Nonlinear distributed-order models: Adaptive synchronization, image encryption and circuit implementation

Tarek M. Abed-Elhameed[a,*], Gamal M. Mahmoud[a], Motaz M. Elbadry[b,c], Mansour E. Ahmed[a]

[a] *Department of Mathematics, Faculty of Science, Assiut University, Assiut 71516, Egypt*
[b] *Department of Electrical Engineering, Faculty of Engineering, Assiut University, Assiut 71516, Egypt*
[c] *Department of Electrical and Computer Engineering, Faculty of Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 1H9*

Abstract

The main aim of present work is to investigate the dynamics of the chaotic nonlinear distributed order Lü model (DOLM). The distributed order (DO) derivative is used for describing the viscoelasticity of various technical models and materials. The modified spectral numerical method is used to evaluate the numerical solutions for DOLM. Using nonlinear feedback control and the Lyapunov direct approach, the adaptive synchronization of two chaotic distributed order models (DOMs) is presented. We state a theorem to drive analytical controllers which are used to achieve our synchronization. The DOLM is introduced as an example of DOMs to verify the validity of our analytical results. Numerical computations are displayed to show the agreement between both analytical and numerical results. The DOMs appear in many applications in engineering and physics, e.g., image encryption and electronic circuits (ECs). Based on our proposed synchronization, the encryption and decryption of color images are studied. Information entropy, visual analysis and histograms are calculated, together with the experimental results of image encryption and decryption. We design the EC of the DOLM using the Multisim circuit simulator for the first time to our knowledge. Using electronic circuit simulation, we achieved the same results for the numerical treatment of our synchronization. Other ECs can be similarly designed for other DOMs. **Keywords:** *Distributed order; Adaptive synchronization; Nonlinear feedback control; Image encryption; Lyapunov direct method; Circuit implementation*

1. Introduction

Distributed order fractional calculus was first proposed by Caputo in 1969 [1]. The DO derivative was considered as a generalization of integer- and fractional-order derivatives. The DO calculus was developed by Caputo and other researchers [2, 3, 4, 5, 6, 7]. An asymptotic stability theorem for DO time-varying models

*Corresponding author
Email addresses: tarekmalsbagh@aun.edu.eg (Tarek M. Abed-Elhameed), gmahmoud@aun.edu.eg, gmahmoud_56@yahoo.com (Gamal M. Mahmoud), elbadrymotaz@aun.edu.eg, mmelbadr@ualberta.ca (Motaz M. Elbadry), m_ahmed@aun.edu.eg, me78ahmed@yahoo.com (Mansour E. Ahmed)

was presented in [4]. The generalized Wright stability of DO dynamical models was stated by Mahmoud *et al.* [5]. A controller design that ensures the DOM's solution converges at a predefined time was proposed [6]. Muñoz-Vázquez *et al.* [7] examined how a class of distributed-order models can be stabilized robustly. There are numerous applications for DO differential equations in physics, engineering, and neural networks [8, 9, 10, 11]. The DO Lorenz model was employed by the authors in [9] for secure communication. Hyperchaotic masking for a text was proposed and studied by DO hyperchaotic van der Pol oscillators [10]. Mahmoud *et al.* presented DO neural networks [11] and Zang *et al.* [12] investigated a technique for image encryption based on DOCM. Due to its potential uses in signal processing, secure communication, etc., the chaos synchronization problem has received a lot of attention. Because it has strong robustness, adaptive control is appealing in several controllers [13, 14, 15]. The adaptive control model offers numerous benefits, including the need for continuous model information extraction during model operation and the ability to adapt to varied update laws. The input and output data of the item can be used to continuously identify the model parameters, which increases the model's accuracy, and it can be applied in many different fields: secure communication, biological technology, image process and information science [16, 17, 18, 19]. On the other hand, chaos synchronization in DO nonlinear dynamical models is a hot topic. The synchronization for chaotic DO Lorenz model was achieved by active control [9]. Based on linear feedback control, the synchronization for chaotic complex DO Lü and Chen was presented [5]. Using direct Lyapunov function and nonlinear feedback control, the synchronization for hyperchaotic complex DO van der Pol was investigated [10]. Chaos synchronization is used in many schemes of image encryption which have been presented with different properties in many applications (e.g. see [20, 21, 22, 23] and references therein).

The modelling and applications in ECs of chaotic dynamical models have received a lot of research attention in recent years. Many researchers designed ECs to simulate the integer-order dynamical models [24, 25, 26]. Wang *et al.* [24] proposed the EC of a four-wing memristive chaotic model. Based on self-directed channel memristors, a fully autonomous chaotic oscillator circuit was realized [25]. By using numerical simulations and circuit implementation, the basic properties of a no-equilibrium chaotic model were studied [26]. On the other hand, the fractional-order chaotic and hyperchaotic models were used to construct ECs [27, 28, 29].

In this paper, we introduce the chaotic nonlinear distributed order Lü model (DOLM) as:

$$\begin{aligned} D^{\psi(\beta)}x_1 =& a(x_2 - x_1),\\ D^{\psi(\beta)}x_2 =& cx_2 - x_1x_3,\\ D^{\psi(\beta)}x_3 =& -bx_3 + x_1x_2, \end{aligned} \tag{1.1}$$

where a, b, c are constant parameters and $D^{\psi(\beta)}$ is the DO derivative [30].

The nonlinear feedback control and the Lyapunov direct approach are applied to investigate the adaptive synchronization of two DOMs. We study numerically the behavoir of the DOLM through the modified spectral numerical method [5]. A theorem to achieve our synchronization is presented and proofed. The model (1.1) is

used as an example to achieve this kind of synchronization. We note that, the numerical simulation shows an agreement between analytical and numerical results. Using the adaptive synchronization, the encryption and decryption of color Peppers image are investigated. We calculate the information entropy, visual analysis and histograms for this image. A circuit diagram is designed for DOLM (1.1) for the first time by Multisim circuit simulator. The electronic circuit results of the adaptive synchronization of the DOLM (1.1) have an agreement with numerical calculations.

The rest of this paper is given as. In section 2, a few theorems, definitions and lemmas are utilised in this work. The basic properties and dynamics of the DOLM (1.1) are investigated in section 3. In section 4, we propose a theorem to achieve the adaptive synchronization between two DOMs. We derive the analytical controllers and the estimated parameters updating law which used to achieve the proposed synchronization. In order to validate our suggested theorem, the numerical simulation is shown. Section 5 contains an application of adaptive synchronization in color image encryption. The information entropy, visual analysis and histograms for color Peppers image are given. Section 6 contains the EC which created for DOLM (1.1). We present the conclusion of our work in section 7.

2. Basic concepts

We stated a few lemmas, theorems and definitions of DO derivative [1, 4, 5, 30], in this section.

Definition 2.1. *[1] The Caputo FO derivative of a continuous function $y(t)$ is:*

$$ {}^{C}D^{\beta}y(t) = \frac{1}{\Gamma(\gamma-\beta)}\int_0^t \frac{y^{(\gamma)}(v)}{(t-v)^{\beta-\gamma+1}}dv, \tag{2.1}$$

where $\gamma \in \mathbb{N}^{+}, (\gamma-1<\beta<\gamma)$.

Definition 2.2. *[30] The DO derivative of $y(t)$ is:*

$$ D^{\psi(\beta)}y(t) = \int_{\gamma-1}^{\gamma} \psi(\beta)\,{}^{C}D^{\beta}y(t)d\beta, \tag{2.2}$$

where $\gamma-1=\tau_0<\tau_1<...\tau_m=\gamma, \Delta\tau_j=\tau_j-\tau_{j-1}=\frac{1}{m}$, $\gamma-1<\beta<\gamma$, $\beta_j=\frac{\tau_j+\tau_{j-1}}{2}=\frac{2j-1}{2m}, j=1,2,...,m, m\in\mathbb{N}$.

We choose the grid $0=\tau_0<\tau_1<...<\tau_m=1$, to discretize the integral interval $(0,1]$.
and take $\Delta\tau_j=\tau_j-\tau_{j-1}=\frac{1}{m}$, $\beta_j=\frac{\tau_j+\tau_{j-1}}{2}=\frac{2j-1}{2m}$, $j=1,2,...,m$, $m\in\mathbb{N}$. Using the mid-point quadrature method, one has:

$$ D^{\psi(\beta)}x(t) = \int_0^1 \psi(\beta)\,{}^{C}D^{\beta}x(t)d\beta \approx \sum_{j=1}^{m}\psi(\beta_j)\,{}^{C}D^{\beta_j}x(t)\Delta\tau_j. \tag{2.3}$$

We expressed the DOM as a multi term FO models as:

$$ \sum_{j=1}^{m}\psi(\beta_j)\,{}^{C}D^{\beta_j}x(x)\Delta\tau_j = F(x(t))A + f(x(t)), \tag{2.4}$$

where $x \in \mathbb{R}^n$.

