# Robust Local Stabilization of Nonlinear Systems With Controller-Dependent Norm Bounds: A Convex Approach With Input-Output Sampling

Sze Kwan Cheah[ID], *Graduate Student Member, IEEE*, Diganta Bhattacharjee[ID],
Maziar S. Hemati[ID], *Member, IEEE*, and Ryan J. Caverly[ID], *Member, IEEE*

*Abstract*—**This letter presents a framework for synthesizing a robust full-state feedback controller for systems with unknown nonlinearities. Our approach characterizes input-output behavior of the nonlinearities in terms of local norm bounds using available sampled data corresponding to a known region about an equilibrium point. A challenge in this approach is that if the nonlinearities have explicit dependence on the control inputs, an *a priori* selection of the control input sampling region is required to determine the local norm bounds. This leads to a "chicken and egg" problem, where the local norm bounds are required for controller synthesis, but the region of control inputs needed to be characterized cannot be known prior to synthesis of the controller. To tackle this issue, we constrain the closed-loop control inputs within the sampling region while synthesizing the controller. As the resulting synthesis problem is non-convex, three semi-definite programs (SDPs) are obtained through convex relaxations of the main problem, and an iterative algorithm is constructed using these SDPs for control synthesis. Two numerical examples are included to demonstrate the effectiveness of the proposed algorithm.**

*Index Terms*—**Stability of nonlinear systems, robust control, LMIs, numerical algorithms.**

## I. INTRODUCTION

A LARGE portion of the existing literature on nonlinear control comprises model-based approaches (e.g., backstepping, feedback linearization [1]) that depend on the availability of a sufficiently accurate analytical model of the system

or plant to be controlled. There is an inherent assumption that such an analytic model can be obtained from the fundamental laws governing the system. However, there are many instances where a reliable and/or accurate analytical model of the system cannot be obtained in practice. Robust control theory can also be used to design a stabilizing controller for a nonlinear system under the assumption that at least a nominal realization of the system, either linear time-invariant (LTI) or linear time-varying (LTV), can be obtained (see, e.g., [2], [3], [4, Ch. 9]). In this setting, the system is described in linear fractional transformation (LFT) form, where the nominal system is connected in feedback with a 'perturbation' that captures all aspects of the system that do not fit within an LTI or LTV framework (e.g., nonlinearity, uncertainty, time delays). The control synthesis is then based on the input-output (I/O) properties of the perturbation, which are assumed to hold for some known structure and/or set of perturbations (e.g., integral quadratic constraints [5], structured singular value [6]). For a complex system, however, it is not trivial to quantify these I/O properties and ascertain if the accompanying assumptions hold, especially when the analytical form of the perturbation is not explicitly known.

Data-driven techniques are now becoming increasingly popular to overcome these issues [7]. For example, strategies are being proposed for determining I/O properties from sampled data (like dissipation inequalities [8] and passivity [9]). Also, data-driven approaches for robust control, with a focus on perturbations representing parametric uncertainties and unknown nonlinearity driven by states and/or parameters, are gaining popularity [10], [11], [12]. However, the class of perturbations for which the unknown nonlinearity is a function of both the states and control inputs is relatively less studied. Existing data-driven methods for this class of perturbations are either guaranteed to work in a small neighborhood of the equilibrium [13] or require extensive tuning of the associated control parameters to satisfy the underlying assumptions [14]. In light of these challenges associated with purely data-driven control, we have adopted an approach where I/O properties of the perturbation are established through data and subsequently used for control synthesis.

We consider a general nonlinear system and partition it into an LFT form, where the nominal system captures the system's LTI dynamics about an equilibrium and the perturbation

comprises higher-order nonlinearities, which are functions of both the states and control inputs and are not necessarily known analytically. Assuming I/O samples of the perturbation are available, local norm bounds are derived. A robust state-feedback controller is synthesized that asymptotically stabilizes the nonlinear system locally within the sampled region for all perturbations satisfying the norm bounds. Since the norm bounds, which depend on control inputs, are used in the synthesis, the synthesized controller must ensure that the control inputs remain within the sampling set. Moreover, the main synthesis problem involves non-convex constraints. We relax the main problem into three semi-definite programs (SDPs) that are solved iteratively. To summarize, the contribution of this letter is an iterative control synthesis method that results in local asymptotic stabilization of a nonlinear system, where the system's nonlinearities are analytically unknown, but available through sampling, and depend explicitly on control inputs.

*Notation:* The symbol $\mathcal{N}_n$ is a shorthand for the set $\{1, 2, \ldots, n\}$, and $\|\cdot\|$ denotes the 2-norm for vectors and spectral norm for matrices. An $n$-dimensional vector of zeros with the $i$-th entry equal to one is denoted by $\mathbf{1}_{n_i}$. We use $\mathbf{A} > 0$ to denote a symmetric, positive definite matrix $\mathbf{A}$. The maximum singular value of $\mathbf{M}$ is denoted by $\bar{\sigma}(\mathbf{M})$. For a given $\mathbf{E} > 0$, an ellipsoid centered at the origin is denoted by $\mathcal{E}_n(\mathbf{E}) = \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{E}^{-1}\mathbf{x}\| \leq 1\}$. Finally, $\mathcal{B}_n$ denotes the closed unit-norm ball in $\mathbb{R}^n$.

## II. Problem Formulation

Consider a nonlinear dynamic system of the form

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

where $\mathbf{x} \in \mathbb{R}^{n_x}$ and $\mathbf{u} \in \mathbb{R}^{n_u}$ are the state and control input vectors, respectively, and $\mathbf{f} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ is a nonlinear function subject to the following assumption.

*Assumption 1:* $\mathbf{f}$ is not precisely known. However, there exists at least one equilibrium $(\mathbf{x}_0, \mathbf{u}_0)$ such that $\mathbf{f}(\mathbf{x}_0, \mathbf{u}_0) = \mathbf{0}$, and $\mathbf{A} = \frac{\partial \mathbf{f}}{\partial \mathbf{x}}|_{(\mathbf{x}_0, \mathbf{u}_0)}$, $\mathbf{B}_1 = \frac{\partial \mathbf{f}}{\partial \mathbf{u}}|_{(\mathbf{x}_0, \mathbf{u}_0)}$ are known.

All the developments in this letter are based on the above assumption. If multiple equilibria are known, we choose the one that is relevant for the problem at hand. Next, by setting $\mathbf{x} = \mathbf{x}_0 + \delta\mathbf{x}$, $\mathbf{u} = \mathbf{u}_0 + \delta\mathbf{u}$, we can rewrite (1) as

$$\delta\dot{\mathbf{x}} = \mathbf{A}\delta\mathbf{x} + \mathbf{B}_1\delta\mathbf{u} + \mathbf{\Delta}(\delta\mathbf{x}, \delta\mathbf{u}), \tag{2}$$

where $\mathbf{A}, \mathbf{B}_1$ are as defined in Assumption 1, and $\mathbf{\Delta} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ is a function that captures the higher-order terms. Note that $\mathbf{\Delta}$, whose analytical form is not available, is a function of both $\delta\mathbf{x}, \delta\mathbf{u}$. Also, we assume that $\mathbf{\Delta}$ is memoryless and static. Thus, system (2) can be expressed in an LFT form having the nominal LTI dynamics $\delta\dot{\mathbf{x}} = \mathbf{A}\delta\mathbf{x} + \mathbf{B}_1\delta\mathbf{u}$ and the perturbation $\mathbf{\Delta}$.

