

Fall 1999

Topics in chaotic secure communication

Andrew Thomas Parker
University of New Hampshire, Durham

Follow this and additional works at: <https://scholars.unh.edu/dissertation>

Recommended Citation

Parker, Andrew Thomas, "Topics in chaotic secure communication" (1999). *Doctoral Dissertations*. 2097.
<https://scholars.unh.edu/dissertation/2097>

This Dissertation is brought to you for free and open access by the Student Scholarship at University of New Hampshire Scholars' Repository. It has been accepted for inclusion in Doctoral Dissertations by an authorized administrator of University of New Hampshire Scholars' Repository. For more information, please contact nicole.hentz@unh.edu.

INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI[®]

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Topics in Chaotic Secure Communication

BY

Andrew T. Parker

B.S., Principia College, 1992
M.S., University of New Hampshire, 1996

DISSERTATION

Submitted to the University of New Hampshire
in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

in

Mathematics

September 1999

UMI Number: 9944000

**Copyright 1999 by
Parker, Andrew Thomas**

All rights reserved.

**UMI Microform 9944000
Copyright 1999, by UMI Company. All rights reserved.**

**This microform edition is protected against unauthorized
copying under Title 17, United States Code.**

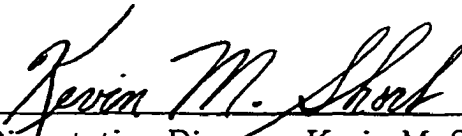
UMI
300 North Zeeb Road
Ann Arbor, MI 48103

ALL RIGHTS RESERVED


©1999

Andrew T. Parker

This dissertation has been examined and approved.



Dissertation Director, Kevin M. Short
Associate Professor of Mathematics



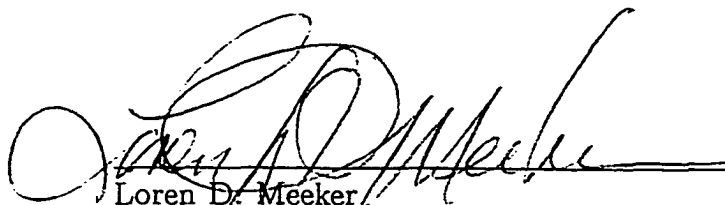
Kelly J. Black
Associate Professor of Mathematics



John B. Geddes
Faculty In Residence, Mathematics



Ernst Linder
Professor of Mathematics



Loren D. Meeker
Professor of Mathematics

Date July 29, 1999

Dedication

With love to mom, dad,
and my sisters.

Acknowledgments

This work could not have been done without the support and guidance of my advisor, Kevin Short. As a result of our joint efforts my professional development has been very well-rounded; I have had the opportunity to participate in the publishing process from beginning to end, prepare and present talks at conferences and interviews, even referee an article. He has shared with me his advice and experience on everything from reading and dissecting a research article to teaching with technology to obtaining tenure. I know that further development of my research skills will be required of me in the future, but thanks to Professor Short's coaching I feel that I am building on a firm foundation.

I also wish to thank the rest of my dissertation committee; Kelly Black, Ernst Linder, Dave Meeker and John Geddes. The time they committed to reviewing and critiquing this work resulted in a much clearer presentation and a more pleasant experience for the reader.

This research has been financially supported by a grant through the Center for Research on Applied Signal Processing at the University of Southern California (Contract No. 012132).

TABLE OF CONTENTS

Dedication	iv
Acknowledgments	v
Abstract	ix
1 Background	1
1.1 Introduction	1
1.2 Chaotic synchronization	1
1.3 Communication methods	8
1.4 Nonlinear dynamic (NLD) forecasting	10
1.5 Other examples of chaotic systems	16
1.6 Using chaos in cryptography	20
2 New applications of NLD forecasting techniques to chaotic communication schemes	23
2.1 Introduction	23
2.2 NLD forecasting techniques applied to hyperchaotic communication systems	24
2.2.1 Active-passive decomposition approach to chaotic synchronization . .	25
2.2.2 Results of NLD forecasting	27
2.2.3 Discussion	33
2.3 Reconstructing the key stream from a chaotic encryption scheme	36
2.3.1 Introduction	36

2.3.2	Hybrid Communication System Using Encryption and Synchronizing Chaos	36
2.3.3	Recover Encrypted Signal	38
2.3.4	Recover Key Stream	42
2.3.5	Recover Encryption Function	43
2.3.6	Trial on Voice Data	48
2.3.7	Discussion	50
2.4	NLD detection of controls	50
2.4.1	Introduction	50
2.4.2	Control method	51
2.4.3	Detection of Controls	55
2.4.4	Discussion	58
3	Applications of NLD forecasting techniques to nonchaotic communications problems	59
3.1	Introduction	59
3.2	Digital communication techniques	61
3.3	Bit error correction	66
3.3.1	Effect of dispersion and fading on error correction	69
3.4	Cochannel demodulation	72
3.5	Examination of real GMSK Data	84
3.5.1	Background	84
3.5.2	Procedure and Results	84

4	Digital chaotic communication system	93
4.1	Introduction	93
4.2	Binary Chaotic Communications	94
4.3	Remote Initialization of Receiver	102
4.4	Binary Communication with One-Dimensional Maps	104
4.4.1	Initialization of map-based systems	112
4.4.2	A class of one-dimensional maps with perfect statistics	113
4.5	Testing for Determinism	118
4.6	Cryptographic context	125
4.6.1	Block cipher modes	127
4.6.2	Stream cipher modes	129
4.7	Discussion and Conclusions	132
5	Future Directions	136
	Bibliography	138

List of Tables

4.1	A portion of the table containing control instructions.	99
4.2	100 “random” numbers from the pseudo-random number generator in [8]. . .	115

List of Figures

1-1 Lorenz attractor projected onto the x - z plane.	4
1-2 Estimating the largest Liapunov exponent: $\lambda \approx 0.9$	5
1-3 A view of the 3-dimensional Lorenz attractor.	12
1-4 Observed quantity $x(t)$	12
1-5 3-dimensional reconstruction using only $x(t)$	13
1-6 Close-up view of a reconstruction of s_t	14
1-7 The Rössler attractor.	18
1-8 Nonlinear resistance $g(v)$	19
1-9 The double scroll attractor.	20
2-1 A portion of the transmitted signal, carrying square wave message.	29
2-2 Three-dimensional reconstruction of signal, with message signal present. . .	30
2-3 Residuals after NLD forecasting in a well-represented region of the attractor.	30
2-4 Residuals after NLD forecasting in a poorly represented region of the attractor.	31
2-5 Filtered residuals.	31
2-6 Original square wave and results from multi-step predictions.	32
2-7 Original voice trace (“testing 1–2–3...”).	34
2-8 Hyperchaotic signal carrying voice data.	34
2-9 Extracted speech.	35
2-10 Encryption function.	38
2-11 Stages in a sample transmission.	39
2-12 Reconstructed intercepted signal.	40
2-13 Reconstructed low-pass filtered signal.	40
2-14 Estimated encrypted signal: dotted line represents the original.	41
2-15 Decryption using estimated encrypted signal.	41
2-16 Singular value decomposition of intercepted signal.	44
2-17 Frequency spectrum of transmitted signal: dotted line indicates the spectral peak of the message signal.	45
2-18 Frequency spectrum of decrypted intercepted signal.	45
2-19 Signals in the time domain.	46
2-20 Results using speech data.	49
2-21 Double scroll oscillator showing surfaces.	52
2-22 Controlled signal showing information bit stream.	53
2-23 Function $r_{12}(x)$	55
2-24 Time-delay reconstruction of controlled signal	56
2-25 Derivative reconstruction of controlled signal	57
2-26 Residuals from NLD forecasting	58
3-1 MSK waveform construction.	64
3-2 (a) Pure MSK signal reconstruction. (b) Reconstruction of signal after pass- ing through a channel.	68
3-3 Clean, interrupted MSK signal, showing two possible correct transmissions.	69

3-4	(a) Pure interrupted MSK signal reconstruction. (b) Reconstruction of interrupted signal after passing through a channel.	70
3-5	Effect of dispersion on MSK signal reconstruction.	71
3-6	Effect of fading on MSK signal reconstruction.	72
3-7	Example of cochannel interference.	75
3-8	Cochannel reconstruction, zero symbol offset.	76
3-9	Cochannel reconstruction with an offset of 0.3 symbol.	76
3-10	Cochannel reconstruction with an offset of 0.5 symbol.	77
3-11	Cochannel reconstruction with an offset of 0.8 symbol.	77
3-12	Determining which bitstream is associated with which signal.	79
3-13	Reconstruction for first case where signal B switches bits.	80
3-14	Reconstruction for second case where signal A switches bits.	80
3-15	Interfering MSK signals at different amplitudes.	81
3-16	Reconstruction of interfering signals transmitting "Eureka" and "Ansatz".	82
3-17	Case where signal carrying "E" is at full power.	82
3-18	Case where signal carrying "A" is at full power.	83
3-19	Raw GMSK data.	85
3-20	Reconstructed raw GMSK data.	86
3-21	Original data and results of basebanding.	87
3-22	Reconstructed basebanded GMSK data.	88
3-23	Reconstructed simulated GMSK data.	88
3-24	Demodulated raw and basebanded signals.	90
3-25	Reconstruction of the demodulated basebanded GMSK signal.	92
4-1	Poincaré section of the double scroll attractor, showing best fit line.	98
4-2	Period-5 orbit, resulting from the initialization code "01011."	103
4-3	One-dimensional Poincaré map for the double scroll oscillator.	106
4-4	Double scroll oscillator, showing surfaces.	106
4-5	Expanded view of Figure 4-3.	107
4-6	Illustrating one cause of grammar limitations	110
4-7	Two candidate maps for this system.	112
4-8	True period-7 orbit, without controls.	113
4-9	Structure in "random" numbers revealed.	116
4-10	Sample map from Zhou <i>et al</i> family.	119
4-11	Histograms showing encodings of all possible 3-bit message words into 3-bit transmitted words.	120
4-12	Correlation plots for the "King Henry V" data using the map $g(x_n)$	122
4-13	Correlation plots for the "King Henry V" data using the map $\phi(x_n)$	123
4-14	16-bit reconstruction of "King Henry V" transmission using $g(x_n)$, $N = 7$	124
4-15	16-bit reconstruction of "King Henry V" transmission using $\phi(x_n)$, $N = 7$	125
4-16	Kolmogorov entropy of "King Henry V" transmission using the map $\phi(x_n)$, $N = 7$	126

ABSTRACT

Topics in Chaotic Secure Communication

by

Andrew T. Parker

University of New Hampshire, September, 1999

Results in nonlinear dynamics and chaos during this decade have been applied to problems in secure communications with limited success. Most of these applications have been based on the chaotic synchronization property discovered by Pecora and Carroll in 1989 [37]. Short [44, 45, 48] demonstrated the effectiveness of nonlinear dynamic (NLD) forecasting methods in breaking this class of communication schemes. In response, investigators have proposed enhancements to the basic synchronization technique in an attempt to improve the security properties. In this work two of these newer communication systems will be analyzed using NLD forecasting and other techniques to determine the level of security they provide. It will be shown that the transmitted waveform alone allows an eavesdropper to extract the message.

During the course of this research, a new impulsively initialized, binary chaotic communication scheme has been developed, which eliminates the most significant weaknesses of its predecessors. This new approach is based on symbolic dynamics and chaotic control, and may be implemented using one-dimensional maps, which gives the designer more control over the statistics of the transmitted binary stream. Recent results in a certain class of one-dimensional chaotic maps will be discussed in this context.

The potential for using NLD techniques in problems from standard digital communications will also be explored. The two problems which will be addressed are bit errors due

to channel effects and co-channel interference. It will be shown that NLD reconstruction methods provide a way to exploit the short-term determinism that is present in these types of communication signals.

Chapter 1

Background

1.1 Introduction

In recent years there has been much interest in the potential application of results in nonlinear dynamics to secure communications. In this chapter, some of the essential developments in the study of nonlinear dynamics and chaos which have applications to communication will be reviewed. We begin in Section 1.2 with the discovery by Pecora and Carroll [37] that two chaotic systems may be synchronized via a unidirectional coupling. Early attempts to use this property in the context of a transmitter and receiver to provide a secure communication channel will be reviewed in Section 1.3. Nonlinear dynamic (NLD) forecasting will be summarized in Section 1.4 and its effectiveness in exposing weaknesses in communication systems based on a synchronized chaotic transmitter and receiver will be discussed. Two important chaotic systems which will be instrumental in this work will be introduced in Section 1.5. Finally, some recent discussions about the use of chaotic systems in a cryptologic setting will be presented in Section 1.6.

1.2 Chaotic synchronization

While there is no universally accepted definition for the mathematical phenomenon known as **chaos** [3], there are three properties [56] which are typically included in its characterization: aperiodicity, determinism and sensitive dependence on initial conditions. The first property

simply means that there exist trajectories which do not converge onto fixed points or periodic or quasiperiodic orbits. The second property excludes random or noisy inputs into the process. The third property implies that nearby trajectories separate over time, usually exponentially fast.

An example of a well-studied chaotic system is the Lorenz system, discovered in 1963 by Edward Lorenz [27] as he was modeling the behavior of atmospheric convection rolls. The system evolves according to the following equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= rx - y - xz \\ \dot{z} &= xy - bz,\end{aligned}\tag{1}$$

where typically $\sigma = 10$, $b = \frac{8}{3}$ and $r = 28$. With only two nonlinear terms, these equations have surprisingly complex solutions. A plot of a long trajectory in the x - z plane is shown in Figure 1-1. The apparent intersection of trajectories is an illusion caused by the two-dimensional projection. The well-defined structure to which the trajectory seems to be attracted and confined is an example of what is commonly referred to as a **strange attractor**. In linear systems, trajectories will either be attracted to a fixed point or else be on a closed orbit, resulting in either a stationary or periodic flow. However, in chaotic systems, which are nonlinear, it is possible for trajectories to be attracted to and confined within a finite volume in phase space, yet never becoming stationary, periodic or even quasiperiodic. In the Lorenz system, this attracting set has two lobes, and trajectories on each lobe spiral outward. However, there is a clear division at the center of the attractor where some trajec-

tories are split off and inserted into the other lobe while the rest remain on the current lobe. This division and reinsertion of trajectories is a common characteristic of three-dimensional chaotic attractors.

Just as it is difficult to define chaos universally, there remains some disagreement on how to define the notion of a strange attractor. Strogatz [56] provides a working definition which will suffice in the present context. Suppose A is a closed subset of the phase space of a dynamical system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$. Then A is a strange attractor if [56]

1. A is invariant: $\mathbf{x}(0) \in A \Rightarrow \forall t > 0, \mathbf{x}(t) \in A$.
2. A attracts an open set of initial conditions: there exists an open set $U, A \subset U$, such that $\mathbf{x}(0) \in U$ implies that

$$\liminf_{t \rightarrow \infty} \{\|\mathbf{x}(t) - a\| : a \in A\} = 0.$$

3. A is minimal: $B \subset A, B \neq A$ implies that B fails at least one of 1 and 2 above.
4. Trajectories on A exhibit sensitive dependence on initial conditions.

In Figure 1-1 it appears that the trajectory does not settle onto a periodic orbit or a fixed point. Also, the equations (1) do not have any random inputs. Thus this system satisfies the two conditions of aperiodicity and determinism. To see how sensitive this system is to initial conditions we can start two trajectories, $(x_1(t), y_1(t), z_1(t))$ and $(x_2(t), y_2(t), z_2(t))$, very close to each other and examine their rate of separation. The graph in Figure 1-2 shows the log of the distance $\ln(\|\delta(t)\|)$ between two solution curves of the Lorenz system, where $\delta(t) = (x_2(t) - x_1(t), y_2(t) - y_1(t), z_2(t) - z_1(t))$ and $\|\delta(0)\| \approx 10^{-14}$. The slope $\lambda \approx 0.9$ of

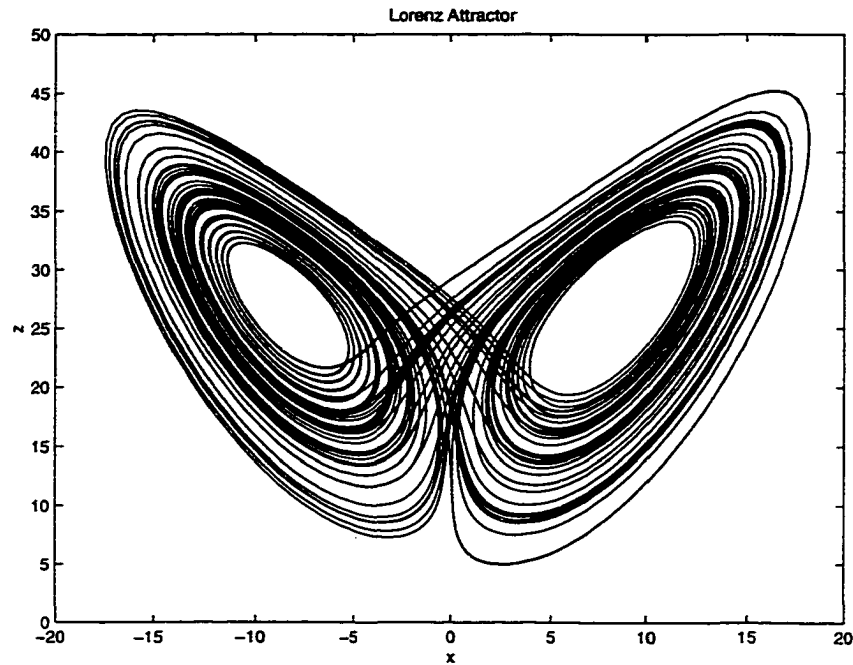


Figure 1-1 Lorenz attractor projected onto the x - z plane.

the line which best fits the graph is then an approximation to the exponential rate of growth of this separation, where $\|\delta(t)\| \sim \|\delta(0)\| e^{\lambda t}$. This number λ is an estimate of the largest **Liapunov exponent** for the system. Any chaotic system will have at least one positive Liapunov exponent.

Pecora and Carroll [37] discovered that if two identical chaotic circuits were coupled in a simple way, their behavior would become synchronized very quickly. Synchronization in this context means that the state space variables in one system become identical to their counterparts in the other system. This was a surprising result, since normally any two trajectories in a chaotic system would separate exponentially fast according to the largest Liapunov exponent. So even if two identical, uncoupled circuits were started with the same initial conditions, material differences in the circuitry or internal noise would decorrelate

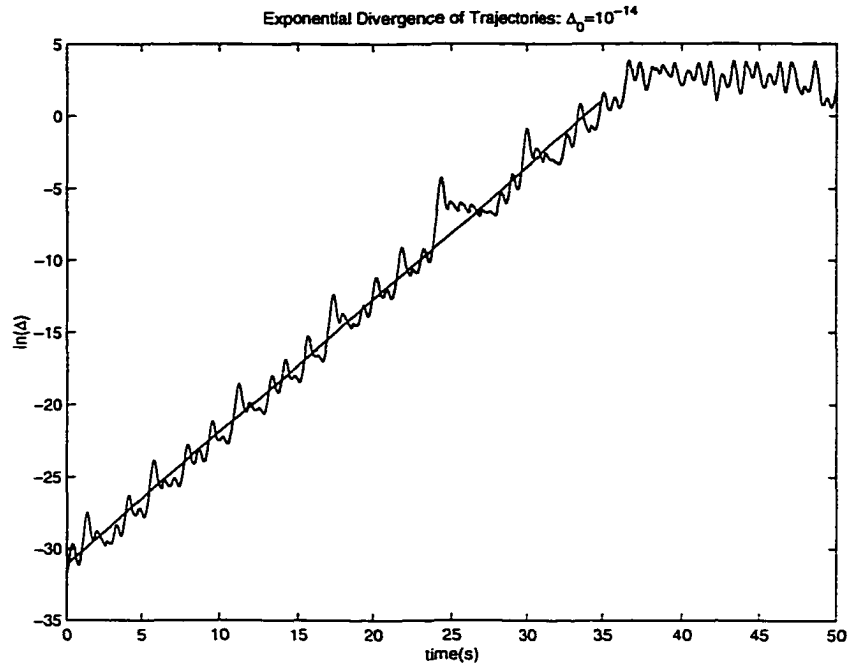


Figure 1-2 Estimating the largest Liapunov exponent: $\lambda \approx 0.9$

them. Cuomo and Oppenheim [11] presented a simplified proof of this synchronization property which will be reviewed here.

Suppose we have a chaotic circuit which we would like to use as a signal generator and which evolves according to Equations (1). While there are several ways to design a response system which will synchronize with the original, in this discussion the response system is given by

$$\dot{x}_r = \sigma(y_r - x_r)$$

$$\dot{y}_r = rx - y_r - xz_r$$

$$\dot{z}_r = xy_r - bz_r$$

where x is the transmitted variable and x_r, y_r, z_r are the receiver variables. In practice, the circuitry simply has the x_r wire cut and replaced by the transmitted signal x . We can now define a time-varying error vector $\mathbf{e} \equiv (x - x_r, y - y_r, z - z_r)$, and thus derive the equations governing the error dynamics along trajectories by using the original equations:

$$\begin{aligned}\dot{e}_1 &= \sigma(e_2 - e_1) \\ \dot{e}_2 &= -e_2 - xe_3 \\ \dot{e}_3 &= xe_2 - be_3.\end{aligned}\tag{2}$$

If we can show that $\mathbf{e} \rightarrow \mathbf{0}$ as $t \rightarrow \infty$ for all initial conditions, then we will have shown that the drive and response Lorenz circuits will synchronize. This can be achieved by finding an appropriate **Liapunov function**, $E(\mathbf{e})$, which is defined to have the following properties [56]:

1. $E(\mathbf{e}) > 0$ for all $\mathbf{e} \neq \mathbf{0}$ and $E(\mathbf{0}) = 0$ (i.e. E is positive definite).
2. $\dot{E} < 0$ for all $\mathbf{e} \neq \mathbf{0}$.

In other words, E acts like a positive measure of the energy of the error system which is decreasing monotonically over time. Therefore $E \rightarrow 0$ and since $E = 0$ only when $\mathbf{e} = \mathbf{0}$, this means that the two coupled systems will synchronize. For this error system we can show that the function

$$E = \frac{1}{2} \left(\frac{1}{\sigma} e_1^2 + e_2^2 + e_3^2 \right)$$

is a Liapunov function. Clearly, for $\sigma > 0$, $E \geq 0$. If we take the derivative of E along

trajectories of \mathbf{e} , using the error system equations to replace \dot{e}_1 , \dot{e}_2 , and \dot{e}_3 , we find

$$\begin{aligned}
\dot{E} &= \frac{1}{\sigma}e_1\dot{e}_1 + e_2\dot{e}_2 + e_3\dot{e}_3 \\
&= \frac{1}{\sigma}e_1\sigma(e_2 - e_1) + e_2(-e_2 - xe_3) + e_3(xe_2 - be_3) \\
&= -e_1^2 + e_1e_2 - e_2^2 - be_3^2 \\
&= -\left(e_1 - \frac{1}{2}e_2\right)^2 - \frac{3}{4}e_2^2 - be_3^2 \\
&\leq 0,
\end{aligned}$$

where in the last step we have completed the square. Thus E is a Liapunov function as long as $b, \sigma > 0$. Therefore, all trajectories in the error system \mathbf{e} are attracted to the origin and the two systems become synchronized.

It would be beneficial for applications if we knew something about the error's rate of convergence toward zero. The proof of synchronization was extended in [9] to show that the error decays exponentially. Consider the function $V = \frac{1}{2}e_2^2 + \frac{1}{2}e_3^2$. Then, since typically $b = \frac{8}{3} > \frac{1}{2}$,

$$\begin{aligned}
\dot{V} &= e_2\dot{e}_2 + e_3\dot{e}_3 \\
&= e_2(-e_2 - xe_3) + e_3(xe_2 - be_3) \\
&= -e_2^2 - 2be_3^2 \\
&\leq -e_2^2 - e_3^2 \\
&= -2V.
\end{aligned}$$

Thus $V = o(e^{-2t})$ for all time t . So e_2 and e_3 decay exponentially fast. Since $\dot{e}_1 =$

$-\sigma e_1 + \sigma e_2$, when e_2 is small, e_1 behaves like $e^{-\sigma t}$. Therefore the entire error system converges exponentially to $\mathbf{0}$.

1.3 Communication methods

The idea of hiding an information-bearing signal in a noisy transmission is not new. Examples of existing techniques include frequency-hopping, where the carrier frequency is switched erratically, making it difficult for an intruder to “tune in” to the signal; and spread-spectrum methods of transmitting information over a broad frequency band. If the spectral range of the message signal is within the spectral range of the noisy carrier, this can be an especially effective method of hiding the information, since spectral filtering techniques used by an intruder on an intercepted signal will have limited success. The problem for the intended receiver (as well as an intruder) is then to reproduce exactly the carrier signal in order to separate the message from the noise. This is what makes the synchronization of two chaotic circuits so interesting: the chaotic transmission has a broad spectral range, behaves unpredictably, yet may be reproduced almost perfectly by the intended receiver. Unfortunately, the proof of synchronization discussed in the previous section holds only for a pure, message-free drive signal. In practice, however, the synchronization is robust enough that small perturbations to the drive signal have little effect on the tendency to synchronize. Therefore, Pecora and Carroll [5] found that a drive signal of the form $s(t) = x(t) + m(t)$, where $m(t)$ is an information-bearing signal whose power is much less than that of $x(t)$, allowed synchronization to occur with enough accuracy to be able to recover the message by calculating $\tilde{m}(t) = s(t) - x_r(t)$. However, because the message signal perturbs the dynamics of the receiver slightly, there is some small error in the recovered message \tilde{m} .

This problem was overcome by Wu and Chua [60] using a feedback mechanism which ensures that the message modulates the dynamics of both the transmitter and receiver in exactly the same way. The equations for the transmitter in this system are

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= (r - \mu)[x + m] + \mu x - y - [x + m]z \\ \dot{z} &= [x + m]y - bz,\end{aligned}$$

and the corresponding receiver equations are

$$\begin{aligned}\dot{x}_r &= \sigma(y_r - x_r) \\ \dot{y}_r &= (r - \mu)[x + m] + \mu x_r - y_r - [x + m]z_r \\ \dot{z}_r &= [x + m]y_r - bz_r\end{aligned}$$

where $\sigma = 16$, $r = 45.6$, $\mu = 0.98$, $b = 4$, and the transmitted signal is $s(t) = x(t) + m(t)$. Notice that the message signal $m(t)$ modulates the dynamics of both the transmitter and the receiver. The error system for this configuration is

$$\begin{aligned}\dot{e}_1 &= \sigma(e_2 - e_1) \\ \dot{e}_2 &= \mu e_1 - e_2 - (x + m)e_3 \\ \dot{e}_3 &= (x + m)e_2 - be_3.\end{aligned}$$

To prove that the errors decay to $\mathbf{0}$ we use the same Liapunov function as before, $E = \frac{1}{2}(\frac{1}{\sigma}e_1^2 + e_2^2 + e_3^2)$ which now has the derivative

$$\dot{E} = -(e_1 - 0.99e_2)^2 - 0.0199e_2^2 - be_3^2$$

when we complete the square and insert the parameter values used for this system. Note that the synchronization criteria places no restrictions on the strength of the message, although in practice one would want to keep the message power reasonably small. Since the message was included in the system equations, the Wu and Chua scheme has perfect message recovery.

1.4 Nonlinear dynamic (NLD) forecasting

The synchronization of chaotic systems is an interesting mathematical result, and has been shown to have potentially useful applications. The broad-band frequency spectrum of many chaotic systems makes it easy to hide an information-bearing signal in the same frequency band as the chaos, which then prevents the application of spectral filtering techniques from having much success in extracting the message. The difficulty for an intruder lies in using the scalar chaotic carrier signal to model the underlying dynamics well enough to draw some conclusions about the state of the transmitter at any given time.

A theorem proved by Takens [57] provides a method of reconstructing the chaotic attractor from which the carrier signal is taken. Suppose $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ is an n -dimensional chaotic dynamical system with $\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_n(t))$. Denote the associated strange attractor by M . Then the theorem from [57] proves that, for any $1 \leq k \leq n$ and any real $\tau \neq 0$, the function $\mathbf{y}(t) = (x_k(t), x_k(t+\tau), x_k(t+2\tau), \dots, x_k(t+2n\tau))$ is generically a diffeomorphism

of M in \mathcal{R}^{2n+1} . In other words, the entire geometrical structure of the original attractor may be reconstructed from a single observed quantity. The theorem is more general than what has been stated here, but this is the form which will be most useful in this work. A similar theorem is proved in [57] showing that the function $\mathbf{y}(t) = (x_k(t), \dot{x}_k(t), \ddot{x}_k(t), \dots, x_k^{(2n)}(t))$, where the derivatives are taken with respect to time, is a diffeomorphism as well. In most of the work presented here, the time-delay method of reconstruction is used, where τ is typically chosen to be the first zero-crossing of the sample autocorrelation $\bar{\rho}_x(\tau)$ of the time series $x(t)$ [1, 31]. There are other methods [13, 14] for choosing the time delay τ ; however, in practice, if the choice based on the autocorrelation function results in a clear reconstruction, then the other techniques will offer at best only slight improvements.

As a simple example, consider the Lorenz system (1). In Figure 1-3 a 3-dimensional phase-space plot of a long trajectory is shown. Suppose we observe only the state variable $x(t)$ for this orbit, shown in Figure 1-4. Using the idea of the theorem by Takens, and by choosing τ as described above, we may reconstruct the shape of the entire 3-dimensional attractor by plotting the 3-dimensional curve $(x(t), x(t + \tau), x(t + 2\tau))$. This reconstruction is shown in Figure 1-5. Note the similarity of the reconstructed attractor to the original.

Now, suppose that a low-power message $m(t)$ is incorporated into the transmitted signal $s(t)$, as in $s(t) = x(t) + m(t)$, either with or without a feedback mechanism. Since $m(t)$ will typically be uncorrelated with the underlying chaotic carrier, this addition will cause the trajectory of $s(t)$ to stray from the reconstructed attractor, as well as cause trajectories in a local region of the reconstruction to cross. This latter behavior is clearly inconsistent with any solution to a set of autonomous ODE's such as the Lorenz system. The problem of extracting the message then becomes one of removing these inconsistencies from the data

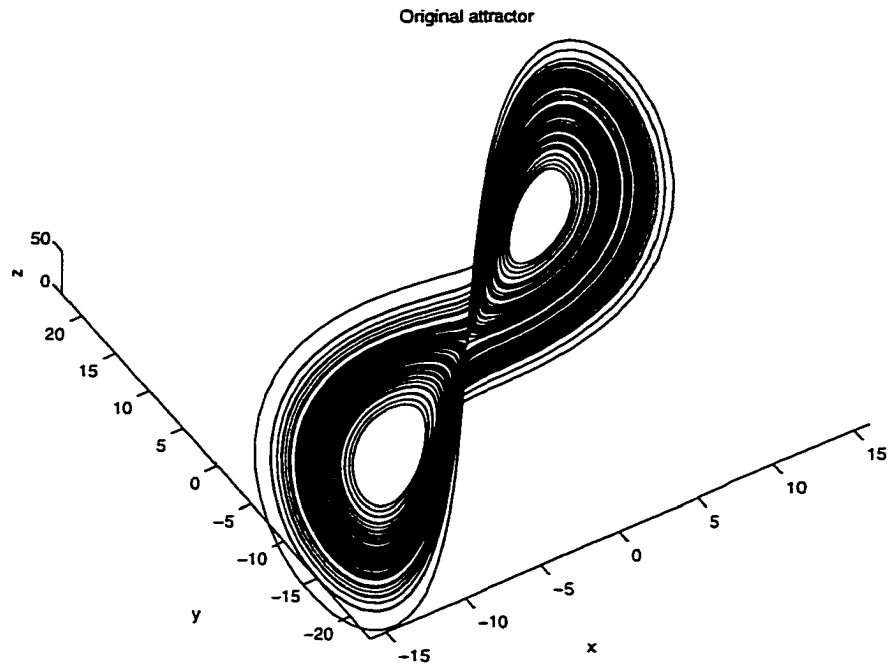


Figure 1-3 A view of the 3-dimensional Lorenz attractor.

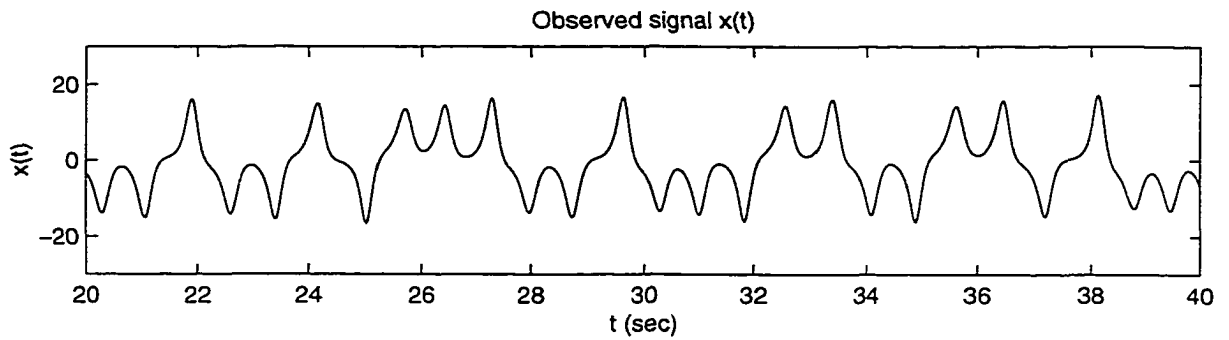


Figure 1-4 Observed quantity $x(t)$.

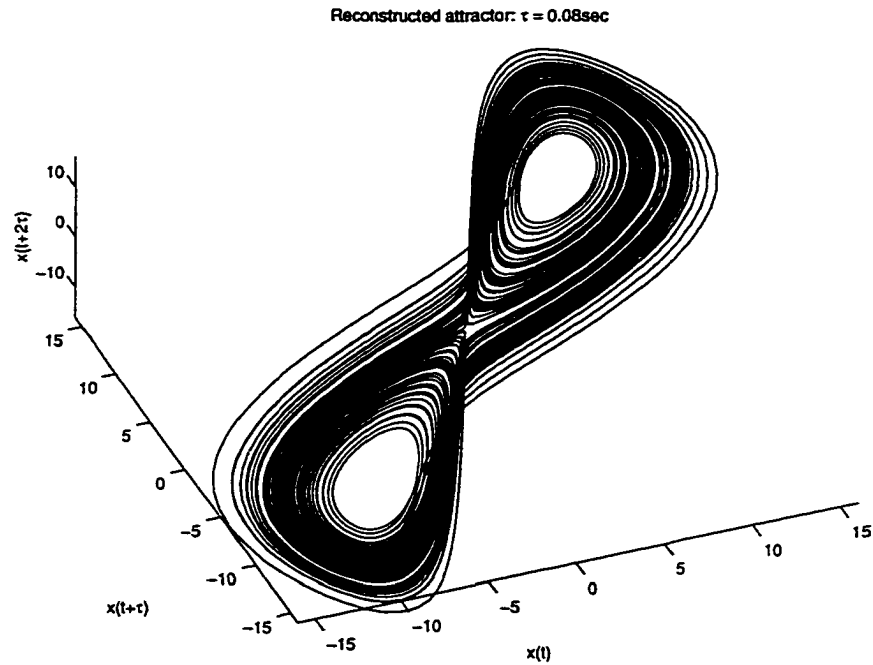


Figure 1-5 3-dimensional reconstruction using only $x(t)$.

and taking the difference between the cleaned and the received signals. Nonlinear dynamic (NLD) forecasting has been used to accomplish this task effectively [44, 45, 48]. A brief description of this technique follows—a more complete presentation may be found in [44].

