Stability and Stabilization Conditions for Takagi-Sugeno Fuzzy Model via Polyhedral Lyapunov Functions

Willem Esterhuizen¹, Hua O. Wang², Kazuo Tanaka³ and Xiangzhou Wang⁴

Abstract— Polyhedral Lyapunov functions (PLFs) are universal for establishing stability of Takagi-Sugeno (T-S) fuzzy models. In this paper, a stability theorem via PLFs is presented for T-S models, and it is shown that stability can be established via linear programming. Furthermore, nonconvex stabilization conditions are stated that, if satisfied, specify a parallel distributed compensation (PDC) controller as well as a PLF which proves stability of the closed loop system. An algorithm is presented as an initial step in working around the nonconvex stabilization conditions, and has shown to be useful in the computation of PDC controllers.

I. INTRODUCTION

The Takagi-Sugeno (T-S) fuzzy model together with parallel distributed compensation forms a very effective framework for modeling, analysis and control design for nonlinear systems. A large body of theory exists that deals with this framework and most of the fundamental notions, such as stability, stabilization, observer design, robustness, optimality etc. have been studied extensively: [18]. A large number of the well-established results have been derived from an initial choice of candidate Lyapunov function (LF): the quadratic function. The main reason for this is that the stability and design conditions under quadratic LFs are in the form of linear matrix inequalities (LMIs) which are easily solvable.

However, quadratic LFs are conservative: the conditions under which a quadratic function is a LF for a T-S model are only sufficient. This can be shown by an example; see [5] and [14]. The class polyhedral Lyapunov functions (PLF) are known to be universal for establishing stability of linear polytopic differential inclusions (LPDI), [10], and this is the main motivation for investigating the this function’s applicability to the T-S model.

The work in this paper is based on findings from the master’s thesis [5] (obtainable from the authors). In that work, the application of PLFs to the T-S model via two methods was investigated: the one based on findings from [11], [12], [13], [14] for LPDIs and the other based on findings from [1] and [2] for uncertain linear systems. In this paper only the results for the method based on the work of Polanski will be presented, as, via this approach, it was possible to come up with a stabilization algorithm.

The paper is structured as follows: section II presents relevant theoretical preliminaries. Section III presents stability conditions for T-S models via PLFs, followed by PDC stabilization conditions in section IV. Section V looks at an algorithm to deal with the nonconvex stabilization conditions. Section VI summarizes the main results and points out future research.

II. PRELIMINARIES AND RELEVANT THEORY

A. Polyhedral Stability of LTI Systems

The candidate Lyapunov function considered in this paper is the infinity norm:

\[ V(x) = \|Wx\|_\infty \]

where \( W \in R^{m \times n}, m \geq n \), and \( \text{rank}(W) = n \). The sublevel sets of this function are positively homogeneous, 0-symmetric polytopes. The unit sub-level set will be denoted by \( \varphi(W) \triangleq \{ x \in R^n : \|Wx\|_\infty \leq 1 \} \). The rows of \( W \) specify the hyperplanes of the unit level-set.

An autonomous system [7] is described by

\[ \dot{x}(t) = f(x) \]

where \( f : D \rightarrow R^n \) is a locally Lipschitz map from a domain \( D \subset R^n \) into \( R^n \).

A linear time-invariant system is given by

\[ \dot{x}(t) = Ax(t) \]

where \( A \in R^{n \times n} \). \( x^* \) is an equilibrium point for an autonomous system if \( f(x^*) = 0 \).

A linear polytopic differential inclusion [10] is given by:

\[ \dot{x} \in F_r(x), \quad F_r(x) = \{ y : y = Ax, A \in \text{co}(A_1, ..., A_r) \} \]

where \( \text{co}(\cdot) \) denotes the convex hull of the elements \( A_1 \) to \( A_r \), i.e., \( A = \sum_{i=1}^{r} h_i A_i \) where \( \sum_{i=1}^{r} h_i = 1, h_i \geq 0 \).

