Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

ARTICLE TYPE

Set-Based Gain-Scheduled Control via Quasi-Convex Difference Inclusions

Antonio Sala[1] | Carlos Ariño[2] | Ruben Robles[3]

[1]Instituto de Automática e Informática Industrial
Universitat Politècnica de València, Cno. Vera s/n, 46022 Valencia, Spain
[2] Department of Industrial Systems Engineering and Design,
Universitat Jaume I, Av. Vicent Sos Baynat, s/n. 12071 Castelló de la Plana, Spain
[3]Universidad Tecmilenio,
Campus Las Torres, Paseo del Acueducto 2610, 64909 Monterrey, N.L., Mexico

Correspondence
*Antonio Sala. Email: asala@isa.upv.es

Summary

A nonlinear system with sector-bounded nonlinearities may be expressed as a quasi-LPV system (convex combination of linear models), being this a well-known fact. The convex difference inclusion (CDI) modelling framework proposed by M. Fiacchini and coworkers in several of their works generalises the quasi-LPV modelling procedure and proposes robust controllers enlarging polytopic domain of attraction estimates. This works further generalises the CDI approach to a gain-scheduled case including, also, some quasi-convex cases. Controller design is based on convexity properties of two set valued maps describing (with some uncertainty) the state evolution and the state-dependent set where scheduling variables take values. As most set-based approaches, the proposal is tractable in low-dimensional cases. The presented results encompass prior quasi-LPV and CDI models as particular cases.

KEYWORDS:
Convex difference inclusion; invariant sets; quasi-LPV systems; gain-scheduling; robust control; polyhedral Lyapunov function; set-based control

1 | INTRODUCTION

Determining invariant and contractive sets for dynamic systems is a control problem which has been addressed in a variety of settings; indeed, invariance and contractiveness are close cousins of stability definitions [1].

Linear parameter-varying models (LPV) are a widely-used modelling paradigm; in this work, for sake of comparison with later improvements, we will consider discrete-time LPV systems $x_+ = A(h)x+B(h)u+E(h)d$, $h \in \mathcal{H}$, whose state evolves in the convex hull of a finite set of linear "vertex models" ($A(h) \in Co\{A_1, \dots, A_r\}$ for all $h \in \mathcal{H}$), also referred to as polytopic *linear difference inclusions* (LDI) [2]. Determining contractive sets for LPV systems is a well-studied topic, and it can be approached either from set-based computations [1,3,4,5,6] or via convex optimisation (linear matrix inequalities, LMI) [7,8,9,10,11].

Smooth nonlinear systems can be easily embedded in a polytopic LDI, giving rise to quasi-LPV models [12,13,14,15,16]; thus, LPV results can be applied to prove stability in some nonlinear control problems, with, of course, a dose of conservatism [9]; actually, the quasi-LPV model of a nonlinear system is not unique, so the best one might depend on the required performance objectives [16].

A broader class of models is that of non-linear parameter varying models (NLPV), $x_+ = f(x, d, h, u)$; however, as such, they are too general to be useful. There is a well-studied particular case when f is a polynomial, giving rise to Sum-of-Squares (SoS) convex optimisation approaches to nonlinear control [17]. Being quasi-LPV models a particular case (degree 1), a Taylor series argumentation can be used to embed a smooth nonlinear system in the convex hull of a finite set of *polynomial* vertex models [18], so SoS techniques can be easily generalised to such polynomial-NLPV case; Positivstellensatz argumentations generalise S-procedure to get convex necessary conditions for local positiveness (or Lyapunov decrescence). There are still some inherent sources of conservatism[9], however. Importantly, from a modelling perspective, as Taylor-series bounds wildly diverge, they might give worse results than LPV counterparts for large modelling regions.

This work will pursue extending the above-mentioned non-LMI set-based approaches, to a class of NLPV which will just require some convexity (or quasi-convexity) conditions on the vertex models: intentionally, neither LMI nor SOS approaches will be pursued any further in this work.

Seminal prior works in this convexity-related line are[19,20,21,22]; they show that set-based LDI ideas can be generalised to the so-called convex differential inclusions (CDI), which are more general models capable of representing nonlinear and uncertain systems with lower conservatism (overbounding) than LDI, being these a particular case. Furthermore, they present iterative scaling/shooting algorithms to compute contractive sets, both in analysis and *robust* control design settings. In the polytopic case, such steps can be carried out with polytope manipulation software such as the multiparametric toolbox (MPT3[6]). The cited works, however, do not discuss the gain-scheduling problem we will address here. There are simplified representations of polyhedral sets which may have computational advantages (at the expense of generality), such as interval-based[23] or zonotopes[24,25]; these descriptions are, nevertheless, also left out of the scope of the present work.

The goal of this work is generalising the CDI ideas in[19,20,21], in order to incorporate gain-scheduling options; as a corollary, this work will also extend the set-based quasi-LPV/LDI developments in[26,27]. The work[28] discusses a generalisation of some concepts in these referred works to a quasi-convex setup, and a preliminary conference work[29] by the authors sketches the ideas that will be fully developed in the present contribution.

The structure of this paper is as follows: next section presents preliminary definitions and problem statement; Section 3 reviews LPV and non-LPV modelling frameworks to which the presented results apply; Section 4 defines gain-scheduled 1-step sets and discusses conditions to prove that a given polytope is a subset of the gain-scheduled 1-step one; actual algorithms to approximately compute such sets as well as contractive set estimates are discussed in section 5; stabilisation via Minkowski polyhedral Lyapunov functions in undisturbed cases is considered in Section 6; Section 7 provides numerical examples, comparing with options in prior literature. Finally, a conclusion section closes the paper.

Notation.

Sets: In the sequel, Δ will denote the standard simplex $\Delta := \{h \in \mathbb{R}^n : \sum_{i=1}^n h_i = 1, h_i \geq 0, i = 1, \dots, n\}$. Given arbitrary sets Σ and Γ in some vector space and a scalar $\alpha \in \mathbb{R}$, $\alpha\Sigma$ will stand for the linear transformation (scaling) of set Σ. Also, $Co(\Sigma)$ will denote the convex hull, and $\Sigma \oplus \Gamma$ will denote the Minkowski sum $\Sigma \oplus \Gamma := \{z : \exists s \in \Sigma, g \in \Gamma \; s.t. \; z = s + g\}$. $\mathcal{P}(\Gamma)$ will denote the power set (set of all subsets) of Γ. Given a polytopic set Γ, $vert(\Gamma)$ will denote the set of its vertices. In the case of a set of matrices, denoted in boldface, $\mathbf{\Sigma} \subset \mathbb{R}^{n\times m}$, given two constant matrices $M \in \mathbb{R}^{q\times n}$,$N \in \mathbb{R}^{m\times s}$, we will define the product $M \cdot \mathbf{\Sigma} \cdot N$ as $M \cdot \mathbf{\Sigma} \cdot N := \{Q \subset \mathbb{R}^{q\times s} : \exists S \in \mathbf{\Sigma} \; s.t. \; Q = M \cdot S \cdot N\}$. Minkowski sum and products of sets of matrices with compatible dimensions will be also denoted with $\oplus$ and $\otimes$, and product notation $\cdot$ or $\otimes$ will be omitted if clear from the context.

Set-valued maps: A set valued map $\mathcal{F}$ with domain Σ and range Γ will be understood as an operator that maps an element of a vector space $x \in \Sigma$ to a *set* $\mathcal{F}(x) \subseteq \Gamma$. Equivalently, it is an ordinary mapping from Σ to the power set of Γ, i.e., $\mathcal{F} : \Sigma \mapsto \mathcal{P}(\Gamma)$. Conversely, ordinary functions $f : \Sigma \mapsto \Gamma$ can be understood as "deterministic" set-valued maps so the image of $x \in \Sigma$ is a set consisting of a single point $\{f(x)\}$, and "constant" set-valued maps, $\mathcal{F}(x) := \Gamma$ for all $x \in \Sigma$ can be understood, abusing the notation, as, plainly, a set Γ. The image of a set Ω under a set-valued map $\mathcal{F}$ will be understood as $\mathcal{F}(\Omega) := \cup_{\xi\in\Omega}\mathcal{F}(\xi)$.

2 | PRELIMINARIES

This paper will consider a discrete-time dynamic system:

$$x_+ = f(x, d, h, u) \tag{1}$$

where $x \in \mathbb{X} \subset \mathbb{R}^n$ is the state vector, lying inside a so-called *modelling region* $\mathbb{X}$, $x_+ \in \mathbb{R}^n$ will be denoted as successor state, $d \in \mathbb{D} \subset \mathbb{R}^v$ is a vector of *unmeasurable* time varying parameters or disturbances inside a disturbance region $\mathbb{D}$, $h \in \mathbb{R}^r$ is a vector of *scheduling parameters*, assumed computable in real-time operation from available measurements, and $u \in \mathbb{U} \subset \mathbb{R}^m$ is the so-called input vector, taking values in a feasible input region $\mathbb{U}$. In the sequel, sets $\mathbb{X}$, $\mathbb{U}$ will be assumed to be convex sets; for computational reasons, $\mathbb{U}$ will be assumed to be polytopic.

Trajectories of (1) are given by $x(t + 1) = f(x(t), d(t), h(t), u(t))$, for given initial conditions $x(0)$. Gain-scheduled state-feedback controllers in the form $u(t) = \bar{u}(x(t), h(t))$ will be considered in this work, as $h(t)$ was assumed to be measurable. The presented framework encompasses the cases where either $h(t)$ is a measurable exogenous signal in, say, a "pure" parameter-varying system, or $h(t) \equiv h(x(t))$ models known nonlinearities in the system (as in quasi-LPV models) or a combination of both $h(t, x(t))$.

As d is not measurable, and f may have a complex structure, in many works in literature, system (1) is *embedded* into an *uncertain* or *parameter-varying* system with a simple-enough structure to allow for various control design techniques. In particular, embedding (1) in the convex hull of a finite set of linear models motivates the well-known LPV approach to gain scheduling, see references in the introduction.

Elements of d may be understood as disturbances or, equivalently, as uncertainty: given x, u and h, the successor state $x_+ = f(x, d, h, u)$ can only be asserted to belong to the set $\cup_{d\in\mathbb{D}} f(x, d, h, u)$; note that the "shape" of such a set depends on the source state x, the scheduling variable h, and the control action u. The mathematical structure that maps a point (x, u, h) onto a set is a set-valued map (see notation at the end of the Introduction section).

Scheduling variable set-valued map. The first set-valued map to consider will be the set of values that parameters $h(t, x)$ can take for a given state x, i.e., in the sequel we will assume $h(t, x) \in \mathcal{H}(x)$, being $\mathcal{H} : \mathbb{X} \mapsto \mathcal{P}(\mathbb{R}^r)$ a known set-valued map. Actually, most LPV and quasi-LPV literature assumes $\mathcal{H}$ to be a constant map (i.e., a given, *fixed* set) such as a parameter box (hyperrectangle) or the standard simplex; a generic $\mathcal{H}(x)$ allows conveniently generalising to a state-dependent parameter set, if so wished.

In the sequel, for notational brevity, possible arguments $h(t, x)$ will be omitted because, as h is directly measurable by assumption, such arguments are actually irrelevant for theoretical developments.

Embeddings. At the core of quasi-LPV/NLPV techniques is the ability to "embed" a "complex" model (1) into a "simpler" one (linear, polynomial, convex...) amenable to some stability analysis or control design techniques. The following definitions will set up the meaning of such "embedding" in the context of this work.

Definition 1. Let us consider a pair of set valued maps, $\mathcal{F} : \Theta \mapsto \mathcal{P}(\mathbb{R}^n)$, $\mathcal{H} : \mathbb{X} \mapsto \mathcal{P}(\mathbb{R}^r)$, where $\Theta := \{(x, u, h) : x \in \mathbb{X}, u \in \mathbb{U}, h \in \mathcal{H}(x)\}$. The pair $(\mathcal{F}, \mathcal{H})$ will be said to be an embedding of the system (1) in modelling region $\mathbb{X}$ if, for each $(x, u, h) \in \Theta$:

$$\bigcup_{d\in\mathbb{D}} f(x, d, h, u) \subseteq \mathcal{F}(x, u, h) \tag{2}$$

Definition 2. We will say that $(\mathcal{F}', \mathcal{H}')$ is an embedding of $(\mathcal{F}, \mathcal{H})$ if $\mathcal{H}(x) \subseteq \mathcal{H}'(x)$ and $\mathcal{F}(x, u, h) \subseteq \mathcal{F}'(x, u, h)$ for all $(x, u, h) \in \Theta$.

Informally, we will understand Definition 1 as the fact that the model (1) can be replaced[1] by the parameter-dependent difference inclusion (PDDI), associated to an embedding of it, denoted by:

$$x_+ \in \mathcal{F}(x, u, h), \quad h \in \mathcal{H}(x) \tag{3}$$

perhaps introducing additional uncertainty due to stating *subsethood* and not *equality* in (2), in exchange of $\mathcal{F}$ being "simpler" in a certain sense, to be later discussed: indeed, the union at the left-hand side of (2) is a set-valued map which may not have, in general, any desirable mathematical property; the goal of the embedding process is finding some $\mathcal{F}$ which (conservatively) overbounds the said union but has a suitable linear/convex structure.

Example 1. The model $x_+ = sin\left(x + x^3 + d\right) + h \cdot u$, $x \in [-1, 1]$, $d \in [-0.1, 0.1]$, $h \in [0, |x|]$ can be embedded[2] into a PDDI (3) with a pair $(\mathcal{F}, \mathcal{H})$ given by

$$\mathcal{F}(x, u, h) := Co\{0.95x + h \cdot u, 1.17x + h \cdot u\} \oplus [-0.1, 0.1], \qquad \mathcal{H}(x) := Co\{0, |x|\}.$$

In this way, the disturbance-dependent nonlinearity $sin\left(x + x^3 + d\right)$ is replaced by the convex hull of two undisturbed linear models plus an interval (which might be interpreted as an additive disturbance, but conceptually different from the original d, which entered the model in a non-linear way). Further modelling examples will be discussed in Section 3.

Definition 3. Given an input sequence $(u_0, u_1, \dots)$, an admissible trajectory of the PDDI (3) will be understood as a sequence of states $\{x_0, x_1, \dots\}$ and scheduling variables $(h_0, h_1, \dots)$ such that $x_{k+1} \in \mathcal{F}(x_k, u_k, h_k)$ and $h_k \in \mathcal{H}(x_k)$ for all $k \geq 0$.