Lemma 2.1. *[4] Suppose that a Lyapunov function $V(t, x(t))$ exists that is sufficient*

$$\gamma_1 \|x\|^a \leq V(t, x(t)) \leq \gamma_2 \|x\|^{ab}, \tag{2.5}$$

$$D^{\psi(\beta)} V(t, x(t)) \leq -\gamma_3 \|x\|^{ab}, \tag{2.6}$$

where $\beta \in (0, 1]$, and $a, b, \gamma_i > 0$, $i = 1, 2, 3$. Therefore $\bar{x} = 0$ is Mittag-Leffler stable, where $\bar{x} = 0$ is a fixed point for DOM (2.4). The $\bar{x} = 0$ is globally Mittag-Leffler stable, if the assumptions hold globally on $\mathbb{R}^n$.

Theorem 2.1. *[5] The zero solution of (2.4) is asymptotically stable, if*

1. $\lim\limits_{\|x(t)\| \to 0} \frac{\|f(x(t))\|}{\|x(t)\|} = 0$;
2. $|arg\lambda_i(-\frac{\theta_{k-1}}{\theta_k})| > \frac{\pi \mu_k}{2}$; k=1, 2, ...,m, i=1,2, ...,n,

where $\mu_m = \beta_m - \beta_{m-1}$ and $(m + 1)$ is the number of steps for $\beta \in (0, 1]$, $\theta_k \in \mathbb{R}^{n \times n}$, $\theta_0 = -A$, $\theta_k = I\Delta\tau_k \psi(\beta_k)$, and $\lambda_i(\frac{\theta_{k-1}}{\theta_k})$ is the eigenvalues of the matrix $\frac{\theta_{k-1}}{\theta_k}$.

Lemma 2.2. *[4] Suppose $y(t) \in R^n$ is differentiable function, then*

$$\frac{1}{2} D^{\psi(\beta)} y^T(t) y(t) \leq y^T(t) D^{\psi(\beta)} y(t). \tag{2.7}$$

3. Dynamics of chaotic distributed order Lü model (1.1)

In this section we study the dynamics of the DOLM (1.1).

3.1. Properties and stability for model (1.1)

By solving model (1.1), the fixed points are: $E_0 = (0, 0, 0, 0)^T$, $E_1 = (\sqrt{bc}, \sqrt{bc}, c)^T$ and $E_2 = (-\sqrt{bc}, -\sqrt{bc}, c)^T$. Using Theorem 2.1, we study the stability of E_0 of model (1.1). The first part of Theorem 2.1 is tested as:

$$\begin{aligned} \lim_{\|x(t)\| \to 0} \frac{\|f(x(t))\|}{\|x(t)\|} &= \lim_{\|x(t)\| \to 0} \frac{\sqrt{(x_1 x_3)^2 + (x_1 x_2)^2}}{\sqrt{x_1^2 + x_2^2 + x_3^2}} \\ &= \lim_{\|x(t)\| \to 0} \frac{\sqrt{x_1^2 (x_2^2 + x_3^2)}}{\sqrt{x_1^2 + x_2^2 + x_3^2}} \\ &\leq \lim_{\|x(t)\| \to 0} \|x(t)\| = 0. \end{aligned} \tag{3.1}$$

For the choice $m = 50$, $\psi(\beta) = \frac{\beta^2}{10^3 (1-\beta)^2}$, $k = 1$ and $\Delta\tau_m = \frac{1}{m}$, we can test the second part in Theorem 2.1. Then $\frac{\theta_0}{\theta_1} = 4 \times 10^9 \begin{pmatrix} -a & a & 0 \\ 0 & c & 0 \\ 0 & 0 & -b \end{pmatrix}$,
then, the corresponding eigenvalues can be written as: $\lambda_1 = -a, \lambda_2 = -b$ and $\lambda_3 = c$.

If we choose the parameters a, b and c such that $|\arg(\lambda_i(-\frac{\theta_0}{\theta_1}))| > \frac{\pi}{100}$, $i = 1, 2, 3$, the zero solution of model (1.1) is stable. For $k = 2, 3, ..., m$, the pervious condition is given as:

$$|\arg(\lambda_i(-\frac{\theta_{k-1}}{\theta_k}))| = \pi > \frac{\mu_m \pi}{2} = \frac{\pi}{100}, \qquad i = 1, 2, 3; \tag{3.2}$$

Therefore the zero solution of model (1.1) is asymptotically stable.

4. Adaptive synchronization for DOMs with uncertain parameters

To achieve adaptive synchronization between two DOMs, we use the Lyapunov direct method and nonlinear feedback control method

We consider the drive DOMs as:

$$D^{\psi(\beta)}x = F(x(t))A + f(x(t)), \tag{4.1}$$

where $x \in \mathbf{R^n}$, the matrix $F \in \mathbf{R^{n\times p}}$, the vector $A \in \mathbf{R^p}$, and $f(x) \in \mathbf{R^n}$. Let the response DOM is:

$$D^{\psi(\beta)}y = F(y(t))\hat{A} + f(y(t)) + u(t), \tag{4.2}$$

where $y \in \mathbf{R^n}$, $\hat{A}$ is the estimate vector of A and $u(t) \in R^n$ is a suitable controller which will be given later.

Definition 4.1. *The drive model* (4.1) *is adaptive synchronization with the response model* (4.2)*, if*

$$\lim_{t\to\infty} \|e\| = \lim_{t\to\infty} \|y(t) - x(t)\| = 0. \tag{4.3}$$

From models (4.1) and (4.2), we can write the model of errors as:

$$D^{\psi(\beta)}e = (F(y) - F(x))A + f(y) - f(x) + F(y)e_A + u(t), \tag{4.4}$$

where $e_A = \hat{A} - A$.

Remark 4.1. *If the distribution function is chosen as $\psi(\beta) = \delta(\beta - \beta)$, then the fractional results with order β that inspired this work are recovered, while one can get the adaptive synchronization between integer models if $\beta = 1$, where $\delta(.)$ is the Dirac delta function.*

Theorem 4.1. *The response model* (4.2) *is synchronized to the drive model* (4.1) *as adaptive synchronization if:*

$$u(t) = (F(x) - F(y))A + f(x) - f(y) - Ke, \tag{4.5}$$

$$D^{\psi(\beta)}e_A = D^{\psi(\beta)}\hat{A} = -(F(y))^T e, \tag{4.6}$$

where $u(t)$ is the controller, Eq. (4.6) *gives the estimated parameters updating law, and $K \in R^{n\times n}$ is the gain matrix.*

Proof. Suppose the Lyapunov direct function as $V(e(t)) = \frac{1}{2}e_A^T(t)e_A(t) + \frac{1}{2}e^T(t)e(t)$, then

$$D^{\psi(\beta)}V(e(t)) = D^{\psi(\beta)}\{\frac{1}{2}e_A^T(t)e_A(t) + \frac{1}{2}e^T(t)e(t)\}. \tag{4.7}$$

Using lemma 2.2 and Eqs. (4.4-4.6), we have

$$\begin{aligned} D^{\psi(\beta)}V(e(t)) &\leq e_A^T(t)D^{\psi(\beta)}e_A(t) + e^T(t)D^{\psi(\beta)}e(t) \\ &= e_A^T(t)(-F^T(y)e(t)) + e^T(t)(F(y)e_A(t) - Ke(t)) \\ &= -e^T(t)Ke(t) \\ &\leq -\mu_{min}V(e(t)). \end{aligned} \tag{4.8}$$

According to Lemma 2.1, therefore $\lim_{t\to\infty}\|e(t)\| = 0$, where μ_{min} is the smallest value of the eigenvalues of K. □

4.1. Numerical simulations

In this subsection, we will give the DOLM as an example of DOMs (4.1) to confirm the validity of the theorem's analytical results 4.1.

$$\begin{aligned} D^{\psi(\beta)}x_1 =& a(x_2 - x_1), \\ D^{\psi(\beta)}x_2 =& cx_2 - x_1x_3, \\ D^{\psi(\beta)}x_3 =& -bx_3 + x_1x_2. \end{aligned} \tag{4.9}$$

We use Matlab software to evaluate the Lyapunov exponents for the model (4.9), we choose $\psi(\beta) = \frac{\beta^2}{10^3(1-\beta)^2}$, $a = 36$, $b = 3$, $c = 20$ and the initial values $(5.706, 6.3379, 17.0355)^T$. These Lyapunov exponents are $\lambda_1 = 8.6868$, $\lambda_2 = -12.1279$, $\lambda_3 = -69.1089$. So that this model has chaotic behaviour as depicted in Fig. 1.

Model (4.9) can be written in form (4.1), where $A = (a, b, c)^T$, $f(x) = (0, -x_1x_3, x_1x_2)^T$ and $F(x) = \begin{pmatrix} (x_2 - x_1) & 0 & 0 \\ 0 & 0 & x_2 \\ 0 & -x_3 & 0 \end{pmatrix}$.

