NORMAL FORMS, NONLOCAL CHAOTIC BEHAVIOR IN SPROTT AND NMR SYSTEMS

by

Nazım Ziya Perdahçı

B.S. in Physics, Boğaziçi University, 1993

M.S. in Physics, Boğaziçi University, 1996

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Doctor
of
Philosophy

Boğaziçi University

2002

To my father

ACKNOWLEDGEMENTS

I owe special thanks to my advisor Prof. Avadis Hacınlıyan for his understanding and support. Despite the fact that it is particularly hard to do theoretical work for a person of experimental physics background he accepted me to his research group and helped me in every way to become a theoretical physicist.

I also owe special thanks to Prof. Orhan Nalcıoğlu and his research group. They were very helpful during my stay in UCI.

I would like to thank the members of our research group, especially Dr. İnanç Birol, for their help and support in the initial years of the work.

I also would like to thank my dear friend Dr. Tonguç Rador for communicating on problems.

I am mostly grateful to Prof. Hilmi Demiray, who as our dean in Işık University, always gave special attention to the work load I am given so that I could finish this work.

My grateful thanks are due to my affectinate parents and my wife for their patience and never ending support.

I am indebted to Prof. Ömer Oğuz who agreed to be my thesis advisor after the retirement of Prof. Avadis Hacınlıyan from Boğaziçi University.

Support by TÜBİTAK under the BDP Joint Doctorate Program and Boğaziçi University Research Fund are gratefully acknowledged.

ABSTRACT

NORMAL FORMS, NONLOCAL CHAOTIC BEHAVIOR IN SPROTT AND NMR SYSTEMS

Poincaré's theory of normal forms is applied to a number of simple chaotic Sprott flows that have resonant eigenvalues. It is shown that the normal form expansion can give significant information not limited to the local properties of chaotic attractors, but also, on nonlocal properties such as positive and zero Liapunov exponents for systems that have the Hopf bifurcation property. Existence of a zero Liapunov exponent is indicated if the system has hyperbolic fixed points. The method is not directly applicable where an eigenvalue of the linearized part vanishes, because of the complexity of the normal form. Rational transformations that change the eigenvalue spectrum of the linearized parts are employed on the Sprott C and E systems to obtain simpler systems. A proposal on the possible use of fractal analysis methods on functional MRI data and preliminary results on possible sources of chaotic behavior inherent in nuclear spin systems are presented.

ÖZET

NORMAL FORMLAR, SPROTT VE NMR SİSTEMLERİNDE YEREL OLMAYAN DÜZENSİZ DAVRANIŞ

Poincaré'nin normal form kuramı rezonant özdeğerleri olan birkaç basit, kaotik Sprott akışına uygulanmıştır. Normal form açılımının sağladığı bilgi yalnızca yerel özelliklerle sınırlı değildir, Hopf dallanma özelliğine sahip sistemler için pozitif ve sıfır Liapunov üstelleri gibi yerel olmayan özellikler de kestirilebilmektedir. Sistemin hiperbolik denge noktaları varsa bir Liapunov üstelinin sıfır olabileceği görülmüştür. Yöntem, ortaya çıkan normal formun karmaşıklığı nedeniyle lineerleştirilmiş kısmın özdeğerlerinden birinin sıfır olduğu durumlarda doğrudan uygulanamamaktadır. Lineerleştirilmiş kısmın özdeğerlerini değiştiren rasyonel dönüşümler Sprott C and E sistemleri üzerinde uygulanarak daha basit sistemler elde edilebileceği gösterilmiştir. Fonksiyonel Manyetik Rezonans Görüntüleme verilerinin incelenmesinde kullanılabilecek bir fraktal analiz yöntemi önerilmiş, nükleer spin sistemlerinde olası kaotik davranış kaynakları hakkında ön sonuçlar takdim edilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

A	tangent flow matrix
$\tilde{A}$	complex diagonal tangent flow matrix
b	system parameter
$\mathbf{B}$	total magnetic field vector
$\mathbf{B_o}$	main static magnetic field vector
$\mathbf{B_1}$	radiofrequency magnetic field vector
B_H	Brownian function
c	system parameter
c_i	constant coefficients
C	correlation coefficient
D	diffusion constant
F_i	components of Taylor expansion of degree n
$\tilde{F}_i$	complex components of Taylor expansion of degree n
g	gyromagnetic ratio
$\mathbf{G}$	gradient field vector
G_k	the null space
h_{ij}	generating functions of the near identity transformation
$\mathbf{L}$	angular momentum vector
$\mathbf{M}$	net magnetization moment vector
n	system dimension
R	rescaled range
S	standard deviation of rescaled range
t	time
T_1	spin-lattice relaxation time constant
T_2	spin-spin relaxation time constant
u	state vector of dimension n for normal form system
$v(\cdot)$	system dynamics vector field
w	z component in cylindrical coordinates
W	z component of resonance functions in cylindrical coordinates

x	state vector of dimension n
$\hat{\mathbf{x}}_i$	ith unit vector
x_k	dynamical variable
$\bar{x}_k$	dynamical variable with equilibrium point shifted to the origin
x_o	homogeneous initial conditions or equilibrium points
X	position in random walk
y	state vector of dimension n for transformed system
α	real component of a complex eigenvalue
β	imaginary component of a complex eigenvalue
γ	purely real eigenvalue
ϵ	scaling parameter
ζ	complex state vector of dimension n
θ	angular component in cylindirical coordinates
λ_i	ith Liapunov exponent or eigenvalue
μ	magnetic moment vector
ξ	step length in random walk
ρ	radial component in cylindirical coordinates
τ	time interval
ω_r	resonant monomials
Θ	angular part of functions in cylindrical coordinates
Ω	radial part of resonance functions in cylindrical coordinates
DDI	Dipole Dipole Interaction
fMRI	Functional Magnetic Resonance Imaging
MR	Magnetic Resonance
MRI	Magnetic Resonance Imaging
NF	Normal Form
NMR	Nuclear Magnetic Resonance
NSS	Nuclear Spin System

1. INTRODUCTION

Nonlinear systems have attracted attention in recent years, since they may show a so-called chaotic behavior due to their sensitive dependence on initial conditions. The systematization of the different mechanisms for chaotic behavior is important for understanding nonlinear models of different physical systems. In order to determine if a system is chaotic, one first calculates the Liapunov spectrum. Chaotic systems have at least one positive Liapunov exponent. Since the calculation of the largest Liapunov exponent is usually easier than the calculation of the whole spectrum, many authors [1, 2, 3] consider only the largest exponent. Since it is very hard to show the existence of a positive exponent algebraically, practical methods for Liapunov exponent calculation involve numerical approaches [4]. Numerical algorithms can be applied to an arbitrary dynamical system, but they are limited by problems in computational stability, convergence and truncation errors. There is also the question of choosing the proper numerical integration algorithm [5]. Zero Liapunov exponents are also important since they have a direct bearing on the dimension of the attractor through the Kaplan-Yorke conjecture [6]. They may affect the dimensionality of the central manifold, if any. Zero Liapunov exponents may also serve as indicators for conserved quantities. Finally, if the trajectory of an attractor does not contain a fixed point at least one Liapunov exponent vanishes [7].

We see that both positive and zero Liapunov exponents are important for quite different reasons. However, errors associated with numerical simulations may leave one in doubt as to whether we have a zero exponent, a slightly positive or slightly negative exponent. For this reason, alternative methods to confirm a zero exponent would be extremely useful. A frequently used approach is to study a chaotic system near an equilibrium point by linearizing it about this point. If the trajectory near the fixed point is a center or focus, this linearized system has complex conjugate pairs of eigenvalues. If this argument could be reversed such that complex conjugate pairs of eigenvalues near an equilibrium point would signal an attractor with no fixed points and hence a zero Liapunov exponent, this would give a very simple criterion.

An attractive approach for estimating zero and perhaps positive Liapunov exponents could, in principle, be based on finding a polynomial transformation which maps the tangent space of a given dynamical system to a special system with a zero (or perhaps positive) Liapunov exponent. In very special cases, such a scheme can be found [8, 9]. Many of the examples in reference [8] can also be integrated by alternative analytical methods. One drawback of such a method is its reliance on local behavior for estimating quantities where a global average is taken over fiducial trajectories. An attempt to make the study more systematic [10] by focusing on the class of systems proposed by Sprott [11] fails to give anything beyond hints on two possible mechanisms for transition to chaotic behavior, based on the number and character of fixed points. Another possibility is the normal form approach. This does furnish a sequence of polynomial transformations that reduce the system to a simpler system, whose structure is determined by resonances dictated by the eigenvalues of the linearized system near an equilibrium point. NF expansions give the local properties of chaotic behavior correctly, but have convergence problems [12]. It may sometimes be desirable to change the resonant structure of the base system by applying nonpolynomial transformations [13], or using rational transformations that introduce a manifold over which the system survives in long time scales in a way analogous to the center manifold.

The purpose of this investigation is to give a systematic treatment of the NF approach in order to understand the transition to chaos and the Liapunov spectrum for systems that exhibit the Hopf bifurcation property [14]. In Part 2, we review the NF methods that are used in the rest of this work [15]. In Part 3, specific results on three Sprott systems are presented, these results help one understand the transition to chaos as reflected by the NF approach and imply a non-unique averaging technique over the attractor for estimating the largest Liapunov exponent. Part 4 summarizes results on systems that do not have the Hopf bifurcation property. Part 5 proposes a rational polynomial transformation which can change the eigenvalue spectrum of the linearized part. Part 6 discusses a proposal for fractal data analysis in fMRI data and also includes a discussion of possible chaoticity in nuclear spin systems. Conclusions are given in Part 7.

Five appendices are provided. The first one explains the symbolic algebra calculations used in the present thesis. Programs for calculating standard nonresonant NFs, resonant NFs and nullspaces for resonant systems are given in Appendices B,C, and D. Appendix E includes the NF expansion for the Sprott C system up to the fifth order.

2. THE METHOD OF NORMAL FORMS

Our objects of investigation in this work are autonomous, continuous time, dynamical systems given by the set of differential equations

$$\dot{x} = v(x) \ , \tag{2.1}$$

where $x \in \mathcal{R}^n$, $t \geq 0$, $v : \mathcal{R}^n \to \mathcal{R}^n$ and the initial values $x(t = 0)$ are given. We wish to consider the system near its fixed points x_0, given by $v(x_0) = 0$. Without loss of generality, one can move the fixed point to the origin by $\bar{x} = x - x_0$. Drop bars and let $\{x_i, i = 1..n\}$ and $\{v_{i,j}\}$ denote the components of x and v. Taylor expansion near $x = 0$ followed by an orthogonal transformation brings Equation 2.1 to the form [16]

$$\dot{x} = Ax + F_2(x) + F_3(x) + ...F_N(x) + ... \tag{2.2}$$

where A is a matrix of constants in either diagonal or Jordan canonical form, $\{F_i, i = 1, 2...\}$ are homogeneous polynomials of degree i. We then transform Equation 2.2 by a near identity, polynomial transformation

$$x_i = u_i + h_{i2}(\{u_k\}) + ... + h_{iM}(\{u_k\}) + ... \tag{2.3}$$

where $u \in \mathcal{R}^n$, $h_{iM}(\{u_k\}) : \mathcal{R}^n \to \mathcal{R}^n$ are homogeneous polynomials of degree M. These polynomials are determined order by order so that F_i are eliminated as far as possible. The condition for the elimination of i'th order terms is [16]

$$Dh_i(u)Au - Ah_i(u) = F_i(u). \tag{2.4}$$

Equation 2.4 is called the homological equation. Note that $L_A = Ah_i(u) - Dh_i(u)Au$ is the Lie Bracket of $h_i(u)$ and Au. These equations form a linear system in a space spanned by vector valued monomials of degree i $(\prod_k u^{m_k})s_j$, where $m_j \geq 0$, $\sum_k m_k = i$, s_j are arbitrary vectors in $\mathcal{R}^n$. Homological equations can be completely solved order

by order, and the system 2.1 can be reduced to a set of linear NF equations

$$\dot{u} = Au \,, \tag{2.5}$$

if the eigenvalues λ_j of the linearized part A do not satisfy a resonance condition of the form $\lambda_j = \sum_{k=1}^{n} m_k \lambda_k$, for some positive or zero integers m_k and $\sum m_k \geq 2$. The integer $\sum m_k$ is called the order of the resonance. This is the fundamental result of Poincaré's dissertation [17].

If there exists eigenvalues that satisfy resonance relations the Poincaré-Dulac theorem asserts that all non-resonant terms in 2.1 can be eliminated by a near identity transformation which reduces the system to the form

$$\dot{u} = Au + \omega_r(u) \,, \tag{2.6}$$

where ω_r consists of only resonant terms.

The eigenvalues are said to be in a Poincaré domain if their convex hull in the complex plane does not include the origin. Otherwise they are said to be in a Siegel domain. Only a finite number of resonant relations is possible in a Poincaré domain. In a Siegel domain, if the eigenvalues resonate, all integer multiples of resonance conditions will resonate as well leading to an infinite number of resonance relations [15].

Convergence problems could arise either in the series of resonant terms or the polynomial transformation. Poincaré's series are convergent if the eigenvalues lie in the Poincaré domain, even in the case of resonance. In contrast to that, if the eigenvalues lie in the Siegel domain, the series leading to the NFs are often divergent in the case of resonance [18]. This polynomial NF expansion also fails to converge if the system does not admit an additional symmetry [12].

The set of all vector valued monomials of degree k forms a linear vector space, which we denote by H_k, and the operation

$$h_{im} \rightarrow Dh_i(u)Au - Ah_i(u) = -L_A(h_{im}) \tag{2.7}$$

is a linear map of H_k onto H_k. $F_i(u)$ can be viewed as an element of H_k, and H_k can be (non uniquely) represented as

$$H_k = L_A(H_k) \oplus G_k \tag{2.8}$$

where G_k contains those basis vectors that are complementary to $L_A(H_k)$. We will term this the null space and it will contain the vectors that can not be eliminated by a polynomial NF transformation.

We will put A into a special form that will ease our calculations considerably. The three dimensional chaotic systems that we will be dealing with in this work are going to have

$$\lambda_{1,2} = \alpha \pm i\beta, \quad and \quad \lambda_3 = \gamma, \tag{2.9}$$

as their eigenvalues. Let the corresponding eigenvectors be $\mathbf{v_1} + \mathbf{iv_2}$, $\mathbf{v_1} - \mathbf{iv_2}$, and $\mathbf{v_3}$ then $\{\mathbf{v_1}, \mathbf{v_2}, \mathbf{v_3}\}$ is a basis for R^3, the matrix

$$M = [\mathbf{v_2}, \mathbf{v_1}, \mathbf{v_3}]$$

is invertible and

$$M^{-1}AM = \begin{bmatrix} \alpha & -\beta & 0 \\ \beta & \alpha & 0 \\ 0 & 0 & \gamma \end{bmatrix} \tag{2.10}$$

a real 3 x 3 matrix with a 2 x 2 block along the diagonal. This matrix can be transformed to a complex diagonal matrix by another transformation

$$D^{-1}M^{-1}AMD = \tilde{A} = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix}, \tag{2.11}$$

where

$$D = \frac{1}{2}\begin{bmatrix} 1 & 1 & 0 \\ -i & i & 0 \\ 0 & 0 & 2 \end{bmatrix}, \quad D^{-1} = \begin{bmatrix} 1 & i & 0 \\ 1 & -i & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The resulting complex, diagonal system with basis $\zeta = (z, z^*, \tilde{x}_3)^T$ becomes

$$\dot{\zeta} = v(\zeta) = \tilde{A}\zeta + \tilde{F}(\zeta), \tag{2.12}$$

where the relation between the new and old coordinates is

$$\tilde{x} = M^{-1}\bar{x}\,, \tag{2.13}$$

$$\zeta = D^{-1}\tilde{x}. \tag{2.14}$$

Also note that the nonlinear part transforms as $\tilde{F} = D^{-1}M^{-1}F$, which results in a system that consists of a complex conjugate pair of equations and a real equation.

2.1. The Normal Form Procedure

In the n'th order approximation to the solution, the near-identity transformation is written as

$$z = u + \sum_{k=1}^{n} h_{1k}(u, u^*, w) \tag{2.15}$$

$$z^* = u^* + \sum_{k=1}^{n} h_{1k}^*(u^*, u, w) \tag{2.16}$$

$$\tilde{x}_3 = w + \sum_{k=1}^{n} h_{2k}(u, u^*, w) \,, \tag{2.17}$$

where the complex h functions are the generators of the transformation. This near identity transformation is not unique, a number of terms with vanishing Lie Brackets can be freely added to the transformation without changing the NF equation. The so-called usual choice is to omit all of these free terms. This convention has been adopted in this work. The homological equations are solved order by order. In each order all terms other than the resonant ones are eliminated. Calculations pertaining to one order of the NF expansion do not affect the homological equation that is solved in higher orders. A higher order NF transformation leaves the lower orders unchanged. Thus, a transformation is not affected by subsequent near identity substitutions. In this way, the original set of equations are converted into the corresponding set of NF equations:

$$\dot{u} = \lambda_1 u + \sum_{k=1}^{n} U_k(z, z^*, w) \tag{2.18}$$

$$\dot{u}^* = \lambda_2 u^* + \sum_{k=1}^{n} U_k^*(z, z^*, w) \tag{2.19}$$

$$\dot{w} = \lambda_3 w + \sum_{k=1}^{n} W_k(z, z^*, w) \,, \tag{2.20}$$

where $\lambda_1^* = \lambda_2$ and the summation part constitutes the resonance terms.

2.1.1. Implementation of Initial Conditions:

To make a numerical comparison between the NF approximation to the solution and the original system one resorts to numerical simulation. The complex conjugate structure was very convenient for the computation of the NF expansion on a digital computer by a symbolic programming language [19]. On the other hand, one has to use the cylindrical coordinates in order to perform numerical calculations. Taking $u = \rho e^{i\theta}$

the NF equations in cylindrical coordinates can be written as:

$$\dot{\rho} = \alpha\rho + \Omega(\rho, \theta, w) \tag{2.21}$$

$$\dot{\theta} = \beta + \Theta(\rho, \theta, w) \tag{2.22}$$

$$\dot{w} = \gamma w + W(\rho, \theta, w) \tag{2.23}$$

These three equations are to be integrated numerically with initial conditions (ρ_o, θ_o, w_o) near the origin. These initial conditions are back transformed by

$$\bar{x} = MD\zeta$$

followed by $x = \bar{x} + x_o$ to get the initial conditions of the original system. Note that the initial conditions will depend on the order of the NF expansion, because ζ are computed from $z = u + \sum T$, $\tilde{x}_3 = w + \sum S$.

If $\alpha = 0, \beta, \gamma \neq 0$, the NF can be manipulated so that $\Omega(\rho, \theta, w)$ and $\Theta(\rho, \theta, w)$ depend only on ρ while $W(\rho, \theta, w)$ becomes w times a function of ρ and θ. This, in principle, allows an analytical solution of the truncated equations by integrating the ρ equation directly, using this to integrate the θ equation and using these two results to integrate the w equation. The question of the extent to which this analytical solution can represent the original system will be analyzed in the next section.

3. SYSTEMS POSSESSING THE HOPF BIFURCATION PROPERTY

Sprott [11] examined three dimensional ordinary differential equations with at most quadratic nonlinearities and found 19 distinct simple examples of chaotic flows with either five terms and two nonlinearities or six terms and one nonlinearity. They are given in Table 3.1. Except for one anomaly all have a Liapunov spectrum (+, 0, −). Half of them are resonant. When they are linearized about their equilibrium points, zero eigenvalues and imaginary conjugate pairs of eigenvalues occur frequently. The NF calculations involving these cases have been extensively studied [20, 21]. Numerical simulation of these systems reveals that they possess zero Liapunov exponents. Our study of some of these systems will involve a comparison of their numerically simulated attractors to the NF representation. We will also see that their resonant properties can be shifted by a nonpolynomial rational transformation in the next chapter.

