AIMS *Mathematics*

Research article

New results on finite-time projective synchronization for memristor-based hybrid delayed BAM neural networks with applications to DNA image encryption

Meng Li[1] and Gulijiamali Maimaitiaili[1,2,*]

[1] School of Mathematics Science, Xinjiang Normal University, Urumqi, Xinjiang 830017, China
[2] Xinjiang Key Laboratory of Mental Development and Learning Science, China

* Correspondence: Email: gulijiamali@xjnu.edu.cn.

Abstract: With the popularization of digital image technology, image information has inevitably developed to involved the disclosure of personal privacy; in this study, a color image encryption algorithm was designed to encrypt and decrypt images by using chaotic sequences of a class of memristor-based hybrid delayed bidirectional associative memory neural networks (MHDBAMNNs) to protect images from illegal acquisition and use. Additionally, the discontinuity problem of the right-hand side of the Filippov system due to the hopping property of the memristor has been treated by using differential inclusion and set-valued mapping theories, and a sufficient criterion for guaranteeing the synchronization of finite-time projections derived based on the drive-response concept, Lyppunov stability theorem, and inequality technique. To improve the security performance, a color image encryption algorithm based on a combination of Chen's hyperchaotic system and a DNA codec operation was adopted, also, the robustness and validity of our proposed approach was demonstrated through image performance analysis. Furthermore, the potential application of the model in secure transmission has been explored.

1. Introduction

Memristors used to replace resistors in artificial neural networks constitute memory neural networks (MNNs), which are steadily garnering more attention, particularly in the fields of brain-like simulation [1,2] and computer science. Owing to the hysteresis loop properties of memristors [3], memory-delayed neural network systems may exhibit special dynamics and produce novel chaotic phenomena [4]. These phenomena allow amnesic time-delay neural networks to be used as chaotic systems in applications such as pseudorandom number generators [5], secure communication [6], image transmission [7], and image encryption [8–10].

A novel flux-controlled memristor model containing a set of hyperbolic tangent functions has been developed by analyzing multiple double-scroll chaotic attractors, amplitude control features, and extreme multi-stability [11]. A five-dimensional grid-scroll memristive chaotic system has been effectively designed by employing two types of flux-controlled memristors and an equivalent circuit that is capable of generating grid-scroll attractors has been devised [12].

The efficiency of image transmission can be improved by incorporating finite-time synchronization. In a previous study on finite-time projection synchronization, aiming to address the unsynchronized oscillations caused by external noise of a nonlinear dynamic system of microgrid hierarchical control, a finite-time function projection synchronization control scheme was developed by combining complex networks and microgrids [13]. A finite-time robust feedback controller of surface-balanced systems with hidden attractors has been constructed by applying phase analysis, bifurcation, and a Poincaré section, and the synchronization of master-slave systems was realized in a finite time [14]. In [15], the finite-time projection synchronization problem for Caputo-type fractional-order complex-valued neural networks with time delay is proposed based on fractional calculus and inequality techniques.

The goodness of the chaotic system properties has been found to determine the effective encryption of chaotic images, and a four-dimensional multi-scroll chaotic system based on two types of flux-controlled non-volatile memristor models with a set of signum functions has been constructed, additionally, the chaotic sequence generated by the proposed system was demonstrated to effectively encrypt the image and have good robustness [16]. The equivalent simulation circuit has been constructed by using discrete components, and a commercial image encryption system for the Internet of Things has been implemented based on the proposed high-dimensional memristive Hopfield neural network model [17]. A 5D fractional-order memristive Hopfield neural network (FOMHNN) under electromagnetic radiation was developed by applying memristors as self-synaptic weights of a Hopfield neural network, and an extremely secure image encryption algorithm has been proposed based on the FOMHNN [18]. Kanwal et al. proposed an image encryption framework based on the integration of chaotic maps and color codes [19]. In [20], a color-image encryption scheme was designed by combining hyperchaotic maps, cross-plane operations, and gene theories.

DNA image encryption schemes based on a high-dimensional chaotic system have very broad development prospects and research value. In the study of Chen's hyperchaotic system, a new hyper-chaotic system based on the three-dimensional Chen system designed by using a smooth continuous nonlinear flux-controlled memristor model [21]. Regarding DNA image encryption schemes, a symmetric cipher based on the permutation substitution structure was comprehensively cryptanalyzed

by combining quantum chaotic maps and DNA coding [22]. A chaotic image encryption scheme based on an improved foraging behavior of the artificial fish school and DNA coding has also been proposed [23].

Motivated by the aforementioned discussions, this study mainly involving investigating the finite-time projection synchronization of bidirectional associative MNNs with leakage delays, time-varying discrete delays, and time-varying distributed delays; the finite-time projection synchronization criterion has been derived under the condition that the negative feedback controller coefficient is related to the connection weight values of other layers; similarly, the color image encryption algorithm based on a combination of Chen's hyperchaotic system and a DNA codec operation has been designed to encrypt and decrypt images by using chaotic sequences of the proposed system, and good results have been achieved to show the superiority of the encryption algorithm.

The remainder of this paper is organized as follows. Section 2 describes the drive-response system model and preliminary preparations. Section 3 presents a time delay independent negative feedback controller that has been designed by using a feedback control strategy to provide a criterion for the finite-time projection synchronization of MHDBAMNNs. Section 4 provides numerical simulation examples to demonstrate the validity of the proposed method. Section 5 presents the color image encryption algorithm based on the combination of Chen's hyperchaotic system and the DNA codec operation that was developed by using chaotic sequences of the proposed system. Section 6 concludes the study and offers perspectives for future research.

Note: $\Xi([-\theta, 0], \mathbb{R}^n)$ is a Banach space consisting of all continuous function maps from $[-\theta, 0]$ to $\mathbb{R}^n$.

2. Model description and preliminaries

In this study, the following bidirectional associative MNNs with mixed time delays (leakage time delays, time-varying discrete time delays, and time-varying distributed time delays) is considered:

$$
\begin{cases}
\dot{x}_i(t) = -\delta_i x_i(t - \sigma) + \sum_{j=1}^m a_{ji}\big(x_i(t)\big) f_j\big(y_j(t)\big) + \sum_{j=1}^m b_{ji}\big(x_i(t)\big) f_j\big(y_j\big(t - \tau(t)\big)\big) \\
\qquad + \sum_{j=1}^m c_{ji}\big(x_i(t)\big) \int_{t-\rho(t)}^t f_j\big(y_j(v)\big) dv + I_i(t), \\
\dot{y}_j(t) = -\omega_j y_j(t - \sigma) + \sum_{i=1}^n k_{ij}\big(y_j(t)\big) g_i\big(x_j(t)\big) + \sum_{i=1}^n l_{ij}\big(y_j(t)\big) g_i\big(x_j\big(t - \tau(t)\big)\big) \\
\qquad + \sum_{i=1}^n m_{ij}\big(y_j(t)\big) \int_{t-\rho(t)}^t g_i\big(x_j(v)\big) dv + H_j(t),
\end{cases}
\tag{1}
$$

where $t \geq 0$, $i = 1,2,\ldots,n, j = 1,2,\ldots,m$; $x_i(t)$ and $y_j(t)$ denotes the states of the i-th neuron in the x-layer and j-th neuron in the y-layer, respectively; $\delta_i > 0$ and $\omega_j > 0$ represent the rates of neuron self-inhibition; $f_j(\cdot)$ and $g_i(\cdot)$ are the neuron activation functions; σ is the leakage time delay, $\tau(t)$ denotes the time-varying discrete time delay, $\rho(t)$ denotes the time-varying distributed time delay; $\sigma > 0$, $0 \leq \tau(t) \leq \tau_0$, $\dot{\tau}(t) \leq \tau_1 < 1$, and $0 \leq \rho(t) \leq \rho_0$; $I_i(t)$ and $H_j(t)$ are the external input variables for the network system. The initial values of system (1) are given as $x_i(\chi) = \phi_i(\chi)$, $\phi(\chi) = [\phi_1(\chi), \phi_2(\chi), \ldots, \phi_n(\chi)] \in \Xi([-\theta, 0], \mathbb{R}^n)$; $y_j(\chi) = \psi_j(\chi)$, $\psi(\chi) = [\psi_1(\chi),$

$\psi_2(\chi), \dots, \psi_m(\chi)] \in \Xi([-\theta, 0], \mathbb{R}^m)$, where $\theta = max\{\sigma, \tau_0, \rho_0\}$; $a_{ij}(x_i(t))$, $b_{ij}(x_i(t))$, $c_{ji}(x_i(t))$, $k_{ij}(y_i(t))$, $l_{ij}(y_i(t))$, $m_{ij}(y_i(t))$ describe the connection weights of the i-th and the j-th nodes, respectively, and they are given as follows:

$$a_{ji}(\gamma) = \begin{cases} \acute{a}_{ji}, & |\gamma| \leq T_i, \\ \grave{a}_{ji}, & |\gamma| > T_i, \end{cases}$$

$$b_{ji}(\gamma) = \begin{cases} \acute{b}_{ji}, & |\gamma| \leq T_i, \\ \grave{b}_{ji}, & |\gamma| < T_i, \end{cases}$$

$$c_{ji}(\gamma) = \begin{cases} \acute{c}_{ji}, & |\gamma| \leq T_i, \\ \grave{c}_{ji}, & |\gamma| > T_i, \end{cases}$$

$$k_{ij}(\gamma) = \begin{cases} \acute{k}_{ij}, & |\gamma| \leq \hat{T}_j, \\ \grave{k}_{ij}, & |\gamma| > \hat{T}_j, \end{cases}$$

$$l_{ij}(\gamma) = \begin{cases} \acute{l}_{ij}, & |\gamma| \leq \hat{T}_j, \\ \grave{l}_{ij}, & |\gamma| > \hat{T}_j, \end{cases}$$

$$m_{ij}(\gamma) = \begin{cases} \acute{m}_{ij}, & |\gamma| \leq \hat{T}_j, \\ \grave{m}_{ij}, & |\gamma| > \hat{T}_j, \end{cases}$$

where $\gamma = x_i(t)$ or $y_j(t)$; the switching jumps T_i, $\hat{T}_j$ are positive constants; and $\acute{a}_{ji}, \grave{a}_{ji}, \acute{b}_{ji}, \grave{b}_{ji}, \acute{c}_{ji}, \grave{c}_{ji}, \acute{k}_{ij}, \grave{k}_{ij}, \acute{l}_{ij}, \grave{l}_{ij}, \acute{m}_{ij}, \grave{m}_{ij}$ are constants. To clearly present the proof of the theoretical process, the following representations are considered:

$$a_{ji}^+ = \max\{\acute{a}_{ji}, \grave{a}_{ji}\}, a_{ji}^- = \min\{\acute{a}_{ji}, \grave{a}_{ji}\}, a_{ji}^m = \max\{|\acute{a}_{ji}|, |\grave{a}_{ji}|\},$$

$$b_{ji}^+ = \max\{\acute{b}_{ji}, \grave{b}_{ji}\}, b_{ji}^- = \min\{\acute{b}_{ji}, \grave{b}_{ji}\}, b_{ji}^m = \max\{|\acute{b}_{ji}|, |\grave{b}_{ji}|\},$$

$$c_{ji}^+ = \max\{\acute{c}_{ji}, \grave{c}_{ji}\}, c_{ji}^- = \min\{\acute{c}_{ji}, \grave{c}_{ji}\}, c_{ji}^m = \max\{|\acute{c}_{ji}|, |\grave{c}_{ji}|\},$$

$$k_{ij}^+ = \max\{\acute{k}_{ij}, \grave{k}_{ij}\}, k_{ij}^- = \min\{\acute{k}_{ij}, \grave{k}_{ij}\}, k_{ij}^m = \max\{|\acute{k}_{ij}|, |\grave{k}_{ij}|\},$$

$$l_{ij}^+ = \max\{\acute{l}_{ij}, \grave{l}_{ij}\}, l_{ij}^- = \min\{\acute{l}_{ij}, \grave{l}_{ij}\}, l_{ij}^m = \max\{|\acute{l}_{ij}|, |\grave{l}_{ij}|\},$$

$$m_{ij}^+ = max\{\acute{m}_{ij}, \grave{m}_{ij}\}, m_{ij}^- = min\{\acute{m}_{ij}, \grave{m}_{ij}\}, m_{ij}^m = max\{|\acute{m}_{ij}|, |\grave{m}_{ij}|\}.$$