We consider model (4.9) is the drive model and the response model can be written in the form (4.2) as:

$$\begin{aligned} D^{\psi(\beta)}y_1 =& \hat{a}(y_2 - y_1) + u_1, \\ D^{\psi(\beta)}y_2 =& \hat{c}y_2 - y_1y_3 + u_2, \\ D^{\psi(\beta)}y_3 =& -\hat{b}y_3 + y_1y_2 + u_3, \end{aligned} \tag{4.10}$$

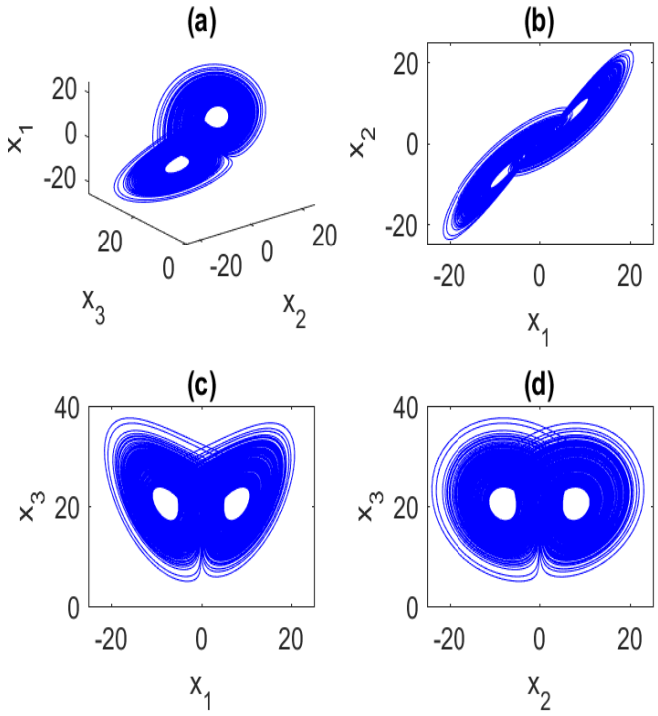

Figure 1: Chaotic solution of model (4.9) for $a = 36$, $b = 3$ and $c = 20$: (a) x_2, x_3, x_1 space, (b) x_1, x_2 space, (c) x_1, x_3 space, (d) x_2, x_3 space.

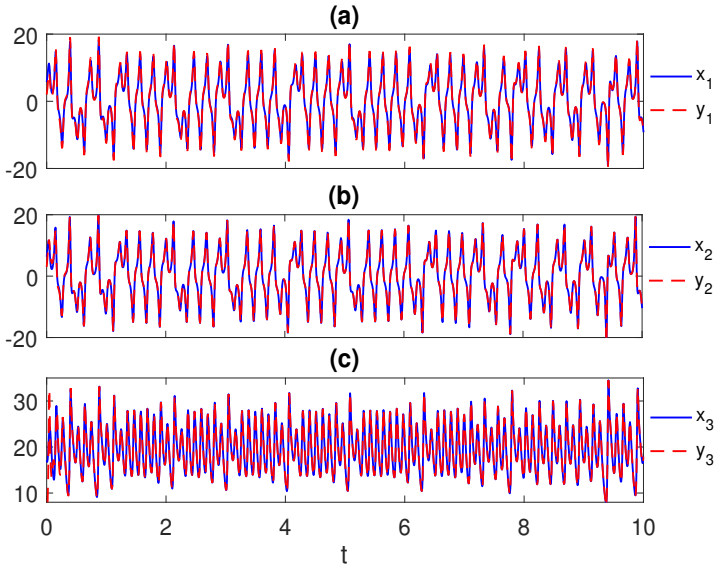

Figure 2: The state variables of adaptive synchronization of the drive model (4.9) and the response model (4.10): (a) x_1 and y_1 vs t, (b) x_2 and y_2 vs t, (c) x_3 and y_3 vs t.

where $\hat{A} = (\hat{a}, \hat{a}, \hat{c})^T$ and $u = (u_1, u_2, u_3)^T$.

The response model (4.10) is synchronized to the drive model (4.9) by theorem 4.1, if $u(t)$ takes:

$$u(t) = \begin{pmatrix} a(x_2 - x_1 - y_2 + y_1) - k_1 e_1 \\ c(x_2 - y_2) - x_1 x_3 + y_1 y_3 - k_2 e_2 \\ b(-x_3 + y_3) + x_1 x_2 - y_1 y_2 - k_3 e_3 \end{pmatrix}, \tag{4.11}$$

where $K = diag(k_1, k_2, k_3)$ is the gain matrix, $e_i = y_i - x_i$, $i = 1, 2, 3$ are the errors of the synchronization.

In numerical simulations, for $K = diag(0.2, 2, 3)$, and the initial values of the drive model (4.9) and the response model (4.10) are, respectively, $x_0 = (5.706, 6.3379, 17.0355)^T$, $y_0 = (2, 3, 4)^T$, $\hat{A}_0 = (37, 2, 21)^T$, the adaptive synchronization is achieved as shown in Figs. 2-4. Fig. 2 depicts the the state variables of adaptive synchronization between the models (4.9-4.10). As seen in Fig. 3, the synchronization errors approach zero. While the estimate of unknown parameters $\hat{A} = (\hat{a}, \hat{b}, \hat{c})^T$ converge to $A = (a, b, c)^T$ in Fig. 4. In order to establish synchronization, the analytical expansion of the controllers (4.11), which is derived from the theorem 4.1, is used.

5. Application of adaptive synchronization in image encryption

In this section, using the adaptive synchronization between DOLMs, we stated the application of image encryption. In the sender, we consider the model (4.9) as the drive model which generates chaotic signals. These signals of (4.9) drive the response model (4.10) to achieve our synchronization.

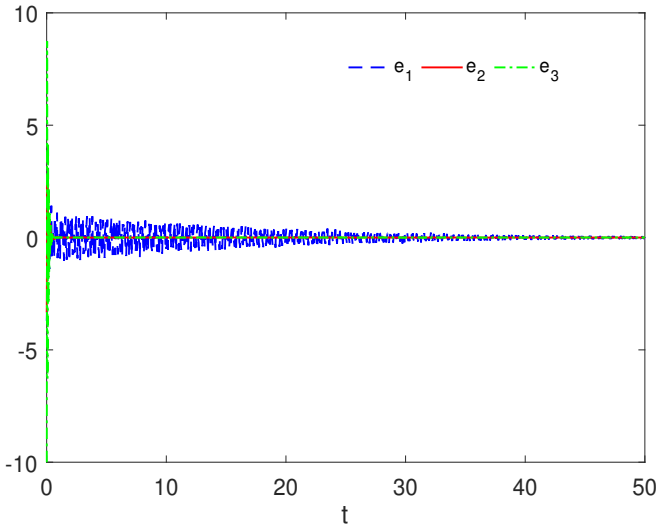

Figure 3: Synchronization errors between the models (4.9) and (4.10).

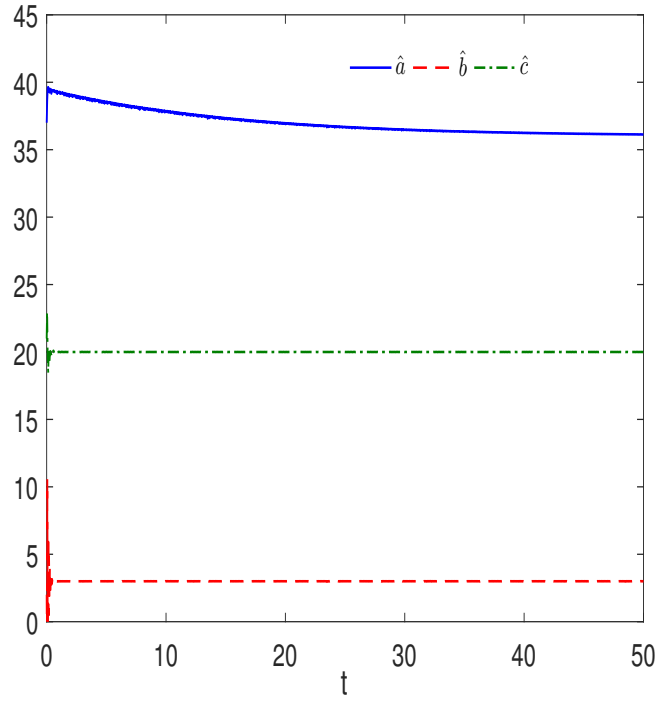

Figure 4: parameters identified for the drive model (4.9).

5.1. Encryption process

The encryption process for color images involves several steps. First, we take the plain image I with a size of $M \times N \times 3$, which is made up of three image components: red (R), green (G), and blue (B). Each component has a size of $M \times N$, and we encrypt each channel separately. We represent the image I as $[I_R, I_G, I_B]$. Next, each channel of the color image I, which has a size of $M \times N$ pixels, is initially represented as an $M \times N$ matrix of pixels. For example, the red channel (I_R) can be represented as a matrix of pixels:

$$I_R = \begin{pmatrix} I_{R11} & I_{R12} & \dots & I_{R1N} \\ \vdots & \vdots & \ddots & \vdots \\ I_{RM1} & I_{RM2} & \dots & I_{RMN} \end{pmatrix}. \tag{5.1}$$

We then transform the 2D matrices of I_R, I_G, and I_B into 1D vectors, P_R, P_G, and P_B, respectively, with a length of MN as:

$$P_R = [I_{R11}, I_{R12}, ..., I_{R1N}, I_{R21}, I_{R22}, ..., I_{R2N}, I_{RM1}, I_{RM2}, ..., I_{RMN}] = [E_{R1}, E_{R2}, ..., E_{RMN}]. \tag{5.2}$$

where, each element in the vectors is a pixel's gray value in the range of 0-255. We utilize the synchronization between the drive model (4.9) and response model (4.10) to generate a chaotic sequence K. The models are iterated for $N_0 + M \times N$ times, and the former N_0 values are discarded. The chaotic sequence K is an MN-dimensional vector as:

$$K = [K_1, K_2, ..., K_{MN}]. \tag{5.3}$$

We sort the chaotic sequence K in ascending order and calculate decimal values M using the sorted sequence.

$$M = mod(floor(K) \times 10^{14}, 256), \tag{5.4}$$

We perform an exclusive XOR operation between M and P_R to generate the encrypted vector H_R.

$$H_R = M \oplus P_R. \tag{5.5}$$

We repeat the same process for P_G and P_B to generate H_G and H_B, respectively. Finally, we convert the vectors H_R, H_G, and H_B to 2D matrices to generate the encrypted color image of size $M \times N$.