3.1. The Sprott C System

The Sprott C system is

$$v = \begin{bmatrix} x_2 x_3 \\ x_1 - x_2 \\ 1 - x_1^2 \end{bmatrix} \tag{3.1}$$

with two critical points at $C_\pm = (\pm 1, \pm 1, 0)$. Moving the fixed points to the origin by $\bar{x}_1 = x_1 \mp 1, \bar{x}_2 = x_2 \mp 1, \bar{x}_3 = x_3$ results in a vector field:

$$v = \begin{bmatrix} \pm\bar{x}_3 + \bar{x}_2\bar{x}_3 \\ \bar{x}_1 - \bar{x}_2 \\ \mp 2\bar{x}_1 - \bar{x}_1^2 \end{bmatrix}. \tag{3.2}$$

Table 3.1. Sprott Systems

	System	Fixed Points	Liapunov Exponents	Eigenvalues of A
A	$\dot{x_1} = x_2$ $\dot{x_2} = -x_1 + x_2x_3,$ $\dot{x_3} = 1 - x_2^2$	none	0 ± 0.014	$(0, \pm i)$
B	$\dot{x_1} = x_2x_3,$ $\dot{x_2} = x_1 - x_2,$ $\dot{x_3} = 1 - x_1x_2$	$(\pm 1, \pm 1, 0)$	$0.210, 0, -1.210$	$(-1.35, 0.176 \pm 1.202i)$
C	$\dot{x_1} = x_2x_3,$ $\dot{x_2} = x_1 - x_2,$ $\dot{x_3} = 1 - x_1^2$	$(\pm 1, \pm 1, 0)$	$0.163, 0, -1.163$	$(-1, \pm\sqrt{2}i)$
D	$\dot{x_1} = -x_2,$ $\dot{x_2} = x_1 + x_3,$ $\dot{x_3} = x_1x_3 + 3x_2^2$	$(0, 0, 0)$	$0.103, 0, -1.320$	$(0, \pm i)$
E	$\dot{x_1} = x_2x_3,$ $\dot{x_2} = x_1^2 - x_2,$ $\dot{x_3} = 1 - 4x_1$	$0.25, 0.063, 0$	$0.078, 0, -1.078$	$(-1, \pm\frac{i}{2})$
F	$\dot{x_1} = x_2 + x_3,$ $\dot{x_2} = -x_1 + 0.5x_2,$ $\dot{x_3} = x_1^2 - x_3$	$(0, 0, 0),$ $(-2, -4, 4)$	$0.117, 0, -0.617$	$(-1, \frac{1}{4} \pm \frac{i\sqrt{15}}{4}),$ $(0.21, -0.35 \pm 2.12i)$
G	$\dot{x_1} = 0.4x_1 + x_3,$ $\dot{x_2} = x_1x_3 - x_2,$ $\dot{x_3} = -x_1 + x_2$	$(0, 0, 0),$ $(-2.5, -2.5, 1)$	$0.034, 0, -0.634$	$(-1, 0.20 \pm 0.97i),$ $(0.29, -0.44 \pm 1.78i)$
H	$\dot{x_1} = -x_2 + x_3^2,$ $\dot{x_2} = x_1 + 0.5x_2,$ $\dot{x_3} = x_1 - x_3$	$(0, 0, 0),$ $(-2, 4, -2)$	$0.117, 0, -0.617$	$(-1, 0.25 \pm 0.97i),$ $(-2.50, 0.96 \pm 0.54i)$
I	$\dot{x_1} = -0.2x_2,$ $\dot{x_2} = x_1 + x_3,$ $\dot{x_3} = x_1 - x_2^2 - x_3$	$(0, 0, 0)$	$0.012, 0, -1.012$	$(0.57, -0.78 \pm 0.28i)$
J	$\dot{x_1} = 2x_3,$ $\dot{x_2} = -2x_2 + x_3,$ $\dot{x_3} = x_1 + x_2 + x_2^2$	$(0, 0, 0)$	$0.076, 0, -2.076$	$(-2.31, 0.16 \pm 1.31i)$
K	$\dot{x_1} = x_1x_2 - x_3,$ $\dot{x_2} = x_1 - x_2,$ $\dot{x_3} = x_1 + 0.3x_3$	$(0, 0, 0),$ $(-10/3, -10/3, 100/9)$	$0.038, 0, -0.089$	$(-1, 0.15 \pm 0.99i),$ $(0.144, -2.088 \pm 1.614i)$
L	$\dot{x_1} = x_2 + 3.9x_3,$ $\dot{x_2} = 0.9x_1^2 - x_2,$ $\dot{x_3} = 1 - x_1$	$(1, 9/10, -3/13)$	$0.061, 0, -1.061$	$(-1.43, 0.22 \pm 1.64i)$
M	$\dot{x_1} = -x_3,$ $\dot{x_2} = -x_1^2 - x_2,$ $\dot{x_3} = 1.7 + 1.7x_1 + x_2$	$(2.406, -5.791, 0),$ $(-0.706, -0.499, 0)$	$0.044, 0, -1.044$	$(-1.38, 0.19 \pm 1.48i),$ $(0.91, -0.95 \pm 1.59i)$
N	$\dot{x_1} = -2x_2,$ $\dot{x_2} = x_1 + x_3^2,$ $\dot{x_3} = 1 + x_2 - 2x_3$	$(-1/4, 0, 1/2)$	$0.076, 0, -2.076$	$(-2.31, 0.16 \pm 1.31i)$
O	$\dot{x_1} = x_2,$ $\dot{x_2} = x_1 - x_3,$ $\dot{x_3} = x_1 + x_1x_3 + 2.7x_2$	$(0, 0, 0),$ $(-1, 0, -1)$	$0.049, 0, -0.319$	$(-0.51, 0.26 \pm 1.38i),$ $(0.43, -0.71 \pm 1.34i)$
P	$\dot{x_1} = 2.7x_2 + x_3,$ $\dot{x_2} = -x_1 + x_2^2,$ $\dot{x_3} = x_1 + x_2$	$(0, 0, 0),$ $(1, -1, 2.7)$	$0.087, 0, -0.281$	$(-0.51, 0.25 \pm 1.37i),$ $(0.38, -1.19 \pm 1.69i)$
Q	$\dot{x_1} = -x_3,$ $\dot{x_2} = x_1 - x_2,$ $\dot{x_3} = 3.1x_1 + x_2^2 + 0.5x_3$	$(0, 0, 0),$ $(-3.1, -3.1, 0)$	$0.109, 0, -0.609$	$(-1, 0.25 \pm 1.74i),$ $(0.83, -0.66 \pm 1.81i)$
R	$\dot{x_1} = 0.9 - x_2,$ $\dot{x_2} = 0.4 + x_3,$ $\dot{x_3} = x_1x_2 - x_3$	$(-0.44, 0.9, -0.4)$	$0.062, 0, -1.062$	$(-1.23, 0.12 \pm 0.85i)$
S	$\dot{x_1} = -x_1 - 4x_2,$ $\dot{x_2} = x_1 + x_3^2,$ $\dot{x_3} = 1 + x_1$	$(-1, 0.25, 1),$ $(-1, 0.25, -1)$	$0.188, 0, -1.188$	$(-1.61, 0.30 \pm 2.21i),$ $(1.20, -1.10 \pm 2.33i)$

Since the fixed points of the system are symmetric about the origin, the NF expansion will be carried out about C_+ only. A and its eigenvectors are

$$A = \begin{bmatrix} 0 & 0 & 1 \\ 1 & -1 & 0 \\ -2 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} \pm i\sqrt{2}/2 \\ (-2 \pm i\sqrt{2})/6 \\ 1 \end{bmatrix}. \tag{3.3}$$

The eigenvalues of A are of the form $(\lambda_{1,2} = \alpha \pm i\beta, \lambda_3 = \gamma)$, where $\alpha = 0, \beta = \sqrt{2}, \gamma = -1$. Since $\alpha = 0$ the system has the Hopf bifurcation property. The eigenvalues of the system satisfy resonance relations in every odd order of the NF expansion starting from the third order. To put the system into a diagonal, complex form, first the transformation matrix M is constructed from the eigenvectors and its inverse M^{-1} is calculated to be:

$$M = \begin{bmatrix} -\sqrt{2}/2 & 0 & 0 \\ -\sqrt{2}/6 & -1/3 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad M^{-1} = \begin{bmatrix} -\sqrt{2} & 0 & 0 \\ 0 & 0 & 1 \\ -1/3 & 1 & 1/3 \end{bmatrix}, \tag{3.4}$$

for C_+. C_- gives a similar result. These matrices transform A into a matrix with 2x2 blocks along the diagonal:

$$M^{-1}AM = \begin{bmatrix} 0 & -\sqrt{2} & 0 \\ \sqrt{2} & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \tag{3.5}$$

which can be transformed into a diagonal, complex system by $z = \tilde{x}_1 + i\tilde{x}_2$:

$$\begin{aligned} \dot{z} &= i\beta z + \frac{\sqrt{2}}{2} i\tilde{x}_3 z - \frac{\sqrt{2}}{2} i\tilde{x}_3 z^* - \frac{\sqrt{2}}{12} z^2 + \frac{\sqrt{2}}{6} zz^* - \frac{\sqrt{2}}{12} z^{*2} - \frac{5}{24} iz^2 \\ &\quad -\frac{1}{4} izz^* - \frac{1}{24} iz^{*2} \\ \dot{\tilde{x}}_3 &= \gamma x_3 - \frac{\sqrt{2}}{72} iz^2 + \frac{\sqrt{2}}{72} iz^{*2} + \frac{1}{6} i\tilde{x}_3 z - \frac{1}{6} i\tilde{x}_3 z^* - \frac{5}{72} z^2 - \frac{1}{36} zz^* - \frac{5}{72} z^{*2} \end{aligned}$$

The U, U^*, and W functions are calculated and yield a Hopf NF:

$$\dot{u} = \beta u + \left(\frac{1}{36} - \frac{1}{9}i\sqrt{2}\right) u^2 u^* + \left(\frac{113}{36936} - \frac{2099}{590976}i\sqrt{2}\right) u^3 u^{*2}$$

$$- \left(\frac{35738557}{35572027392} + \frac{23284783}{14821678080}i\sqrt{2}\right) u^4 u^{*3} \tag{3.6}$$

$$\dot{w} = \gamma w - \frac{1}{9} uu^* w - \frac{887}{36936} u^2 u^{*2} w + \frac{146441791}{80037061632} u^3 u^{*3} w \tag{3.7}$$

through the seventh order. Writing $u = \rho e^{i\theta}$ gives

$$\dot{\rho} = \frac{1}{36}\rho^3 \left(1 + \frac{113}{1026}\rho^2 - \frac{35738557}{988111872}\rho^4\right) \tag{3.8}$$

$$\dot{\theta} = \beta - \frac{\sqrt{2}}{9}\rho^2 \left(1 + \frac{2099}{65664}\rho^2 + \frac{23284783}{1646853120}\rho^4\right) \tag{3.9}$$

$$\dot{w} = \gamma w - \frac{1}{9}\rho^2 w \left(1 + \frac{887}{4104}\rho^2 - \frac{146441791}{8893006848}\rho^4\right) \tag{3.10}$$

The origin is a stable equilibrium point for the $\dot{w}$ equation, so that as the NF equations evolve, w vanishes. This serves as an indication of a two dimensional central manifold and a negative Liapunov exponent. Since the coefficient of ρ^3 is positive the Hopf bifurcation is of subcritical type [22]. Note that the $\dot{\rho}$ equation has two fixed points. The one at the origin is unstable whereas $\rho_o = 2.645$ is a stable one. This fixed point can be ascribed to the size of the chaotic attractor. The equilibrium point ρ_o is only an equilibrium point for the effective one dimensional $\dot{\rho}$ equation, since at that point $\dot{\theta} \neq 0$, so that the point corresponds to a quasi-periodic orbit.

3.1.1. Dependence on a Scale Parameter, ϵ

The method of NFs is generally known as a perturbation method that has been used to overcome the problem of secular terms of the naive perturbation theory. To

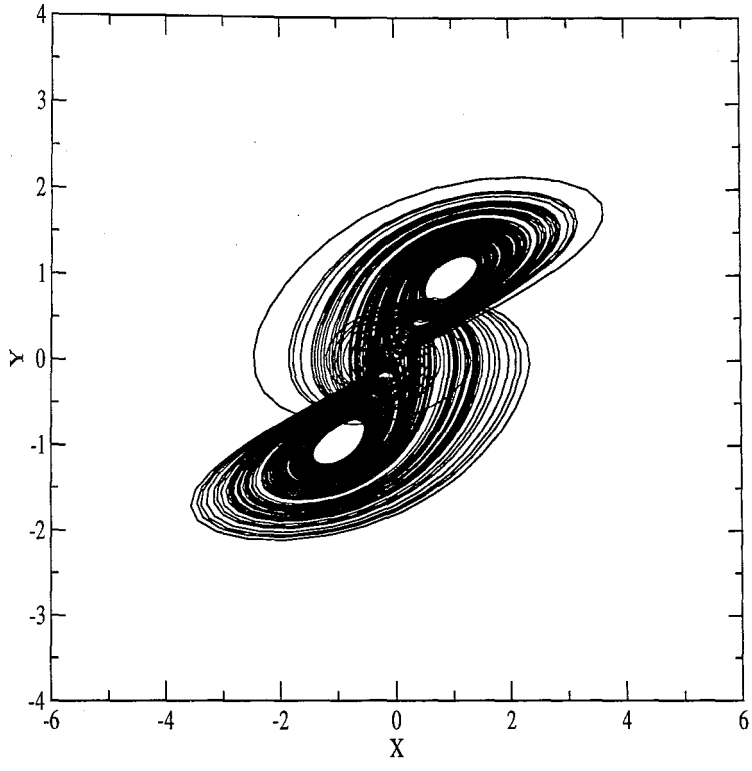

Figure 3.1. Phase space of the Sprott C system, with x_1 along X and x_2 along Y

perform a perturbation expansion, the state variables are rescaled as:

$$x_1 \to \epsilon x_1 \tag{3.11}$$

$$x_2 \to \epsilon x_2 \tag{3.12}$$

$$x_3 \to \epsilon x_3 \tag{3.13}$$

where ϵ is a small, positive number. With the transformed variables the system becomes

$$v = \begin{bmatrix} \pm\bar{x}_3 + \epsilon\bar{x}_2\bar{x}_3 \\ \bar{x}_1 - \bar{x}_2 \\ \mp 2\bar{x}_1 - \epsilon\bar{x}_1^2 \end{bmatrix}. \tag{3.14}$$

The Liapunov spectrum under this transformation is shown in Figure 3.2 as a function of ϵ. Clearly, in the limit $\epsilon \to 0$ the system is not chaotic and all exponents vanish. As ϵ starts increasing the system becomes chaotic after about $\epsilon = 0.2$ beyond which numerical simulations show that the Liapunov spectra is invariant under this transfor-

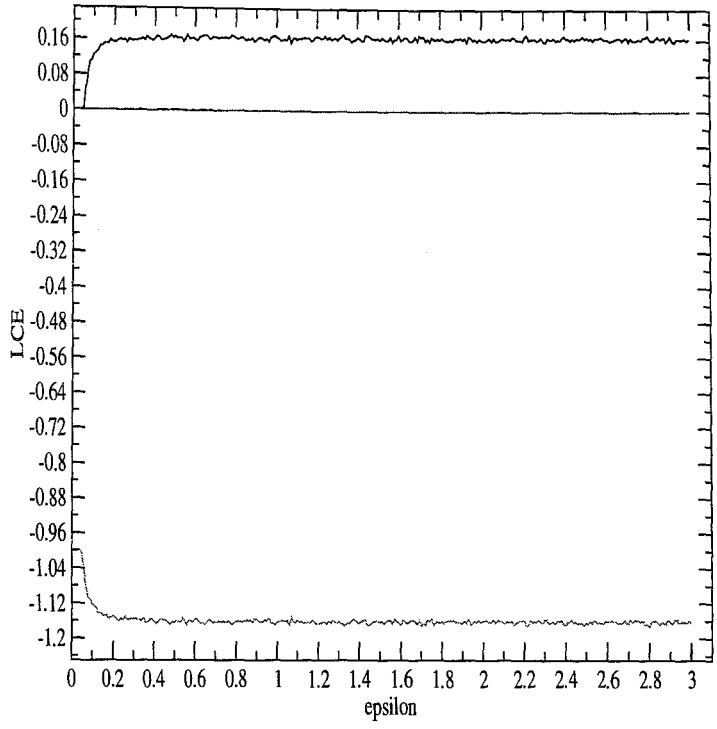

Figure 3.2. Liapunov characteristic exponent (LCE) spectrum of Sprott C under ϵ scaling

mation, although the transformation affects the size of the attractor. For example the attractor for $\epsilon = 0.5$ is shown in Figure 3.3. As ϵ increases the size of the attractor decreases. This change in the attractor size can be estimated by the NF analysis. The NF calculations lead to

$$\dot{\rho} = \epsilon \frac{1}{36}\rho^3 + \epsilon^2 \frac{113}{36936}\rho^5 - \epsilon^3 \frac{35738557}{35572027392}\rho^7 \tag{3.15}$$

$$\dot{\theta} = \beta - \epsilon \frac{\sqrt{2}}{9}\rho^2 + \epsilon^2 \frac{2099}{590976}\rho^4 + \epsilon^3 \frac{23284783}{14821678080}\rho^6 \tag{3.16}$$

$$\dot{w} = \gamma w - \epsilon \frac{1}{9}\rho^2 w - \epsilon^2 \frac{887}{36936}\rho^4 w + \epsilon^3 \frac{146441791}{80037061632}\rho^6 w\,. \tag{3.17}$$

Again the size of the attractor can be estimated from the two fixed points of the $\dot{\rho}$ equation; an unstable one at the origin, and a stable one at $\rho_o = 2.645/\epsilon$. Thus the size of the attractor is scaled as $1/\epsilon$. This value is 5.290 for $\epsilon = 0.5$. This result agrees with Figure 3.3.

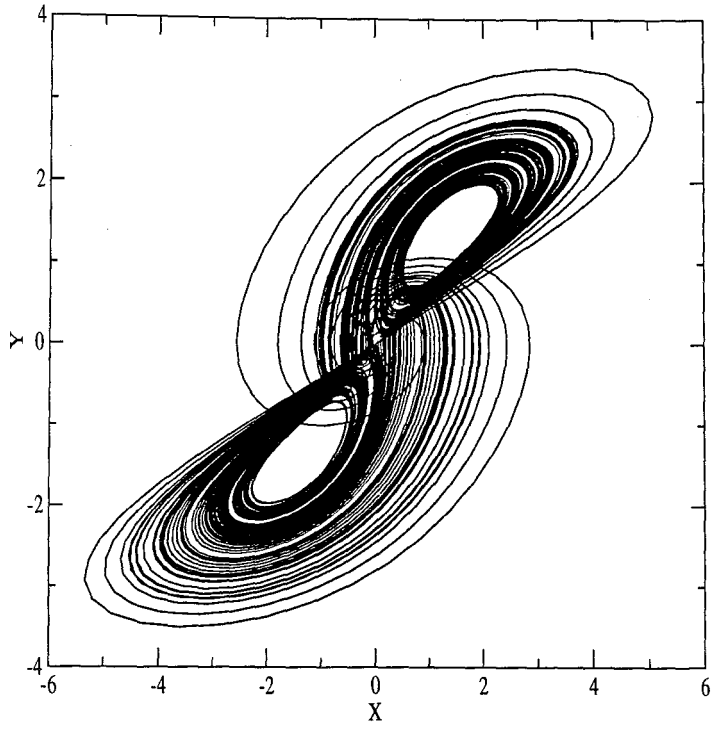

Figure 3.3. Phase space of the Sprott C system with $\epsilon = 0.5$

3.1.2. Numerical Results

In Figure 3.4 the numerical solution of the Hopf NF equations for the initial conditions ($x_o = -0.07255$, $y_o = 0.07883$, $z_o = -0.00388$) that correspond to ($\rho = 0.1$, $\theta = 0$, $w = 0.1$) are compared with seventh order approximations. Solutions overlap up to $t = 1800$ units [23]. Numerical integration of NF equations shows that this is precisely the time it takes for the NF system to reach $\rho_o = 2.645$. Also it confirms the so-called throw-and-catch mechanism from one leaf of the attractor to the other that was heuristically proposed by Umur [24]. It seems that it is the chaotic behavior alone that is responsible for the failure of the approximation. Since w approaches to zero and ρ settles on ρ_o one has to search for chaos on the (ρ , θ) space in spite of the fact that three dimensions are required for chaotic behavior. We remark that the exact solution is only valid for the NF equations. The nonlinear dependence of θ on ρ

$$\frac{d\rho}{d\theta} = f(\rho) = (\frac{1}{72}\sqrt{2}\rho^3 + \frac{227}{73872}\sqrt{2}\rho^5 - \frac{2646071}{23714684928}\sqrt{2}\rho^7 + O\left(\rho^9\right)) , \qquad (3.18)$$

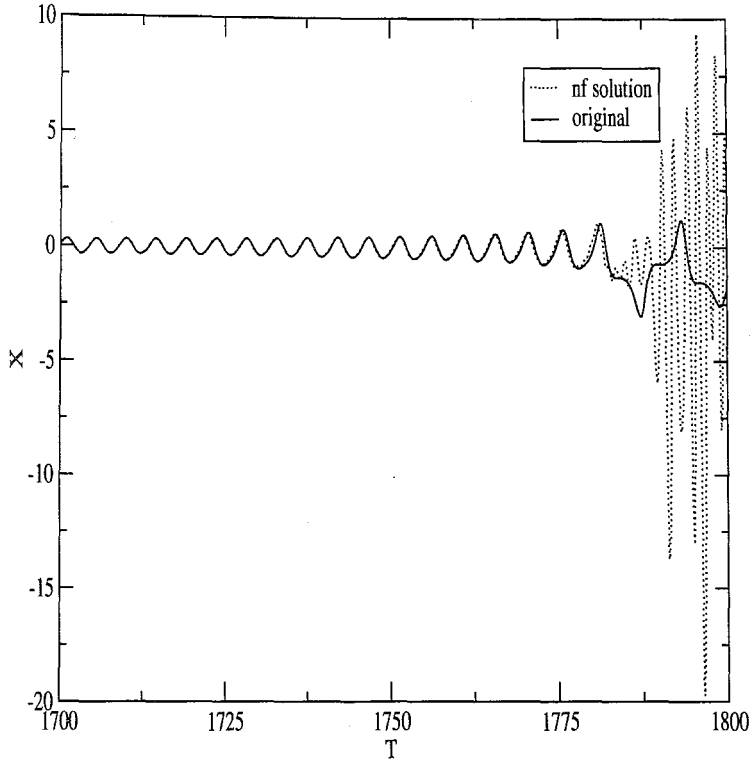

Figure 3.4. Solution and approximation to solution of Equation 3.1

allows us to assume that on the average solutions can diverge from each other exponentially up to the point where our approximation is valid. Let such a divergence be given by $\rho = Ae^{\lambda\theta}$, where A is some constant. Then $d\rho/d\theta = \lambda\rho$. The equation can not admit such a solution exactly, so we expect that this solution is valid as an average over the extent of the trajectory,

$$\langle \lambda\rho - f(\rho) \rangle \approx 0, \quad \text{with} \langle x \rangle = \int_o^{\rho_o} x d\rho \tag{3.19}$$

This can be considered as an average over the attractor,

$$\lambda \approx \frac{\langle f(\rho) \rangle}{\langle \rho \rangle} . \tag{3.20}$$

This results in $\lambda = 0.126$, that agrees with the value 0.163 as the largest Liapunov exponent found by numerical simulation. Only a heuristic justification can be given to this procedure, since there could be a number of alternative numerical approaches. The statistical error bound on this estimate can be tested by calculating the spread of

Equation 3.20 around zero:

$$\sigma = \left[\frac{\langle(\lambda\rho - f(\rho))^2\rangle}{\langle\rho^2\rangle}\right]^{\frac{1}{2}} \tag{3.21}$$

For our estimate $\lambda = 0.126$, this statistical error is 0.088. The result is independent of ϵ.