[1] Note that, in the same way as h, the disturbance parameters d may take values in a state-dependent set, say $d \in Co\{-0.1x, 0.1x\}$, and even the scheduling variables might take state-and-disturbance-dependent values, so there are more general versions of the sets in which the arguments of f in (1) may take values. They are omitted to avoid notational clutter, as the ensuing results would be just minor variations of the ones presented in this paper.

[2] In this simplistic example, bounds were manually obtained by inspection of the plots.

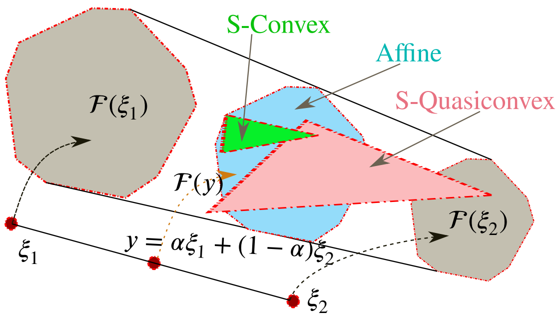

FIGURE 1 Graphical illustration of the concept of affine, S-convex and S-quasiconvex maps.

Convexity-related properties. The embeddings $(\mathcal{F}, \mathcal{H})$ in this paper will have, as an assumption in later results, some of the convexity-related properties below for the intervening set-valued maps $\mathcal{F}$ and $\mathcal{H}$.

Definition 4. Let us consider a set-valued map $\mathcal{F} : \mathcal{D} \mapsto \mathcal{P}(\mathbb{R}^n)$ with $\mathcal{D}$ assumed to be a convex set. The set-valued map $\mathcal{F}$ is :

- **convex-valued**, if $\mathcal{F}(\xi)$ is non-empty and convex for every $\xi \in \mathcal{D}$. Additionally, it will be denoted as **polytopic** if $\mathcal{F}(\xi)$ is a compact polytope for all ξ in $\mathcal{D}$.
- **affine**, if $\mathcal{F}(\alpha_1\xi_1 + \alpha_2\xi_2) = \alpha_1\mathcal{F}(\xi_1) \oplus \alpha_2\mathcal{F}(\xi_2)$, for all α_1, α_2 in $\mathbb{R}$; **linear** if, additionally, $\mathcal{F}(0) = \{0\}$.
- **S-convex**, if $\mathcal{F}(\alpha\xi_1 + (1-\alpha)\xi_2) \subseteq \alpha\mathcal{F}(\xi_1) \oplus (1-\alpha)\mathcal{F}(\xi_2)$ for all ξ_1, ξ_2 in $\mathcal{D}$ and for all $0 \leq \alpha \leq 1$.
- **S-quasiconvex**, if $\mathcal{F}(\alpha\xi_1 + (1-\alpha)\xi_2) \subseteq Co\left(\mathcal{F}(\xi_1) \cup \mathcal{F}(\xi_2)\right)$ for all ξ_1, ξ_2 in $\mathcal{D}$ and for all $0 \leq \alpha \leq 1$.

Evidently from the definitions, all affine set-valued maps are S-convex, and all S-convex ones are S-quasiconvex.

Figure 1 illustrates Definition 4 with a graphical interpretation: the gray sets depict $\mathcal{F}(\xi_1)$ and $\mathcal{F}(\xi_2)$; considering now an interpediate point $y := \alpha\xi_1 + (1-\alpha)\xi_2$, the blue set depicts $\alpha\mathcal{F}(x_1) \oplus (1-\alpha)\mathcal{F}(x_2)$ (the exact image of an affine map, in which $\mathcal{F}(\xi_1)$ would gradually "morph" to $\mathcal{F}(\xi_2)$ as α decreases from 1 to zero), the pink set depicts a possible $\mathcal{F}(y)$ of an S-quasiconvex map, and the green set depicts a possible $\mathcal{F}(y)$ of an S-convex map. For convex-valued maps, Definition 4 can be equivalently stated in terms of linearity, convexity or quasi-convexity of the support function of a non-empty set Ω, given by $\Phi_\Omega := \sup_{x\in\Omega} \eta^T x$, details omitted for brevity, see[21] for details.

The key aspect of S-quasiconvexity is that a bound of the image of a polytope can be easily computed from that of its vertices, as the following theorem states (the result is inspired in the S-convex case in[19]):

Theorem 1. Given a polytope $P \subset \mathcal{D}$, and an S-quasiconvex set-valued map $\mathcal{F} : \mathcal{D} \mapsto \mathcal{P}(\mathbb{R}^n)$, we have

$$\mathcal{F}(\xi) \subseteq Co\left(\bigcup_{\zeta\in vert(P)} \mathcal{F}(\zeta)\right) \quad \forall \xi \in P \tag{4}$$

Proof. Denote the vertices of P as $\{\zeta_1, \ldots, \zeta_r, \ldots, \zeta_{N_P}\}$, and denote the convex hull of their images (right-hand side of (4) above) as Ξ . Every point $\xi \in P$ can be expressed as a convex combination $\xi = \sum_{i=1}^{N_p} \eta_i(\xi)\zeta_i$, with $\eta_i \in \Delta$. Let us assume that r of the coefficients η_i are non-zero; we will denote such situation by saying that ξ can be expressed as a non-zero convex combination of r vertices. Now, let us assume that the non-zero convex combination of $r-1$ vertices fulfills the assertion in the theorem statement. Any point $\xi \in P$ which is not a vertex and is a non-zero convex combination of r vertices can be expressed, for some $\eta_i(x) > 0$, $i = 1, \ldots, r$ as $\xi = \sum_{i=1}^{r} \eta_i(\xi)\zeta_i = \eta_r\zeta_r + (1-\eta_r)\hat{z}_{r-1}$ being $\eta_r \neq 1$ and

$$\hat{z}_{r-1} := \left(\sum_{i=1}^{r-1} \frac{\eta_i(x)}{1-\eta_r(\xi)}\zeta_i\right) \in P \tag{5}$$

i.e., as the non-zero convex combination of a vertex of P and a point $\hat{z}_{r-1}$ which is a non-zero convex combination of $r-1$ vertices. By the induction assumption, $\mathcal{F}(\hat{z}_{r-1}) \subseteq \Xi$; thus, as $\mathcal{F}(\zeta_r) \subseteq \Xi$, Definition 4 entails $\mathcal{F}(\xi) \subseteq \Xi$. As the theorem is true for combinations of 2 vertices ($r = 2$), and so it is trivially for $r = 1$, it is true for any integer r. □

Problem statement

A plethora of results exist for quasi-LPV embeddings of nonlinear models proving stability, finding inescapable sets, gain-scheduled state-feedback controller design, etc. using either set-manipulation software [4,6,27] via the polytopic representation (7), or convex linear matrix inequalities (LMI), see for instance [7,8,10,11]. However, generic LMI approaches do not easily apply to more general [3] non-LPV models, or to non-symmetric constraints, so the scope of this paper will be directed to the set-based approach.

Basically, the objective of this work is extending to the gain-scheduled case the ideas in [20] and related works [19,21,32]. Indeed, the cited works discuss how to embed nonlinear systems $x_+ = f(x, d, u)$ into a single set-valued map $x_+ \in \mathcal{F}(x, u)$; later on, they propose non-scheduled (robust) control of the resulting uncertain dynamic system, using a set-based approach, in the case $\mathcal{F}$ is S-convex.

This work pursues generalising the above-cited results to a gain-scheduled case, using the pair $(\mathcal{F}, \mathcal{H})$, by exploiting S-convexity or S-quasiconvexity properties of $\mathcal{F}$ or $\mathcal{H}$, to design gain-scheduled controllers $u = \bar{u}(x, h)$. As in the cited literature, the developments in this work will use a set-based approach to control, seeking to compute the so-called invariant and contractive sets [1]: the class of models used in set-based gain-scheduled LPV developments [27] can be considered a particular case of the ones considered here. Of course, jumping to nonlinear (even non-polynomial) vertex models comes at the expense of compuational cost: efficient LMI solvers cannot apply to the class of models in consideration. The advantages of the set-based approach in this work (and the above-cited cited ones) are its applicability to S-convex (but not necessarily linear) set-valued maps, and the ability to naturally consider saturation and possible non-symmetric constraints in state and input. The disadvantage comes from convex hull computations which make it impractical for high-dimensional systems in a general case.

3 | MODELLING

Let us now present some examples of particular cases of the above embeddings, illustrating its generality and motivating the usefulness of later results. Due to their popularity, motivating LPV examples are presented first, even if well known, for the sake of completeness. Actual non-LPV embeddings, which are the main objective of this work, will come immediately afterwards. Nevertheless, as disscussed at several places in this section, LPV and S-convex modelling issues are comprehensively dealt with in other literature references, to which the interested reader is referred.

3.1 | LPV embeddings

Uncertain LPV models. LPV modelling is a well-studied field nowadays, so the reader is referred to, for instance, [13,16,12,9,15] for ample detail. In order to understand how to express such models in the notation required in this work, consider an LPV model with disturbances $\eta \in \mathbb{D}$ and uncertain model matrices:

$$x_+ = \textstyle\sum_{i=1}^{r} h_i((A_i + M_i\Gamma N_i)x + E_i\eta) + Bu \tag{6}$$

with $h \in \Delta$, being Δ the $r-1$-dimensional standard simplex, and being Γ an unknown time-varying matrix such that $\Gamma \in \mathbf{\Gamma}$, being $\mathbf{\Gamma} := Co(\{\Gamma_1, \dots, \Gamma_s\})$ a polytope of matrices. Now, considering the set-valued map:

$$\mathcal{F}(x, u, h) := \bigoplus_{i=1}^{r} h_i((A_i \oplus M_i\mathbf{\Gamma} N_i)x \oplus E_i\mathbb{D}) \oplus Bu \tag{7}$$

we have an embedding of (6) in the form (3), considering d in (1) to be $d := (\eta, \Gamma)$, –i.e., d contains both the uncertain input and the uncertain model matrix–, and $\mathcal{H}$ being the constant map $\mathcal{H}(x) := \Delta$. Following Definition 4, the map $\mathcal{F}$ in (7) is affine in arguments (x, u) for fixed h, and affine in (h, u) for fixed x.

Quasi-LPV models of nonlinear systems. The quasi-LPV models are a well-known particular case of the embeddings in Definition 1. Let us illustrate the idea with a straightforward example, to be also used in later results.

[3] As discussed in the introduction, another way of obtaining more general models is using polynomial bounds and using Sum-of-squares convex programming techniques for them [17,30,31]. The sum-of-squares generalisation of LMIs, however, will not be considered in the scope of this paper, either.

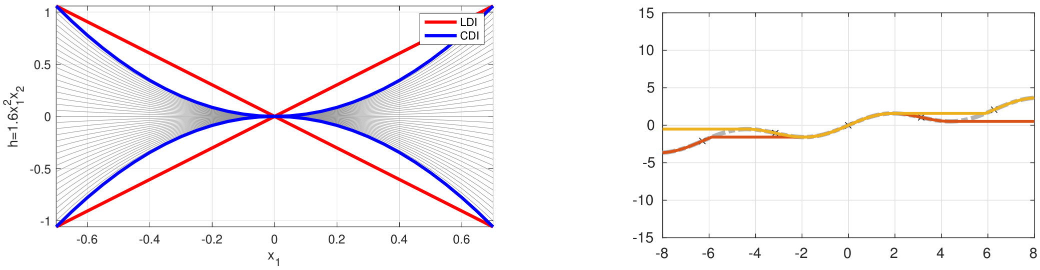

FIGURE 2 –*Left:* quasi-LPV (LDI) bound of $h = 1.6x_1^2x_2$ (red), *vs.* S-convex bound of the same function (blue); the representation depicts a projection on the plane (x_1, h).
–*Right:* a quasiconvex(brown)–quasiconcave(red) bound inducing a S-quasiconvex set-valued map, embedding the nonlinearity $f(x) = \sin(x) + x/3$ (grey, dashed).

Example 2. Consider an undisturbed nonlinear system $x_+ = f(x, u)$ so:

$$\begin{aligned} x_{1+} &= x_1 + 0.6x_2 - 1.6x_2x_1^2 + 1.5u \\ x_{2+} &= -1.2x_1 + 0.4x_2 + 0.4u \end{aligned} \tag{8}$$

Trivially, the above expression (8) can be equivalently rewritten as:

$$f(x, u) \in \mathcal{F}(x, u, h) := \begin{pmatrix} x_1 + 0.6x_2 - 1.6h + 1.5u \\ -1.2x_1 + 0.4x_2 + 0.4u \end{pmatrix} \tag{9}$$

with $h \equiv x_2x_1^2$, so $\mathcal{F}$ is a *deterministic* set-valued map and, in order to express the above as in Definition 1, we just need to set up a second deterministic set-valued map $\mathcal{H}$ for the scheduling variables and assert $h \in \mathcal{H}_{nonlin}(x) := \{x_2x_1^2\}$. Evidently, as $\mathcal{H}_{nonlin}(x)$ is a single point and so it is $\mathcal{F}$, no "uncertainty" in the embedding $(\mathcal{F}, \mathcal{H})$ is present at this stage, i.e., (9) is an apparently linear expression that, once the explicit "shape" of h is plugged into it, renders equivalent to (8).

Considering now a modelling region $\mathbb{X} := \{(x_1, x_2) : |x_1| \leq a, |x_2| \leq b\}$ we can bound the nonlinearity $h = x_2x_1^2$ by (conservatively) asserting that $h \in \mathcal{H}_{LPV}(x) := Co(\{-ab\cdot x_1, ab\cdot x_1\})$, i.e., assuming h is bounded by two linear vertex models; uncertainty is now present as $\mathcal{H}_{LPV}(x)$ is an interval for $x_1 \neq 0$. As $\mathcal{H}_{nonlin}(x) \subseteq \mathcal{H}_{LPV}(x)$, then $(\mathcal{F}, \mathcal{H}_{LPV})$ is an embedding of $(\mathcal{F}, \mathcal{H}_{nonlin})$ in the sense of Definition 2, henceforth, it is an embedding of (8). Expressing a nonlinearity as a point in the convex hull of linear functions of the state, $\mathcal{H}_{LPV}(x)$, actually amounts to the well-known quasi-LPV modelling, thus, our proposal includes it as a particular case. Actually, expressing (9) as (6) is straightforward, left to the reader.

3.2 | NLPV embedding examples

Even if quasi-LPV cases have been presented in first place, the modelling setup with a pair of set-valued maps $(\mathcal{F}, \mathcal{H})$ discussed in Section 2 is more general; it is actually inspired in the non-scheduled cases in [19,21].

