

## MIT Open Access Articles

*A Theoretical Overview of Neural Contraction Metrics for Learning-based Control with Guaranteed Stability*

The MIT Faculty has made this article openly available. **Please share** how this access benefits you. Your story matters.

**Citation:** Tsukamoto, Hiroyasu, Chung, Soon-Jo, Slotine, Jean-Jacques and Fan, Chuchu. 2021. "A Theoretical Overview of Neural Contraction Metrics for Learning-based Control with Guaranteed Stability." 2021 60th IEEE Conference on Decision and Control (CDC).

**As Published:** 10.1109/CDC45484.2021.9682859

**Publisher:** Institute of Electrical and Electronics Engineers (IEEE)

**Persistent URL:** <https://hdl.handle.net/1721.1/145339>

**Version:** Author's final manuscript: final author's manuscript post peer review, without publisher's formatting or copy editing

**Terms of use:** Creative Commons Attribution-Noncommercial-Share Alike



# A Theoretical Overview of Neural Contraction Metrics for Learning-based Control with Guaranteed Stability

Hiroyasu Tsukamoto\*, Soon-Jo Chung\*, Jean-Jacques Slotine†, and Chuchu Fan‡

**Abstract**—This paper presents a theoretical overview of a Neural Contraction Metric (NCM): a neural network model of an optimal contraction metric and corresponding differential Lyapunov function, the existence of which is a necessary and sufficient condition for incremental exponential stability of non-autonomous nonlinear system trajectories. Its innovation lies in providing formal robustness guarantees for learning-based control frameworks, utilizing contraction theory as an analytical tool to study the nonlinear stability of learned systems via convex optimization. In particular, we rigorously show in this paper that, by regarding modeling errors of the learning schemes as external disturbances, the NCM control is capable of obtaining an explicit bound on the distance between a time-varying target trajectory and perturbed solution trajectories, which exponentially decreases with time even under the presence of deterministic and stochastic perturbation. These useful features permit simultaneous synthesis of a contraction metric and associated control law by a neural network, thereby enabling real-time computable and probably robust learning-based control for general control-affine nonlinear systems.

## I. INTRODUCTION

Since the development of reinforcement learning and neural networks [1], [2], leveraging Artificial Intelligence (AI) and Machine Learning (ML) methods in the field of systems and control theory has been an active field of research. Although successful, conventional black-box AI and ML approaches often lack formal robustness and stability guarantees, which are essential in implementing safety-critical control schemes for, e.g., robotic and aerospace systems.

*Related Work:* Contraction theory is one of the most powerful mathematical tools for providing such guarantees for learning-based control frameworks [3], [4]. Generalizing Lyapunov theory, it studies differential dynamics of a nonlinear system using a Riemannian contraction metric and its uniformly positive definite matrix, thereby designing a differential Lyapunov function to characterize a necessary and sufficient condition for exponential convergence of any couple of the system trajectories. Due to the differential nature of contraction theory which allows utilizing LTV system-type approaches [4]–[8], its nonlinear stability analysis reduces to the problem of finding a contraction metric that satisfies a Linear Matrix Inequality (LMI) [9].

Obtaining contraction metrics analytically is challenging for general nonlinear systems, and thus several techniques have been proposed to find them numerically using the LMI property. In particular, it is proposed in [10] that sufficient conditions for incremental exponential stabilizability of nonlinear

systems can be expressed as convex constraints, enabling systematic synthesis of a differential feedback control. Such an idea is further extended to develop the method of ConVex optimization-based Steady-state Tracking Error Minimization (CV-STEM) [4]–[7], which computes a contraction metric and its associated feedback control and state estimation laws via convex optimization for optimal deterministic/stochastic disturbance attenuation.

A critical limitation of these optimization-based schemes is their computational complexity. In essence, a Neural Contraction Metric (NCM) and its extensions [4], [6]–[8], [11], [12] are developed to model the CV-STEM contraction metric using a Deep Neural Network (DNN), achieving real-time implementable control, estimation, and motion planning of nonlinear systems with guaranteed robustness and stability. We again remark that such formal guarantees are difficult to quantify for black-box AI and ML techniques without considering a contracting stability property.

*Contribution:* This paper presents rigorous proofs on the aforementioned theoretical guarantees of the NCM methods [4], [6]–[8], [11], [12], which provide an explicit exponential bound on the distance between a target trajectory and solution trajectories with the NCM learning-based control, even under the presence of learning errors and deterministic/stochastic perturbation. We also derive one explicit way to optimally sample NCM training data by formulating the CV-STEM, to design contracting differential feedback control that minimizes the steady-state of the computed bound for optimal disturbance attenuation. Furthermore, we provide an algorithmic overview on DNN training for simultaneous synthesis of the contraction metric and control, to see how the NCM robustness and stability guarantees corroborate the success of learning-based control.

*Notation:* For  $x \in \mathbb{R}^n$  and  $A \in \mathbb{R}^{n \times m}$ , we let  $\|x\|$ ,  $\delta x$ ,  $\|A\|$ , and  $\|A\|_F$  denote the Euclidean norm, infinitesimal displacement at a fixed time, induced 2-norm, and Frobenius norm, respectively. For  $A \in \mathbb{R}^{n \times n}$ , we use  $A \succ 0$ ,  $A \succeq 0$ ,  $A \prec 0$ , and  $A \preceq 0$  for the positive definite, positive semi-definite, negative definite, negative semi-definite matrices, respectively, and  $\text{sym}(A) = (A + A^\top)/2$ . Also,  $\mathbb{E}$  denotes the expected value operator.

## II. NEURAL CONTRACTION METRICS FOR LEARNING-BASED CONTROL [4], [6]–[8], [12]

In this paper, we consider the following smooth non-autonomous (i.e. time-varying) nonlinear system, perturbed by deterministic disturbances  $d(x, t)$  with  $\sup_{x,t} \|d(x, t)\| = \bar{d}$  and Gaussian white noise  $\mathcal{W}(t)$  with  $\|G(x, t)\|_F \leq \bar{g}$ :

$$dx = h(x, u, t)dt + d(x, t)dt + G(x, t)d\mathcal{W}(t) \quad (1)$$

\* Graduate Aerospace Laboratories, Caltech, Pasadena, CA, {htsukamoto, sjchung}@caltech.edu.

