PERGAMON Mathematical and Computer Modelling 35 (2002) 145–163

MATHEMATICAL AND COMPUTER MODELLING

www.elsevier.com/locate/mcm

Hybrid Chaos Synchronization and Its Application in Information Processing

QINGXIAN XIE AND GUANRONG CHEN
Department of Electrical and Computer Engineering
University of Houston, Houston, TX 77204-4793, U.S.A.

E. M. BOLLT
Department of Mathematics, U.S. Naval Academy
Annapolis, MD 21402-5002, U.S.A.

(Received October 1999; revised and accepted December 2000)

Abstract—In this paper, through numerical studies, we explore a new methodology for chaos synchronization via a hybrid (generalized plus identical) synchronization. An arbitrary signal, generated by an unknown dynamical system, can be synchronized by the hybrid chaotic system. The signal can then be stored for future application such as password and message identification. Each finite-length signal can, in principle, be labelled and stored by a unique number, provided that the key hybrid system parameter used for the purpose is suitably chosen within a one-to-one mapping range. The new methodology enables us to encode an arbitrary signal accurately and efficiently. Sufficient numerical simulations are shown to verify the proposed design. Potential applications of the developed hybrid chaos synchronization system include information storage, message identification, and certain types of secure signal and image communication.

Keywords—Chaos synchronization, Dynamical system, Information processing, Generalized synchronization.

1. INTRODUCTION

There are many ways to encode and represent a finite-length signal or a piece of information about the output of an unknown dynamical system. Within the context of chaos synchronization for information processing, it is worth mentioning that Pecora and Carroll [1,2] proposed a chaos synchronization scheme based on two identical chaotic systems, called identical synchronization (IS). Rulkov *et al.* [3] put forward the concept of generalized synchronization (GS) for two different chaotic systems. Ever since then, there have been a large number of publications devoted to this research topic and its potential applications in information processing such as secure communication ([4–10] and references therein).

Since IS can considered as a special case of GS, we first review the concept of GS. Consider two unidirectionally coupled systems

$$\dot{x} = r(x), \qquad \dot{y} = g(y, s) = g(y, h(x)), \tag{1.1}$$

where $x \in R^n$, $y \in R^m$, $s \in R^k$, $h \in R^k$, and $s(t) = (s_1(t), \ldots, s_k(t))$ with $s_j(t) = h_j(x(t, x_0))$ being the coupling signal. Systems in the form of (1.1) are said to possess the characteristic of GS

E.B. is supported by the National Science Foundation under Grant DMS-9704639. G.C. is supported by the U.S. Army Research Office under Grant DAAG55-1-98-0198.

0895-7177/01/$ - see front matter Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-TEX
PII: S0895-7177(01)00157-1

between x and y if for a continuous (and nontrivial) transformation, $H : R^n \to R^m$, there are a manifold $M = \{(x, y) : y = H(x)\}$ and a subset $B = B_x \times B_y \subset R^n \times R^m$ with $M \subset B$, such that all trajectories of the coupled systems, with initial conditions in B, approach the manifold M as time goes to infinity. In particular, if $H = I$ (I is the identify transformation), then this GS reduces to the IS.

2. THE HYBRID CHAOS SYNCHRONIZATION SYSTEM

It has been observed that IS and GS for chaos synchronization are studied separately, and they are not put together in both theoretic and practical studies. In this paper, we integrate them in an appropriate way, with the attempt of enhancing the information processing capability of the existing chaos synchronization based techniques. More specifically, we explore a simple hybrid chaos synchronization system (HCSS) which can be used to encode and represent various kinds onf finite-length signals.

The term HCSS used in this paper means that both IS and GS are used in a system. The main idea of using GS in HCSS is the primary advantage of robustness of GS, which can make HCSS synchronize with an arbitrary input signal (under certain amplitude). The idea of using IS in HCSS is due to the exact synchronization of IS which can estimate the parameters of HCSS.

2.1. Structure of the Proposed HCSS

The block diagram of the proposed HCSS is shown in Figure 1.

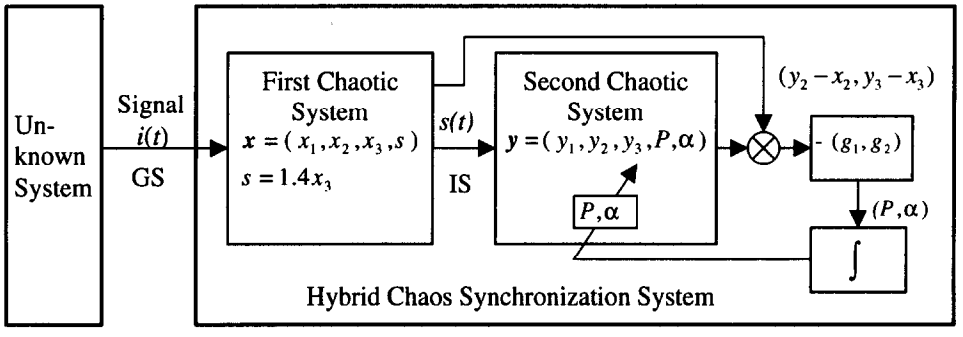


Figure 1. Block diagram of the proposed hybrid chaos synchronization system.

In this study, the Rossler chaotic oscillator is used in the HCSS. The equations of this HCSS are as follows:

$$\left.\begin{aligned} \dot{x}_1 &= 2 + x_1(x_2 - 4) \\ \dot{x}_2 &= -x_1 - x_3 \\ \dot{x}_3 &= x_2 - x_3 + s + i \end{aligned}\right\} \quad \text{first chaotic system,}$$

$$\left.\begin{aligned} \dot{y}_1 &= 2 + y_1(y_2 - P) \\ \dot{y}_2 &= -y_1 - y_3 \\ \dot{y}_3 &= y_2 - y_3 + s + \alpha \end{aligned}\right\} \quad \text{second chaotic system,} \tag{2.1}$$

$$\left.\begin{aligned} \dot{\alpha} &= -g_1(y_3 - x_3) \\ \dot{P} &= -g_2(y_2 - x_2) \end{aligned}\right\} \quad \text{adaptive control,}$$

where $x = (x_1, x_2, x_3)$ are the variables of the first chaotic system; $y = (y_1, y_2, y_3)$ are the variables of the second chaotic system; $i(t)$ is the input signal (to be encoded) from the unknown dynamical system; $s(t)$ is the drive variable from the first chaotic system (i.e., the coupling variable between the two chaotic systems); $\alpha(t)$ is the parameter function used to trace the input signal $i(t)$; $P(t)$ is the parametric function to be estimated; g_1 and g_2 are the control functions (which will be

discussed in detail later). In this study, we chose $s(t) = 1.4x_3$, where the magnitude is chosen due to the coefficient 0.4 of the term x_3 is the Rossler system. This coupling approach is also called active-passive-decomposition (APD) in [11]. When $i(t), P(t)$, and $\alpha(t)$ are considered as state variables of the augmented system (2.1), the whole system is autonomous (of higher dimensionality). Substituting $s(t)$ in (2.1) with $s(t) = 1.4x_3$, we obtain

$$\left.\begin{aligned} \dot{x}_1 &= 2 + x_1(x_2 - 4) \\ \dot{x}_2 &= -x_1 - x_3 \\ \dot{x}_3 &= x_2 + 0.4x_3 + i \end{aligned}\right\} \quad \text{first chaotic system,}$$

$$\left.\begin{aligned} \dot{y}_1 &= 2 + y_1(y_2 - P) \\ \dot{y}_2 &= -y_1 - y_3 \\ \dot{y}_3 &= y_2 - y_3 + 1.4x_3 + \alpha \end{aligned}\right\} \quad \text{second chaotic system,} \tag{2.2}$$

$$\left.\begin{aligned} \dot{\alpha} &= -g_1(y_3 - x_3) \\ \dot{P} &= -g_2(y_2 - x_2) \end{aligned}\right\} \quad \text{adaptive control.}$$

The HCSS shown in Figure 1 works as follows. The signal to be encoded is generated by the unknown system and is input to the first chaotic system. GS will occur between the unknown system and the first chaotic system [12]. GS is used here because the underlying system that generated the signal is unknown. Then, IS occurs between the first and the second chaotic systems. IS is used here because parameter identification is needed to achieve. When exact synchronization occurs, the parameter value of P of the second chaotic system can be determined (estimated) via an adaptive parameter control algorithm [13–15]. Due to the GS between the unknown system and the first chaotic system, the attractor of the first chaotic system is a nonlinear image of the input signal. This nonlinear transformation should be carefully designed so that the resulting parameter is a single valued function of the input signal in the monotone region of the parameter P. After the parameter of the nonlinear image has been estimated, this parameter effectively "encodes" (represents) the input signal uniquely [6]. Details will be provided throughout the paper.

