Pramana – J. Phys. (2018) 90:14
https://doi.org/10.1007/s12043-017-1507-8

FPGA implementation of fractional-order discrete memristor chaotic system and its commensurate and incommensurate synchronisations

ANITHA KARTHIKEYAN[1,2] and KARTHIKEYAN RAJAGOPAL[1,2,*]

[1]Department of Electrical and Communication Engineering, University of Technology, Lae, Papua New Guinea
[2]Center for Nonlinear Dynamics, Defence University, Bishoftu, Ethiopia
*Corresponding author. E-mail: rkarthiekeyan@gmail.com

MS received 10 August 2017; revised 4 September 2017; accepted 5 September 2017; published online 30 December 2017

Abstract. A new fourth-order memristor chaotic oscillator is taken to investigate its fractional-order discrete synchronisation. The fractional-order differential model memristor system is transformed to its discrete model and the dynamic properties of the fractional-order discrete system are investigated. A new method for synchronising commensurate and incommensurate fractional discrete chaotic maps are proposed and validated. Numerical results are established to support the proposed methodologies. This method of synchronisation can be applied for any fractional discrete maps. Finally the fractional-order memristor system is implemented in FPGA to show that the chaotic system is hardware realisable.

Keywords. Memristor; discretisation; fractional order; synchronisation; field programmable gate arrays.

PACS No. 05.45.–a

1. Introduction

Chaotic systems are a special case of nonlinear systems which can be categorised as chaotic if the system possesses at least one positive Lyapunov exponent and hyperchaotic if the system possesses two or more positive Lyapunov exponents. Lyapunov exponents and fractal dimension studies are important in defining the complexity of chaotic and hyperchaotic systems [1,2]. Chaotic systems found significant importance after the discovery of a 3D weather model by Lorentz [3].

By early 21st century, many researchers have announced different chaotic systems such as Chen system [4], Liu system [5], Sundarapandian system [6], Vaidhyanathan system [7], Pham system [8], etc. Chaotic systems with no equilibrium are of great interest in chaos literature. If the sum of all the Lyapunov exponents is zero then the system is a conservative system [9].

Memristors, popularly known as the fourth fundamental circuit element characterised as a nonlinear and low power device, was proposed by Chua [10,11]. In 2008 Hewlett-Packard (HP) engineers announced the first physical realisation of memristors. In memristor literatures several models had been presented such as linear and nonlinear ion drift model, threshold adaptive memristor model, etc. [12–14]. Murali–Lakshmanan–Chua's circuit with a piecewise linear active flux controlled memristors with hyperchaotic behaviour was investigated by Ahamed and Lakshmanan [15]. A memristor-based hyperchaotic complex Lu system and its adaptive synchronisation were studied by Wang *et al* [16].

Synchronisation of chaotic systems are of great importance when one chaotic oscillator drives the other. Because of high sensitivity to initial conditions, two identical chaotic systems may have exponentially diverging state trajectories. Many methods have been proposed in the literature such as active control method [17,18], adaptive control method [19,20], extended back stepping control [21,22], sliding mode control [23,24], adaptive sliding mode [25,26], etc.

Fractional calculus has fully emerged into a mathematical field with applications in nonlinear controls, electrical and mechanical controls, etc. Fractional-order chaotic systems have been investigated by many researchers [25–28]. Fractional-order controllers are more effective compared to its integer-order models especially in chaos control and synchronisation [29]. Fractional-order systems with no equilibrium are announced and investigated in [29,30]. For the numerical simulation of fractional-order chaotic system, Petras [31] explained a methodology with a register memory component.

Many methods for solving the fractional-order differential equations are also presented by many researchers [32–40]. There are many recent literatures on fractional-order memristor chaotic systems [41,42] and time-delayed memristor oscillators [43] with time-delayed synchronisation are discussed in [44]. All these literatures deal with chaotic flows and their fractional-order or time-delay analysis.

Some recent literatures have also been discussing about control and synchronisation in various chaotic flows and maps. Controlling chaotic oscillations in a Josephson junction oscillator with a modified Lyapunov function method is proposed in [45].

Fractional-order systems have been of great interest for many years [46–50]. Also, stability of fractional-order systems using Lyapunov stability theory has attracted lots of attentions [51,52]. Fractional-order controllers have been proposed for many applications such as stabilising the unstable fixed points of an unstable open-loop system [53], delayed feedback control (DFC) based on the act-and-wait concept for nonlinear dynamical systems [54] etc. Control and synchronisation of an induction motor system are investigated in [46–50,55].

The dynamical behaviour of a class of dynamical system is investigated at fixed parameter region and it is shown that the selection of the chaotic state behaviours depends on the initial conditions of the system [56]. An autapse-modulated neuron model is presented and the effect of electromagnetic induction is considered by using magnetic flux and discussed in [57]. An improved neuron model is derived to study the effect of electromagnetic induction described by using magnetic flux, and the modulation of magnetic flux on membrane potential is realised by using memristor coupling in [58].

Inspired by all the above works, in this paper we analyse a fractional-order memristor system and then transform it into its discrete version by using finite truncation method. Dynamic properties of the fractional-order discrete system are investigated. A new and novel synchronisation method for synchronising same-order and different-order fractional discrete systems are proposed and validated with fractional discrete memristor system. Numerical simulations are conducted to validate theoretical results. The 4D fractional-order memristor system is implemented in field programmable gate arrays (FPGA) using Xilinx vivado tools to show that the chaotic system is hardware realisable. The proposed synchronisation scheme is more reliable and to the best of our knowledge there are not much works related to discrete fractional-order system synchronisations. Also the proposed method can be easily realised in hardware using FPGA.

2. Fourth-order memristor system

We considered a fourth-order memristor system with its dynamics described as below:

$$\begin{aligned}
\dot{\phi}(t) &= v_1(t)\\
\dot{v}_1(t) &= \frac{1}{C_aR_a}v_2(t) - \frac{1}{C_aR_a}v_1(t) + \frac{G}{C_a}v_1(t)\\
&\quad - \frac{1}{C_a}v_1(t)W(\phi(t))\\
\dot{v}_2(t) &= \frac{1}{C_bR_a}v_1(t) - \frac{1}{C_bR_a}v_2(t) + \frac{1}{C_b}i(t)\\
\dot{i}(t) &= -\frac{1}{L}v_2(t) - \frac{R_b}{L}i(t), \qquad (1)
\end{aligned}$$

where $v_1(t)$ and $v_2(t)$ are voltages, $i(t)$ is the current, C_a, C_b, R_a, R_b represent capacitance and resistance respectively. $W(\phi(t))$ denotes the memductance function with $\phi(t)$ as the magnetic flux. L and G represent inductance and conductance. Using the mathematical model of a cubic memristor [29,30], the memductance function is given by

$$W(\phi(t)) = a + 3b\phi(t)^2\,, \qquad (2)$$

where a and b are parameters. From (1) and (2) it follows that

$$\begin{aligned}
\dot{\phi}(t) &= v_1(t)\\
\dot{v}_1(t) &= \frac{1}{C_aR_a}v_2(t) - \frac{1}{C_aR_a}v_1(t) + \frac{G}{C_a}v_1(t)\\
&\quad - \frac{1}{C_a}v_1(t)(a + 3b\phi(t)^2)\\
\dot{v}_2(t) &= \frac{1}{C_bR_a}v_1(t) - \frac{1}{C_bR_a}v_2(t) + \frac{1}{C_b}i(t)\\
\dot{i}(t) &= -\frac{1}{L}v_2(t) - \frac{R_b}{L}i(t). \qquad (3)
\end{aligned}$$

