CHAOTIC SYNCHRONIZATION USING BACKSTEPPING METHOD WITH APPLICATION TO THE CHUA'S CIRCUIT AND LORENZ SYSTEM

TAO YANG, XIAO FENG LI, HUI HE SHAO
Department of Automation, Shanghai Jiao Tong University, Shanghai 200030, China

In this paper, backstepping method is proposed for synchronizing chaotic systems. The tool consists in a recursive procedure that combines the choice of a Lyapunov function with the design of feedback control. It offers a systematic design procedure for the synchronization of a large class of continuous-time hyper-chaotic systems, which implies that much complicated high-order chaotic systems can be used to improve the security in chaos communications. In order to illustrate the general applicability of backstepping design, the tool is used to synchronize the dynamics of Chua's circuits and Lorenz systems. Numerical simulations are presented to show the effectiveness of this approach.

1. Introduction

In past decade, synchronization of chaotic systems has become an area of active research, especially in the light of its potential application in secure communications, modeling brain activity, (see[1] and references therein). Recently, nonlinear control theory was widely used in the study of chaos synchronization and its application in communications. Backstepping design (12) and references therein), one of the most popular methods for nonlinear control, can guarantee global stabilities, tracking and transient performances for a broad class of strict-feedback systems. It has been shown that many well-known chaotic systems, like the Duffing oscillator, van der Pol oscillator, Rössler systems, and so on, can be transformed into a class of nonlinear system in the so-called strict-feedback form, and backstepping design has been used to control these chaotic systems to a steady as well as tracking of any desired trajectory.

In this paper, we propose an approach for the synchronization of chaotic systems based on the backstepping design. Compared with the observer based synchronization schemes (5), one drawback of this method is that several states of the master and response system should be used to form the control, which make the control more complicated to some extent. But this approach can be used to synchronize a wide class of nonlinear systems, including most of the continuous-time chaotic and hyper-chaotic systems, and many modern control theory, such as the State Observe theory, and so on, may be employed to compensate this drawback.

2. Backstepping Design

Backstepping design consists in a recursive procedure that interlaces the choice of a Lyapunov function with the design of feedback control. The key idea is to utilize the Lyapunov's method by breaking the design problem into a sequence of design problem for lower-order (even scalar) system8. Namely, a nonlinear system is described by $\dot{x} = f(x) + g(x)u$, where $x \in R^n$ is the state variable, $u \in R$ is the scalar control input and $f, g$ are nonlinear functions. According to the backstepping design, let the nonlinear system be augmented by the so-called strict-feedback form:

$$\dot{\xi} = f_1(x, \xi, \xi_2, \ldots, \xi_m) + g_1(x, \xi, \xi_2, \ldots, \xi_m)u$$

where $\xi, \xi_2, \ldots, \xi_m$ are scalars and the nonlinear functions $f, g$, depending only on $x, \xi_1, \ldots, \xi_m$. Many chaotic systems, such as Rössler system, Chua's circuit, Lorenz system and so on, can be described in this form.

According to (1), consider $\xi_1$ as a control input to stabilize the first equation, when $\xi_1$ has been designed, let $\xi_2$ be the virtual control to control the second equation of (1), ..., and so on. Therefore the actual control input $u$ usually depending on $x, \xi, \ldots, \xi_m$ can be systematically designed in $m$ steps. It's worth noting that backstepping is suitable for chaotic synchronization because chaotic synchronization is the same as a tracking problem, which control the slave system to track the master system. So the control force $u$, used to synchronize the master-slave system, can be achieved systematically based on the backstepping idea.