Although $\mathbf{\Delta}$ is not known analytically, we assume that a finite number of input-output samples of $\mathbf{\Delta}$ are available from either experiments or high-fidelity numerical simulations. We also assume that the sampling is carried out in a known region around the equilibrium, such as over a $N$-point grid where $N$ is large. We refer to this region as the sampling region $\mathcal{S} = \mathbb{X} \times \mathbb{U}$, where $\mathbb{X} \subset \mathbb{R}^{n_x}$ and $\mathbb{U} \subset \mathbb{R}^{n_u}$ are known, compact sets that contain the respective origins in the interiors. Therefore, we have access to $N$ input-output samples

$\{(\delta\mathbf{x}^{(k)}, \delta\mathbf{u}^{(k)}), \mathbf{\Delta}(\delta\mathbf{x}^{(k)}, \delta\mathbf{u}^{(k)})\}_{k \in \mathcal{N}_N}$, where $(\delta\mathbf{x}^{(k)}, \delta\mathbf{u}^{(k)}) \in \mathcal{S}$ for all $k \in \mathcal{N}_N$. Upon investigating these samples, we can deduce the following:

- If the vector $\mathbf{\Delta}(\delta\mathbf{x}^{(k)}, \delta\mathbf{u}^{(k)})$ corresponding to $\delta\mathbf{x}^{(k)} \neq \mathbf{0}$ and $\delta\mathbf{u}^{(k)} \neq \mathbf{0}$ contains elements that are identically 0, then the original system (1) has states that are governed by purely LTI dynamics. This, in conjunction with the nonzero entries (say, $n_w$ of those), can be used to find a realization of $\mathbf{\Delta}(\delta\mathbf{x}^{(k)}, \delta\mathbf{u}^{(k)})$ of the form $\mathbf{\Delta}(\delta\mathbf{x}^{(k)}, \delta\mathbf{u}^{(k)}) = [\Delta_1 \cdots 0 \Delta_{n_w} \cdots]^\mathsf{T}$, where $\Delta_i : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$, $i \in \mathcal{N}_{n_w}$ are the nonlinearties corresponding to the nonzero entries. Thus, the above can be reformulated as $\mathbf{\Delta}(\delta\mathbf{x}^{(k)}, \delta\mathbf{u}^{(k)}) = \mathbf{B}_2\mathbf{w}^{(k)}$ where $\mathbf{w}^{(k)} = \left[w_1^{(k)}\ w_2^{(k)}\ \cdots\ w_{n_w}^{(k)}\right]^\mathsf{T} \in \mathbb{R}^{n_w}$ contains outputs of all the nonlinearities and $\mathbf{B}_2 \in \mathbb{R}^{n_x \times n_w}$ properly distributes the elements of $\mathbf{w}^{(k)}$.

- We can also identify the individual states and control inputs that drive each nonlinear function $\Delta_i$. Based on this, the sampled input to $\Delta_i$ takes the form $\mathbf{v}_i^{(k)} = [\delta x_1^{(k)}\ \cdots\ 0\ \cdots\ \delta u_1^{(k)}\ \cdots\ 0\ \cdots]^\mathsf{T} \in \mathbb{R}^{n_x + n_u}$ which is equivalent to $\mathbf{v}_i^{(k)} = \mathbf{C}_i\delta\mathbf{x}^{(k)} + \mathbf{D}_i\delta\mathbf{u}^{(k)}$ with $\mathbf{C}_i \in \mathbb{R}^{(n_x + n_u) \times n_x}$, $\mathbf{D}_i \in \mathbb{R}^{(n_x + n_u) \times n_u}$ known. Therefore, for each $k \in \mathcal{N}_N$, we have $w_i^{(k)} = \Delta_i(\mathbf{v}_i^{(k)})$, $i \in \mathcal{N}_{n_w}$. Although expressed in this form, it is understood that each $\Delta_i$ maps $\mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$ to $\mathbb{R}$.

- The samples can be used to prescribe *empirical norm bounds* on the inputs-outputs of each $\Delta_i$. Specifically, we intend to find $\gamma_i > 0$ for each $i \in \mathcal{N}_{n_w}$ such that $(w_i^{(k)})^2 \leq \gamma_i^2 \left\|\mathbf{v}_i^{(k)}\right\|^2$ holds for all $k \in \mathcal{N}_N$. To this end, for each pair $(i, k) \in \mathcal{N}_{n_w} \times \mathcal{N}_N$, we define $\gamma_i^{(k)} = ((w_i^{(k)})^2 / \left\|\mathbf{v}_i^{(k)}\right\|^2)^{1/2}$ and stack all such bounds in a matrix $\Gamma = [\Gamma_1\ \Gamma_2\ \cdots\ \Gamma_{n_w}]$, where $\Gamma_i = \left[\gamma_i^{(1)}\ \gamma_i^{(2)}\ \cdots\ \gamma_i^{(N)}\right]^\mathsf{T}$. The empirical bounds are then specified as the maximum over each column, i.e., $\gamma_i = \max \Gamma_i$. If we seek the bounds over a region $\mathbb{D} \subseteq \mathcal{S}$, the same procedure can be repeated with the sampled data corresponding to $\mathbb{D}$.

Using the above information, system (2) can be rewritten as

$$\begin{aligned} \delta\dot{\mathbf{x}} &= \mathbf{A}\delta\mathbf{x} + \mathbf{B}_1\delta\mathbf{u} + \mathbf{B}_2\mathbf{w}, \\ w_i &= \Delta_i(\mathbf{v}_i), \ i \in \mathcal{N}_{n_w}, \\ \mathbf{v}_i &= (\mathbf{C}_i\delta\mathbf{x} + \mathbf{D}_i\delta\mathbf{u}), \ i \in \mathcal{N}_{n_w}, \end{aligned} \tag{3}$$

with the following standing assumption for a given $\mathbb{D} \subseteq \mathcal{S}$.

*Assumption 2:* Let $\gamma_i$, $i \in \mathcal{N}_{n_w}$ be the empirical bounds corresponding to $\mathbb{D}$. Each input-output tuple $(\mathbf{v}_i, w_i)$ satisfies the bound $\gamma_i$ in $\mathbb{D}$, i.e., $w_i^2 \leq \gamma_i^2\|\mathbf{v}_i\|^2$, for all $(\delta\mathbf{x}, \delta\mathbf{u}) \in \mathbb{D}$ and for all $i \in \mathcal{N}_{n_w}$. Also, for each $i \in \mathcal{N}_{n_w}$, let $\mathbf{\Delta}_{\mathbb{D}_i}$ be the set of functions $\Delta_i : \mathbb{D} \to \mathbb{R}$ for which the bound $\gamma_i$ holds.

