Alexandria Engineering Journal (2020) 59, 601–611

HOSTED BY

ELSEVIER

Alexandria University

Alexandria Engineering Journal

www.elsevier.com/locate/aej
www.sciencedirect.com

ORIGINAL ARTICLE

A robust spectral integral method for solving chaotic finance systems

Claude Rodrigue Bambe Moutsinga *, **Edson Pindza**, **Eben Maré**

Department of Mathematics and Applied Mathematics, University of Pretoria, Pretoria 002, South Africa

Received 7 November 2019; revised 5 January 2020; accepted 8 January 2020

KEYWORDS

Operational matrices;
Chebyshev polynomials;
Chaos;
Synchronization;
Lyapunov theory;
Spectral integral method

Abstract Nonlinear chaotic finance systems are represented by nonlinear ordinary differential equations and play a significant role in micro-and macroeconomics. In general, these systems do not have exact solutions. As a result, one has to resort to numerical solutions to study their dynamics. However, numerical solutions to these problems are sensitive to initial conditions, and a careful choice of the suitable parameters and numerical method is required. In this paper, we propose a robust spectral method to numerically solve nonlinear chaotic financial systems. The method relies on spectral integration diagonal matrices coupled with a domain decomposition method to preserve the high accuracy of our methodology on a long time period. In addition, we investigate stability of chaotic finance systems using the Lyapunov theory, and a two sliding controller mode synchronisation to regulate the synchronisation of these systems. Numerical experiments reveal the high accuracy and the robustness of our method and validate the synchronisation of chaotic finance systems.

1. Introduction

Nonlinear chaotic systems have attracted many research works in the sense that they can describe the evolution of more complex systems in a reasonable manner. The presence of parameters is typical for many models of economic processes. For example, in economic growth models, they may represent tools for influencing the economy, while the aim of the analysis is to find such quantities that would lead to the optimal path of growth. However, if the analyzed model has chaotic dynamics, the matter is essentially complicated. The high sensitivity of chaotic system to a change in the initial conditions makes it impossible to predict the effects of economic decisions in a long time scale. Considering that government are usually interested in stimulating investment in order to cope with unemployment rate, exports, etc..., this may cause a total new trajectory for the system. Therefore, an effective and rapid control method is very much needed when chaos appears in order to avoid undesired trajectory and make suitable economic adaptation and prediction possible, specially from government and investors side. Ma and Chen [11,12] provided a practical way to analyse and predict the chaotic economic systems from a bifurcation approach.

By control, we refer to redesigning the system in which parameters are added and controlled in order to eradicate the chaotic behaviour of the system and reach a desired goal. Lots of research has been conducted on the nonlinear chaotic finance problem, mostly with the aim of achieving control and synchronization. Several techniques are used for control of the

* Corresponding author.
E-mail address: bamberodrigue@gmail.com (C.R. Bambe Moutsinga).
Peer review under responsibility of Faculty of Engineering, Alexandria University.

https://doi.org/10.1016/j.aej.2020.01.016
1110-0168

chaotic system including sliding mode, feedback control [1], integral sliding mode control [24], inverse optimal control [4], passive control [17], adaptive control [23], backstepping control [22] to name these only.

In the synchronization process, we are given two identical systems with different starting points. One initial system, also called driving, and a second system called response, or slave, similar to the driving system. The aim is that by adding some parameters on the slave system we should match the driving system after some time. In the world of finance, this means from one chaotic finance system generated from a certain economy we can add some parameters to it in order to match another desired chaotic system generated by another economy. Various synchronization methods have been introduced so far and some with extension to fractional cases, namely, the projective synchronisation [27,29], sliding mode [21,13], and the nonlinear control [15,28]. In this paper, a two sliding controller mode synchronisation is used as it regulates synchronisation of chaotic finance system more effectively than passive control, while keeping also the system internally stable [13].

Analytical solutions for nonlinear chaotic systems are almost nonexistent. Therefore, we rely on numerical methods to study these systems. In the field of numerical methods for solving differential equations, two main classes can be distinguished, classical methods and spectral methods. By classical methods we refer to the class of finite difference, finite element methods. These methods are very accurate, but computationally costly.

However, spectral methods have the advantage of being fast converging methods. Their truncation error decays as fast as the global smoothness of the underlying solution permits. Their definite integrals are calculated once by the quadrature rule [8], see also [18,9] for more on spectral methods. For ordinary differential equations in which some coefficient functions or solutions are not analytic, Babolian [2] introduced a modified spectral method that is more efficient than the existing spectral methods. Various quadrature and modified quadrature rules can be found in the literature of spectral methods, including quadrature based on Chebyshev polynomials. Shifted Chebyshev polynomials for instance are used to solve the Klein-Gordon equation, [10]. The method is referred to as shifted Chebyshev-Tau method. An extension of this method is applied in the case of fractional differential equations [6]. Two years later, Bhrawy [3] introduced an operational matrix to the shifted Chebyshev method to generate a faster algorithm for fractional integration in the sense that only a small number of shifted Chebyshev polynomials is needed to obtain a satisfactory result. Liu [14] applies a quasi-inverse technique to solve differential equations directly. The method performs very well and shows obvious advantage especially when it comes to multi-dimensional cases.

Driscoll [7] presents a fast algorithm based on operational matrices in which the matrices have a lower density. In integral form, large condition numbers associated with differentiation matrices in high-order problems are avoided. The Chebfun package of Matlab [19] is used in the algorithm, as it exploits results from approximation theory, spectral methods, and object-oriented software design to reduce the distance between analytical expressions and numerical solutions for one-dimensional problems. Like the Matlab Differentiation Matrix Suite (DMS) package [26], Chebfun also suffers from the fact that the differentiation matrix gets full (while it is sparse for the finite difference or the finite element method) and, more importantly, it is very sensitive to rounding errors. Reason being that these two packages are based on the spectral collocation method where the approach focuses more on the physical space generated from the quadrature.

Following the same matrix based operational approach, an improvement of these packages is brought by Trif [20] in introducing the chebpack package that is based on the Chebyshev-Tau method where the focus is more on the spectral space of coefficients rather than the physical space. This approach takes advantage of the spectral properties of Chebyshev polynomials resulting in avoiding full matrices. Actually the obtained matrices are sparse upper triangular and for the particular case of constant coefficient in the system, the matrices become diagonal almost everywhere. Hence a tremendous gain in computation is achieved.

In this paper we intend to solve the chaotic finance system by means of the robust spectral integral method (RSIM), to compute the solution of three dimensional and four dimensional problems. In addition, a splitting method is used in order to achieve fast convergence without compromising on the accuracy over a long time period.

This paper is organised as follows, Section 2 presents the chaotic finance system with a brief analysis on the stability. In Section 3 we introduce the robust spectral integral method In Section 4, we present numerical results and conduct an error analysis as well as a synchronisation via sliding mode. The last section is devoted to the conclusion.

2. Chaotic finance systems

The chaotic finance system, under study, is driven by the interaction of three main variables influencing the market economy. This interaction is modeled in the form of three nonlinear simultaneous ordinary differencial equations (ODEs) as follows (see [11])

$$\begin{cases} \dot{x} = z + (y - a)x \\ \dot{y} = 1 - by - x^2 \\ \dot{z} = -x - cz. \end{cases} \tag{2.1}$$

where x component stands for the interest rate dynamics which is defined as the percentage amount of the principal a borrower promises to pay the lender, y is the investment demand which is the desired capital and inventories by firms, and z represents the price index of a stock. The positive constant parameters a, b, c are the saving, the per-investment cost and the elasticity of the demand, respectively.

