T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YENİ KAOTİK SİSTEMLER İLE RASGELE SAYI ÜRETECİ TASARIMI VE ÇOKLU-ORTAM VERİLERİNİN YÜKSEK GÜVENLİKLİ ŞİFRELENMESİ

DOKTORA TEZİ

Akif AKGÜL

Enstitü Anabilim Dalı	**:**	**ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ**
Enstitü Bilim Dalı	**:**	**ELEKTRONİK**
Tez Danışmanı	**:**	**Doç. Dr. İhsan PEHLİVAN**

Mayıs 2015

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YENİ KAOTİK SİSTEMLER İLE RASGELE SAYI ÜRETECİ TASARIMI VE ÇOKLU-ORTAM VERİLERİNİN YÜKSEK GÜVENLİKLİ ŞİFRELENMESİ

DOKTORA TEZİ

Akif AKGÜL

Enstitü Anabilim Dalı : ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ

Bu tez 26 / 05 /2015 tarihinde aşağıdaki jüri tarafından oybirliği / ~~oyçokluğu~~ ile kabul edilmiştir.

Doç. Dr.
İhsan PEHLİVAN
Jüri Başkanı

Doç. Dr.
Yılmaz UYAROĞLU
Üye

Doç. Dr.
Ayhan İSTANBULLU
Üye

Doç. Dr.
Uğur YÜZGEÇ
Üye

Yrd. Doç. Dr.
Devrim AKGÜN
Üye

BEYAN

Tez içindeki tüm verilerin akademik kurallar çerçevesinde tarafımdan elde edildiğini, görsel ve yazılı tüm bilgi ve sonuçların akademik ve etik kurallara uygun şekilde sunulduğunu, kullanılan verilerde herhangi bir tahrifat yapılmadığını, başkalarının eserlerinden yararlanılması durumunda bilimsel normlara uygun olarak atıfta bulunulduğunu, tezde yer alan verilerin bu üniversite veya başka bir üniversitede herhangi bir tez çalışmasında kullanılmadığını beyan ederim.

Akif AKGÜL

26.05.2015

TEŞEKKÜR

Doktora eğitimim süresince değerli birikimlerini aktaran, tezimin başlangıcından bitimine kadar çalışmalarıma yön veren ve değerli zamanını sorunlarımın çözümüne ayıran tez danışmanım Sayın Doç. Dr. İhsan PEHLİVAN'a, maddi olarak destek sağlayan SAÜ Bilimsel Araştırma Projeleri Komisyonu Başkanlığı'na ve ayrıca emeği geçen herkese teşekkür ederim.

Maddi ve manevi olarak desteklerini esirgemeyen anne babama, tez çalışmam boyunca bana destek olan eşime, oğlum Muhammed Ali'ye ve üzerimde emeği olan herkese ayrıca teşekkürlerimi sunarım.

İÇİNDEKİLER

SİMGELER VE KISALTMALAR LİSTESİ

$\Delta^2\Psi_m^2(\text{obs})$	: m-bit örneğin beklenen frekansı
$\nu_{i_1 \cdots i_m}$	: m bitlik $i_1 \ldots i_m$ örneklerin frekansı
$\nu_{i_1 \cdots i_{m-1}}$	: m-1 bitlik $i_1 \ldots i_{m-1}$ örneklerin frekansı
$\nu_{i_1 \cdots i_{m-2}}$	: m-2 bitlik $i_1 \ldots i_{m-2}$ örneklerin frekansı
AD	: Analog Devices
AES	: Advanced Encryption Standard
C	: Kondansatör değeri
CMOS	: Complementary Metal Oxide Semiconductor
cov	: Koveryans
DES	: Data Encryption Standard
ei	: Beklenen frekans
erfc	: The Complementary Error Function
exp	: Üs bitleri değeri
FIPS	: Federal Information Processing Standard
FPAA	: Field Programmable Analog Array
FPGA	: Field Programmable Gate Array
GRSÜ	: Gerçek Rasgele Sayı Üreteçleri
IDEA	: International Data Encryption Algorithm
IEEE	: The Institute of Electrical and Electronical Engineers
IEEE-754	: IEEE Kayan noktalı sayı formatı
igamc	: Incomplete Complementary Gamma Function
j	: Kesir bitlerinin sayısı
J	: i. L-bit bloğun onluk sayı sistemindeki değeri
j	: Kesir bitlerinin sayısı
K	: Bağımsızlık katsayısı