Figure 1-6 shows a close-up of a region of a reconstructed Lorenz attractor where the observed chaotic signal contains a low-amplitude message. The trajectories in this figure are sparse for the purpose of illustration—in practice one would typically have more data than what is shown. The dotted curves show where the pure chaotic trajectories would lie if there were no message signal present. In NLD forecasting, a prediction about the evolution of each point in the time series is made based on the local dynamics in reconstructed space, rather than on neighboring points in time. Suppose that the signal $s(t)$ is sampled evenly, such that $s_i = s(i\Delta t)$, and suppose a point s_p is chosen in the reconstruction. In order

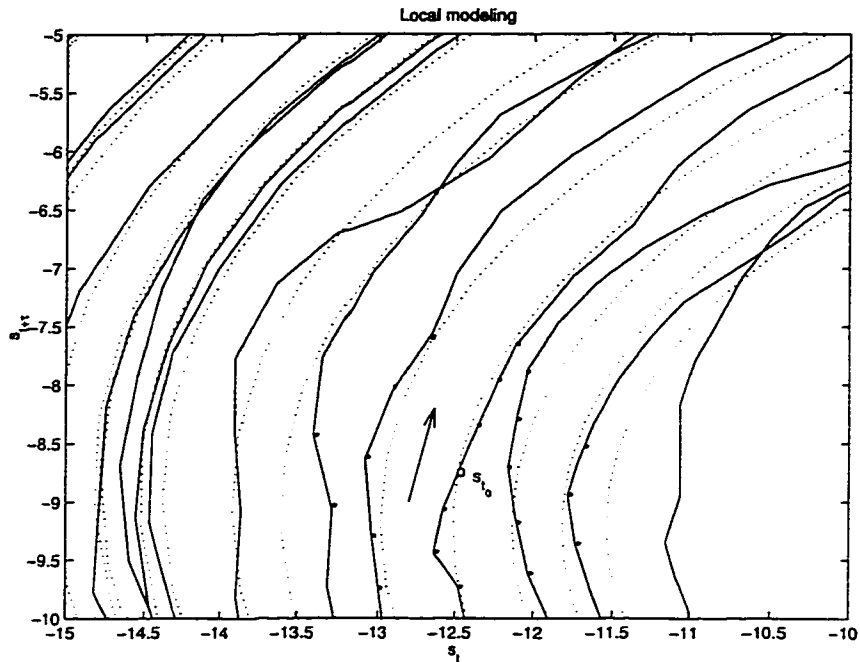


Figure 1-6 Close-up view of a reconstruction of s_t .

to determine the evolution of this point based on the dominant local flow on the attractor, neighboring points are chosen and their actual movements are recorded. In practice, 25 neighbors are usually sufficient. Although the effect of the message will be to introduce components lateral to the dominant local flow direction, these components will tend to average out over the neighborhood of s_p . This is because the message and the chaos are dynamically uncorrelated in the reconstruction. That is, there will be no consistent relationship over the local geometry between the message signal and the chaotic carrier. The movements of neighboring points may be modeled in a least-squares sense, usually using polynomial basis functions of degree 2. Suppose $\{s_{q_i}\}$ is the chosen set of neighbors of s_p , m is the embedding dimension and $\{\phi_k\}$ is the chosen set of basis functions. Then for the predictor function

$\mathbf{F}(\mathbf{s}_{q_i}) = (f_0(\mathbf{s}_{q_i}), f_1(\mathbf{s}_{q_i}), \dots, f_{m-1}(\mathbf{s}_{q_i}))$ we have

$$f_j(\mathbf{s}_{q_i}) = \sum_{k=0}^n \lambda_{jk} \phi_k(\mathbf{s}_{q_i})$$

where n is the number of terms in the basis and λ_{jk} are the expansion coefficients. The system of equations $\mathbf{F}(\mathbf{s}_{q_i}) = \mathbf{s}_{q_{i+1}}$ may now be solved in a least-squares sense for the coefficients λ_{jk} . Once this model is constructed, a prediction $\tilde{\mathbf{s}}_{p+1} = \mathbf{F}(\mathbf{s}_p)$ may be obtained about the location of the point \mathbf{s}_{p+1} . The difference $\mathbf{r}_{p+1} = \tilde{\mathbf{s}}_{p+1} - \mathbf{s}_{p+1}$ should now reflect more the effect of the message than of the chaotic carrier. However, in regions where the attractor is poorly represented, or where there is an apparent self-intersection of the attractor, bad predictions may occur which degrade the message extraction.

To combat these and other difficulties, some important enhancements to this technique were presented in [44]. For example, to make the calculations more numerically stable, local coordinate axes are chosen which are aligned with the dominant flow direction. Then, since many chaotic attractors, such as the Lorenz attractor, locally approximate a 2-dimensional surface, components of the neighboring trajectories which are orthogonal to this surface may be ignored or zeroed out. Of course, doing this for real data requires estimating the relative importance of the dimensions. This may be done by performing a singular value decomposition (SVD) on the local trajectory matrix \mathbf{R} , where $R_{ij} = (\mathbf{s}_{q_{i+1}} - \mathbf{s}_{q_i})_j$, and eliminating the dimensions associated with very small singular values; see [44] for details. This reduces the size of the prediction problem to a subspace of the reconstructed space, simplifying the calculations and improving the robustness of the predictions. Another problem arises in regions where there is an apparent self-intersection of the attractor. Neighbors might be

chosen which are actually on different parts of the attractor, and which evolve in very different directions, resulting in poor predictions. A solution to this problem involves choosing neighbors based not only on proximity in reconstructed space but also on consistent tangent vectors in these regions [44].

Successful applications of NLD forecasting to problems in breaking chaotic communication schemes, detecting teleseismic events and other applications may be found in [47, 46, 52, 48, 45, 44]. In 1994, Short [44] broke the original additive-message communication technique developed by Pecora and Carroll. In 1996, the Wu and Chua message-modulated scheme was also broken by Short [45]. In response to these results, researchers have proposed possible improvements to the design of communication schemes which are based on chaotic synchronization, specifically attempting to foil the NLD attack. In 1995, Kocarev and Parlitz [22] investigated and developed a theory for synchronizing high-dimensional (> 3) chaotic systems using only a scalar signal. Another communication scheme developed in 1997 by Yang, Wu and Chua [63] takes a more cryptographic approach and includes a signal scrambling stage before transmission, with the intent that the signal may only be unscrambled by using a coordinate of the chaotic system that is not transmitted. Results from the analysis of these new systems will be presented in Chapter 2.

1.5 Other examples of chaotic systems

Besides the Lorenz system, there are many other well-studied chaotic systems. Two of these need to be introduced here since they will be used in later chapters. The first is a simple chaotic system which was discovered by Rössler in 1976 [39]. Drawing inspiration from a taffy-pulling machine, Rössler's intent was to derive a continuous chaotic system

whose strange attractor had a much simpler structure than that of the Lorenz system. The equations for the Rössler system are

$$\begin{aligned}\dot{x} &= -y - z \\ \dot{y} &= x + ay \\ \dot{z} &= b + z(x - c)\end{aligned}\tag{3}$$

where a chaotic attractor exists, for example, for $a = b = 0.2$, $c = 5.7$. A plot of a long trajectory for this system is shown in Figure 1-7. This system has proven to be much simpler to analyze, both quantitatively and qualitatively, than the Lorenz system, because of the single nonlinear term and the attractor's simple structure. Similar to the Lorenz system, trajectories in the x-y plane tend to spiral outward, until they reach an outer limit, beyond which they are stretched into the third dimension, folded over, and reinserted into the spiral in the x-y plane. This picture provides some intuition into the reason why three dimensions are required for continuous chaos to exist. The Rössler system will be mentioned and used briefly in Section 2.2.1.

In 1983 (see footnote in [29]) a simple electrical circuit was discovered which exhibited chaotic behavior. This system was named the “double scroll” oscillator after the shape of the resulting chaotic attractor. Since its discovery, a detailed analysis of the dynamics appeared in 1985 [29], and a proof of the existence of chaos for this circuit first appeared in 1986 [6]. The circuit's behavior is described by the equations

$$\frac{dv_{C_1}}{dt} = \frac{1}{C_1}[G(v_{C_2} - v_{C_1}) - g(v_{C_1})]$$

Rössler System

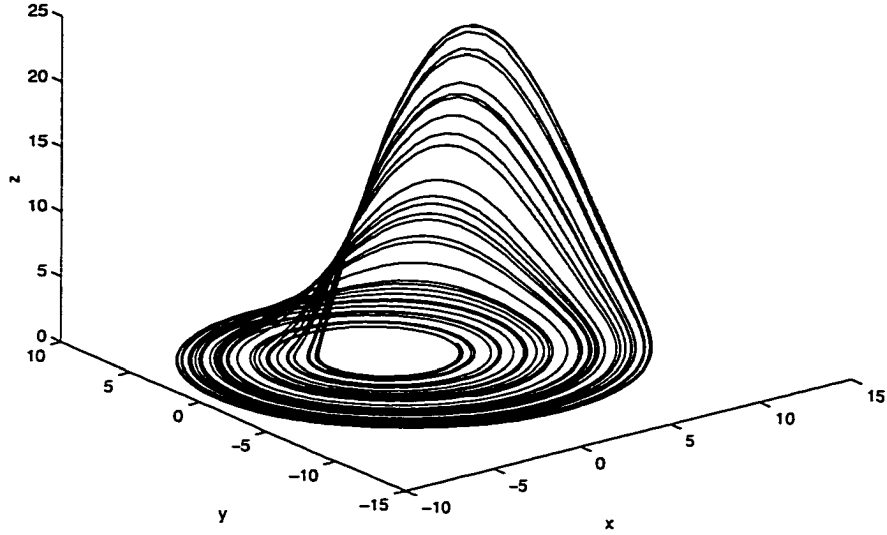


Figure 1-7 The Rössler attractor.

$$\begin{aligned}\frac{dv_{C_2}}{dt} &= \frac{1}{C_2}[G(v_{C_1} - v_{C_2}) + i_L] \\ \frac{di_L}{dt} &= \frac{-1}{L}v_{C_2},\end{aligned}\tag{4}$$

where

$$g(v) = \begin{cases} m_1 v, & \text{if } -B_p \leq v \leq B_p; \\ m_0(v + B_p) - m_1 B_p, & \text{if } v \leq -B_p; \\ m_0(v - B_p) + m_1 B_p, & \text{if } v \geq B_p, \end{cases}$$

and common parameter settings are $C_1 = \frac{1}{9}$, $C_2 = 1$, $L = \frac{1}{7}$, $G = 0.7$, $m_0 = -0.5$, $m_1 = -0.8$, and $B_p = 1$. The parameter range for which chaos persists is described in [29]. The state variable v_{C_1} represents the voltage across the capacitor C_1 , v_{C_2} the voltage across C_2 , and i_L measures the current through the inductor L . Notice that the only nonlinear term is

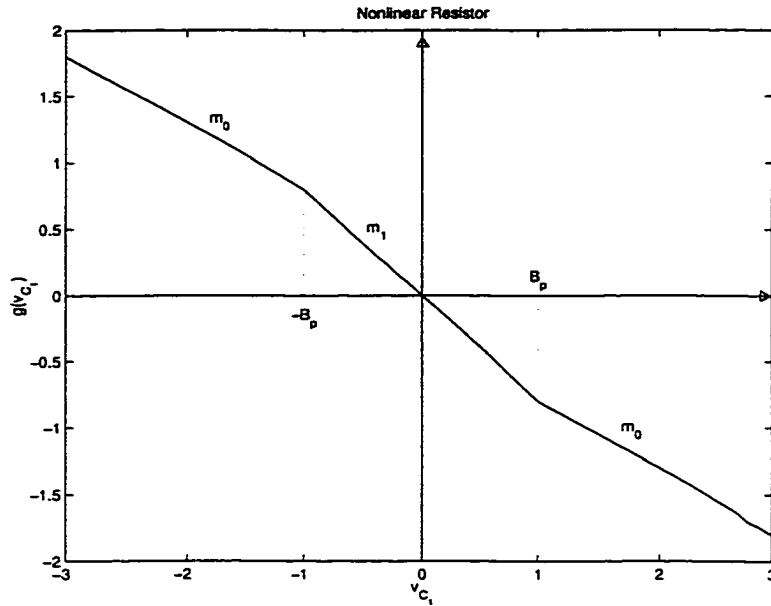


Figure 1-8 Nonlinear resistance $g(v)$.

the function $g(v)$, shown in Figure 1-8, which is itself *piecewise* linear. The function $g(v)$ represents a simple nonlinear resistor. A plot of a long trajectory is shown in Figure 1-9. Similar to the Lorenz system, this attractor has two lobes, encircling unstable fixed points at $(v_{C_1}^*, v_{C_2}^*, i_L^*) = \left(\pm \frac{(m_0 - m_1)B_p}{G + m_0}, 0, \pm \frac{(m_1 - m_0)B_p G}{G + m_0} \right)$. Following a common theme, trajectories on each lobe tend to spiral outward, until they reach a point where some trajectories are folded back and reinserted into the same lobe, like the Rössler system, while others are split off and inserted into the opposite lobe, like the Lorenz system. Once on a new lobe, any trajectory will cycle around that lobe at least twice before traveling to the other lobe. More details about the structure of this attractor may be found in [29, 6]. The double scroll oscillator will play a central role in much of Chapters 2 and 4.

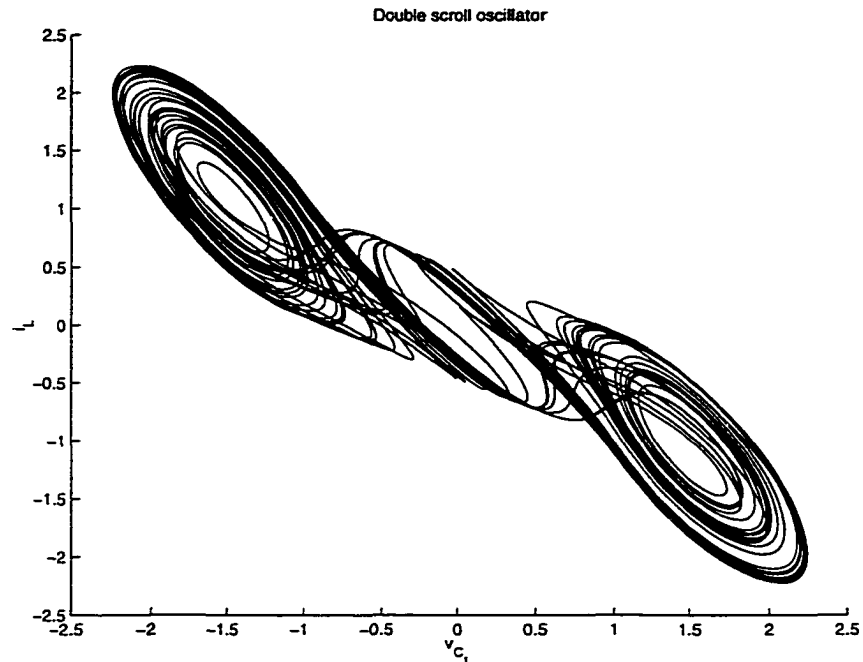


Figure 1-9 The double scroll attractor.

1.6 Using chaos in cryptography

Most research in secure communication using chaotic systems has been concerned with analog signals. However, there have been several recent results and discussions in the literature concerning the possible meeting of the fields of nonlinear dynamics and digital cryptography. For example, an interesting interchange occurred earlier this decade in the journal CRYPTOLOGIA about the utility of chaotic systems in cryptography. The discussion began with an article by Robert Matthews [30] in which the author describes a simple system for generating a random stream of letters (here called a **key stream**) using the logistic map. The idea is to calculate iterates of the map using previously agreed-upon precision, parameter and initial values. Then the last two digits in the decimal representation of the iterates

are converted modulo 25 and added to the message text modulo 25, assuming a 25-letter alphabet. A responding article by Daniel Wheeler [59] alerted readers to the potential problem that this or any digitally executed chaotic system may have with repetitions in the key stream, or key stream cycling. The problem lies in the fact that the digital representation of the chaotic system is only a finite-state machine, so that an orbit will eventually return to a previously occupied state. Since the equation governing the dynamics of the logistic map depends on only the previous iteration, the orbit is then necessarily locked in a cycle. A collaboration between Matthews and Wheeler [58] presented some evidence that this cycling problem can be prevented to any degree by using a sufficiently high precision. But the final word in this interchange came from Ross Anderson in a letter to the editor [2] where he claims that this system has no advantages over any other random function. He further complained about the possibly short cycle length and the difficulty in calculating iterates of the logistic equation versus a linear feedback shift register.

The system described above, as a free-running chaotic pseudo-random number generator, may have a problem with producing cycles. However, in Chapter 4 a chaotic encryption scheme will be presented which is a *message modulated* system [35, 34], where each iteration is dependent on the previous history of the state of the system and the message itself. Thus as long as there are no cycles in the message, there will not be any cycles in the key stream. Even if there are some repeated elements in the message text, as long as the length of the repeated message segments are incommensurate with any possible natural cycle lengths of the chaotic system, there will be no cycles in the key stream.

Fridrich [15] observes several natural connections between a good cryptographic algorithm and a chaotic dynamical system. For example, a cryptographic system must be

sensitive to changes in the message text; that is, changing one letter in a block of message text should result in a completely different encryption result for that block. This is clearly parallel to the characteristic of sensitive dependence on initial conditions in chaotic systems. Also, the encrypted message must appear random, i.e. not show any patterns or periodicities; a chaotic system's aperiodic behavior suggests the presence of a random component where there is none. "However," the author writes, "there is one important difference between chaos and encryption. Cryptosystems work on finite sets, while chaotic systems have meaning only on a continuum, an infinite set." Thus, in order for there to be a useful interchange between the two fields, "the main problem that needs to be solved is a correct generalization of chaos from a continuum to finite sets." Or, looking at the problem from the other side, a solution might be found in generalizing cryptographic algorithms to functions on continuous domains. A complete answer to this question is beyond the scope of this work, although some results in the field of chaotic encryption will be presented in Chapter 4.

Chapter 2

New applications of NLD forecasting techniques to chaotic communication schemes

2.1 Introduction

It has been shown that nonlinear dynamic (NLD) forecasting techniques have had considerable success in breaking communication schemes which are based on principles of chaotic synchronization. This has prompted researchers to propose more complex schemes in an attempt to foil the NLD attack. One suggestion has been to employ higher-dimensional chaotic systems [22]. A representative system of this type will be introduced and analyzed in Section 2.2. Another approach involves a signal encryption step which attempts to scramble the message before feeding into the chaotic circuit [63]. Several weaknesses of this approach will be revealed in Section 2.3. A completely new technique for encoding binary information in a chaotic signal using chaotic control [17] will be analyzed for security weaknesses in Section 2.4. The results on the high-dimensional chaotic systems in Section 2.2 and the chaotic control method in Section 2.4 were produced jointly with Kevin Short using software tools already developed to implement NLD forecasting. The analysis of the signal encryption approach in Section 2.3 was done primarily by the author.

2.2 NLD forecasting techniques applied to hyperchaotic communication systems

Despite the weaknesses that were exposed in the early attempts to design a secure chaotic communication system, new schemes are being designed and proposed at an increasingly rapid pace. Many of these systems specifically address and attempt to overcome the weaknesses that NLD forecasting techniques have been able to exploit. One problem with previous attempts is that the chaotic systems which have been used are low-dimensional, which makes it easy for an intruder to reconstruct accurately the chaotic attractor from an intercepted signal. This allows NLD forecasting techniques to be applied effectively. To combat this weakness, considerable interesting research has been done in the area of synchronization of high-dimensional (> 3) chaotic (hyperchaotic) systems.

In a paper by Kocarev and Parlitz [22] a generalization is made of the synchronization property of chaotic systems which accomodates hyperchaotic systems. They illustrate their theory with an example that is six-dimensional, yet which synchronizes perfectly via a scalar signal even when modulated by a message. However, it will be shown that an intruder may reconstruct much of the dynamical behavior by embedding the signal in only *three* dimensions, thereby showing that the proposed hyperchaotic system does not provide a significant improvement in security. An outline of the theoretical results in [22] as well as the six-dimensional example developed by Kocarev and Parlitz will be presented in the next section, and results from NLD forecasting will be presented and discussed in Section 2.2.2.

2.2.1 Active-passive decomposition approach to chaotic synchronization

Suppose we have a chaotic system

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, s(t)) \quad (1)$$

where $s(t)$ is a driving function determined by either $s(t) = h(\mathbf{x}(t))$ or $\dot{s} = h(\mathbf{x}, s)$. If an information signal $i(t)$ is to be incorporated, these equations become $s(t) = h(\mathbf{x}(t), i(t))$ or $\dot{s} = h(\mathbf{x}, s, i)$. Then an identical copy of the above system,

$$\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, s(t)),$$

driven by the same signal $s(t)$, will synchronize with the first system as long as the error system $\mathbf{e} = \mathbf{x} - \mathbf{y}$, or

$$\dot{\mathbf{e}} = \mathbf{f}(\mathbf{x}, s) - \mathbf{f}(\mathbf{y}, s) = \mathbf{f}(\mathbf{x}, s) - \mathbf{f}(\mathbf{x} - \mathbf{e}, s),$$

has a stable fixed point at $\mathbf{e} = \mathbf{0}$. To be practical for applications, we usually require that the origin be globally asymptotically stable.

As an example, Kocarev and Parlitz present in this context the synchronization scheme discussed in Section 1.2:

$$\begin{aligned} \dot{x}_1 &= \sigma(s - x_1) \\ \dot{x}_2 &= rx_1 - x_2 - x_1x_3 \\ \dot{x}_3 &= x_1x_2 - bx_3 \end{aligned} \quad (2)$$

where $s = h(\mathbf{x}) = x_2$. The slave system

$$\begin{aligned}
 \dot{y}_1 &= \sigma(s - y_1) \\
 \dot{y}_2 &= ry_1 - y_2 - y_1y_3 \\
 \dot{y}_3 &= y_1y_2 - by_3
 \end{aligned} \tag{3}$$

is easily seen to match the form $\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, s(t))$. The proof of synchronization, i.e. the proof that $\mathbf{e} = \mathbf{x} - \mathbf{y}$ implies $\mathbf{e} = \mathbf{0}$ is a globally asymptotically stable fixed point, is similar to that in Section 1.2; see [22]. The advantage of this formulization, however, is that the generality attained by defining the transmitted driving signal $s(t)$ in terms of a function $h(\mathbf{x})$ gives us greater freedom in choosing the actual form of the signal.

The hyperchaotic scheme under investigation in this section is composed of two familiar, three-dimensional attractors, the Lorenz and Rössler systems (see Section 1.5), coupled by an intermediate variable s_{aux} . The governing equations are:

(Rössler)	(Lorenz)
$\dot{x}_1 = 2 + x_1(x_2 - 4)$	$\dot{x}_4 = -10x_4 + s$
$\dot{x}_2 = -x_1 - x_3$	$\dot{x}_5 = 28x_4 - x_5 - x_4x_6$
$\dot{x}_3 = x_2 - 2.45x_3 + s_{aux}$	$\dot{x}_6 = x_4x_5 - 2.666x_6$
$s_{aux} = i + 3x_3$	$s = 10x_5 + 30s_{aux}/x_6$

where i is the information signal and the transmitter has the property that $x_6 > 0$, so the division causes no problems. The parameters were carefully chosen in a region where a chaotic attractor existed. Notice that i is only directly added into s_{aux} , and then s_{aux} is

coupled into the second level of the system through s . The transmitted signal is $s(t)$ alone, and it would appear that the presence of the information signal is completely hidden in the dynamics. The equations governing the receiver are essentially the same, but coupled in the reverse order:

$$\begin{aligned}
 \dot{y}_4 &= -10y_4 + s & \dot{y}_1 &= 2 + y_1(y_2 - 4) \\
 \dot{y}_5 &= 28y_4 - y_5 - y_4y_6 & \dot{y}_2 &= -y_1 - y_3 \\
 \dot{y}_6 &= y_4y_5 - 2.666y_6 & \dot{y}_3 &= y_2 - 2.45y_3 + \bar{s}_{aux} \\
 \bar{s}_{aux} &= (s - 10y_5)y_6/30 & i_R &= (s - 10y_5)y_6/30 - 3y_3
 \end{aligned}$$

where i_R is the recovered information signal. Except for an initial transient, the signal is recovered exactly. This communication scheme has been programmed so that a numerically simulated transmission could be generated, and in the next section the results of the analysis of the transmitted signal will be presented.

2.2.2 Results of NLD forecasting

To test the hyperchaotic communication scheme, both a square wave and the more complicated case of a speech waveform were used as simulated information signals. In both cases, good signal extraction was obtained. To generate the data sets, the system was numerically integrated with a time step $\Delta t = 0.01$ using a fourth order Runge–Kutta scheme. To extract the signal the attractor was reconstructed in three dimensions using a time-delay embedding with $\tau = 10$. In the NLD forecasting, second degree polynomials were used and 25 neighbors were chosen in a region of the time series where no message is present, called a “comparison region,” and predictions based on their evolution were made. Comparison regions may

be blindly detected by performing NLD forecasting once and looking for regions where the prediction errors are relatively small.

For the square wave trial, i was set to be a periodic square wave of amplitude $1/2$. The numerical integration of the system provided the simulated transmission $s(t)$. For this trial the comparison region consisted of the first 70000 points and contained no information signal, so the square wave was only turned on for the last 30000 points. We then performed NLD forecasting using second order polynomials and made one-step predictions for each point in the time series, using the techniques described in Section 1.4. Determining if signal extraction is possible is then a matter of subtracting the predicted time series from the original and checking the result.

For this data, a portion of the transmitted signal can be seen in Figure 2-1. The 3-dimensional reconstruction is shown in Figure 2-2. One can see that three dimensions is enough to capture much of the dynamics for this six-dimensional system. An NLD forecasting model of this data was calculated and the predicted dynamics were subtracted from the transmitted signal. In Figure 2-3 a part of the residual after subtracting away the predicted data is shown, and it can be seen that where the attractor was well-represented by the reconstructed time series, the predictions were accurate enough to show the edge transitions of nearly every square wave, which is the expected result for one-step predictions. However, in regions where trajectories are sparse, prediction errors are larger, as seen in Figure 2-4. Even though the predictions may be poorly behaved in these regions, the edges of the square waves can still be seen as spikes. Also, it is important to realize that since the predictions were one-step predictions, the extracted signal really just picks up the transitions in the square wave. These one-step residuals can be filtered to reconstruct an approximation of

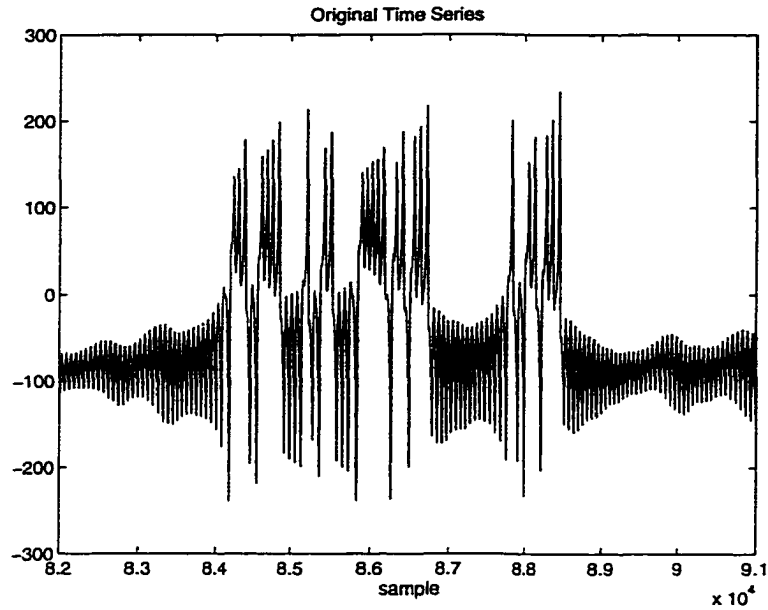


Figure 2-1 A portion of the transmitted signal, carrying square wave message.

the actual hidden signal, as described in [48]. In this case, the residual was integrated over an exponentially decaying backward window to give the result in Figure 2-5 for one of the best regions.

The use of multi-step NLD forecasting methods was considered to extract a faithful representation of the hidden square wave. Multi-step block forecasting was used to do this [48]. This means that predictions were made in blocks, where the block length was chosen to correspond to the interpeak distance detected in the square wave from the one-step predictions. Within each block, predictions are based on previously predicted data points. Then, at the end of a block, the predictions are resynchronized with the data by taking an actual data point before going on to predict the next block. In other words, a predictor function F_0 is calculated based on the dynamics in a neighborhood of a point s_{t_0} and a prediction is made by $\bar{s}_{t_0+\Delta t} = F_0(s_{t_0})$ as before. However, the next local predictor

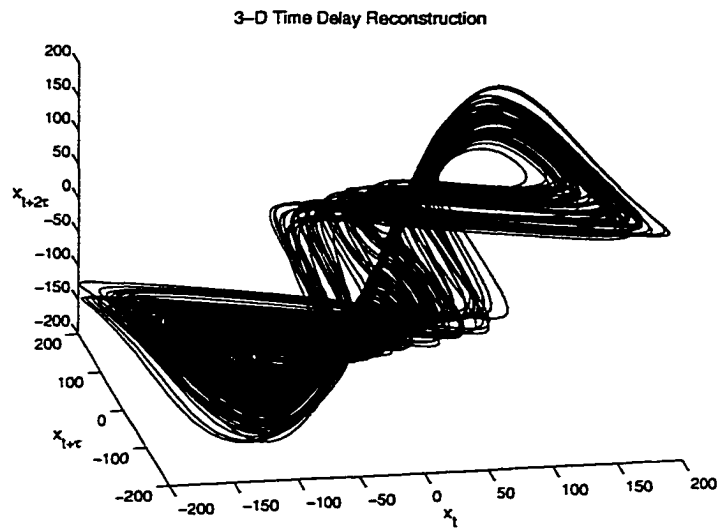


Figure 2-2 Three-dimensional reconstruction of signal, with message signal present.

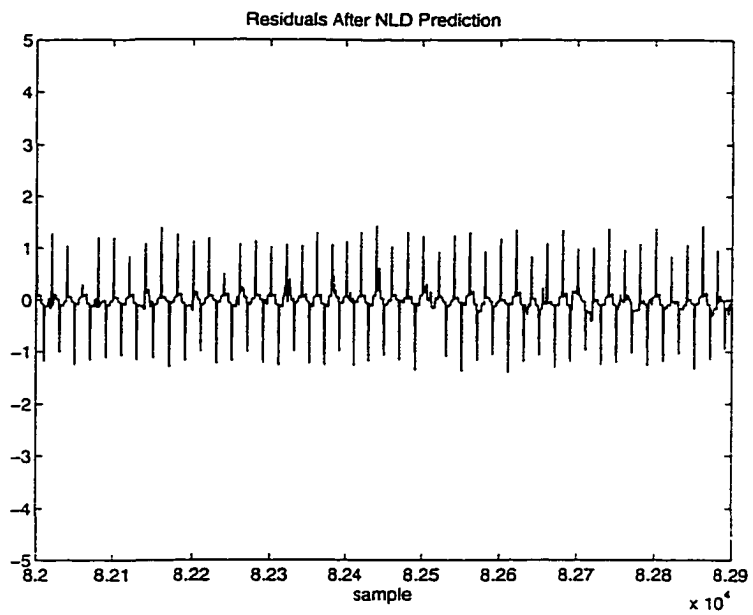


Figure 2-3 Residuals after NLD forecasting in a well-represented region of the attractor.

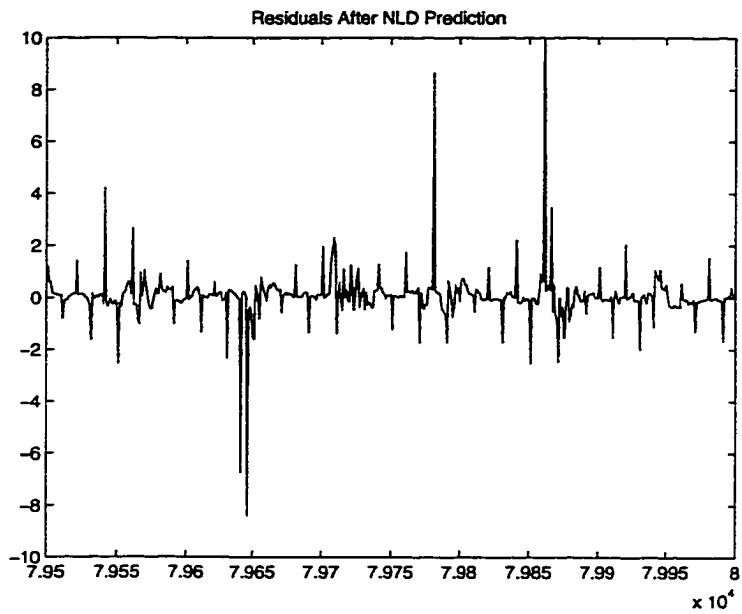


Figure 2-4 Residuals after NLD forecasting in a poorly represented region of the attractor.

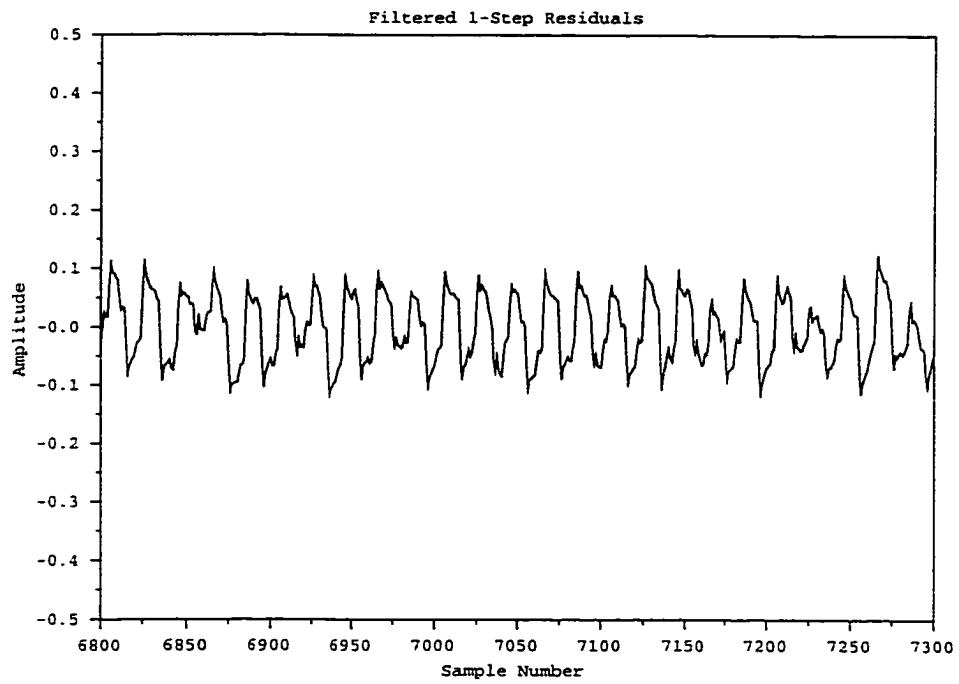


Figure 2-5 Filtered residuals.

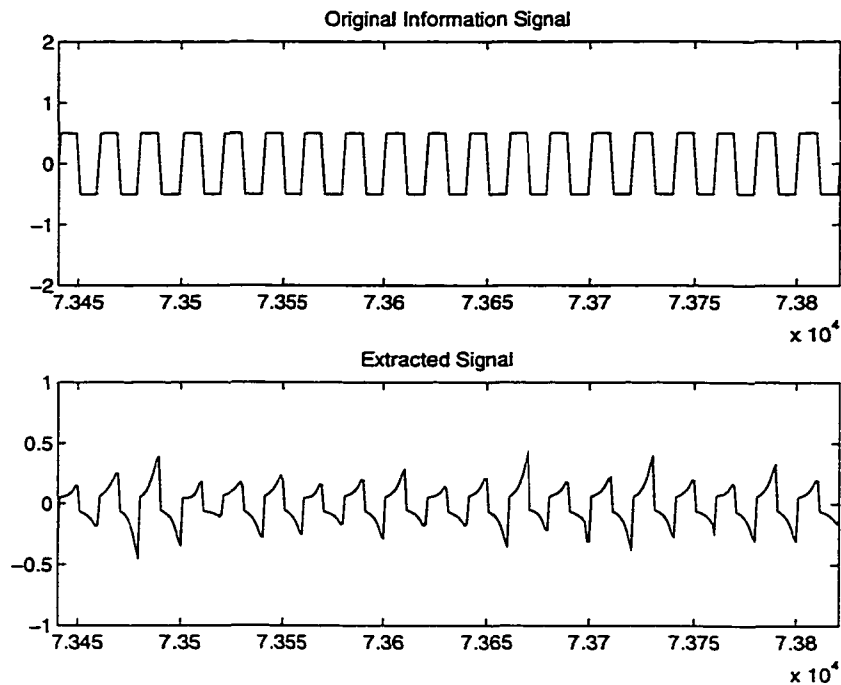


Figure 2-6 Original square wave and results from multi-step predictions.

function \mathbf{F}_1 is calculated based on the dynamics in a neighborhood of $\tilde{s}_{t_0+\Delta t}$ rather than $s_{t_0+\Delta t}$, and the next prediction is $\tilde{s}_{t_0+2\Delta t} = \mathbf{F}_1(\tilde{s}_{t_0+\Delta t})$. This continues for n iterations, corresponding to the length of the block, at which time \mathbf{F}_n again is based on a true data point $s_{t_0+n\Delta t}$ and the process is repeated.

Figure 2-6 shows the original square wave and the extracted square wave. It is clear that the information content of the extracted signal is the same as the original; however, the flat tops of the square waves are distorted because of the local divergence of trajectories. It is notable that in the 3-dimensional reconstruction, the local divergence is very strong, so the hyperchaotic nature of the chaos does have a significant effect.

To test the capability of the NLD forecasting on a signal carrying a speech message, a voice trace of the phrase, “testing, one, two, three . . . , testing, one, two, three,” sampled at

22050 Hz was used as the information signal in the modulated hyperchaotic communication scheme. The original voice trace is shown in Fig. 2-7. The transmission from the transmitter to the receiver was then modulated by the voice, where the speech begins at sample 95850 (although a lead-in hiss begins at point 83850). The resulting signal is shown in Fig. 2-8. The three-dimensional time-delay reconstruction is nearly identical in appearance to Figure 2-2. NLD forecasting was then used to predict the background dynamics. The extracted speech appears in Fig. 2-9, and although there is some error introduced by the forecasting process, it does not interfere with intelligibility, and all listeners found it easy to understand the speech. For the example in Figs. 2-7-2-9, the data was scaled so that the range of the chaos was $[-25.9, 25.1]$ and the maximum amplitude in the speech was 1.74. However, the speech extraction could be achieved for a wide range of voice amplitudes. since we were able to extract intelligible speech for maximum speech amplitudes of 0.9, 0.32, and 0.16. At amplitudes much greater than 1.74, the speech can be heard unaided in the chaotic transmission. Consequently, there does not appear to be a range of values of the speech power which yields a secure communication system.

2.2.3 Discussion

While the test information signals used here included a quiet region to improve the local modeling, several trials were also done where the message was present throughout the signal. This had the effect of enlarging some of the residuals in the extracted signals. resulting in only a few more potential bit errors in the square wave, and only a slight reduction in intelligibility in the voice signal.