Rearranging (3)

$$\begin{aligned}
\dot{\phi}(t) &= v_1(t)\\
\dot{v}_1(t) &= \frac{1}{C_aR_a}v_2(t) - \left[\frac{1}{C_aR_a} - \frac{G}{C_a}v_1(t) + \frac{a}{C_a}\right]v_1(t)\\
&\quad - \frac{3b}{C_a}b\phi(t)^2v_1(t)
\end{aligned}$$

$$\dot{v}_2(t) = \frac{1}{C_b R_a} v_1(t) - \frac{1}{C_b R_a} v_2(t) + \frac{1}{C_b} i(t)$$
$$\dot{i}(t) = -\frac{1}{L} v_2(t) - \frac{R_b}{L} i(t). \quad (4)$$

Let

$$x(t) = \phi(t), \quad y(t) = v_1(t), \quad z(t) = v_2(t), \quad w(t) = i(t),$$
$$\alpha = \frac{1}{C_a R_a}, \quad \beta = \frac{1}{C_a R_a} - \frac{G}{C_a} v_1(t) + \frac{a}{C_a},$$
$$\varepsilon = \frac{3b}{C_a}, \quad \lambda = \frac{1}{C_b R_a}, \quad \mu = \frac{1}{C_b R_a}, \quad \sigma = \frac{1}{C_b},$$
$$\rho = \frac{1}{L}, \quad \tau = \frac{R_b}{L}.$$

Equation (4) becomes

$$\dot{x} = y$$
$$\dot{y} = \alpha z - \beta y - \varepsilon x^2 y$$
$$\dot{z} = \lambda y - \mu z + \sigma w$$
$$\dot{w} = -\rho z - \tau w. \quad (5)$$

The parameters of the above equation are chosen as follows for the system to exhibit chaos: $\alpha = 16.4$, $\beta = -3.28$, $\varepsilon = 19.68$, $\lambda = 1$, $\mu = 1$, $\sigma = 1$, $\rho = 15$, $\tau = 0.5$. Figure 1 shows the 3D state portraits of the fourth-order memristor system (5).

3. Fractional-order memristor system

The fractional-order operator is the generalisation of integer-order operator. There are three commonly used definitions of the fractional-order differential operator, viz. Grunwald–Letnikov, Riemann–Liouville and Caputo [35,36]. In this section, we shall study the dynamical behaviour of fractional-order system derived from system (5) with the Grunwald–Letnikov (GL) definition, which is defined as

$$_aD_t^q f(t) = \lim_{h\to 0} \left\{ \frac{1}{h^q} \sum_{j=0}^{\left[\frac{t-a}{h}\right]} (-1)^j \binom{q}{j} f(t - jh) \right\}$$
$$= \lim_{h\to 0} \left\{ \frac{1}{h^q} \Delta_h^q f(t) \right\}, \quad (6)$$

where a and t are limits of the fractional-order equation, $\Delta_h^q f(t)$ is the generalised difference, h is the step size and q is the fractional order of the differential equation. For numerical calculations, the above equation is modified as

$$_{(t-L)}D_t^q f(t) = \lim_{h\to 0} \left\{ h^{-q} \sum_{j=0}^{N(t)} b_j (f(t - jh) \right\}. \quad (7)$$

Theoretically, fractional-order differential equations use infinite memory. Hence, when we want to numerically calculate or simulate the fractional-order equations we have to use finite memory principle, where L is the memory length and h is the time sampling.

$$N(t) = \min \left\{ \left[\frac{t}{h}\right], \left[\frac{L}{h}\right] \right\}. \quad (8)$$

The binomial coefficients required for the numerical simulation is calculated as

$$b_j = \left(1 - \frac{a+q}{j}\right) b_{j-1}. \quad (9)$$

Applying these fractional-order approximations into the integer-order system (5) yields

$$D_t^{q_1} x = y$$
$$D_t^{q_2} y = \alpha z - \beta y - \varepsilon x^2 y$$
$$D_t^{q_3} z = \lambda y - \mu z + \sigma w$$
$$D_t^{q_4} w = -\rho z - \tau w. \quad (10)$$

The parameters of the above equation are chosen as follows for the system to exhibit chaos: $\alpha = 16.4$, $\beta = -3.28$, $\varepsilon = 19.68$, $\lambda = 1$, $\mu = 1$, $\sigma = 1$, $\rho = 15$, $\tau = 0.5$ and order $q_1 = q_2 = q_3 = q_4 = 0.95$. Figure 2 shows the 3D state portraits of the fractional-order system (10).

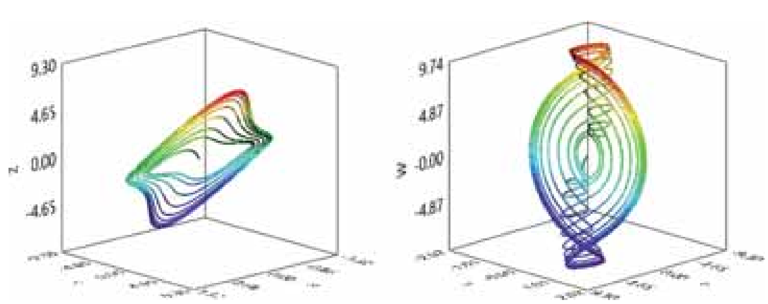

Figure 1. 3D state portraits of the fourth-order memristor system.

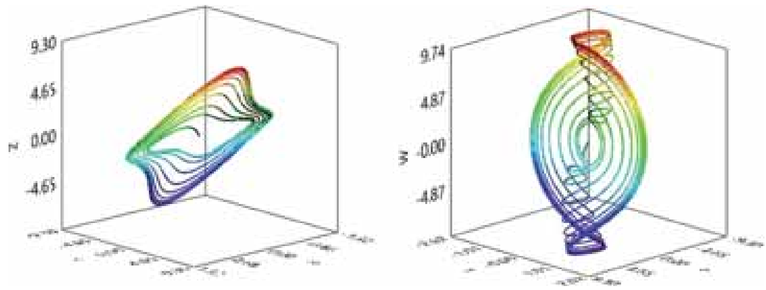

Figure 2. 3D state portraits of the fractional-order memristor system ($q = 0.95$).

4. Fractional-order chaotic maps

4.1 *Problem formulation and preliminaries*

In this section we derive the preliminaries for a fractional-order discrete time system. Let T be the sampling time of a fractional-order system of order q. Then the difference operation of a fractional-order system can be defined as [32,33],

$$\nabla^q x(k) = \sum_{j=0}^{k} (-1)^j \binom{q}{j} x(k-j) \tag{11}$$

when $q > 0$ in fractional order

$$\binom{q}{j} = \begin{cases} 1 & \text{for } j = 0 \\ \dfrac{q(q-1).....(q-j+1)}{j!} & \text{for } j > 0 \end{cases}. \tag{12}$$

Let us define a nonlinear discrete system as

$$x(k+1) = f(x(k)). \tag{13}$$

The qth order differentiation of (13) can be defined as

$$\nabla^q x(k+1) = f(x(k)) - x(k). \tag{14}$$

Comparing (11) and (14),

$$\nabla^q x(k+1) = x(k+1) - qx(k) + \sum_{j=2}^{k+1} (-1)^j \binom{q}{j} x(k-j+1). \tag{15}$$

Let us define $p = j - 1$, then (15) simplifies to

$$\nabla^q x(k+1) = x(k+1) - qx(k) + \sum_{p=1}^{k} (-1)^{p+1} \binom{q}{p+1} x(k-p). \tag{16}$$

Let

$$L_p = (-1)^p \binom{q}{p+1},$$

substituting (16) in (14),

$$x(k+1) = f(x(k)) + (q-1)x(k) + \sum_{p=1}^{k} L_p x(k-p). \tag{17}$$

Let us define a truncation length L for the fractional-order discrete system (17),

$$x(k+1) = f(x(k)) + (q-1)x(k) + \sum_{p=1}^{L} L_p x(k-p). \tag{18}$$

Let us apply the fractional-order discrete system approximations to the fractional-order memristor system (10).

