Control Design and Analysis for Discrete Time Bilinear Systems Using Sum of Squares Methods

Mohsen Vatani, Morten Hovd, and Sorin Olaru

***Abstract*— In this paper, stabilization of discrete time bilinear systems is investigated by using Sum of Squares (SOS) programming methods and a quadratic Lyapunov function. Starting from the fact that global asymptotic stability cannot be proven with a quadratic Lyapunov function if the controller is polynomial in the states, the controller is instead proposed to be a ratio of two polynomials of the states. First, a simple one-step optimal controller is designed, and it is found that it is indeed defined as a ratio of two polynomials. However, this simple controller design does not result in any stability guarantees. For stability investigation, the Lyapunov difference inequality is converted to a SOS problem, and an optimization problem is proposed to design a controller which maximizes the region of convergence of the bilinear system. Input constraints can also be accounted for in the optimization problem.**

I. INTRODUCTION

In this paper, bilinear discrete-time systems are considered. Bilinear systems are a special class of nonlinear systems, where the nonlinearity consists of products between the states and inputs. Although bilinear systems may be said to be a class of nonlinear systems that is 'close' to linear systems, linearization results in neglect of the main challenge in controller design for these systems. They have many practical applications in various fields (power systems as an example [1]) and many nonlinear systems could be approximated by bilinear models.

A substantial number of works have been devoted to control and analysis of continuous time bilinear systems over the last fifty years. A representative overview of these works is beyond the scope of this paper, but some inspiration from Gutman [2] is acknowledged. In his paper, Gutman introduces a quadratic stabilizing feedback controller and applied to a biochemical process. Compared to the amount of published work on continuous time bilinear systems, there are relatively few publications on discrete time bilinear systems. Most works have been devoted to model predictive control, e.g., [3], [4]. In [5] a nonlinear state feedback control based on passivity design has been proposed to asymptotically stabilize a neutrally stable system. References [6] and [7] investigate the constrained and unconstrained stabilization of discrete time bilinear systems using polyhedral Lyapunov functions and a systematic method for designing a stabilizing linear state feedback control is introduced. This method is used in [8] to handle a discrete bilinear system with additive bounded disturbances. In [9], controllers for discrete time bilinear systems are designed which take the form of a ratio of two polynomials in the state. The same functional form of the controller is used in the present paper, but the design procedure is different (being based on Sum of Squares programming) and can also handle systems with multiple inputs.

Sum of Squares (SOS) programming is a technique for testing non-negativity of polynomial functions, and can be used in implementation of system analysis tools. In [10] a general framework for analyzing nonlinear systems is discussed by using SOS programming method. There are also a few of papers proposing SOS programming for control of continuous time bilinear systems, see, e.g., [11], [12] and references therein. In [6] it is stated that *'It is not surprising that very few works dealing with the stabilization problem of discrete-time systems have been reported. This is due to the fact that quadratic functions which can be viewed as the "natural" Lyapunov functions for linear systems lead to very complex computational problems when applied to nonlinear discrete-time systems.'* Nevertheless, the present paper comes back to the class of quadratic Lyapunov functions for discrete time bilinear systems. Although the complexity of the resulting computational problems is undisputable, it is found that software for SOS programming are now of a quality that makes this technique useful and relatively accessible. The software package YALMIP [13], [14] has been used for all SOS problems in this paper. To the best of the authors' knowledge, this is the first work specifically addressing the stabilization of discrete time bilinear systems using SOS programming.

This paper is organized as follows: in section II, the problem is defined and preliminary information is provided. The proposed controller is defined as the ratio of two polynomials and quadratic Lyapunov function is considered for stability investigation. In the next section, a direct one-timestep optimal controller design is considered, optimizing the one step ahead tracking error and input cost. It is found that the structure of the controller agrees with the findings in section II, but no stability guarantee is given. In section IV, an SOS optimization problem is proposed to design a controller which maximizes the region of convergence of the bilinear system. This region is defined as a level set of the quadratic Lyapunov function, and input constraints may be accounted for in the problem formulation. In section V, an optimization problem is developed to improve the rate of convergence inside a given Lyapunov function level set.

Mohsen Vatani and Morten Hovd are with the Engineering Cybernetics Department, Norwegian University of Science and Technology, N-7491 Trondheim, Norway mohsen.vatani,morten.hovd@itk.ntnu.no

Sorin Olaru is with SUPELEC Systems Science, Automatic Control Department and EPI INRIA DISCO, 3 rue Joliot Curie, 91192 France sorin.olaru@supelec.france

Three illustrative numerical examples are provided in section VI. Then paper ends with conclusions and discussion on further work.