Two important points must be emphasized here. First, in this hyperchaotic circuit it

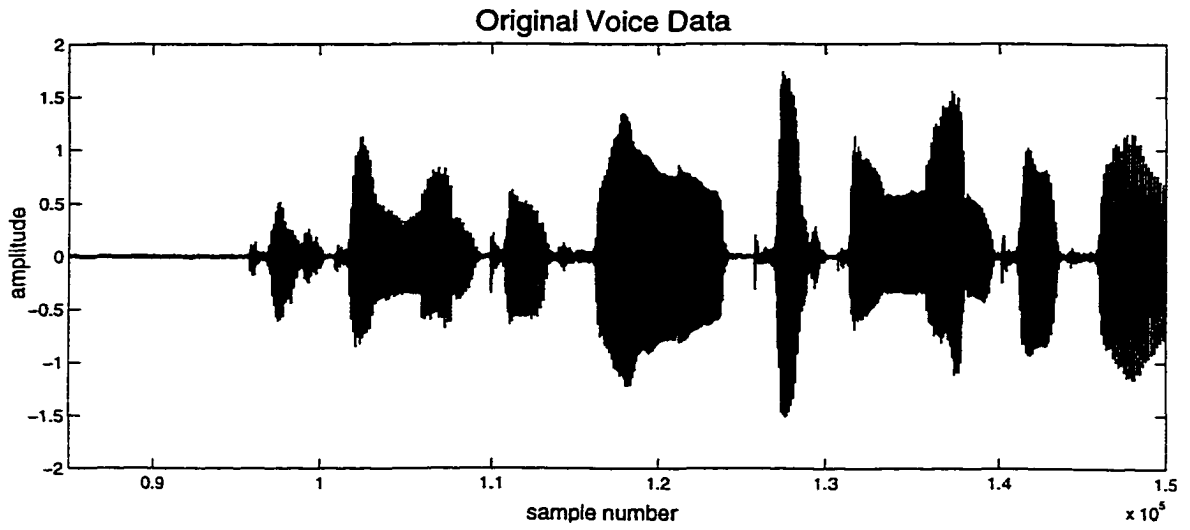


Figure 2-7 Original voice trace (“testing 1-2-3...”).

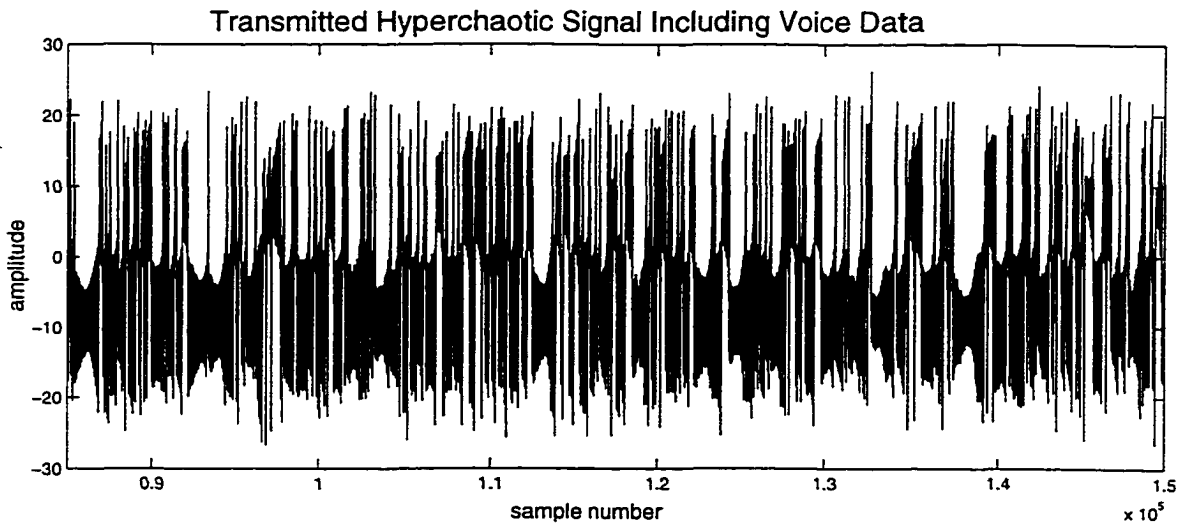


Figure 2-8 Hyperchaotic signal carrying voice data.

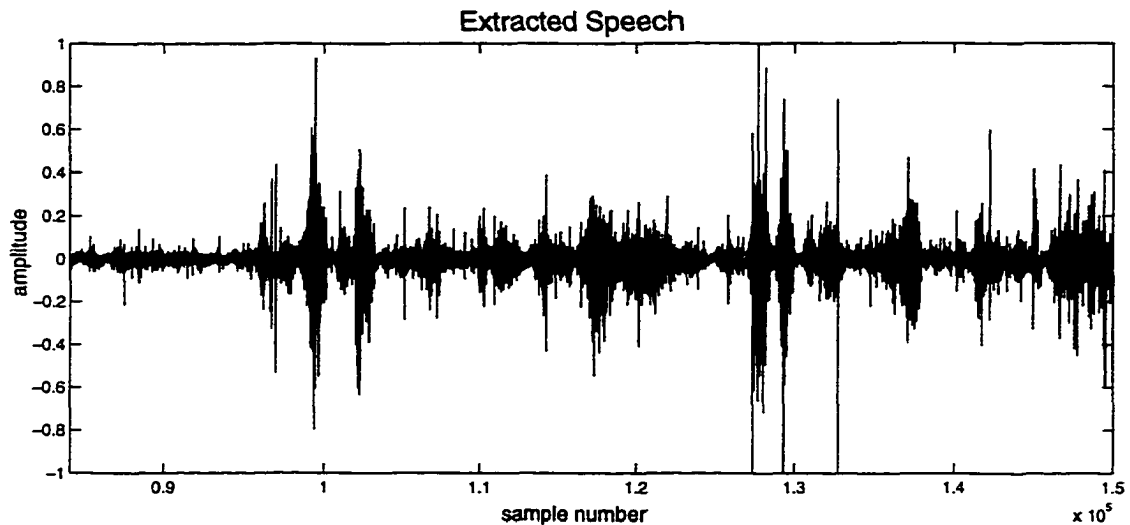


Figure 2-9 Extracted speech.

appeared that the information signal was buried deeply in the dynamics of the transmitter and consequently modulated the transmitter dynamics. However, the resulting perturbations apparently resembled the original message enough that the extracted message was easily recognizable as a human voice, and the words were clearly understood. Thus a suggestion for future work might be to study the manner in which an information-bearing signal modulates the carrier dynamics, to have the resulting perturbations differ significantly from the original message.

The second and perhaps more important point is that high-dimensional chaotic systems do not necessarily exhibit an attractor which is locally high-dimensional. Just as the 3-D Lorenz attractor is locally approximated by a 2-D surface, the six-dimensional hyperchaotic system studied in this section is also suspected to have a locally low-dimensional attractor, which makes the NLD forecasting process much easier. Thus it is important when designing a high-dimensional chaotic circuit to determine if the local dynamics are also high-dimensional.

2.3 Reconstructing the key stream from a chaotic encryption scheme

2.3.1 Introduction

The results in the previous section and in [54, 52, 44, 45, 48] show that it is possible to use NLD forecasting to break keyless communication schemes which are based on the synchronization property of chaotic systems. Consequently, researchers have developed chaotic cryptographic techniques which attempt to foil the NLD forecasting attack by using key systems which increase the sensitivity to modeling errors [24, 63]. That is, achieving synchronization is only the first step in the recovery of a message: an additional decryption stage, which depends on a secret “key” or *key stream*, is required to extract the message. The encryption/decryption process is made to be sensitive to errors in this key stream. The key stream does not have to be communicated to the receiver because it can be dynamically reconstructed from the state of the receiver, once synchronization has been achieved. Nonetheless, it will be shown that it is possible, although computationally difficult, to take an intercepted chaotic transmission and recover the plaintext message with good accuracy.

2.3.2 Hybrid Communication System Using Encryption and Synchronizing Chaos

One of the more interesting examples of this type of system is the hybrid chaotic communication scheme developed by Yang, Wu and Chua [63]. This new scheme combines key-based nonlinear data encryption with chaotic communication. As mentioned above, the crucial development is that the message is encrypted with one component of a chaotic signal playing the role of the key stream, but this component is *not* transmitted to the receiver. However,

the component that is transmitted can be used by the receiver to recreate the key stream, allowing decryption to occur. This design is intended to thwart an eavesdropper, since the information necessary to decode the message is not in the transmission.

This scheme is applied to the double scroll oscillator, which is described by the equations

$$\begin{aligned}\frac{dv_1}{dt} &= \frac{1}{C_1}[G(v_2 - v_1) - g(v_R)] \\ \frac{dv_2}{dt} &= \frac{1}{C_2}[G(v_1 - v_2) + i_L] \\ \frac{di_L}{dt} &= \frac{1}{L}[-v_2]\end{aligned}\tag{4}$$

where

$$g(v) = \begin{cases} m_1 v, & \text{if } -B_p \leq v \leq B_p; \\ m_0(v + B_p) - m_1 B_p, & \text{if } v \leq -B_p; \\ m_0(v - B_p) + m_1 B_p, & \text{if } v \geq B_p. \end{cases}$$

Parameter settings are $C_1 = 5.56$, $C_2 = 50$, $G = .7$, $L = 7.14$, $m_0 = -0.5$, $m_1 = -0.8$, and $B_p = 1$. The transmitted signal is obtained by $v_R(t) = v_1(t) - e(p(t))$, where $p(t)$ is the plain text signal and $e(p(t))$ is the encrypted plain text signal. The encryption function is

$$e(p(t)) = \underbrace{f_1(\cdots f_1(f_1(p(t), v_2(t)), v_2(t)), \dots, v_2(t))}_n$$

where f_1 is the piecewise linear function

$$f_1(x, k) = \begin{cases} (x + k) + 2h, & \text{if } -2h \leq (x + h) \leq -h; \\ (x + h), & \text{if } -h < (x + h) < h; \\ (x + h) - 2h, & \text{if } h \leq (x + h) \leq 2h. \end{cases}$$

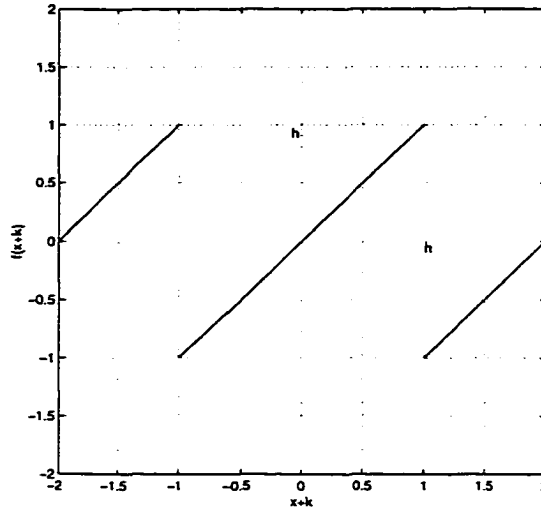


Figure 2-10 Encryption function.

This function may be seen in Fig. 2-10. The parameter h must be chosen such that both $p(t)$ and $v_2(t)$ lie in $(-h, h)$. For this example $h = 0.4$ and $n = 30$. When the sinusoidal signal in Fig. 2-11a is used for $p(t)$, the cipher function produces the sawtooth-like wave in Fig. 2-11b which is then used to find v_R by $v_R = v_1 - e(p(t))$, as shown in Fig. 2-11c.

2.3.3 Recover Encrypted Signal

The assault on this hybrid scheme proceeded in stages. First, a reconstruction of the transmitted signal v_R was done to reveal the underlying structure, as seen in Fig. 2-12. While the random-like behavior of the signal makes NLD forecasting difficult, a low-pass filtered version \bar{v}_R shows the underlying structure quite well, as seen in Fig. 2-13. It turns out that this is a sufficiently accurate estimate of v_1 for the encrypted plain text signal to be recovered from the difference $\bar{e}(p(t)) = v_R(t) - \bar{v}_R(t)$. Consequently, while the transmission will deny the interceptor the proper chaotic signal, it may give access to the encrypted signal.

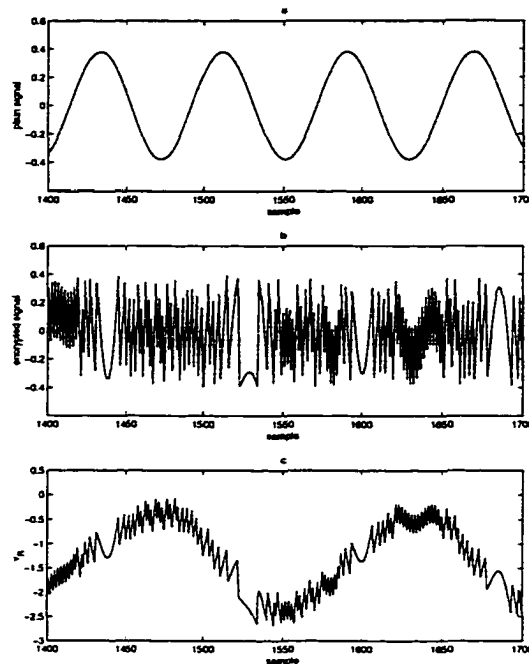


Figure 2-11 Stages in a sample transmission.

Thus the problem really reduces to the well-established category of breaking an encrypted message without access to the key stream. So, the strength of the scheme would seem to lie in the strength of the encryption and the fact that the transmitted information allows the remote receiver to reconstruct the key stream, with the intent to deny the eavesdropper the same opportunity. Both the original encrypted signal and this estimate are shown in Fig. 2-14. For a measure of progress at this stage, the author assumed knowledge of the key signal $v_2(t)$ and the encryption function. The estimated encrypted signal $\tilde{e}(p(t))$ could then be fed into the decrypting function which is the same as the encrypting function. The result is shown in Fig. 2-15, where the smoother line indicates the original plain text signal. Notice that there is a clear correlation between the estimated and the original signals, even though noise introduced by filtering was amplified by the decryption process.

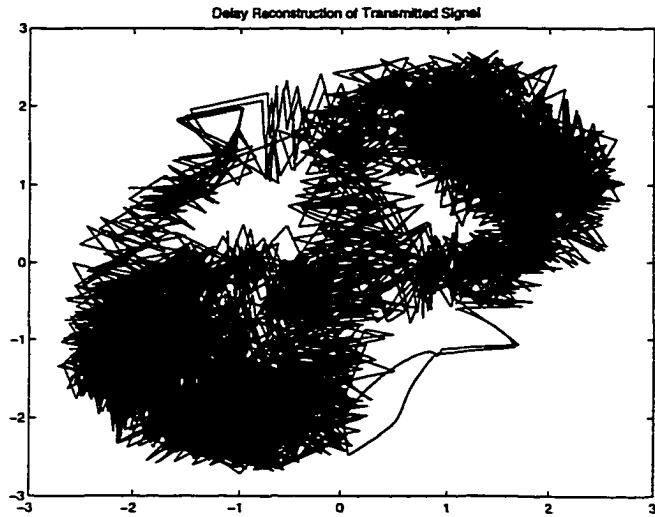


Figure 2-12 Reconstructed intercepted signal.

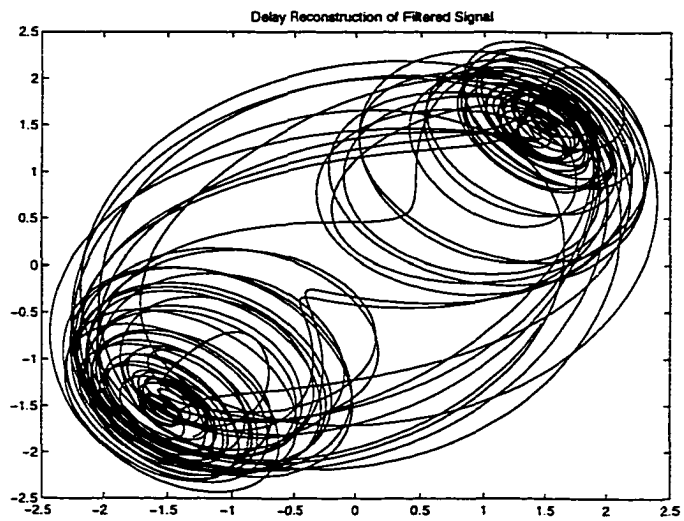


Figure 2-13 Reconstructed low-pass filtered signal.

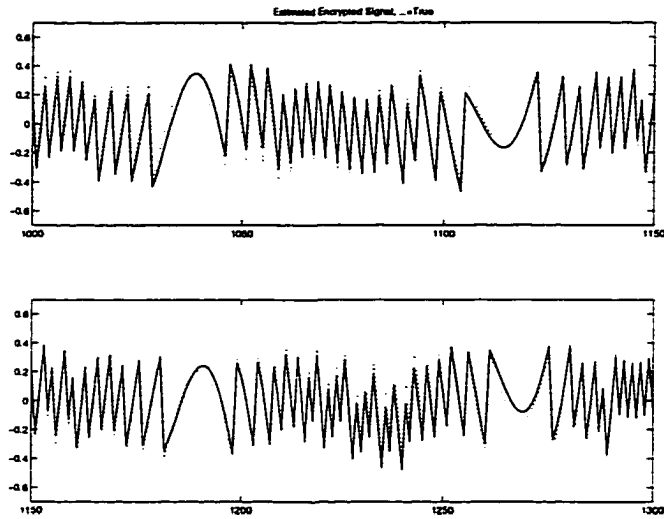


Figure 2-14 Estimated encrypted signal: dotted line represents the original.

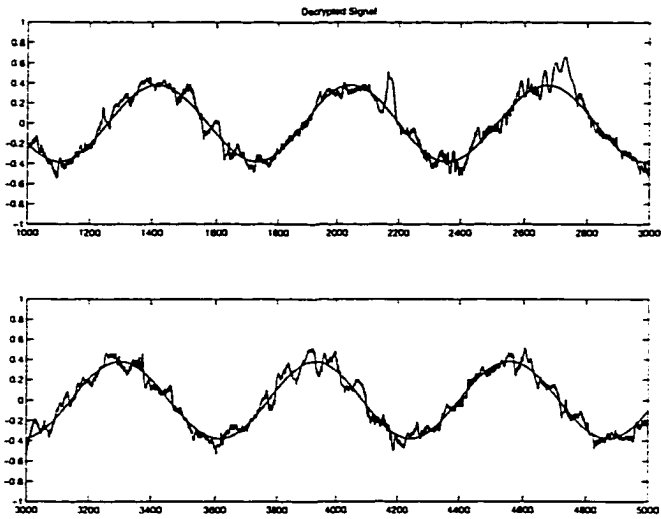


Figure 2-15 Decryption using estimated encrypted signal.

To proceed with this problem and decrypt the message, one still needs to know the encryption technique and the key stream, but the key stream is not included in the transmission. Rather, what is transmitted allows the receiver to reconstruct the key stream once synchronization is achieved, assuming that the true equations of motion are known by the receiver. The idea is that an eavesdropper would not be able to perform the key reconstruction step of the process, since there would be (presumably) a large space of functions from which to choose. So, the eavesdropper may be able to reconstruct the ciphertext, but would not have access to the key stream, leaving a ciphertext-only attack which can presumably be made difficult. However, in the next section, a method of reconstructing the key stream from only an intercepted signal will be presented.

2.3.4 Recover Key Stream

The next stage of progress came when it was shown that an intruder can use the intercepted signal alone to reconstruct the key stream with enough accuracy to partially decrypt the plaintext signal, without knowledge of the carrier system. This is a much harder problem, because the encrypting and decrypting function magnifies errors in the key stream. The true key stream is shown in Fig. 2-16a. From the delay reconstruction in Fig. 2-13, it can be inferred by the intercepting party that the carrier system has a double-scroll shaped attractor. Assuming that the intruder has some prior knowledge about this type of encryption scheme, it can be expected that he or she will know that the key stream will be some component of the chaotic system other than what is transmitted. To estimate v_2 it was clear that one needed to rotate and project the delay reconstruction in a way that produced a time series similar to the true v_2 . To accomplish this a reconstructed trajectory matrix R was created where each row

i is given by $R_i = \begin{bmatrix} \bar{v}_R(i) & \bar{v}_R(i + \tau) & \bar{v}_R(i + 2\tau) \end{bmatrix}$. The singular value decomposition on R returns a matrix V which can be used to rotate the reconstructed attractor along principal axes. Let the rotated coordinates be given by $\tilde{R} = RV$. The resulting projections are shown in Fig. 2-16b-d. It turns out that a simple linear combination of these principal components does quite well in approximating the key stream v_2 . An example where $\bar{v}_2 = -0.3\tilde{R}_{.1} + 0.5\tilde{R}_{.3}$ is shown in Fig. 2-16e. In practice, the correct rotation would be difficult to find, so one presumably would be forced to do an extensive search. However, partial decryptions result when the estimated key stream is close to the original, so the search is directed.

This estimated key stream can now be tested by feeding it into the (assumed known) decrypting function with the estimated encrypted signal. As one measure of progress, the frequency spectra of the message, transmitted and extracted signals may be compared. The spectra of the transmitted signal and the plain signal appear in Fig. 2-17. Notice that there is no spectral evidence of the message in the transmission. The spectrum of the decrypted signal appears in Fig. 2-18. The presence of the sine wave clearly has been revealed. The equivalent signals in the time domain appear in Fig. 2-19.

2.3.5 Recover Encryption Function

The final stage of the analysis of the hybrid chaotic communication scheme involves developing methods for determining the encryption function by comparing the reconstructed key stream to the encrypted signal. Similar weaknesses in encrypting with chaotic maps have been discussed by Zhou *et al* in [64, 65]. The shift map for this system was seen in Fig. 2-10. Notice that it simply sums the arguments and wraps the sum into the interval $[-h, h]$. Assuming for the moment that the plaintext is zero, the effect of iterating this map

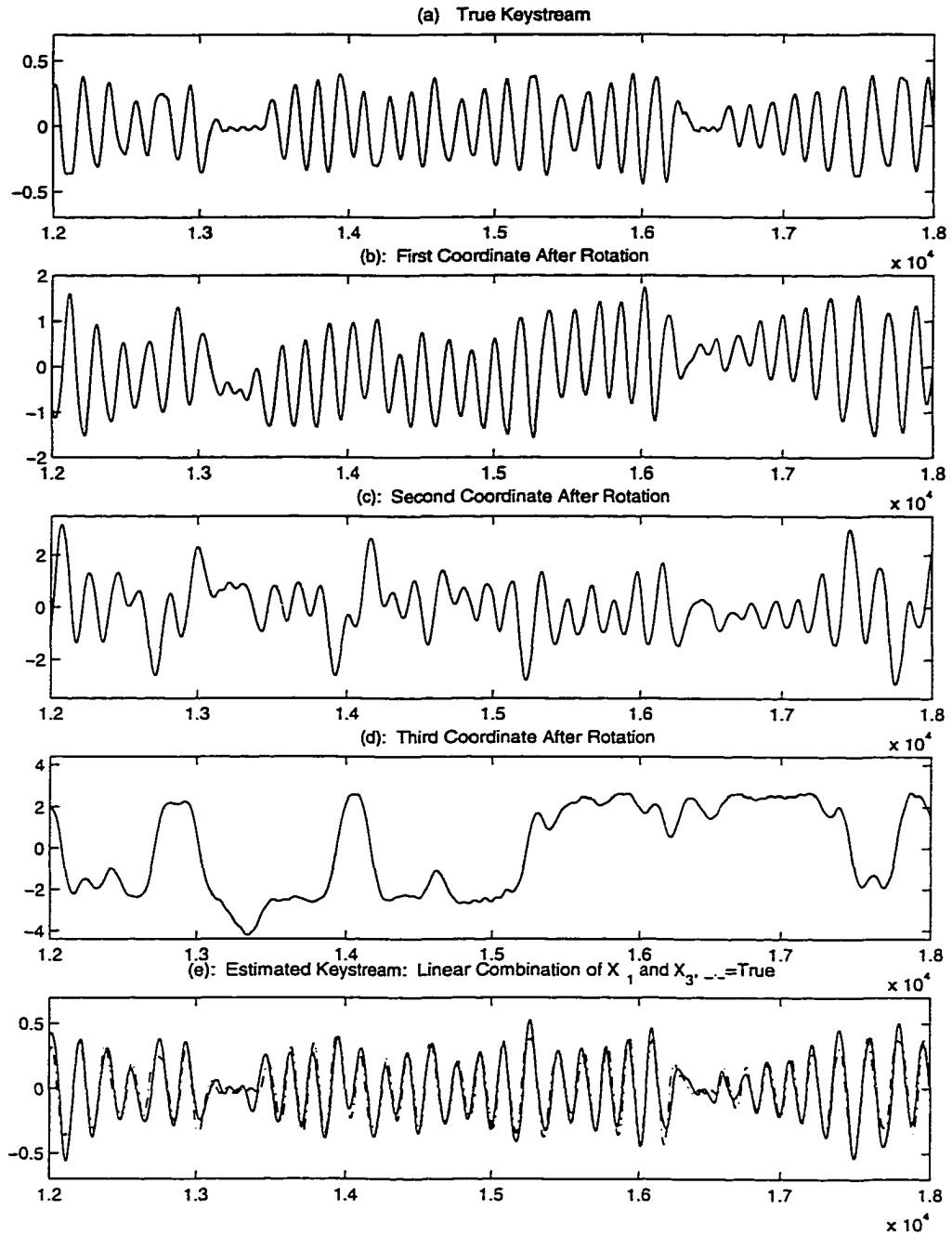


Figure 2-16 Singular value decomposition of intercepted signal.

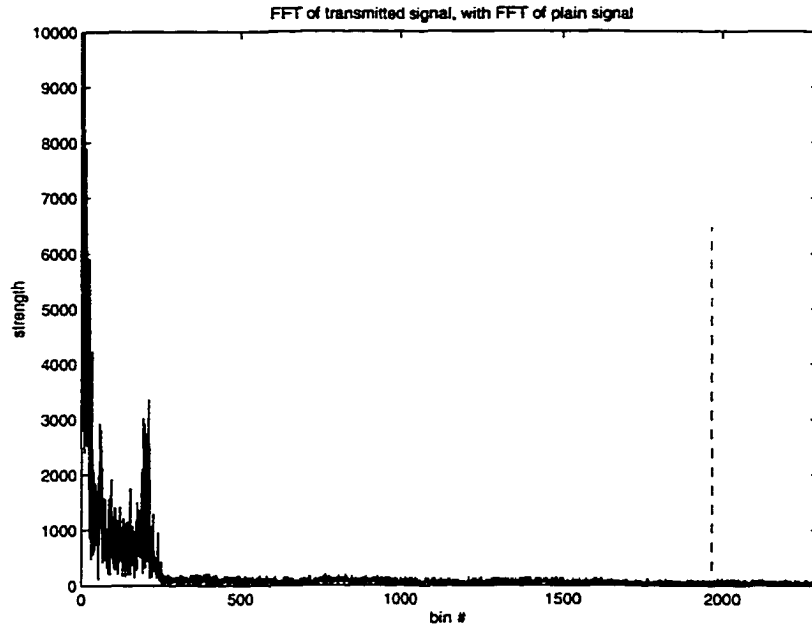


Figure 2-17: Frequency spectrum of transmitted signal: dotted line indicates the spectral peak of the message signal.

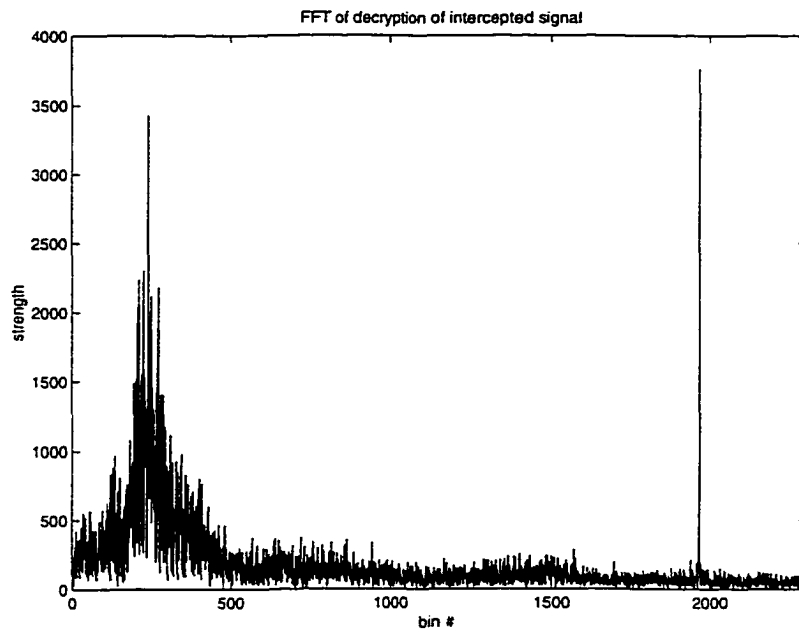


Figure 2-18 Frequency spectrum of decrypted intercepted signal.

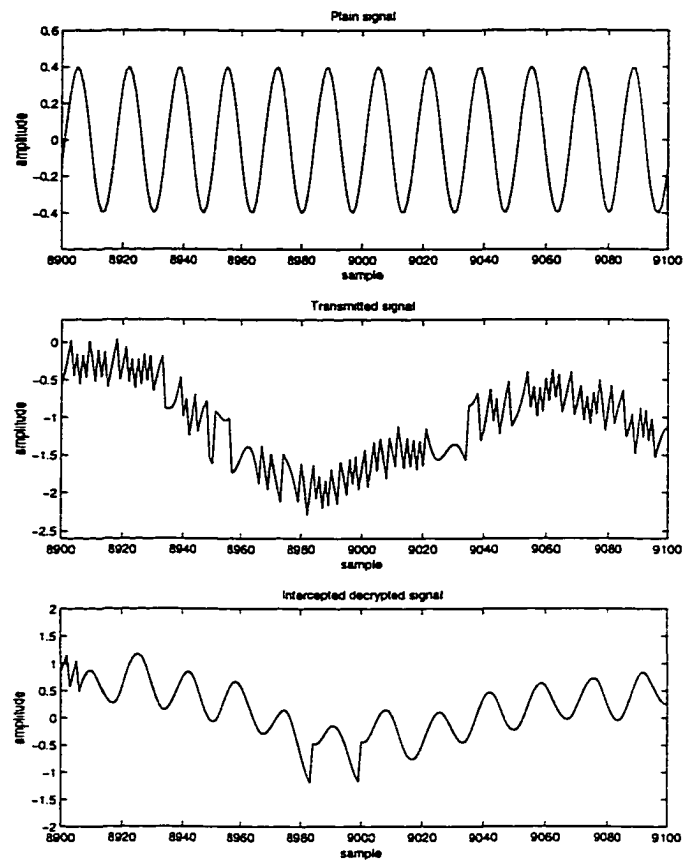


Figure 2-19 Signals in the time domain.

in the encryption function $e(p(t))$ is that of adding the key stream v_2 to itself n times, while wrapping around the interval $[-h, h]$. Suppose two adjacent samples in the key stream differ by a small amount δ . Then, the slope of the line connecting these points is $\frac{\delta}{\Delta t}$. After iterating the function f_1 as above (with a zero plaintext) these points will be separated by $n\delta$, modulo the interval $[-h, h]$. If these points are on the same (continuous) segment of the encrypted signal, the slope of the line connecting these adjacent points is $\frac{n\delta}{\Delta t}$, or n times the slope of the line joining the original points in the key stream. Therefore a comparison of slopes of the segments in our estimated encrypted signal to the corresponding slopes in the estimated key stream provides an approximation to the number of iterations n in the cipher function. The effect of the plain signal, since it is only added in once during the encryption, is negligible, especially if it has a mean of zero.

A more common and slightly more complex case would be when the linear segments in the continuous shift cipher f_1 had a slope p greater than one in magnitude. Then the information obtained by the above comparison should approximate the value np , and an assumption must be made about either n or p . The authors of [64, 65] propose a method of choosing n large enough such that there are no adjacent pairs of points on the same segment of the encrypted signal, thus destroying any information about n , p , and the shift cipher f_1 . This would foil the approach described here.

To extract the characteristics of the shift cipher function f_1 , the estimates of the value np were obtained by searching for points on the same linear segments of the cipher $\bar{e}(t)$ and their corresponding points in the estimated key stream $\bar{k}(t)$. Then an estimate for the value

np was obtained by

$$np \approx \frac{\bar{e}(t_i) - \bar{e}(t_{i-1})}{\bar{k}(t_i) - \bar{k}(t_{i-1})}.$$

An average value was taken over the entire time series. Preliminary trials yielded estimates between 25 and 35 for $n = 30$ and $p = 1$. More precision in these estimates would likely be attained by calculating weighted averages based on the actual distribution function of the quotient.

The final parameter necessary in determining encryption functions of this type is the cutoff h . This parameter can be estimated by looking at the mean range of the estimated encrypted signal. An average of the peak values of this time series gives a reasonable approximation for h . Also, in the case presented in [63], the value of h is chosen to reflect the range of the key stream, although this is not necessary. Therefore, in this case the range of the estimated key stream will provide a rough estimate of h as well.

2.3.6 Trial on Voice Data

So far our trial plaintexts have only consisted of sine wave signals. To test the ability of these eavesdropping techniques on speech, $p(t)$ was set equal to a voice trace of the words "testing, 1-2-3, testing, 1-2-3." The results are shown in Fig. 2-20. The first plot is of the message signal $p(t)$. The transmitted signal is shown in the second plot. The original speech is preceded by a moment of silence, during which the transmitter seems to be driven onto a periodic orbit. This is surprising behavior for a chaotic system, and must be related to the modulation of the system by the encrypted signal. It is important to remember that feeding the system a null message signal still results in a nonzero encrypted signal, since the

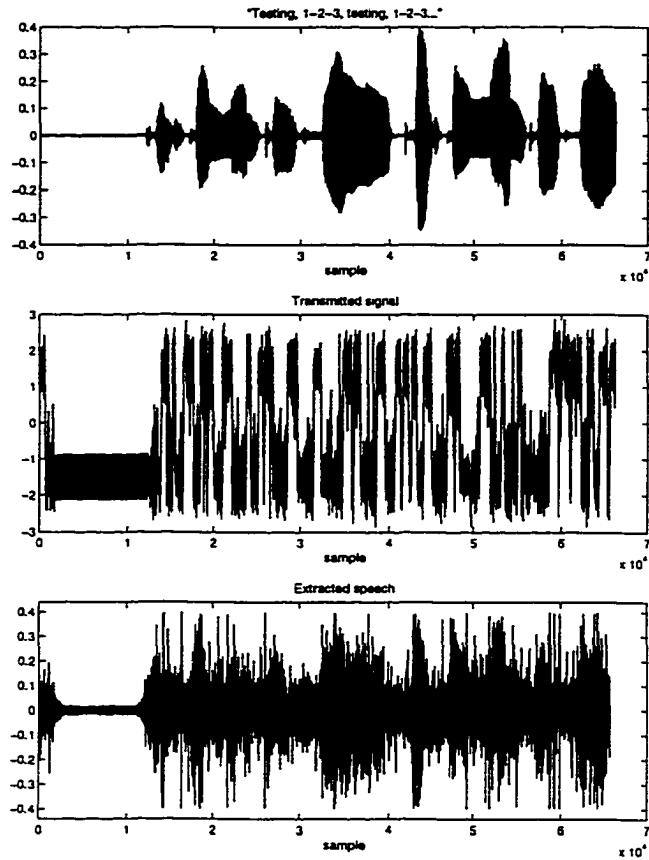


Figure 2-20 Results using speech data.

encryption stage relies heavily on $v_2(t)$. To see if this periodic orbit reappeared occasionally during the actual speech, the spectrum of the transmitted signal in the periodic region was compared to spectra taken from other portions of the signal, and no evidence of the orbit was found elsewhere. The results from the attempt to decrypt the message are shown in the last plot. While the extracted voice signal looks noisy, the words are clear and intelligible.

2.3.7 Discussion

It is clear now that the chaotic encryption system developed by Yang *et al* reveals information which may cause security weaknesses. The fact that one can reconstruct the key stream from the transmitted signal alone is problematic, since the main advantage of the scheme was in the key stream's secrecy. But there are also several clues left about the cipher function itself, so that all of the components of this encryption system can be reproduced from only one brief transmission. Although the cipher function in the cases studied above was quite simple, a motivated intruder is likely to determine characteristics of a more complex function in a way similar to what has been presented here.

2.4 NLD detection of controls

2.4.1 Introduction

The primary focus of chaotic communication schemes discussed above has been the use of synchronizing chaotic circuits to produce a communication channel where the receiver could be made to synchronize with the transmitter. Then, a message signal may be encoded into the transmission, and may be recovered without the need to exchange keys [37, 38, 5, 10, 12, 16, 18, 23, 36, 60, 22]. A different approach to chaotic communication is developed by Hayes, Grebogi and Ott (HGO) [17], in which they use a new way of controlling chaos to transmit binary information. The scheme controls the double scroll oscillator using *small* perturbations to follow a "prescribed symbolic sequence" which contains the encoded information. Although this was not intended to be a secure communication method, the design of the system attempts to hide the presence of the controls in the transmitted signal.

Since this is an innovative control technique, it becomes an important question whether NLD techniques can be successful in detecting evidence of a control mechanism in the transmitted signal. Here we show that, in the HGO implementation, the controls are detectable, compromising security. However, this control method will in fact be an important part of the digital chaotic communication scheme which will be developed in Chapter 4.