3.1.3. An Approximately Conserved Quantity

To the lowest order the NF equations for the Sprott C system are

$$\begin{aligned} \dot{\rho} &= \frac{1}{36}\rho^3 + \cdots \\ \dot{\theta} &= \beta + \cdots \end{aligned} \tag{3.22}$$

which result in

$$\frac{d\rho}{d\theta} = \frac{\sqrt{2}}{72}\rho^3 + O(\rho^5)\,. \tag{3.23}$$

This expression can be integrated to give

$$\frac{1}{2\rho^2} + \frac{\sqrt{2}}{72}\theta = Constant\,. \tag{3.24}$$

One has to back transform this quantity to see how it behaves over the phase space up to the point where the NF expansion is valid. This has been done numerically and for the same initial conditions as in Figure 3.4 the result is shown in Figure 3.5. An approximately conserved quantity indicates that in the phase space the corresponding combination of variables neither shrink nor expand exponentially, indicating the existence of a zero Liapunov exponent or a Liapunov exponent indistinguisible from zero by numerical simulation [25]. In principle an approximately conserved quantity that includes ρ^5 terms can also be constructed using the w equation. Numerical simulations reveal that this quantity is not any better. The reason is that w rapidly goes to zero,

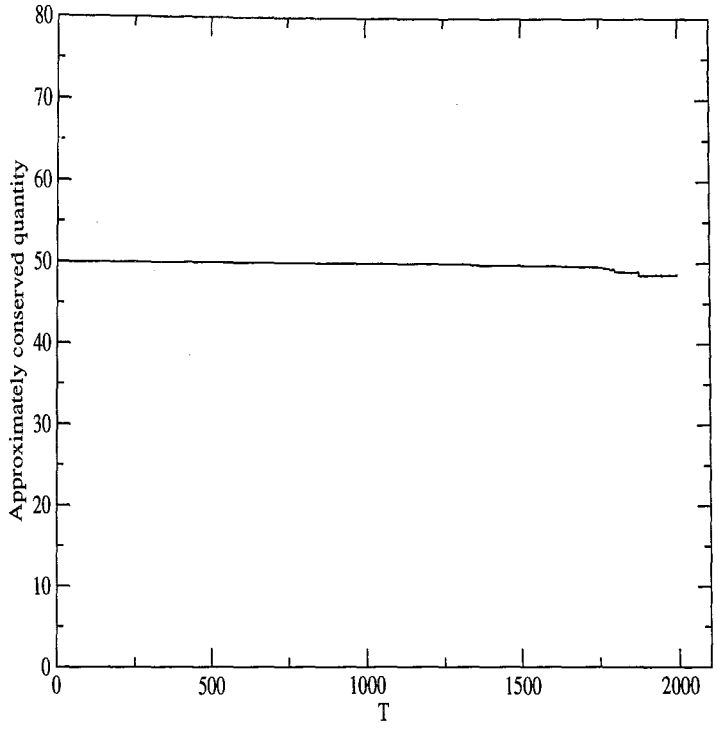

Figure 3.5. Behavior of the approximately conserved quantity in Sprott C

the system settles on the (ρ, θ) space. This fact also agrees with the fact that the fractal dimension of the attractor is slightly above two [11].

3.2. The Sprott E System

The Sprott E system is given as

$$v = \begin{bmatrix} x_2 x_3 \\ x_1^2 - x_2 \\ 1 - 4x_1 \end{bmatrix} . \tag{3.25}$$

There is only one equilibrium point located at $C = (1/4, 1/16, 0)$. When this point is moved to the origin the system becomes:

$$v = \begin{bmatrix} \bar{x}_3/16 + \bar{x}_2\bar{x}_3 \\ \bar{x}_1/2 - \bar{x}_2 + \bar{x}_1^2 \\ -4\bar{x}_1 \end{bmatrix} \tag{3.26}$$

A and its eigenvectors are

$$A = \begin{bmatrix} 0 & 0 & 1/16 \\ 1/2 & -1 & 0 \\ -4 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -i/8 \\ -(1+2i)/40 \\ 1 \end{bmatrix}. \tag{3.27}$$

The eigenvalues of A are again of the form ($\lambda_{1,2} = \alpha \pm i\beta, \lambda_3 = \gamma$), where $\alpha = 0, \beta = 1/2, \gamma = -1$. As in the Sprott C system, they satisfy resonance relations in every odd order of the NF expansion starting from the third order. Note that the system has the Hopf bifurcation property. To put the system into a diagonal, complex form, first a transformation matrix M is constructed from the eigenvectors and its inverse M^{-1} is calculated:

$$M = \begin{bmatrix} -1/8 & 0 & 0 \\ -1/20 & -1/40 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad M^{-1} = \begin{bmatrix} -8 & 0 & 0 \\ 0 & 0 & 1 \\ -2/5 & 1 & 1/40 \end{bmatrix}. \tag{3.28}$$

These matrices transform A into a matrix with 2x2 blocks along the diagonal:

$$M^{-1}AM = \begin{bmatrix} 0 & -1/2 & 0 \\ 1/2 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \tag{3.29}$$

which can be transformed into a diagonal, complex system by $z = \tilde{x}_1 + i\tilde{x}_2$:

$$\dot{z} = \beta i z + 4 i \tilde{x}_3 z - 4 i \tilde{x}_3 z^* - \frac{1}{10} i z^2 + \frac{1}{10} i z^{*2} - \frac{1}{20} z^2 + \frac{1}{10} z z^* - \frac{1}{20} z^{*2}$$

$$\dot{\tilde{x}}_3 = \gamma \tilde{x}_3 + \frac{1}{5} i \tilde{x}_3 z - \frac{1}{5} i \tilde{x}_3 z^* - \frac{1}{200} i z^2 + \frac{1}{200} i z^{*2} + \frac{9}{6400} z^2 + \frac{41}{3200} z z^*$$

$$+ \frac{9}{6400} z^{*2}$$

The U, U^*, and W functions are calculated and yield a NF:

$$\begin{aligned}\dot{u} &= \beta iu + \frac{23}{3200}u^2u^* + \frac{113}{9600}iu^2u^* \\ &\quad + \frac{4273991}{1497600000}u^3u^{*2} - \frac{14470361}{17971200000}iu^3u^{*2} \qquad (3.30)\end{aligned}$$

$$\dot{w} = \gamma w - \frac{23}{800}uu^*w - \frac{5584999}{374400000}u^2u^{*2}w \qquad (3.31)$$

through the seventh order. In cylindrical coordinates they read

$$\dot{\rho} = \frac{1}{3200}\rho^3\left(23 + \frac{4273991}{468000}\rho^2 - \frac{5732625917801}{7008768000000}\rho^4\right) \qquad (3.32)$$

$$\dot{\theta} = \beta + \frac{1}{9600}\rho^2\left(113 - \frac{14470361}{1872000}\rho^2 - \frac{30250848101177}{2336256000000}\rho^4\right) \qquad (3.33)$$

$$\dot{w} = \gamma w - \frac{1}{800}\rho^2 w\left(23 + \frac{5584999}{468000}\rho^2 - \frac{2063305495801}{7008768000000}\rho^4\right). \qquad (3.34)$$

The structure of the NF equations are very similar to case C, undergoing again a subcritical Hopf bifurcation (the coefficient of ρ^3 is positive), thus amenable to the same analysis. The radius of the chaotic attractor can be estimated from $\dot{\rho} = 0$ which yields $\rho_o = 3.6445$. This agrees with the phase space plot (Figure 3.6) of the system.

$$\frac{d\rho}{d\theta} = f(\rho) = \left(\frac{23}{1600}\rho^3 - \frac{8041177}{1497600000}\rho^5 + \frac{6890526323021}{11214028800000000}\rho^7 + O\left(\rho^9\right)\right) \qquad (3.35)$$

The averaging method described above,

$$\langle \lambda\rho - f(\rho) \rangle \approx 0 \qquad (3.36)$$

yields the estimate $\lambda = 0.0512$ for the positive Liapunov exponent. This result also agrees with the value 0.078 as the largest Liapunov exponent found by numerical methods. The statistical error on this result is 0.000085.

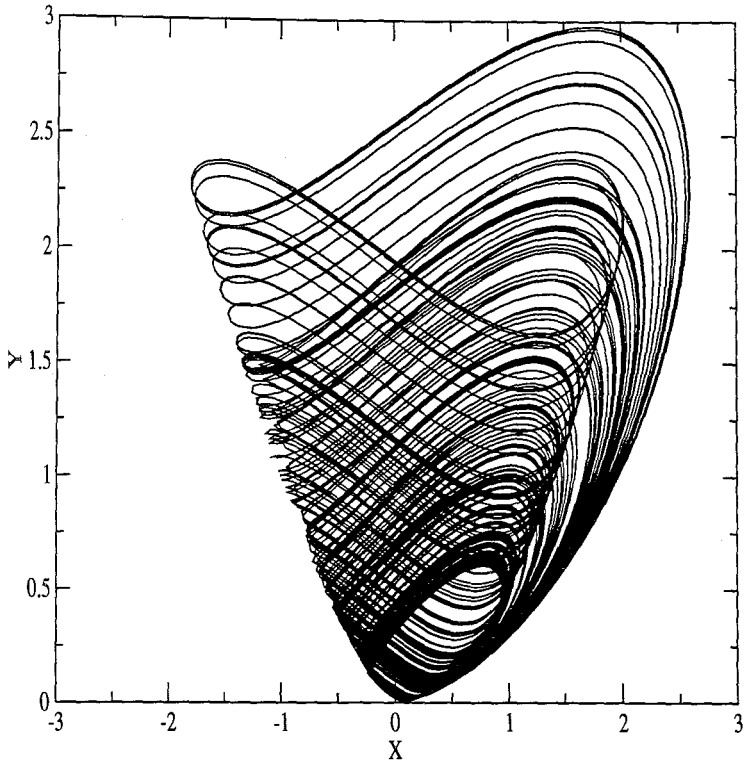

Figure 3.6. Phase space of the Sprott E system, with x_1 along X and x_2 along Y

3.2.1. Numerical Results

The estimate of the positive Liapunov exponent is not a very strong test, since estimating the largest Liapunov exponent is relatively easy [26] and other approaches may exist. For this reason, results of numerical integration of the original equations with a fixed point at the origin for the initial condition ($x_{10} = 0.0326$, $x_{20} = 0.2232$, $x_{30} = 0.0966$) obtained by computing the initial conditions of the original system by Equation 2.15 is compared to the seventh order NF approximation with the corresponding initial condition ($\rho = 0.25$, $\theta = 0$, $w = 0.25$). Results are shown in Figure 3.7. Solutions overlap up to $t = 1008$ time units. This is comparable to the time it takes the system to settle on the value ρ_o ($t = 1025$) and the attractor then jumps away from the origin at ($t = 1028$) time units [23].

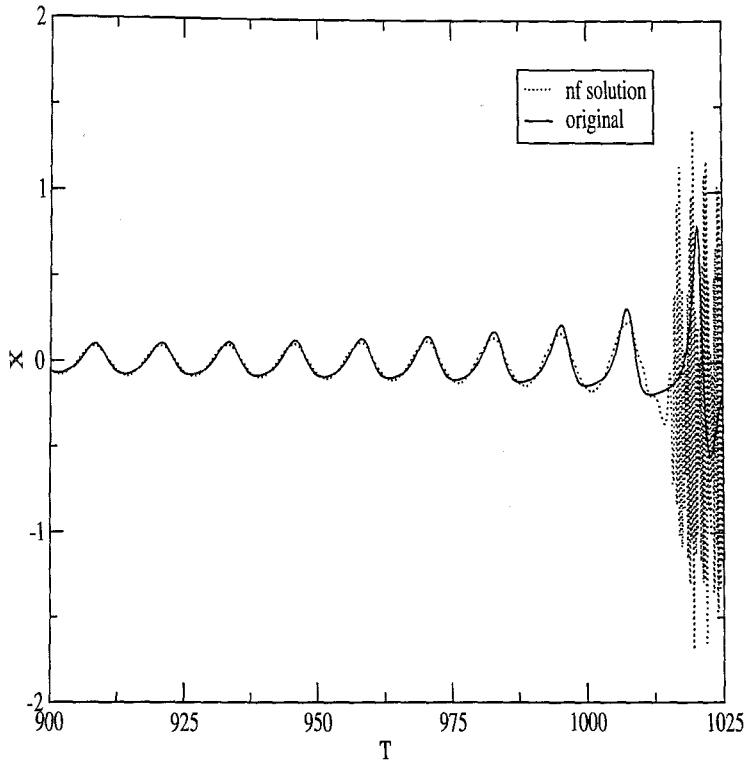

Figure 3.7. Solution and approximation to solution of Equation 3.25

3.2.2. An Approximately Conserved Quantity

To the lowest order the NF equations for the Sprott E system are

$$\begin{aligned} \dot{\rho} &= \frac{23}{3200}\rho^3 + \cdots \\ \dot{\theta} &= \beta + \cdots \end{aligned} \tag{3.37}$$

which result in

$$\frac{d\rho}{d\theta} = \frac{46}{3200}\rho^3 + O(\rho^5)\,. \tag{3.38}$$

This expression can be integrated to give

$$\frac{1}{2\rho^2} + \frac{46}{3200}\theta = Constant\,. \tag{3.39}$$

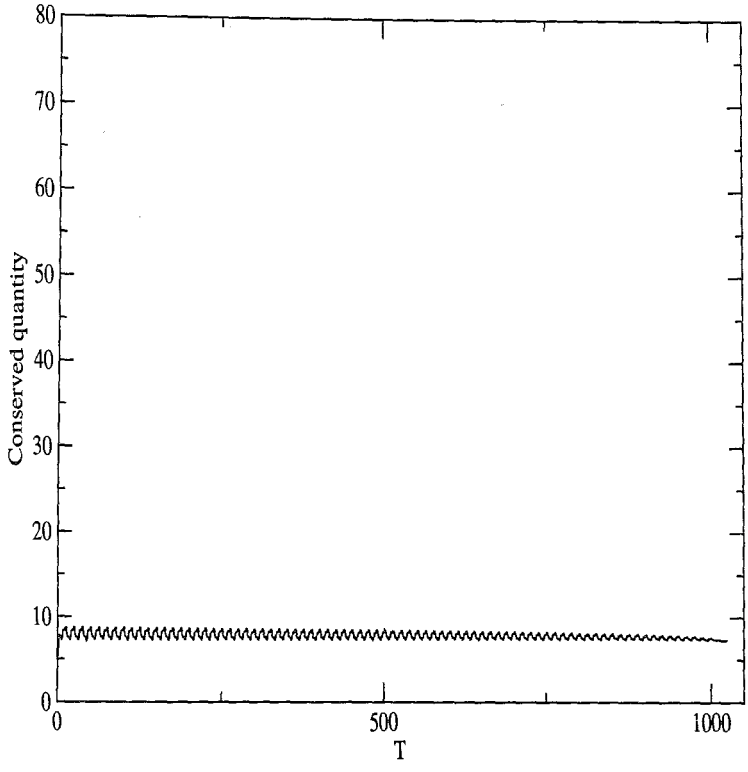

Figure 3.8. Behavior of the approximately conserved quantity in Sprott E

Again this quantity is back transformed numerically to see how it behaves over the phase space up to the point where the NF expansion is valid. This has been done for the same initial conditions (Figure 3.7). The result shown in the Figure 3.8 is similar to that of the Sprott C system, which again indicates the existence of a zero Liapunov exponent.

3.3. The Modified Sprott F System

The last example that we will consider is the modified Sprott F system, given by

$$v = \begin{bmatrix} x_2 + x_3 \\ -x_1 + ax_2 \\ x_1^2 - x_3 \end{bmatrix} . \tag{3.40}$$

The system that Sprott gives corresponds to $a = 1/2$. Here we consider a as a bifurcation parameter. For $a = 0$ the system has $(\lambda_{1,2} = 0, \lambda_3 = -1)$ as its Liapunov

exponents. Thus the system is not chaotic. Eigenvalues of the linearized system are $(\lambda_{1,2} = \pm i, \lambda_3 = -1)$ with $\alpha = 0, \beta = 1, \gamma = -1$. The following NF expansion through the seventh order,

$$\dot{\rho} = -\frac{1}{20}\rho^3 - \frac{251}{9000}\rho^5 - \frac{1850797}{88128000}\rho^7 \tag{3.41}$$

$$\dot{\theta} = \beta + \frac{1}{60}\rho^2 + \frac{3499}{108000}\rho^4 + \frac{12692663}{440640000}\rho^6 \tag{3.42}$$

$$\dot{w} = \gamma w + \frac{1}{5}\rho^2 w + \frac{497}{4500}\rho^4 w + \frac{44446901}{550800000}\rho^6 w \tag{3.43}$$

shows that contrary to the first two cases, both the ρ and w equations have a stable equilibrium point at the origin and the system undergoes a supercritical Hopf bifurcation. This indicates that two of the Liapunov exponents are zero or negative. Since our approach involves an average over the attractor it is not applicable to this supercritical case. As mentioned above, a two dimensional space is required to perform the averaging.

4. OTHER RESONANT SYSTEMS

The NF structure of chaotic systems that does not have the Hopf bifurcation property exhibit coupled resonant terms. Two of these systems are studied.