† Nonlinear Systems Laboratory, MIT, Cambridge, MA, jjs@mit.edu.

‡ Reliable Autonomous System Laboratory, MIT, Cambridge, MA, chuchu@mit.edu.

where  $t \in \mathbb{R}_+$ ,  $x: \mathbb{R}_+ \mapsto \mathbb{R}^n$  is the system state,  $u \in \mathbb{R}^m$  is the system control input,  $h: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}_+ \rightarrow \mathbb{R}^n$  is a known smooth function,  $d: \mathbb{R}^n \times \mathbb{R}_+ \mapsto \mathbb{R}^n$  and  $G: \mathbb{R}^n \times \mathbb{R}_+ \rightarrow \mathbb{R}^{n \times w}$  are unknown bounded functions for external disturbances,  $\mathcal{W}: \mathbb{R}_+ \mapsto \mathbb{R}^w$  is a  $w$ -dimensional Wiener process, and we consider the case where  $\bar{d}, \bar{g} \in [0, \infty)$  are given. Let us first introduce the following definition of contraction [3].

**Definition 1.** The system (1) is said to be contracting if there exist a uniformly positive definite matrix  $M(x, t) = \Theta(x, t)^T \Theta(x, t) \succ 0$ ,  $\forall x, t$  and a feedback control law  $u = u^*(x, x_d, u_d, t)$  s.t. the following condition holds  $\forall x, t$ :

$$\dot{M} + 2 \text{sym} \left( M \frac{\partial h(x, u^*(x, x_d, u_d, t), t)}{\partial x} \right) + \Xi(x, t) \preceq 0 \quad (2)$$

where  $\Xi(x, t) \succ 0$ ,  $\forall x, t$ ,  $M = M(x, t)$ , and  $x_d$  is the target trajectory of (1) given as follows:

$$\dot{x}_d = h(x_d, u_d, t) \text{ s.t. } u_d = u^*(x_d, x_d, u_d, t). \quad (3)$$

We call  $M$  and  $u^*$  satisfying (2) a contraction metric and contracting control law, respectively.

It is shown that (2) is a necessary and sufficient condition for incremental exponential stability of the system trajectories with respect to each other [3], [4].

#### A. Problem Formulation

Finding an optimal contraction metric  $M$  for general nonlinear systems is challenging and could involve solving optimization problems at each time instant [5], [13] (see Sec. III), which is not suitable for systems with limited computational capacity. A Neural Contraction Metric (NCM) and its extensions [4], [6]–[8], [12] have been proposed to approximate a contracting control law  $u^*$ , i.e., a computationally-expensive control law parameterized by the contraction metric given by (2), utilizing a Deep Neural Network (DNN).

**Definition 2.** A Neural Contraction Metric (NCM) is a DNN model of a contraction metric given by Definition 1.

We consider the problem of modeling the contracting control law  $u^*$  by a learning-based control law  $u_L(x, x_d, u_d, t)$  parameterized by the NCM of Definition 2 (see [5], [10] and Sec. III for how to obtain  $u^*$ ). Let  $x$  be the trajectory of (1) controlled by  $u = u_L$ , and  $x_d$  be the target trajectory of (3). Since we have  $h(x, u_L, t) = h(x, u^*, t) + (h(x, u_L, t) - h(x, u^*, t))$ , a virtual system of  $q$ , which has  $q = x$  and  $q = x_d$  as its particular solutions, is given as follows:

$$dq = h(q, u^*(q, x_d, u_d, t), t)dt + d_q(q, t)dt + G_q(q, t)d\mathcal{W} \quad (4)$$

where  $d_q$  and  $G_q$  are parameterized linearly to verify  $d_q(q = x_d, t) = 0$ ,  $d_q(q = x, t) = h(x, u_L(x, x_d, u_d, t), t) - h(x, u^*(x, x_d, u_d, t), t) + d(x, t)$ ,  $G_q(q = x_d, t) = 0$ , and  $G_q(q = x, t) = G(x, t)$ .

**Assumption 1.** In this section, we assume the following.

- For all  $(x, x_d, u_d, t) \in S$ ,  $u_L$  is learned to satisfy

$$\|u_L(x, x_d, u_d, t) - u^*(x, x_d, u_d, t)\| \leq \varepsilon_{\ell 0} + \varepsilon_{\ell 1} \int_{x_d}^x \|\delta q\| \quad (5)$$

where  $S = S_x \times S_x \times S_u \times S_t$ ,  $S_x \subset \mathbb{R}^n$ ,  $S_u \subset \mathbb{R}^m$ , and  $S_t \subset \mathbb{R}_+$  are some given compact sets, and  $\int_{x_d}^x \|\delta q\|$  is a path integral from  $x_d$  to  $x$  parameterized by  $q$  of (4).

- $h$  of (1) is Lipschitz with respect to  $u$ , i.e.,  $\exists L_u \in [0, \infty)$  s.t.

$$\|h(x, u, t) - h(x, u', t)\| \leq L_u \|u - u'\|, \quad \forall u, u' \in \mathbb{R}^m \quad (6)$$

for all  $x \in S_x$  and  $t \in S_t$ .

- Let  $S_{xt} = S_x \times S_t$ . When  $G \neq 0$  in (1),  $\partial M / \partial x_i$  of  $M$  in (2) is Lipschitz with respect to  $x$ , i.e.,  $\exists L_m \in [0, \infty)$  s.t.

$$\left\| \frac{\partial M}{\partial x_i}(x, t) - \frac{\partial M}{\partial x_i}(x', t) \right\| \leq L_m \|x - x'\|, \quad \forall (x, t), (x', t) \in S_{xt}. \quad (7)$$

**Remark 1.** (5) can be empirically guaranteed, e.g., by DNNs, which have been shown to generalize well to the set of unseen events that are from almost the same distribution as their training set [14], [15], and consequently, obtaining a tighter and more general bound for the learning error has been an active field of research [16]. The condition (5) has thus become a common assumption in analyzing the performance of learning-based control [4], [8], [12], [15].

