

# Continuous-time Dynamic Realization for Nonlinear Stabilization via Control Contraction Metrics

Ruigang Wang and Ian R. Manchester

**Abstract**—Nonlinear stabilization using control contraction metric (CCM) method usually involves an online optimization problem to compute a minimal geodesic (a shortest path) between pair of states, which is not desirable for real-time applications. This paper introduces a continuous-time dynamic realization which distributes the computational cost of the optimization problem over the time domain. The basic idea is to force the internal state of the dynamic controller to converge to a geodesic using covariant derivative information. A numerical example illustrates the proposed approach.

## I. INTRODUCTION

Stabilization of arbitrary trajectories of nonlinear dynamical systems is a challenging problem. One solution is to linearize the dynamics around the equilibrium manifold and apply the linear parameter-varying (LPV) control design methods [1]. However, these approaches generally lack global stability guarantees for the closed-loop nonlinear system. Another approach is to apply nonlinear model predictive control (NMPC) [2], which solves an optimal control problem (OCP) in a moving horizon way. Due to the complex dynamic constraints, the computational cost often limits its applications in real-time systems.

Contraction theory [3] is an attractive tool for the nonlinear stabilization problem because it provides formal global stability guarantees of the nonlinear system via simple local linear analysis. The underlying idea is to integrate the local stability results along a geodesic (a shortest path w.r.t. certain Riemannian metric) connecting the measured and reference states. Extensions to control design were developed in [4], [5] by introducing the concept of control contraction metric (CCM). Specifically, a CCM is a Riemannian metric for which the Riemannian energy functional of the geodesic between the measured and reference states can be made to decrease exponentially by choosing proper control action. Thus, the CCM can be understood as a differential version of control Lyapunov function (CLF). Further extensions to distributed control can be found in [6], [7]. Connections and comparisons with LPV based control was discussed in [8].

A static state-feedback realization based on integration along a geodesic was proposed in [4]. Implementation of this controller involves solving an optimization problem to find a geodesic. This online computation is similar to NMPC, but is of lower dimension without dynamic constraints. There exist some indirect methods for geodesic computation, such as phase flow method [9], fast marching [10] and

graph cuts [11]. One drawback of these approaches is the small convergence radii. Direct methods construct a finite-dimensional approximation of the online OCP and solve it via nonlinear programming (NLP). Typical discretization methods include single/multiple shooting [12] and global pseudospectral [13]. Recently, an efficient approach using the Chebyshev pseudospectral was proposed in [14]. Although the computational time is significantly reduced compared with the shooting method, online optimization is still not desirable for time-critical applications.

In this paper, we propose a continuous-time dynamic realization approach to address this issue. Inspired by a recent continuous-time MPC scheme [15], [16], the proposed approach makes continuous improvements to the integral path rather than solving a full optimization problem online. Specifically, the dynamic controller use forward flows generated by the plant model and gradient information of Riemannian energy functional to force its internal state (a path connecting the reference point to the measured state) to converge to a geodesic. The control output uses the same integration technique of [4] with integrals computed over the dynamic controller's internal state. We will consider state-feedback realization for both nominal and perturbed systems. It is shown that the nominal closed-loop system is globally exponentially stable and the path converges to a geodesic if the controller dynamics are sufficiently fast with respect to the plant dynamics. For the robust case where the system is perturbed by bounded additive disturbances, one endpoint of the path would deviate from the measured state, which may lead to closed-loop instability. Robust stability is achieved by adding state feedback to the path dynamics.

The structure of the paper is as follows. Section II gives some preliminaries results on CCM-based control design. In Section III we detail the proposed continuous-time dynamic realization. A numerical example is presented in Section IV to illustrate the effectiveness of this approach.

## II. PRELIMINARIES

### A. Notation

We use  $|x|$  to denote the standard Euclidean norm of a real vector  $x$ . The nonnegative reals are denoted  $\mathbb{R}^+ := [0, \infty)$ . The space  $\mathcal{L}_2^c$  is the set of vector signals  $f$  on  $\mathbb{R}^+$  whose causal truncation to any finite interval  $[0, T]$  has finite squared norm, i.e.  $\sqrt{\int_0^T |f(t)|^2 dt} < \infty$ . For symmetric matrices  $A$  and  $B$ , the notation  $A \prec B$  ( $A \preceq B$ ) means that  $B - A$  is positive (semi)definite.

This work was supported by the Australian Research Council.  
The authors are with the Australian Centre for Field Robotics, The University of Sydney, Sydney, NSW 2006, Australia (e-mail: ian.manchester@sydney.edu.au).

A Riemannian metric on  $\mathbb{R}^n$  is a symmetric positive-definite matrix function  $M(x)$ , smooth in  $x$ , which defines a smooth inner product  $\langle \delta_1, \delta_2 \rangle := \delta_1^\top M(x) \delta_2$  for any two tangent vectors  $\delta_1, \delta_2$  at the point  $x$ , and the norm  $\|\delta\|_M = \sqrt{\langle \delta, \delta \rangle}$ . A metric is called *uniformly bounded* if  $\alpha_1 I \preceq M(x) \preceq \alpha_2 I$ ,  $\forall x$ , for some constants  $\alpha_2 \geq \alpha_1 > 0$ .