By applying some appropriate change of coordinate system and settings, different views of the chaotic finance system can be presented [5]. In this paper we shall stick to the presentation given in (2.1). The system admits three equilibrium points $X_0 = (0, \frac{1}{b}, 0)$, $X_1 = \left(\sqrt{1 - ba - \frac{b}{c}}, a + \frac{1}{c}, -\frac{1}{c}\sqrt{1 - ba - \frac{b}{c}}\right)$ and $X_2 = \left(-\sqrt{1 - ba - \frac{b}{c}}, a + \frac{1}{c}, \frac{1}{c}\sqrt{1 - ba - \frac{b}{c}}\right)$. The Jacobian matrix is

$$J_x = \begin{bmatrix} y-a & x & 1 \\ -2x & -b & 0 \\ -1 & 0 & -c \end{bmatrix}, \tag{2.2}$$

and at the equilibrium point X_0,

$$J_{x_0} = \begin{bmatrix} \frac{1}{b}-a & 0 & 1 \\ 0 & -b & 0 \\ -1 & 0 & -c \end{bmatrix}.$$

The characteristic polynomial is

$$P(\lambda) = \lambda^3 - \left(\frac{1}{b} - a - b - c\right)\lambda^2 - \left(\frac{c}{b} - ab - ac - bc\right)\lambda - (c - b - abc). \tag{2.3}$$

According to the Routh-Hurwitz criterion for polynomial of order 3, the real parts of the three eigenvalues are all negative if the following simultaneous inequalities hold:

$$-\left(\frac{1}{b} - a - b - c\right) > 0, \tag{2.4}$$

$$-(c - b - abc) > 0, \tag{2.5}$$

$$\left(\frac{1}{b} - a - b - c\right)\left(\frac{c}{b} - ab - ac - bc\right) + (c - b - abc) > 0. \tag{2.6}$$

In addition, we also see that the root $\lambda = -b$ of the characteristic polynomial (2.3) has a negative real part if $b > 0$. Now, if we arbitrarily consider $b = 0.1$, $c = 1$ and if we take a to be our control parameter, then the above set of conditions is written as:

$$a > \tfrac{89}{10},$$
$$a > 9,$$
$$\left(a - \tfrac{89}{10}\right)\left(\tfrac{11a-99}{10}\right) - \tfrac{a-9}{10} > 0.$$

This means for $a > 9$ all the eigenvalues will have a negative real part. As a result, X_0 is asymptotically stable.

At the equilibrium point $X_1 = \left(\sqrt{1 - ba - \frac{b}{c}}, a + \frac{1}{c}, -\frac{1}{c}\sqrt{1 - ba - \frac{b}{c}}\right)$, the Jacobian is given by

$$J_{x_1} = \begin{bmatrix} \frac{1}{c} & \sqrt{1 - ba - \frac{b}{c}} & 1 \\ -2\sqrt{1 - ba - \frac{b}{c}} & -b & 0 \\ -1 & 0 & -c \end{bmatrix}.$$

Therefore, the corresponding characteristic polynomial is obtained in Eq. (2.7)

$$P(\lambda) = \lambda^3 + \left(b + c - \frac{1}{c}\right)\lambda^2 + \left(2 + bc - 2ab - \frac{3b}{c}\right)\lambda + (2c - 2b - 2abc). \tag{2.7}$$

Accordingly, the real parts of all the eigenvalues are all negative if

$$b + c - \frac{1}{c} > 0, \tag{2.8}$$

$$2c - 2b - 2abc > 0, \tag{2.9}$$

$$\left(b + c - \frac{1}{c}\right)\left(2 + bc - 2ab - \frac{3b}{c}\right) - (2c - 2b - 2abc) > 0. \tag{2.10}$$

Choosing the constants $b = 0.1$, $c = 1$, the above set of conditions result to

$$a < 9,$$
$$a > 9$$

implying X_1 cannot be stable no matter the value of a. The same analysis can also be conducted for the equilibrium point $X_2 = \left(-\sqrt{1 - ba - \frac{b}{c}}, a + \frac{1}{c}, \frac{1}{c}\sqrt{1 - ba - \frac{b}{c}}\right)$. Elouahab et al. [1] proved the existence of a chaotic behaviour of problem (2.1) for the constants $b = 0.1$, $c = 1$ and $0 < a < 7$ from the Lyapunov theory.

In the next section, we present a pseudo-spectral method used to solve the system of ODEs (2.1). The main advantage of this method is that it can handle large time values while preserving high accuracy.

3. Chebyshev polynomials

The Chebyshev polynomial $T_n(x)$ of 1st kind is a polynomial in $x \in [-1, 1]$ of degree $n > 0$ defined by the relation:

$$T_n(x) = \cos n\theta, \text{for } x = \cos\theta,$$
$$ie. \quad T_n(x) = \cos(n \arccos(x)).$$

Note that the definition of the Chebyshev polynomials can easily be extended to any interval $[a, b]$ by just applying a shift mapping $s : x \to s(x) = \frac{2}{b-a}x - \frac{b+a}{b-a}$. For this reason, we shall work on the interval $[-1, 1]$ then applying the inverse shift mapping we can always get back to any interval $[a, b]$. From the trigonometric relation.

$$\cos(n\theta) + \cos(n-2)\theta = 2\cos\theta\cos(n-1)\theta, \tag{3.11}$$

we get

$$T_0(x) = 1, \tag{3.12}$$

$$T_1(x) = x, \tag{3.13}$$

$$T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), \quad n = 2, 3, \ldots \tag{3.14}$$

which in turn can be expressed in a matrix form as:

$$\begin{bmatrix} 1 & & & & \\ -2x & 1 & & & \\ 1 & -2x & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2x & 1 \end{bmatrix} \begin{bmatrix} T_0(x) \\ T_1(x) \\ T_2(x) \\ \vdots \\ T_n(x) \end{bmatrix} = \begin{bmatrix} 1 \\ -x \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{3.15}$$

The zeros of T_n are the points

$$x_k = -\cos\frac{(k - \frac{1}{2})\pi}{n}, \qquad k = 1, 2, \ldots, n. \tag{3.16}$$

The set $\{x_k\}_k$ is termed as collocation points, also called Chebyshev points of first kind. For any point x, the set $\{T_0(x), T_1(x), \ldots\}$ is an orthogonal basis according to the weighted inner product defined by:

$$< f, g >= \int_{-1}^{1} \frac{f(x)g(x)}{\sqrt{1 - x^2}}\, dx, \tag{3.17}$$

for any continuous function f, g defined on $[-1, 1]$. This means that for any polynomial of degree $n > 0$, there exists a unique set of coefficients $\{c_1, c_2, \ldots, c_n\}$ such that

$$p_n(x) = \sum_{k=0}^{n} c_k T_k(x). \tag{3.18}$$

Considering the fact that polynomials are dense in $C([-1,1])$, and the set of Chebyshev polynomials is complete, we can therefore have the following theorem

Theorem 3.1. *Let u be a Lipschitz continuous function on the interval [-1,1]. Then u admits a unique representation as a series of the form:*

$$u(x) = \frac{c_0}{2} + \sum_{k=1}^{\infty} c_k T_k(x). \tag{3.19}$$

where $T_k(x)$ *are Chebyshev polynomials,*

$$c_k = \frac{2}{\pi}\int_{-1}^{1} \frac{u(x)T_k(x)}{\sqrt{1-x^2}}dx, \qquad k = 0,1,2,3,\ldots \tag{3.20}$$

This series converges uniformly and absolutely.