#### B. Nonlinear Robustness and Stability Analysis

The following theorem rigorously studies robustness and stability of the NCM-parameterized control law  $u_L$ .

**Theorem 1.** Suppose that Assumption 1 holds. Suppose also that the contraction metric  $M(x, t)$  of (2) is bounded as

$$\underline{m}I \preceq M(x, t) \preceq \bar{m}I, \quad \forall x, t \quad (8)$$

for  $\underline{m}, \bar{m} \in (0, \infty)$ , and  $u^*$  of (2) is given with  $\Xi$  defined as

$$\Xi(x, t) = 2\alpha M(x, t) + \alpha_s I \quad (9)$$

where  $\exists \alpha \in (0, \infty)$  is the contraction rate and  $\alpha_s = L_m \bar{g}^2 (\alpha_G + 1/2)$  with an arbitrary constant  $\alpha_G \in (0, \infty)$  (see (14)). If the learning error  $\varepsilon_{\ell 1}$  of (5) and another arbitrary constant  $\alpha_d \in (0, \infty)$  (see (15)) are selected to satisfy

$$\exists \alpha_\ell \in (0, \infty) \text{ s.t. } \alpha_\ell = \alpha - (\alpha_d/2 + L_u \varepsilon_{\ell 1} \sqrt{\bar{m}/\underline{m}}) > 0 \quad (10)$$

then we have the following bound for all  $(x, t) \in S_x \times S_t$ :

$$\mathbb{E} [\|x(t) - x_d(t)\|^2] \leq \frac{C}{2\alpha_\ell \underline{m}} + \frac{\mathbb{E}[V_{sl}(0)]e^{-2\alpha_\ell t}}{\underline{m}} \quad (11)$$

where  $V_{sl}(t) = \int_{x_d}^x \delta q(t)^T M(q(t), t) \delta q(t)$ ,  $C = \bar{g}^2 (2\alpha_G^{-1} + 1) + (L_u \varepsilon_{\ell 0} + \bar{d})^2 \alpha_d^{-1}$ ,  $\bar{d}$ ,  $\bar{g}$ , and  $q$  are given in (1) and (4),  $x$  is the trajectory of (1) controlled by  $u = u_L$ , and  $x_d$  is the target trajectory given by (3).

*Proof.* Since  $\partial M / \partial x_i$  is Lipschitz as in (7), we have  $\|\partial^2 M / (\partial x_i \partial x_j)\| \leq L_m$  and  $\|\partial M / \partial x_i\| \leq \sqrt{2L_m \bar{m}}$  using (8) as derived in [7]. Also,  $d_q$  of (4) is bounded as

$$\|d_q(x, t)\| \leq L_u (\varepsilon_{\ell 0} + \varepsilon_{\ell 1} \int_{x_d}^x \|\delta q\|) + \bar{d} \quad (12)$$

by the Lipschitz condition (6) and learning error bound (5). Let  $\mathcal{L}$  be the infinitesimal differential generator in [17, p. 15]. Since we have  $\int_{x_d}^x \|\delta q\| \leq V_\ell / \sqrt{\underline{m}}$  and  $V_\ell^2 \leq V_{sl}$  [4] for

$V_\ell = \int_{x_d}^x \|\Theta \delta q\|$  and  $M = \Theta^\top \Theta$ , computing  $\mathcal{L}V_{sl}$  using these bounds as in [5] yields

$$\begin{aligned} \mathcal{L}V_{sl} &\leq \int_{x_d}^x \delta q^\top (\dot{M} + 2\text{sym}(Mh_x)) \delta q + 2L_u \varepsilon_{\ell 1} \sqrt{\bar{m}/\underline{m}} V_{sl} \\ &\quad + 2\sqrt{\bar{m}d_\varepsilon} V_\ell + \bar{g}^2(\bar{m} + \int_{x_d}^x L_m \|\delta q\|^2/2 + 2\sqrt{2L_m \bar{m}} \|\delta q\|) \end{aligned} \quad (13)$$

where  $h_x = \partial h / \partial x$  and  $d_\varepsilon = L_u \varepsilon_{\ell 0} + \bar{d}$ . Using the relation  $2ab \leq c^{-1}a^2 + cb^2$  which holds for any  $a, b \in \mathbb{R}$  and  $c \in (0, \infty)$ , we have that

$$2\sqrt{\bar{m}d_\varepsilon} \|\Theta \delta q\| \leq \alpha_d \|\Theta \delta q\|^2 + \bar{m}d_\varepsilon^2 \alpha_d^{-1} \quad (14)$$

$$2\sqrt{2L_m \bar{m}} \|\delta q\| \leq L_m \alpha_G \|\delta q\|^2 + 2\bar{m} \alpha_G^{-1} \quad (15)$$

for any  $\alpha_d, \alpha_G \in (0, \infty)$ . Substituting (14) and (15) into (13) along with the contraction condition (2) for  $u^*$  yields

$$\mathcal{L}V_{sl} \leq \int_{x_d}^x \delta q^\top (-\Xi + (\alpha_d + 2L_u \varepsilon_{\ell 1} \sqrt{\chi})M + \alpha_s I) \delta q + \bar{m}C$$

where  $\chi = \bar{m}/\underline{m}$ ,  $\alpha_s = L_m \bar{g}^2(\alpha_G + 1/2)$ , and  $C = \bar{g}^2(2\alpha_G^{-1} + 1) + \bar{d}_\varepsilon^2 \alpha_d^{-1}$ . Thus, by definition of  $\Xi$  in (9), we have  $\mathcal{L}V_{sl} \leq -2\alpha_\ell V_{sl} + \bar{m}C$  due to the condition (10). This results in the desired relation (11) due to Theorem 1 of [5].  $\square$

Theorem 1 states that, as long as we have a contracting controller  $u^*$  which satisfies (2) of Definition 1, the NCM controller  $u_L$  of (5) ensures that the mean-squared distance from any target trajectory  $x_d$  controlled by  $u_d = u^*(x_d, x_d, u_d, t)$  to the one controlled by  $u_L$  with deterministic and stochastic perturbation is exponentially bounded in time, even under the presence of the learning error  $\varepsilon_{\ell 0}$  and  $\varepsilon_{\ell 1}$ . Such an explicit exponential bound is difficult to obtain for systems without a contraction property.