2.4.2 Control method

Any chaotic attractor may be used, but for illustrative purposes, the technique will be developed for the double scroll chaotic attractor. Recall that this system evolves according to the equations

$$\begin{aligned}\frac{dv_{C_1}}{dt} &= \frac{1}{C_1}[G(v_{C_2} - v_{C_1}) - g(v_{C_1})] \\ \frac{dv_{C_2}}{dt} &= \frac{1}{C_2}[G(v_{C_1} - v_{C_2}) + i_L] \\ \frac{di_L}{dt} &= \frac{-1}{L}v_{C_2},\end{aligned}\tag{5}$$

where

$$g(v) = \begin{cases} m_1 v, & \text{if } -B_p \leq v \leq B_p; \\ m_0(v + B_p) - m_1 B_p, & \text{if } v \leq -B_p; \\ m_0(v - B_p) + m_1 B_p, & \text{if } v \geq B_p. \end{cases}$$

The parameters used were the same as those used by Hayes *et al*: $C_1 = \frac{1}{9}$, $C_2 = 1$, $L = \frac{1}{7}$, $G = 0.7$, $m_0 = -0.5$, $m_1 = -0.8$, and $B_p = 1$. A plot of a long trajectory appears in Figure 2-21. It may be seen in this figure that the attractor has two lobes, each of which surrounds an unstable fixed point. It is easy to show that these fixed points are at $(v_{C_1}^*, v_{C_2}^*, i_L^*) = (\pm \frac{(m_0 - m_1)B_p}{G + m_0}, 0, \pm \frac{(m_1 - m_0)B_p G}{G + m_0})$.

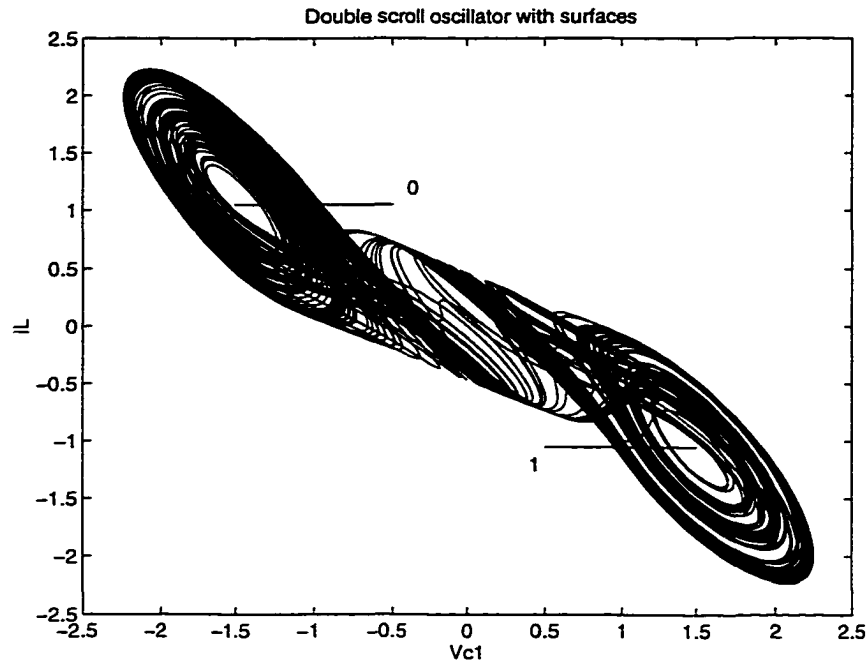


Figure 2-21 Double scroll oscillator showing surfaces.

Because of the chaotic nature of this system's dynamics, it is possible to take advantage of sensitive dependence on initial conditions by carefully choosing small perturbations to direct trajectories around each of the loops of the attractor. In this way, a desired message bit stream can be transmitted by steering the trajectories around the appropriate lobes of the attractor, suitably labeled 0 and 1, and then transmitting one of the variables. At the receiver end, the message is just read from the peaks of the transmission, where a positive peak would indicate a 1 and a negative peak would correspond to a 0 (or *vice versa*). An example of this appears in Fig. 2-22.

The generation of the desired transmission requires a control mechanism for the transmitter. A useful technique in studying the qualitative behavior of an n -dimensional continuous chaotic system is to observe the intersection of trajectories along an $n - 1$ dimensional sur-

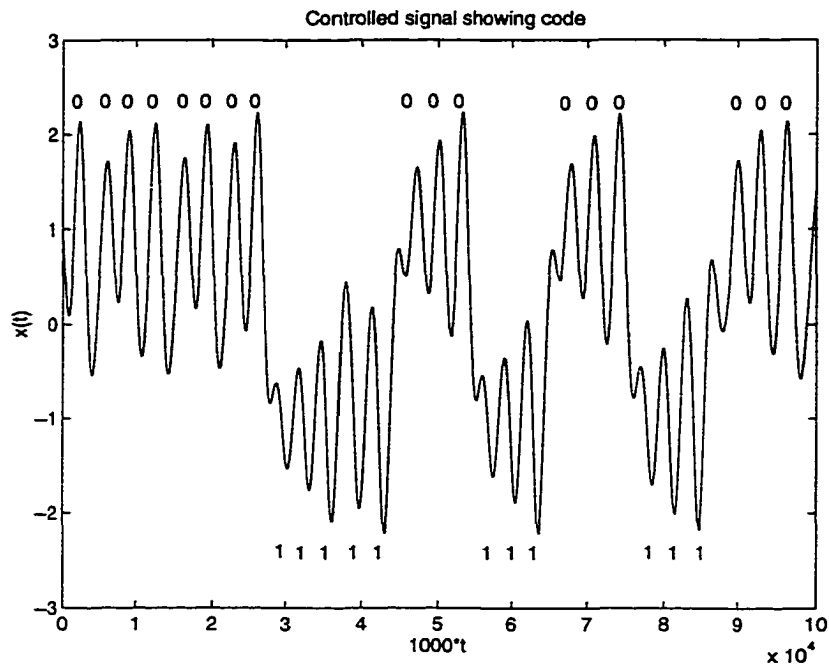


Figure 2-22 Controlled signal showing information bit stream.

face which intersects the attractor, called a *Poincaré surface of section*, or more simply a Poincaré section. So for this system, to specify the bits “1” and “0” more precisely, a Poincaré section is defined on each lobe by intersecting the attractor with the half-planes $i_L = GF$ with $v_{C_1} \geq F$, and $i_L = -GF$ with $v_{C_1} \leq F$, where $F = B_p(m_0 - m_1)/(G + m_0)$. The edge of each half-plane intersects the fixed point at the center of each lobe. One half-plane is labeled “1” and the other is labeled “0”. This is indicated in Fig. 2-21. When a trajectory intersects one of these sections, the corresponding bit can be recorded. Then, a function $r_N(x)$ is designed to take any point on either section and return the future symbolic sequence of length N for trajectories passing through that point. If l_1, l_2, l_3, \dots represent the lobes that are visited on the attractor (so l_i is either a 0 or a 1), and the future evolution of a given point x_0 is such that $x_0 \rightarrow l_1, l_2, l_3, \dots, l_N$ for some number N of loops around

the attractor, then the function $r_N(x)$ is chosen to map x_o to an associated binary fraction, i.e. $x_o \xrightarrow{r_N} 0.\ell_1\ell_2\ell_3\dots\ell_N$. Then, when $r_N(x)$ is calculated for every point on the cross-section, the future evolution of any point on the cross-section is known for N iterations. The resulting function has the interesting fractal-like structure seen in Fig. 2-23, where $r_N(x)$ has been calculated for $N = 12$.

To control the trajectory, wait for it to pass through one of the sections, say at x_0 . Now, the value of $r_N(x_0)$ tells us the future symbolic sequence followed by the current trajectory for N loops. If transmission of our desired message bit stream requires a different symbol in the N th position of the sequence, search $r_N(x)$ for the nearest point on the section which will produce the desired symbolic sequence. As long as the trajectory has been controlled for at least N loops, only the N th bit will need to be changed. Simply perturb the trajectory to this new point, and let the system continue to its next encounter with a surface. Repeat this procedure until all desired information has been transmitted. Notice that it requires N loops for any given message bit to appear in the transmitted signal.

It should be noted that trajectories on the double scroll attractor exhibit a "limited grammar," which means that not all sequences of 1's and 0's can be directly encoded. This is because trajectories always loop at least twice in succession around each lobe. For example, a sequence of bits containing 00100 is not in the grammar since it requires a single loop around the 1-lobe. A simple remedy is to repeat every bit in the code or append a 1- or 0-bit to each contiguous grouping of 1- or 0-bits, respectively. Obeying the grammar of the chaotic system is required to guarantee that searching $r_N(x)$ for a new position as above will be successful. This grammar limitation is the cause of the large discrete jumps in $r_N(x)$ seen in Fig. 2-23.

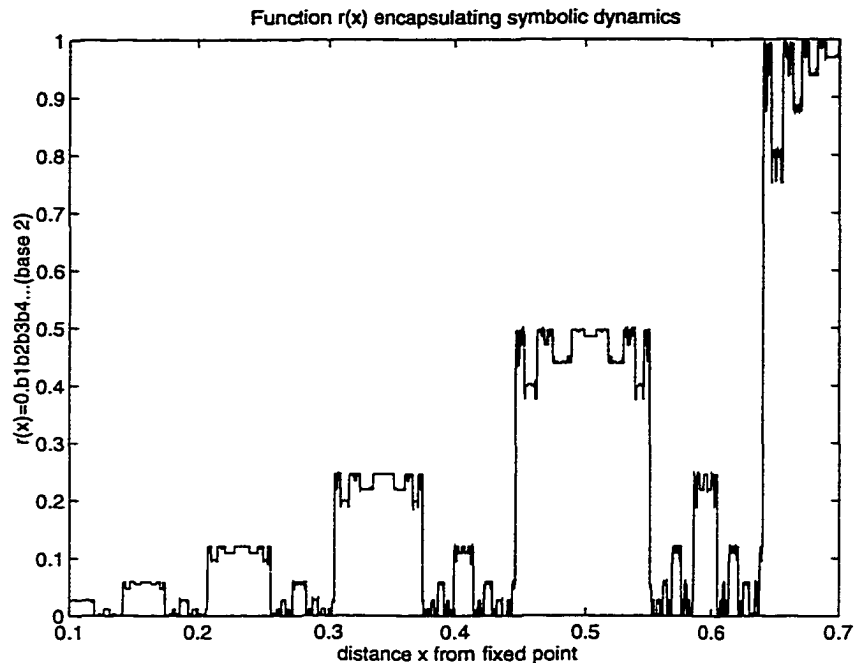


Figure 2-23 Function $r_{12}(x)$

For this system, the actual transmitted signal is the coordinate i_L , so the message bit stream is simply read off from the peaks and valleys in i_L (there are small loops/minor peaks which occur as the trajectory is switching lobes of the attractor, but these are ignored). An important point to notice is that the perturbation is applied at constant i_L , so there is no discontinuity in the transmitted trajectory.

2.4.3 Detection of Controls

To analyze the effects and detectability of these controls, a simulated controlled signal was generated. The controlled transmission was then analyzed using reconstruction techniques and NLD forecasting [44, 48]. Both approaches can be used to detect the controls [49].

The analysis of most chaotic communication schemes usually begins with a reconstruc-

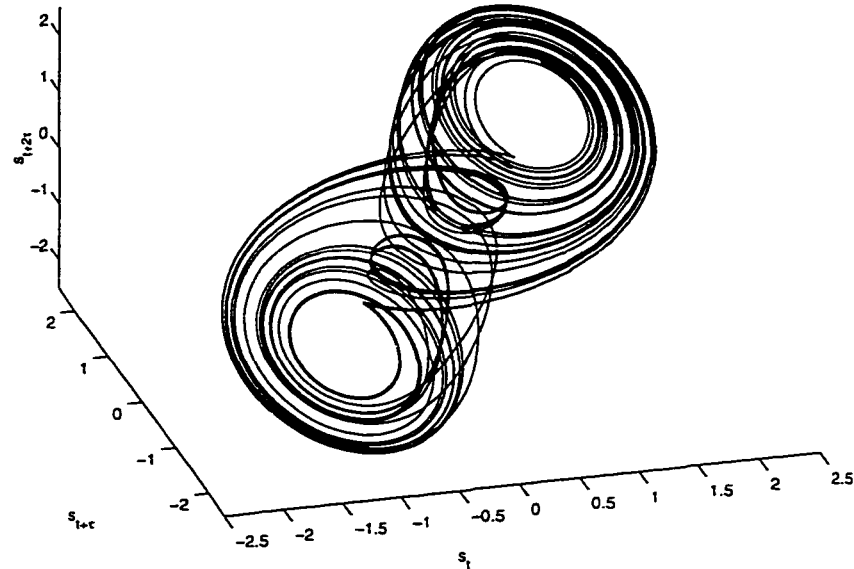


Figure 2-24 Time-delay reconstruction of controlled signal

tion of the phase space dynamics. Applying the time-delay reconstruction technique to the transmitted i_L coordinate reveals the underlying double scroll attractor seen in Fig. 2-24, but there is no sign of the controls. The perturbations in this scheme are cleverly applied in a direction perpendicular to the transmitted coordinate (i.e., at constant i_L), so no discrete jumps can be seen in the original time series.

An alternative reconstruction uses numerical derivatives with respect to time, i.e. $(x(t), \dot{x}(t), \ddot{x}(t))$, where $x(t)$ is the signal. When forming the approximations of the derivatives for this data, the first derivative of the signal does not reveal much about the controls. The second derivative, however, exhibits spikes at the moments when the controls were applied. As can be seen in Fig. 2-25, a derivative state space reconstruction shows not only that the system is being controlled, but it also shows approximately where the surface of section is on the

Derivative Reconstruction of Controlled Signal

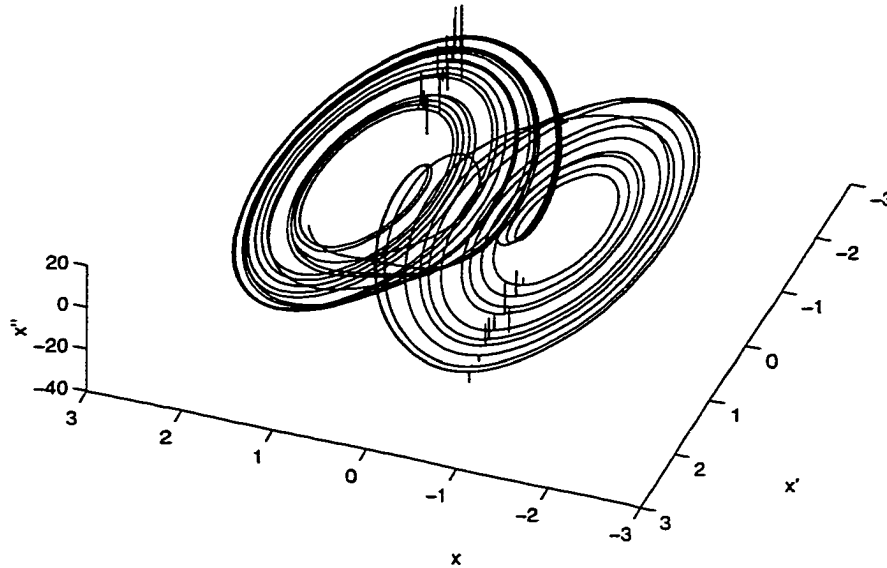


Figure 2-25 Derivative reconstruction of controlled signal

attractor. Once the controls are detected, it becomes clearer that the information is coded into the loops around the two lobes. The binary sequence can be read by inspection and the problem becomes one of interpreting the message.

Previously it has been shown that when the intercepted signal allows for a reconstruction of the phase space dynamics, NLD forecasting can often detect hidden signals in chaotic systems, even when the presence of signals is not obvious in the reconstruction. As a test for this scheme, NLD forecasting was applied to the time delay reconstruction to see if the controls can be detected. In this case, for each data point in the reconstruction, a prediction was made about its future evolution. Wherever the controls were applied, the future evolution was altered and these trajectory changes were readily detected by the algorithm, as seen in Fig. 2-26, where the sharp spikes in the residual represent places where the application of

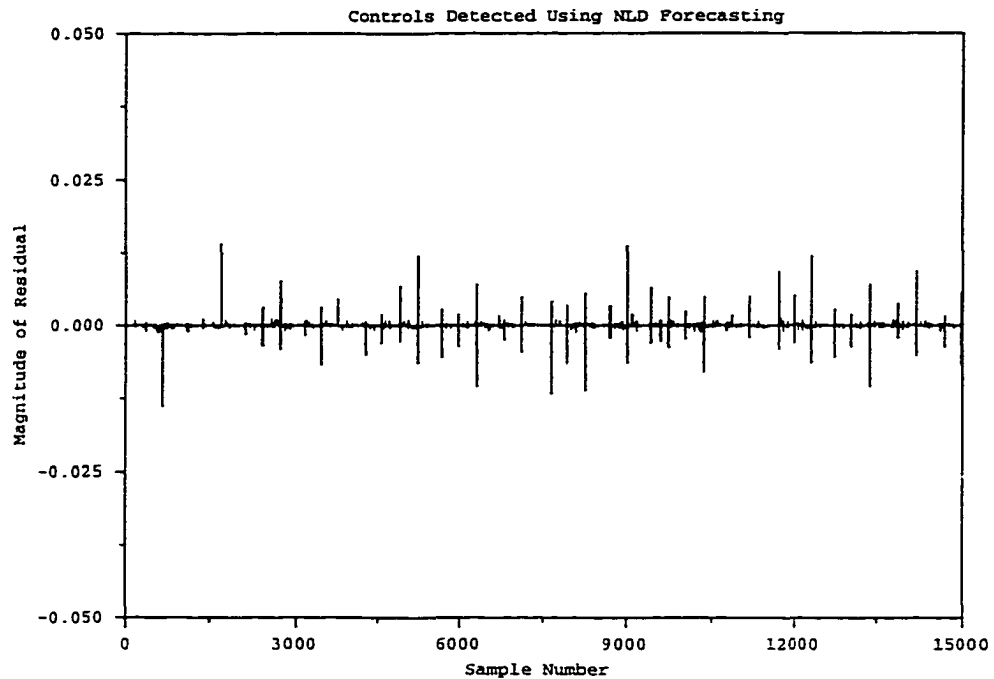


Figure 2-26 Residuals from NLD forecasting

the controls caused the dynamics to deviate from that predicted by the forecasting.

2.4.4 Discussion

Although hiding the evidence of the controls in the signal was not necessarily a primary concern to the inventors, the fact that the controlling mechanism was not obvious in the transmission made it an interesting detection problem. The results here provide further proof that any chaotic communication scheme which reveals information that may yield a reconstruction of the phase space of the transmitter is difficult to make secure. However, in Chapter 4 the HGO chaotic control method will be used to build a new digital chaotic communication which eliminates the weakness inherent in revealing the state of the transmitter.

Chapter 3

Applications of NLD forecasting techniques to nonchaotic communications problems

3.1 Introduction

Two problems that plague standard digital communication systems are bit errors and co-channel interference. Bit errors occur when message symbols are not received correctly, and are commonly caused by either added noise or distortions to the signal which occur in transit. The medium through which a signal passes, e.g. an optical fiber, a copper wire or the atmosphere, is called a *channel*, and the distortions to the signal caused by passing through these channels are called *channel effects*. For example, a sound wave traveling through a long corridor will be distorted not only by such things as differences in air temperature and movement along the corridor, but also by reflections of the sound wave off of the walls, ceiling and floor. These are examples of channel effects associated with a particular channel. All channels introduce channel effects independent of the digital modulation technique used to generate the signal. Distortions may also be introduced by various pre-transmission filtering. For example, the FCC has divided the usable radio frequency range into equal sections, or *bands*, and requires that any given radio transmission be limited to only one band. Therefore, radio signals often have to be passed through a band-limiting filter to satisfy this regulation. Distortions caused by filtering or channel effects can increase the probability that a bit will be received incorrectly. However, most channel effects may be modeled by convolving the

signal with a particular response function which represents the channel. For example, if there is a single echo introduced by the channel, such as a reflection off of a wall, the associated response function will be the sum of two delta functions, $\delta(0) + \delta(t_1)$, where t_1 is the echo delay. If the channel filters out high-frequency components in the signal, this filtering may be modeled by either a low-pass filter in the frequency domain or by a convolution with that filter's inverse Fourier transform in the time domain. See [41] for a good introduction to the principles of channel effects and response functions.

Co-channel interference occurs when two or more communication signals are received over the same or overlapping frequency bands. This problem is particularly common in the use of mobile phones when two users in adjacent cells in a cellular network are near a common border between the cells. One receiving tower in one cell may pick up the signals from both users simultaneously. Another example occurs when your vehicle is at the edge of the range of one radio station and your car radio begins receiving another station's signal before the first has completely faded out. Co-channel interference can also occur when signals are transmitted along adjacent copper wires, where the current through one wire may induce a current in another, partially transferring the signal. The goal of digital co-channel demodulation is, ideally, to separate and demodulate all of the received signals, or at least to separate the strongest signal from the interferers, with as few bit errors as possible. Fortunately, all communication systems are subject to conditions, such as channel effects or FCC mandated filtering, which introduce a certain degree of short-term determinism into the transmitted signal. This short-term determinism contains information about the recent past and near future evolution of the signal, and will be studied from a geometric viewpoint. It will be shown that this information can be useful to combat the problems of bit errors

and co-channel interference.

The digital communication techniques which will be used to generate simulated signals in this chapter will be introduced in Section 3.2. The problem of bit error correction will be addressed in Section 3.3. Dynamic co-channel demodulation will be discussed in Section 3.4. Finally, the geometric approach will be applied to a hardware-generated digital signal prepared by the National Security Agency (NSA) in Section 3.5.

3.2 Digital communication techniques

There are many different methods for modulating a carrier signal to transmit binary data. The simplest method is called *on-off keying* (OOK) where the carrier is turned on or off according to the bit stream. That is, if the bit stream b_i is a sequence of 1's and 0's, then an OOK signal may be written as $s_i(t) = b_i \cos 2\pi f_c t$ for $(i-1)T \leq t \leq iT$, where T is the length of time that one bit (or *symbol*) is transmitted and f_c is the carrier frequency. A slightly more complex technique is called *frequency-shift keying* (FSK), where the carrier signal oscillates between two frequencies according to the bit stream. This may be written as $s_i(t) = \cos 2\pi(f_c + b_i \Delta f)t$ where $-\frac{T}{2} \leq t \leq \frac{T}{2}$ and the bit stream b_i now has values of -1 and 1 . *Phase-shift keying* (PSK) encodes the bit stream in 180° phase shifts of a single carrier frequency, as in $s_i(t) = b_i \cos 2\pi f_c t$ where b_i again take on values of -1 and 1 . It may be shown that PSK has an advantage over the first two modulation techniques in combating channel noise [33].

These techniques transmit only one bit per T -second interval. A method for transmitting two bits at a time is called *quaternary PSK* (QPSK) and may be written as $s_i(t) = \cos(2\pi t + \theta_k)$, where $k = 1, 2, 3, 4$ and θ_k is a phase angle associated with one of the four possible pairs

of bits [41]. A natural mapping of these angles to the pairs of bits arises from rewriting the formula for $s_i(t)$ using the trigonometric identity for a sum:

$$s_i(t) = a_i \cos 2\pi f_c t + b_i \sin 2\pi f_c t.$$

After a scaling by a factor of $\frac{\sqrt{2}}{2}$, it may be shown that the angles $-\frac{\pi}{4}$, $\frac{\pi}{4}$, $\frac{3\pi}{4}$, $-\frac{3\pi}{4}$ correspond to the pairs (1, 1), (1, -1), (-1, -1), (-1, 1), respectively. Notice that when both a_i and b_i switch, the result is a phase shift of π radians. This maximum phase shift may be reduced by allowing only one bit to switch at a time. This may be done by offsetting one bit in the pair of bits by $T/2$ seconds, with the result that the maximum phase shift is now $\pi/2$ radians. This method is called *offset* QPSK (OQPSK).

The benefit of limiting the phase discontinuity is realized when the resulting signal is passed through a band-limiting filter to satisfy FCC requirements. Large phase discontinuities result in variations in the amplitude after band-limiting. Therefore, to maintain a stable amplitude, a digital signal with zero phase shift would be an improvement. An example of a signal with this characteristic is called **minumum shift keying (MSK)**, and because of its phase continuity it has become a common modulation scheme in practice. This scheme may be derived as a special case of OQPSK, where the square binary signals a_i and b_i are shaped by multiplying them with a sinusoidal weighting term. This signal is written

$$s_i(t) = a_i \cos\left(\frac{\pi t}{T}\right) \cos 2\pi f_c t + b_i \sin\left(\frac{\pi t}{T}\right) \sin 2\pi f_c t. \quad (1)$$

In Figure 3-1 the various stages in the construction of this signal are shown for a sample trans-

mission. The bitstream shown is $\{\dots, -1, 1, -1, 1, 1, 1, -1, -1, -1, -1, -1, 1, -1, 1, -1, -1, \dots\}$, where alternate bits are respectively labeled a_i and b_i . Figures 3-1(a) and 3-1(c) show the product of the bit stream square waves and the sinusoidal weighting, and 3-1(b) and 3-1(d) show the product of these signals with waves at the carrier frequency, $\cos 2\pi f_c t$ and $\sin 2\pi f_c t$. The modified bit streams act as an envelope for these carrier signals, while the resulting sum $s_i(t)$ shown in Figure 3-1(e) has a constant envelope. There will be no discontinuity in phase as long as the carrier frequency f_c is a multiple of $\frac{1}{2T}$ [41].

From Figure 3-1(e) it appears that an MSK signal oscillates between two distinct frequencies with zero phase shift. This relationship with FSK becomes clearer when the MSK signal (1) is rewritten using a simple trigonometric identity [41]:

$$s_i(t) = \begin{cases} \cos(2\pi f_c t \mp \frac{\pi t}{T}) & a_i = 1, b_i = \pm 1 \\ \cos(2\pi f_c t \pm \frac{\pi t}{T} + \pi) & a_i = -1, b_i = \pm 1 \end{cases} \quad (2)$$

or

$$s_i(t) = \cos\left(2\pi f_c t - \frac{a_i b_i \pi t}{T} + \theta\right) \quad (3)$$

where

$$\theta = \begin{cases} 0 & a_i = 1 \\ \pi & a_i = -1. \end{cases}$$

The spacing between the carrier and either transmitted frequency is thus $\Delta f = \frac{1}{2T}$, and it may be shown [41] that this is the *minimum* frequency spacing which allows for robust demodulation of an FSK signal. Hence the designation as minimum-shift keying.

Since the MSK modulation technique was derived as a case of QPSK, it may be de-

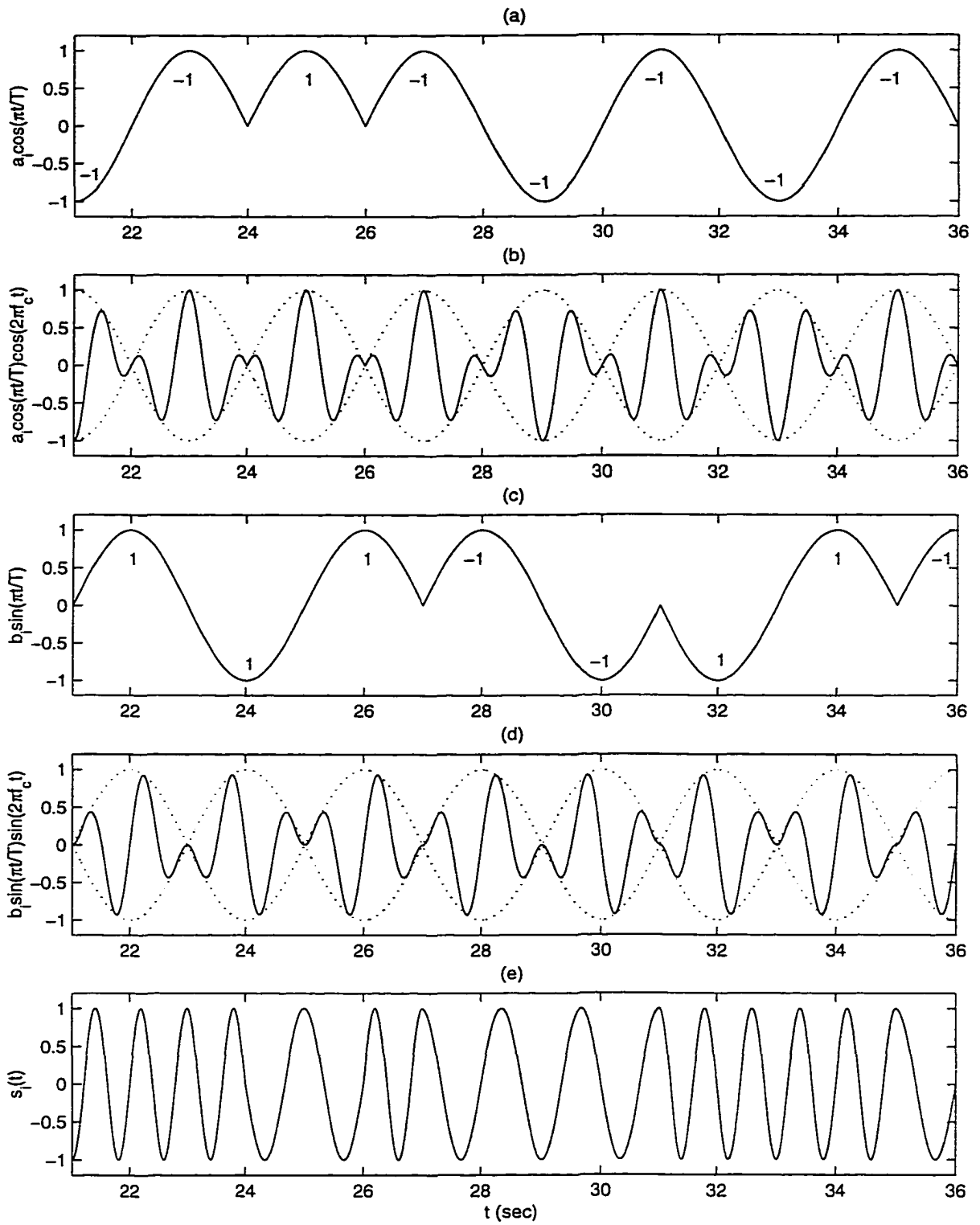


Figure 3-1 MSK waveform construction.

modulated using methods associated with that class of signals [33]. But MSK may also be represented as a special type of FSK, although since *two* bits are being transmitted at the same time, the frequency shifts have different meanings than in true FSK. High frequency tones are now associated with a transition from one bit to the opposite bit, while low frequency tones mean that the bit parity stays the same. That is, high frequencies will be transmitted for bit pairs (1, -1) and (-1, 1), and low frequencies for pairs (1, 1) and (-1, -1).

As was mentioned previously, to satisfy FCC narrow-band regulations it is often necessary to limit the frequency bandwidth of a digital signal. While the phase continuity of MSK results in a signal with a narrower frequency range than PSK, **Gaussian-filtered MSK (GMSK)** modulation provides even greater improvement. This is simply MSK modulation where the data stream is passed as a square wave through a Gaussian filter. The impulse response of a Gaussian filter may be obtained from the Gaussian distribution function

$$pdf(t) = \frac{1}{\sqrt{\pi}} e^{-t^2}.$$

The result of passing the data stream through this type of filter is to smooth out the edges of the square wave and to reduce the high frequencies associated with these discontinuities. The frequency shifts in the transmitted signal now occur more gradually than in the original MSK signal. This has the benefit of producing a signal which uses a narrower frequency band, at the cost of introducing some level of *inter-symbol interference* (ISI), or blurring of symbol transitions. The hardware-generated data in Section 3.5 will be in this form.

3.3 Bit error correction

The basic technique used to study the geometric properties of a digital signal such as MSK was a two-dimensional time-delay reconstruction with $\tau = T/4$, seen in Figure 3-2(a), where T is the symbol interval. In this figure, T corresponds to 100 samples. As was mentioned in the introduction to this chapter, channel effects may be modeled by convolving the signal with an equivalent channel response function. To model a simple channel which introduces a detectable level of ISI, the MSK signal from Figure 3-2(a) was convolved with an exponentially decaying response function one symbol in width. The resulting reconstruction is shown in Figure 3-2(b). Since trajectories in both reconstructions exhibit a clear structure, relative to the reconstructions of chaotic signals seen in Chapters 1 and 2, they will be referred to as *wire diagrams*.

In the previous section it was shown that the transmitted MSK signal oscillates between two frequencies, which are $f_c \pm \frac{1}{2T}$. If the signal is a single pure sine wave, as in $s(t) = \sin 2\pi f_c t$, then the reconstruction for $\tau \neq \frac{k}{2f_c}$ for $k = 0, \pm 1, \pm 2, \dots$ will be in the shape of an oval. If $\tau = \frac{1}{4f_c} + \frac{k}{f_c}$, then the reconstruction will be a perfect circle, since the first coordinate will be $\sin 2\pi f_c t$ and the delayed coordinate will be

$$\begin{aligned} \sin 2\pi f_c(t + \tau) &= \sin 2\pi f_c \left(t + \frac{1}{4f_c} + \frac{k}{f_c} \right) \\ &= \sin \left(2\pi f_c t + \frac{\pi}{2} + 2\pi k \right) \\ &= \cos 2\pi f_c t. \end{aligned}$$

In both reconstructions in Figure 3-2, the high and low frequencies appear as two ovals with

major axes along the lines $y = -x$ and $y = x$, respectively. In Figure 3-2(a), the ovals are equal in size, while on the right, due to the convolutional effects of the channel, the oval with major axis along $y = -x$ has become small enough to fall within the boundary of the other. This is because the channel response attenuates high frequencies while allowing lower frequencies to pass. The key observation to make, however, is that the convolution with the exponentially decaying window introduces slight variations in trajectories, both near the ovals and in transitional paths, which are dependent on the recent past and near future dynamics. That is, the convolution introduces short-term determinism in the signal, even if the transmitted bit stream is random. In particular, because the response function was one symbol wide and the coordinates of each point in the time-delay reconstruction are separated by $T/4$, points on the reconstruction contain information from as many as three neighboring symbols. For example, if we consider a point on the inner oval, by carefully studying which part of the trajectory the point is on, one may be able to determine if the next symbol will occur on the inner oval or if the point will evolve towards the outer oval. Similarly, we may look backwards to the previous symbol. Therefore, given a point on a trajectory, it is possible in this simple example to determine up to three bits in the transmission. In the reconstruction of the clean signal, if a point is chosen on either oval, there is no evidence of the signal's previous or future evolution.

The fact that short-term correlations introduce separation of trajectories in the phase-space reconstruction implies that the geometry can indicate a great deal about the underlying bit stream. This has the potential to provide an error detection and correction method. For example, suppose that a transmission was interrupted, or an error was introduced due to excessive noise, as in Figure 3-3. In this example two symbols were lost, and the

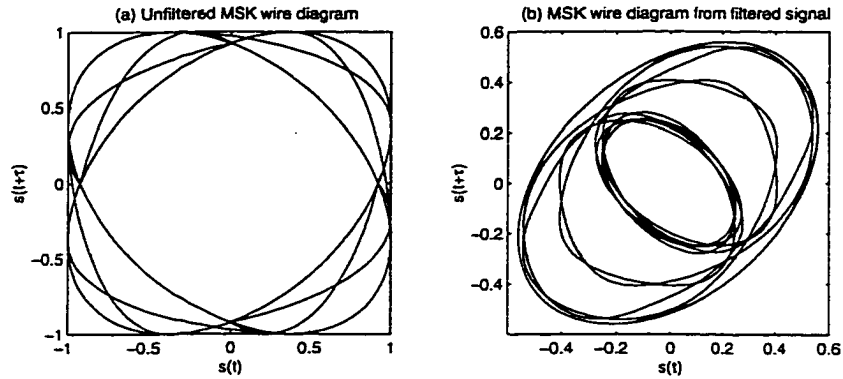


Figure 3-2: (a) Pure MSK signal reconstruction. (b) Reconstruction of signal after passing through a channel.

dotted curves in the figure show two possible correct transmissions which would fill the void. The correct curve has a local maximum at sample number 400. To correct this error, one may form a reconstruction and find where the signal was interrupted and regained. The reconstructions for the pure signal in Figure 3-3 as well as the same signal convolved with the channel response are shown in Figure 3-4, where the dotted curve indicates the correct but missing section of the transmission. Trajectories evolve in a clockwise direction over the reconstructions. Notice that there is no evidence of the correct signal in the unconvolved diagram: with the dotted portion missing, there is nothing to indicate that the signal ever deviated from the solid oval. But in Figure 3-4(b) where the signal that passed through a channel is regained, i.e. when the trajectory in Figure 3-4(b) goes from dotted to solid, there is enough separation present to determine that the trajectory is heading towards the outer oval from the inner one. This observation provides the receiver with enough information to

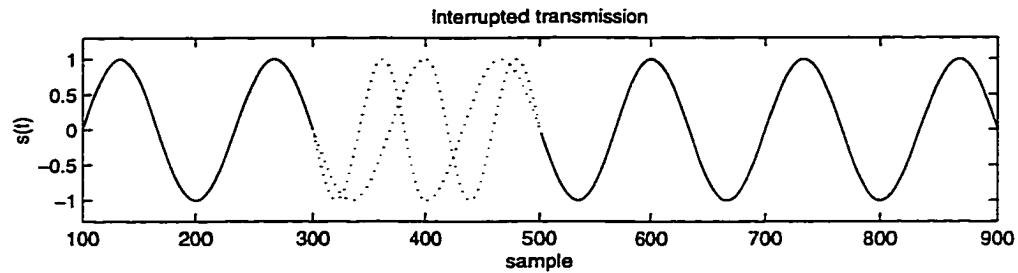


Figure 3-3 Clean, interrupted MSK signal, showing two possible correct transmissions.

correctly deduce the two missing symbols in the transmission. The receiver will typically have the entire wire diagram for comparison, so that the dotted line will simply be tracing over a previously observed trajectory, making it easier to predict the evolution of the signal.

