RESEARCH ARTICLE

WILEY

Virtual control contraction metrics: Convex nonlinear feedback design via behavioral embedding

Ruigang Wang[1] | **Roland Tóth[2,3]** | **Patrick J. W. Koelewijn[2]** | **Ian R. Manchester[1]**

[1]Australian Centre for Robotics, School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, Sydney, New South Wales, Australia

[2]Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

[3]Systems and Control Lab, HUN-REN Institute for Computer Science and Control, Budapest, Hungary

Correspondence
Ruigang Wang, Australian Centre for Robotics, School of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney, Sydney, NSW 2006, Australia.
Email: ruigang.wang@sydney.edu.au

Abstract

This article presents a systematic approach to nonlinear state-feedback control design that has three main advantages: (i) it ensures exponential stability and $\mathcal{L}_2$-gain performance with respect to a user-defined set of reference trajectories, (ii) it provides constructive conditions based on convex optimization and a path-integral-based control realization, and (iii) it is less restrictive than previous similar approaches. In the proposed approach, first a virtual representation of the nonlinear dynamics is constructed for which a behavioral (parameter-varying) embedding is generated. Then, by introducing a virtual control contraction metric, a convex control synthesis formulation is derived. Finally, a control realization with a virtual reference generator is computed, which is guaranteed to achieve exponential stability and $\mathcal{L}_2$-gain performance for all trajectories of the targeted reference behavior. We show that the proposed methodology is a unified generalization of the two distinct categories of linear-parameter-varying (LPV) state-feedback control approaches: global and local methods. Moreover, it provides rigorous stability and performance guarantees as a method for nonlinear tracking control, while such properties are not guaranteed for tracking control using standard LPV approaches. Code is available at https://github.com/ruigangwang7/VCCM.

KEYWORDS

contraction, linear-parameter varying systems, nonlinear systems, stabilization

1 | INTRODUCTION

For linear time-invariant (LTI) systems, essentially all reasonable definitions of stability coincide, and if a particular solution (such as the zero solution) is locally stable, then all solutions are globally stable. This favorable property extends to *stabilization*, and furthermore there are many well-established methods for computing stabilizing feedback controllers.

For nonlinear systems, however, the picture is more nuanced: distinct and practically-motivated notions of stability are not necessarily equivalent. For example, stability of a particular set-point (equilibrium) does not imply stability of all set-points, which in turn does not imply stability of all non-equilibrium trajectories. Furthermore, even in the full-state-feedback case, the computation of stabilizing feedback controllers is an on-going research topic.

In this article, we use the term *regulation* to denote stabilization of a particular set-point, usually the origin under suitable choice of coordinates. For the regulation problem, the concept of a *control Lyapunov function* (CLF)[1,2] plays an important role. Given a CLF, a simple, but universal construction yields stabilizing controllers.[3] For certain classes of nonlinear systems, CLFs can be constructed based on energy[4] or back-stepping techniques.[5] For more general systems, it is desirable to construct a CLF via optimization. For linear systems, the CLF criteria can be converted into a convex *linear matrix inequality* (LMI),[6] but for nonlinear systems, the set of CLFs for a particular system is not necessarily convex or even connected.[7] Certain dual notions of a CLF lead to convexity of synthesis,[7,8] but these can only imply almost-global stabilization, and are difficult to extend to disturbance rejection or robust stabilization.

In contrast, for the much stronger problem of *universal stabilization*—that is, global exponential stabilization of *all* trajectories—the concept of a *control contraction metric* (CCM) leads to convex conditions analogous to those for the LTI case,[9] and has been extended to disturbance rejection and robust stabilization.[10] This concept builds on contraction analysis,[11] extending it to address constructive control design. The main idea of contraction analysis is that local stability of *all* trajectories of a nonlinear system implies global stability of all trajectories. Hence, stability can be addressed via analysis of an infinite family of linear systems (the local linearizations along trajectories) and is decoupled from the specification of particular trajectories. To establish stability of an LTI system or to stabilize one, it is sufficient to search via semidefinite programming for a quadratic Lyapunov function or CLF.[6] In CCM synthesis, this is replaced by a smoothly state-varying quadratic form that measures infinitesimal distances, that is, a Riemannian metric. The resulting search is still convex, and is defined via state-dependent *point-wise* LMIs. Due to these advantageous features, contraction-based methods have recently also received attention in other parts of the nonlinear control field, such as adaptive control,[12,13] model predictive control[14–17] and learning-based control[18–21] and so forth.

Nevertheless, for some systems, the requirements for existence of a CCM may be too stringent. For example, in a *set-point tracking* problem, it may only be desired to globally stabilize a particular family of set-points: a problem that is stronger than regulation, but weaker than universal stabilization.[22,23] Motivated by such cases, in this article we propose methods for achieving global exponential stability of all trajectories from a user-specified reference set $\mathcal{B}^*$. We call this problem $\mathcal{B}^*$-*universal stabilization*, which covers a wide range of stabilization problems from regulation to tracking.

Linear parameter-varying (LPV) control is a widely-applied framework that also extends LTI design techniques to nonlinear systems.[24,25] There are two distinct categories of LPV state-feedback control approaches: local and global methods. The local LPV (a.k.a. LPV gain scheduling) approach is based on the widely-applied idea of gain scheduling.[26] It contains three stages[24]: linearize around a family of operating points (**modeling**), design controllers for those points (**synthesis**), and then interpolate in some way (**realization**). Although successfully applied in many settings, it is well known that "hidden couplings" between system dynamics and parameter variations can lead to closed-loop instability. Previous work to analyse this problem has led to an approach that is similar to contraction analysis, interpreting local linearization as the Gâteaux derivative of a nonlinear operator.[27]

The global LPV approach uses the same synthesis techniques as local methods, but different LPV modeling and realization methods. In this approach, the nonlinear system is modeled as an LPV system by choosing the scheduling variable as a function of state and input. This scheduling variable is treated as a "free" (external independent) variable throughout the synthesis step. However, it becomes in general[*] an internal variable for the control realization. The fact that stability must hold for all possible trajectories of the scheduling variable, including those incompatible with the true system dynamics, results in an inherent conservatism that can be seen as a trade-off for a convex synthesis procedure. Closed-loop stability and performance with respect to a particular set-point is guaranteed by the so-called *LPV (behavior) embedding principle*, that is, any solution of the nonlinear system is also a solution of the LPV system. However, this approach can fail to guarantee asymptotic convergence in set-point tracking.[28,29]

In this article, we propose a systematic approach to design nonlinear state-feedback controllers for $\mathcal{B}^*$-universal stabilization based on behavioral embedding. In this approach, we first generate behavior embeddings via the concept of a *virtual system*.[30,31] The main idea of virtual system is that a nonlinear system—which is not itself contracting—may still have stability properties that can be established via construction of an auxiliary observer-like system which *is* contracting. This contracting virtual system takes the original state as a "free" parameter, and thus naturally induces a nonlinear behavior embedding of the original system, that is, any solution of the original system is also a particular solution of the virtual system.

By extending this virtual contraction concept, a $\mathcal{B}^*$-universal stabilizing controller can be constructed if the virtual system satisfies two conditions: (**C1**) the virtual system is universally stabilizable, and (**C2**) any state reference from $\mathcal{B}^*$ is admissible to the virtual system. For the virtual feedback design, we provide constructive conditions based on parametric LMIs via the new concept of *virtual control contraction metrics* (VCCMs), which extends the CCM approach[9] to weak stabilization problems. Moreover, since part of the system nonlinearities can be hidden into the "free" parameter, the synthesis problem is simplified compared to the CCM approach. For Condition **C2**, we use several examples to show that it may lead to closed-loop instability for tracking tasks if such condition does not hold. However, this condition is quite stringent as it does not hold for many virtual systems. We present a relaxed condition by introducing the concept of *virtual reference generators* (VRGs), which produces a virtual reference that temporarily deviates from, but then exponentially converges to the original reference.

We also show that the proposed VCCM approach provides a unified framework for the two distinct categories of LPV control approaches: local and global methods. First, the virtual system can be taken as a generalization of the concept of *global LPV embedding* to *nonlinear embedding of the behavior* and the virtual differential dynamics can be seen as an extension of local linearization at operating points to continuous linearization across the entire space, that is, taking the first-order Taylor approximation at each state-input pair rather than those on the equilibrium manifold. Second, the VCCM approach provides a path-integral-based realization, which we argue is the correct realization of gain scheduling control that has been searched for in the past. We provide numerical examples in which local and global LPV approaches result in closed-loop instability at some set-points, whereas the VCCM approach guarantees global exponential stability of all set-points. We also discuss how to use our theoretic framework to explain the loss of stability guarantees for tracking control by standard LPV methods.

Contributions

This article presents a systematic behavioral embedding based nonlinear state-feedback approach for various stabilization problems—ranging from regulation to universal stabilization. Our work has the following contributions:

- We extend the CCM approach[9] to solve weak stabilization problems. Moreover, our proposed approach enables the embedding of the original dynamics into a virtual system with a simpler structure, which can reduce the complexity of contraction based control synthesis.
- We show that existing virtual contraction based control methods[32,33] require existence of a virtual feedforward controller that has restrictions on the achievable performance and even feasibility of stabilization of the system. We relax the underlying conditions by introducing VRGs and a novel feed-forward synthesis approach that ensures implementability of dynamic $\mathcal{B}^*$-universal stabilizing controllers, see Theorem 3.
- We show that the proposed VCCM approach is a unified theoretic framework for two distinct LPV state-feedback control methods. Our proposed approach can provide rigorous stability and performance guarantees, while such properties are not guaranteed for tracking control using standard LPV approaches.

Related works

Virtual or partial contraction analysis was first introduced in Reference 30 to extend contraction analysis to weaker stability notions. A more formal treatment appeared later in Reference 31. The first work to use virtual systems for constructive control design was Reference 32, however this was limited to Lagrangian systems. Subsequently, a virtual-contraction based approach to control of port-Hamiltonian systems was presented in Reference 33. In this article, a parametric-LMI based computational method is proposed to remove the structural assumption. For control realization, we introduce VRGs to relax the strong requirement on virtual feedforward control design.[32–34] Recently, we applied the proposed method to various stabilization tasks of a control moment gyroscope.[34] Compared to Reference 34, besides introducing VRGs and integrating it into the VCCM method, we provide a comprehensive description of the proposed VCCM approach, comparison with the CCM based approach, and an extensive discussion on the connection of the VCCM approach to local and global LPV methods.

Notation

$\mathbb{R}$ is the set of real numbers, while $\mathbb{R}_+$ is the set of non-negative reals. We use lower-case normal letters such as x to denote vector signals on $\mathbb{R}_+$. $\mathcal{L}_2$ is the space of square-integrable signals, that is, $x \in \mathcal{L}_2$ if $\|x\| := \sqrt{\int_0^\infty |x(t)|^2 dt} < \infty$ where $|\cdot|$ is the Euclidean norm. $\mathcal{L}_2^e$ is the extended $\mathcal{L}_2$ space where $\|x\|_T := \sqrt{\int_0^T |x(t)|^2 dt} < \infty$ for all $T \in \mathbb{R}_+$. $A > (\succeq) B$ means that the matrix $A - B$ is positive (semi)-definite. A Riemannian metric is a smooth matrix function $M : \mathbb{R}^m \to \mathbb{R}^{n \times n}$ with $M(x) = M(x)^\top > 0$ for all $x \in \mathbb{R}^m$. A Riemannian metric M is said to be uniformly-bounded if there exist $a_2 \geq a_1 > 0$ such that $a_2 I \succeq M(x) \succeq a_1 I$ for all $x \in \mathbb{R}^m$.

