Output feedback controller design for discrete LTI systems with polytopic uncertainty via direct searching of the design space

Roozbeh Abolpour\(^1\) | Maryam Dehghani\(^1\) | Heidar Ali Talebi\(^2\)

\(^1\)School of Electrical and Computer Engineering, Shiraz University, Shiraz, Iran
\(^2\)Department of Electrical Engineering, Amirkabir University of Technology, Tehran, Iran

Abstract

This paper concerns static output feedback stabilization of polytopic discrete linear time-invariant (LTI) systems. The previous related studies were mainly based on linear matrix inequality (LMI) approaches which are naturally conservative. In this paper, a novel design algorithm is presented that iteratively partitions a primary design space to subspaces. Then, by assessing stabilizability status of each generated subspace, the algorithm determines the total stabilizable parts and removes the undesired parts of the design space. Mathematical theories are developed to determine the total de-stabilizability or stabilizability of a given subspace. These subspaces’ properties are detected through checking the existence of critical polynomials (which have roots on the unit circle of the complex plane) on their exposed edges. By omitting the undesired parts of the design space, the algorithm just searches the desired parts which are far smaller than the primary design space. This strategy improves the feasibility performance of the algorithm. Some illustrating examples are provided to show the steps of the design algorithm. Furthermore, 100 random models are generated to evaluate the feasibility performance of the proposed algorithm as compared to some existing methods. The results reveal the superiority of the proposed algorithm.

KEYWORDS
direct searching, exposed edges, output feedback controller, polytopic systems, stability of convex polynomial spaces

1 | INTRODUCTION

The polytopic uncertain model includes a type of parametric uncertainty which is exploited to model the existing uncertainty in practical models [1]. This type of uncertainty is frequently encountered in the analysis of various kinds of discrete-time systems [2,3]. In this type of model uncertainty, the system matrices are considered to be a convex combination of some known corner matrices [4–6].

One appropriate controller is the static output feedback (SOF) controller [7,8] due to its easy implementation. In the presence of polytopic uncertainties, robust SOF synthesis of LTI systems is widely investigated in the previous studies [9–11]. Although, some novel and valuable approaches are developed, this problem is still an...
open challenge [12]. It is mainly because, there is a lack of a suitable and acceptable approach that is able to solve this problem by a set of sufficient and necessary conditions [13,14]. Among various approaches in this area, the Lyapunov-based methods are the most widely popular and common approaches.

In the Lyapunov-based approaches, two groups of variables are considered as the design and Lyapunov variables [15,16]. The design variables are free variables in the control law which exist in the gain of the controller and the Lyapunov variables are used in the Lyapunov candidate functions. The main issue which arises in these approaches is the existence of coupling terms between the Lyapunov and design variables [10]. Indeed, the stability conditions are given in bilinear matrix inequality (BMI) form which cannot be efficiently solved by available linear matrix inequality (LMI) solvers. To cope with this issue, sufficient BMI conditions are developed to convexify these BMI conditions via utilizing LMI tricks or limiting assumptions [12,17]. Clearly, this approach is conservative due to the assumptions made during converting the problem into an LMI form. For instance, in Crusius and Trofino [17], the Lyapunov function is forced to have a special structure. Thus, the BMI conditions are converted to LMIs by inserting an LMI equality constraint on the Lyapunov variables. This change imposes conservativeness to the problem because the equality constraints result in a set of approximate conditions. The LMI tricks such as congruence transformation are also used to convexify the main nonconvex conditions [18].

Some approaches consider the parameter-dependent Lyapunov functions to decrease the conservativeness of the resulting stability criteria [19]. In such problems, the Lyapunov candidate function is considered to depend affinely on the uncertain parameters. Similar to the previous approaches, some LMI tricks are adopted to convexify the final conditions. These approaches significantly reduce the conservativeness which are also more feasible than the methods pertaining to the common Lyapunov functions. However, some conservatism still remains in their approach.

To overcome the disadvantages of the present approaches, this paper extends the direct searching idea which is recently developed to design robust stabilizing controllers for continuous LTI systems [20,21] to LTI discrete models. In other words, here, a design algorithm is proposed for robust SOF stabilization of discrete LTI systems in the presence of polytopic uncertainties. The controller may have some known and unknown variables in its structure where the unknown variables are noted by design variables. It is supposed that the design variables belong to a compact convex polygonal space which can be located anywhere in the whole possible real spaces, and its shape does not restrict the design methodology. A design algorithm tries to find a point inside the design space that is capable of stabilizing the uncertain model where the uncertain variables belong to the unit simplex.

The algorithm iteratively divides the primary design space to smaller parts and predicts the existence or nonexistence of feasible points in the given subspaces. If it detects that there is no feasible point in the design subspace, it marks it as total-destabilizability (TD) space. On the other hand, if it predicts that the whole subspace can robustly stabilize the model, it marks it as the total-stabilizability (TS) space. Investigation of the TD property of any subspaces is one of the most important novelties of this paper. The previous studies focus on the TS conditions, but this paper proposes a systematic way to predict the TD subspaces, and this achievement enables us to detect the undesired parts of the design space and delete them rapidly. Obviously, this tool may significantly improve the convergence rate of the design algorithm.

Briefly, the paper contributions are as follows:

First, the present approaches for SOF design for discrete LTI systems with polytopic uncertainties are conservative. This paper presents a less-conservative approach for SOF design of such systems through direct searching of the design space. The method considers the whole design space and checks all parts for finding a feasible solution.

Second, to increase the convergence rate of the method, the design space is partitioned to smaller subspaces, and they are searched for TS, TD, or undermined parts. Finding a TS subspace is satisfactory and ends the method, while finding a TD subspace means that no solution exists in this subspace. Thus, the algorithm omits this part and shrinks the design space. If the subspace is undetermined (neither TS nor TD), the method partitions that special subspace and continues the method. All the theories and the method for detecting TD and TS subspaces and the criteria for continuing the design space partitioning are expanded in the body of the paper.

Third, the TS and TD determination of the subspaces depend on finding critical polynomials which are defined in the paper and the related theories are mentioned.

The paper is organized as follows. In Section 2, the uncertain plant model and its algebra are presented. In Section 3, the design algorithm which is the main contribution of this paper is mentioned. Extensive simulation examples are provided in Section 4 to assess the trueness of the proposed contributions of this paper. Finally, Section 5 concludes the paper.
2 | PLANT MODEL

Consider the following uncertain LTI model, in which, the uncertain parameters are assumed to have polytopic form:

$$\begin{align*}
    x(k+1) &= A_p(\theta)x(k) + B_p(\theta)u(k) \\
    y(k) &= C_p x(k)
\end{align*}$$

(1)

where \( k \) is the time index and \( x \in \mathbb{R}^n \), \( u \in \mathbb{R}^{n_u} \), and \( y \in \mathbb{R}^n \) are the states, inputs, and outputs, respectively. Uncertain parameters are shown by vector \( \theta \in \mathbb{R}^n \) assumed to be constant and belong to a unit simplex with the following definition:

$$\Delta_n = \{ \theta \in \mathbb{R}^n | \forall i \in \{1,...,n_i\} : 0 \leq \theta_i \land \theta_1 + ... + \theta_{n_i} = 1 \}$$

(2)

In (1), the system matrix \( C_p \in \mathbb{R}^{p \times n} \) is supposed to be constant and known. However, the matrices \( A_p(\theta) \), \( B_p(\theta) \) affinely depend on the uncertain parameters:

$$\begin{align*}
    A_p(\theta) &= \sum_{i=1}^{n_i} A_i \theta_i \\
    B_p(\theta) &= \sum_{i=1}^{n_i} B_i \theta_i
\end{align*}$$

(3)

(4)

where \( \{A_i\}_{i=1}^{n_i} \subset \mathbb{R}^{n \times n} \) and \( \{B_i\}_{i=1}^{n_i} \subset \mathbb{R}^{n \times n_u} \) are corner matrices of these system matrices.