Let  $\Gamma(x, y)$  be the set of smooth paths joining two points  $x$  and  $y$  in  $\mathbb{R}^n$ , where each  $c \in \Gamma(x, y)$  is a smooth map  $c : [0, 1] \rightarrow \mathbb{R}^n$  and satisfying  $c(0) = x$  and  $c(1) = y$ . We use the notation  $c(s)$ ,  $s \in [0, 1]$  and  $c_s := \frac{\partial c}{\partial s}$ . Given a metric  $M(x)$ , we can define the Riemannian length and energy functional of  $c$  as follows

$$L(c) := \int_0^1 \|c_s\|_M ds, \quad E(c) := \int_0^1 \|c_s\|_M^2 ds$$

respectively. The Riemannian distance  $d(x, y)$  between two points is the length of the shortest path between them, i.e.,  $d(x, y) := \inf_{c \in \Gamma(x, y)} L(c)$ . Under the conditions of the Hopf-Rinow theorem, there exists a *geodesic* (minimum-length curve)  $\gamma \in \Gamma(x, y)$  such that  $d(x, y) = L(\gamma)$ . Furthermore, we have  $E(\gamma) = L(\gamma)^2 = \langle \gamma_s, \gamma_s \rangle$ ,  $\forall s \in [0, 1]$ .

Let  $\Gamma(x, y, t)$  be the set of smooth time-varying paths  $c : \mathbb{R} \times [0, 1] \rightarrow \mathbb{R}^n$  connecting smooth signals  $x(t)$  and  $y(t)$ . We also use  $c(t) := c(t, \cdot)$  and  $\dot{c} := \frac{dc}{dt}$ . The formula for first variation of energy [17, p. 195] gives the time derivative of the energy functional  $E(t) := E(c(t))$  as follows

$$\frac{1}{2} \frac{dE}{dt} = \langle \dot{c}, c_s \rangle \Big|_{s=0}^{s=1} - \int_0^1 \langle \dot{c}, \nabla_{c_s} c_s \rangle ds \quad (1)$$

where  $\nabla$  is the Riemannian connection induced by the metric  $M(x)$ , and  $\nabla_{c_s} c_s$  is the covariant derivative. A smooth curve  $c$  is a geodesic if and only if  $\nabla_{c_s} c_s = 0$ .

### B. Control Contraction Metrics

Consider nonlinear control-affine systems of the form

$$\dot{x} = F(x, u) := f(x) + B(x)u \quad (2)$$

where  $x(t) \in \mathbb{R}^n$  and  $u(t) \in \mathbb{R}^m$  are state and control at time  $t \in \mathbb{R}^+ := [0, \infty)$ , respectively. For simplicity,  $f$  and  $B$  are assumed to be smooth and time-invariant. We denote the  $i$ th column of  $B(x)$  by  $b_i(x)$ . For the system (2) we define a *reference trajectory* to be any set of signals  $x^*, u^*$  all in  $\mathcal{L}_2^e$  and satisfying (2) on  $\mathbb{R}^+$ . A reference trajectory  $(x^*, u^*)$  is said to be globally exponentially stabilized by a feedback controller  $u = \kappa(x, x^*, u^*)$  if for any initial state  $x(0) \in \mathbb{R}^n$ , a unique closed-loop solution  $x(t)$  exists for all  $t \in \mathbb{R}^+$  and satisfies

$$\|x(t) - x^*(t)\| \leq R e^{-\lambda t} \|x(0) - x^*(0)\| \quad (3)$$

where  $R > 0$  is the overshoot, and  $\lambda > 0$  the rate. System (2) is said to be *universally exponentially stabilizable* if every forward-complete solution  $(x^*, u^*)$  is globally exponentially stabilizable. Note that universal stabilizability is a strong condition than global stabilizability of a particular solution.

Nonlinear stabilization using control contraction metric (CCM) ([4]) is a constructive approach to achieve universal stability. For the offline design stage, it applies linear system

theory to the control synthesis of the local linearized system – *differential dynamics*:

$$\dot{\delta}_x = A(x, u) \delta_x + B(x) \delta_u \quad (4)$$

where  $A = \frac{\partial f}{\partial x} + \sum_{i=1}^m \frac{\partial b_i}{\partial x} u_i$ . Specifically, we construct a differential feedback law:

$$\delta_u = K(x) \delta_x \quad (5)$$

where  $K = YW^{-1}$  with  $W(x) \in \mathbb{R}^{n \times n}$  and  $Y(x) \in \mathbb{R}^{m \times n}$  obtained from the following parameter-dependent linear matrix inequality (LMI):

$$-\dot{W} + AW + WA^\top - BY - Y^\top B^\top + 2\lambda W \preceq 0. \quad (6)$$

From the above inequality, the controller (5) achieves exponential stability for (4):

$$\frac{d}{dt} \|\delta_x\|_M^2 = \delta_x^\top \dot{M} \delta_x + 2\delta_x^\top M(A + BK) \delta_x \leq -2\lambda \|\delta_x\|_M^2 \quad (7)$$

where  $M(x) = W^{-1}(x)$  is called a CCM.