2 | PROBLEM FORMULATION AND PRELIMINARIES

2.1 | Problem formulation

Consider a nonlinear systems of the form

$$\dot{x}(t) = f(x(t), u(t)), \tag{1}$$

where $x(t) \in \mathbb{R}^n$, $u(t) \in \mathbb{R}^m$ are the state and control input, respectively, at time $t \in \mathbb{R}_+$, and f is a smooth function of their arguments. We define the *behavior* $\mathcal{B}$ as the set of all forward-complete solutions of (1), that is, $\mathcal{B} = \{(x, u)\}$ with $x : \mathbb{R}_+ \to \mathbb{R}^n$ piecewise differentiable and $u : \mathbb{R}_+ \to \mathbb{R}^m$ piecewise continuous satisfying (1) for all $t \in \mathbb{R}_+$. A state trajectory x is said to be admissible to system (1) if there exists an input signal u such that $(x, u) \in \mathcal{B}$. Let $\mathcal{B}^* \subset \mathcal{B}$ be a (feasible) set of desired behaviors. We consider state-feedback controllers that explicitly depend on $(x^*, u^*) \in \mathcal{B}^*$:

$$u(t) = \kappa(x(t), x^*(t), u^*(t)), \tag{2}$$

where $\kappa : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$. By applying (2) to system (1) we obtain the closed-loop system

$$\dot{x}(t) = f(x(t), \kappa(x(t), x^*(t), u^*(t))). \tag{3}$$

A reference trajectory (x^*, u^*) is said to be globally exponentially stabilizable if one can construct a feedback controller (2) such that for any initial condition $x(0) \in \mathbb{R}^n$, a unique solution x exists for (3) and satisfies

$$|x(t) - x^*(t)| \leq R e^{-\lambda t} |x(0) - x^*(0)|, \tag{4}$$

where rate $\lambda > 0$ and overshoot $R > 0$ are constants independent of initial conditions but may depend on the choice of reference trajectory. If every $(x^*, u^*) \in \mathcal{B}^*$ is globally exponentially stabilizable, then the system is said to be $\mathcal{B}^*$-*universally exponentially stabilizable*. If $\mathcal{B}_1^* \supset \mathcal{B}_2^*$, then $\mathcal{B}_1^*$-universal stabilization is stronger than $\mathcal{B}_2^*$-universal stabilization. The strongest case is $\mathcal{B}^* = \mathcal{B}$, which is called universal stabilization.[9]

We will also consider the disturbance rejection problem for the perturbed system of (1):

$$\dot{x}(t) = f(x(t), u(t), w(t)), \quad z(t) = h(x(t), u(t), w(t)), \tag{5}$$

where $w(t) \in \mathbb{R}^p, z(t) \in \mathbb{R}^q$ are collections of external disturbances (load, noise, etc.) and performance outputs (tracking error, actuator usage, etc.), respectively. With slight abuse of notation, we use $\mathcal{B}$ to denote the behavior of (5). The set $\mathcal{B}^* \subset \mathcal{B}$ is called a reference behavior if each $(x^*, u^*, w^*, z^*) \in \mathcal{B}^*$ satisfies $w^* = 0$ (i.e., the nominal value of disturbance is 0). Similar to the stabilization problem, we consider state-feedback controllers of the form (2), leading to the closed-loop system:

$$\dot{x}(t) = f(x(t), \kappa(x(t), x^*(t), u^*(t)), w(t)), \quad z(t) = h(x(t), \kappa(x(t), x^*(t), u^*(t)), w(t)). \tag{6}$$

The above controlled system is said to have $\mathcal{B}^*$-*universal $\mathcal{L}_2$-gain* bound of α if for each $(x^*, u^*, w^*, z^*) \in \mathcal{B}^*$, any initial condition $x(0) \in \mathbb{R}^n$ and any input $w \in \mathcal{L}_2^e$, a unique solution (x, w, z) of (6) exists and satisfies

$$\|z - z^*\|_T^2 \leq \alpha^2 \|w - w^*\|_T^2 + \beta(x(0), x^*(0)), \tag{7}$$

for all $T > 0$ and some continuous function $\beta(x, y) \geq 0$ with $\beta(x, x) = 0$.

The controller synthesis problems we address in this article are (i), synthesizing a state-feedback controller of the form (2) for a nonlinear system of the form (1) such that the resulting closed-loop system (given by (3)) is $\mathcal{B}^*$-universally exponentially stable, and (ii), synthesizing a state-feedback controller of the form (2) for a nonlinear system of the form (5) such that the resulting closed-loop system (given by (6)) is $\mathcal{B}^*$-universally exponentially stable and satisfies a prescribed $\mathcal{B}^*$-universal $\mathcal{L}_2$-gain bound.

In the next section, we will briefly discuss contraction and virtual contraction theory, which will be used to address these controller synthesis problems.

2.2 | Contraction analysis

Contraction analysis studies the convergence between arbitrary trajectories of the system, which has proven to be a useful tool for constructive design of nonlinear tracking controllers.[9,35] Here we summarize the main results of contraction theory, see details in References [11,36], and [37]. Consider a nonlinear time-varying system of the form:

$$\dot{x}(t) = f(t, x(t)), \tag{8}$$

where $x(t) \in \mathbb{R}^n$ is the state, and f is a smooth function. Later on, time variation will appear in the virtual systems associated with time-invariant systems (1) and (5). System (8) is said to be *contracting* if any solution pair (x_1, x_2) satisfies

$$|x_2(t) - x_1(t)| \leq Re^{-\lambda t}|x_2(0) - x_1(0)|,$$

where λ, R are some positive constants. To analyze the contraction property, we utilize the "extended" system consisting of (8) and its differential dynamics:

$$\dot{\delta}_x(t) = A(t, x)\delta_x(t) := \frac{\partial f(t, x)}{\partial x}\delta_x(t), \tag{9}$$

defined along the solutions x. The state $\delta_x(t)$ is the infinitesimal variation[11,37] (or can be seen as a tangent vector[36]) at the point $x(t)$. A uniformly-bounded Riemannian metric $M(x, t)$ is called a contraction metric for (8) if

$$\dot{M} + MA + A^\top M \preceq -2\lambda M, \tag{10}$$

for all $x \in \mathbb{R}^n, t \in \mathbb{R}_+$, where $\dot{M} = \frac{\partial M}{\partial t} + \sum_{i=1}^n f_i \frac{\partial M}{\partial x_i}$. The contraction metric $M(x, t)$ also induces a quadratic differential Lyapunov function $V(x, t, \delta_x) = \delta_x^\top M(x, t)\delta_x$ for (9), that is, $\dot{V} \leq -2\lambda V$. A central result of Reference [11] is that the existence of a contraction metric for system (1) implies that it is contracting with λ and $R = \sqrt{a_2/a_1}$.

Contraction analysis can be extended to the system with external input $w(t) \in \mathbb{R}^p$ and performance output $z(t) \in \mathbb{R}^q$:

$$\dot{x} = f(t, x, w), \quad z = h(t, x, w), \tag{11}$$

whose differential dynamics are of the form:

$$\dot{\delta}_x = A(t, x, w)\delta_x + B(t, x, w)\delta_w, \quad \delta_z = C(t, x, w)\delta_x + D(t, x, w)\delta_w, \tag{12}$$

where $A = \frac{\partial f}{\partial x}$, $B = \frac{\partial f}{\partial w}$, $C = \frac{\partial h}{\partial x}$, and $D = \frac{\partial h}{\partial w}$. System (11) is said to have a *differential $\mathcal{L}_2$-gain bound* of α, if for all $T > 0$,

$$\|\delta_z\|_T^2 \leq \alpha^2 \|\delta_w\|_T^2 + b(x(0), \delta_x(0)), \tag{13}$$

where $b(x, \delta_x) \geq 0$ with $b(x, 0) = 0$ for all x. From Reference 38 (Theorem 3.1.11), a sufficient, and in some cases necessary, condition for (13) is the existence of a differential storage function $V(x, t, \delta_x) \geq 0$ with $V(x, t, 0) = 0$ that verifies

$$V_{t_2} - V_{t_1} \leq \int_{t_1}^{t_2} \left(-\delta_z^\top \delta_z + \alpha^2 \delta_w^\top \delta_w \right) dt, \tag{14}$$

where $V_t = V(x(t), t, \delta_x(t))$. For smooth systems, the differential $\mathcal{L}_2$-gain bound is equivalent to the incremental $\mathcal{L}_2$-gain bound.[27]

2.3 | Behavior embedding and virtual contraction analysis

Virtual contraction analysis extends contraction theory to study convergence between behaviors (i.e., trajectory sets). For the sake of simplicity, let us first consider a time-invariant smooth autonomous system

$$\dot{x} = f(x), \tag{15}$$

with $x \in \mathbb{R}^n$. A *virtual system* is a new system of the form

$$\dot{y} = \mathfrak{f}(y, x), \tag{16}$$

with $\mathfrak{f}(x, x) = f(x)$ and $\mathfrak{f}$ being smooth, where the virtual state y lives in a copy of the original state space $\mathbb{R}^n$, and the variable x, taken as an exogenous input, is the state associated with the original system (15). One way to construct a virtual system is based on the factorization of the dependency of f on x, for example, $f(x) = x^2$ can be re-cast as $\mathfrak{f}(y, x) = xy$, or y^2. Hence, construction of (16) is non-unique. A virtual system introduces an *embedding* relationship between the behavior $\mathcal{B}$ of (15) and the virtual behavior $\mathcal{V}$ of (16). That is, $x \in \mathcal{B}$ is equivalent to $(x, x) \in \mathcal{V}$. For any $x \in \mathcal{B}$, we can define a *projected virtual behavior* $\mathcal{V}_x = \{y | (y, x) \in \mathcal{V}\}$, which is the state trajectory set of (16) with fixed external signal x.

Associated with $\mathcal{V}_x$, the virtual differential dynamics is

$$\dot{\delta}_y = \frac{\partial \mathfrak{f}(y, x)}{\partial y} \delta_y. \tag{17}$$

Note that $\delta_x(t) = 0$ for all t since x is a fixed external signal for all $y \in \mathcal{V}_x$. System (15) is said to be *virtually contracting* if there exists a metric $M(y, x)$ such that for any $x \in \mathcal{B}$, $\hat{M}(y, t) = M(y, x(t))$ is a contraction metric for the time-varying system (16). Here, $M(y, x)$ is also referred to as a virtual contraction metric. The existence of a virtual contraction metric implies that all virtual trajectories in $\mathcal{V}_x$ converge to each other, and thus they converge to a particular solution x as $x \in \mathcal{V}_x$. Furthermore, if $\mathcal{V}_x$ also contains $x^* \in \mathcal{B}^*$, we can conclude that $|x(t) - x^*(t)|$ vanishes exponentially.

Theorem 1 (Virtual contraction[31]). *Consider a system (15) and suppose that the virtual system (16) is virtually contracting. Given a trajectory $x \in \mathcal{B}$ of (15) and a reference trajectory $x^* \in \mathcal{B}^*$, suppose that $x^* \in \mathcal{V}_x$. Then x converges to $x^* \in \mathcal{B}^*$ exponentially.*

3 | NONLINEAR STABILIZATION VIA VCCM

In this section, we first present the behavioral embedding based control framework for the $\mathcal{B}^*$-universal stabilization problem. In this framework there are two main components: virtual feedforward and feedback controllers. We give a convex formulation for designing the virtual feedback controller by introducing the new concept of VCCMs. After that, we introduce the concept of virtual reference generator, which can be understood as an extension of the virtual feedforward controller. Finally, we discuss the differences between the proposed approach and the CCM-based control approach. For the sake of simplicity, the dependency of time t is dropped whenever it can be clearly inferred from the context.

3.1 | Behavior embedding based control framework

A virtual system of (1) can be represented by

$$\dot{y} = \mathfrak{f}(y, x, v), \tag{18}$$

with $\mathfrak{f}(x, x, u) = f(x, u)$, where $y(t) \in \mathbb{R}^n$, $v(t) \in \mathbb{R}^m$ are the virtual state and input, respectively, and the variable x, taken as an exogenous variable, is the state of the original system (1).