4.1. The Original Sprott F System

Consider the Sprott F system again:

$$v = \begin{bmatrix} x_2 + x_3 \\ -x_1 + x_2/2 \\ x_1^2 - x_3 \end{bmatrix} \tag{4.1}$$

The system has fixed two points at $C_1 = (0,0,0), C_2 = (-2,-4,4)$. A and its eigenvectors are

$$A = \begin{bmatrix} 0 & 1 & 1 \\ -1 & 1/2 & 0 \\ 0 & 0 & -1 \end{bmatrix}, \begin{bmatrix} -3/5 \\ -2/5 \\ 1 \end{bmatrix}, \begin{bmatrix} 1/4 \pm i\sqrt{15}/4 \\ 1 \\ 0 \end{bmatrix}. \tag{4.2}$$

The eigenvalues of A are $(\lambda_{1,2} = 1/4 \pm i\sqrt{15}/4, \lambda_3 = -1)$. Thus $\alpha = 1/4$, $\beta = \sqrt{15}/4$, $\gamma = -1$. Resonance relations are present starting from the sixth order. The next resonance happens at eleventh order. This is a chaotic system with Liapunov exponents $(\lambda_1 = 0.117,\ \lambda_2 = 0,\ \lambda_3 = -0.617)$ [11]. To put the system into diagonal, complex form, first a transformation matrix M is constructed from the eigenvectors as

$$M = \begin{bmatrix} -\sqrt{15}/4 & 1/4 & -3/5 \\ 0 & 1 & -2/5 \\ 0 & 0 & 1 \end{bmatrix}, \tag{4.3}$$

then M^{-1} is calculated to be

$$M^{-1} = \begin{bmatrix} -4\sqrt{15}/15 & 1/15 & -2/15 \\ 0 & 1 & 2/5 \\ 0 & 0 & 1 \end{bmatrix}, \tag{4.4}$$

to transform A into

$$M^{-1}AM = \begin{bmatrix} 1/4 & -\sqrt{15}/4 & 0 \\ \sqrt{15}/4 & 1/4 & 0 \\ 0 & 0 & -1 \end{bmatrix}. \tag{4.5}$$

Now the system can be easily transformed into the diagonal, complex notation by $z = \tilde{x}_1 + i\tilde{x}_2$.

$$\begin{aligned} \dot{z} &= \lambda_1 z + \frac{\sqrt{15}}{25} i\tilde{x}_3 z + \frac{2\sqrt{15}}{25} i\tilde{x}_3 z^* - \frac{6\sqrt{15}}{125}\tilde{x}_3^2 - \frac{\sqrt{15}}{24} z^2 - \frac{\sqrt{15}}{15} zz^* \\ &\quad -\frac{\sqrt{15}}{60} z^{*2} + \frac{18}{125} i\tilde{x}_3^2 + \frac{1}{40} iz^2 + \frac{1}{5} izz^* + \frac{3}{20} iz^{*2} - \frac{9}{25}\tilde{x}_3 z - \frac{6}{25}\tilde{x}_3 z^* \\ \dot{\tilde{x}}_3 &= \lambda_3 \tilde{x}_3 + \frac{\sqrt{15}}{32} iz^2 - \frac{1}{32}\sqrt{15} iz^{*2} + \frac{3}{20}\sqrt{15}\tilde{x}_3 z + \frac{3}{20}\sqrt{15}\tilde{x}_3 z^* \\ &\quad +\frac{3}{20} i\tilde{x}_3 z - \frac{3}{20} i\tilde{x}_3 z^* + \frac{9}{25}\tilde{x}_3^2 + \frac{7}{32} z^2 + 1/2\, zz^* + \frac{7}{32} z^{*2} \end{aligned}$$

The U, U^*, and W functions are calculated and yield a NF:

$$\dot{u} = i\beta u + 1/4\, u + \frac{3}{61} i c_1 \sqrt{15} u^3 u^{*2} w - c_1 u^3 u^{*2} w \tag{4.6}$$

$$\dot{u}^* = -i\beta u^* + 1/4\, u^* - \frac{3}{61} i c_1 \sqrt{15} u^2 u^{*3} w - c_1 u^{*3} u^2 w \tag{4.7}$$

$$\dot{w} = \gamma w + 3 c_1 u^2 u^{*2} w^2 , \tag{4.8}$$

where $c_1 = 0.5158987484$. This suggests taking the combination $u\bar{u}$

$$\frac{d(u\bar{u})}{dt} = \frac{1}{2} u\bar{u} - 2c_1 (u\bar{u})^3 w \tag{4.9}$$

$$\frac{dw}{dt} = -w + 3c_1 (u\bar{u})^2 w^2 \tag{4.10}$$

Writing $u = \rho e^{i\theta}$ gives

$$\dot{\rho} = \alpha\rho - c_1\rho^5 w \tag{4.11}$$

$$\dot{\theta} = \beta + \frac{3}{61}\sqrt{15}c_1\rho^4 w \tag{4.12}$$

$$\dot{w} = \gamma w + 3\,c_1\rho^4 w^2\,. \tag{4.13}$$

Note that the origin is a fixed point for both ρ and w. Furthermore, the expansion is in powers of $\rho^4 w$. This can still be analytically solved by integrating

$$4\frac{\dot{\rho}}{\rho} + \dot{w}w = \frac{d}{dt}ln\rho^4 w = -c_1\rho^4 w. \tag{4.14}$$

For ρ it is a source, therefore ρ escapes to infinity as the system evolves. Higher order resonant terms would result in an upper bound for ρ. Unfortunately, the NF expansion up to the eleventh order was beyond the power of the underlying portable standard lisp system in our symbolic routine. In principle it can be calculated. But even if such an upper bound is found the averaging approach for the estimation of the largest Liapunov exponent fails due to coupled resonance terms and the nonlinear combination $\rho^4 w$. On the other hand the origin becomes a sink for w. This can be considered as an indication of a negative Liapunov exponent. Numerical results are given in the Figure 4.1. We see that the validity of the NF solution is not as good as in the previous cases. The approximation diverges at about t = 70 units.

4.2. The Case with a Zero Eigenvalue: The Sprott D System

The system dynamics is given by

$$v = \begin{bmatrix} -x_2 \\ x_1 + x_3 \\ x_1 x_3 + 3{x_2}^2 \end{bmatrix}. \tag{4.15}$$

The system has its only fixed point at the origin. The eigenvalues of the linearized system are $(\lambda_{1,2} \pm i, \lambda_3 = 0)$. It is a resonant system in every order. Following the same

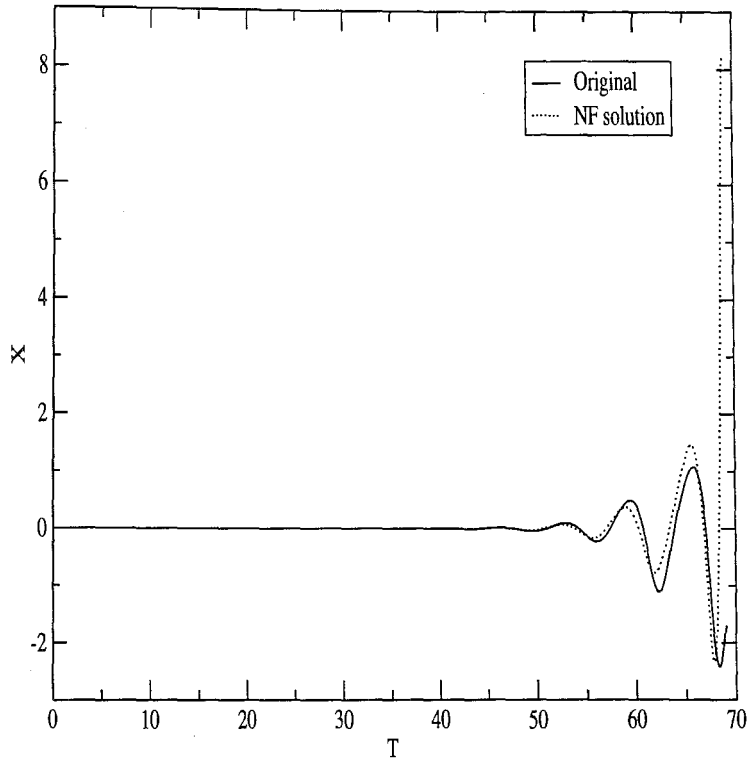

Figure 4.1. Solution and approximation to solution of Equation 4.1

steps as in the other cases the NF form equations are found to be

$$\dot{\rho} = \frac{1}{2}\rho w - \frac{113}{16}\rho w^3 + \frac{29411}{256}\rho w^5 + \frac{358651}{4096}\rho^5 w - \frac{100027}{768}\rho^3 w^3 + \frac{129}{32}\rho^3 w \tag{4.16}$$

$$\dot{\theta} = \beta - \frac{57}{16}\rho^2 + \frac{31}{8}w^2 + \frac{16069}{256}\rho^2 w^2 - \frac{34047}{1024}\rho^4 - \frac{5817}{128}w^4 \tag{4.17}$$

$$\dot{w} = \frac{3}{2}\rho^2 + \frac{147}{16}\rho^4 + \frac{945579}{8192}\rho^6 + 5\,w^4 - w^2 - \frac{409}{32}\rho^2 w^2 - \frac{84141}{256}\rho^4 w^2 + \frac{84235}{256}\rho^2 w^4 - 120\,w^6\,, \tag{4.18}$$

where $\alpha = 0$, $\beta = 1$, $\gamma = 0$. The NF equations are quite complicated making it very hard to draw any sound conclusions. One can not find a combination such as ρ in Sprott C and E or $\rho^4 w$ in Sprott F.

5. THE RATIONAL NONPOLYNOMIAL TRANSFORMATION

In this work, we have been analyzing the NF approach, in which the system is replaced by a simpler system dictated by the resonances and a sequence of polynomial transformations. We have seen that a rational nonlinear transformation like $\xi = \rho^4 w$ can simplify a NF. Actually, a slightly more general approach can be used along these lines. It is sometimes possible to change the order at which a system becomes resonant and thus use a different simpler system by a rational nonpolynomial transformation [27]. We will first discuss an example given by Wiggins [16] which is resonant in the second order. Its linear part is given by

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \tag{5.1}$$

with the corresponding null space containing the two vectors,

$$\begin{bmatrix} 2x_1^2 \\ x_1 x_2 \end{bmatrix}, \begin{bmatrix} 0 \\ x_1^2 \end{bmatrix}. \tag{5.2}$$

This implies that the NF through the second order is given by

$$\dot{x_1} = x_1 + 2c_1 x_1^2 \tag{5.3}$$

$$\dot{x_2} = c_1 x_1 x_2 + c_2 x_1^2 . \tag{5.4}$$

The nonlinear transformation $x_1 = y_1, x_2 = y_2 x_1^2$ yields the fixed point $y_1 = c_2/2c_1^2$, $y_2 = -2c_1$ and the system in which this newly found fixed point is moved to the origin becomes

$$\dot{y_1} = \frac{c_2}{c_1^2} y_2 + 2 y_1 y_2 \tag{5.5}$$

$$\dot{y}_2 = -2c_1^2 y_1 + \frac{5}{2}\frac{c_2}{c_1} y_2 - \frac{c_2}{c_1^2} y_2^2 - 2y_2^2 y_1 + 5c_1 y_2 y_1 \,. \tag{5.6}$$

The eigenvalues of the linearized part A of this new matrix are $\lambda_1 = c_2/c_1^2$ and $\lambda_2 = \frac{5}{2}(c_2/c_1)$. In general, the new system is non resonant, except for the special cases $c_1 = 0$, $c_2 = 0$, $c_1 = 2k/5$ or $c_1 = 5k/2$. We now use this approach on some of the Sprott systems that we analyzed in chapter three.

5.1. The Sprott Systems

We start from the Sprott C system given in Equation 3.2, where the fixed points have been moved to the origin. Its linear part has the eigenvalues ($\alpha = 0$, $\beta = \sqrt{2}$, $\gamma = -1$) which make the system resonant in the third order. Drop bars, and consider the following nonlinear rational transformation:

$$x_1 = y_1 \tag{5.7}$$

$$x_2 = x_1^2 y_3 \tag{5.8}$$

$$x_3 = x_1 y_2 \tag{5.9}$$

which transforms the flow to

$$\dot{y}_1 = y_1 y_2(\pm 1 + y_1 y_3) \tag{5.10}$$

$$\dot{y}_2 = \mp 2 - y_1 - y_2^2(\pm 1 + y_1 y_3) \tag{5.11}$$

$$\dot{y}_3 = 1 - y_3 - y_2 y_3(\pm 1 + y_1 y_3) \tag{5.12}$$

This system has the following four fixed points:

$$(y_1 = \mp 2, y_2 = 0, y_3 = 1)\,, \left(y_1 = 0, y_2 = \pm\sqrt{2}i, y_3 = \frac{1}{1 \pm \sqrt{2}i}\right)$$

If we move the origin to the first fixed point by $y_1 \to y_1 - 2$, $y_2 \to y_2$, $y_3 \to y_3 + 1$, we obtain

$$\dot{y_1} = y_2(y_1 - 2)(-1 + y_1 - 2y_3 + y_1y_3) \tag{5.13}$$

$$\dot{y_2} = -y_1 - y_2^2(-1 + y_1 - 2y_3 + y_1y_3) \tag{5.14}$$

$$\dot{y_3} = 1 - (1 + y_3) - y_2(1 + y_3)(1 + (1 + y_3)(y_1 - 2)) \tag{5.15}$$

whose linear part has eigenvalues ($\lambda_{1,2} = \pm i\sqrt{2}$, $\lambda_3 = -1$), and the system still goes into resonance in the third order. The second fixed point gives an identical result.

Now we move the system to the origin about the third fixed point. This time the linear part reads

$$\begin{bmatrix} -\sqrt{2}i & 0 & 0 \\ \frac{-4-\sqrt{2}i}{9} & -1+\sqrt{2}i & \frac{-1-\sqrt{2}i}{3} \\ \frac{-1+2\sqrt{2}i}{3} & 0 & 2\sqrt{2}i \end{bmatrix} \tag{5.16}$$

with eigenvalues $(-\sqrt{2}i, 2\sqrt{2}i, -1+\sqrt{2}i)$. Notice that the eigenvalue structure emerges as $(\lambda_1, \lambda_2 = -2\lambda_1, \lambda_3)$ and the system goes into resonance in the fourth order. The fourth fixed point is the complex conjugate of the third one and gives a similar result.

The eigenvectors corresponding to these eigenvalues

$$\begin{bmatrix} -9(4+\sqrt{2}i) \\ \frac{-1+11\sqrt{2}i}{3} \\ 7+4\sqrt{2}i \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ -3 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}. \tag{5.17}$$

span a complete space.

We now treat the Sprott E system with the fixed point moved to the origin given in Equation 3.26. We have seen that the system is resonant in the third order. Again

we drop bars and consider two alternatives for the nonlinear transformation:

$$x_1 = y_1 \tag{5.18}$$

$$x_2 = y_2 x_1 \tag{5.19}$$

$$x_3 = y_3 x_1 \tag{5.20}$$

and

$$x_3 = y_3 \tag{5.21}$$

$$x_1 = y_1 x_3 \tag{5.22}$$

$$x_2 = y_2 x_3 \tag{5.23}$$

Under the first alternative the flow becomes

$$\dot{y}_1 = \frac{y_1 y_3}{16} + y_1^2 y_2 y_3 \tag{5.24}$$

$$\dot{y}_2 = \frac{1}{2} + y_1 - y_2 - \frac{y_2 y_3}{16} - y_1 y_2 y_3^2 \tag{5.25}$$

$$\dot{y}_3 = -4 - \frac{y_3^2}{16} - y_1 y_2 y_3^2 \tag{5.26}$$

This system has two fixed points:

$$\left(y_1 = 0, y_2 = \tfrac{2+i}{5}, y_3 = -8i\right)$$

$$\left(y_1 = 0, y_2 = \tfrac{2-i}{5}, y_3 = 8i\right)$$

When the system is moved to the origin about the first fixed point the linearized part becomes

$$\begin{bmatrix} -\frac{i}{2} & 0 & 0 \\ \frac{-7+24i}{25} & \frac{-2+i}{2} & -\frac{2+i}{80} \\ \frac{64(2+i)}{5} & 0 & i \end{bmatrix}, \tag{5.27}$$

for which the eigenvalues $(-i/2, i, -1 + i/2)$ have the familiar structure $(\lambda_1, \lambda_2 =$

$-2\lambda_1, \lambda_3)$ and the system is resonant in the fourth order. The eigenvectors corresponding to these eigenvalues are

$$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -\frac{3}{128} - \frac{3i}{64} \\ \frac{31}{1280} + \frac{3i}{1280} \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ -\frac{1}{40} \\ 1 \end{bmatrix}, \tag{5.28}$$

and they form a complete space. The second alternative for transformation results in

$$\dot{y}_1 = \frac{1}{16} + 4y_1^2 + y_2 y_3 \tag{5.29}$$

$$\dot{y}_2 = \frac{y_1}{2} - y_2 + y_1^2 + 4y_1 y_2 \tag{5.30}$$

$$\dot{y}_3 = -4y_2 y_3 \tag{5.31}$$

This system has fixed points at

$$\left(y_1 = \tfrac{i}{8}, y_2 = -\tfrac{1}{40} + \tfrac{i}{20}, y_3 = 0\right)$$
$$\left(y_1 = -\tfrac{1}{8}, y_2 = -\tfrac{1}{40} + \tfrac{i}{20}, y_3 = 0\right)$$

If the system is moved to the origin about the first fixed point the coefficient matrix of the linearized part is as follows

$$\begin{bmatrix} -\frac{i}{2} & 0 & 0 \\ -\frac{1}{64} & \frac{-2+i}{2} & \frac{2+i}{5} \\ \frac{-1+2i}{40} & 0 & i \end{bmatrix}, \tag{5.32}$$

with the same eigenvalues $(\lambda_1 = -i/2, \lambda_2 = i, \lambda_3 = -1 + i/2)$ as above and the system is again resonant in the fourth order. The corresponding eigenvectors

$$\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} -24 + 12i \\ \frac{61+63i}{160} \\ 1 \end{bmatrix}, \begin{bmatrix} 0 \\ \frac{2}{5} \\ 1 \end{bmatrix} \tag{5.33}$$

again span a complete space. Thus, the second fixed point is the complex conjugate of the first one, and gives a similar result.

We observe that resonance is pushed from the third to the fourth order when the fixed points in the transformed variables are complex. It is easy to see that complex fixed points in the transformed variables mean centers or foci, while real fixed points imply centers by looking at a simple example. If the system $\dot{x}_1 = x_2,\ \dot{x}_2 = -x_1$ is transformed by $x_2 = y_2 x_1$, we get $\dot{y}_1 = y_1 y_2,\ \dot{y}_2 = -(1 + y_2^2)$. This system has the complex fixed point, $y_1 = 0,\ y_2 = \pm i$. On the other hand, if the system is replaced by $\dot{x}_1 = x_2,\ \dot{x}_2 = x_1$, the same transformation gives $\dot{y}_1 = y_1 y_2,\ \dot{y}_2 = (1 - y_2^2)$ with the real fixed point $y_1 = 0,\ y_2 = \pm 1$. The strength of the transformation is that nonlinear terms would influence the equilibrium point and the eigenvalues of the linearized system in the new variables.

The eigenvalue spectrum $(\lambda_1, \lambda_2 = -2\lambda_1, \lambda_3)$ that is obtained in the transformed Sprott C and E cases gives the NF:

$$\frac{du_1}{dt} = \lambda_1(u_1 + c_1 u_1^3 u_2 + c_2 u_1^5 u_2^2) \tag{5.34}$$

$$\frac{du_2}{dt} = -2\lambda_1(u_2 + c_3 u_1^2 u_2^2 + c_4 u^4 u_2^3) \tag{5.35}$$

$$\frac{du_3}{dt} = \lambda_3(u_3 + c_5 u_1^2 u_2 u_3 + c_6 u_1^4 u_2^2 u_3) \tag{5.36}$$

through the ninth order. λ_i and c_i are constants whose values depend on the nonlinear terms. Using $u_4 = u_1^2 u_2$, this NF can be transformed to:

$$\frac{du_1}{dt} = \lambda_1 u_1(1 + c_1 u_4 + c_2 u_4^2) \tag{5.37}$$

$$\frac{du_2}{dt} = -2\lambda_1 u_2(1 + c_3 u_4 + c_4 u_4^2) \tag{5.38}$$

$$\frac{du_3}{dt} = \lambda_3 u_3(1 + c_5 u_4 + c_6 u_4^2) \tag{5.39}$$

$$\frac{du_4}{dt} = c_7 u_4^2 + c_8 u_4^3 \tag{5.40}$$

which looks like a Hopf bifurcation NF.

6. CHAOTIC BEHAVIOR POSSIBILITIES IN NMR

MR is a subject of intensive research in atomic, molecular and nuclear physics, a great deal of useful information about the structure of matter has been obtained from resonance studies. NMR has also been very useful in medicine [28, 29].

The dynamics of the basic nuclear magnetic resonance phenomenon is governed by the Bloch equation that describes the motion of net magnetization moment vector $\mathbf{M}$ in the presence of a magnetic field $\mathbf{B(t)}$:

$$\frac{d\mathbf{M}}{dt} = \mathbf{M} \times g\mathbf{B} - \frac{M_{x_1}\hat{\mathbf{x}}_1 + M_{x_2}\hat{\mathbf{x}}_2}{T_2} - \frac{(M_{x_3} - M_o)\hat{\mathbf{x}}_3}{T_1} \,. \tag{6.1}$$

Here g is the gyromagnetic ratio, T_1 is the spin-lattice (longitudinal) relaxation time constant, T_2 is the spin-spin (transverse) relaxation time constant. In MR imaging, $\mathbf{B}$ consists of three types of magnetic fields: 1) the main static field $\mathbf{B_o}$ applied in the longitudinal direction that causes precession, 2) radio frequency fields $\mathbf{B}_1(t)$ applied in the transverse direction which are used for excitation; and 3) the linear gradient fields $\mathbf{G(t)}$, which are used for spatial localization.