A Chebyshev approximation of order $n > 0$ of a function u continuous on an interval $[-1,1]$ is defined by

$$u_n(x) = \sum_{k=0}^{n} c_k T_k(x) \tag{3.21}$$

$$= \underline{c} \cdot T(x) \tag{3.22}$$

where $\underline{c} = (c_0, c_1, \ldots, c_n)$ is the coefficient vector associated with the approximation u_n. It is usually termed as the spectral representation of u_n. The set of Chebyshev coefficient vectors $\{\underline{c}\}$ of continuous functions on $[-1,1]$ is referred to as the frequency space.

For simplicity of notation, we shall write $u(x)$ in place of $u_n(x)$ to denote the Chebyshev approximation of order n of u at x. Another discrete representation of the function u is to directly interpolate u at the collocation points $x_k's$. This means u can be represented by a vector $\underline{v}$ of its values on the grid $\underline{x} = (x_0, x_1, \ldots, x_n)$, that is $\underline{v} = (u(x_0), u(x_1), \ldots, u(x_n))$. We shall call $\underline{v}$ the physical representation of u.

On the collocation point, one writes

$$\underline{v}(x) = T(x).\underline{c}, \tag{3.23}$$

$$(v(x_0),\ldots,v(x_n)) = \left(\sum_{k=0}^{n} c_k T_k(x_0), \ldots, \sum_{k=0}^{n} c_k T_k(x_n),\right) \tag{3.24}$$

where T is the matrix defined as follows

$$T = \begin{bmatrix} T_0(x_0) & T_1(x_0) & \ldots & T_n(x_0) \\ T_0(x_1) & T_1(x_1) & \ldots & T_n(x_1) \\ \vdots & & \ddots & \vdots \\ T_0(x_n) & T_1(x_n) & \ldots & T_n(x_n) \end{bmatrix}.$$

Since

$\underline{v} = T\underline{c}$,

this implies that $\underline{c} = T^{-1}\underline{v}$.

From the nature of $T_k's$, the matrix T is sparse and FFT enables to get T^{-1}.

3.1. Some useful properties

Consider two functions a and u of a variable x, with spectral representation $\underline{a}$ and $\underline{u}$ respectively. Then the product $a(x)\cdot u(x)$ admits also a spectral representation, denoted as $\underline{\phi}$ which is defined by

$$\underline{\phi} = \mathbf{a}.\underline{c} \tag{3.25}$$

where $\mathbf{a}$ is termed as the matrix representation of the function $a(x)$ and $\underline{c}$ is the spectral representation of function u, see [7]

An efficient way of getting matrix $\mathbf{a}$ is to write the product in its discrete form.

$$\text{Since} \quad a(x)u(x) = \left[\sum_{k=0}^{n} a_k T_k(x)\right]\left[\sum_{k=0}^{n} c_k T_k(x)\right], \tag{3.26}$$

$$\text{then} \quad \sum_{k=0}^{n} \phi_k T_k(x) = \sum_{k=0}^{n}\sum_{l=0}^{n} \alpha_{kl} a_k c_l T_k T_l \tag{3.27}$$

for some coefficients α_{kl}, $0 \leqslant k,l \leqslant n$. In addition, given the following relation

$$T_k(x)T_l(x) = \frac{1}{2}\left[T_{k+l}(x) + T_{|k-l|}(x)\right], \qquad \text{forall } k,l = 0,1,\ldots n \tag{3.28}$$

and in rearranging terms properly, it brings to existence a matrix $\mathbf{a}$ such that

$$\sum_{k=0}^{n} \phi_k T_k(x) = \sum_{k=0}^{n}\left[\sum_{l=0}^{n} \mathbf{a}_{kl} c_l\right] T_k(x).$$

In the frequency space, this will written in the form

$$\underline{\phi} = \mathbf{a}\underline{c}. \tag{3.29}$$

3.2. Differentiation and integration

In view of Eq. (3.21) and by differentiation, $u'(x)$ is given by

$$u'(x) = \sum_{k=0}^{n} c_k T_k'(x). \tag{3.30}$$

The differentiation of relation (3.14) and (3.13) implies

$$T_0 = T_1', \tag{3.31}$$

$$T_1 = \frac{T_2'}{2}, \tag{3.32}$$

$$T_{n+1}(x) = nT_{n-1}'(x) - 2(1-x^2)T_n'(x) \tag{3.33}$$

$$\text{ie.} \quad T_n = \frac{T_{n+1}'}{2(n+1)} - \frac{T_{n-1}'}{2(n-1)}, \quad n = 2,3,\ldots. \tag{3.34}$$

Inserting this back into (3.30) shows the existence of a matrix $D = (d_{kl})_{0<k,l<n}$ such that

$$\sum_{k=0}^{n} \underline{c}_k' T_k(x) = \sum_{k=0}^{n}\sum_{l=0}^{n} d_{kl} c_l T_k(x) \tag{3.35}$$

$$\text{ie.} \quad \underline{c}' = D\underline{c} \tag{3.36}$$

where $\underline{c}'$ is the spectral representation of the derivative function u' and moreover D is a sparse upper triangular matrix, with the following properties

$$\begin{cases} d_{kl} = 0, \text{ for } k \leqslant l, \\ d_{kl} = 0, \text{ if } l-k \text{ is even}, \\ d_{kl} = 2k, \text{ if } l-k \text{ is odd}. \end{cases} \tag{3.37}$$

Applying the above result recursively, we get the spectral representation $c^{(p)}$ of the derivative with order p of u stated by

$$\underline{c}^{(p)} = D^p \underline{c}. \tag{3.38}$$

For the case of integration, we recall again the relation

$$T_{n+1}(x) = nT_{n-1}(x) - 2(1-x^2)T_n(x) \tag{3.39}$$
$$\int T_n(x)dx = \frac{1}{2}\left[\frac{T_{n+1}(x)}{n+1} - \frac{T_{n-1}(x)}{n-1}\right], n = 2, 3, \tag{3.40}$$
$$\int T_1(x)dx = \frac{1}{4}T_2(x), \tag{3.41}$$
$$\int T_0(x)dx = \frac{1}{2}T_1(x). \tag{3.42}$$

As a linear operator, the integral of u will also be a continuous Lipschitz function in $[-1,1]$, which will in turn have a unique expansion series of the form

$$\int u(x)dx = \sum_{k=0}^{n} I_k T_k(x), \qquad x \in [a,b],$$

where I_k's are coefficients of the integral of u, and similarly as with differentiation there exists a $n \times n$-matrix J such that

$$I_k = \sum_{l=0}^{n} J_{kl} c_l, \tag{3.43}$$

or simply

$$\underline{I} = J \cdot \underline{c}, \tag{3.44}$$

where $\underline{I}$ is the spectral representation of the integral of u. In fact,

$$\int u(x)dx = \int \sum_{k=0}^{N-1} c_k T_k(x)$$
$$i.e \qquad \sum_{k=0}^{N-1} I_k T_k(x) = \int \sum_{k=0}^{N-1} c_k T_k(x)\,dx$$
$$= \sum_{k=0}^{N-1} c_k \int T_k(x)\,dx$$
$$\sum_{k=0}^{N-1}\sum_{j=2}^{N-1} J_{kj} c_j T_k(x) = \sum_{k=2}^{n} c_k \tfrac{1}{2}\left[\tfrac{T_{k+1}}{k+1} - \tfrac{T_{k-1}}{k-1}\right].$$