The above assumption is reasonable since our knowledge is restricted to the extent provided by the sampled data and analytical forms of $\Delta_i$s are unknown. Now, we are interested in designing a state-feedback control law $\delta\mathbf{u} = \mathbf{K}\delta\mathbf{x}$ for the open-loop system (3), under Assumption 2 with $\mathbb{D} = \mathbb{X}_c \times \mathbb{U}_c$ for some $\mathbb{X}_c \subseteq \mathbb{X}$, $\mathbb{U}_c \subseteq \mathbb{U}$ containing the respective origins in the interiors. The closed-loop system thus becomes

$$\delta\dot{\mathbf{x}} = (\mathbf{A} + \mathbf{B}_1\mathbf{K})\delta\mathbf{x} + \mathbf{B}_2\mathbf{w},$$

$$w_i = \Delta_i(\mathbf{v}_i), \ i \in \mathcal{N}_{n_w},$$
$$\mathbf{v}_i = (\mathbf{C}_i + \mathbf{D}_i\mathbf{K})\delta\mathbf{x}, \ i \in \mathcal{N}_{n_w}. \quad (4)$$

The goal is to synthesize $\mathbf{K}$ to certify the closed-loop system (4) asymptotically stable in the largest local region $\mathbb{X}_c \subseteq \mathbb{X}$, for a choice of $\mathbb{U}_c \subseteq \mathbb{U}$ and the corresponding bounds $\gamma_i$ and set of functions $\mathbf{\Delta}_{\mathbb{D}_i}$, $i \in \mathcal{N}_{n_w}$. However, as the synthesis uses $\gamma_i$, we need to verify whether the closed-loop control trajectories satisfy $\delta\mathbf{u} = \mathbf{K}\delta\mathbf{x} \in \mathbb{U}_c$ for $\mathbf{K}$ to be consistent with the data. This requires the use of an iterative approach, which is described in detail in the synthesis presented in the next section. Before discussing the control synthesis, we outline a few useful matrix inequality results.

*Lemma 1 (Young's Relation [15]):* Consider $\mathbf{X} \in \mathbb{R}^{m \times n}$ and $\mathbf{Y} \in \mathbb{R}^{m \times n}$. For any $\mathbf{S} > 0$, it holds that

$$\mathbf{X}^\mathsf{T}\mathbf{Y} + \mathbf{Y}^\mathsf{T}\mathbf{X} \leq \mathbf{X}^\mathsf{T}\mathbf{S}^{-1}\mathbf{X} + \mathbf{Y}^\mathsf{T}\mathbf{S}\mathbf{Y}. \quad (5)$$

A special case of Lemma 1 that will prove useful in the control synthesis presented in the next section considers $\mathbf{S} = \mathbf{I}$, $\mathbf{X} = \mathbf{H} \in \mathbb{R}^{n_x \times n_x}$ and $\mathbf{Y} = \mathbf{H}_0 \in \mathbb{R}^{n_x \times n_x}$, which leads to

$$\mathbf{H}^\mathsf{T}\mathbf{H} \geq \mathbf{H}^\mathsf{T}\mathbf{H}_0 + \mathbf{H}_0^\mathsf{T}\mathbf{H} - \mathbf{H}_0^\mathsf{T}\mathbf{H}_0. \quad (6)$$

## III. CONTROL SYNTHESIS

This section describes the synthesis of a static state-feedback controller $\mathbf{K}$, which requires breaking the main synthesis problem into different sub-problems and iterating over these sub-problems to obtain a controller that is certified to render the closed-loop system (4) asymptotically stable within $\mathbb{X}_c$. We will start by specifying structures of the sets $\mathbb{X}_c$ and $\mathbb{U}_c$ that will be used in the remainder of this letter. The local region $\mathbb{X}_c$ is taken to be a family of ellipsoids parameterized by $\mathbf{W} > 0$, i.e., $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W})$, where $\mathbf{W}$ is chosen appropriately such that $\mathbb{X}_c \subseteq \mathbb{X}$. Similarly, we take $\mathbb{U}_c = r\mathcal{B}_{n_u}$, where $r > 0$ is the parameter related to the control input magnitude and is chosen such that $\mathbb{U}_c \subseteq \mathbb{U}$. With these sets defined, the main synthesis problem is summarized in the next result.

*Theorem 1:* Let $\mathbf{W} > 0$ and $r > 0$ be chosen such that $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W}) \subseteq \mathbb{X}$ and $\mathbb{U}_c = r\mathcal{B}_{n_u} \subseteq \mathbb{U}$, respectively. Suppose Assumption 2 holds with $\mathbb{D} = \mathbb{X}_c \times \mathbb{U}_c$. Then, the closed-loop system (4) is locally asymptotically stable in $\mathbb{X}_c$ for all $\Delta_i \in \mathbf{\Delta}_{\mathbb{D}_i}$, $i \in \mathcal{N}_{n_w}$, if there exist $\mathbf{P} > 0$, $\mathbf{K} \in \mathbb{R}^{n_u \times n_x}$, $\tau > 0$, and $\lambda_i > 0$, $i \in \mathcal{N}_{n_w}$, such that

$$\begin{bmatrix} \mathbf{P}(\mathbf{A}+\mathbf{B}_1\mathbf{K}) + (\mathbf{A}+\mathbf{B}_1\mathbf{K})^\mathsf{T}\mathbf{P} & \mathbf{PB}_2 & \mathbf{\Theta} \\ \mathbf{B}_2^\mathsf{T}\mathbf{P} & \mathbf{\Lambda} & \mathbf{0} \\ \mathbf{\Theta}^\mathsf{T} & \mathbf{0} & \mathbf{\Xi} \end{bmatrix} < 0, \quad (7)$$

$$\begin{bmatrix} \tau^2\mathbf{I} & \mathbf{K} \\ \mathbf{K}^\mathsf{T} & \mathbf{W}^{-1}\mathbf{W}^{-1} \end{bmatrix} \geq 0, \quad (8)$$

$$\tau \leq r, \quad (9)$$

where $\mathbf{\Lambda} = -\mathrm{diag}(\lambda_1, \ldots, \lambda_{n_w})$, and

$$\mathbf{\Xi} = -\mathrm{diag}(\frac{\lambda_1}{\gamma_1^2}\mathbf{I}, \ldots, \frac{\lambda_{n_w}}{\gamma_{n_w}^2}\mathbf{I}), \ \mathbf{\Theta} = \begin{bmatrix} \lambda_1\mathbf{\Phi}_1, \ldots, \lambda_n\mathbf{\Phi}_{n_w} \end{bmatrix}, \quad (10)$$

with $\mathbf{\Phi}_i = \mathbf{C}_i^\mathsf{T} + \mathbf{K}^\mathsf{T}\mathbf{D}_i^\mathsf{T}$, $i \in \mathcal{N}_{n_w}$.

*Proof:* We establish the proof in three parts: first, we derive a condition that ensures asymptotic stability of the closed-loop system within the local region $\mathbb{X}_c$ under the assumption that $\delta\mathbf{u} = \mathbf{K}\delta\mathbf{x} \in \mathbb{U}_c$ for all $\delta\mathbf{x} \in \mathbb{X}_c$; then, we obtain the equivalent

stability condition (7); finally, we constrain the control signal such that $\delta\mathbf{u} = \mathbf{K}\delta\mathbf{x} \in \mathbb{U}_c$ for all $\delta\mathbf{x} \in \mathbb{X}_c$, which leads to (8) and (9).