3.3.1 Effect of dispersion and fading on error correction

Digital signals are often subject to a number of distorting channel effects. Two of these effects are dispersion and fading, where energy is lost and the signals become distorted. Fading occurs when power is lost in transmission. Commonly, this power loss is frequency-dependent; higher frequencies are absorbed more quickly than low frequencies. Dispersion occurs when different frequency components propagate at different rates through a medium, like light through a prism, introducing a frequency-dependent phase shift. These effects were investigated to determine what impact they would have on the geometric approach to signal demodulation. Some tests were run on a simulated MSK signal using simplified models for the different distortions. In Figure 3-2(a) the time-delay reconstruction of an MSK signal was shown. This will be used as a reference diagram for comparing the distorting effects of dispersion and fading on the signal reconstruction. To model the dispersion ef-

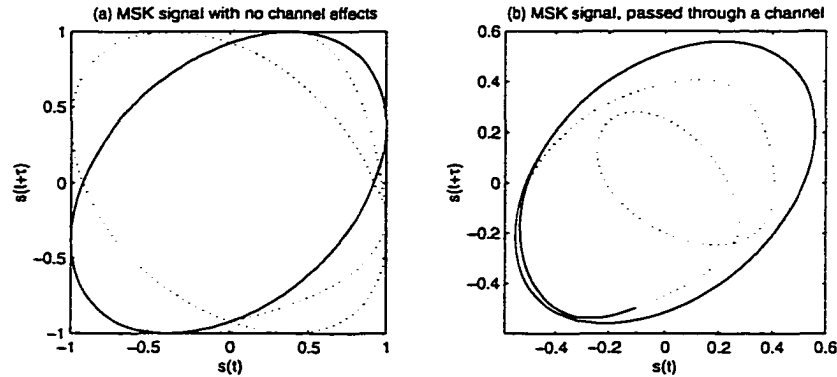


Figure 3-4: (a) Pure interrupted MSK signal reconstruction. (b) Reconstruction of interrupted signal after passing through a channel.

fect, a frequency-dependent phase shift was introduced by taking the FFT of the signal and converting the real and imaginary parts of the complex amplitudes to an amplitude-phase representation. Then the phase relationships were changed by adding a linear component to the argument of the FFT of the signal. The time-delay reconstruction of the distorted signal is shown in Figure 3-5. Notice that although some of the nodes/vertices have undergone splitting, the overall structure in this reconstruction is essentially the same as before. There remain two distinguishable ovals oriented as in Figure 3-2(a), representing the two frequencies, although the phase shift has reduced the amplitude of the higher frequency more than the low frequency. Transitional trajectories on the reconstruction in Figure 3-5 from one oval to the other begin earlier than in Figure 3-2(a), which indicates the introduction of some level of ISI. This was shown to provide error-correcting information in Section 3.3.

The effect of fading was tested similarly by decreasing the magnitude of a portion of

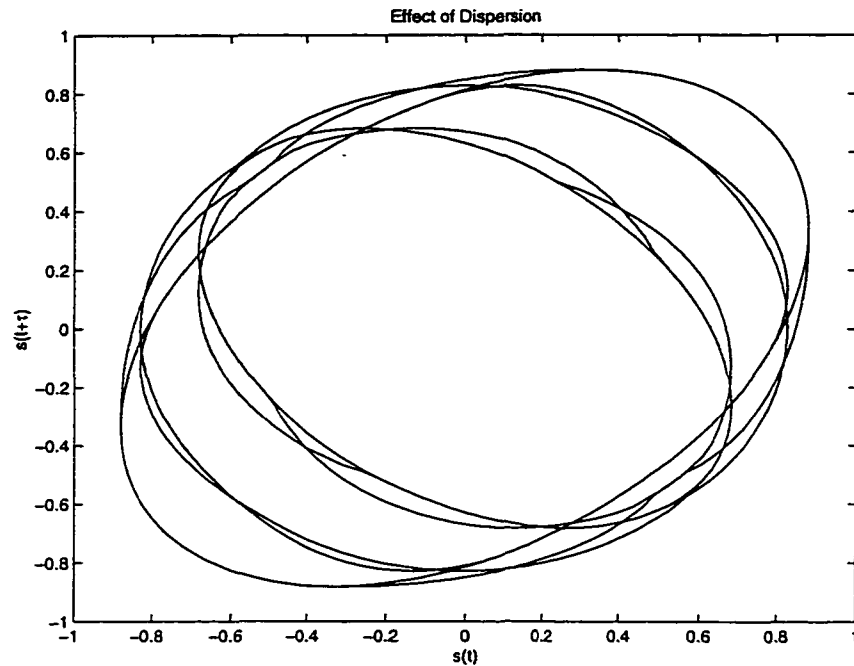


Figure 3-5 Effect of dispersion on MSK signal reconstruction.

the FFT linearly from low to high frequencies by a factor from 1 to 0, and comparing the reconstruction of the inverse transform to the original. Figure 3-6 shows the reconstruction of the result. Again, other than a simple deformation, the overall structure is unchanged. In [33], Pahlavan and Levesque show that “frequency-selective fading in the frequency domain is manifested as ISI in the time domain.” Since the type of fading introduced by this simple model was frequency-dependent, we can view the splitting of trajectories in the deformed reconstruction as a type of ISI. So the error-correction technique presented in the previous section may be applied to this signal in much the same way. Thus it seems from these preliminary tests that fading and dispersion will produce deformations which should not greatly affect the potential for signal classification or error-correction.

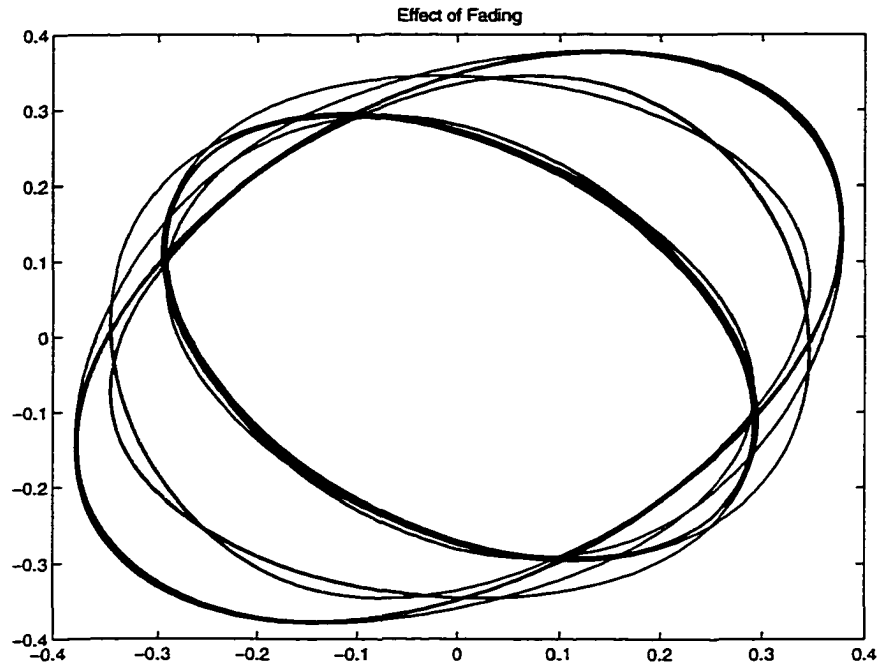


Figure 3-6 Effect of fading on MSK signal reconstruction.

3.4 Cochannel demodulation

Thus far, properties of MSK digital signals have been discussed, and time-delay reconstructions have been shown to provide a certain amount of error-correction capability. One motivation for studying these digital signals was to attempt to solve the problem of separating two signals that have been passed through the same channel, i.e. reducing cochannel interference. Now the case when two signals are being received in the same frequency band will be considered. The cochannel data initially examined was generated using a simple model, where it was assumed that the bit transmission rate and the amplitudes for both signals were identical. The signals were MSK modulated, sampled at a rate of 100 samples per symbol. A time scale was not defined, but could be assigned arbitrarily to determine

a bit transmission rate and sampling rate in Hertz. The assumption of equal amplitudes may actually make the problem more difficult since it removes one way of distinguishing one signal from the other.

To obtain the data two MSK signals were simulated with identical bit transmission rates and amplitudes, transmitting messages that were independent random bitstreams. It is assumed that the two signals are transmitted independently, but using the same MSK communication technique on the same frequencies. Consequently, the independence means that there can be a lag between the symbol timing in each channel. That is, the beginning of one symbol in signal A may not coincide with the beginning of a symbol in signal B. If this is the case, let the *symbol offset* be defined to be the shortest length of time between the beginning of a symbol in signal A to the beginning of a symbol in B. Given that the signals are transmitting data at the same rates, this symbol offset will remain fixed for the duration of the transmissions. Two typical signals from these systems are seen in Figures 3-7a and 3-7b, where random bits are being transmitted. They can be added together assuming various symbol offsets between the channels. When the symbol offset between channels is zero, complete cancellation of certain bits of information can occur, as can be seen in Figure 3-7c. If the symbol offset is not equal to zero, then the compound signal is similar to that in Figure 3-7d. Some portions of the compound signal reveal which two symbols are being transmitted. For example, it is sometimes easy to detect when both signals are transmitting a high-frequency symbol, because the combined signal will also reflect the higher frequency, such as near the point 21400 in Figure 3-7c or d. However, it is difficult using only the one-dimensional time series data to assign symbols to the separate transmitters and thus to keep track of the data encoded in each signal. The geometric approach provides better

separability.

The compound signal may be reconstructed using time-delay coordinates as described previously. The resulting structure provides a wire diagram which catalogs all possible combinations of symbols and transitions from both signals. So, for a fixed time lag between the channels, the states of the channels are encoded into the branches of the wire diagram. If a different time lag existed between the channels, a somewhat different wire diagram results, but it has similar topological structure. Since the structure changes (apparently smoothly) with each different time lag between the channels, the time lag was incrementally varied and the corresponding wire diagram was generated; then a set of frames was compiled into an MPEG movie to see how the wire diagram deforms. Several frames of this movie are seen in Figures 3-8-3-11, and the entire MPEG movie may be viewed by following the link from the Web page at <http://www.math.unh.edu/~kmshort>. It seems that as long as the phase lag is not zero there are two distinct sets of trajectories, and the wire diagram can be decomposed into two congruent parts with interconnecting transitions. Given an unknown data stream composed of two transmitters, one would be able to perform the reconstruction of the wire diagram, and then compare the result with the configurations in the MPEG movie to find the symbol offset between the channels. Once this is known, it is possible to determine when the different channels are changing symbols.

To illustrate, let A and B be two such signals, and suppose that the symbol offset is one tenth of a symbol. Suppose both signals are transmitting all -1 's, until at some point one of them changes to all 1 's. We would like to know if a difference can be seen between the two cases where either A or B is the signal which changes bit streams. These cases with the resulting (very similar-looking) compound signals may be seen in Figures 3-12a-f. In the first

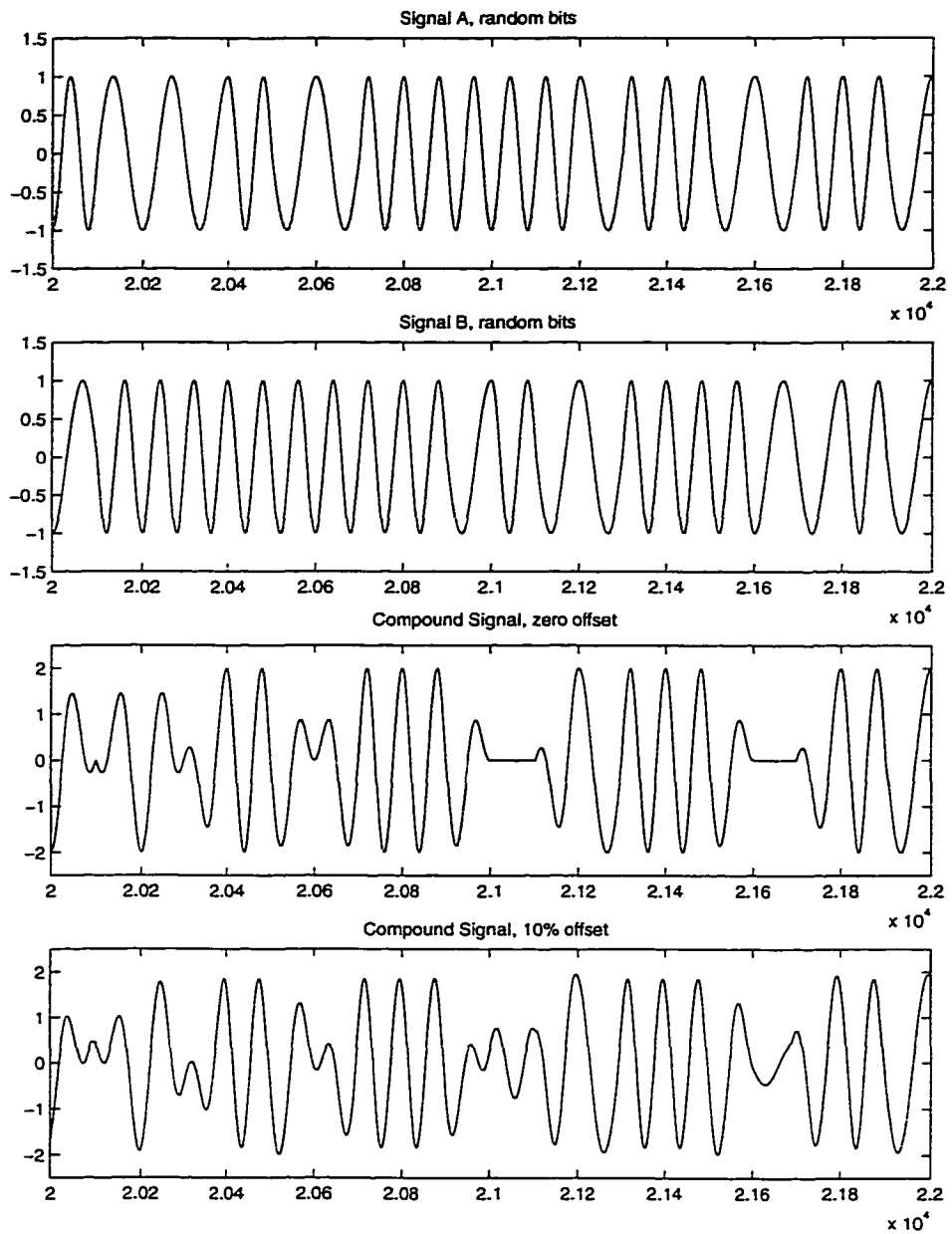


Figure 3-7 Example of cochannel interference.

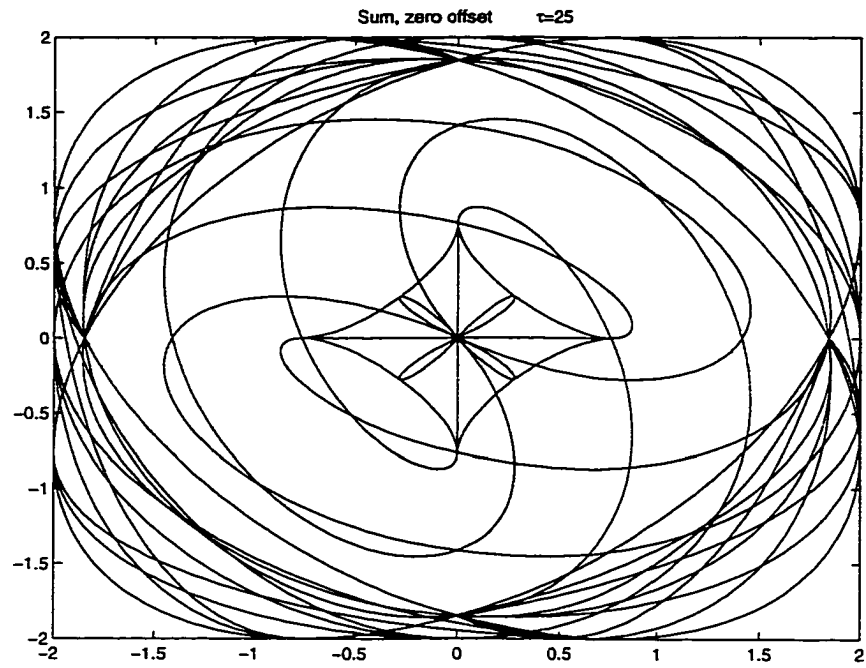


Figure 3-8 Cochannel reconstruction, zero symbol offset.

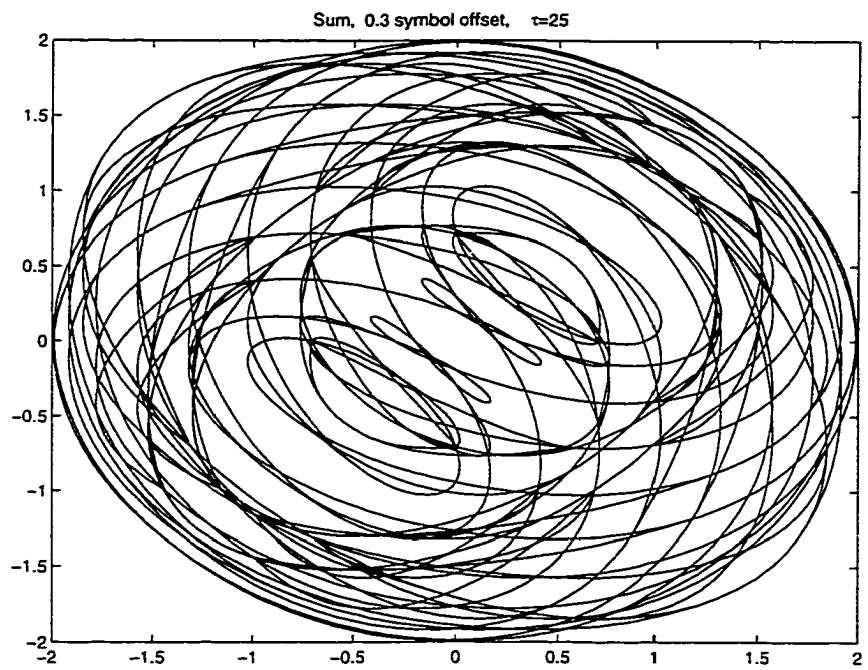


Figure 3-9 Cochannel reconstruction with an offset of 0.3 symbol.

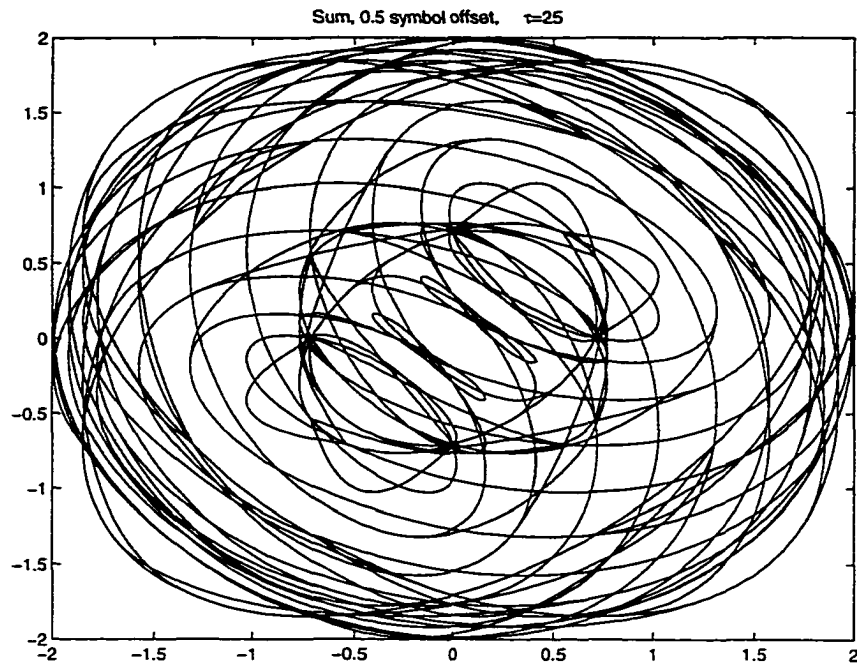


Figure 3-10 Cochannel reconstruction with an offset of 0.5 symbol.

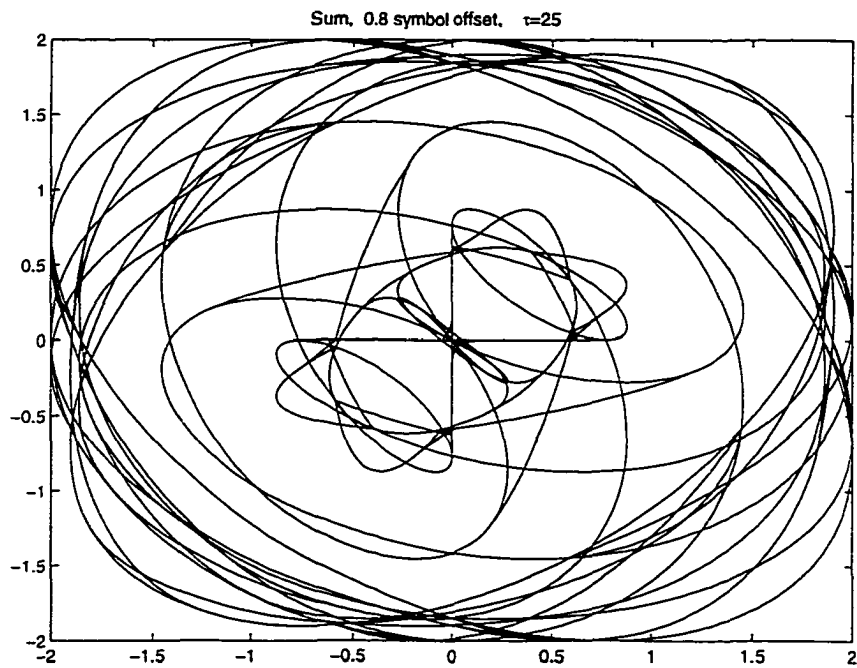


Figure 3-11 Cochannel reconstruction with an offset of 0.8 symbol.

case, signal B switches bits, while in the second case signal A switches bits. Remember that the symbol offset between the two signals remains the same for both cases. The compound signals for both cases are seen in Figures 3-12c and 3-12f. It may be difficult distinguish between the two cases using standard demodulation techniques, since the structures of the received signals are very similar; the high-amplitude sine wave is suddenly reduced by some unknown interfering signal. The differences between these two compound signals are amplified by the time-delay reconstructions in Figures 3-13 and 3-14. Both reconstructions exhibit a large and a small oval oriented with major axes along $y = x$. Keeping in mind that the trajectory generally travels in a clockwise direction, we notice that not only does the trajectory in case 2 diverge earlier from the outer oval, at the point marked with the symbol " \oplus ," but its path towards the inner oval is vastly different than the path taken by the trajectory in case 1. Thus in this elementary example is it easy to see that one can distinguish between the two cases and therefore be able to assign the appropriate bit stream to each signal.

To further test this ability to separate bit streams by studying the wire diagram, two MSK signals transmitting the ASCII equivalents of the words "Eureka" and "Ansatz" were numerically simulated, then were added together at different powers and with a symbol offset of $0.3T$ where T is the symbol interval. The amplitude of the signal carrying the word "Ansatz" was 77% that of the other signal, and identical transmission rates were assumed. The signals and sum are shown in Figure 3-15, and the reconstruction of the compound signal appears in Figure 3-16. After determining the appropriate wire diagram, it is a simple matter of tracing the evolution of the compound signal along the reconstruction, and properly identifying the individual arcs in the diagram with the correct bits in both bit

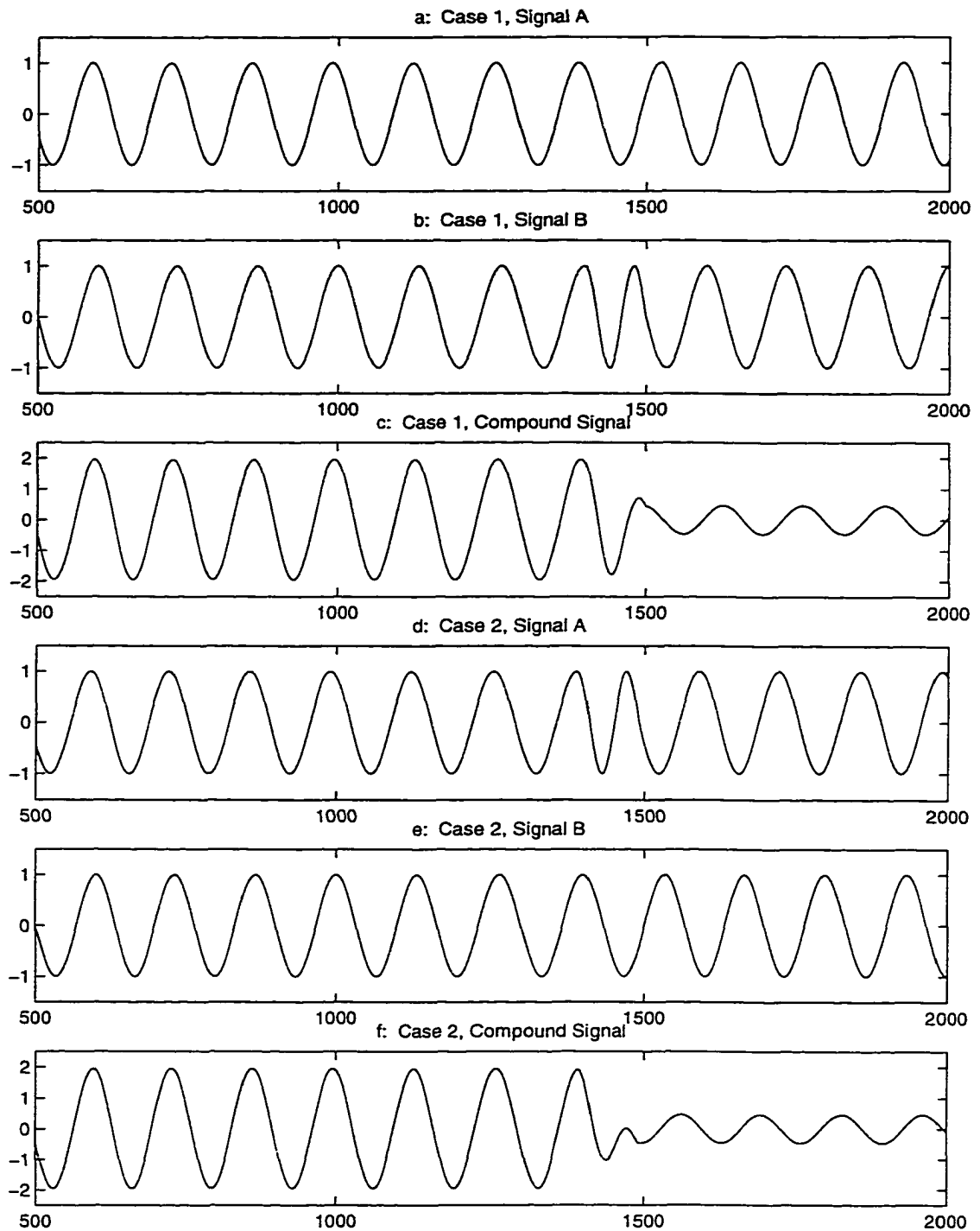


Figure 3-12 Determining which bitstream is associated with which signal.

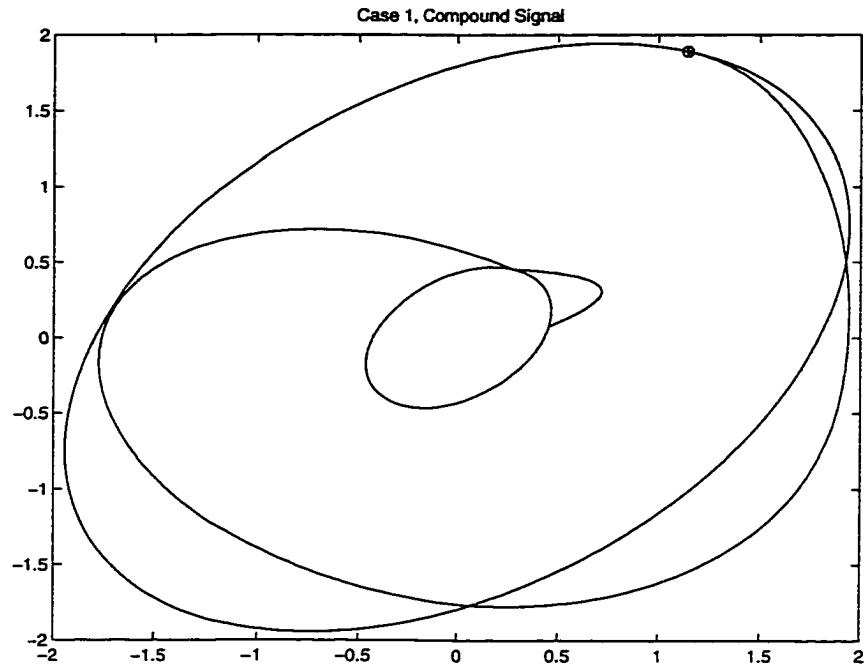


Figure 3-13 Reconstruction for first case where signal B switches bits.

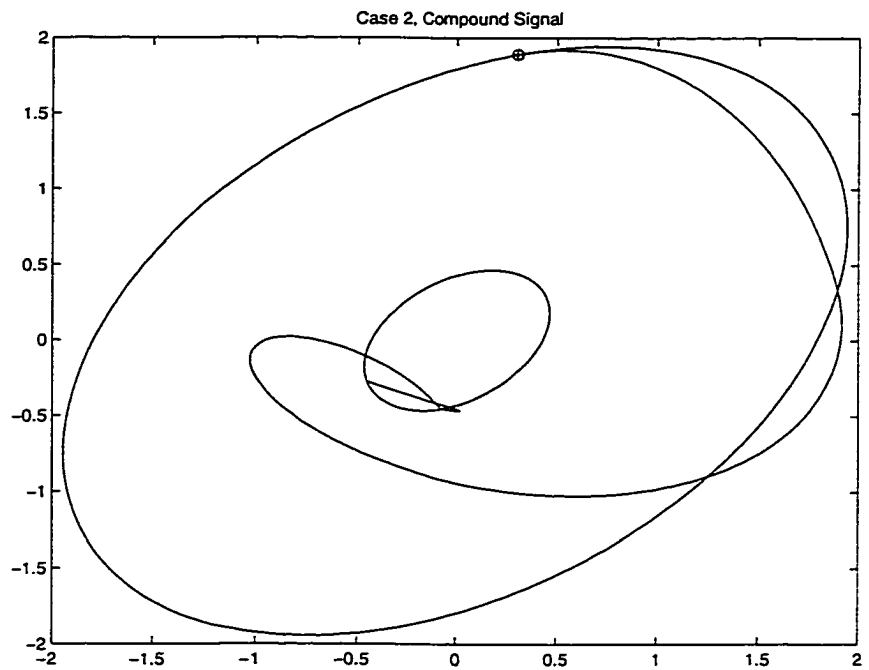


Figure 3-14 Reconstruction for second case where signal A switches bits.

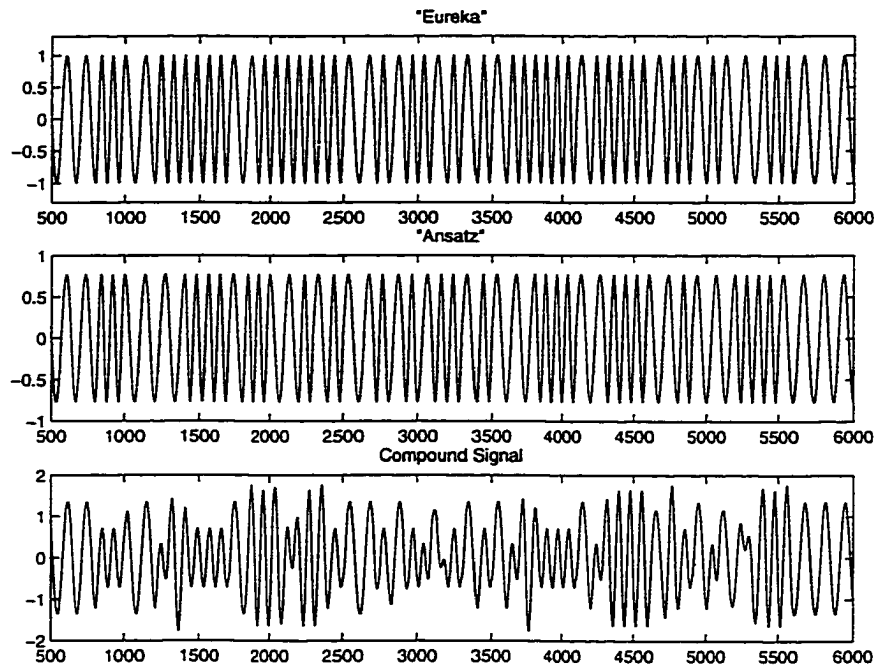


Figure 3-15 Interfering MSK signals at different amplitudes.

streams. It is much easier in the reconstruction to determine which symbol transitions are associated with each signal source. To simplify this example, Figure 3-17 shows just one ASCII letter from each signal: "E" from the full power signal, and "A" at reduced power, offset by $0.3T$. If we compare this case to one where the letters have switched roles, i.e. "A" at full power and "E" from the weaker signal, seen in Figure 3-18, one can readily detect the difference in trajectories over the reconstruction. Since there are 8 bits (symbols) per ASCII letter, these reconstructions are more complex than those in Figures 3-13 and 3-14, but by demodulating one pair of transmitted symbols at a time we reduce the problem of separating the bitstreams to one which was solved in the previous paragraph.

To summarize the results in this chapter thus far, the first step in the study of digital communication from a geometric viewpoint was to determine whether a typical communica-

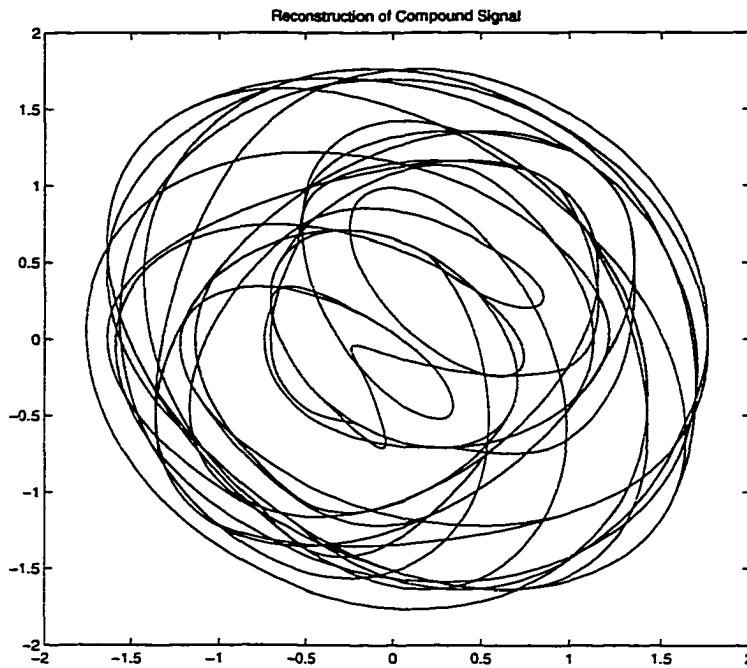


Figure 3-16 Reconstruction of interfering signals transmitting “Eureka” and “Ansatz”.

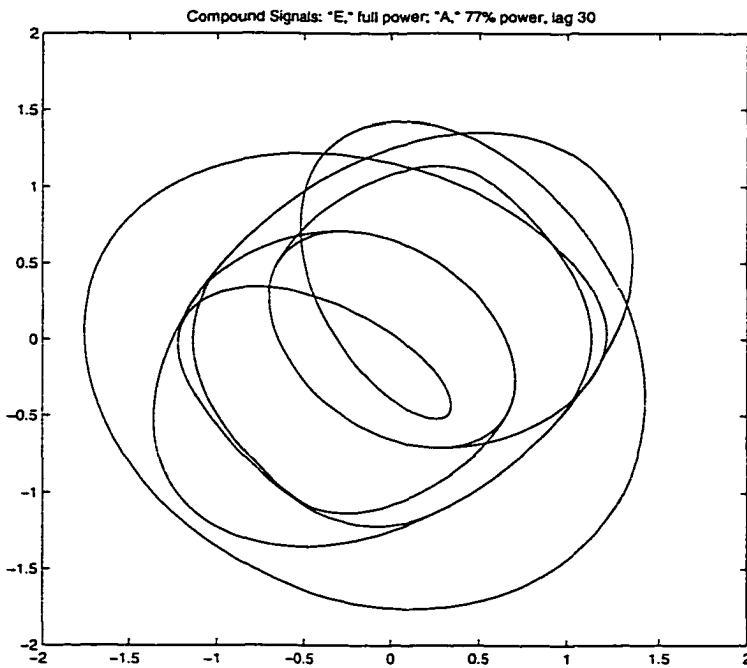


Figure 3-17 Case where signal carrying “E” is at full power.