k1 : RK algoritmasında ilk hesaplanan değişken

k2 : RK algoritmasında ikinci hesaplanan değişken

k3 : RK algoritmasında üçüncü hesaplanan değişken

k4 : RK algoritmasında dördüncü hesaplanan değişken

Kbit : Kilobit

L : Üniversal testinde her bir bloğun uzunluğu

M : Bit dizisinde belirli sayıdaki bitlerinden oluşan blok

m : Örtüşen şablon eşleştirme testinde özel blokların bit sayısı

MATLAB : Matrix laboratory

Mbit : Megabit

MD5 : Message Digest Algorithm

ms : Mili Saniye

n : Bit dizisinin uzunluğu

N_0 : T değerinden daha küçük beklenen değeri

NIST : National Institute of Standards and Technology

o_i : Gözlemlenen frekans

P-değeri : NIST-800-22 testinde rasgelelik ölçütü

Q : İkili matris derece testinde sütun sayısı

R : Direnç değeri

RC5 : Rivest Cipher 5

RC6 : Rivest Cipher 6

RK4 : Dördüncü dereceden Runge-Kutta algoritması

RK5 : Beşinci dereceden Runge-Kutta algoritması

RL : Direnç ve Bobinden oluşan Devre

RLC : Direnç, Bobin ve Kondansatörden oluşan Devre

RSA : Ronald, Shamir, Adleman

RSÜ : Rasgele Sayı Üreteçleri

SEA : Scalable Encryption Algorithm

sign : İşaret biti

S_n : Normalizasyon işleminden elde edilen değer

sn : Saniye

S_{obs} : Gözlemlenen değer

SRSÜ : Sözde Rasgele Sayı Üreteçleri

TEA : Tiny Encryption Standard

T_i : Dağılımın rasgele değişkeni

T_j : Muhtemel L-bit değerleri

V : Gerilim

v : Onluk sayı değeri

V(obs) : Bit osilasyon sayısı

var : Varyans

VHDL : VHSICircuit Hardware Description Language

Vi : En uzun 1 dizisinin akış frekansı

Wj : Özel B şablonunun frekansı

XOR : Exclusive Or (Özel Veya)

XTEA : eXtended Tiny Encryption Standard

y_λ : Algoritma ilk değeri

$y_{\lambda+1}$: Algoritma sonraki değeri

γ : Sistem parametresi

Δh : Algoritma adım miktarı

ε : Bit dizisi

ε' : Artırım dizisi

εi : Bit dizisinin i. elemanı

λ : Öz değerler

λ_σ : Algoritma parametreleri

μ : Beklenen değer

μs : Mikro Saniye

ξ : Rasgele yürüyüşlerde ziyaret edilen durumların toplam sayısı

ξ_σ : Algoritma parametreleri

π : Bit dizisindeki 1 değerlerinin sayısı

σ^2 : Varyans

τ : Test için gerekli parametre şartı

$\varphi^{(m)}$: Blokların ampirik dağılım frekansı

$\Phi(z)$: Olasılık yoğunluk fonksiyonu

ŞEKİLLER LİSTESİ

TABLOLAR LİSTESİ

ÖZET

Anahtar kelimeler: Kaos, Kriptoloji, Kaotik Sistemler, Kaos Tabanlı Şifreleme, Rasgele Sayı Üreteci, İstatistiksel Rasgelelik Testleri, NIST Rasgelelik Testi, Çoklu-Ortam Veri Güvenliği, Güvenlik Analizleri

Bu tez çalışmasında, literatürdeki şifreleme algoritmalarından daha hızlı ve güvenli olan kaos tabanlı bir şifreleme algortimasının tasarımı ile çoklu ortam verilerinin şifrelenmesi amaçlanmıştır.