In particular, when there is no stochastic disturbance in the system (1), i.e.,  $G = 0$ , we have the following result.

**Theorem 2.** Suppose (5) and (6) of Assumption 1 hold for  $u^*$  that satisfies (2) with  $\Xi = 2\alpha M(x, t)$ , where  $\exists \alpha \in (0, \infty)$  is the contraction rate. Now let  $x_d$  be the target trajectory of (3), and  $x$  be the trajectory controlled by  $u_L$  with  $G = 0$ . If  $M$  is bounded as in (8), and if the learning error  $\varepsilon_{\ell 1}$  of (5) is sufficiently small to satisfy

$$\exists \alpha_\ell \in (0, \infty) \text{ s.t. } \alpha_\ell = \alpha - L_u \varepsilon_{\ell 1} \sqrt{\bar{m}/\underline{m}} > 0 \quad (16)$$

then we have the following bound:

$$\|x(t) - x_d(t)\| \leq \frac{\|V_\ell(0)\|}{\sqrt{\bar{m}}} e^{-\alpha_\ell t} + \frac{L_u \varepsilon_{\ell 0} + \bar{d}}{\alpha_\ell} \sqrt{\frac{\bar{m}}{\underline{m}}} (1 - e^{-\alpha_\ell t}) \quad (17)$$

where  $V_\ell(t) = \int_{x_d}^x \|\Theta(q(t), t) \delta q(t)\|$  and  $q$  is given by (4) with the same  $d_q$  and  $G_q = 0$  (note that this virtual system indeed has  $q = x$  and  $q = x_d$  as its particular solutions).

*Proof.* Let  $V = \delta q^\top M \delta q$ . Using (13) computed in the proof of Theorem 1, along with the bound on  $d_q$  (12) and the contraction condition (2) for  $\Xi = 2\alpha M$ , we have that

$$\dot{V} \leq -2\alpha V + 2\sqrt{\bar{m}}(L_u(\varepsilon_{\ell 0} + \varepsilon_{\ell 1} V_\ell / \sqrt{\bar{m}}) + \bar{d}) \|\Theta \delta q\| \quad (18)$$

where the relation  $\int_{x_d}^x \|\delta q\| \leq V_\ell / \sqrt{\bar{m}}$  is used. Since we have  $\dot{V} = \|\Theta \delta q\| (d \|\Theta \delta q\| / dt)$  and  $V = \|\Theta \delta q\|^2$ , (18) implies that

$\dot{V}_\ell \leq -(\alpha - L_u \varepsilon_{\ell 1} \sqrt{\bar{m}/\underline{m}}) V_\ell + \sqrt{\bar{m}}(L_u \varepsilon_{\ell 0} + \bar{d})$ , which results in (17) by applying the comparison lemma [18, pp.102-103, pp.350-353] as long as the condition (16) holds.  $\square$

### III. A CONVEX OPTIMIZATION APPROACH FOR NCM ROBUST CONTROL [4]–[8]

Although Theorems 1 and 2 are already useful for systems with a known contraction metric and corresponding differential Lyapunov function, finding them for general nonlinear systems is not always straightforward. This section delineates one approach, called ConVex optimization-based Steady-state Tracking Error Minimization (CV-STEM), to optimally design  $u^*$  and  $M$  of Theorem 1 for control-affine nonlinear systems, thereby demonstrating how to construct the NCM of Definition 2 in practice for probably stable learning-based robust control.

Let us consider the following control-affine nonlinear system:

$$dx = (f(x, t) + B(x, t)u)dt + d(x, t)dt + G(x, t)d\mathcal{W}(t) \quad (19)$$

where  $t, x, u, d, G$ , and  $\mathcal{W}$  are as defined in (1), and  $f: \mathbb{R}^n \times \mathbb{R}_+ \rightarrow \mathbb{R}^n$  and  $B: \mathbb{R}^n \times \mathbb{R}_+ \rightarrow \mathbb{R}^{n \times m}$  are known smooth functions.

**Remark 2.** If  $h(x, u, t) - (f(x, t) + B(x, t)u)$  is bounded in (1) and (19), we can still apply the technique to be discussed in this section even for control non-affine systems (1), by treating this modeling error as external disturbance  $d$  in (19). For situations where such an assumption is not valid, we could apply (learning-based) adaptive control [8], [15], [19], [20] to estimate the non-affine part of the dynamics.

#### A. Contraction Theory-based Robust Control (CV-STEM)

We design a contracting control law  $u = u^*$  for (19) by the following differential feedback control law [10] (see Remark 3 for more options):

$$u^*(x, x_d, u_d, t) = u_d - \int_{x_d}^x R(q, t)^{-1} B(q, t)^\top M(q, t) \delta q \quad (20)$$

where  $R(x, t) \succ 0$  is a given weight matrix,  $M(x, t) \succ 0$  is a contraction metric to be defined in Theorem 3,  $q$  is a smooth path that connects  $x$  to  $x_d$  as in (4), and  $x_d$  is a given target trajectory of (3) with  $h(x, u, t) = f(x, t) + B(x, t)u$ . Note that (20) yields  $\delta u^* = -R(x, t)^{-1} B(x, t)^\top M(x, t) \delta x$ .

**Remark 3.** We could utilize other types of control laws in (20), e.g.,  $\delta u = k(x, \delta x, u, t)$  (see Sec. IV) [10] or  $u = u_d - K(x, x_d, u_d, t)(x - x_d)$  [5]–[8]. See [4] for their trade-offs.

The CV-STEM is for designing  $u^*$  and  $M$  of Theorem 1 explicitly by (20) to minimize the steady-state bound of (11).