4.2 *Fractional-order discrete memristor system*

In this section, we derive the fractional-order discrete model of the memristor system (10). To start with, let us first define the integer-order discrete model of the memristor model (5).

$$\begin{aligned} x(k+1) &= x(k) + (y(k))h \\ y(k+1) &= y(k) + \{\alpha z(k) - \beta y(k) - \varepsilon x^2(k) y(k)\}h \\ z(k+1) &= z(k) + \{\lambda y(k) - \mu z(k) + \sigma w(k)\}h \\ w(k+1) &= w(k) + \{-\rho z(k) - \tau w(k)\}h \end{aligned} \tag{19}$$

where $\alpha = 16.4$, $\beta = -3.28$, $\varepsilon = 19.68$, $\lambda = 1$, $\mu = 1$, $\sigma = 1$, $\rho = 15$, $\tau = 0.5$ and the step time $h = 0.005$. Applying (18) in (19) we get the fractional-order map

$$\begin{aligned} x(k+1) &= x(k) + (y(k))h + (q_1 - 1)x(k) \\ &\quad + \sum_{p=1}^{L} L_p x(k-p) \\ y(k+1) &= y(k) + \{\alpha z(k) - \beta y(k) - \varepsilon x^2(k) y(k)\}h \\ &\quad + (q_2 - 1)y(k) + \sum_{p=1}^{L} L_p y(k-p) \end{aligned}$$

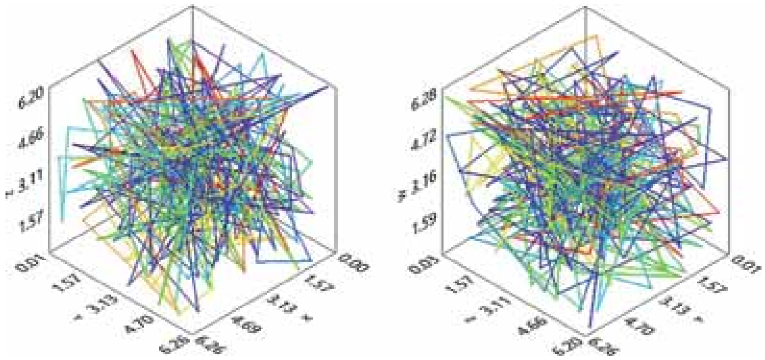

Figure 3. 3D state portraits of the fractional-order discrete memristor system ($q = 0.95$).

$$
\begin{aligned}
z(k+1) &= z(k) + \{\lambda y(k) - \mu z(k) + \sigma w(k)\}h \\
&\quad + (q_3 - 1)z(k) + \sum_{p=1}^{L} L_p z(k-p) \\
w(k+1) &= w(k) + \{-\rho z(k) - \tau w(k)\}h \\
&\quad + (q_4 - 1)w(k) + \sum_{p=1}^{L} L_p w(k-p), \quad (20)
\end{aligned}
$$

where $\alpha = 16.4, \beta = -3.28, \varepsilon = 19.68, \lambda = 1, \mu = 1, \sigma = 1, \rho = 15, \tau = 0.5$ and order $q_1 = q_2 = q_3 = q_4 = 0.95$. Figure 3 shows the 3D state portraits of the discrete fractional-order memristor system.

5. Dynamics of the fractional-order discrete memristor system

5.1 *Bifurcation*

By fixing $\alpha = 16.4, \beta = -3.28, \lambda = 1, \mu = 1, \sigma = 1, \rho = 15, \tau = 0.5$ and $q = 0.9$ and varying ε, the new system (20) is investigated. The bifurcation diagram is shown in figure 4a. By fixing $\varepsilon = 19.68, \beta = -3.28, \lambda = 1, \mu = 1, \sigma = 1, \rho = 15, \tau = 0.5$ and varying α the bifurcation is investigated and shown in figure 4b. By fixing $\varepsilon = 19.68, \alpha = 16.4, \lambda = 1, \mu = 1, \sigma = 1, \rho = 15, \tau = 0.5$ and varying β the bifurcation is investigated and shown in figure 4c. Generally speaking, when the system's biggest Lyapunov exponents is larger than zero, and the points in the corresponding bifurcation diagram are dense, the chaotic attractor will be found to exist in this system. Therefore, from the Lyapunov exponents and bifurcation diagrams in figures 4a, 4b and 4c, it can be concluded that chaos exist in the fractional-order memristor system when selecting a certain range of parameters.

Figure 5 shows the bifurcation of the discrete fractional-order memristor system with the initial conditions ($x(t) = 0.01, y(t) = 0.01, z(t) = 0.01, w(t) = 0.01$) changed in every iteration to the end values of the respective state variables and the local maximum of the state variables are plotted. It can be seen that the system takes period doubling route to chaos and shows chaotic oscillations for the commensurate fractional-order $q \geq 0.95$.

5.2 *Lyapunov exponents*

The initial conditions are randomly chosen but kept close to [0.01, 0.01, 0.01, 0.01] and the fractional order is taken as $q = 0.95$. The Lyapunov exponents of system (20) are 0.3417, 0, −0.0078, −7.9116.

The numerical results of the simulation are shown in figure 6.

6. Synchronisation of the fractional-order discrete memristor system

Synchronisation of chaotic systems with active control method [17,18], adaptive control method [19,20], extended back stepping control [21,22], sliding mode control [23,24], adaptive sliding mode [25,26], projective synchronisation [59], modified projective synchronisation [60], feedback control method [61], fractional-order synchronisation [62–64] and applying a new synchronisation scheme for image encryption [65] are studied in detail in the recent years. Most of these methods are more suited for chaotic flows and when considering chaotic maps with fractional orders, these synchronisation techniques seem to have a problem with varying stability of fractional orders and with incommensurate orders of the systems. Also, when dealing with real-time implementation of such synchronisation scheme using FPGS/DSP, we have a problem with limited memory availability. Hence, discrete forms of