Owing to the discontinuity problem of the system caused by the hopping property of the memristor, which must be treated by using the theory of differential inclusion and set-valued mapping, system (1) can be transformed into the following system:

$$
\begin{cases}
\dot{x}_i(t) \in -\delta_i x_i(t-\sigma) + \sum_{j=1}^{m} co[a_{ji}(x_i(t))]f_j\left(y_j(t)\right) \\
\qquad + \sum_{j=1}^{m} co[b_{ji}(x_i(t))]f_j\left(y_j(t-\tau(t))\right) \\
\qquad + \sum_{j=1}^{m} co[c_{ji}(x_i(t))] \int_{t-\rho(t)}^{t} f_j\left(y_j(v)\right) dv + I_i(t), \\
\dot{y}_j(t) \in -\omega_j y_j(t-\sigma) + \sum_{i=1}^{n} co\left[k_{ij}\left(y_j(t)\right)\right] g_i\left(x_j(t)\right) \\
\qquad + \sum_{i=1}^{n} co\left[l_{ij}\left(y_j(t)\right)\right] g_i\left(x_j(t-\tau(t))\right) \\
\qquad + \sum_{i=1}^{n} co\left[m_{ij}\left(y_j(t)\right)\right] \int_{t-\rho(t)}^{t} g_i\left(x_j(v)\right) dv + H_j(t),
\end{cases}
\tag{2}
$$

where $co[a_{ji}(x_i(t))]$, $co[b_{ji}(x_i(t))]$, $co[c_{ji}(x_i(t))]$, $co[k_{ij}(y_j(t))]$, $co[l_{ij}(y_j(t))]$, and $co[m_{ij}(y_j(t))]$ denote the convex closures of the assemblies $[a_{ji}(x_i(t))]$, $[b_{ji}(x_i(t))]$, $[c_{ji}(x_i(t))]$, $[k_{ij}(y_j(t))]$, $[l_{ij}(y_j(t))]$, and $[m_{ij}(y_j(t))]$, respectively, and they are given as follows:

$$
co[a_{ji}(\gamma)] = \begin{cases}
\acute{a}_{ji}, & |\gamma| < T_i, \\
[a_{ji}^-, a_{ji}^+], & |\gamma| = T_i, \\
\grave{a}_{ji}, & |\gamma| > T_i,
\end{cases}
$$

$$
co[b_{ji}(\gamma)] = \begin{cases}
\acute{b}_{ji}, & |\gamma| < T_i, \\
[b_{ji}^-, b_{ji}^+], & |\gamma| = T_i, \\
\grave{b}_{ji}, & |\gamma| > T_i,
\end{cases}
$$

$$
co[c_{ji}(\gamma)] = \begin{cases}
\acute{c}_{ji}, & |\gamma| < T_i, \\
[c_{ji}^-, c_{ji}^+], & |\gamma| = T_i, \\
\grave{c}_{ji}, & |\gamma| > T_i,
\end{cases}
$$

$$
co[k_{ij}(\gamma)] = \begin{cases}
\acute{k}_{ij}, & |\gamma| < \hat{T}_j, \\
[k_{ij}^-, k_{ij}^+], & |\gamma| = \hat{T}_j, \\
\grave{k}_{ij}, & |\gamma| > \hat{T}_j,
\end{cases}
$$

$$
co[l_{ij}(\gamma)] = \begin{cases}
\acute{l}_{ij}, & |\gamma| < \hat{T}_j, \\
[l_{ij}^-, l_{ij}^+], & |\gamma| = \hat{T}_j, \\
\grave{l}_{ij}, & |\gamma| > \hat{T}_j,
\end{cases}
$$

$$
co[m_{ij}(\gamma)] = \begin{cases}
\acute{m}_{ij}, & |\gamma| < \hat{T}_j, \\
[m_{ij}^-, m_{ij}^+], & |\gamma| = \hat{T}_j, \\
\grave{m}_{ij}, & |\gamma| > \hat{T}_j.
\end{cases}
$$

Let $a_{ji}^x(t) \in co[a_{ji}(x_i(t))]$, $b_{ji}^x(t) \in co[b_{ji}(x_i(t))]$, $c_{ji}^x(t) \in co[c_{ji}(x_i(t))]$, $k_{ij}^y(t) \in co[k_{ij}(y_j(t))]$, $l_{ij}^y(t) \in co[l_{ij}(y_j(t))]$, and $m_{ij}^y(t) \in co[m_{ij}(y_j(t))]$ [24]. Then, the system (2) can be written as follows:

$$\begin{cases} \dot{x}_i(t) = -\delta_i x_i(t-\sigma) + \sum_{j=1}^{m} a_{ji}^x(t) f_j\left(y_j(t)\right) + \sum_{j=1}^{m} b_{ji}^x(t) f_j\left(y_j(t-\tau(t))\right) \\ \quad + \sum_{j=1}^{m} c_{ji}^x(t) \int_{t-\rho(t)}^{t} f_j\left(y_j(v)\right) dv + I_i(t), \\ \dot{y}_j(t) = -\omega_j y_j(t-\sigma) + \sum_{i=1}^{n} k_{ij}^y(t) g_i\left(x_j(t)\right) + \sum_{i=1}^{n} l_{ij}^y(t) g_i\left(x_j(t-\tau(t))\right) \\ \quad + \sum_{i=1}^{n} m_{ij}^y(t) \int_{t-\rho(t)}^{t} g_i\left(x_j(v)\right) dv + H_j(t), \end{cases} \tag{3}$$

The finite-time projection synchronization criterion is derived by using the drive-response concept, and system (1) is considered as a drive system; the response system of MHDBAMNNs is described as follows:

$$\begin{cases} \dot{\tilde{x}}_i(t) = -\delta_i \tilde{x}_i(t-\sigma) + \sum_{j=1}^{m} a_{ji}(\tilde{x}_i(t)) f_j\left(\tilde{y}_j(t)\right) + \sum_{j=1}^{m} b_{ji}(\tilde{x}_i(t)) f_j\left(\tilde{y}_j(t-\tau(t))\right) \\ \quad + \sum_{j=1}^{m} c_{ji}(\tilde{x}_i(t)) \int_{t-\rho(t)}^{t} f_j\left(\tilde{y}_j(v)\right) dv + u_i(t) + I_i(t), \\ \dot{\tilde{y}}_j(t) = -\omega_j \tilde{y}_j(t-\sigma) + \sum_{i=1}^{n} k_{ij}\left(\tilde{y}_j(t)\right) g_i\left(\tilde{x}_j(t)\right) + \sum_{i=1}^{n} l_{ij}\left(\tilde{y}_j(t)\right) g_i\left(\tilde{x}_j(t-\tau(t))\right) \\ \quad + \sum_{i=1}^{n} m_{ij}\left(\tilde{y}_j(t)\right) \int_{t-\rho(t)}^{t} g_i\left(\tilde{x}_j(v)\right) dv + v_j(t) + H_j(t), \end{cases} \tag{4}$$

where $u_i(t)$ and $v_j(t)$ are the negative feedback controllers to be designed.

To obtain the criterion for finite-time projection synchronization, the following assumptions must be made.

Assumption 2.1. If the nonlinear activation functions $f_j(\cdot)$ and $g_i(\cdot)$ are Lipschitz continuous on a real number field $\mathbb{R}$, then for any $x, y \in \mathbb{R}$, the following inequality condition holds:

$$|f_j(y) - f_j(x)| \le \xi_j |y - x|,$$
$$|g_i(y) - g_i(x)| \le \zeta_i |y - x|,$$

where ξ_j, $\zeta_i > 0$ are constants, with $j = 1,2,..,m, i = 1,2,...,n$.

Assumption 2.2. If $f_j(\cdot)$ and $g_i(\cdot)$ are continuous bounded functions, then for any x_j、 $y_i \in \mathbb{R}$, a constant $p_j, q_i > 0$ exists such that

$$\left|f_j\left(x_j\right)\right| \le p_j, j = 1,2\ldots\ldots m,$$

$$|g_i(y_i)| \le q_i, i = 1,2\ldots\ldots n.$$

Assumption 2.3. If the functions $I_i(t)$ and $H_j(t)$ are bounded, then for any t, nonnegative constants I_i^m, H_j^m exist such that the following conditions hold:

$$|I_i(t)| \le I_i^m, i = 1,2,\ldots,n,$$

$$\left|H_j(t)\right| \le H_j^m, j = 1,2,\ldots,m.$$

Definition 2.1. [25] A constant $t' > 0$ exists such that the error of the drive and response system satisfies that $\lim_{t \to t'} e_i^x(t) = \lim_{t \to t'} e_j^y(t) = 0$ for $\forall t \ge t'$; also, $e_i^x(t) \equiv 0$, and $e_j^y(t) \equiv 0$ hold. Thus, we claim that the drive and response system can achieve finite time synchronization, and t' is the finite synchronization time of the system.

Definition 2.2. [26] Suppose that constant $t_0(e(0)) \geq 0$ satisfies the following conditions:

$$\begin{cases} \lim_{t \to t_0(e(0))} \|e_i^x(t)\| = \lim_{t \to t_0(e(0))} \|\tilde{x}_i(t) - \alpha_i x_i(t)\| = 0, \\ \lim_{t \to t_0(e(0))} \|e_j^y(t)\| = \lim_{t \to t_0(e(0))} \|\tilde{y}_j(t) - \beta_j y_j(t)\| = 0, \end{cases}$$

where $i = 1, 2, \ldots, n$; $j = 1, 2, \ldots, m$; $|\alpha_i| \leq g, |\beta_j| \leq \hbar$; g and $\hbar$ are constants; $t_0(e(0))$ depends on the initial value $e(0)$, and $e(t) = (e_1^x(t), e_2^x(t), \ldots, e_n^x(t), e_1^y(t), e_2^y(t), \ldots, e_m^y(t))^T$. Additionally, the finite-time projection synchronization is achieved between driving system (1) and the response system (4), and α_i and β_i are the projection coefficients.

3. Synchronized control of finite-time projections of memristor-based chaotic systems

In this section, a study on the finite-time projection synchronization problem for bidirectional associative MNNs with leakage, discrete, and distributed delays is presented. Using MHDBAMNNSs (1) and (4), the form of error definition for system-corrected projection synchronization is derived as follows:

$$e_i^x(t) = \tilde{x}_i(t) - \alpha_i x_i(t), \tag{5}$$

$$e_j^y(t) = \tilde{y}_j(t) - \beta_i y_j(t), \tag{6}$$

where $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$.

$$\dot{e}_i^x(t) = \dot{\tilde{x}}_i(t) - \alpha_i \dot{x}_i(t), \tag{7}$$
$$\dot{e}_j^y(t) = \dot{\tilde{y}}_j(t) - \beta_i \dot{y}_j(t). \tag{8}$$

