Article:

Abolpour, R., Dehghani, M. and Sad Abadi, M.S. orcid.org/0000-0002-4609-9152 (2021) Designing controller parameters of an LPV system via design space exploration. European Journal of Control, 59. pp. 47-57. ISSN 0947-3580

https://doi.org/10.1016/j.ejcon.2021.02.001

Designing Controller Parameters of an LPV System via Design Space Exploration

Roozbeh Abolpour[a], Maryam Dehghani [a], Mahdieh S. Sadabadi[b]

[a] School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran

[b] Department of Automatic Control and Systems Engineering, University of Sheffield, United Kingdom

r.abolpour@shirazu.ac.ir, mdehghani@shirazu.ac.ir, m.sadabadi@sheffield.ac.uk

Abstract- **This paper deals with the stabilizability problem of linear parameter varying (LPV) systems. It is assumed that LPV models affinely depend on time-varying uncertain and time-invariant design parameters. The uncertain parameters, their time-derivations, and design parameters belong to polygonal convex spaces. The stabilizability problem of such systems is studied. Extending the stability conditions to stabilizability conditions generally causes nonlinearity issues due to the coupling between the Lyapunov and design variables. To cope with this issue, a design space exploration algorithm (DSEA) is proposed to accurately determine the design parameters with a feasibility performance similar to stability analysis approaches. DSEA removes the undesired parts of the design subspace that cannot stabilize the model. Then, it checks the corner points of the remaining subspaces to find a stabilizing point. This procedure continues until a stabilizing point is found or the whole design subspaces are detected to be undesirable. Three hundred random LPV systems are generated to compare the feasibility performance of DSEA with existing approaches. Also, the proposed approach is used to stabilize the LPV model of a microgrid consisting of several distributed generation units and energy storage systems. The simulation results show the superiority of DSEA over the existing approaches.**

Keywords: Characteristic polynomial; design space exploration algorithm; LPV system; undesired infeasible points; Stabilizability

1. Introduction

The linear parameter varying (LPV) framework is widely applicable for equivalently or approximately modeling of time-varying or nonlinear dynamic behaviors of control systems [1]. Various methods are employed to convert control systems into LPV systems [1]. Among these methods, performing exact mathematical transformations to model nonlinear dynamics in varying parameters [2], approximating the Jacobian linearization of nonlinear systems around some equilibrium interest points [3], and exploiting data-driven identification [4] have gained the utmost importance. Recently, the LPV framework has been widely utilized in the modeling of real applications involving aviation [5], aero-elastic dynamics [6], robotics [7], and biological systems [8].

The stability analysis and control of LPV systems is of paramount interest in literature [9]-[14]. Generally, convex spaces are considered for the system's parameters and their time-differentiation to analyze the system stability or design a proper controller. Lyapunov-based approaches with parameter-dependent Lyapunov functions are frequently used for these purposes in previous studies [15]. In these methods, the candidate Lyapunov function explicitly depends on the system's time-varying parameters [15].

Various types of parameter-dependent Lyapunov functions are developed, namely affine [16]- [17], piecewise [18], and polynomial functions [19]. Although these approaches certainly include conservativeness, the negativity condition of their time-derivatives along system's trajectories is not naturally convex. This implies that the negativity conditions cannot be easily implemented by semidefinite programming

(SDP) solvers such as Linear Matrix Inequality (LMI)-based software [15]. Furthermore, some of these approaches have an infinite-dimensional issue which is related to their special Lyapunov function structure [17].

To overcome the above-mentioned difficulties, relaxation techniques have been used to convert the non-convex conditions to LMIs in parameter-dependent Lyapunov approaches [17]. Among many of them, one can list sum of squares [20], Polya's theorem [21], slack variable [22], and partitioning the uncertain space [23]. In [15], a relaxation technique is used which introduces two new slack variables and proposes LMI stability conditions based on affine parameter-dependent Lyapunov functions. However, these techniques affect the feasibility performance of the approaches as they sufficiently convert the non-convex conditions into LMIs.

The aforementioned approaches are only applicable to LPV systems in which the parameters and their time-derivative belong to convex spaces [24]. The gridding-based approaches can handle the non-convex spaces through defining the novel parameter-dependent Lyapunov functions. These approaches usually do not develop a systematic way to find their candidate Lyapunov functions. To cope with this issue, [24] exploits a Haar transform to systematically select the candidate Lyapunov functions.

The coupling between the Lyapunov and design variables is the major drawback of the previous stabilizability-based approaches. It is mainly because the stability conditions contain some coupling terms between the Lyapunov and design parameters. Hence, the final stability conditions (after the relaxation technique) are not inherently in terms of LMIs due to the mentioned coupling. To deal with this issue, previous studies have used a group of numerical approximations or LMI tricks; however, these methods usually lead to conservative results. Although several pieces of research have been carried out on the stabilizability of LPV systems, a systematic approach that leads to less conservative results is still lacking and can benefit from further research.

Motivated by aforementioned challenges, this paper develops a novel approach to deal with the stabilizability problem in LPV systems. The LPV model under study affinely depends on two groups of parameters which are time-varying uncertain and design parameters. The uncertain and design parameters are assumed to belong to uncertain and design spaces which are convex polygonal. The main aim of this paper is to systematically determine design parameters inside the given design space that stabilizes the LPV model. To this end, an algorithm is proposed that iteratively searches the design space to find an appropriate design vector. The algorithm is known as design space exploration algorithm (DSEA) in this paper.

It is worth mentioning that the idea of removing the undesired parts of the design space enables us to overcome the coupling issue between the Lyapunov and design variables. Furthermore, DSEA only removes the undesired parts. This implies that DSEA does not omit any stabilizing point. A theorem is presented to mathematically investigate DSEA's convergence in this paper.

The proposed algorithm is directly applicable to a wide range of LPV systems whose closed loop system's matrices affinely depend on the uncertain and design parameters. The algorithm can be used to solve all LPV control problems in which controllers do not explicitly depend on the uncertain parameters. However, the method can be extended to systems with non-affine matrices using an over-parameterization technique.

The rest of this paper is organized as follows. Section 2 presents the LPV system under study and its related definitions and assumptions. Section 3 presents the related preliminaries of the DSEA. Section 4 proposes the main contributions of this paper which is the DSEA. The simulation examples are presented in Section 5. Finally, Section 6 concludes the paper.

Notations: The basic notations of this paper are presented in Table 1.

Table 1. Basic notations used in this paper

Notation	Definition
R	Real number space.
$R^{n\times m}$	Space of real matrices with n rows and m columns.
R^{+}	Space of positive real numbers.
$\rho(t)$	Time-varying parameters of LPV models.
α	Design parameters of LPV models.
Ω_0	Space of time-varying parameters.
Ω_1	Space of time-derivative of time-varying parameters.
ϕ	Primary design space.
$A(\rho(t),\alpha)$	State matrix of LPV models.
$d(s,\rho,\alpha)$	Characteristic polynomial of LPV models, considering ρ as a constant
$co\{\cdot,\dots,\cdot\}$	Convex combination operator.

2. System description

Consider the following linear parameter varying (LPV) system:

$$\dot{x}(t) = A(\rho(t),\alpha)x(t) \tag{1}$$

where $x(t) \in \mathbb{R}^n$ is the state vector and n is the number of states. $\rho(t) \in \mathbb{R}^p$ denotes the time-varying parameters that are assumed to belong to a convex polygonal space $\Omega_0 \subset \mathbb{R}^p$ which is known as the parameter space. Additionally, the time-derivative of these parameters $\dot{\rho}(t)$ are assumed to be inside another convex polygonal space $\Omega_1 \subset \mathbb{R}^p$ that is known as time-derivative parameter space.

In (1), $\alpha \in \mathbb{R}^d$ is a vector of some unknown and time-invariant design parameters that should be precisely adjusted to establish LPV system's stability. It is assumed that α belongs to a given convex polygonal space $\phi \subset \mathbb{R}^d$ that is denoted by the design space.

In (1), the system matrix $A(\rho(t), \alpha) \in \mathbb{R}^{n \times n}$ is assumed to be an affine function of the time-varying parameters $\rho(t)$ that can be written as follows:

$$A(\rho(t), \alpha) = A_0(\alpha) + \sum_{i=1}^{p} A_i(\alpha)\rho_i(t) \tag{2}$$

where ρ_i is the i^{th} entry of ρ and $\{A_i(\alpha)\}_{i=0}^{p} \subset \mathbb{R}^{n \times n}$ is a set of matrix functions that affinely depend on the design parameters α.

Obviously, for the stability analysis of this system, we need to show the system (1) is stable for time-varying $\rho(t)$. This paper utilizes the concept of instability since the instability analysis of system (1) can be shown even if the system is unstable for a specific constant $\bar{\rho}$. This means, if we demonstrate that the system is unstable for a special constant uncertain point, we can conclude that special point is totally unstable. Let $d(s, \bar{\rho}, \alpha) = |sI_n - A(\bar{\rho}, \alpha)|$ be the system's characteristic polynomial that can be defined upon the consideration of ρ and α which are time-independent.

To summarize the assumptions of LPV model (1), the model's conditions are briefly listed in the following:

- Matrix $A(\rho(t), \alpha)$ should be an affine function of system's parameters $\rho(t)$.
- Matrix $A(\rho(t), \alpha)$ should affinely depend on design parameters α.