2.2. Stability Analysis of the HCSS

Stability is the first concern before this HCSS is put in use. To analyze the stability of the HCSS, we define the error variables as

$$e_1 = y_1 - x_1, \qquad e_2 = y_2 - x_2, \qquad e_3 = y_3 - x_3, \qquad e_4 = \alpha - i, \qquad e_5 = P - 4. \tag{2.3}$$

The synchronization error system between the first and second chaotic systems is obtained as

$$\begin{aligned} \dot{e}_1 &= y_1y_2 - x_1x_2 - Py_1 + 4x_1, & \dot{e}_2 &= -e_1 - e_3, \\ \dot{e}_3 &= e_2 - e_3 + (\alpha - i), & \dot{e}_4 &= (\alpha - i)' = -g_1(e_3), \\ \dot{e}_5 &= (P - 4)' = -g_2(e_2). \end{aligned} \tag{2.4}$$

From the fifth equation of (2.4), we know that we should choose the control function g_2 such that an equilibrium state, $P - 4 = C$, can be achieved; that is, $P = 4 + C$, where C is a constant. To find the equilibrium state of (2.4), we rewrite its first equation as follows and equate it to zero,

$$\begin{aligned} \dot{e}_1 &= y_1y_2 - x_1x_2 - Py_1 + 4x_1 = -x_1x_2 - y_1x_2 + y_1x_2 + y_1y_2 + 4x_1 - (4 + C)y_1 \\ &= (y_1 - x_1)x_2 + y_1(y_2 - x_2) - 4(y_1 - x_1) - Cy_1 = e_1x_2 + e_2y_1 - 4e_1 - Cy_1 \\ &= (x_2 - 4)e_1 + (e_2 - C)y_1 = 0. \end{aligned}$$

Because $x_2 - 4 \neq 0$ and $y_1 \neq 0$, system (2.4) must have the following equilibrium states:

$$e_1 = 0, \qquad e_2 = C.$$

Demanding that all of the other equations of system (2.4) are zero, we get the following. First, $e_1 = -e_3$, and so $e_3 = 0$. Then, by letting

$$\dot{e}_3 = e_2 - e_3 + (\alpha - i) = 0,$$

we have $e_2 + (\alpha - i) = 0$, that is, $e_2 = -e_4$. Moreover, it follows from (2.3) that $e_4 = \alpha - i = -e_2 = -C$, so that $\dot{e}_4 = 0$, and $i = \alpha + C$. By letting

$$\dot{e}_5 = (P - 4)' = 0,$$

we have

$$e_5 = P - 4 = C_1,$$

where C_1 is a constant. When the synchronization error system is in its equilibrium state, $e_1 = y_1 - x_1 = 0$, that is, $x_1 = y_1$, and so $\dot{x}_1 = \dot{y}_1$. Therefore, from the HCSS (2.2), the following equation can be introduced:

$$2 + x_1(x_2 - 4) = 2 + y_1(y_2 - P).$$

This gives

$$x_2 - 4 = y_2 - P,$$

that is,

$$y_2 - x_2 = P - 4.$$

Since

$$y_2 - x_2 = e_2 = C$$

and

$$P - 4 = e_5,$$

we have

$$e_5 = e_2 = C.$$

In summary, the synchronization error system has the following equilibrium state:

$$e^* = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{bmatrix} = \begin{bmatrix} 0 \\ C \\ 0 \\ -C \\ C \end{bmatrix}, \tag{2.5}$$

where C is a constant.

Therefore, if the hybrid chaos synchronization takes place when $t \to \infty$, then the result we can expect is

$$y_1(t) = x_1(t), \quad y_2(t) - C = x_2(t), \quad y_3(t) = x_3(t), \quad i(t) - C = \alpha(t), \quad P - C = 4. \tag{2.6}$$

Strictly speaking, this is not an IS between the first and the second chaotic systems because only $y_1(t) = x_1(t)$ and $y_3(t) = x_3(t)$, but $y_2(t) \neq x_2(t)$, $i(t) \neq \alpha(t)$, and $P \neq 4$. This phenomenon is not phase synchronization nor is it orbital synchronization [1,2,16]. It is considered as a particular GS with constant offsets in this study.

2.3. Adaptive Control of the HCSS

For the expected hybrid chaos synchronization to take place, we need to control the error synchronization system (2.4) to reach its equilibrium state. For this purpose, we use two unimodal control functions, $g_1(\cdot)$ and $g_2(\cdot)$, respectively,

$$g_1(e_3) = \frac{ke_3}{1 + (e_3/e_{01})^4} \tag{2.7}$$

and

$$g_2(e_2) = \frac{\varepsilon e_2}{1 + (e_2/e_{02})^6}, \tag{2.8}$$

where $k = g_1'(0)$ and $\varepsilon = g_2'(0)$ are constants, called the stiffness of the controller. The constants e_{01} and e_{02}, and the powers of (e_3/e_{01}) and (e_2/e_{02}) in (2.7) and (2.8) are determined based on our design consideration, as further explained.

Figure 2 shows these two functions, with $g_1(0) = 0$, $g_2(0) = 0$, and $g_2(e_2) \to 0$ when $e_2 \to \pm\infty$.

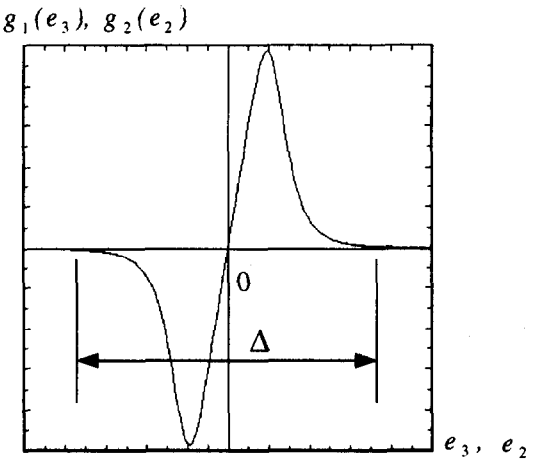


Figure 2. Unimodal function used as control functions $g_1(e_3)$ and $g_2(e_2)$.

This kind of unimodal function can supply three possible equilibrium states for the synchronization error system (2.4), that is, e_3 (or e_2) $= 0$ and e_3 (or e_2) $\to \pm\infty$.

The unimodal function g_1 supplies one possible equilibrium state for the negative feedback control

$$\dot{e}_4 = -g_1(e_3) \to 0, \qquad \text{as } e_3 \to 0.$$

The unimodal function g_2 supplies two possible equilibrium states for the negative feedback control

$$\dot{e}_5 = -g_2(e_2) \to 0, \qquad \text{as } e_2 \to \pm\infty.$$

The equilibrium state of the synchronization error can be explained as follows. From (2.5), we know that the equilibrium state has the synchronization error $e_3 = 0$. So, the equilibrium state must fall to the point $g_1(e_3) \to 0$, when $e_3 \to 0$ on the unimodal function g_1.

Since $i(t)$ is an arbitrary input signal, $\alpha(t)$ tracks $i(t)$ via adaptive control which usually has a nonzero residue, i.e., $e_4(t) = \alpha(t) - i(t)$ is usually nonzero. Once $|e_4(t)| > \Delta/2$, that is, $|e_2(t)| > \Delta/2$, the equilibrium state of the error system (2.4) will jump over the unimodal peak of the control function g_2 from $g_2(0) = 0$ and converge to the region of $|e_2(t)| > \Delta/2$, it will lead to $g_2(e_2) \to 0$, when $e_2 \to \pm\infty$.