Theorem 3.1. To realize the projection synchronization of a memristor-based neural network, under the condition that Assumptions 2.1–2.3 hold, a delayed independent negative feedback controller based on the systematic error can be designed as follows:

$$\begin{cases} u_i(t) = -\lambda_{1i} sign(e_i^x(t)) - \lambda_{2i} e_i^x(t), \\ v_j(t) = -\mu_{1j} sign(e_j^y(t)) - \mu_{2j} e_j^y(t), \end{cases} \tag{9}$$

where $i = 1, 2, \ldots, n$, $j = 1, 2, \ldots, m$, $\lambda_{1i}, \lambda_{2i}, \mu_{1j}, \mu_{2j}$ are parameters of the controller given by (9) and the following conditions are derived:

$$\lambda_{1i} \geq (1 + g) \sum_{j=1}^{n} (a_{ji}^m + b_{ji}^m + \rho_0 c_{ji}^m) p_j + (1 + g) I_i^m,$$

$$\mu_{1j} \geq (1 + \hbar) \sum_{i=1}^{n} (k_{ij}^m + l_{ij}^m + \rho_0 m_{ij}^m) q_i - (1 + \hbar) H_j^m,$$

$$\lambda_{2i} > \delta_i + \sum_{j=1}^{n} \zeta_j (k_{ji}^m + \frac{l_{ji}^m}{1 - \tau_1} + \rho_0 m_{ji}^m),$$

$$\mu_{2j} > \omega_j + \sum_{i=1}^{m} \xi_j (a_{ij}^m + \frac{b_{ij}^m}{1 - \tau_1} + \rho_0 c_{ij}^m),$$

where $i = 1,2, \dots, n$, $j = 1,2, \dots, m$. Then, MHDBAMNNSs (1) and (4) under controller (9) can achieve projection synchronization in finite time, and the time can be calculated as follows:

$$t_0' \leq \frac{V(0)}{\varpi}, \tag{10}$$

where

$$V(0) = \sum_{i=1}^{n} |e_i^x(0)| + \sum_{i=1}^{n} \delta_i \int_{-\sigma}^{0} |e_i^x(v)| dv + \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\xi_j b_{ji}^m}{1-\tau_1} \int_{-\tau(0)}^{0} |e_j^y(v)| dv$$

$$+ \sum_{i=1}^{n} \sum_{j=1}^{n} \xi_j c_{ji}^m \int_{-\rho_0}^{0} \int_{v}^{0} |e_j^y(w)| dw dv + \sum_{j=1}^{m} |e_j^y(0)|$$

$$+ \sum_{j=1}^{m} \omega_j \int_{-\sigma}^{0} |e_j^y(v)| dv + \sum_{j=1}^{m} \sum_{i=1}^{m} \frac{\zeta_i l_{ij}^m}{1-\tau_1} \int_{-\tau(0)}^{0} |e_i^x(v)| dv$$

$$+ \sum_{j=1}^{m} \sum_{i=1}^{m} \zeta_i m_{ij}^m \int_{-\rho_0}^{0} \int_{v}^{0} |e_i^x(w)| dw dv,$$

$$\varpi = \min_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} \{\lambda_{1i} - (1+\mathcal{g}) \sum_{j=1}^{n} (a_{ji}^m + b_{ji}^m + \rho_0 c_{ji}^m) p_j - (1+\mathcal{g}) I_i^m,$$

$$\mu_{1j} - (1+\hbar) \sum_{i=1}^{n} (k_{ij}^m + l_{ij}^m + \rho_0 m_{ij}^m) q_i - (1+\hbar) H_j^m \} > 0.$$

Proof of Theorem 3.1. Consider the following Lyapunov–Krasovsky function:

$$V(t) = \sum_{k=1}^{8} V_k(t),$$

where

$$V_1(t) = \sum_{i=1}^{n} |e_i^x(t)|,$$

$$V_2(t) = \sum_{i=1}^{n} \delta_i \int_{t-\sigma}^{t} |e_i^x(v)| dv,$$

$$V_3(t) = \sum_{i=1}^{n} \sum_{j=1}^{n} \frac{\xi_j b_{ji}^m}{1-\tau_1} \int_{t-\tau(t)}^{t} |e_j^y(v)| dv,$$

$$V_4(t) = \sum_{i=1}^{n} \sum_{j=1}^{n} \xi_j c_{ji}^m \int_{-\rho_0}^{0} \int_{t+v}^{t} |e_j^y(w)| dw dv,$$

$$V_5(t) = \sum_{j=1}^{m} |e_j^y(t)|,$$

$$V_6(t) = \sum_{j=1}^{m} \omega_j \int_{t-\sigma}^{t} |e_j^y(v)| dv,$$

$$V_7(t) = \sum_{j=1}^{m} \sum_{i=1}^{m} \frac{\zeta_i l_{ij}^m}{1-\tau_1} \int_{t-\tau(t)}^{t} |e_i^x(v)| dv,$$

$$V_8(t) = \sum_{j=1}^{m} \sum_{i=1}^{m} \zeta_i m_{ij}^m \int_{-\rho_0}^{0} \int_{t+v}^{t} |e_i^x(w)| dw dv.$$

Thus, the derivative of $V_1(t)$ can be calculated as follows:

$$\dot{V}_1(t) = \sum_{i=1}^n sign(e_i^x(t))\dot{e}_i^x(t) = \sum_{i=1}^n sign(e_i^x(t))[\dot{\tilde{x}}_i(t) - \alpha_i \dot{x}_i(t)]$$

$$= \sum_{i=1}^n sign(e_i^x(t))\{-\delta_i \tilde{x}_i(t-\sigma) + \alpha_i \delta_i x_i(t-\sigma)$$

$$+ \sum_{j=1}^m a_{ji}(\tilde{x}_i(t))f_j\left(\tilde{y}_j(t)\right) - \sum_{j=1}^m a_{ji}(x_i(t))f_j\left(\beta_j y_j(t)\right) + \sum_{j=1}^m a_{ji}(x_i(t))f_j\left(\beta_j y_j(t)\right)$$

$$- \alpha_i \sum_{j=1}^m a_{ji}(x_i(t))f_j\left(y_j(t)\right) + \sum_{j=1}^m b_{ji}(\tilde{x}_i(t)) \cdot f_j\left(\tilde{y}_j(t-\tau(t))\right)$$

$$- \sum_{j=1}^m b_{ji}(x_i(t))f_j\left(\beta_j y_j(t-\tau(t))\right) + \sum_{j=1}^m b_{ji}(x_i(t))f_j\left(\beta_j y_j(t-\tau(t))\right)$$

$$- \alpha_i \sum_{j=1}^m b_{ji}(x_i(t)) f_j\left(y_j(t-\tau(t))\right) + \sum_{j=1}^m c_{ji}(\tilde{x}_i(t)) \int_{t-\rho(t)}^t f_j\left(\tilde{y}_j(v)\right) dv$$

$$- \sum_{j=1}^m c_{ji}(x_i(t)) \int_{t-\rho(t)}^t f_j\left(\beta_j y_j(v)\right) dv + \sum_{j=1}^m c_{ji}(x_i(t)) \int_{t-\rho(t)}^t f_j\left(\beta_j y_j(v)\right) dv$$

$$- \alpha_i \sum_{j=1}^m c_{ji}(x_i(t)) \int_{t-\rho(t)}^t f_j\left(y_j(v)\right) dv + (1-\alpha_i)I_i(t) + u_i(t)\}$$

$$= \sum_{i=1}^n sign(e_i^x(t)) \cdot \{-\delta_i e_i^x(t-\sigma) + \sum_{j=1}^m a_{ji}^x(t)\Lambda_j\left(e_j^y(t)\right) + \sum_{j=1}^m a_{ji}^x(t)f_j\left(\beta_j y_j(t)\right)$$

$$- \alpha_i \sum_{j=1}^m a_{ji}^x(t)f_j\left(y_j(t)\right) + \sum_{j=1}^m b_{ji}^x(t)\Lambda_j\left(e_j^y(t-\tau(t))\right)$$

$$+ \sum_{j=1}^m b_{ji}^x(t)f_j\left(\beta_j y_j(t-\tau(t))\right) - \alpha_i \sum_{j=1}^m b_{ji}^x(t) \cdot f_j\left(y_j(t-\tau(t))\right)$$

$$+ \sum_{j=1}^m c_{ji}^x(t) \int_{t-\rho(t)}^t \Lambda_j\left(e_j^y(v)\right) dv + \sum_{j=1}^m c_{ji}^x(t) \int_{t-\rho(t)}^t f_j\left(\beta_j y_j(v)\right) dv$$

$$- \alpha_i \sum_{j=1}^m c_{ji}^x(t) \int_{t-\rho(t)}^t f_j\left(y_j(v)\right) dv + (1-\alpha_i)I_i(t) + u_i(t)\},$$

where $\Lambda_j(e_j^y(t)) = f_j(\tilde{y}_j(t)) - f_j(\beta_j y_j(t))$, $i = 1,2,\ldots,n$ and $j = 1,2,\ldots,m$. Given that Assumptions 2.1–2.3 hold, each term of $\dot{V}_1(t)$ can be computed separately by using the linear matrix inequality method, as follows:

$$sign(e_i^x(t))[-\delta_i e_i^x(t-\sigma)] \le \delta_i |e_i^x(t-\sigma)|,$$

$$sign(e_i^x(t))[\sum_{j=1}^m a_{ji}^x(t)\Lambda_j\left(e_j^y(t)\right)] \le \sum_{j=1}^m |a_{ji}^x(t)|\xi_j|e_j^y(t)|) \le \sum_{j=1}^m a_{ji}^m \xi_j |e_j^y(t)|,$$

$$sign(e_i^x(t))[\sum_{j=1}^m b_{ji}^x(t)\Lambda_j\left(e_j^y(t-\tau(t))\right)] \le \sum_{j=1}^m |b_{ji}^x(t)|\xi_j|e_j^y(t-\tau(t))|)$$

$$\le \sum_{j=1}^m b_{ji}^m \xi_j |e_j^y(t-\tau(t))|.$$

The model precondition $0 \le \rho(t) \le \rho_0$, combined with Assumption 2.1, yields the following conclusion:

$$sign(e_i^x(t)) \left[\sum_{j=1}^m c_{ji}^x(t) \int_{t-\rho(t)}^t \Lambda_j\left(e_j^y(v)\right) dv\right] \le \sum_{j=1}^m |c_{ji}^x(t)| \int_{t-\rho(t)}^t \xi_j |e_j^y(v)|dv$$

$$\le \sum_{j=1}^m c_{ji}^m \xi_j \int_{t-\rho_0}^t |e_j^y(v)|dv.$$

Under Assumption 2.2, the linear matrix inequality method yields

$$sign(e_i^x(t)) \left[\sum_{j=1}^m a_{ji}^x(t) f_j \left(\beta_j y_j(t) \right) \right] \le \sum_{j=1}^m \gamma_i |a_{ji}^x(t)| \left| f_j \left(\beta_j y_j(t) \right) \right|$$

$$\le \sum_{j=1}^m \gamma_i |a_{ji}^x(t)| p_j \le \sum_{j=1}^m a_{ji}^m p_j \gamma_i,$$

where $e_i^x(t) = 0$ at $\gamma_i = 0$; otherwise, $\gamma_i = 1$. It can be deduced as follows:

$$sign(e_i^x(t))[-\alpha_i \sum_{j=1}^m a_{ji}^x(t) f_j(y_j(t))] \le |\alpha_i| \sum_{j=1}^m \gamma_i |a_{ji}^x(t)| p_j \le |\alpha_i| \sum_{j=1}^m a_{ji}^m p_j \gamma_i,$$

$$sign(e_i^x(t))[\sum_{j=1}^m b_{ji}^x(t) f_j \left(\beta_j y_j(t - \tau(t)) \right)] \le \sum_{j=1}^m \gamma_i |b_{ji}^x(t)| p_j \le \sum_{j=1}^m b_{ji}^m p_j \gamma_i,$$

$$sign(e_i^x(t))[-\alpha_i \sum_{j=1}^m b_{ji}^x(t) \cdot f_j(y_j(t - \tau(t)))] \le |\alpha_i| \sum_{j=1}^m \gamma_i |b_{ji}^x(t)| p_j \le |\alpha_i| \sum_{j=1}^m b_{ji}^m p_j \gamma_i,$$

$$sign(e_i^x(t)) \left[\sum_{j=1}^m c_{ji}^x(t) \int_{t-\rho(t)}^t f_j \left(\beta_j y_j(v) \right) dv \right] \le \sum_{j=1}^m \gamma_i |c_{ji}^x(t)| \int_{t-\rho_0}^t \left| f_j \left(\beta_j y_j(t) \right) \right| dv$$

$$\le \rho_0 \sum_{j=1}^m c_{ji}^m p_j \gamma_i,$$

$$sign(e_i^x(t))[-\alpha_i \sum_{j=1}^m c_{ji}^x(t) \int_{t-\rho(t)}^t f_j(y_j(v)) dv] \le \rho_0 |\alpha_i| \sum_{j=1}^m \gamma_i |c_{ji}^x(t)| \int_{t-\rho_0}^t \left| f_j \left(y_j(v) \right) \right| dv$$

$$\le \rho_0 |\alpha_i| \sum_{j=1}^m c_{ji}^m p_j \gamma_i.$$

Additionally, Assumption 2.3 yields

$$sign(e_i^x(t))(1 - \alpha_i) I_i(t) \le \left| sign(e_i^x(t)) \right| |(1 - \alpha_i) I_i(t)| \le |1 - \alpha_i| I_i^m \gamma_i,$$
$$sign(e_i^x(t)) u_i(t) = sign(e_i^x(t))[-\lambda_{1i} sign(e_i^x(t)) - \lambda_{2i} e_i^x(t)] = -\lambda_{1i} \gamma_i - \lambda_{2i} |e_i^x(t)|.$$