Remark 1. The results of this paper are also applicable to state matrices $A(\rho(t), \alpha)$ that does not affinely depend on $\rho(t)$. In fact, the nonlinear terms of ρ in $A(\rho(t), \alpha)$ can be replaced by members of an over-parameterized vector $\bar{\rho}(t)$ that are appropriately defined based on ρ. In this case, spaces Ω_0 and Ω_1 should be redefined based on the over-parameterized vector $\bar{\rho}(t)$ such that $\bar{\rho} \in \bar{\Omega}_0$ and $\dot{\bar{\rho}} \in \bar{\Omega}_1$.

3. DSEA's preliminaries

DSEA searches a given design space to find a feasible point which is able to stabilize the LPV model. DSEA considers an initial design convex space and checks the feasibility of its corner points. The algorithm detects the undesired parts of the design space (the subspaces that certainly do not include any feasible point) and divides the remaining into smaller subspaces. DSEA iteratively continues until finding a feasible corner point or all design subspaces are detected to be undesirable.

To detect the undesired parts, two indicators are exploited which are Indicator Point (IP) and Admissible Closed Path (ACP). These indicators enable DSEA to detect the undesirability of a design subspace that are described in this section.

An IP is a specific point in the uncertain space such as $\bar{\rho} \in \Omega_0$ which is constant and fixed. In fact, a design subspace can be concluded to be undesirable if the LPV model is unstable for all its internal points considering $\rho(t) = \bar{\rho}$ for all times. An IP is used to detect the undesirability of a design space through the characteristic polynomial concept.

ACP is a periodic closed path in the uncertain space Ω_0 which its time-derivation belongs to Ω_1, as well. A continuous function is considered as a candidate that increments at each period of an ACP. This result can directly conclude the instability of the delay model which is described in the next section. In fact, an ACP is considered to detect the undesirability of a design subspace via exploiting these ACPs.

In the following, the ACP is introduced. The time-varying vector $\rho(t)$ can freely move inside the parameter space Ω_0 such that $\dot{\rho}(t) \in \Omega_1$. The ACP is defined exactly in the following definition.

Definition 1. *Periodic continuous function* $C: \mathbb{R}_+ \to \Omega_0$ *is an ACP if* $\dot{C}(t)$ *belongs to* Ω_1 *for all* $t \in \mathbb{R}_+$.

According to Definition 1, an ACP determines an admissible path that belongs to the defined Ω_0 and its derivative has some important properties including periodicity, continuity and admissibility. To clarify the ACP concept, the following example is provided. Consider an arbitrary LPV system (1) with the following spaces Ω_0 and Ω_1:

$$\Omega_0 = \{\rho \in R^2 | \ \ \rho^T \rho \leq 4\} \tag{3}$$

$$\Omega_1 = co\{[-10 \ \ -10]^T, [-10 \ \ 10]^T, [10 \ \ -10]^T, [10 \ \ 10]^T\} \tag{4}$$

Then, two different ACPs are presented in the following:

$$C_1(t) = \{\rho \in R^2 | \ \ \rho_1 = 1.5\cos(2\pi t), \ \ \rho_2 = 1.5\sin(2\pi t)\} \tag{5}$$

$$C_2(t) = \begin{cases} [-1 \ \ 2r_t^2 - 1]^T, & 0 \leq r_t < 1 \\ [2(r_t - 1)^2 - 1 \ \ 1]^T, & 1 \leq r_t < 2 \\ [1 \ \ 1 - 2(r_t - 2)^2]^T \ \ , & 2 \leq r_t < 3 \\ [1 - 2(r_t - 3)^2 \ \ -1]^T, & 3 \leq r_t < 4 \end{cases} \tag{6}$$

where r_t is the reminder after division of t by 4. Paths (5) and (6) satisfy the ACP's properties shown on Figs. 1 and 2 that indicate C_1 and C_2 and their time-derivatives:

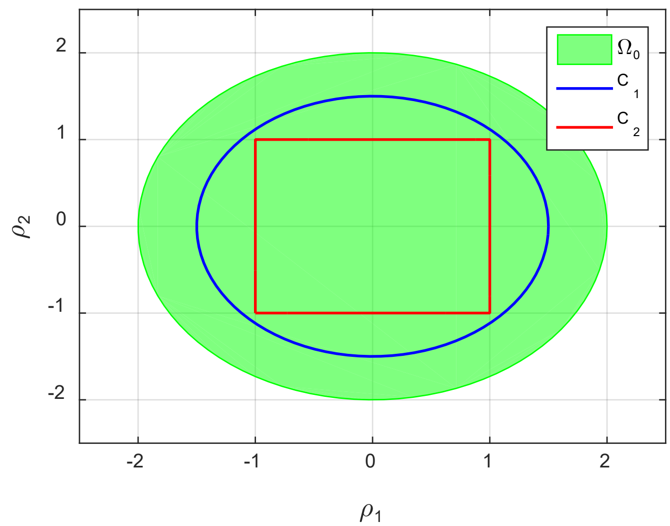

Fig. 1. Parameter space (3) and ACPs C_1 and C_2 defined in (5) and (6)

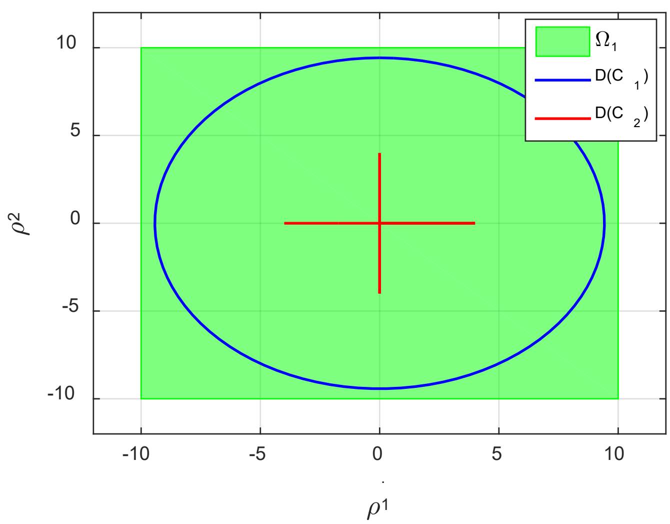

Fig. 2. Parameter time-derivative space (4) and time-derivatives of ACPs C_1 and C_2 defined in (5) and (6)

Assume C and $\rho(t)$ are an ACP and its arbitrary member at instant time t. Then, the time-derivative vector $\dot{\rho}(t)$ can be explicitly obtained based on $\rho(t)$. This means $\dot{\rho}$ only depends on ρ for all $\rho \in C$ and this relation can be easily obtained upon derivation of ACP C. For instance, consider ACP C_1 defined in (5), in which $\dot{\rho}$ equals to $[-2\pi\rho_2 \quad 2\pi\rho_1]^T$ for all $\rho = [\rho_1 \quad \rho_2]^T$ inside C_1.

4. Proposed Approach for Control Parameters Design

This section finally proposes the DSEA as a main contribution. DSEA initially meshes the design space to obtain a set of small simplexes which are known as design subspaces. Then, it checks the stabilizability status of the corner points of each design subspace. If there exists a corner point that stabilizes the model, the algorithm terminates (note that, it suffices to check stability of the LPV model at a specific and known design vector which is a stability analysis problem). Otherwise, DSEA checks the undesirability of the design subspaces and removes the undesired subspaces and divides the rest of them into two smaller sub-simplexes (If a design subspace does not contain any stabilizing point, it is labeled to be undesirable). DSEA repeats these steps to reach a stabilizable corner deign point or the whole generated subspaces are detected to be undesirable.

The main idea of this paper is to determine the design parameters via eliminating the undesired parts and checking the corner points of the rest. The undesired parts are the design subspaces of the original design space ϕ that do not surely contain any stabilizing point.

To detect the undesirability of an individual design subspace, two indicators are used which are noted by indicator point (IP) and ACP. An IP is a specific time-invariant vector in the uncertain space that assesses the total instability of the LPV model for the design subspace. The other indicator is ACP which is an allowable closed path in the uncertain space such that its time-derivative belongs to the supposed the time-derivative uncertain space. The IPs and ACPs aim to detect the undesirability of a design subspace with constant and periodic trajectories in the uncertain space. If they successfully detect the undesirability of a design subspace, the design space is instantly removed. Otherwise, the subspace is halved to two smaller subspaces that will be checked in the next algorithm's iterations. It is apparent that the failure of these indicators to exactly detect the undesirability of design subspace influences the computational time of the algorithm, but not its feasibility performance. The indicators should be simple, since they are evaluated for all generated design subspaces and they should have an acceptable level of conservativeness to control the computational time of the algorithm. In order to achieve a trade-off between the level of conservativeness and the computational burden, the above-mentioned indicators are considered in this paper.