Definition 1: Upper-Directional Derivative [9]

The upper-directional derivative of \( V \) w.r.t. (2), in the direction given by the vector \( f(x) \) is defined as follows:
\[ D^+_f(x)V(x) \triangleq \limsup_{h \to 0^+} \frac{V(x + hf(x)) - V(x)}{h} \tag{4} \]

**Definition 2: Polyhedral Stability (Adapted from [1])**

The linear time-invariant system (3) is said to be polyhedrally stable if there exists an infinity norm LF (1) such that the upper-directional derivative of V with respect to (3), in the direction given by Ax is negative definite:

\[ \dot{V}(x) = \limsup_{h \to 0^+} \frac{V(x + hAx) - V(x)}{h} < 0 \tag{5} \]

The symbol \( \prec \) indicates that the function is negative definite.

**B. Takagi-Sugeno (T-S) Fuzzy Model Approach**

A nonlinear system is approximated by a T-S model by defining \( r \) “extreme subsystems”, \( r \) fuzzy if-then rules as well as fuzzy membership functions. The dynamics of the system at any time is then contained within the convex hull of the subsystems. See [18] for excellent treatment of the T-S model. What is important is to note that the dynamics are described by

\[ \dot{x}(t) = \sum_{i=1}^{r} h_i(z(t))\{A_i x(t) + B_i u(t)\} \tag{6} \]

where \( \sum_{i=1}^{r} h_i = 1 \), \( h_i \geq 0 \). \( z_1(t),...,z_p(t) \) are the known “premise variables” and can be functions of the state (most often), external disturbances and/or time. The T-S model with \( u(t) = 0 \) is therefore an LPDI where the weighting that each subsystem contributes to the convex combination changes with the evolution of the system.

An effective controller for stabilizing a T-S model is through the technique called “parallel distributed compensation (PDC)” [19], [20]. In PDC scheme, \( r \) “extreme” full-state feedback matrices are specified, and along with the same if-then rules and fuzzy membership functions, a controller is specified as being contained in the convex hull of these extreme feedback matrices. More clearly: \( u(t) = \sum_{j=1}^{r} h_j F_j x(t) \), \( \sum_{j=1}^{r} h_j = 1 \), \( h_j \geq 0 \). The closed loop system then becomes:

\[ \dot{x} = \sum_{i=1}^{r} h_i^2 G_{ii} + \sum_{i=1}^{r} \sum_{i<j} h_i h_j \left( \frac{G_{ij} + G_{ji}}{2} \right) \]

where \( G_{ij} \triangleq (A_i - B_i F_j) \). With a PDC controller, the dynamics are contained within the convex hull of the extreme matrices \( G_{i}, i = 1,\ldots,r \) and \( \frac{G_{ij} + G_{ji}}{2} \), \( i = 1,\ldots,r, i < j \), because with \( h_i \geq 0 \), \( \sum_{i=1}^{r} h_i = 1 \).

In [18, Ch.14], it is shown that T-S models are universal approximators: any smooth nonlinear system can be approximated by a T-S model, and any smooth nonlinear state feedback controller can be approximated by a PDC controller, under some mild conditions.

**III. Stability of T-S Fuzzy Models via Polyhedral Lyapunov Functions**

Molchanov and Pyatnitskiy showed that for an equilibrium point of an LPDI to be stable, it is necessary and sufficient that there exists a PLF (1) (with a sufficiently large \( m \)) [10]. As discussed in the previous section, the T-S model is an LPDI where the weighting of each subsystem changes with the evolution of the system. Therefore, this significant result also holds for the T-S model.

In [10] algebraic necessary and sufficient conditions for an infinity norm to be a Lyapunov Function for an LPDI were also derived. This theorem, which is also applicable to the T-S model, is stated next.

**Theorem 1**: The function \( V(x) = \|W x\|_{\infty} \) is a global polyhedral Lyapunov function for the T-S model (6) with \( u(t) = 0 \) if there exists \( r \) \( (m \times m) \)-matrices \( Q_i \) such that

\[ WA_i = Q_i W, i = 1,\ldots,r \tag{8} \]

2) for each matrix \( Q_i \):

\[ q_{gg} + \sum_{j=1, j \neq g}^{m} |q_{gj}| < 0 \forall g = 1,\ldots,m \tag{9} \]

The theorem states that the PLF needs to be common to all the subsystems of the T-S model.