Thus, the derivatives of $V_2(t)$, $V_3(t)$, and $V_4(t)$ can be calculated as follows:

$$\dot{V}_2(t) = \sum_{i=1}^n \delta_i [|e_i^x(t)| - |e_i^x(t - \sigma)|],$$

$$\dot{V}_3(t) = \sum_{i=1}^n \sum_{j=1}^n \frac{\xi_j b_{ji}^m}{1-\tau_1} [|e_j^y(t)| - (1 - \dot{\tau}(t))|e_i^y(t - \tau(t))|]$$

$$\le \sum_{i=1}^n \sum_{j=1}^n \frac{\xi_j b_{ji}^m}{1-\tau_1} |e_j^y(t)| - \sum_{i=1}^n \sum_{j=1}^n \xi_j b_{ji}^m |e_j^y(t - \tau(t))|,$$

$$\dot{V}_4(t) = \sum_{i=1}^n \sum_{j=1}^n \xi_j c_{ji}^m \int_{-\rho_0}^0 [|e_j^y(t)| - |e_j^y(t + v)|] dv$$

$$= \rho_0 \sum_{i=1}^n \sum_{j=1}^n \xi_j c_{ji}^m |e_j^y(t)| - \sum_{i=1}^n \sum_{j=1}^n \xi_j c_{ji}^m \int_{t-\rho_0}^t |e_j^y(v)| dv.$$

Given that $|\alpha_i| \le \mathcal{g}$ and $|\beta_j| \le \hbar$, in summary, it can be obtained that

$$\dot{V}_1(t) + \dot{V}_2(t) + \dot{V}_3(t) + \dot{V}_4(t)$$

$$\le \sum_{i=1}^n \sum_{j=1}^n [\xi_j (a_{ji}^m + \frac{b_{ji}^m}{1-\tau_1} + \rho_0 c_{ji}^m)] |e_j^y(t)| + \sum_{i=1}^n (\delta_i - \lambda_{2i}) |e_i^x(t)|$$

$$- \sum_{i=1}^n [\lambda_{1i} - (1 + \mathcal{g}) \sum_{j=1}^n (a_{ji}^m + b_{ji}^m + \rho_0 c_{ji}^m) p_j - (1 + \mathcal{g}) |I_i^m|] \gamma_i.$$

Similarly, it can be determined that

$$\dot{V}_4(t) + \dot{V}_5(t) + \dot{V}_6(t) + \dot{V}_7(t)$$

$$\leq \sum_{j=1}^{n} \sum_{i=1}^{n} [\zeta_i(k_{ij}^m + \frac{l_{ij}^m}{1-\tau_1} + \rho_0 m_{ij}^m)]|e_i^x(t)| + \sum_{j=1}^{n}(\omega_j - \mu_{2j})|e_j^y(t)|$$

$$- \sum_{j=1}^{n}[\mu_{1j} - (1+\hbar)\sum_{i=1}^{n}(k_{ij}^m + l_{ij}^m + \rho_0 m_{ij}^m)q_i - (1+\hbar)H_j^m]\gamma_j,$$

where $e_j^y(t) = 0$ at $\gamma_j = 0$; otherwise, $\gamma_j = 1$. Thus, we obtain

$$\dot{V}(t) \leq \sum_{j=1}^{n} \sum_{i=1}^{n} [\delta_i - \lambda_{2i} + \zeta_i(k_{ij}^m + \frac{l_{ij}^m}{1-\tau_1} + \rho_0 m_{ij}^m)]|e_i^x(t)|$$

$$- \sum_{i=1}^{n}[\lambda_{1i} - (1+g)\sum_{j=1}^{n}(a_{ji}^m + b_{ji}^m + \rho_0 c_{ji}^m)p_j - (1+g)I_i^m]\gamma_i$$

$$+ \sum_{i=1}^{n} \sum_{j=1}^{n} [\omega_j - \mu_{2j} + \xi_j(a_{ji}^m + \frac{b_{ji}^m}{1-\tau_1} + \rho_0 c_{ji}^m)]|e_j^y(t)|$$

$$- \sum_{j=1}^{n}[\mu_{1j} - (1+\hbar)\sum_{i=1}^{n}(k_{ij}^m + l_{ij}^m + \rho_0 m_{ij}^m)q_i - (1+\hbar)H_j^m]\gamma_j.$$

Given that λ_{1i}, λ_{2i}, μ_{1j}, and μ_{2j} satisfy the conditions in Theorem 3.1, $V(t)$ can be derived as follows:

$$\dot{V}(t) \leq - \sum_{i=1}^{n}[\lambda_{1i} - (1+g)\sum_{j=1}^{n}(a_{ji}^m + b_{ji}^m + \rho_0 c_{ji}^m)p_j - (1+g)I_i^m]\gamma_i$$

$$- \sum_{j=1}^{n}[\mu_{1j} - (1+\hbar)\sum_{i=1}^{n}(k_{ij}^m + l_{ij}^m + \rho_0 m_{ij}^m)q_i - (1+\hbar)H_j^m]\gamma_j$$

$$\leq -\varpi(\sum_{i=1}^{n} \gamma_i + \sum_{j=1}^{m} \gamma_j),$$

where $\varpi = \min_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} \{\lambda_{1i} - (1+g)\sum_{j=1}^{n}(a_{ji}^m + b_{ji}^m + \rho_0 c_{ji}^m)p_j - (1+g)I_i^m,$

$$\mu_{1j} - (1+\hbar)\sum_{i=1}^{n}(k_{ij}^m + l_{ij}^m + \rho_0 m_{ij}^m)q_i - (1+\hbar)H_j^m\} > 0.$$

When $||e_i^x(t)|| \neq 0$ and $||e_j^y(t)|| \neq 0$, $\dot{V}(t) \leq -\varpi$. Finite-time synchronization analysis of Definition 2.1 indicates that for $\forall t \in [0, t_0]$, $V(t_0) - V(0) \leq -\varpi t_0$. The above-mentioned properties of $V(t)$ suggest that a constant $t_0 = V(0)/\varpi$ exists, which satisfies that $\lim_{t \to t_0} ||e(t)|| = 0$ and $||e(t)|| \equiv 0$, $\forall t \geq t_0$.

By setting $t_0' = \inf\{t \in (0, t_0]: ||e(t')|| \equiv 0, t' > t\}$, we obtain that $\lim_{t \to t_0'} ||e(t)|| = 0$ and $||e(t)|| \equiv 0$ for $\forall t \geq t_0'$. The MHDBAMNNs (1) and (4) can be synchronized with finite-time projections according to Definition 2.2. Thus, the proof of Theorem 3.1 is complete.

4. Numerical simulations

In this section, numerical simulation examples are provided to demonstrate the validity of the obtained results and their potential applications in image encryption. Here, MHDBAMNNs based on Eqs (11) and (12) as the drive system and its corresponding response system with $m = 2$ and $n = 2$, respectively, are considered as follows:

$$
\begin{cases}
\dot{x}_i(t) = -\delta_i x_i(t-\sigma) + \sum_{j=1}^{2} a_{ji}(x_i(t)) f_j\left(y_j(t)\right) + \sum_{j=1}^{2} b_{ji}(x_i(t)) \cdot f_j\left(y_j(t-\tau(t))\right) \\
\qquad + \sum_{j=1}^{2} c_{ji}(x_i(t)) \int_{t-\rho(t)}^{t} f_j\left(y_j(v)\right) dv + I_i(t), \\
\dot{y}_j(t) = -\omega_j y_j(t-\sigma) + \sum_{i=1}^{2} k_{ij}\left(y_j(t)\right) g_j\left(x_j(t)\right) + \sum_{i=1}^{2} l_{ij}\left(y_j(t)\right) \cdot g_j\left(x_j(t-\tau(t))\right) \\
\qquad + \sum_{i=1}^{2} m_{ij}\left(y_j(t)\right) \int_{t-\rho(t)}^{t} g_j\left(x_j(v)\right) dv + H_j(t),
\end{cases}
\tag{11}
$$

$$
\begin{cases}
\dot{\tilde{x}}_i(t) = -\delta_i \tilde{x}_i(t-\sigma) + \sum_{j=1}^{2} a_{ji}(\tilde{x}_i(t)) f_j\left(\tilde{y}_j(t)\right) + \sum_{j=1}^{2} b_{ji}(\tilde{x}_i(t)) \cdot f_j\left(\tilde{y}_j(t-\tau(t))\right) \\
\qquad + \sum_{j=1}^{2} c_{ji}(\tilde{x}_i(t)) \int_{t-\rho(t)}^{t} f_j\left(\tilde{y}_j(v)\right) dv + u_i(t) + I_i(t), \\
\dot{\tilde{y}}_j(t) = -\omega_j \tilde{y}_j(t-\sigma) + \sum_{i=1}^{2} k_{ij}\left(\tilde{y}_j(t)\right) g_j\left(\tilde{x}_j(t)\right) + \sum_{i=1}^{2} l_{ij}\left(\tilde{y}_j(t)\right) \cdot g_j\left(\tilde{x}_j(t-\tau(t))\right) \\
\qquad + \sum_{i=1}^{2} m_{ij}\left(\tilde{y}_j(t)\right) \int_{t-\rho(t)}^{t} g_j\left(\tilde{x}_j(v)\right) dv + v_i(t) + H_j(t),
\end{cases}
\tag{12}
$$

where $\delta_1 = 2.5$, $\delta_2 = 2.2$, $\omega_1 = 2.3$, and $\omega_2 = 2.1$, with the following memristor parameters:

$$a_{11}(\gamma) = \begin{cases} 0.1, |\gamma| < 1, \\ -0.8, |\gamma| > 1, \end{cases} \qquad a_{21}(\gamma) = \begin{cases} 0.9, |\gamma| < 1, \\ -0.5, |\gamma| > 1, \end{cases}$$

$$a_{12}(\gamma) = \begin{cases} 1.8, |\gamma| < 1, \\ 0.9, |\gamma| > 1, \end{cases} \qquad a_{22}(\gamma) = \begin{cases} -0.3, |\gamma| < 1, \\ 0.6, |\gamma| > 1, \end{cases}$$

$$b_{11}(\gamma) = \begin{cases} 2.3, |\gamma| < 1, \\ 1.7, |\gamma| > 1, \end{cases} \qquad b_{21}(\gamma) = \begin{cases} 1.2, |\gamma| < 1, \\ -1.9, |\gamma| > 1, \end{cases}$$

$$b_{12}(\gamma) = \begin{cases} 2.8, |\gamma| < 1, \\ -1.2, |\gamma| > 1, \end{cases} \qquad b_{22}(\gamma) = \begin{cases} -0.3, |\gamma| < 1, \\ 0.5, |\gamma| > 1, \end{cases}$$

$$c_{11}(\gamma) = \begin{cases} 0.6, |\gamma| < 1, \\ -1.5, |\gamma| > 1, \end{cases} \qquad c_{21}(\gamma) = \begin{cases} -0.7, |\gamma| < 1, \\ 1.3, |\gamma| > 1, \end{cases}$$

$$c_{12}(\gamma) = \begin{cases} -1.5, |\gamma| < 1, \\ -0.9, |\gamma| > 1, \end{cases} \qquad c_{22}(\gamma) = \begin{cases} 1, |\gamma| < 1, \\ -0.3, |\gamma| > 1, \end{cases}$$

$$k_{11}(\gamma) = \begin{cases} -2.2, |\gamma| < 1, \\ 0.7, |\gamma| > 1, \end{cases} \qquad k_{21}(\gamma) = \begin{cases} -2.7, |\gamma| < 1, \\ -0.8, |\gamma| > 1, \end{cases}$$

$$k_{12}(\gamma) = \begin{cases} 1, |\gamma| < 1, \\ -0.6, |\gamma| > 1, \end{cases} \qquad k_{22}(\gamma) = \begin{cases} 1.2, |\gamma| < 1, \\ 0.5, |\gamma| > 1, \end{cases}$$

$$l_{11}(\gamma) = \begin{cases} 0.75, |\gamma| < 1, \\ -1, |\gamma| > 1, \end{cases} \qquad l_{21}(\gamma) = \begin{cases} 2.7, |\gamma| < 1, \\ 2, |\gamma| > 1, \end{cases}$$

$$l_{12}(\gamma) = \begin{cases} -1.4, |\gamma| < 1, \\ 0.5, |\gamma| > 1, \end{cases} \qquad l_{22}(\gamma) = \begin{cases} 2.8, |\gamma| < 1, \\ -1, |\gamma| > 1, \end{cases}$$

$$m_{11}(\gamma) = \begin{cases} -0.3, |\gamma| < 1, \\ 1.5, |\gamma| > 1, \end{cases} \qquad m_{21}(\gamma) = \begin{cases} -1.8, |\gamma| < 1, \\ 0.7, |\gamma| > 1, \end{cases}$$

$$m_{12}(\gamma) = \begin{cases} 0.2, |\gamma| < 1, \\ 1.1, |\gamma| > 1, \end{cases} \qquad m_{22}(\gamma) = \begin{cases} 0.1, |\gamma| < 1, \\ -1.9, |\gamma| > 1. \end{cases}$$