Consequently, $\dot{e}_5 = -g_2(e_2) \to 0$, namely

$$e_5 \to C,$$

which is the equilibrium state of the synchronization error system. This means that the HCSS will stay in the state where both y_1 tracks x_1 exactly, and y_3 tracks x_3 exactly, but y_2 tracks x_2 with a constant offset, $e_2(t) = C$, which has an opposite sign to the difference $e_4(t)$. And the parametric function $P(t)$ traces the value 4.0 by a constant offset, $e_5(t) \to C$, since $\dot{e}_5 = -g_2(e_2) \to 0$. This proves that if $|e_2(t)| > \Delta/2$, the HCSS is in its stable equilibrium state. Thus, the parametric function $P(t)$ converges to a constant value, $C + 4$.

3. NUMERICAL EXPERIMENTS AND DISCUSSIONS

To verify the design of the HCSS, we have performed a large number of numerical experiments, some of which are reported in this section. Since our numerical experiments are computer-based, a signal is actually a time series with discrete values in double precision. This is a difference from the "typical" or "theoretical" master-slave synchronization, in which one might think of the master as an oscillator creating the drive signal at the same time as the slave's response.

3.1. Various Input Signals

To experimentally verify the capability of the HCSS, we tested many finite-length signals sampled from handwritten numerals (input using a writing-tablet), chaotic trajectories, constant numbers, sine and cosine curves, and impulse signals, etc. We have selected some of these signals to be representative of "common" types of functions "in practice".

For standardization, we first normalize all input signals to the range $[-0.5, 0.5]$. Figures 3–5 show the finite-length normalized time series which we use in our numerical experiments.

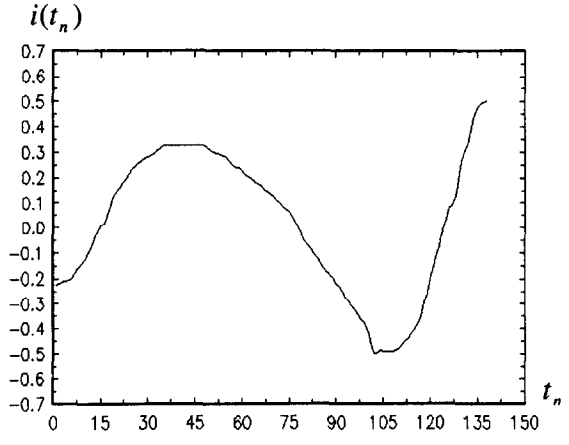


Figure 3. The input signals sampled from the x coordinates of handwritten numeral "2" after normalization.

We also include an example of a chaotic trajectory as the input signal in our numerical experiments, which was generated by Chua's circuit, as shown in Figure 4.

For our final example, we have chosen the impulse signal shown in Figure 5 as input to the HCSS.

3.2. HCSS in Practice

To encode a signal by using the designed HCSS, equation (2.1), the length of the input signal must be determined as a reference. This is because the convergent P is also relative to the length of the input signal. After we encode and store the input signals by means of keeping their corresponding parameter values P, the HCSS is readily used for future input signal matching (or identification). When a new signal is input to the HCSS for identification or recognition, the same length of this signal must be used. We call this total number of signal points the "critical

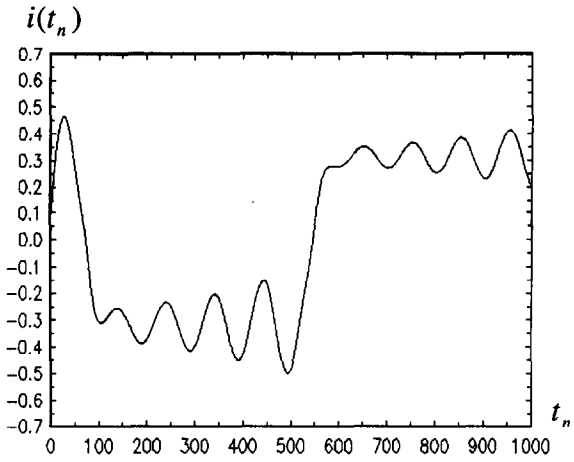


Figure 4. The x coordinate of the input chaotic trajectory of Chua's circuit.

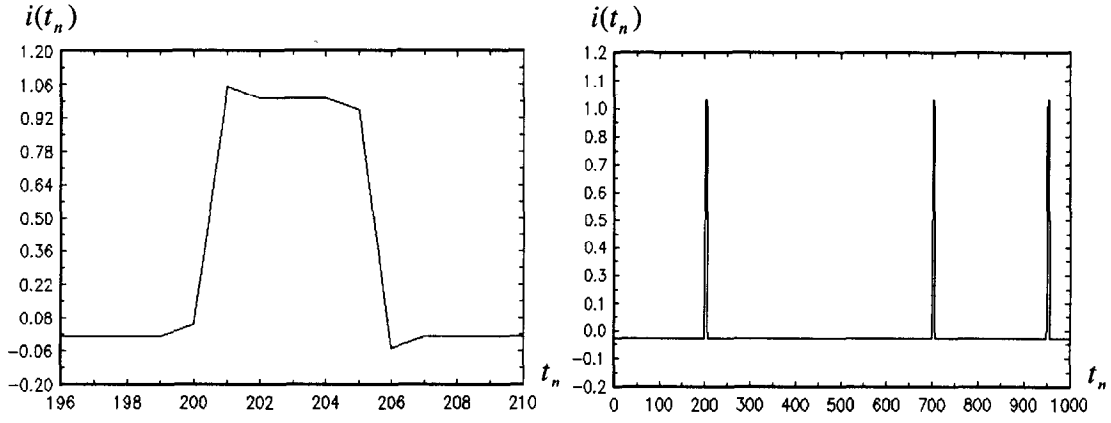


Figure 5. The input impulse signal with amplitude 1.0 and width five time units, (a) one enlarged impulse, (b) three impulses in 1000 points.

length" of the encoding. Since the encoding process is by chaos synchronization, which includes a transient period, the critical length of the input signals in application should be chosen longer than that of the transient period, in order to ensure convergence of the parametric function P.

The initial setup of our numerical experiments was such that it can ensure the parameter P to converge within (less than) 1000 point units, that is, 1000 integration steps. On the other hand, the input signals used in the numerical experiments are also less than or equal to 1000 points long (see Figures 3–5). So, we chose the critical length input signals to be 1000 points. For an input signal that is shorter than the critical length, we augment it with appended zeros, but when longer, we put it into another class of signals, with a longer critical length.

In the following numerical experiments, the first chaotic system has initial conditions:

$$x_1(0) = 0.5, \qquad x_2(0) = 0.5, \qquad x_3(0) = 0.7, \tag{3.1}$$

and the second has

$$y_1(0) = 0.1, \qquad y_2(0) = 0.2, \qquad y_3(0) = 0.3, \qquad \alpha(0) = 3.0, \qquad P(0) = 5.0. \tag{3.2}$$

All of these numerical experiments were performed using the fourth-order Runge-Kutta integration algorithm with integration step of 0.05.

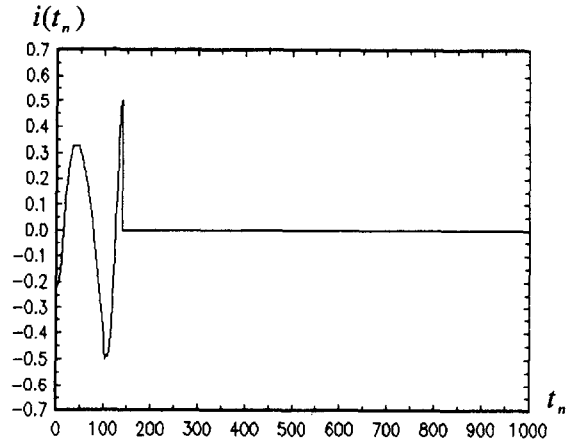


Figure 6. The x coordinates of the extended signal sampled from the handwritten numeral "2".