A static (memoryless) realization of the controller was proposed in [4], which includes three steps:

- 1) Compute a minimal geodesic

$$\gamma(t) := \underset{c \in \Gamma(x^*(t), x(t))}{\operatorname{argmin}} E(c). \quad (8)$$

- 2) Integrate (5) over  $\gamma$ , i.e.,

$$\kappa_\gamma(t, s) := u^*(t) + \int_0^s K(\gamma(t, \mathfrak{s})) \gamma_s(t, \mathfrak{s}) d\mathfrak{s}. \quad (9)$$

- 3) Implement the state-feedback control

$$u = \kappa(x, x^*, u^*) := \kappa_\gamma(t, 1). \quad (10)$$

This static realization achieves universally exponential stability with overshoot  $R = \sqrt{\frac{\alpha_2}{\alpha_1}}$  and rate  $\lambda$ . If there exists a smooth coordinate transformation  $z = h(x)$  satisfying  $\delta_z^\top \delta_z = \delta_x^\top M(x) \delta_x$ , we can compute the geodesics directly via  $\gamma(s) = h^{-1}(z^*(1-s) + z s)$  where  $z^* = h(x^*)$  and  $z = h(x)$ . However, for general cases, the computation of geodesics involves an optimization problem (8), which is not desired for time-critical applications.

### III. CONTINUOUS-TIME DYNAMIC REALIZATION

In this section, we introduce a continuous dynamic control realization which keeps Step 2) and 3) unchanged but replace Step 1) with a dynamical system whose internal state is a path joining  $x^*(t)$  to  $x(t)$ . This path dynamics solves a geodesic computation problem in parallel with the plant system. As shown in Fig. 1, we will consider two scenarios: nominal and robust state feedback. In particular, for the robust case, we assume that the system (2) is perturbed by bounded additive disturbances, i.e.,

$$\dot{x} = F(x, u) + d \quad (11)$$

with  $\|d(t)\| \leq \Delta$  for all  $t \in \mathbb{R}^+$ .

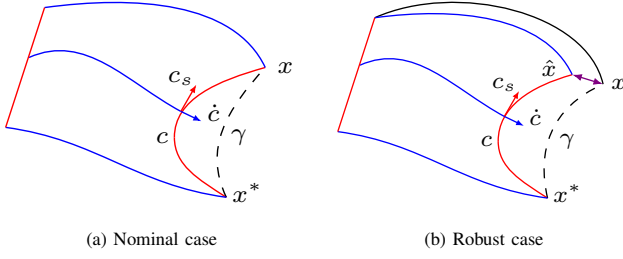


Fig. 1. Geometric illustrations of the continuous-time dynamic realization: red – path  $c(t, \cdot)$ , blue – flows  $c(\cdot, s)$ , dash – geodesic  $\gamma(t)$ .

### A. Nominal State Feedback

First, we consider the continuous-time dynamic realization via the forward flow defined by (2):

$$\begin{aligned}\dot{c} &= f(t, s) := F(c(t, s), \kappa_c(t, s)) \\ u &= \kappa_c(t, 1)\end{aligned}\quad (12)$$

where the initial state is  $c(0, s) = sx(0) + (1-s)x^*(0)$ . Then the endpoint dynamics can be represented by

$$\dot{c}(t, 0) = F(x^*, u^*), \quad (13a)$$

$$\dot{c}(t, 1) = f(x, u). \quad (13b)$$

It is easy to verify that  $c(t, 0) = x^*(t)$  and  $c(t, 1) = x(t)$  for all  $t \in \mathbb{R}^+$  since  $c(0, 0) = x^*(0)$  and  $c(0, 1) = x(0)$ . Moreover, integration of (7) over  $c(t)$  gives

$$\frac{1}{2}\dot{E} = \langle f(t, s), c_s \rangle \Big|_{s=0}^{s=1} - \int_0^1 \langle f(t, s), \nabla_{c_s} c_s \rangle ds \leq -\lambda E. \quad (14)$$

Globally exponential stability is achieved but perhaps with larger overshoot since  $c(t)$  generally does not converge to a geodesic  $\gamma(t)$ .

Now we consider an alternative path dynamics:

$$\dot{c} = f(t, s) + \alpha(s) \nabla_{c_s} c_s \quad (15)$$

where  $\alpha : [0, 1] \rightarrow \mathbb{R}^+$  is a smooth weighting function satisfying  $\alpha(0) = \alpha(1) = 0$ . Here the covariant derivative  $\nabla_{c_s} c_s$  can be taken as the gradient information of the geodesic optimization problem (8). We define the normalized weighting function as  $\eta(s) = \alpha(s)/\bar{\alpha}$  with  $\bar{\alpha} = \max_{s \in [0, 1]} \alpha(s)$ . Then, the nominal stability is given as follows.

**Theorem 1.** *For any weighting function  $\alpha(s)$ , the system (2) subject to the control law (15) is universally exponentially stable. If the parameter  $\bar{\alpha}$  is chosen to be sufficiently large, then the controller internal state  $c(t, \cdot)$  converges to a geodesic  $\gamma(t, \cdot) \in \Gamma(x^*, x, t)$  before  $x(t)$  converges to  $x^*(t)$ .*

*Proof.* From (14), we have

$$\frac{1}{2}\dot{E} \leq -\lambda E - \bar{\alpha} \int_0^1 \eta(s) \|\nabla_{c_s} c_s\|_M^2 ds \leq -\lambda E. \quad (16)$$

Universal stability of the nominal system (2) follows as the length of  $c(t)$  shrinks exponentially.