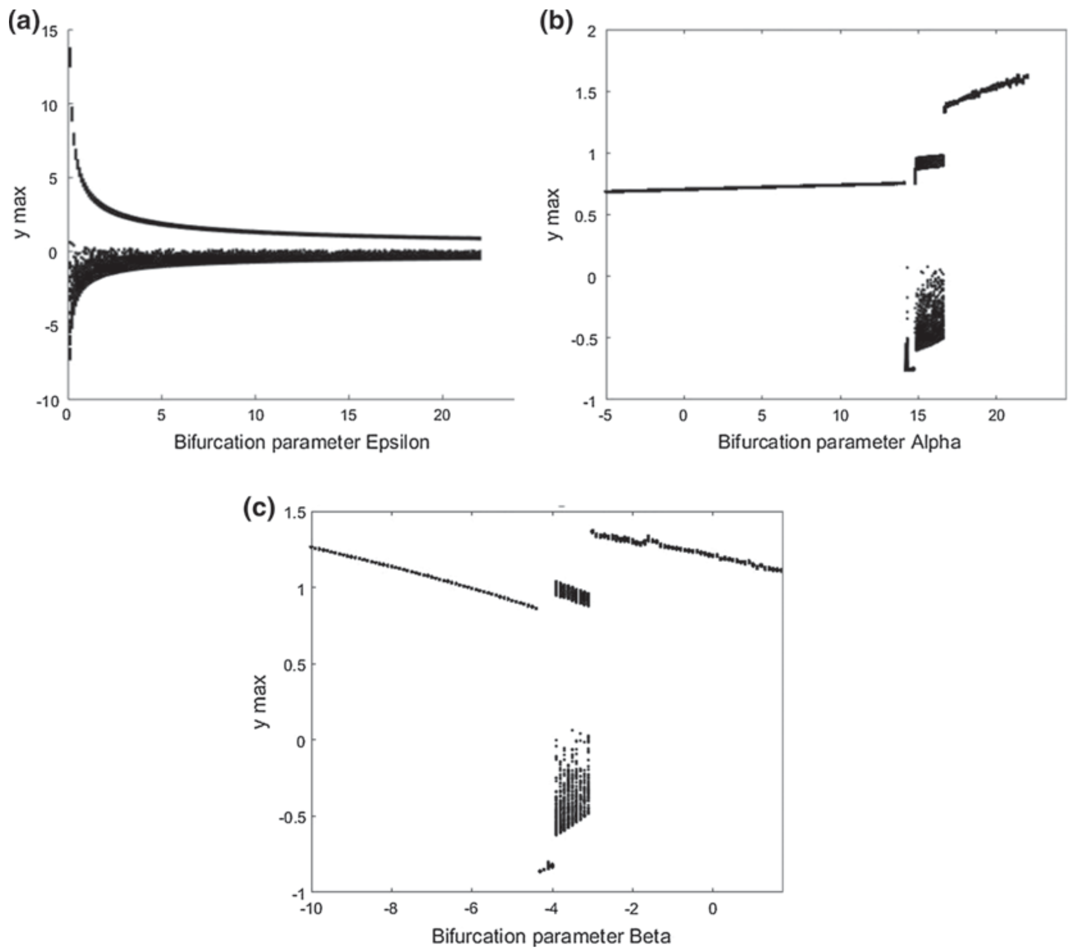

Figure 4. (**a**) Bifurcation plot vs. ε, (**b**) bifurcation plot vs. α and (**c**) bifurcation plot vs. β.

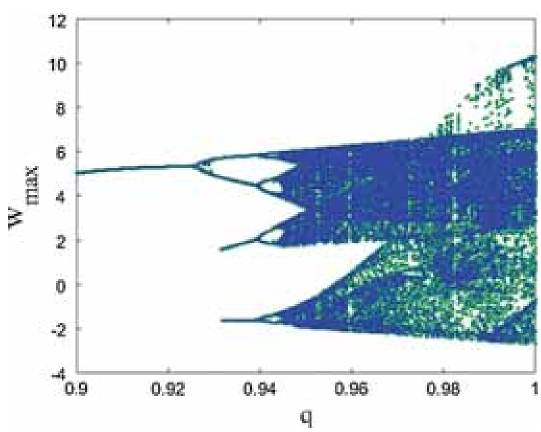

Figure 5. Bifurcation of the system with fractional-order q.

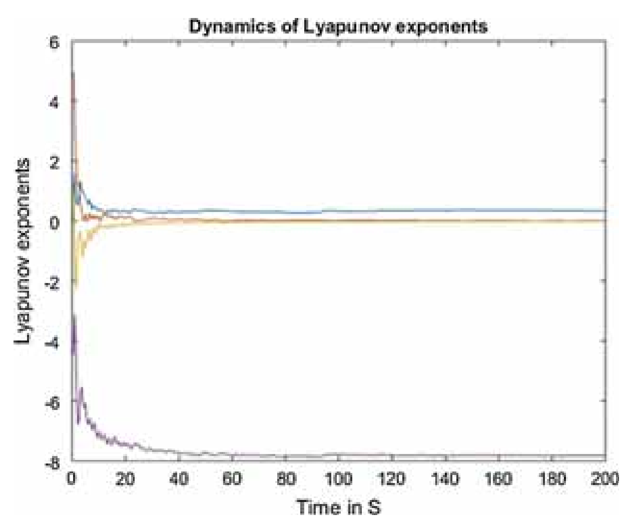

Figure 6. Lyapunov exponents of system (20).

fractional-order chaotic flows are considered for real-time scientific applications and fractional-order chaotic maps, and we have to derive a new technique which gives better performance for the synchronisation of the proposed systems.

In this section, we discuss about the synchronisation of the fractional-order discrete system (20). We analyse two cases of synchronisation. In Case 1 we discuss about the master and the slave systems with the same fractional order. In Case 2 we discuss about the master and the slave systems with different fractional orders.

6.1 *Commensurate-order systems*

In this case we consider the case where both the master and the slave systems have the same fractional order. Let us consider the master system as

$$x(k+1) = f(x(k)) + (q-1)x(k) + \sum_{p=1}^{L} L_p x(k-p) \quad (21)$$

and the slave system with the discrete controller as

$$y(k+1) = f(y(k)) + (q-1)y(k) + \sum_{p=1}^{L} L_p y(k-p) + u(k). \quad (22)$$

Let us define the synchronisation error as

$$e(k+1) = y(k+1) - x(k+1). \quad (23)$$

By simplifying (23) with (21) and (22),

$$\begin{aligned} e(k+1) &= f(y(k)) + (q-1)y(k) + \sum_{p=1}^{L} L_p y(k-p) - f(x(k)) - (q-1)x(k) - \sum_{p=1}^{L} L_p x(k-p) + u(k) \\ e(k+1) &= f(y(k)) - f(x(k)) + (q-1)[y(k)-x(k)] + \sum_{p=1}^{L} L_p[y(k-p) - x(k-p)] + u(k) \\ e(k+1) &= f(y(k)) - f(x(k)) + (q-1)[e(k)] + \sum_{p=1}^{L} L_p[e(k-p)] + u(k). \end{aligned} \quad (24)$$

Let us define the discrete controller to synchronise the same order systems as

$$u(k) = ke(k) - f(y(k)) + f(x(k)) - \sum_{p=1}^{L} L_p[e(k-p)]. \quad (25)$$

Substituting (25) in (24), we redefine the errors as

$$\begin{aligned} e(k+1) &= (q-1)e(k) + ke(k) \\ e(k+1) &= (q-1+k)e(k). \end{aligned} \quad (26)$$

If $k \in (q-2, q)$ and $(q-1+k) < 1$, the error (26) converges to zero.

The stability of the proposed synchronisation scheme is analysed as follows:

Let

$$x(k+1) = A_q x(k) + \sum_{p=1}^{L} L_p x(k-j). \quad (27)$$

The discrete system (27) is stable if $x(k+1) = A_q x(k)$ is stable, the term $\sum_{j=1}^{L} L_{xp} e(k-p)$ can be removed from the controller.

We apply the proposed synchronisation technique to the master and the slave fractional-order discrete memristor systems (20).