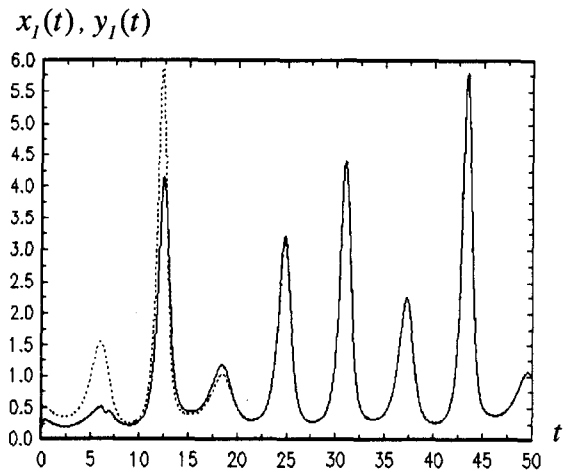


Figure 7. Trajectories $x_1(t)$ (dotted line) and $y_1(t)$ (solid line) synchronize identically in the HCSS when the input signal sampled from the handwritten numeral "2" is encoded (integration step of 0.05 is used). Likewise, $e_3(t) = y_3(t) - x_3(t) \to 0$, not shown.

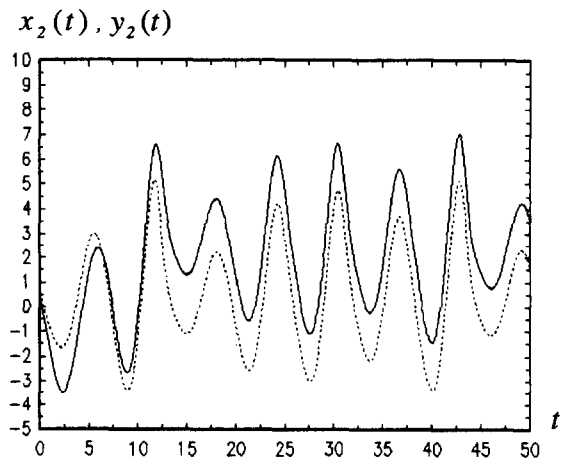


Figure 8. Trajectories $x_2(t)$ (dotted line) and $y_2(t)$ (solid line) of the HCSS when the input signal sampled from the handwritten numeral "2" is encoded (integration step of 0.05 is used).

3.2.1. Encoding the handwritten numeral "2"

We now discuss the example of an input signal sampled from the handwritten numeral "2" shown in Figure 3. We must append zeros to augment to a total of 1000 points, shown in Figure 6, before it is input to the HCSS.

Figures 7 and 8 show the synchronization process, the trajectories x_1 and y_1, x_2 and y_2, (and likewise for x_3 and y_3 not shown), in the HCSS, respectively, when the extended signal sampled from the handwritten numeral "2" is encoded.

Figure 8 shows the diagrams of the input signal $i(t)$, its trace. Figure 9 shows the diagrams of three error states $e_1(t), e_2(t), e_3(t)$ and the resulting parameter P.

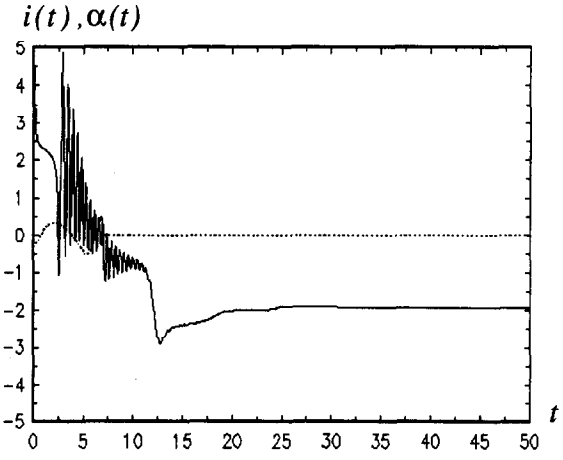
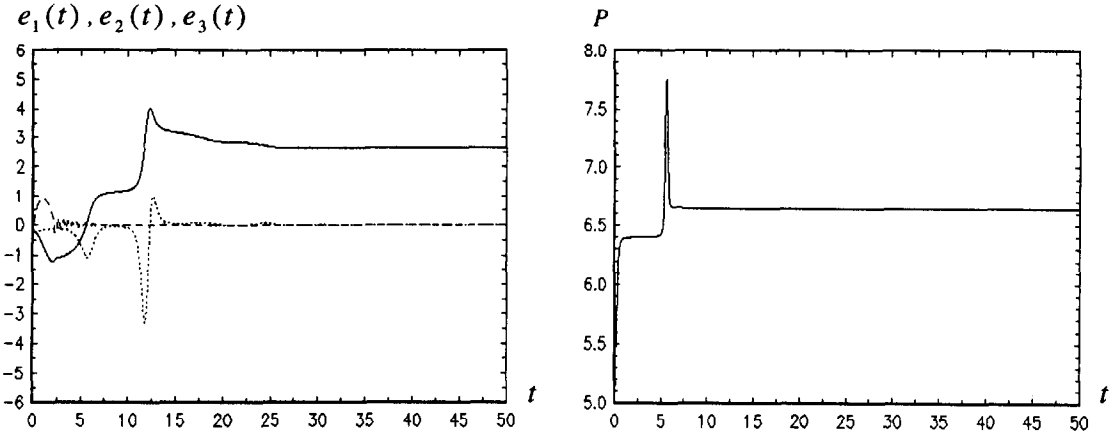


Figure 9. Input signal $i(t)$ (dotted line) and $\alpha(t)$ (solid line) in the HCSS, when the input signal sampled from the handwritten numeral "2" is encoded (integration step of 0.05 is used).



(a) (b)

Figure 10. (a) Errors $e_1(t) = y_1(t) - x_1(t)$ (dotted line), $e_2(t) = y_2(t) - x_2(t)$ (solid line), and $e_3(t) = y_3(t) - x_3(t)$ (dashed line), (b) the resulting parameter P of the HCSS, when the input signal sampled from the handwritten numeral "2" is encoded (integration step of 0.05 is used).

From these phase diagrams, one can see that the first chaotic system exactly synchronizes with the second chaotic system in trajectories x_1 and y_1, (and x_3 to y_3, not shown), that is, $e_1(t) = y_1(t) - x_1(t) \to 0$, $e_3(t) = y_3(t) - x_3(t) \to 0$, but trajectories x_2 and y_2 synchronize with a constant offset, $e_2(t) = y_2(t) - x_2(t) \to C$ (constant), which determines the resulting

parameter P. Also, $e_4(t) = \alpha(t) - i(t) \to -C$ (constant), so that $P = 4 - e_4(t) \to 4 + C$. All of these are consistent with the theoretical analysis given in Section 2.

It is very interesting to see the phenomenon that the two Rossler attractors run synchronously but are separated by a constant offset forever, *even if the input signal is taken away after synchronization*, as shown in Figure 11. This special property of the designed HCSS may have potential for new information processing and memory applications—a topic for future investigation elsewhere.

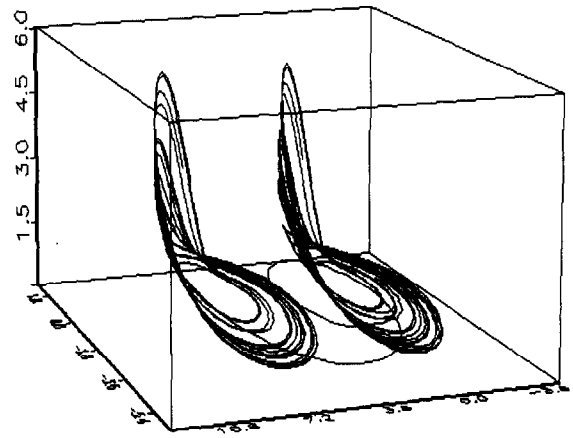


Figure 11. Two Rossler attractors (of the first and the second chaotic systems) in the HCSS run separately forever with a constant distance depending on the input signal, even if the input signal is taken away after synchronization.

3.2.2. Encoding a chaotic trajectory of Chua's circuit

As the next example, we now input into the HCSS the 1000 point chaotic trajectory of Chua's circuit, shown in Figure 4.

Figures 12–15 show the synchronization process.