5.2. Decryption process

As we know the decryption is the reverse process of encryption and the key used for encryption and decryption are the same. The generated chaotic sequence is consistent for the encryption and decryption process. Therefore, the encryption application is symmetric and reversible, and we can easily decrypt the encrypted image with the inverse steps of encryption process.

Image	Red	Green	Blue
Original Peppers	7.3564	7.6187	7.1658
Encrypted Peppers	7.3563	7.6187	7.1663

Table 1: The information entropy of original and decrypted Peppers images.

5.3. Experimental results

It seems that in this subsection, the effectiveness of encrypting and decrypting color images is being assessed. The performance of the application is being evaluated using various methods such as visual analysis, information entropy, and histogram analysis. The test images used for this evaluation is ***Peppers***, which is 512×512 in size and can be seen in Figure 5(a).

5.3.1. Visual analysis

This subsubsection uses visual analysis to evaluate the similarity and dissimilarity of original, and decrypted images. Figure 5 displays the simulation results, with original image in Figure 5(a), and decrypted image in Figure 5(b). The encrypted image is noise-like, and the decrypted image is identical to the original one. This demonstrates the success of our image encryption algorithm in encrypting and decrypting color images.

5.3.2. Information Entropy

Information entropy is used to measure the randomness of an image, which is calculated as follows:

$$H(I) = \sum_{i=1}^{N} P(i) \log_2 \frac{1}{P(i)}, \tag{5.6}$$

where N is the maximum possible intensity value, and $P(i)$ denotes the probability of intensity value i in image I.

The information entropy values of encrypted and original images in RGB color images generated by the encryption algorithm in the Table 1, respectively. The proposed approach has an information entropy of almost 8, which means that there is a significant amount of randomness and it is close to the ideal value. This indicates that there is very little chance of information leakage in the proposed algorithm.

5.3.3. Histogram analysis

The histogram $h(i)$ for an image with intensity values between 0 and L gives the total number of pixels with intensity i. This histogram can be used to analyze the frequency of encrypted and decrypted images. However, it is important to note that the histogram of an encrypted image should not be similar to the histogram of the image before encryption. Fig. 6 shows the histograms of the B channel of original Peppers, and decrypted images respectively.

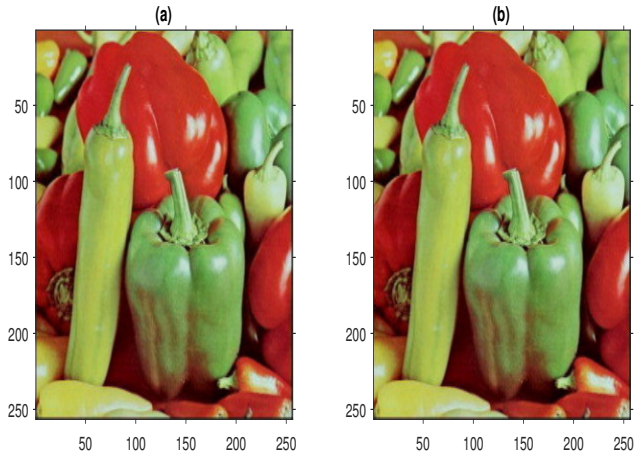

Figure 5: Original and decrypted Peppers images.

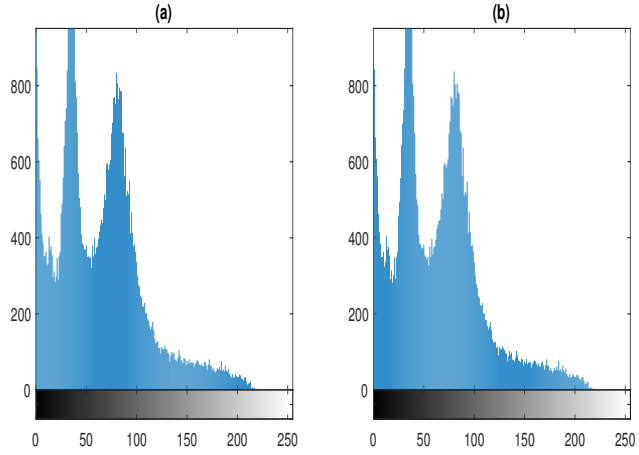

Figure 6: Analysis of histogram of B component of original image Peppers and its decrypted image.

The histogram of a color image component is not uniformly distributed. Therefore, the proposed encryption algorithm can hide statistical information of the plain image component by evenly distributing pixels in the encrypted image.

6. Electronic Circuit

In this section, we want to design an EC to simulate both our chaotic drive and response models. This requires a circuit that is able to simulate the frequency response of the DO operator. In [31], the authors proposed a method that uses successive pole-zero pairs that form a zig-zag pattern to approximate the frequency response of a single fractional order operator in a specific frequency range and for a particular maximum discrepancy value. The authors formulated a set of equations where the fractional order value, the frequency range, and the required

discrepancy are given and the values of the poles and zeros of the approximating transfer function are returned. However, this method is viable in the single fractional order case because the slope of the frequency response of the transfer function is constant which is not the same for the DO operator.

We propose a similar method where instead of using a set of equations to find the location of the pole-zeros pairs of the approximating transfer function, we implement an algorithm that uses an iterative method to achieve the same goal. This way even though the frequency response of the fractional order operator does not have a constant slope we are able to find an approximating function in a specific range of frequencies for a particular maximum discrepancy. In fact, our proposed method has the flexibility to be generalized for any transfer function with a low-pass frequency response which is monotonically decreasing beyond the corner frequency. The pseudo-code of this method is shown in Algorithm 1 whereas the Bisection method, in Algorithm 2, is used as the iterative method to find the exact locations of the poles and zeros of the approximate transfer function.

Algorithm 1 Algorithm for finding approximate function to the distributed fractional order operator

$a \leftarrow \omega_{min}$
$b \leftarrow \omega_{max}$
$flag \leftarrow$ **false** ▷ flag for pole/zero
$F_{approx} \leftarrow F(\omega_{min})$
$array[1] \leftarrow bisection(a, b, F, F_{approx})$
$const \leftarrow F(\omega_{min}) + 20log(array[1])$
$F_{approx}(\omega) \leftarrow const - 20log(\omega)$
while $abs(F(\omega_{max}) - F_{approx}(\omega_{max})) > \delta$ **do** ▷ δ = required discrepancy
  $a \leftarrow array[end]$
  $array[end+1] \leftarrow bisection(a, b, F, F_{approx})$
  if $flag$ is **false then**
    $F_{approx}(\omega) \leftarrow F_{approx}(array[end])$
    $flag \leftarrow$ **true**
  else
    $const \leftarrow F(\omega_{min}) + 20log(array[end])$
    $F_{approx}(\omega) \leftarrow const - 20log(\omega)$
    $flag \leftarrow$ **false**
  end if
end while

Recalling the definition of the proposed distributed fractional order derivative operator in (2.3). by taking

Algorithm 2 Algorithm of Bisection method

$c \leftarrow (a+b)/2$
while $abs(abs(F(c) - F_{approx}(c)) - \delta) > 10^{-6}$ **do**
  if $sign(abs(F(c) - F_{approx}(c)) - \delta) = sign(abs(F(\omega_{max}) - F_{approx}(\omega_{max})) - \delta)$ **then**
    $b \leftarrow c$
  else
    $a \leftarrow c$
  end if
  $c \leftarrow (a+b)/2$
end while
return c

the Laplace transform for both sides

$$\mathscr{L}\{D^{\psi(\beta)}x(t)\} \approx \sum_{j=1}^{m} \psi(\beta_j)\Delta\tau_j s^{\beta_j}\mathscr{L}\{\boldsymbol{x}(t)\}, \tag{6.1}$$

where $m = 50$, $\beta_j = \frac{2j-1}{2m}$, $\psi(\beta) = \frac{\beta^2}{10^3(1-\beta)^2}$ and $\Delta\tau_m = \frac{1}{m}$, thus the integral operator can be written as:

$$F(s) = \frac{1}{\sum_{j=1}^{m} \psi(\beta_j)\Delta\tau_j s^{\beta_j}} \tag{6.2}$$