**Theorem 3.** Suppose that the contraction metric  $M(x, t) = W(x, t)^{-1} \succ 0$  of (20) is designed by the following convex optimization (CV-STEM) [4]–[7], which minimizes the steady-state upper bound of (11):

$$J_{CV}^* = \min_{v > 0, \chi \in \mathbb{R}, \bar{W} \succ 0} \frac{C}{2\alpha_\ell} \chi \text{ s.t. (22), (23), and (24)} \quad (21)$$

with the convex constraints (22), (23), and (24) given as

$$\begin{bmatrix} H(v, \bar{W}) + 2\alpha\bar{W} & \bar{W} \\ \bar{W} & v\alpha_s^{-1}I \end{bmatrix} \preceq 0, \forall x, t \quad (22)$$

$$-\partial_{b_i}\bar{W} + 2\text{sym}\left(\frac{\partial b_i}{\partial x}\bar{W}\right) = 0, \forall i = 1, \dots, m, \forall x, t \quad (23)$$

$$I \preceq \bar{W}(x, t) \preceq \chi I, \forall x, t \quad (24)$$

where  $\partial_p F = \sum_{k=1}^n (\partial F / \partial x_k) p_k$  for  $p(x, t) \in \mathbb{R}^n$  and  $F(x, t) \in \mathbb{R}^{n \times n}$ ,  $v = \bar{m}$ ,  $\chi = \bar{m}/\underline{m}$ ,  $\bar{W} = vW$ ,  $f$  and  $B$  are given in (19),  $b_i$  is the  $i$ th column of  $B$ , and

$$H(v, \bar{W}) = -\frac{\partial \bar{W}}{\partial t} - \partial_f \bar{W} + 2\text{sym}\left(\frac{\partial f}{\partial x}\bar{W}\right) - vBR^{-1}B^\top.$$

with the other variables defined in Theorem 1. If  $u^*$  is designed as (20) for  $M$  of (21), the contraction condition (2) of Definition 1 and (8) of Theorem 1 hold with  $\Xi$  given by (9),  $h(x, u, t) = f(x, t) + B(x, t)u$ , and  $u = u^*$ .

*Proof.* Applying Schur's complement lemma [9, pp. 7] to the constraint (22), we have that

$$H(v, \bar{W}) + 2\alpha\bar{W} - \alpha_s\bar{W}^2/v \preceq 0. \quad (25)$$

It can be easily verified that, by multiplying the constraints (25), (23), and (24) by  $M = W^{-1}$  from both sides and by  $v^{-1}$ , they can be equivalently rewritten as [5]

$$\frac{\partial M}{\partial t} + \partial_f M + 2\text{sym}(M(\partial f / \partial x)) - MBR^{-1}B^\top M \preceq -\Xi \quad (26)$$

$$\partial_{b_i} M + 2\text{sym}(M(\partial b_i / \partial x)) = 0 \quad (27)$$

$$\underline{m}I \preceq M \preceq \bar{m}I \quad (28)$$

respectively, for  $\Xi$  of (9), where (28) indicates that (24) is indeed equivalent to (8). Combining (26) and (27), we get

$$\dot{M} + 2\text{sym}(M(\partial(f(x, t) + B(x, t)u^*)/\partial x)) \preceq -\Xi$$

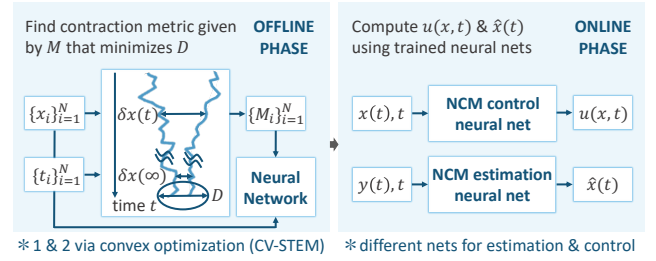
which implies the contraction condition (2) holds for  $h(x, u, t) = f(x, t) + B(x, t)u$  and  $u = u^*$ .

Furthermore, it can be seen that the problem (21) minimizes the steady-state upper bound of (11) due to the relation  $\lim_{t \rightarrow \infty} \mathbb{E}[\|x(t) - x_d(t)\|^2] = (C/(2\alpha_\ell))(\bar{m}/\underline{m}) = (C/(2\alpha_\ell))\chi$ . Note that (21) is convex as the objective is affine in  $\chi$ , and (22), (23), and (24) are linear matrix inequalities in terms of  $v$ ,  $\chi$ , and  $\bar{W}$  [9, pp. 7].  $\square$

### B. NCM Construction using CV-STEM

Now that we can construct the contracting control law  $u^*$  of Theorem 1 by using Theorem 3, we are ready to explicitly design the NCM of Definition 2. Since the NCM only requires one DNN evaluation at each time instant without solving any optimization problems unlike the CV-STEM of Theorem 3, it enables real-time computation of optimal feedback control using (20) in most engineering and science applications. We formally derive its robustness and stability guarantees in the following theorems.

**Theorem 4.** Let  $u_L$  be a learning-based controller, parameterized by the NCM of Definition 2, which approximates the contracting control law  $u^*$  of (20). If Assumption 1 holds for such



\* 1 & 2 via convex optimization (CV-STEM) \* different nets for estimation & control  
Fig. 1. Illustration of NCM ( $x$ : system state;  $M$ : positive definite matrix that defines optimal contraction metric;  $x_i$  and  $M_i$ : sampled  $x$  and  $M$ ;  $\hat{x}$ : estimated system state;  $y$ : measurement; and  $u$ : system control input). Note that the target trajectory ( $x_d, u_d$ ) is omitted in the figure for simplicity. (see [4], [6] for state estimation).

$u_L$ ,  $u^*$ , and  $M$  of Theorem 3 with  $h(x, u, t) = f(x, t) + B(x, t)u$  of (19), then Theorem 1 holds, i.e., the system trajectory of (19) controlled by  $u_L$  exponentially converges to a ball around the target trajectory  $x_d$  of (3) with  $h(x, u, t) = f(x, t) + B(x, t)u$ .

*Proof.* Since  $u^*$  of (20) with  $M$  of (21) guarantees the assumptions of Theorem 1 due to Theorem 3, the exponential bound (11) holds as long as Assumption 1 is satisfied.  $\square$

Although Theorem 4 considers the modeling error on the learned control law,  $\|u_L - u^*\|$ , it is also useful to study how the learning error on the contraction metric,  $\|M_L - M\|$ , affects the NCM stability performance.