4.1. Undesirability checking via IPs

The LPV system (1) will be converted into an LTI system through the following assumptions. First, vector $\rho(t)$ is assumed to be time-invariant that equals to $\bar{\rho}$ inside Ω_0. Second, vector α is assumed to belong to a convex polygonal subspace φ inside ϕ, (i.e. $\varphi \subset \phi$). Using these assumptions, the LPV system (1) can be converted into the following uncertain LTI system:

$$\dot{x}(t) = A(\bar{\rho}, \alpha)x(t) \colon \alpha \in \varphi \tag{7}$$

It should be emphasized that α plays the uncertainty role in the LTI system (7). Therefore, α is noted by the uncertain vector to highlight the uncertain nature of the LTI system (7). It is assumed that the characteristic polynomial of LTI system (7) affinely depends on the uncertain parameters. This means that the characteristic polynomial $d(s, \bar{\rho}, \alpha) = |sI_n - A(\bar{\rho}, \alpha)|$ is an affine function of the uncertain vector α.

Remark 2. The characteristic polynomial of the closed loop system requires to be an affine function of the uncertain parameters only to exploit the IPs. It is not needed to be satisfied for all cases. In fact, ACPs can be used to detect the undesirability whenever the characteristic polynomial is not affine.

The uncertain LTI system (7) is totally unstable if and only if $d(s,\bar{\rho},\alpha)$ is a non-Hurwitz polynomial for all $\alpha \in \varphi$. Hence, the main aim of this subsection will be followed through Theorem 1 that proposes a set of conditions to evaluate total non-Hurwitzness of $d(s,\bar{\rho},\alpha)$. Before we proceed with Theorem 1, Definition 2 and Lemma 1 are presented.

Definition 2. *A real coefficient polynomial is said to be critical if it has pure imaginary roots.*

Lemma 1 [25]**.** *Assume $p(s)$ and $q(s)$ are two real coefficient polynomials with the same orders. There exists a critical polynomial on the polynomial segment $co\{p(s),q(s)\}$ if and only if there exists a real number ω that satisfies the following conditions:*

$$p_r(\omega)q_i(\omega) = p_i(\omega)q_r(\omega) \tag{8}$$

$$p_r(\omega)q_r(\omega) + p_i(\omega)q_i(\omega) < 0 \tag{9}$$

Based on Definition 2 and Lemma 1, Theorem 1 will be presented in the following.

Theorem 1 [25]**.** *The characteristic polynomial $d(s,\bar{\rho},\alpha) = |sI_n - A(\bar{\rho},\alpha)|$ is totally non-Hurwitz for all $\alpha \in \varphi$ (φ is a convex polygonal subspace of ϕ) if the following conditions hold:*

$$\forall\alpha \in \partial_c(\varphi): \quad d(s,\bar{\rho},\alpha)\ \ is\ not\ Hurwitz \tag{10}$$

$$\forall(\alpha_1,\alpha_2) \in \partial_e(\varphi):$$

$$co\{d(s,\bar{\rho},\alpha_1), d(s,\bar{\rho},\alpha_2)\}\ \ does\ not\ contain\ any\ critical\ polynomial \tag{11}$$

Remark 3. Conditions (10) and (11) can be easily investigated by computing roots of corner polynomial $\{d(s,\bar{\rho},\alpha)\}_{\alpha\in\partial_c(\Omega_0)}$ and checking the existence of any critical polynomial on exposed edges of $\{co\{d(s,\bar{\rho},\alpha_1),d(s,\bar{\rho},\alpha_2)\}\}_{(\alpha_1,\alpha_2)\in\partial_e(\varphi)}$ via applying (8) and (9) in Lemma 1.

Theorem 1 can check the total non-Hurwitzness of the characteristic polynomial $d(s,\bar{\rho},\alpha)$ that means it can assess the total unstability of uncertain LTI system (7).

Theorem 2 proposes conditions to check the undesirability of design subspace $\varphi \subset \phi$ for $\bar{\rho}$.

Theorem 2. *Design subspace* φ *is* undesirable *for LPV model (1) if there exists* $\bar{\rho} \in \Omega_0$ *that holds the following condition:*

$$\forall\alpha \in \varphi: \quad A(\bar{\rho},\alpha) \; is \; unstable \tag{12}$$

Proof. Assume $\bar{\rho} \in \Omega_0$ forces $A(\bar{\rho},\alpha)$ to be unstable for all $\alpha \in \varphi$. Obviously, LPV system (1) is converted into an LTI system at special case $\rho(t) = \bar{\rho}$. In this special case, system $\dot{x}(t) = A(\bar{\rho},\alpha)x(t)$ will be entirely unstable for all $\alpha \in \varphi$ because its system matrix $A(\bar{\rho},\alpha)$ is supposed to be unstable. This fact implies the statement of the theorem and completes the proof.

Remark 4. The feasibility of the condition given in (12) can be investigated by Theorem 1. In fact, this condition is applicable via checking the corners and exposed edges of φ based on Theorem 1.

4.2. Undesirability checking via ACPs

This subsection exploits the ACPs to detect the undesired parts of the design space. For this purpose, Theorem 3 is presented in the following.

Theorem 3. *Assume* C *is an ACP with a period of* T *and* $D \subset \mathbb{R}^n$ *is a compact space that contains the origin. Then, design subspace* φ *is undesirable for the LPV model*

(1) if there exist continuous positive function $V(x,\rho) \in D \times C \to \mathbb{R}_+$ and continuous function $W(x,\rho) \in D \times C \to \mathbb{R}$ that satisfy the following conditions:

$$\forall t \in [0,T]:\ \ V\big(0, C(t)\big) = 0 \tag{13}$$

$$\forall x \in D\backslash\{0\}:\ \ \int_0^T W\big(x, C(t)\big)dt > 0 \tag{14}$$

$$\forall \alpha \in \varphi, \forall x \in D, \forall t \in [0,T]: \frac{\partial V\big(x, C(t)\big)}{\partial x} A(C(t), \alpha)x + \frac{\partial V\big(x, C(t)\big)}{\partial \rho} \dot{C}(t) \geq W(x, C(t)) \tag{15}$$

Note that $x(0) \neq 0$.

Proof. Assume $S_\varepsilon = \{x \in R^n |\ \|x\| \leq \varepsilon\}$ is a hyper-sphere around the origin with radius $\varepsilon > 0$. Obviously, one can find a sufficient small number ε such that the following conditions hold:

$$S_\varepsilon \subset D \tag{16}$$

$$x(0) \notin S_\varepsilon \tag{17}$$

$$\forall x \in S_\varepsilon, \forall t \in [0,T]: \quad V\big(x, C(t)\big) \leq V\big(x(0), \rho(0)\big) \tag{18}$$

Note that since D is a compact space around the origin, $V(x,\rho)$ is continuous over $D \times C$ and (13).

Assume the LPV model is asymptotically stable which implies system's trajectory $x(t)$ asymptotically reaches the origin. Hence, system's trajectory reaches the hyper-sphere S_ε in a finite period of time that is denoted by τ in this proof. It results in the following inequality considering $\rho(t) = C(t)$ for all $t \in [kT, kT+T]$ and $k \in \{1,2,3 \ldots\}$:

$$\forall t \geq \tau: \quad x \in S_\varepsilon \to\ V(x, \rho(t)) \leq V\big(x(0), \rho(0)\big) \tag{19}$$

Now, assume k is an arbitrary positive integer number. Using (15), one can easily obtain the following inequality:

$$\int_{kT}^{kT+T} \dot{V}\big(x(t), \rho(t)\big)dt \geq \int_{kT}^{kT+T} W\big(x(t), \rho(t)\big)dt \tag{20}$$

Inequality (20) obviously implies the following condition:

$$V\big(x(kT+T), \rho(kT+T)\big) - V\big(x(kT), \rho(kT)\big) \geq \int_{kT}^{kT+T} W\big(x(t), \rho(t)\big)dt \tag{21}$$

According to (14) and (21), one obtains that $V\big(x(kT), \rho(kT)\big) \geq V\big(x(0), \rho(0)\big)$. Let $k > \frac{\tau}{T}$ that implies $V\big(x(kT), \rho(kT)\big)$ is larger than $V\big(x(0), \rho(0)\big)$ which clearly contradicts (19). □

Remark 5. It is worth mentioning that the continuous function $V(x, \rho)$ in the statement of Theorem 3 is not needed to be radially unbounded. Additionally, the compact space D can be a very small space which may reduce the conservativeness of conditions (13-15) in some special cases.

The periodic nature of the ACPs guarantees that the system's state vector does not reach the origin at all. The reason is that the candidate Lyapunov function is positive everywhere except at the origin of the state space model and it increases at all points of the ACP's trajectory in the uncertain space. It should be noted that the ACP should satisfy conditions $\rho \in \Omega_0$ and $\dot{\rho} \in \Omega_1$ that suffices to be checked only at one period of the ACP.

Theorem 3 proposes conditions to investigate the undesirability of a given design subspace φ. Notice that, conditions (13-15) cannot be directly investigated by mathematical tools due to their generality issue. To cope with this issue, another lemma is presented that relaxes these conditions to obtain required LMIs by considering a special function form.