Choose a constant  $\tau \in (0, 1)$ , from Lemma 3 we can online adjust the weighting function  $\eta$  such that the following inequality holds:

$$\frac{1}{2}\dot{E} \leq -\lambda E - \bar{\alpha}\tau \int_0^1 \|\nabla_{c_s} c_s\|_M^2 ds. \quad (17)$$

With a sufficiently large  $\bar{\alpha}$ , the closed-loop system can be decomposed into two time-scale subsystems: slow dynamics (2) and fast dynamics (15). The covariant derivative  $\nabla_{c_s} c_s$  will be forced to converge to 0 (i.e., the path  $c(t, \cdot)$  converges to a geodesic  $\gamma(t, \cdot) \in \Gamma(x^*, x, t)$ ) before the convergence of the state  $x(t)$  to  $x^*(t)$ .  $\square$

*Remark 1.* As shown in Section III-C, the online implementation only computes a finite number of flows  $c(t, s_j)$ ,  $j = 0, 1, \dots, N$  digitally using forward-Euler or Runge-Kutta methods with a sufficiently small sampling time  $\tau_s$ . Thus,  $\bar{\alpha}$  cannot be chosen to be arbitrary large due to numerical stability consideration and the parameters  $(s_0, s_1)$  for the weighting function  $\eta(s)$  in (30) cannot be chosen to be arbitrary close to  $(0, 1)$ . Although the path  $c(t, \cdot)$  may not follow  $\gamma(t, \cdot)$  exactly, it can still converge to a small neighborhood of  $\gamma(t, \cdot)$ .

### B. Robust State Feedback

When system (2) is perturbed by external disturbances, the state trajectory  $x(\cdot)$  generally does not coincide with the endpoint trajectory  $c(\cdot, 1)$  generated by the path dynamics (15). Let  $\hat{x}(t) = c(t, 1)$  and  $\tilde{x}(t) = x(t) - \hat{x}(t)$ . To reduce the disturbance effect on  $\tilde{x}$ , we use the following path dynamics

$$\dot{c} = f(t, s) + \alpha(s) \nabla_{c_s} c_s + \beta(s) \tilde{x}(t) \quad (18)$$

where  $\beta(s) = \bar{\beta}\zeta(s)$  with  $\zeta : [0, 1] \rightarrow [0, 1]$  as a nondecreasing function satisfying  $\zeta(0) = 0$  and  $\zeta(1) = 1$ . Note that, for the nominal case, the above system is equivalent to the path dynamics (15).

If the disturbance bound  $\Delta$  is sufficiently small, the dynamics of  $\tilde{x}(\cdot)$  can be approximated by

$$\dot{\tilde{x}} = (A_{cl}(\hat{x}, \hat{u}) - \bar{\beta}I)\tilde{x} + d \quad (19)$$

where  $A_{cl}(\hat{x}, \hat{u}) = A(\hat{x}, \hat{u}) + B(\hat{x})K(\hat{x})$  with  $\hat{u} = \kappa_c(t, 1)$ . From (7) we have that the maximum eigenvalue of  $A_{cl}(\hat{x}, \hat{u})$  is no larger than  $-\lambda$ . Therefore, the error bound for  $\tilde{x}$  is

$$|\tilde{x}(t)| \leq \frac{\Delta}{\beta + \lambda}, \quad \forall t \in \mathbb{R}^+. \quad (20)$$

The time derivative of the energy functional satisfies

$$\begin{aligned}\frac{1}{2}\dot{E} &\leq -\lambda E + \langle \bar{\beta}\tilde{x}, c_s(t, 1) \rangle - \int_0^1 \langle \bar{\beta}\tilde{x}, \zeta \nabla_{c_s} c_s \rangle ds \\ &\quad - \bar{\alpha}\tau \int_0^1 \|\nabla_{c_s} c_s\|_M ds \\ &\leq -\lambda E + \langle \bar{\beta}\tilde{x}, c_s(t, 1) \rangle + \frac{\epsilon^2}{4} \|\bar{\beta}\tilde{x}\|_M^2 \\ &\quad - \frac{\bar{\alpha}\tau}{2} \int_0^1 \|\nabla_{c_s} c_s\|_M ds\end{aligned}\quad (21)$$

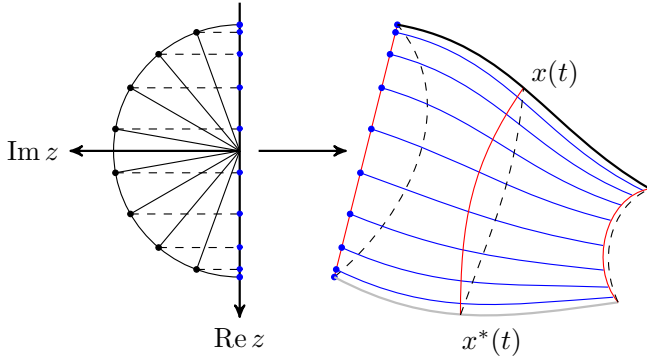


Fig. 2. Illustration of discretization of path  $c(t)$  using Chebyshev polynomials.

where  $\epsilon \geq \sqrt{2/\bar{\alpha}\tau}$ . The closed-loop robust stability is given as follows.

