World Journal of Mechanics, 2012, 2, 334-338
doi:10.4236/wjm.2012.26039 Published Online December 2012 (http://www.SciRP.org/journal/wjm)

Stability Control of Stretch-Twist-Fold Flow by Using Numerical Methods

Shahab Ud-Din Khan[1], **Yonglu Shu**[1], **Salah Ud-Din Khan**[2]
[1]Department of Mathematics and Statistics, Chongqing University, Chongqing, China
[2]Sustainable Energy Technologies Center, King Saud University, Riyadh, KSA
Email: shahab.furqan@gmail.com

Received September 21, 2012; revised October 24, 2012; accepted November 5, 2012

ABSTRACT

In this study, the multistep method is applied to the STF system. This method has been tested on the STF system, which is a three-dimensional system of ODE with quadratic nonlinearities. A computer based Matlab program has been developed in order to solve the STF system. Stable and unstable position of the system has been analyzed graphically and finally a comparison as well as accuracy between two-step sizes with detail. Newton's method has been applied to show the best convergence of this system.

Keywords: STF System; Chaos; Modified Method; Fixed Point Iteration Method; Newton's Method

1. Introduction

The distinction between slow and fast dynamos was first drawn by Vainshtein & Zeldovich (1972) in this research; we describe the stretch-twist-fold (STF) fast dynamo, which is the archetype of the elementary models of the process. Basically, stretch-twist-fold is applied in fluid mechanics in aerospace. In space, any fluid can be D-Tracked easily so a magnetic field is required to compel the fluid to be in the same orbit and this method is called STF system. In this paper, we will investigate the accuracy of numerical method. The Multistep method was first introduced by Goldstine, Herman H. in the beginning of 1977's. This iterative method has proven rather successful in dealing with various scientific problems [1-4] since it provides analytical solutions, which is a standard numerical method. This method has also been applied to solve nonlinear systems of ordinary differential equations. For example, H. B. Keller [5] presented an extensive comparative study on the accuracy of the multistep method and C. Lubich [6] studied the effects of time steps on the stiff problem. J. O. Fatokun and I. K. O. Ajibola [7] studied multistep method for integrating ordinary differential equations on manifolds. Differential equations are used to model problems in science and engineering that involve the change of some variables with respect to another. Most of their problems require the solution to an initial-value problem that is the solution to a differential equation that satisfies a given initial condition. In most real-life situations the differential equation that models the problem is too complicated to solve exactly and one of two approaches is taken to approximate the solution. The first approach is to simplify the differential equation to one that can be solved exactly and then use the solution of the simplified equation to approximate the solution to the original equation. The other approach, which we will examine in this paper, uses methods for approximating the solution of the original problem. This is the approach that is most commonly taken, since the approximation methods give more accurate results and realistic error information. The objective of this research is to solve STF system and test nonlinear behavior with different time steps. This modified method is able to find a stable and unstable position of STF system. This method can also give the exact values after iteration results. Newton's method is able to show the best convergence than fixed point iteration method.

2. Stretch-Twist-Fold Flow (STF)

The STF flow is defined as

$$\begin{aligned}\dot{x}(t) &= \alpha z - 8xy,\\ \dot{y}(t) &= 11x^2 + 3y^2 + z^2 + \beta xz - 3,\\ \dot{z}(t) &= -\alpha x + 2yz - \beta xy,\end{aligned} \tag{1}$$

where $\alpha = 0.1$, $\beta = 1$ are positive real parameters and related to the ratios of intensities of the stretch, twist and fold ingredients of the flow.

3. Description of Methods

The methods we consider in this section do not produce a continuous approximation to the solution of the initial-value problem. Rather, approximations are found at certain specified and often equally spaced points. Some method of interpolation is used if intermediate values are needed. We need some definitions and results from the theory of ordinary differential equations before considering methods for approximating the solutions to initial-value problems.