Remark 1. If $\mathfrak{f}$ is chosen to be linear in y and v, the virtual system (18) becomes a conventional global LPV embedding.[39] By allowing for more general behavior embeddings (e.g., nonlinear parameter-varying embedding), we can further reduce the conservatism resulting from the embedding while still providing a similar convex control formulations.

To obtain a $\mathcal{B}^*$-universal stabilizing controller, we construct virtual system (18) satisfying the following two conditions:

C1 (virtual feedback) For any admissible state trajectory x of $\mathcal{B}$, the projected virtual behavior $\mathcal{V}_x := \{(y, v) | (y, x, v) \in \mathcal{V}\}$ is universally exponentially stabilizable, that is, for any virtual reference $(y^*, v^*) \in \mathcal{V}_x$ there exists a *virtual feedback controller* of the form

$$v = \kappa^{\mathrm{fb}}(y, x, y^*, v^*), \tag{19}$$

where $\kappa^{\mathrm{fb}} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ is a locally Lipschitz continuous map, such that (18) is exponentially stabilized by (19) at (y^*, v^*).

C2 (virtual feedforward) For any admissible trajectory x of $\mathcal{B}$ and any reference trajectory $(x^*, u^*) \in \mathcal{B}^* \subseteq \mathcal{B}$, there exists a *virtual feedforward controller* of the form

$$v^* = \kappa^{\mathrm{ff}}(x, x^*, u^*), \tag{20}$$

where $\kappa^{\mathrm{ff}} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ is a locally Lipschitz continuous map, such that $u^* = \kappa^{\mathrm{ff}}(x^*, x^*, u^*)$ and $(x^*, v^*) \in \mathcal{V}_x$ hold, that is, any state reference from $\mathcal{B}^*$ is admissible to the virtual system.

By substituting (20) into (19) and setting $y^* = x^*$, system (18) is globally exponentially stabilized at the reference x^* via the following controller:

$$v = \overline{\kappa}(y, x, x^*, u^*) := \kappa^{\mathrm{fb}}(y, x, x^*, \kappa^{\mathrm{ff}}(x, x^*, u^*)). \tag{21}$$

By setting $y = x$ and $v = u$, we obtain an overall controller for system (1):

$$u = \kappa(x, x^*, u^*) := \overline{\kappa}(x, x, x^*, u^*). \tag{22}$$

Closed-loop stability with κ is guaranteed by the following theorem.

Theorem 2. *Consider the system (1) and a reference behavior $\mathcal{B}^*$. Assume that there exists a virtual system representation (18) such that Conditions **C1** and **C2** hold. Then, for any κ^{fb} and κ^{ff} that satisfy **C1** and **C2**, the closed-loop system (1) under (22) is $\mathcal{B}^*$-universally exponentially stable.*

Proof. From (18) and (21), we can obtain a virtual closed-loop system of the form

$$\dot{y} = \mathfrak{f}(y, x, \overline{\kappa}(y, x, x^*, u^*)). \tag{23}$$

Since $\overline{\kappa}$ is locally Lipschitz continuous due to the definition of κ^{fb} and κ^{ff}, then (23) is forward complete by Angeli and Sontag[40(Theorem 2)]. We use $\mathcal{V}^\kappa$ to denote the behavior of (23), that is, $\mathcal{V}^\kappa = \{(y, x)\}$ where (y, x) are

the feed-forward solution of (23). Given a trajectory x, we can obtain the projected behavior of $\mathcal{V}^\kappa$ as $\mathcal{V}^\kappa_x = \{y | (y, x) \in \mathcal{V}^\kappa\}$. For any $x^* \in \mathcal{B}^*$, Condition **C2** implies that $x^* \in \mathcal{V}^\kappa_x$ for all x. Furthermore, by Condition **C1** we have that all $y \in \mathcal{V}^\kappa_x$ converge to x^* exponentially.

Let $\mathcal{B}^\kappa$ be the closed-loop behavior of the original dynamics (1) and the controller (22). Since the choice of virtual system (18) satisfies $\mathfrak{f}(x, x, u) = f(x, u)$, then (23) is a virtual system of (3). For any $x \in \mathcal{B}^\kappa$, we have $x \in \mathcal{V}^\kappa_x$ by the behavior embedding principle, which further implies that x converges to $x^* \in \mathcal{B}^*$ exponentially. ∎

Example 1. Consider a system with state $x = [x_1 \ x_2]^\top$ and nonlinear dynamics

$$\dot{x} = A(x)x + Bu := \begin{bmatrix} -1 & 0 \\ -\sin x_2 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u. \tag{24}$$

This system is not universally stabilizable as any trajectory starting in the plane $x_2 = 0$ will remain in this plane. However, it can be $\mathcal{B}^*$-universally stabilized if all reference trajectories in $\mathcal{B}^*$ satisfy $x_2^*(t) = 0, \ \forall t \in \mathbb{R}_+$. First, we choose the virtual system

$$\dot{y} = A(x)y + Bv, \tag{25}$$

and then take the virtual feedback controller as

$$v = v^* + (y_2 - y_2^*) \sin x_2, \tag{26}$$

where (v^*, y^*) is the virtual reference trajectory, that is, $\dot{y}^* = A(x)y^* + Bv^*$. The closed-loop system of (25) and (26) becomes

$$\dot{\tilde{y}} = A_{\text{cl}}(x)\tilde{y} = \begin{bmatrix} -1 & \sin x_2 \\ -\sin x_2 & -1 \end{bmatrix} \tilde{y}, \tag{27}$$

where $\tilde{y} = y - y^*$. Since $A_{\text{cl}}(x) + A_{\text{cl}}^\top(x) = -2I$ for all $x \in \mathbb{R}^2$, the above system is globally exponentially stable, which further implies that the virtual feedback controller (26) satisfies Condition **C1**. Moreover, Condition **C2** also holds if we take the virtual feedforward controller as $y_2^* = x_2^* = 0$ and $v^* = u^*$. By the embedding principle, we obtain a $\mathcal{B}^*$-universal stabilizing controller $u = u^* + x_2 \sin x_2$ for the original nonlinear system (24).

3.2 | Virtual control contraction metrics

Here we present a constructive method for designing a virtual feedback controller satisfying Condition **C1** via VCCMs. It can be viewed as an extension of the CCM approach[9] to $\mathcal{B}^*$-universal stabilization.

For any fixed admissible state trajectory x of $\mathcal{B}$, the virtual system (18) becomes a time-varying nonlinear system. Its associated differential dynamics can be given by

$$\dot{\delta}_y = A(\sigma)\delta_y + B(\sigma)\delta_v, \tag{28}$$

where $\sigma = (y, x, v)$, $A = \frac{\partial \mathfrak{f}}{\partial y}$ and $B = \frac{\partial \mathfrak{f}}{\partial v}$.

Remark 2. Note that (28) fulfills the properties of an LPV system (see Reference 39) with σ as the scheduling-variable. In fact, if one takes the original nonlinear system (1) as a virtual system (18) (i.e., $\mathfrak{f} = f$), then (28) is a generalized local LPV embedding, see more details in Section 5.

Definition 1. A uniformly-bounded Riemannian metric $M(y, x)$ is said to be a virtual control contraction metric (VCCM) for (1), if there exists a matrix function $K(y, x) \in \mathbb{R}^{n \times m}$ satisfying

$$\dot{M} + M(A + BK) + (A + BK)^\top M + 2\lambda M \preceq 0 \tag{29}$$

for some constant $\lambda > 0$.

The following result is a direct application of Reference 9 (Theorem 1).

Proposition 1. *Given a virtual system (18) based embedding of (1), Condition **C1** holds if there exists a VCCM.*

3.2.1 | Control synthesis

The VCCM-based control synthesis problem can be formulated as the following conditions on a uniformly-bounded metric $W(y,x) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ and matrix function $Y(y,x) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{m \times n}$:

$$-\dot{W} + AW + WA^\top + BY + Y^\top B^\top + 2\lambda W \preceq 0, \tag{30}$$

which implies (29) with $M(y,x) = W^{-1}(y,x)$ and $K(y,x) = Y(y,x)W^{-1}(y,x)$. Moreover, (28) is exponentially stable under the differential controller

$$\delta_v = K(y,x)\delta_y. \tag{31}$$

The formulation in (30) is convex in W, Y, but infinite dimensional as the decision variables are sets of smooth matrix functions. There are various finite-dimensional LMI approximations to turn (30) into efficiently computable synthesis problems. One way is to apply an LPV synthesis technique, since the differential dynamics (28) vary with σ, which can be seen as the scheduling-variable. By computing a convex outer approximation (e.g., a convex polytope) of the possible signal variations $(\sigma, \dot{x})$ in $\mathcal{V}$, (30) can be transformed to a finite set of LMI constraints using LPV state-feedback synthesis techniques if A, B, and Y are rational matrix functions in σ, and W is rational matrix function in y, x (see Reference 25 for an overview). An alternative representation for A, B, Y is the gridding based approach, see Reference 41 for details. Another way is to approximate the components of (g, B, Y, W) by polynomials up to a chosen order, and verifying the inequalities by the sum-of-squares relaxation.[42]

3.2.2 | Control realization

We construct the virtual feedback realization by integrating the above differential controller (31) along a particular path, that is,

$$v(t) = \kappa^{\text{fb}}(y(t), x(t), y^*(t), v^*(t)) := v^*(t) + \int_0^1 K(\gamma(s), x(t))\gamma_s(s)ds, \tag{32}$$

with $\gamma_s := \frac{d\gamma}{ds}$, where the path $\gamma(s) : [0,1] \to \mathbb{R}^n$ is a (shortest path) geodesic γ connecting $y^*(t)$ to $y(t)$ w.r.t. the metric $M(y,x)$. Based on Reference 9, we can conclude that (32) is a virtual feedback controller satisfying Condition **C1**.

Note that if the metric M is independent of y, then γ is a straight line $\gamma(s) = sy(t) + (1-s)y^*(t)$, which allows to compute an analytic form of the controller through (32). For general y-dependent metric, the proposed control realization (32) requires solving an online optimization problem to compute the geodesic γ:

$$\min_{c:[0,1]\to\mathbb{R}^n} \int_0^1 c_s^\top M(c(s), x(t))c_s ds \quad \text{s.t.} \quad c(0) = y^*(t), \quad c(1) = y(t), \tag{33}$$

where $c_s := \frac{dc}{ds}$ and then using numerical integration to compute (32). In Reference 43, a pseudo-spectral approach was used to construct an approximate problem for the geodesic computation (33), which can reduce the computational load of the controller computation significantly while maintaining a good approximation quality. A recent work[44] developed a dynamical control realization method, which uses gradient flows to approximate the geodesics. By applying that approach to (32), no online optimization is required to compute (32) at the price of slightly-reduced convergence rate for the closed-loop system.

3.2.3 | Case study

We illustrate our approach on the pendubot system from Reference 45. The configuration, coordinates, and relative parameters of the pendubot are depicted in Figure 1. The pendubot dynamics is a nonlinear system of the form

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G(\theta) = R\tau, \tag{34}$$

where $\theta = [\theta_1, \theta_2]^\top$ is the angle configuration of two links and τ is the control input torque at the base joint. Here the coefficient matrices are

$$M = \begin{bmatrix} (m_1 + m_2)l_1^2 & m_2 l_1 l_2 \cos(\theta_1 - \theta_2) \\ m_2 l_1 l_2 \cos(\theta_1 - \theta_2) & m_2 l_2^2 \end{bmatrix}, \qquad C = \begin{bmatrix} f_{v1} & m_2 l_1 l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_2 \\ -m_2 l_1 l_2 \sin(\theta_1 - \theta_2)\dot{\theta}_1 & f_{v2} \end{bmatrix},$$

$$G = \begin{bmatrix} -(m_1/2 + m_2 + m_h)l_1 g \sin \theta_1 \\ -m_2 l_2 g \sin \theta_2 \end{bmatrix}, \quad R = \begin{bmatrix} 1 \\ 0 \end{bmatrix},$$

where f_{v1}, f_{v2} are the friction coefficients of two joints. We consider the stabilization problem of unstable set-points over a large range, that is, the reference $\theta^* \in := \{(\theta_1^*, \theta_2^*) \mid |\theta_1^*| \leq \pi/3, \theta_2^* = 0\}$. Our proposed approach takes the following steps.