Lemma 2. *Assume C is an ACP with $T > 0$ as its period. Then, design subspace $\varphi \in \phi$ is undesirable for the LPV model (1) if there exist a sufficiently large integer number N and symmetric matrices $\{P_i\}_{i=0}^{p} \in \mathbb{R}^{n\times n}$ and $\{X_i\}_{i=0}^{p} \in \mathbb{R}^{n\times n}$ such that following conditions are satisfied:*

$$\forall t \in \left\{\frac{T}{N}, \frac{2T}{N}, \dots, T\right\}: \quad P_0 + \sum_{i=1}^{p} P_i C_i(t) > 0 \tag{22}$$

$$\forall \alpha \in \partial_c(\varphi), \forall j \in \{1, \dots, N\}, \forall t, \bar{t} \in \left\{\frac{(j-1)T}{N}, \frac{jT}{N}\right\}:$$

$$He\left\{\left(P_0 + \sum_{i=1}^{p} P_i C_i(t)\right) A(C(\bar{t}), \alpha)\right\} + \sum_{i=1}^{p} P_i \dot{C}_i(t) > X_0 + \sum_{i=1}^{p} X_i C_i(t) \tag{23}$$

$$X_0 T + \sum_{i=1}^{p} X_i \int_0^T C_i(t) dt \geq 0 \tag{24}$$

where $C_i(t)$ is the i^{th} entry of ACP C.

Proof. It is obvious that conditions (22-23) will result in the following conditions for a sufficiently large integer N:

$$\forall t \in [0, T]: \quad P_0 + \sum_{i=1}^{p} P_i C_i(t) > 0 \tag{25}$$

$$\forall \alpha \in \partial_c(\varphi), \forall t \in [0, T]:$$

$$He\left\{\left(P_0 + \sum_{i=1}^{p} P_i C_i(t)\right) A(C(t), \alpha)\right\} + \sum_{i=1}^{p} P_i \dot{C}_i(t) > X_0 + \sum_{i=1}^{p} X_i C_i(t) \tag{26}$$

Let $V(x, \rho) = x^T\left(P_0 + \sum_{i=1}^{p} P_i \rho_i\right)x$ and $W(x, \rho) = x^T\left(X_0 + \sum_{i=1}^{p} X_i \rho_i\right)x$ that are defined over $\mathbb{R}^n \times C$. Clearly, conditions (22) and (25-26) imply (13-15), that means the undesirability of the design subspace φ based on Theorem 3.

4.3. Stabilizability checking of a design point

This section investigates the stability of an individual point in the design space. Indeed, it has been investigated whether the special point $\alpha \in \phi$ is able to stabilize the LPV system (1) through an existing theorem. The theorem suggests a set of stability conditions for this purpose based on LMIs.

Theorem 4 [26]**.** *The special point $\bar{\alpha} \in \phi$ is a stabilizing point for LPV system (1) if and only if there exists a symmetric matrix function $P(\rho) \in \Omega_0 \to \mathbb{R}^{n\times n}$ that satisfies the following conditions:*

$$\forall\rho \in \Omega_0: \quad P(\rho) > 0 \tag{27}$$

$$\forall\rho \in \Omega_0, \forall\dot{\rho} \in \Omega_1: \quad He\{P(\rho)A(\rho,\bar{\alpha})\} + \sum_{i=1}^{p} \frac{\partial P(\rho)}{\partial \rho_i}\dot{\rho}_i < 0 \tag{28}$$

Since the matrix function $P(\rho)$ has a general form, it is not possible to check conditions (27-28) by convenient mathematical solvers. To overcome this problem, the following lemma is presented that relaxes these conditions by considering a special form of the matrix function $P(\rho)$. In the following lemma, an over-parameterized vector $\bar{\rho}$ is used to cope with the existent bilinear terms of the uncertain parameters ρ in (28).

Lemma 3. *The special point $\alpha \in \phi$ is a stabilizing point for LPV model (1) if there exists a set of symmetric matrices $\{P_i\}_{i=0}^{p}$ that hold the following conditions:*

$$\forall\rho \in \partial_c(\Omega_0): \quad P_0 + \sum_{i=1}^{p} P_i\rho_i > 0 \tag{29}$$

$$\forall\rho,\bar{\rho} \in \partial_c(\Omega_0), \forall\dot{\rho} \in \partial_c(\Omega_1):$$

$$He\left\{\left(P_0 + \sum_{i=1}^{p} P_i\rho_i\right)A(\bar{\rho},\alpha)\right\} + \sum_{i=1}^{p} P_i\dot{\rho}_i < 0 \tag{30}$$

Proof. Assume $P(\rho) = P_0 + \sum_{i=1}^{p} P_i\rho_i$. Then, conditions (29)-(30) directly imply conditions (27)-(28) in Theorem 4. According to Theorem 4, α is a stabilizing point for the LPV model.

4.4. DSEA presentation

The DSEA iteratively eliminates the undesired parts of the design space ϕ until it finds a stabilizing point. DSEA exploits Theorem 2 and Lemma 2 to determine whether a design subspace is undesirable or not. The undesired parts are omitted and DSEA searches the remained parts. Before presenting the DSEA's steps, the following definition should be mentioned:

Definition 3. *Set* $IP = \{\rho^{(i)}\}_{i=1}^{N} \subset \Omega_0$ *and* $IC = \{C^{(i)}\}_{i=1}^{M}$ *contain the indicator points and indicator ACPs in which* $\rho^{(i)} \in \Omega_0$ *and* $C^{(i)} \subset \Omega_0$ *are the* i^{th} *indicator point and* i^{th} *ACP.*

Then, DSEA's steps can be briefly presented in the following:

Design Space Exploration Algorithm (DSEA):

1. *Initialize design space* ϕ *and sets* IP *and* IC.
2. *Set* $D = \{\phi\}$, $\varphi^{(0)} = \phi$ *and* $k = 0$.
3. *If* $k > |D|$, *return "there is not any stabilizing points.".*
4. *If there exists a stabilizing point in the corner set of* $\varphi^{(k)}$ *based on Lemma 3, return this point as the feasible solution.*
5. *Check the existence of any* $\rho \in IP$ *or* $C \in IC$ *that guarantee the undesirability of* $\varphi^{(k)}$ *for LPV model (1) based on Theorem 2 and Lemma 2, respectively.*
6. *If* $\varphi^{(k)}$ *is detected to be undesirable or volume of* $\varphi^{(k)}$ *is smaller than threshold value* δ, *set* $k = k + 1$ *and go to Step 3.*
7. *Divide* $\varphi^{(k)}$ *into two smaller convex polygonal subspaces* $\varphi_1^{(k)}$ *and* $\varphi_2^{(k)}$.
8. *Insert* $\varphi_1^{(k)}$ *and* $\varphi_1^{(k)}$ *into* D, *set* $k = k + 1$ *and go to Step 3.*

Furthermore, a flowchart is provided to schematically describe the procedure of the DSEA that is shown in Fig. 3.

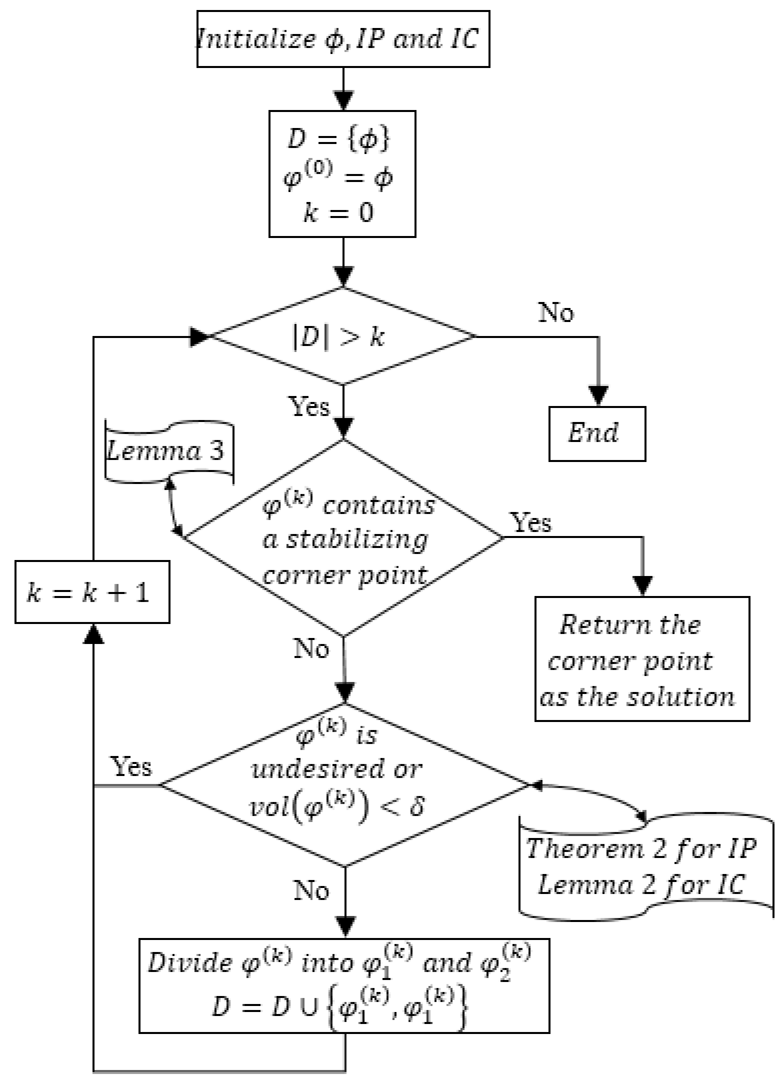

Fig. 3. DSEA's flowchart

The DSEA is initialized by assumptions of ϕ, IP and IC to be the design space, indicator point and indicator ACP sets in Step 1. In Step 2, $D = \{\phi\}$, $\varphi^{(0)} = \phi$ and $k = 0$ which are the set of generated convex-polygonal subspaces, current design subspace and the iteration number. Note that, each iteration separately investigates one design subspace to detect its undesirability or existence of at least one stabilizing corner point. This design space is called the current design subspace.