Definition 3.1: A function $f(x,y)$ is said to satisfy a Lipschtiz condition in the variable y on a set

$$D = \{(x,y) \mid a \le x \le b, -\infty < y < +\infty\}$$

If a constant $L > 0$ exists with the property that

$$\left|f(x,y) - f(x,y^*)\right| \le L\left|y - y^*\right|$$

$$\forall (x,y),(x,y^*) \in D$$

This first part of this section is concerned with approximation the solution $y(x)$ to a problem of the form

$$\frac{\mathrm{d}y}{\mathrm{d}x} = f(x,y), \text{ for } a \le x \le b$$

Subject to an initial conditions $y(a) = y_0$.

Lemma 3.1: Suppose that $f(x,y)$ is continuous on D if f satisfies a Lipschitz condition on D in the variable y,

Then the initial-value problem

$$\begin{cases} \dfrac{\mathrm{d}y}{\mathrm{d}x} = f(x,y), \text{ for } a \le x \le b \\ y(a) = y_0 \end{cases}$$

has a unique solution $y(x)$ for $a \le x \le b$.

The methods of Euler and Runge-kutta are called one-step methods because the approximation for the mesh point $x_i + 1$ involves information from only one of the previous mesh points x_i although these methods can use functional evaluation information at points between x_i and $x_i + 1$, they do not retain that information for direct use in future approximations. All the information used by these methods is obtained within the subinterval over which the solution is being approximated. Since the approximate solution is available at each of the mesh points $x_0, x_1, \cdots, x_i$ before the approximation at $x_i + 1$ is obtained and because the error $\left|y_{i+1} - y(x_{i+1})\right|$ tends to increase with I, it seems reasonable to develop methods that these more accurate previous data when approximation the solution at $x_i + 1$.

Methods using the approximation at more than one previous mesh point to determine the approximation at the next point are called multistep methods.

Definition 3.2: An m-step multistep method for solving the initial-value problem (3.1) is one whose difference equation for finding the approximation y_{i+1}.

At the mesh point $x_i + 1$ can be represented by the following equation,

where p is an integer greater than 0

$$\begin{aligned} y_{i+1} &= a_0 y_1 + a_1 y_{i-1} + \cdots + a_m y_{i-m+1} + h\left[b_{-1} f(x_{i+1}, y_{i+1})\right. \\ &\quad \left. + b_0 f(x_i, y_i) + \cdots + b_m f(x_{i-m+1}, y_{i-m+1})\right] \qquad \text{(A)} \\ &= \sum_{j=0}^{m-1} a_j y_{i-1} + h \sum_{j=-1}^{m-1} b_j f_{i-j} \end{aligned}$$

When $b_{-1} = 0$ then the method is called explicit or open. Since Equation (A) then gives y_{i+1} explicitly in terms of previously determine values. When $b_{-1} \ne 0$ then the method is called implicit or closed. Since y_{i+1} occurs on both sides of Equation (A) and is specified only implicitly.

To begin the derivation of the multistep methods, note that the solution to the initial-value problem (3.1), if integrated over the interval $[x_i, x_{i+1}]$ has the property that

$$y(x_{i+1}) = y(x_i) + \int_{x_i}^{x_{i+1}} f(x, y(x))\mathrm{d}x \qquad \text{(B)}$$

Since we cannot integrate $f(x, y(x))$ without knowing $y(x)$ the solution to the problem, we instead integrate an interpolating $L(x)$ to $f(x, y(x))$ that is determined by some of the previously obtained data points $(x_0, y_0), (x_1, y_1), \cdots, (x_i, y_i)$ Equation (B) becomes

$$y(x_i) \approx y_i + \int_{x_i}^{x_{i+1}} L(x)\mathrm{d}x$$

3.1. Modified Method

Use the modified APC method to solve STF system. This method is derived from ABF-Explicit m-step technique and AM-Implicit m-step technique. The simulation done of this paper is for the time range $t \in [0,1]$ with two time steps $\Delta t = 0.01$ and $\Delta t = 0.001$.