**Theorem 5.** Let  $M_L$  be the NCM of Definition 2 that models  $M$  of Theorem 3. Suppose that (7) holds for such  $M$ , and that the system (19) is controlled by (20) with  $M$  replaced by  $M_L$ . Suppose also that  $\exists \bar{b}, \bar{\rho} \in [0, \infty)$  s.t.  $\|B(x, t)\| \leq \bar{b}$  and  $\|R^{-1}(x, t)\| \leq \bar{\rho}$ ,  $\forall (x, t) \in S_x \times S_t$ , for  $B$  in (19) and  $R$  in (20), where  $S_x$  and  $S_t$  are given compact sets of (5). If the NCM satisfies

$$\|M_L(x, t) - M(x, t)\| \leq \varepsilon_\ell, \forall (x, t) \in S_x \times S_t$$

for  $\exists \varepsilon_\ell \in [0, \infty)$ , Theorem 1 holds for  $\varepsilon_{\ell 0} = 0$  and  $\varepsilon_{\ell 1} = \bar{\rho}\bar{b}^2\varepsilon_\ell$ .

*Proof.* This follows from Theorem 4, due to the facts that the learning error of (5) is given as  $\|u_L - u^*\| \leq \bar{\rho}\bar{b}\varepsilon_\ell \int_{x_d}^x \|\delta q\|$  due to (20), and the Lipschitz constant of (6) is as  $L_u = \bar{b}$ .  $\square$

Theorems 4 and 5 indeed guarantee robustness and incremental stability of the NCM learning-based control law  $u_L$ , which imitates the contraction theory-based robust control  $u^*$  of Theorem 3 for its real-time implementability.

## IV. LEARNING CERTIFIED CONTROL USING CONTRACTION METRICS [11]

This section delineates one framework to train a DNN for jointly synthesizing a contraction metric and contraction control law with the robustness and stability guarantees of Theorems 4 and 5, treating the contraction constraints (2) and (8) as the loss functions. Intuitively, this can be viewed as a way to solve (21) of Theorem 3 directly by a DNN, while also avoiding the integral evaluation required in (20).

**Remark 4.** Although here we consider the system (19) with  $G = 0$ , stochastic perturbation could be incorporated in a similar way using the constraints introduced in (22) and (23).

For deterministic systems (19) with  $G = 0$ , it is shown in [10] that the following conditions weaker than (22) and (23) are sufficient for the existence of a contracting differential feedback controller, which satisfies the contraction condition (2) of Definition 1 with  $\Xi = 2\alpha M = 2\alpha W^{-1}$ :

$$B_{\perp}^{\top} \left( -\frac{\partial W}{\partial t} - \partial_f W + 2 \text{sym} \left( \frac{\partial f}{\partial x} W \right) + 2\alpha W \right) B_{\perp} \prec 0 \quad (29)$$

$$B_{\perp}^{\top} \left( \partial_{b_i} W - 2 \text{sym} \left( \frac{\partial b_i}{\partial x} W \right) \right) B_{\perp} = 0, \forall j, x, t \quad (30)$$

where  $B_{\perp}(x, t)$  is an annihilator matrix of  $B(x, t)$  satisfying  $B_{\perp} B = 0$ , and the other notations and variables are as defined in Theorem 3. Thus, in this section, we consider the constraints (29) and (30), instead of (22) and (23), without specifying the form of the contracting control  $u^*$  as in (20).

#### A. Cost Functions in Neural Network Training

The purpose of this section is to find a DNN-based controller  $u_L(x, x_d, u_d, t; \theta_u)$  that models a contracting control law  $u^*$  and dual contraction metric  $W_L$  designed as

$$W_L(x, t; \theta_w) = \Theta_L(x, t; \theta_{\vartheta})^{\top} \Theta_L(x, t; \theta_{\vartheta}) + \bar{m}_L^{-1} I \quad (31)$$

that models  $W = M^{-1}$  of Theorem 3, where  $\theta_u$  and  $\theta_w = \{\theta_{\vartheta}, \bar{m}_L\}$  with  $\bar{m}_L > 0$  are hyper-parameters. Note that by definition of  $W_L$  in (31), it is already a positive definite matrix satisfying  $W_L(x, t; \theta_w) \succeq \bar{m}_L^{-1} I$  as in (8) or (24).

Let  $C_u(x, x_d, u_d, t; \theta_w, \theta_u)$  be the left-hand side of (2) with  $W (= M^{-1}) = W_L$ ,  $u^* = u_L$ ,  $h(x, u, t) = f(x, t) + B(x, t)u$ , and  $\Xi = 2\alpha M$ , and  $\rho(S)$  be the uniform distribution over the compact set of (5). We define the contraction loss as follows:

$$\mathcal{L}_u(\theta_w, \theta_u) = \mathbb{E}_{(x, x_d, u_d, t) \sim \rho(S)} L_{PD}(-C_u(x, x_d, u_d, t; \theta_w, \theta_u)) \quad (32)$$

where  $L_{PD}(\cdot) \geq 0$  is a function for penalizing non-positive definiteness, i.e.,  $L_{PD}(A) = 0$  if and only if  $A \succeq 0$ . Also, since we already have  $W_L \succeq \bar{m}_L^{-1} I$  by (31), the condition (8) on the boundedness of  $W_L$  can be considered by the following loss function with another hyper-parameter  $\underline{m}_L$  in  $\theta_w$ :

$$\mathcal{L}_c(\theta_w) = \mathbb{E}_{(x, x_d, u_d, t) \sim \rho(S)} L_{PD}(\underline{m}_L^{-1} I - W_L(x, t; \theta_w)). \quad (33)$$

Furthermore, we utilize the weaker contraction conditions (29) and (30) also as the following loss functions to provide more guidance for DNN optimization, as simultaneously learning  $u_L$  and  $W_L$  by minimizing  $\mathcal{L}_u$  and  $\mathcal{L}_c$  solely is challenging as demonstrated in [11]:

$$\begin{aligned} \mathcal{L}_{w1}(\theta_w) &= \mathbb{E}_{(x, x_d, u_d, t) \sim \rho(S)} L_{PD}(-C_1(x, t; \theta_w)) \\ \mathcal{L}_{w2}(\theta_w) &= \sum_{j=1}^m \mathbb{E}_{(x, x_d, u_d, t) \sim \rho(S)} \|C_2^j(x, t; \theta_w)\|_F \end{aligned} \quad (34)$$

where  $C_1(x, t; \theta_w)$  and  $\{C_2^j(x, t; \theta_w)\}_{j=1}^m$  are the left-hand side of (29) and (30) with  $W = W_L$ , respectively.