Tezin ilk aşamasında, literatürde olmayan yeni kaotik sistemlerin tasarım ve analizleri için; öngörülen sistemlerin denge noktaları bulunmuş, zaman serileri ve faz portreleri elde edilmiş, lyapunov üstelleri hesaplatılmış, parametre değişimine göre lyapunov üstellleri spektrumları ve çatallaşma diyagramları çizdirilmiştir. Ardından tasarlanan yeni kaotik sistemlere ait elektronik devreler modellenerek, ORCAD-PSpice simülasyonları ve deneysel olarak devre uygulamaları gerçekleştirilmiştir. Yapılan tüm dinamik analizler, devre simülasyonları, devre gerçeklemelerine ait çıkışlar ve karşılaştırmalar ile sistemlerin kaotik yapıları ispatlanmış, daha iyi anlaşılmış ve denklemlere son halleri verilmiştir. İkinci aşamada, öncelikle sürekli zamanlı kaotik sistemler Runge Kutta-4 yöntemi ile ayrıklaştırılmış ve elde edilen sayılar ikili sayı formatına çevrilerek, yeni ve özgün RSÜ tasarımları yapılmıştır. Tasarlanan RSÜ'ler uluslararası en üst standart olan NIST-800-22 ve FIPS-140-1 istatistiksel testlerinden başarıyla geçirilerek, yeni bir kaos tabanlı şifreleme algoritmasının geliştirilmesinde temel alınmıştır.

Üçüncü aşamada; yeni kaotik RSÜ tabanlı özgün bir şifreleme algoritması geliştirilerek sinyal, metin, ses, resim ve video gibi farklı çoklu-ortam verileri ayrı ayrı şifrelenmiştir. Şifrelenen multimedya verilerinin korelasyon, histogram gibi güvenlik analizleri yapılarak başarımları ölçülmüştür. Son aşamada ise; AVR Studio 5.1 programı ile yeni kaotik RSÜ tabanlı özgün şifreleme yöntemi ile güncel literatürdeki bazı şifreleme yöntemleri, bellek ve hız bakımından karşılaştırılarak, gerçek ortam uygulamaları için performans değerlendirmeleri sunulmuştur.

Sonuç olarak; geliştirilen kaos tabanlı şifreleme yönteminin yeni ve özgün özellikleri şunlardır: dinamik yapısı karmaşık ve rasgeleliği yüksek yeni kaotik sistemler içermektedir, NIST-800-22 ve FIPS-140-1 rasgelelik testleri ile daha üstün rasgeleliğe sahip olan özgün RSÜ tabanlı bir yapıdadır, daha kolay ve hızlı işlenebilen bir algoritma yapısına sahip olduğundan diğer şifreleme algoritmalarına (AES, Skipjack, RC5, vb.) göre genel olarak daha hızlı, bellek olarak çok daha az yer kaplamakta ve en önemlisi de tüm çoklu ortam verilerini (ses, görüntü, video, metin, vb.) daha güvenli olarak şifreleyebilmektedir.

DESIGN OF RANDOM NUMBER GENERATORS WITH NOVEL CHAOTIC SYSTEMS AND HIGH SECURE MULTIMEDIA DATA ENCRYPTION

SUMMARY

Keywords: Chaos, Cryptology, Chaotic Systems, Chaos Based Encryption, Random Number Generator, Statistical Randomness Tests, Multimedia Data Security

In this thesis, the encryption of multimedia data with the design of a new chaos based encryption algorithm, which is much more secure and faster than the encryption algorithm in the literature, is aimed.