**Theorem 2.** Consider the perturbed system (11) and the continuous-time dynamic control realization (18). If the parameter  $\bar{\alpha}, \bar{\beta}$  are sufficiently large, the closed-loop system is robust stable with respect to the set

$$\Omega(x^*) = \{x \in \mathbb{R}^n : |x - x^*| \leq \bar{R}\Delta/\lambda\} \quad (22)$$

where  $\bar{R} = \frac{(1+\sqrt{1+\lambda\epsilon^2})\bar{\beta}}{2(\bar{\beta}+\lambda)}R + \frac{\lambda}{\bar{\beta}+\lambda}$ .

*Proof.* If  $\bar{\alpha}$  is sufficiently large, we can conclude from (21) that  $c(t)$  converges to  $\gamma(t)$  for  $t \geq T$  where  $T$  is sufficiently large. This leads to

$$\frac{1}{2}\dot{E} \leq -\lambda E + \langle \bar{\beta}\tilde{x}, \gamma_s(t, 1) \rangle + \frac{\epsilon}{4}\|\bar{\beta}\tilde{x}\|_M^2. \quad (23)$$

With the facts that  $E(\gamma) = \langle \gamma_s, \gamma_s \rangle, \forall s \in [0, 1]$  and  $|\tilde{x}| \leq \frac{\Delta}{\bar{\beta}+\lambda}$ , there exists a  $T' > T$  such that the following inequality holds for  $t \geq T'$ :

$$|\hat{x}(t) - x^*(t)| \leq \frac{(1 + \sqrt{1 + \lambda\epsilon^2})\bar{\beta}}{2(\bar{\beta} + \lambda)}R\Delta \quad (24)$$

which leads to

$$|x(t) - x^*(t)| \leq |\hat{x}(t) - x^*(t)| + |\tilde{x}| \leq \bar{R}\Delta/\lambda. \quad (25)$$

□

**Remark 2.** The parameter  $\bar{\alpha}$  represents the convergence speed of  $c(t)$  to a geodesic  $\gamma(t)$  connecting  $x^*(t)$  to  $\hat{x}(t)$  while the parameter  $\bar{\beta}$  controls the convergence speed of  $\hat{x}(t)$  to  $x(t)$ . The parameter  $\epsilon$  affects the size of invariant set. When  $\bar{\alpha}, \bar{\beta} \rightarrow \infty$  and  $\epsilon \rightarrow 0$ , we have  $\bar{R} \rightarrow R$  which implies that dynamic realization achieves the same invariant set as the geodesic based static realization (10).

### C. Implementation

The path dynamics is an infinite-dimensional system as its internal state  $c(t)$  is a smooth function over  $[0, 1]$ . For online implementation, we approximate  $c(t)$  with Chebyshev polynomial expansion at time  $t$ . This is a finite-dimensional approximation based on the samples of the path at Chebyshev nodes. In this way, the path dynamics is discretized into

a finite set of dynamical systems whose state dimension is same as the original nonlinear plant. Those systems are solved in parallel with the nonlinear plant and the solutions are used to construct an approximate path at the next time step. As time involves, this path converges to a geodesic due to the forward and gradient descent flows. This approach is different from [14] which uses Chebyshev polynomials to discretize the geodesic computation problem (8) at each time point. A finite-dimensional NLP is iteratively solved online, and the optimal solution is then used to construct a geodesic. Thus, the online computation time of the proposed approach is expected to be much smaller, compared with the optimization based approach [14].

Firstly, we recall some standard results of approximation theory using Chebyshev polynomials (see [18] for details). The first-kind Chebyshev polynomials  $T_k(x)$  over the interval  $[-1, 1]$  are defined recursively by

$$T_{k+1}(x) = 2xT_k(x) - T_{k-1}(x), \quad k = 1, 2, 3, \dots \quad (26)$$

with starting values  $T_0(x) = 1$  and  $T_1(x) = x$ . Under the coordinate transform  $x = \cos(\theta)$ ,  $\theta \in [0, \pi]$ , we have  $T_k(\cos \theta) = \cos(k\theta)$ . A continuous function  $f(x)$  over the interval  $[-1, 1]$  can be approximated by

$$\tilde{f}(\cos \theta) = \frac{a_0}{2} + \sum_{k=1}^N a_k \cos(k\theta) \quad (27)$$

where the coefficients  $\{a_k\}_{0 \leq k \leq N}$  can be obtained by applying discrete cosine transform to the samples of  $f$  at the Chebyshev nodes  $x_j = \cos(j\pi/N)$ ,  $j = 0, 1, \dots, N$ . Other operations on  $f$  such as integration and differentiation can also be efficiently approximated by Chebyshev polynomials.