Approximating the expectations of (32)–(34) using sampled data, the empirical loss function for the NCM training is defined as follows:

$$\begin{aligned} \mathcal{L}(\theta_u, \theta_w) &= \frac{1}{N} \sum_{i=1}^N \mathcal{L}_{m,i}(\theta_u, \theta_w) + \mathcal{L}_{w,i}(\theta_w) \\ \mathcal{L}_{m,i} &= L_{PD}(-C_u(\xi_i; \theta_w, \theta_u)) + L_{PD}(\underline{m}_L^{-1} I - W(x_i, t_i; \theta_w)) \\ \mathcal{L}_{w,i} &= L_{PD}(-C_1(x_i, t_i; \theta_w)) + \sum_{j=1}^m \|C_2^j(x_i, t_i; \theta_w)\|_F \end{aligned} \quad (35)$$

where the training samples  $\{\xi_i = (x_i, x_{d,i}, u_{d,i}, t_i)\}_{i=1}^N$  are drawn independently from  $\rho(S)$ , and  $L_{PD}$  is computed as  $L_{PD}(A) = \frac{1}{K} \sum_{i=1}^K \min\{0, -p_i^{\top} A p_i\}$  for a given matrix  $A \in \mathbb{R}^{n \times n}$  and randomly sampled  $K$  points in the set  $\{p_i \in \mathbb{R}^n \mid \|p_i\| = 1\}_{i=1}^K$ . The robustness and stability guarantee of  $u_L$  is given in the following theorem.

**Theorem 6.** Suppose Assumption 1 holds for the DNN-based NCM controller  $u_L$  learned by minimizing the loss function (35). If (19) with  $G = 0$  is controlled by  $u_L$ , (17) of Theorem 2 holds, i.e., the distance between the target trajectory  $x_d$  and perturbed trajectories  $x$  is exponentially bounded.

*Proof.* This follows from the proof of Theorem 2 as long as  $u_L$  is learned to satisfy (5) with the loss function (35).  $\square$

**Remark 5.** The objective function (21) of the CV-STEM method in Theorem 3 can also be considered in (35) as a loss function  $\mathcal{L}_{CV}(\theta_w) = (C/2\alpha_e)(\bar{m}_L/\underline{m}_L)$ , where  $\bar{m}_L$  and  $\underline{m}_L$  are given in (31) and (33). Using it in (35) would augment  $u_L$  of Theorem 6 with some CV-STEM-type optimality property.

#### B. Numerical Simulation

We briefly summarize the simulation results presented in [11] to see how the theoretical guarantees in Theorems 1–6 lead to exponential stabilization of nonlinear systems in practice (more simulation results can be found also in [4], [6]–[8], [12]). The loss function (35) is used for DNN training, and we consider (a) PVTOL: Planar Vertical-Takeoff-Vertical-Landing (PVTOL) system for drones [21], [22]; (b) Quadrotor: Physical quadrotor system [21]; (c) Neural lander: Drone flying close to the ground, where the ground effect is non-negligible [15] (the ground effect is learned from empirical data using a 4-layer neural network); and (d) SEGWAY: Real-world Segway robot [23].

The performance is compared with the Sum of Squares (SoS)-based method [21], Model Predictive Control (MPC) [24], [25], and proximal policy optimization reinforcement learning (RL) algorithm [26].

*1) DNN Training:* For  $\Theta_L$  of the dual metric in (31), we use a 2-layer neural network with 128 neurons. For  $u_L$  of Theorem 6, we use  $u_L(x, x_d, u_d, t; \theta_u) = w_2 \cdot \tanh(w_1 \cdot (x - x_d)) + u_d$  where  $w_1 = w_1(x, x_d; \theta_{u1})$  and  $w_2 = w_2(x, x_d; \theta_{u2})$  are two 2-layer neural networks with 128 neurons with the hyperparameters  $\theta_u = \{\theta_{u1}, \theta_{u2}\}$ . We train the networks for 20 epochs with a training set with 130K samples. See [6], [7], [11], [12] for more details on the NCM dataset generation.



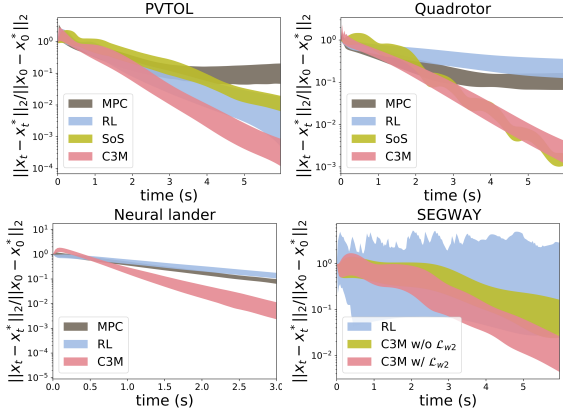


Fig. 2. Normalized tracking error on the benchmarks using different approaches. The y axes are in log-scale. The tubes are tracking errors between mean plus and minus one standard deviation over 100 trajectories (C3M: Proposed approach in Theorem 6).

2) *Target Trajectory Generation*: The target control inputs  $u_d(t)$  are generated as the linear combination of the elements in a set of sinusoidal signals with some fixed frequencies and randomly sampled a weight for each frequency component. The initial states of the target trajectories  $x_d(0)$  are uniformly randomly sampled from a compact set, and then  $x_d$  are obtained by integrating (3). The initial errors  $e(0)$  are uniformly randomly sampled from a compact set, and the initial states  $x(0)$  are computed as  $x(0) = x_d(0) + e(0)$ .