At the first stage of the thesis, to design and analyze the novel chaotic system, the equilibrium points are found, time series and phase portraits are acquired, lypapunov exponents of the systems are calculated, the spectrums of lypanov exponents with respect to the system parameters and bifurcation diagram of the systems are plotted. Then, the electronic circuit model of the designed chaotic systems are simulated in ORCAD-PSpice and the circuits are realized at the laboratory. At the second stage, the continuous time chaotic systems are discretized with Runge Kutta-4 numerical algorithm. The new novel RNGs are designed by converting the numbers obtained after the discretization process into binary. The RNGs pass the NIST-800-22 and FIPS-140-1 statistical tests, which are the highest international standards, successfully and then these RNGs are used as a base for developing a new chaos based encryption algorithm. At the third stage; by developing a new novel chaotic RNG based encryption algorithm, multimedia data like signal, text, audio, image and video is encrypted. Security analyses like correlation and histogram analysis, of the encrypted data are made to evaluate the performance of the encryption. At the last stage, the performance analyses of the new chaotic RNG based encryption algorithm and some encryption algorithms in the recent literature with are made AVR Studio 5.1 program and comparison with respect to speed and memory is done to present the evaluation of the performance of the new encryption algorithm for real time application.

In conclusion; the novelty of the developed chaos based encryption method is: it consist chaotic systems with complex dynamic structure and high randomness, The designed original RNGs are more random structure with successfull NIST-800-22 and FIPS-140-1 statistical tests, due to the algorithm has a structure that is easier and faster to process, the new chaos based algorithm is faster and uses less memory than other encryption algorithm (AES, Skipjack, RC5) and more importantly for all type of the multimedia data(audio, image, video, text) the new chaos based encryption algorithm provides more reliable encryption than other algorithms.

BÖLÜM 1. GİRİŞ

Kaos bilimi, dinamik sistemlerde bilinen en karmaşık kararlı hal davranışıdır ve doğrusal olmayan olayları açıklamaya yarayan bir bilim dalıdır. Diğer bir ifadeyle kısaca kaos, düzensizliğin düzenidir. Kaosun ve kaotik işaretlerin başlıca önemli özellikleri; zaman boyutunda düzensizliği, başlangıç şartlarına hassas bağımlılığı, sınırsız sayıda değişik periyodik salınımlar içermesi, gürültü benzeri geniş güç spektrumuna sahip olması, limit kümesinin parçalı (fraktal) boyutlu olması, genliği ve frekansı tespit edilemeyen, ancak sınırlı bir alanda değişen işaretler içermesidir. Doğrusal olmayan sistem teorilerindeki ilerleme, yeni deneysel teknikler, pahalı ve işlem gücü yüksek bilgisayarların ucuzlayıp yaygınlaşması, karmaşık ve doğrusal olmayan davranışları daha iyi analiz etmeye ve anlamaya sebep olmuş ve sonuç olarak kaos bilimi gelişmiştir. Son yıllarda kaos ve karmaşıklıkla ilgili gözlemlere paralel olarak, bu olayın mekanizmasının anlaşılması, kaotik davranışın nitelendirilmesi, özelliklerinin belirlenmesi, deneysel verilerin ölçülmesi ve analizinin yapılması ile ilgili araştırmalarda çok hızlı gelişmeler kaydedilmiştir.

Kaotik sistemler, son yıllarda bilim ve mühendislik çevrelerinde geniş bir alanda çalışma konusu olmakta, [1-7] literatüre yeni kaotik sistemler sunulmakta, ve kullanım alanlarında artış sağlanmaktadır. Teknolojinin gelişmesiyle birlikte kaos bilimi, haberleşme, görüntü işleme, bulanık mantık, kontrol, fizik, optimizasyon, ve mekatronik gibi pek çok alanın yanında özellikle şifreleme çalışmalarında da kullanılmaya başlamıştır. Bunun en önemli nedeni ise; kaotik işaretlerin geniş bantlı, gürültü benzeri, önceden tahmin edilmesi zor ve periyodik olmayan özelliklere[6] sahip olması ve şifrelenen veriler üzerindeki karıştırma ve yayılmayı önemli ölçüde arttırmasıdır. Şifrelenmiş verilerin karmaşıklık ve hassasiyet düzeyinin yüksekliği ile, şifreleme algoritmalarının yapısı gibi etkenler şifrelemede en önemli unsurlardır. Standart şifreleme algoritmalarına alternatif olarak, kaos tabanlı şifreleme algoritmalarıyla yapılan çalışmalar son zamanlarda artış göstermektedir.