Given these parameters, the activation functions are denoted by $f_1(\gamma) = f_2(\gamma) = g_1(\gamma) = g_2(\gamma) = (|\gamma + 1| - |\gamma - 1|)/2$; $\sigma = 0.2$, $\tau(t) = 0.3\cos(t) + 0.5$, $\rho(t) = 0.6\sin(t) + 0.9$; $I_1(t) = \sin(t)$, $I_2(t) = \cos(t)$, $H_1(t) = \cos(2t)$, and $H_2(t) = \sin(2t)$ are the external inputs. Thus, the constraints were computed to obtain $\tau_0 = 0.8$, $\rho_0 = 1.5$, and $\tau_1 = 0.8$. According to Assumptions 2.2 and 2.3, $p_1 = p_2 = q_1 = q_2 = 1$, $I_1 = I_2 = H_1 = H_2 = 1$, the initial values of the system are given by $\phi(\chi) = (0.8, -1.3)^T$, $\psi(\chi) = (-1,0.5)^T$, $\tilde{\phi}(\chi) = (-0.6,1.1)^T$, and $\tilde{\psi}(\chi) = (0.5, -1)^T$, and the controller coefficients were calculated as $\lambda_{11} \geq 18.87$, $\lambda_{12} \geq 17.765$, $\mu_{11} \geq 24.735$, $\mu_{12} \geq 20.23$, and $\lambda_{21} > 30.85$, $\lambda_{22} > 29.9$, $\mu_{21} > 29.2$, $\mu_{22} > 24.75$.

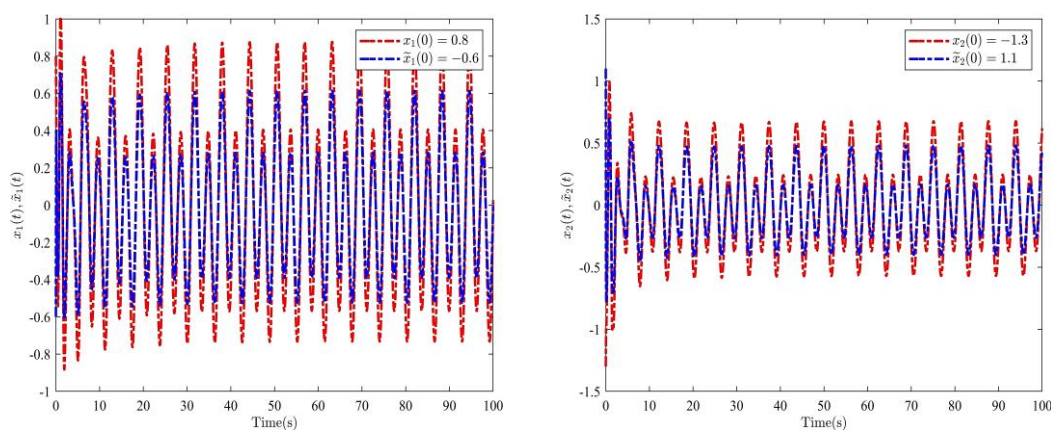The dynamic trajectories of states x and y in the system are shown in Figures 1 and 2, respectively.

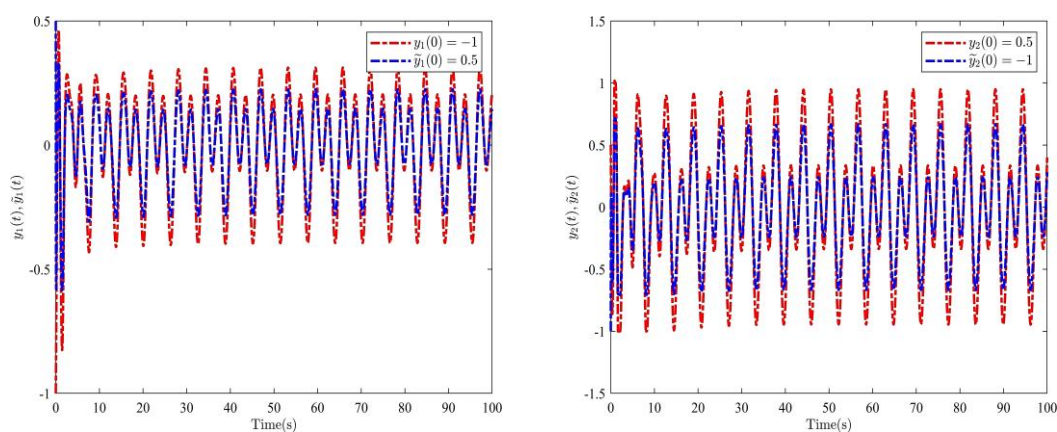**Figure 1.** The curves for state x in the drive-response system.

Figure 2. The curves for state y in the drive-response system.

Figures 3(b) and 4(b) show the synchronization errors curves $e^x(t)$ and $e^y(t)$, respectively, with projection coefficients $\alpha_1 = 0.7$, $\alpha_2 = 0.7$, $\beta_1 = 0.7$, and $\beta_2 = 0.7$. The controller coefficients $\lambda_{11} = 20$, $\lambda_{12} = 19$, $\mu_{11} = 26$, $\mu_{12} = 21$ and $\lambda_{21} = 32$, $\lambda_{22} = 31.2$, $\mu_{21} = 30.5$, $\mu_{22} = 26$ were applied under the control of (9). A comparison of the state trajectories of the synchronization errors

$e^x(t)$ and $e^y(t)$ are shown in Figure 4(a) without controller (9), respectively, revealing the uncontrolled error system is unstable. Accordingly, the driving system and response system were not synchronized. The synchronization simulation results presented in Figures 3(b) and 4(b) indicate that the synchronization errors $e^x(t)$ and $e^y(t)$ converge to zero within a finite time period of 0.32 seconds. Additionally, the two error systems can be quickly stabilized to the zero point; thus, a finite time projection synchronization could be achieved between systems (11) and (12). Furthermore, the theoretical estimation of the upper bound of the time was determined to be $t_0' > 5.72$ seconds to reach synchronization, and the time of the experimental synchronization was found to be less than the theoretically estimated time t_0', which verifies the validity of the finite-time synchronization theory adopted in Theorem 3.1.
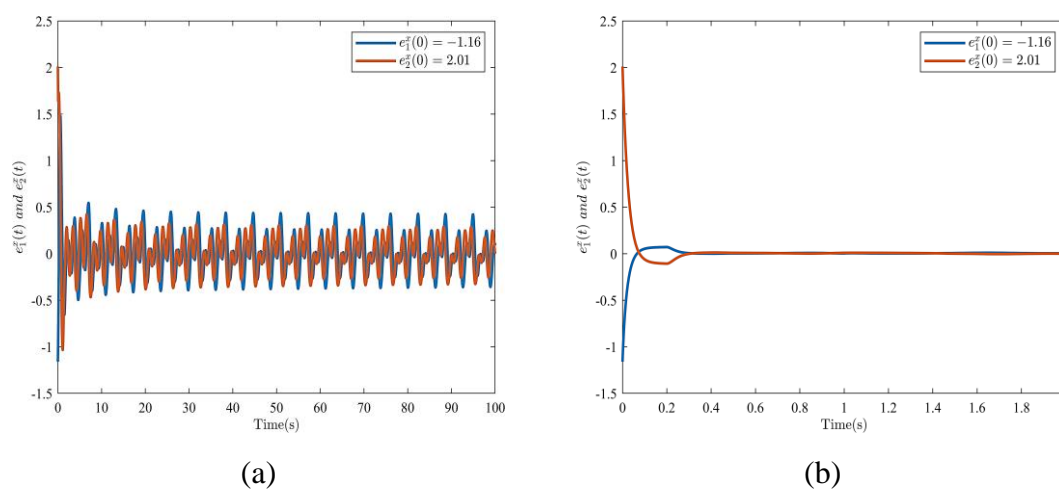
Figure 3. (a) Synchronization error $e^x(t)$ without control. (b) Synchronization error $e^x(t)$ under controller (9).
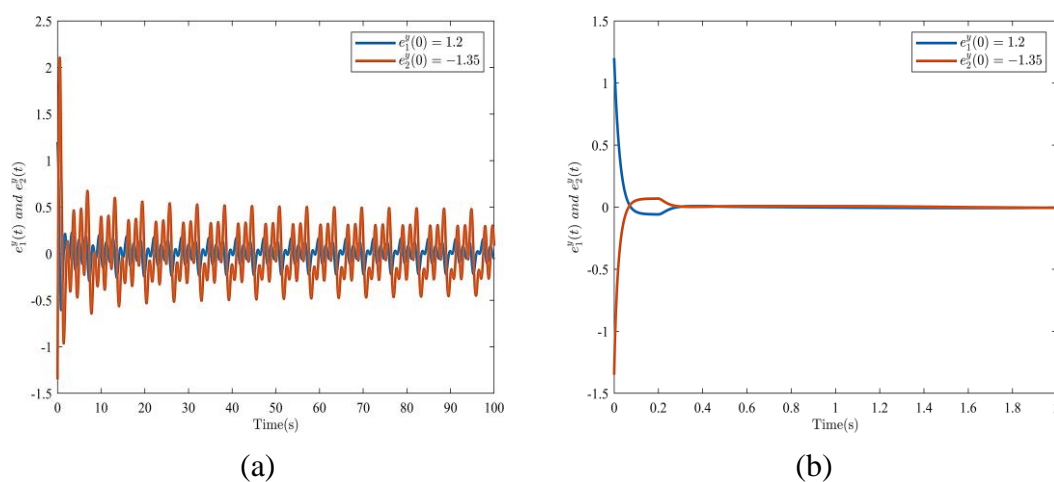

Figure 4. (a) Synchronization error $e^y(t)$ without control. (b) Synchronization error $e^y(t)$ under controller (9).

The chaotic attractors of the systems are illustrated in Figure 5. Evidently, Eqs (11) and (12) can be effectively applied to chaotic image algorithms.
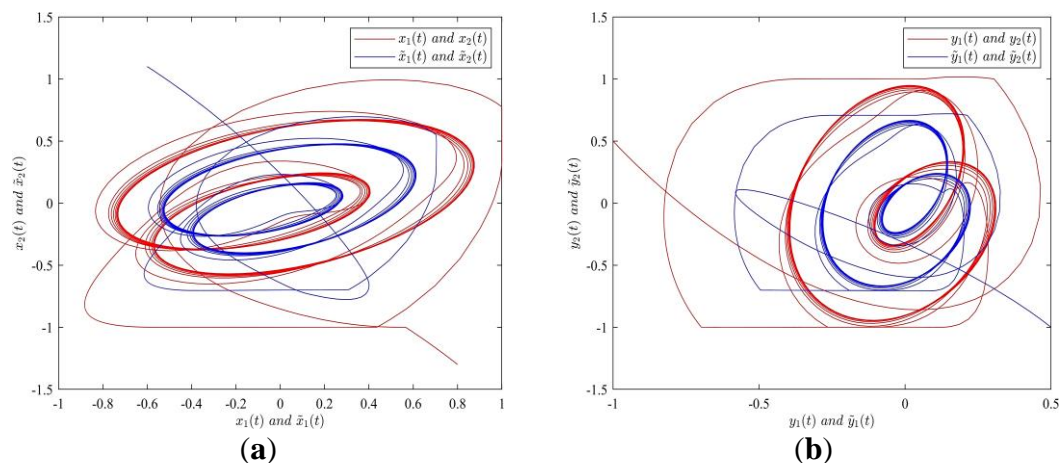
Figure 5. Chaotic attractors of systems (11) and (12).