Recent developments in MRI techniques have made it possible to examine functions of the human brain (fMRI) [30]. During experiments human subjects are being stimulated in different ways. The problem is to apply a possible visual or audio excitation and determine the response of the subject to the stimulus. In particular, the degree of response of different sections of brain is measured [31]. Since the stimulus is not measured, one is led to analyze a statistical parameter belonging to the response from different regions [32]. The classical approach uses statistical methods such as the Student t-test, cross-correlation methods, analysis of variance (ANOVA) to analyze this time series of observations. There are a number of software packages devoted to fMRI analysis, most notable of which are SPM [33] and AFNI [34]. Since fractional Brownian motion is useful in modeling time series, we have proposed the rescaled range (R/S) analysis as a statistic on responses originating from different regions of the brain.

Such analysis on the fMRI signals has been performed [35].

6.1. Fractal Brownian Motion

We consider the one dimensional random walk problem in order to analyze the structure of the brain and the possible excitation and response. We can assume that the step length ξ in the random walk has a Gaussian distribution

$$p(\xi,\tau) = \frac{1}{\sqrt{4\pi D\tau}} exp\left(-\frac{\xi^2}{4D\tau}\right) . \tag{6.2}$$

The mean square jump distane of this process is

$$\langle \xi^2 \rangle = 2D\tau . \tag{6.3}$$

Here D is the diffusion constant. The observed series $\xi_1, \xi_2, \xi_3, \ldots$ are assumed to be steps of the random walk so that the position of the particle on the x axis is

$$X(t = n\tau) = \sum_{i=1}^{n} \xi_i . \tag{6.4}$$

The position of the particle undergoing Brownian motion is a random function of time. Wiener has defined this random function as follows [37]: For any time interval (t_o, t), if the change in the position of the Brownian particle is given by

$$X(t) - X(t_o) = \xi |t - t_o|^H \qquad (t \geq t_o) \tag{6.5}$$

the particle is undergoing random motion. Here H=1/2 signifies ordinary Brownian motion. Mandelbrot [36] has generalized ordinary Brownian motion by generalizing H so as it can assume values in the interval zero to one and named this the fractal Brownian motion, $B_H(t)$. The important point is that unlike ordinary Brownian motion past and future steps in fractal motion are correlated with a coefficient of correlation

$$C = 2^{2H-1} - 1 . \tag{6.6}$$

We can see that there is no correlation for H=1/2. For $H > 1/2$, $C > 0$, this means that an increase in the past will imply an increase in the future (persistence).

6.1.1. Empirical Hurst Law and R/S Analysis

Let us assume that we have an experimental time series $\xi(t)$. The average of $\xi(t)$ over the period of τ time interval $[t = 1, t = \tau]$ is given by

$$\langle \xi \rangle_\tau = \frac{1}{\tau} \sum_{t=1}^{\tau} \xi(t) \, . \tag{6.7}$$

$X(t, \tau)$ is defined for any t as the following sum of deviations from the average.

$$X(t, \tau) = \sum_{u=1}^{\tau} \xi(u) - \langle \xi \rangle_\tau \, . \tag{6.8}$$

The difference between the maximum value and minimum value of this series, R 's given by

$$R(\tau) = max \; X(t, \tau) - min \; X(t, \tau), \qquad t \in [1, \tau] \tag{6.9}$$

and R is defined as the range of the series $X(t, \tau)$. The standard deviation can be calculated from

$$S(\tau) = \left(\frac{1}{\tau} \sum_{t=1}^{\tau} [\xi(t) - < \xi >_\tau]^2 \right)^{1/2} \, . \tag{6.10}$$

For many time series, Hurst studied the dimensionless ratio, $R(\tau)/S(\tau)$ and proposed that observed time series follow the empirical law,

$$\frac{R}{S} = (\tau/2)^H . \tag{6.11}$$

H is called the Hurst exponent. Feder [37] has shown that for self-affine series, the following relation exists between the localized fractal dimension and the Hurst exponent

H,

$$d = 2 - H \qquad 0 < H < 1 \,. \tag{6.12}$$

From here, one can see that one can estimate H by a log-log plot of R/S against τ and estimate the fractal dimension.

Preliminary results of such an analysis have been reported by Nalcıoğlu et. al. [35]. The adverse effects of noise, the fact that the brain has spatial fractal character and the applied excitation has temporal fractal character forces one to perform a detailed analysis with relatively large amount of data and relatively long computer time usage. These requirements present two important practical drawbacks; one can not keep subjects for times long enough to get a low signal to noise ratio in every voxel and one can not wait for the long computation time necessary, if one wants results rapidly. These drawbacks force one to rely on the classical methods for this purpose.

6.2. Possible Chaoticity Deep in the Atomic Level

The relatively inconclusive nature of our findings leaves one in doubt whether the chaotic behavior, if any, is due to the fractal nature of the tissues in question or chaotic behavior inherent in the MRI system. To this end it might be useful to search for chaos deep in the atomic level.

In the simplest approximation the Bloch equations are linear. However they possess damping terms whose coefficients have magnitudes which are much smaller than the magnitudes of the oscillatory terms caused by the direct current magnetic field. Under two circumstances, radiation damping [38] and DDI [39], nonlinear terms have been reported to arise. In the former case the equations can be linearized by Lie group analysis. Chaotic attractors have been observed in the NSS with DDI proposed by Khomeriki. The system is governed by the following set of six nonlinear equations

[39]:

$$v = \begin{bmatrix} -x_1 + 6cx_5x_6 \\ -x_2 + 2bx_3 - 6cx_4x_6 \\ 2(-bx_2 - x_3 + 1) \\ -x_4 + 4cx_2x_6 + 2cx_3x_5 \\ -4cx_1x_6 - x_5 + 2bx_6 - 2cx_3x_4 \\ 2(-bx_5 - x_6 - cx_1x_5 + cx_2x_4) \end{bmatrix}, \tag{6.13}$$

where x_1 and x_2 are the components of the transverse polarization in the rotating frame, x_3 is the longitudinal polarization, x_4, x_5 , and x_6 are the components of the quantum-statistical average of the spin located in the lattice site as reported by Khomeriki.

The author claims that chaos is observed for the parameter values of $c = 10$ and $b = 2.69$ by just showing a limited portion of the phase portrait for transverse polarization in the rotating frame $p_t = \sqrt{x_1^2 + x_2^2}$ versus longitudinal polarization p_l (Figure 6.1). Although this line of reasoning is compelling, further analysis reveals a more complicated picture. The power spectrums obtained by a Fourier transform for the parameter values presented in the mentioned work are shown in the following figures. As observed $b = 2.42$ (Figure 6.2) gives a period one and $b = 2.49$ (Figure 6.4) gives a period two limit cycle. On the other hand $b = 2.69$ (Figure 6.3) gives a very interesting result: although this system could be classified chaotic from one point of view (it has $1/f$ "background" noise) it also mimics the power spectrum of the period one system. So we might coin the phrase "pseudo-chaotic" for this phenomenon. The three real valued fixed points and the corresponding eigenvalues of A for the parameters $c = 10, b = 2.69$ are given in table 6.1. The other six complex fixed points are not considered. A rich structure of eigenvalues that give rise to resonances in the Siegel domain are apparent.

Numerical simulations give ($\lambda_1 = 2.315, \lambda_2 = 0.008, \lambda_3 = -1.298, \lambda_4 = -2.835, \lambda_5 = -3.123, \lambda_6 = -6.619$) as the Liapunov exponents. Despite the fact the system is "pseudo-chaotic" numerical simulation gives a positive Liapunov exponent. This prob-

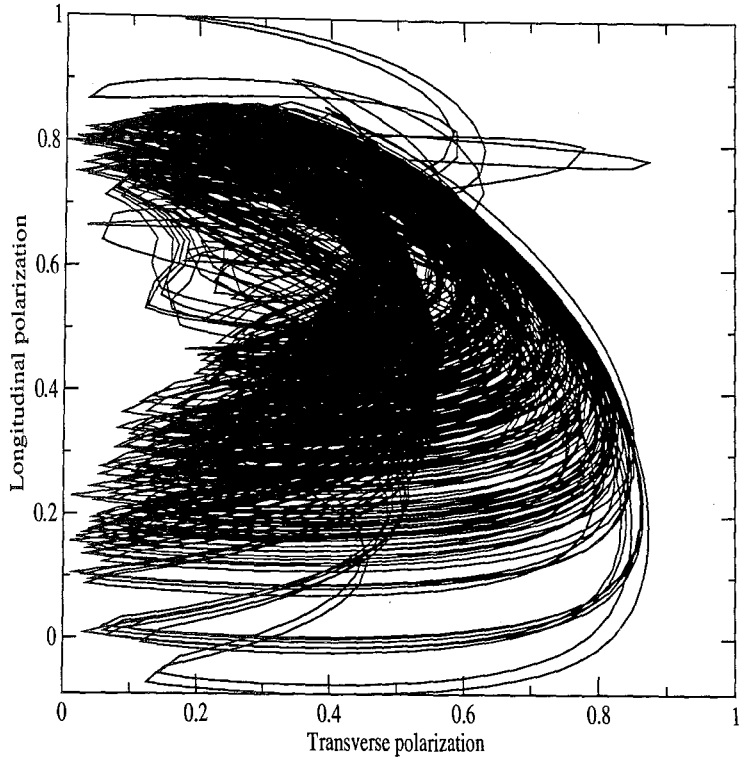

Figure 6.1. Phase portrait for p_l versus p_t polarization

Table 6.1. Fixed points and eigenvalues of A for the NSS

Fixed Points	**Eigenvalues of A**
$x_1 = -0.3899, x_2 = 0.0847,$ $x_3 = 0.7721, x_4 = \pm 0.3044,$ $x_5 = \mp 0.0291, \pm x_6 = 0.2227$	$(1.9893, -1.7525, \pm(6.8799 \pm 16.2881)$
$x_1 = 0, x_2 = 0.3477,$ $x_3 = 0.0646, x_4 = 0,$ $x_5 = 0, x_6 = 0$	$(\pm 5.5501i, \pm(1.3957 \pm 3.5342i)$

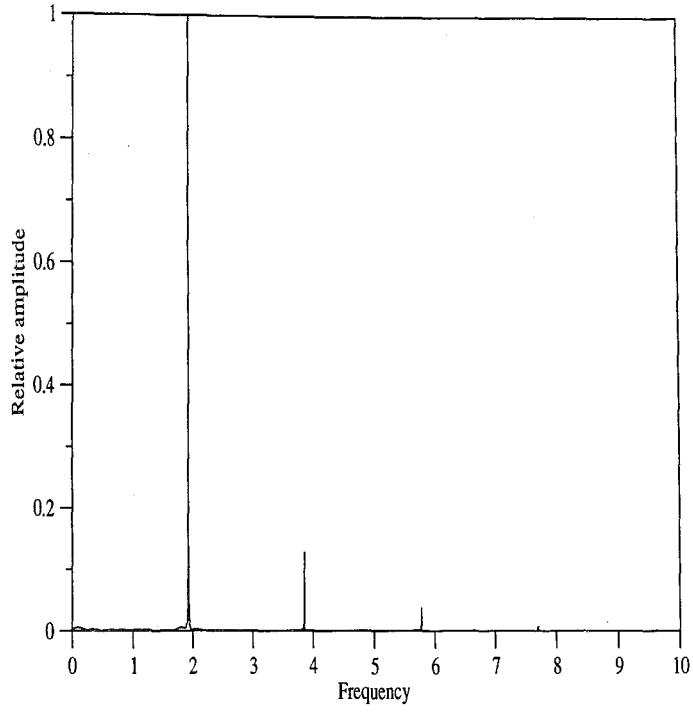

Figure 6.2. Power spectrum for b=2.42

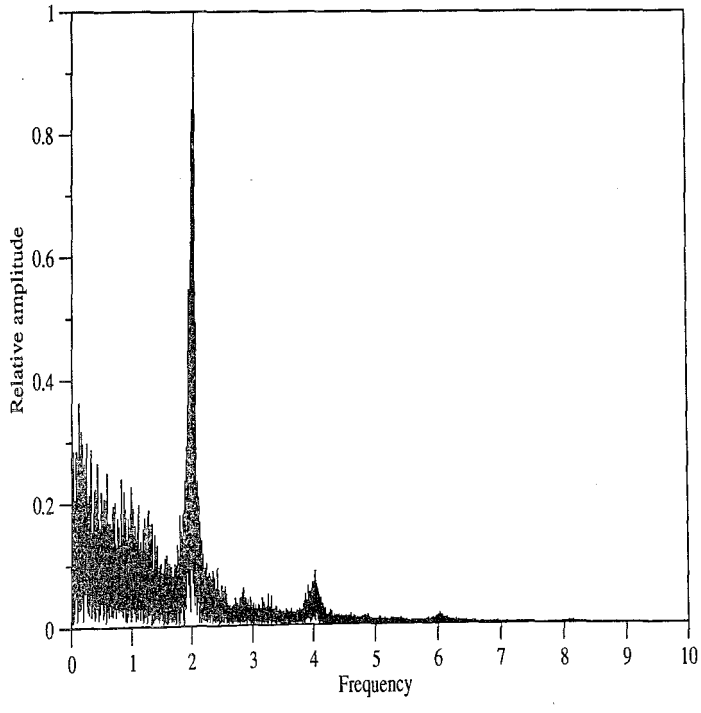

Figure 6.3. Power spectrum for b=2.69

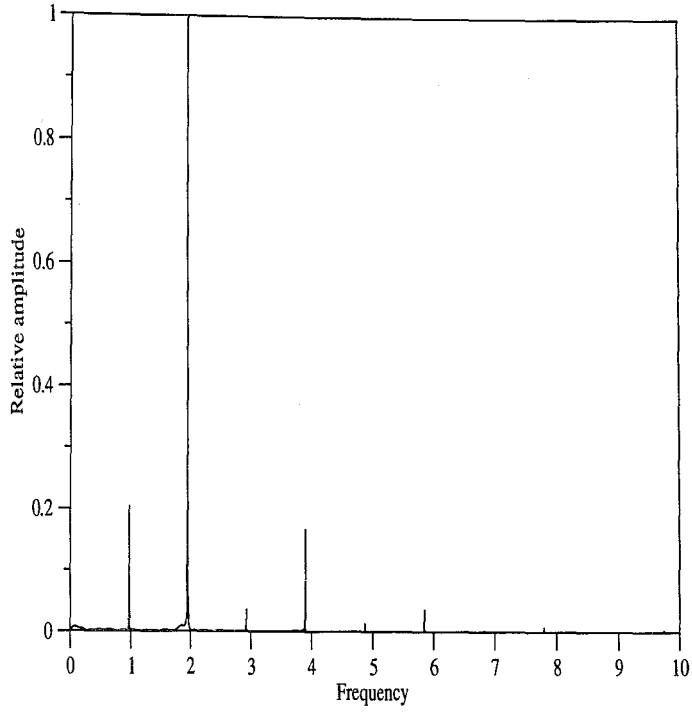

Figure 6.4. Power spectrum for b=2.49

ably means numerical methods for the estimation of the Liapunov spectrum overemphasizes the background noise rather than the sharp peaks. The reason is that they mostly take the volume of the attractor which is dominated by the background noise.

7. CONCLUSION

Although the linearized form of a dynamical system near its equilibrium point can give a partial understanding of the Liapunov spectrum, more powerful techniques than simple scenarios based on the linearized system are needed. Developing such techniques is a difficult and system specific task. In this work, some attempts at finding such techniques are reported. A number of the three variable systems with quadratic nonlinearities proposed by Sprott have been studied by the NF technique.

Chaotic properties of systems are best studied by using the Liapunov exponent spectrum. The usual method of obtaining this spectrum is numerical simulation. Numerical errors can often lead to difficulties in distinguishing a zero Liapunov exponent from a slightly positive one. Hence, other methods should be used to understand the road to chaos for dynamical systems. Mapping the system to a simpler one with known properties is a possibility.

The NF transformation replaces the original system by a simpler system dictated by resonances and a sequence of polynomial, near identity transformations. In spite of convergence problems, we have demonstrated that the resulting representation gives an adequate picture of certain nonlocal properties of attractors near Hopf bifurcation points. Systems that have the Hopf bifurcation property are frequently used to model natural phenomena. Numerical work reveals good agreement between the original system and the NF representation truncated at the seventh order. Nonlinear dependence between the variable θ on ρ that occur in the NF parametrization reveals the route to chaos and provides an estimate of the largest Liapunov exponent. There is a connection between complex conjugate eigenvalue pairs in the linearized system and a zero Liapunov exponent in all systems with the Hopf bifurcation property that we have studied. However, this property does not persist if there is both a complex conjugate pair and a zero eigenvalue, which introduces additional resonant terms. The linearized part of the Sprott D system has eigenvalues ($\lambda_{1,2} = \pm i, \lambda_3 = 0$), but its Liapunov spectrum is (0.163, 0, -1.163). There is only one zero Liapunov Exponent. A candidate

for a comparable counterexample has been observed in the nuclear spin system with dipole dipole interaction.

Since resonances dictate the simpler system, non polynomial transformations that change the linearized system can be useful. Singularities seem to be necessary in resonant cases, as first pointed out by Warner et. al. [13]. Rational nonlinear transformations have been studied, they show the promise of answering these problems in specific cases, although a recipe for handling the general case is not yet at hand, since there are too many possibilities. There is a connection between the complex equilibrium points of the transformed system and presence of centers or foci. The eigenvalue spectrum $(\lambda_1, \lambda_2 = -2\lambda_1, \lambda_3)$ together with the associated NF seem to arise frequently in resonant cases. Nonlinear rational transformations have resemblances to the Linstedt and central manifold approaches and clearly help enhance our understanding of such systems.

A general recipe, that we have not yet been able to find, would be helpful in attacking bifurcation problems. Furthermore, non uniqueness of the NF transformation, the possibility of alternative schemes for estimating the largest Liapunov exponent and clarification of the resonant system when the linearized system has the eigenvalue spectrum $(0, \pm i\beta)$ leading to a more complicated NF, are problems that merit further study.

The Nuclear magnetic resonance system has been studied. A fractal dimension estimate procedure using the R/S analysis on responses originating from different regions of tissue has been proposed. There are practical limitations of this proposal. The nuclear spin system with dipole-dipole interactions proposed by Khomeriki has been investigated, evidences of chaotic behavior and its possible transient character has been observed.

APPENDIX A: SYMBOLIC CALCULATIONS

Symbolic algebra systems were originally designed to supplement numerically oriented languages such as Fortran with limited algebraic capabilities of expanding, factoring, substitution and differentiation. They were later expanded into systems that can algebraically perform vector and matrix operations, symbolic integration, solve simultaneous equations and pattern matching. Programming capabilities, integration of numerical techniques and graphical capabilities are built into virtually all of the symbolic programming packages. All symbolic algebra systems have an interface to emit output in the Fortran or C language format, thus enabling the user to use symbolic processing results in numerical work.

Nowadays, the two most widely used symbolic algebra systems are Mathematica [41] and Maple [42]. Their programming environment resembles that of the C language. Both of them are based on the C programming language and represent algebraic expressions by trees. Mathematica has been developed from Wolfram's symbolic manipulation package for evaluating Feynman Diagrams. Maple is also used as the symbolic algebra interface in well-known program packages, such as Matlab or Scientific Workplace.

Three older symbolic manipulation packages, REDUCE [43], MACSYMA [44] and DERIVE are based on the LISP language and represent algebraic expressions by linked lists. REDUCE and MAXIMA are widely used in the Physics community. REDUCE is commercial, but available relatively cheaply for personal use, MACSYMA was originally a commercial software, subsequently a version of the language MAXIMA has been released under GNU General Public License [45], and is freely available. Their programming environment resembles that of the ALGOL and PASCAL languages.

Symbolic algebra packages are widely used for accurately carrying out manipulations involving repeated application of routine algebraic operations. In nonlinear dynamical systems, many approaches, including the NF expansion, involve such oper-

ations. Before the availability and use of symbolic algebra computer programs, calculations were generally carried out only through the first or second orders. Higher order calculations involve significant algebraic complexity, they are best done by computer algebra software accurately. Many authors have published symbolic algebra codes for NF calculation [46, 47, 14]. Rand and Armbuster [48] have written a book containing MAXIMA codes for nonlinear dynamical systems. Their NF expansion code has been adapted to REDUCE and expanded to provide treatment of the resonant cases in this thesis. Three REDUCE [43] routines are presented in Appendices A, B and C. The first one, given in Appendix A is named NF. It is a direct translation of the Rand Armbuster MAXIMA code into REDUCE. A Maple translation of this code is also available.