Since the CCM-based control design is invariant under coordinate transformations [4], we can reparameterize the path  $c(t, \cdot)$  from  $[0, 1]$  to  $[-1, 1]$ . For the online implementation of dynamic controllers (12), (15) and (22), instead of computing the infinite-dimensional state  $c(t, \cdot)$ , we only compute the flows at  $s_j = \cos(j\pi/N)$ ,  $j = 0, 1, \dots, N$ , as shown in Fig. 2. Based on the values of  $c(t, s_j)$ , the state  $c(t, \cdot)$  is reconstructed as  $c(t, s) = \mathbf{c}(t)T(s)$  where  $\mathbf{c}(t) \in \mathbb{R}^{n \times (N+1)}$  and  $T(s) = [T_0(s), T_1(s), \dots, T_N(s)]^T$ . With this, we can compute smooth representations of the derivative  $\frac{\partial c}{\partial s}$ , covariant derivative  $\nabla_{c_s} c_s$ , differential control  $\delta_u$  and its integration  $\kappa_c$ . By taking samples of these functions at the Chebyshev nodes, we can obtain the right hand side of the dynamic controllers (12), (15) and (22).

**Remark 3.** Note that the above computation only involves a series of simple online operations such as additions, multiplications, differentiation and integration over a smooth function  $c(t, \cdot)$ . Due to the absence of complex operations (e.g. solutions of optimization problems) the online computational time can be estimated *a priori*. This information can be used to choose a sufficiently small sampling time  $\tau_s$  such that the flows  $c(\cdot, s_j)$ ,  $j = 0, 1, \dots, N$  can be computed digitally by using forward-Euler or Runge-Kutta approximation methods.

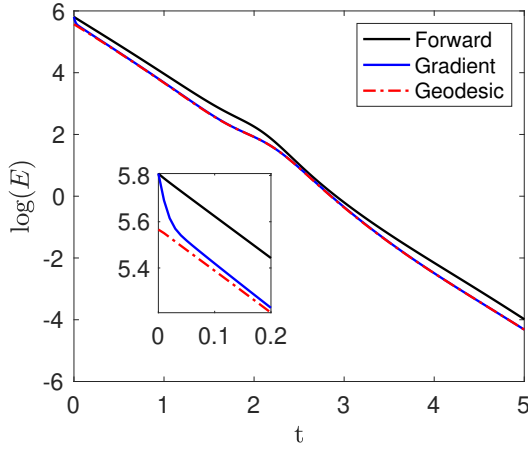


Fig. 3. Exponential decay rate of the Riemannian energy of integral paths.

#### IV. ILLUSTRATIVE EXAMPLE

We consider the following nonlinear system

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} -x_1 + x_3 \\ x_1^2 - x_2 - 2x_1x_2 + x_3 \\ -x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} u. \quad (28)$$

This system is not feedback linearizable and highly unstable. The control synthesis problem (6) was solved by SOS programming with LMI toolbox - Yalmip [19]. A control contraction metric with  $\lambda = 1$  was found to be

$$W(x) = W_0 + W_1x_1 + W_2x_1^2$$

where

$$W_0 = \begin{bmatrix} 2.686 & 0.237 & -1.816 \\ 0.237 & 16.265 & 2.006 \\ -1.816 & 2.006 & 6.395 \end{bmatrix}$$

$$W_1 = \begin{bmatrix} 0 & -5.373 & 0 \\ -5.373 & -0.948 & 3.631 \\ 0 & 3.631 & 0 \end{bmatrix}, W_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 10.747 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

and  $Y(x) = -\frac{1}{2}\rho(x)B^\top$  with  $\rho(x) = 19.614 + 1.386x_1 + 9.616x_1^2$ . From [4, Lemma 1], this metric is complete and thus a minimal geodesic exists for every pair of points. For online implementation, we use Chebyshev basis functions with maximal order  $N = 4$  to reconstruct the path  $c(t, \cdot)$ . The toolbox for Chebyshev polynomials manipulation is called *chebfun* [20], which is an open source software.

For the nominal case, we compare the results of three different realizations: forward flow based dynamic controller (12), gradient flow based dynamic controller (15) and geodesic based static controller (10). The initial condition and setpoint are chosen as  $x(0) = [9, 9, 9]^\top$  and  $x^* = [0, 0, 0]^\top$ , respectively. From Fig. 3, the Riemannian energy functional of the integral path  $c$  decays exponentially with rate of  $2\lambda$  for these three controllers. The proposed approach converges to a geodesic within time of 0.05 by feeding the covariant derivative to the path dynamics. Without this term, the path in forward flow based approach does not converge to a geodesic, which leads to a larger overshoot estimation

for exponential stability. Fig. 4 depicts the time evolution of integral paths  $c(t, \cdot)$  and geodesics  $\gamma(t)$  for different controllers. Compared with the forward flow approach, given the same initial condition (a straight line), the state  $c(t, \cdot)$  of the proposed approach converges to the neighborhood of a geodesic  $\gamma(t)$ .

For the robust case where the dynamics of  $x_1$  is perturbed by a persistent external disturbance  $d(t) = 2$ , we test those three controllers using the same initial state and setpoint. Fig. 5 shows that the forward flow approach is unstable due to the lack of feedback, although the state prediction  $\hat{x}(t)$  converges to the setpoint. For the proposed approach, the state prediction  $\hat{x}(t)$  remains in a neighborhood of  $x(t)$  due to the feedback term in (22). And the closed-loop system has a similar response compared to the geodesic based approach.

#### V. CONCLUSION

In this paper we proposed a continuous-time dynamic realization for control contraction metrics based nonlinear stabilization. It distributes the online geodesic computation across the time domain. Both universal stability for the nominal system and robust stability for the perturbed system are guaranteed. Simulation results demonstrated the effectiveness of the proposed approach.