Algorithm 1 is implemented in Matlab to find the approximate transfer function of (6.2). Since the frequency response of the integral operator in (6.2) tends to infinity at low frequency, a constant value of 0.001 is added to the denominator to obtain a finite value which is required for any practical application. Setting the value of $\omega_{min} = 10^{-5}$ rad/s, $\omega_{max} = 10$ rad/s and $\delta = 1$ dB, the frequency response of the integral operator and the zig-zag pattern is shown in Fig. 7 while the frequency responses of the integral operator with the resulting approximating function are presented in Fig. 8. The equation of the transfer function of the approximating function can be shown as:

$$F(s) \approx \frac{3.9827(s+0.002159)(s+0.007596)}{(s+0.001615)(s+0.003995)(s+0.01013)} \tag{6.3}$$

A tree-shaped model (TSM), shown in Fig. 9, was presented in [32] which uses resistors and capacitors in a configuration that has a transfer function of successive poles and zeros as the one represented in (6.3). The transfer function of the TSM can be written as:

$$-F(s) = \frac{-R}{C_0}\frac{(\frac{C_0}{C_1}+\frac{C_0}{C_3})(s+\frac{R_a+R_b}{R_aC_2R_c})(s+\frac{1}{C_1R_c+C_3R_c})}{s^3+(\frac{R_a+R_b}{R_aC_2R_B}+\frac{1}{C_3R_c}+\frac{C_1+C_3}{C_1R_aC_3})s^2+(\frac{R_a+R_b}{R_aC_2R_bC_3R_c}+\frac{1}{C_1R_aC_3R_c}+\frac{C_1+C_3}{C_1R_aC_2R_bC_3})s+\frac{1}{C_1R_aC_2R_bC_3R_c}}, \tag{6.4}$$

where C_0 is a unit parameter used to adjust the resistors and capacitors to values practical for ECs. Choosing $R = 1M\Omega$, $C_0 = 1\mu F$, $R_a = 854.389M\Omega$, $R_b = 145.611M\Omega$, $R_c = 66.151M\Omega$, $C_1 = 294.732nF$, $C_2 = 3.722\mu F$

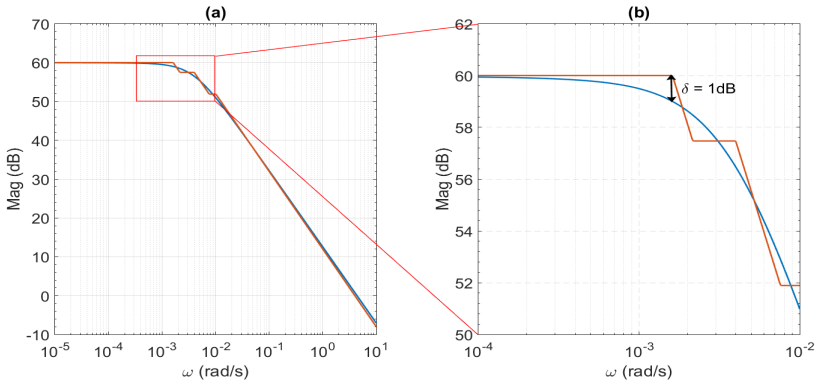

Figure 7: The frequency response of the integral operator and the zig-zag pattern: (a) over the complete range and (b) magnified area for more detail.

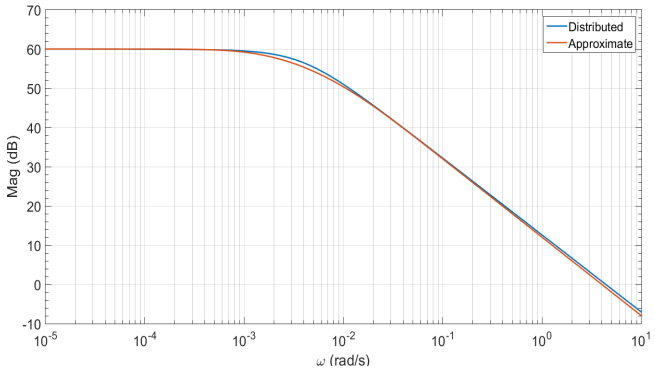

Figure 8: The frequency response of the integral operator and the approximating function.

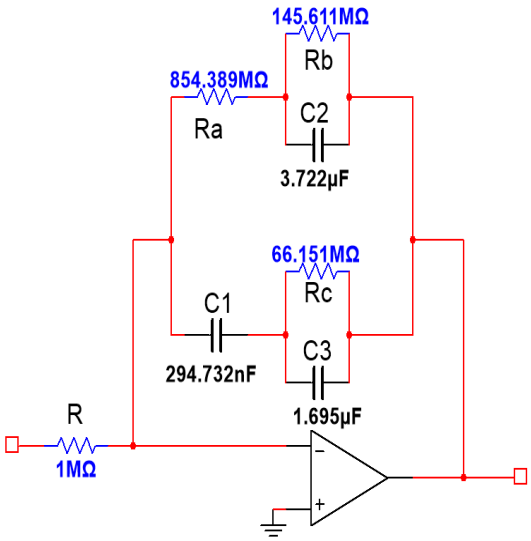

Figure 9: Tree-shaped model.

and $C_3 = 1.695\mu F$, the transfer function becomes equal to (6.3) with a negative sign due to inverting amplifier which can be accounted for in the completed circuit.

The drive model in (4.9) can be implemented as an EC using the TSM model where the schematic diagrams of x_1, x_2 and x_3 equations are shown in Fig. 10, Fig. 11 and Fig. 12 respectively. The equations obtained from schematic diagrams for the drive model in the Laplace domain are derived as:

$$\begin{aligned}
\frac{X_1(s)}{F(s)} =& \frac{R_1}{R}(X_2(s) - X_1(s)),\\
\frac{X_2(s)}{F(s)} =& \frac{R_2}{R}X_2(s) - \mathscr{L}\{x_1(t)x_3(t)\},\\
\frac{X_3(s)}{F(s)} =& \mathscr{L}\{x_1(t)x_2(t)\} - \frac{R_3}{R}X_3(s),
\end{aligned} \tag{6.5}$$

where $R = 1M\Omega$, $R_1 = 36M\Omega$, $R_2 = 20M\Omega$ and $R_3 = 3M\Omega$. The schematic diagram is implemented with ideal components in the Multisim schematic solver tool and the simulation results of the drive model circuit are shown in Fig. 13

Applying the same procedure for the estimated parameters updating equation in (4.6), the schematic diagrams for $\hat{a}$, $\hat{b}$ and $\hat{c}$ are shown in Fig. 14, Fig. 15 and Fig. 16 respectively. The equations derived from the schematic diagrams in the Laplace domain can be written as:

$$\begin{aligned}
\frac{\hat{A}(s)}{F(s)} =& \mathscr{L}\{-(y_2(t) - y_1(t))(y_1(t) - x_1(t))\},\\
\frac{\hat{B}(s)}{F(s)} =& \mathscr{L}\{y_3(t)(y_3(t) - x_3(t))\},\\
\frac{\hat{C}(s)}{F(s)} =& \mathscr{L}\{-y_2(t)(y_2(t) - x_2(t))\},
\end{aligned} \tag{6.6}$$

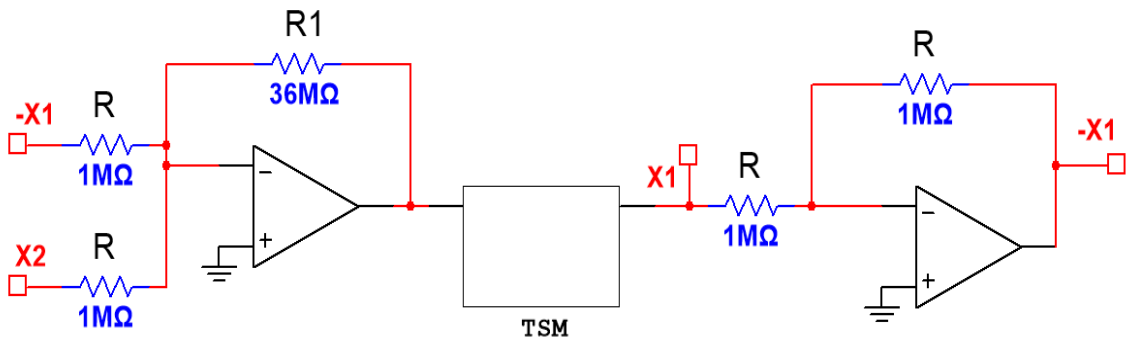

Figure 10: Schematic diagram of equation for x_1 of model (6.5).

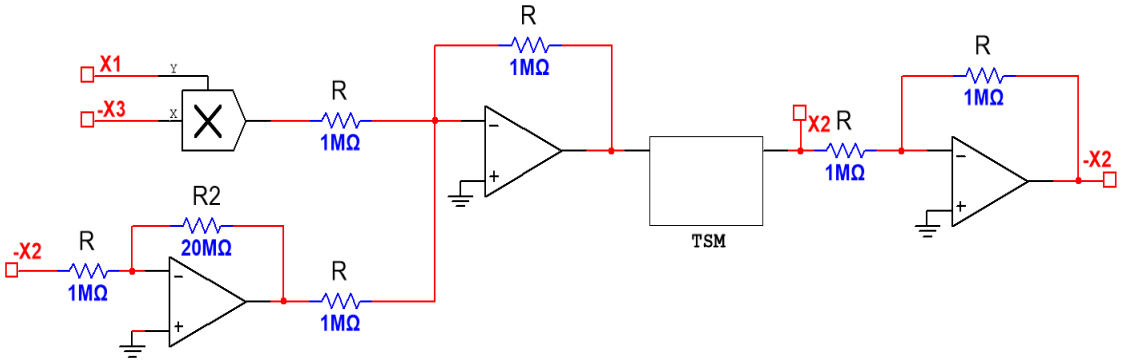

Figure 11: Schematic diagram of equation for x_2 of model (6.5).