From (24), the synchronisation errors of the master and the slave fractional-order memristor systems can be defined as

$$\begin{aligned} e_x(k+1) &= e_x(k) + e_y(k)h + (q-1)e_x(k) + \sum_{p=1}^{L} L_p e_x(k-p) + u_x(k) \\ e_y(k+1) &= e_y(k) + [\alpha e_z(k) - \beta e_y(k) - \varepsilon(x_2^2(k)y_2(k) - x_1^2(k)y_1(k)]h \\ &\quad + (q-1)e_y(k) + \sum_{p=1}^{L} L_p e_y(k-p) + u_y(k) \\ e_z(k+1) &= e_z(k) + [\lambda e_y(k) - \mu e_z(k) + \sigma e_w(k)] \\ &\quad + (q-1)e_z(k) + \sum_{p=1}^{L} L_p e_z(k-p) + u_z(k) \\ e_w(k+1) &= e_w(k) - [\rho e_z(k) + \tau e_w(k)] \\ &\quad + (q-1)e_w(k) + \sum_{p=1}^{L} L_p e_x(k-p). \end{aligned} \quad (28)$$

To synchronise the systems let us define the controllers as

$$\begin{aligned} u_x(k) &= ke_x(k) - e_x(k) - e_y(k)h - \sum_{p=1}^{L} L_p e_x(k-p) \\ u_y(k) &= ke_y(k) - e_y(k) - [\alpha e_z(k) - \beta e_y(k) - \varepsilon(x_2^2(k)y_2(k) - x_1^2(k)y_1(k))]h - \sum_{p=1}^{L} L_p e_y(k-p) \\ u_z(k) &= ke_z(k) - e_z(k) - [\lambda e_y(k) - \mu e_z(k) \end{aligned}$$

$$+\sigma e_w(k)]h - \sum_{p=1}^{L} L_p e_z(k-p)$$

$$u_w(k) = ke_w(k) - e_w(k) - [\alpha e_z(k) - \beta e_y(k) - \varepsilon(x_2^2(k)y_2(k) - x_1^2(k)y_1(k))]h - \sum_{p=1}^{L} L_p e_y(k-p). \quad (29)$$

Substituting (29) in (28), the error dynamics simplifies to

$$e_x(k+1) = e_x(k)[q-1+k]$$
$$e_y(k+1) = e_y(k)[q-1+k]$$
$$e_z(k+1) = e_z(k)[q-1+k]$$
$$e_w(k+1) = e_w(k)[q-1+k]. \quad (30)$$

If $k \in (q-2, q)$ and $(q-1+k) < 1$, error (30) converges to zero.

For numerical simulations, the initial conditions of the master system are taken as $[1, 3, 2, -1]$ and the slave system as $[3, 1, 1, 5]$.

Figure 7 shows the synchronisation errors of the same order fractional discrete memristor system.

6.2 *Incommensurate order systems*

In this case, we assume that the orders of master and the slave systems are different. Let us define the master system as

$$x(k+1) = f(x(k)) + (q_1 - 1)x(k) + \sum_{p=1}^{L} L_p x(k-p) \quad (31)$$

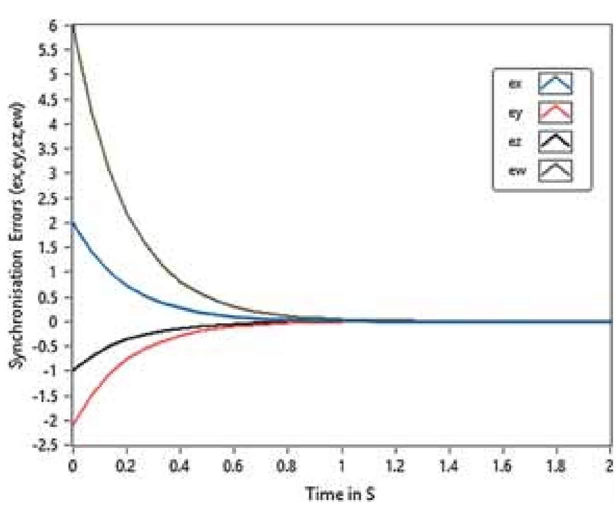

Figure 7. Synchronisation errors of the commensurate-order fractional discrete system.

and the slave system with the discrete controllers as

$$y(k+1) = f(y(k)) + (q_2 - 1)y(k) + \sum_{p=1}^{L} L_p y(k-p) + u(k), \quad (32)$$

where q_1 and q_2 are the fractional orders of the master and the slave systems respectively. Let us define the different order error dynamics as

$$e(k+1) = f(y(k)) - f(x(k)) + q_2 y(k) - q_1 x(k) - e(k) + \sum_{p=1}^{L} L_{p2} y(k-p) - \sum_{p=1}^{L} L_{p1} x(k-p) + u(k). \quad (33)$$

Let us define

$$q_2 y(k) = (q_2 - q_1)y(k) + q_1 y_1(k)$$

and

$$L_{p2} y(k-p) = (L_{p1} - L_{p2})y(k-p) + L_{p1} y(k-p).$$

Substituting the assumptions in (32) we simplify the error dynamics as

$$e(k+1) = f(y(k)) - f(x(k)) + (q_2 - q_1)y(k) - e(k) + q_1 y_1(k) + \sum_{p=1}^{L} (L_{p2} - L_{p1})y(k-p) - q_1 x(k) + \sum_{p=1}^{L} L_{p1} y(k-p) - \sum_{p=1}^{L} L_{p1} x(k-p) + u(k). \quad (34)$$

Rearranging (34),

$$e(k+1) = f(y(k)) - f(x(k)) + (q_2 - q_1)y(k) - (q_1 - 1)e(k) + \sum_{p=1}^{L} (L_{p2} - L_{p1})y(k-p) + \sum_{p=1}^{L} L_{p1} e(k-p) + u(k). \quad (35)$$

Let us define the controller $u(k)$ needed for synchronising the different fractional-order discrete systems as

$$u(k) = -f(y(k)) + f(x(k)) \\ -(q_1 - q_2)y(k) - ke(k) \\ -\sum_{p=1}^{L} L_{p1}e(k-p) \\ -\sum_{p=1}^{L} (L_{p2} - L_{p1})y(k-p). \quad (36)$$

By solving (35) with (36),

$$e(k+1) = (q_1 - 1 - k)e(k). \quad (37)$$

If $k \geq 0$ then the error dynamics $e(k+1)$ converges to zero.

Let

$$x(k+1) = Ax(k) + \sum_{p=1}^{L} L_p x(k-p). \quad (38)$$

System (37) is stable if $x(k+1) = Ax(k)$ is also stable and if $0 < q_1 < 1$ then, $\sum_{p=1}^{L} L_p e(k-p)$ can be deleted in the controller $u(k)$ as $e(k)$ approaches zero, $e(k-p)$ also approaches zero. Hence, controller (36) simplifies to

$$u(k) = -f(y(k)) + f(x(k)) - (q_2 - q_1)y(k) \\ + ke(k) - \sum_{p=1}^{L} (L_{p2} - L_{p1})y(k-p). \quad (39)$$

Applying the proposed different-order fractional discrete synchronisation to the master and the slave fractional-order systems (20), we can define the error dynamics as

$$e_x(k+1) = x_2(k) + y_2(k)h + (q_2 - 1)x_2(k) \\ + \sum_{p=1}^{L} L_p x_2(k-p) - x_1(k) \\ - y_1(k)h - (q_1 - 1)x_1(k) \\ - \sum_{p=1}^{L} L_{p1}x_1(k-p) + u_x(k)$$

$$e_y(k+1) = y_2(k) + [\alpha z_2(k) - \beta y_2(k) \\ - \varepsilon x_2^2(k)y_2(k)]h + (q_2 - 1)y_2(k) \\ + \sum_{p=1}^{L} L_{p2}y_2(k-p) - y_1(k) - [\alpha z_1(k) \\ - \beta y_1(k) - \varepsilon x_1^2(k)y_1(k)]h \\ - (q_1 - 1)y_1(k) - \sum_{p=1}^{L} L_{p1}y_1(k-p) \\ + u_y(k)$$

$$e_z(k+1) = z_2(k) + [\lambda y_2(k) - \mu z_2(k) \\ + \sigma w_2(k)]h + (q_2 - 1)z_2(k) \\ + \sum_{p=1}^{L} L_{p2}z_2(k-p) - z_1(k) \\ + [\lambda y_1(k) - \mu z_1(k) + \sigma w_1(k)]h \\ + (q_1 - 1)z_1(k) + \sum_{p=1}^{L} L_{p1}z_1(k-p) \\ + u_z(k)$$

$$e_w(k+1) = w_2(k) - [\rho z_2(k) + \tau w_2(k)]h \\ + (q_2 - 1)w_2(k) + \sum_{p=1}^{L} L_{p2}w_2(k-p) \\ - w_1(k) + [\rho z_1(k) + \tau w_1(k)]h \\ - (q_1 - 1)w_1(k) + \sum_{p=1}^{L} L_{p1}w_1(k-p) \\ + u_w(k). \quad (40)$$