See the master’s thesis [5, Ch.3] for a sufficiency proof of this theorem that explicitly uses the T-S model. As a sketch: the sufficiency proof shows that condition (8) implies that:

\[ \dot{V}(x) \leq \mu \left( \sum_{i=1}^{r} h_i(t) Q_i \right) \|W x\| \]

where \( \mu (\cdot) \) is the logarithmic norm:

\[ \mu (Q) = \max_{g} \left( r e(q_{gg}) + \sum_{l=1, l \neq g}^{m} |q_{gl}| \right) \]

Condition (9) then implies that \( \dot{V}(x) \prec 0 \), which implies that the origin of the T-S model is globally polyhedrally asymptotically stable.

In [14], Polanski presents a necessity proof of this theorem for LTI systems as well as LPDIs. This result also holds for the T-S model and enables the computation of PLFs for T-S models via linear programming (LP). This LP approach is presented in the next section for completeness and because the stabilization result, which is the main contribution of the paper, builds on this method.

**A. Establishing Stability via linear programming**

This means of computing PLFs for LPDIs was derived in [14]. Consider the function \( V(x) = \|D^{-1} U x\| \) where \( U \) is chosen such that \( \varphi(U) \) approximates the unit sphere. This is trivial in 2 dimensions: \( U = [\cos \left( \frac{i-1}{m} \pi \right), \sin \left( \frac{i-1}{m} \pi \right)] \in \mathbb{R}^{m \times 2}, i = 1,\ldots,m \). Let \( D = \text{diag}(d_1, d_2,\ldots,d_m) \in \mathbb{R}^{m \times m} \), and \( d_i > 0 \). Then
\(V \leq 1\) specifies an approximation of the unit sphere with faces that are then scaled by the entries of \(D\).

As mentioned in [14], it is reasonable to consider \(U\) as an approximation of the unit sphere, as the goal is to find a function \(V(x)\) such that \(\varphi(D^{-1}U)\) approximates some unknown convex shape.

Let
\[
Q_{D,i} = D^{-1}Q_iD = \frac{d_j}{d_g} q_{gj}
\]

\(Q_{D,i}\) and \(W = D^{-1}U\) satisfying (8) is equivalent to \(Q_i\) and \(U\) satisfying (8):
\[
WA_i = Q_{D,i}W \quad i = 1, \ldots, r
\]
\[
\Leftrightarrow D^{-1}UA_i = D^{-1}Q_iDD^{-1}U \quad i = 1, \ldots, r
\]
\[
\Leftrightarrow UA_i = Q_iU \quad i = 1, \ldots, r
\]

Therefore, for \(V(x) = \|Wx\|_\infty\) to be a PLF for the system (6) with \(u(t) = 0\), the matrix \(U\) and \(r\) matrices \(Q_i, (i = 1, \ldots, r)\) need to satisfy (10), and each matrix \(Q_{D,i}\) needs to satisfy (9), rewritten as: (11).

\[
q_{gg} + \sum_{j=1, j \neq g}^{m} \frac{d_j}{d_g} |q_{gj}| < 0 \forall g = 1, \ldots, m
\]

**Linear Program 1:**

In [14] it was shown that feasible solutions to the \(m\) LPs \((g = 1, \ldots, m)\):

\[
\begin{align*}
\min_{\lambda} & \quad \lambda^T \mathcal{I}(2m, m + g) \\
\text{s.t.} & \quad [U \quad -U] \lambda = A^T[U_{g\bullet}]^T \\
& \quad \lambda \geq 0
\end{align*}
\]

produces a matrix \(Q_i\) that satisfies (10) for a matrix \(A_i\). \(\mathcal{I}(2m, m + g)\) denotes the \(2m\)-column vector with all its elements equal to 1, except for the \((m+g)\)-th element which is set to -1. \(U_{g\bullet}\) is \(U\)'s \(g\)-th row. In [14] it was shown that:

\[
\lambda_j - \lambda_{m+j} = q_{gj}, \quad j = 1, \ldots, m
\]

\[
\lambda_j + \lambda_{m+j} = |q_{gj}|, \quad j = 1, \ldots, m, \quad j \neq i
\]

The entries of the \(g\)-th row of a matrix \(Q_i\) are thus specified by the solution of the \(g\)-th LP, and (12) and (13).