Now, consider the following control law,

$$u = F(f)y$$

(5)

where \( f \in \mathbb{R}^{n_u} \) is the vector of design parameters and \( F(f) \in \mathbb{R}^{p \times n_u} \) is an affine function of these design parameters.

According to (1) and (5), the closed loop system will be obtained as given below:

$$x(k+1) = (A_p(\theta) + B_p(\theta)F(f)C_p)x(k)$$

(6)

The following assumptions are considered for the closed loop model (6):

- The design parameters \( f \) should belong to a primary space \( \Omega \), called primary design space.
- The characteristic polynomial of the closed loop system should affinely depend on \( f \).

Definition 1. Büyükköroğlu et al. [22]. Let \( A \in \mathbb{R}^{n \times n} \) be an arbitrary square matrix. This matrix is noted to be Schur stable if all its eigenvalues are inside the unit circle of the complex plane (in this paper, the unit circle is shown by \( \mathcal{C} \)).

Definition 2. Let \( \psi \) be an arbitrary subspace inside the design space \( \Omega \). This subspace is said to be totally stabilizable (TS) if it satisfies the following condition:

$$\forall f \in \psi, \forall \theta \in \Delta_n : A_p(\theta) + B_p(\theta)F(f)C_p$$

is Schur stable

(7)

Definition 3. Let \( \psi \) be an arbitrary subspace in the design space \( \Omega \). It is noted to be totally de-stabilizable (TD) if it holds the following condition:

$$\forall f \in \psi, \exists \theta \in \Delta_n : A_p(\theta) + B_p(\theta)F(f)C_p$$

is not Schur stable

(8)

Definition 4. The real coefficient polynomial is critical if it has some roots on the boundary of the unit circle in the complex plane.

Definition 5. If \( \mu \) is a compact convex polygonal space, then \( \partial(\mu), \partial_1(\mu) \) and \( \partial_2(\mu) \) stand for the boundary, corners, and exposed edges of \( \mu \), respectively.

3 | MAIN RESULTS

This section presents the main contribution of the paper that is an algorithm to find a feasible point inside the primary design space \( \Omega \) that stabilizes model (6). Assume the primary design space as a convex and compact space. At first, the whole space is considered, and its stability status is evaluated. The evaluation result is whether it is TS, TD, or none of them. If it is TS, one of the points inside it is returned as the feasible solution. If it is TD, there will be no feasible point inside this space, and the algorithm terminates with infeasibility. If the space is neither TS, nor TD, it is divided to two smaller subspaces through halving its largest edge. The method is repeated for the new subspaces, and it is continued until a feasible point is found or all the smaller parts are marked as TD. It is apparent that the sufficient conditions which are presented to exploit the TS or TD properties are more feasible in the smaller subspaces in comparison to larger ones.
Finding critical polynomials in a polynomial segment (a convex combination space of just two polynomials) plays a key role to develop TD conditions. For this purpose, an efficient method is proposed that can find all critical polynomials on a given polynomial segment. The critical polynomials are detected through finding roots of another polynomial that is defined based on the real and imaginary parts of the complex variable. Theorem 1. Assume \( p \) is a real coefficient polynomial and \( v \) is an arbitrary complex number on the boundary of \( C \). Then, \( p(v) \) can be decomposed as follows:

\[
p(v) = p_r(\sigma) + j\omega p_i(\sigma)
\]  

where \( \sigma \) and \( \omega \) are the real and imaginary parts of \( v \) and \( p_r(\sigma) \) and \( p_i(\sigma) \) are real coefficient polynomials of \( \sigma \).

**Proof.** An induction technique can be used to successively prove the statement of the theorem for all polynomials with different degrees. Indeed, this technique shows the trueness of the following logical statement for each \( m \in \{1,2,3,\ldots\} \):

\[
Q[m] \equiv \text{all polynomials with degree } m \text{ that satisfy (9)}
\]  

Firstly, it is shown that \( Q[1] \) is true. Then, it will be proved that \( Q[m] \rightarrow Q[m+1] \) which means \( Q[m+1] \) is true only if \( Q[m] \) is true, as well.

Note that each polynomial with degree 1 can be written as \( p(z) = a_0 + a_1 z \) where \( a_0 \) and \( a_1 \) are real numbers. Obviously, \( p(v) = a_0 + a_1 \sigma + j a_1 \omega \) that implies

\[
p_r(\sigma) = a_0 + a_1 \sigma.
\]  

(11)

\[
p_i(\sigma) = a_1.
\]  

(12)

According to (11) and (12), we can see that \( Q[1] \) is true.

Now, assume \( Q[m] \) is true which means all polynomials with degree \( m \) satisfy (9). Let \( p(z) = \sum_{i=0}^{m+1} a_i z^i \) be an arbitrary real polynomial with degree \( m+1 \) and \( v = \sigma + j \omega \) be an arbitrary complex number on \( \partial(C) \). Thus, one can obtain

\[
p(z) = z q(z) + a_0
\]  

(13)

where \( q(z) = \sum_{i=0}^{m} a_{i+1} z^i \) is another polynomial with degree \( m \). Since \( q \) has degree \( m \) and \( Q[m] \) is considered to satisfy the theorem hypothesis, \( q(v) \) can be decomposed as follows:

\[
q(v) = q_r(\sigma) + j\omega q_i(\sigma)
\]  

(14)

Substituting (14) in (13) results in

3.1 Finding critical polynomials on a polynomial segment

It is known that a polynomial segment is a convex combination of two polynomials [23]. The proposed method detects the critical polynomials on a polynomial segment via finding possible roots on the unit circle of the complex plane.

Before presenting the proposed methodology, the following theorems are needed. Theorem 1 mentions that the real coefficient polynomial in the complex plane can be decomposed to real and imaginary parts, in which, the real part only depends on the real complex variable, while the imaginary part depends on both real and imaginary parts of the complex variable. Theorem 2 derives two conditions to find critical polynomials on a polynomial segment based on Theorem 1.
\[ p(v) = (\sigma + j\omega)q_r(\sigma) + (\sigma + j\omega)jq_l(\sigma) + a_0. \]  

(15)

By expanding (15), the above equation can be expressed:

\[ p(v) = a_0 + \sigma q_r(\sigma) - \omega^2 q_l(\sigma) + j\omega(q_r(\sigma) + \sigma q_l(\sigma)) \]  

(16)

Regarding \( v \in \partial(C) \), one can easily conclude that \( \sigma^2 + \omega^2 = 1 \) which implies that

\[ p(v) = a_0 + \sigma q_r(\sigma) - (1 - \sigma^2)q_l(\sigma) + j\omega(q_r(\sigma) + \sigma q_l(\sigma)). \]  

(17)

Then, by using (17), it can be easily concluded that

\[ p_r(\sigma) = a_0 + \sigma q_r(\sigma) - (1 - \sigma^2)q_l(\sigma). \]  

(18)

\[ p_i(\sigma) = q_r(\sigma) + \sigma q_l(\sigma). \]  

(19)

Hence, \( p \) satisfies (9) which completes the proof.