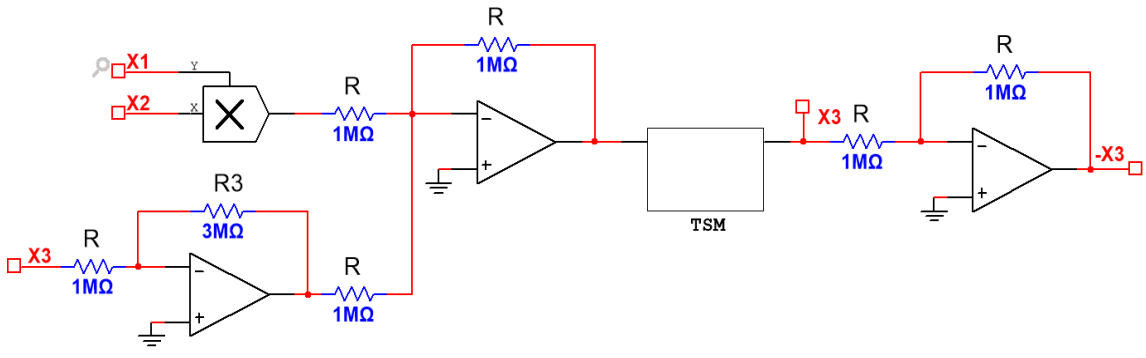

Figure 12: Schematic diagram of equation for x_3 of model (6.5).

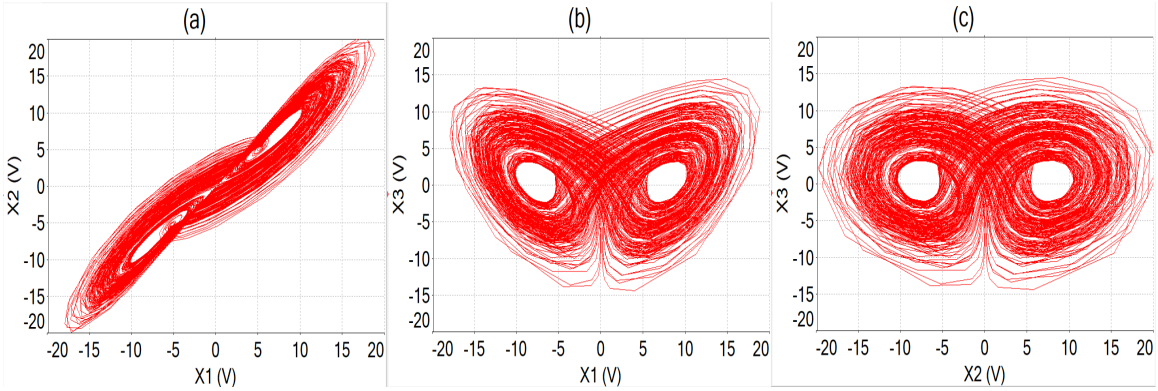

Figure 13: The simulation results of the drive model circuit: (a)x_1, x_2 space, (b)x_1, x_3 space and (c)x_2, x_3 space.

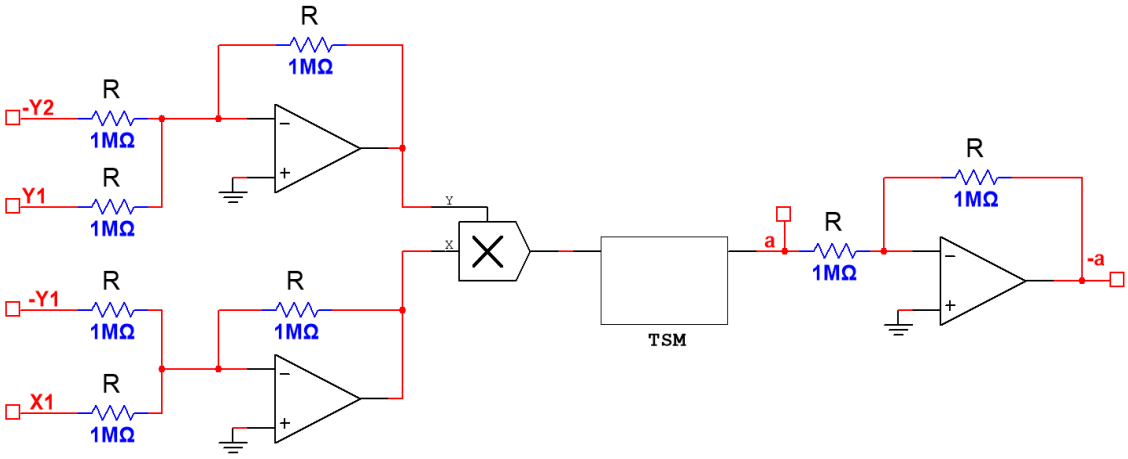

Figure 14: Schematic diagram of equation for $\hat{a}$ of model (6.6).

Similarly, we can design the EC for the response model in (4.10), the schematic diagrams for y_1, y_2 and y_3 are shown in Fig. 17, Fig. 18 and Fig. 19 respectively. The equations derived from the schematic diagrams in the Laplace domain can be written as:

$$\begin{aligned}
\frac{Y_1(s)}{F(s)} =& \mathscr{L}\{\hat{a}(t)(y_2(t)-y_1(t))\} + \frac{R_1}{R}(X_2(s)-X_1(s)) + \frac{R_1}{R}(Y_1(s)-Y_2(s)) + \frac{R_4}{R}(X_1(s)-Y_1(s)),\\
\frac{Y_1(s)}{F(s)} =& \hat{C}(s)Y_2(s) + \frac{R}{R_6}X_2(s) - \mathscr{L}\{x_1(t)x_3(t))\} - \frac{R}{R_6}Y_2(s) + \frac{R_5}{R}(X_2(s)-Y_2(s)),\\
\frac{Y_1(s)}{F(s)} =& -\mathscr{L}\{\hat{b}(t)y_3(t))\} + \mathscr{L}\{x_1(t)x_2(t)\},
\end{aligned} \tag{6.7}$$

where $R = 1M\Omega$, $R_1 = 36M\Omega$, $R_4 = 200k\Omega$, $R_5 = 2M\Omega$ and $R_6 = 50k\Omega$. The simulation results of synchronization errors between the drive and response models are shown in Fig. 20 while the values of the estimated parameters are presented in Fig. 21.

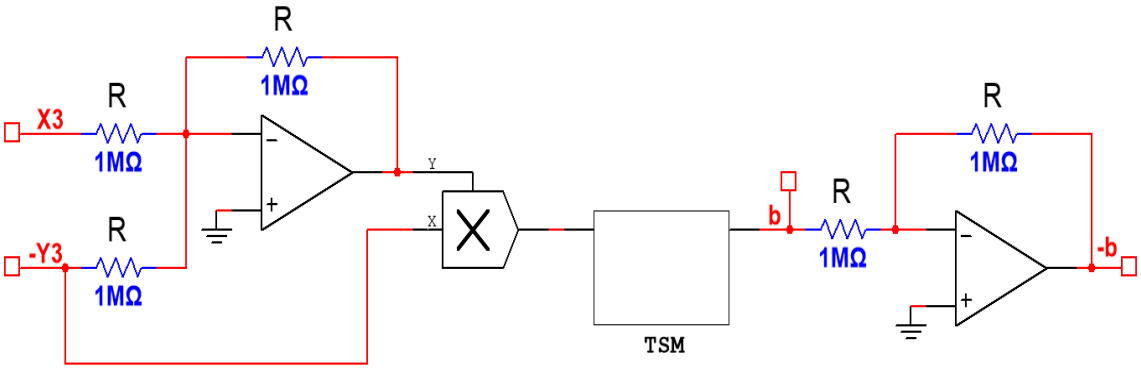

Figure 15: Schematic diagram of equation for $\hat{b}$ of model (6.6).

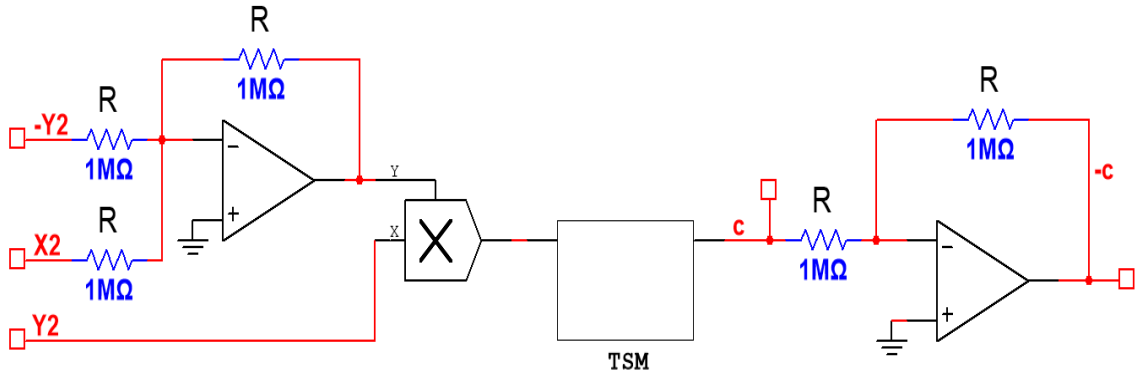

Figure 16: Schematic diagram of equation for $\hat{c}$ of model (6.6).