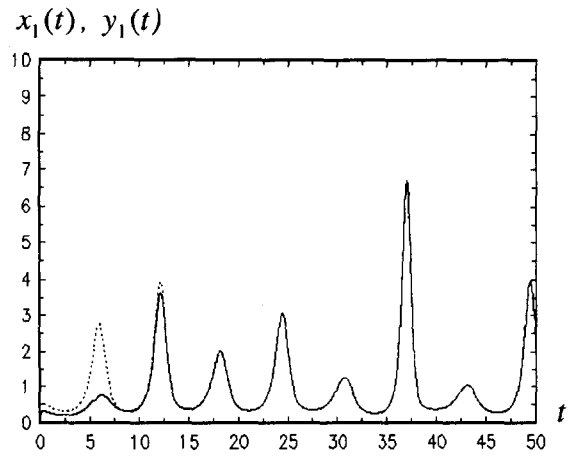


Figure 12. Trajectories $x_1(t)$ (dotted line) and $y_1(t)$ (solid line) in the HCSS, when a chaotic trajectory of Chua's circuit is encoded (integration step of 0.05 is used).

3.2.4. Encoding an impulse signal

The 1000 point impulse input signal used for this next numerical experiment is shown in Figure 5. Figures 16–19 show the resulting synchronization process.

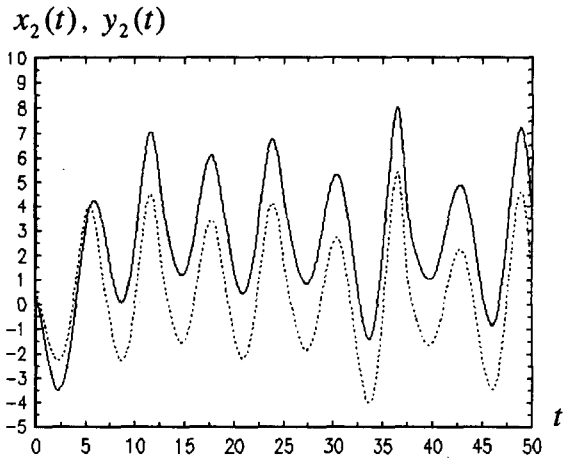


Figure 13. Trajectories $x_2(t)$ (dotted line) and $y_2(t)$ (solid line) of the HCSS, when a chaotic trajectory of Chua's circuit is encoded (integration step of 0.05 is used).

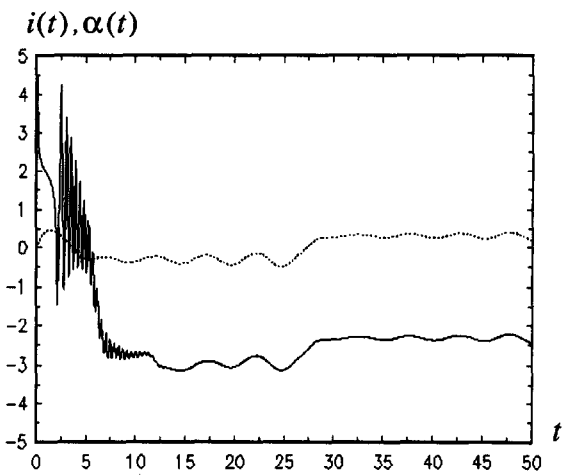
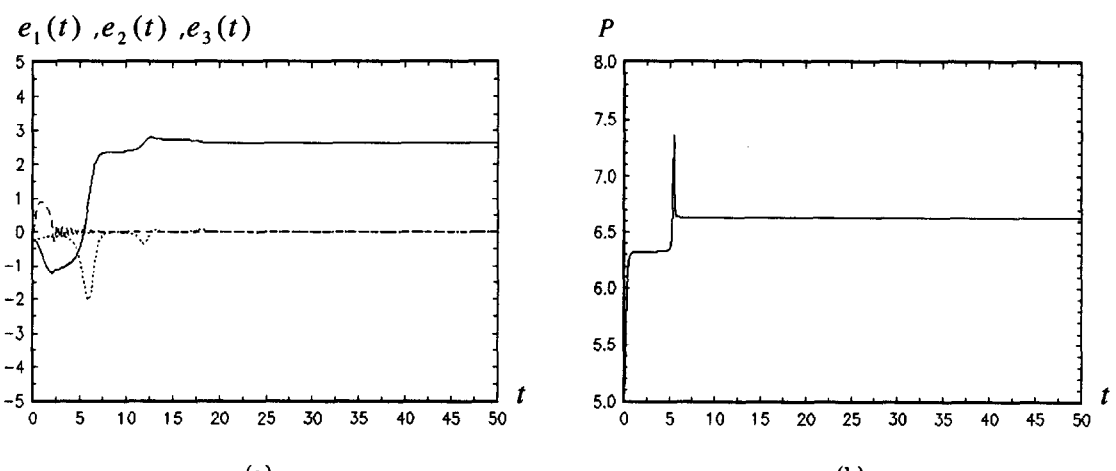


Figure 14. Input signal $i(t)$ (dotted line) and $\alpha(t)$ (solid line) in the HCSS, when a chaotic trajectory of Chua's circuit is encoded (integration step of 0.05 is used).



(a) (b)

Figure 15. (a) Errors $e_1(t) = y_1(t) - x_1(t)$ (dotted line) and $e_2(t) = y_2(t) - x_2(t)$ (solid line) and $e_3(t) = y_3(t) - x_3(t)$ (dashed line), (b) the resulting parameter P in the HCSS, when a chaotic trajectory of Chua's circuit is encoded (integration step of 0.05 is used).

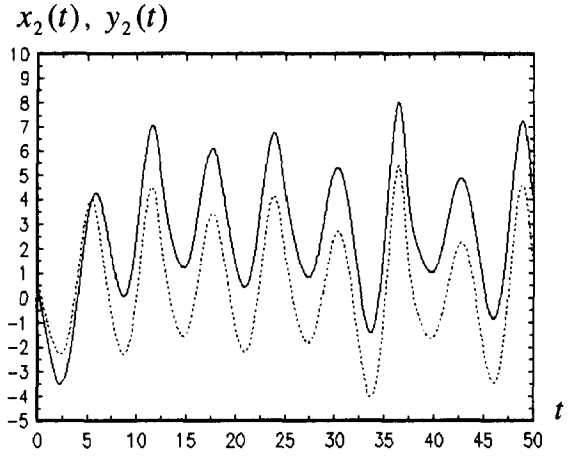


Figure 16. Trajectories $x_1(t)$ (dotted line) and $y_1(t)$ (solid line) in the HCSS, when the impulse signal is encoded (integration step of 0.05 is used).

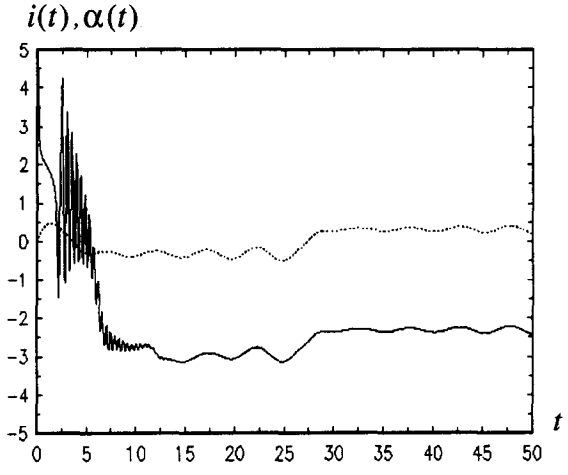


Figure 17. (a) Trajectories $x_2(t)$ (dotted line) and $y_2(t)$ (solid line), (b) error diagram in the HCSS, when the impulse signal is encoded (integration step of 0.05 is used).