6.3 *Simplifying individual error dynamics*

6.3.1 *Error $e_x(k+1)$.* Let us take the error dynamics of state x from (40)

$$e_x(k+1) = e_y(k)h + q_2 x_2(k) - q_1 x_1(k) \\ + \sum_{p=1}^{L} L_{p2}x_2(k-p) \\ - \sum_{p=1}^{L} L_{p1}x_1(k-p) + u_x(k). \quad (41)$$

Let us define

$$q_2 x_2(k) = (q_2 - q_1)x_2(k) + q_1 x_2(k) \\ L_{p2}x_2(k-p) = (L_{p2} - L_{p1})x_2(k-p) \\ + L_{p1}x_2(k-p). \quad (42)$$

Simplifying (41) with (42),

$$e_x(k+1) = e_y(k)h + (q_2 - q_1)x_2(k) + q_1 e_x(k) \\ + \sum_{p=1}^{L} (L_{p2} - L_{p1})x_2(k-p) \\ + \sum_{p=1}^{L} L_{p1}e_x(k-p) + u_x(k). \quad (43)$$

Let us define the controller for synchronising states x as

$$u_x(k) = -k_x e_x(k) - e_y(k)h - (q_2 - q_1)x_2(k) + \sum_{p=1}^{L}(L_{p2} - L_{p1})x_2(k-p) - \sum_{p=1}^{L} L_{p1}e_x(k-p) \quad (44)$$

by eq. (39) we can remove $\sum_{p=1} L_{p1}e_x(k-p)$ from (44),

$$u_x(k) = -k_x e_x(k) - e_y(k)h - (q_2 - q_1)x_2(k) - \sum_{p=1}^{L}(L_{p2}-L_{p1})x_2(k-p). \quad (45)$$

Simplifying (43) with (45),

$$e_x(k+1) = -k_x e_x(k) + q_1 e_x(k) = (-k_x e + q_1)e_x(k). \quad (46)$$

Error goes to zero if $k_x > q_1$.

6.3.2 *Error* $e_y(k+1)$. Let us take the error dynamics of state y from (40)

$$e_y(k+1) = e_y(k) + \left[\alpha e_z(k) - \beta e_y(k) - \varepsilon(x_2^2(k)y_2(k) - x_1^2(k)y_1(k)\right]h + q_2 y_2(k) - y_2(k) + \sum_{p=1}^{L} L_{p2}y_2(k-p) - y_1(k) - q_1 y_1(k) + y_1(k) - \sum_{p=1}^{L} L_{p1}y_1(k-p) + u_y(k). \quad (47)$$

Let us define

$$q_2 y_2(k) = (q_2 - q_1)y_2(k) + q_1 y_2(k)$$
$$L_{p2}y_2(k-p) = (L_{p2} - L_{p1})y_2(k-p) + L_{p1}y_2(k-p). \quad (48)$$

Simplifying (47) with (48),

$$e_y(k+1) = [\alpha e_z(k) - \beta e_y(k) - \varepsilon(x_2^2(k)y_2(k) - x_1^2(k)y_1(k))]h + (q_2 - q_1)y_2(k) + q_1 e_y(k) + \sum_{p=1}^{L}(L_{p2} - L_{p1})y_2(k-p) + \sum_{p=1}^{L} L_{p1}e_y(k-p). \quad (49)$$

Let us define the synchronisation controllers for state y as

$$u_y(k) = -k_y e_y(k) - [\alpha e_z(k) - \beta e_y(k) - \varepsilon(x_2^2(k)y_2(k) - x_1^2(k)y_1(k)]h - (q_2 - q_1)y_2(k) - \sum_{p=1}^{L}(L_{p2} - L_{p1})y_2(k-p). \quad (50)$$

Substituting (50) in (49),

$$e_y(k+1) = -k_y e_y(k) + q_1 e_y(k) = e_y(k)(-k_y + q_1) \quad (51)$$

which approaches zero if $k_y > q_1$.

6.3.3 *Error* $e_z(k+1)$. Let us take the error dynamics of state z from (40)

$$e_z(k+1) = e_z(k) + [\lambda e_y(k) - \mu e_z(k) + \sigma e_w(k)]h + q_2 z_2(k) - e_z(k) + \sum_{p=1}^{L} L_{p2}z_2(k-p) - q_1 z_1(k) - \sum_{p=1}^{L} L_{p1}z_1(k-p) + u_z(k). \quad (52)$$

Let us define

$$q_2 z_2(k) = (q_2 - q_1)z_2(k) + q_1 z_2(k)$$
$$L_{p2}z_2(k-p) = (L_{p2} - L_{p1})z_2(k-p) + L_{p1}z_2(k-p). \quad (53)$$

Simplifying (52) with (53)

$$e_z(k+1) = [\lambda e_y(k) - \mu e_z(k) + \sigma e_w(k)]h + (q_2 - q_1)z_2(k) + q_1 e_z(k) + \sum_{p=1}^{L}(L_{p2} - L_{p1})z_2(k-p) + \sum_{p=1}^{L} L_{p1}e_z(k-p) + u_z(k). \quad (54)$$

Let us define the synchronisation controllers as

$$u_z(k) = -[\lambda e_y(k) - \mu e_z(k) + \sigma e_w(k)]h + (q_2 - q_1)z_2(k) + \sum_{p=1}^{L}(L_{p2} - L_{p1})z_2(k-p) - \sum_{p=1}^{L} L_{p1}e_z(k-p). \quad (55)$$

By applying (38) in (55), term $\sum_{p=1}^{L} L_{p1}e_z(k-p)$ can be removed,

$$u_z(k) = -[\lambda e_y(k) - \mu e_z(k) + \sigma e_w(k)]h + (q_2 - q_1)z_2(k) + \sum_{p=1}^{L}(L_{p2} - L_{p1})z_2(k-p). \quad (56)$$

Simplifying (54) with (56),

$$e_z(k+1) = -k_z e_z(k) + q_1 e_z(k) = (-k_z + q_1)e_z(k). \quad (57)$$

Error dynamics (57) converges to zero if $k_z > q_1$.