1. *Behavior embedding via virtual system.* Here we construct three virtual systems via different treatment of nonlinearities in M, C, G:

$$M(\theta)\ddot{\phi} + C(\phi, \dot{\phi})\dot{\phi} + G(\phi) = Rv, \tag{35a}$$

$$M(\theta)\ddot{\phi} + C(\theta, \dot{\theta})\dot{\phi} + G(\phi) = Rv, \tag{35b}$$

$$M(\theta)\ddot{\phi} + C(\theta, \dot{\theta})\dot{\phi} + H(\theta)\phi = Rv, \tag{35c}$$

where ϕ, v are the virtual angle and input, respectively. The matrix $H(\theta) \in \mathbb{R}^{2 \times 2}$ is chosen such that $H(\theta)\theta = G(\theta)$ for all $\theta \in \mathbb{R}^2$. Note that such H always exists as G is continuously differentiable and $G(0) = 0$.[46] System (35a) only hides the nonlinear terms $M(\theta)\ddot{\theta}$ of (34) into the linear embedding $M(\theta)\ddot{\phi}$ while keeping the rest unchanged. System (35c) takes a full LPV embedding for all the nonlinear terms. System (35b) sits between (35a) and (35c). By introducing the virtual state $y = [\phi, \dot{\phi}]^\top$ and external parameter $x = [\theta, \dot{\theta}]^\top$, we rewrite (35a)–(35c) as virtual systems (18).

2. *Control synthesis.* For each virtual system representation, we apply the gridding method to solve (30) with $\lambda = 0.5$ to search for a constant $W \succ 0$ and a matrix function $Y(y, x)$ over the operating region $|\theta_1| \leq \pi/3, |\theta_2| \leq \pi/18$, $|\dot{\theta}_1| \leq 1, |\dot{\theta}_2| \leq 1$.

3. *Control realization.* As the VCCM $M = W^{-1}$ is a constant metric, the virtual feedback controller in (32) can be written as

$$v = \kappa^{\text{fb}}(y, x, y^*, v^*) := v^* + \left(\int_0^1 K(\gamma(s), x)\,ds \right)(y - y^*), \tag{36}$$
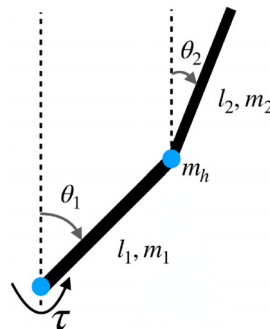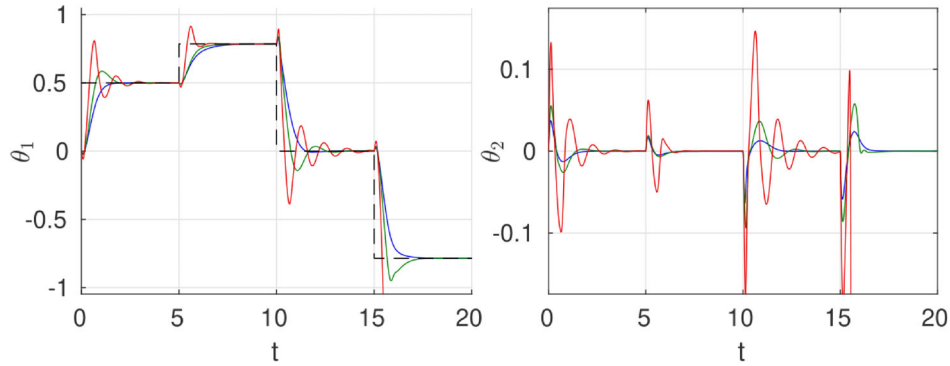

FIGURE 1 The pendubot system (see physical parameters in Reference 45).

FIGURE 2 State responses of (34) with VCCM controllers based on different virtual systems: blue—(35a), green—(35b), red—(35c), black dashed—reference trajectory.

with $K(y,x) = Y(y,x)W^{-1}$ is the differential control gain and $\gamma(s) = (1-s)y^* + sy$ is the geodesic connecting y^* to y. Note that the above virtual feedback controller satisfies Condition **C1** when the scheduling variable (y,x) is within the operating region. For virtual feedforward design, we choose $y^* = x^* := [\theta^*, 0]^\top$ and $v^* = \kappa^{\text{ff}}(x, x^*, u^*) := u^*$ where u^* is the nominal input torque satisfying $Ru^* = G(\theta^*)$. Finally, by applying the embedding rule $y(t) = x(t)$ and $u(t) = v(t)$ for all t, we obtain the overall controller for the (34) as

$$u = u^* + \left(\int_0^1 K(\gamma(s),x)ds \right)(x - x^*), \qquad (37)$$

where $\gamma(s) = (1-s)x^* + sx$.

Figure 2 depicts the closed-loop responses for control design based on virtual systems (35a)–(35c). The controllers from (35a) and (35b) can stabilize the system at different set points while unstable behavior is observed for the controller based on (35c) at the reference point $\theta_1^* = -\pi/4$. The explanation is as follows. First, we can easily verify that the chosen feedforward controller κ^{ff} satisfies Condition **C2** under the virtual systems (35a) and (35b). Thus, $\mathcal{B}^*$-universal stability follows by Theorem 2. Second, for the virtual system (35c), Condition **C2** requires $H(x)x^* = Ru^*, \forall x \in \mathbb{R}^2$, which only holds when $\theta_1^* = 0$. For this commonly used feedforward, a.k.a. input offset or trimming, in LPV gain scheduled control, Condition **C2** explains why the closed-loop response is stable for the large set-point jump from $\theta_1^* = \pi/4$ to $\theta_1^* = 0$, but unstable for the same amount of jump from $\theta_1^* = 0$ to $\theta_1^* = -\pi/4$. The instability is mainly due to the violation of Condition **C2** by the chosen κ^{ff}, showing that an inappropriate design of κ^{ff} can reduce the stability margin obtained from feedback control synthesis. For further analysis of this issue, see Section 5.3.

3.3 | Virtual reference generator

As shown in Section 3.2.3, if Condition **C2** is not satisfied then closed-loop stability can suffer. However, the more reference trajectories $\mathcal{B}^*$ contains, the more restrictive this condition becomes. Taking the virtual control-affine system $\dot{y} = g(y,x) + B(y,x)v$ as an example, Condition **C2** becomes

$$\dot{x}^*(t) - g(x^*(t), x(t)) \in \text{Span}\{B(x^*(t), x(t))\}, \quad \forall x^* \in \mathcal{B}^*, x \in \mathcal{B}, \qquad (38)$$

where $\text{Span}\{B\}$ denotes the column space of the matrix B. Such a condition often does not hold if the input dimension is smaller than the state dimension.

In this section, we relax Condition **C2** by introducing *virtual reference generators* (VRGs). Roughly speaking, instead of forcing $y^*(t) = x^*(t)$ for all $t \in \mathbb{R}_+$, a VRG allows to find a virtual reference $y^*(t)$ that can deviate from $x^*(t)$, but it represents an admissible virtual state trajectory that converges to $x^*(t)$, guiding $x(t)$ towards $x^*(t)$ as the virtual feedback controller (19) ensures that all $y(t)$ (including $x(t)$) converge to $y^*(t)$.

We create a copy of the virtual system (18) to govern the dynamics of the VRG:

$$\dot{y}^* = \mathfrak{f}(y^*, x, v^*), \tag{39}$$

with initial condition $y^*(0) = x^*(0)$, where the exogenous signal x is generated by system (1). Let $d := x - y^*$, then (39) can be rewritten as

$$\dot{y}^* = F(y^*, v^*, d) := \mathfrak{f}(y^*, y^* + d, v^*). \tag{40}$$

Here $d(t)$ is a known time-varying parameter. As one of our contributions, we replace Condition **C2** with the following weaker condition.

C2′: For any reference trajectory $(x^*, u^*) \in \mathcal{B}^*$, there exists a virtual reference controller

$$v^* = \kappa^{\text{ff}}(y^*, x, x^*, u^*), \tag{41}$$

with $u^* = \kappa^{\text{ff}}(x^*, x^*, x^*, u^*)$, where $\kappa^{\text{ff}} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^m$ is a locally Lipschitz continuous map, such that for any signal $d = x - y^*$ with $|d(t)| \leq Re^{-\lambda t}|d(0)|$, the closed-loop solution y^* of (40) exists and

$$|y^*(t) - x^*(t)| \leq R'e^{-\lambda t}|x(0) - y^*(0)|, \tag{42}$$

for some constant $R' > 0$.

Theorem 3. *Consider the system (1) and a reference behavior $\mathcal{B}^*$. Assume that there exists a virtual system representation (18) such that Condition **C1** and **C2′** hold. Then, for any κ^{fb} and κ^{ff} that satisfy **C1** and **C2′**, system (1) is $\mathcal{B}^*$-universally exponentially stable under the following controller*

$$\dot{y}^* = \mathfrak{f}(y^*, x, \kappa^{\text{ff}}(y^*, x, x^*, u^*)), \quad y^*(0) = x^*(0), \quad u = \kappa^{\text{fb}}(x, x, x^*, \kappa^{\text{ff}}(y^*, x, x^*, u^*)). \tag{43}$$

Proof. Condition **C1** implies $|x(t) - y^*(t)| \leq Re^{-\lambda t}|x(0) - x^*(0)|$ since $x(t)$ is also a feasible virtual state trajectory of (18). By Condition **C2′** we have $|y^*(t) - x^*(t)| \leq R'e^{-\lambda t}|x(0) - y^*(0)|$, which further leads to $|x(t) - x^*(t)| \leq |x(t) - y^*(t)| + |y^*(t) - x^*(t)| \leq (R + R')e^{-\lambda t}|x(0) - x^*(0)|$. ∎

Example 2. Consider the following nonlinear system

$$\dot{x} = f(x) + Bu := \begin{bmatrix} 2x_1 + x_2 \\ x_1 + x_1^2 - x_1^3 - x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u, \tag{44}$$

and the reference set $\mathcal{B}^* = \{(x^*, u^*) | x^*, u^*$ are constant signals such that $f(x^*) + Bu^* = 0\}$. We construct a control-affine virtual system $\dot{y} = g(y, x) + Bv$ with

$$g(y, x) = \begin{bmatrix} 2y_1 + y_2 \\ y_1 - y_2 + x_1^2 - x_1^3 \end{bmatrix}.$$

Simple incremental analysis yields that the virtual feedback controller $\kappa^{\text{fb}}(y, x, y^*, v^*) = v^* - K(y - y^*)$ with $K = [\, 5 \quad 1.5 \,]$ satisfies Condition **C1**. For the virtual feedforward design, we can verify that (38) does not hold for g since for any set-point x^*, there are infinitely many $x_1 \in \mathbb{R}$ such that $x_1^2 - x_1^{*2} + x_1^2 - x_1^3 \neq 0$. In fact, Condition **C2** cannot be satisfied for this choice of control-affine virtual embedding, as (38) does not hold for any control-affine virtual embedding of (44) that tries to hide all or part of the nonlinear term $x_1^2 - x_1^3$. Here we
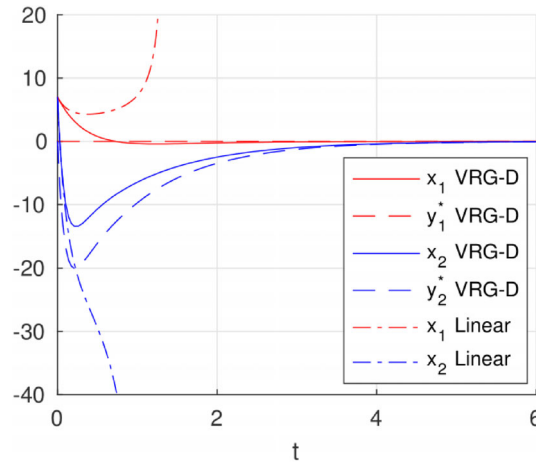
FIGURE 3   State responses of (44) with VRG-D controller (47) and linear controller (45) from initial state $x(0) = [\ 7\ \ 7\ ]^\top$. The virtual reference $y_2^*(t)$ first deviates from x_2^* due to the large initial value of $x_1(t)$ and then converges to x_2^* when $x_1(t)$ approaches to x_1^*. The state $x_2(t)$ converges to x_2^* as it follows $y_2^*(t)$.