Set D contains the whole generated design subspaces in all previous iterations. Notation $\varphi^{(k)}$ stands for the current design subspace in the current iteration that is indexed by k. Step 3 checks the existence of any design subspace in D that is not already investigated in previous iterations. If the whole subspaces are undesirable, the algorithm returns "there is not any feasible solution." and terminates. Otherwise, the algorithm goes to Step 4 to check the existence of any indicator point in IP or any ACP in IC that can establish the undesirability of current design subspace $\varphi^{(k)}$. If the current design subspace is detected to be undesirable or its volume is smaller than threshold value δ, DSEA increases k by one and goes to Step 3. Otherwise, $\varphi^{(k)}$ will be divided into two smaller sub-simplexes $\varphi_1^{(k)}$ and $\varphi_2^{(k)}$, the generated subspaces $\varphi_1^{(k)}$ and $\varphi_2^{(k)}$ are added to D and DSEA goes to Step 3.

It is worth mentioning that the algorithm will certainly terminate because the volume of the generated design subspaces is limited by threshold value δ. It forces the new subspaces to be larger than a specific volume to guarantee the termination of the algorithm.

Finally, Theorem 5 analyzes the DSEA's convergence through proving its ability to find any existing stabilizing point in the primary design space ϕ. In fact, this theorem proves that if design space ϕ contains any stabilizing point, the algorithm will find it. This convergence analysis mathematically reveals DSEA's feasibility performance.

Theorem 5. *Assume there exists convex polygonal R in the primary design space* ϕ *such that its internal points can be detected to be stabilizing by Lemma 3. If volume of R is larger than* δ*, then DSEA assuredly reaches a stabilizing point (the reach point may be inside R or not).*

Proof. It is apparent that DSEA will terminate if one of these two possible cases occur which are discussed in the following. First, DSEA converges to a stabilizing point that reveals the correctness of the theorem. Second, DSEA may not find any feasible solution in its iteration. In this case, the whole design space ϕ will be divided into smaller subspaces which are shown by notations S_1 and S_2. Set S_1 contains the generated subspaces that are detected to be undesirable by DSEA. Set S_2 includes the subspaces which their volumes are smaller than δ. Notice that, the whole subspaces are generated in DSEA's iterations before it stops. Since R is a subspace of ϕ which consists of stabilizing points, none of S_1's subspaces belong to R. Hence, space R is fully included in the union of S_2's subspaces. It clearly implies the following result:

$$R \subset \bigcup_{\varphi \in S_2} \varphi \tag{31}$$

Subspace R cannot fully belong to only one of the subspaces of S_2. It is mainly because volume of S_1's subspaces is smaller than δ, whereas volume of R is larger than δ based on the hypothesis of the theorem. Therefore, subspace R will certainly contain at-least one corner point of one subspace of S_2. Let $\varphi \in S_2$ and $\alpha \in \partial_c(\varphi)$ be the subspace and its corner point that is concluded to be inside R. Since R contains only the points that can be detected to be stabilizing by Lemma 3 and $\alpha \in R$, this corner point should be detected to be stabilizing, as well.

It should be noted that the whole corner points of the generated subspaces including α was checked to be stabilizing by Lemma 3 in DSEA's iterations. Hence, corner point α was checked and detected to be not stabilizing by Lemma 3. Clearly, this fact contradicts the membership of α to R and it proves the theorem.

Remark 6. DSEA's feasibility performance only depends on the feasibility performance of Lemma 3. Indeed, DSEA can totally overcome the nonlinearity issue which rises whenever the stability conditions are extended to stabilizability conditions for LPV models (the coupling between the Lyapunov and design variables). Thus, proposing a new stability condition that is less conservative than Lemma 3 can directly improve the DSEA's feasibility performance via replacing Lemma 3 by the new one in DSEA's steps.

5. Simulation results

Three detailed simulation examples are provided in this section. Example 1 illustrates the notations and basic definitions of this paper for a sample LPV model. Example 2 and 3 compare the feasibility performance of the suggested algorithm to some previous approaches.

Example 1. Consider the following LPV model:

$$\dot{x} = (A_0(\alpha) + A_1\rho_1 + A_2\rho_2)x \tag{32}$$

where $\rho = [\rho_1 \quad \rho_2]^T$ is the vector of time-varying uncertain parameters, α is the vector of design parameters that should be precisely adjusted to establish LPV system's stability and $x \in \mathbb{R}^2$ is the state vector. Matrices $A_0(\alpha)$, A_1 and A_2 are supposed to be as follows:

$$A_0(\alpha) = \begin{bmatrix} 0 & 1 \\ \alpha_1 & \alpha_2 \end{bmatrix}, \ A_1 = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}, \ A_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{33}$$

In this example, the design spaces $\Omega_0 \subset \mathbb{R}^2$ and $\Omega_1 \subset \mathbb{R}^2$ are assumed to be as given below:

$$\Omega_0 = co\begin{Bmatrix} [-0.4 \quad -0.4]^T, [-0.4 \quad 0.4]^T, \\ [0.4 \quad -0.4]^T, [0.4 \quad 0.4]^T \end{Bmatrix} \tag{34}$$

$$\Omega_1 = co\begin{Bmatrix} [-2 \quad -2\,]^T, [-2 \quad 2]^T, \\ [2 \quad -2]^T, [2 \quad 2]^T \end{Bmatrix} \tag{35}$$

Also, the design space is supposed to be as follows, in this example:

$$\phi = co\{[-160 \quad 0]^T, [0 \quad -160]^T, [160 \quad 160]^T\} \tag{36}$$

In this example, five IPs and two ACPs are used which are shown in Fig. 4:

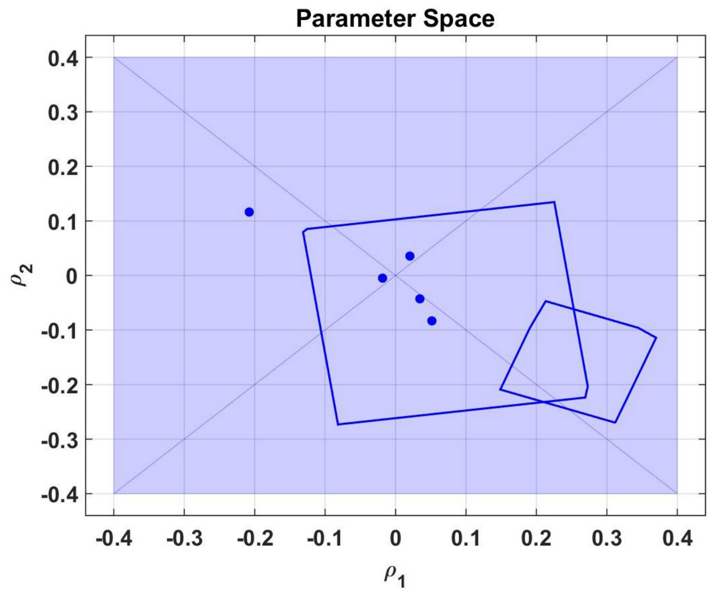

Fig. 4. Parameter space Ω_0 defined in (3) and IPs and ACPs utilized in Example 1

The characteristic polynomial of this LPV model will be obtained as follows:

$$d = s^2 - (\alpha_2 + 0.2\rho_1)s + 0.1\alpha_2\rho_1 - \rho_2 - \alpha_1\rho_2 - \alpha_1 + 0.01\rho_1^2 - \rho_2^2 \tag{37}$$

Due to (37), LPV model (32) satisfies the whole assumption of LPV model (1).

Fig. 5 shows the searching procedure of DSEA to solve the design problem. In these figures, the highlighted simplex presents the selected simplex in the current iteration, not-filled simplexes refer to undesired subspaces that are detected by IPs or ACPs and the transparent simplexes show the spaces that are generated in previous iteration, but not checked until this iteration.