6.3.4 *Error* $e_w(k+1)$. Let us take the error dynamics of state w from (40)

$$e_w(k+1) = e_w(k) - [\rho z_2(k) + \tau w_2(k)]h + q_2 w_2(k) - e_w(k) - q_1 w_1(k) + \sum_{p=1}^{L} L_{p2} w_2(k-p) - \sum_{p=1}^{L} L_{p1} w_1(k-p) + u_w(k). \quad (58)$$

Let us define

$$q_2 w_2(k) = (q_2 - q_1)w_2(k) + q_1 w_2(k)$$
$$L_{p2} w_2(k-p) = (L_{p2} - L_{p1})w_2(k-p) + L_{p1} w_2(k-p). \quad (59)$$

Solving (58) with (59),

$$e_w(k+1) = -[\rho e_z(k) + \tau e_w(k)]h + (q_2 - q_1)w_2(k) + q_1 e_w(k) + \sum_{p=1}^{L}(L_{p2} - L_{p1})w_2(k-p) + \sum_{p=1}^{L} L_{p1} e_w(k-p) + u_w(k). \quad (60)$$

Let us define the synchronisation controllers by applying (38) with (60) and removing term $\sum_{p=1}^{L} L_{p1}e_w \times (k-p)$ from the controller,

$$u_w(k) = -k_w e_w(k) - [\rho e_z(k) + \tau e_w(k)]h - (q_1 - q_2)w_2(k) - \sum_{p=1}^{L}(L_{p2} - L_{p1})w_2(k-p). \quad (61)$$

By simplifying (60) with (61),

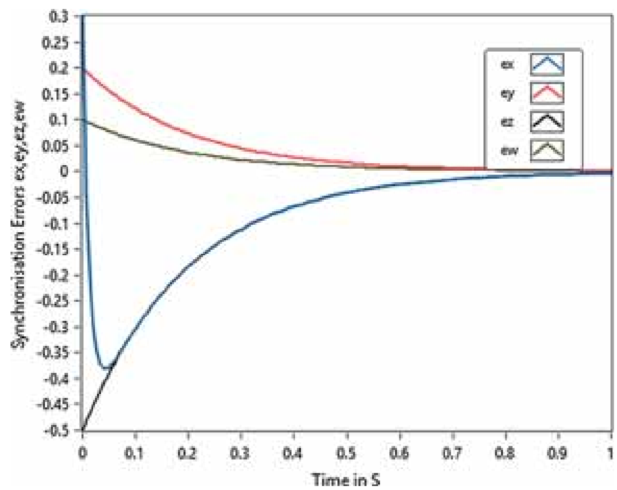

Figure 8. Synchronisation errors of the different-order fractional discrete memristor systems.

$$e_w(k+1) = -k_w e_w(k) + q_1 e_w(k) = (-k_w + q_1)e_w(k). \quad (62)$$

Error (62) converges to zero if $k_w > q_1$.

For numerical simulations the initial conditions of the master system are taken as [0.7, 0.8, 1, 0.9] and the slave system as [1, 1, 0.5, 1].

Figure 8 shows the synchronisation errors of the different-order fractional discrete memristor chaotic systems.

6.4 *FPGA implementation of the fractional-order memristor 4D systems*

In this section we discuss about the implementation of the proposed fractional-order systems in FPGA using the Xilinx (Vivado) System Generator toolbox in Simulink. First we configure the available built-in blocks of the system generator toolbox. The Add/Sub blocks are configured with zero latency and 32/16 bit fixed point settings. The output of the block is configured to rounded quantisation in order to reduce the bit latency. For the multiplier block, a latency of 3 is configured and the other settings are the same as in Add/Sub block. Next we shall have to design the fractional-order integrator which is not a readily available block in the System Generator. Hence, we implement the integrators using the mathematical relation discussed in (8) and (9) and the value of h is taken as 0.001 and the initial conditions are fed into the forward register. Figure 9 shows the Xilinx schematics of the fractional-order memristor system implemented in Kintex 7 (Device = 7k160t Package = fbg484 S) and figure 10 shows the Xilinx Kintex 7 schematics of the fractional-order integrator. Here we used a sampling period of 0.01 s and

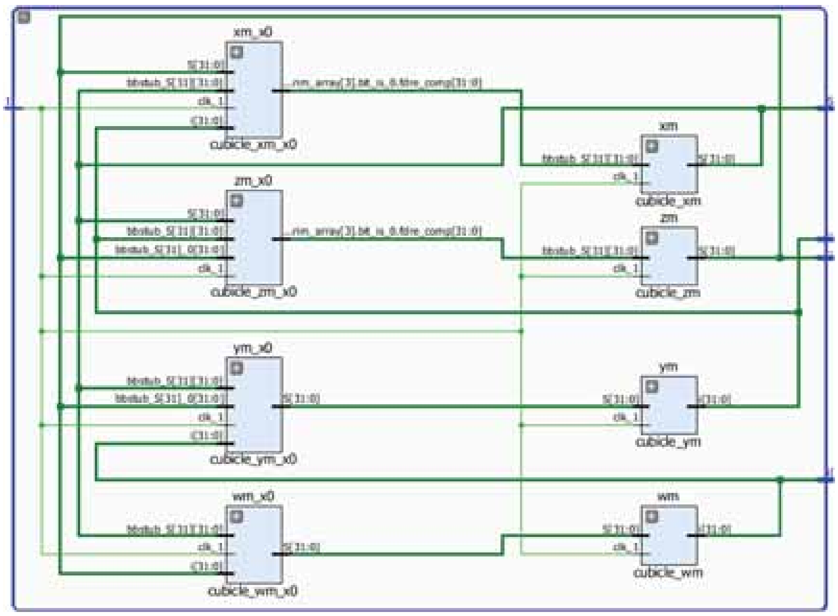

Figure 9. Xilinx schematics of the system implemeted in Kintex 7.

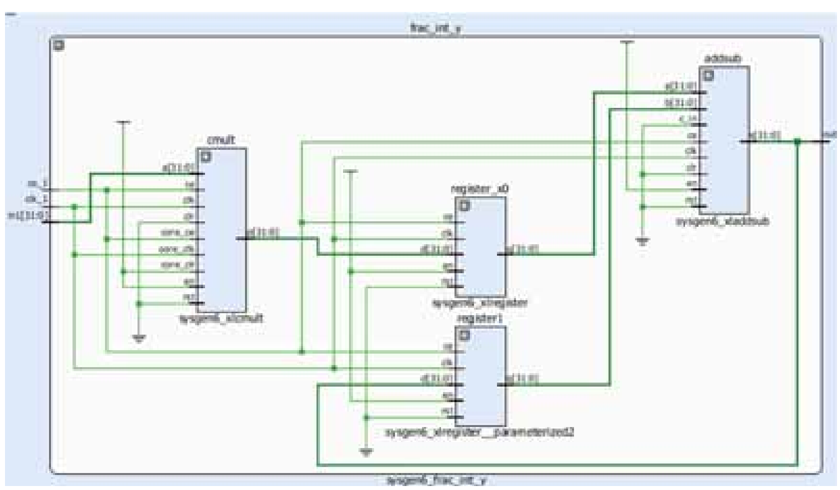

Figure 10. Xilinx schematics of the fractional-order integrator memory block implemeted in Kintex 7.

E14 as the clock input pin. Increasing the sampling time period in some implementations may lead to a clock frequency mismatch and hence plays a critical role in implementation. Also negative time slack may also create problems while implementing the design and hence choosing constraints may also be critical in cases where the number of logical operations are more. Also avoiding DDR clocks help in reducing the route delays.

7. Conclusions

In this paper, we investigated the discrete fractional-order model of a fourth-order memristor chaotic system. The discrete model is formed by transforming the differential version of the system using finite truncation method. The Lyapunov exponents and bifurcation analysis of the discrete fractional-order memristor system are investigated. Many synchronisation methods are available for fractional-order systems but the hardware realisation of such schemes are limited because of the poor memory efficiency of the algorithm. Hence, to solve this problem, we propose a novel methodology to synchronise the same order and different-order fractional discrete systems. The proposed scheme can be used in cryptography and other chaos-based security protocols because of the ease of implementation.