To clarify the results of Theorem 1, the following numerical example is presented. Let \( p(z) \) be a real polynomial such as

\[ p(z) = z^3 + 2z^2 - z + 1. \]  

(20)

By substituting \( v = \sigma + j\omega \) in (20), one can get:

\[ p(v) = \sigma^3 + 3j\omega\sigma^2 - 3\sigma^2\sigma - j\omega^3 + 2\sigma^2 - 2\omega^2 + 4j\omega\sigma - \sigma - j\omega + 1. \]  

(21)

Now, (21) can be factorized as follows:

\[ p(v) = (\sigma^3 + 2\sigma^2 - 3\sigma^2\sigma - 2\sigma^2 - \sigma + 1) + j\omega(3\sigma^2 - \omega^2 + 4\sigma - 1) \]  

(22)

Since \( \omega^2 = 1 - \sigma^2 \), it can be easily concluded that

\[ p(v) = (4\sigma^3 + 4\sigma^2 - 4\sigma - 1) + j\omega(4\sigma^2 + 4\sigma - 2). \]  

(23)

Therefore,

\[ p_r(\sigma) = 4\sigma^3 + 4\sigma^2 - 4\sigma - 1 \]  

(24)

which approves the result of Theorem 1. In the following, Theorem 2 is presented for finding the critical polynomials in a given polynomial segment.

**Theorem 2.** Let \( p(z) \) and \( q(z) \) be two polynomials with similar degrees. Then, there are \( \alpha \in [0,1] \) and \( v = \sigma + j\omega \in \partial(C) \) such that \( \alpha p(v) + (1 - \alpha)q(v) = 0 \) if and only if the following conditions hold:

\[ p_r(\sigma)q_r(\sigma) - p_i(\sigma)q_i(\sigma) = 0 \]  

(26)

\[ p_r(\sigma)q_r(\sigma) + p_i(\sigma)q_i(\sigma) \leq 0 \]  

(27)

where \( p_r(\sigma), q_r(\sigma), p_i(\sigma), \) and \( q_i(\sigma) \) are defined in Theorem 1.

**Proof.** To prove the necessity part, let \( \alpha \in [0,1] \) and \( v \in \partial(C) \) which leads to \( \alpha p(v) + (1 - \alpha)q(v) = 0 \). Thus, the following results can be obtained due to Theorem 1 and the assumption \( \omega \neq 0 \):

\[ \alpha(q_r(\sigma) - p_i(\sigma)) = q_i(\sigma) \]  

(28)

\[ \alpha(q_i(\sigma) - p_r(\sigma)) = q_r(\sigma) \]  

(29)

According to (28), (29), and \( \alpha \in [0,1] \), \( q_r(\sigma) \) and \( p_i(\sigma) \) cannot have the same sign. The same result can be obtained for \( q_i(\sigma) \) and \( p_r(\sigma) \). Therefore, \( q_r(\sigma)p_i(\sigma) \leq 0 \) and \( q_i(\sigma)p_r(\sigma) \leq 0 \) which proves condition (27) [20]. By multiplying both sides of (28) by \( q_i(\sigma) \) and (30) by \( q_r(\sigma) \), it is easy to see that (26) is satisfied, as well. Finally, the necessity part of the theorem is proved.

To show the sufficiency part, let a \( v \in \partial(C) \) exist which satisfies (26) and (27). Therefore, there are real numbers \( \alpha_i, \alpha_r \in [0,1] \) such that:

\[ \alpha_i(q_i(\sigma) - p_i(\sigma)) = q_i(\sigma) \]  

(30)

\[ \alpha_r(q_r(\sigma) - p_r(\sigma)) = q_r(\sigma) \]  

(31)

Assume \( \alpha = \max(\alpha_i, \alpha_r) \) in which \( \alpha_i \) and \( \alpha_r \) are defined, in the following:

\[ \alpha_i = \begin{cases} 0, & \text{if } p_i(\sigma) = q_i(\sigma) = 0 \\ \alpha, & \text{otherwise} \end{cases} \]  

(32)
\begin{equation}
\alpha_r = \begin{cases}
0, & p_r(\sigma) = q_r(\sigma) = 0 \\
\alpha_r, & \text{otherwise}
\end{cases}
\end{equation}

Clearly, \(\alpha\) satisfies (28) and (29). On the other hand, \(ap(v)+(1-\alpha)q(v)\) can be decomposed as given below, according to Theorem 1:

\begin{equation}
ap(v) + (1 - \alpha)q(v) = (ap_\sigma(\sigma) + (1 - \alpha)q_\sigma(\sigma)) \\
+ j\omega (ap_\sigma(\sigma) + (1 - \alpha)q_\sigma(\sigma))
\end{equation}

Since \(\alpha\) satisfies (28) and (29), one can conclude that \(ap(v) + (1 - \alpha)q(v) = 0\).

Therefore, it proves the sufficiency of the theorem.

Based on Theorem 2, the following method is presented to find the critical polynomials in a given polynomial segment which is noted by Method 1.

Method 1: Finding the critical polynomials in a given polynomial segment

1. Assume real polynomials \(p(z)\) and \(q(z)\) are given.
2. Find the root set of polynomial \(p_\sigma(\sigma)q_\sigma(\sigma) - p(\sigma)q(\sigma)\), namely, \(S\).
3. Remove all \(\sigma\) in \(S\) which \(p_\sigma(\sigma)q_\sigma(\sigma) + p(\sigma)q(\sigma) > 0\).
4. Remove all \(\sigma\) in \(S\) where \(|\sigma| > 1\).
5. Solve the equalities \(ap(v)+(1 - \alpha)q(v) = 0\) for all \(\sigma \in S\) considering \(v = \sigma + j\sqrt{1 - \sigma^2}\) to find \(\alpha\)'s.
6. The polynomials \(ap(v) + (1 - \alpha)q(v)\) are critical for all \(\alpha\) which are obtained in the previous step.

### 3.2 Investigating the TS property of design subspaces

In this subsection, a theorem is presented to detect the TS property of a given subspace of the design space. The theorem is mainly based on the Polya’s approach which is frequently utilized in the control literature [24]. To completely explain the concepts of the main theorem of this subsection, the Polya’s theorem which discusses the positive definiteness of homogenous polynomial matrices and Lemma 1 which discusses the robust stability of a point in the design space are presented in the following.

**Theorem 3.** (Polya’s theorem) Sala and Arino [24]. Assume \(P(w)\) is a homogenous polynomial matrix in which \(\{w_i\}_{i=1}^n\) is the set of parameters and \(L\) is a large enough positive integer number. Then, the following statements are equivalent:

i. \(\forall w \in \Delta_n : \ P(w) > 0\)

ii. Matrix coefficients of all monomials in \(\sum (w_1 + ... + w_n)^i P(w)\) are positive definite.