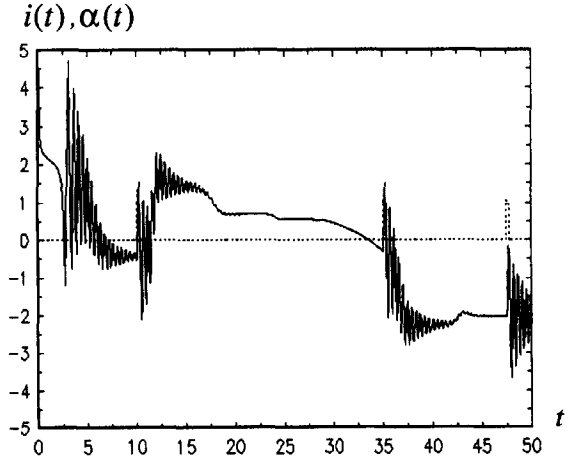
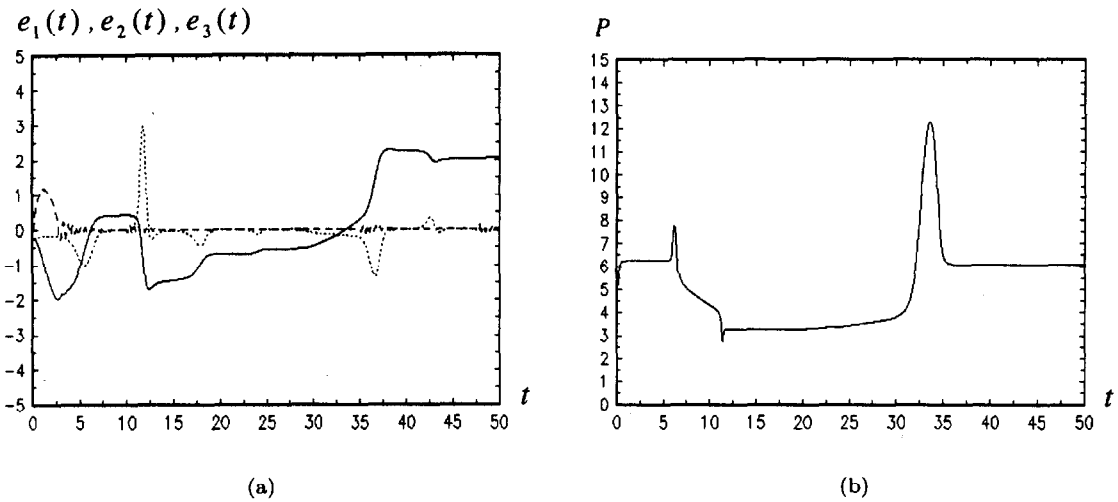


Figure 18. Input signal $i(t)$ (dotted line) and $\alpha(t)$ (solid line) in the HCSS, when the impulse signal is encoded (integration step of 0.05 is used).



(a) (b)

Figure 19. (a) Errors $e_1(t) = y_1(t) - x_1(t)$ (dotted line) and $e_2(t) = y_2(t) - x_2(t)$ (solid line) and $e_3(t) = y_3(t) - x_3(t)$ (dashed line), (b) the resulting parameter P diagram in the HCSS, when the impulse signal is encoded (integration step of 0.05 is used).

3.2.5. A brief summary

We draw the following conclusions from the above numerical experiments of the hybrid chaos synchronization processes in encoding different input signals.

First, each of these experiments is consistent with the theoretical analysis given in Section 2.

Second, under the initial conditions (3.1) and (3.2), chaos synchronization of the HCSS passes through its transient period within a time period less than the critical length (1000 points in this study). Therefore, HCSS can be considered to have a transient phase and then a steady phase. The transient phase of the HCSS mainly determines the first several digits of the resulting parameter value P, and the steady phase affects the high-precision digits of this parameter.

Third, concerning the unimodal control function $g_2(e_2)$ (see Figure 2), if the stable equilibrium point falls into the range of $e_2 = 2.0 \sim 3.0$, then the precision of the resulting parameter P obtained is approximately $10^{-5} \sim 10^{-7}$ (further discussion will be given later). The convergence of the resulting parameter P can be adjusted by changing the shape of the unimodal control function. This means that the higher the power of (e_2/e_{02}) is, the faster the resulting parameter P converges, so that the later digital points in the input signal have less effect on the resulting parameter value P.

Finally, the initial value $\alpha(t)$ has a close relationship to the transient time of chaos synchronization of the HCSS according to our numerical experiments. If the input is normalized to be within the range $[-0.5, 0.5]$ and if the initial value $\alpha(0)$ is 3.0, then the transient time of the HCSS will be less than two hundred time units out of 1000 points for most of input signals simulated. If the initial value $\alpha(0)$ is chosen to be 5.0, the transient time will be less than 400 time units. That is, adjusting the initial values from $\alpha(0) = 3.0$ to $\alpha(0) = 5.0$ will prolong the transient phase. In general, the longer the transient phase, the longer the input signal the HCSS can encode. In practice, this property can be used for encoding different lengths of different input signals.

3.3. Resulting Parameters P from Different Input Signals

For the signals used in the above numerical experiments, the resulting parameters P and the changes corresponding to variations of the second or the last digital of different input signals are shown in Tables 1–3. The first column of each table shows the digital index number of an input signal. The second column shows the normalized values of every point of the input signal. The third column shows the point value, which is the same as that of the second column except the

Table 1. Encoding the signal of handwritten numeral "2".

Sample	Sample Value	Change the 2nd Sample	Change Last Sample
1	−0.22316	−0.22316	−0.22316
2	−0.22029	−0.22028	−0.22029
...	...	...	...
137	0.49809	0.49809	0.49809
138	0.50000	0.50000	0.50001
139	0.00000	0.00000	0.00000
140	0.00000	0.00000	0.00000
...	...	...	...
1000	0.00000	0.00000	0.00000
P	5.92134767725156	5.92135**419545282**	5.92134767**531791**

Table 2. Encoding a chaotic orbit of Chua's circuit.

Sample	Sample Value	Change the 2nd Sample	Change Last Sample
1	0.083475	0.083475	0.083475
2	0.108855	0.108854	0.108855
...	...	...	...
998	0.208520	0.208520	0.208520
999	0.205635	0.205635	0.20564**5**
1000	0.203150	0.203150	0.203150
P	6.63184318818670	6.63184**285487491**	6.63184318818671

Table 3. Encoding an impulse signal.

Sample	Sample Value	Change the 201st Sample	Change the 956th Sample
1	0.00000	0.00000	0.00000
...	...	...	...
199	0.00000	0.00000	0.00000
200	0.05000	0.05000	0.05000
201	1.05000	1.05001	1.05000
202	1.00000	1.00000	1.00000
203	1.00000	1.00000	1.00000
204	1.00000	1.00000	1.00000
205	0.95000	0.95000	0.95000
206	−0.05000	−0.05000	−0.05000
207	0.00000	0.00000	0.00000
...	...	...	...
949	0.00000	0.00000	0.00000
950	0.05000	0.05000	0.05000
951	1.05000	1.05000	1.05000
952	1.00000	1.00000	1.00000
953	1.00000	1.00000	1.00000
954	1.00000	1.00000	1.00000
955	0.95000	0.95000	0.95000
956	−0.05000	−0.05000	−0.05001
957	0.00000	0.00000	0.00000
...	...	...	...
999	0.00000	0.00000	0.00000
1000	0.00000	0.00000	0.00000
P	6.09405030372185	6.093**99622554569**	6.0940503037**2003**

fifth digit (which has a change in the hundred-thousandth precision indicated in bold font). The last column shows the same point values as that of the second column except that the last point value has a change in the hundred-thousandth precision (in bold font). The last row of each table shows the resulting parameter P estimated by the HCSS from the input signal located in the same column. The bold fonts in parameter P tell their differences from the original parameter value of P.

Considering Tables 1–3, we observe the following. First, the parameter P is sensitive to small changes in the input signal. So, it can be used to encode (represent) this signal distinguishably, and these representations are unique within the monotone region of the parameter P (we will discuss this issue further).

Second, in general, the beginning points in column two of each table are more significant than the later points, in the sense that they have greater effects on the final value of the resulting parameter P. This reflects the sensitivity to the changes of its initial conditions of chaotic dynamical systems. Therefore, the output trajectory will be increasingly effected as time evolves.

3.4. Control Functions in the HCSS

In all of the numerical experiments described above, we used the control functions

$$g_1(e_3) = \frac{ke_3}{1 + (e_3/e_{01})^4}$$

and

$$g_2(e_2) = \frac{\varepsilon e_2}{1 + (e_2/e_{02})^6}$$

to force the HCSS to converge quickly to its stable equilibrium state. As pointed out earlier, the power of (e_2/e_{02}) in the control function g_2 will affect the precision of convergence of the parameter P, which was chosen carefully based on trial-and-error. Besides, large stiffness k and ε values of the unimodal control functions g_1 and g_2 should be chosen, and the constant parameter e_{02} (or e_{01}) determines the desired Δ of the unimodal control functions. For all numerical experiments reported above, $k = 211$, $\varepsilon = 51$, $e_{01} = 0.2$, and $e_{02} = 0.2$ were used, again, on the basis of trial-and-error.