5. Image encryption algorithms and statistical performance analysis

Since MHDBAMNNs have large chaotic attractors, the above simulation results can be applied to image encryption experiments. Consider the color image P as the experimental object, whose size is $m \times n \times 3$. First, P is divided into two-dimensional matrices comprising red, green and blue components, which greatly reduces the required time and space resources. Second, DNA coding and operations are performed on each 2D matrix chunk; also, after encryption, row and column substitution is performed again to enhance the anti-cropping property, and finally the three channels are merged to obtain the encrypted image. By implementing the MHDBAMNN synchronous simulation algorithm in the drive system (11), we generated three chaotic sequences x_1, y_1 and y_2, where the product of x_1 and the average gray value of the three channels is used as a random chaos matrix for the DNA encryption operation on original image, The index sequences obtained by multiplying the average grayscale values of channels y_1 and y_2 with the red and green channels, respectively, determine the permutation of rows and columns after encryption to enhance the scrambling effect of the ciphertext image and resist differential attacks. The initial values of Chen's hyperchaotic system are associated with the original image. The generated four chaotic sequences determine the DNA encoding, decoding, and computational rules for each small block of the image, significantly enhancing the system's resistance to attacks.

The encryption steps are as follows:

1) Split P into three 2D pixel matrices, denoted as $P_R(i,j), P_G(i,j)$ and $P_B(i,j)$, according to Eq (11), where $i \in \{1, 2, \dots, n\}$ and $j \in \{1, 2, \dots, m\}$. Divide the matrices $P_R(i,j), P_G(i,j)$ and $P_B(i,j)$ into evenly sized blocks of $t \times t$, where $t = 4$.

2) In the synchronous simulation example of the MHDBAMNN, the chaotic sequence x_1 generated by the drive system (11) is multiplied by the average grayscale values I_1 of the three channels. This result is then transformed into a random chaotic matrix Q of size $m \times n$, where the initial value of sequence $m \times n$ and I_1 serve as one of the encryption keys.

$$I_1 = sum(P_R(:) + P_G(:) + P_B(:))/(255 \times 3 \times m \times n). \tag{13}$$

3) Set the initial values and parameters for the Chen hyperchaotic system. Utilize Matlab built-in functions to compute four sequences $\{X_i\}$, $\{Y_i\}$, $\{Z_i\}$, $\{H_i\}$, where $i \in \{1,2, \dots, m \times n/t^2\}$. Specifically, the initial value $X(0)$ is obtained by the *bitand* operation between $P_R(i,j)$ and the 1st and 5th bit planes, $Y(0)$ is derived from $P_G(i,j)$ with the 2nd and 6th, $Z(0)$ is calculated by using $P_B(i,j)$ with the 3rd and 7th, and $H(0)$ is derived from $P_R(i,j)$ with the 4th and 8th, as specified in Eqs (14)–(17). Different plaintext images have different initial values; hence, these four initial values also serve as one of the encryption keys.

$$X(0) = sum(sum(bitand(P_R, 17))) / (17 \times m \times n), \tag{14}$$
$$Y(0) = sum(sum(bitand(P_G, 34))) / (34 \times m \times n), \tag{15}$$
$$Z(0) = sum(sum(bitand(P_B, 68))) / (68 \times m \times n), \tag{16}$$
$$H(0) = sum(sum(bitand(P_R, 136))) / (136 \times m \times n). \tag{17}$$

4) Apply the same DNA coding scheme to the corresponding sub-blocks of $P_R(i,j)$, $P_G(i,j)$ and $P_B(i,j)$ at the same positions to enhance encryption efficiency. The coding is represented by A, G, C, and T, determined by the sequence $\{X_i\}$. The encoding scheme for each sub-block of the Q matrix is determined by $\{Y_i\}$. Apply the same operation rules, determined by $\{Z_i\}$, between the coded $P_R(i,j)$, $P_G(i,j)$ and $P_B(i,j)$ and the corresponding blocks of Q. Here, $Z_i = 1$ represents addition, $Z_i = 2$ represents subtraction, $Z_i = 3$ represents *bitxor* operation, and $Z_i = 4$ represents the *bitand* operation. The specific algorithm is outlined in Eqs (18)–(20).

$$X = mod(round(X \times 10^4), 8) + 1, \tag{18}$$
$$Y = mod(round(Y \times 10^4), 8) + 1, \tag{19}$$
$$Z = mod(round(Z \times 10^4), 3) + 1. \tag{20}$$

5) In addition to the first sub-block, the encryption results of the previous sub-block and the current sub-block are again subjected to DNA operations according to $\{Z_i\}$ to obtain better diffusion results.

6) Decoding is the inverse process of encoding, where the matrix chunks after DNA operations are DNA decoded by the sequence $\{H_i\}$, i.e., A, G, C, T are decoded into specific values.

7) To perform row substitution and column substitution on the matrix of the three channels after DNA decoding, the index sequence K_1 after multiplying the chaotic sequence y_1 generated by the driving system (11) in the synchronous simulation algorithm of the MHDBAMNNs with the average grey value I_2 of the red channel is multiplied by the index sequence K_1; furthermore, the exchange coordinates of K_1 and its corresponding index as the rows are used to determine the post-cryptographic row substitution, and the index sequence K_2 after multiplying the chaotic sequence y_1 generated by the driving system (11) in the synchronous simulation algorithm of the MHDBAMNNs is multiplied by the average grey value I_3 of the green channel. The index sequence K_2 after multiplication, with K_2 and its corresponding index as the exchange coordinates of columns is used to determine the encrypted column replacement; the lengths K_1 and K_2 are denoted by m and n, respectively, as shown in Eqs (21)–(26), where the initial values of the sequences y_1 and y_2 and I_2 and I_3 constitute one of the keys.

$$I_2 = sum(P_R(:))/(255 \times m \times n), \tag{21}$$

$$I_3 = sum(P_G(:))/(255 \times m \times n), \tag{22}$$
$$U_1 = y_1 \times I_2, \tag{23}$$
$$U_2 = y_2 \times I_3, \tag{24}$$
$$[\sim, K_1] = sort(U_1, 'descend'), \tag{25}$$
$$[\sim, K_2] = sort(U_2, 'descend'). \tag{26}$$

8) Combine the three two-dimensional matrices obtained after row and column substitution into a three-dimensional matrix to obtain the ciphertext image.

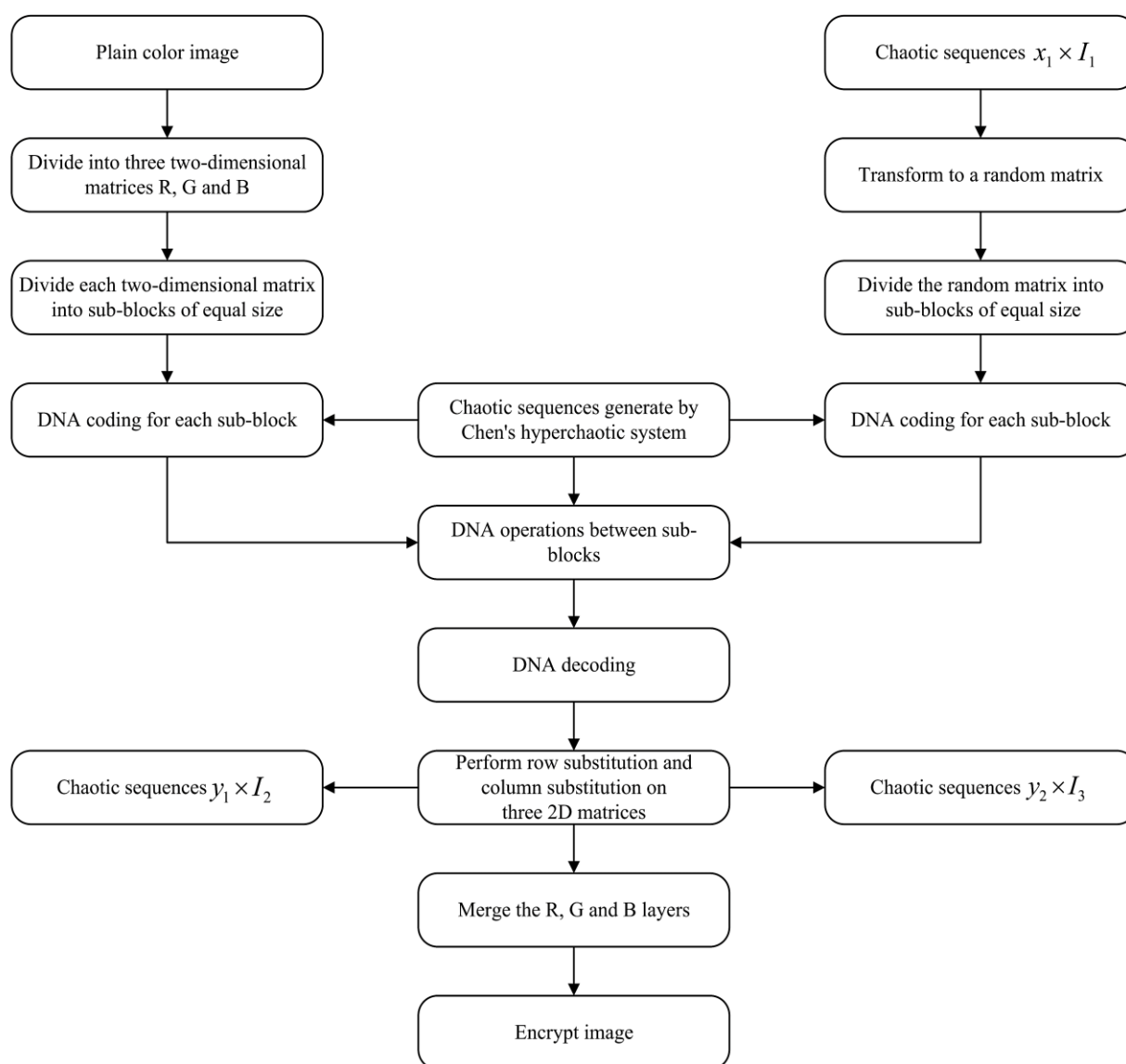The encryption algorithm flowchart is as shown in Figure 6.

Figure 6. Encryption algorithm flowchart.

With the help of the key, we encrypted the color baboon image, the encryption result and decryption result are shown in Figure 7.
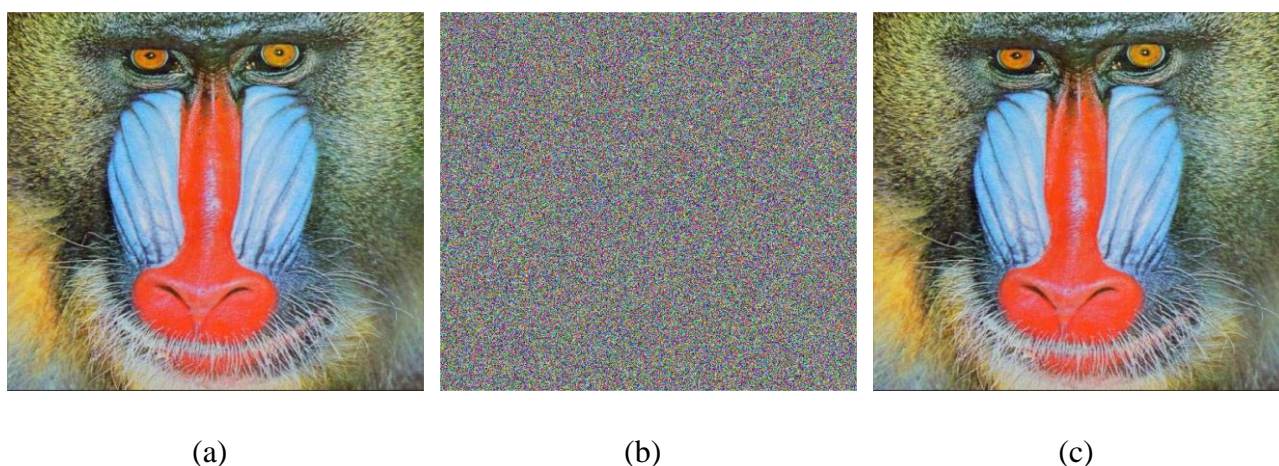

(a)   (b)   (c)

Figure 7. Scheme results: (a) original image; (b) encrypted image; (c) decrypted image.