II. PROBLEM STATEMENT AND PRELIMIARIES

The dynamic of class of discrete bilinear systems under interest in this note is described by the following difference equation:

$$x_{k+1} = Ax_k + \sum_{i=1}^{m}(B_i x_k + b_i)u_{i,k} \tag{1}$$

where $x_k \in \Re^n$ is the state vector at time k, $u_{i,k}$ is the ith element of input vector $u_k \in \Re^m$ at time k, and $A \in \Re^{n\times n}, B_i \in \Re^{n\times n}, b_i \in \Re^{n\times 1}$ are system matrices. For the sake of simplicity of notation, the system dynamics may alternatively be expressed as:

$$x_{k+1} = Ax_k + (B_x + B)u_k \tag{2}$$

where $B_x = [\ B_1 x_k\ B_2 x_k\ \cdots\ B_m x_k\ ]$ and $B = [\ b_1\ b_2\ \cdots\ b_m\ ]$.

The problem considered here is the stabilization of the bilinear system (1) to the origin by designing a controller which satisfies input constraints. The main tool for analyzing stability of nonlinear systems is Lyapunov's direct method. This is well known in the control community, and the following theorem is therefore stated without proof:

Theorem 1. *Lyapunov uniform asymptotic stability for discrete time systems: If in a neighborhood D of the equilibrium state $x = 0$ of the discrete time system $x_{k+1} = f(x_k)$, there exist a function $V(.): D \to \Re$ such that:*

$$W_1(x) \le V(x) \le W_2(x) \tag{3}$$

where W_1 and W_2 are time invariant positive definite functions (with $W_1(0) = W_2(0) = 0$) , and the rate of change $\Delta V(x_k) = V(x_{k+1}) - V(x_k)$ is negative definite in D, then the equilibrium state is uniformly asymptotically stable in D.

A quadratic Lyapunov function $V_k = x_k^T P x_k$ is often used for some given weighting matrix $P > 0$. The first assumption in Theorem 1 is satisfied for this function by considering:

$$\lambda_{min}(P)\,\|x_k\|_2^2 \le x_k^T P x_k \le \lambda_{max}(P)\,\|x_k\|_2^2 \tag{4}$$

where λ_{min} and λ_{max} are minimum and maximum eigenvalues of P. Consequently, the closed loop stability is guaranteed by ensuring that the candidate quadratic Lyapunov function is decreasing in each time step:

$$V(x_k) - V(x_{k+1}) = x_k^T P x_k - x_{k+1}^T P x_{k+1} > 0 \tag{5}$$

The stabilizing controller is considered in the form of ratio of two polynomials as follows:

$$u_i(x) = \frac{c_i(x_k)}{c_0(x_k)} \tag{6}$$

where $c_i(x_k)$ are polynomials in the state with lowest order one and highest order n_n, and $c_0(x_k)$ is a polynomial of lowest order zero and highest order n_d. All inputs share the same denominator polynomial $c_0(x_k)$. Note that for a given x_k, these polynomials are linear in the polynomial coefficients (c_{ci}), a fact that is important when optimizing over polynomial coefficients in the controller design.

Definition 2. *A system is defined to be globally quadratically stable if a quadratic Lyapunov function fulfilling Theorem 1 can be found for $D = \Re^n$.*

In the following two propositions, a bilinear system is termed *open loop unstable* if it is unstable with a zero input, *i.e.*, if the linear dynamics described by the matrix A is unstable.

Proposition 3. *Global quadratic stability of open loop unstable discrete-time bilinear systems requires the controller (6) to have a denominator polynomial of order at least as high as the numerator polynomial ($n_d \ge n_n$).*

Proof: If the order of numerator would be higher than denominator, then the second term in the Lyapunov difference inequality (5) will be of higher order than the first, and therefore will dominate for large norm of the state vector. ■

Proposition 4. *Global quadratic stability of open loop unstable discrete-time bilinear systems requires the controller (6) to have a numerator polynomial of order at least as high as the denominator polynomial ($n_n \ge n_d$).*

Proof: If the numerator order is lower than the denominator order, the norm of the input signal will approach zero far from the origin and as the result the controller could not satisfy (5). ■

From the propositions above, it is concluded that for open loop unstable bilinear discrete time systems the highest order of the controller numerator and denominator polynomials should be the same.

The controller is obliged to satisfy the control constraints of the form:

$$|u_i(x)| \le u_{i,max} \tag{7}$$

To design the controller (6) to satisfy (5) and (7), Sum of Squares (SOS) methods are exploited. The basic idea behind the SOS approach for checking the positivity of a polynomial $p(x)$, is to replace the positivity with the condition that the polynomial can be transformed to SOS terms [15]:

$$p(x) = \sum_{i=1}^{N} h_i^2(x) = \sum_{i=1}^{M}(q_i^T v(x))^2 = v^T(x)Qv(x) \tag{8}$$

where $Q = Q^T > 0$. As the result, if it is possible to find a vector of monomials $v(x)$ and a positive definite matrix Q, positivity of $p(x)$ is guaranteed.