*Part-1:* Define the candidate Lyapunov function $V = \delta\mathbf{x}^\mathsf{T}\mathbf{P}\delta\mathbf{x}$ with $\mathbf{P} > 0$. Taking the time-derivative of $V$ and using (4) results in

$$\dot{V} = \delta\mathbf{x}^\mathsf{T}\mathbf{P}\delta\dot{\mathbf{x}} + \delta\dot{\mathbf{x}}^\mathsf{T}\mathbf{P}\delta\mathbf{x}$$
$$= \begin{bmatrix} \delta\mathbf{x} \\ \mathbf{w} \end{bmatrix}^\mathsf{T} \begin{bmatrix} \mathbf{P}(\mathbf{A}+\mathbf{B}_1\mathbf{K}) + (\mathbf{A}+\mathbf{B}_1\mathbf{K})^\mathsf{T}\mathbf{P} & \mathbf{PB}_2 \\ \mathbf{B}_2^\mathsf{T}\mathbf{P} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \mathbf{w} \end{bmatrix}.$$

The inputs and outputs of each $\Delta_i$ can be rewritten using (4) as

$$\begin{bmatrix} \mathbf{v}_i \\ w_i \end{bmatrix} = \begin{bmatrix} (\mathbf{C}_i + \mathbf{D}_i\mathbf{K}) & \mathbf{0} \\ \mathbf{0} & \mathbf{1}_{n_{w_i}}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \mathbf{w} \end{bmatrix}. \quad (11)$$

Now, suppose $\delta\mathbf{u} = \mathbf{K}\delta\mathbf{x} \in \mathbb{U}_c$ holds for all $\delta\mathbf{x} \in \mathbb{X}_c$. Then, under Assumption 2 with $\mathbb{D} = \mathbb{X}_c \times \mathbb{U}_c$, we have $w_i^2 \leq \gamma_i^2 \|\mathbf{v}_i\|^2$ for each $\Delta_i \in \mathbf{\Delta}_{\mathbb{D}_i}$, $i \in \mathcal{N}_{n_w}$, and for all $\delta\mathbf{x} \in \mathbb{X}_c$, which is equivalently given by

$$\begin{bmatrix} \mathbf{v}_i \\ w_i \end{bmatrix}^\mathsf{T} \begin{bmatrix} \gamma_i^2\mathbf{I} & \mathbf{0} \\ \mathbf{0} & -1 \end{bmatrix} \begin{bmatrix} \mathbf{v}_i \\ w_i \end{bmatrix} \geq 0. \quad (12)$$

This, along with (11), leads to a quadratic constraint (QC) in $\delta\mathbf{x}$ and $\mathbf{w}$ for each $i \in \mathcal{N}_{n_w}$, expressed as

$$\begin{bmatrix} \delta\mathbf{x} \\ \mathbf{w} \end{bmatrix}^\mathsf{T} \begin{bmatrix} \gamma_i^2\mathbf{\Phi}_i\mathbf{\Phi}_i^\mathsf{T} & \mathbf{0} \\ \mathbf{0} & -\mathbf{1}_{n_{w_i}}\mathbf{1}_{n_{w_i}}^\mathsf{T} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x} \\ \mathbf{w} \end{bmatrix} \geq 0, \quad (13)$$

where $\mathbf{\Phi}_i = \mathbf{C}_i^\mathsf{T} + \mathbf{K}^\mathsf{T}\mathbf{D}_i^\mathsf{T}$. In the current setting, all QCs of the form (13) for $\Delta_i \in \mathbf{\Delta}_{\mathbb{D}_i}$, $i \in \mathcal{N}_{n_w}$ hold for all $\delta\mathbf{x} \in \mathbb{X}_c$. This would imply, through the S-procedure, that $\dot{V} < 0$ for all $\delta\mathbf{x} \in \mathbb{X}_c$ if there exists $\lambda_i \geq 0$, $i \in \mathcal{N}_{n_w}$ such that

$$\begin{bmatrix} \mathbf{P}(\mathbf{A}+\mathbf{B}_1\mathbf{K}) + (\mathbf{A}+\mathbf{B}_1\mathbf{K})^\mathsf{T}\mathbf{P} & \mathbf{PB}_2 \\ \mathbf{B}_2^\mathsf{T}\mathbf{P} & \mathbf{0} \end{bmatrix}$$
$$+ \sum_{i=1}^{n_w} \lambda_i \begin{bmatrix} \gamma_i^2\mathbf{\Phi}_i\mathbf{\Phi}_i^\mathsf{T} & \mathbf{0} \\ \mathbf{0} & -\mathbf{1}_{n_{w_i}}\mathbf{1}_{n_{w_i}}^\mathsf{T} \end{bmatrix} < 0. \quad (14)$$

This concludes the first part of the proof where we have derived a condition for local asymptotic stability.

*Part-2:* Making the restriction that $\lambda_i > 0$ for all $i \in \mathcal{N}_{n_w}$, allows for (14) to be rewritten as

$$\begin{bmatrix} \mathbf{P}(\mathbf{A}+\mathbf{B}_1\mathbf{K}) + (\mathbf{A}+\mathbf{B}_1\mathbf{K})^\mathsf{T}\mathbf{P} & \mathbf{PB}_2 \\ \mathbf{B}_2^\mathsf{T}\mathbf{P} & \mathbf{\Lambda} \end{bmatrix}$$
$$- \sum_{i=1}^{n_w} \begin{bmatrix} -\lambda_i\mathbf{\Phi}_i\frac{\gamma_i^2}{\lambda_i}\mathbf{\Phi}_i^\mathsf{T}\lambda_i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} < 0, \quad (15)$$

where $\mathbf{\Lambda} = -\sum_{i=1}^{n_w} \lambda_i\mathbf{1}_{n_{w_i}}\mathbf{1}_{n_{w_i}}^\mathsf{T} = -\mathrm{diag}(\lambda_1, \ldots, \lambda_{n_w})$. Each matrix in the sum above can be expressed as

$$\begin{bmatrix} -\lambda_i\mathbf{\Phi}_i\frac{\gamma_i^2}{\lambda_i}\mathbf{\Phi}_i^\mathsf{T}\lambda_i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \lambda_i\mathbf{\Phi}_i \\ \mathbf{0} \end{bmatrix} \left( -\frac{\gamma_i^2}{\lambda_i}\mathbf{I} \right) \begin{bmatrix} \lambda_i\mathbf{\Phi}_i^\mathsf{T} & \mathbf{0} \end{bmatrix},$$

which leads to

$$\sum_{i=1}^{n_w} \begin{bmatrix} -\lambda_i\mathbf{\Phi}_i\frac{\gamma_i^2}{\lambda_i}\mathbf{\Phi}_i^\mathsf{T}\lambda_i & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{\Theta} \\ \mathbf{0} \end{bmatrix} (\mathbf{\Xi})^{-1} \begin{bmatrix} \mathbf{\Theta} \\ \mathbf{0} \end{bmatrix}^\mathsf{T}, \quad (16)$$

where $\boldsymbol{\Xi}$ and $\boldsymbol{\Theta}$ are as shown in (10). Thus, (15) becomes

$$\begin{bmatrix} \mathbf{P}(\mathbf{A}+\mathbf{B}_1\mathbf{K})+(\mathbf{A}+\mathbf{B}_1\mathbf{K})^\mathsf{T}\mathbf{P} & \mathbf{P}\mathbf{B}_2 \\ \mathbf{B}_2^\mathsf{T}\mathbf{P} & \boldsymbol{\Lambda} \end{bmatrix}$$
$$ -\begin{bmatrix} \boldsymbol{\Theta} \\ \mathbf{0} \end{bmatrix}(\boldsymbol{\Xi})^{-1}\begin{bmatrix} \boldsymbol{\Theta} \\ \mathbf{0} \end{bmatrix}^\mathsf{T} < 0. \quad (17)$$

Applying the Schur complement to (17) leads to (7), which completes the proof for the equivalent stability condition.