The second one, given in Appendix B, is named NFRES has the same functionality as NF if there are no resonances, and contains enhancements to calculate the Hopf NF of a system of dimension n up to the eleventh order (the limitation to the eleventh order is due to the underlying LISP system). Complex conjugate pairs are directly created for faster execution, more efficient bookkeeping and memory allocation. It takes about ten minutes to compute the NF of a three dimensional complex diagonal system. The program is very well tested and to our best knowledge has no bugs up to the sixth order. In the sixth order there still remains a bug that we are aware of. The bug stems from the fact that, if there is a resonance, not all variables are included in the solution, there is an arbitrariness both in the NF expansion and in variable selection in the resolving equations. The variable selection process sometimes pollutes the arbitrary terms that enter the NF expansion. We have been forced to use manual selection for the variables, this removal by hand leaves no trace on the higher orders.

The third one, called NFRNULL is given in Appendix C. It extends the regular NFRES() utility to give both the null space in orthonormal form and the removable vector space in case of resonance. It was originally intended to automate the selection procedure for NFRES. Memory limitations in the Portable Standard Lisp implementation of REDUCE has once more blocked our attempts at this automation. The program

has been used to generate NFs for diagonal and non diagonal linear parts. The basis vectors so found are occasionally used in the hand selection procedure.

APPENDIX B: NF CALCULATION PROGRAM

```
COMMENT PROGRAM NUMBER 5: NF(), NORMAL FORM TRANSFORMATIONS. SEE PAGE 62 IN
  "PERTURBATION METHODS, BIFURCATION THEORY AND COMPUTER ALGEBRA" BY RAND AND
  ARMBUSTER FOR THE ORIGINAL MAXIMA CODE;

COMMENT THIS FILE CONTAINS NF(), A NORMAL FORM UTILITY FUNCTION.
   IT ALSO CONTAINS THESE ADDITIONAL FUNCTIONS
   GEN(N) WILL GENERATE A HOMOGENEOUS ORDER N TRANSFORMATION.
   DECOMPOSE() ISOLATES THE COEFFICIENTS OF THE NEW EQUATIONS.
   VARS(N) GENERATES A LIST OF UNKNOWN COEFFICIENTS OF DEGREE N.
   HOPF(K), FOR K=2,3,4,5,6,7,8 SOLVES FOR THE COEFFICIENTS OF A SYSTEM OF
   N DE'S SO AS TO PUT THE EQS IN HOPF NORMAL FORM;

PROCEDURE SERI(ifade);
comment this expands a polynomial expression into a series containing
homogeneous terms of order up to m;
begin scalar ifade2,sonuc,set3,set4,j;
on div;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
set3:=coeff(ifade2,zt);
sonuc:=for j:=1:m+1 sum Part(set3,j);
write(sonuc);
return(sonuc);
end;
PROCEDURE NF();
BEGIN
COMMENT NEW VARIABLE NAMES?;
WRITE "DO YOU WANT TO ENTER NEW VARIABLE NAMES (Y/N)?";
OPERATOR X,Y;
TEST := XREAD (1);
IF TEST = N THEN GO TO JUMP;
WRITE "HOW MANY EQS";
N := XREAD (1);
ARRAY TRF(n);
FOR I:=1:N DO BEGIN
WRITE "SYMBOL FOR OLD X(",I,")";
X(I) := XREAD (1);
END;
FOR I:=1:N DO BEGIN
WRITE "SYMBOL FOR NEW X(",I,")";
Y(I) := XREAD (1);
END;
FOR I:=1:N DO DEPEND X(I),TT;
FOR I:=1:N DO DEPEND Y(I),TT;
```

```
JUMP:
COMMENT  NEW D.E.'S?;
WRITE "DO YOU WANT TO ENTER NEW D.E.'S (Y/N)?";
TEST:=XREAD(1);
IF TEST = N THEN GO TO LOOP;
ARRAY RH(N);
FOR I:=1:N DO BEGIN
WRITE "ENTER RHS OF EQ. NO.",I,",D",X(I),"/DT =";
RH(I):=XREAD(1);
WRITE "D",X(I),"/DT =",RH(I);
end;

LOOP:

COMMENT NEAR-IDENTITY TRANSFORMATION;
WRITE "INPUT NEAR-IDENTITY TRANSFORMATION";
WRITE "(USE PREV FOR PREVIOUS TRANSFORMATION)";
FOR I:=1:N DO BEGIN
ROW:=I;
PREV :=TRF(I);
WRITE X(I),"=",Y(I),"+ ?";
TRF(I) :=XREAD (1);
WRITE X(I),"=",Y(I)+TRF(I);
END;
COMMENT INPUT TRUNCATION ORDER;
TRANS := for i:=1:n collect X(I)=Y(I)+TRF(I);
WRITE "ENTER TRUNCATION ORDER (HIGHEST ORDER TERMS TO BE KEPT)";
M := XREAD(1);
Matrix newrhs(n,1);
detdf();
FOR I:=1:N DO WRITE PART(NEWDES,I);
COMMENT ENTER ANOTHER TRANSFORMATION?;
WRITE "DO YOU WANT TO ENTER ANOTHER TRANSFORMATION (Y/N)";
BRANCH:=XREAD(1);
IF BRANCH = Y THEN GO to LOOP;
write NEWDES;
END;

procedure detdf;
COMMENT THIS SUBROUTINE SUBSTITUTES THE TRANSFORMATION INTO THE
SYSTEM AND GENERATES THE DIFFERENTIAL EQUATIONS TO ORDER M;
begin scalar i,j,sublist,invsublist;
operator ww;
Matrix Jacob2(n,n),eye(n,n);
for i:=1:n do eye(i,i):=1;
invsublist:=for i:=1:n collect ww(i)=y(i);
sublist:=for i:=1:n collect y(i)=ww(i);
weight(ww(1)=1,ww(2)=1,ww(3)=1,ww(4)=1,ww(5)=1,ww(6)=1,ww(7)=1,ww(8)=1);
```

```
wtlevel(m);
for i:=1:n do for j:=1:n do
JACOB2(I,J):=sub(sublist,DF(TRF(I),Y(J)));
write "Jacobian OK";
COMMENT TRANSFORM THE D.E.'S;
MATRIX TEMP2(n,1),temp3(n,n);
for i:=1:n do TEMP2(i,1) :=sub(sublist,sub(TRANS,rh(i)));
write "transformation substituted";
COMMENT SOLVE FOR THE TRANSFORMED DERIVATIVES;
temp3:=eye;
for i:= 1:m-1 do
begin
temp3:=eye- Jacob2*temp3;
end;
temp2:=temp3*temp2;
write "Jacobian inverted";
COMMENT TAYLOR EXPAND THE RESULTING EQS;
write "begin Taylor";
for i:=1:N DO NEWRHS(I,1):=sub(invsublist,temp2(i,1));
NEWDES:=for i:=1:n collect DF(Y(I),TT)=NEWRHS(I,1);
clear(ww(1),ww(2),ww(3),ww(4),ww(5),ww(6),ww(7),ww(8));
clear ww;
end;

procedure hopf(m1);
COMMENT SPLITS OFF EQUATIONS AND SOLVES FOR THE NORMAL FORM;
begin scalar i,ii,neqtns,eqtns,expression,sublist,sublist2;
eqtns:={};
sublist:=for i:=1:n collect y(i)=wtt**(for ii:=0:(i-1) sum m1**ii);
sublist2:=for i:=1:n collect y(i)=wtt*y(i);

for i:=1:n do BEGIN
  expression:=sub(sublist2,newrhs(i,1));
  expression:=expression-coeffn(expression,wtt,0)-wtt*coeffn(expression,wtt,1);
  expression:=sub(sublist,expression);
  neqtns:=coeff(expression,wtt);
  eqtns:=append(neqtns,eqtns);
   END;
   solution:=solve(eqtns,slist);
end;

operator cc;

procedure gen(m1);
comment iord is the order, n is number of variables, m is degree
This generates a homogeneous polynomial of order m in n variables
with arbitrary coefficients for use in generating the arbitrary
near identity transformation;
```

```
begin scalar s,i1,i2,i3,i4,i5,i6,i7,i8,i9,j;
if(not(numberp(iord)) or mod(iord,n)=0) then
<<iord:=1;slistold:={}>>
else <<iord:=iord+1;slistold:=slist>>;

if n=2 then begin
 s:=for i1:=0:m1 sum cc(iord,i1,m1-i1)*y(1)**i1*y(2)**(m1-i1);
 slist:=for i1:=0:m1 collect cc(iord,i1,m1-i1);
 end;

if n=3 then begin
 s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum cc(iord,i1,i2,m1-i1-i2)
   *y(1)**i1*y(2)**i2*y(3)**(m1-i1-i2);
 slist:=for i1:=0:m1 join for i2:=0:m1-i1 collect cc(iord,i1,i2,m1-i1-i2);
 end;

if n=4 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum for i3:=0:m1-i1-i2 sum
  cc(iord,i1,i2,i3,m1-i1-i2-i3)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**(m1-i1-i2-i3);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join for i3:=0:m1-i1-i2 collect
  cc(iord,i1,i2,i3,m1-i1-i2-i3);
end;

if n=5 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
  cc(iord,i1,i2,i3,i4,m1-i1-i2-i3-i4)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**(m1-i1-i2-i3-i4);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 collect
  cc(iord,i1,i2,i3,i4,m1-i1-i2-i3-i4);
end;

if n=6 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
  cc(iord,i1,i2,i3,i4,i5,m1-i1-i2-i3-i4-i5)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**(m1-i1-i2-i3-i4-i5);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 collect
cc(iord,i1,i2,i3,i4,i5,m1-i1-i2-i3-i4-i5);
end;
```

```
if n=7 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
for i6:=0:m1-i1-i2-i3-i4-i5 sum
  cc(iord,i1,i2,i3,i4,i5,i6,m1-i1-i2-i3-i4-i5-i6)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**i6*y(7)**(m1-i1-i2-i3-i4-i5-i6);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 join
for i6:=0:m1-i1-i2-i3-i4-i5 collect
cc(iord,i1,i2,i3,i4,i5,i6,m1-i1-i2-i3-i4-i5-i6);
end;

if n=8 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
for i6:=0:m1-i1-i2-i3-i4-i5 sum
for i7:=0:m1-i1-i2-i3-i4-i5-i6 sum
  cc(iord,i1,i2,i3,i4,i5,i6,i7,m1-i1-i2-i3-i4-i5-i6-i7)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**i6*y(7)**i7*y(8)**(m1-i1-i2-i3-i4-i5-i6-i7);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 join
for i6:=0:m1-i1-i2-i3-i4-i5 join
for i7:=0:m1-i1-i2-i3-i4-i5-i6 collect
cc(iord,i1,i2,i3,i4,i5,i6,i7,m1-i1-i2-i3-i4-i5-i6-i7);
end;
slist:=append(slistold,slist);
return(s);
end;

END;
```