3) *Simulation Results and Discussions [11]*: Figure 2 shows the normalized tracking error defined as  $x_e(t) = \|x(t) - x_d(t)\| / \|x(0) - x_d(0)\|$  for each model and method. For PVTOL and the quadrotor models, both SoS and our learning-based method find a contracting tracking controller that renders  $x_e$  decrease rapidly, successfully achieving exponential stabilization. While the SoS of [21] cannot solve for the Neural lander and SEGWAY examples due to their complex and non-polynomial dynamics, and MPC produces large numerical errors for the SEGWAY example due to the sensitivity of the SEGWAY model, the proposed control of Theorem 6 yields consistently higher tracking performance for both of these models. The performance of the RL method varies with tasks: it achieves comparable results for PVTOL and the neural lander models but fails to find a contracting tracking controller within a reasonable time for the quadrotor and Segway models, as their state spaces are larger than the previous two cases.

Figure 2 thus implies that our proposed method outperforms both the learning and non-learning-based methods, producing a contracting controller with smaller tracking errors. These errors are guaranteed to be bounded exponentially with time even with the learning error and external disturbances, as rigorously proven in Theorems 1–6. Again, more simulation results can be found in [4], [6]–[8], [12].

## V. CONCLUSION

In this paper, we presented a theoretical overview of the NCM for provably stable, robust, and optimal learning-based control of non-autonomous nonlinear systems. It is formally shown that, even under the presence of learning errors and deterministic/stochastic disturbances, the NCM control ensures the distance between a target trajectory and learned system

trajectories to be bounded exponentially with time, enabling safe and robust use of AL and ML-based automatic control frameworks in real-world scenarios. The numerical simulation results of [11] are presented to validate the effectiveness of the NCM framework.

## REFERENCES

- [1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- [2] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Athena Scientific, 1996.
- [3] W. Lohmiller and J.-J. E. Slotine, “On contraction analysis for nonlinear systems,” *Automatica*, no. 6, pp. 683 – 696, 1998.
- [4] H. Tsukamoto, S.-J. Chung, and J.-J. E. Slotine, “Contraction theory for nonlinear stability analysis and learning-based control: A tutorial overview,” *Annu. Rev. Control, minor revision*, 2021.
- [5] H. Tsukamoto and S.-J. Chung, “Robust controller design for stochastic nonlinear systems via convex optimization,” *IEEE Trans. Autom. Control*, vol. 66, no. 10, pp. 4731–4746, 2021.
- [6] H. Tsukamoto and S.-J. Chung, “Neural contraction metrics for robust estimation and control: A convex optimization approach,” *IEEE Control Syst. Lett.*, vol. 5, no. 1, pp. 211–216, 2021.
- [7] H. Tsukamoto, S.-J. Chung, and J.-J. E. Slotine, “Neural stochastic contraction metrics for learning-based control and estimation,” *IEEE Control Syst. Lett.*, vol. 5, no. 5, pp. 1825–1830, 2021.
- [8] H. Tsukamoto, S.-J. Chung, and J.-J. Slotine, “Learning-based adaptive control via contraction theory,” in *IEEE CDC*, Dec. 2021.
- [9] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, ser. Studies in Applied Mathematics. Philadelphia, PA: SIAM, Jun. 1994, vol. 15.
- [10] I. R. Manchester and J.-J. E. Slotine, “Control contraction metrics: Convex and intrinsic criteria for nonlinear feedback design,” *IEEE Trans. Autom. Control*, vol. 62, no. 6, pp. 3046–3053, Jun. 2017.
- [11] D. Sun, S. Jha, and C. Fan, “Learning certified control using contraction metric,” in *CoRL*, Nov. 2020.
- [12] H. Tsukamoto and S.-J. Chung, “Learning-based robust motion planning with guaranteed stability: A contraction theory approach,” *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 6164–6171, 2021.
- [13] A. P. Dani, S.-J. Chung, and S. Hutchinson, “Observer design for stochastic nonlinear systems via contraction-based incremental stability,” *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 700–714, 2015.
- [14] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, “Understanding deep learning requires rethinking generalization,” in *Int. Conf. Learn. Represent.*, Toulon, France, Apr. 2017.
- [15] G. Shi *et al.*, “Neural lander: Stable drone landing control using learned dynamics,” in *IEEE ICRA*, May 2019.
- [16] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky, “Spectrally-normalized margin bounds for neural networks,” in *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.
- [17] H. J. Kushner, *Stochastic Stability and Control*. Academic Press New York, 1967.
- [18] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice-Hall, 2002.
- [19] B. T. Lopez and J. E. Slotine, “Adaptive nonlinear control with contraction metrics,” *IEEE Control Syst. Lett.*, vol. 5, no. 1, pp. 205–210, 2021.
- [20] B. T. Lopez and J.-J. E. Slotine, “Universal adaptive control of nonlinear systems,” arXiv:2012.15815, 2021.
- [21] S. Singh, A. Majumdar, J.-J. E. Slotine, and M. Pavone, “Robust online motion planning via contraction theory and convex optimization,” in *IEEE ICRA*, May 2017, pp. 5883–5890.
- [22] S. Singh, S. M. Richards, V. Sindhwani, J.-J. E. Slotine, and M. Pavone, “Learning stabilizable nonlinear dynamics with contraction-based regularization,” *Int. J. Robot. Res.*, Aug. 2020.
- [23] A. Taylor, A. Singletary, Y. Yue, and A. Ames, “Learning for safety-critical control with control barrier functions,” in *L4DC*, 2020, pp. 708–717.
- [24] B. Amos *et al.*, “Differentiable MPC for end-to-end planning and control,” in *Adv. Neural Inf. Process. Syst.*, 2018.
- [25] Y. Tassa, N. Mansard, and E. Todorov, “Control-limited differential dynamic programming,” in *IEEE ICRA*, 2014, pp. 1168–1175.
- [26] J. Schulman *et al.*, “Proximal policy optimization algorithms,” arXiv:1707.06347, 2017.