Running these \(m\) LPs \(r\) times (i.e. for each \(A_i\) matrix) will then produce the \(r\) \(Q_i\) matrices required to satisfy (10).

Now, values for the elements of the matrix \(D\) need to be computed in order for (11) to be satisfied by each \(Q_{D,i}\). This can again be done via linear programming:

**Linear Program 2:**

Find a \(d = [d_1, \ldots, d_m]^T\) such that

\[
\begin{bmatrix}
|Q_1| \\
\vdots \\
|Q_r|
\end{bmatrix} d < 0
\]

is feasible. \(|Q_i|\) is defined to be the matrix \(Q_i\) with all its entries replaced by their absolute value, except for the diagonal entries.

In [14] an example is worked through where an LPDI’s stability is proven via a PLF that was computed using this LP approach. The example is equally applicable to the T-S model, and it will not be shown here in the interest of conciseness.

**IV. Stabilization of T-S Fuzzy Models via Polyhedral Lyapunov Functions**

Methods have been specified for computing PLFs along with Lipschitz variable structure controllers for LPDIs [3], [4]. Here a new stabilization theorem is derived which presents conditions that, if satisfied, specify a PLF as well as a PDC controller. To the author’s knowledge, this is a new contribution to T-S fuzzy theory.

**Theorem 2:** The function \(V(x) = \|Wx\|_\infty\) is a global polyhedral Lyapunov function for the T-S model (6) with \(u(t) = -\sum_{j=1}^{r} h_j F_j x(t), \sum_{j=1}^{r} h_j = 1, h_j \geq 0\) iff there exists \((m \times m)\)-matrices \(Q_i\) and \(Q_{ij}\); and \(r\) \((p \times n)\)-matrices \(F_j\) such that

1) \(WG_{ii} = Q_iW \quad i = 1, \ldots, r\)

2) for each matrix \(Q_i\) and \(Q_{ij}:

\[
q_{gg} + \sum_{h=1, h \neq g}^{m} |q_{gh}| < 0 \forall g = 1, \ldots, m
\]

where \(G_{ij} \triangleq (A_i - B_i F_j)\)

See the master’s thesis [5, Ch.4] for a sufficiency proof of this theorem that explicitly uses the T-S model. The sufficiency proof uses the fact that the closed loop system is also an LPDI with the extreme matrices now specified as \(G_{ii}, i = 1, \ldots, r\) and \((G_{ij} + G_{ji}), i = 1, \ldots, r\). In [14] (which exposed the LP approach to the computation of PLFs) can also be modified to prove necessity of this stabilization theorem.

In the stability section a candidate Lyapunov function \(V(x) = \|Wx\|_\infty\) with \(W = D^{-1}U\) was considered. The problem then became that of solving for the \(Q\) matrices through Linear Program 1, and then to use the computed \(Q\)’s to solve for \(D\) through Linear Program 2.

In the stabilization problem there are also \(r\) \(F\) matrices that need to be solved along with the \(Q\)’s and the \(D\). As in the stability section, if the matrices
\[ Q_{D,i} = D^{-1}Q_iD = \frac{d_h}{d_g} q_{gh} \]

and

\[ Q_{D,ij} = D^{-1}Q_{ij}D = \frac{d_h}{d_g} q_{gh} \]

are considered then \( Q_{D,i} \) and \( W = D^{-1}U \) satisfying (14), and \( Q_{D,ij} \) and \( W = D^{-1}U \) satisfying (15) is equivalent to: \( Q_i \) and \( W = U \) satisfying (14), and \( Q_{ij} \) and \( W = U \) satisfying (15). For the function \( V(x) = \|D^{-1}U\| \) to now be a PLF, each \( Q_{D,i} \) and \( Q_{D,ij} \) needs to satisfy (16), i.e.:

\[
q_{gg} + \sum_{h=1, h\neq g}^{m} \frac{d_h}{d_g} |q_{gh}| < 0 \quad \forall g = 1, \ldots, m \quad (17)
\]

It is desired to solve the stabilization problem by simultaneously finding entries of the \( Q \)'s, \( F \)'s and the \( D \) that satisfy the stabilization conditions of theorem 2.