Trivially, a function $f : \mathbb{R}^n \mapsto \mathbb{R}$ can be embedded into a set-valued map $\mathcal{F}$, i.e., $f(x) \in \mathcal{F}(x)$, if two bounding functions f_-, f_+ can be found such that $f_-(x) \leq f(x) \leq f_+(x)$; indeed, $\mathcal{F}$ is the interval $\mathcal{F}(x) := [f_-(x), f_+(x)]$.

Now, given such bounding, if f_- is concave (quasiconcave) and f_+ is convex (quasiconvex), then $\mathcal{F}(x)$ is a S-convex (S-quasiconvex) map, see [19,28] for details. Figure 2 (right) illustrates a simple example. This is not the only way of getting such set-valued embeddings: for instance, as another NLPV modelling option, Property 3.18 in [19] discusses how to obtain an S-convex bounding of a function which can be expressed as the difference of two convex functions; the same work discusses additional options.

Example 3. Considering the same system and modelling region as the quasi-LPV bound in Example 2, we can embed the nonlinear system (8) by expressing it as (9) and considering $h \in \mathcal{H}_Q(x) := [-bx_1^2, bx_1^2]$: we are bounding the nonlinearity within *quadratic* (concave/convex) vertex models. The difference between the linear and the S-convex embedding appears in Figure 2(left).

No convexity properties in Definition 4 can be asserted for $\mathcal{H}_{nonlin}$. However, $\mathcal{F}$ in (9) is a linear deterministic set-valued map (indeed, it is actually an ordinary linear function); $\mathcal{H}_{LPV}$ in Example 2 is, too, a linear set-valued map (but not *deterministic*, because its output is an interval), and the just introduced $\mathcal{H}_Q$ is S-convex.

Note that $\mathcal{H}_{nonlin}(x) \subseteq \mathcal{H}_Q(x) \subseteq \mathcal{H}_{LPV}(x)$ in the chosen modelling region, so that $(\mathcal{F}, \mathcal{H}_{LPV})$ is an embedding of $(\mathcal{F}, \mathcal{H}_Q)$ and $(\mathcal{F}, \mathcal{H}_Q)$ is an embedding of $(\mathcal{F}, \mathcal{H}_{nonlin})$, where the latter pair faithfully replicates (8) as both maps are deterministic. As each embedding operation enlarges the uncertainty, more conservative results should arise from $\mathcal{H}_{LPV}$ than from $\mathcal{H}_Q$ (i.e., smaller set of admissible trajectories in the former, Def. 3); obviously, the least conservative would be considering $\mathcal{H}_{nonlin}$ which is what "pure nonlinear" control would ideally do, out of the scope of this work.

The main motivation of this manuscript is the fact that relaxing "linearity" in $\mathcal{H}_{LPV}$ to just "convexity" in $\mathcal{H}_Q$ (even quasi-convexity in other cases) will enable obtaining larger domain of attraction estimates for (8) in a gain-scheduled setting. As usual, when using $\mathcal{H}_{LPV}$ or $\mathcal{H}_Q$, the actual fact that $h \equiv x_2 x_1^2$ will *not* be used at design time in the gain-scheduled designs: only the bounds (vertex models) will be relevant; of course, in on-line operation gain-scheduled controllers will indeed be implemented as $u = \bar{u}(x, x_2 x_1^2)$.

Example 4. If multiple nonlinearities are present, the idea can be generalised to "box" NLPV models in the form[29]:

$$x_+ = \sum_{l=1}^{m} \sum_{i_l=0}^{1} h_{i_l} \cdot F_l \cdot f_{l,i_l}(x) + Gu + Ed \tag{10}$$

being F_l a constant vector, and $f_{l,0}(\cdot)$ concave and $f_{l,1}(\cdot)$ convex functions for each l, such that $f_{l,0}(x) \le f_{l,1}(x)$. Understanding (10) as a deterministic $\mathcal{F}(x,u,h)$ and letting $\mathcal{H}(x) := \Delta$, the "reduced" map $\mathcal{G}(x,u) := \bigcup_{h\in\Delta} \mathcal{F}(x,u,h)$ is S-convex. In general, if the vertex models f_{l,i_l} have uncertainty, results in this work are able to deal with models in, for instance, the form:

$$x_+ \in \bigoplus_{i=1}^{r} h_i \cdot \left(\mathcal{F}_i(x) \oplus \mathbf{G}_i u\right) \tag{11}$$

3.3 | Convexification of product nonlinearities

Let us assume that a given nonlinearity $\xi(h,x,u)$ can be factored as $\xi(h,x,u) = f(h,x,u)g(h,x,u)$. If separate maps such that $f(h,x,u) \in \mathcal{F}(h,x,u)$ and $g(h,x,u) \in \mathcal{G}(h,x,u)$ are built, then the product $\mathcal{F} \otimes \mathcal{G}$ loses the convexity properties that might be present in $\mathcal{F}$ or $\mathcal{G}$. For instance, (6) is not convex in (x,u,h) due to the products hx and hu. Nevertheless, results requiring convexity in fewer arguments will be later provided in particular cases.

In a generic product case, we can conservatively bound one of the factors by a set (i.e., a *constant* set-valued map) and keep S-convexity (and S-quasiconvexity) properties of the other one:

Proposition 1. Consider an expression $\rho(x) = A(x)f(x)$ and assume that vector $f(x)$ can be embedded into an S-quasiconvex (S-convex) map, i.e., $f(x) \in \mathcal{F}(x)$. Then, if a polytopic set of matrices $\mathbf{A}$ is found such that $A(x) \subseteq \mathbf{A}$ for all $x \in \mathbb{X}$, with $vert(\mathbf{A}) = \{A_i, i = 1,\dots,r\}$, the map $\mathcal{G}(x) := \mathbf{A} \otimes \mathcal{F}(x) = Co\left(\cup_{i=1}^{r} A_i \cdot \mathcal{F}(x)\right)$ is S-quasiconvex (S-convex).

Proof. Note that for any set $\mathbb{S} \subseteq \mathbb{R}^n$ and a linear transformation given by matrix A, we have that $Co(A\mathbb{S}) = ACo(\mathbb{S})$. Thus, consider $\tilde{x} = \gamma x_1 + (1-\gamma)x_2$, with $0 \le \gamma \le 1$. In the S-quasiconvex case, as $\mathcal{F}(\tilde{x}) \subseteq Co(\mathcal{F}(x_1) \cup \mathcal{F}(x_2))$, we can assert that

$$\begin{aligned}\mathcal{G}(\tilde{x}) = Co\left(\cup_{i=1}^{r} A_i \mathcal{F}(\tilde{x})\right) &\subseteq Co\left(\cup_{i=1}^{r} A_i Co(\mathcal{F}(x_1) \cup \mathcal{F}(x_2))\right)\\ &\subseteq Co(Co\left(\cup_{i=1}^{r} A_i \cdot \mathcal{F}(x_1)\right) \cup Co\left(\cup_{i=1}^{r} A_i \cdot \mathcal{F}(x_2)\right)) = Co(\mathcal{G}(x_1) \cup \mathcal{G}(x_2))\end{aligned} \tag{12}$$

The proof of the S-convex case roots on:

$$\begin{aligned}\mathcal{G}(\tilde{x}) = Co\left(\cup_{i=1}^{r} A_i \mathcal{F}(\tilde{x})\right) &\subseteq Co\left(\cup_{i=1}^{r} A_i \cdot (\gamma \mathcal{F}(x_1) \oplus (1-\gamma)\mathcal{F}(x_2))\right)\\ &= \gamma Co\left(\cup_{i=1}^{r} A_i \cdot \mathcal{F}(x_1)\right) \oplus (1-\gamma) Co\left(\cup_{i=1}^{r} A_i \cdot \mathcal{F}(x_2)\right) = \gamma \mathcal{G}(x_1) \oplus (1-\gamma)\mathcal{G}(x_2)\end{aligned} \tag{13}$$

□

Affine-in-control models

An affine-in-control NLPV system

$$x_+ = f(h,x,d) + G(h,x,d) \cdot u \tag{14}$$

is a widely-used model structure, hence deserving special attention. The left term $f(\cdot,\cdot,\cdot)$ can be embedded into an S-convex map with the ideas earlier discussed in this section or the ones in [20]. However, due to the product $G(h,x,d)u$, no convexity-related properties can be usually stated on the input term, as above discussed. One (quite conservative) option is using Proposition 1 to cover $G(h,x,d)u$ with an uncertain linear set-valued map $G(h,x,d)u \in \mathbf{G}u$, being $\mathbf{G}$ a set of matrices encompassing all possible values of $G(h,x,d)$. Conservatism arises because subsequent control designs would not be able to schedule on the nonlinearities in G, not present in $\mathbf{G}$ (they are masked as uncertainty).

In order to remove part of such conservatism, there are at least two possible options, both of which use a vector of *artificial* input variables, denoted as $\bar{u}$.

a) Basis-function expressions (≈ feedback linearisation). We may consider an input change of variable in the form:

$$u = \Upsilon(x,h)\bar{u}, \tag{15}$$

with $\Upsilon \in \mathbb{R}^s \mapsto \mathbb{R}^m$ being a matrix of user-defined "basis functions" and $\bar{u}$ being "artificial" input variables; note that the size s of $\bar{u}$ may not be coincident with that of u, if so wished –actually, such an option is later pursued in this section, see (18)–.

Using (15) may remove conservatism if $\Upsilon(x,h)$ is suitably selected to carry out, for instance, cancellation/linearisation/pseudoinverse of some elements of $G(h,x,d)$, so $G(h,x,d)\Upsilon(x,h)$ renders in a "simpler" form. The drawback of this approach is that the choice of Υ is problem-dependent: in some problems straightforward cancellations can be carried out (for instance, with $g(x,h,d)u = \frac{1}{1+x^2}u$, setting $\Upsilon(x) = (1+x^2)$ trivially results in a linearised expression $g\cdot\Upsilon\cdot\bar{u} = \bar{u}$) –indeed, motivation for (15) comes from feedback linearisation–; however, in many higher-order models, direct input-channel linearisation is not possible. Note also that, for computational reasons, input bounds in u must be translated to polytopic bounds on $\bar{u}$; this step may introduce conservatism in a general case.

b) Polya relaxations. Polya (copositivity) relaxations are widely used in LPV gain-scheduling [33,27], as well as in Sum-of-Squares frameworks [17]. In a set-based LPV approach, they are used in [27], for instance. Let us assume that G in the NLPV system (14) depends only on h and it can be expressed as:

$$G(h) = \textstyle\sum_{i=1}^r \mu_i(h)G_i, \quad 0 \le \mu_i \le 1,\ \sum_{i=1}^r \mu_i = 1 \tag{16}$$

where, if x is measurable, it can be assumed to be part of h with no loss of generality. Thus, $G(h)$ is, by assumption, an homogeneous polynomial of degree 1 in $\mu = \{\mu_1,\dots,\mu_r\}$. For the moment being, we disregard dependence of μ on h writing $G(\mu)$ instead.

We will represent an homogeneous polynomial of degree d with notation $\sum_{|s|=d}\mu^s q_s$, being $s := \{s_1,\dots,s_r\}$ an r-dimensional multi-index composed of non-negative integers, with $|s| := \sum_{i=1}^r s_i$ and $\mu^s := \Pi_{i=1}^r \mu_i^{s_i}$. In particular, we will express

$$1 = (\sum_{i=1}^r \mu_i)^d = \sum_{i_1=1}^r \cdots \sum_{i_d=1}^r \mu_{i_1}\cdot\mu_{i_d} = \sum_{|s|=d}\mu^s n_s \tag{17}$$

where $n_s = \frac{d!}{\Pi_{i=1}^r s_i!}$. For instance, for $d=3$ and $r=2$ we would have $(\sum_{i=1}^2 \mu_i)^3 = \mu_1^3 + 3\mu_1^2\mu_2 + 3\mu_1\mu_2^2 + \mu_2^3$, so for $s=\{3,0\}$ and $s=\{0,3\}$ we have $n_s = 1$, whereas for $s=\{1,2\}$ and $s=\{2,1\}$ we have $n_s = 3$.

In general, we will assume that $G(\mu)$ can be expressed as an homogeneous polynomial on μ (degree d_g, being (16) the particular case $d_g = 1$), and artificial $\bar{u}$ are introduced so a gain-scheduled u is also defined as an homogeneous polynomial in μ (degree d_u), expressed as

$$u(\mu,\bar{u}) := \sum_{|s|=d_u}\mu^s\cdot\bar{u}_s \tag{18}$$

Now, multiplying by $(\sum_{i=1}^r \mu_i)^{d_P}$ (degree $d_P \ge 0$ is a complexity parameter, arbitrarily chosen), we can write $G(\mu)u(\mu)$ as an homogeneous polynomial of degree $d := d_P + d_g + d_u$ in μ, i.e., $G(\mu)u(\mu,\bar{u}) = \sum_{|s|=d}\mu^s n_s q_s(\bar{u})$, where coefficients $n_s q_s(\bar{u})$ are linear functions of $\bar{u}$. Then, an expression in the form (15) immediately arises.

Proposition 2. Under the above assumptions and notation, $G(\mu)u(\mu,\bar{u}) \in \mathcal{G}(\bar{u}) := Co\left(\{q_s(\bar{u}) : |s| = d\}\right)$.

Proof. Evident from the fact that $\sum_{|s|=d}\mu^s n_s = 1$ and $\mu^s n_s \ge 0$. □

The usefulness of Proposition 2 lies in the fact that a non-convex $G(h)u$ can be embedded into a polytopic linear set-valued map $\mathcal{G}(\bar{u})$ which depends on the new artificial inputs (decision variables), such that a gain-scheduled controller can be designed from $\bar{u}$. Note that Polya relaxations do apply, with straightforward modifications, for the case of common nonlinearities in f

and G, such as $x_+ = \sum_{i=1}^r \mu_i(h)\left(f_i(h,x,d) + G_i u\right)$, as well as in expressions with products of nonlinearities giving rise to tensor-product expressions of μ_i, details omitted for brevity, see for instance the notation in[34].