Similarly, a symmetric polynomial matrix $M(x)$ is said to be an SOS matrix if it can be decomposed into

$$M(x) = H^T(x)H(x) \tag{9}$$

The SOS decomposition can be computed by semi-definite programming with the help of available software [14].

In the following, conditions will be presented that require some state-dependent matrix $M(x)$ to be positive definite,

denoted $M(x) > 0$ (corresponding to $z^T M(x) z > 0 \; \forall x, \forall z \neq 0$). In practice, in the resulting optimization problem formulations, such conditions are replaced with the slightly more restrictive condition that $M(x)$ should be an SOS matrix.

III. DIRECT OPTIMAL CONTROLLER DESIGN

In this section, a controller for the discrete time bilinear system will be designed, based on the direct minimization of a cost function over one timestep. The cost function accounts for the tracking error at the next timestep and cost of control inputs at the present time:

$$J(x,u) = x_{k+1}^T Q x_{k+1} + u_k^T R u_k \tag{10}$$

where Q is a positive semidefinite matrix and R is a positive definite matrix. Direct minimization of the cost function results in the controller:

$$u_k = -\left[(B_x+B)^T Q (B_x+B) + R\right]^{-1} (B_x+B)^T Q A x_k \tag{11}$$

While this controller is optimal according to the chosen cost function, it does not guarantee global stability, so the region of stability should be assessed by other means - which will be addressed in subsequent sections. The denominator term in the controller comes from $det((B_x+B)^T Q (B_x+B) + R)$ which is common for all inputs and is positive definite, so it is possible to write the optimal controller in the form of ratio of polynomials as in (6).

Remark 5. *It is straight forward to extend (11) to the case with non-zero references for states and inputs. However, inclusion of non-zero references may raise additional questions about the admissibility of the references with regards to the system dynamics. This issue is not pursued any further in the present paper.*

IV. MAXIMIZING REGION OF CONVERGENCE

This section addresses controller design, using controllers on the form (6), to maximize the region of convergence. The denominator polynomial $c_0(x)$ will be assumed to be an SOS polynomial. However, there exists a possibility of using excessively large inputs, if all terms $c_0(x)$ have roots near the same point in the state space. To guard against this situation, the denominator polynomial is specified as $c_0(x) = \acute{c}_0(x) + 1$, with $\acute{c}_0(x)$ an SOS polynomial, thus ensuring that the denominator polynomial cannot be very small anywhere in $\Re^n$. One may note that the controller in (11) also has a constant term - that can be set to unity by multiplying the numerator and denominator polynomials with a common scaling factor.

Theorem 6. *Region of convergence: Given a quadratic function $V(x) = x_k^T P x_k$, polynomials $c_i(x_k), i \in [1,\ldots,m]$, and SOS polynomials $\acute{c}_0(x_k)$ and $s_1(x_k)$, a bilinear discrete time system (1) in closed loop with the control law (6) is stable $\forall x_k | x_k^T P x_k < \gamma$, provided*

$$\begin{bmatrix} (\acute{c}_0(x_k)+1) x_k^T P x_k - s_1(x_k)(\gamma - x_k^T P x_k) & \left(\left(\acute{c}_0(x_k)+1\right) A x_k + (B_x+B) \begin{bmatrix} c_1(x_k) \\ \vdots \\ c_m(x_k) \end{bmatrix}\right)^T P \\ P \left(\left(\acute{c}_0(x_k)+1\right) A x_k + (B_x+B) \begin{bmatrix} c_1(x_k) \\ \vdots \\ c_m(x_k) \end{bmatrix}\right) & (\acute{c}_0(x_k)+1) P \end{bmatrix} > 0 \tag{12}$$

Proof: Dividing (12) with the strictly positive $(\acute{c}_0(x_k)+1)$:

$$\begin{bmatrix} x_k^T P x_k - \frac{s_1(x_k)}{(\acute{c}_0(x_k)+1)}(\gamma - x_k^T P x_k) & \left(A x_k + (B_x+B)\frac{1}{c_0(x_k)} \begin{bmatrix} c_1(x_k) \\ \vdots \\ c_m(x_k) \end{bmatrix}\right)^T P \\ P \left(A x_k + (B_x+B)\frac{1}{c_0(x_k)} \begin{bmatrix} c_1(x_k) \\ \vdots \\ c_m(x_k) \end{bmatrix}\right) & P \end{bmatrix} > 0 \tag{13}$$