Recall that the constraints of Linear Program 1 took the form:

\[
\begin{bmatrix}
U \\
-U
\end{bmatrix}^T \lambda = A^T[U_i^*]^T
\]

\[
\lambda \geq 0
\]

where \( \lambda_l - \lambda_{m+l} = q_{gl} \), and \( \lambda_l + \lambda_{m+l} = |q_{gl}| \), \( l = 1, \ldots, m \), \( l \neq g \). With \( A \) replaced with \( G = A - BF \):

\[
\begin{bmatrix}
U \\
-U
\end{bmatrix}^T \lambda = (A - BF)^T[U_i^*]^T
\]

This can be re-written as:

\[
\begin{bmatrix}
U \\
-U
\end{bmatrix}^T \lambda + T(i)f = (U_i^*A)^T
\]

\[
\lambda \geq 0
\]

\[
f \text{ unbounded}
\]

with

\[
T(i) = \begin{bmatrix}
U_{i*}B & 0 & 0 & \ldots & 0 \\
0 & U_{i*}B & 0 & \ldots & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \ldots & U_{i*}B
\end{bmatrix}
\]

and

\[
f = [f_{11}, f_{21}, \ldots, f_p, f_{12}, f_{22}, \ldots, f_{p2}, \ldots, f_{1n}, f_{pn}],
\]

\[
F = \begin{bmatrix}
f_{11} & \ldots & f_{1n} \\
\vdots & \ddots & \vdots \\
f_{p1} & \ldots & f_{pn}
\end{bmatrix}
\]

Constraints (17) can be written in terms of \( \lambda \)'s from (12) and (13). With these observations, a feasibility problem (FP) can now be formed, the solution of which will specify the \( Q \) matrices, the \( r \) feedback matrices \( F_i \)'s, as well as the PLF that proves stability of the closed loop system.

Recall that a solution to LP1 presents a row of a \( Q \) matrix, and that this LP needs to be run \( m \) times to produce one \( Q \) matrix. This then needs to be done \( r \) times, to produce all the \( Q \) matrices. These LP constraints can in fact be “stacked” along with the constraints on the feedback matrices as well as the constraints (17), to form the FP.

**Feasibility Problem 1:**

\[
\text{Find } \lambda, f \text{ subject to }
\]

\[
A_{eq} \lambda = b_{eq}
\]

\[
A_{ineq} \lambda < 0
\]

\[
\lambda \geq 0
\]

\[
f \text{ unbounded}
\]

\[
\Lambda = [\lambda; f]^T,
\]

where \( A_{eq} \) and \( b_{eq} \) are appropriately defined so as to enforce conditions (14) and (15) (with \( W = U \)), and \( A_{ineq} \) is appropriately defined so as to enforce condition (16), with \( Q_{D,i} \) and \( Q_{D,ij} \) (i.e. condition (17)) [5, Ch.4, Sec. 4.3.2] shows an example with the matrices fully specified that clarifies the approach.

The problem lies with the nonconvex constraints (17) as this prevents finding a solution via linear programming. In the next section, a few observations regarding these constraints are made that can be exploited to arrive at an algorithm for simultaneously computing a PLF as well as the feedback matrices.

V. STABILIZATION ALGORITHM

Fact 1: With \( q \in \mathbb{R} \) and \( |q_h| \geq 0 \): \( q + \sum_{h=1}^{m} a_h |q_h| < 0 \Rightarrow q + \sum_{h=1}^{m} a_h |q_h| < 0 \) for all \( a_h \leq \bar{a}_h \).

Note also that each of the constraints \( q_{gg} + \sum_{h=1, h\neq g}^{m} \bar{a}_{h,g} |q_{gh}| < 0 \), \( g = 1, \ldots, m \) becomes stricter with larger \( \bar{a}_{h,g} \).

Each one of the values \( \frac{d_h}{d_g} \) specifies a ratio between the scaling of two hyperplanes that make up the sub-level set \( \psi(W) \), where \( W = D^{-1}U \). By replacing each of the values \( \frac{d_h}{d_g} \) by a constant, (17) becomes linear, and thus convex, with respect to the \( \lambda \)'s.