#### APPENDIX

**Lemma 3.** For any  $c \in \Gamma(x^*, x)$  and any  $\tau \in (0, 1)$ , there exists a weighting function  $\eta : [0, 1] \rightarrow [0, 1]$  such that

$$\int_0^1 \eta(s) \|\nabla_{c_s} c_s\|_M^2 ds \geq \tau \int_0^1 \|\nabla_{c_s} c_s\|_M^2 ds. \quad (29)$$

*Proof.* We define  $\mu_c(s) := \int_0^s \|\nabla_{c_s} c_s\|_M^2 ds$ . Since  $M(x)$  is a uniformly bounded metric and  $c$  is a smooth curve, the covariant derivative  $\nabla_{c_s} c_s$  is smooth and bounded for any  $s \in [0, 1]$ . Thus,  $\mu_c$  is a nondecreasing function with  $\mu_c(0) = 0$  and  $\mu_c(1) = C < \infty$ . If  $c$  is a geodesic (i.e.,  $C = 0$ ), the weighting function  $\eta(s) = 0$  satisfies (29). Otherwise, for any  $\tau \in (0, 1)$ , we can find  $s_0 = \arg \inf \mu_c^{-1}((1 - \tau)C/2)$  and  $s_1 = \arg \sup \mu_c^{-1}((1 + \tau)C/2)$ . It is easy to check that  $0 < s_0 < s_1 < 1$  and  $\mu_c(s_1) - \mu_c(s_0) \geq \tau C$  since  $\mu_c$  is nondecreasing. Now we choose the weighting function to be

$$\eta(s) = \pi_0^{s_0}(s) [1 - \pi_1^{s_1}(s)] \quad (30)$$

where  $\pi_a^b : \mathbb{R} \rightarrow [0, 1]$  is a smooth and nondecreasing such that  $\pi_a^b(s) = 0, \forall s \leq a$  and  $\pi_a^b(s) = 1, \forall s \geq b$ . This weighting function satisfies (29) as

$$\int_0^1 \eta(s) \|\nabla_{c_s} c_s\|_M^2 ds \geq \int_{s_0}^{s_1} \|\nabla_{c_s} c_s\|_M^2 ds \geq \tau C.$$

□

#### REFERENCES

- [1] W. J. Rugh and J. S. Shamma, "Research on gain scheduling," *Automatica*, vol. 36, pp. 1401–1425, 2000.
- [2] F. Allgöwer and A. Zheng, *Nonlinear model predictive control*. Birkhäuser, 2012, vol. 26.
- [3] W. Lohmiller and J.-J. E. Slotine, "On contraction analysis for nonlinear systems," *Automatica*, vol. 34, pp. 683–696, 1998.

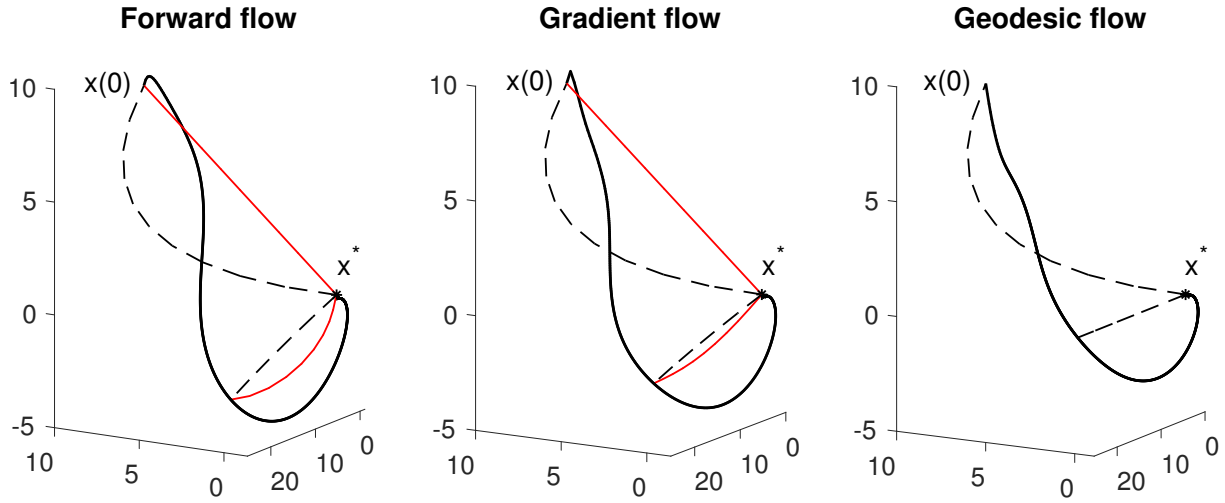


Fig. 4. Nominal state-feedback control: red – path  $c(t, \cdot)$  at two time points, black dash – geodesic  $\gamma(t, \cdot)$ , black solid – state trajectory  $x(t)$ .