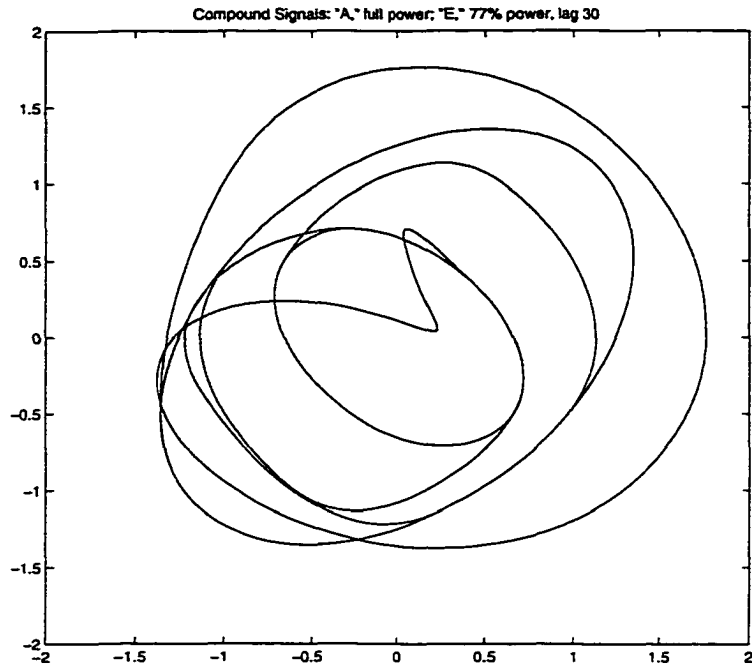


Figure 3-18 Case where signal carrying "A" is at full power.

tion signal could be reconstructed. It was found that the resulting wire diagram of a common type of digital signal (MSK) was clear and consistent. Then it was considered what deformations of the wire diagrams resulted if the communication signal was convolved with certain channel effects. There it was found that the overall topological structure was generally preserved. In addition, it was found that if the channel introduced short-term determinism into the signal, then the nodes in the wire diagram split, and this had the desirable property that it allowed for a form of dynamical error correction. When the channel introduced fading or dispersion, the wire diagram was deformed, but retained its topological structure. A geometric approach to cochannel demodulation was also considered. Given a compound signal composed of two interfering digital transmissions, the resulting reconstructed wire diagram was essentially a combination of the wire diagrams of the two individual channels

themselves, and the specific wire diagram was related to the symbol offset between the channels. The geometric structure allows for the separation of the bit streams transmitted by the interfering MSK signals. In the next section, these techniques will be applied to a real, hardware-generated GMSK signal.

3.5 Examination of real GMSK Data

3.5.1 Background

The potential application of NLD techniques to common problems in the demodulation of standard digital signals has been discussed, including channel distortion effects, error-correction and co-channel demodulation. Most of the experiments and results were based on simple simulations of digital modulation methods, MSK in particular. For more realistic testing, hardware-generated GMSK data was obtained from an unclassified CD-ROM prepared at the National Security Agency. The signal data was examined for properties which were similar to those exhibited by the simulated signals from the previous section.

3.5.2 Procedure and Results

The GMSK data examined was generated using an HP ESG-D4000A analog signal generator, sampled at 5.12023Msamples/sec. Once the raw data was extracted from the CD-ROM, the examination began by looking at the sampled time series shown in Figure 3-19. The data at the top of Figure 3-19 exhibits an intermittent behavior due to a communication standard called *time-division multiple access* (TDMA) which allows many users to communicate over the same channel by transmitting in short, rapid bursts, alternating among the users. In this sample, only one user was transmitting. The sampling rate is close to Nyquist, i.e. there are

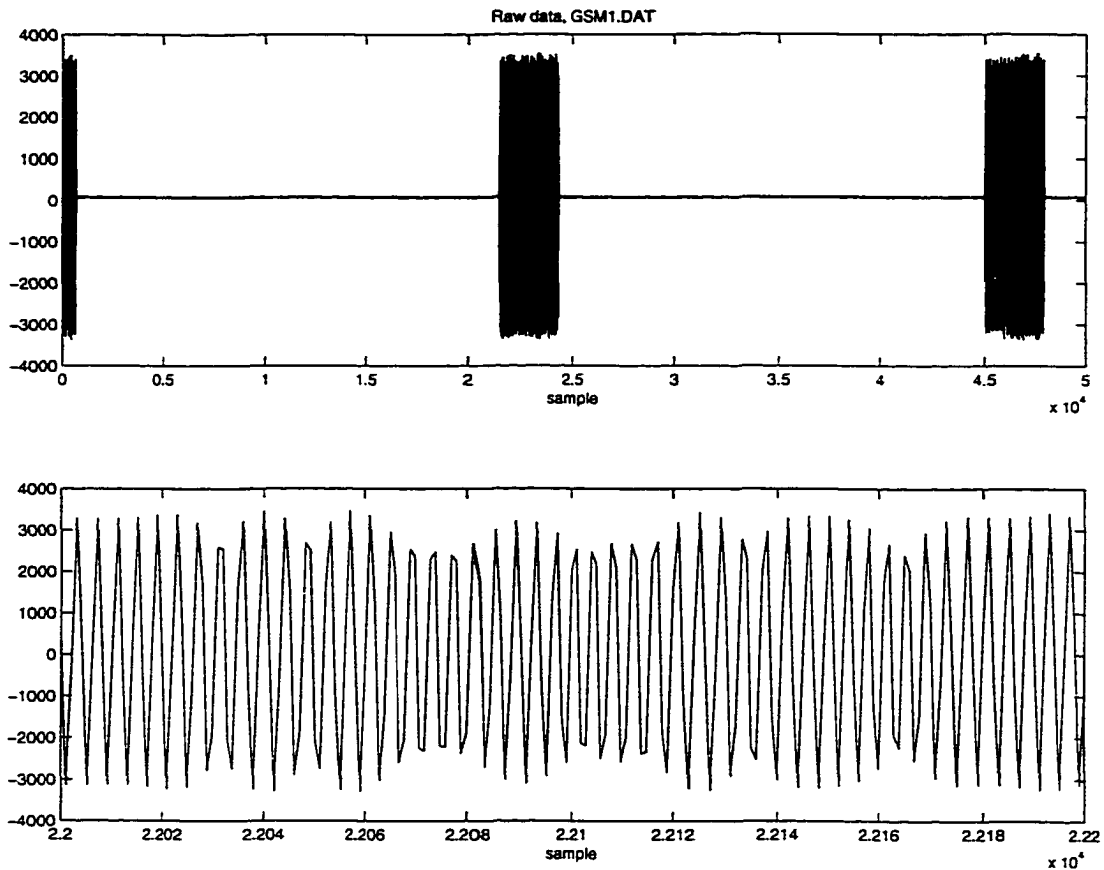


Figure 3-19 Raw GSMK data.

only approximately two samples per cycle, so to determine the presence of any geometric structure which could be exploited, a time-delay reconstruction was generated with delay $\tau = 1.953 \times 10^{-5}$ sec (i.e. $1/\text{sampling rate}$), plotted without connecting the points, as shown in Figure 3-20. This view shows a simple underlying structure to the signal, but the sample rate is too slow to obtain useful dynamical information. Thus the next step was to remove the carrier signal from the transmission, which is called *basebanding* the signal.

Basebanding a digital signal can be achieved by first transforming the signal into the frequency domain via a Fourier transform. The resulting spectrum is then convolved with

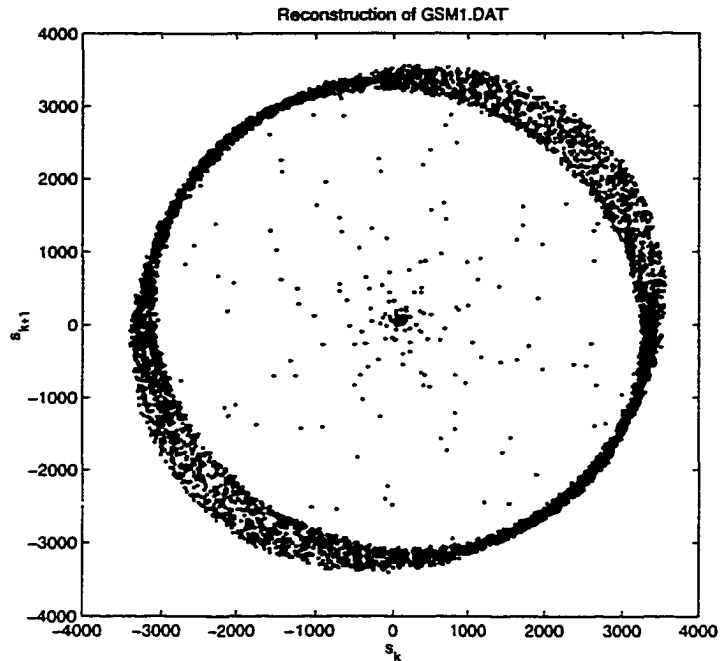


Figure 3-20 Reconstructed raw GSMK data.

a delta function chosen at the desired base frequency. The result is transformed back to the time domain. The basebanded data can be seen in Figure 3-21b, and a delay reconstruction is shown in Figure 3-22. As expected, this reconstruction has a shape that is much closer to the reconstruction of numerically simulated baseband GSMK data seen in Figure 3-23. Both exhibit the characteristic ovals associated with the two transmitted frequencies. However, the transitional trajectories in the basebanded real GSMK data do not line up as cleanly as in the simulated signal reconstruction. The cause for this is that the sample rate of the original signal as well as the result of the basebanding did not result in an integer number of samples per cycle, either for the high or low frequency symbols. This causes the point at which a trajectory leaves one oval to precess around that oval. Correcting for this precession will be the subject of future investigation.

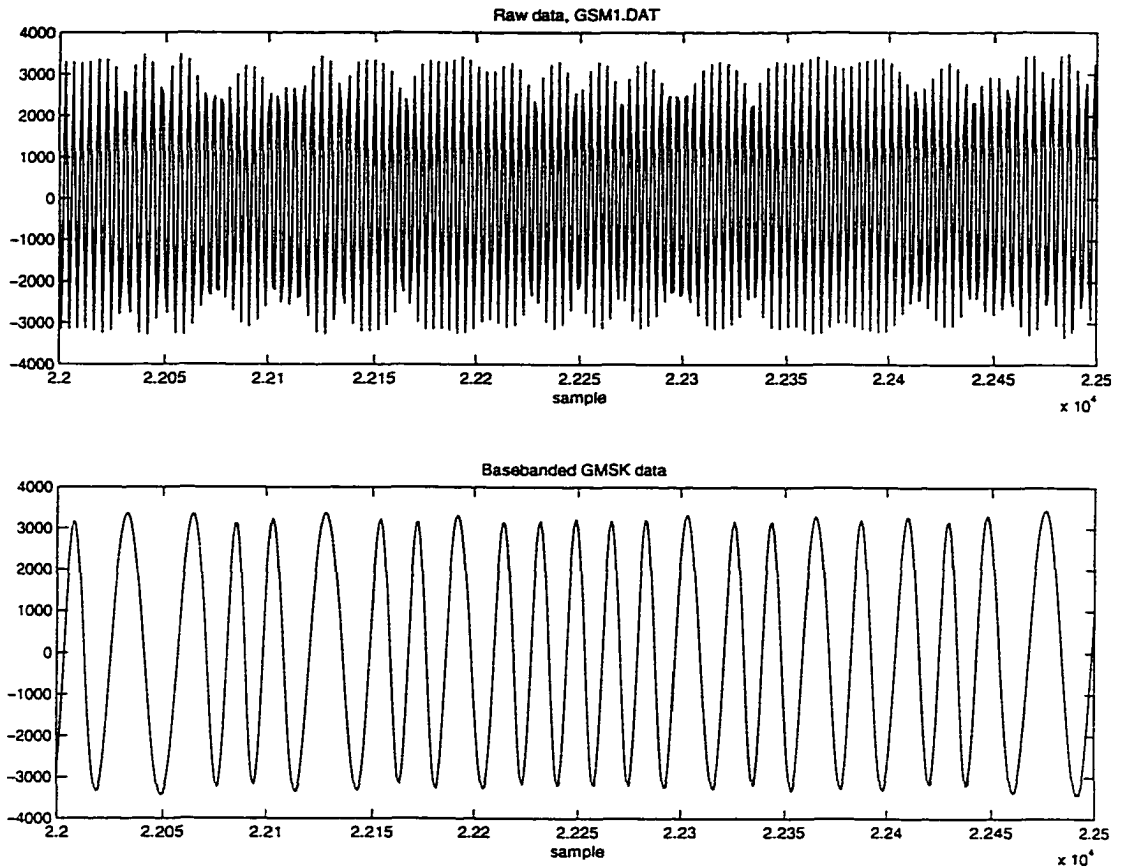


Figure 3-21 Original data and results of basebanding.

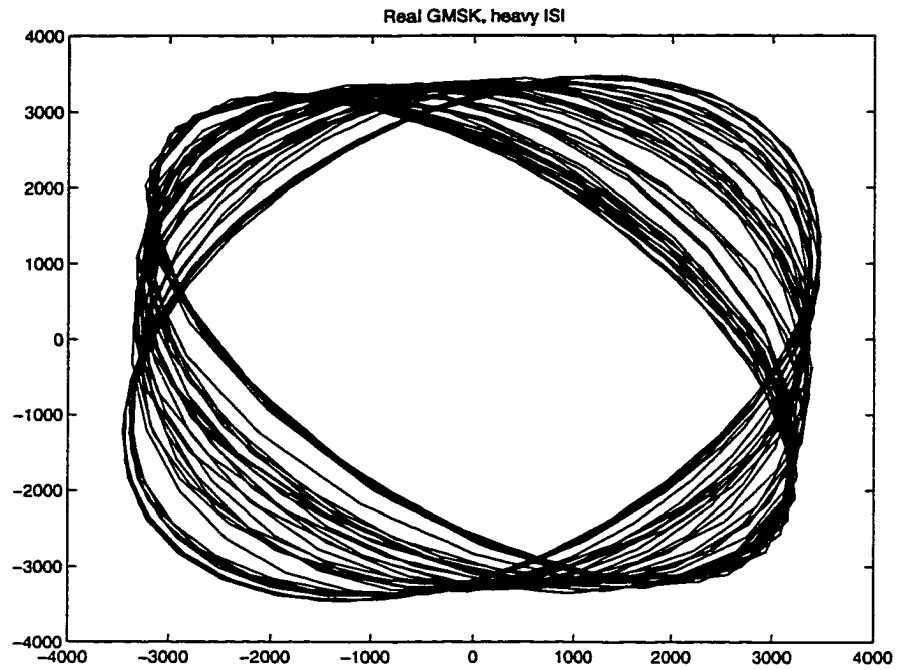


Figure 3-22 Reconstructed basebanded GSMK data.

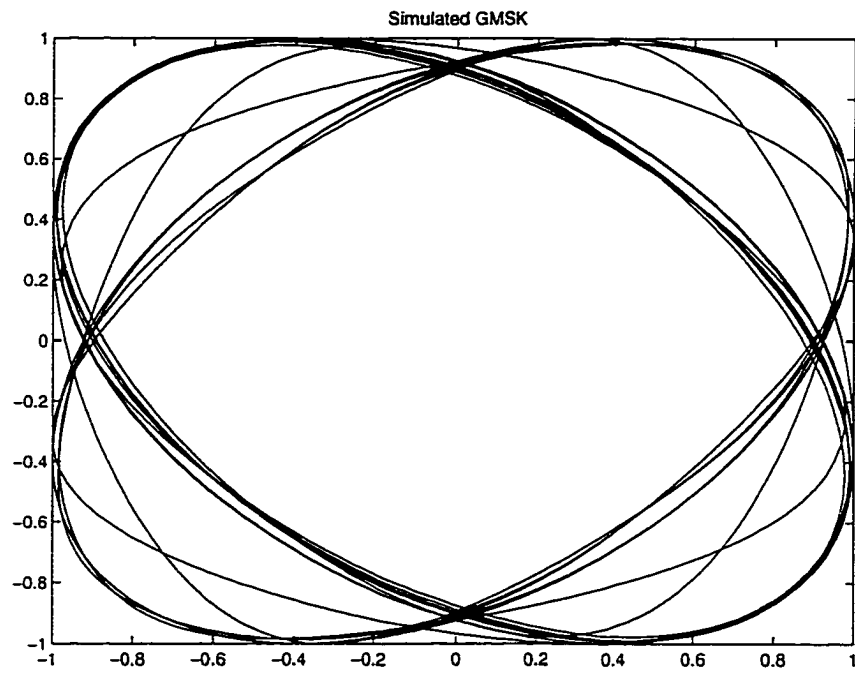


Figure 3-23 Reconstructed simulated GSMK data.

MSK signals can be described as a special case of FSK modulation, which also alternates between high and low frequency symbols, although the data is encoded somewhat differently into the frequency shifts. In fact, in practice, MSK signals are often generated and demodulated directly in its FSK form [33]. Using a numerical signal demodulation routine in the MATLAB signal processing toolbox, it was possible to demodulate the GMSK data files (both the real and the simulated series) as FSK signals. A comparison of the demodulation of the basebanded signal to that of the raw data (using the hardware settings as parameters) provided a check that the basebanding process preserved the symbol structure. The results of the demodulations of both signals are shown in Figure 3-24, where peaks and valleys represent high- and low-frequency symbols, respectively (the basebanding process apparently stabilizes the frequency of the received signal, which would account for the smoother demodulated signal in 3-24(a) versus 3-24(b)). To extract the correct bit stream, recall that a high frequency symbol represents a pair of alternate bits, and a low frequency symbol represents a pair of identical bits. Also recall that only one bit in a pair will switch at any symbol transition. In Figure 3-24(a), the sequence and widths in symbols of the peaks and valleys, beginning with the first large peak, are: five-symbol peak, four-symbol valley, one-symbol peak, five-symbol valley, etc. These symbol widths may be determined by correcting the signal for the mean and recording the length of time between zero-crossings. Thus, assuming that the bit stream begins with a 1, we can write down the transmitted sequence:

...10101000001111110100111101110110101001001101110...

The main tradeoff involved in using GMSK over MSK is the introduction of greater

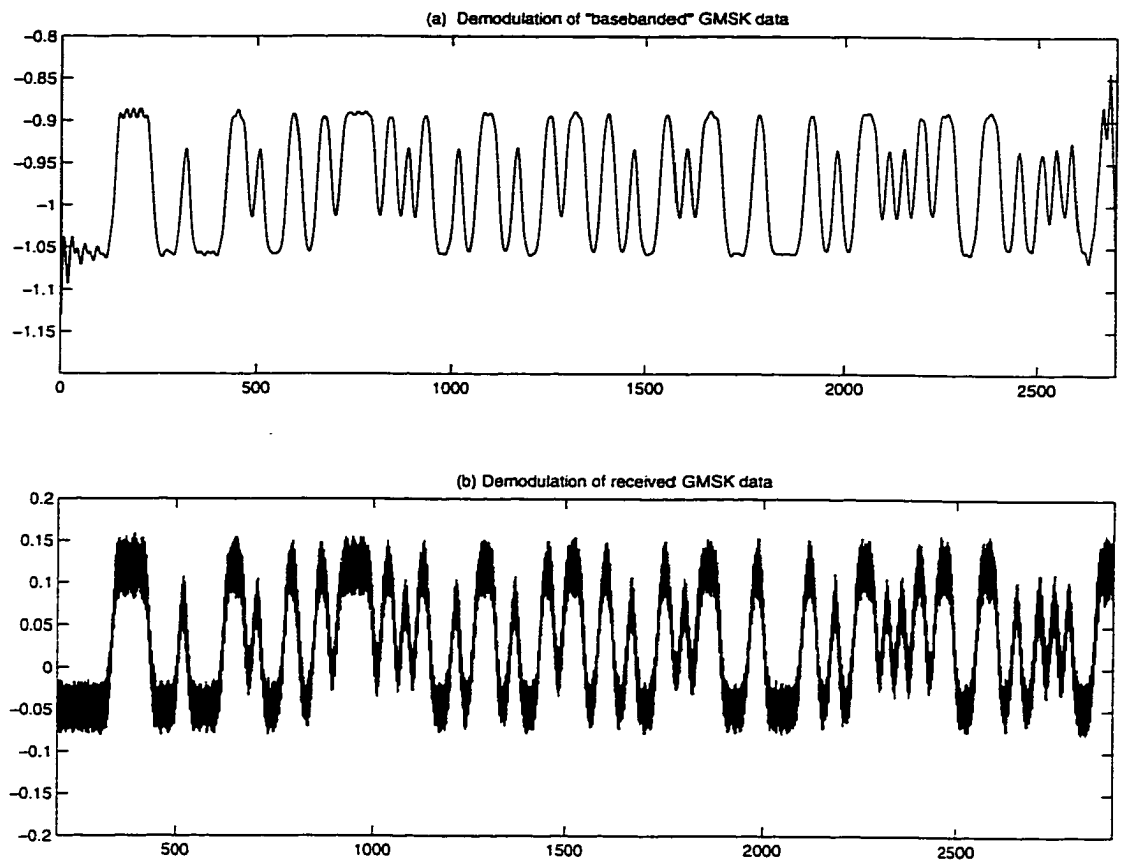


Figure 3-24 Demodulated raw and basebanded signals.

ISI. This is one of the most noticeable characteristics of the signals in Figure 3-24(a)—it may be harder to distinguish symbols in regions where the frequency shifts occur in rapid succession. This is because the difference between the high and low frequency levels in Figure 3-24(a) is smaller for peaks that are only one symbol in width. However, in a way similar to channel effects, this type of ISI introduces short-term determinism which is preserved and used for error-correction when reconstructed. A reconstruction of the demodulation of the basebanded real GMSK signal appears in Figure 3-25. The separation of trajectories in this reconstruction provides valuable information about the past and future symbolic evolution of the signal. The reconstruction may be generally described as having the shape of a football, where the corners of the football fall along the line $y = x$. The points at the upper right and lower left corners of the reconstruction represent high and low frequency symbols, respectively, and the paths the signal takes from one corner to the other determine the sequence of symbols, as was described in the previous paragraph. Near the corners appear distinct paths which pass close to, but do not actually reach, the corners of the football. These paths represent the peaks and valleys in Figure 3-24(a) which are two symbols in width. Two other paths which pass across the middle of the football represent the peaks and valleys which are one symbol in width. Trajectories which arrive at the corners represent peaks or valleys which are three or more symbols in width. These features provide a clearer picture of the short-term determinism present in the signal than the transitional orbits between ovals do in Figure 3-22, and convert the higher level of ISI into useful information for dynamical error correction.

In conclusion, while ISI may introduce bit errors in demodulating digital communication signals using standard techniques, it has been shown that reconstruction techniques used

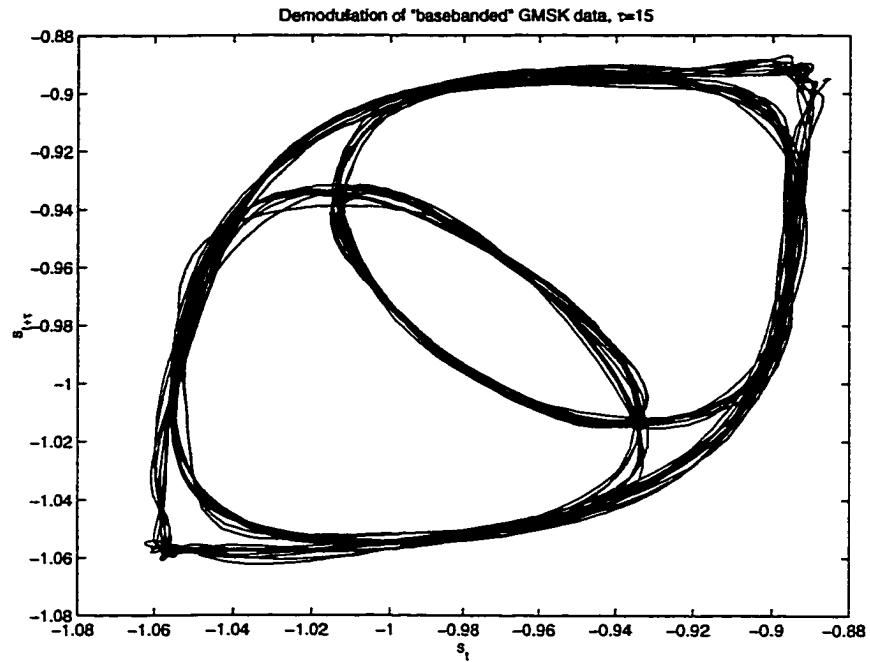


Figure 3-25 Reconstruction of the demodulated basebanded GMSK signal.

in NLD forecasting can unlock a considerable amount of error-correcting information contained in ISI. Other types of signals besides MSK and GMSK need to be examined in the same manner, although the expectation is that only the reconstructed wire diagram will be different, while the methods presented here may be applied with little modification.

Chapter 4

Digital chaotic communication system

4.1 Introduction

In this chapter a chaotic communication scheme will be developed which uses control of chaos to produce a digital chaotic communication channel. The approach uses the control scheme developed by Hayes, Grebogi & Ott (HGO) [17], introduced in Section 2.4.2, where small controls are used to steer the trajectories of the double scroll attractor around each of the two loops in the attractor, with each circuit around a loop corresponding to a 1- or 0- bit in an encoded message. In the HGO approach, the transmitted signal is essentially analog and the bit value is determined by observing whether the oscillation is above or below a reference value, as was shown in Section 2.4.2. Such a scheme was not intended to be secure; however, it will be shown that the scheme can be adapted so that if an identical transmitter and receiver are used, communication can be achieved simply by transmitting a binary signal corresponding to *control* or *no control*. This binary signal can be made to bear no correlation to the original message to be encoded and to have a delta-like autocorrelation function. In fact, the encryption and decryption will depend on the history of both the chaos *and* the message. Further, the receiver can be initialized remotely. There are three main points in this chapter. One is to show that such a communication channel effectively nullifies the nonlinear dynamic forecasting attack which has been effective on chaotic communication schemes which transmit a chaotic waveform. The second point is to show that a binary stream is all that is

necessary to achieve remote initialization and to maintain synchronization. The final point is to show that chaotic systems may be able to make a legitimate contribution to the field of cryptography.

In Section 4.2 it will be shown how the control scheme may be adapted so that a binary data stream can be used to keep an identical transmitter and receiver synchronized. This allows an encoded message to be delivered while making it impossible to extract the message by applying the techniques in Section 2.4.3. Section 4.3 shows that it is possible to initialize the receiver remotely by driving it onto a periodic orbit. Section 4.4 describes how this new system may be based on one-dimensional chaotic maps, which may be chosen to improve the statistics of the transmitted bit stream. Section 4.5 then analyzes some of the statistical properties of the binary communications and several failed attempts to find and exploit determinism in the transmitted signal. Section 4.6 discusses this scheme in a more traditional cryptographic context. Finally, Section 4.7 will offer some discussion of the security aspects of the approach, and the potential and probable security flaws.

The work in Section 4.4 and much of Section 4.5 was done primarily by the author, while the work in the other sections are the result of a collaboration with Kevin Short.

4.2 Binary Chaotic Communications

The chaotic control scheme discussed in Section 2.4.2 can be used to develop a communication protocol which involves a *digital* transmission between a chaotic transmitter and an identical receiver to achieve communications in a manner which is not amenable to breaking by NLD techniques [50, 51, 53]. What is needed is a way to send the receiver information that lets the receiver remain synchronized with the transmitter without transmitting a con-

tinuous waveform. To do this, the function $r_N(x)$ defined in Section 2.4.2 is used, which can be determined independently when calibrating the transmitter and receiver, and can be assumed to be known information for the transmitter and receiver. If the two systems are initially synchronized, so long as both the transmitter and the receiver have knowledge of $r_N(x)$, then all that must be done for the synchrony to be preserved is for the transmitter to tell the receiver when it has applied a control, under the assumption that the control moves the trajectory to the nearest location on the cross-section which gives the desired symbolic sequence.

For the purpose of simulation some techniques have been implemented which improve the performance of the numerically implemented control system from Section 2.4.2. Since the controls are executed along the two-dimensional surface of the Poincaré section defined on page 53, it is imperative that accurate calculations of the intersection between trajectories on the chaotic attractor and the Poincaré section be obtained. However, many common numerical integration algorithms result in a sequence of integration points which are evenly spaced in time, and would rarely, if ever, fall on the Poincaré section in phase space. This is because time is an independent variable, and it is difficult to calculate a time step which will result in a solution point on the Poincaré section. Henon [19] described a simple trick for autonomous dynamical systems which temporarily reverses the roles of time and space, allowing an integration step to be calculated using a spatial variable. This technique will be discussed here for the double scroll equations and the Poincaré section previously defined—a more general treatment may be found in [19].

Recall the equations governing the double scroll system:

$$\begin{aligned}
\frac{dv_{C_1}}{dt} &= \frac{1}{C_1}[G(v_{C_2} - v_{C_1}) - g(v_{C_1})] \\
\frac{dv_{C_2}}{dt} &= \frac{1}{C_2}[G(v_{C_1} - v_{C_2}) + i_L] \\
\frac{di_L}{dt} &= \frac{-1}{L}v_{C_2},
\end{aligned} \tag{1}$$

where

$$g(v) = \begin{cases} m_1 v, & \text{if } -B_p \leq v \leq B_p; \\ m_0(v + B_p) - m_1 B_p, & \text{if } v \leq -B_p; \\ m_0(v - B_p) + m_1 B_p, & \text{if } v \geq B_p. \end{cases}$$

The Poincaré section consists of two half-planes, one intersecting each lobe, fixed at a constant level $i_L = \pm GF$, $|v_{C_1}| \leq F$, where $F = B_p(m_0 - m_1)/(G + m_0)$. When a trajectory passes through these half-planes, the quantity $S = i_L \mp GF$ will change sign. When this happens, we want to back up one iteration and define a new integration step that will allow us to use the same integration scheme but will provide a solution point on the Poincaré section. We would like to make i_L the independent variable in the differential system. This is done by dividing the first two equations in (1) by the third one and inverting the third equation:

$$\begin{aligned}
\frac{dv_{C_1}}{di_L} &= \frac{-L}{v_{C_2} C_1}[G(v_{C_2} - v_{C_1}) - g(v_{C_1})] \\
\frac{dv_{C_2}}{di_L} &= \frac{-L}{v_{C_2} C_2}[G(v_{C_1} - v_{C_2}) + i_L] \\
\frac{dt}{di_L} &= \frac{-L}{v_{C_2}}.
\end{aligned} \tag{2}$$

Now the system is in a form which makes i_L the independent variable. It is therefore trivial

to calculate an integration step which will bring the trajectory to the Poincaré section, and it is given by $\Delta i_L = \pm GF - i_L$. This is just the distance in the i_L direction from the last (or previous) integration point to the half-plane which was crossed in the previous iteration. Once the integration is performed, continue along the trajectory using the original system (1) until the next time a sign change in S is detected.

To simplify the implementation, the systems (1) and (2) may be combined into one form:

$$\begin{aligned}
\frac{dv_{C_1}}{d\tau} &= K \frac{1}{C_1} [G(v_{C_2} - v_{C_1}) - g(v_{C_1})] \\
\frac{dv_{C_2}}{d\tau} &= K \frac{1}{C_2} [G(v_{C_2} - v_{C_2}) + i_L] \\
\frac{di_L}{d\tau} &= K \frac{-v_{C_2}}{L} \\
\frac{dt}{d\tau} &= K,
\end{aligned} \tag{3}$$

where τ is the current independent variable and $K = 1$ or $K = \frac{-L}{v_{C_2}}$ as appropriate.

The Poincaré section is two-dimensional, but because the attractor is also nearly two-dimensional near these half-planes, the intersection between the attractor and the Poincaré section is approximately one-dimensional. Figure 4-1 shows a top view of one branch of the Poincaré section, where the asterisks mark the intersections of a trajectory with the half-plane. This set of points may be approximated quite accurately by a line extending from the corresponding unstable fixed point fitted with a least-squares method. This is an important simplification, because it allows us to calculate the symbolic dynamics (i.e. $r_N(x)$) on a one-dimensional domain versus a two-dimensional surface. Now r_N may be defined as $r_N : [a, b] \times \{0, 1\} \rightarrow [0, 1]$ such that $(d, \ell_0) \mapsto \sum_{i=1}^N \ell_i 2^{-i}$ where d is the distance along the fitted line between the appropriate fixed point and the point of interest on the

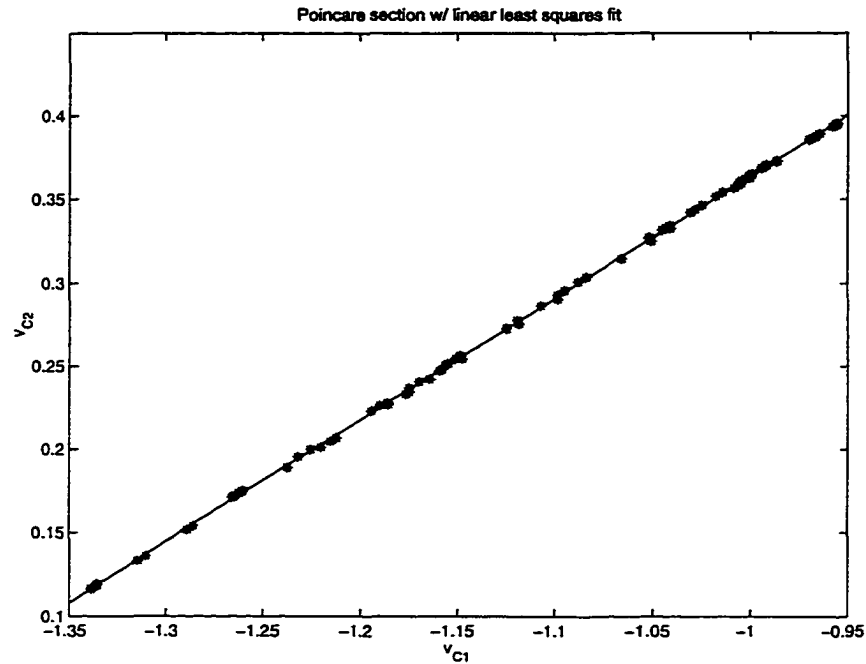


Figure 4-1 Poincaré section of the double scroll attractor, showing best fit line.

Poincaré section, ℓ_0 is an indicator variable denoting the current lobe and $\{\ell_1, \ell_2, \dots, \ell_N\}$ is the future sequence of lobes visited by a trajectory starting at the current point on the surface. The interval $[a, b]$ is chosen to be large enough to contain the attractor, since the attractor has finite width. The values $a = 0.1$ and $b = 0.7$ are sufficient for the present implementation of the double scroll system. The distance d may be found by the formula $d = (F - |v_{C1}|) \cos \theta + |v_{C2}| \sin \theta$, where θ is the angle between the fitted line and the plane defined by $v_{C2} = 0$ [17]. By only considering the distance from the corresponding fixed point, the point of intersection is rotated slightly about the fixed point onto the line before proceeding.

In the numerical implementation, the calculation of $r_N(x)$ was done discretely by dividing up each of the cross-sections into 2001 partitions and calculating the future evolution of the

Index	d	$r_N(d, 0)$	v_{C_1}	v_{C_2}	Control
⋮					
717	0.3148	0.18750	-1.245731	0.185597	5
718	0.3151	0.18750	-1.245488	0.185774	4
719	0.3154	0.18750	-1.245246	0.185951	3
720	0.3157	0.18750	-1.245004	0.186127	2
721	0.3160	0.18750	-1.244761	0.186304	1
722	0.3163	0.21875	-1.244519	0.186481	-1
723	0.3166	0.21875	-1.244277	0.186658	-2
724	0.3169	0.21875	-1.244035	0.186835	-3
725	0.3172	0.21875	-1.243792	0.187012	-4
726	0.3175	0.21875	-1.243550	0.187189	-5
727	0.3178	0.21875	-1.243308	0.187366	-6
728	0.3181	0.21875	-1.243065	0.187542	-7
⋮					

Table 4.1 A portion of the table containing control instructions.

central point in the partition for up to 12 loops around the attractor. However, the controls were applied so that effects of a perturbation to a trajectory will be evident after only 5 loops around the attractor, i.e. $N = 5$. In addition to recording $r_5(x)$, a look-up table was constructed for each branch which contains the coordinates for the central points in the partitions, as well as instructions concerning the controls at these points. The table has one row for each bin. A portion of the table for the “0” branch appears in Table 4.1. The columns are: the row number; the distance d from the fixed point; $r_N(d, 0)$; the coordinates v_{C_1} and v_{C_2} for the central point in the partition; and control instructions telling how many partitions up or down one needs to perturb in order to change the N th symbol in the future. Because of the symmetry of the double scroll system, the table for the “1” branch is essentially the same as above: the first two columns are the same; $r_N(d, 1) = 1 - 2^N - r_N(d, 0)$; change the signs on the 4th and 5th columns; and the 6th column is the same. The control instructions may be

found well ahead of time. Since we have already found $r_N(d, \ell_0)$ for all partitions, it is simply a matter of finding a location d' on the section for which $|r_N(d', \ell_0) - r_N(d, \ell_0)| = 2^{-N}$. For example, at an intersection point x_0 of a trajectory with a cross-section, if $r_5(x_0)$ indicates that the trajectory will trace out the sequence 11001, then one searches for the bin nearest to x_0 which will give the sequence 11000 and places this information in M (if the nearest bin is not unique, then there must be an agreement about which bin to take; one may take the bin further from fixed point at the center of the loop). If, however, the current point will trace out the sequence 10011, the search will fail to find any location, near or not, which will produce the sequence 10010: this is due to the grammar limitation discussed on p. 54 for the double scroll oscillator. These impossible bit sequences result in large gaps in the values of r_N , seen in Figure 2-23.

Since the new starting point after a perturbation will visit the same lobes as the point x_0 until the N th loop, only two options at each intersection need to be considered; *control* or *no control*. Consequently, when the chaotic dynamics of the transmitter are being perturbed to trace out a given message, the set of controls which are applied can be translated into another binary sequence. It will be shown later that the map between a string of message bits and the associated binary sequence of controls changes as a function of the history of both the transmitter's chaotic dynamics and the message.