Performing a smart multiplication and rearranging terms we get the coefficients of J recursively as follows:

$$J_{kk} = 0, \quad J_{01} = \frac{1}{2}, \quad J_{k,k-1} = -J_{kk+1} = \frac{1}{k}. \tag{3.45}$$

So then, the spectral representation of the integral of u is the vector $\underline{d} = J.\underline{c}$, and for any continuous function $a(x)$, the corresponding spectral representation for the integral of the product $a(x)u(x)$ is $J\mathbf{a}\,\underline{c}$ where $\mathbf{a}$ is the matrix representation of the function a. We shall write

$$\int a(x)u(x)dx \to J\mathbf{a}\,\underline{c}. \tag{3.46}$$

Consequently it can be seen with the help of elementary technique of integration by parts that

$$\int a_1(x)u'(x)dx \to (\mathcal{I} - JD)\mathbf{a}_1\,\underline{c}$$
$$\iint a_2(x)u''(x)dx \to (\mathcal{I} - JD)^2\mathbf{a}_2\,\underline{c}.$$
$$\int a_3(x)\tfrac{d^3u}{dx^3}(x)dx \to (\mathcal{I} - JD)^3\mathbf{a}_3\,\underline{c}$$
$$\vdots$$
$$\int \ldots \int a_m(x)\tfrac{d^mu}{dx^m}(x)dx\ldots dx \to (\mathcal{I} - JD)^m\mathbf{a}_m\,\underline{c}$$

where $\mathcal{I}$ stands for the identity matrix. Thus, for a general linear differential operator L

$$L\,u(x) = \sum_{i=0}^{m} a_i(x)\frac{d^iu}{dx^i}(x) \tag{3.47}$$

$$\text{we have} \quad \int \ldots \int L\,u(x)dx^m \to \sum_{i=0}^{m} J^{m-i}(\mathcal{I} - JD)^i\mathbf{a}_i\,\underline{c}. \tag{3.48}$$

The matrix

$$A = \sum_{i=0}^{m} J^{m-i}(\mathcal{I} - JD)^i\mathbf{a}_i \tag{3.49}$$

is the spectral representation of the integral operator of L.

If we consider a general differential equation $\mathcal{A}u = f$ of order m for which the differential operator can be written as $\mathcal{A} = L + \mathcal{N}$ where L and $\mathcal{N}$ are respectively the linear part and the nonlinear part, then the differential equation then writes as

$$Lu(t) + \mathcal{N}u(t) = f(t) \tag{3.50}$$
$$Lu(t) = -\mathcal{N}u(t) + f(t) \tag{3.51}$$
$$\int \ldots \int Lu(t) \to A\underline{c} = -\mathbf{n} + J^m\underline{f} \tag{3.52}$$
$$A\underline{c} = \mathbf{f} \tag{3.53}$$
$$\text{implying} \qquad \underline{c} = A^{-1}\mathbf{f} \tag{3.54}$$

where $\mathbf{n}$ is the spectral representation of the integral of $\mathcal{N}u$ at order m, and $\mathbf{f} = -\mathbf{n} + J^m\underline{f}$ is the spectral representation of $-\mathcal{N}u + f(t)$. We use the following Algorithm 1:

Algorithm 1. Pseudo code

```
u0 ← initial solution
INITIALIZE L
Evaluate 𝒩, and f at u0
u := L^{-1} * (𝒩 + f)
while ||u − u0|| > ε do
   u0 ← u
   Evaluate 𝒩, and f at u0
   u = L^{-1} * (𝒩 + f)
RETURN u
```

3.3. The robust spectral integral method

For this section we consider I_h to be a mesh on the interval $[0,T]$ and N be the number of subintervals and

$$I_h := \{t_n : 0 = t_0 < t_1 < \cdots < t_N = T\}.$$

We denote by $\Lambda_n = [t_{n-1}, t_n], h_n = t_n - t_{n-1}$ and $u^n(t)$ the solution of (3.50) on the n-th element, namely

$$u^n(t) = u(t), \qquad \forall t \in \Lambda_n, \qquad 1 \leqslant n \leqslant N.$$

Let $M_n > 0$ be an integer and consider $\mathcal{P}_{M_n}$ to be the space of polynomials of order at most M_n built on Λ_n. We apply the spectral method as described in the Algorithm 1 to obtain a numerical solution $U_{M_n} \in \mathcal{P}_{M_n}$ on Λ_n. The Robust Spectral Integral Method on the interval $[0,T]$ consists of a successive application of the spectral method on each Λ_n to obtain a global numerical solution $U_M(t)$ of (3.50) defined in such way that

$$U_M(t) = U_{M_n}(t), \quad t \in \Lambda_n, \quad 1 \leqslant n \leqslant N.$$

where M is taken to be the smallest of the M_n's. That is,

$$M = \inf_{0<n\leqslant N} M_n,$$

For each subinterval $[t_i, t_{i+1}]$, Eq. (3.53) is applied.

$$A^{(i)}\mathbf{c}^{(i)} = \mathbf{f}^{(i)}, \qquad i = 0, \ldots, m-1. \tag{3.55}$$

The overall matrix A of the entire problem is then a diagonal of the block of matrices $A^{(i)}$.

$$\begin{pmatrix} A^{(1)} & 0 & & \\ 0 & A^{(2)} & 0 & \\ & \ddots & \ddots & \\ & & 0 & A^{(m)} \end{pmatrix} \begin{pmatrix} \mathbf{c}^{(1)} \\ \mathbf{c}^{(2)} \\ \vdots \\ \mathbf{c}^{(m)} \end{pmatrix} = \begin{pmatrix} \mathbf{f}^{(1)} \\ \mathbf{f}^{(2)} \\ \vdots \\ \mathbf{f}^{(m)} \end{pmatrix}. \tag{3.56}$$

By inversion of the matrix $A^{(i)}$ on each domain Λ_i, we obtain $c^{(i)}$ and therefore u_{M_i} which is U_M on Λ_i.

In this case a global error can arise and jeopardise the convergence. However the following Theorem 3.2 still guarantees an exponential convergence even after discretization.

Theorem 3.2. *Assume that u belongs to the broken Sobolev space: $u \in H^1(0,T)$ and $u_{|\Lambda_n} \in H^{r_n}(\Lambda_n)$, $1 \leqslant n \leqslant N$ with integers $2 \leqslant r_n \leqslant M_n + 1$, and there exists a constant $L \geqslant 0$ such that for any z_1 and z_2,*

$$|f(z_1, t) - f(z_2, t)| \leqslant L|z_1 - z_2|. \tag{3.57}$$

Then for

$$2\sqrt{2\pi}h_{\max}L \leqslant \beta < 1, \tag{3.58}$$

we have

$$\|u - U_M\|^2_{H^1(0,T)} \leqslant c_\beta T \exp(c_\beta T)\sum_{i=1}^{N} h_i^{2r_i-2} M_i^{2-2r_i} |u|^2_{H^{r_i}(\Lambda)}, \tag{3.59}$$

where c_β is a positive constant depending only on β.

Where $\Lambda_n = [t_{n-1}, t_n], h_n = t_n - t_{n-1}$ and the constant M_n is of the order of the Chebyshev polynomial of approximation u_n defined on Λ_n. The proof can be found in [11,25].