choose the virtual feedforward input as $v^* = u^* = \arg\min_{v \in \mathbb{R}^m} |g(x^*, x) - Bv|^2$. The overall controller (22) is a linear law:

$$u = u^* - K(x - x^*). \tag{45}$$

Simulation results in Figure 3 show that this controller fails to stabilize the system for some large initial states. Now we consider the following VRG

$$y_1^* = x_1^*, \quad \dot{y}_2^* = y_1^* - y_2^* + x_1^2 - x_1^3, \quad v^* = u^* + x_2^* - y_2^*, \tag{46}$$

which satisfies Condition **C2′** as $y_2^*(t)$ converges to x_2^* if $x_1(t)$ approaches to y_1^*. Then, the overall controller (43) becomes

$$\dot{y}_2^* = x_1^* - x_2^* + x_1^2 - x_1^3, \quad y_2^*(0) = x_2^*, \quad u = u^* + x_2^* - y_2^* - K\left(x - \begin{bmatrix} x_1^* \\ y_2^* \end{bmatrix}\right). \tag{47}$$

The closed-loop stability follows by Theorem 3 and Figure 3 also shows stable response to a large initial state.

3.4 | Comparison with CCM-based control

A special case of the VCCM approach is when we take the original system itself as a virtual representation. In this case, both control synthesis and realization are the same as in the CCM approach.[9,10] With the proposed generalization based on the behavior embedding, our method can be applied to weak stabilization problems where the CCM approach fails to produce a feedback controller, see Example 1. Additionally, the complexity of the control design can be simplified since some of the system nonlinearities can be treated as external parameters at the synthesis stage.

Example 3. Consider an example taken from Reference 47 with state $x = [\ x_1\ \ x_2\ \ x_3\ ]^\top$ and nonlinear dynamics $\dot{x} = f(x) + Bu$ where

$$f(x) = \begin{bmatrix} -x_1 + x_3 \\ x_1^2 - x_2 - 2x_1x_3 + x_3 \\ -x_2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \tag{48}$$
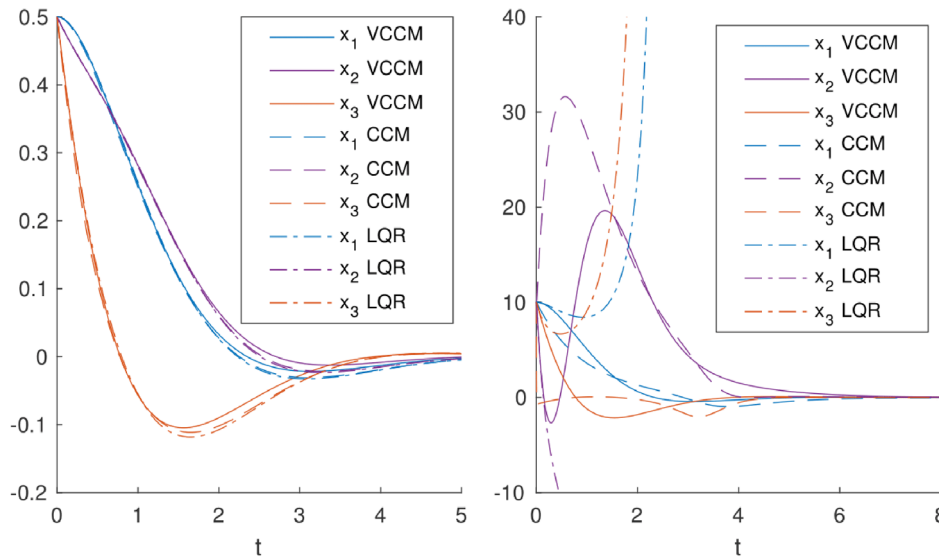
FIGURE 4 State responses of (48) with VCCM, CCM and LQR control from initial state $x(0) = [\ 0.5\ \ 0.5\ \ 0.5\ ]^\top$ (left) and $x(0) = [\ 10\ \ 10\ \ 10\ ]^\top$ (right).

The task is set-point tracking, that is, $\mathcal{B}^* = \{(x^*, u^*) | f(x^*) + Bu^* = 0\}$ with linear quadratic regulation (LQR) cost $\int_0^\infty (x^\top x + u^2)\, dt$. As pointed out in Reference 9, the above system does not allow for a uniform CCM. Then, a pair of state-dependent controller and metric have been computed by solving state-dependent LMIs. However, implementation of this CCM controller requires solving an online optimization problem at each time step due to the non-uniform metric.

For the proposed VCCM approach, we consider the virtual embedding $\dot{y} = g(y, x) + Bv$ with

$$
g(y, x) = \begin{bmatrix} -y_1 + y_3 \\ -y_2 + y_3 + x_1^2 - 2x_1 x_3 \\ -y_2 \end{bmatrix}.
$$

Since the associated virtual differential dynamics becomes an LTI system, we can obtain a constant metric and an LQR gain K based on the cost functional $\int_0^\infty \left(\delta_y^\top \delta_y + \delta_v^2\right) dt$. We choose the VRG with $y_1^* = x_1^*$, $\dot{y}_2^* = -y_2^* + y_3^* + x_1^2 - 2x_1 x_3$, $y_3^* = x_3^*$ and $v^* = y_2^*$, which satisfies Condition **C2′**. Then, our method gives a simple dynamic controller for (48):

$$
\dot{y}_2^* = -y_2^* + x_3^* + x_1^2 - 2x_1 x_3, \quad y_2^*(0) = x_2^*, \quad u = y_2^* + K(x - y^*). \tag{49}
$$

Figure 4 shows that for small initial conditions, the responses of VCCM, CCM, and LQR controllers are almost identical, where the LQR controller is constructed based on the local linearized model at the origin. In contrast, for larger initial conditions, the LQR controller was not stabilizing, while both the VCCM and CCM controllers were. On the other hand, the implementation of the VCCM controller (49) is much simpler than the CCM controller in Reference 9 as it does not require online optimization for computing geodesics.

4 │ DISTURBANCE REJECTION VIA ROBUST VCCM

Building on the results of the previous sections, we can extend the VCCM approach for general disturbance rejection. Given the system (5), we first construct a virtual system

$$
\dot{y} = \mathfrak{f}(y, x, v, w), \quad \zeta = \mathfrak{h}(y, x, v, w), \tag{50}
$$

with the property that $\mathfrak{f}(x, x, u, w) = f(x, u, w)$ and $\mathfrak{h}(x, x, u, w) = h(x, u, w)$, where $y(t) \in \mathbb{R}^n, v(t) \in \mathbb{R}^m, \zeta(t) \in \mathbb{R}^q$ are the virtual state, virtual control input and virtual performance output, respectively, and $x(t)$ is the real state of (5).

The associated differential dynamics of the projected virtual behavior $\mathcal{V}_x$ are given by

$$\dot{\delta}_y = A(\sigma)\delta_y + B(\sigma)\delta_v + B_w(\sigma)\delta_w, \quad \delta_\zeta = C(\sigma)\delta_y + D(\sigma)\delta_v + D_w(\sigma)\delta_w, \tag{51}$$

where $\sigma = (y, x, v, w)$, $A = \frac{\partial \mathfrak{f}}{\partial y}$, $B = \frac{\partial \mathfrak{f}}{\partial v}$, $B_w = \frac{\partial \mathfrak{f}}{\partial w}$, $C = \frac{\partial \mathfrak{h}}{\partial y}$, $D = \frac{\partial \mathfrak{h}}{\partial v}$ and $D_w = \frac{\partial \mathfrak{h}}{\partial w}$. Applying the differential state feedback (31) to (51) gives the closed-loop differential dynamics:

$$\dot{\delta}_y = (A + BK)\delta_y + B_w\delta_w, \quad \delta_\zeta = (C + DK)\delta_y + D_w\delta_w. \tag{52}$$

A *robust virtual control contraction metric* (RVCCM) is a uniformly bounded metric $M(y, x)$ used to establish an $\mathcal{L}_2$ gain bound for (52), that is,

$$\dot{V}(y, x, \delta_y) \leq -\delta_\zeta^\top \delta_\zeta + \alpha^2 \delta_w^\top \delta_w, \tag{53}$$

where $V(y, x, \delta_y) = \delta_y^\top M(y, x)\delta_y$ is called a virtual differential storage function. The following result gives a sufficient condition to search for the pair (K, M).

Proposition 2. *Suppose that there exist matrix functions* $Y(y, x) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^m \times \mathbb{R}^n$ *and* $W(y, x) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n \times \mathbb{R}^n$ *with* $c_2 I \succeq W(y, x) \succeq c_1 I$ *such that*

$$\begin{bmatrix} H & B_w & (CW + DY)^\top \\ B_w^\top & -\alpha I & D_w^\top \\ (CW + DY) & D_w & -\alpha I \end{bmatrix} \prec 0, \quad \forall x, y \in \mathbb{R}^n, \tag{54}$$

where $H = -\dot{W} + AW + WA^\top + BY + Y^\top B^\top$. *Then, the controlled system (52) with* $K = YW^{-1}$ *satisfies the* $\mathcal{L}_2$-*gain condition (53) with* $M = W^{-1}$.

Proof. The proof follows similarly as in Reference 10. ∎

The transformation from (53) to (54) is similar to the case of H_∞ state-feedback control for linear systems (e.g., Reference 48). Here (54) takes the form of a point-wise LMI in W and Y, that is, it is still convex, but infinite dimensional. Numerically tractable solutions are discussed in Section 3.2.

For realization of the controller in case of disturbance rejection, we consider a VRG of the form

$$\dot{y}^* = \mathfrak{f}(y^*, x, v^*, w^*), \quad \zeta^* = \mathfrak{h}(y^*, x, v^*, w^*), \tag{55}$$

with initial condition $y^*(0) = x^*(0)$, where the exogenous input x is generated by the original system (5) under the control law $u = \kappa^{\text{fb}}(x, x, y^*, v^*)$ where κ^{fb} is given in (32). From Reference 10 (Theorem 1) and the behavioral embedding principle, the real closed-loop system achieves an $\mathcal{L}_2$-gain bound of α from $w - w^*$ to $z - \zeta^*$. We can also obtain the $\mathcal{L}_2$-gain bound (denoted as α_{wy}) from δ_w to δ_y of (52) under the differential gain $K = YW^{-1}$. The performance bound from $w - w^*$ to $z - z^*$ is given as follows.