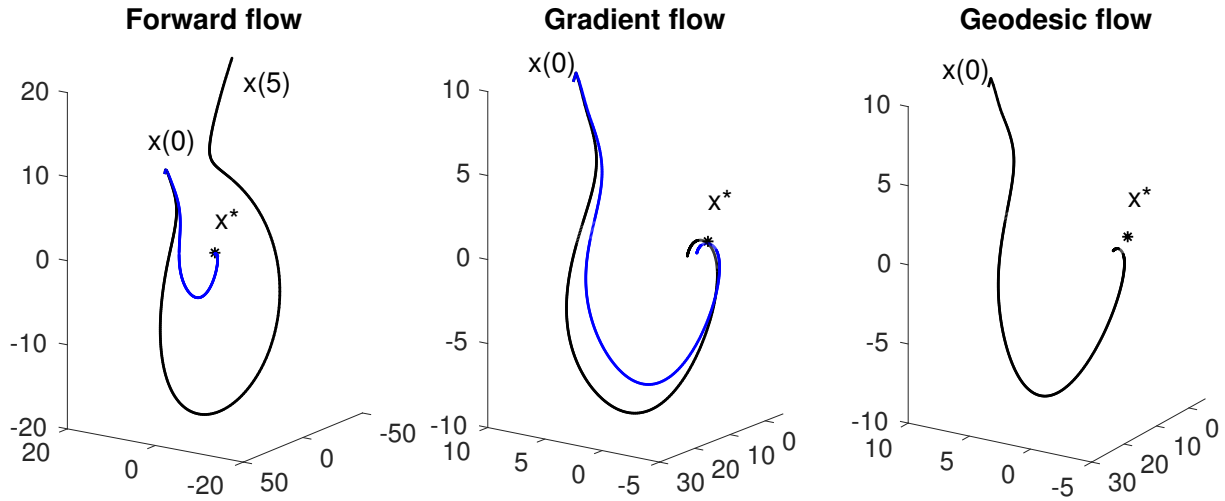


Fig. 5. Robust state-feedback control: blue – predicted state trajectory  $c(t, 1)$ , black solid – state trajectory  $x(t)$ .

- [4] 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.
- [5] —, “Robust control contraction metrics: A convex approach to nonlinear state-feedback  $H_\infty$  control,” *IEEE Control Syst. Lett.*, vol. 2, no. 3, pp. 333–338, Jul. 2018.
- [6] R. Wang, I. R. Manchester, and J. Bao, “Distributed economic MPC with separable control contraction metrics,” *IEEE Control Syst. Lett.*, vol. 1, pp. 104–109, 2017.
- [7] H. S. Shiromoto, M. Revay, and I. R. Manchester, “Distributed nonlinear control design using separable control contraction metrics,” *IEEE Trans. Control Netw. Syst.*, 2018.
- [8] R. Wang, R. Tóth, and I. R. Manchester, “A comparison of LPV gain scheduling and control contraction metrics for nonlinear control,” accepted by IFAC Workshop on Linear Parameter-Varying Systems, 2019.
- [9] L. Ying and E. J. Candes, “Fast geodesics computation with the phase flow method,” *J. Comput. Phys.*, vol. 220, no. 1, pp. 6–18, 2006.
- [10] R. Kimmel and J. A. Sethian, “Computing geodesic paths on manifolds,” *Proc. National Academy of Sciences*, vol. 95, no. 15, pp. 8431–8435, 1998.
- [11] Y. Boykov and V. Kolmogorov, “Computing geodesics and minimal surfaces via graph cuts,” in *Proc. IEEE Int. Conf. Comput. Vision*, 2003, pp. 26–33.
- [12] B. Houska, H. J. Ferreau, and M. Diehl, “An auto-generated real-time iteration algorithm for nonlinear MPC in the microsecond range,” *Automatica*, vol. 47, no. 10, pp. 2279–2285, 2011.
- [13] D. Garg, M. Patterson, W. W. Hager, A. V. Rao, D. A. Benson, and G. T. Huntington, “A unified framework for the numerical solution of optimal control problems using pseudospectral methods,” *Automatica*, vol. 46, no. 11, pp. 1843–1851, 2010.
- [14] K. Leung and I. R. Manchester, “Nonlinear stabilization via control contraction metrics: A pseudospectral approach for computing geodesics,” in *Proc. Amer. Control Conf. (ACC)*, Seattle, WA, 2017, pp. 1284–1289.
- [15] C. Feller and C. Ebenbauer, “Continuous-time linear MPC algorithms based on relaxed logarithmic barrier functions,” *IFAC Proc. Vol.*, vol. 47, no. 3, pp. 2481–2488, 2014.
- [16] M. M. Nicotra, D. Liao-McPherson, and I. V. Kolmanovsky, “Embedding constrained model predictive control in a continuous-time dynamic feedback,” *IEEE Trans. Autom. Control*, vol. 64, no. 5, pp. 1932–1946, 2018.
- [17] M. P. Do Carmo, *Riemannian geometry*. Boston, MA: Springer, 1992.
- [18] L. N. Trefethen, *Approximation Theory and Approximation Practice*. Philadelphia, PA: SIAM, 2013.
- [19] J. Lofberg, “YALMIP: A toolbox for modeling and optimization in MATLAB,” in *Proc. CACSD conf., New Orleans, LA, USA*, 2004, pp. 284–289.
- [20] T. A. Driscoll, N. Hale, and L. N. Trefethen, *Chebfun Guide*. Pafnuty Publications, 2014.