Polya’s theorem can be very useful to check the positive definiteness of a given homogenous polynomial matrix. As stated in the theorem, the statements are equivalent only if \(L\) is sufficiently large. Indeed, if the second statement is not satisfied for a specific \(L\), it cannot be concluded that the first statement satisfies because \(L\) may be considered larger. This fact is the main drawback of Polya’s theorem because it is not known how much the parameter \(L\) should be large.

**Lemma 1.** Lee et al. [25]. Assume \(f \in \Omega\) is an arbitrary member of the design space. Then, the closed loop model 6 is robustly stable if and only if there exists symmetric homogenous polynomial matrix \(P(\theta)\) that holds the following conditions:

\begin{equation}
\forall \theta \in \Delta_n, \forall f \in \varphi : \ P(\theta, f) > 0
\end{equation}

\begin{equation}
\forall \theta \in \Delta_n, \forall f \in \varphi : \ (A_p(\theta) + B_p(\theta)F(f)C_p)^T P^{-1}(\theta,f) (A_p(\theta) + B_p(\theta)F(f)C_p) - P^{-1}(\theta,f) < 0
\end{equation}

Note that the conditions (36) and (37) in Lemma 1 are not in LMI form which means the feasibility of these conditions cannot be investigated by convex optimization methods. The nonconvexity of these conditions can critically affect the feasibility region of the methods that apply to this problem. Thus, elementary LMI techniques are utilized to convexify the mentioned nonconvex conditions in Theorem 4.

**Theorem 4.** The convex polygonal space \(\varphi \subset \Omega\) is TS if and only if there exists a symmetric homogenous polynomial matrix \(P(\theta, f)\) consisting of monomials with sufficient large degree \(d\) that hold the following conditions:

\begin{equation}
\forall \theta \in \Delta_n, \forall f \in \varphi : \ P(\theta, f) > 0
\end{equation}
\[ \forall \theta \in \Delta_n, \forall f \in \varphi: \begin{bmatrix} -P(\theta, f) & A_p(\theta)P(\theta, f) + B_p(\theta)F(f)C_pP(\theta, f) \\ * & -P(\theta, f) \end{bmatrix} < 0 \]  

(39)

**Proof.** Assume matrix \( G(\theta, f) \) is defined as follows:

\[ G(\theta, f) = \begin{bmatrix} I & 0 \\ * & P^{-1}(\theta, f) \end{bmatrix} \]

By congruence transformation of the matrix in (39) by \( G(\theta, f) \), the following result will be equivalently obtained:

\[ \begin{bmatrix} -P(\theta, f) & A_p(\theta) + B_p(\theta)F(f)C_p \\ * & -P^{-1}(\theta, f) \end{bmatrix} < 0 \]

(41)

Using Schur theorem, the above inequality equals to:

\[
\begin{align*}
(A_p(\theta) + B_p(\theta)F(f)C_p)^T P^{-1}(\theta, f) \\
(A_p(\theta) + B_p(\theta)F(f)C_p) - P^{-1}(\theta, f) < 0
\end{align*}
\]

(42)

Therefore, the conditions are equivalent to conditions of Lemma 1 which conclude the robust stability of the model.

The feasibility of conditions (38) and (39) can be investigated by Polya’s approach, the details of which are presented in Theorem 3. The degree of monomials of \( P(\theta, f) \) is considered to be finite and noted by \( d \), in this paper.

As stated earlier in this subsection, Polya’s approach cannot equivalently assess the feasibility conditions in practical applications. Furthermore, the monomials of \( P(\theta, f) \) in Theorem 4 have no upper bound \( f \). Although (38) and (39) are stated to be equivalent to the robust stability, the equivalency will be obtained if \( d \) goes to infinity. Since the monomials cannot have infinite degrees in practical application, these conditions will be conservative. Therefore, there are two sources of conservativeness in this approach which are related to the values of parameters \( L \) (in Theorem 3) and \( d \) (in Theorem 4).

**Remark 1.** It is apparent that the conservativeness of conditions (38) and (39) reduces whenever space \( \varphi \subset \Omega \) becomes smaller. Indeed, the main idea behind the proposed design algorithm of this paper is to investigate the feasibility of conditions (38) and (39) in a small part of the design space. This fact can improve the feasibility of conditions which are used to check the TS property of each subspace of the design space.

### 3.3 Investigating the TD property of design subspaces

The proposed design algorithm in this paper aims to find the feasible solution more quickly by removing the TD regions of the design space. In the following, the required theory for TD detection of subspaces is elaborated. First, the exposed edge theorem is mentioned which demonstrates that the root space of all polynomials in a given compact convex polynomial space is a subspace of the roots of its exposed edges. This result can be used to detect the TD property of a subspace of the design space.

**Theorem 5.** (Exposed edges) Bartlett et al. [26]. Assume \( p(\omega; \mu) \) is a parametric polynomial whose parameters belong to a compact space. Then, the boundary curve of its root space belongs to the root pace of its exposed edges.

A set of indicator points will be randomly selected (shown by \( N \subset \Delta_n \)) to explore the TD property of subspaces.

**Theorem 6.** The convex polygonal space \( \varphi \subset \Omega \) is TD if there exists \( \theta \in N \) which holds:

\[
\forall f \in \partial_\omega(\varphi): d(z; f, \theta) \text{ is not Schur stable} \quad (43)
\]

\[
\forall f^{(1)}, f^{(2)} \in \partial_\omega(\varphi), \exists g(z) \in Y: g(z) \text{ is critical} \quad (44)
\]

where \( d(z; f, \theta) \) is the characteristic polynomial of the closed loop model (6) and \( Y = \text{co}(d(s; f^{(1)}, \theta), d(s; f^{(2)}, \theta)) \) is a polynomial segment.

**Proof.** Assume \( \theta \) is an arbitrary member of the evaluating set \( N \) satisfying conditions (43) and (44). All internal polynomials of \( \{d(s; f, \theta) | f \in \varphi\} \) are not Schur stable regarding conditions (43) and (44) and exposed edges theorem. Hence, the existence of \( \theta \in N \) that holds these conditions apparently completes the theorem proof.

In other words, the theorem mentions that a subspace of the design space is TD if all the polynomials on the
exposed edges of the corresponding polynomial subspace are not Schur stable and there is no critical point on its exposed edges.

3.4 The proposed algorithm

This subsection proposes the design algorithm which is the main contribution of this paper. The steps of the design algorithm can be briefly presented as follows:

Design algorithm

1. Consider \( \Omega \) to be a compact simplex space.
2. Set \( S^{(1)} = \Omega \) and \( i = 1 \).
3. Do the following steps till \( i \leq |S^{(i)}| \), otherwise terminate the algorithm.
4. Let \( s \) be the first simplex of set \( S^{(i)} \).
5. If \( s \) is TS, \( i = i+1 \), return \( s \) as the feasible solution and terminate the algorithm.
6. If \( s \) is TD, \( i = i+1 \) and go to Step 3.
7. If \( \text{lel}(s) \leq \delta \), \( i = i+1 \) and go to Step 3.
8. Divide \( s \) by halving its largest edge such that \( s = s_1 \cup s_2 \) and go to Step 3.
9. Set \( S^{(i+1)} = S^{(i)} \cup \{s_1, s_2\} - \{s\}, i = i+1 \) and go to Step 3.

