N.P., Zhang, X., Wang, C., Choi, J., Tomizuka, M., and Horowitz, R. (2023a). Contact-rich se(3)-equivariant robot manipulation task learning via geometric impedance control. *IEEE Robotics and Automation Letters*.
- Seo, J. et al. (2023b). Geometric impedance control on SE(3) for robotic manipulators. *IFAC World Congress 2023, Yokohama, Japan*.

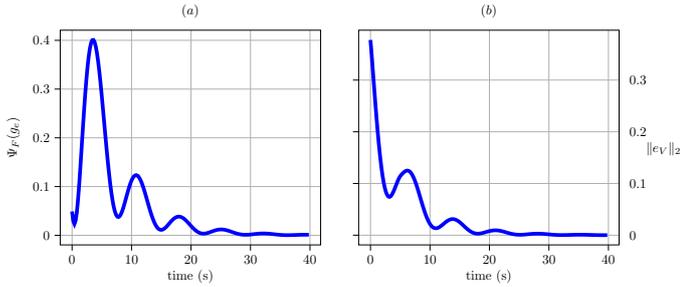


Fig. 2. Randomly Initialized Network: (a) Configuration error trajectory and (b) Corresponding velocity error trajectory for a rigid body with initial rotation error $R_e(0) = 0.1\pi$ rotated along the z-axis, $p_e = [1, 1, 1]^T$ and zero initial velocity error.

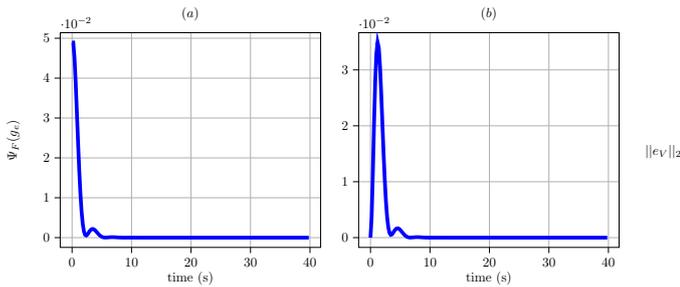


Fig. 3. Trained Network: (a) Configuration error trajectory and (b) Corresponding velocity error trajectory for a rigid body with initial rotation error $R_e(0) = 0.1\pi$ rotated along the z-axis, $p_e = [1, 1, 1]^T$ and zero initial velocity error.

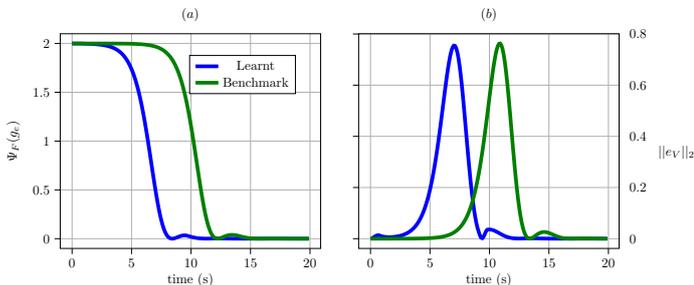


Fig. 4. Trained Network: (a) Configuration error trajectory and (b) Corresponding velocity error trajectory for a rigid body with initial rotation error $R_e(0) = 0.999\pi$ rotated along the z-axis, $p_e = [1, 1, 1]^T$ and zero initial velocity error.

8. CONCLUSIONS

In this paper, a generic design methodology for designing trainable conservative and dissipative potential functions for fully actuated dynamical systems evolving on manifolds has been presented. The corresponding elastic and damping wrenches/forces obtained from the potential functions were shown to be stabilizing irrespective of the initialization of the network. These wrenches/forces were used to formulate the loss functions which can be minimized using gradient descent algorithms to achieve user’s requirements. Analysis of the potential functions and stability for various manifolds of interest has also been presented. Finally, the methodology was demonstrated on two problems - a satellite orientation control and a robotic manipulator impedance control.