Considering the controller in (6) and the bilinear system dynamics in (2), this corresponds to

$$\begin{bmatrix} x_k^T P x_k - \frac{s_1(x_k)}{(\acute{c}_0(x_k)+1)}(\gamma - x_k^T P x_k) & x_{k+1}^T P \\ P x_{k+1} & P \end{bmatrix} > 0 \tag{14}$$

and using the Schur complement one obtains

$$x_k^T P x_k - x_{k+1}^T P x_{k+1} - \frac{s_1(x_k)}{(\acute{c}_0(x_k)+1)}(\gamma - x_k^T P x_k) > 0 \tag{15}$$

Noting that $\frac{s_1(x_k)}{(\acute{c}_0(x_k)+1)} > 0$ and $(\gamma - x_k^T P x_k) > 0 \forall x_k^T P x_k < \gamma$, it follows from Theorem 1 that

$$x_k^T P x_k - x_{k+1}^T P x_{k+1} > 0 \; \forall \; x | x_k^T P x_k < \gamma. \qquad \blacksquare$$

Theorem 7. *Given the polynomial $c_i(x_k)$, SOS polynomials $\acute{c}_0(x_k)$ and $q_i(x_k)$, the input constraint in (7) is satisfied $\forall \; x_k | x_k^T P x_k < \gamma$ provided*

$$\begin{bmatrix} (\acute{c}_0(x_k)+1) u_{max,i}^2 - q_i(x_k)(\gamma - x_k^T P x_k) & c_i(x_k) \\ c_i(x_k) & \acute{c}_0(x_k)+1 \end{bmatrix} > 0 \tag{16}$$

Proof: Following the same approach as in the proof of Theorem 6, it can be shown that (16) is equivalent to

$$u_{max,i}^2 - u_i^2 - \frac{q_i(x_k)}{(\acute{c}_0(x_k)+1)}(\gamma - x_k^T P x_k) > 0,$$

and hence $u_{max,i}^2 - u_i^2 > 0 \; \forall \; x_k \in \{x_k | x_k^T P x_k < \gamma\}$. $\blacksquare$

Observe that the parameters in the polynomials $\acute{c}_0(x_k)$, $c_i(x_k)$, $s_1(x_k)$, and $q_i(x_k)$ enter linearly in (12) and (16). One may therefore maximize the region of convergence by iteratively increasing γ while verifying (12) and (16), with the polynomial coefficients as free variables. Such problems can easily be formulated and solved using YALMIP (with an appropriate semidefinite programming solver). If γ is treated

as a variable, (12) and (16) contain bilinear terms in γ and the parameters of $s_1(x_k)$ and $q_i(x_k)$, respectively. Thus, if one has access to a solver handling bilinear terms, one may formulate an optimization problem to directly maximize γ.

V. IMPROVING RATE OF CONVERGENCE

Maximizing the region of convergence may lead to rather slow control, i.e., the rate of convergence may be poor, in particular near the boundary of the region in question. To improve the rate of convergence, the decrease in Lyapunov function in each step could be increased. This can be formulated as:

$$\max_{\alpha, c_{c0}, c_{ci}} \alpha \tag{17}$$
$$\text{such that}$$
$$x_k^T P x_k - x_{k+1}^T P x_{k+1} \quad - \frac{s_1(x_k)}{(\acute{c_0}(x_k)+1)}(\gamma - x_k^T P x_k) > \alpha x_k^T P x_k$$

where again the region to be considered is bounded by $(\gamma - x_k^T P x_k) > 0$.

This would change (12) to the constraint[1]:

$$\begin{bmatrix} (1-\alpha)(\acute{c_0}(x_k)+1)x_k^T P x_k - s_1(x_k)(\gamma - x_k^T P x_k) & P\left((\acute{c_0}(x_k)+1)Ax_k + (B_x+B)\begin{bmatrix} c_1(x_k) \\ \vdots \\ c_m(x_k)\end{bmatrix}\right) \\ \left((\acute{c_0}(x_k)+1)Ax_k + (B_x+B)\begin{bmatrix} c_1(x_k) \\ \vdots \\ c_m(x_k)\end{bmatrix}\right)^T P & (\acute{c_0}(x_k)+1)P \end{bmatrix} > 0 \tag{18}$$

There are bilinear terms between α and the parameters of $\acute{c_0}(x_k)$, so the problem has to be solved iteratively, for a fixed γ iteratively increasing α (within the range $0 < \alpha < 1$). Clearly, one must expect to have to decrease γ from the maximal value found in Section IV, in order to be able to increase the rate of convergence.