To clarify the abovementioned steps, some explanatory notes are presented in the following:

The design space is initialized in the first step. Notation \( S^{(i)} \) stands for remained sub-simplexes of the primary simplex \( \Omega \) that will be subsequently investigated after the \( i \)th iteration. Steps 5 and 6 determine the TS or TD properties of the current simplex \( s \) by utilizing Theorems 4 and 6, respectively. If \( s \) is TS, the algorithm returns it as the feasible solution and finalizes the iterations. If \( s \) is TD, the algorithm marks it as TD and goes to Step 3. To confirm when the algorithm should terminate, the length of the largest edge of \( s \) is used as a terminating condition for the algorithm (in step 7, \( \text{lel}(s) \) stands for the largest edge length of \( s \)). In Step 8, \( s \) is divided into two smaller simplexes via halving the largest edge of \( s \), and these simplexes are inserted into \( S^{(i)} \), and the algorithm goes to Step 3. It is worth mentioning that the design algorithm will terminate based on a threshold value for the size of the obtained simplexes at each iteration.

In the following, some properties of the design algorithm are investigated involving the finite termination and special conservativeness. For this purpose, some notations should be defined involving \( S^{(i)}_{TD} \) and \( S^{(i)}_{\text{non-TD}} \) that contain the TD and non-TD simplexes of \( S^{(i)} \), respectively. Notice that TS and TD are not direct-opposite properties.

Theorem 7. The design algorithm will certainly terminate in up to \( \left( \frac{1}{\delta} \right)^{n_d} \) number of iterations where \( 1 \) is the largest-edge length of the primary simplex \( \Omega \), \( \delta \) is the threshold value for the largest-edge length of the generated simplexes and \( n_d \) is the number of design variables.

Proof. Let \( i = \left( \frac{1}{\delta} \right)^{n_d} \). Three possible cases can happen about the termination of the algorithm: It may reach a TS simplex before the \( i \)th iteration, all generated simplexes can be detected to be TD before the \( i \)th iteration, or the generated simplexes can be divided into \( S^{(i)}_{TD} \) and \( S^{(i)}_{\text{non-TD}} \). Obviously, the algorithm terminates before the \( i \)th iteration in the first and second cases. Thus, it suffices to prove the theorem for the third case to complete the proof.

It can be easily concluded that \( i \geq 2^{n_d \log_2 (\frac{1}{\delta})} \) which concludes the largest-edge length of all simplexes of \( S^{(i)}_{\text{non-TD}} \) is smaller than \( \frac{1}{2^{n_d \log_2 (\frac{1}{\delta})}} \delta \) based on Lemma A3 (please see Appendix A). Thus, the largest-edge length of all simplexes of \( S^{(i)}_{\text{non-TD}} \) is smaller than the threshold value \( \delta \) which implies the termination of the design algorithm based on its seventh step.

Theorem 8. The design algorithm converges to a feasible solution if the primary design simplex \( \Omega \) includes a feasible simplex such as \( s \) that satisfies the following conditions:

\[
\min_{v \in \partial(s)} \|c_v - v\| > \delta
\]

\[
\min_{v_1, v_2 \in \partial(s)} \|v_1 - v_2\| > \delta
\]

where \( c_v \) is the center of \( s \).

Proof. In the worst situation, the algorithm continues up to \( i = \left( \frac{1}{\delta} \right)^{n_d} \) number of iterations without finding any TS simplex based on Theorem 7 \( (i = \text{lel}(\Omega)) \). According to Theorem 7, the largest-edge length of all simplexes of \( S^{(i)}_{\text{non-TD}} \) is certainly smaller than \( \delta \).

It is clear that the intersection of simplex \( s \) with all simplexes of \( S^{(i)}_{TD} \) is empty due to the definition of TD and TS simplexes. This fact can be mathematically mentioned as follows:
Using (47), simplex $s$ entirely belongs to the union space of simplexes of $S_{\text{non-TD}}$. Since the smallest-edge length of $s$ is larger than $\delta$ and $\text{lel}(s) < \delta$ for all $s \in S_{\text{non-TD}}$, no vertex pair of $s$ can be inside one simplex of $S_{\text{non-TD}}$ that results in the following relation:

$$\forall \bar{s} \in S^{(i)}_{\text{TD}}: s \cap \bar{s} = \emptyset$$  \hfill (47)

Assume $c_i$ lies within simplex $s \in S^{(i)}_{\text{non-TD}}$ that differs with simplexes $\{S_v\}_{v \in \partial_i(s)}$. It is obvious that the distance between center and each vertex of $s$ is surely larger than $\delta$ based on the assumption of this theorem. Due to $\text{lel}(s) < \delta$ and condition (45), simplex $s$ cannot include any boundary point of simplex $\bar{s}$ which concludes $s$ fully belongs to $s$ (i.e., $\bar{s} \subset s$). This fact contradicts the first assumption of the proof about the failure of the design algorithm to find any TS simplex because the generated simplex $s$ completely belongs to the feasible simplex $s$ that should be detected to be TS.

Theorem 8 states that the design algorithm will surely find at least one TS simplex if and only if there is a feasible simplex in the primary design space satisfying conditions (45) and (46). It is worth mentioning that these conditions directly depend on the threshold value $\delta$ such that the size of the feasible simplex can be forced to have an appropriate volume dependent on $\delta$. In fact, the design algorithm finds all feasible parts of the primary space that are appropriately small while the LMI-based approaches ideally explore the largest convex subspace of the design space. This fact is one of the most important advantages of the design algorithm as compared to the previous ones.

### 4 SIMULATION RESULTS

In this section, four simulation examples are provided to evaluate the performance of the proposed algorithm. The first two examples analyze a specific model, while in Example 3, the results of the proposed approach are compared to the results of a group of existing approaches. The final example elaborates the ability of the design algorithm to be applied to a real application.

**Example 1.** Consider the discrete model in (1) with the following corner matrices:

$$A_1 = \begin{bmatrix} -2.18 & 0.89 \\ -1.41 & 0.12 \end{bmatrix}, \quad A_2 = \begin{bmatrix} -1.74 & 0.89 \\ -0.58 & 0.59 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 1.06 & 1.01 \\ -2.87 & -4.21 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 1.55 & 1.90 \\ -2.04 & -2.21 \end{bmatrix}$$

$$C = [4.33 - 2.06]$$

The proposed algorithm is used to find a proper controller. The algorithm proceeds sequentially in 21 iterations to reach a feasible solution in the design space which is $\text{co}([-10 0]^T, [0 10]^T]$. Figure 1 shows meshes obtained in the algorithm’s procedure. The center of the feasible region shown in iteration 21 (Figure 1C) is one of the solutions of this problem given by the following:

$$F(f) = [-0.83\, 0.83]^T$$  \hfill (50)

Figure 1D shows the TD simplexes which are detected in the whole iterations of the algorithm. TD detection of the parts of the primary space will improve the convergence rate of the algorithm because by detecting them, and there is no need to search the TD simplexes at all. In this example, a set of large sub-simplexes of the design space are detected to be TD, and deleting them enhances the convergence rate of the algorithm.