*Part-3:* To obtain a controller $\mathbf{K}$ that satisfies the assumption used in *Part-1* (i.e., $\delta\mathbf{u} = \mathbf{K}\delta\mathbf{x} \in \mathbb{U}_c$ holds for all $\delta\mathbf{x} \in \mathbb{X}_c$) and is, therefore, consistent with the values of $\gamma_i$ used, we need to ensure $\|\delta\mathbf{u}\| = \|\mathbf{K}\delta\mathbf{x}\| \le r$. A bound on $\|\delta\mathbf{u}\|$ is found using the definition of $\bar{\sigma}(\cdot)$ and knowing that $\|\mathbf{W}^{-1}\delta\mathbf{x}\| \le 1$, for all $\delta\mathbf{x} \in \mathbb{X}_c$, which yields

$$\|\delta\mathbf{u}\| = \|\mathbf{K}\delta\mathbf{x}\| = \left\|\mathbf{K}\mathbf{W}\mathbf{W}^{-1}\delta\mathbf{x}\right\|$$
$$ \le \bar{\sigma}(\mathbf{K}\mathbf{W})\left\|\mathbf{W}^{-1}\delta\mathbf{x}\right\| \le \bar{\sigma}(\mathbf{K}\mathbf{W}). \quad (18)$$

Thus, ensuring $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le r$ guarantees that $\|\delta\mathbf{u}\| \le r$. We first find a $\tau > 0$ such that $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le \tau$. This condition can be expressed equivalently as $\tau^2\mathbf{I} \ge (\mathbf{K}\mathbf{W}\mathbf{W}\mathbf{K}^\mathsf{T})$, applying the Schur complement to which leads to (8). Finally, specifying $\tau \le r$ means $\|\delta\mathbf{u}\| \le \bar{\sigma}(\mathbf{K}\mathbf{W}) \le \tau \le r$. This completes the last part of the proof. ∎

*Remark 1:* Note that although we have considered the 2-norm bound in this letter, Theorem 1 is suitable for other quadratic characterizations of I/O behavior (of the form (12)) of the nonlinearities (e.g., weighted 2-norm bounds). This would involve suitably modifying the stability condition (7).

*Remark 2:* Theorem 1 holds for any $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W}) \subseteq \mathbb{X}$ and $\mathbb{U}_c = r\mathcal{B}_{n_u} \subseteq \mathbb{U}$, with the values of $\gamma_i$ computed from the sampled data corresponding to the chosen region $\mathbb{D} = \mathbb{X}_c \times \mathbb{U}_c$. Hence, one could ideally select $\mathbf{W} > 0$ such that $\mathbb{X}_c$ is the largest ellipsoid contained in $\mathbb{X}$, choose a $r > 0$, and find a feasible point satisfying all the constraints in Theorem 1 to obtain a controller which would asymptotically stabilize the closed-loop system (4) for all initial conditions within this ellipsoid. However, it is not trivial to find such a feasible point for the constraints involved. Specifically, the matrix inequality (7) is non-convex, as it is bilinear in the variables $\mathbf{P}$, $\mathbf{K}$. Also, $\boldsymbol{\Theta}$ is quadratic in the variables $\lambda_i$, $\mathbf{K}$. The bilinearity issue is well-known and can be addressed by applying a congruence transformation and a change of variables (see, e.g., [15, p. 119]). However, the matrix inequality in (8) then becomes non-convex in the transformed variables. We address these issues by reformulating and/or relaxing these constraints into convex ones. The convex constraints are then used to set up three different semi-definite programs (SDPs) for the controller synthesis.

We start by deriving an alternative form of (7) using a congruence transformation with $\mathrm{diag}(\mathbf{P}^{-1}, \mathbf{I}, \mathbf{I})$ as

$$\begin{bmatrix} \mathbf{A}\mathbf{R}+\mathbf{B}_1\mathbf{F}+\mathbf{R}\mathbf{A}^\mathsf{T}+\mathbf{F}^\mathsf{T}\mathbf{B}_1^\mathsf{T} & \mathbf{B}_2 & \tilde{\boldsymbol{\Theta}} \\ \mathbf{B}_2^\mathsf{T} & \boldsymbol{\Lambda} & \mathbf{0} \\ \tilde{\boldsymbol{\Theta}}^\mathsf{T} & \mathbf{0} & \boldsymbol{\Xi} \end{bmatrix} < 0, \quad (19)$$

where $\mathbf{R} = \mathbf{P}^{-1}$, $\mathbf{F} = \mathbf{K}\mathbf{P}^{-1} = \mathbf{K}\mathbf{R}$, and $\tilde{\boldsymbol{\Theta}} = \left[\lambda_1\tilde{\boldsymbol{\Phi}}_1, \ldots, \lambda_{n_w}\tilde{\boldsymbol{\Phi}}_{n_w}\right]$ with $\tilde{\boldsymbol{\Phi}}_i = \mathbf{R}\mathbf{C}_i^\mathsf{T} + \mathbf{F}^\mathsf{T}\mathbf{D}_i^\mathsf{T}$. Note that (19) is not an LMI in the variables $\mathbf{R}, \mathbf{F}, \lambda_i$. However, if the values

of $\lambda_i$s are known or given, (19) is an LMI in $\mathbf{R}, \mathbf{F}$. Another approach for deriving an LMI form of (19) is by setting $\lambda_i = \lambda$ and applying a congruence transformation on (19) with $\mathrm{diag}(\sqrt{\lambda}\mathbf{I}, (1/\sqrt{\lambda})\mathbf{I}, (1/\sqrt{\lambda})\mathbf{I})$. These steps lead to

$$\begin{bmatrix} \mathbf{A}\mathbf{R}+\mathbf{B}_1\mathbf{F}+\mathbf{R}\mathbf{A}^\mathsf{T}+\mathbf{F}^\mathsf{T}\mathbf{B}_1^\mathsf{T} & \mathbf{B}_2 & \bar{\boldsymbol{\Theta}} \\ \mathbf{B}_2^\mathsf{T} & -\mathbf{I} & \mathbf{0} \\ \bar{\boldsymbol{\Theta}}^\mathsf{T} & \mathbf{0} & \bar{\boldsymbol{\Xi}} \end{bmatrix} < 0, \quad (20)$$

where the $\lambda$ is absorbed into the definitions of $\mathbf{R}$ and $\mathbf{F}$ (i.e., $\mathbf{R} = \lambda\mathbf{R}$, $\mathbf{F} = \mathbf{K}\lambda\mathbf{R}$) for consistent notation, $\bar{\boldsymbol{\Xi}} = \mathrm{diag}(-\frac{1}{\gamma_1^2}\mathbf{I}, \ldots, -\frac{1}{\gamma_{n_w}^2}\mathbf{I})$, and $\bar{\boldsymbol{\Theta}} = \left[\tilde{\boldsymbol{\Phi}}_1, \ldots, \tilde{\boldsymbol{\Phi}}_{n_w}\right]$.

Because of the change of variables introduced above, we need to suitably modify the constraint $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le \tau$. To this end, we state our first convex reformulation of (8) next.