VI. NUMERICAL EXAMPLES

Example 1: In the following, a second-order bilinear system, proposed initially in [7], is considered:

$$A = \begin{bmatrix} 1 & 0.01 \\ 0.01 & 1 \end{bmatrix}, B1 = \begin{bmatrix} 0.001 & 0 \\ 0 & -0.004 \end{bmatrix}, b1 = \begin{bmatrix} 0.09 \\ 0.09 \end{bmatrix}$$

The input is constrained to $|u| \leq 2$. The problem to be solved is the determination of the controller which stabilizes the system in the maximum possible region of $x_k^T P x_k < \gamma$. P is considered as identity matrix.

YALMIP [14] is used to solve the SOS problems in MATLAB environment. First, the region of convergence is maximized according to Section IV. The highest order input polynomials considered is $n_p = 2$. The maximum region where YALMIP could find a controller to stabilize the system is given by $\gamma = 150$. The designed controller is as follows:

$$u(x_k) = \frac{-256.1x_1 - 253.9x_2 + 1.8x_1^2 + 1.3x_1x_2 - 9.5x_2^2}{498.3 + 0.6x_1 + 6.3x_2 + 39.1x_1^2 - 28.8x_1x_2 + 34.4x_2^2}$$

[1] $\acute{c_0}(x_k)$ and $s_1(x_k)$ are still assumed to be SOS polynomials.

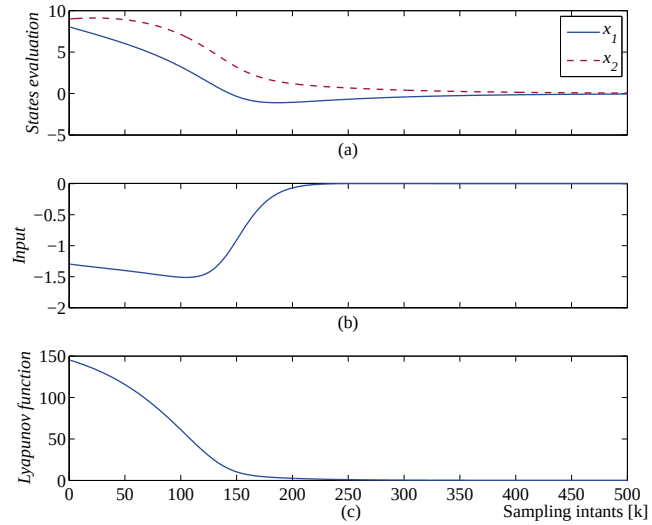

Fig. 1. Simulation results for example 1 system controlled by SOS method: (a) states, (b) input, and (c) cost function

The state evolution in time, input and cost function for designed controller are shown in Fig. 1 for the initial state of $x_0 = [8, 9]^T$. Note that, although (12) cannot be verified for $\gamma > 150$, this does not mean that the system is necessarily unstable in that region.

In Fig. 2, phase portraits of the closed loop system for initial states belonging to the $x_1^2 + x_2^2 = 150$ is depicted.

Remark 8. *The problem formulation in [7] includes the state constraints $|x_i| \leq 4$, $i \in \{1, 2\}$, which makes the objective of the controller design different from the one in the present paper. Nevertheless, Fig. 2 shows that the controller presented here practically makes the set $\{x| \, |x_1| \leq 4, \, |x_2| \leq 4\}$ positively invariant, and thus that the state constraints are fulfilled for any initial condition within this set.*

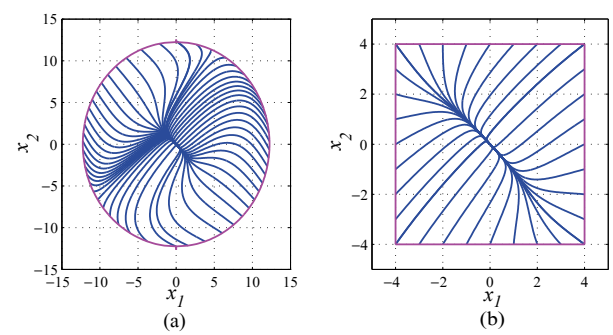

Fig. 2. State trajectories for example 1: (a) starting from the border of the maximum region of stability (b) starting from $|x_i| \leq 4$ as in [7]

To improve the rate of convergence, (18) is solved in YALMIP by specifying $\alpha = 0.015$. Note that by adding α to the problem, the maximum region of convergence will decrease. In this example, it decreases to $\gamma = 120$. The designed controller is as follows:

$$u(x_k) = \frac{-137.4x_1 - 148.9x_2 + 0.6x_1^2 - 0.6x_1x_2 - 6.35x_2^2}{429.2 + 5.1x_1 + 11.2x_2 + 7.5x_1^2 + 1.4x_1x_2 + 6.3x_2^2}$$