Figure 20 shows the convergence segments in a small scale for the stable equilibrium state of the HCSS when the parameter P is within the range [6, 7] (that is, e_2 is in the range [2, 3], since

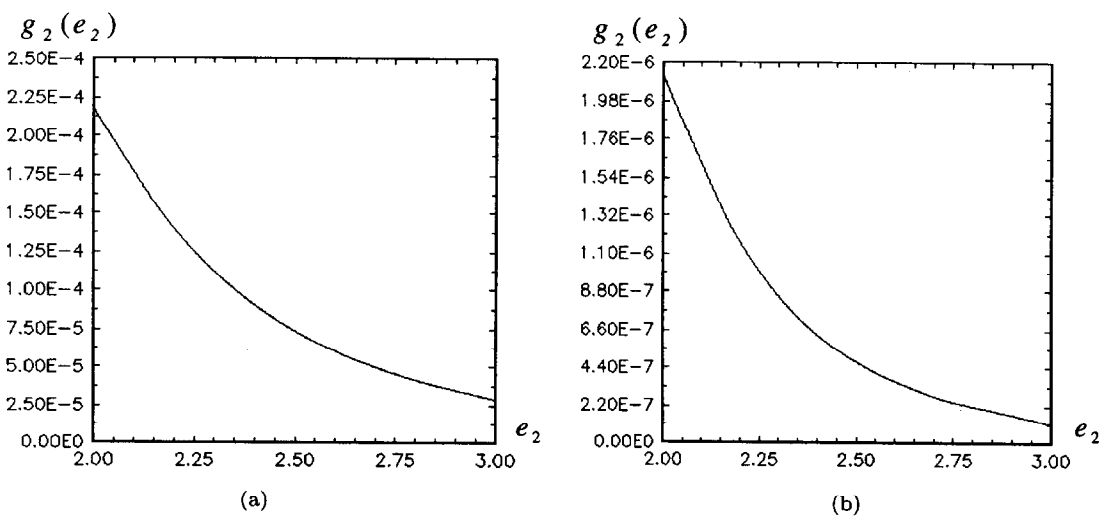


Figure 20. A segment of unimodal function (a) with the sixth power, (b) with the eighth power of (e_2/e_{02}) used as control function $g_2(e_2)$.

$P = e_2 + 4$). In this case, the feedback control function $g_2(e_2)$ has about 10^{-5} or 10^{-7} in precision for the sixth power or the eighth power of (e_2/e_{02}), respectively, as can be seen in Figure 20.

4. OBSERVATIONS AND DISCUSSIONS

The methodology explored in this paper enables us to encode an arbitrary signal accurately and efficiently. The crucial condition for the success of this method is a noise-free environment for the precise representations of different signals. Therefore, potential applications of the designed HCSS are likely to be in information encoding and storage, and certain security systems such as password check and key identification, rather than noisy signal communication and pattern recognition. This HCSS has some useful properties:

(1) a unique parameter value for each input signal within a monotone region,
(2) continuity of the parameter with respect to the amplitude of the input signals, and
(3) sensitivity of the parameter with respect to the time shift of the input signals, etc.

4.1. Continuity of the Parametric Function P to the Amplitude of the Input Signals

Since an input signal $i(t)$ has a continuous GS with the first chaotic system, the output $v(t)$ of the first chaotic system must be a continuous image of $i(t)$. Similarly, the $v(t)$ has IS with the second chaotic system, so the output parametric function P must also be continuous. Therefore, the resulting one-dimensional output parameter P, with respect to the one-dimensional input signal $i(t)$ of the HCSS, is determined by some continuous mapping. That is, the parameter P must have a (small) defining domain with respect to very small changes of the amplitude of the input signal $i(t)$. This means, that if the amplitude of the input signal increases or decreases within a very small range, then the resulting parameter P will vary monotonously, also within a very small domain.

The following numerical experimental results, with a small variation of the amplitude of the input signal sampled from the handwritten numeral "0", agree that the resulting parameter P indeed changes slightly and monotonously. These numerical experiments use initial conditions (3.1) and (3.2), a critical length of 1000 points, and the fourth-order Runge-Kutta integration algorithm with integration step size 0.1. The sixth power of (e_2/e_{02}) is used in the control function $g_2(e_2)$. Other setups are the same as the numerical experiments reported above. Table 4 and Figure 21 show the experimental data, which agree with the continuity property of HCSS. It can be seen from Figure 21 that there is a large monotone domain $(-0.015, 0.005)$ within which there is a linear domain $(-0.01, 0.003)$. This is a useful region to be used for real applications, since it guarantees the uniqueness of the mapping $i(t) \to P$.

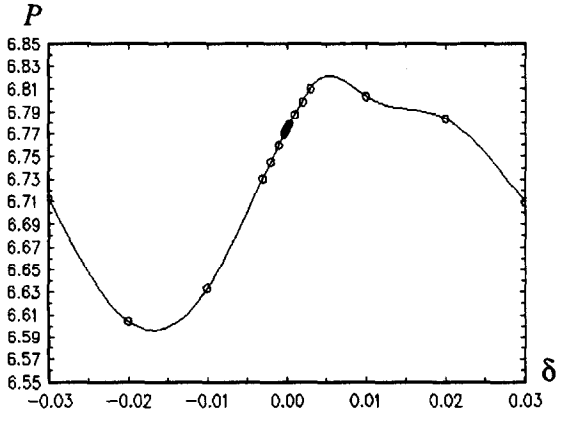


Figure 21. Continuity of the resulting parameter P vs. small amplitude changes δ of the input signal for handwritten numeral "0".

Table 4. Continuity of the resulting parameter P vs. small changes δ of the amplitude of the input signal, handwritten numeral "0".

Changes δ of Amplitude of Input Signal	Parameter P	Changes δ of Amplitude of Input Signal	Parameter P
+0	6.77418287166439	−0	6.77418287166439
+0.00001	6.77432310165669	−0.00001	6.77404250139414
+0.00002	6.77446319010803	−0.00002	6.77390199211087
+0.00003	6.77460313575773	−0.00003	6.77376134508150
...	...	...	...
+0.00010	6.77557865151863	−0.00010	6.77277306571141
+0.00020	6.77695915637546	−0.00020	6.77135050347900
+0.00030	6.77832317811666	−0.00030	6.76991645819652
...	...	...	...
+0.00100	6.78732810978530	−0.00100	6.75965825142054
+0.00200	6.79855498498728	−0.00200	6.74482639857012
+0.00300	6.81020751576172	−0.00300	6.73027901435812
...	...	...	...
+0.01000	6.80340083107163	−0.01000	6.63345935170303
+0.02000	6.78345089962921	−0.02000	6.60426930043395
+0.03000	6.70932426282247	−0.03000	6.71172291891259

Here, we outline proof of the property that the resulting parameter P has a unique value for each input signal within its monotone region.

REMARK 4.1. Given a chaotic dynamical system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$, which satisfies the hypothesis of the existence and uniqueness theorem (see, e.g., [17]) for all time t, and given two sets of initial conditions $\{\mathbf{x}(t_0), \mathbf{x}(t_1), \ldots, \mathbf{x}(t_n)\}$ and $\{\mathbf{x}'(t_0), \mathbf{x}'(t_1), \ldots, \mathbf{x}'(t_n)\}$, where $t_j > t_{j-1}$, $1 \leq j \leq n$, the phase trajectories starting from these two initial condition sets will coincide if and only if $\{\mathbf{x}(t_0), \mathbf{x}(t_1), \ldots, \mathbf{x}(t_n)\} = \{\mathbf{x}'(t_0), \mathbf{x}'(t_1), \ldots, \mathbf{x}'(t_n)\}$. This statement follows by considering two initial conditions $\mathbf{x}(t_j) \neq \mathbf{x}'(t_j)$ at time $t = t_j$. Since the overall system is autonomous, phase trajectories starting from these two initial conditions never intersect, and since the system is chaotic, the two trajectories are expected to diverge at approximately the rate of the largest positive Lyapunov exponent of the system. Therefore, the two trajectories cannot coincide. Conversely, if two phase trajectories of system starting from these two initial condition sets coincide, then by uniqueness of solutions, these two trajectories must be identical. Hence, $\{\mathbf{x}(t_0), \mathbf{x}(t_1), \ldots, \mathbf{x}(t_n)\} = \{\mathbf{x}'(t_0), \mathbf{x}'(t_1), \ldots, \mathbf{x}'(t_n)\}$.