4. Applications and numerical results

In this section, we apply our method to different problems found in financial economics and test the convergence, and efficiency of the proposed method against the existing Chebfun method [16]. In addition we provide an application of our method for synchronization. Since the exact solution is not available we choose the ODE15s with relative and absolute tolerance 10^{-14} to serve as the benchmark solution. The error E is the maximal error given by:

$$\|E\| = \|Sol_{Benchmark} - Sol_{Numerical}\|_\infty. \tag{4.60}$$

Let us apply the above technique described in Section 3 to the nonlinear chaotic problems stated in Section 2. First lets recall the problem (2.1) in its initial form

$$\begin{cases} \dot{x} + ax - z & = xy \\ \dot{y} + by & = 1 - x^2 \\ \dot{z} + x + cz & = 0, \end{cases} \tag{4.61}$$

which can also be written as:

$$\underline{a}u'(t) + Bu(t) = f(t), \quad t \in [0, T] \tag{4.62}$$

where $\underline{a} = (1,1,1)$, $u(t) = [x(t), y(t), z(t)]$, $B = \begin{bmatrix} a & 0 & -1 \\ 0 & b & 0 \\ 1 & 0 & c \end{bmatrix}$ and $f(t) = (x(t)y(t), 1 - x^2(t), 0)$. Integrating (4.62) yields

$$\underline{a}u(t) + B\int_0^T u(t)\,dt = \int_0^T f(t)\,dt. \tag{4.63}$$

The Chebyshev approximation of problem (4.63) at order n in the space of Chebyshev polynomials yields the following simultaneous equations

$$\begin{cases} (\mathcal{I} + aJ)\mathbf{x} - J\mathbf{z} & = \mathbf{f}^1 \\ (\mathcal{I} + bJ)\mathbf{y} & = \mathbf{f}^2 \\ (\mathcal{I} + cJ)\mathbf{z} + J\mathbf{x} & = \mathbf{f}^3. \end{cases} \tag{4.64}$$

where $\mathcal{I}$ is the identity matrix of order 3, J is the integration matrix as defined in (3.43) and (3.45), $\mathbf{x}, \mathbf{y}, \mathbf{z}$ are the spectral representation of the unknown functions $[x(t), y(t), z(t)]$ respectively, and similarly $[\mathbf{f}^1, \mathbf{f}^2, \mathbf{f}^3]$ which represent the coefficient vectors of the nonlinear part $[xy, 1 - x^2, 0]$ respectively. In other words,

$$\begin{bmatrix} \mathcal{I} + aJ & 0 & -J \\ 0 & \mathcal{I} + bJ & 0 \\ J & 0 & \mathcal{I} + cJ \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^1 \\ \mathbf{f}^2 \\ \mathbf{f}^3 \end{bmatrix}. \tag{4.65}$$

This problem is nonlinear, we will apply an iterative method to Eq. (4.65) and the aim is to get the coefficient vector $\underline{c}$ of $u(t) = [x(t), y(t), z(t)]$.

Lets then consider the fix point problem

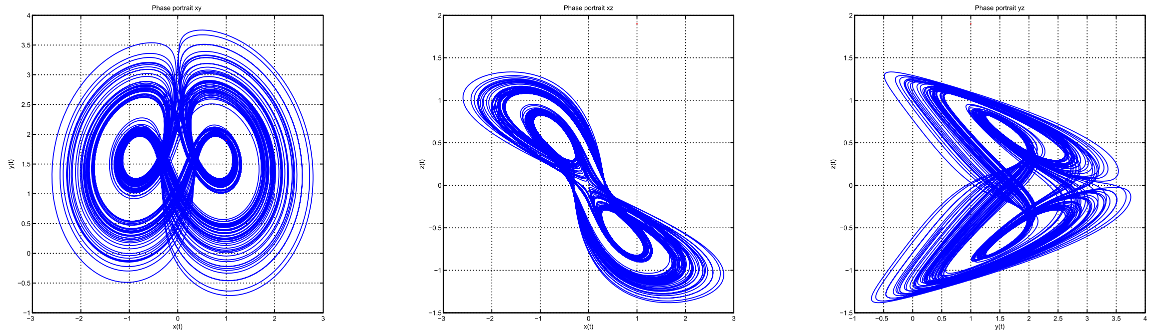
$$A\underline{c} = \mathbf{f}. \tag{4.66}$$

We shall start with an initial guess coming out of the initial condition $[1, 1, 1]$ then get the new $\underline{c}$ by $\mathbf{c} = A^{-1}\mathbf{f}$ where the old $\underline{c}$ is used to compute $\mathbf{f}$ in the iterations. Keeping in mind that the chaotic finance (2.1) is also highly nonlinear on some interval, and in order to speed up convergence, we suggest the use of a splitting method on the interval $[0, T]$ into N-subintervals $0 = t_0 < t_1 < \ldots < t_N = T$ and apply the robust spectral integral method.

The results are implemented for $a = 0.9, b = 0.2, c = 1.2$. Fig. 1 shows the chaotic behaviour of the finance system as expected. The solution functions $x(t), y(t), z(t)$ are plotted in Fig. 2 where a 4-domains decomposition has been used with 8 collocation points per domain.

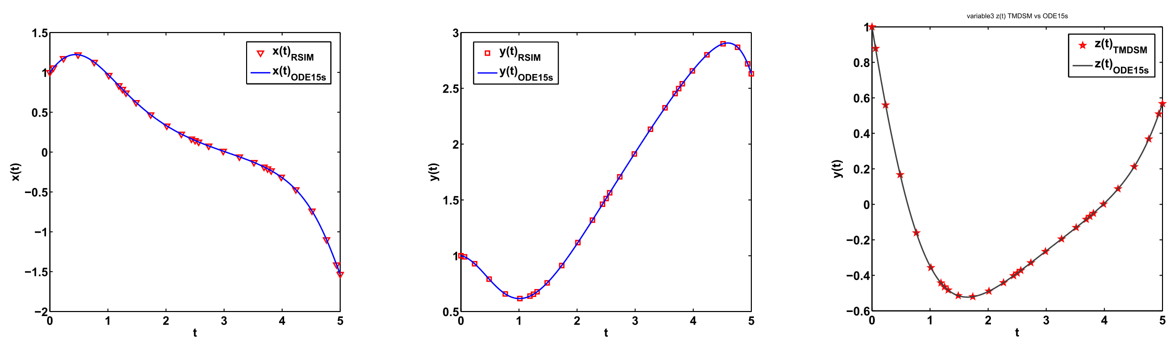
As we vary the number of collocation points from $n = 4, 8, 16, 32, 64, 128, 256$ and 512, we record in Table 1 the error on each variable x, y and z.

Given that Chebfun returns the solution in 1.05 s, we also record the accuracy achieved as the running time increases and with respect to the number of domains (here we consider 1, 2 and 4 subintervals), for a total number of Chebyshev collocation points varying from 256, 512, 1024, 2048 and 4096. Lets also record the error as well as the CPU running time in the Table 2. Fig. 3 shows the efficiency, on one variable,



(a) xy-plot (b) xz-plot (c) yz-plot

Fig. 1 Phase portraits for $T = 1000$.



(a) $x(t)$ (b) $y(t)$ (c) $z(t)$

Fig. 2 Plot of the 3 variables for $T = 5$ using 4-domains decomposition.

Table 1 Convergence of the error of the variables x, y and z fot $T = 1$ with 1 domain only.