3. Synchronization of Chua's Circuits

3.1 Steps of synchronization

In order to show how backstepping design works, in this section the approach is used to synchronize two Chua's systems, which are described by the following two sets of dynamic equations (8):

$$\dot{x} = a\left(y - x - cx\right), \dot{y} = x - y + z, \dot{z} = -by$$

$$\dot{x}^* = a\left(y^* - x^* - cx^*\right) + u, \dot{y}^* = x^* - y^* + z^*, \dot{z}^* = -by^*$$

where $a, b, c$ are the system parameters. The object is to design the control $u$, so that $\lim_{t \to \infty} [x - x^*] \to 0$ is fulfilled. According to equations (2) and (3), we get the error dynamics as follows:

$$\dot{e}_1 = -be_2, \dot{e}_2 = e_1 - e_1 + e_1 = ae_3 - ace_1 - ax^* + ax^* - u$$

where $e_1 = x - x^*, e_2 = y - y^*, e_3 = x - x$, the synchronizing error. Then the aim of synchronization is to design $u$ so that system (4) is stabilized to the origin. According to backstepping design, the following steps must be done:

Step1. Choose the Lyapunov function as $V_1 = \frac{1}{2}e_1^2$, a stabilizing function $a_1(e_1)$ must be designed to make the derivative of $V_1$:

$$\dot{V}_1 = -be_2e_2 < 0$$

So we choose $a_1(e_1) = e_1$ and get $\dot{V}_1 = -be_2e_2 < 0$.

By define the error variable $f_1$ as $f_1 = e_1 - e_2$.

(5)

We get the following $(e_1, f_1)$-subsystem:

$$\dot{\xi}_1 = -b(f_2 + e_1), \dot{\xi}_2 = e_1 + (b - 1)f_1 + b\xi_1$$

(6)

Step2. We choose a candidate Lyapunov function for (6) as $V_2 = \frac{1}{2}Sf_1^2$. When we choose the virtual control $e_1$ as follows:

$$e_1 = a_2(e_1, f_2) = -bf_1$$

the derivative of $V_2$ becomes negative, that is:

$$\dot{V}_2 = -be_2 - f_1 + f_2 + b\xi_1 < 0$$

in the same way, we define error variable $f_2$ as $f_2 = e_2 + b\xi_2$.

So we get the $(e_1, f_1, f_2)$-subsystem:

$$\dot{\xi}_1 = -b(f_2 + e_1), \dot{\xi}_2 = e_1 + (b - 1)f_1 + b\xi_1, \dot{\xi}_3 = e_2 + b\xi_2$$

(7)
Step3. By iterating the previous steps, we have the third Lyapunov function as follows: \( V_i = V_i + 0.5x_i^2 \). Combine with equation (5), (7), when choosing the control \( u \) as:
\[
u = A(x^2 - x^*) + (1 + a + acb + b + b')(y - y^*) + (b - 1 - bcb)(y - y^*) + b(x - x^*) \]

(9)

We have \( V_i < 0 \), which prove that system (8) is has been stabilized to the origin. In the view of (3) and (7), the origin in the \( (\varepsilon_i, \varepsilon_j, \varepsilon_k) \) coordinates has the same properties, which means the synchronization of (2) and (3) is fulfilled.

3.2 simulation results

Numerical simulations are carried out to show the effectiveness of the previous approach. In the simulation process, we have the system parameters as: \( a = 10.0, b = 16, c = 0.143 \) and the initial condition: \( X(0) = (2.31, 2.31, 1) \) and \( Y(0) = (0.13, 2.1, 1) \). Under the control of (9), we get the perfect synchronization. The results are on show as the follows.

\[ \begin{align*}
(a) & \quad \text{Tracking error: (a) } z - z^*; \quad (b) \quad y - y^*; \quad (c) \quad x - x^*
\end{align*} \]

4. Synchronization of Lorenz systems

In order to show the general applicability of backstepping design in chaotic synchronization, the attention is now focused on two Lorenz systems:
\[
\begin{align*}
X: & \quad \dot{x} = \sigma(y - x), y = rx - xz - y, z = xy - bz \\
X^*: & \quad \dot{x}^* = \sigma(y^* - x^*), y^* = r^*x^* - x^*z^* - y^* + u, z = x^*y^* - b^*z^* + u^*
\end{align*}
\]

(10)

The objective is to design two scalar controls \( u_1, u_2 \) so that \( \lim_{t \to \infty} X = X^* \).