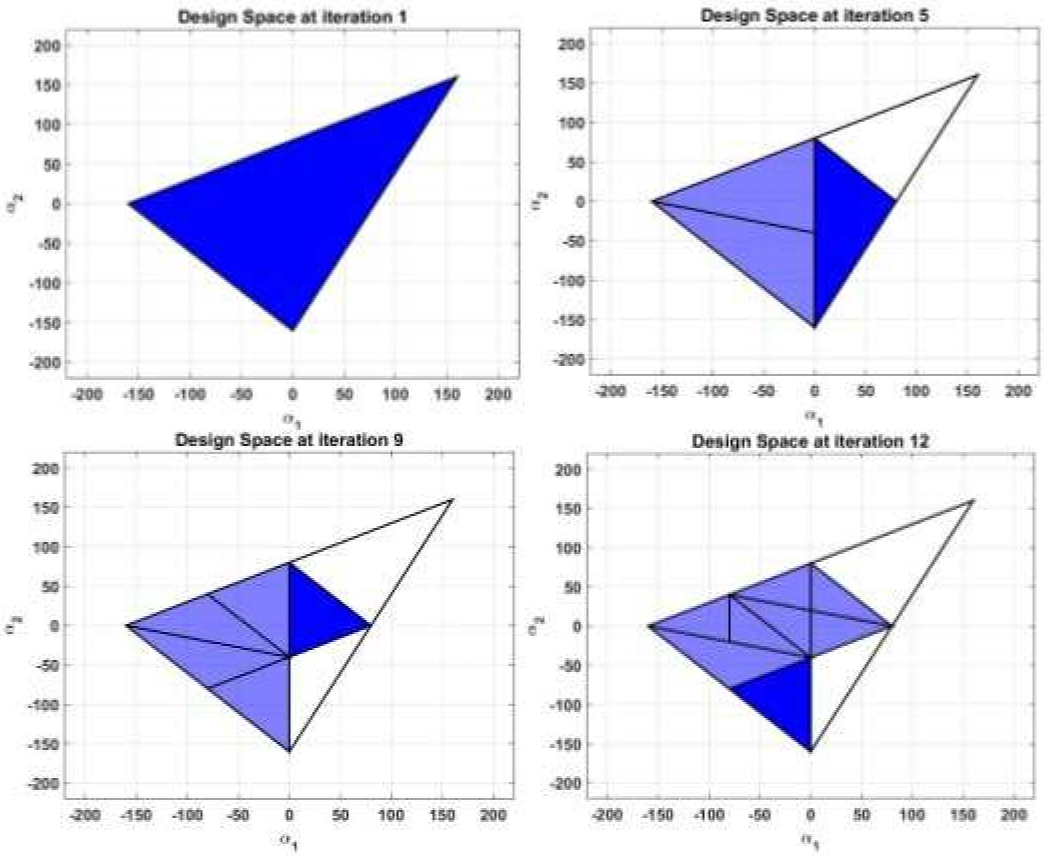

Fig. 5. Design space defined in (41) and DSEA's current space at some iterations (highlighted)

It should be noted that DSEA is successful in finding an appropriate feasible stabilizing point and the following point is obtained at its twelfth iteration:

$$\alpha = [-80 \quad -80]^T \tag{38}$$

Example 2. This example compares the feasibility performance of DSEA to some existing ones. Three hundred models are randomly generated. These models have different number of states and parameters. Then, DSEA and previous methods in [14], [15], [27], and [28] are evaluated for each generated system. The generated models are supposed to have the following structure:

$$\dot{x}(t) = \left(A_0 + \textstyle\sum_{i=1}^{2} A_i\rho_i(t)\right)x(t) + \left(B_0 + \textstyle\sum_{i=1}^{2} B_i\rho_i(t)\right)u(t) \tag{39}$$

$$u(t) = Kx(t) \tag{40}$$

where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^{n_u}$ are the state vector and control input that is supposed to be single. Matrices $\{A_i\}_{i=0}^2 \in \mathbb{R}^{n\times n}$ and vector $\{B_i\}_{i=0}^2 \in \mathbb{R}^{n\times n_u}$ are constant matrices that are randomly generated. The controller aim is to precisely design feedback gain K to stabilize the closed loop model presented in the following:

$$\dot{x}(t) = \left(A_0 + B_0 K + \sum_{i=1}^{2}(A_i + B_i K)\,\rho_i(t)\right)x(t) \tag{41}$$

The parameter spaces Ω_0 and Ω_1 are considered to be as follows for the whole generated systems:

$$\Omega_0 = co\begin{Bmatrix} [-0.5 \quad -0.5]^T, [-0.5 \quad 0.5]^T, \\ [0.5 \quad -0.5]^T, [0.5 \quad 0.5]^T \end{Bmatrix} \tag{42}$$

$$\Omega_1 = co\begin{Bmatrix} [-0.1 \quad -0.1\,]^T, [-0.1 \quad 0.1]^T, \\ [0.1 \quad -0.1]^T, [0.1 \quad 0.1]^T \end{Bmatrix} \tag{43}$$

The number of systems which are stabilized by DSEA and exiting methods are mentioned in Table 2. Fifty models are generated for each system's order and the number of feasible cases are mentioned for each method. The details of the results are available at *https://figshare.com/s/48bbd3d53f2a3f737fd3*.

Table 2. Number of feasible models

System order	Method in [15]	Method in [14]	Method in [27]	Method in [28]	IDA
$n = 2, n_u = 1$	30	24	26	33	46
$n = 3, n_u = 1$	21	8	11	25	42
$n = 3, n_u = 2$	43	21	33	45	47
$n = 4, n_u = 1$	8	0	8	20	41
$n = 4, n_u = 2$	35	9	36	41	44
$n = 6, n_u = 1$	0	0	0	3	36

Table 2 demonstrates that DSEA is more appropriate in stabilizing the LPV models than the existing approaches. In the sixth row of Table 2, DSEA stabilizes 36 systems, while the previous methods stabilize none of them.

Example 3 (Microgrid model). This example evaluates the ability of the proposed algorithm to stabilize a sample microgrid model, borrowed from [29]. The model involves Photovoltaic (PV), Wind Turbine (WT), Fuel Cell (FC), Diesel Generator (DEG), Gas Turbine (GS), Battery Energy System (BES), and Flywheel Energy System (FES) as shown in Fig. 6. Each of these systems are modeled by a simple first-order transfer function, but the time constant in the model is considered as a varying parameter in order to consider the model nonlinearity. The aim of this example is to design a stabilizing controller which damps the frequency deviation in the microgrid.

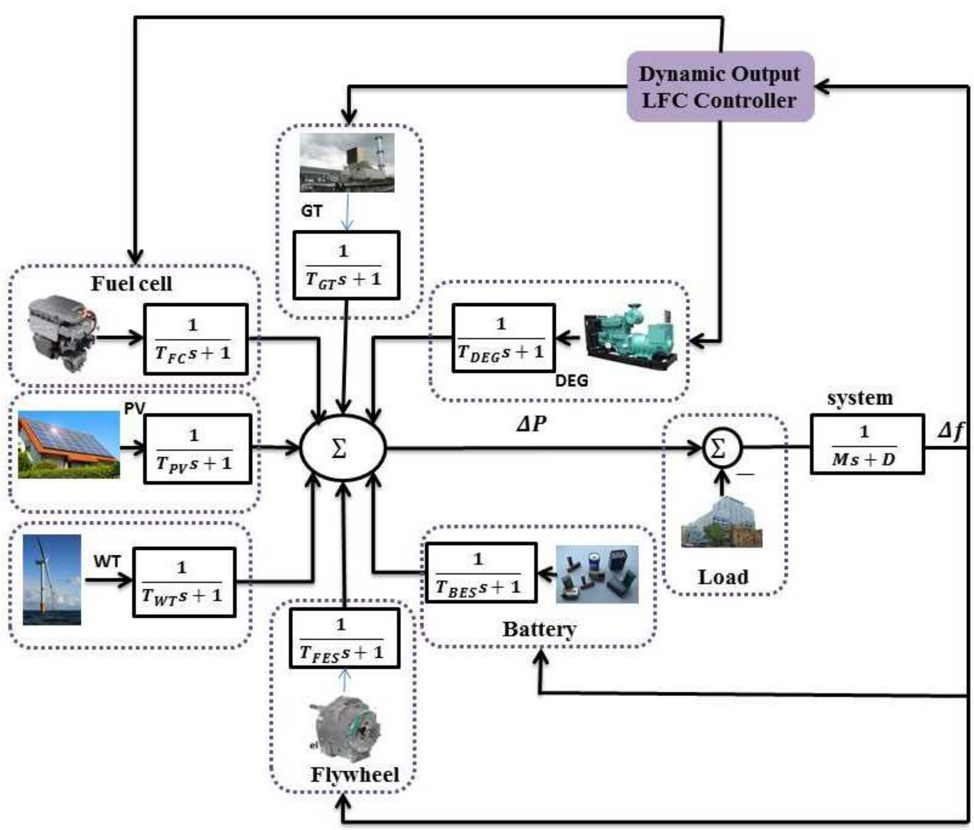

Fig. 6. The schematic diagram of a microgrid model [29]

The values of the system's parameters and their definitions are given in Table 3.