APPENDIX C: RESONANT NF CALCULATION PROGRAM

```
COMMENT THIS FILE CONTAINS NFRES(), A NORMAL FORM UTILITY FUNCTION.
   IT ALSO CONTAINS THESE ADDITIONAL FUNCTIONS
   GEN(N) WILL GENERATE A HOMOGENEOUS ORDER N TRANSFORMATION.
   DECOMPOSE() ISOLATES THE COEFFICIENTS OF THE NEW EQUATIONS.
   VARS(N) GENERATES A LIST OF UNKNOWN COEFFICIENTS OF DEGREE N.
   HOPF(K), FOR K=2,3,4,5,6,7,8,9,10,11
   SOLVES FOR THE COEFFICIENTS OF A SYSTEM OF
   N DE'S SO AS TO PUT THE EQS IN HOPF NORMAL FORM;

COMMENT GENERATES COMPLEX C ONJUGATE PAIRS DIRECTLY;

PROCEDURE SERI(ifade);
comment this expands a polynomial expression into a series containing
homogeneous terms of order up to m;
% works for polynomial cases, otherwise use SERINONP which is slower;
% expands n variable function of variables y(k) up to order m.
begin scalar ifade2,sonuc,set3,set4,j;
on div;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
set3:=coeff(ifade2,zt);
sonuc:=for j:=1:m+1 sum Part(set3,j);
write(sonuc);
return(sonuc);
end;

procedure S(W);
comment this routine gives a list of variables in a list of
expressions given as its argument;

begin scalar i,sw,mv;
mvarlst:={};
operator mvlst;
uzunluk:=0;
sw:=w;
LOOP1;
IF(MV:= MAINVAR NUM(SW)) NEQ 0 THEN
begin
uzunluk:=uzunluk+1;
mvlst(uzunluk):=MV;
SW:=SUB(MV=-1.728374658609345172109,SW);
GOTO LOOP1;
```

```
END
ELSE
BEGIN
for i:=1:uzunluk do
if freeof(mvlst(i),cc) then 1 else mvarlst:=append(mvarlst,{mvlst(i)});
clear(mvlst);
return(mvarlst);
END;
end;

PROCEDURE CCONJ(ifade);

comment this avoids taking real and imaginary parts of variables;

begin scalar donusum;
donusum:=sub(i=-i,sublst,ifade);
return(donusum);
end;

PROCEDURE SERICN(ifade,mm);
% works for polynomial cases, otherwise use SERINONPCN which is slower;
% expands n variable function of variables y(k) giving terms of order m.;
begin scalar ifade2,sonuc,set4,j;
on div;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
sonuc:=coeffn(ifade2,zt,mm);
write(sonuc);
return(sonuc);
end;

PROCEDURE SERINONPCN(ifade,mm);
% Works in general, requires that Taylor be loadad within calling routine;
% Gives m'th order terms;
begin scalar ifade2,sonuc;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
sonuc:=coeffn(taylortostandard(taylor(ifade2,zt,0,mm)),zt,mm);
write(sonuc);
return(sonuc);
end;

PROCEDURE SERINONP(ifade);
% Works in general, requires that Taylor be loadad within calling routine;
begin scalar ifade2,sonuc;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
sonuc:=sub(zt=1,taylortostandard(taylor(ifade2,zt,0,m)));
```

```
write(sonuc);
return(sonuc);
end;

procedure coef(ifade1,ifade2);

comment this routine isolates the coefficient of a nonsimple kernel
        in an expression;

begin scalar expr2,expr2,res;
   expr2:=ifade2;
   match expr2 = sts;
   expr1:=ifade1;
   res:=coeffn(expr1,sts,1);
   clear expr2;
return res;
end;

procedure mod(u,v);
comment this bypasses a REDUCE 3.6 bug;
remainder(u,v);

procedure dumpresults();
comment dumps the n'th order normal form and transformation;
begin
operator dydx;
on rationalize;
out "results";
off nat;
write "%%%%%%%%%%%%%%%  transformations %%%%%%%%%%%%%%%";
for i:=1:n do write trf(i):=trf(i);
write "%%%%%%%%%%%%%%%  differential equations %%%%%%%%%%%%%%%";
write NEWDES;
shut "results";
comment terminal dump;
off rationalize;
on nat;
on rat;
write "%%%%%%%%%%%%%%%  transformations %%%%%%%%%%%%%%%";
for i:=1:n do write trf(i):=y(i)+trf(i);
write "%%%%%%%%%%%%%%%  differential equations %%%%%%%%%%%%%%%";
write NEWDES;

off rat;
end;

PROCEDURE NFRES();
```

```
BEGIN

COMMENT NEW VARIABLE NAMES?;
WRITE "DO YOU WANT TO ENTER NEW VARIABLE NAMES (Y/N)?";

OPERATOR X,Y;
TEST := XREAD (1);
IF TEST = N THEN GO TO JUMP;
WRITE "HOW MANY COMPLEX CONJUGATE PAIRS";
NCC:= XREAD (1);

FOR I:=1 STEP 2 UNTIL 2*NCC DO BEGIN
WRITE "SYMBOL FOR OLD X(",I,")";
X(I) := XREAD (1);
X(I+1):=MKID(X(I),B);
WRITE "SYMBOL FOR NEW X(",I,")";
Y(I) := XREAD (1);
Y(I+1):=MKID(Y(I),B);
WRITE " X(",I+1,") IS SET TO ",X(I+1);
WRITE " Y(",I+1,") IS SET TO ", Y(I+1);
END;

WRITE "HOW MANY ADDITIONAL EQS ARE THERE ? ";
NADD := XREAD (1);
N:=NADD+NCC+NCC;

ARRAY TRF(n);
IF NADD > 0 THEN BEGIN
WRITE "FOR ADDITIONAL EQUATIONS";
FOR I:=2*NCC+1:N DO BEGIN
WRITE " SYMBOL FOR OLD X(",I,")";
X(I) := XREAD (1);
WRITE "SYMBOL FOR NEW X(",I,")";
Y(I) := XREAD (1);
END;
END;

FOR I:=1:N DO DEPEND X(I),TT;
FOR I:=1:N DO DEPEND Y(I),TT;

JUMP:

COMMENT  NEW D.E.'S?;
WRITE "DO YOU WANT TO ENTER NEW D.E.'S (Y/N)?";
TEST:=XREAD(1);
IF TEST = N THEN GO TO LOOP;
ARRAY RH(N);
```

```
sublst:=FOR I:=1 STEP 2 UNTIL 2*NCC join {x(i)=x(i+1),x(i+1)=x(i)};
invsublst:=FOR I:=1 STEP 2 UNTIL 2*NCC join {y(i)=y(i+1),y(i+1)=y(i)};
alfalst:=for i:=1:100 join {mkid(al,i)=mkid(alc,i)};
alfaclst:=for i:=1:100 join {mkid(alc,i)=mkid(al,i)};

FOR II:=1 STEP 2 UNTIL 2*NCC DO BEGIN
WRITE "ENTER RHS OF EQ. NO.",II,",D",X(II),"/DT =";
RH(II):=XREAD(1);
WRITE "D",X(II),"/DT =",RH(II);
RH(II+1):=sub(i=-i,sublst,RH(II));
WRITE " EQ. NO.", II+1, " SET AS ";
WRITE "D",X(II+1),"/DT =",RH(II+1);
end;

IF NADD > 0 THEN BEGIN
FOR II:=2*NCC+1:N DO BEGIN
WRITE "ENTER RHS OF EQ. NO.",II,",D",X(II),"/DT =";
RH(II):=XREAD(1);
WRITE "D",X(II),"/DT =",RH(II);
END;
END;

LOOP:

COMMENT NEAR-IDENTITY TRANSFORMATION;
WRITE "INPUT NEAR-IDENTITY TRANSFORMATION";
WRITE "(USE PREV FOR PREVIOUS TRANSFORMATION)";
WRITE "(USE PREVCC FOR CX. CONJ. OF PREVIOUS VARIABLE)";

FOR II:=1:N DO BEGIN
ROW:=II;
PREV :=TRF(II);
WRITE X(II),"=",Y(II),"+ ?";
TRF(II) :=XREAD (1);
WRITE X(II),"=",Y(II)+TRF(II);
TRFI:=TRF(II);
PREVCC:=sub(i=-i,sublst,invsublst,alfalst,alfaclst,TRFI);
END;
COMMENT INPUT TRUNCATION ORDER;
TRANS := for i:=1:n collect X(I)=Y(I)+TRF(I);
WRITE "ENTER TRUNCATION ORDER (HIGHEST ORDER TERMS TO BE KEPT)";
M := XREAD(1);
Matrix newrhs(n,1);
detdf();
FOR I:=1:N DO WRITE PART(NEWDES,I);

COMMENT ENTER ANOTHER TRANSFORMATION?;
WRITE "DO YOU WANT TO ENTER ANOTHER TRANSFORMATION (Y/N)";
```

```
BRANCH:=XREAD(1);
IF BRANCH = Y THEN GO to LOOP;
write NEWDES;
END;

procedure detdf;
begin scalar i,j,sublist,invsublist;
operator ww;
Matrix Jacob2(n,n),eye(n,n);
for i:=1:n do eye(i,i):=1;
invsublist:=for i:=1:n collect ww(i)=y(i);
sublist:=for i:=1:n collect y(i)=ww(i);
weight(ww(1)=1,ww(2)=1,ww(3)=1,ww(4)=1,ww(5)=1,ww(6)=1,ww(7)=1,ww(8)=1);
wtlevel(m);
for i:=1:n do for j:=1:n do
JACOB2(I,J):=sub(sublist,DF(TRF(I),Y(J)));
write "Jacobian OK";

COMMENT TRANSFORM THE D.E.'S;
MATRIX TEMP2(n,1),temp3(n,n);
for i:=1:n do TEMP2(i,1) :=sub(sublist,sub(TRANS,rh(i)));
write "transformation substituted";

COMMENT SOLVE FOR THE TRANSFORMED DERIVATIVES;
temp3:=eye;
for i:= 1:m-1 do
begin
temp3:=eye- Jacob2*temp3;
end;
temp2:=temp3*temp2;
write "Jacobian inverted";
COMMENT TAYLOR EXPAND THE RESULTING EQS;
write "begin Taylor";
for i:=1:N DO NEWRHS(I,1):=sub(invsublist,temp2(i,1));
NEWDES:=for i:=1:n collect DF(Y(I),TT)=NEWRHS(I,1);
clear(ww(1),ww(2),ww(3),ww(4),ww(5),ww(6),ww(7),ww(8));
clear ww;
end;

procedure hopf(m1);
begin scalar i,ii,neqtns,expression,sublist,sublist2,tryagain,st4,st5,trm;
array rhold(n),rrh(n);
comment eqtns are the bona-fide processed equations;
comment eqtnsdum are the equations stemming from complex conjugates;

out"sonuclar";
eqtns:={};
eqtnsdum:={};
```

```
sublist:=for i:=1:n collect y(i)=wtt**(for ii:=0:(i-1) sum m1**ii);
sublist2:=for i:=1:n collect y(i)=wtt*y(i);

comment separate complex conjugate pairs, flag if this has been done;
comment neq_cc labels the index, so that we can generate the complex conjugate;
comment         solution;

cc_flag:=0;
neq_cc:={};

for i:=1:n do BEGIN
  expression:=sub(sublist2,newrhs(i,1));
  expression:=expression-coeffn(expression,wtt,0)-wtt*coeffn(expression,wtt,1);
  expression:=sub(sublist,expression);
  neqtns:=coeff(expression,wtt);
    if ((i<=(2*ncc)) and (mod(i,2)=0))
          then <<
     eqtnsdum:=append(neqtns,eqtnsdum);
     cc_flag:=1;
     neq_cc:=append({i},neq_cc)>>
          else
     eqtns:=append(neqtns,eqtns);
END;
write eqtns:=eqtns;
write slist:=slist;
%normal case;
solution:=solve(eqtns,slist);
write solution;
if(solution neq {}) then goto sonuc;
%first eliminate superfluous equations;
eqtns2:={};
for i:=1:length(eqtns) do begin
      trm:=Part(eqtns,i);
      if freeof(trm,cc) then 1 else eqtns2:=append(eqtns2,{trm});
end;
eqtns:=eqtns2;
%next limit variables to those in the equations;
dm:=length(eqtns);
%slistnew:=slist;
slist:={};
for i:=1:length(eqtns) do begin
    s(Part(eqtns,i));
    slist:=append(mvarlst,slist);
end;

write eqtns;
write slist;
%try again;
```

```
solution:=solve(eqtns,slist);

if (solution neq {}) then goto sonuc2;

%now resonance is obvious, eliminate as many eqtns as the rank permits;

nodk:=length(eqtns);
novr:=length(slist);
sze:=if novr<nodk then novr else nodk;
matrix jac(sze,sze);
for i:=1:sze do for j:=1:sze do
jac(i,j):=Coeffn(Part(eqtns,i),Part(slist,j),1);
jac;
rnk:=rank(jac);
write "nodk=",nodk," novr=",novr," rank= ",rnk;
matrix jac2(rnk,rnk);
for i:=1:rnk do for j:=1:rnk do
jac2(i,j):=Coeffn(Part(eqtns,i),Part(slist,j),1);
jac2;
if(det(jac2) neq 0) then goto bulundu;
for i:=1:rnk do begin
    eqtns:=append(rest(eqtns),{first(eqtns)});
    slist:=append(rest(slist),{first(slist)});
    for i:=1:rnk do for j:=1:rnk do
    jac2(i,j):=Coeffn(Part(eqtns,i),Part(slist,j),1);
    deter:=det(jac2);
    write deter;
    if(deter neq 0) then goto bulundu;
end;
bulundu:
   rdeqtns:={};
   rdslist:={};
   for i:=1:rnk do begin
      rdeqtns:=append(rdeqtns,{Part(eqtns,i)});
      rdslist:=append(rdslist,{Part(slist,i)});
   end;
   solution:=solve(rdeqtns,rdslist);
write "rezonans";
write solution;
sonuc2:

sublist:={};
for i:=1:length(slistorig) do begin
  trm:=Part(slistorig,i);
  trminset:=0;
  ccvar:=0;
  for j:=1:length(neq_cc) do if (Part(trm,1) eq Part(neq_cc,j)) then ccvar:=1;
```

```
  for j:=1:length(slist) do if (trm eq Part(slist,j)) then trminset:=1;

  if ((trminset=0) and (ccvar=0))  then begin
     solution:=append(solution,{trm=MKID(AL,NOOFALFAS)});
     write  trm," = ",MKID(AL,NOOFALFAS);
     NOOFALFAS:=NOOFALFAS+1;
  end;

end;
sonuc:
comment these lines were intended for automatic handling
of arbitrary terms, currently under development;
solution;

shut"sonuclar";
end;

operator cc;

procedure gen(m1);
begin scalar s,i1,i2,i3,i4,i5,i6,i7,i8,i9,i10,i11,j;
comment iord is the order, n is number of variables, m is degree
This generates a homogeneous polynomial of order m in n variables
with arbitrary coefficients for use in generating the arbitrary
near identity transformation;

if(not(numberp(iord)) or mod(iord,n)=0) then
<<iord:=1;slistold:={}>>
else <<iord:=iord+1;slistold:=slist>>;
if n=2 then begin
 s:=for i1:=0:m1 sum cc(iord,i1,m1-i1)*y(1)**i1*y(2)**(m1-i1);
 slist:=for i1:=0:m1 collect cc(iord,i1,m1-i1);
 end;
if n=3 then begin
 s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum cc(iord,i1,i2,m1-i1-i2)
   *y(1)**i1*y(2)**i2*y(3)**(m1-i1-i2);
 slist:=for i1:=0:m1 join for i2:=0:m1-i1 collect cc(iord,i1,i2,m1-i1-i2);
 end;
if n=4 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum for i3:=0:m1-i1-i2 sum
  cc(iord,i1,i2,i3,m1-i1-i2-i3)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**(m1-i1-i2-i3);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join for i3:=0:m1-i1-i2 collect
  cc(iord,i1,i2,i3,m1-i1-i2-i3);
end;

if n=5 then begin
```

```
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
  cc(iord,i1,i2,i3,i4,m1-i1-i2-i3-i4)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**(m1-i1-i2-i3-i4);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 collect
  cc(iord,i1,i2,i3,i4,m1-i1-i2-i3-i4);
end;

if n=6 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
  cc(iord,i1,i2,i3,i4,i5,m1-i1-i2-i3-i4-i5)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**(m1-i1-i2-i3-i4-i5);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 collect
cc(iord,i1,i2,i3,i4,i5,m1-i1-i2-i3-i4-i5);
end;

if n=7 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
for i6:=0:m1-i1-i2-i3-i4-i5 sum
  cc(iord,i1,i2,i3,i4,i5,i6,m1-i1-i2-i3-i4-i5-i6)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**i6*y(7)**(m1-i1-i2-i3-i4-i5-i6);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 join
for i6:=0:m1-i1-i2-i3-i4-i5 collect
cc(iord,i1,i2,i3,i4,i5,i6,m1-i1-i2-i3-i4-i5-i6);
end;

if n=8 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
for i6:=0:m1-i1-i2-i3-i4-i5 sum
for i7:=0:m1-i1-i2-i3-i4-i5-i6 sum
  cc(iord,i1,i2,i3,i4,i5,i6,i7,m1-i1-i2-i3-i4-i5-i6-i7)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**i6*y(7)**i7*y(8)**(m1-i1-i2-i3-i4-i5-i6-i7);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
```

```
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 join
for i6:=0:m1-i1-i2-i3-i4-i5 join
for i7:=0:m1-i1-i2-i3-i4-i5-i6 collect
cc(iord,i1,i2,i3,i4,i5,i6,i7,m1-i1-i2-i3-i4-i5-i6-i7);
end;

if n=9 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
for i6:=0:m1-i1-i2-i3-i4-i5 sum
for i7:=0:m1-i1-i2-i3-i4-i5-i6 sum
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7 sum
  cc(iord,i1,i2,i3,i4,i5,i6,i7,i8,m1-i1-i2-i3-i4-i5-i6-i7-i8)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**i6*y(7)**i7*y(8)**i8*y(9)**(m1-i1-i2-i3-i4-i5-i6-i7-i8);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 join
for i6:=0:m1-i1-i2-i3-i4-i5 join
for i7:=0:m1-i1-i2-i3-i4-i5-i6 join
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7 collect
cc(iord,i1,i2,i3,i4,i5,i6,i7,i8,m1-i1-i2-i3-i4-i5-i6-i7-i8);
end;

if n=10 then begin
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
for i6:=0:m1-i1-i2-i3-i4-i5 sum
for i7:=0:m1-i1-i2-i3-i4-i5-i6 sum
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7 sum
for i9:=0:m1-i1-i2-i3-i4-i5-i6-i7-i8 sum
  cc(iord,i1,i2,i3,i4,i5,i6,i7,i8,i9,m1-i1-i2-i3-i4-i5-i6-i7-i8-i9)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**i6*y(7)**i7*y(8)**i8*y(9)**i9*y(10)*(m1-i1-i2-i3-i4-i5-i6-i7-i8-i9);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 join
for i6:=0:m1-i1-i2-i3-i4-i5 join
for i7:=0:m1-i1-i2-i3-i4-i5-i6 join
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7 join
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7-i8 collect
cc(iord,i1,i2,i3,i4,i5,i6,i7,i8,i9,m1-i1-i2-i3-i4-i5-i6-i7-i8-i9);
end;

if n=11 then begin
```

```
s:=for i1:=0:m1 sum for i2:=0:m1-i1 sum
for i3:=0:m1-i1-i2 sum for i4:=0:m1-i1-i2-i3 sum
for i5:=0:m1-i1-i2-i3-i4 sum
for i6:=0:m1-i1-i2-i3-i4-i5 sum
for i7:=0:m1-i1-i2-i3-i4-i5-i6 sum
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7 sum
for i9:=0:m1-i1-i2-i3-i4-i5-i6-i7-i8 sum
for i10:=0:m1-i1-i2-i3-i4-i5-i6-i7-i8-i9 sum
  cc(iord,i1,i2,i3,i4,i5,i6,i7,i8,i9,10,m1-i1-i2-i3-i4-i5-i6-i7-i8-i9-i10)
   *y(1)**i1*y(2)**i2*y(3)**i3*y(4)**i4
   *y(5)**i5*y(6)**i6*y(7)**i7*y(8)**i8*y(9)**i9*y(10)**i10*y(11)*(m1-i1-i2-i3-i4-i5-i6-i7-i8-i9-10);
slist:=for i1:=0:m1 join for i2:=0:m1-i1 join
for i3:=0:m1-i1-i2 join for i4:=0:m1-i1-i2-i3 join
for i5:=0:m1-i1-i2-i3-i4 join
for i6:=0:m1-i1-i2-i3-i4-i5 join
for i7:=0:m1-i1-i2-i3-i4-i5-i6 join
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7 join
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7-i8 join
for i8:=0:m1-i1-i2-i3-i4-i5-i6-i7-i8-i9 collect
cc(iord,i1,i2,i3,i4,i5,i6,i7,i8,i9,i10,m1-i1-i2-i3-i4-i5-i6-i7-i8-i9-i10);
end;

slist:=append(slistold,slist);
slistorig:=slist;
return(s);
end;

NOOFALFAS:=1;
END;
```

APPENDIX D: NULLSPACE CALCULATION IN RESONANT CASES

```
COMMENT NFRNULL(), NORMAL FORM TRANSFORMATIONS
        IN CASE OF RESONANCE. EXTENDS THE REGULAR NF UTILITY
        GIVES BOTH THE NULLSPACE IN ORTHONORMAL FORM AND THE
        REMOVABLE VECTOR SPACE;

PROCEDURE SERI(ifade);
% works for polynomial cases, otherwise use SERINONP which is slower;
% expands n variable function of variables y(k) up to order m.
begin scalar ifade2,sonuc,set3,set4,j;
on div;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
set3:=coeff(ifade2,zt);
sonuc:=for j:=1:m+1 sum Part(set3,j);
write(sonuc);
return(sonuc);
end;

PROCEDURE SERICN(ifade,mm);
% works for polynomial cases, otherwise use SERINONPCN which is slower;
% expands n variable function of variables y(k) giving terms of order m.;
begin scalar ifade2,sonuc,set4,j;
on div;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
sonuc:=coeffn(ifade2,zt,mm);
write(sonuc);
return(sonuc);
end;

PROCEDURE SERINONPCN(ifade,mm);
% Works in general, requires that Taylor be loadad within calling routine;
% Gives m'th order terms;
begin scalar ifade2,sonuc;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
sonuc:=coeffn(taylortostandard(taylor(ifade2,zt,0,mm)),zt,mm);
write(sonuc);
return(sonuc);
end;

PROCEDURE SERINONP(ifade);
```

```
% Works in general, requires that Taylor be loadad within calling routine;
begin scalar ifade2,sonuc;
set4:=for j:=1:n collect y(j)=zt*y(j);
ifade2:=sub(set4,ifade);
sonuc:=sub(zt=1,taylortostandard(taylor(ifade2,zt,0,m)));
write(sonuc);
return(sonuc);
end;

PROCEDURE NFRNULL();

BEGIN

operator poli;
COMMENT NEW VARIABLE NAMES?;
WRITE "DO YOU WANT TO ENTER NEW VARIABLE NAMES (Y/N)? N MEANS VARS ARE IN Y(I)";

OPERATOR Y;
TEST := XREAD (1);
IF TEST = N THEN GO TO JUMP;
WRITE "HOW MANY EQS";
N := XREAD (1);

FOR I:=1:N DO BEGIN
WRITE "SYMBOL FOR NEW X(",I,")";
Y(I) := XREAD (1);
END;

FOR I:=1:N DO DEPEND Y(I),TT;

JUMP:

COMMENT  NEW D.E.'S?;
WRITE "DO YOU WANT TO ENTER NEW D.E.'S (Y/N)? IF NO, DE'S ARE IN RHS(I)";
TEST:=XREAD(1);
IF TEST = N THEN GO TO LOOP;
ARRAY RH(N);
FOR I:=1:N DO BEGIN
WRITE "ENTER RHS OF EQ. NO.",I,",D",Y(I),"/DT =";
RH(I):=XREAD(1);
WRITE "D",Y(I),"/DT =",RH(I);
end;

LOOP:

COMMENT EVALUATE LINEAR PART J;

MATRIX JJ(N,N);
```

```
st4:=for j:=1:n collect y(j)=0;
FOR I:=1:N DO FOR J:=1:N DO JJ(I,J):=SUB(ST4,DF(RH(I),Y(J)));

WRITE "ENTER TRUNCATION ORDER (HIGHEST ORDER TERMS TO BE KEPT)";
M := XREAD(1);

matrix xvec(n,1);
for i:=1:n do xvec(i,1):=y(i);
for m1:=2:m do begin

nt:=genpoli(m1);

matrix nbase(n,n*nt);
comment first index is the 3nt basis vectors,
        second index is the n components;
icompt:=0;
for i:=1:n do begin
for j:=1:nt do begin

out"monomial.res";
matrix vec(n,1);
vec(i,1):=poli(j);
write " vec = ";
write vec:=vec;

matrix lierslt(n,1);

matrix derivs(n,n);

for ii:=1:n do for ij:=1:n do derivs(ii,ij):=df(vec(ii,1),y(ij));
lierslt:=jj*vec-derivs*jj*xvec;

icompt:=icompt+1;

write" result = ";
write lierslt:=lierslt;

for ii:=1:n do nbase(ii,icompt):=lierslt(ii,1);

end;
end;

 matrix liebraket(n*nt,n*nt);
 for ii:=1:n*nt do begin
 for ij:=1:n do begin
 for ik:=1:nt do begin
 liebraket(nt*(ij-1)+ik,ii):=coef(nbase(ij,ii),poli(ik));
```

```
 end;
 end;
 end;

matrix rhand(n*nt,1);
for ij:=1:n do begin
for ik:=1:nt do begin
rhand(nt*(ij-1)+ik,1):=coef(sericn(RH(ij),m1),poli(ik));
end;
end;

write "liebraket=", liebraket;

bosuzay:=nullspace(tp(liebraket));

write "nullspace = ", bosuzay;
if bosuzay = {} then write " Complimentary space empty "
   else begin
   nvect:=length(bosuzay);
   for ii:=1:nvect do
       begin
       matrix nulvect(n*nt,1);
       nulvect:=Part(bosuzay,ii);
       matrix nullbasis(n,1);
       for ij:=1:n*nt do
           begin
           if nulvect(ij,1) neq 0 then begin
           inr:=fix((ij-1)/nt);
           inp:=ij-nt*inr;
           inr:=inr+1;
           nullbasis(inr,1):=nullbasis(inr,1)+poli(inp)*nulvect(ij,1);
           end;
       end;
       write " Complimentary space basis ",ii;
       write nullbasis:=nullbasis;
       clear nullbasis;
   end;
end;

veclst:={};
matrix bos(n*nt,1);
array nnzo(length(bosuzay));
for ii1:=1:length(bosuzay) do begin
bos:=Part(bosuzay,ii1);
nnzero:=0;
nnelt:=0;
for ii2:=1:n*nt do if bos(ii2,1) neq 0 then begin
      nnzero:=nnzero+1;
```

```
      nnelt:=nnelt+bos(ii2,1)**2;
      end;
nnzo(ii1):=nnzero+nnelt/10000;
listi:=for ii2:=1:n*nt collect bos(ii2,1);
veclst:=append(veclst,{listi});
end;

write "veclst = ", veclst;

array nnzsrt(length(bosuzay));
for ii1:=1:length(bosuzay) do nnzsrt(ii1):=ii1;

for ii1:=1:length(bosuzay) do begin
for ii2:=ii1+1:length(bosuzay) do
if nnzo(ii2) < nnzo(ii1) then begin
gec1:=nnzo(ii1);
nnzo(ii1):=nnzo(ii2);
nnzo(ii2):=gec1;

gec1:=nnzsrt(ii1);
nnzsrt(ii1):=nnzsrt(ii2);
nnzsrt(ii2):=gec1;
end;
end;

veclst2:={};
for ii1:=1:length(bosuzay) do
veclst2:=append(veclst2,{Part(veclst,nnzsrt(ii1))});

comment veclst2 contains vectors in the nullspace in sorted form.
Now find first single element orthogonal vectors to it;

operator sirano;
nsingle:=0;
for ii1:=1:n*nt do begin
    for jj1:=1:length(bosuzay) do begin
    nulvct:=part(veclst2,jj1);
    if (Part(nulvct,ii1) neq 0 ) then goto olmadi;
    end;
nsingle:=nsingle+1;
sirano(nsingle):=ii1;
listi:=for ii2:=1:n*nt collect 0;
listi:=(part(listi,ii1):=1);
veclst2:=append(veclst2,{listi});
%  write ii1," veclst2 ",veclst2;
% goto oldu;
olmadi: %write ii1, " is dependent on nullspace";
% oldu:
```

```
 end;

 comment single element vectors complete, fill up rest with two element
 vectors;
 for ii1:=1:n*nt-length(bosuzay) do begin;
  for jj2:=1:nsingle do if(ii1 eq sirano(jj2)) then goto atla;
  listi:=for ii2:=1:n*nt collect 0;
  listi:=(part(listi,ii1):=1);
  % ********* This commented list is adding a second element;
  % uncomment if single element fails;
  %ii1lg:=ii1+length(bosuzay);
  %listi:=(part(listi,ii1lg):=-1);
  %even a third element can be added here;
  %if((nsingle>0) and (ii1lg+nsingle leq n*nt)) then
  %listi:=(part(listi,ii1lg+nsingle):=ii1+2);
  veclst2:=append(veclst2,{listi});
  % write veclst2:=veclst2;
  atla:
 end;

% write veclst2:=veclst2;
out t;
load_package linalg;
off nat;
out temp;
write"bosorth:=gram_schmidt(",veclst2,");";
shut temp;
out"monomial.res";
on nat;
in temp;

%*************************************************************;
% OBTAIN ORTHONORMALIZED NULL AND REDUCIBLE SPACES;
%*************************************************************;
nvect:=length(bosuzay);
nspace:=n*nt;
   matrix scalarprod(1,1),nullcont(nspace,1),remcont(nspace,1);
   for ii:=1:nspace do
       begin
       matrix nulvect(n*nt,1);
       for iij:=1:nspace do nulvect(iij,1):=Part(Part(bosorth,ii),iij);
%************************************************************************;
%* Here remove the common factor for normalization, just report factor  *;
%************************************************************************;
       minmum:=1.1;
            on rounded;
            array numvect(nspace);
            for iij:=1:nspace do numvect(iij):=abs(nulvect(iij,1));
```

```
        for iij:=1:nspace do
        if((numvect(iij) leq minmum) and
             (numvect(iij) > 0)) then
            <<minmum:=numvect(iij);ormin:=iij>>;
         off rounded;
         minmum:=nulvect(ormin,1);
         clear numvect;

    scalarprod:=tp(rhand)*nulvect;
    if(ii leq nvect) then nullcont:=nullcont+scalarprod(1,1)*
           nulvect else remcont:=remcont+scalarprod(1,1)*nulvect;
    for iij:=1:nspace do
           nulvect(iij,1):=nulvect(iij,1)/minmum;
        matrix nullbasis(n,1);
    for ij:=1:n*nt do
        begin
        if nulvect(ij,1) neq 0 then begin
        inr:=fix((ij-1)/nt);
        inp:=ij-nt*inr;
        inr:=inr+1;
        nullbasis(inr,1):=nullbasis(inr,1)+poli(inp)*nulvect(ij,1);
        end;
    end;
    if (ii leq nvect) then write " Complimentary space basis ",ii
                    else write " Reducible space basis ",ii-nvect;
    write minmum:=minmum;
    write "right hand cont = ",scalarprod(1,1);
    write nullbasis:=nullbasis;
    clear nullbasis;
end;

        matrix nullpart(n,1);
    for ij:=1:n*nt do
        begin
        if nullcont(ij,1) neq 0 then begin
        inr:=fix((ij-1)/nt);
        inp:=ij-nt*inr;
        inr:=inr+1;
        nullpart(inr,1):=nullpart(inr,1)+poli(inp)*nullcont(ij,1);
        end;
    end;

     matrix rempart(n,1);
    for ij:=1:n*nt do
        begin
        if remcont(ij,1) neq 0 then begin
        inr:=fix((ij-1)/nt);
        inp:=ij-nt*inr;
```

```
            inr:=inr+1;
            rempart(inr,1):=rempart(inr,1)+poli(inp)*remcont(ij,1);
            end;
        end;

write nullpart:=nullpart;
write rempart:=rempart;
write nullpart+rempart;

end;
shut"monomial.res";
end;

comment this routine isolates the coefficient of a nonsimple kernel
        in an expression;

procedure coef(ifade1,ifade2);
begin scalar expr2,expr2,res;
   expr2:=ifade2;
   match expr2 = sts;
   expr1:=ifade1;
   res:=coeffn(expr1,sts,1);
   clear expr2;
return res;
end;

comment this routine returns the homogeneous monomials in poli
and the number of such terms as genpoli;

procedure genpoli(m1);
comment iord is the order, n is number of variables, m1 is degree;
begin scalar s,i1,i2,i3,i4,i5,i6,i7,i8,i9,j;
j:=0;
if n=2 then begin
 for i1:=0:m1 do
 <<j:=j+1;poli(j):=y(2)**i1*y(1)**(m1-i1)>>;
 end;
if n=3 then begin
 for i1:=0:m1 do for i2:=0:m1-i1 do
    <<j:=j+1;poli(j):=y(3)**i1*y(2)**i2*y(1)**(m1-i1-i2)>>;
 end;
if n=4 then begin
  for i1:=0:m1 do for i2:=0:m1-i1 do for i3:=0:m1-i1-i2 do
  <<j:=j+1;poli(j):=y(4)**i1*y(3)**i2*y(2)**i3*y(1)**(m1-i1-i2-i3)>>;
end;

if n=5 then begin
 for i1:=0:m1 do for i2:=0:m1-i1 do
```

```
 for i3:=0:m1-i1-i2 do for i4:=0:m1-i1-i2-i3 do
    <<j:=j+1;poli(j):=y(5)**i1*y(4)**i2*y(3)**i3*y(2)**i4
    *y(1)**(m1-i1-i2-i3-i4)>>;
end;

if n=6 then begin
 for i1:=0:m1 do for i2:=0:m1-i1 do
 for i3:=0:m1-i1-i2 do for i4:=0:m1-i1-i2-i3 do
 for i5:=0:m1-i1-i2-i3-i4 do
 <<j:=j+1;poli(j):=y(6)**i1*y(5)**i2*y(4)**i3*y(3)**i4
   *y(2)**i5*y(1)**(m1-i1-i2-i3-i4-i5)>>;
end;

if n=7 then begin
 for i1:=0:m1 do for i2:=0:m1-i1 do
 for i3:=0:m1-i1-i2 do for i4:=0:m1-i1-i2-i3 do
 for i5:=0:m1-i1-i2-i3-i4 do
 for i6:=0:m1-i1-i2-i3-i4-i5 do
 <<j:=j+1;poli(j):=y(7)**i1*y(6)**i2*y(5)**i3*y(4)**i4
   *y(3)**i5*y(2)**i6*y(1)**(m1-i1-i2-i3-i4-i5-i6)>>;
end;

if n=8 then begin
 for i1:=0:m1 do for i2:=0:m1-i1 do
 for i3:=0:m1-i1-i2 do for i4:=0:m1-i1-i2-i3 do
 for i5:=0:m1-i1-i2-i3-i4 do
 for i6:=0:m1-i1-i2-i3-i4-i5 do
 for i7:=0:m1-i1-i2-i3-i4-i5-i6 do
 <<j:=j+1;poli(j):=y(8)**i1*y(7)**i2*y(6)**i3*y(5)**i4
   *y(4)**i5*y(3)**i6*y(2)**i7*y(1)**(m1-i1-i2-i3-i4-i5-i6-i7)>>;
end;
return(j);
end;
;end;
```