Theorem 4. *Consider the system (5) and a reference behavior $\mathcal{B}^*$. Assume that there exists a virtual embedding such that (54) is feasible and there exists a virtual feed-forward controller κ^{ff} achieving $\mathcal{L}_2$ gain of $\alpha_{y\zeta}$ from $x - x^*$ to $\zeta^* - z^*$ for the VRG (55). Then, the realization of the overall control law (43) achieves a $\mathcal{B}^*$-universal $\mathcal{L}_2$-gain bound of $\tilde{\alpha} = \sqrt{\alpha^2 + \alpha_{wy}^2 \alpha_{y\zeta}^2}$ for the system (5).*

Proof. From the above analysis, the upper bound of the $\mathcal{L}_2$ gain can be established by

$$\|z - z^*\|_T^2 \leq \|z - \zeta^*\|_T^2 + \|\zeta^* - z^*\|_T^2 \leq \alpha^2 \|w - w^*\|_T^2 + \alpha_{y\zeta}^2 \|x - x^*\|_T^2 \leq (\alpha^2 + \alpha_{wy}^2 \alpha_{y\zeta}^2)\|w - w^*\|_T^2.$$

∎

5 | A UNIFIED GENERALIZATION FOR LPV CONTROL

In this section, we first summarize the design procedure of the proposed VCCM approach. Then, by step-by-step comparisons, we show that the VCCM approach is a unified generalization for two distinct categories of LPV control methods, namely, global and local LPV state-feedback control. Moreover, through the lens of Theorem 2, we provide explanations to the question why both the local and global standard LPV control realizations do not provide rigorous global stability and performance guarantees for tracking control.

5.1 | Overview of VCCM-based control

We summarize the procedure of the VCCM-based control as follows (see also Figure 5):

- *Two-stage modeling*: We first choose a virtual system by treating some state-dependent terms as exogenous signals, resulting in a *nonlinear parameter-varying* (NPV) behavior embedding. Then, by taking continuous linearization w.r.t. the virtual state and input, we obtain a virtual differential dynamics, which can be treated as an LPV system.
- *Control synthesis*: We use the LMI-based control synthesis framework for LPV systems to design a linear differential state-feedback controller.
- *Control realization*: We construct a virtual feedback controller by integrating the differential controller over a particular path. This realization ensures that Condition **C1** holds. We also need to design a virtual feedforward controller based on the NPV embedding such that either Condition **C2** or **C2′** holds. Finally, the overall controller based on behavior embedding principle provides rigorous stability and performance guarantees.

5.2 | Local LPV control

We rephrase a classic local LPV control method—the equilibrium linearization approach[49] in the VCCM framework. For simplicity, we only consider control design for the nominal system (1).
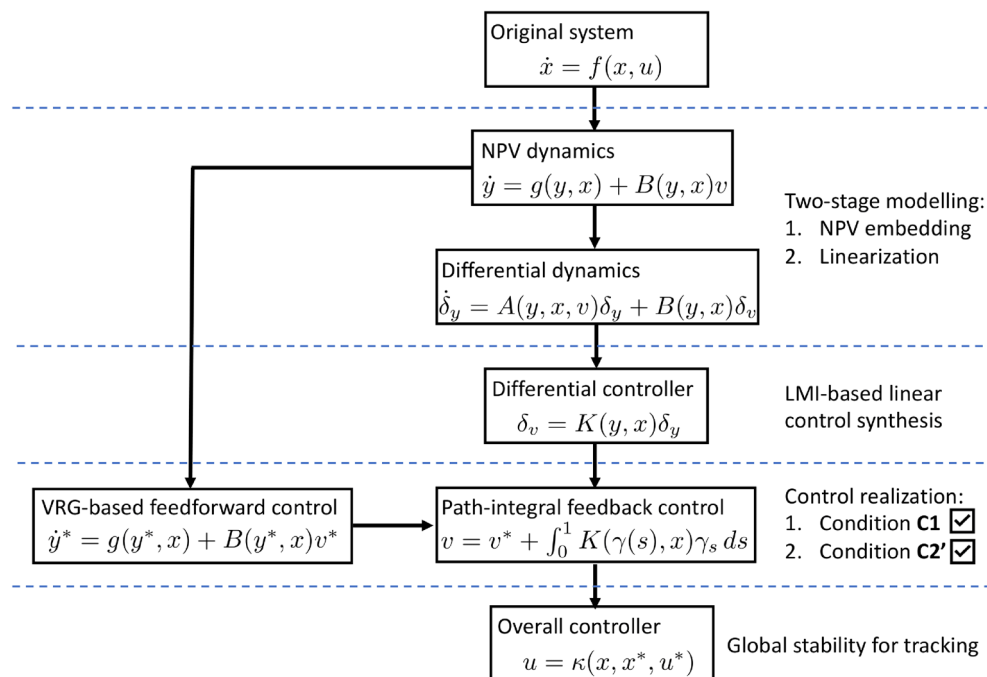

FIGURE 5   Control design procedure for the proposed VCCM approach.

5.2.1 | Two-stage modeling

The local LPV control method takes a *trivial* NPV embedding, that is, the virtual dynamics is a copy of the original nonlinear dynamics (1). Let $\{(x^*, u^*)(\sigma)\}_{\sigma \in \Omega}$ be a smooth equilibrium family of (1) where $\sigma \in \Omega \subset \mathbb{R}^{n_\sigma}$ is a scheduling variable depending on the internal state x, that is, $\sigma = \psi(x)$ is a smooth mapping. By performing Jacobian linearization of (1) around the equilibrium family, we obtain an LPV model as follows:

$$\frac{d}{dt}\delta x = A(\sigma)\delta x + B(\sigma)\delta u := \frac{\partial f}{\partial x}(x^*(\sigma), u^*(\sigma))\delta x + \frac{\partial f}{\partial u}(x^*(\sigma), u^*(\sigma))\delta u, \quad \sigma \in \Omega, \tag{56}$$

where $\delta x = x - x^*(\sigma)$, $\delta u = u - u^*(\sigma)$ are the *deviation variables*. The VCCM approach extends the modeling of local LPV control in two aspects. First, it allows for more general NPV embeddings, which can hide some "troublesome" nonlinear terms into the exogenous variables and result in a simple control synthesis problem. Second, linearization is extended from the equilibrium manifold to the entire state/input space (see Figure 6), which allows for tracking control design for time-varying references.
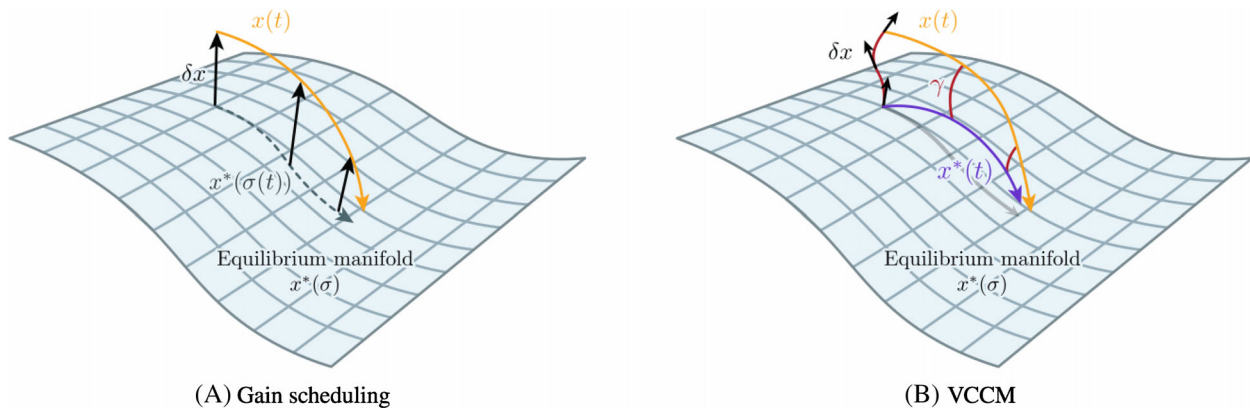
5.2.2 | Control synthesis

By solving the same control synthesis problem (30) as the VCCM approach, we obtain an LPV controller of the form

$$\delta u = K(\sigma)\delta x, \tag{57}$$

with a given parameterization of K (e.g., an affine function of σ). That is, there exists an Lyapunov function $V(\delta x) = \delta x^\top M(\sigma)\delta x$ with $M(\sigma) \succ 0$ such that

$$\frac{d}{dt}V(\delta x) = \delta x^\top \overbrace{\left[\frac{\partial M}{\partial \sigma}\dot{\sigma} + A_c(\sigma)^\top M(\sigma) + M(\sigma)A_c(\sigma)\right]}^{P(\sigma)}\delta x < 0, \tag{58}$$

for all $\delta x \neq 0$, where $A_c(\sigma) = A(\sigma) + B(\sigma)K(\sigma)$.

FIGURE 6 Geometric illustration of gain-scheduling and VCCM-based control realizations. For the gain-scheduling approach, the realization constrains the reference trajectories to the equilibrium manifold and takes the deviation $x(t) - x^*(t)$ as the variable δx, which can violate conditions **C1** and **C2** and result in a loss of stability. Our proposed VCCM approach takes a path integral along the geodesic γ and allows the references $x^*(t)$ temporally deviating from the equilibrium manifold. These two features make the resulting control realization satisfy the conditions **C1** and **C2** and hence guarantee stability.

5.2.3 | Control realization

The realization task for a local LPV controller (57) is to construct a gain-scheduled law $u = \kappa(x, \sigma)$ for (1) satisfying

$$u^*(\sigma) = \kappa(x^*(\sigma), \sigma), \tag{59a}$$

$$\frac{\partial \kappa}{\partial y}(y^*(\sigma), \sigma) = K(\sigma). \tag{59b}$$

Condition (59a) means that the reference point x^* is also an equilibrium of the closed-loop system. Condition (59b) states that linearization of $u = \kappa(x, \sigma)$ at this equilibrium is the local LPV controller (57), which is the key to guarantee closed-loop stability. However, such a realization generally does not exist when $K(\psi(x))$ is not completely-integrable. A typical realization for (57) used in the LPV literature is

$$u = u^*(\sigma) + K(\sigma)(x - x^*(\sigma)), \tag{60}$$

leading to a nonlinear controller of the form

$$u = u^*(\psi(x)) + K(\psi(x))(x - x^*(\psi(x))). \tag{61}$$

As pointed out in Reference 24, the closed-loop stability requirements of the above gain-scheduling controller suffer from two drawbacks: first, the state x needs to be sufficiently close to the equilibrium manifold $x^*(\sigma)$; second, the scheduling signal σ needs to be sufficiently "slowly varying." We now give explanations to the above issues by checking the conditions of Theorem 2.

First, we show that Condition **C1** may not hold for the feedback controller (61). Although σ is implicitly involved in (61) via equilibrium parameterization, linearization of (61) at the equilibrium manifold yields

$$\delta u = K(\sigma)\delta x + \overbrace{\left(\frac{\partial u^*(\sigma)}{\partial \sigma} - K(\sigma)\frac{\partial x^*(\sigma)}{\partial \sigma}\right)\frac{\partial \psi}{\partial x}(x^*(\sigma))}^{K_h(\sigma)}\delta x.$$

Compared with (57), the above equation contains a so-called *hidden coupling term* $K_h(\sigma)\delta x$. Then, the derivative of Lyapunov function in (58) becomes

$$\frac{d}{dt}V(\delta x) = \delta x^\top P(\sigma)\delta x + \delta x^\top \left[B(\sigma)K_h(\sigma)^\top M(\sigma) + M(\sigma)B(\sigma)K_h(\sigma)\right]\delta x, \tag{62}$$

which may not always be negative. Thus, Condition **C1** may be violated by the virtual feedback realization (61), regardless the fact that exponential stability is achieved in the local control synthesis step. This could lead to performance deterioration or even closed-loop instability.[24]

Then, we explain the necessity of sufficiently "slow-varying" scheduling signals for closed-loop stability through the lens of Condition **C2**. The feedforward component in (61) is $x^*(t) = x^*(\sigma(t))$ and $u^*(t) = u^*(\sigma(t))$. Substituting $(x^*, u^*)(\sigma(t))$ into (1) yields a residual term $\frac{\partial x^*(\sigma)}{\partial \sigma}\dot{\sigma}$, that is, Condition **C1** does not hold as $(x^*, u^*)(\sigma(t))$ is not an admissible trajectory to the virtual system (1). If $\dot{\sigma}$ is not sufficiently small, the residual term can drive the state away from the close neighborhood of $x^*(\sigma)$, where the local stability guarantees of the design hold.