	$n = 4$	$n = 8$	$n = 16$	$n = 32$	$n = 64$	$n = 128$	$n = 256$	$n = 512$
$x(t)$	2.390E−2	4.528E−4	1.992E−5	1.013E−6	5.844E−8	3.501E−9	2.145E−10	1.158E−11
$y(t)$	2.031E−2	4.484E−4	1.673E−5	9.079E−7	5.203E−8	3.139E−9	1.918E−10	1.0752E−11
$z(t)$	6.135E−3	1.001E−4	3.990E−6	2.2074E−7	1.182E−8	7.127E−10	4.361E−11	2.570E−12

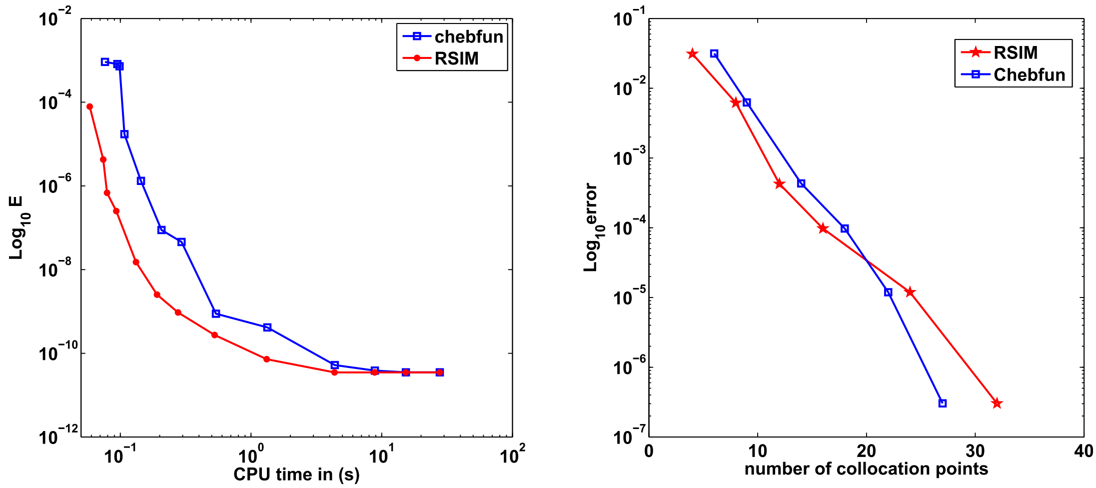
Table 2 Convergence and Efficiency of RSIM with 1, 2 and 4 domains at $T = 5$.

		$n = 256$	$n = 512$	$n = 1024$	$n = 2048$	$n = 4096$
$N = 1$	CPU	0.271	0.746	2.543	10.593	40.02
	Error	6.925E−9	4.304E−10	2.570E−11	3.431E−12	2.814E−12
$N = 2$	CPU	0.128	0.344	0.998	3.823	15.508
	Error	1.372E−8	8.429E−10	E−11	4.231E−12	2.961E−12
$N = 4$	CPU	0.104	0.174	0.525	1.525	5.451
	Error	3.646E−8	2.189E−9	1.348E−10	9.730E−12	2.762E−12

of our method compare to Chebfun in solving the chaotic finance system. The method is reliable on this problem.

Clearly the discretization of our interval $[0, T]$ is uniform, that is, $h_N = h = \frac{T}{N}$, where N is the number of domains. In addition we consider $M_N = M$ to be constant since we generate the same number of Chebyshev points in each domain. Moreover, it is not difficult to see that our function f here adheres to the Lipschitz conditions. Therefore we should expect an exponential decay of the error as shown in Fig. 4b where we also considered an additional case of $N = 8$ domains.

Table 2 shows that as the number of collocation points get larger (here $n > 1000$) on each subinterval, the method tends to suffer in terms of rapidity. Indeed matrix A gets very large, making inversion a complicated task. But if the structure of



(a) Efficiency (b) convergence

Fig. 3 Convergence and efficiency RSIM vs Chebfun.

A gets more porous (increase in sparsity) then the spectral method would still be capable of handling an even larger problem without losing much in accuracy. This explains why in the Table 2, our algorithm, RSIM, performs faster in larger time scale when splitting is done (see Fig. 5).

As mentioned earlier, the chaotic finance system is highly sensitive to the initial conditions, which can be a problem for an economical system. Controlling such system is of great importance in order to match a desired way of functioning. This is achieved by means of synchronization.

5. Synchronization

This section is devoted to the synchronization mentioned earlier. The sliding mode is applied using two controller parameters. We depart the section by considering our driving system with variables (x, y, z) to be the initial finance system (2.1) and let the response system be defined with variables x_s, y_s, z_s in the following way:

$$\begin{cases} \dot{x}_s = z_s + (y_s - a)x_s & +u_1(t) \\ \dot{y}_s = 1 - by_s - x_s^2 & +u_2(t) \\ \dot{z}_s = -x_s - cz_s & +u_3(t). \end{cases} \tag{5.67}$$

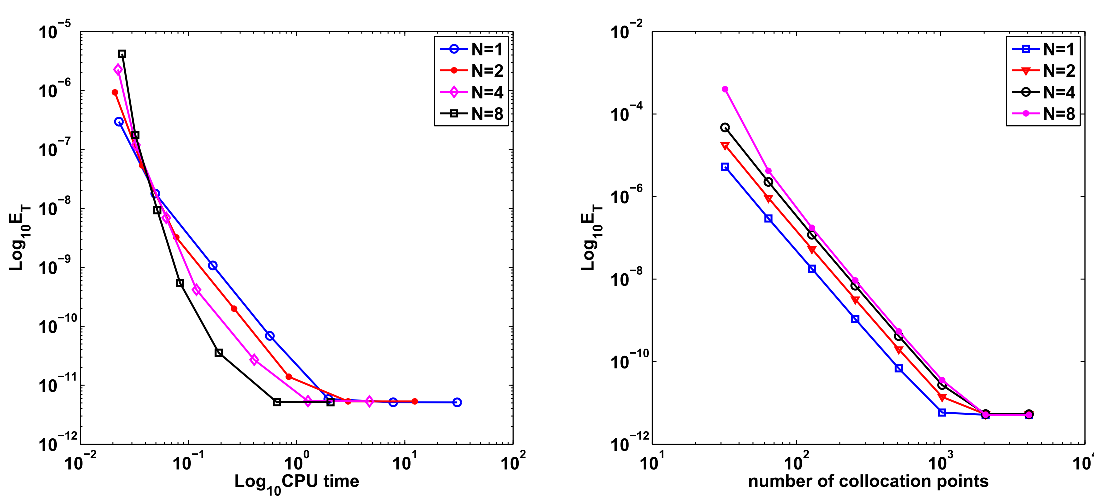
where $u = (u_1, u_2, u_3)$ is a suitable sliding control function to be determined in order to achieve synchronization. The error function from solution $e = (e_1, e_2, e_3)$ is defined by

$$\begin{cases} e_1 = x_s - x \\ e_2 = y_s - y \\ e_3 = z_s - z. \end{cases} \tag{5.68}$$