5.1. Histogram analysis

The original, encrypted and decrypted images are shown in Figure 8. Evidently, the uniform distribution of the encrypted image indicates that it does not display any information about the original image.
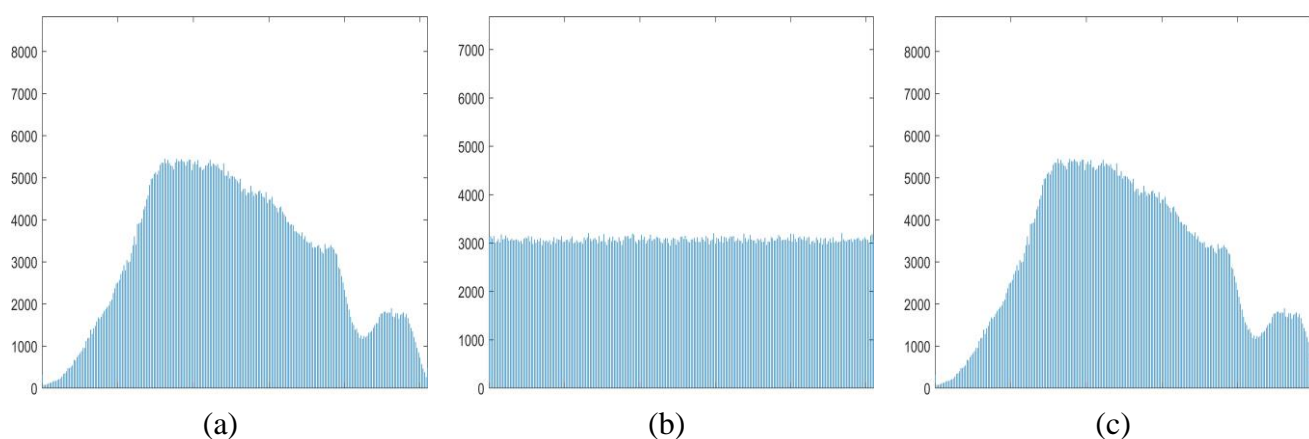

(a)   (b)   (c)

Figure 8. Histogram analysis: (a) histogram of original image; (b) histogram of encrypted image; (c) histogram of decrypted image.

5.2. Correlation coefficient analysis

Correlation analysis is a very important image evaluation index; the calculation method for the correlation coefficient was employed as described in [16].

The correlation between neighboring pixels in the original and encrypted images in different directions (horizontal, vertical, and diagonal) is shown in Figure 9. A strong correlation between

neighboring pixels in the original image can be observed in Figure 9(a–c), whereas the correlation between neighboring pixels in the encrypted image, shown in Figure 9(d–e), was found to be weak. This is further evidenced by the data in Table 1. The weak correlation of encrypted images demonstrates the potential application of MHDBAMNNs for image encryption; thus, the validity of Theorem 3.1 is verified.

Table 1. The correlation coefficients for the original image and encrypted image.

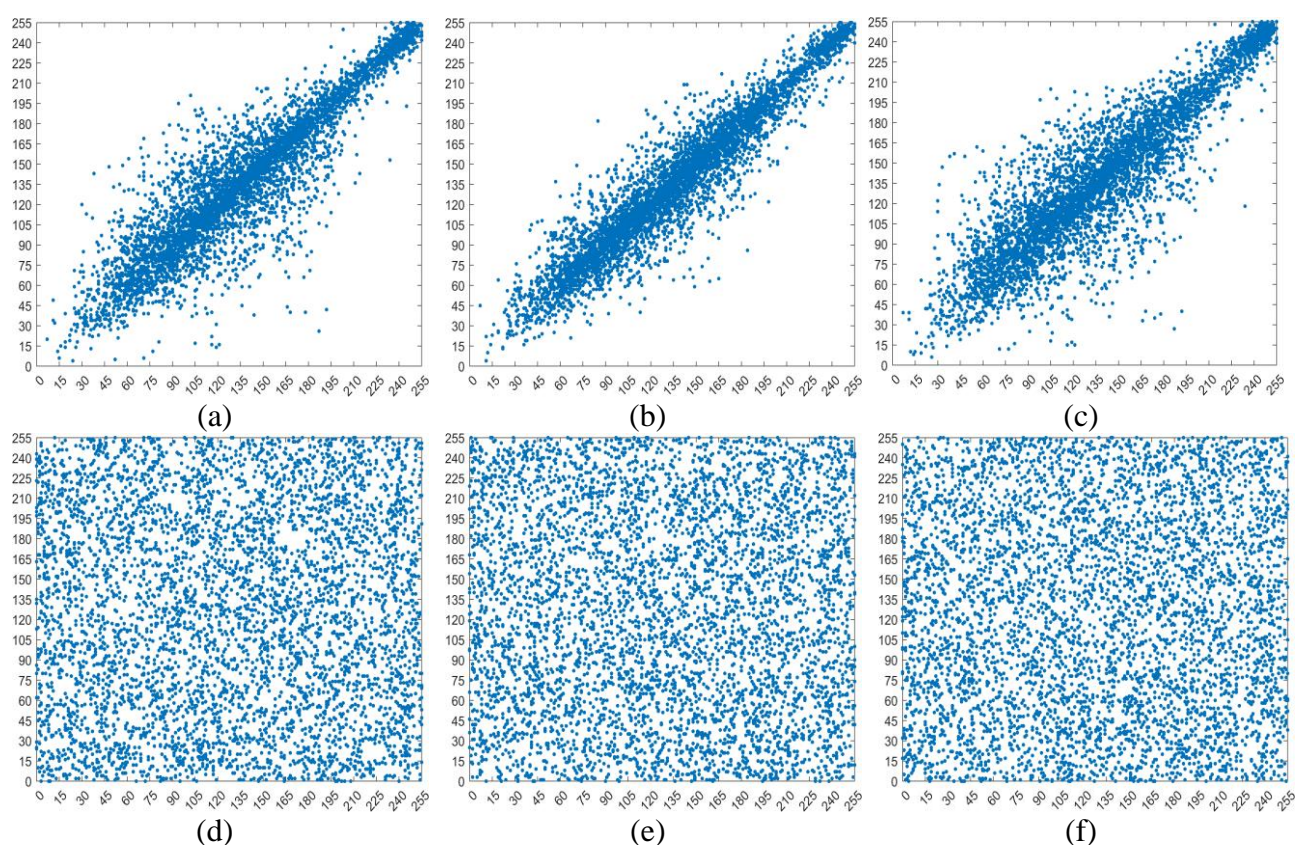|	Original image	Encrypted image
Horizontal direction	0.9237	0.0012
Vertical direction	0.8792	0.0076
Diagonal direction	0.8437	-0.0034

Figure 9. Correlation of adjacent pixels: (a) horizontal direction of image in Figure 7(a); (b) vertical direction of image in Figure 7(a); (c) diagonal direction of image in Figure 7(a); (d) horizontal direction of image in Figure7(b); (e) vertical direction of image in Figure 7(b); (f) diagonal direction of image in Figure 7(b).

5.3. Information entropy

Information entropy represents the degree of confusion of image pixel values, and the mathematical definition is as follows:

$$H = -\sum_{i=0}^{2^N-1} p(x_i) log_2 p(x_i) \tag{27}$$

where $p(x_i)$ represents the proportion of the gray value x_i in an image, and 2^N is the gray level of the image. The information entropy results for different channels of the original and encrypted images are shown in Table 2. It can be ascertained from the results that the information entropy of each channel is very close to the theoretical maximum of 8, and that the encryption algorithm can effectively resist the attack based on the image information entropy.

Table 2. The information entropy results for the original image and encrypted image.

	Original image	Encrypted image
R	7.7063	7.9994
G	7.4105	7.9993
B	7.7285	7.9994

5.4. Analysis of key sensitivity

If a small change in the key causes the decrypted image to be completely unrelated to the original image, this characteristic is called strong key sensitivity. The key $X(0)$ was changed from 0.5146 to 0.5146000000000001 during decryption, and the decrypted image under the wrong key was obtained as shown in Figure 10. It can be ascertained from the results that the key changed by 10^{-16}, but the decrypted image cannot be recovered, demonstrating the strong key sensitivity of the algorithm.
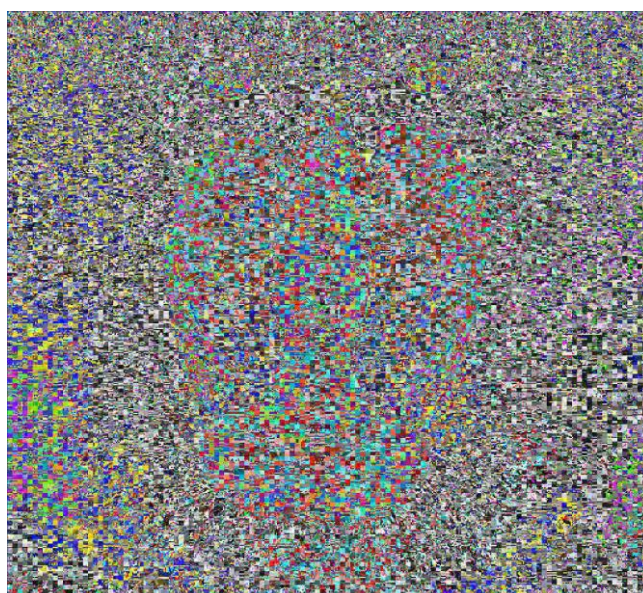

Figure 10. Decrypted image under the wrong key.

5.5. Ability to resist differential attacks

One or several pixel values of the original image was changed and encrypted according to the encryption algorithm by the attacker; some relationships existing between the images were obtained by comparing the pixel values of the encrypted and the original encrypted images, which can be represented by the change rate of the number of pixels (NPCR) and uniform average intensity of change (UACI); the corresponding formulas are as follows:

$$NPCR = \frac{\sum_{i,j} D(i,j)}{m \times n} \times 100, \tag{28}$$

$$UACI = \frac{1}{m \times n} \left[\sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255} \right] \times 100, \tag{29}$$

$$D(i,j) = \begin{cases} 1, C_1(i,j) \neq C_2(i,j) \\ 0, C_1(i,j) = C_2(i,j) \end{cases}, \tag{30}$$

where m and n are the height and width of the original image, respectively, C_1 and C_2 are the two encrypted images mentioned above, and the ideal values of NPCR and UACI are 99.6094% and 33.4635%, respectively. The pixel value in the baboon image with the coordinate of (158,164) was added by 1, and the specific numerical results were obtained according to Eqs (28)–(30) as shown in Table 3. The results show that the scheme can resist differential attacks.

Table 3. The test results for NPCR and UACI.