Compared to gain-scheduling, the proposed VCCM control scheme achieves rigorous global stability and performance guarantees based on the following two facts. First, it does not contain any residual term as its reference trajectory is a feasible solution to the system dynamics. Second, thanks to continuous linearization and contraction theory, it does not suffer from hidden coupling effects as it only requires K to be path-integrable. By integrating the differential controller along geodesics, it achieves the designed stability and performance.[9] In Figure 6, we also show a illustration of the geometrical comparison between gain-scheduling and VCCM-based control realizations.

We argue that the proposed path-integral based realization is the correct realization of gain scheduling that has been searched for in the past. Considering a linear differential controller $\delta_u = K(x)\delta_x$, its path-integral realization is $v = v(1)$ where $v : [0,1] \to \mathbb{R}^m$ is a control path satisfying

$$v(s) := v^* + \int_0^s K(\gamma(\tau))\gamma_s(\tau)d\tau, \tag{63}$$

with $\gamma : [0,1] \to \mathbb{R}^n$ as a geodesic connecting y^* to y. Note that Condition (59a) holds as $v(1) = v(0) = u^*$ if $x = x^*$. Condition (59b) holds along the path γ, that is, $v_s = K(\gamma(s))\gamma_s(s)$ for all $s \in [0,1]$. We connect it to the gain-scheduling controller (60) by taking N sampled points of γ where N is sufficiently large. A simple numerical integration scheme for the path integral (63) is

$$v(s_{j+1}) \approx v(s_j) + K(\gamma(s_j))\big(\gamma(s_{j+1}) - \gamma(s_j)\big),$$

where $v(s_0) = v^*$. Based on this observation, the control path v integrates a series of gain-scheduling controllers (60) along the path γ, which corresponds to a "scheduled path" to reach reference trajectory, and the VCCM control realization is the corresponding control action to the state $x = \gamma(1)$ at one end-point of the path.

Here we give a comparison example of the gain-scheduling and path-integral realization schemes.

Example 4. Consider the following nonlinear system from Reference 26:

$$\dot{x}_1 = -x_1 - x_2 + r, \quad \dot{x}_2 = 1 - e^{-x_2} + u, \tag{64}$$

where $r(t)$ is a measurable reference. Define the equilibrium family by $x^*(r) = (0, r)$ and $u^*(r) = e^{-r} - 1$, and introduce the scheduling variable $\sigma = e^{-r}$. Pole placement at $\lambda_{1,2} = -2$ gives an LPV controller (57) with $K(\sigma) = [\ 1\ \ (-3 - \sigma)\ ]$. Substituting $\sigma = e^{-r}$ into (60) gives a gain-scheduled controller (denoted by GSC 1):

$$u = e^{-r} - 1 + x_1 - (3 + e^{-r})(x_2 - r). \tag{65}$$

For this controller, unstable closed-loop responses are observed in Figure 7. Alternatively, applying the equilibrium relation $\sigma = e^{-x_2}$ gives a controller (denoted by GSC 2):

$$u = x_1 + e^{-x_2} - 1, \tag{66}$$

which contains a hidden coupling term with $K_h = [\ 0\ \ 3\ ]^\top$. As a direct consequence, the closed-loop differential dynamics have eigenvalues with larger real parts, that is, $\lambda_{1,2} = -1/2 \pm \sqrt{3}/2i$, leading to slower responses as shown Figure 7. Moreover, due to the non-zero residual term, the above controller cannot achieve error-free tracking for time-varying references $r(t)$. The proposed realization (32) takes a path integral of the local LPV
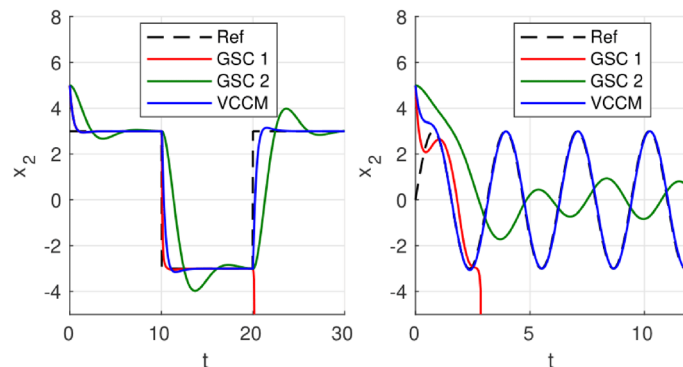

FIGURE 7 State responses of (64) with gain-scheduling and VCCM control for set-point and reference tracking.

controller along a geodesic γ (in this case, γ is a straight line connecting $x^* = [0, r]^\top$ to x), which can be analytically computed via

$$u = u^* + \int_0^1 K(\gamma(s))(x - x^*)ds = \dot{r} + x_1 - 3(x_2 - r) + e^{-x_2} - 1, \tag{67}$$

where $u^* = e^{-r} - 1 + \dot{r}$. Figure 7 shows that this controller can follow both piecewise constant (set-point) and time-varying reference.

Among other local LPV control methods, the so-called *velocity linearization* approach[50] uses a similar technique as continuous linearization, that is, it uses time derivatives of $u(t)$ and $x(t)$ to produce δ_u and δ_x, respectively. Then, the LPV control design is executed on this form and the controller is intuitively, realized by integrating its output δ_u and fed by differentiating the state to produce δ_x. However, the controlled system can converge to an arbitrary response and suffers from implementation issues, due to required differentiation of the measured/estimated state signal. The work[51] proposed a gain scheduling law based on the velocity form as a solution to the realization problem, but without any theoretical analysis of its implications on stability and performance of the resulting closed-loop system. Moreover, this approach will suffer from convergence problems as it fails to guarantee Condition **C2**.

5.3 | Global LPV control

Similarly, we rephrase the global LPV control approach[52] in the VCCM framework.

5.3.1 | Two-stage modeling

In global LPV control, the applied LPV embedding principle can be understood in our setting as follows: the nonlinear system model (1) is rewritten as a virtual system

$$\dot{y} = \mathfrak{f}(y, x, v) := \hat{A}(x)y + \hat{B}(x)v, \tag{68}$$

with $\mathfrak{f}(x, x, u) = f(x, u)$. The main difference w.r.t. the NPV embedding (18) is that (68) has to be linear in (y, v). Note that construction of (68) is not unique and in some cases existence of such a virtual form requires dependence of $\hat{A}$ and $\hat{B}$ on u as well.

By introducing a scheduling map $\sigma = \psi(x)$, where ψ is a vector function such that $\hat{A}(x) = A(\sigma)$ and $\hat{B}(x) = B(\sigma)$ are affine, polynomial or rational functions of σ and restricting the variation of $\sigma(t)$ to a convex set $\Omega \subset \mathbb{R}^{n_\sigma}$, a global LPV model of (5) is formulated as

$$\dot{y} = A(\sigma)y + B(\sigma)u, \quad \sigma \in \Omega. \tag{69}$$

From the modeling viewpoint, the VCCM approach extends the LPV embedding to NPV embeddings, which can reduce the conservativeness of the behavioral embedding principle. Recent research results in the LPV literature also try to formulate extensions of the available toolset in this direction.[53,54]

5.3.2 | Control synthesis

Based on the same synthesis formulation (30) of the VCCM approach, we can obtain an LPV controller

$$v = K(\sigma)y, \tag{70}$$

with a given parameterization of K (e.g., an affine function of σ).

5.3.3 | Control realization

Due to the behavior embedding principle, the real controller takes the form of

$$u = K(\psi(x))x, \tag{71}$$

which stabilizes (69) if it maintains $\sigma(t) \in \Omega$. For set-point tracking, a possible realization[28,29] of (70) is

$$u = u^* + K(\psi(x))(x - x^*). \tag{72}$$

It might be assumed that due to the linearity of (69), stability guarantee for the origin trivially extends to any other set-point (x^*, u^*) by applying a state transformation in the constructed Lyapunov/storage function V. However, it has been recently shown that there is a fundamental gap in the reasoning extending the resulting Lyapunov/storage function for such cases.[29]

For the LPV embedding (68), the VCCM synthesis does not require any Jacobian linearization, and hence it is equivalent to the LPV synthesis. Since the metric is independent of y, the realization for (70) takes the form of

$$u = \kappa^{\text{ff}}(x, x^*, u^*) + K(\psi(x))(x - x^*), \tag{73}$$

where the feed-forward term κ^{ff} is chosen such that Condition **C2** holds. From Theorems 2 and 4, the above controller achieves global stability and performance for all set-points. Thus, the possible loss of stability and performance guarantees of the LPV realization (72) is due to the fact that Condition **C2** generally does not hold for the virtual feed-forward controller $v^* = u^*$. Here we give a brief explanation, see Reference 29 for details.

For simplicity, we consider a simple LPV embedding:

$$\dot{y} = A(\sigma)y + Bv, \quad \sigma = \psi(x). \tag{74}$$

Suppose that there exist an LPV controller $v = K(\sigma)y$ and a quadratic Lyapunov function $V(y) = y^\top My$ with $M \succ 0$ such that the closed-loop system $\dot{y} = A_c(\sigma)y := (A(\sigma) + BK(\sigma))y$ is asymptotically stable, that is, $\forall \sigma \in \Omega$

$$P(\sigma) := A_c^\top(\sigma)M + MA_c(\sigma) \prec 0. \tag{75}$$

For the set-point $(x^*, u^*) \neq (0, 0)$, the closed-loop system under the realization (72) becomes

$$\dot{x} = A(x)x + B[u^* + K(\psi(x))(x - x^*)]. \tag{76}$$

By introducing $\delta x = x - x^*$, the closed-loop error dynamics can be rewritten as

$$\dot{\delta x} = A(\sigma)x - A(\sigma^*)x^* + BK(\sigma)\delta x = (A_c(\sigma) + \Delta(x, x^*))\,\delta x, \tag{77}$$

where Δ is defined by $\Delta(x, x^*)(x - x^*) := (A(\sigma) - A(\sigma^*))x^*$ with $\sigma^* = \psi(x^*)$. By using the quadratic Lyapunov function $V(\delta x) = \delta x^\top M \delta x$, we obtain

$$\dot{V}(\delta x) = \delta x^\top \left(P + \Delta^\top M + M\Delta\right)\delta x. \tag{78}$$

If there exist (x, x^*) with $\psi(x), \psi(x^*) \in \Omega$ such that $P_\Delta(x, x^*) := P(\psi(x)) + \Delta^\top(x, x^*)M + M\Delta(x, x^*)$ is indefinite, then the stability guarantee of the origin cannot be extended to the set-point (x^*, u^*). Performance deterioration or even instability can be observed when Δ is sufficiently large, see Example 5. However, for the VCCM-based realization (73), the term $\Delta(x, x^*)$ in (77) can be compensated by the feed-forward law κ^{ff} since Condition **C2** implies

$$A(\psi(x))x^* + B\kappa^{\text{ff}}(x, x^*, u^*) = 0. \tag{79}$$

Thus, quadratic stability guarantee of the origin is correctly extended to the set-point (x^*, u^*).
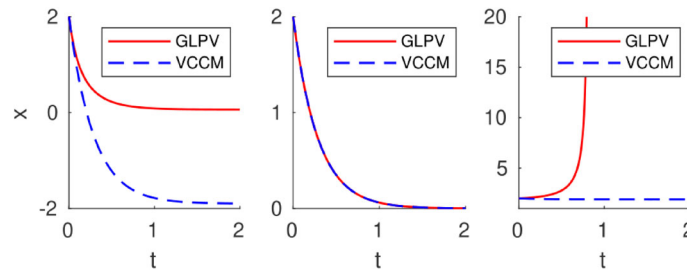
FIGURE 8   State responses of (80) with GLPV and VCCM control for different set-points: $x^* = -1.9$ (left), $x^* = 0$ (middle), and $x^* = 1.9$ (right). For the special case with $x^* = 0$, the VCCM controller is identical to the GLPV controller. For other reference points, the GLPV control may converge to a different equilibrium or yield unstable closed-loop behavior while our VCCM controller converges to the desired equilibrium.