Table 3. Microgrid model's parameters

Parameter	Definition	Value	Parameter	Definition	Value
$\boldsymbol{T_{GT}}$	GT time constant	$2\ s$	T_{BES}	BES time constant	$0.1\ s$
$\boldsymbol{T_{DEG}}$	DEG time constant	2 s	T_{FES}	FES time constant	$0.1\ s$
$\boldsymbol{T_{FC}}$	FC time constant	$4\ s$	T_{WEC}	WEC time constant	$1.5\ s$
$\boldsymbol{T_{PV}}$	PV time constant	$1.8\ s$	D	Power system gain	$0.012(pu/Hz)$
$\boldsymbol{M}$	Power system time constant	$0.2(pu/s)$			

According to Fig. 6, the state space model of the microgrid can be obtained as follows:

$$\dot{x}_s(t) = A\big(\rho(t)\big)x_s(t) + B\big(\rho(t)\big)u(t)$$

$$y(t) = Cx_s(t) \tag{44}$$

where $x_s(t) = [\Delta P_{WEC} \quad \Delta P_{PV} \quad \Delta P_{DEG} \quad \Delta P_{FC} \quad \Delta P_{GT} \quad \Delta P_{BES} \quad \Delta P_{FES} \quad \Delta f]^T$ is the vector of internal states and $u(t)$ and $y(t)$ are the input and output of the model. The state vector respectively consists of the power deviation of WEC, PV, DEG, GT, BES, FES, and the frequency deviation of the grid. Also, system matrices $A\big(\rho(t)\big)$, $B\big(\rho(t)\big)$, and C are as follows:

$$A = \begin{bmatrix} -\rho_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\rho_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\rho_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\rho_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\rho_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\rho_6 & 0 & \rho_6 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\rho_7 & \rho_7 \\ \frac{1}{M} & \frac{1}{M} & \frac{1}{M} & \frac{1}{M} & \frac{1}{M} & \frac{1}{M} & \frac{1}{M} & -\frac{D}{M} \end{bmatrix} \tag{45}$$

$$B = [0 \quad 0 \quad \rho_3 \quad \rho_4 \quad \rho_5 \quad 0 \quad 0 \quad 0]^T \tag{46}$$

$$C = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1] \tag{47}$$

where $\rho = [\rho_1 \;\; \rho_2 \;\; \dots \;\; \rho_7]^T$ is the vector of uncertain parameters defined as follows:

$$\rho_1(t) = \frac{1}{T_{WEC}} v_1(t),\ \rho_2(t) = \frac{1}{T_{PV}} v_2(t),\ \rho_3(t) = \frac{1}{T_{DEG}} v_3(t), \rho_4(t) = \frac{1}{T_{FC}} v_4(t)$$

$$\rho_5(t) = \frac{1}{T_{GT}} v_5(t),\ \rho_2(t) = \frac{1}{T_{BES}} v_6(t),\ \rho_3(t) = \frac{1}{T_{FES}} v_7(t) \tag{48}$$

where $\{v_i(t)\}_{i=1}^{7}$ represent the time-dependent uncertain nature of the microgrid model. These uncertain functions are supposed to satisfy $|v_i(t)| \le v$ and $|\dot{v}_i| \le v_d$ in which v and v_d determines the scale of uncertainty in this model.

The controller is assumed to have a dynamic output feedback form as given in (49):

$$\dot{x}_c(t) = \alpha_1 x_c(t) + y(t)$$

$$u(t) = \alpha_2 x_c(t) + \alpha_3 y(t) \tag{49}$$

where x_c is the internal state of the controller and $\alpha = [\alpha_1 \;\; \alpha_2 \;\; \alpha_3]$ is the control parameter.

Using (44) and (49), the closed loop system is obtained as follows:

$$\dot{x}(t) = \begin{bmatrix} A(\rho) + \alpha_3 B(\rho) C & \alpha_2 B(\rho) \\ C & \alpha_1 \end{bmatrix} x(t) \tag{50}$$

It is obvious that the system matrix of the closed loop model is an affine function of both design and uncertain parameters. Thus, the proposed algorithm can be simply applied to this model. For this simulation example, $v = 0.1$, $v_d = 10$, and the primary design space $\alpha \in \phi$ is assumed to be as given in (51):

$$\phi = co\{[10^5 \;\; 0 \;\; 0], [0 \;\; 10^5 \;\; 0], [0 \;\; 0 \;\; 0], [-10^5 \; -10^5 \; -10^5]\} \tag{51}$$

The proposed algorithm iteratively searches the above primary design space and it finally finds the following feasible point:

$$\alpha = [-37.9137 \quad 2714 \quad -336] \tag{52}$$

According to (49) and (52), the appropriate controller is obtained as follows:

$$\dot{x}_c(t) = -37.9137x_c(t) + y(t)$$

$$u(t) = 2714x_c(t) - 336y(t) \tag{53}$$

In the following, the closed loop model is simulated for some cases of the uncertain parameters. The results are shown in Figs. 7 and 8:

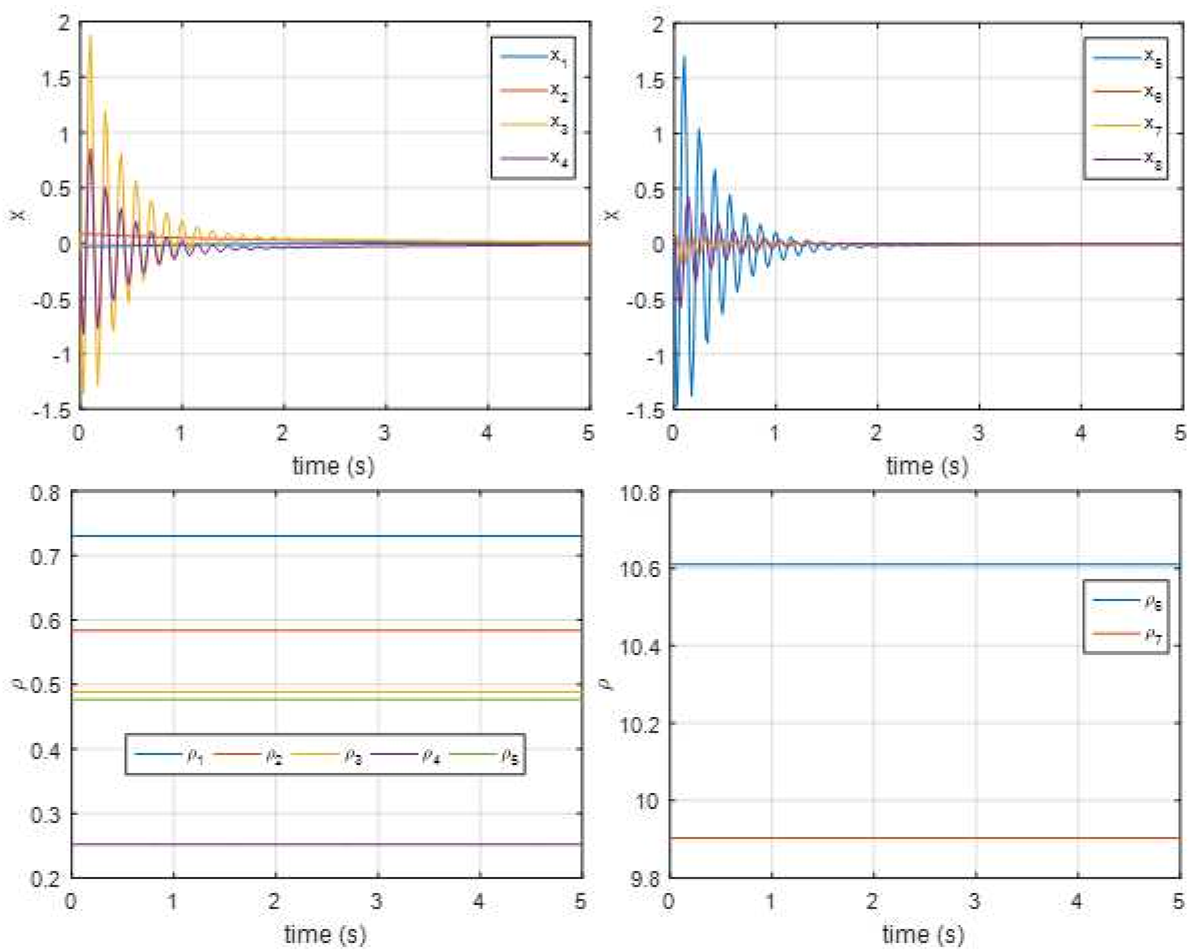

Fig. 7. The simulation results of model (44) considering controller (53) with constant ρ