Kaotik sistemler, ayrık zamanlı veya sürekli zamanlı olarak sistemdeki denklem sayısına göre sınıflandırılabilir. Sistem boyutu arttıkça, denklemlerdeki parametreler ve başlangıç değerlerinin sayısı da artabilmektedir. Şifreleme çalışmalarında ne kadar fazla bilinmeyen olursa, üçüncü kişiler tarafından şifreli verilerin çözülmesi o derece zor olacaktır. Ayrıca konu veri güvenliği ve gizli haberleşme olunca, bilinen kaotik sistemler yerine, yeni karmaşık sistemler kullanmak çok daha önemli hale gelmektedir.

Kaos tabanlı yöntemlerle şifrelenmiş bir veriyi çözebilmek için, kullanılan kaotik sistemi, kaotik sistemdeki tüm denklemler, parametre ve başlangıç değerlerini bilmek gerekmektedir. Şifre çözme esnasındaki her hangi bir hata durumunda şifreli verinin çözümü mümkün değildir. Bu hassasiyetlerden dolayı şifreleme biliminde kaotik dinamikleri kullanmak tercih nedenlerinden birisi olmuştu

Geliştiren yeni şifreleme yöntemlerinde, matematiksel ve mantıksal ifadeler ile algoritmalardaki karmaşıklığı arttırmak bazı durumlarda dezavantajlı hale gelebilir. Çünkü, genel olarak her türlü bilgi iletiminde, güvenli bir haberleşmenin sağlanabilmesi için minimum bazı gereksinimler bulunmaktadır. Bunlar; gizlilik, bütünlük, doğrulama gibi temel etkenlerdir. Mesajın sadece yetkili kişiler tarafından görülebilmesi(gizlilik), mesajın göndericiden başka hiçkimse tarafından değiştirilememesi(bütünlük), mesaja sadece yetkili kişilerin erişebilmesi ve mesajın bozulmamış olması (doğrulama) güvenli bir haberleşmede olması gereken temel özelliklerdir[8]. Şifreleme sistemlerini ve algoritmalarını çok karmaşık hale getirmek, bahsedilen temel şartların sağlanamamasına neden olabilmektedir.

Şifrelemenin en temel unsurlarından birisi kullanılan anahtarlardır. Bu anahtarların üretilmesi ve saklanması en önemli problemlerden birisidir. Kaotik sistemler çok karmaşık dinamik özellikler gösterdiklerinden dolayı, rasgele anahtar üretiminde ön plana çıkmaktadırlar. Kriptolojik uygulamalarda kullanılan Rasgele Sayı Üreteçlerinin (RSÜ) ürettiği sayıların rasgeleliği, şifreleme uygulamaların güvenliğini doğrudan etkilediklerinden, kriptolojik uygulamalar için kritik öneme sahiptiler. Son zamanlarda kaotik sistemler ile RSÜ tasarımında artış meydana gelmiştir. Üretilen rasgele sayılar

FIPS-140-1 ve NIST-800-22 gibi uluslararası kabul görmüş testlerden geçtikten sonra şifreleme uygulamalarında kullanılabilmektedir.

Sürekli zamanlı kaotik sistemler, genellikle Euler, Heun, RK4, RK5 gibi numerik analiz algoritmalarıyla ayrık zamanlı hale getirilerek birçok farklı uygulama alanında kullanılabilmektedir. Ayrıklaştırma işlemi sonucu elde edilen veriler, bilgisayar ortamında ve birçok gerçek zamanlı sistemde kullanılabilmektedir. Sürekli zamanlı kaotik sistemlerden ayrıklaştırılarak elde edilen veriler farklı şifreleme yöntemleri yardımıyla şifreleme çalışmalarında da kullanılabilmektedir. Şifreleme uygulamalarında verilerin sadece şifrelenmesi yeterli değildir. Şifrelenmiş verilerin güvenilirliğin olabildiğince üst seviyede olması gerekmektedir. Güvenilirliğin üst seviyede olduğunu göstermek için veri türüne göre bazı güvenlik analizlerinin yapılması gerekmektedir. Literatürde yaygın olarak kullanılan bilgi entropi, korelasyon, kaos etkisi, hız etkisi, differantial attack, histogram analiz gibi güvenlik analizleri bulunmaktadır[9-14]. Bu analizlerin sonuçları ne kadar iyi çıkarsa, veriler o derece güvenlidir ve şifreli verileri çözmek de o derece zordur. Güvenlik düzeyi yüksek olan bir şifreleme, maliyet, hız, bellek vb. faktörler konusunda sınırlama yoksa tercih edilebilir bir yöntemdir. Güçlü bir şifreleme yöntemini gerçek ortam uygulamalarında kullanmak, verilerin güvenliği ve yaygın kullanım alanı için önemli avantajlar sağlamış olacaktır.