As previous, we define the error dynamics E as follows:
\[
\begin{align*}
\varepsilon_t = -\varepsilon_1 + \varepsilon_2, \quad \varepsilon_j = -\varepsilon_2 + \varepsilon_3 - xz + x^*z^* - u \\
\varepsilon_k = -\varepsilon_3 + \varepsilon_1 - y^* + u^* - u_1
\end{align*}
\]

(12)

where \( E = (\varepsilon_1, \varepsilon_2, \varepsilon_3) = (z - z^*, y - y^*, x - x^*) \). Now we have to choose proper scalar control input \( u_1, u_2 \) to stabilize (12) to (0,0,0), this is a classical control problem and we use backstepping design to solve it.

First, choose the Lyapunov function as: \( V_i(\varepsilon_i) = \varepsilon_i^2/2 \). According to the idea of backstepping design, a stabilizing function \( \alpha(\varepsilon_i) \) has to be designed to make the derivative of \( V_i \) become negative, that is, to have \( -\varepsilon_1^2 + \varepsilon_2^2 < 0 \). Choose \( \varepsilon_1 = \alpha(\varepsilon_i) = 0 \), the goal can be fulfilled, and define variable:
\[
\begin{align*}
\varepsilon_t = & \quad \varepsilon_1 - \varepsilon_1
\end{align*}
\]

(13)

We get the following \( (\varepsilon_t, \varepsilon_j) \)-subsystem:
\[
\begin{align*}
\varepsilon_t = -\varepsilon_1 + \varepsilon_2, \quad \varepsilon_j = -\varepsilon_2 + \varepsilon_3 - xz + x^*z^* - u
\end{align*}
\]

(14)

for which a candidate Lyapunov function is \( V_i(\varepsilon_t, \varepsilon_j) = \varepsilon_t^2/2 \), and its derivative is:\n\[
\begin{align*}
\dot{V}_i = -\varepsilon_t^2 + \varepsilon_j^2 [\sigma(r + r) - xz + x^*z^* - u].
\end{align*}
\]

(15)

So when we choose \( \varepsilon_t = \alpha(\varepsilon_t, \varepsilon_j) \), that is:
\[
\begin{align*}
u = (\sigma + r)\varepsilon_t - xz + x^*z^*
\end{align*}
\]

We get \( V_i < 0 \). Now define another variable: \( \varepsilon_3 = \varepsilon_1 - \varepsilon_2, \varepsilon_j = \varepsilon_2 - \varepsilon_3 \), and we get the following subsystem:
\[
\begin{align*}
\varepsilon_t = -\varepsilon_1 + \varepsilon_2, \quad \varepsilon_j = \varepsilon_2 - \varepsilon_3 - xz + x^*z^* - u
\end{align*}
\]

(17)

As before, choose Lyapunov function as form of \( V_i = \varepsilon_t^2/2 \).

Its derivative is \( \varepsilon_t = \varepsilon_t^2/2, \) and when choose the control \( u_2 \) has the form: \( u_2 = xy - x^*y^* \).

(18)

We have \( V_i < 0 \), which means that system (17) has been controlled to the origin. According to (13) and (16), we know that synchronization occurs.

In the numerical simulation, we choose the system parameters as follows: \( \sigma = 16.0, b = b^* = 45.92, r = r^* = 4.0 \) and with the following initial condition: \( X(0) = (0.231, 3.1, 0.0) \), \( Y(0) = (2.1, 0.52) \), under the control of (15) and (18), we get:
\[
u = -xz + x^*z^* + 61.92(x - x^*), u_2 = xy - x^*y^*
\]

(19)

The tracking error of (10) and (11) in Fig.2, shows the perfect synchronization, under the control of the form of (19), is completed.

\[ \begin{align*}
(a) & \quad \text{Tracking error: (a) } x - x^*; \quad (b) \quad y - y^*; \quad (c) \quad z - z^*
\end{align*} \]

5. Conclusion

In this letter a Lyapunov-based method, called backstepping design, has been proposed for chaotic synchronization. It is a systematic procedure for synchronizing chaotic systems, with its successfully application to the Chua's circuit as well as Lorenz systems. But it has drawback of its own. The control form we get through backstepping design may be complicated and it may be hard to fulfill in engineer practice. The development of State Observer and many other modern control ideas may be used to compensate this problem to some extent. All in all, the backstepping design, which is very simple to execute, may has a promising future in the field of chaotic synchronization, and this topic is worthy of our further efforts.

References


2300