Example 5. Consider $x_+ = f(x)+(2+sin(h))u$. The model can be expressed as $x_+ = f(x)+G(\mu)u$, being $G(\mu) = \sum_{i=1}^2 \mu_i(h)G_i$, with $G_1 = 1$, $G_2 = 3$, $\mu_1 = \frac{1-sin(h)}{2}$, $\mu_2 = \frac{sin(h)+1}{2}$, i.e., we have $d_g = 1$. Let us consider $d_u = 4$ so the control law proposal (18) results in the following expression involving five augmented input decision variables:

$$u(\mu,\bar{u}) = \mu_1^4 \cdot \bar{u}_{40} + \mu_1^3\mu_2 \cdot \bar{u}_{31} + \mu_1^2\mu_2^2 \cdot \bar{u}_{22} + \mu_1\mu_2^3 \cdot \bar{u}_{13} + \mu_2^4 \cdot \bar{u}_{04} \tag{19}$$

Last, we will chose complexity parameter $d_P = 1$. In total, we can build a degree 6 homogeneous polynomial:

$$\begin{aligned} G(\mu)u(\mu,\bar{u}) = {} & \mu_1^6 \cdot G_1\bar{u}_{40} + \mu_1^5\mu_2 \cdot \left(G_1\bar{u}_{31} + (G_1+G_2)\cdot\bar{u}_{40}\right) + \mu_1^4\mu_2^2 \cdot \left(G_1\bar{u}_{22} + G_2 u_{40} + (G_1+G_2)\bar{u}_{31}\right) \\ & + \mu_1^3\mu_2^3 \cdot \left(G_1\bar{u}_{13} + G_2\bar{u}_{31} + (G_1+G_2)\bar{u}_{22}\right) + \mu_1^2\mu_2^4 \cdot \left(G_1\bar{u}_{04} + G_2 u_{22} + (G_1+G_2)\bar{u}_{13}\right) \\ & + \mu_1\mu_2^5 \cdot \left(G_2\bar{u}_{13} + (G_1+G_2)\cdot\bar{u}_{04}\right) + \mu_2^6 \cdot G_2\bar{u}_{04} \end{aligned} \tag{20}$$

and, as $1 = (\mu_1+\mu_2)^6 = \mu_1^6 + 6\mu_1^5\mu_2 + 15\mu_1^4\mu_2^2 + 20\mu_1^3\mu_2^3 + 15\mu_1^2\mu_2^4 + 6\mu_1\mu_2^5 + \mu_2^6$, Proposition 2 asserts that

$$\begin{aligned} G(\mu)u(\mu,\bar{u}) \in Co\Big\{ & G_1\bar{u}_{40},\ \frac{1}{6}\left(G_1\bar{u}_{31} + (G_1+G_2)\cdot\bar{u}_{40}\right),\ \frac{1}{15}\left(G_1\bar{u}_{22} + G_2\bar{u}_{40} + (G_1+G_2)\bar{u}_{31}\right), \\ & \frac{1}{20}\left(G_1\bar{u}_{13} + G_2\bar{u}_{31} + (G_1+G_2)\bar{u}_{22}\right),\ \frac{1}{15}\left(G_1\bar{u}_{04} + G_2\bar{u}_{22} + (G_1+G_2)\bar{u}_{13}\right),\ \frac{1}{6}\left(G_2\bar{u}_{13} + (G_1+G_2)\cdot\bar{u}_{04}\right),\ G_2\bar{u}_{04}\Big\} \end{aligned} \tag{21}$$

and, hence, we can formally design an augmented non-scheduled controller (i.e., obtaining $\bar{u}_{40}(x)$, $\bar{u}_{31}(x)$, $\bar{u}_{22}(x)$, $\bar{u}_{13}(x)$, $\bar{u}_{04}(x)$ which do not depend on h) with the 7-vertex 5-input polytopic model (21), and later on reduce it to a gain-scheduled control law (19). The same idea will be later used in Section 7, see (42) and (44) for instance.

Note that, in order to avoid notational clutter with double subscripts, we could equivalently have written $u(\mu,\bar{u}) = \mu_1^4\bar{u}_1 + \mu_1^3\mu_2\bar{u}_2 + \mu_1^2\mu_2^2\bar{u}_3 + \mu_1\mu_2^3\bar{u}_4 + \mu_2^4\bar{u}_5$ in terms of a 5-dimensional augmented input $(\bar{u}_1,\dots,\bar{u}_5)$, instead of the multi-index in (19).

Further modelling examples

Additional examples of embedding nonlinearities into S-convex uncertainty (set-valued maps) are presented in[19], including parametric/additive uncertainty elements, ellipsoidal uncertainty, difference-of-convex nonlinearities, etc. The reader is referred to the cited work and later ones by the same team for in-depth coverage of S-convex NLPV embedding (and robust, i.e., non-gain-scheduled, controller design). Some S-quasiconvex options appear in[28], but S-quasiconvexity is lost even with linear transformations so options in this respect are much more limited. Note, finally, that non-scheduled controller design does not need Polya relaxations; due to this fact, such relaxations are not covered in the cited literature dealing with S-convex modelling.

For brevity, further NLPV modelling details are intentionally out of the scope of this paper.

4 | GAIN-SCHEDULED 1-STEP SETS

Basic definitions

Most concepts in set-based control design root on the well-established one-step set[4] concept, i.e., the set of “source” states that can be driven in one step to a given “target set” Ω. Its gain-scheduled generalisation will be defined as:

Definition 5. Given a target set Ω, the **gain-scheduled one-step set** of Ω for a pair $(\mathcal{F},\mathcal{H})$ in a modelling region $\mathbb{X}$ is defined as:

$$\mathcal{Q}(\Omega) := \{x \in \mathbb{X} : \exists\, \upsilon_x : \mathcal{H}(x) \mapsto \mathbb{U} \text{ such that } \mathcal{F}(x,\upsilon_x(h),h) \subseteq \Omega,\ \ \forall h \in \mathcal{H}(x)\} \tag{22}$$

Conversely, if a given point x belongs to $\mathcal{Q}(\Omega)$ it will be said to be **1-step feasible** for Ω.

In plain words, the gain-scheduled one-step set requires, for each x in $\mathcal{Q}(\Omega)$, the existence of a *scheduling control function*, denoted as $\upsilon_x(h)$, whose argument are the scheduling variables h. The function must, obviously, be defined over the set $\mathcal{H}(x)$ of possible values of h for a given x. Then, using $u = \upsilon_x(h)$, the set of all possible successor states, x_+ in (3), of the source state x can be steered to Ω. As υ_x will be, in general, a different function for different source states x, the above definition implies that there exists a gain-scheduled controller $u = \bar{u}(h,x) \equiv \nu_x(h)$ such that, for all $x \in \mathcal{Q}(\Omega)$, and for all $h \in \mathcal{H}(x)$ we have $\mathcal{F}(x,\bar{u}(h,x),h) \subseteq \Omega$. Note that, in the already cited prior non-scheduled S-convex control literature[4,19,32], for each x, only a *constant u* is considered (amounting to just a state-feedback $u(x)$), instead of a function υ_x of the scheduling variables, as done in the gain-scheduled generalisation presented in this section.

From Definition 5, $\mathcal{Q}^N(\Omega) := \mathcal{Q}(\mathcal{Q}(\dots \mathcal{Q}(\Omega)))$ is the set of states that can be driven in N steps to Ω with a gain-scheduled control law, to be denoted as the gain-scheduled N-step set of Ω; in other words, $\mathcal{Q}^N(\Omega)$ contains all of the N-step feasible states for Ω.

In general, the "exact" computation of $\mathcal{Q}^N(\Omega)$ for a generic nonlinear system or a PDDI (3) is a difficult problem. As discussed in earlier sections, linear (i.e., LPV) settins in a set-based approach appear in[27], non-scheduled S-convex cases appear in Fiacchini's works[20,21] and gain-scheduled options in S-convex and S-quasiconvex cases are the goal of this work. Actually, numerical algorithms, to be later discussed, will only be able to output a polyhedron Ξ which is an inner approximation to $\mathcal{Q}(\Omega)$, i.e., $\Xi \subseteq \mathcal{Q}(\Omega)$.

The larger "modelling uncertainty" is, the smaller the N-step sets estimates will be:

Proposition 3. If $(\mathcal{F}', \mathcal{H}')$ is an embedding of $(\mathcal{F}, \mathcal{H})$ then the one-step set for $(\mathcal{F}', \mathcal{H}')$, denoted as $\mathcal{Q}'(\Omega)$ is a subset of the one-step set $\mathcal{Q}(\Omega)$ for $(\mathcal{F}, \mathcal{H})$.

Proof. Straightforward from the fact that $\mathcal{F} \subseteq \mathcal{F}'$ and $\mathcal{H} \subseteq \mathcal{H}'$ in (22). □

The following notation will be useful for later developments.

Definition 6. Let us define $\mathcal{F}_x(u,h) := \mathcal{F}(x,u,h)$ as the restriction to x of $\mathcal{F}$, i.e., a two-argument set-valued map $\mathcal{F}_x : \mathbb{U} \times \mathcal{H}(x) \mapsto \mathcal{P}(\mathbb{R}^n)$. Likewise, we will denote the restriction to h, $\mathcal{F}_h(x,u) := \mathcal{F}(x,u,h)$, i.e., considering $\mathcal{F}_h : \widetilde{\mathbb{X}}_h \times \mathbb{U} \mapsto \mathcal{P}(\mathbb{R}^n)$, with $\widetilde{\mathbb{X}}_h := \{x : h \in \mathcal{H}(x)\}$, and the restriction to (x,h) as $\mathcal{F}_{xh}(u)$, ie., $\mathcal{F}_{xh}(u) : \mathbb{U} \mapsto \mathcal{P}(\mathbb{R}^n)$.

Additionaly, in the sequel, $\mathcal{F}$ and $\mathcal{H}$ will be assumed to be polytopic (Def. 4), for computational reasons.

Conditions for 1-step feasibility (single point)

Under some convexity assumptions on $(\mathcal{F}, \mathcal{H})$, we will first present conditions to check if a single point x belongs to $\mathcal{Q}(\Omega)$, for a given "target polytope" Ω, and obtain the control action ensuring feasibility as a by-product.

Proposition 4. Consider a pair $(\mathcal{F}, \mathcal{H})$, and a given source state $x \in \mathbb{X}$.

If the restriction to x, $\mathcal{F}_x(u,h)$, is S-quasiconvex and, for each $\zeta \in vert(\mathcal{H}(x))$, there exists $\upsilon \in \mathbb{U}$ such that $\mathcal{F}(x, \upsilon, \zeta) \subseteq \Omega$, then, the following two assertions hold:

a) 1-step feasibility: $x \in \mathcal{Q}(\Omega)$.

b) (*Control law construction* from vertex actions υ) Denote the elements of $vert(\mathcal{H}(x))$ as ζ_i, $i = 1, \dots, N_{\mathcal{H}}$, being $N_{\mathcal{H}}$ the number of such vertices. For any $h \in \mathcal{H}(x)$, denote as $\eta(h) \in \Delta$ any arbitrary choice of convex coordinates such that $h = \sum_{i=1}^{N_{\mathcal{H}}} \eta_i(h)\zeta_i$, $\eta_i(h) \geq 0$. The scheduling control function:

$$\upsilon_x(h) := \textstyle\sum_{i=1}^{n_{\mathcal{H}}} \eta_i(h)\upsilon_i \tag{23}$$

achieves $\mathcal{F}(x, \upsilon_x(h), h) \subseteq \Omega$ for all $h \in \mathcal{H}(x)$.

Proof. Theorem 1 ensures that for any $h \in \mathcal{H}(x)$, we have

$$\mathcal{F}(x, \upsilon_x(h), h) = \mathcal{F}(x, \sum_{i=1}^{n_{\mathcal{H}}} \eta_i(h)\upsilon_i, \sum_{i=1}^{n_{\mathcal{H}}} \eta_i(h)\zeta_i) \subseteq Co\left(\{\mathcal{F}(x, \upsilon_i, \zeta_i),\ i = 1, \dots N_{\mathcal{H}}\}\right)$$

due to the S-quasiconvexity assumption on $\mathcal{F}_x$. Now, convexity of the polytope Ω and $\mathcal{F}(x, \upsilon_i, \zeta_i) \subseteq \Omega$ ensure that the convex hull at the right-hand side is a subset of Ω, so the control law (23) steers x to Ω, whatever the value of h happens to occur. □

4.1 | Conditions for 1-step feasibility of a source polytope Ξ

We will now check if a given "candidate polytope" in the state space, say $\Xi \subset \mathbb{X}$, is a subset of $\mathcal{Q}(\Omega)$. For the moment being, the target and candidate polytopes will be assumed to be explicitly known; later on, they will be conveniently generated by iterative algorithms.

For convenience, there is no loss of generality in assuming that $\mathcal{F}$ in a pair $(\mathcal{F}, \mathcal{H})$ does *not* depend on the state. First of all, $\mathcal{F}$ might not depend on "all" the state components in the original model, i.e., it might be expressed as $\mathcal{F}(\hat{x}, u, h)$ being $\hat{x}$ only the

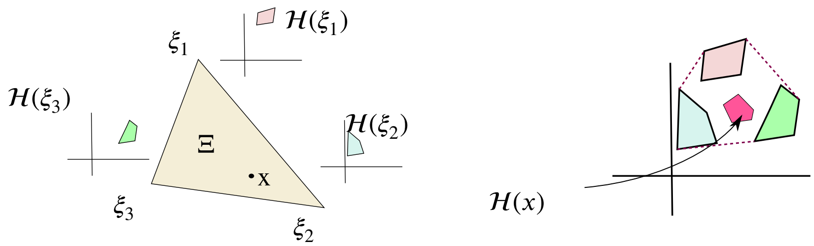

FIGURE 3 Illustration on a polytope in state space Ξ with vertices (ξ_1, ξ_2, ξ_3), and in scheduling variables' space $(\mathcal{H}(\xi_1), \dots)$.

states that *explicitly* appear on the model $\mathcal{F}$. Now, as the state is assumed measurable in gain-scheduled state-feedback control, we can define $\bar{h} = (\hat{x}, h)$, and consider a PDDI $x_+ \in \overline{\mathcal{F}}(u, \bar{h})$, $\bar{h} \in \bar{\mathcal{H}}(x)$, i.e., a pair $\left(\overline{\mathcal{F}}(u, \bar{h}), \bar{\mathcal{H}}(x)\right)$, with $\bar{\mathcal{H}}(x) = (\hat{x}, \mathcal{H}(x))$. With this notation, a gain-scheduled control law will now be written as $u = \bar{u}(\bar{h})$.

Theorem 2. Given a pair $\left(\overline{\mathcal{F}}(u, \bar{h}), \bar{\mathcal{H}}(x)\right)$, assume that both $\overline{\mathcal{F}}$ and $\bar{\mathcal{H}}$ are S-quasiconvex. Consider a known "target" polytope Ω, and a candidate "source" polytope Ξ. If every $\xi \in vert(\Xi)$ is 1-step feasible, i.e., $vert(\Xi) \subseteq \mathcal{Q}(\Omega)$, then $\Xi \subseteq \mathcal{Q}(\Omega)$.