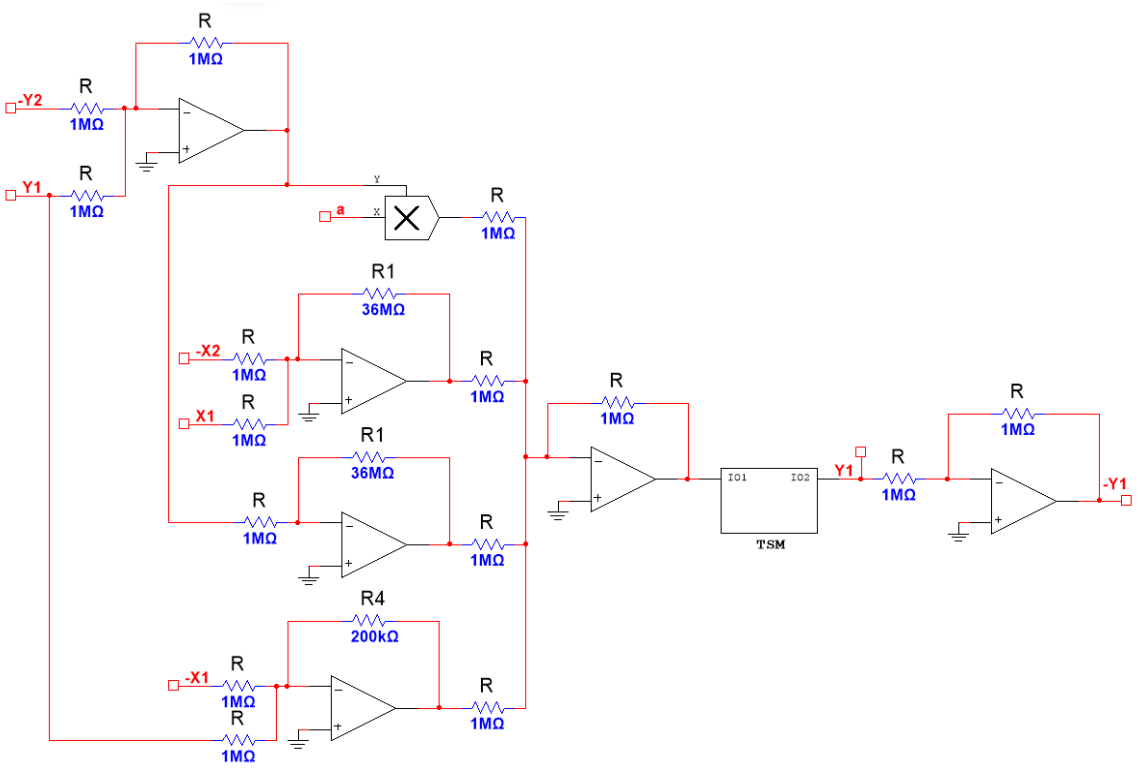

Figure 17: Schematic diagram of equation for y_1 of model (6.7).

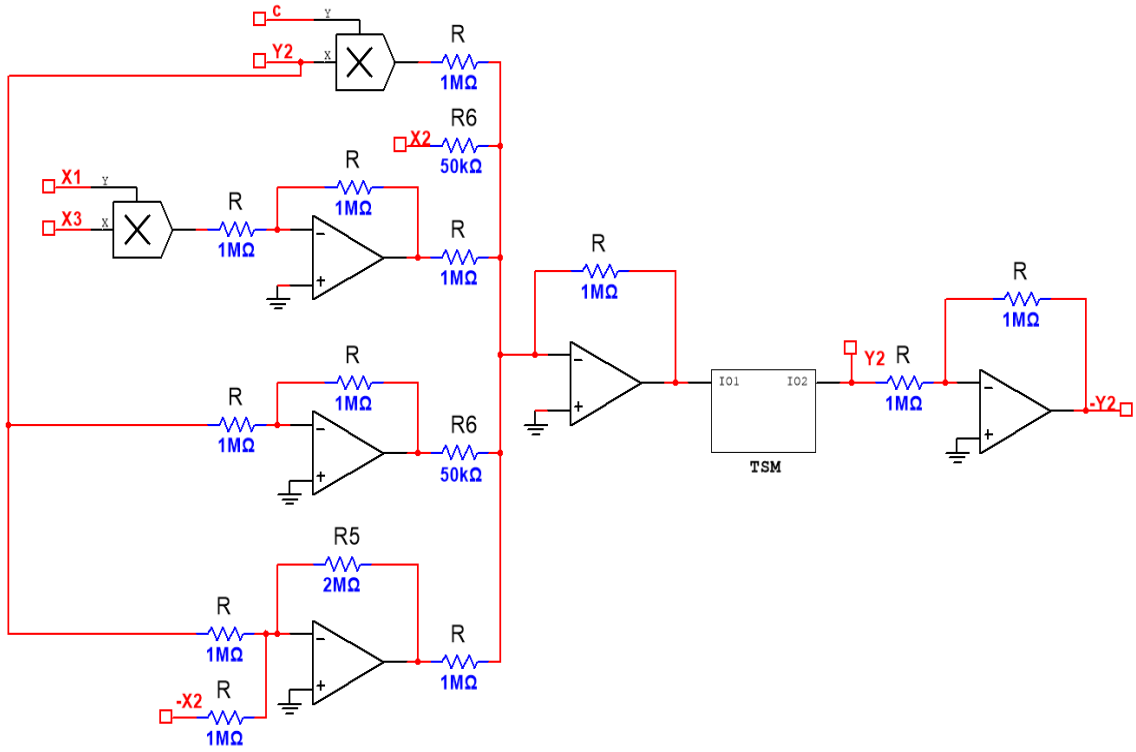

Figure 18: Schematic diagram of equation for y_2 of model (6.7).

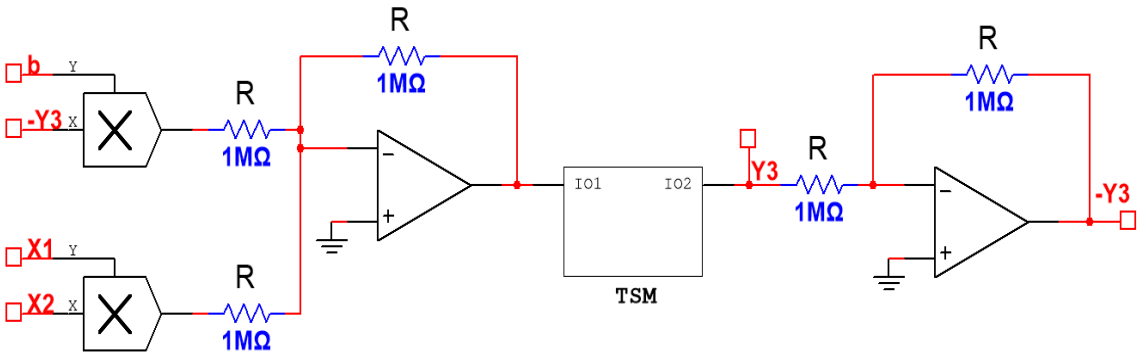

Figure 19: Schematic diagram of equation for y_3 of model (6.7).

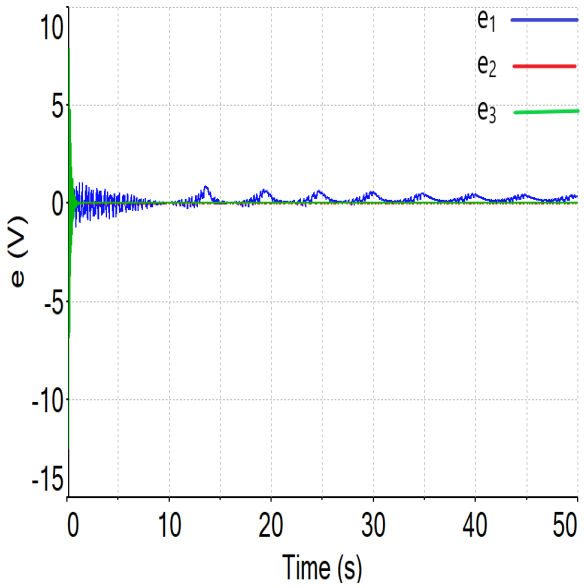

Figure 20: Synchronization errors of circuit simulation.