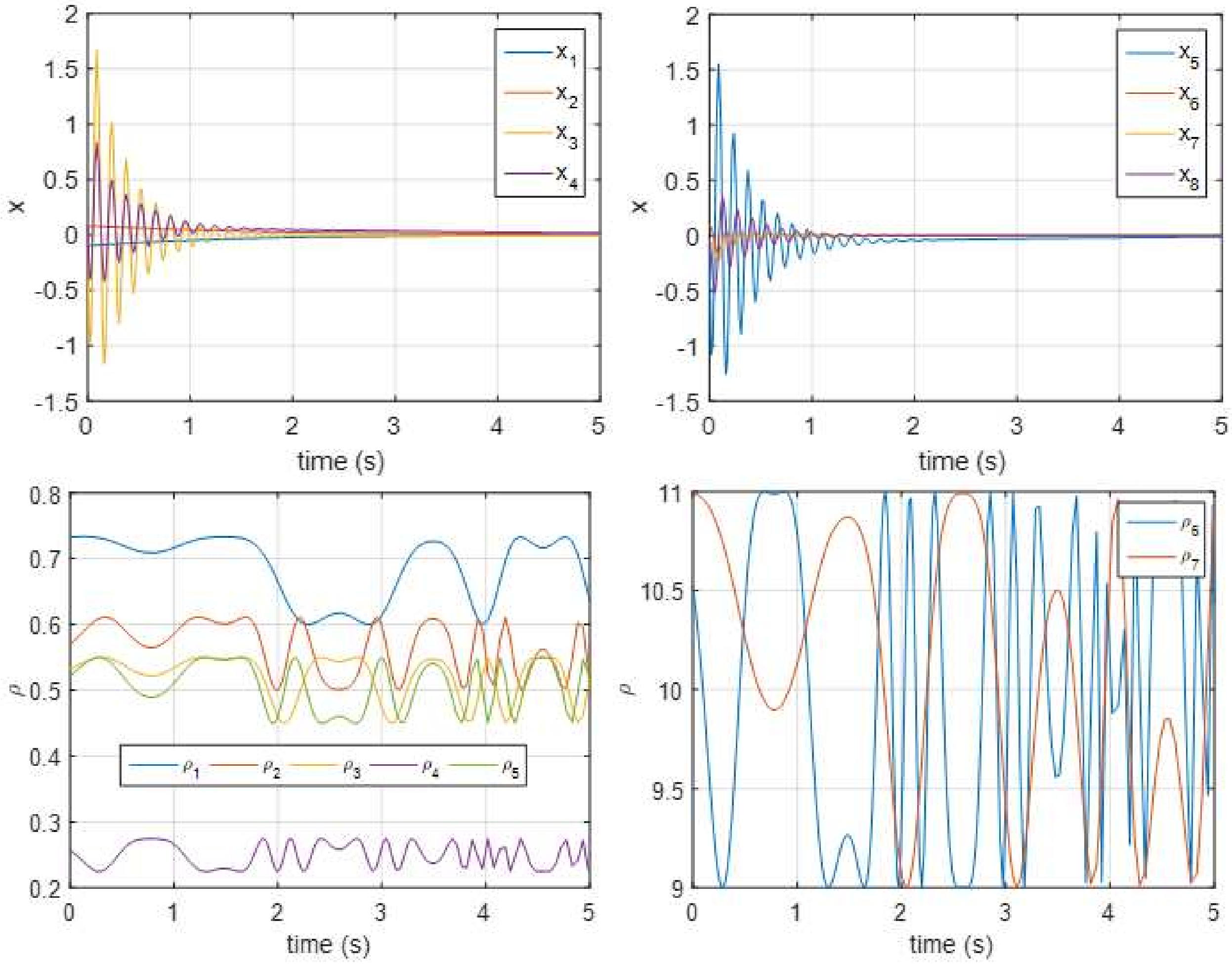

Fig. 8. The simulation results of model (44) considering controller (53) with sinusoidal ρ

Figs. 7 and 8 reveal the stability of the closed loop model which is controlled by (53). According to these figures, the obtained controller is able to stabilize the closed loop model with uncertain parameters shapes involving special cases of constant and sinusoidal uncertain parameters.

6. Conclusion

This paper investigates the stabilizability problem of LPV systems with a particular emphasis on coupling issue between Lyapunov and design variables. The LPV model is supposed to affinely depend on the design and uncertain parameters. An algorithm is presented to find an appropriate point in the design space via iteratively removing the undesired design subspaces and checking the stabilizability of the generated subspaces corner points. It has been shown that the proposed algorithm does not remove any feasible solution and it converges to the existed feasible point at some special situations. Three simulation examples are provided to compere feasibility performance of the proposed algorithm with the existing methods. The simulation results reveal the superiority of the proposed algorithm in stabilizing control design for LPV systems.

References

[1] D. Corona, A. Cristofaro and D. Rotondo, "Reachability and stabilization of scheduled steady-states for LPV single-input systems.," *Journal of the Franklin Institute,* vol. 356, no. 8, pp. 4478-4495, 2019.

[2] D. Rotondo, V. Puig, F. Nejjari and M. Witczak, "Automated generation and comparison of Takagi–Sugeno and polytopic quasi-LPV models.," *Fuzzy Sets and Systems,* vol. 277, pp. 44-64, 2015.

[3] O. Sename, P. Gaspar and J. Bokor, Robust control and linear parameter varying approaches: application to vehicle dynamics., Springer, 2013.

[4] M. Mejari, D. Piga and A. Bemporad, "A bias-correction method for closed-loop identification of Linear Parameter-Varying systems.," *Automatica,* vol. 87, pp. 128-141, 2018.

[5] K. Zhu, D. Ma and J. Zhao, "Event triggered control for a switched LPV system with applications to aircraft engines.," *IET Control Theory & Applications,* vol. 12, no. 10, pp. 1505-1514, 2018.

[6] A. K. Al-Jiboory, G. Zhu, S. M. Swei, W. Su and N. T. Nguyen, "LPV modeling of a flexible wing aircraft using modal alignment and adaptive gridding methods.," *Aerospace science and technology,* vol. 66, pp. 96-102, 2017.

[7] E. Alcalá, V. Puig and J. Quevedo, "LPV-MP planning for autonomous racing vehicles considering obstacles.," *Robotics and Autonomous Systems,* vol. 124, 2020.

[8] G. Eigner, M. Siket, A. Szakál, I. Rudas and L. Kovács, "Discrete LPV Modeling of Diabetes Mellitus for Control Purposes.," *In 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC) ,* pp. 2558-2563, 2018.

[9] G. Becker and A. Packard, "Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback.," *Systems & Control Letters,* vol. 23, no. 3, pp. 205-215, 1994.

[10] N. A. Keles, C. M. Agulhari, M. J. Lacerdaand, "Stability analysis and robust performance of periodic discrete-time uncertain systems via structured Lyapunov functions.," *European Journal of Control*, 2020.

[11] L. Na, K. Y. Zhao and M. Yuan, "Robust stability analysis of time-varying parameter polynomial-dependent linear systems," *In 2007 IEEE International Conference on Automation and Logistics,* pp. 719-723, 2007.

[12] M. S. de Oliveira and R. L. Pereira, "IOn Unknown Input Observers Designs for Discrete-time LPV Systems with Bounded Rates of Parameter Variation," *European Journal of Control,* 2020.

[13] D. Peaucelle and D. Arzelier, "Robust performance analysis with LMI-based methods for real parametric uncertainty via parameter-dependent Lyapunov functions.," *IEEE Transactions on Automatic Control,* vol. 46, no. 4, pp. 624-630, 2001.

[14] Y. Y. Cau and Z. Lin, "A descriptor system approach to robust stability analysis and controller synthesis.," *Transactions on Automatic Control,* vol. 49, no. 11, pp. 2081-2084, 2004.

[15] L. Song and J. Yang, "An improved approach to robust stability analysis and controller synthesis for LPV systems.," *International Journal of Robust and Nonlinear Control,,* vol. 21, no. 13, pp. 1574-1586, 2011.

[16] R. C. Oliveira and P. L. Peres, "Stability of polytopes of matrices via affine parameter-dependent Lyapunov functions: Asymptotically exact LMI conditions.," *Linear algebra and its applications,* vol. 405, pp. 209-228, 2005.

[17] P. B. Cox, S. Weiland and R. Tóth, "Affine parameter-dependent Lyapunov functions for LPV systems with affine dependence.," *IEEE Transactions on Automatic Control,* vol. 63, no. 11, pp. 3865-3872, 2018.

[18] B. Lu and F. Wu, "Switching LPV control designs using multiple parameter-dependent Lyapunov functions.," *Automatica,* vol. 40, no. 11, pp. 1973-1980, 2004.

[19] N. Aouani, S. Salhi, G. Garcia and M. Ksouri, "Parameter dependent Lyapunov functions for stability of linear parameter varying systems," *In 2010 17th IEEE International Conference on Electronics, Circuits and Systems,* pp. 1002-1005, 2010.

[20] C. W. Scherer, "LMI relaxations in robust control.," *European Journal of Control,* vol. 12, no. 1, pp. 3-29, 2006.

[21] A. Delibaşı, I. B. Kucukdemiral and G. Cansever, "control of LPV systems with saturating actuators: Pólya approach.," *Optimal Control Applications and Methods,* vol. 34, no. 1, pp. 17-34, 2013.

[22] J. Daafouz and J. Bernussou, "Parameter dependent Lyapunov functions for discrete time systems with time varying parametric uncertainties.," *Systems & control letters,* vol. 43, no. 5, pp. 355-359, 2001.

[23] F. Amato, M. Mattei and A. Pironti, "Gain scheduled control for discrete-time systems depending on bounded rate parameters.," *International Journal of Robust and Nonlinear Control,* vol. 15, no. 11, pp. 473-494, 2005.

[24] P. T. Bandeira, P. C. Pellanda and L. O. de Araújo, "New Haar-based algorithms for stability analysis of LPV systems.," *IEEE control systems letters,* vol. 2, no. 4, pp. 605-610, 2018.

[25] R. Abolpour, M. Dehghani and H. A. Talebi, "Output feedback controller for polytopic systems exploiting the direct searching of the design space," *International Journal of Robust and Nonlinear Control,* vol. 29, p. 5164– 5177, 2019.

[26] H. K. Khalil, Nonlinear systems, Upper Saddle River, 2002.

[27] P. Park, N. K. Kwon and B. Y. Park, "State-feedback control for LPV systems with interval uncertain parameters," *Journal of the Franklin Institute*, vol. 352, no. 11, pp. 5214-5225, 2015.

[28] F. Großmann and F. Svaricek, "Parameter dependent static output feedback control-An LPV approach," *In 2019 18th European Control Conference (ECC)*, pp. 3322-3327, 2019.

[29] H. Javanmardi, M. Dehghani, M. Mohammadi, S. Siamak, and M.R. Hesamzadeh, 2021. BMI-based Load Frequency Control in Microgrids Under False Data Injection Attacks. *IEEE Systems Journal*.