The dynamics of the error is thus driven by

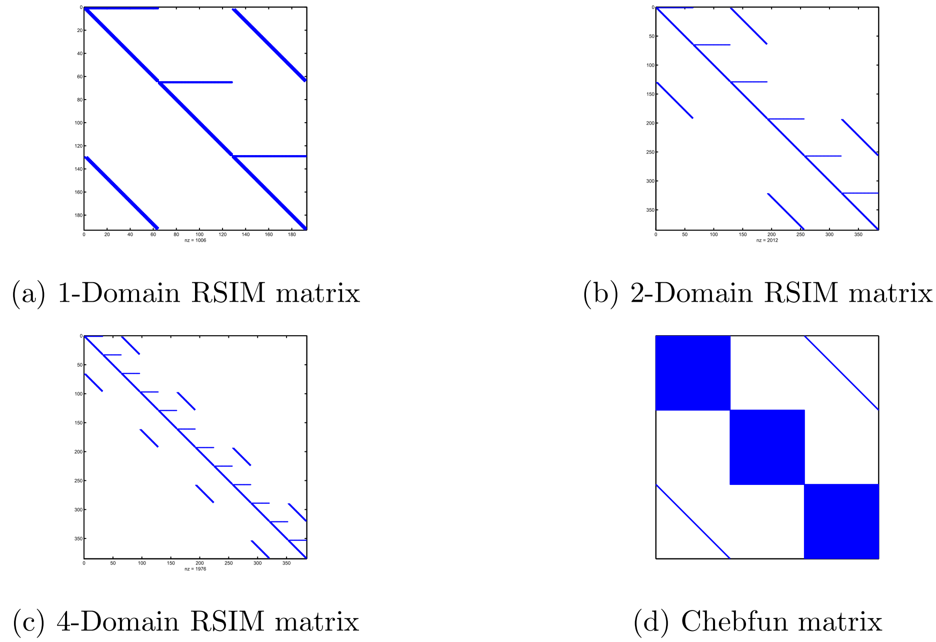
$$\begin{cases} \dot{e}_1 & = e_3 - ae_1 + x_s y_s - xy + u_1(t) \\ \dot{e}_2 & = -be_2 - x_s^2 + x^2 + u_2(t) \\ \dot{e}_3 & = -e_1 - ce_3 + u_3(t). \end{cases} \tag{5.69}$$

From sliding mode control theory, Kocamaz et al. [13] show that in order to achieve synchronization while maintain the



(a) Efficiency (b) Convergence

Fig. 4 Convergence and efficiency as we vary the number of domains on x-variable.



(a) 1-Domain RSIM matrix (b) 2-Domain RSIM matrix
(c) 4-Domain RSIM matrix (d) Chebfun matrix

Fig. 5 Plots of the underlying matrix A for 1, 2, and 4-Domain RSIM vs Chebfun matrix.

system stable, the required sliding control function u must satisfy

$$\begin{cases} u_1(t) = -x_s y_s + xy + v(t) \\ u_2(t) = x_s^2 - x^2 + v(t) \\ u_3(t) = 0, \end{cases} \tag{5.70}$$

where $v(t) = a_1(x_s - x) - b_1(y_s - y) + c_1(z_s - z) - q\,\text{sign}(-1.75(x_s - x) + 2.75(y_s - y))$ and $a_1 = 1.75(k - a)$, $b_1 = 2.75(k - b)$, $c_1 = 1.75$, and k, q are some parameters to be adjusted. In other words, sliding mode control achieves synchronisation only requires to act on interest rates and investment demand. Introducing all this back into (5.67) we obtain

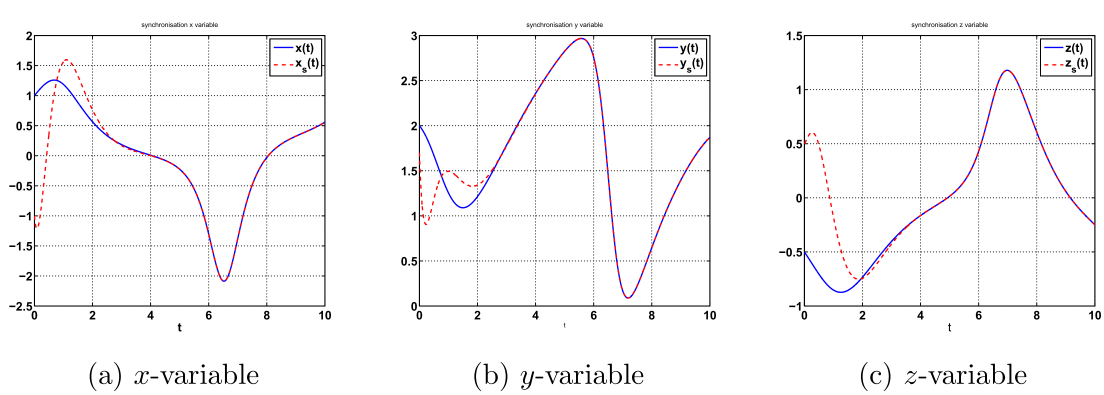
$$\begin{aligned} &\dot{x}_s + (a - a_1)x_s + b_1 y_s - (1 + c_1)z_s = -a_1 x + b_1 y - c_1 z + xy \\ &-q\,\text{sign}((-1.75(x_s - x) + 2.75(y_s - y)) \\ &\dot{y}_s + a_1 x_s + (b_s + b)y_s + c_1 z_s = -a_1 x + b_1 y - c_1 z + 1 - x^2 \\ &-q\,\text{sign}(-1.75(x_s - x) + 2.75(y_s - y)) \\ &\dot{z}_s + x_s + c z_s = 0. \end{aligned}$$

Applying integration and expressing it in a matrix form as in Section 4 yields

$$\begin{bmatrix} \mathcal{I} + (a - a_1)J & b_1 J & -(1 + c_1)J \\ -a_1 J & \mathcal{I} + (b + b_1)J & -c_1 J \\ J & 0 & \mathcal{I} + cJ \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} = \begin{bmatrix} \mathbf{f}^1 \\ \mathbf{f}^2 \\ \mathbf{f}^3, \end{bmatrix} \tag{5.71}$$

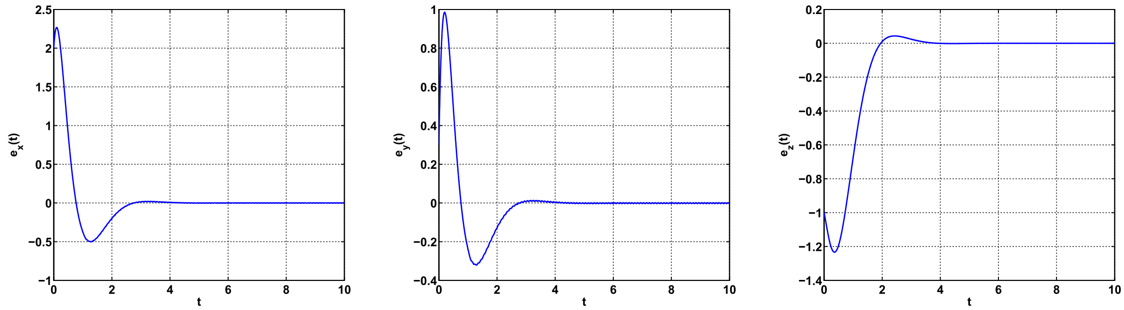
where $\mathbf{f}^1$ is coefficient vector of $-a_1 x + b_1 y - c_1 z + xy - q\,\text{sign}(-1.75(x_s - x) + 2.75(y_s - y))$ and $\mathbf{f}^2$ is coefficient vector of $-a_1 x + b_1 y - c_1 z + 1 - x^2 - q\,\text{sign}(-1.75(x_s - x) + 2.75(y_s - y))$.