Proof. Let us enumerate the vertices of Ξ as ξ_j, $j = 1, \dots, N_\Xi$. S-quasiconvexity of $\overline{\mathcal{H}}$ ensures that for any $x \in \Xi$ (i.e., not necessarily a vertex point), we will have $\overline{\mathcal{H}}(x) \subseteq Co\left(\bigcup_{j=1}^{N_\Xi} \overline{\mathcal{H}}(\xi_j)\right)$. Equivalently, for any $\bar{h} \in \overline{\mathcal{H}}(x)$, there exist convex coordinates $\left(\theta_1, \dots, \theta_{N_\Xi}\right) \in \Delta$, and points $\bar{h}_j \in \overline{\mathcal{H}}(\xi_j)$ such that:

$$\bar{h} = \sum_{j=1}^{N_\Xi} \theta_j \bar{h}_j \tag{24}$$

Hence S-quasiconvexity of $\mathcal{F}$ ensures that the following definition of the gain-scheduled control law $\bar{u}(\bar{h})$, in terms of the scheduling control functions from the vertices of Ξ (denoted as $v_{\xi_j}(h)$, which exist by assumption in the theorem statement; note that h is the second component of $\bar{h} = (\hat{x}, h)$):

$$\bar{u}(\bar{h}) := \sum_{j=1}^{N_\Xi} \theta_j v_{\xi_j}(h_j) \tag{25}$$

fulfills:

$$\overline{\mathcal{F}}(\bar{u}(\bar{h}), \bar{h}) = \overline{\mathcal{F}}\left(\sum_{j=1}^{N_\Xi} \theta_j v_{\xi_j}(h_j), \sum_{j=1}^{N_\Xi} \theta_j \bar{h}_j\right) \subseteq Co\left(\bigcup_{j=1}^{N_\Xi} \overline{\mathcal{F}}(v_{\xi_j}(h_j), \bar{h}_j)\right) \subseteq \Omega \tag{26}$$

because $\overline{\mathcal{F}}(v_{\xi_j}(h_j), \bar{h}_j) \subseteq \Omega$ and Ω is a convex set. □

Constant $\mathcal{H}$. In many modelling cases, there are products between scheduling parameters and functions of the states, LPV models being the paradigmatic well-known case; however, scheduling parameters are set to lie in a constant (state-independent) box, or the standard simplex Δ. Products usually break the convexity properties; conditions below, based on the convexity properties of the restrictions (Def. 6), apply in such frequent cases.

Theorem 3. Given a pair $(\mathcal{F}, \mathcal{H})$, assume that $\mathcal{F}_h(x, u)$ is an S-quasiconvex map, and $\mathcal{H}$ is a constant map $\mathcal{H}(x) := \mathbb{H}$. Consider a known "target" polytope Ω, and a candidate source polytope Ξ. If every $\xi \in vert(\Xi)$ is 1-step feasible, then $\Xi \subseteq \mathcal{Q}(\Omega)$.

Proof. Let us enumerate the vertices of Ξ as ξ_j, $j = 1, \dots, N_\Xi$. For any $x \in \Xi$, denote by $\theta(x) \in \Delta$ a set of convex coordinates such that $x = \sum_{j=1}^{N_\Xi} \theta_j(x)\xi_j$. By assumption, functions $v_{\xi_j}(h)$ exist proving $\xi_j \in \mathcal{Q}(\Omega)$. With them, let us define the scheduling function:

$$v_x(h) := \sum_{j=1}^{N_\Xi} \theta_j(x) v_{\xi_j}(h) \tag{27}$$

Then:

$$\mathcal{F}\left(x, v_x(h), h\right) = \mathcal{F}\left(\sum_{j=1}^{N_\Xi} \theta_j(x)\xi_j, \sum_{j=1}^{N_\Xi} \theta_j(x) v_{\xi_j}(h), h\right) \subseteq Co\left(\bigcup_{j=1}^{N_\Xi} \mathcal{F}\left(\xi_j, v_{\xi_j}(h), h\right)\right) \subseteq \Omega \tag{28}$$

□

Even if products appear, in order to apply the above results, in some cases changes of variable can be made. The following corollary casts our preliminary conference paper[29] in the context of the just presented results (in fact, a generalisation incorporating *quasi*convexity, disturbances and Polya relaxations).

Corollary 1 ([29])**.** Consider the affine-in-control PDDI given by:

$$x_+ \in \sum_{i=1}^{r} \mu_i f_i(x) \oplus \mathbf{G} \cdot u \oplus \mathbb{D} \tag{29}$$

with $\mu \in \Delta$, $f_i : \mathbb{R}^n \mapsto \mathbb{R}^n$ being the so-called "vertex models", possibly nonlinear, $\mathbf{G}$ being a polytopic set of matrices modelling "input uncertainty" and Polya relaxations, if needed (as discussed in Section 3), and $\mathbb{D}$ is a polytopic set where bounded additive disturbances may take values. Denote $\mathcal{H}(x) := Co(f_i(x))$, and assume that $\mathcal{H}$ is a S-quasiconvex map. Consider a "target" polytope Ω, and a candidate source polytope Ξ. The vertices of Ξ will be enumerated as ξ_j, $j = 1, \dots, N_\Xi$; those of $\mathbb{D}$ will be enumerated as d_k, $k = 1, \dots, N_D$. If, for each $i \in \{1, 2, \dots, r\}$, $j \in \{1, \dots, N_\Xi\}$ there exist v_{ij} such that the inclusion condition below holds:

$$f_i(\xi_j) \oplus \mathbf{G} \cdot v_{ij} \oplus \mathbb{D} \in \Omega \tag{30}$$

then $\Xi \subseteq \mathcal{Q}_{\mathcal{F}}(\Omega)$.

Proof. Alternatively from the proof in[29], we can make the change of variable $h = \sum_{i=1}^{r} \mu_i f_i(x)$. Then, $x_+ \in \mathcal{F}(x, h, u) = h \oplus \mathbf{G}u \oplus \mathbb{D}$, and, trivially $h \in \mathcal{H}(x)$.

Now, in order to apply Theorem 2, we can state that $\hat{x}$ is empty, so $h \equiv \bar{h}$, and, with a slight abuse of notation, $\mathcal{F} \equiv \overline{\mathcal{F}}, \mathcal{H} \equiv \overline{\mathcal{H}}$.

Thus, $\overline{\mathcal{F}}$ is affine (hence S-quasiconvex), and $\overline{\mathcal{H}}$ is S-quasiconvex. Also, (30) for fixed j asserts that each vertex of Ξ fulfills conditions in Proposition 4 so vertices are 1-step feasible (S-quasiconvexity of $\mathcal{F}_x$ trivially holds as $\overline{\mathcal{F}}$ does not depend on the state). Hence, conditions for Theorem 2 apply, and a gain-scheduled control law can be built. ☐

Corollary 2. Given a PDDI defined by a pair $(\mathcal{F}, \mathcal{H})$ or the equivalent representation $(\overline{\mathcal{F}}, \overline{\mathcal{H}})$, if a collection of points $\xi_1, \dots, \xi_N$ is 1-step feasible for a target set Ω, and the convexity conditions in the theorem statements in this section hold, the polyhedron $\Xi = Co\left(\bigcup_{j=1}^{N} \xi_j\right)$ fulfills $\Xi \subseteq \mathcal{Q}(\Omega)$.

Proof. Omitted, as it's trivial from Theorem 1. ☐

5 | APPROXIMATION OF 1-STEP SETS: ALGORITHMS

In order to build tractable algorithms based on the above results, we need a polynomial-time sufficient condition to check, given x, ζ and a target set Ω, if there exists v such that $\mathcal{F}(x, v, \zeta) \subseteq \Omega$. Indeed, Proposition 4 states that if such check is successful for all $\zeta \in vert(\mathcal{H}(x))$, then $x \in \mathcal{Q}(\Omega)$.

Note first that, under the polytopic assumption for $\mathcal{F}$ and Ω, Proposition 4 amounts to checking a *finite* number of conditions in decision variable v involving vertices of $\mathcal{H}(x)$, vertices of $\mathcal{F}$, and faces (inequalities) of Ω.

In affine-in-control cases[4] such as (29), this check is easy. For instance, consider polytope Ω described as a finite set of linear inequalities, $\Omega := \{x \in \mathbb{R}^n : Rx \leq s\}$. Then, checking that one vertex is 1-step feasible with Proposition 4 reduces to checking (30) for fixed j, suitably enumerating the vertices of $\mathbf{G}$ as G_k and those of $\mathbb{D}$ as d_l, i.e., for all i, k, l in their respective ranges:

$$R\left(f_i(\xi_j) + G_k \cdot v_{ij} + d_l\right) \leq s, \tag{31}$$

This is a finite set of linear inequalities, whose feasibility is easily assessed by standard Linear Programming code, such as MPT3[6], used in later examples.

In order to determine an inner approximation of the gain-scheduled one-step set of a set Ω for a DI $x_+ \in \mathcal{F}(x, u, h)$, we can use the following "ray-tracing" algorithm:

[4]In the non affine-in-control case, if the support function of $\mathcal{F}_{xh}(u)$, i.e., $\Phi_{\mathcal{F}_{xh}(u)}(\eta, u) := \sup_{y \in \mathcal{F}_{xh}(u)} \eta^T y$, is convex or quasi-convex in u, computationally viable conditions may be cast[19,35] as $\mathcal{F}_{xh}(u) \subseteq \Omega$ is equivalent to a finite set of convex (or quasi-convex) constraints $\Phi_{\mathcal{F}_{xh}(u)}(\eta, u) \leq \Phi_\Omega(\eta)$, see[19] for details, omitted here for brevity.

Algorithm 1 [one-step set, inner polytopic approximation]
Inputs: Ω (target set, verifying $\Omega \in \mathbb{X}$).

1. Generate an arbitrary set of K unit-norm vectors $\rho_1, \ldots, \rho_K$ in $\mathbb{R}^n$ (state space).
2. For each ρ_k, $k = 1, \ldots, K$, determine by bisection the largest scaling γ_k such that $\xi_k := \gamma_k \rho_k \in \mathcal{Q}(\Omega)$, using Proposition 4.
3. Form $\Xi := Co(\{\xi_k, k = 1, \ldots, K\})$.
4. End. Corollary 2 ensures that $\Xi \subseteq \mathcal{Q}(\Omega)$. We will denote the resulting polyhedral approximation as $\widetilde{\mathcal{Q}}(\Omega) := \Xi$.

5.1 | Contractive set computation

Definition 7. A set $\Omega \subseteq \mathbb{X}$ is said to be **gain-scheduled control λ-contractive** for a pair $(\mathcal{F}, \mathcal{H})$ if $\Omega \subseteq \mathcal{Q}(\lambda\Omega)$. Setting $\lambda = 1$, Ω will be said to be **gain-scheduled invariant** if $\Omega \subseteq \mathcal{Q}(\Omega)$.

The above definition is the generalisation of the non-scheduled control λ-contractive sets[1,4,19] to the scheduled case[27]: λ-contractive sets are the basic element of geometric decay-rate stability analysis around an equilibrium point set as the origin $x = 0$, later analysed on in Section 6. Of course, if only computations regarding "invariance" ($\lambda = 1$) were pursued, then the origin would have no special meaning in such a case.

As, by definition contractiveness of Ω is equivalent to $\Omega \subseteq \mathcal{Q}(\lambda\Omega)$, the one-step set operator approximated by Algorithm 1 can be used to compute a contractive set by repeated application of it.

Indeed, starting with an arbitrary set $\Xi_0 := \mathbb{S}$, we can repeat $\Xi_{k+1} = \widetilde{\mathcal{Q}}(\lambda\Xi_k)$ until $\Xi_k \subseteq \Xi_{k+1}$; then, $\Xi_k \subseteq \Xi_{k+1} \subseteq \mathcal{Q}(\lambda\Xi_k)$ so it is λ-contractive.

In a LPV case (see[4] for non-scheduled versions, and[27] for a gain-scheduled case), there is alternative implementation of Algorithm 1 that converges to a set Ω fulfilling $\Omega = \mathcal{Q}(\lambda\Omega)$ in a finite number of repetitions if the LPV system is quadratically asymptotically stabilizable. However, in the general S-quasiconvex case dealt with here, there is no guarantee that repeated application of the operator $\widetilde{\mathcal{Q}}$ (result of Algorithm 1) reaches a contractive set, due to the approximate nature of $\widetilde{\mathcal{Q}}$. So, if Algorithm 1 does not succeed in finding a contractive set for a given nonlinear system (perhaps because number of rays K is too small), an alternative option (at least in a disturbance-free case) is setting a "small enough" modelling region around the origin, and use an auxiliary LPV embedding. Indeed, if the classical Jacobian linearisation is stabilizable, a small modelling region will render an LPV representation of a nonlinear systems with vertex models close to such linearisation[16], so feasibility of LPV algorithms can be ultimately guaranteed. This is, for instance, the suggestion in[20].

In order to expand the resulting "small" contractive set using less-conservative S-quasiconvex models, we can use Algorithm 2 below, which actually is a gain-scheduled reinterpretation of the shooting algorithm arising from equation (11) in[20]. The algorithm can obtain progressively larger λ-contractive sets if one of them (to be denoted as "seed" set) is available. It should run "forever" in theory or, in practice, until some termination criteria (maximum iterations, volume of the set, etc.) is met.

Algorithm 2 [contractive set expansion[20]]
Inputs: Ω_0, an initial gain-scheduled λ-contractive polyhedral "seed" set.

1. Set $k = 0$.
2. Choose a random unit-norm vector ρ.
3. Determine, by bisection and Proposition 4, the largest scaling $\gamma > 0$ such that $\gamma\rho \in \mathcal{Q}(\lambda\Omega_k)$.
4. Set $\Omega_{k+1} = Co(\Omega_k \cup \{\gamma\rho\})$. As $\Omega_k \subseteq \mathcal{Q}(\lambda\Omega_k)$, Corollary 2 ensures $\Omega_{k+1} \subseteq \mathcal{Q}(\lambda\Omega_k) \subseteq \mathcal{Q}(\lambda\Omega_{k+1})$, because $\Omega_k \subseteq \Omega_{k+1}$ implies $\mathcal{Q}(\lambda\Omega_k) \subseteq \mathcal{Q}(\lambda\Omega_{k+1})$. Thus, Ω_{k+1} is gain-scheduled control λ-contractive.
5. Set $k = k + 1$. If $k < k_{max}$, go to step 2, otherwise end. The algorithms provides a sequence of λ-contractive sets which verify $\Omega_0 \subseteq \Omega_1 \subseteq \Omega_2 \subseteq \ldots$

Example 6. In order to illustrate Algorithm 2, let us consider the PDDI

$$x_+ = \begin{pmatrix} 0.5x_1 + h \\ 0.5x_2 + h \end{pmatrix} + \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} u, \qquad \mathcal{H}(x) = [-0.1(1 + |x_1|), 0.1(1 + x_1^2 + x_2^2)]$$

and constraint $|u| \leq 2$. Note that $\mathcal{H}$ is S-convex as it has concave/convex bounds.