Image	NPCR	UACI
Baboon	99.6102%	33.4547%
[16] Lena	99.6063%	33.4607%
[16] 5.1.09.tiff	99.6124%	33.4790%
[16] 5.1.10.tiff	99.6109%	33.4533%

5.6. Robustness against noise and data loss

Some information may be lost due to noise during transmission. Salt and pepper noise with an intensity of 10% and a center clipping attack with an intensity of 20% were introduced into the encrypted image. The encrypted images after disturbance, as well as the corresponding decrypted images are shown in Figure 11, which shows that the decrypted images can be recognized, and that the proposed scheme is robust against noise.
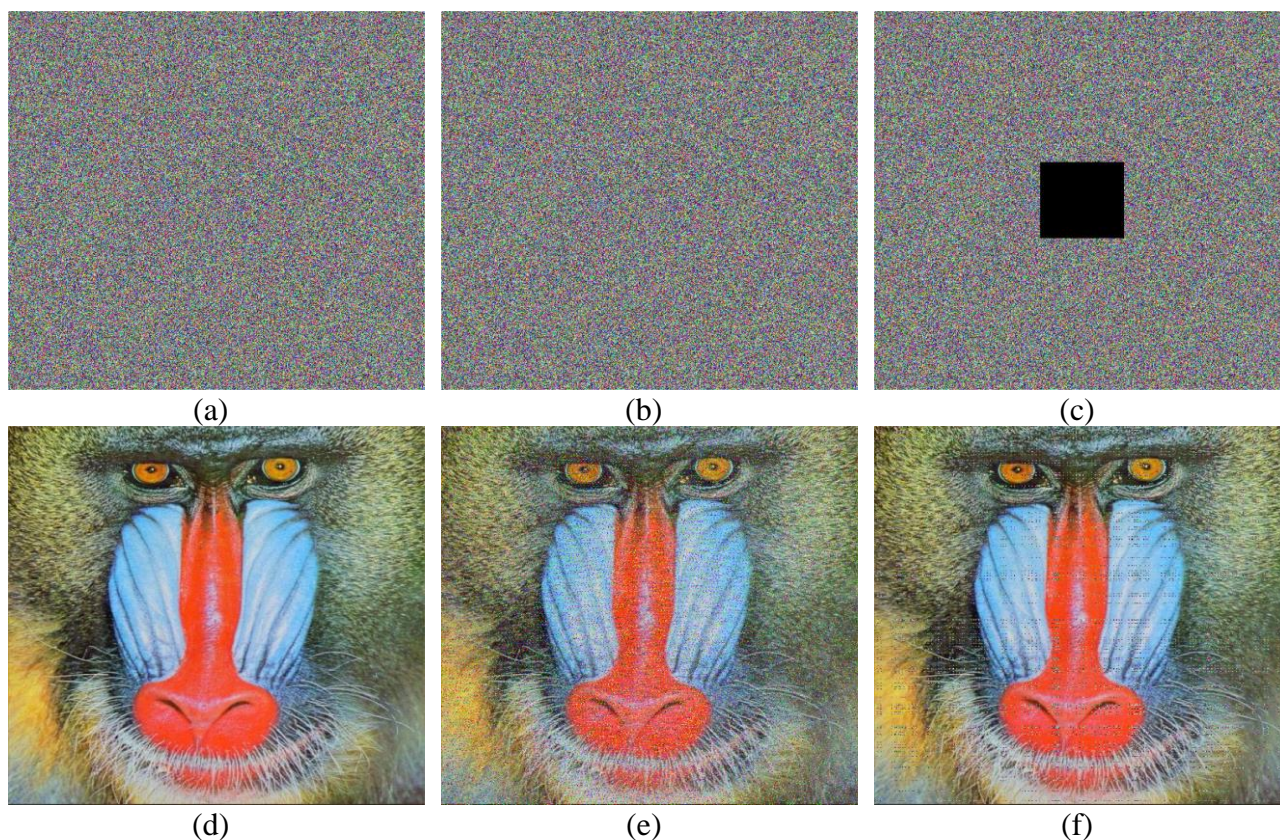
Figure 11. Robustness analysis: (a) normal encrypted image; (b) encrypted image with 10% salt-and-pepper noise; (c) encrypted image with 20% data loss, (d) normal decrypted image; (e) decrypted image with 10% salt-and-pepper noise; (f) decrypted image with 20% data loss.

5.7. Analysis of peak signal-to-noise ratio and mean square error

The difference between the encrypted image F and the original image P of a good encryption algorithm should be huge. We applied the peak signal-to-noise ratio (PSNR) and the mean square error (MSE) as two image quality evaluation indexes, where the better effect of the encryption is a larger MSE value and a smaller PNSR value. The PSNR and MSE can be calculated as shown below:

$$MSE = \frac{1}{m \times n} \sum_{i=1}^{m} \sum_{j=1}^{n} [P(i,j) - F(i,j)]^2, \tag{31}$$

$$PSNR = 10 \times log_{10} \left(\frac{255^2}{MSE} \right), \tag{32}$$

where m and n are the height and width of the original image, respectively, and the MSE and PSNR results for the original image, as compared to those of [16], are shown in Table 4. The results show that the proposed algorithm can effectively encrypt images.

Table 4. The MSE and PSNR test results.

Image	MSE	PSNR
Baboon	9741.2	8.2447
[16] Lena	8861.4	8.6558
[16] 5.1.09.tiff	6233.4	10.1836
[16] 5.1.10.tiff	7696.4	9.2680

6. Conclusions

In this study, an MHDBAMNN was investigated for finite-time projection synchronization, along with its application to image encryption. To achieve secure image transmission, a time delay independent negative feedback controller was constructed by using a feedback control strategy and the criterion for finite-time projection synchronization of MHDBAMNN was derived. Numerical simulation examples demonstrated that the drive-response system realized finite-time projection synchronization, thus validating the finite-time synchronization theory proposed in this study. Given that the MHDBAMNN model has good chaotic properties; in the diffusion stage, the combination of Chen's hyperchaotic system and the DNA method was used to achieve a better diffusion effect. The experimental results show that the proposed algorithm can effectively encrypt images and has good robustness. The security of the proposed algorithm and the potential application of the model in secure image transmission were demonstrated by analyzing the encryption effect. All of these satisfactory characteristics make the proposed scheme a potential candidate for multimedia data encryption, and studying the dynamics of fractional order neural networks with hidden hyperchaos or hidden multi-scrolls is our future work.

Use of AI tools declaration

The authors declare they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

Conflicts of interest

The authors declare no conflict of interest.

References

1. Z. Wang, X. Wang, Z. Zeng, Memristive circuit design of brain-like emotional learning and generation, *IEEE Trans. Cybernet*, **53** (2023), 222–235. https://doi.org/10.1109/TCYB.2021.3090811

2. B. Gao, Y. Zhou, Q. Zhang, S. Zhang, P. Yang, Y. Xi, et al., Memristor-based analogue computing for brain-inspired sound localization with in situ training, *Nat. Commun.*, **13** (2022), 2026. https://doi.org/10.1038/s41467-022-29712-8

3. A. A. Kausani, M. Anwar, Behavioral modeling of the pinched hysteresis loop of a Pt/TiO$_2$/Pt memristor, *Int. J. High Speed Electronics Syst.*, **32** (2023), 2350008. https://doi.org/10.1142/S0129156423500088

4. F. Min, L. Xue, Routes toward chaos in a memristor-based Shinriki circuit, *Chaos*, **33** (2023), 023122. https://doi.org/10.1063/5.0126900

5. R. B. Naik, U. Singh, A review on applications of chaotic maps in pseudo-random number generators and encryption, *Ann. Data Sci.*, **11** (2022), 25–50. https://doi.org/10.1007/s40745-021-00364-7

6. I. Ahmad, M. Shafiq, B. Naderi, Finite-time synchronization of four-dimensional memristor-based chaotic oscillator and applied to secure communication systems, *Franklin Open*, **3** (2023), 100015. https://doi.org/10.1016/j.fraope.2023.100015

7. B. Alhayani, A. A. Abdallah, Manufacturing intelligent Corvus corone module for a secured two way image transmission under WSN, *Eng. Comput.*, **38** (2021), 1751–1788. https://doi.org/10.1108/EC-02-2020-0107

8. S. Sun, H. Shi, S. Duan, L. Wang, Memristor-based time-delay hyperchaotic system with circuit simulation and image encryption, *Phys. Scr.*, **97** (2022), 035204. https://doi.org/10.1088/1402-4896/ac4cfb

9. D. Ding, H. Xiao, Z. Yang, H. Luo, Y. Hu, X. Zhang, et al., Coexisting multi-stability of Hopfield neural network based on coupled fractional-order locally active memristor and its application in image encryption, *Nonlinear Dyn.*, **108** (2022), 4433–4458. https://doi.org/10.1007/s11071-022-07371-0

10. L. Chen, H. Yin, T. Huang, L. Yuan, S. Zheng, L. Yin, Chaos in fractional-order discrete neural networks with application to image encryption, *Neural Netw.*, **125** (2020), 174–184. https://doi.org/10.1016/j.neunet.2020.02.008

11. Q. Lai, Z. Wan, H. Zhang, G. Chen, Design and analysis of multiscroll memristive hopfield neural network with adjustable memductance and application to image encryption, *IEEE Trans. Neural Netw. Learn Syst.*, **34** (2022), 7824–7837. https://doi.org/10.1109/TNNLS.2022.3146570

12. Q. Lai, Z. Chen, Dynamical analysis and finite-time synchronization of grid-scroll memristive chaotic system without equilibrium, *Chaos Solitons Fractals*, **176** (2023), 114118. https://doi.org/10.1016/j.chaos.2023.114118

13. X. Yang, W. Chen, L. Wu, N. Qiu, Synchronous oscillation characteristic and finite-time function projection synchronization control method for microgrids, *Electr. Power Syst. Res.*, **221** (2023), 109492. https://doi.org/10.1016/j.epsr.2023.109492

14. R. Zhang, X. Xi, H. Tian, Z. Wan, Dynamical analysis and finite-time synchronization for a chaotic system with hidden attractor and surface equilibrium, *Axioms*, **11** (2022), 579. https://doi.org/10.3390/axioms11110579

15. S. Wang, H. Zhang, W. Zhang, H. Zhang, Finite-time projective synchronization of Caputo type fractional complex-valued delayed neural networks, *Mathematics*, **9** (2021), 1406. https://doi.org/10.3390/math9121406

16. Q. Lai, Z. Chen, Grid-scroll memristive chaotic system with application to image encryption, *Chaos Solitons Fractals*, **170** (2023), 113341. https://doi.org/10.1016/j.chaos.2023.113341

17. C. Wang, D. Tang, H. Lin, F. Yu, Y. Sun, High-dimensional memristive neural network and its application in commercial data encryption communication, *Expert Syst. Appl.*, **242** (2024), 122513. https://doi.org/10.1016/j.eswa.2023.122513

18. X. Kong, F. Yu, W. Yao, S. Cai, J. Zhang, H. Lin, Memristor-induced hyperchaos, multiscroll and extreme multistability in fractional-order HNN: Image encryption and FPGA implementation, *Neural Netw.*, **171** (2024), 85–103. https://doi.org/10.1016/j.neunet.2023.12.008

19. S. Kanwal, S. Inam, O. Cheikhrouhou, K. Mahnoor, A. Zaguia, H. Hamam, Analytic study of a novel color image encryption method based on the chaos system and color codes, *Complexity*, **2021** (2021), 5499538. https://doi.org/10.1155/2021/5499538

20. X. Gao, B. Sun, Y. Cao, S. Banerjee, B. Mou, A color image encryption algorithm based on hyperchaotic map and DNA mutation, *Chinese Phys. B*, **32** (2023), 030501. https://doi.org/10.1088/1674-1056/ac8cdf

21. X. Xie, S. Wen, Y. Feng, B. O. Onasanya, Three-stage-impulse control of memristor-based chen hyper-chaotic system, *Mathematics*, **10** (2022), 4560. https://doi.org/10.3390/math10234560

22. H. Wen, Y. Lin, Cryptanalysis of an image encryption algorithm using quantum chaotic map and DNA coding, *Expert Syst. Appl.*, **237** (2024), 121514. https://doi.org/10.1016/j.eswa.2023.121514

23. Y. Zhu, C. Wang, J. Sun, F. Yu, A chaotic image encryption method based on the artificial fish swarms algorithm and the DNA coding, *Mathematics*, **11** (2023), 767. https://doi.org/10.3390/math11030767

24. C. Xu, X. Yang, J. Lu, J. Feng, F. E. Alsaadi, T. Hayat, Finite-time synchronization of networks via quantized intermittent pinning control, *IEEE Trans. Cybern.*, **48** (2018), 3021–3027. https://doi.org/10.1109/TCYB.2017.2749248

25. X. Yang, Can neural networks with arbitrary delays be finite-timely synchronized? *Neurocomputing*, **143** (2014), 275–281. https://doi.org/10.1016/j.neucom.2014.05.064

26. H. Wu, L. Wang, P. Niu, Y. Wang, Global projective synchronization in finite time of nonidentical fractional-order neural networks based on sliding mode control strategy, *Neurocomputing*, **235** (2017), 264–273. https://doi.org/10.1016/j.neucom.2017.01.022