Represented formula:

$$\bar{y}_{n+1} = y_n + h(b_1 g_1 - b_2 g_2 + b_3 g_3 - b_4 g_4)$$

$$y_{n+1} = y_n + h(b_4 g_5 + b_5 g_1 - b_6 g_2 - b_7 g_3)$$

where,

$$b_1 = 55, b_2 = 59, b_3 = 37, b_4 = 9$$
$$b_5 = 19, b_6 = 5, b_7 = 1.$$

$$g_1 = f_n, g_2 = f_{n-1}, g_3 = f_{n-2},$$
$$g_4 = f_{n-3}, g_5 = \bar{y}_{n+1}.$$

3.2. Unstable Position

When $\alpha = 1$, $\beta = 0.1$ and $h = 0.01$ then we can determine the unstable position of the system that is shown in **Table 1** and easily analyzed by the **Figure 1**.

3.3. Stable Position

When $\alpha = 0.1$, $\beta = 1$ and $h = 0.001$ then we can determine the stable position of the system that is shown in **Table 2** and easily analyzed by the **Figure 2**.

Table 1. ***X, Y, Z*****-Direction for *β* = 0.1.**

T	Δx	Δy	Δz
0	0	0	0
0.1	0.0533	0.5102	0.0081
0.2	0.0801	0.8065	0.0210
0.3	0.0947	0.8007	0.0386
0.4	0.1035	0.7961	0.0605
0.5	0.1097	0.7909	0.0863
0.6	0.1147	0.7854	0.1155
0.7	0.1191	0.7796	0.1473
0.8	0.1233	0.7734	0.1811
0.9	0.1275	0.7670	0.2158
1	0.1319	0.7602	0.2507

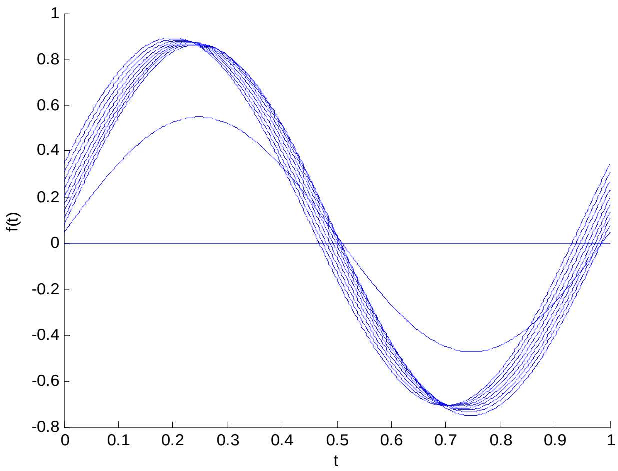


Figure 1. The unstable position of the system when *α* = 1, *β* = 0.1 and *h* = 0.01.

Table 2. ***X, Y, Z*****-Direction for *β* = 1.**

T	Δx	Δy	Δz
0	0	0	0
0.1	0.0051	0.0508	0.0001
0.2	0.0100	0.1013	0.0004
0.3	0.0146	0.1511	0.0004
0.4	0.0190	0.1998	0.0006
0.5	0.0232	0.2472	0.0009
0.6	0.0271	0.2929	0.0012
0.7	0.0309	0.3367	0.0016
0.8	0.0345	0.3785	0.0021
0.9	0.0379	0.4180	0.0025
1	0.0411	0.4551	0.0031