If each of the ratios \( \frac{d_h}{d_g} \) are bounded:

\[
\bar{a}_{h,g} \leq \frac{d_h}{d_g} \leq \bar{a}_{h,g} \quad \forall h > g
\]

then (17) can be made linear in the \( \lambda \)'s by setting \( \frac{d_h}{d_g} = \bar{a}_{h,g} \) if \( h > g \) and setting \( \frac{d_h}{d_g} = \frac{1}{\bar{a}_{h,g}} \) if \( h < g \).

By setting each \( \bar{a}_{h,g} \) to a very small value and each \( \bar{a}_{h,g} \) to a very large value, each of the constraints in (17) will be very relaxed. However, this conflicts with the requirement that \( \bar{a}_{h,g} \leq \frac{d_h}{d_g} \leq \bar{a}_{h,g} \).

The idea behind the algorithm is to start with very large and very small values of \( \bar{a}_{h,g} \) and \( \bar{a}_{h,g} \), \( h > g \) respectively,
as this provides the greatest relaxation of the constraints (17). This makes it the most likely that a feasible solution will be found to FP1 without regard as to whether \( a_{h,g} \) and \( \pi_{h,g} \) \( h > g \) satisfy (18). Values of the \( \pi_{h,g} \)'s and \( a_{h,g} \)'s are then found that satisfy both FP 1 and (18) by iteratively increasing or decreasing the \( \pi_{h,g} \)'s and \( a_{h,g} \)'s and checking whether there exists a feasible solution to FP1 and (18) for each iteration. Fact 1 makes it possible to do this “searching” via binary search.

The algorithm is loosely specified as follows:

**Algorithm 1**

1. Identify the relations between the \( a_{h,g} \)'s and \( \pi_{h,g} \)'s that guarantee a solution to (18). (i.e. such that there exists \( d_1, d_2, \ldots, d_m \) that satisfy (18))
2. Define values \( \text{max} \) and \( \text{min} \)
3. Let \( a_{h,g} = \text{max} \) for \( h > g \)
4. Let \( \pi_{h,g} = \text{min} \) for \( h > g \)
   {Steps 3 and 4 relax the constraints (17) as much as possible with the defined \( \text{max} \) and \( \text{min} \)}
5. Choose a particular \( h, g \) pair, say \( h_1, g_1 \), and find the smallest \( a_{h_1,g_1} \), within some error, such that there exists a solution to FP 1. {Fact 1 allows this to be done through a binary search}
6. With this value of \( a_{h_1,g_1} \), find the largest corresponding \( \pi_{h_1,g_1} \) such that there exists a solution to FP 1, again through binary search.
7. **if** The largest \( \pi_{h_1,g_1} < \) the smallest \( a_{h_1,g_1} \) **then**
   8. Relax the constraints (17) by increasing \( a_{h_1,g_1} \). Go to step 6
9. **else**
10. Continue
11. **end if**
12. With these calculated values of the particular \( a_{h_1,g_1} \) and \( \pi_{h_1,g_1} \), choose a new \( h, g \) pair, say \( h_2, g_2 \), and find the smallest \( a_{h_2,g_2} \) and largest \( \pi_{h_2,g_2} \) such that there exists a solution to FP 1, again through binary search.
13. **if** If the largest \( \pi_{h_2,g_2} < \) smallest \( a_{h_2,g_2} \) **then**
14. Relax the constraints (17) by increasing \( a_{h_1,g_1} \) and/or decreasing \( \pi_{h_1,g_1} \). Go to step 12.
15. **else**
16. Continue
17. **end if**
18. Continue finding smallest/largest \( a_{h,g} \)'s and \( \pi_{h,g} \)'s as specified in the previous steps for all the \( h, g \) pairs. If necessary, relax the constraints (17) by increasing/decreasing previously calculated \( a_{h,g} \)'s and/or \( \pi_{h,g} \)'s, and loop as in previous if statements
19. If there is a solution to FP 1 and the relations between the \( a_{h,g} \)'s and \( \pi_{h,g} \)'s are as required from step 1, a solution has been found to the stabilization problem.

**Comment 1:** The candidate infinity-norm LF is positively homogeneous: i.e. \( V(\mu x) = \mu V(x) \) \( \forall \mu \geq 0 \). Therefore, one only needs to consider the unit sub-level set when making stability deductions. Therefore, it is intuitive to consider the ratios between the scaling of each of the hyperplanes of the sub-level set, as specified by the constraints (18).