At each intersection where no control needs to be applied, the trajectory may be reset so that it starts at the central point of whatever partition it is in (this resetting process may be thought of as a system of *microcontrols*). This removes any accumulation of round-off error and minimizes the effects of sensitive dependence on initial conditions when the simulations are run, making the communication technique more robust. It also has the

effect of restricting the dynamics of the transmitter to a finite subset of the full chaotic attractor, although the dynamics still visit the full phase space. These restrictions can be relaxed by calculating $r(x)$ and M on a finer grid at the outset.

The communications possibility arises since both transmitter and receiver have a copy of $r_N(x)$ and the table M , so assuming that a protocol has been established such that the receiver knows when to start applying the controls, all that must be passed between the transmitter and receiver is the sequence of control instructions in binary form, telling the receiver when to perturb the trajectory. The table M holds the information about which partition should hold the new starting point for the perturbed trajectory, so once the receiver is told to perturb the orbit, it immediately knows where and how to achieve the desired perturbation. As the transmitter is controlled to trace out the desired trajectory, it is noted at each intersection whether or not a perturbation is needed. A “1” now indicates that a control was applied, and a “0” means that the trajectory was left to pass through the section unperturbed. This new sequence now forms the transmitted signal. The transmitted signal is thus a digital stream, which should have the added benefit of producing a more robust communication technique which could be transmitted using current hardware and could incorporate error-correction technology to produce the digital transmission. The receiver has an identical system, along with a copy of M and $r_N(x)$, so all the receiver needs is some starting point and the transmitted control information. As the receiver’s trajectory passes through some prescribed partition, or, as will be described in Section 4.3, after the receiver has been driven onto a periodic orbit, the control sequence is applied. The receiver is then controlled to follow the same dynamics as the transmitter and the message can be read simply by observing the sequence of lobes of the attractor visited by the receiver.

4.3 Remote Initialization of Receiver

One crucial element of this binary chaotic communication scheme remains to be developed. So far, it has been assumed that the transmitter and receiver have been synchronized from the start. For this scheme to be practical, there must be a mechanism in place to initialize the receiver into a known state. Promising research is currently being conducted into related questions from the perspective of impulsive differential equations [26] and several researchers have investigated the possibility of using impulses to synchronize a receiver [62, 55], or to control a chaotic system onto some regular behavior [28, 61]. Consequently, this discussion will be restricted to a numerical demonstration that it is quite easy to send a sequence of controls to the receiver which will drive the receiver onto a periodic orbit. Once on the periodic orbit, the message bits can be incorporated into the dynamics of the transmitter, with the resulting transmitted bits causing the receiver to leave the periodic orbit, which serves to alert the receiver to the beginning of the message.

At a fundamental level, when microcontrols are used in the binary communication scheme, there are only a finite number of orbits on the attractor, so periodicity of the dynamics would eventually be guaranteed. However, it is true that there are short periodic orbits which are (numerically) stabilized by the microcontrols. More importantly, it was found that the receiver could be driven onto a periodic orbit by sending it a repeating code. Different repeating codes led to different periodic orbits. The resulting periodic orbit was dependent only on the code segment which was repeated, and not on the initial state of the receiver (although the time to get on the periodic orbit can vary depending on the initial state). Consequently, it may be possible to send an initialization control sequence to the

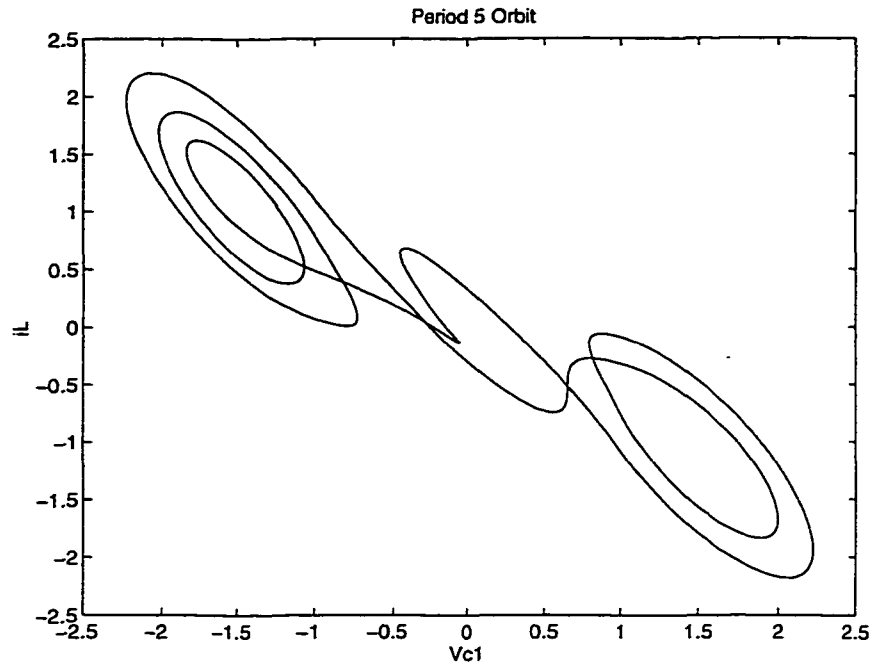


Figure 4-2 Period-5 orbit, resulting from the initialization code "01011."

receiver which drives the receiver and transmitter onto the same periodic orbit.

This initialization property will be illustrated with a single example. Using the double scroll system, the binary sequence of control instructions 01011 is repeatedly fed through both the transmitter and receiver. The result is that both the transmitting and receiving systems quickly settle on a periodic orbit, in phase (modulo the length of the periodic orbit), no matter what their separate initial conditions were. This orbit is shown in Figure 4-2. Only one slight perturbation is noticeable in the figure which is due to one of the control bits. This controlled orbit then seems to occur very close to a true unstable periodic orbit of order 5 in the uncontrolled system. The task of providing a rigorous explanation of this apparent stabilization of unstable periodic orbits is progressing and will appear in a future paper.

The remote initialization property provides the final component of the binary chaotic communication scheme. There may be other, better ways to achieve initialization, but they could simply be used to replace the current initialization scheme without changing the remainder of the method.

4.4 Binary Communication with One-Dimensional Maps

The symbolic dynamics of this scheme can be reproduced exactly by an approximate one-dimensional Poincaré map. The Poincaré surface in this case has two branches, one on each lobe of the attractor. The partitioning of the surface and the use of microcontrols allows for the easy calculation of a map which exhibits all of the symbolic dynamics of the full microcontrolled system. The evaluation of this map is much simpler and faster than integrating between intersections with the Poincaré surface. To find the map, the center point in each bin of the partition is taken as an initial condition (since these are the points to which the microcontrols “reset” trajectories) which is then integrated forward in time until the next intersection with either branch of the Poincaré surface, and the branch and distance d at which the trajectory lands is noted. For any two consecutive intersections with the surface, represented by (d_0, ℓ_0) and (d_1, ℓ_1) , this map may be written as

$$M : [a, b] \times \{0, 1\} \rightarrow [a, b] \times \{0, 1\}$$

such that $M(d_0, \ell_0) = (d_1, \ell_1)$, where the interval $[a, b]$ is the same as in Section 4.2. For a given set of integration parameters (time step, method, etc.) and for a given partition of the surface, the trajectory from the center of any bin in the partition to its next intersection

with the surface will not vary. Therefore, the map mimics exactly the behavior of the full microcontrolled system for a given integration method.

A plot of the map described above for the double scroll system is shown in Fig. 4-3. The primarily unimodal shape is not surprising, since it is known that unimodal maps can exhibit chaotic properties. However, we do not yet know, using only this map, when trajectories will switch lobes, which is crucial information about the communication scheme. By studying the structure of the double scroll attractor, seen again in Figure 4-4, it may be discerned that there is a sharp division between trajectories which remain on the same lobe and those which travel towards the opposite lobe. It turns out that, for the current parameter settings, any trajectory which begins on a section within about 0.641 units from either fixed point will intersect the Poincaré surface on the same lobe. Otherwise it will traverse to the other lobe before intersecting with the surface again. This transition point will be denoted x_{lobe} . This region is magnified in Figure 4-5. More details about the dynamics of this one dimensional, approximate map may be found in [7].

To implement this map in the communication scheme on the computer, two more columns may be placed in the instruction table corresponding to the value of $M(d_i, \ell)$ at each row i . The bin number can be calculated by the function

$$\text{index}_p(d) = \frac{d - a}{b - a}(p - 1) + 1,$$

where p is the number of bins defined by the partition. Simulated data transmission and reception using this new table is essentially the same as transmission and reception using integration. For a given intersection with the surface (d_0, ℓ_0) , the transmitter still uses the

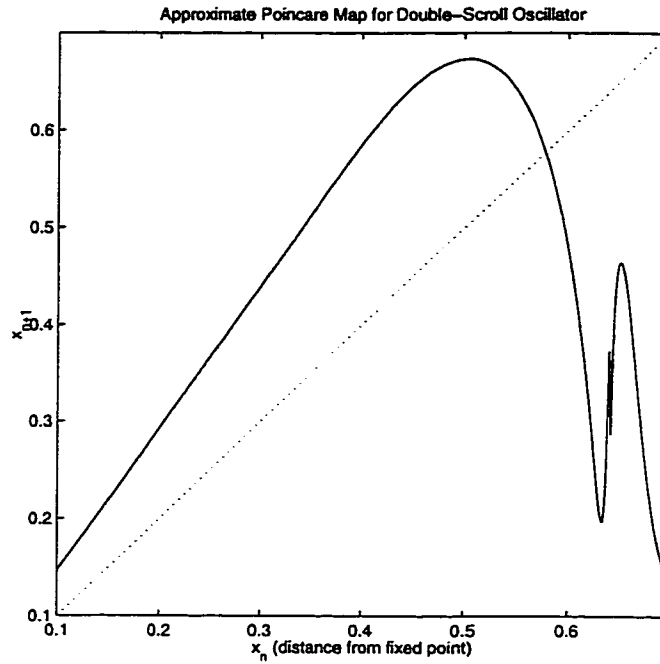


Figure 4-3 One-dimensional Poincaré map for the double scroll oscillator.

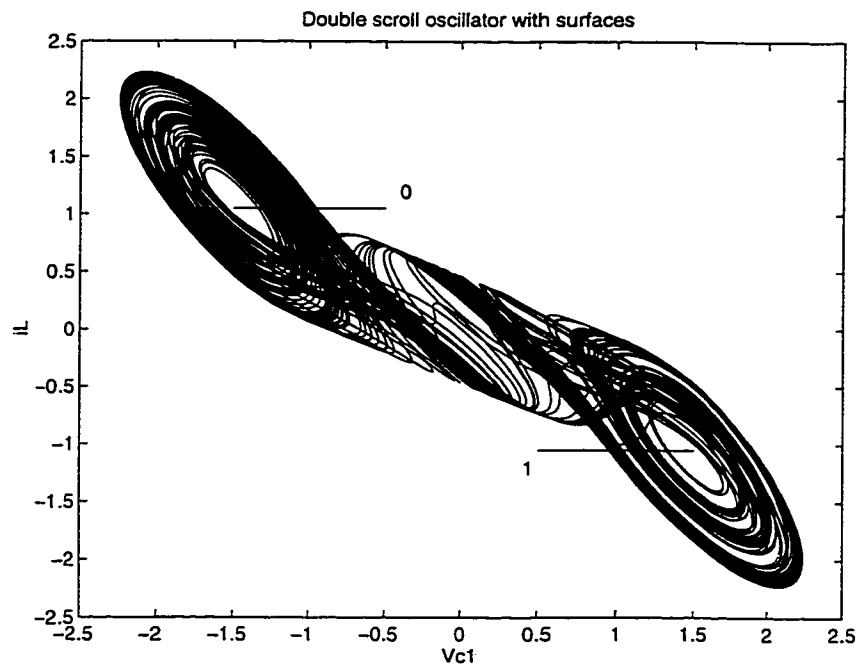


Figure 4-4 Double scroll oscillator, showing surfaces.

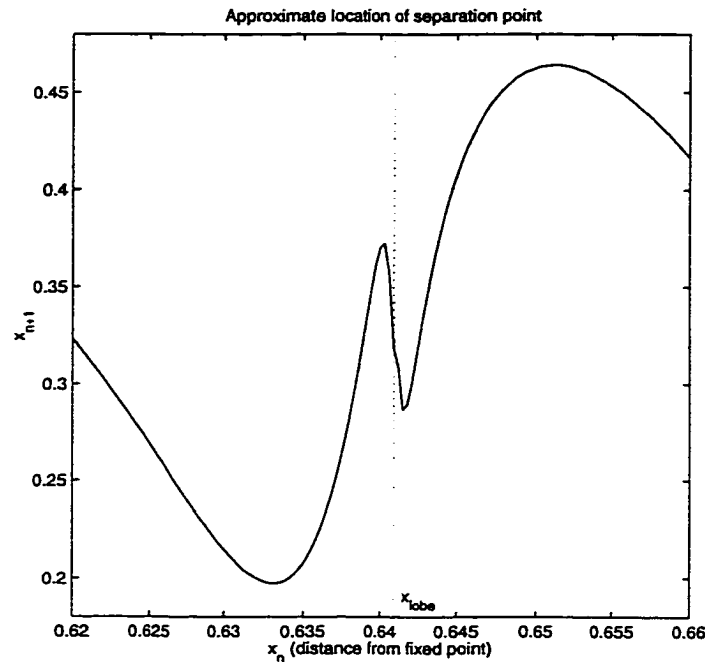


Figure 4-5 Expanded view of Figure 4-3.

function $r_N(d_0, \ell_0)$ to compare the symbolic dynamics N bits in the future with the message bit. If the N th bit in the future symbolic sequence for that bin differs from the current message bit, the control instructions are used to move to the nearest bin that will produce the desired sequence, and a “1” is sent. Otherwise, a “0” is sent. Then the new columns are used to find the location of the next intersection with the surface under the map, and the process is repeated with the next message bit. The receiver uses the map in a similar fashion, allowing the transmitted bitstream to dictate when to use the control instructions to find a new bin. The use of this map eliminates all of the time-consuming numerical integration, allowing for faster and more extensive testing.

To summarize the results so far, the entire continuous double scroll system has been replaced by an iterated, primarily unimodal chaotic map. Although in the 3-D system

there were *two* branches of the Poincaré section, one on each lobe, the attractor is perfectly symmetric about the origin, so that the Poincaré maps on both branches are identical. This allows us to iterate only one version of the map, while keeping track of the lobes with an indicator variable $\ell_i \in \{0, 1\}$.

This map differs from a conventional Poincaré map in a couple of aspects. First, our Poincaré section is two-dimensional, but it is approximated by a pair of lines extending from the unstable fixed points fitted with a least-squares method. Whenever a trajectory intersects the section, by considering only the distance from the corresponding fixed point, the point of intersection is essentially rotated about the fixed point onto the line before proceeding. In this way the three-dimensional dynamical system is reduced to a one-dimensional map. Secondly, the point of intersection is reset to the center of its current bin to simulate the microcontrols. However, since we are only concerned with the dynamics occurring within a finite number of iterations, N , of the map, the actual resolution required in this discretization may be quite coarse. Thus the microcontrols are not viewed as restricting significantly the chaotic dynamics.

Periodic orbits are of particular interest for the remote initialization of the receiver, and can be found through the study of the approximate Poincaré map defined above. Periodicity is guaranteed in the free-running (null message) microcontrolled system as soon as a trajectory lands in a bin which it has visited before. Running the microcontrolled system corresponds to iterating the Poincaré map, so periodic orbits for all points in our partition may be found by starting trajectories in each bin and iterating the map until periodicity is reached. Some preliminary results will be discussed in Section 4.3.

This reduction of the binary chaotic communication scheme to the Poincaré map version

allows for the generalization of the scheme to *any* chaotic one-dimensional map. It is simply a matter of defining “lobes”—what section of the domain implies a switching of bits—recording the symbolic dynamics in $r_N(x)$ and finding appropriate controls as before. For example, one could take the logistic map $x_n = ax_{n-1}(1 - x_{n-1})$ defined on $[0, 1]$ and arbitrarily say that for any $x_k \geq x_{lobe}$, where $0 < x_{lobe} < 1$, the next bit ℓ_{k+1} will be $\ell_{k+1} = 1 - \ell_k$; otherwise $\ell_{k+1} = \ell_k$. This provides the symbolic dynamics necessary to build the system. In Section 4.4.2 the symbolic dynamics will be simplified by explicitly labelling disjoint intervals with “1” or “0”, but for now the former definition will continue to be used, which parallels the 3-D double scroll system.

The system designer now has the freedom to improve the scheme in at least two ways. One can choose maps which would eliminate any grammar restrictions as well as optimize the system statistically, in the sense of generating random-looking bit streams. Minimally one would want a uniform distribution and a δ -like autocorrelation function—see [21] for many other statistical tests of randomness. Eliminating the grammar restrictions will help in many ways to improve the statistics. To eliminate the restriction that bits must at least come in pairs, the map has to allow trajectories to remain in the “switching” region for two or more iterations in a row. While the dynamics of the 1-D map for the double scroll system have been studied thoroughly in [7], a simple illustration will be used here to explain how to choose a map and define the symbolic dynamics such that there are fewer restrictions on the grammar.

In Figure 4-6 the 1-D map associated with the double scroll system again is shown, with the different symbolic intervals labelled and shaded. Let the function $M : I \rightarrow I$ represent the curve in the figure, where $I = [0.1, 0.7]$, and let A and B be the intervals under the

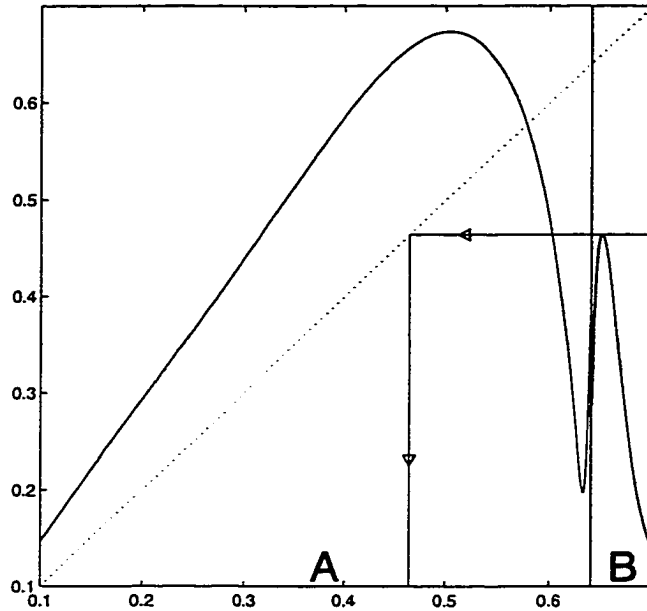
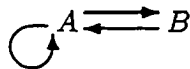


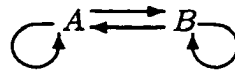
Figure 4-6 Illustrating one cause of grammar limitations

unshaded and shaded regions, respectively. Suppose $x_n \in I$ with associated bit b_n . Using the symbolic dynamics from the 3-D system, if $x_n \in B$, then $b_{n+1} = 1 - b_n$; otherwise $x_n \in A$ and $b_{n+1} = b_n$. It should be clear from the figure that if $x_n \in B$, then $x_{n+1} = M(x_n) \in A$. The rules governing the symbols then imply that $b_{n+1} = 1 - b_n$, but $b_{n+2} = b_{n+1}$. Thus there will never be more than one successive transition of bits, which eliminates the possibility that a singleton will appear in any symbolic sequence. We can summarize the possible sequence of intervals visited by a trajectory by the following simple diagram:



In order to allow a trajectory to remain in B for more than one iteration, a portion of the interval B must be mapped into itself by M . Stated more precisely, there

should exist a non-empty open set $U \subseteq I$ such that $U \subseteq B \cap M[B]$, where $M[B] = \{y \in I \mid \exists x \in B \text{ such that } y = M(x)\}$. This may be achieved simply by widening the interval B to include a fixed point of the map M , or by raising a section of $M|_B$ above the line $y = \inf B$. If B includes a fixed point of M , then trajectories will be allowed to remain in B for many more than one or two iterations. These are guidelines toward choosing a new chaotic map which has no grammar limitations. The desired diagram will then look like:



As a first example, one could consider the second iterate of the logistic map, $x_{n+1} = f(x_n) = a^2 x_n(1 - x_n)(1 - ax_n(1 - x_n))$, with $a = 3.99$. Since this map is symmetric about the vertical line $x = 0.5$, a logical choice for x_{lobe} to preserve the symmetry would be $x_{lobe} = 0.5$. All short n -bit words are possible under this map since both regions of the map contain an unstable fixed point, which may hold trajectories close for several iterations. Therefore, preprocessing the message bitstream is unnecessary. Another possibility is the simple, tent-like piecewise-linear map

$$x_{n+1} = g(x_n) = \begin{cases} 4x_n, & \text{if } 0 < x_n < \frac{1}{4}; \\ 2 - 4x_n, & \text{if } \frac{1}{4} \leq x_n < \frac{1}{2}; \\ 4x_n - 2, & \text{if } \frac{1}{2} \leq x_n < \frac{3}{4}; \\ 4 - 4x_n, & \text{if } \frac{3}{4} \leq x_n < 1. \end{cases} \quad (4)$$

This map will be used below in some of our statistical testing and experiments. Both examples are shown in Fig. 4-7.

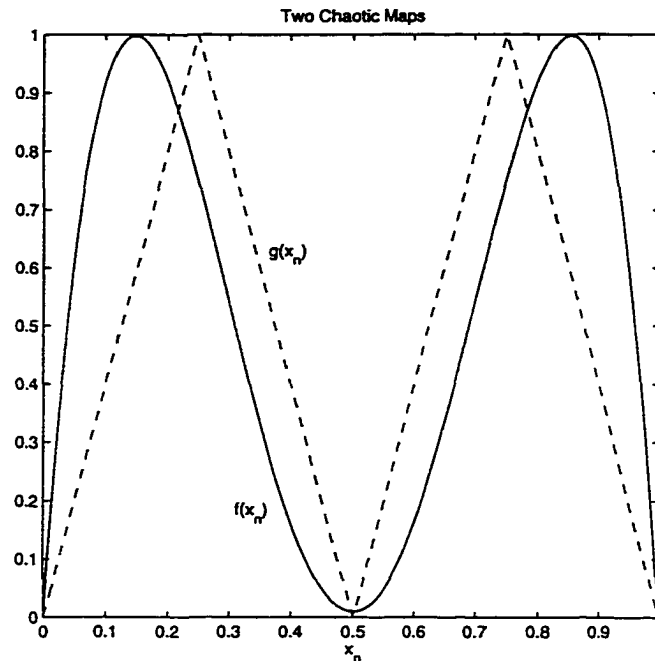


Figure 4-7 Two candidate maps for this system.

4.4.1 Initialization of map-based systems

Remote initialization of map-based systems may be achieved in exactly the same way as before. For an example of initialization using a system based on the piecewise-linear map $g(x_n)$ defined above, the sequence 1111000 was used as an initialization code as in Section 4.3. The resulting orbit for this code appears as a dotted line in Figure 4-8. Remarkably, this orbit is very close to a true periodic orbit of the uncontrolled iterated map, represented by a solid line in Figure 4-8. The slight differences between the two orbits are due to the small controls. As was mentioned before in Section 4.3, a rigorous proof of the relationship between the initialization process and unstable periodic orbits is in progress and will be the focus of future work.

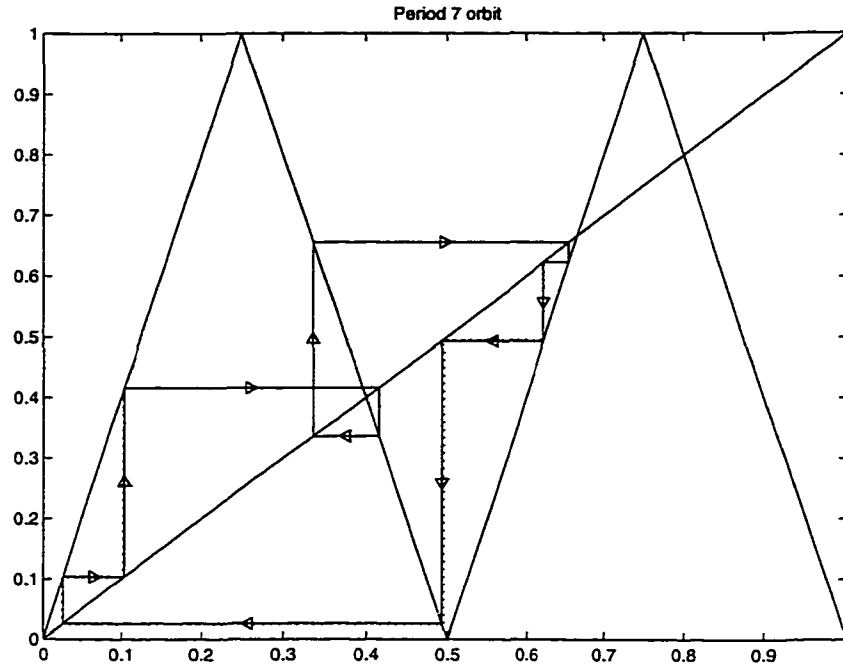


Figure 4-8 True period-7 orbit, without controls.

4.4.2 A class of one-dimensional maps with perfect statistics

The dynamics of one-dimensional chaotic maps have been thoroughly studied, but only recently have these maps been considered in a cryptographic context as pseudorandom number generators. Some maps which have been proven to exhibit a δ -like autocorrelation function may provide a good foundation for the digital chaotic communication scheme, and will be discussed here.

In addition to many recent attempts to design a secure communication scheme based on chaotic systems, there have been several proposed pseudo-random number generators based on chaotic one-dimensional maps [8, 20]. In [8], the focus is to generate numerical sequences which have a certain distribution using a simple one-dimensional map. The authors consider

the piecewise linear map

$$x_{n+1} = \mathcal{F}(x_n, \lambda) = \begin{cases} \lambda x_n & \text{if } x_n \leq \frac{1}{\lambda} \\ \frac{\lambda}{\lambda-1} (1 - x_n) & \text{if } x_n > \frac{1}{\lambda} \end{cases} \quad (5)$$

where $\lambda > 1$. The function \mathcal{F} is often called the *skew tent map*. A proof is given of the uniformity of the distribution on $[0, 1]$ of iterates $\{x_k\}$, as well as the existence and instability of periodic orbits of length n , where n is any natural number. A sequence of numbers with a desired cumulative distribution function $F(x)$ is then obtained by simply transforming the iterates according to

$$y_n = F^{-1}(x_n).$$

For example, to transform to a uniform distribution on $[a, b]$, use $y_n = (b - a)x_n + a$. For an exponential (μ, λ) distribution, transform according to $y_n = \mu - \ln(1 - x_n)/\lambda$ [8]. It is clear that, if x_n is uniformly distributed on $[0, 1]$, then the resulting sequence y_n will have the desired distribution. However, the authors of [8] repeatedly refer to the sequences as “random signals,” implying a time ordering. It is when this time ordering is considered that the structure of the one-dimensional map appears. The authors of [8] provide a table of 100 “random” points with a uniform distribution on $[-2, 2]$. This table is reproduced in Table 4.2. Plotting a delay reconstruction of these numbers as a time series with time delay equal to 1 reveals the hidden structure in this sequence (rounded to the nearest hundredth), seen in Figure 4-9. This is the simplest form of this type of reconstruction. The authors suggest that their number generator may be appropriate for Monte Carlo analysis, but it is clearly inappropriate for time-ordered cryptographic use.

n	x_n	n	x_n	n	x_n	n	x_n
1	-1.17	26	-1.36	51	0.46	76	-1.47
2	0.41	27	0.08	52	0.56	77	-0.11
3	0.80	28	1.54	53	-1.79	78	0.77
4	-1.31	29	-0.92	54	-0.66	79	-1.51
5	0.17	30	0.85	55	1.30	80	-0.18
6	1.91	31	-1.98	56	-1.47	81	0.48
7	-0.02	32	-0.99	57	-0.11	82	1.17
8	1.13	33	0.73	58	0.76	83	-1.80
9	-1.87	34	-1.82	59	-1.63	84	-0.67
10	-0.80	35	-0.71	60	-0.39	85	1.28
11	1.06	36	1.21	61	1.77	86	-1.53
12	-1.12	37	-1.70	62	-0.37	87	-0.21
13	0.51	38	-0.50	63	1.79	88	0.35
14	0.59	39	1.57	64	-0.31	89	1.73
15	-1.78	40	-0.85	65	1.90	90	-0.47
16	-0.65	41	0.98	66	-0.05	91	1.62
17	1.32	42	-1.45	67	1.00	92	-0.71
18	-1.43	43	-0.07	68	-1.36	93	1.21
19	-0.04	44	0.95	69	0.08	94	-1.70
20	1.06	45	-1.57	70	1.55	95	-0.50
21	-1.10	46	-0.28	71	-0.89	96	1.57
22	0.54	47	1.95	72	0.89	97	-0.84
23	-2.00	48	0.05	73	-1.79	98	0.99
24	-1.02	49	1.44	74	-0.66	99	-1.39
25	0.68	50	-1.14	75	1.30	100	0.03

Table 4.2 100 “random” numbers from the pseudo-random number generator in [8].

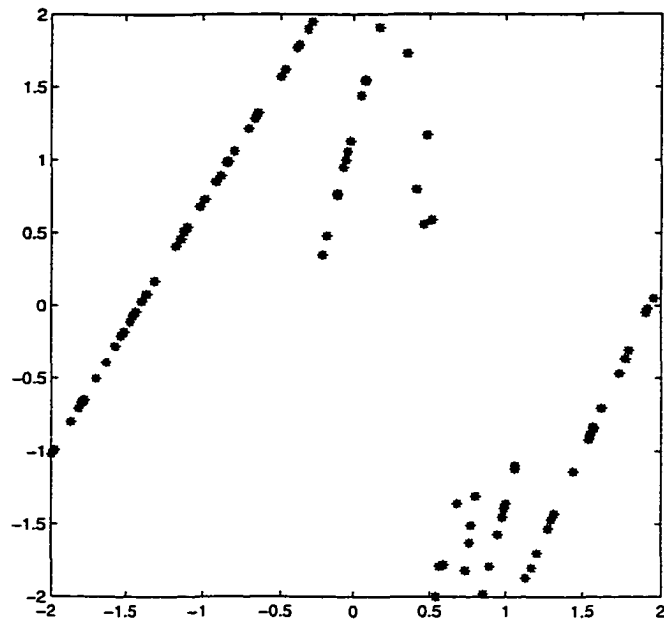


Figure 4-9 Structure in “random” numbers revealed.

The pseudo-random number generator described in [20] may provide considerably more security. Rather than taking numbers directly calculated by a one-dimensional map, the authors of [20] propose to partition the interval into subintervals labeled “1” and “0,” and letting the bits of the pseudo-random numbers be determined by the sequence of intervals through which a trajectory travels. This abstraction provides much needed separation between the chaotic attractor and the output, and denies an intruder much of the information needed to expose any hidden structure.

The class of functions considered in [20] is given by

$$\phi_p(x) = \begin{cases} -1 + 2(x - a_i)/(a_{i+1} - a_i) & \text{if } x \in [a_i, a_{i+1}) \\ \phi_p(-x) & \text{if } x \in [-1, 0) \\ 1 & \text{if } x = 1 \end{cases} \quad (6)$$

where $p \geq 1$, $0 = a_0 < a_1 < \dots < a_p < a_{p+1} = 1$, and the set $\{a_i\}$ divide $[0, 1]$ into $p + 1$ subintervals. The proof of the ergodicity and uniform distribution of these maps is similar to that which appears in [8]. Now divide the interval $[-1, 1]$ into two sets I_0 and I_1 of equal measure and which are both finite unions of intervals such that $I_0 \cup I_1 = [-1, 1]$ and $I_0 \cap I_1 = \emptyset$. When an iterate x_n falls in I_0 , a “0” is recorded; otherwise, a “1” is recorded. Bitstreams are then generated by the sequence of intervals through which a trajectory passes. The authors prove that the sequences have a delta-like autocorrelation function, the probabilities of a “1” and a “0” are equal, and the cross-correlation between any two distinct sequences is zero. In the next section, it will be shown that various reconstructions have so far revealed little structure in the bitstream.

A “key” may now be defined for this system in the following way. For any natural number n , divide the interval $[-1, 1]$ into 2^n equal subintervals, labelled in order $I_1^n, I_2^n, \dots, I_{2^n}^n$ as in [20]. Now when an iterate x_n lands in an odd-labelled interval, record a “1”; otherwise, record a “0”. We now have a “key space” that consists of the set

$$\mathcal{K} = \{(p, a_1, a_2, \dots, a_p, n) : p > 0, 0 < a_1 < a_2 < \dots < a_p < 1, n > 0\}$$

where n and p are natural numbers.

The method in [20] holds much more promise as a pseudo-random number generator than that presented in [8]. More work needs to be done in the area of detecting patterns in this type of scheme, although it seems to pass the simplest reconstruction and information measure tests. The performance of this map in the chaotic communication scheme will be compared to the tent-like map described earlier.

The symbolic dynamics may be defined in any number of ways, and for this map the implementation differed from previous work in the following manner. The symbolic dynamics had previously been defined to mimic that of the double scroll Poincaré map, where there was a “switching” region which indicated a transition of bits (or “lobes”) at the *next* iteration of the map, i.e. $b_{n+1} = 1 - b_n$. Otherwise, $b_{n+1} = b_n$. Rather than following this convention for this map, the symbolic dynamics were defined more simply by breaking the interval into subintervals labelled “0” and “1” in accord with [20]. While this slightly affects how the symbolic dynamics is recorded, the remainder of the algorithm is unchanged—controls are chosen to alter the symbolic dynamics only after the N th iteration.

4.5 Testing for Determinism

To generate a sample bitstream from the Zhou *et al* binary scheme, given by Equation (6), let $p = 1$, $a_1 = 0.33$, $I_0 = [-1, -0.5) \cup [0, 0.5)$ and $I_1 = [-0.5, 0) \cup [0.5, 1]$. The resulting map ϕ is shown in Figure 4-10.

For an experimental data set, the entire ASCII text of Shakespeare’s “King Henry V” was encoded using the double scroll Poincaré map, the tent-like map $g(x_n)$ given on page 111, and $\phi(x_n)$ described above. This provided 1,298,040 bits for statistical testing. As the first test of the binary communication process, the mapping between a sequence of message bits

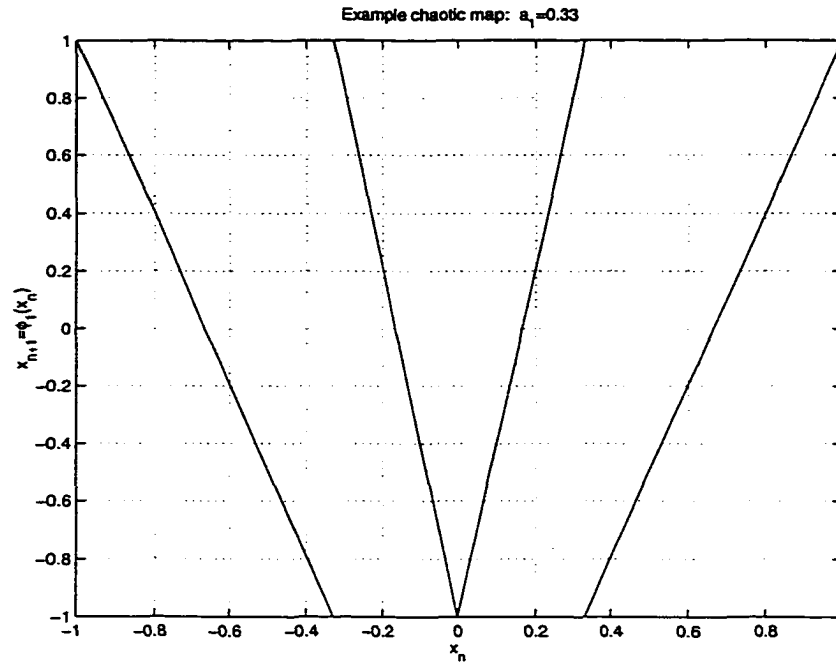


Figure 4-10 Sample map from Zhou *et al* family.

and the corresponding transmitted bits was examined, and it was found that the process was many-to-one and one-to-many. For example, Figure 4-11 illustrates in histograms the mapping between 3-bit words in the message and the corresponding 3-bit words in the transmission for the map $g(x_n)$ using the “King Henry” data. The numbers on the x-axes and in the titles represent base-10 conversions of 3-bit words, i.e. $3_{10} = 011_2$, $5_{10} = 101_2$, etc. The histograms show that a given sequence of message bits can be encoded in many ways: similarly, a given sequence of transmitted bits can represent many different sequences of message bits. It is only the dynamics of the chaotic transmitter which allows the proper meaning to be discerned, and the encoding is entirely dependent on the history of *both* the chaos and the message.