**Example 2.** This example is provided to compare the conservativeness of the proposed algorithm to a group of previous methods in a specific model. The previous methods are previous works [17, 18, 27].

Consider a discrete LTI system 1 with the following corner matrices:

$$A_1 = \begin{bmatrix} -0.46 & 0.51 & 0.42 \\ 2.10 & -0.69 & -0.92 \\ a & -1.98 & 0.62 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1.83 & -0.82 & -0.72 \\ a & -1.60 & 0.53 \end{bmatrix}$$

$$B_1 = \begin{bmatrix} -0.32 & 3.06 \\ 0.77 & 3.05 \\ 2.43 & 0.29 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -0.73 & 2.27 \\ 0.87 & 2.92 \\ 2.15 & 0.19 \end{bmatrix}$$

$$C = \begin{bmatrix} -1.14 & b \\ -0.63 & 3.77 \end{bmatrix}$$

where $a$ and $b$ are positive real numbers.
The proposed algorithm and previous methods are used to control this model for different values of $a$ and $b$ in $\{a, b\} \in [0,0.1] \times [2,2.5]$. Figure 2 shows the feasibility results of the proposed algorithm for each pair of $\{a, b\}$ defined in 51. In other words, it shows the points in the space $\{a, b\} \in [0,0.1] \times [2,2.5]$ where the proposed algorithm will converge to a stabilizing point. For instance, the algorithm can stabilize the closed loop model if $a = 0.05$ and $b = 2.25$. It must be emphasized that none of the previous methods can obtain any stabilizing point, in this example.

In the following, the feasible solution of the proposed algorithm for the defined model is presented for $a = 0$ and $b = 2$.

$$F(f) = \begin{bmatrix} 0.50 & -0.12 \\ -0.07 & 0.07 \end{bmatrix}$$

Then, the closed loop model is simulated under the assumptions of $x(0) = [2 \ 3 \ -1]^T$ and $\theta = [0.4 \ 0.6]^T$. Figure 3 shows the internal states of the closed loop model which shows that the closed loop model is stable in the mentioned situation.

**Example 3.** The feasibility performance of the proposed design algorithm in this paper is compared to some existing approaches. For this purpose, 100 polytopic uncertain models are randomly generated that are supposed to have different dimensions.

The proposed algorithm and some of the previous methods are evaluated to compare their feasibilities. Table 1 shows the number of feasible models for the methods and model's dimensions (50 random models are generated for each system's orders). Table 1 reveals that the design algorithm results in a less-conservative
behavior as compared to the previous methods. The detailed results of this algorithm are available at [https://doi.org/10.6084/m9.figshare.12103143](https://doi.org/10.6084/m9.figshare.12103143).

Notice that some of the previous methods focus on the optimal robust controller design problem. To fairly adopt these methods, they are evaluated by considering sufficiently large predefined optimal gains which causes their optimality conditions do not affect their results.

**Example 4.** This example reveals the ability of the design algorithm to be applied to real applications. For this purpose, the simplified model of vertical dynamics of a helicopter is considered which is borrowed from [28,29]:

\[
\begin{align*}
x(k+1) &= (A_p + \tilde{A}_p(\theta))x(k) + B_p u(k) \\
y(k) &= C_p x(k)
\end{align*}
\]

where \(x(k) \in \mathbb{R}^4\) consists of four internal states, namely, horizontal velocity, vertical velocity, pitch rate, and pitch angle, respectively [28]. Also, notation \(\theta \in \Delta_4\) stands for the uncertain vector of the system. System matrices \(A_p\), \(B_p\), and \(C_p\) are given from da Silva and Tarbouriech [29], as follows:

\[
A_p = \begin{bmatrix}
0.9964 & 0.0026 & -0.0004 & -0.0460 \\
0.0045 & 0.9038 & -0.0188 & -0.3834 \\
0.0097 & 0.0263 & 0.9379 & 0.1223 \\
0.0005 & 0.0014 & 0.0968 & 1.0063
\end{bmatrix},
\]

\[
B_p = \begin{bmatrix}
0.0444 & 0.0167 \\
0.2935 & -0.7252 \\
-0.5298 & 0.4726 \\
-0.0268 & 0.0241
\end{bmatrix},
\]

\[
C_p = I_4
\]

In model (53), matrix \(\tilde{A}\) plays the uncertainty role of the system with the following definition [28]:

\[
\tilde{A}_p(\theta) = \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix} \theta_1
+ \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix} \theta_2
+ \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix} \theta_3
+ \begin{bmatrix}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix} \theta_4
\]

It is apparent that these system matrices can be equivalently rewritten as given in Equations (3) and (4).

The primary simplex \(\Omega \subset \mathbb{R}^8\) is considered as follows (please note that the control problem includes 8 design variables):

\[
\Omega = 10 \begin{bmatrix}
I_8 \\
-e^T_d
\end{bmatrix}
\]

where \(I_8\) is the eighth dimensional identity matrix and \(e_d \in \mathbb{R}^8\) is a column vector whose entries are 1. The design algorithm reaches a TS simplex in the above primary design space whose center is given in the following as a robust stabilizer point:
\[ f = 1.25 \times [-1 \ 2 \ 1 \ -1 \ -1 \ -1]^T \quad (57) \]

The obtained feasible solution is implemented in the closed loop system to evaluate its performance. The simulation results are given in Figure 4 for \( x(0) = [1 \ -1 \ 2 \ -2]^T \) and all corners of the uncertain space \( \Delta \): Figure 4 demonstrates that the point in (57) robustly stabilizes the closed loop model while the convergence rate is relevantly small. To increase the convergence rate of the closed loop model, the design algorithm is exploited to solve the other design problem in which the system matrix \( (A_p + \bar{A}_p(\theta)) + B_pF(f)C_p \) is replaced by

**FIGURE 4** The states of the closed loop model controlled by (57) in Example 4. In top left \( \theta = [1 \ 0 \ 0 \ 0]^T \), top right \( \theta = [0 \ 1 \ 0 \ 0]^T \), down left \( \theta = [0 \ 0 \ 1 \ 0]^T \), and down right \( \theta = [0 \ 0 \ 0 \ 1]^T \)

**FIGURE 5** The states of the closed loop model controlled by (58) in Example 4. In top left \( \theta = [1 \ 0 \ 0 \ 0]^T \), top right \( \theta = [0 \ 1 \ 0 \ 0]^T \), down left \( \theta = [0 \ 0 \ 1 \ 0]^T \), and down right \( \theta = [0 \ 0 \ 0 \ 1]^T \)
\[ y \left( \left( A_p + \tilde{A}_p(\theta) \right) + B_p F(f) C_p \right) \] where \( \gamma > 1 \) imposes the convergence rate of the model. The design algorithm reaches the following feasible point for \( \gamma = 1.05 \).

\[ f = [0 \ 2.5 \ 2.5 \ 2.5 \ 0 \ 2.5 \ 0]^T \] (58)

Figure 5 shows the internal states of the closed loop model that is controlled by the controller whose parameters are given in (58).

Figure 5 depicts the faster convergence of the closed loop which is controlled by (58) as compared to the previous closed loop response in Figure 4.