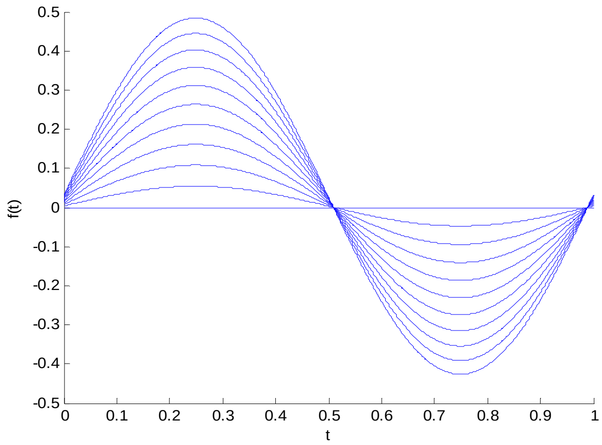


Figure 2. The stable position of the system when *α* = 0.1, *β* = 1 and *h* = 0.001.

4. Fixed Points for Function of Several Variables

In this section, we will discuss about fixed point iteration method and Newton's method.

A system of nonlinear equations has the form

$$\begin{cases} f_1(x_1, x_2, \cdots, x_n) = 0 \\ f_2(x_1, x_2, \cdots, x_n) = 0 \\ \vdots \\ f_m(x_1, x_2, \cdots, x_n) = 0 \end{cases}$$

Here each function f_i can be thought of as mapping a vector $x = (x_1, x_2, \cdots, x_n)^{\mathrm{T}}$ of n-dimensional space R^n into the real line R.

The system of n nonlinear equations in n unknowns can alternatively be represented by defining a function f, mapping R^n into R^n by $f = (f_1, f_2, \cdots, f_n)^{\mathrm{T}}$.

Then we have

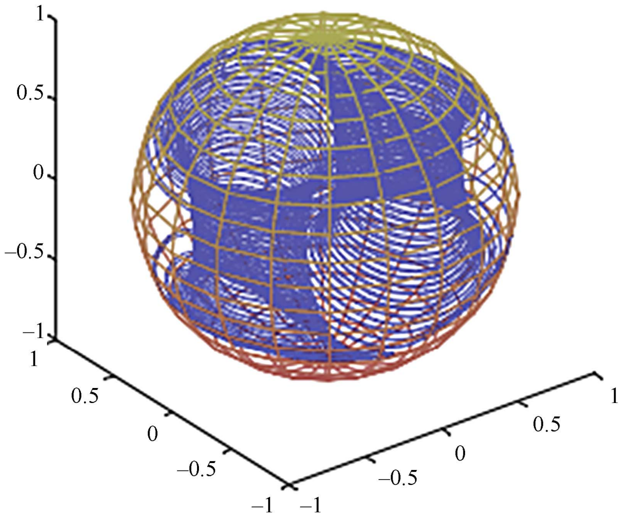
$$f(x) = 0$$

In an iterative process for solving an equation $f(x) = 0$ was developed by transforming the equation into one of the form $x = g(x)$. The function g is defined to have fixed points precisely at solutions to the original equation. A similar procedure will be investigated for function from R^n to R^n.

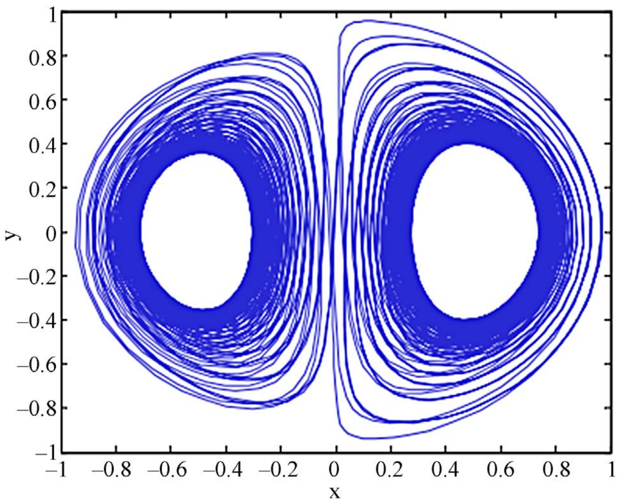
Definition 4.1: A function g from $D \subset R^n$ into R^n has a fixed point at $x^* \in D$ if $g(x^*) = x^*$.