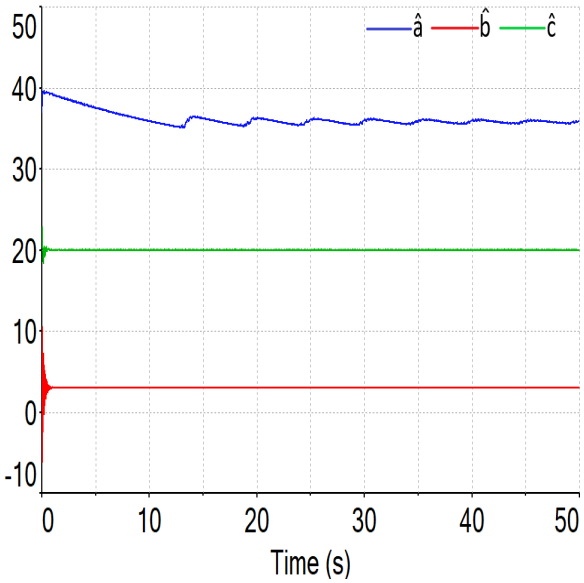

Figure 21: Estimated parameters of circuit simulation.

7. Conclusion

The dynamics of the proposed chaotic distributed order Lü model (DOLM) is examined, and its chaotic solution is demonstrated in Fig. 1. The Lyapunov direct approach and nonlinear feedback control are used to achieve the adaptive synchronization between two DOMs. We presented theorem 4.1 to drive the analytical controllers (4.5) and the estimated parameters updating law (4.6). In order to verify the accuracy of the analytical results obtained by Theorem 4.1, we used the DOLM as an example. The numerical results are calculated to show the effectiveness of our synchronization technique, see Figs. 2-4.

A color image encryption scheme is provided using the adaptive synchronization. The simulation results for original and decrypted images are shown in Fig. 5. In Table 1, we calculated and analysed the information entropy of encrypted and plain images. They are clearly approaching the ideal value of 8. The ciphered image histograms are almost flat and the encrypted image distributions are really uniform as shown in Fig. 6. The results of encryption and decryption for color images were presented using Matlab programme. A novel approach for approximating the distributed order operator is proposed. This approach uses an iterative method to find the location of the poles and zeros of the approximating transfer function. Our proposed approach can be implemented to approximate any distributed order operator with the desired level of accuracy. The ECs for the DOLM (1.1) are given in Figs. 10-19. The results of the adaptive synchronization of the ECs as shown in Figs. 20-21 agree with the numerical results of Figs. 3-4.

Data Availability

The datasets of the current study are available from the corresponding author on reasonable request.

Conflict of interest

The authors declare that they have no conflict of interest.

References

[1] M. Caputo, Elasticita e dissipazione, Zanichelli, 1969.

[2] M. Caputo, Mean fractional-order-derivatives differential equations and filters, Annali dell'Universita di Ferrara 41 (1995) 73–84.

[3] R. Bagley, P. Torvik, On the existence of the order domain and the solution of distributed order equations-part I, International Journal of Applied Mathematics 2 (2000) 865–882.

[4] G. Fernández-Anaya, G. Nava-Antonio, J. Jamous-Galante, R. Muñoz-Vega, E. G. Hernández-Martínez, Asymptotic stability of distributed order nonlinear dynamical systems, Communications in Nonlinear Science and Numerical Simulation 48 (2017) 541–549.

[5] G. M. Mahmoud, T. Aboelenen, T. M. Abed-Elhameed, A. A. Farghaly, Generalized Wright stability for distributed fractional-order nonlinear dynamical systems and their synchronization, Nonlinear Dynamics 97 (2019) 413–429.

[6] A. J. Muñoz-Vázquez, G. Fernández-Anaya, J. D. Sánchez-Torres, F. Meléndez-Vázquez, Predefined-time control of distributed-order systems, Nonlinear Dynamics 103 (2021) 2689–2700.

[7] A. J. Muñoz-Vázquez, G. Fernández-Anaya, J. D. Sánchez-Torres, S. Boulaaras, Robust stabilisation of distributed-order systems, Mathematical Methods in the Applied Sciences (2022).

[8] T. Aboelenen, Local discontinuous Galerkin method for distributed-order time and space-fractional convection–diffusion and Schrödinger-type equations, Nonlinear dynamics 92 (2018) 395–413.

[9] J. Chen, C. Li, X. Yang, Chaos synchronization of the distributed-order Lorenz system via active control and applications in chaotic masking, International Journal of Bifurcation and Chaos 28 (2018) 1850121.

[10] G. M. Mahmoud, A. A. Farghaly, T. M. Abed-Elhameed, S. A. Aly, A. A. Arafa, Dynamics of distributed-order hyperchaotic complex van der Pol oscillators and their synchronization and control, The European Physical Journal Plus 135 (2020) 1–16.

[11] G. M. Mahmoud, T. Aboelenen, T. M. Abed-Elhameed, A. A. Farghaly, On boundedness and projective synchronization of distributed order neural networks, Applied Mathematics and Computation 404 (2021) 126198.

[12] H. Zang, M. Tai, X. Wei, Image encryption schemes based on a class of uniformly distributed chaotic systems, Mathematics 10 (2022) 1027.

[13] N. Li, J. Cao, New synchronization criteria for memristor-based networks: adaptive control and feedback control schemes, Neural Networks 61 (2015) 1–9.

[14] Y. Wei, J. Qiu, H. R. Karimi, Reliable output feedback control of discrete-time fuzzy affine systems with actuator faults, IEEE Transactions on Circuits and Systems I: Regular Papers 64 (2016) 170–181.

[15] H. Wang, B. Chen, C. Lin, Y. Sun, F. Wang, Adaptive neural control for mimo nonlinear systems with unknown dead zone based on observers, International Journal of Innovative Computing, Information and Control 14 (2018) 1339–1349.

[16] J. Cao, J. Lu, Adaptive synchronization of neural networks with or without time-varying delay, Chaos: An Interdisciplinary Journal of Nonlinear Science 16 (2006) 013133.

[17] H. Zhang, W. Huang, Z. Wang, T. Chai, Adaptive synchronization between two different chaotic systems with unknown parameters, Physics Letters A 350 (2006) 363–366.

[18] H. Bao, J. H. Park, J. Cao, Adaptive synchronization of fractional-order memristor-based neural networks with time delay, Nonlinear Dynamics 82 (2015) 1343–1354.

[19] Y. Song, W. Sun, Adaptive synchronization of stochastic memristor-based neural networks with mixed delays, Neural Processing Letters 46 (2017) 969–990.

[20] W. S. Sayed, A. G. Radwan, Generalized switched synchronization and dependent image encryption using dynamically rotating fractional-order chaotic systems, AEU-International Journal of Electronics and Communications 123 (2020) 153268.

[21] S. Moon, J.-J. Baik, J. M. Seo, Chaos synchronization in generalized lorenz systems and an application to image encryption, Communications in Nonlinear Science and Numerical Simulation 96 (2021) 105708.

[22] M. Yuan, X. Luo, X. Mao, Z. Han, L. Sun, P. Zhu, Event-triggered hybrid impulsive control on lag synchronization of delayed memristor-based bidirectional associative memory neural networks for image hiding, Chaos, Solitons & Fractals 161 (2022) 112311.

[23] G. Biban, R. Chugh, A. Panwar, Image encryption based on 8D hyperchaotic system using Fibonacci Q-Matrix, Chaos, Solitons & Fractals 170 (2023) 113396.

[24] M. Wang, Y. Deng, X. Liao, Z. Li, M. Ma, Y. Zeng, Dynamics and circuit implementation of a four-wing memristive chaotic system with attractor rotation, International Journal of Non-Linear Mechanics 111 (2019) 149–159.

[25] L. Minati, L. Gambuzza, W. Thio, J. Sprott, M. Frasca, A chaotic circuit based on a physical memristor, Chaos, Solitons & Fractals 138 (2020) 109990.

[26] V.-T. Pham, A. Akgul, C. Volos, S. Jafari, T. Kapitaniak, Dynamics and circuit realization of a no-equilibrium chaotic system with a boostable variable, AEU-International Journal of Electronics and Communications 78 (2017) 134–140.

[27] M. Wang, X. Liao, Y. Deng, Z. Li, Y. Su, Y. Zeng, Dynamics, synchronization and circuit implementation of a simple fractional-order chaotic system with hidden attractors, Chaos, Solitons & Fractals 130 (2020) 109406.

[28] X. Zhang, Z. Li, D. Chang, Dynamics, circuit implementation and synchronization of a new three-dimensional fractional-order chaotic system, AEU-International Journal of Electronics and Communications 82 (2017) 435–445.

[29] G. M. Mahmoud, T. M. Abed-Elhameed, M. M. Elbadry, A class of different fractional-order chaotic (hyper-chaotic) complex duffing-van der Pol models and their circuits implementations, Journal of Computational and Nonlinear Dynamics 16 (2021).

[30] Z. Jiao, Y.-Q. Chen, I. Podlubny, Distributed-Order Dynamic Systems: Stability, Simulation, Applications and Perspectives, SpringerBriefs in Electrical and Computer Engineering. Springer, 2012.

[31] A. Charef, H. Sun, Y. Tsao, B. Onaral, Fractal system as represented by singularity function, IEEE Transactions on automatic Control 37 (1992) 1465–1470.

[32] C. Xiang-Rong, L. Chong-Xin, W. Fa-Qiang, Circuit realization of the fractional-order unified chaotic system, Chinese Physics B 17 (2008) 1664.