5 | CONCLUSION

This paper studied the problem of robust SOF controller for discrete LTI systems with polytopic uncertainty. The existing methods try to find the controller parameters based on LMI aspects which are conservative. This paper presents a design algorithm which is able to search any compact convex design space. The algorithm iteratively finds the feasible solution by partitioning the design space into TD, TS, or undetermined subspaces. A novel method is presented to explore the existence of critical polynomials on a polynomial segment. Then, this concept is utilized in determining the TD subspaces in the design space. These subspaces are instantly eliminated from the design space, and the algorithm changes the way to other subspaces. Through determination of TS subspaces, the algorithm is able to find a feasible solution to the design problem. If the subspace is neither TS nor TD, the algorithm halves it and continues the same procedure in each simplex.

The numerical examples show the less conservativeness and efficiency of the design parameters in comparison to the previous methods.

For the future researches, development of the direct search algorithm to solve the robust and optimal stabilization problems of polytopic time-varying uncertain discrete LTI systems is suggested. The design algorithm can be extended to the more complicated models such as time-varying uncertain systems with some minor modifications. It is also applicable for solving optimal problems such as optimal \( H_\infty \) performance.

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

AUTHOR CONTRIBUTIONS
Roozbeh Abolpour: Conceptualization, methodology. Maryam Dehghani: Supervision, validation. Heidar Ali Talebi: Supervision, validation.

ORCID
Roozbeh Abolpour https://orcid.org/0000-0002-4268-7598
Maryam Dehghani https://orcid.org/0000-0002-8628-2699
Heidar Ali Talebi https://orcid.org/0000-0001-6038-2109

REFERENCES
1. V. Ghaifari, A model predictive scheme to integral controller design for uncertain LTI systems with nonzero reference, ISA Trans. 102 (2020), 43–55.
2. X. Zhang and H. Zhu, Robust stability and stabilization criteria for discrete singular time-delay LPV systems, Asian J. Control 14 (2012), no. 4, 1084–1094.
3. X. Ji, Z. Yang, and H. Su, Robust stabilization for uncertain discrete singular time-delay systems, Asian J. Control 12 (2010), no. 2, 216–222.
4. D. Saifi, M. Chadli, S. Labiod, and T. M. Guerra, Robust \( H_\infty \) static output-feedback control for discrete-time fuzzy systems with actuator saturation via fuzzy Lyapunov functions, Asian J. Control 22 (2020), no. 2, 611–623.
5. F. Chen, S. Kang, S. Qiao, and C. Guo, Exponential stability and stabilization for quadratic discrete-time systems with time delay, Asian J. Control 20 (2018), no. 1, 276–285.
6. L. Wang and X. Liu, Parameter-varying state feedback control for discrete-time polytopic systems, Int. J. Syst. Sci. 42 (2011), no. 6, 997–1005.
7. D. Rosinov, V. Veselý, and V. Kučera, A necessary and sufficient condition for static output feedback stabilizability of linear discrete-time systems, Kybernetik 39 (2003), no. 4, 447–459.
8. Y. Tian, W. Liu, and X. Que, Feedback stabilization of discrete time singularly perturbed systems with communication constraints, Asian J. Control 22 (2020), no. 4, 1484–1493.
9. C. M. Agulhari, R. C. Oliveira, and P. L. Peres, Robust \( H_\infty \) static output-feedback design for time-invariant discrete-time polytopic systems from parameter-dependent state-feedback gains, Proc. 2010 Am. Control Conf. (2010), 4677–4682.
10. J. Dong and G. H. Yang, Robust static output feedback control synthesis for linear continuous systems with polytopic uncertainties, Automatica 49 (2013), no. 6, 1821–1829.
11. H. R. Moreira, R. C. Oliveira, and P. L. Peres, Robust \( H_2 \) static output feedback design starting from a parameter-dependent state feedback controller for time-invariant discrete-time polytopic systems, Opt. Control Appl. Methods 32 (2011), no. 1, 1–13.
12. S. Kanev, C. Scherer, M. Verhaegen, and B. de Schutter, Robust output-feedback controller design via local BMI optimization, Automatica 40 (2004), no. 7, 1115–1127.
13. V. L. Syrmos, C. T. Abdallah, P. Dorato, and K. Grigoriadis, Static output feedback—a survey, Automatica 33 (1997), no. 2, 125–137.
14. L. Frezzatto, R. C. Oliveira, and P. L. Peres, \( H_\infty \) and \( H_2 \) memory static output-feedback control design for uncertain discrete-time linear systems, IFAC-PapersOnLine 51 (2018), no. 25, 90–95.
15. D. Mehdi, E. K. Boukas, and O. Bachelier, Static output feedback design for uncertain linear discrete time systems, IMA J. Math. Control Inf. 21 (2004), no. 1, 1–13.
16. D. Arzelier, D. Peaucelle, and S. Salhi, Robust static output feedback stabilization for polytopic uncertain systems: improving the guaranteed performance bound, IFAC Proc. Vol. 36 (2003), no. 11, 425–430.
17. C. A. Crusius and A. Trofino, Sufficient LMI conditions for output feedback control problems, IEEE Trans. Autom. Control 44 (1999), no. 5, 1053–1057.
18. X. H. Chang, J. H. Park, and J. Zhou, Robust static output feedback $H_\infty$ control design for linear systems with polytopic uncertainties, Syst. Control Lett. 85 (2015), 23–32.
19. Y. Y. Cao and K. Fang, Parameter-dependent Lyapunov function approach to stability analysis and design for polytopic systems with input saturation, Asian J. Control 9 (2007), no. 1, 1–10.
20. R. Abolpour, M. Dehghani, and H. A. Talebi, Output feedback controller for polytopic systems exploiting the direct searching of the design space, Int. J. Robust Nonlinear Control 29 (2019), no. 15, 5164–5177.
21. R. Abolpour, M. Dehghani, and M. S. Sadabadi, Designing controller parameters of an LPV system via design space exploration, Eur. J. Control 59 (2021), 47–57.
22. T. G. Büyükköröglü, G. Çelebi, and V. Dzhafarov, Stabilisation of discrete-time systems via Schur stability region, Int. J. Control 91 (2018), no. 7, 1620–1629.
23. B. Aguirre-Hernández, F. R. García-Sosa, C. A. Loredo-Villalobos, R. Villafuerte-Segura, and E. Campos-Cantón, Open problems related to the Hurwitz stability of polynomials segments, Math. Problems Eng. 2018 (2018). https://doi.org/10.1155/2018/2075903
24. A. Sala and C. Arino, Asymptotically necessary and sufficient conditions for stability and performance in fuzzy control: Applications of Polya’s theorem, Fuzzy Set. Syst. 158 (2007), no. 24, 2671–2686.
25. D. H. Lee, Y. H. Joo, and M. H. Tak, A convex optimization approach to adaptive stabilization of discrete-time LTI systems with polytopic uncertainties, Int. J. Adapt. Control Signal Process. 29 (2015), no. 9, 1116–1134.
26. A. C. Bartlett, C. V. Hollot, and H. Lin, Root locations of an entire polytope of polynomials: It suffices to check the edges, Math. Control Signals Syst. 1 (1988), no. 1, 61–71.
27. X. H. Chang and G. H. Yang, New results on output feedback $H_\infty$ control for linear discrete-time systems, IEEE Trans. Autom. Control 59 (2013), no. 5, 1355–1359.
28. W. E. Schmidtendorf, Designing stabilizing controllers for uncertain systems using Riccati equation approach, IEEE Trans. Autom. Control 33 (1988), no. 4, 36–379.
29. J. G. da Silva and S. Tarbouriech, Local stabilization of discrete-time linear systems with saturating controls: an LMI-based approach, IEEE Trans. Autom. Control 46 (2001), no. 1, 119–125.