Literatürde kaotik sistemlerin kullanımları, kaos tabanlı yöntemlerle rasgele sayı üretimi ve şifreleme çalışmalarına yönelik, çeşitli ortamlarda (network, stenegrofi, FPGA vb.) ve farklı multimedya verileriyle gerçekleştirilmiş uygulamalar bulunmaktadır. Kaotik sistemler ve şifreleme ile ilgili aşağıdaki bilimsel çalışmalar örnek olarak verilebilirler:

1963 yılında Edward Lorenz'in öncülüğünde gelişmeye başlayan [15] "Kaos Bilimi", Rössler [16], Chua [17] gibi bilim adamları ile hızlı ilerlemeler kaydederek, günümüzde de birçok alanda gelişmesine devam etmektedir. Kaos olayını ilk basit elektronik uygulama ile açıklayan model devreyi Chua gerçekleştirmiştir [18]. Geliştirilen bu devre, kaos üreteci olarak birçok yerde kullanılmıştır. Daha sonraları basit RLC, RC devreleri [19- 23], osilatörler [24, 25], güç devreleri [26-28], sayısal

filtreler [29-32] ve kapasitör devreleri gibi kaotik davranış gösteren birçok elektronik devre geliştirilmiştir.

Son yıllarda birçok alanda kullanılmak üzere ilginç özellikli ve pratik uygulamalarda kullanım potansiyeline sahip yeni ve farklı kaotik ve hiperkaotik (3 boyutdan fazla olan sistemler) sistemler literature sunulmuştur [33-38]. Bazı sistemler var olan sistemlerde değişiklikler yapılarak, bazıları ise tamamen yeni sistemler olarak geliştirilmiştir. Chen ve Ueta, Lorenz sistemini referans alarak, Chen sistemini geliştirmişlerdir [39]. Lü ve Chen ise, Lorenz ve Chen sisteminden yeni bir kaotik sistem tasarlamışlardır [1]. Sprott, kapsamlı araştırmalar sonucu 19 farklı yeni kaotik sistem bulmuştur [40, 41, 3]. Geliştirilen ve üzerinde bilimsel çalışmalar yapılan kaotik sistemlere Rabinovich [42], Rikitake [43], Burke-Shaw [44] ve Sundarapandian-Pehlivan [33] sistemleri de örnek verilebilir. Uygulama alanlarına göre, farklı dinamik özeliklerdeki kaotik sistemleri kullanmak avantajlı olabilmektedir. Örneğin; şifreleme uygulamaları için, son zamanlarda keşfedilen, gizli çekicili kaotik istemler olarak da adlandırılan, denge noktasız kaotik sistemler bulunmaktadır [45- 48, 7]. Bu sistemlerin analizleri Shilnikov metodu gibi yöntemlerle yapılamadığından, karmaşıklık gerektiren uygulamalarda tercih edilebilmektedir. Bilinen kaotik analiz yöntemleri ile analizleri yapılamayan bir sistem, dinamik yapısı iyi anlaşılamayacağından dolayı, şifrelemede kullanıldığında şifreli verinin çözülmesi de çok zor olacaktır.