With a driving system starting at initial condition $[1, 2, -0.5]$, and a response system starting with initial condition $[-1, 1.7, 0.5]$, and a time factor varying from 0 to 10, we see from Fig. 6 that synchronization is achieved quite fast from $t = 3.5$ on all three variables x, y, z. In what follows we plot the error dynamics function $e_1 = x_s - x$, $e_2 = y_s - y$ and $e_3 = z_s - z$ in Fig. 7. Again, we have a better confirmation of that actually from $t = 4$, the two systems behave the same.



(a) x-variable (b) y-variable (c) z-variable

Fig. 6 Drive and response system behaviour for $k = 5, q = 0.1$ and $0 \leqslant t \leqslant 10$.



(a) error in the x-variable (b) error in the y-variable (c) error in the z-variable

Fig. 7 Error behaviour for $k = 5$, $q = 0.1$ and $0 \leqslant t \leqslant 10$.

In other words, the controller defined in (5.70) is switched at time $t = 4$. Making therefore the chaotic finance system (2.1) rapidly controllable.

6. Conclusion

In this paper, we proposed a time multiple domain spectral method based on integration matrices to solve chaotic finance system. We first investigated the efficiency and the convergence of our method for different number of domains. In all the cases, the method displayed exponential convergence. We also compared the performance our method to that of Chebfun. We achieve good accuracy in very short time as result of using integration in the frequency space. The method also proves to be reliable for synchronization of chaotic finance system. We are currently extending the method to tackle fractional chaotic finance systems.

Declaration of Competing Interest

The authors declare that they have no conflict of interest.

References

[1] M.S. Abd-Elouahab, N.-E. Hamri, J. Wang, Chaos control of a fractional-order financial system, Math. Probl. Eng. 2010 (2010).
[2] E. Babolian, M. Hosseini, A modified spectral method for numerical solution of ordinary differential equations with non-analytic solution, Appl. Math. Comput. 132 (2) (2002) 341–351.
[3] A. Bhrawy, A. Alofi, The operational matrix of fractional integration for shifted chebyshev polynomials, Appl. Math. Lett. 26 (1) (2013) 25–31.
[4] C. Chen, T. Fan, B. Wang, Inverse optimal control of hyperchaotic finance system, World J. Model. Simul. 10 (2) (2014) 83–91.
[5] J. Ding, W. Yang, H. Yao, A new modified hyperchaotic finance system and its control, Int. J. Nonlinear Sci. 8 (1) (2009) 59–66.
[6] E. Doha, A. Bhrawy, S. Ezz-Eldien, A Chebyshev spectral method based on operational matrix for initial and boundary value problems of fractional order, Comput. Math. Appl. 62 (5) (2011) 2364–2373.
[7] T.A. Driscoll, Automatic spectral collocation for integral, integro-differential, and integrally reformulated differential equations, J. Comput. Phys. 229 (17) (2010) 5980–5998.
[8] G.N. Elnagar, M. Kazemi, Chebyshev spectral solution of nonlinear volterra-hammerstein integral equations, J. Comput. Appl. Math. 76 (1) (1996) 147–158.
[9] B. Fornberg, T.A. Driscoll, A fast spectral algorithm for nonlinear wave equations with linear dispersion, J. Comput. Phys. 155 (2) (1999) 456–467.
[10] N. Gurefe, E.G. Kocer, Y. Gurefe, Chebyshev-tau method for the linear klein-gordon equation, Int. J. Phys. Sci. 7 (43) (2012) 5723–5728.
[11] Y.C.J. Ma, Study for the bifurcation topological structure and the global complicated character of a kind of non-linear finance system (i), Appl. Math. Mech. 22 (11) (2001) 1240–1251.
[12] Y.C.J. Ma, Study for the bifurcation topological structure and the global complicated character of a kind of non-linear finance system (ii), Appl. Math. Mech. 22 (11) (2001) 1375–1382.
[13] U.E. Kocamaz, A. Göksu, H. Taşkın, Y. Uyaroğlu, Synchronization of chaos in nonlinear finance system by means of sliding mode and passive control methods: a comparative study, Inform. Technol. Control 44 (2) (2015) 172–181.
[14] F. Liu, X. Ye, X. Wang, Efficient chebyshev spectral method for solving linear elliptic pdes using quasi-inverse technique, Numer. Math.: Theory, Methods Appl. 4 (02) (2011) 197–215.
[15] J.H. Park, Chaos synchronization of a chaotic system via nonlinear control, Chaos, Solit. Fract. 25 (3) (2005) 579–584.
[16] R.B. Platte, N. Trefethen, Chebfun: a new kind of numerical computing, Prog. Industr. Math. at ECMI 2008. Math. Indust., 15, 69–87, 2010.
[17] A. Sambas, M. Mamat, A.A.A.G.M. Mahmoud, M.A. Mohamed, W.M. Sanjaya, A new chaotic system with line of equilibria: dynamics, passive control and circuit design, Int. J. Electr. Comput. Eng. (IJECE), 9 (4), 2019, pp. 2365–2376.
[18] J. Shen, T. Tang, L.-L. Wang, Spectral Methods: Algorithms, Analysis and applications, volume 41, Springer Science & Business Media, 2011.
[19] L.N. Trefethen. Spectral methods in MATLAB, volume 10. Siam, 2000.
[20] D. Trif, Matrix based operatorial approach to differential and integral problems, In MATLAB-A Ubiquitous Tool for the Practical Engineer, IntechOpen, 2011.
[21] S. Vaidyanathan, Global chaos synchronisation of identical li-wu chaotic systems via sliding mode control, Int. J. Model., Ident. Control 22 (2) (2014) 170–177.
[22] S. Vaidyanathan, S.T. Kingni, A. Sambas, M.A. Mohamed, M. Mamat, A new chaotic jerk system with three nonlinearities and synchronization via adaptive backstepping control, Int. J. Eng. Technol. 7 (3) (2018) 1936–1943.
[23] S. Vaidyanathan, A. Sambas, M. Mamat, A new chaotic system with axe-shaped equilibrium, its circuit implementation and adaptive synchronization, Arch. Control Sci. 28 (2018).
[24] S. Vaidyanathan, L.G. Dolvis, K. Jacques, C.-H. Lien, A. Sambas, A new five-dimensional four-wing hyperchaotic system with hidden attractor, its electronic circuit realisation and

synchronisation via integral sliding mode control, Int. J. Model., Ident. Control 32 (1) (2019) 30–45.
[25] Z.-Q. Wang, J. Mu, A multiple interval chebyshev-gauss-lobatto collocation method for ordinary differential equations, Numer. Math.: Theory, Methods Appl. 9 (4) (2016) 619–639.
[26] J. Weideman, S. Reddy, A matlab differentiation matrix suite, ACM Trans. Math. Software (TOMS) 26 (4) (2000) 465–519.
[27] M. Yang, B. Cai, G. Cai, Projective synchronization of a modified three-dimensional chaotic finance system, Int. J. Nonlinear Sci. 10 (1) (2010) 32–38.
[28] X. Zhao, Z. Li, S. Li, Synchronization of a chaotic finance system, Appl. Math. Comput. 217 (13) (2011) 6031–6039.
[29] S. Zheng, G. Dong, Q. Bi, Adaptive modified function projective synchronization of hyperchaotic systems with unknown parameters, Commun. Nonlinear Sci. Numer. Simul. 15 (11) (2010) 3547–3556.