**A. Stabilization Implementation Example**

In this section, the two-rule fuzzy model of the inverted pendulum on a cart system from [20] will be stabilized via a PDC controller and a three plane PLF. Details which are discussed in [20] regarding the T-S model of this system have been omitted in the interest of conciseness. The extreme matrices are given by:

\[
A_1 = \begin{bmatrix}
0 & 1 \\
\frac{2g}{4f^2 - aml} & 0
\end{bmatrix}, \quad A_2 = \begin{bmatrix}
0 & 1 \\
-\frac{2g}{\pi(4f^2 - aml/\beta^2)} & 0
\end{bmatrix}
\]

\[
B_1 = \begin{bmatrix}
0 \\
-\frac{g}{4f^2 - aml}
\end{bmatrix}, \quad B_2 = \begin{bmatrix}
0 \\
-\frac{g}{4f^2 - aml/\beta^2}
\end{bmatrix}
\]

with \( m = 2kg \), \( M = 8kg \), \( 2l = 1m \), \( a = \frac{1}{m+M} \) and \( \beta = 88^\circ \). For this example, three Q matrices need to be computed, along with two \( F \) feedback matrices as well as the three entries of the \( D \) matrix.

Let the bounds on the ratios of the \( d \)'s be specified as follows: \( a_{h_1} \leq \frac{d_h}{d_1} \leq a_{h_2} \); \( a_{h_3} \leq \frac{d_h}{d_2} \leq a_{h_4} \); \( a_{h_5} \leq \frac{d_h}{d_3} \leq a_{h_6} \);

In step 1 of the algorithm, the relations between the \( a_{h,g} \)'s and \( \pi_{h,g} \)'s must be as follows: \( a_{h_1} \leq \pi_{h,1} = \pi_{h,2} \leq \pi_{h,3} \); \( a_{h_4} \leq \pi_{h,4} = \pi_{h,5} \leq \pi_{h,6} \); \( a_{h_7} \leq \pi_{h,7} \) to guarantee a solution to (18).

The algorithm was implemented with \( \text{max} = 100 \) and \( \text{min} = 0.01 \). With each relaxation of the constraints (17) the other \( a_{g,h} \)'s were increased/decreased by 0.5. The \( f \)'s, which specify the feedback matrices, were also bounded for smaller gains in the final solution.

The following solution was found:

\[
F_1 = \begin{bmatrix}
-1470.28 & -5000
\end{bmatrix}, \quad F_2 = \begin{bmatrix}
-3143.29 & -5000
\end{bmatrix}
\]

\[
D = \begin{bmatrix}
0.0204 & 0 & 0 \\
0 & 0.01 & 0 \\
0 & 0 & 0.0307
\end{bmatrix}
\]

As a check, note that each of the subsystems \( G_{11}; G_{22}; G_{12} + G_{21} \) are stable matrices. Recall that this is a necessary condition from Theorem 2.

In Fig. 1 each of the subsystems’ trajectories are simulated starting from each of the PLF’s level set’s vertices. It can be shown, see for e.g. [5], that these trajectories staying in the set prove the set’s invariance with respect to the T-S model. Note that some of the trajectories run almost flush against the hyperplanes of the PLF’s level set before making their way to the origin.
VI. CONCLUSIONS

This paper presented stability and stibilization theorems via polyhedral Lyapunov functions for the T-S fuzzy model. An algorithm was presented that worked around the nonconvex conditions of the stabilization theorem and allows computation of PDC controllers.

T-S fuzzy models, along with PDC controllers, are known to be universal approximators of smooth nonlinear systems with smooth state feedback controllers. The results of this paper are potentially significant due to the fact that PLFs are universal for proving stability of T-S fuzzy systems.

These theorems, along with the linear programming approach to computing PLFs for T-S models and the stabilization algorithm, are valid for general n-dimensional systems.

At present, however, applying these results to systems of dimension higher than 2 is not a routine task; the reason being that it is a challenge to specify a matrix $U$ such that $\psi(U)$ is an n-dimensional approximation of the unit sphere. This is a possible focus of further research. In [11], a method is presented by which a 3 dimensional LPD is proven to be stable via a PLF.

REFERENCES