*Lemma 2:* Let $\mathbf{W} > 0$ be given and $\mathbf{K} = \mathbf{F}\mathbf{R}^{-1}$. Then, $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le \sqrt{\beta}$ if there exists $\beta > 0$ such that

$$\begin{bmatrix} \beta\mathbf{I} & \mathbf{F} \\ \mathbf{F}^\mathsf{T} & (\mathbf{W}^{-1}\mathbf{R})^\mathsf{T} + (\mathbf{W}^{-1}\mathbf{R}) - \mathbf{I} \end{bmatrix} \ge 0. \quad (21)$$

*Proof:* Performing a congruence transformation with $\mathrm{diag}(\mathbf{I}, \mathbf{R})$ on (8) results in

$$\begin{bmatrix} \tau^2\mathbf{I} & \mathbf{F} \\ \mathbf{F}^\mathsf{T} & (\mathbf{W}^{-1}\mathbf{R})^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}) \end{bmatrix} \ge 0. \quad (22)$$

Using Lemma 1 with $\mathbf{S} = \mathbf{X} = \mathbf{I}$ and $\mathbf{Y} = \mathbf{W}^{-1}\mathbf{R}$, we relax the bilinear term $(\mathbf{W}^{-1}\mathbf{R})^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R})$ as

$$(\mathbf{W}^{-1}\mathbf{R})^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}) \ge (\mathbf{W}^{-1}\mathbf{R})^\mathsf{T} + (\mathbf{W}^{-1}\mathbf{R}) - \mathbf{I}. \quad (23)$$

Therefore, (22) is implied by (21), which ensures that $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le \tau$. Defining $\beta = \tau^2$ completes the proof. ∎

Alternatively, we can find a different relaxation of the bilinear term $(\mathbf{W}^{-1}\mathbf{R})^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R})$ for a given $\mathbf{R}_0$. This is analogous to linearizing the bilinear term about $\mathbf{R}_0$, which is similar to the convex overbounding approach in [16].

*Lemma 3:* Let $\mathbf{W} > 0$, $\mathbf{R}_0 > 0$ be given and $\mathbf{K} = \mathbf{F}\mathbf{R}^{-1}$. Then, $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le \sqrt{\beta}$ if there exists $\beta > 0$ such that

$$\begin{bmatrix} \beta\mathbf{I} & \mathbf{F} \\ \mathbf{F}^\mathsf{T} & \mathbf{T}_1 \end{bmatrix} \ge 0, \quad (24)$$

where $\mathbf{T}_1 = (\mathbf{W}^{-1}\mathbf{R})^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}_0) + (\mathbf{W}^{-1}\mathbf{R}_0)^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}) - (\mathbf{W}^{-1}\mathbf{R}_0)^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}_0)$.

*Proof:* We have already established that $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le \tau \iff$ (22). Then, using (6) with $\mathbf{H} = \mathbf{W}^{-1}\mathbf{R}$ and $\mathbf{H}_0 = \mathbf{W}^{-1}\mathbf{R}_0$ leads to

$$(\mathbf{W}^{-1}\mathbf{R})^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}) \ge (\mathbf{W}^{-1}\mathbf{R})^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}_0)$$
$$ + (\mathbf{W}^{-1}\mathbf{R}_0)^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}) - (\mathbf{W}^{-1}\mathbf{R}_0)^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}_0). \quad (25)$$

Therefore, (22) is implied by (24), which ensures that $\bar{\sigma}(\mathbf{K}\mathbf{W}) \le \tau$. Denoting $\beta = \tau^2$ completes the proof. ∎

Finally, to express (24) in terms of the variables $\mathbf{K}$ and $\mathbf{P}$, instead of $\mathbf{F}$ and $\mathbf{R}$, we perform a congruence transformation with $\mathrm{diag}(\mathbf{I}, \mathbf{P})$ on (24) to obtain

$$\begin{bmatrix} \beta\mathbf{I} & \mathbf{K} \\ \mathbf{K}^\mathsf{T} & \mathbf{T}_2 \end{bmatrix} \ge 0, \quad (26)$$

where $\mathbf{T}_2 = \mathbf{W}^{-1}\mathbf{W}^{-1}\mathbf{R}_0\mathbf{P} + \mathbf{P}(\mathbf{W}^{-1}\mathbf{R}_0)^\mathsf{T}\mathbf{W}^{-1} - \mathbf{P}(\mathbf{W}^{-1}\mathbf{R}_0)^\mathsf{T}(\mathbf{W}^{-1}\mathbf{R}_0)\mathbf{P}$. The Schur complement is then

---

**Algorithm 1:** Controller Synthesis Algorithm

1 Initialization: Choose $\mathbf{W}_0, \mathbf{r}, n_{\max}$.
2 **for** $j = 1 : n_r$ **do**
3      Set $r = r_j$ where $r_j$ is the $j$-th entry of $\mathbf{r}$ and $\mathbf{W} = \mathbf{W}_0$. Set $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W})$ and $\mathbb{U}_c = r\mathcal{B}_{n_u}$.
4      **repeat**
5          Get the sampled data points corresponding to $\mathbb{X}_c \times \mathbb{U}_c$. Use those to compute $\gamma_i, i \in \mathcal{N}_{n_w}$.
6          Get $\beta^\star, \mathbf{R}^\star, \mathbf{F}^\star$ from (28). Set $\mathbf{K} = \mathbf{F}^\star(\mathbf{R}^\star)^{-1}$, $\mathbf{R}_0 = \mathbf{R}^\star, c_t = 1$.
7          **while** $c_t \leq n_{\max}$ & $\bar{\sigma}(\mathbf{KW}) \geq r$ **do**
8              Get $\beta^\star, \mathbf{P}^\star, \{\lambda_i^\star\}$ from (29) using $\mathbf{R}_0, \mathbf{K}$. Set $\mathbf{R}_0 = (\mathbf{P}^\star)^{-1}$.
9              Get $\beta^\star, \mathbf{R}^\star, \mathbf{F}^\star$ from (30) using $\mathbf{R}_0, \{\lambda_i^\star\}$.
10             Set $\mathbf{K} = \mathbf{F}^\star(\mathbf{R}^\star)^{-1}, \mathbf{R}_0 = \mathbf{R}^\star, c_t = c_t + 1$.
11          **end**
12          **if** $\bar{\sigma}(\mathbf{KW}) \leq r$ **then**
13             Update $\mathbf{W}$ to get a larger $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W}) \subseteq \mathbb{X}$.
14          **else**
15             Update $\mathbf{W}$ to get a smaller $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W}) \subseteq \mathbb{X}$.
16          **end**
17      **until** The largest $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W})$ is certified;
18      Output: $\mathbf{K}$ and $\mathbf{W}$.
19 **end**