The responses of the system for both controllers designed

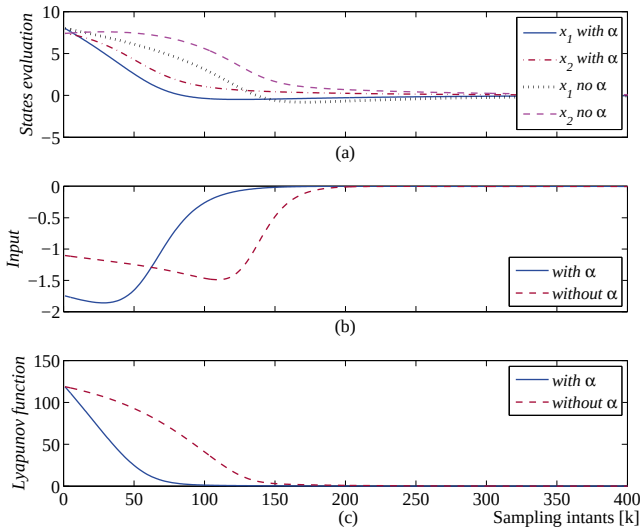

Fig. 3. Improvement of the response in example 1 by specifying $\alpha = 0.015$, ($\gamma = 120$): (a) states, (b) input, and (c) cost function

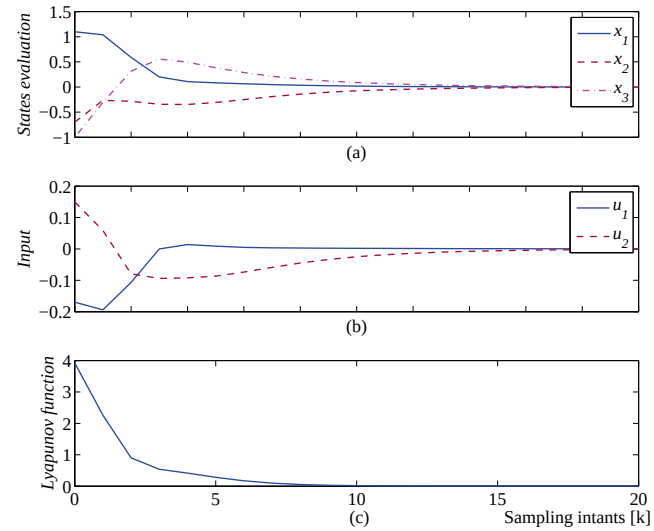

Fig. 4. Simulation results for example 2 system controlled by SOS method: (a) states, (b) input, and (c) cost function

by (12) and (18) are shown in fig.3, which shows that by adding the term α, the rate of convergence is increased.

Example 2: Consider the third-order bilinear system with two inputs found in [7]:

$$A = \begin{bmatrix} 1.10 & -0.2 & -0.34 \\ -0.06 & 0.7 & -0.42 \\ 0.41 & 0.41 & 0.90 \end{bmatrix}, \quad b1 = \begin{bmatrix} 3.75 \\ 1.05 \\ -0.85 \end{bmatrix}$$
$$b2 = \begin{bmatrix} 0 \\ -1.33 \\ -0.49 \end{bmatrix}, \quad B1 = \begin{bmatrix} -0.12 & -0.22 & 0.36 \\ -0.32 & 0.48 & 0.36 \\ -0.35 & 0.36 & -0.18 \end{bmatrix} \quad (19)$$
$$B2 = \begin{bmatrix} -0.18 & 0.30 & 0.07 \\ -0.03 & -0.18 & -0.38 \\ 0.55 & -0.74 & -0.77 \end{bmatrix}$$

Both control inputs have to respect the linear constraints $-1 \leq u \leq 1$. The matrix P in the cost function is chosen as:

$$P = \begin{bmatrix} 2 & 0.1 & 0.1 \\ 0.1 & 1.5 & 0.1 \\ 0.1 & 0.1 & 1 \end{bmatrix} \quad (20)$$

Using SOS programming, a region of stability parametrized by $\gamma = 4$ results. The designed controller based on (6) is as follows:

$$\begin{aligned} c_1(x_k) =& -7.1x_1 + 0.8x_2 + 2.7x_3 + 0.4x_1^2 + 0.6x_1x_2 - 0.8x_2^2 \\ & - 0.6x_1x_3 - 0.2x_2x_3 + 1.1x_3^2 \\ c_2(x_k) =& -0.9x_1 + 6.7x_2 - 1.7x_3 + 2.4x_1^2 - 0.3x_1x_2 + 0.8x_2^2 \\ & - 4.3x_1x_3 + 1.9x_2x_3 + 3.1x_3^2 \\ c_0(x_k) =& 26.3 + 0.1x_1 - 0.6x_2 + 1.6x_3 + 11.9x_1^2 - 0.2x_1x_2 \\ & + 13.2x_2^2 + 0.3x_1x_3 + 0.9x_2x_3 + 11.6x_3^2 \end{aligned}$$

The state responses for the calculated controller for the initial state $x_0 = [1.1, -0.7, -1]^T$ is depicted in Fig. 4 along with input and cost function. The region of convergence ($x_k^T P x_k < \gamma$) calculated for this example is shown in Fig. 5.