AUTHOR BIOGRAPHIES

Roozbeh Abolpour is a Ph.D. from Electrical and Computer Engineering School of Shiraz University, in 2020. He received two B.Sc. degrees, one in Electrical Engineering, and the other in Computer Engineering, in 2012 and 2013, respectively, both from Shiraz University, and a M.Sc. degree in Control Engineering from Sharif University of Technology in 2014. His current research interests include stability and stabilisability analysis of nonlinear systems with uncertainty, bio-inspired nonlinear systems, and their applications in real dynamical systems, offshores wind turbines, and biomedical engineering problems.

Maryam Dehghani received her B. Sc. and M. Sc. degrees in electrical engineering from Shiraz university, Shiraz, Iran, in 1999 and 2002, respectively and her PhD from Amirkabir University of Technology, Tehran, Iran in 2008. She is currently an associate professor in the school of electrical and computer engineering, Shiraz university, Shiraz, Iran. Her research interests include Linear Matrix Inequality (LMI), LPV systems and their applications in power systems, energy systems and biomedical systems.

Heidar Ali Talebi received the B.Sc. degree in electrical engineering from Ferdowsi University, Mashhad, Iran, in 1988, the M.Sc. degree in electrical engineering from Tarbiat Modarres University, Tehran, Iran, in 1991, and the Ph.D. degree in electrical engineering from Concordia University, Montreal, QC, Canada, in 1997. He was a Postdoctoral Fellow and a Research Fellow at Concordia University and The University of Western Ontario. In 1999, he joined Amirkabir University of Technology, Tehran, Iran. He is currently a professor and chair in the Department of Electrical Engineering, and his research interests include control, robotics, medical robotics, fault diagnosis and recovery, intelligent systems, adaptive control, nonlinear control, and real-time systems.
How to cite this article: R. Abolpour, M. Dehghani, and H. A. Talebi, Output feedback controller design for discrete LTI systems with polytopic uncertainty via direct searching of the design space, Asian J Control 24 (2022), 3080–3094. https://doi.org/10.1002/asjc.2710

APPENDIX A

Lemma A1. If simplex \( s \subset \mathbb{R}^{n_d} \) is divided \( n_d \) times to obtain simplexes \( \{s_i\}_{i=0}^{n_d} \) in which \( s_0 = s \) and \( s_i + 1 \) is obtained through halving the largest-edge of \( s_i \) for all \( i \in \{0, ..., n_d - 1\} \), then \( \text{lel}(s_{n_d+1}) \) is smaller than \( \frac{1}{2} \text{lel}(s_0) \).

Proof. Assume \( e_{i,j} \) is the \( j \)th largest edge of simplex \( s_i \) for all \( i \in \{0, ..., n_d\} \). Based on the statement of this lemma, simplex \( s_{i+1} \) consists of edges \( \left\{ \frac{1}{2} e_{i,1}, \left\{ e_{i,j} \right\}_{j=2}^{n_d} \right\} \) which results in the following relations:

\[
\forall i \in \{0, ..., n_d\}, \forall j \in \{1, ..., n_d - 1\}:
\begin{align*}
e_{i+1,j} &= \begin{cases} 
1/2 e_{i,1}, & \|e_{i,j+1}\| < 1/2 \|e_{i,1}\| \\
e_{i,j}, & \|e_{i,j}\| < 1/2 \|e_{i,1}\|
\end{cases} \quad (A1)
\end{align*}
\]

\[
\forall i \in \{0, ..., n_d\} : e_{i+1,n_d} = \begin{cases} 
1/2 e_{i,1}, & 1/2 \|e_{i,1}\| \leq \|e_{i,n_d}\| \\
e_{i,n_d}, & \|e_{i,n_d}\| < 1/2 \|e_{i,1}\|
\end{cases} \quad (A2)
\]

Using A1 and A2, one can successively obtain the following relations:

\[
\forall i \in \{0, ..., n_d\}, \forall j \in \{1, ..., n_d - 1\}:
\|e_{i+1,j}\| \leq \max \left( \frac{1}{2} \|e_{i,1}\|, \|e_{i,j+1}\| \right) \quad (A3)
\]

\[
\forall i \in \{0, ..., n_d\} : \|e_{i+1,n_d}\| \leq \frac{1}{2} \|e_{i,1}\| \quad (A4)
\]

\[
\|e_{n_d,1}\| \leq \max \left( \frac{1}{2} \|e_{n_d-1,1}\|, \|e_{n_d-1,2}\| \right) \leq \max \left( \frac{1}{2} \|e_{n_d-2,1}\|, \|e_{n_d-2,2}\| \right) \leq \cdots \quad (A5)
\]

Since \( \text{lel}(s_n) = \|e_{n,1}\| \) and \( \text{lel}(s_0) = \|e_{0,1}\| \), inequality A5 proves the lemma.

Lemma A2. Each simplex of set \( S_{n_{\text{TD}}}^{(n_{\text{TD}})} \) such as \( s \) has a generation sequence denoted by \( G(s) = \{s_i\}_{i=0}^{n_d} \) such that \( s_{n_d} = s \) and \( s_{i+1} \) is obtained via halving the largest edge of simplex \( s_i \) for all \( i \in \{0, ..., n_d - 1\} \).

Proof. Assume none of the generated simplexes is detected to be TD which is the worst situation because all generated simplexes are needed to be halved in the algorithm’s iterations. Hence, it suffices to prove the lemma considering \( S_{n_{\text{TD}}}^{(n_{\text{TD}})} \) is empty as a worst situation.

According to the algorithm’s steps, the generated simplexes can be grouped into different levels which are the number of required halving procedures to be obtained by the primary simplex. For instance, the halved sub-simplexes of the primary simplex \( \Omega \) which are \( S_{n_{\text{TD}}}^{(n_{\text{TD}})} \) belong to the first level, and all simplexes of \( S_{n_{\text{TD}}}^{(n_{\text{TD}})} \) are inside the second level (note that none of the generated simplexes is detected to be TD). Therefore, one can iteratively conclude that all simplexes of \( S_{n_{\text{TD}}}^{(n_{\text{TD}})} \) belong to the \( n_{\text{th}} \) level which concludes the existence of the generation sequence as described in the statement of this lemma.

Lemma A3. The largest-edge length of all simplexes of set \( S_{n_{\text{TD}}}^{(n_{\text{TD}})} \) is smaller than or equivalent to \( \frac{1}{2} \text{lel}(\Omega) \).

Proof. Let \( s \) be an arbitrary simplex of \( S_{n_{\text{TD}}}^{(n_{\text{TD}})} \). According to Lemma A2, there exists the generation sequence \( G(s) \) as described in Lemma A2. The existence of this simplex sequence concludes the statement of this lemma based on Lemma A1.