---

applied to (26) to yield

$$\begin{bmatrix} \beta\mathbf{I} & \mathbf{K} & \mathbf{0} \\ \mathbf{K}^\mathsf{T} & \mathbf{T}_3 & \mathbf{P} \\ \mathbf{0} & \mathbf{P} & \mathbf{R}_0^{-1}\mathbf{WWR}_0^{-1} \end{bmatrix} \geq 0, \qquad (27)$$

where $\mathbf{T}_3 = \mathbf{W}^{-1}\mathbf{W}^{-1}\mathbf{R}_0\mathbf{P} + \mathbf{P}(\mathbf{W}^{-1}\mathbf{R}_0)^\mathsf{T}\mathbf{W}^{-1}$. Finally, we are ready to state the convex optimization problems (namely, SDPs) that are involved in the iterative controller synthesis. The SDPs, along with the respective solutions, are given by

$$(\beta^\star, \mathbf{R}^\star, \mathbf{F}^\star) = \arg\min_{\beta,\mathbf{R},\mathbf{F}}\{\beta \mid (20), (21), \mathbf{R} > 0, \beta > 0\}, \quad (28)$$

$$(\beta^\star, \mathbf{P}^\star, \{\lambda_i^\star\}) = \arg\min_{\beta,\mathbf{P},\{\lambda_i\}}\{\beta \mid (7), (27), \mathbf{P} > 0, \beta > 0,$$
$$\lambda_i > 0, i \in \mathcal{N}_{n_w}\}, \qquad (29)$$

$$(\beta^\star, \mathbf{R}^\star, \mathbf{F}^\star) = \arg\min_{\beta,\mathbf{R},\mathbf{F}}\{\beta \mid (19), (24), \mathbf{R} > 0, \beta > 0\}. \quad (30)$$

Note that the SDP in (29) requires known values of $\mathbf{K}$ and $\mathbf{R}_0$, similar to the SDP in (30) which requires the values of $\lambda_i$ and $\mathbf{R}_0$ to be known. Thus, the SDP in (28) can be initially solved to obtain $\beta^\star, \mathbf{R}^\star, \mathbf{F}^\star$ and set $\mathbf{K} = \mathbf{F}^\star(\mathbf{R}^\star)^{-1}$, $\mathbf{R}_0 = \mathbf{R}^\star$. The values of $\mathbf{K}$ and $\mathbf{R}_0$ can then be utilized in solving the SDP in (29) to get $\beta^\star, \mathbf{P}^\star, \{\lambda_i^\star\}$. Now, the tuple $(\mathbf{P}, \mathbf{K}, \tau, \{\lambda_i\})$ with $\mathbf{P} = \mathbf{P}^\star$, $\tau = \sqrt{\beta^\star}$, $\{\lambda_i\} = \{\lambda_i^\star\}$ satisfies (7), (8). However, (9) might not hold and our approach involves iterating between (29) and (30) to satisfy $\bar{\sigma}(\mathbf{KW}) \leq r$, if that is possible without modifying $\mathbf{W}$. The control synthesis starts from a small ellipsoid $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W})$ for a given $r > 0$. Iterations are then carried out to certify the largest possible ellipsoid $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W})$ for that $r$, while simultaneously satisfying $\bar{\sigma}(\mathbf{KW}) \leq r$. The overall procedure for control synthesis is summarized in Algorithm 1. Given the sampling region $\mathbb{X} \times \mathbb{U}$, Algorithm 1 should be initialized by choosing a $\mathbf{W} = \mathbf{W}_0 > 0$ such that $\mathbb{X}_c = \mathcal{E}_{n_x}(\mathbf{W}) \subseteq \mathbb{X}$ is sufficiently small. Also as a part of the initialization, $r_m = \max_{r>0}\{r \mid \mathbb{U}_c = r\mathcal{B}_{n_u} \subseteq \mathbb{U}\}$ should be determined to specify an $n_r$-point grid $\mathbf{r} = (r_0, \ldots, r_m)$ where $r_0 > 0$
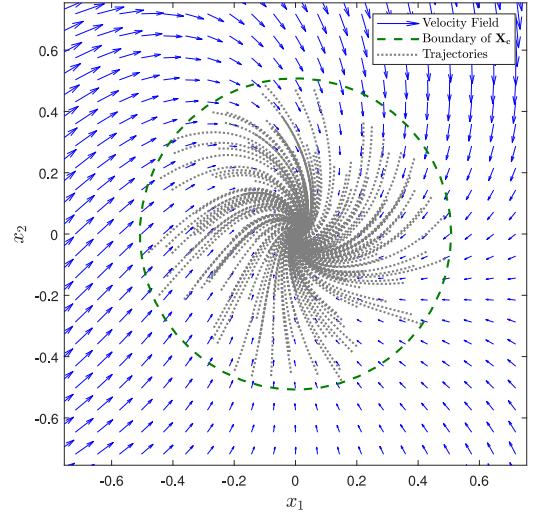


Fig. 1. Simulation results of the closed-loop system (32) with $r = 0.5$ and $\mathbf{K} = [-0.7151 \quad -0.6762]$.

is chosen to be small. Finally, a maximum iteration number $n_{\max}$ should be chosen when implementing Algorithm 1. Note that a variation of the Algorithm 1 can be obtained where only (28) is utilized (i.e., without the iteration between (29) and (30)). However, in our experience, this generally leads to more conservative results.

## IV. NUMERICAL EXAMPLES

Two numerical examples are included in this section, concerning two-dimensional single-input systems. The nonlinearity explicitly depends on control inputs in the first example. The second example considers a system where the nonlinearity is a function of the states only. We choose $\mathbf{W} = \alpha\mathbf{I}, \alpha > 0$, meaning $\mathbb{X}_c = \mathcal{E}_2(\mathbf{W})$ is a circle of radius $\alpha$ in these examples. Also, the SDPs in (28)-(30) are solved in MATLAB using YALMIP [17] and MOSEK [18].

*Example-1:* Consider a nonlinear system of the form

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} -0.1x_1 + x_2 + u - x_1x_2 + u^2 \\ -0.1x_2 + u + x_1^2 - u^2 \end{bmatrix}, \qquad (31)$$

with the corresponding equilibrium $(\mathbf{x}_0, u_0) = (\mathbf{0}, 0)$. The nonlinear system in (31) is cast in the form of (2) with

$$\mathbf{A} = \begin{bmatrix} -0.1 & 1 \\ 0 & -0.1 \end{bmatrix}, \ \mathbf{B}_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix},$$
$$\mathbf{\Delta}(\delta\mathbf{x}, \delta u) = \begin{bmatrix} -\delta x_1\delta x_2 + \delta u^2 \\ \delta x_1^2 - \delta u^2 \end{bmatrix} = \begin{bmatrix} \Delta_1(\delta\mathbf{x}, \delta u) \\ \Delta_2(\delta\mathbf{x}, \delta u) \end{bmatrix}.$$

The system is then expressed in closed-loop form as

$$\delta\dot{\mathbf{x}} = (\mathbf{A} + \mathbf{B}_1\mathbf{K})\delta\mathbf{x} + \mathbf{w}, \ \mathbf{w} = [\Delta_1 \ \Delta_2]^\mathsf{T},$$
$$\mathbf{v}_1 = ([\mathbf{1}_{3_1} \ \mathbf{1}_{3_2}] + \mathbf{1}_{3_3}\mathbf{K})\delta\mathbf{x}, \mathbf{v}_2 = ([\mathbf{1}_{3_1} \ \mathbf{0}] + \mathbf{1}_{3_3}\mathbf{K})\delta\mathbf{x}. \ (32)$$