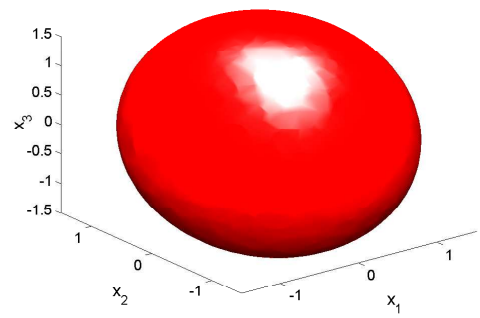

Fig. 5. Region of convergence calculated for example 2 by SOS method

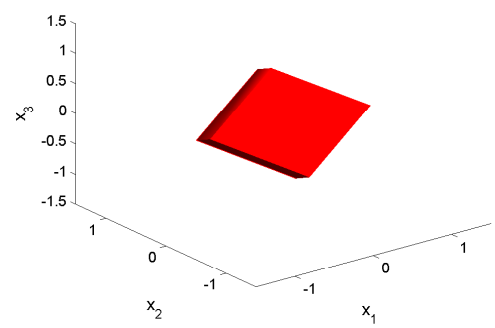

Fig. 6. Region of convergence calculated for example 2 using polyhedral Lyapunov function in [7]

This problem is also solved in [7] using polyhedral Lyapunov functions and the calculated region of convergence is depicted in Fig. 6. Comparing Fig. 5 and Fig. 6, it is clear that the region of convergence calculated using SOS methods is much larger than the one calculated in [7].

Example 3: Consider the following second order bilinear system: [6]

$$A = \begin{bmatrix} 0.8 & 0.5 \\ 0.4 & 1.2 \end{bmatrix}, B1 = \begin{bmatrix} 0.45 & 0.45 \\ 0.3 & -0.3 \end{bmatrix}, b1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \quad (21)$$

The input is constrained to $|u| \leq 0.5$. The problem to be solved is the determination of the controller which stabilizes

the system in the maximum possible region of $x_k^T P x_k < \gamma$. The matrix P is chosen as

$$P = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \tag{22}$$

Solving the problem in YALMIP for maximum γ results in $\gamma = 6$. The designed controller is as follows:

$$u(x_k) = \frac{-6.0x_1 - 9.7x_2 - 0.2x_1^2 - 0.9x_1x_2 - 0.3x_2^2}{27.9 + 5.1x_1 - 0.48x_2 + 2.1x_1^2 + 2.3x_1x_2 + 4.0x_2^2}$$

State responses, input and cost function evolution in time is depicted in Fig. 7. In addition, the calculated region of convergence for SOS method is shown in Fig. 8. This problem is also solved in [6] using polyhedral Lyapunov functions and calculated region of convergence is also shown the same figure. It is clearly shown that region of convergence using SOS method is much larger than the region of convergence calculated in [6] using polyhedral Lyapunov functions.

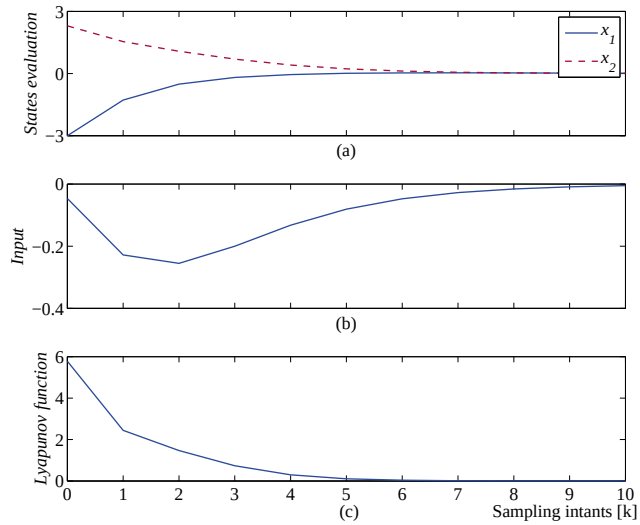

Fig. 7. Simulation results for example 3 system controlled by SOS method: (a) states, (b) input, and (c) cost function