Step 1. The square $\Omega_0 := Co\{(-1,-1),(-1,1),(1,-1),(1,1)\}$ is gain-scheduled invariant. Indeed, in all four vertices $h \in [-0.2, 0.3]$, so with $u = 0$ the successor state of each vertex lies inside the square Ω_0; as the model fulfills the S-quasiconvexity conditions required in Theorem 2, invariance of the whole square is guaranteed from 1-step feasibility of its vertices.

Step 2. Now, considering Ω_0 as seed set, we wish to extend the invariant set in the (arbitrary) direction $\rho = (0.8, 0.6)$.

Step 3. Setting, say, $\gamma = 2$, the resulting candidate point $p = \gamma\rho = (1.6, 1.2)^T$ has $\mathcal{H}(p) = [-0.26, 0.5]$. We need to determine if p is 1-step feasible for Ω_0, applying Proposition 4, which allows us to set up independent conditions for each vertex of $\mathcal{H}(p)$.

For the first vertex of $\mathcal{H}(p)$, replacing in the model equation, we must find u_1 such that $x_+ = \begin{pmatrix} 0.54 \\ 0.34 \end{pmatrix} + \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} u_1 \in \Omega_0$; this is trivially fulfilled with $u_1 = 0$. For the second vertex of $\mathcal{H}(p)$, replacing in the model equation, we must find u_2 such that $x_+ = \begin{pmatrix} 1.3 \\ 1.1 \end{pmatrix} + \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} u_2 \in \Omega_0$, fulfilled with, say, $u_2 = -0.3$.

As $\gamma = 2$ renders a 1-step feasible point, we can try a larger γ. Bisection ends up finding $\gamma = 3.1097$ rendering $p = (2.4878, 1.8658)^T$ so that $\mathcal{H}(p) = [-0.3488, 1.0670]$. In this case, the first vertex requires that $\begin{pmatrix} 0.8951 \\ 0.5841 \end{pmatrix} + \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} u_1 \in \Omega_0$ renders feasible, which is the case with $u_1 = 0$ and the second vertex requires $\begin{pmatrix} 2.3109 \\ 1.9999 \end{pmatrix} + \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} u_1 \in \Omega_0$, true with $u_1 = -2$. Larger values of γ would need u above its saturation limit.

Step 4. The convex hull of the original square and the point $p = (2.4878, 1.8658)^T$ is, then, gain-scheduled invariant. This new polyhedron would replace Ω_0 and a new iteration would be made. Checking that there exists an input driving a point to a square can be done "by hand" in the first iteration, but it would need, say, linear programming for an arbitrary Ω_k, see Section 5.2 next.

5.2 | Control law computation

In this section, we will address the problem of, given (x, h), with $x \in Q(\Omega)$, finding u such that $\mathcal{F}(x, u, h) \subseteq \Omega$.

Note that, even if the prior theorems and algorithms discussed computations with vertices of one-step sets and vertices of the scheduling variable set, the argumentation in (31) can be applied to a single point x. Under the same assumptions, once x and h are known, computing a control action requires finding a feasible solution for the linear constraints $R(\sum_{i=1}^{r} h_i f_i(x) + G_k u + d_l) \leq s$, with k and l taking all values in their respective ranges.

Linear constraints arise in all affine-in-control modelling setups. Given that the solution u may not be unique, one-step optimisation may be used in implementation targeting, say, minimum $\|u\|_2$ (quadratic programming), minimum $\|u\|_1$ or $\|u\|_\infty$ (linear programming, LP), or optimal contraction, obtaining $\gamma^* := \min_u \gamma$, subject to

$$R(\sum_{i=1}^{r} h_i f_i(x) + G_k u + d_l) \leq \gamma s \tag{32}$$

Note that x and h are known in actual operation to compute $u(x, h)$; given that disturbances d are not assumed to be measurable, their extreme vertex values must be used to build the above constraints for all l ranging over the said vertices.

For instance, if Ω is gain-scheduled λ-contractive, this ensures, in the optimal contraction setup, that $\gamma^* \leq \lambda$ for all $x \in \Omega$. This is parallel to LDI cases[4,27], so details are left to the reader; an example of such computation will be used to generate the closed-loop trajectories in the examples in Section 7, with a single `linprog` statement in Matlab.

Note also that, in a general case, to avoid on-line optimisation, the theorems are constructive so they actually build gain-scheduling control laws in (23), (25), (27) from the solutions of (31) in the successive steps; details are left to the reader, for brevity.

6 | GAIN-SCHEDULED STABILISATION

In *undisturbed* LPV systems the origin is an invariant set (equilibrium point). In such a setting, it is well known that, if a compact set Ξ is gain-scheduled λ-contractive, so they are its scalings (until saturation or modelling region bounds are hit), and Ξ can be considered as the level set of a Lyapunov function proving stability with geometric decay λ, see[1,4,21,27] for details.

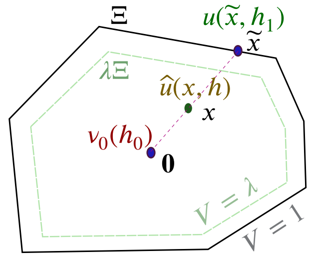

FIGURE 4 Illustration of the points and sets involved in the argumentations in the section: $\hat{u}$ is built from $u(\widetilde{x}, h_1)$ and $v_0(h_0)$.

The problem to be addressed in this section consists on generalising the idea to non-LPV set-valued embeddings $(\mathcal{F}, \mathcal{H})$, so the gain-scheduled LPV case will be a particular one of Theorem 4, the main result of this section. The S-convex stability analysis, *cf.* Proposition 5 in[21], will, too, be a particular case of the results here.

Definition 8. A pair $(\mathcal{F}, \mathcal{H})$ will be said to have a *gain-scheduled controllable equilibrium* at the origin if $0 \in \mathbb{X}$ and there exists a function $v_0 : \mathcal{H}(0) \mapsto \mathbb{U}$ such that $\mathcal{F}(0, v_0(h), h) = \{0\}$ for all $h \in \mathcal{H}(0)$, i.e., if the origin is gain-scheduled invariant.

Definition 9. A PDDI (3) is (locally) gain-scheduled stabilisable with geometric decay $0 \leq \lambda < 1$, if there exists a control law $u(x, h)$ such that:

1. There exists a a compact set Ξ, containing the origin in its interior, which is gain-scheduled invariant with a known control law $u(x, h)$, i.e., $\mathcal{F}(x, u(x, h), h) \in \Xi$ for all $x \in \Xi$, $h \in \mathcal{H}(x)$,
2. there exists an homogeneous Lyapunov function $V : \Xi \mapsto \mathbb{R}^+$, with $V(x) > 0$ for $x \neq 0$ such that, for all $x \in \Xi$, $V(\kappa x) = \kappa V(x)$, and for all $h \in \mathcal{H}(x)$, for all $x_+ \in \mathcal{F}(x, u(x, h), h)$, $V(x_+) \leq \lambda V(x)$.

Indeed, let us consider $x_0 \in \Xi$ and an admissible trajectory of the uncertain system (Def. 3) under the gain scheduled law in Definition 9, i.e., such that $x_{k+1} \in \mathcal{F}(x_k, u(x_k, h_k), h_k)$ and $h_k \in \mathcal{H}(x_k)$ and, from the above definition (item 1), $x_k \in \Xi$ for all k. If conditions in Definition 9 hold, we have $V(x_k) \leq \lambda^k V(x_0)$ for each k. Homogeneity entails that the level set $\Omega_k := \{x \in \Xi : V(x) \leq \lambda^k V(x_0)\}$ coincides with the scaling $\lambda^k \Omega_0$, so we can ensure $x_k \in \lambda^k \Omega_0$, and that is what we understand as *geometric* decay in the above definition[5]. Of course, the state converges to the origin, being its norm bounded by $\|x_k\|_2 \leq M \lambda^k \|x_0\|_2$, with $M = \max_{\xi \in \Omega_k} \frac{\|\xi\|}{\|x_0\|}$, for any k such that Ω_k lies entirely in the interior of Ξ, details left to the reader.

Assumption 1. Let us assume that there exists a compact, convex, shape-independent gain-scheduled control λ-contractive set Ξ for the PDDI (3), i.e., a control law $u(x, h)$ exists so $\mathcal{F}(x, u(x, h), h) \in \lambda\Xi$ for all $x \in \Xi$ and $h \in \mathcal{H}(x)$. Let us additionally assume that the pair $(\mathcal{F}, \mathcal{H})$ has a gain-scheduled controllable equilibrium at the origin, and $0 \in \Xi$.

In the sequel, $V(x)$ will denote the Minkowski function of the set Ξ in the above assumption, i.e.,

$$V(x) := \min \{\gamma : \gamma \geq 0,\ x \in \gamma\Xi\}, \tag{33}$$

which is, trivially, homogeneous. and plays a key role in set-based control[1].

Note that, for any $x \in \Xi$, $x \neq 0$, the point $\widetilde{x} := V^{-1}(x)x \in \Xi$ lies in the boundary of Ξ, i.e., $V(\widetilde{x}) = 1$ (an illustration on the meaning of the different points and sets appears on Figure 4).

Assumption 2. Considering that Assumption 1 holds, let us additionally assume that, for all $x \in \Xi$, $x \neq 0$, for all $h \in \mathcal{H}(x)$, being $\widetilde{x} := V^{-1}(x)x$, there exist $h_0 \in \mathcal{H}(0)$, $h_1 \in \mathcal{H}(\widetilde{x})$ such that, defining the interpolated control law:

$$\widehat{u}(x, h) := V(x) \cdot u(\widetilde{x}, h_1) + (1 - V(x)) \cdot v_0(h_0) \tag{34}$$

where $v_0(\cdot)$ comes from Definition 8, the map $\mathcal{F}$ and $\widehat{u}(x, h)$ verify:

$$\mathcal{F}(x, \widehat{u}(x, h), h) \subseteq V(x) \cdot \mathcal{F}(\widetilde{x}, u(\widetilde{x}, h_1), h_1), \tag{35}$$

Note that Assumption 2 seems a convoluted one at first glance, but it applies to some of the previously discussed modelling examples, as follows.

[5]Well, as the Lyapunov level sets must be included in Ξ, shape may change in the first instants of the trajectory: consider the scaling argumentation as an informal interpretation of Definition 9.

Proposition 5. Given a pair $(\mathcal{F}, \mathcal{H})$ having a gain-scheduled controllable equilibrium at the origin, consider Assumption 1 as true. Then, Assumption 2 holds if any of the following conditions is true:

1. Both $\mathcal{F}$ and $\mathcal{H}$ are S-convex,
2. $\mathcal{F}_h(x, u)$ is S-convex, and $\mathcal{H}(x)$ is a constant map $\mathcal{H}(x) = \mathbb{H}$.

Proof. [Case 1] Assumption 2 is fulfilled because, by S-convexity of $\mathcal{H}$, considering any $x \in \Xi$, $h \in \mathcal{H}(x)$, given that $x = V(x)\widetilde{x} + (1 - V(x)) \cdot 0$, there exist $h_0 \in \mathcal{H}(0)$, $h_1 \in \mathcal{H}(\widetilde{x})$ such that $h = (1 - V(x)) \cdot h_0 + V(x) \cdot h_1$. Then,

$$\mathcal{F}(x, \hat{u}(x, h), h) = \mathcal{F}(V(x)\widetilde{x}, V(x) \cdot u(\widetilde{x}, h_1) + (1 - V(x)) \cdot v_0(h_0), V(x) \cdot h_1 + (1 - V(x)) \cdot h_0)$$
$$\subseteq V(x) \cdot \mathcal{F}(\widetilde{x}, u(\widetilde{x}, h_1), h_1) \oplus (1 - V(x)) \cdot \mathcal{F}(0, v_0(h_0), h_0) = V(x) \cdot \mathcal{F}(\widetilde{x}, u(\widetilde{x}, h_1), h_1) \oplus \{0\} = V(x) \cdot \mathcal{F}(\widetilde{x}, u(\widetilde{x}, h_1), h_1)$$

[Case 2] Assumption 2 is fulfilled because, as $\mathcal{H}(0) = \mathcal{H}(x) = \mathcal{H}(\widetilde{x}) = \mathbb{H}$, considering $h_1 = h_0 = h \in \mathbb{H}$, we have:

$$\mathcal{F}(x, \hat{u}(x, h), h) = \mathcal{F}(V(x)\widetilde{x}, V(x) \cdot u(\widetilde{x}, h) + (1 - V(x)) \cdot v_0(h), h)$$
$$\subseteq V(x) \cdot \mathcal{F}(\widetilde{x}, u(\widetilde{x}, h), h) \oplus (1 - V(x)) \cdot \mathcal{F}(0, v_0(h), h) = V(x) \cdot \mathcal{F}(\widetilde{x}, u(\widetilde{x}, h), h) \oplus \{0\}$$

□

Theorem 4. If assumptions 1 and 2 hold, the gain-scheduled control law $\widehat{u}(x, h)$ in (34) ensures that Ξ is gain-scheduled stabilisable with geometric decay λ.

Proof. First, convexity of Ξ and $0 \in \Xi$ ensure that $\lambda\Xi \subseteq \Xi$, so the first condition (invariance) in Definition 9 is fulfilled. Let us now prove that the Minkowski function $V(x)$ in (33) is a Lyapunov function with the required properties in Definition 9. Under the definition of $V(x)$, $\kappa\Xi = \{x : V(x) \leq \kappa\}$ for any $\kappa \geq 0$. Also, $V(x)$ is homogeneous, in the sense that $V(\kappa x) = \kappa V(x)$.