Now, Algorithm 1 is implemented with 11 values of $r$ between 0.01 and 0.5. The largest radius certified is $\alpha = 0.508$, which corresponds to $r = 0.5$ and $\mathbf{K} = \begin{bmatrix} -0.7151 & -0.6762 \end{bmatrix}$. The simulation results of the closed-loop system with this controller are shown in the form of a phase portrait plot in Fig. 1. Closed-loop trajectories starting from different initial conditions in the set $\mathbb{X}_c$ converge to the origin, illustrating asymptotic convergence in the certified region. The velocity
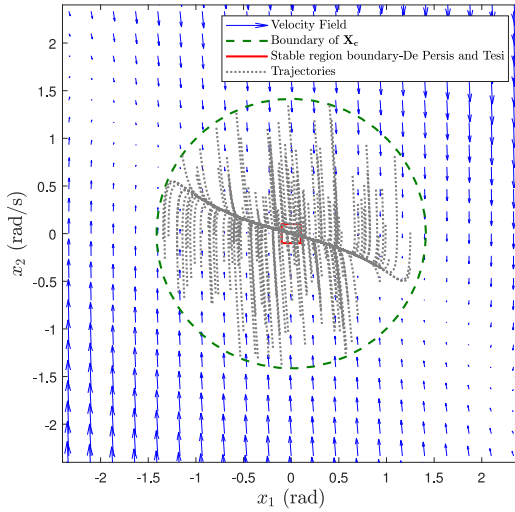
Fig. 2. Simulation results of the closed-loop system (33) with $\mathbf{K} = [-13.4283 \quad -13.5242]$.

field indicates that the largest set $\mathbb{X}_c$ certified is contained within an even larger asymptotically stable region. This is likely due to the local norm bounds holding true for this larger region. In summary, this example demonstrates that the proposed method is able to certify local asymptotic stability of the nonlinear system with state feedback, using only sampling and no explicit knowledge of the system's nonlinearities.

*Example-2:* Consider the inverted pendulum example in [13, Sec. V-B] in continuous time with the same unstable equilibrium at $(\mathbf{x}_0, u_0) = (\mathbf{0}, 0)$, which corresponds to the pendulum in upright position. The governing system is expressed in the closed-loop form as

$$\delta\dot{\mathbf{x}} = (\begin{bmatrix} 0 & 1 \\ g & -\mu \end{bmatrix} + \mathbf{1}_{2_2}\mathbf{K})\delta\mathbf{x} + \begin{bmatrix} 0 \\ g(\sin(\delta x_1) - \delta x_1) \end{bmatrix}, \quad (33)$$

where $m = l = 1$, $g = 9.8$, and $\mu = 0.01$ (as in [13]). In this example, $\mathbf{B}_2 = \mathbf{1}_{2_2}$, $\mathbf{C}_1 = \begin{bmatrix} \mathbf{1}_{3_1} & \mathbf{0} \end{bmatrix}$ and $\mathbf{D}_1 = \mathbf{0}$. Since the nonlinearity is independent of $\delta u$, we do not necessarily need to constrain the control input. However, performing the iterations in Algorithm 1 to reduce $\bar{\sigma}(\mathbf{KW})$ in turn reduces the control effort required, and we let these iterations continue for $n_{\max} = 20$. In this setup, the maximum certified radius is $\alpha = \sqrt{2}$, along with the controller $\mathbf{K} = \begin{bmatrix} -13.4283 & -13.5242 \end{bmatrix}$. This controller is therefore able to drive the pendulum to its upright position from an initial displacement of approximately 81 degrees. The simulation results of the system (33) with this controller are depicted in Fig. 2 where, similar to Example-1, the vector field indicates that the closed-loop system can be driven to the equilibrium from a much larger region than the certified region $\mathbb{X}_c$. Indeed, there appears to be a stable manifold with the vector field converging to it (see Fig. 2). Also in Fig. 2, the red square denotes the local region certified in [13]. In comparison, the proposed controller is able to certify a much larger region. This improvement was achieved, in part, by utilizing the knowledge of the nominal LTI system whereas, the controller in [13] is purely data-driven. This example thus demonstrates the efficacy of the proposed method over a purely data-driven framework, given the nominal LTI system is known.

## V. CONCLUSION AND FUTURE WORK

We presented an iterative method of local stabilization for nonlinear systems using sampled I/O data. Our approach uses I/O data to derive local norm bounds and synthesizes a robust state-feedback controller that is guaranteed to stabilize the system within the sampling region for the set of nonlinearities satisfying the norm bounds. The iterative steps require solving SDPs which can be done efficiently using freely available solvers. One of the numerical examples highlighted the reduced conservatism in our proposed synthesis method compared to a purely data-driven approach. Our future efforts will involve introducing parametric uncertainties and exogenous signals into the proposed framework. Also, we will investigate other ways to characterize I/O behavior of the nonlinearities (e.g., weighted 2-norm bounds) and extend our formulation to the output-feedback case.

## REFERENCES

[1] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[2] J. Buch and P. Seiler, "Finite horizon robust synthesis using integral quadratic constraints," *Int. J. Robust Nonlinear Control*, vol. 31, no. 8, pp. 3011–3035, 2021.

[3] K.-C. Goh, M. G. Safonov, and J. H. Ly, "Robust synthesis via bilinear matrix inequalities," *Int. J. Robust Nonlinear Control*, vol. 6, nos. 9–10, pp. 1079–1095, 1996.

[4] G. E. Dullerud and F. Paganini, *A Course in Robust Control Theory: A Convex Approach*. New York, NY, USA: Springer, 2000.

[5] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints," *IEEE Trans. Autom. Control*, vol. 42, no. 6, pp. 819–830, Jun. 1997.

[6] A. Packard and J. Doyle, "The complex structured singular value," *Automatica*, vol. 29, no. 1, pp. 71–109, 1993.

[7] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Inf. Sci.*, vol. 235, pp. 3–35, Jun. 2013.

[8] A. Romer, J. Berberich, J. Köhler, and F. Allgöwer, "One-shot verification of Dissipativity properties from input–output data," *IEEE Contr. Syst. Lett.*, vol. 3, pp. 709–714, 2019.

[9] M. Tanemura and S.-I. Azuma, "Efficient data-driven estimation of passivity properties," *IEEE Contr. Syst. Lett.*, vol. 3, pp. 398–403, 2019.

[10] J. Berberich, A. Koch, C. W. Scherer, and F. Allgöwer, "Robust data-driven state-feedback design," in *Proc. Amer. Control Conf.*, 2020, pp. 1532–1538.

[11] J. Na, J. Zhao, G. Gao, and Z. Li, "Output-feedback robust control of uncertain systems via online data-driven learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2650–2662, Jun. 2021.

[12] D. Wang, D. Liu, and H. Li, "Policy iteration algorithm for online design of robust control for a class of continuous-time nonlinear systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 2, pp. 627–632, Apr. 2014.

[13] C. De Persis and P. Tesi, "Formulas for data-driven control: Stabilization, optimality, and robustness," *IEEE Trans. Autom. Control*, vol. 65, no. 3, pp. 909–924, Mar. 2020.

[14] M. Tanaskovic, L. Fagiano, C. Novara, and M. Morari, "Data-driven control of nonlinear systems: An on-line direct approach," *Automatica*, vol. 75, pp. 1–10, Jan. 2017.

[15] R. J. Caverly and J. R. Forbes, "LMI properties and applications in systems, stability, and control theory," 2019, *arXiv:1903.08599*.

[16] E. C. Warner and J. T. Scruggs, "Iterative convex overbounding algorithms for BMI optimization problems," *IFAC PapersOnLine*, vol. 50, no. 1, pp. 10449–10455, 2017.

[17] J. Lofberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Conf. Robot.*, 2004, pp. 284–289.

[18] *The MOSEK Optimization Toolbox for MATLAB Manual. Version 9.0.*, MOSEK ApS, Copenhagen, Denmark, 2019.