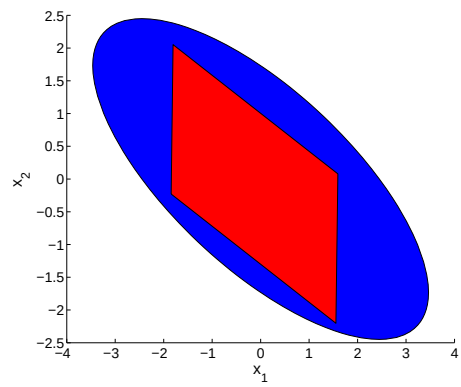

Fig. 8. Region of convergence calculated for example 3 using polyhedral Lyapunov function in [6] in red and using SOS method in blue

VII. Conclusions and Future works

In this paper SOS programming methods are used to investigate controller design for discrete time bilinear systems. First, an optimization problem is proposed to design a controller to maximize the region of convergence. The input is defined in the form of ratio of two polynomials and quadratic Lyapunov function is considered. Then the rate of convergence is improved by increasing the rate of decrease in the Lyapunov function. Finally the direct optimal controller is considered and maximum rate of convergence is calculated for this controller. Numerical examples from literature are provided and solved using YALMIP software.

In this project, it was assumed that matrix P is given. The next step would be to consider that input (c_0, c_i) and stability region (s_1, γ) are available and the problem is to find a new P which maximizes the volume of the ellipsoid $x_k^T P x_k < \gamma$ where the system is stable. After finding new P, the optimization process could be repeated to find new c_0, c_i and γ and this process could iterated to find larger ellipsoids which guarantee stability.

References

[1] V. Spinu, N. Athanasopoulos, M. Lazar, and G. Bitsoris, "Stabilization of Bilinear Power Converters by Affine State Feedback Under Input and State Constraints," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 59, no. 8, pp. 520–524, Aug. 2012.

[2] P.-O. Gutman, "Stabilizing controllers for bilinear systems," *IEEE Transactions on Automatic Control*, vol. 26, no. 4, pp. 917–922, Aug. 1981.

[3] A. B. Fontes, C. E. Dorea, and M. R. da S. Garcia, "An iterative algorithm for constrained MPC with stability of bilinear systems," in *2008 16th Mediterranean Conference on Control and Automation*. IEEE, June 2008, pp. 1526–1531.

[4] H. Bloemen, M. Cannon, and B. Kouvaritakis, "Interpolation in MPC for discrete time bilinear systems," in *Proceedings of the 2001 American Control Conference*, vol. 5, 2001, pp. 3061–3066.

[5] M.-S. Chen, Y.-R. Hwang, and K.-C. Huang, "Nonlinear controls for a class of discrete-time bilinear systems," *International Journal of Robust and Nonlinear Control*, vol. 13, no. 11, pp. 1079–1090, Sept. 2003.

[6] G. Bitsoris and N. Athanasopoulos, "Constrained Stabilization of Bilinear Discrete-Time Systems Using Polyhedral Lyapunov Functions," in *17th World Congress of the International Federation of Automatic Control*, vol. 17, no. 1, July 2008, pp. 2502–2507.

[7] N. Athanasopoulos and G. Bitsoris, "Unconstrained and constrained stabilisation of bilinear discrete-time systems using polyhedral Lyapunov functions," *International Journal of Control*, vol. 83, no. 12, pp. 2483–2493, Dec. 2010.

[8] N. Athanasopoulos, G. Bitsoris, and M. Vassilaki, "Ultimate boundedness and robust stabilization of bilinear discrete-time systems," in *IEEE Conference on Decision and Control and European Control Conference*, Dec. 2011, pp. 4622–4627.

[9] M. Bacic, M. Cannon, and B. Kouvaritakis, "Constrained control of SISO bilinear systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 8, pp. 1443–1447, Aug. 2003.

[10] E. J. Hancock and A. Papachristodoulou, "Generalised absolute stability and sum of squares," *Automatica*, vol. 49, no. 4, pp. 960–967, 2013.

[11] D. Kang, S. Won, and Y. J. Jang, "Guaranteed cost control for bilinear systems by static output feedback," *Applied Mathematics and Computation*, vol. 219, no. 14, pp. 7398–7405, 2013.

[12] E. Nobuyama, T. Aoyagi, and Y. Kami, "A Sum of Squares Optimization Approach to Robust Control of Bilinear Systems, ser. Recent Advances in Robust Control - Theory and Applications in Robotics and Electromechanics." InTech, Nov. 2011, pp. 39–54.

[13] J. Löfberg, "YALMIP : A Toolbox for Modeling and Optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taiwan, 2004.

[14] ——, "Pre- and Post-Processing Sum-of-Squares Programs in Practice," *IEEE Transactions on Automatic Control*, vol. 54, no. 5, pp. 1007–1011, May 2009.

[15] P. A. Parrilo, "Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization," Ph.D. dissertation, California Institute of Technology, 2000.