References

[1] T Geisel, *Nature* **298**, 322 (1982)
[2] O E Rössler, *Phys. Lett. A* **71**, 155 (1979)

[3] E N Lorentz, *J. Atmos. Sci.* **20**, 130 (1963)
[4] G Chen and T Ueta, *Int. J. Bifurc. Chaos* **9**, 1465 (1999)
[5] C X Liu, T Liu, L Liu and K Liu, *Chaos Solitons Fractals* **22**, 1031 (2004)
[6] V Sundarapandian and I Pehlivan, *Math. Comput. Model* **55**, 1904 (2012)
[7] V Sundarapandian, *J. Eng. Sci. Technol. Rev.* **6**, 45 (2013)
[8] V T Pham, C Volos, S Jafari, Z Wei and X Wang, *Int. J. Bifurc. Chaos* **24**, 1450073 (2014)
[9] V Sundarapandian and C Volos, *Arch. Control Sci.* **25**(**3**), 333 (2015)
[10] L O Chua*, IEEE Trans. Circuit Theory* **18**, 507 (1971)
[11] L Chua and S M Kang, *Proc. IEEE* **64**(**2**), 209 (1976)
[12] Z Biolek, D Biolek and V Biolková, *Radioengineering* **18**(**2**), 210 (2009)
[13] Robinson E Pino and Kristy A Campbell, *Proceeding of the International Symposium on Nanoscale Architecture* (2010) pp. 1–4
[14] Rak and G Cserey, *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **29**(**4**), 632 (2010)
[15] A Ishaq Ahamed and M Lakshmanan, *Int. J. Bifurc. Chaos* **23** (2013)
[16] Shibing Wang, X Wang, Y Zhou and B Han, *Entropy* **18**, 58 (2016), https://doi.org/10.3390/e18020058
[17] B A Idowu, U E Vincent and A N Njah, *Chaos Solitons Fractals* **39**, 2322 (2009)
[18] S Vaidyanathan and K Rajagopal, *Int. J. Syst. Signal Control Eng. Appl.* **4**, 55 (2011)
[19] V Sundarapandian and R Karthikeyan, *Eur. J. Sci. Res.* **64**, 94 (2011)
[20] V Sundarapandian and R Karthikeyan, *J. Eng. Appl. Sci.* **7**, 45 (2012)
[21] Sajjad Shoja Majidabad and Heydar Toosian Shandiz, *J. Control Syst. Eng.* **1**(**1**), 1 (2013)
[22] A N Njah, *Nonlinear Dyn.* **61**(**1–2**), 1 (2010)
[23] O S Onma, O I Olusola and A N Njah, *Nonlinear Dyn.* Article ID 861727, 15 pages, http://dx.doi.org/10.1155/2014/861727 (2014)
[24] B Wang, Y Li and D L Zhu, *Int. J. Comput. Appl.* **8**(**8**), 425 (2015)
[25] Chun Yin, Sara Dadras, Shou-ming Zhong and Yang Quan Chen, *Appl. Math. Model.* **37**(**4**), 2469 (2013)
[26] Haorui Liu and Juan Yang, *Entropy* **17**, 4202 (2015)
[27] S Wang, Y Yu and M Diao, *Physica A* **389**, 4981 (2010)
[28] Z Wanga, X Huang and H Shend, *Neurocomputing* **83**, 83 (2012)
[29] Shaojuan Ma, Jie Zheng and Yuqin Li, *J. Inf. Comput. Sci.* **11**(**10**), 3469 (2014)
[30] R H Li and W S Chen, *Chin. Phys. B* **22**, 040503 (2013)
[31] I Petras, *Chaos Solitons Fractals* **38**, 140 (2008)
[32] A M A El-Sayed, Z F El-Raheem and S M Salman, *Adv. Differ. Equ.* Article ID: 2014-66 (2014)
[33] A M A El-Sayed and S M Salman, *J. Frac. Calc. Appl.* **4**(**2**), 251 (2013)
[34] D Baleanu, K Diethelm, E Scalas and J J Trujillo, *Fractional calculus: Models and numerical methods* (World Scientific, Singapore, 2014)
[35] Y Zhou, *Basic theory of fractional differential equations* (World Scientific, Singapore, 2014)
[36] K Diethelm, *The analysis of fractional differential equations* (Springer, Berlin, 2010)
[37] W Trzaska Zdzislaw, *Matlab solutions of chaotic fractional-order circuits* (Intech Open, 2014)
[38] B Bao, P Jiang, H Wu and F Hu, *Nonlinear Dyn.* **79**, 2333 (2015)
[39] K Rajagopal, S Vaidhyanathan, A Karthikeyan and P Duraisamy, *Electr. Eng.* **99**(**2**), 721 (2017)
[40] H K Khalil, *Nonlinear systems* (Prentice Hall, New York, 2002)
[41] K Rajagopal, L Guessas, A Karthikeyan, A Srinivasan and G Adam, *Complexity* https://www.hindawi.com/journals/complexity/2017/1892618/ (2017)
[42] K Rajagopal, A Karthikeyan and A K Srinivasan, *Nonlinear Dyn.* **87**(**4**), 2281 (2017)
[43] K Rajagopal, A Karthikeyan and A Srinivasan, *Chaos Solitons Fractals* **103**, 347 (2017)
[44] K Rajagopal, S Jafari and G Laarem, *Pramana – J. Phys.* (accepted)
[45] C Wang, R Chu and J Ma, *Complexity* **21**, 370 (2015)
[46] Y Jin, Y Q Chen and D Xue, *IET Control Theory Appl.* **5**, 164 (2011)
[47] H Zhang, X Y Wang and X H Lin, *Asian J. Control* **19**, 106 (2017)
[48] I Petráš, *Fractional-order nonlinear systems: Modeling, analysis and simulation* (Springer, 2011)
[49] U N Katugampola, *Bull. Math. Anal. Appl.* **6**, 1 (2014)
[50] M A Herzallah, *J. Fract. Calc. Appl.* **5**, 1 (2014)
[51] Y Li, Y Chen and I Podlubny, *Comput. Math. Appl.* **59**, 1810 (2010)
[52] J A Gallegos and M A Duarte-Mermoud, *Appl. Math. Comput.* **287**, 161 (2016)
[53] M S Tavazoei and M Haeri, *Phys. Lett. A* **372**, 798 (2008)
[54] K Konishi, H Kokame and N Hara, *Nonlinear Dyn.* **63**, 513 (2011)
[55] D Chen, P Shi and X Ma, *Int. J. Innov. Comput. Inf.* **8**, 7237 (2012)
[56] Jun Ma, Fuqiang Wu, Guodong Ren and Jun Tang, *Phys. Lett. A* **298**, 65 (2017)
[57] Y Xu, H Ying, Y Jia, J Ma and T Hayat, *Sci. Rep.* **7**, 43452 (2017)
[58] Fuqiang Wu, Chunni Wang, Wuyin Jin and Jun Ma, *Physica A* **469**, 81 (2017)
[59] X Wang and Y He, *Phys. Lett. A* **372**(**4**), 435 (2008)
[60] X Wang, X Zhang and C Ma, *Nonlinear Dyn.* **69**(**1–2**), 511 (2012)
[61] X Wang, Y He and M Wang, *Nonlinear Anal. Theory Methods Appl.* **71**(**12**), 6126 (2009)
[62] C Luo and X Wang, *Nonlinear Dyn.* **71**(**1–2**), 241 (2013)
[63] X Wang and J M Song, *Commun. Nonlinear Sci. Numer. Simul.* **14**(**8**), 3351 (2009)
[64] X Wang and M Wang, *Chaos* **17**, 033106 (2007)
[65] Y Zhanga and X Y Wang, *Inf. Sci.* **273**, 329 (2014)