REMARK 4.2. For the HCSS shown in Figure 1, input two signals, $\{i(t_0), i(t_1), \ldots, i(t_n)\}$ and $\{i'(t_0), i'(t_1), \ldots, i'(t_n)\}$, where $t_j > t_{j-1}$, $1 \leq j \leq n$. Then, the corresponding parametric functions $P(t)$ and $P'(t)$ of the HCSS are identical if and only if $\{i(t_0), i(t_1), \ldots, i(t_n)\} = \{i'(t_0), i'(t_1), \ldots, i'(t_n)\}$. This follows from the fact that the HCSS in Figure 1 is an augmented autonomous system, in which $i(t)$, $v(t)$, $P(t)$, and $\alpha(t)$ are (new) state variables. By Remark 4.1, the corresponding parametric functions $P(t)$ and $P'(t)$ of the HCSS are identical if and only if $\{i(t_0), i(t_1), \ldots, i(t_n)\} = \{i'(t_0), i'(t_1), \ldots, i'(t_n)\}$.

4.2. Sensitivity of the Parameter P to Time Shifts of Input Signals

To verify the property of sensitivity of the parameter P with respect to the time shifts of the input signals, input a rectangular signal $i(t_n)$, beginning at the time $t_n = 700$ with height 0.5 and width 50, is shifted on the time axis from the left to the right, or *vice versa*. There is no amplitude change throughout the process, as shown in Figure 22. As is clear, the resulting value of the parameter P is changed dramatically by these time shifts.

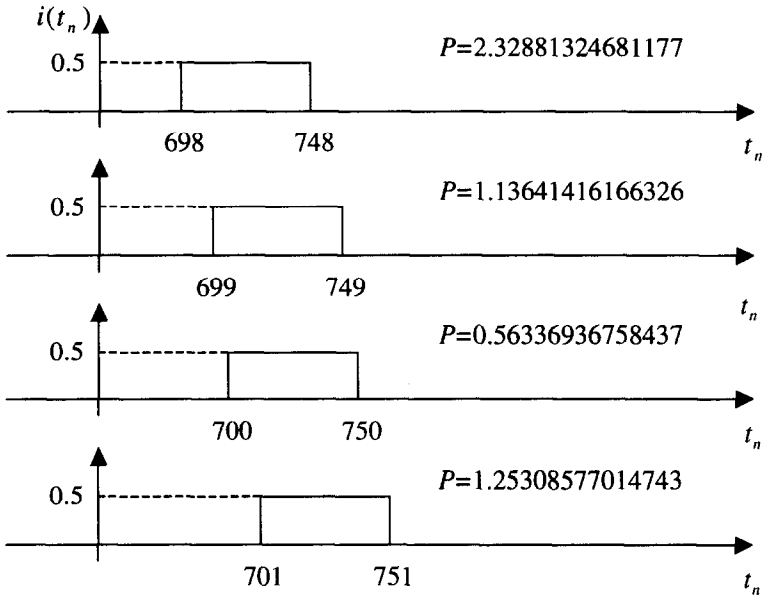


Figure 22. Rectangular input signal $i(t)$, with height 0.5 and width 50, arrives at time t_{698}, t_{699}, t_{700}, and t_{701}, respectively, and the corresponding parameters P.

5. CONCLUSIONS

Through sufficient numerical studies, this paper explores a new methodology for chaos synchronization via a hybrid (generalized plus identical) synchronization. An arbitrary signal, generated by an unknown dynamical system, is synchronized by the hybrid chaotic system and then stored for recognition or identification. Each finite-length signal can in principle be labelled and stored by a unique number, provided that the key hybrid system parameter used for this labelling is suitably chosen within a one-to-one mapping range. The new methodology enables us to encode an arbitrary signal accurately and efficiently. The Rossler chaotic oscillator is used as the platform for the designed hybrid chaos synchronization system. In addition to some basic theoretical analysis, many numerical simulations have been tested which verify the proposed design. A basic requirement for this hybrid chaos synchronization system is a noise-free environment. Therefore, potential applications of the system is in exact information processing such as message storage and password recognition, as well as high-accuracy secure systems.

REFERENCES

1. L.M. Pecora and T.L. Carroll, Synchronization in chaotic systems, *Physical Review Letters* **64**, 821–823 (1990).
2. L.M. Pecora and T.L. Carrol, Driving systems with chaotic signals, *Physical Review A* **44**, 2374–2383 (1991).
3. N.F. Rulkov, M.M. Sushchik and L.S. Tsimring, Generalized synchronization of chaos in directionally coupled chaotic systems, *Physical Review E* **51**, 980–994 (1995).
4. M.J. Ogorzalek, Taming chaos: Part I—Synchronization. Part II—Control, *IEEE Transactions on Circuits and Systems* **40**, 693–706 (1993).
5. Lj. Kocarev, T. Stojanovski and L. Panovski, Generalized synchronization of chaos, In *Proceedings of IEEE International Symposium on Circuits and Systems, Volume 3*, Atlanta, GA, pp. 116–119, (1996).
6. U. Parlitz, Lj. Kocarov, T. Stojanovski and H. Preckel, Encoding messages using chaotic synchronization, *Physical Review E* **53**, 4351–4361 (1996).
7. G. Kolumban, M.P. Kennedy and L.O. Chua, The role of synchronization in digital communications using chaos, *IEEE Transactions on Circuits and Systems* **44**, 927–936 (1997).
8. G. Chen and X. Dong, *From Chaos to Order: Methodologies, Perspectives, and Applications*, World Scientific, Singapore, (1998).
9. G. Chen, Editor, *Controlling Chaos and Bifurcations in Engineering Systems*, CRC Press, Boca Raton, FL, (1999).

10. M.P. Kennedy and G. Kolumban, Digital communications using chaos, In *Controlling Chaos and Bifurcations in Engineering Systems*, (Edited by G. Chen), pp. 464–487, CRC Press, Boca Raton, FL, (1999).
11. U. Parlitz, L. Junge and Lj. Kocarev, Synchronization-based parameter estimation from time series, *Physical Review E* **56**, 6253–6259 (1996).
12. Lj. Kocarev and U. Parlitz, Generalized synchronization, predictability, and equivalence of unidirectionally coupled dynamical systems, *Physical Reviews Letters* **76**, 1816–1819 (1996).
13. C.W. Wu, T. Yang and L.O. Chua, On adaptive synchronization and control of nonlinear dynamical systems, *International Journal of Bifurcation and Chaos* **6**, 455–471 (1996).
14. L.O. Chua, T. Yang, G.Q. Zhong and C.W. Wu, Adaptive synchronization of Chua's oscillators, *International Journal of Bifurcation and Chaos* **6**, 189–201 (1996).
15. P. Celka, Synchronization of chaotic systems through parameter adaptation, In *Proceedings of 1995 IEEE International Symposium on Circuits and Systems*, Seattle, WA, pp. 692–695, (1995).
16. Q. Xie and G. Chen, Synchronization stability analysis of the chaotic Rossler system, *International Journal of Bifurcation and Chaos* **6**, 2153–2161 (1996).
17. C. Robinson, *Dynamical Systems: Stability, Symbolic Dynamics, and Chaos*, CRC Press, Boca Raton, FL, (1995).
18. E. Castillo and J.M. Gutierrez, Nonlinear time series modelling and prediction using functional networks: Extracting information masked by chaos, *Phys. Lett. A* **244**, 71–84 (1998).
19. H. Dedieu and M.J. Ogorzalek, Time series coding and compression based on chaos control techniques, In *Proceedings of 1995 IEEE International Symposium on Circuits and Systems*, Seattle, WA, pp. 1191–1194, (1995).
20. A.N. Kolmogorov and S.V. Fomin, *Introductory Real Analysis*, Dover, New York, (1970).
21. T.K. Lim, K. Kwak and M. Yun, An experimental study of storing information in a controlled chaotic system with time-delayed feedback, *Phys. Lett. A* **240**, 287–294 (1998).