Consider the STF system (see **Figure 3**):

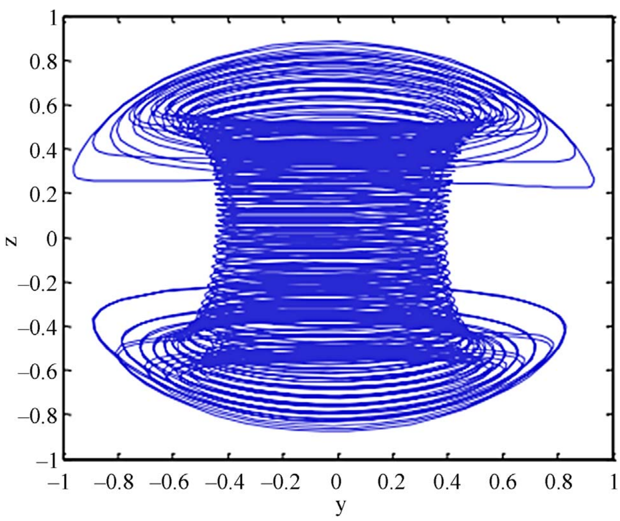
$$\begin{cases} x_1 = \dfrac{\alpha x_3}{8 x_2} \\ x_2 = \sqrt{1 - \left(\dfrac{11 x_1^2 - x_3^2 - \beta x_1 x_3}{3} \right)} \\ x_3 = \dfrac{1}{2} \left(\dfrac{\alpha x_1}{2 x_2} + \beta x_1 \right) \end{cases} \tag{C}$$



(a) *x-y-z*



(b) *x-y*



(c) *y-z*

Figure 3. Portrait and *x*, *y*, *z* direction of STF system.

To approximate the fixed point x^* we choose $x^0 = (0.3, 0.2, -0.1)^{\mathrm{T}}$ the sequence of vectors generated by

$$\begin{cases} x_1^{k+1} = \dfrac{\alpha x_3^k}{8 x_2^k} \\ x_2^{k+1} = \sqrt{1 - \left(\dfrac{11\left(x_1^2\right)^k - \left(x_3^2\right)^k - \beta x_1^k x_3^k}{3} \right)} \\ x_3^{k+1} = \dfrac{1}{2}\left(\dfrac{\alpha x_1^k}{2 x_2^k} + \beta x_1^k \right) \end{cases}$$

If $k = 1, 2, 3, 4$ then

$$x^1 = (0.00625, 0.82664, 0.22500)^{\mathrm{T}}$$
$$x^2 = (0.00340, 0.99136, 0.003503)^{\mathrm{T}}$$
$$x^3 = (0.00004, 0.99998, 0.00187)^{\mathrm{T}}$$

$$\left\| x^{k+1} - x^k \right\| = \left\{ 0.62664, 0.221497, 8.6 \times 10^{-3} \right\}$$

Now we have tested this system in Newton's method and comparison with fixed point iteration results.

Newton's method for systems, like the one-dimensional Newton's method, a fixed point iteration based on a linearization of $f(x)$. If $f : R^n \to R$ then the Taylor series for $f(x)$ has the form

$$f(x) = f\left(x^k\right) + J\left(x^k\right)\left(x - x^k\right) + E(x)$$

Newton's method is derived just as it was for the one-dimensional case: neglecting the remainder term, we have

$$f(x) \approx f\left(x^k\right) + J\left(x^k\right)\left(x - x^k\right)$$

And setting $f(x) = 0$ gives what we hope is an improved estimate

$$f\left(x^k\right) + J(x)\left(x - x^k\right) \approx 0$$

If $\det\left(J\left(x^k\right)\right) \neq 0,$ the iteration

$$x^{k+1} = x^k - \left[J\left(x^k\right)\right]^{-1} f\left(x^k\right), \quad k = 0, 1, 2, \cdots$$

This is Newton's method for systems.