Now, considering any $x \in \Xi$, from (35) and λ-contractiveness of Ξ, there exists $h_1 \in \mathcal{H}(\tilde{x})$ such that:

$$\mathcal{F}(x, \widehat{u}(x, h), h) \subseteq V(x) \cdot \mathcal{F}(\widetilde{x}, u(\widetilde{x}, h_1), h_1) \subseteq V(x) \cdot \lambda\Xi = \lambda V(x) \cdot \Xi \tag{36}$$

Given that the set $\lambda V(x) \cdot \Xi$ can be expressed as $\{\phi : V(\phi) \leq \lambda V(x)\}$, we can assert that for every $x_+ \in \mathcal{F}(x, \widehat{u}(x, h), h)$, we have $V(x_+) \leq \lambda V(x)$. □

The above result generalises well-known results prior literature, in an LPV case (corollary below) and in a non-LPV one[21]:

Corollary 3. If a compact, convex, gain-scheduled λ-contractive set Ξ containing the origin is found for the uncertain LPV model (7) with $\mathbb{D} = \{0\}$, then it is gain-scheduled stabilizable.

Proof. Indeed, $\mathcal{F}_h$ is linear, hence S-convex, $\mathcal{F}(0, 0, h) = \{0\}$, and $\mathcal{H}(x) = \Delta$, so case (2) in Proposition 5 holds, hence Assumption 2 also does. □

Corollary 4. The results in this section imply the stability results for the non-scheduled S-convex case discussed in Proposition 5 in[21].

Control law computation

Regarding actual computation of a stabilising gain-scheduled control law, as Theorem 4 is constructive, the control law (34) can be actually computed from the one which keeps the origin invariant, $v_0(h)$, and the control actions on the boundary of Ξ, i.e., $u(\widetilde{x}, h)$. Each of them can be easily computed, at least in affine-in-control cases, with the ideas in Section 5.2.

Nevertheless, from a generic point of view, once a gain-scheduled control Lyapunov function is available, computing the associated gain-scheduled state-feedback control action is straightforward; indeed, if on-line optimisation were to be used, (36) guarantees that the optimal solution, say γ^*, of the LP problem of minimising γ subject to (32) fulfills $\gamma^* \leq \lambda V(x)$ for all $x \in \Xi$. This on-line option has been the choice for the computation of closed-loop trajectories in the example in Section 7 below.

7 | NUMERICAL EXAMPLES

Example 7. This example will consider a variation of the 2nd-order nonlinear system (8) presented in Example 2, now with nonlinearity in the input channel, as follows:

$$x_+ = f(x) + g(x)u \tag{37}$$

where

$$f(x) := \begin{pmatrix} x_1 + 0.6x_2 - 1.6x_2x_1^2 \\ -1.2x_1 + 0.4x_2 \end{pmatrix} \in \mathcal{F}(x,h) = \begin{pmatrix} x_1 + 0.6x_2 - 1.6h \\ -1.2x_1 + 0.4x_2 \end{pmatrix} \tag{38}$$

and

$$g(x) := \begin{pmatrix} \frac{1.5}{1+x_2^2} \\ \frac{0.5}{1+x_2^2} - 0.1 \end{pmatrix} \tag{39}$$

with $h = x_2x_1^2$, and a control input u constrained to $\mathbb{U} := [-0.8, 0.8]$. We will consider a modeling region $\mathbb{X} = \{(x_1, x_2) : x_1 \in [-a, a], x_2 \in [-b, b]\}$, for some values of $a > 0$ and $b > 0$.

Input nonlinearity modelling. Let us first consider modeling the input-channel nonlinearity $g(x)$. As a first bound, given that $1/(1 + x_2^2)$ is a monotonic function, we can state that:

$$g(x) \in \overline{\mathbf{G}}_{[1]} := Co(\{g(0),\ g(b)\}). \tag{40}$$

Thus, considering nonlinearities in $g(x)$ as plain polytopic uncertainty, we can state $g(x)u \in \overline{\mathbf{G}}_{[1]}u$.

We will now consider a less-conservative convexification of the input-channel nonlinearity $g(x)u$, as discussed in Section 3.3. Defining:

$$\mu_2(x_2) = \frac{x_2^2(1+b^2)}{b^2(1+x_2^2)}, \qquad \mu_1(x_2) = 1 - \mu_2(x_2) \tag{41}$$

we have that $(\mu_1(x_2), \mu_2(x_2)) \in \Delta$ for every $x_2 \in [-b, b]$, and $g(x) = \mu_1(x_2)G_1 + \mu_2(x_2)G_2$, denoting $G_1 := g(0)$, $G_2 := g(b)$, as done in (16); for the moment being, this is just a rewriting of (40). However, now the functions μ can be considered to be part of the scheduling vector in state-feedback control, and we can propose a control law $u(\mu, x) := \mu_1(x_2)\bar{u}_1(x) + \mu_2(x_2)\bar{u}_2(x)$ which would correspond to using $\Upsilon(x, h) := (\mu_1\ \mu_2)$ in (15); note that, in the sequel, arguments of μ are omitted for a simpler notation. In this way, the control synthesis problem is recast to finding a suitable $\bar{u} = (\bar{u}_1\ \bar{u}_2)^T$ augmented control law.

Now, analogously to (20) and (21) in Example 5, using $d_P = d_g = d_u = 1$, a Polya expansion of degree 3 allows writing:

$$\begin{aligned} G(x)u(x) &= (\mu_1+\mu_2)(\mu_1G_1+\mu_2G_2)(\mu_1\bar{u}_1+\mu_2\bar{u}_2) \\ &= \mu_1^3G_1\bar{u}_1 + \mu_1^2\mu_2((G_1+G_2)\bar{u}_1 + G_1\bar{u}_2) + \mu_1\mu_2^2(G_2\bar{u}_1 + (G_1+G_2)\bar{u}_2) + \mu_2^3G_2\bar{u}_2 \end{aligned} \tag{42}$$

and, as $1 = (\mu_1 + \mu_2)^3 = \mu_1^3 + 3\mu_1^2\mu_2 + 3\mu_1\mu_2^2 + \mu_2^3$, following the argumentation in Proposition 2 and the referred example, we can define:

$$\overline{\mathbf{G}}_{[3]} := Co\Big([G_1\ \ 0],\quad [0\ \ G_2],\quad \frac{1}{3}[G_1+G_2\ \ G_1],\quad \frac{1}{3}[G_2\ \ G_1+G_2]\Big) \tag{43}$$

In this way, the input nonlinearity $g(x)u$ in (37) with the above-proposed expression $u(\mu, x)$ can be embedded[6] in the polytopic set-valued map:

$$g(x)\,u(\mu,x) \in \overline{\mathbf{G}}_{[3]}\begin{pmatrix} \bar{u}_1 \\ \bar{u}_2 \end{pmatrix} \tag{44}$$

Actually, the more conservative bound arising from (40) would be recovered if the control law were constrained to $\bar{u}_1 = \bar{u}_2$.

State nonlinearity modelling. The embedding options for $f(x)$ have already been considered in examples 2 and 3, arising from (38) and different boundings of $h := x_2x_1^2$. For convenience, we will recall such results here:

1. LPV bound, $h \in \mathcal{H}_{LPV} = [-bax_1, bax_1]$.
2. NLPV (S-convex) bound, $h \in \mathcal{H}_Q = [-bx_1^2, bx_1^2]$.

As discussed in Example 3, the S-convex bound is tighter than the LPV one for the same value of b. Note, additionally, that only the bound on x_2 has been used to craft the above S-convex embedding. Thus, the S-convex bound actually applies to a modelling region $\widehat{\mathbb{X}} := \{(x_1, x_2) : -b \le x_2 \le b\}$, thus x_1 being unconstrained.

[6] The bound (44) is *shape-independent*, in the sense that it holds for any μ_1 and μ_2 in the standard simplex, without any explicit knowledge of the actual "shape" of the functions in (41).

Test cases: We will compute λ-contractive sets for one LPV setup (prior literature) and three NLPV ones:

1. Gain-scheduled LPV approach (*cf.* LMIs in well-known literature, or set-based approach in[27]) with $\mathcal{F}$ from (38) and degree-3 Polya relaxation:
$$x_+ = f(x) + g(x)u(\mu, h, x) \in \mathcal{F}(x, h) \oplus \overline{\mathbf{G}}_{[3]} \cdot \bar{u}(x, h) \qquad h \in \mathcal{H}_{LPV}(x) \tag{45}$$
2. Robust (non-scheduled) CDI approach in[19],
$$x_+ = f(x) + g(x)u(x) \in \mathcal{F}_{CDI}(x) \oplus \overline{\mathbf{G}}_{[1]} \cdot u(x) \tag{46}$$
with
$$\mathcal{F}_{CDI}(x) := \begin{pmatrix} x_1 + 0.6x_2 - 1.6[-bx_1^2, bx_1^2] \\ -1.2x_1 + 0.4x_2 \end{pmatrix} \tag{47}$$
i.e., considering the nonlinearity bounds as state uncertainty.
3. Gain-scheduled CDI model, with non-augmented input:
$$x_+ = f(x) + g(x)u(h, x) \in \mathcal{F}(x, h) \oplus \overline{\mathbf{G}}_{[1]} \cdot u(x, h) \qquad h \in \mathcal{H}_Q(x) \tag{48}$$
4. Gain-scheduled CDI model, with augmented input (Polya relaxation):
$$x_+ = f(x) + g(x)u(\mu, h, x) \in \mathcal{F}(x, h) \oplus \overline{\mathbf{G}}_{[3]} \cdot \bar{u}(x, h) \qquad h \in \mathcal{H}_Q(x) \tag{49}$$

Computation of λ-contractive sets.

In order to compute shape-independent gain-scheduled contractive sets, a contraction rate of $\lambda := 0.99$ was sought, and the size of modelling region $\mathbb{X}$ was chosen by, for instance, setting ($a = 0.7$, $b = 1.35$). The used software for polyhedron manipulation was MPT3[6].

• **Test case 1 (LPV gain scheduling):** For test case 1, a first option would be finding solutions to, for instance, the LMIs, for decision variables $X = X^T \in \mathbb{R}^{2\times2}$, $F_i =\in \mathbb{R}^{2\times2}$, $i = 1, 2$, $j = 1, \dots, 4$, $k = 1, 2$:
$$\begin{pmatrix} \lambda^2 X & * \\ A_i X + B_j F_i & X \end{pmatrix} \geq 0, \qquad X_{11} \leq a^2, \quad X_{22} \leq b^2, \quad \begin{pmatrix} X & (F_i^{row\,k})^T \\ F_i^{row\,k} & 0.8^2 \end{pmatrix} \geq 0 \tag{50}$$
being B_j each of the four vertices of $\overline{\mathbf{G}}_{[3]}$ in (43), and A_i the two vertices:
$$A_1 = \begin{pmatrix} 1 - 1.6ab & 0.6 \\ -1.2 & 0.4 \end{pmatrix} \qquad A_2 = \begin{pmatrix} 1 + 1.6ab & 0.6 \\ -1.2 & 0.4 \end{pmatrix} \tag{51}$$
so conditions amount to requiring that the ellipsoid $\mathcal{E} := \{x : x^T X^{-1} x \leq 1\}$ is contained in $\mathbb{X}$ and, also, $u = F_i X^{-1} x$ does not saturate in $\mathcal{E}$. Maximisation of the geometric mean of the eigenvalues of X with YALMIP+SEDUMI[36] produced the dashed-line ellipsoid in Figure 5, labelled as 'Case 1 (LMI)'. Nevertheless, the above LMIs are given just as a simple example for the sake of illustration, because any shape-independent LMIs will yield a contractive set estimate included, by sheer definition, into the maximal gain-scheduled shape-independent λ-contractive set whose (approximate, asymptotically exact) computation is discussed in[27], in a non-LMI, set-based, approach. This justifies the use of Algorithm 1 in[27] to compute the green polyhedron in Figure 5, labelled as 'Case 1 (MPT3)'; as discussed in the cited work, it includes the convex hull of all feasible ellipsoids from the above LMIs as well as these from other parameter-dependent options (for the same Polya-relaxation complexity). Details on these LPV-specific issues are out of the scope of the present manuscript.

• **Test case 2 (non-scheduled S-convex models):** Algorithm 2 has been used to compute the cyan set in Figure 5 labelled as 'Case 2', with a non-scheduled controller, thus reducing to the proposals in[20]. Note that the cited Algorithm needs a contractive *seed set*. Such as set was obtained by classical set-based robust polytopic control[4] in a smaller modelling region (reducing the range of x_1, i.e., the value of a). Note that the green LPV set is not "comparable" to this cyan one, i.e., there is no inclusion relation that can be actually proved: neither $(\mathcal{F}, \mathcal{H})$ from (45) are an embedding of those from (46), nor vice-versa. As test cases 1 and 2 are from prior literature, details are not relevant to the discussion here.

• **Test cases 3 and 4 (gain-scheduled S-convex models):**

For test case 3, we defined:
$$\bar{h} \equiv f, \quad \bar{\mathcal{H}}(x) = \mathcal{F}_{CDI}(x), \quad \bar{\mathcal{F}}(u, \bar{h}) := \bar{h} \oplus \overline{\mathbf{G}}_{[1]} u$$

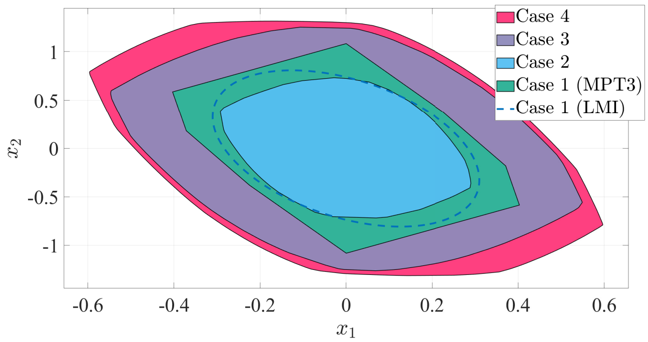

FIGURE 5 Results of the shooting/LDI algorithms with different set-valued maps and gain-scheduling options embedding the model (38)–(39).

For test case 4, the definition of $\overline{\mathcal{F}}$ is changed to $\overline{\mathcal{F}}(u, \bar{h}) := \bar{h} \oplus \overline{\mathbf{G}}_{[3]}\bar{u}$, being the rest of elements as in case 3.

Using Algorithm 2 in this work, test case 3 produced the gray contractive set in gray, and test case 4 yields the red one in Figure 5. The contractive seed sets were the same as in case 2. Given that our gain-scheduled proposal generalises the LPV and robust cases 1 and 2, the resulting contractive sets are larger than these from the mentioned cases, as expected.