APPENDIX E: NF EXPANSION OF SPROTT C UP TO THE FIFTH ORDER

Near identity transformations of the Sprott C system up to the fifth order are given below:

$$\Omega_2 = \frac{5}{9}\sqrt{2}\rho\, w \sin(\theta) - \frac{7}{36}\sqrt{2}\rho^2 (\cos(\theta))^2 + \frac{2}{9}\sqrt{2}\rho^2 - \frac{2}{9}\rho^2 \cos(\theta)\sin(\theta) + \frac{2}{9}\rho\, w \cos(\theta)$$

$$\Theta_2 = -\frac{4}{9}\rho\sqrt{2} w \cos(\theta) - \frac{2}{9}\sqrt{2}\rho^2 \cos(\theta)\sin(\theta) + \frac{2}{9}\rho^2 (\cos(\theta))^2 + \frac{1}{9}\rho^2 - \frac{2}{9}\rho\, w \sin(\theta)$$

$$W_2 = -\frac{1}{18}\sqrt{2}\rho^2 \cos(\theta)\sin(\theta) + \frac{1}{6}\rho\sqrt{2} w \cos(\theta) - \frac{1}{18}\rho^2 (\cos(\theta))^2$$

$$\Omega_3 = -\frac{7}{288}\sqrt{2}\rho^3 \sin(\theta) + \frac{5}{72}\sqrt{2}\rho^3 (\cos(\theta))^2 \sin(\theta) + \frac{40}{171}\sqrt{2}\rho^2 w (\cos(\theta))^2$$

$$-\frac{11}{288}\rho^3 \cos(\theta) + \frac{1}{36}\rho^3 (\cos(\theta))^3 - 1/3\,\rho\, w^2 \cos(\theta) - \frac{1}{171}\sqrt{2}\rho^2 w$$

$$-\frac{31}{171}\rho^2 w \cos(\theta)\sin(\theta)$$

$$\Theta_3 = -\frac{1}{864}\rho^3 \sin(\theta) + \frac{2}{27}\rho^3 (\cos(\theta))^2 \sin(\theta) - \frac{13}{216}\sqrt{2}\rho^3 (\cos(\theta))^3$$

$$+\frac{154}{513}\rho^2 w (\cos(\theta))^2 - \frac{134}{513}\rho^2 w + \frac{1}{9}\rho\sqrt{2} w^2 \cos(\theta) + \frac{11}{288}\sqrt{2}\rho^3 \cos(\theta)$$

$$+\frac{70}{513}\sqrt{2}\rho^2 w \cos(\theta)\sin(\theta) - \frac{1}{9}\rho\, w^2 \sin(\theta)$$

$$W_3 = -\frac{2}{27}\rho\sqrt{2} w^2 \cos(\theta) - \frac{8}{1539}\sqrt{2}\rho^3 (\cos(\theta))^3 + \frac{43}{324}\rho^2 w (\cos(\theta))^2$$

$$+\frac{245}{6156}\rho^3 (\cos(\theta))^2 \sin(\theta) + \frac{2}{27}\rho\, w^2 \sin(\theta) - \frac{43}{648}\rho^2 w - \frac{7}{1539}\rho^3 \sin(\theta)$$

$$-\frac{7}{162}\sqrt{2}\rho^2 w \cos(\theta)\sin(\theta) - \frac{83}{3078}\sqrt{2}\rho^3 \cos(\theta)$$

$$\Omega_4 = -\frac{5051}{184680}\rho^4(\cos(\theta))^3\sin(\theta) - \frac{23}{738720}\rho^4\cos(\theta)\sin(\theta) + \frac{106}{16929}\sqrt{2}\rho^2w^2(\cos(\theta))^2$$

$$+\frac{12221}{369360}\sqrt{2}\rho^4 - \frac{69727}{1477440}\sqrt{2}\rho^4(\cos(\theta))^2 + \frac{4961}{738720}\sqrt{2}\rho^4(\cos(\theta))^4$$

$$+\frac{2366}{16929}\rho^2w^2\cos(\theta)\sin(\theta) - \frac{1987}{33858}\sqrt{2}\rho^2w^2 - \frac{90133}{609444}\rho^3w(\cos(\theta))^3$$

$$+\frac{20}{459}\rho w^3\cos(\theta) + \frac{31862}{152361}\rho^3w\cos(\theta) - \frac{55}{1836}\sqrt{2}\rho w^3\sin(\theta)$$

$$-\frac{2971}{152361}\sqrt{2}\rho^3w(\cos(\theta))^2\sin(\theta) - \frac{40913}{541728}\sqrt{2}\rho^3w\sin(\theta)$$

$$\Theta_4 = \frac{14}{459}\rho w^3\sin(\theta) + \frac{13}{1836}\rho\sqrt{2}w^3\cos(\theta) - \frac{623}{5643}\rho^2w^2(\cos(\theta))^2 + \frac{625}{5643}\rho^2w^2$$

$$+\frac{83579}{2437776}\rho^3w\sin(\theta) - \frac{25501}{152361}\rho^3w(\cos(\theta))^2\sin(\theta) - \frac{11323}{369360}\rho^4(\cos(\theta))^2$$

$$+\frac{808}{23085}\rho^4(\cos(\theta))^4 + \frac{7511}{369360}\rho^4 - \frac{317}{92340}\sqrt{2}\rho^4(\cos(\theta))^3\sin(\theta)$$

$$-\frac{11081}{369360}\sqrt{2}\rho^4\cos(\theta)\sin(\theta) - \frac{24413}{609444}\sqrt{2}\rho^3w(\cos(\theta))^3 + \frac{345467\sqrt{2}\rho^3w\cos(\theta)}{2437776}$$

$$W_4 = \frac{12221}{369360}\sqrt{2}\rho^4 - \frac{69727}{1477440}\sqrt{2}\rho^4(\cos(\theta))^2 + \frac{4961}{738720}\sqrt{2}\rho^4(\cos(\theta))^4$$

$$+\frac{106}{16929}\sqrt{2}\rho^2w^2(\cos(\theta))^2 - \frac{5051}{184680}\rho^4(\cos(\theta))^3\sin(\theta)$$

$$-\frac{23}{738720}\rho^4\cos(\theta)\sin(\theta) - \frac{90133}{609444}\rho^3w(\cos(\theta))^3 + \frac{31862}{152361}\rho^3w\cos(\theta)$$

$$+\frac{2366}{16929}\rho^2w^2\cos(\theta)\sin(\theta) - \frac{1987}{33858}\sqrt{2}\rho^2w^2 + \frac{20}{459}\rho w^3\cos(\theta)$$

$$-\frac{55}{1836}\sqrt{2}\rho w^3\sin(\theta) - \frac{2971}{152361}\sqrt{2}\rho^3w(\cos(\theta))^2\sin(\theta) - \frac{40913\sqrt{2}\rho^3w\sin(\theta)}{541728}$$

REFERENCES

1. Rosenstein, M. T., J. J. Collins and C. J. De Luca, "A Practical Method for Calculating Largest Liapunov Exponents from Small Data Sets", *Physica D*, Vol. 65, pp. 117-134, 1993.

2. Kantz, H., "A Robust Method to Estimate the Maximal Liapunov Exponent of a Time Series", *Physiscs Letters A*, Vol. 185, pp. 77-87, 1994.

3. Amitrano, C. and R. S. Berry, "Probability Distributions of Local Liapunov Exponents for Hamiltonian Systems", *Physical Review E*, Vol. 47, pp. 3158-3173, 1993.

4. Geist, K., U. Parlitz and W. Lauterborn, "Comparison of Different Methods for Computing Liapunov Exponents", *Progress of Theoretical Physics*, Vol. 83, pp. 875-893, 1990.

5. Hacınlıyan, A., N. Z. Perdahçı, "Liapunov Üstellerinin hesaplanmasında tümleyici seçimi", *Teorik ve Uygulamalı Mekanik Türk Milli Komitesi 12th Ulusal Mekanik Kongresi Bildiriler Kitabı* , Selçuk-Konya, 10-14 September 2001, pp. 473-481, Cenkler Matbaası , İstanbul, 2002.

6. Kaplan, J. and J. Yorke, *Functional Differential Equations and Approximation of Fixed Points*, Springer-Verlag, Heilderberg, 1979.

7. Haken, H., "At Least One Liapunov Exponent Vanishes If the Trajectory of an Attractor Does Not Contain a Fixed Point", *Physiscs Letters A*, Vol. 94, pp. 71-72, 1983.

8. Roepstorff, G., "Liapunov Exponents of Simple Smooth Flows", *Journal of Mathematical Physics*, Vol. 34, pp. 5221-5237, 1993.

9. Birol, İ. and A. Hacınlıyan, "Signs and Approximate Magnitudes of Liapunov

Exponents in Continuous Time Dynamical Systems", *Journal of Mathematical Physics*, Vol. 38, pp. 4594-4605, 1997.

10. Umur A., İ. Birol and A. Hacınlıyan, "Chaos Mechanisms of Third Order Continuous Time Dynamical Systems", *Balkan Physics Letters*, Vol. 5, pp. 188-197, 1997.

11. Sprott, J. C., "Some Simple Chaotic Flows", *Physical Review E*, Vol. 50, pp. R647-R650, 1994.

12. Cicogna G., "Symmetries of Dynamical Systems and Convergent Normal Forms", *Journal of Physics A: Math. Gen.*, Vol. 28, pp. L179-L182, 1995.

13. Warner, W. H. , P. R. Sethna and J. P. Sethna, "A Generalization of the Theory of Normal Forms", *Journal of Nonlinear Science*, Vol. 6, pp. 499-506, 1996.

14. Bi, Q. S. and P. Yu, "Symbolic Software Development for Computing the Normal Form of Double Hopf Bifurcation", *Mathematical and Computer Modelling*, Vol. 29, pp. 49-70, 1999.

15. Kahn, P. B. and Y. Zarmi, "Nonlinear Dynamics: A Tutorial on the Theory of Normal Forms", *American Journal of Physics*, Vol. 10, pp. 907-919, 2000.

16. Wiggins, S., *Introduction to Applied Nonlinear Dynamical Systems and Chaos*, Springer-Verlag, New York, 1990.

17. Arnold, V. I., *Geometrical Methods in the Theory of Ordinary Differential Equations*, Springer-Verlag, New York, 1988.

18. Bruno, A. D., *Local Methods in Nonlinear Differential Equations*, Springer-Verlag, Berlin, 1989.

19. Yu, P., "Computation of the Simplest Normal Forms with Perturbation Parameters Based on Lie Transform and Rescaling", *Journal of Computational and Applied*

Mathematics, Vol. 144, pp. 359-373, 2002.

20. Fu, J. H., "On Liapunov Stability and Normal Forms of Nonlinear Systems with a Nonsemisimple Critical Mode, Part I: Zero Eigenvalue", *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, Vol. 47, pp. 838-849, 2000, "On Liapunov Stability and Normal Forms of Nonlinear Systems with a Nonsemisimple Critical Mode, Part II: Imaginary Eigenvalues Pair", Vol. 47, pp. 850-859, 2000.

21. Kahn, P. B. and Y. Zarmi, *Nonlinear Dynamics, Exploration through Normal Forms*, John Wiley and Sons Inc., New York, 1998.

22. Guckenheimer, J. and P. Holmes, *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer-Verlag, Virginia, 1990.

23. Birol İ., A. Hacınlıyan and N. Z. Perdahçı, "Simulating a Dynamical System in the Presence of a 1/f noise", *International Journal of Engineering Sciences*, Vol. 36, pp. 265-271, 1998.

24. Umur, A., *Local Nonperturbative Methods for Algebraic Calculation of Liapunov Exponents*, M.S. Thesis, Boğaziçi University, 1997.

25. Birol, İ., *Signs and Magnitudes of Liapunov Exponents in Continuous Time Dynamical Systems*, Ph.D. Thesis, Boğaziçi University, 1997.

26. Steeb, W. H., "Computer Algebra and its Applications in Physics", *International Journal of Modern Physics C*, Vol. 3, pp. 1333-, 1992.

27. Hacınlıyan A., N. Z. Perdahçı and F. G. Fırat Acar, "Nonpolynomial Expansion Near Fixed Points", *Continuum Models and Discrete Systems (CMDS9), Proceedings of the 9th International Symphosium*, Maçka-İ stanbul, June 29 - July 3 1998, pp. 39-46, World Scientific Publishing Co., 1998.

28. Slichter, C. P., *Principles of Magnetic Resonance*, Harper-Row, New York, 1963.

29. Nishimura, D. G., *Principles of Magnetic Resonance Imaging*, Unpublished Lecture Notes, Stanford University, 1996.

30. Bihan, D. L. and A. Karni, "Applications of Magnetic Resonance Imaging to the Study of Human Brain Function", *Current Opinion in Neurobiology*, Vol. 5, pp. 231-237, 1995.

31. Kim, S., K. Ugurbil, "Functional Magnetic Resonance Imaging of the Human Brain", *Journal of Neuroscience Methods*, Vol. 74, pp. 229-243, 1997.

32. Bodner, M., T. Muftuler and O. Nalcioglu et al., "fMRI Study Relevant to the Mozart Effect Brain Areas Involving Spatial Temporal Reasoning", *Neurol Resonance*, Vol. 23, pp. 683-690, 2001.

33. Penny, W. D., J. Ashburner, S. Kiebel, R. Henson, D. E. Glaser, C. Phillips and K. Fristo, *Statistical Parametric Mapping: An Annotated Bibliography*, 2001, http://www.fil.ion.ucl.ac.uk/spm/bib.htm

34. Robert, C. W. and J. S. Hyde, "Software Tools for Analysis and Visualization of fMRI Data", *NMR in Biomedicine*, Vol. 10, pp. 171-178, 1997.

35. Nalcıoğlu, O., A. Hacınlıyan, N. Z. Perdahçı, "Fonksiyonel MR görüntülemede Fraktal Yöntemler", *Teorik ve Uygulamalı Mekanik Türk Milli Komitesi 11th Ulusal Mekanik Kongresi Bildiriler Kitabı* , Abant-Bolu, 6-10 September 1999, pp. 479-484, İTÜ İnşaat Fakültesi Matbaası, İstanbul, 2000.

36. Mandelbrot, B. B., *The fractal geometry of nature*, W. H. Freeman and Company, New York, 1983.

37. Feder, J., *Fractals*, Plenum Press, New York, 1988.

38. David, R. E., "Solutions and Linearization of the Nonlinear Dynamics of Radiation Damping", Notthingham University, preprint, 2001.

39. Khomeriki, R., "Route to and from the NMR Chaos in Diamagnets", *The European Physical Journal B*, Vol. 10, pp. 99-103, 1999.

40. Gerald, C. F. and Wheatley, P.O., *Applied Numerical Analysis*, Addison-Wesley, California, 1994.

41. Wolfram, S., *Mathematica, A System for Doing Mathematics by Computer*, Addison Wesley Publishing Company, Redwood City, 1996.

42. Char, B. W., K. O. Geddes, G.H. Gonnet, B. L. Leong, M. B. Monagan and S. M. Watt, *Maple V Language Reference Manual*, Springer-Verlag, New York, 1991.

43. Hearn, A. C., *REDUCE 3.7 User's and Contributed Packages Manual, Codemist Ltd.*, Bath, 1999.

44. Schelter, W. F., *Maxima Manual*, 1998, http://www.ma.utexas.edu

45. Free Software Foundation, Inc., *GNU General Public License*, 1991, http://www.gnu.org/copyleft/gpl.html.

46. Mikram, J. and F. Zinoun, "Normal Form Methods for Symbolic Creation of Approximate Solutions of Nonlinear Dynamical Systems", *Mathematics and Computers in Simulation*, Vol. 57, pp. 253-289, 2001.

47. Qinsheng B. and P. Yu, "Symbolic Computation of Normal Forms for Semi-Simple Cases", *Journal of Computational and Applied Mathematics*, Vol. 102, pp. 195-220, 1999.

48. Rand, R. H., and D. Armbruster, *Perturbation Methods, Bifurcation Theory and Computer Algebra*, Springer-Verlag, New York, 1987.