Where, $\left[J\left(x^k\right)\right]^{-1}$ is the inverse of $J(x)$.

In practice it is preferable to solve $J\left(x^k\right)\Delta x^k = -f\left(x^k\right)$ for Δx^k and then add this quantity to x^k we have

$$x^{k+1} = x^k + \Delta x^k$$

where

$$\Delta x^k = x^{k+1} - x^k$$

By system (C), the Jacobi matrix $J(x)$ for this system is

Table 3. Convergence rate of STF.

k	x_1^k	x_2^k	x_3^k	$\left\|x^{k+1}-x^k\right\|$
0	0.3	0.2	–0.1	
1	0.007523	0.00430	0.00005	2.9×10^{-1}
2	0.962467	0.10136	0.10002	9.5×10^{-1}
3	0.338002	0.00450	0.00258	6.2×10^{-1}

$$J(x) = \begin{bmatrix} -8x_2 & -8x_1 & 0.1 \\ 22x_1 + x_3 & 6 & 2x_3 + x_1 \\ -0.1 - x_2 & 2x_3 - x_1 & 2x_2 \end{bmatrix}$$

The results are given in **Table 3**.

According to previous examples, we can easily analyze that Newton's method is more accurate than fixed point iteration method.

5. Conclusion

In this paper, MATLAB programming has been used to solve the STF system with variable time steps ($\Delta t = 0.01$, 0.001). We have obtained good results by using two methods applied to the STF system concerning the system is stable and unstable state. The modified method was computed by developing simple algorithm without perturbation techniques *i.e*. linearization or discretization. In all the considered cases, it has been proved that the modified multistep method appears to be the best method to approximate this solution based on its accuracy and Newton's method is a good example to solve root finding problem in STF system. Newton's method is able to show the best convergence than fixed point iteration method.

6. Acknowledgements

The first author is very thankful to all of his co-authors and especially to Professor Shu Yonglu for advising and giving me the opportunity to conduct this research and also very much thankful to Sustainable Energy Technologies Centre, King Saud University for funding the research.

REFERENCES

[1] C. Baker and E. Buckwar, "Numerical Analysis of Explicit One-Step Methods for Stochastic Delay Differential Equations," *LMS Journal of Computation and Mathematics*, Vol. 3, No. 3, 2000, pp. 315-335. doi:10.1112/S1461157000000322

[2] R. H. Bokor, "On Two-Step Methods for Stochastic Differential Equations," *Acta Cybernetica*, Vol. 13, No. 1, 1997, pp. 197-207.

[3] L. Brugnano, K. Burrage and P. Burrage, "Adams-Type Methods for the Numerical Solution of Stochastic Ordinary Differential Equations," *BIT Numerical Mathematics*, Vol. 40, No. 3, 2000, pp. 451-470. doi:10.1023/A:1022363612387

[4] E. Buckwar and R. Winkler, "On Two-Step Schemes for SDEs with Small Noise," *Proceedings in Applied Mathematics and Mechanics*, Vol. 4, No. 1, 2004, pp. 15-18. doi:10.1002/pamm.200410004

[5] H. B. Keller, "Approximation Method for Nonlinear Problem with Application to Two Point Value Boundary Problem," *Mathematics of Computation*, Vol. 29, No. 130, 1975, pp. 464-474.

[6] C. Lubich, "On the Convergence of Multistep Methods for Nonlinear Stiff Differential Equations," *Numerische Mathematik*, Vol. 61, No. 1, 1992, pp. 277-279. doi:10.1007/BF01385657

[7] J. O. Fatokun and I. K. O. Ajibola, "A Collocation Multistep Method for Integrating Ordinary Differential Equations on Manifolds," *African Journal of Mathematics and Computer Science Research*, Vol. 2, No. 4, 2009, pp. 51-55.