Stability: Note that $\bar{\mathcal{H}}(0) = \{0\}$ and $u = 0$ keeps the state at the origin, ensuring that the proposed models have a gain-scheduled equilibrium at the origin in the sense of Definition 8. Hence, conditions in Proposition 5 hold for all 4 cases (of course, stability proofs for cases 1 and 2, once feasible LMIs or contractive sets are found, are discussed in the above-cited works). Thus, the found (gain-scheduled) control λ-contractive sets are a level set of the Minkowski Lyapunov function (33) proving closed-loop stability with their respective controllers.

Controller simulation. Regarding explicit control law computations, in this particular undisturbed example, once the Minkowski function is proven to be a gain-scheduled control Lyapunov function, computation of a control law is straightforward as solving for u such that $V(f(x) + g(x)u) \leq \gamma V(x)$ is guaranteed feasible for any $\gamma \leq 0.99$, as discussed in the central sections of this work. Now, LP comes handy to minimise γ, i.e., targeting maximum contraction. If the red contractive polytope in Figure 5 is described as $\{x : Rx \leq S\}$, control computation amounts to the following Matlab code:

```
u_ga=linprog([0 1],[R*g(x) -S],-R*f(x),[],[],[-.8 0],[.8 0.99]); u=u_ga(1);
```

Of course, in generic models with disturbances or unknown nonlinearities, constraints for LP would need to be built considering all vertices of the uncertain input set, as previously discussed in (32), instead of only $f(x)$ and $g(x)$ in the above `linprog` statement. Note, importantly, that gain scheduling is not "explicitly" visible; indeed, this roots on the same idea that well-known quasi-LPV models for nonlinear control: quasi-LPV or S-convex embeddings enable gain-scheduled controllers to be designed (assuming scheduling variables unknown at design time) but, later on, when scheduling variables are replaced by known nonlinearities they become, plainly, nonlinear controllers, and the Minkowski functions end up being standard control Lyapunov functions: gain-scheduling was merely an instrumental tool for nonlinear control.

The resulting closed-loop trajectories for 90 points on the boundary of the contractive set are depicted on Figure 6. All trajectories, after just 4 time steps reach a small neigbourhood of the origin (brown dots), faster than the proven *worst-case* geometric decay of 0.99 from uncertain embeddings, as expected.

Example 8 (3rd-order system and seed set generalisation)**.** Consider a 3rd-order nonlinear system, whose model in the region $-2.99 \leq x_3$ is given by:

$$x_{1+} = 0.8x_1 + 0.5x_2 + 0.1x_3 + \phi_1(x) + 0.1u \qquad 0 \leq \phi_1(x) \leq 0.2(x_1^2 + x_2^2) \tag{52}$$

$$x_{1+} = 0.8x_2 + 0.5x_3 + \psi_2(x) + 0.2\mu(x) \cdot u \qquad -0.1|x_1| \leq \psi_2(x) \leq 0.2e^{(x_2/3)^4} - 0.2 \tag{53}$$

$$x_{3+} = 0.9x_3 + \psi_3(x) + (1 - 0.4\mu(x)) \cdot u \qquad 0.5\log(1 + \frac{x_3}{2}) \leq \psi_3(x) \leq 0.25x_3, \quad 0 \leq \mu(x) \leq 1 \tag{54}$$

where ϕ_1 is an unknown-but-bounded nonlinearity, but $h_1 := \psi_2$, $h_2 := \psi_3$ and $h_3 := \mu$ are considered measurable so the control action can depend on them. Control action is saturated to $|u| = 1.5$. In order to convexify the product $\mu \cdot u$, Polya

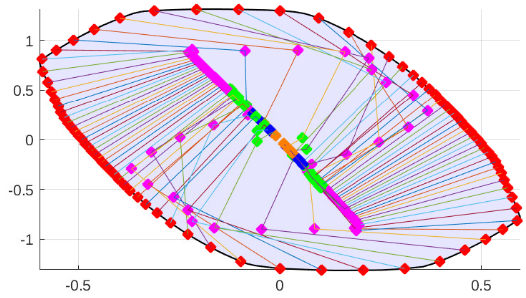

FIGURE 6 Closed-loop trajectories (model and Minkowski LF from test case **4**, *cf.* Fig. 5) with initial conditions (red) on the frontier of the computed contractive set; simulations are carried out with the original nonlinear model (38)–(39).

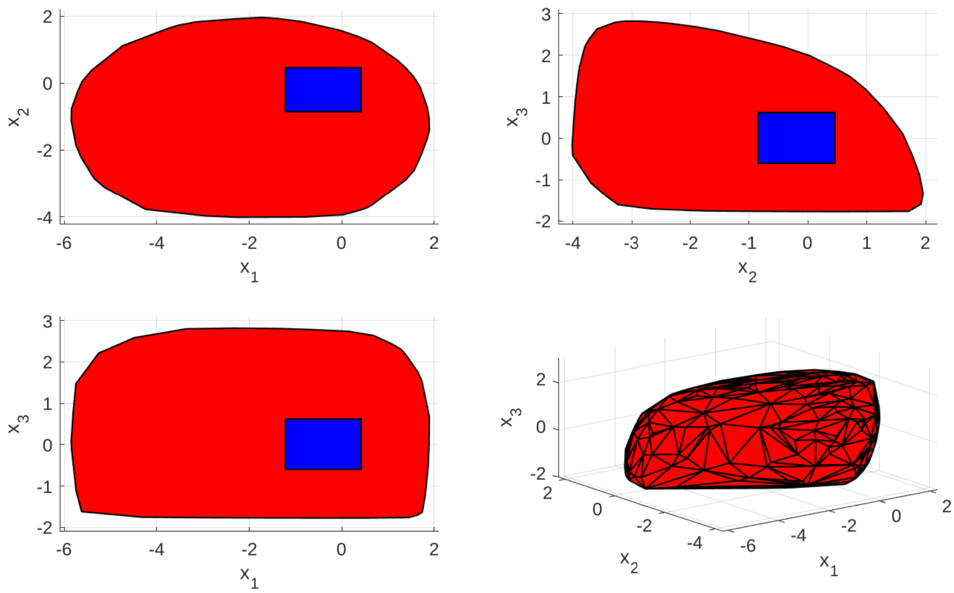

FIGURE 7 Projections and perspective view of the obtained 0.999-contractive set from Algorithm 2 in Example 8. Seed set is a prism whose projections are shown in blue.

.

relaxation with $d_u = 1$, $d_P = 0$ introduces an augmented input $(\bar{u}_1, \bar{u}_2)$ and a 3-vertex polytopic input uncertainty matrix. As all lower/upper bounds in (52)–(54) are concave/convex, the results in this paper do apply to the above model. Aiming for a 0.999-contractive set in closed-loop, after 1700 iterations of Algorithm 2, the resulting gain-scheduled contractive set (544 vertices, 1084 faces) appears in Figure 7, jointly with the initial seed[7] set (blue) Ω_0. As the uncertainty and scheduling variables (ϕ_1, ψ_2, ψ_3) are not necessarily odd functions, a non-symmetric contractive set estimate is obtained as a result of the algorithm.

[7]Note, importantly, that the chosen seed set Ω_0 (an arbitrary prism) is not proven to be gain-scheduled contractive. Nevertheless, once Algorithm 2 (fortunately) obtains a set Ω_k with Ω_0 strictly in its interior, then all vertices of Ω_k can be driven to the λ-scaling of $\Omega_{k-1} \subseteq \Omega_k$, so Ω_k is λ-contractive. Note that, in a general case (specially with disturbed systems), there is no guarantee that Algorithm 2 ends up with a set having an arbitrary seed one in its interior. Details omitted, for brevity.

8 | CONCLUSIONS

This paper has presented a gain-scheduled generalisation of set-based algorithms to compute control λ-contractive sets in prior literature, designing gain-scheduled state-feedback for discrete-time systems with nonlinear vertex models. The results in this work prove larger contractive set estimates than these prior works, due to the additional information available to compute parameter-dependent control actions (compared to robust S-convex literature) and to the less conservative convex or quasi-convex vertex models (instead of plain linear ones in gain-scheduled LPV literature). Actual computations for affine-in-control systems in the examples are based on polyhedron manipulation software MPT3, and gain-scheduled controllers are computed with linear programming code in on-line operation.

Acknowledgements

The authors are grateful to the financial support of Spanish ministry of Economy and European Union, grant DPI2016-81002-R (AEI/FEDER, UE) to this line of research.

References

1. Blanchini F. Set invariance in control. *Automatica* 1999; 35(11): 1747–1767.
2. Kouramas KI, Rakovic SV, Kerrigan EC, Allwright J, Mayne DQ. On the minimal robust positively invariant set for linear difference inclusions. In: Proc. 44th IEEE Conf. on Decision and Control, European Control Conf. ; 2005: 2296–2301.
3. Mayne DQ. Control of constrained dynamic systems. *European Journal of Control* 2001; 7(2): 87–99.
4. Kerrigan EC. *Robust constraint satisfaction: Invariant sets and predictive control*. PhD thesis. University of Cambridge, 2001.
5. Rakovic S, Grieder P, Kvasnica M, Mayne D, Morari M. Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances. *43rd IEEE Conf. on Decision and Control* 2004; 2: 1418–1423.
6. Herceg M, Kvasnica M, Jones C, Morari M. Multi-Parametric Toolbox 3.0. *Proc. of the European Control Conference* 2013: 502–510.
7. Boyd S, Vandenberghe L. *Convex optimization*. Cambridge University Press . 2004.
8. Guerra TM, Vermeiren L. LMI-based relaxed nonquadratic stabilization conditions for nonlinear systems in the Takagi-Sugeno's form. *Automatica* 2004; 40(5): 823 - 829.
9. Sala A. On the conservativeness of fuzzy and fuzzy-polynomial control of nonlinear systems. *Annual Reviews in Control* 2009; 33(1): 48–58.
10. Wu F, Dong K. Gain-scheduling control of LFT systems using parameter-dependent Lyapunov functions. *Automatica* 2006; 42(1): 39–50.
11. Sala A. Stability analysis of LPV systems: scenario approach. *Automatica* 2019; 104: 233–237.
12. Rugh WJ, Shamma JS. Research on gain scheduling. *Automatica* 2000; 36(10): 1401–1425.
13. Tanaka K, Wang HO. *Fuzzy control systems design and analysis: a linear matrix inequality approach*. John Wiley & Sons . 2004.
14. Bianchi F, Mantz R, Christiansen C. Gain scheduling control of variable-speed wind energy conversion systems using quasi-LPV models. *Control Engineering Practice* 2005; 13(2): 247–255.
15. Kwiatkowski A, Werner H. PCA-based parameter set mappings for LPV models with fewer parameters and less overbounding. *IEEE Transactions on Control Systems Technology* 2008; 16(4): 781–788.

16. Robles R, Sala A, Bernal M. Performance-Oriented Quasi-LPV Modelling of Nonlinear Systems. *Int. J. Robust Nonlinear Control* 2019; 29(5): 1230–1248.

17. Prajna S, Papachristodoulou A, Wu F. Nonlinear control synthesis by sum of squares optimization: A Lyapunov-based approach. *Proc. 5th Asian Control conference* 2004; 1: 157–165.

18. Sala A, Ariño C. Polynomial Fuzzy Models for Nonlinear Control: A Taylor Series Approach. *IEEE Transactions on Fuzzy Systems* 2009; 17(6): 1284-1295.

19. Fiacchini M. *Convex difference inclusions for systems analysis and control*. PhD thesis. Universidad de Sevilla, Spain, 2010.

20. Fiacchini M, Alamo T, Camacho EF. On the computation of convex robust control invariant sets for nonlinear systems. *Automatica* 2010; 46(8): 1334–1338.

21. Fiacchini M, Alamo T, Camacho EF. Invariant sets computation for convex difference inclusions systems. *Systems & Control Letters* 2012; 61(8): 819–826.

22. Riah R, Fiacchini M, Alamir M. Iterative method for estimating the robust domains of attraction of non-linear systems: Application to cancer chemotherapy model with parametric uncertainties. *European Journal of Control* 2019; 47: 64 - 73.

23. Rotondo D, Cristofaro A, Johansen TA, Nejjari F, Puig V. State estimation and decoupling of unknown inputs in uncertain LPV systems using interval observers. *International Journal of Control* 2018; 91(8): 1944-1961.

24. Pourasghar M, Puig V, Ocampo-Martinez C. Interval observer versus set-membership approaches for fault detection in uncertain systems using zonotopes. *Int. J. Robust Nonlinear Control* 2019; 29: 2819–2843.

25. Li J, Wang Z, Shen Y, Rodrigues M. Zonotopic fault detection observer for linear parameter-varying descriptor systems. *International Journal of Robust and Nonlinear Control*; 29(11): 3426-3445.

26. Shamma JS, Xiong D. Set-valued methods for linear parameter varying systems. *Automatica* 1999; 35(6): 1081–1089.

27. Ariño C, Sala A, Pérez E, Bedate F, Querol A. Asymptotically exact stabilization for constrained discrete Takagi–Sugeno systems via set-invariance. *Fuzzy Sets and Systems* 2017; 316: 117–138.

28. Sala A. Generalising quasi-LPV and CDI models to Quasi-Convex Difference Inclusions. *IFAC-PapersOnLine* 2017; 50(1): 7560 - 7565. 20th IFAC World Congress.

29. Sala A, Ariño C, Robles R. Gain-Scheduled Control via Convex Nonlinear Parameter Varying Models. *IFAC-PapersOnLine* 2019; 52(28): 70 - 75. 3rd IFAC Workshop on Linear Parameter Varying Systems LPVS 2019.

30. Chesi G. Estimating the domain of attraction for non-polynomial systems via LMI optimizations. *Automatica* 2009; 45(6): 1536 - 1541.

31. Sala A, Pitarch JL. Optimisation of transient and ultimate inescapable sets with polynomial boundaries for nonlinear systems. *Automatica* 2016; 73: 82–87.

32. Fiacchini M, Prieur C, Tarbouriech S, others . Necessary and sufficient conditions for invariance of convex sets for discrete-time saturated systems. *Proc. 52nd IEEE Conf. on Decision and Control* 2013.

33. Scherer CW. LMI relaxations in robust control. *European Journal of Control* 2006; 12(1): 3–29.

34. González T, Bernal M, Sala A, Aguiar B. Cancellation-Based Nonquadratic Controller Design for Nonlinear Systems via Takagi–Sugeno Models. *IEEE Transactions on Cybernetics* 2017; 47(9): 2628-2638.

35. Eppstein D. Quasiconvex Programming. *arXiv preprint* 2004; http://arxiv.org/abs/cs/0412046.

36. Löfberg J. YALMIP : A Toolbox for Modeling and Optimization in MATLAB. *Proc. IEEE Int. Conf. Robotics and Automation* 2004.