An attempt was made to investigate whether it is possible to find a way to construct some

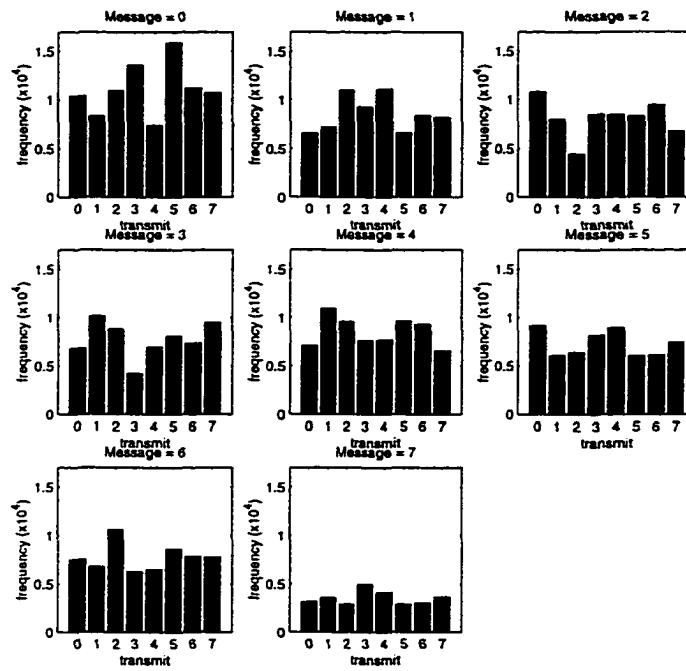


Figure 4-11: Histograms showing encodings of all possible 3-bit message words into 3-bit transmitted words.

kind of dynamical model from the transmitted bit stream. In order to do this, sequences of bits from the transmitted signal were interpreted as integer binary numbers. By considering sequences of bits, it is possible to consider statistical and dynamical tests for determinism. To do this, sequences of 4, 8, and 16 bits taken from the transmitted signal were considered. As an example, consider the following sequence of transmitted data:

0011100011001011011100001100100101001111110...

Assume that disjoint sequences of 4 bits are to be interpreted as decimal numbers. Then the first “data value” would be $s_1 = 0011_2 = 3$, the second would be $s_2 = 1000_2 = 8$ and so on.

For these tests $N = 7$. The autocorrelation of the transmitted bit stream using $g(x_n)$ gives the δ -like plot at the top of Figure 4-12. If the cross correlation between the message bit stream and the transmitted bit stream is calculated, the result in the second plot in Fig. 4-12 shows that there is very little correlation, since a strong correlation at any lag would have a value close to 1. However, there appears to be remaining evidence of the 8-bit ASCII structure in the cross-correlation plot, seen as a small bump occurring at regular intervals. The correlation plots, seen in Figure 4-13, of the transmitted bit stream using the map ϕ reveal none of the ASCII patterns.

As the next test, it was considered whether the reconstructed data points taken from the disjoint time series would fill all possible positions in phase space. The results shown here will be limited to 2-dimensional reconstructions, but it will be shown below that the results will hold in higher dimensions as well. In Fig. 4-14, reconstructed data points of the form $x_i = (s_i, s_{i+1})$ are plotted, where s_i is defined as above, except that disjoint 16-bit

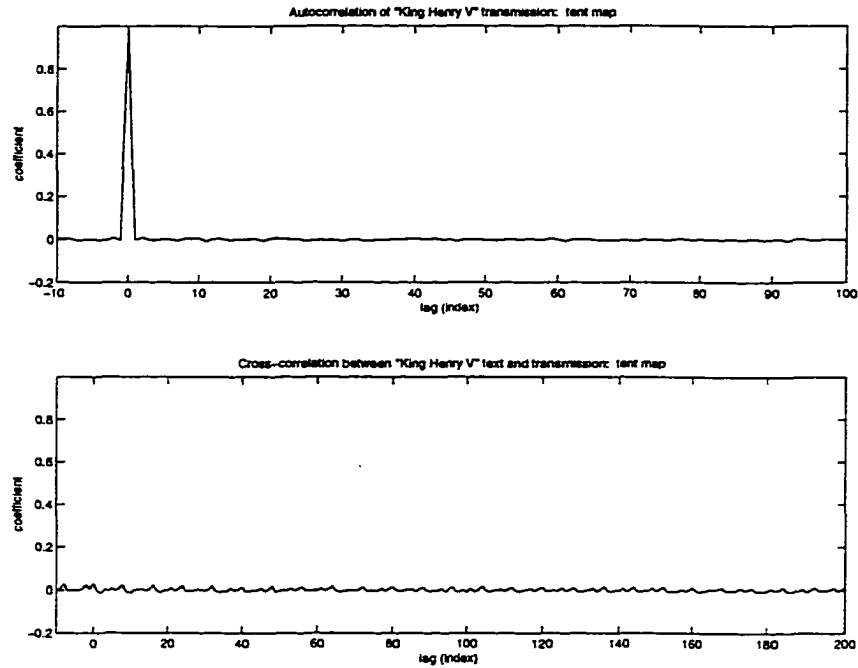


Figure 4-12 Correlation plots for the “King Henry V” data using the map $g(x_n)$.

sequences are used. This plot shows that most grid points are covered for 16-bit sequences. Consequently, it seems as though it will not be possible to use reconstructions to find a distinguished subset of reconstructed points that can be used to determine the state of the hidden chaotic transmitter.

Metric (or Kolmogorov) entropy measurements provide a way to quantify the rate at which dynamical data fills k -dimensional phase space, based on information theory [42]. If the metric entropy is denoted K , then $K = 0$ for periodic or quasi-periodic data, $0 < K < \infty$ for chaotic data and $K \rightarrow \infty$ for random data. The presence of predictable structure would lead to a lower entropy value. A plot of the metric entropy measurements for the sample bit stream for $1 \leq k \leq 22$ can be seen in Fig. 4-16, where it is superimposed on the entropy plot of random data. It is clear that the transmission does not reveal any nonrandom sequence

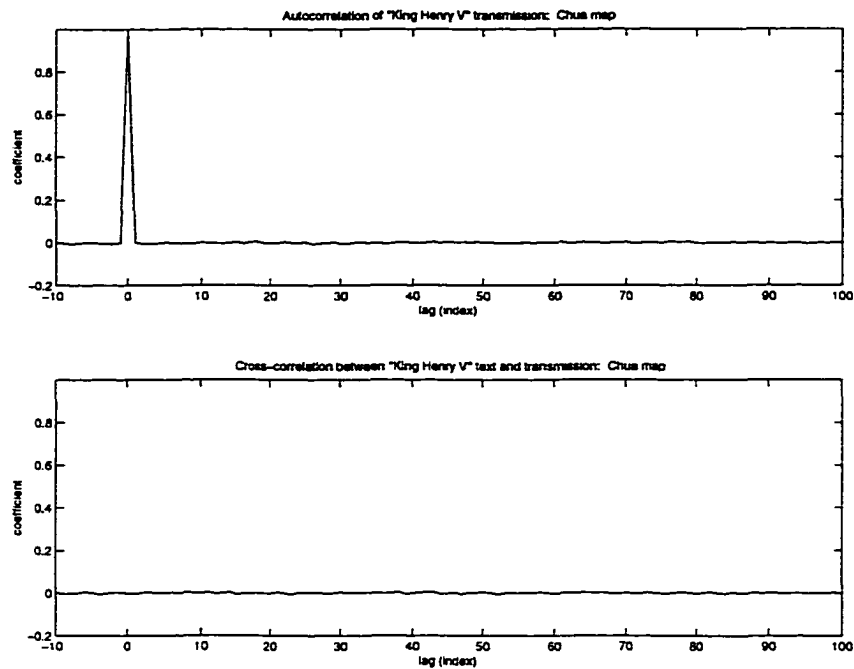


Figure 4-13 Correlation plots for the “King Henry V” data using the map $\phi(x_n)$.

structure for reasonable sequence lengths (the deviations at the upper end are more likely to be related to the size of the data set—larger data sets would have to be used to probe a regime of longer sequences). The statistical imperfections remaining in the tent map transmission seem to have been eliminated or significantly reduced by this type of chaotic map.

It was considered whether there was a consistent pattern to the dynamical evolution of the reconstructed points. To examine this question, phase space reconstructions were created to search for a consistent flow pattern, i.e. for some regularity to the plotted points or to the dynamical behavior as $s_1 \rightarrow s_2 \rightarrow s_3 \dots$. The hope was that if any predictable dynamical behavior was revealed, it might be possible to determine something about the hidden chaotic system which composes the transmitter and receiver. However, it was not surprising to find that the flow patterns appeared as random lines connecting the grid points.

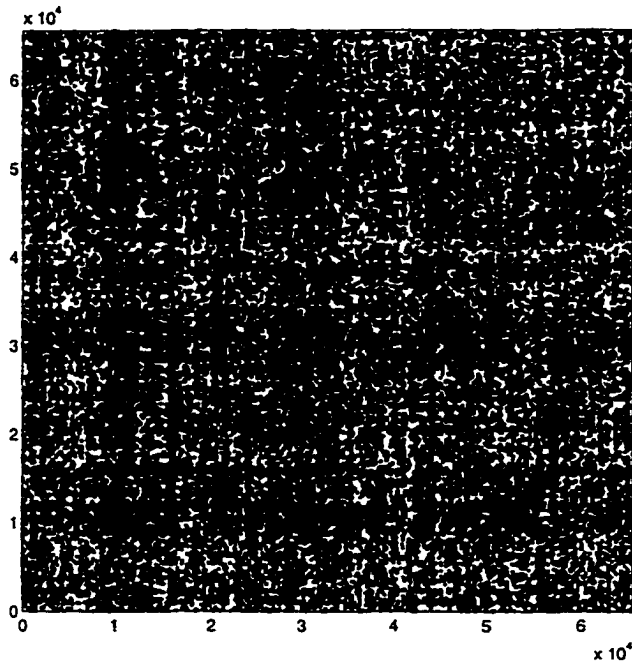


Figure 4-14 16-bit reconstruction of “King Henry V” transmission using $g(x_n)$, $N = 7$.

so there was no dynamical information which could be gleaned from the transmitted data, and NLD forecasting was completely ineffective.

The important aspect of the development of this communication technique is that since the transmitted signal is just a binary sequence, there is no information which can be used to produce a time-delay phase space reconstruction in the usual sense. Consequently, the techniques which were used to break chaotic communication schemes in [48, 45, 43, 44] are no longer applicable to this problem, since there is no obvious way to extract geometric information from the transmitted signal. Even from these preliminary tests, it appears that this chaotic communication scheme is much more difficult to analyze from an NLD perspective than earlier chaotic communication techniques which transmitted a chaotic waveform. This does not mean that other techniques will not work.

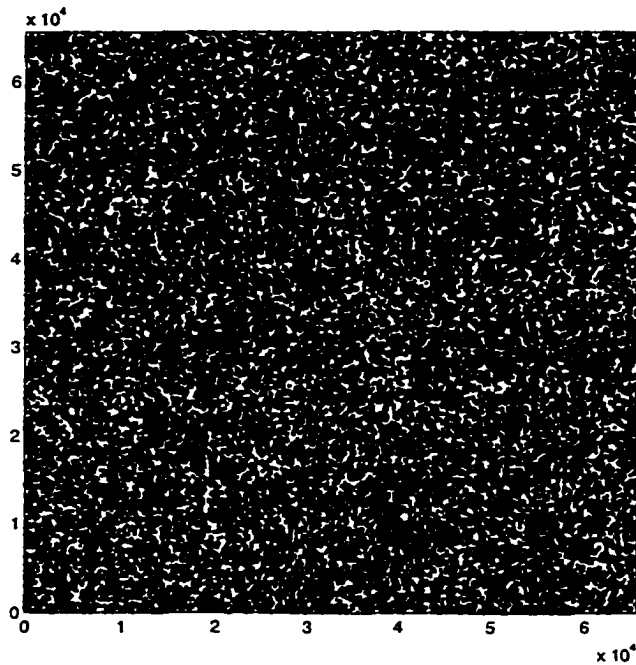


Figure 4-15 16-bit reconstruction of “King Henry V” transmission using $\phi(x_n)$, $N = 7$.

4.6 Cryptographic context

To understand the advantages of this digital chaotic communication scheme when compared to existing encryption algorithms, it is helpful to review some of the general classes of encryption methods. Then common features will be discussed as well as those features which set this scheme apart. Much of the following information on standard encryption techniques has been condensed from Schneier’s book *Applied Cryptography* [40].

There are two general classes of encryption algorithms—block ciphers and stream ciphers. A block cipher performs its functions on sequential blocks of data, usually 64 bits at a time. A stream cipher operates on individual bits or bytes. While the digital chaotic communication scheme may be facially classified as a stream cipher based on these defini-

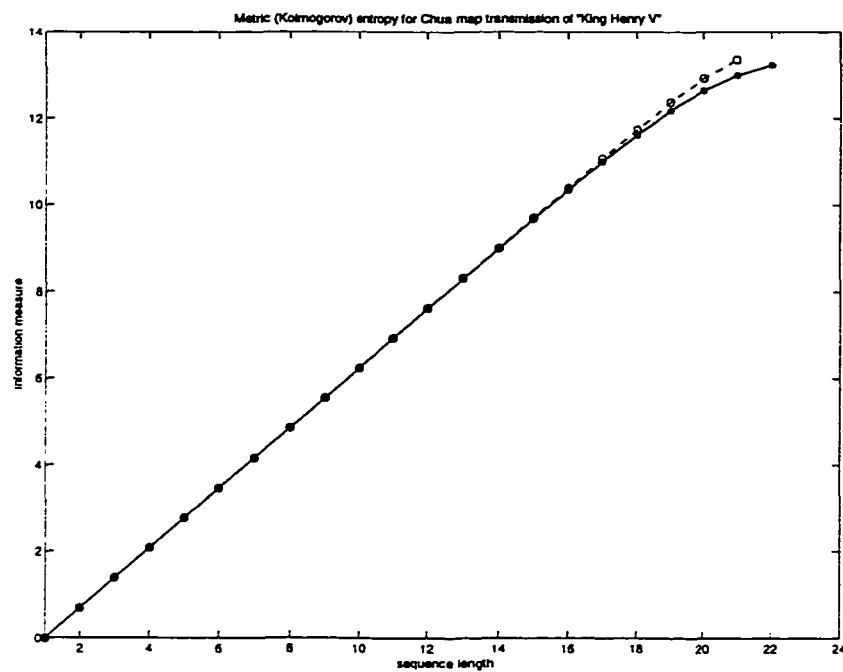


Figure 4-16: Kolmogorov entropy of “King Henry V” transmission using the map $\phi(x_n)$, $N = 7$.

tions, there are characteristics of both types of schemes which are shared and need to be discussed.

There are several different modes in which traditional encryption algorithms may operate. These are usually a combination of feedback loops, chaining and simple operations which assist in hiding patterns in the plaintext and randomizing the input to the encryption algorithm. These modes have various effects on the security, efficiency and error-propagation properties of the implementation of the pure algorithms.

4.6.1 Block cipher modes

The simplest implementation of a block cipher is to encode each block of plaintext with a given algorithm and transmit the result without any feedback loop. A given (e.g., 64-bit) ciphertext block is dependent *only* on the corresponding (64-bit) plaintext block. This is called **electronic code book** mode, or **ECB**. The name comes from the purely theoretical possibility that one could keep track of all possible corresponding ciphertext and plaintext blocks in a code book, since the same block of plaintext will always encrypt to the same block of ciphertext. If K is any given key, E_K and D_K are the encryption and decryption functions, respectively, and P_i and C_i are the i th plaintext and ciphertext blocks, respectively, then this mode is easily represented by

$$C_i = E_K(P_i)$$

$$P_i = D_K(C_i).$$

The main advantage of this mode is that since there is no interdependency between blocks, encryption and decryption may be done in parallel, improving the efficiency. This also makes it a good mode to implement for access to files in a database. The main disadvantages are that patterns in the plaintext are not completely hidden, and that the plaintext is easy for an intruder to manipulate; it is difficult to detect when blocks are omitted, repeated or rearranged.

The other block cipher mode which will be discussed here (there are several others) is called **cipher block chaining (CBC)**. In this mode each plaintext block is added bitwise (modulo 2) to the previous ciphertext block before encryption. Using the notation from above, this can be denoted

$$C_i = E_K(P_i \oplus C_{i-1})$$

$$P_i = C_{i-1} \oplus D_K(C_i)$$

where “ \oplus ” is bitwise addition modulo 2. Thus a given ciphertext block is dependent not only on the corresponding plaintext block but also on every previous plaintext block. This mode helps to obscure patterns in the plaintext. In order to start the chaining and to randomize the first block of plaintext, an **initialization vector (IV)** is needed. This does not need to be kept secret, since the security of any good encryption algorithm should be in its key. It is important to change the IV for each message, but it does not need to be random. CBC is generally best used for encrypting files.

4.6.2 Stream cipher modes

Stream ciphers are methods of generating a stream of bits, k_i , called the **key stream**, which is then added modulo 2 to the bits in the plaintext to form the ciphertext, or $c_i = p_i \oplus k_i$. The key stream must look random, yet it must be deterministic in order to be reproduced exactly at the receiving end where the message is recovered by $p_i = c_i \oplus k_i$. Most common stream cipher modes have an analogous block cipher mode, where individual bits are replaced by blocks.

The simplest mode is called a **synchronous stream cipher**, where the key stream is independent of the plaintext or ciphertext, although the key stream may be fed back into the key stream generator. This mode has the advantage that bit errors in the ciphertext occurring in transmission will only affect the corresponding bit in the plaintext upon decryption. The main disadvantage is that a loss of synchronization is unrecoverable, unless there is some structure present which allows both the sender and receiver to resynchronize. On the other hand, this can be viewed as a security advantage, since manipulative insertions or deletions by an intruder will be immediately detected.

A **self-synchronizing stream cipher** generates each key stream bit using a function of a key and a fixed number of *ciphertext* bits. In this mode the receiver will be fully synchronized with the transmitter as soon as n ciphertext bits are received. This eliminates the need for an IV—the sender can encrypt n bits of random data before starting the message. The receiver will decrypt the first n bits incorrectly, but thereafter will be synchronized just in time to decrypt the message properly. Also, if synchronization is lost during the transmission, it will be regained after n bits, unlike the synchronous stream cipher mode.

where a loss of synchronization is unrecoverable. But this means that one bit error in the ciphertext results in n errors in the plaintext, an increase in the error expansion over the previous mode.

The main advantage which the digital chaotic communication presented in this chapter has over other chaotic communication systems is that the state space of the transmitter is completely hidden. Hiding the state space of the transmitter was the primary objective in its design.

The digital chaotic communication scheme has several characteristics in common with established encryption methods, but it is impossible to classify fully this scheme using traditional cryptographical terminology. The main difference is that the internal state of standard cryptographical algorithms are composed of shift registers or blocks of bits, whereas the digital chaotic scheme's internal state involves an orbit on a chaotic attractor. This issue of cryptography using discrete versus continuous sets has been discussed briefly in [15, 25]. The notion of a "key" is also unique for this system. The chaotic system may be chosen from a large class of simple one-dimensional maps, each of which may be compactly represented. The maps along with the number N of bits chosen to record the symbolic dynamics as well as the choice of controls form a very large key space.

Aside from these unique characteristics, because the ciphertext is generated one bit at a time, this scheme may be classified as a type of a stream cipher. The following table summarizes some of the security, efficiency and error propagation properties of these communication methods.

Property	Block cipher modes		Stream cipher modes		Digital Chaotic
	ECB	CFB	Synchronous	Self-synchron.	
Initializa- tion	not needed	needs IV, does not need to be secret, should change for each message	needs an IV, does not need to be secret, must change for each message	not needed: initialization may be done dynamically	dynamic: receiver may be initialized remotely, initialization does not need to be secret
Bit Error Propaga- tion	corrupts one full block of plaintext	corrupts one full block of plaintext plus one bit in the next block	corrupts only one bit in the plaintext	corrupts n bits of plaintext	unrecoverable
Synchron. Error	unrecoverable	unrecoverable	unrecoverable	resynchronizes after n bits	unrecoverable
Security	plaintext patterns not concealed; plaintext easily manipulated by an intruder	plaintext patterns are very well concealed; harder to manipulate plaintext undetected	patterns well concealed; plaintext easy to manipulate bitwise, but insertions and deletions are immediately detectable	patterns well concealed; plaintext harder to manipulate	patterns are very well concealed by the chaotic dynamics; plaintext very hard to manipulate since all changes to ciphertext are immediately detected
Efficiency/ Applica- tions	most efficient: very good for software implementations, especially for random-access databases	slightly less efficient: good for software implementations, especially for general data encryption and storage	most efficient in a hardware implementation	excellent choice for constant flow of data, e.g. cellular phones or T-1 computer connections	has the hardware-efficiency advantage of a stream cipher, but is also easily implemented in software

While a synchronization error causes difficulty with most schemes, including the present system, it is common to impose structures on the transmitted bit stream which would allow the receiver to resynchronize with the transmitter. Bit errors can also be corrected using existing error-correction technology, so that neither type of error shown in the table above are foreseen to pose a problem with the implementation of this system.

The advantages of this digital chaotic communication scheme include:

- **Efficiency**—One of the reasons chaotic systems are increasingly being considered as a basis for cryptographic algorithms [63, 4, 30] is that they produce highly complex dynamics from very simple mathematical equations. This makes the digital chaotic scheme very efficient to implement, particularly in silicon, but also in software.
- **Security**—The chaotic system to be used may be chosen such that the output has perfect statistics, no matter how repetitive the plaintext. It also entirely defends itself against intruders attempting to manipulate the plaintext, since any change to the transmission will be immediately detectable. Also, the nonlinear dynamic forecasting attack which has been successful against all major proposed chaotic communication systems has been completely ineffective.
- **Initialization**—The receiver may be remotely synchronized with the transmitter dynamically, similar to the self-synchronizing stream cipher, without any need to exchange an IV.
- **Laser Technology**—There is very good potential that this communication scheme may be combined with current research in chaotic lasers to provide an extremely high data transmission rate.

4.7 Discussion and Conclusions

The work in this chapter represents just a first step toward developing chaotic communication schemes which bring together aspects of chaotic control, impulsive differential equations, and transmissions which hide the phase space. The key elements of this binary chaotic communication scheme are the fact that the message and transmitted bit streams are independent,

the encoding of message bits to transmitted bits is many-to-one while the decoding of transmitted bits to recovered message bits is one-to-many, and the binary information passed between transmitter and receiver cannot be used for reconstructions or NLD forecasting. Also, the ability to do remote initialization is especially interesting, since it is in some ways equivalent to communicating without key exchange. Instead, the security aspects are encompassed in the shared knowledge of $r_N(x)$ and M , as well as the fact that the transmitter and receiver circuits would have to be well-matched or tuned to behave the same. There is work underway on problems related to remote initialization [61], but there is a great deal more which needs to be done on the theoretical side. This method of communication completely foils the NLD forecasting attack, at least to date. Of course, that does not imply that the method is particularly secure, so potential security weaknesses will be discussed below.

An interesting perspective on the binary communication approach can be gained by considering the chaotic transmitter as a key generating device. In fact, as long as the microcontrols are non-zero, the system will have only a finite number of possible trajectories, so it is fair to consider this as a key generation scheme. From this viewpoint, the interesting aspect of this approach is that the “key” would change at every iteration, but the changes would not follow a pre-determined functional pattern and would, in fact, be a function of the previous history of the chaos and the message. This would alter the nature of a brute-force attack on the transmission, since it would make little sense to try all possible keys when the key changes at every iteration. Other attacks might be more successful.

In [63] the maps discussed in Section 4.4.2 are presented in the context of a pseudo-random keystream generator for one-time pad cryptographic systems. Problems with keystream cycling in digital implementations of chaotic maps have been discussed in [59, 58, 2]. How-

ever, the digital chaotic communication system described in this chapter is a *message modulated* system, where each iteration is dependent on the previous history of both the state of the system and the message itself. Thus as long as there are no cycles in the message, there will not be any cycles in the keystream. Even if there are some repeated elements in the plain text, as long as the length of the repeated message segments are incommensurate with any possible natural cycle lengths of the chaotic system, there will be no cycles in the keystream.

Now that the positive aspects of the approach have been discussed, it is worth considering potential security flaws. The first obvious security flaw is that the microcontrols stop the dynamics from being truly chaotic. It is certainly possible to calculate $r(x)$ and M on a much finer grid, thereby reducing the size of the microcontrols. In theory, the microcontrols could be eliminated; however, with real circuitry, some level of stabilization would probably be necessary. The most obvious weakness of the scheme is also one of its most desirable properties, namely the remote initialization of the receiver. In this case, since the initialization code is repeated, it is detectably periodic with a short period. This would flag the beginning of the message for the party intercepting the transmission. Again, it remains to be seen how this could be exploited. However, a protocol could be developed where the initialization code is sent only once, perhaps hidden among some random bits, or encrypted using a public-key algorithm.

One important consideration in determining the security potential of a communication scheme is that it is usually the case that the method of communication is assumed to be known, so the security must be in something like private keys. For the binary communication scheme discussed here, that would imply that the system generating the chaos would be

known (although it is arguable whether the intercepting party would need to know all of the operating parameters of the circuitry). If it is assumed that the transmitter and receiver systems are known, the security in this approach lies in the private function $r(x)$, which could be calculated using any number of loops around the attractor, as well as the perturbation rules stored in M . As the method has been described here, a brute-force approach to breaking the transmission would be to calculate a set of functions $r_i(x)$, where i represents the number of loops which were used in the calculation. Then, the intercepting party could try each potential key function sequentially until the message was decoded. Of course, the goal of the system designers must be to try to make this a difficult calculation, thereby achieving some degree of computational security.

The final message is that it appears to be possible to create chaotic communication schemes which nullify the NLD forecasting phase-space attack. However, while it appears that the binary chaotic communication scheme developed here achieves this goal, it does not necessarily imply that the technique provides any practical security. Such considerations can only be made after studying the method from the perspective of key generation schemes. It is hoped that further developments in chaotic communication using impulsive control and a binary communication channel will lead to a communication scheme which provides a high degree of computational security.

Chapter 5

Future Directions

Current research is continuing along the interface between nonlinear dynamics and cryptography. Skeptics would say that there is no legitimate overlap between the fields—the author believes that the potential exists for a beneficial interchange between these two subdisciplines of mathematics, but that there is much work left to be done. In particular, for trustworthy algorithms to be developed, there must be a balance between the designers of chaos-based cryptographic algorithms and the evaluators of the corresponding security claims, or “cryptanalysts.” It seems at present that the scale is tipped heavily towards those that propose systems with limited security testing. It is the author’s intent to continue to examine new work in chaotic cryptography for security weaknesses. As an example, a recent paper by Minai and Pandian [32] presents a system which uses elements from the scheme by Yang *et al* [63], which was shown to have serious flaws in Chapter 2. It is important that these flaws be discussed openly.

To aid in the reconciliation of chaotic cryptography to traditional cryptographic algorithms, the author intends to broaden his background in the field of standard pseudorandom number generation. This background would help put newer results in a more realistic and traditional context.

In addition, the potential for application of the HGO control scheme seems far from exhausted. The author intends to investigate the possibility of using the control system in a data compression algorithm. Also, the many well-studied 2-dimensional chaotic maps

may provide an interesting source of systems for the binary chaotic communication scheme presented in Chapter 4.

The pursuit of a chaotic communication scheme which offers a high level of security has generated some interesting mathematical results, and continues to produce ideas that must be evaluated carefully and objectively.

Bibliography

- [1] A.M. Albano, J. Muench, C. Schwartz, A.I. Mees, and P.E. Rapp. Singular value decomposition and the grassberger-procaccia algorithm. *Phys. Rev. A*, 38:3017–3026, 1988.
- [2] Ross Anderson. Letter to the editor—chaos and random numbers. *CRYPTOLOGIA*, 16(3):226, 1992.
- [3] Ray Brown and Leon O. Chua. Clarifying chaos: Examples and counterexamples. *Intl. J. of Bifurcation and Chaos*, 6(2):219–249, 1996.
- [4] John M. Carroll, Jeff Verhagen, and Perry T. Wong. Chaos in cryptography: The escape from the strange attractor. *CRYPTOLOGIA*, 16(1):52–72, 1992.
- [5] T.M. Carroll and L.M. Pecora. Synchronizing chaotic circuits. *IEEE Trans. Circuits and Systems*, 38:453–456, 1991.
- [6] Leon O. Chua, Motomasa Komuro, and Takashi Matsumoto. The double scroll family. *IEEE Transactions on Circuits and Systems*, CAS-33(11):1072–1118, November 1986.
- [7] Leon O. Chua and Irene Tichonicky. 1-d map for the double scroll family. *IEEE Transactions on Circuits and Systems*, 38(3):233–243, 1991.
- [8] Leon O. Chua, Yong Yao, and Qing Yang. Generating randomness from chaos and constructing chaos with desired randomness. *International Journal of Circuit Theory and Applications*, 18:215–240, 1990.
- [9] Kevin M. Cuomo, Alan V. Oppenheim, and Steven H. Strogatz. Synchronization of lorenz-based chaotic circuits with applications to communications. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 40(10):626–633, October 1993.
- [10] K.M. Cuomo and A.V. Oppenheim. Chaotic signals and systems for communications. In *Proc. IEEE ICASSP*, Piscataway, NJ, 1993. IEEE.
- [11] K.M. Cuomo and A.V. Oppenheim. Circuit implementation of synchronized chaos with applications to communications. *Phys. Rev. Lett.*, 71:65–68, 1993.
- [12] K.M. Cuomo, A.V. Oppenheim, and S.H. Strogatz. Robustness and signal recovery in a synchronized chaotic system. *Intl. J. of Bifurcation and Chaos*, 3:1629–1638, 1993.
- [13] A.M. Fraser. Reconstructing attractors from scalar time series—a comparison of singular system and redundancy criteria. *Physica D*, 34:391–404, 1989.
- [14] A.M. Fraser and H. Swinney. Independent coordinates for strange attractors from mutual information. *Phys. Rev. A*, 33:1134–1140, 1986.

- [15] Jiri Fridrich. Symmetric ciphers based on two-dimensional chaotic maps. *International Journal of Bifurcation and Chaos*, 8(6), 1998.
- [16] K.S. Halle, C.W. Wu, M. Itoh, and L.O. Chua. Spread spectrum communications through modulation of chaos. *Intl. J. of Bifurcation and Chaos*, 3:469–477, 1993.
- [17] S. Hayes, C. Grebogi, and E. Ott. Communicating with chaos. *Physical Review Letters*, 70:3031, 1993.
- [18] R. He and P.G. Vaidya. Analysis and synthesis of synchronous periodic and chaotic systems. *Phys. Rev. A*, 46:7387–7392, 1992.
- [19] M. Henon. On the numerical computation of poincaré maps. *Physica D*, 5:412–414, 1982.
- [20] Zhou Hong and Ling Xieting. Generating chaotic secure sequences with desired statistical properties and high security. *International Journal of Bifurcation and Chaos*, 7(1):205–213, 1997.
- [21] Donald E. Knuth. *The Art of Computer Programming*, volume 2. Addison-Wesley, Reading, Massachusetts, 2 edition, 1981.
- [22] L. Kocarev and U. Parlitz. General approach for chaotic synchronization with applications to communication. *Phys. Rev. Letters*, 74:5028, 1995.
- [23] Lj. Kocarev, K.S. Halle, K. Eckert, L.O. Chua, and U. Parlitz. Experimental demonstration of secure communications via chaotic synchronization. *Intl. J. of Bifurcation and Chaos*, 2:709–713, 1992.
- [24] Lj. Kocarev and Toni D. Stojanovski. A model for secret-key cryptography using chaotic synchronisation. In *Proceedings of the International Symposium on Information Theory & Its Applications 1994*. Institution of Engineers, Australia, 1994.
- [25] Ljupčo Kocarev, Goce Jakimoski, Toni Stojanovski, and Ulrich Parlitz. From chaotic maps to encryption schemes. In *IEEE International Symposium on Circuits and Systems*, volume 4, pages 514–517, 1998.
- [26] V. Lakshmikantham, D.D. Bainov, and P.S. Simeonov. *Theory of Impulsive Differential Equations*. World Scientific, Singapore, 1989.
- [27] Edward N. Lorenz. Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20:130–141, 1963.
- [28] M.A. Matias and J. Guemez. Stabilization of chaos by proportional pulses in the system variable. *Phys. Rev. Lett.*, 72:1455–1458, 1994.
- [29] Takashi Matsumoto, Leon O. Chua, and Motomasa Komuro. The double scroll. *IEEE Transactions on Circuits and Systems*, CAS-32(8):797–818, August 1985.
- [30] Robert Matthews. On the derivation of a “chaotic” encryption algorithm. *CRYPTOLOGIA*, 13(1):29–42, 1989.

- [31] A.I. Mees, P.E. Rapp, and L.S. Jennings. Singular value decomposition and embedding dimension. *Phys. Rev. A*, 36:340–346, 1987.
- [32] Ali A. Minai and T. Durai Pandian. Communicating with noise: How chaos and noise combine to generate secure encryption keys. *CHAOS*, 8(3):621–628, 1998.
- [33] Kaveh Pahlavan and Allen H. Levesque. *Wireless Information Networks*, page 265. John Wiley and Sons, Inc., 1995.
- [34] A.T. Parker and K.M. Short. A binary chaotic communication scheme with improved security aspects. To be submitted to *Intl. J. of Bifurcation and Chaos*, 1997.
- [35] A.T. Parker and K.M. Short. An impulsively initialized binary chaotic communication scheme. To be submitted to *Phys. Lett. A*, 1997.
- [36] U. Parlitz, L.O. Chua, Lj. Kocarev, K.S. Halle, and A. Shang. Transmission of digital signals by chaotic synchronization. *Intl. J. of Bifurcation and Chaos*, 2:973–977, 1992.
- [37] L.M. Pecora and T.M. Carroll. Synchronization in chaotic systems. *Phys. Rev. Lett.*, 64:821–824, 1990.
- [38] L.M. Pecora and T.M. Carroll. Synchronized chaotic signals and systems. In *Proc. IEEE ICASSP*, Piscataway, NJ, 1992. IEEE.
- [39] O.E. Rossler. An equation for continuous chaos. *Phys. Lett. A*, 57:397, 1976.
- [40] Bruce Schneier. *Applied Cryptography: Protocols, Algorithms, and Source Code in C*. John Wiley & Sons, Inc., New York, 1996.
- [41] Mischa Schwartz. *Information Transmission, Modulation, and Noise*. McGraw-Hill, New York, 1990.
- [42] Kevin M. Short. Direct calculation of metric entropy from time series. *J. Computational Phys.*, 104(1):162, 1993.
- [43] K.M. Short. Computational issues in unmasking chaotic communication. Invited paper delivered at Physics Computing '95 conference, June 1995.
- [44] K.M. Short. Steps toward unmasking secure communications. *Intl. J. of Bifurcation and Chaos*, 4:959–977, 1994.
- [45] K.M. Short. Unmasking a modulated chaotic communications scheme. *Intl. J. of Bifurcation and Chaos*, 6:367–375, 1996.
- [46] K.M. Short. Detection of teleseismic events in seismic sensor data using nonlinear dynamic forecasting. *Intl. J. of Bifurcation and Chaos*, 7:1833–1845, 1997.
- [47] K.M. Short. Nonlinear dynamic forecasting for echo detection and removal in nonlinear signals. Submitted to *IEEE Transactions on Speech and Audio Processing*, 1997.
- [48] K.M. Short. Signal extraction from chaotic communications. *Intl. J. of Bifurcation and Chaos*, 7:1579–1597, 1997.

- [49] K.M. Short and A.T. Parker. Detection of controls in a controlled chaotic communication scheme, Oct. 1996. Technical Report CRASP012132.
- [50] K.M. Short and A.T. Parker. Development of a binary chaotic communication scheme with improved security, Nov. 1996. Technical Report CRASP012132.
- [51] K.M. Short and A.T. Parker. Security tests on the binary chaotic communication scheme, Dec. 1996. Technical Report CRASP012132.
- [52] K.M. Short and A.T. Parker. Detecting and extracting messages from chaotic communication schemes. Paper delivered at the SIAM Conf. on Dynamical Systems, Snowbird, UT, May 18-22, 1997.
- [53] K.M. Short and A.T. Parker. Extracting speech and square wave signals from modulated hyperchaotic communications, Feb. 1997. Technical Report CRASP012132.
- [54] K.M. Short and A.T. Parker. Unmasking a hyperchaotic communication scheme. *Phys. Rev. E*, 58:1159–1162, 1998.
- [55] T. Stojanovski, L. Kocarev, and U. Parlitz. Driving and synchronizing by chaotic impulses. *Phys. Rev. E*, 54:2128–2131, 1996.
- [56] Steven H. Strogatz. *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. Addison-Wesley, Reading, MA, 1994.
- [57] F. Takens. Detecting strange attractors in turbulence. In *Dynamical Systems and Turbulence, Warwick 1980*, volume 898 of *Lect. Notes in Math*, page 366. Springer, Berlin, 1981.
- [58] Daniel D. Wheeler and Robert A.J. Matthews. Supercomputer investigations of a chaotic encryption algorithm. *CRYPTOLOGIA*, 15(2):140–152, 1991.
- [59] D.D. Wheeler. Problems with chaotic cryptosystems. *CRYPTOLOGIA*, 13:243–250, 1989.
- [60] C.W. Wu and L.O. Chua. A simple way to synchronize chaotic systems with applications to secure communications. *Intl. J. of Bifurcation and Chaos*, 3:1619–1627, 1993.
- [61] T. Yang, L.-B. Yang, and C.-M. Yang. Control of rossler system to periodic motions using impulsive control methods. Preprint, 1997.
- [62] T. Yang, L.-B. Yang, and C.-M. Yang. Impulsive synchronization of lorenz systems. *Phys. Lett. A*, 226:349–354, 1997.
- [63] Tao Yang, Chai Wah Wu, and Leon O. Chua. Cryptography based on chaotic systems. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 44(5):469, 1997.
- [64] Hong Zhou and Xie-Ting Ling. Problems with the chaotic inverse system encryption approach. *IEEE Transactions on Circuits and Systems—I: Fundamental Theory and Applications*, 44(3):268–271, 1997.

- [65] Hong Zhou, Xie-Ting Ling, and Jun Yu. Secure communication via one-dimensional chaotic inverse systems. In *IEEE International Symposium on Circuits and Systems*. IEEE, 1997.