Examples have been reported in References 28 and 55 to show the loss of $\mathcal{L}_2$-gain performance guarantees for set-point tracking when an external disturbance is present. Here, we give an example where, besides loss of performance guarantees, closed-loop instability occurs.

Example 5. Consider the nonlinear scalar system:

$$\dot{x} = f(x) + u := -x + x^3 + u, \tag{80}$$

and its LPV embedding $\dot{y} = -(1 - \sigma)y + v$ where $\sigma = x^2$. We obtain an LPV control gain $K(\sigma) = -2.49 - 1.01\sigma$ by solving (54) with $\alpha = 0.5$, $B_w = 0$, $D_w = 0$, $C = [\ 1\quad 0\ ]^\top$, $D = [\ 0\quad 0.1\ ]^\top$ and $\Omega = [0, 4]$. The closed-loop responses of the global LPV (GLPV) controller (72) and VCCM-based controller (73) for the reference behavior $\mathcal{B}^* = \{(x^*, u^*)|u^* = f(x^*)\}$ are depicted in Figure 8. When the set-point x^* is the origin, both controllers yield the same closed-loop response. When the magnitude of x^* increases, undesired responses (e.g., convergence to a different equilibrium or instability) can be observed with the global LPV controller. In contrast, the VCCM controller remains stable and achieves the setpoint tracking objective.

6 | CONCLUSION

In this article, we present a nonlinear state feedback design approach based on the concept of VCCMs. This approach can be understood as the generalization of several nonlinear control methods. First, it has weaker assumptions than the existing virtual contraction based control methods. Second, it generalizes the CCM based control design to weak stabilization problems. Finally, we argue that the VCCM framework provides a unified generalization of two distinct categories of LPV state-feedback control methods. The VCCM control realization provides rigorous stability and performance guarantees for tracking problems while such properties generally do not hold for conventional LPV-based control realizations.

CONFLICT OF INTEREST STATEMENT
The authors declare that they have no conflict of interest.

DATA AVAILABILITY STATEMENT
The data that support the findings of this study are openly available in VCCM at https://github.com/ruigangwang7/VCCM.

ENDNOTE

*Except in cases where the scheduling variable is truly independent input to the system, for example, outside temperature.

ORCID

Ruigang Wang 🔘 https://orcid.org/0000-0002-9954-4547
Roland Tóth 🔘 https://orcid.org/0000-0001-7570-6129
Patrick J. W. Koelewijn 🔘 https://orcid.org/0000-0003-2378-6264

REFERENCES

1. Sontag E. A Lyapunov-like characterization of asymptotic controllability. *SIAM J Control Optim*. 1983;21(3):462-471.
2. Artstein Z. Stabilization with relaxed controls. *Nonlinear Anal Theory Methods Appl*. 1983;7(11):1163-1173.
3. Sontag E. A 'universal' construction of Artstein's theorem on nonlinear stabilization. *Syst Control Lett*. 1989;13(2):117-123.
4. Slotine JJ, Li W. *Applied Nonlinear Control*. Vol 199. Prentice Hall; 1991.
5. Krstic M, Kanellakopoulos I, Kokotovic P. *Nonlinear and Adaptive Control Design*. Wiley; 1995.
6. Boyd S, El Ghaoui L, Feron E, Balakrishnan V. *Linear Matrix Inequalities in System and Control Theory*. Vol 15. SIAM; 1994.
7. Rantzer A. A dual to Lyapunov's stability theorem. *Syst Control Lett*. 2001;42(3):161-168.
8. Lasserre JB, Henrion D, Prieur C, Trélat E. Nonlinear optimal control via occupation measures and LMI-relaxations. *SIAM J Control Optim*. 2008;47(4):1643-1666.
9. Manchester I, Slotine JJ. Control contraction metrics: convex and intrinsic criteria for nonlinear feedback design. *IEEE Trans Automat Control*. 2017;62(6):3046-3053.
10. Manchester I, Slotine JJ. Robust control contraction metrics: a convex approach to nonlinear state-feedback H_∞ control. *IEEE Control Syst Lett*. 2018;2(3):333-338.
11. Lohmiller W, Slotine JJ. On contraction analysis for non-linear systems. *Automatica*. 1998;34:683-696.
12. Lopez BT, Slotine JJE. Adaptive nonlinear control with contraction metrics. *IEEE Control Syst Lett*. 2020;5(1):205-210.
13. Tsukamoto H, Chung SJ, Slotine JJ. Learning-based adaptive control via contraction theory. *IEEE Conference on Control and Decision*. IEEE; 2021.
14. Duan Z, Wei C, Xie W. Robust model predictive control of uncertain nonlinear time-delay systems via control contraction metric technique. *Int J Robust Nonlinear Control*. 2021;31(8):3330-3353.
15. Sasfi A, Zeilinger MN, Köhler J. Robust adaptive MPC using control contraction metrics. *Automatica*. 2023;155:111169.
16. McCloy R, Wei L, Bao J. A contraction-constrained economic model predictive control for nonlinear processes using multi-timescale models. *J Process Control*. 2023;122:199-207.
17. Singh S, Landry B, Majumdar A, Slotine JJ, Pavone M. Robust feedback motion planning via contraction theory. *Int J Rob Res*. 2023;42(9):655-688.
18. Kong FH, Manchester IR. Contraction analysis of nonlinear noncausal iterative learning control. *Syst Control Lett*. 2020;136:104599.
19. Tsukamoto H, Chung SJ, Slotine JJE. Neural stochastic contraction metrics for learning-based control and estimation. *IEEE Control Syst Lett*. 2020;5(5):1825-1830.
20. Tsukamoto H, Chung SJ, Slotine JJE. Contraction theory for nonlinear stability analysis and learning-based control: a tutorial overview. *Annu Rev Control*. 2021;52:135-169.
21. Singh S, Richards SM, Sindhwani V, Slotine JJE, Pavone M. Learning stabilizable nonlinear dynamics with contraction-based regularization. *Int J Rob Res*. 2021;40(10-11):1123-1150.
22. Hines G, Arcak M, Packard A. Equilibrium-independent passivity: a new definition and numerical certification. *Automatica*. 2011;47:1949-1956.
23. Simpson-Porco J. Equilibrium-independent Dissipativity with quadratic supply rates. *IEEE Trans Autom Control*. 2019;64(4):1440-1455.
24. Rugh W, Shamma J. Research on gain scheduling. *Automatica*. 2000;36:1401-1425.
25. Hoffmann C, Werner H. A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations. *IEEE Trans Control Syst Technol*. 2015;23(2):416-433.
26. Rugh WJ. Analytical framework for gain scheduling. *IEEE Control Syst Mag*. 1991;11:74-84.
27. Fromion V, Scorletti G. A theoretical framework for gain scheduling. *Int J Robust Nonlinear Control*. 2003;13(10):951-982.
28. Scorletti G, Fromion V, De Hillerin S. Toward nonlinear tracking and rejection using LPV control. *IFAC Pap OnLine*. 2015;48(26):13-18.
29. Koelewijn P, Mazzoccante GS, Tóth R, Weiland S. Pitfalls of guaranteeing asymptotic stability in LPV control of nonlinear systems. *2020 European Control Conference (ECC)*. IEEE; 2020:1573-1578.
30. Jouffroy J, Slotine JJ. Methodological remarks on contraction theory. *2004 43rd IEEE Conference on Decision and Control (CDC)*. IEEE; 2004:2537-2543.

31. Wang W, Slotine JJ. On partial contraction analysis for coupled nonlinear oscillators. *Biol Cybern*. 2005;92(1):38-53.

32. Manchester I, Tang J, Slotine JJ. Unifying robot trajectory tracking with control contraction metrics. *Robotics Research*. Springer; 2018:403-418.

33. Reyes-Báez R, van der Schaft A, Jayawardhana B, Pan L. A family of virtual contraction based controllers for tracking of flexible-joints port-Hamiltonian robots: theory and experiments. *Int J Robust Nonlinear Control*. 2020;30(8):3269-3295.

34. Wang R, Koelewijn PJW, Tóth R, Manchester IR. Nonlinear parameter-varying state-feedback design for a gyroscope using virtual control contraction metrics. *Int J Robust Nonlinear Control*. 2021;31(17):8147-8164.

35. Wang R, Manchester I, Bao J. Distributed economic MPC with separable control contraction metrics. *IEEE Control Syst Lett*. 2017;1:104-109.

36. Forni F, Sepulchre R. A differential Lyapunov framework for contraction analysis. *IEEE Trans Automat Control*. 2014;3(59):614-628.

37. Aminzare Z, Sontagy ED. Contraction methods for nonlinear systems: a brief introduction and some open problems. *53rd IEEE Conference on Decision and Control*. IEEE; 2014:3835-3847.

38. van der Schaft A. *L₂-Gain and Passivity in Nonlinear Control*. Springer-Verlag; 1999.

39. Tóth R. *Modeling and Identification of Linear Parameter-Varying Systems*. Springer; 2010.

40. Angeli D, Sontag E. Forward completeness, unboundedness observability, and their Lyapunov characterizations. *Syst Control Lett*. 1999;38(4-5):209-217.

41. Hjartarson A, Seiler P, Packard A. LPVTools: a toolbox for modeling, analysis, and synthesis of parameter varying control systems. *IFAC Pap OnLine*. 2015;48(26):139-145.

42. Parrilo P. Semidefinite programming relaxations for semialgebraic problems. *Math Program*. 2003;96(2):293-320.

43. Leung K, Manchester IR. Nonlinear stabilization via control contraction metrics: a pseudospectral approach for computing geodesics. *2017 American Control Conference (ACC)*. IEEE; 2017:1284-1289.

44. Wang R, Manchester IR. Continuous-time dynamic realization for nonlinear stabilization via control contraction metrics. *2020 American Control Conference (ACC)*. IEEE; 2020:1619-1624.

45. Cisneros PS, Werner H. Wide range stabilization of a pendubot using quasi-LPV predictive control. *IFAC Pap OnLine*. 2019;52(28):164-169.

46. Iacob LC, Tóth R, Schoukens M. Koopman form of nonlinear systems with inputs. *Automatica*. 2024;162:111525.

47. Andrieu V, Prieur C. Uniting two control Lyapunov functions for affine systems. *IEEE Trans Automat Contr*. 2010;55(8):1923-1927.

48. Dullerud G, Paganini F. *A Course in Robust Control Theory: A Convex Approach*. Springer Science & Business Media; 2013.

49. Shamma J, Athans M. Analysis of gain scheduled control for nonlinear plants. *IEEE Trans Automat Contr*. 1990;35(8):898-907.

50. Leith D, Leithhead W. Gain-scheduled nonlinear systems: dynamic analysis by velocity based linearization families. *Int J Control*. 1998;70:289-317.

51. Mehendale CS, Grigoriadis KM. Performance of LPV gain-scheduled systems. *2006 American Control Conference*. IEEE; 2006:6.

52. Shamma JS, Athans M. Guaranteed properties of gain scheduled control for linear parameter-varying plants. *Automatica*. 1991;27(3):559-564.

53. Rotondo D, Witczak M. Analysis and design of quadratically bounded QPV control systems. *IFAC Pap OnLine*. 2019;52(28):76-81.

54. Sala A, Ariño C, Robles R. Gain-scheduled control via convex nonlinear parameter varying models. *IFAC Pap OnLine*. 2019;52(28):70-75.

55. Koelewijn P, Tóth R, Nijmeijer H. Linear parameter-varying control of nonlinear systems based on incremental stability. *IFAC Pap OnLine*. 2019;52(28):38-43.