İlk yıllarda kaotik maskeleme, kaotik modülasyon ve kaotik anahtarlama gibi şifreleme işlemleri analog olarak gerçekleştirilmiştir. Fakat günümüzde kaos tabanlı şifreleme işlemleri genellikle dijital tabanlı olarak gerçekleştirilmektedir. Kaos tabanlı dijital şifrelemenin, analog tabanlı kaotik şifrelemeye göre temel avantajları arasında; şifre çözme işlemi için şifreleme algoritmasının tersini almanın yeterli olması, şifreleme algoritmalarının kolaylıkla değiştirilip güncellenebilmesi, analog şifrelemede sorun olan senkranizasyon işlemlerine gerek duyulmaması, gürültü, sıcaklık ve nem gibi bozucu etkilerden etkilenmemesi sayılabilir. Kaotik sistemlerle ilgili şifreleme çalışmaları pek çok farklı platformda gerçekleştirilmiştir. Pehlivan ve arkadaşları sinyal gizleme ile ilgili yaptıkları bazı çalışmalarında, Şekil 1.1'deki gibi, kaotik sinyallere bilgi sinyalini ekleme ile gerçekleştirilen, maskeleme (masking)

yöntemini kullanmışlardır [34-36, 2]. Merah ve arkadaşları, bilgi güvenliğini sağlamak amacıyla, VHDL (Very High Speed Integrated Circuit Hardware Description Language) dili ile Lorenz kaotik sistemini FPGA üzerinde modellemişlerdir [49]. Ses, resim, video gibi multimedya verilerini kaos tabanlı şifrelemek için genellikle bilgisayar ortamı kullanılmıştır. Bu çalışmalarda, multimedya verisinin türüne, göre pek çok farklı yöntem kullanılmıştır. Sakthidasan ve Santhosh; orijinal veri ile kaotik sistemden gelen veriyi karıştırarak, kaos ile şifreleme işlemi gerçekleştirmişlerdir [50].

Şekil 1.1. Kaos tabanlı bir şifrelemenin blok diyagramı

Oğraş ve Türk; doğrusal olmayan bir denklem kullanarak, resim verisi üzerinde şifreleme işlemi yapmışlardır [51]. Bu tür şifrelenmiş verileri çözmek için, ayrıca doğrusal olmayan fonksiyon ve onun tüm parametrelerinin de bilinmesi gerekmektedir. Fındık; kaotik sistem tabanlı ve kaotik olmayan şifreleme algoritmalarını karıştırarak metin şifreleme gerçekleştirmiştir [52]. Gerçek ortam uygulamalarında, resim, video, ses gibi verilerin boyutları büyük olduğu için, bu tür hibrid yöntemle şifreleme işlemi gerçekleştirmek, hız açısından dezavantajlıdır. Yardım ve Afacan; kaotik sinyal verilerine gecikme ve anahtarlama işlemleri uygulayarak şifreleme ve şifre çözme işlemlerine yönelik zamanlama ile ilgili çalışmalar yapmışlardır [53]. Bu türden bir yöntemle şifrelenmiş verileri çözmek için, hangi verinin, ne zaman, hangi sırada şifrelenmiş olduğunu bilmek gerekmektedir. Aksi halde şifrelenmiş veriler çözülemeyecektir. Sohby ve Shehata ise; Lorenz sistemine veriyi ekleyerek kaotik şifreleme yapmışlardır [54].

Oğraş ve arkadaşları; iki kaos üreteci ile anahtarlama yaparak, kaos tabanlı şifreleme işlemleri gerçekleştirmişlerdir [55]. Abdulkareem ve Abduljaleel; ses verilerini

şifrelemek için, tek boyutlu kaos üreteci ile kaotik olmayan bir şifreleme yöntemi olan Blowfish algoritmasını kullanarak, yeni bir şifreleme yöntemi geliştirmişlerdir [56]. Zhang ve Min; haberleşme için, simetrik olmayan sayısal bir şifreleme algoritması geliştirerek, geliştirdikleri sistemin güvenlik analizlerini yapmışlardır [57]. Prabu ve arkadaşları; tek boyutlu ayrık kaotik Logistic Map sistemi ile, ses şifreleme çalışması yaparak, gerçek zamanlı bir uygulama gerçekleştirmişlerdir [58]. Bu yöntemlerde birçok farklı multimedya verisi, kaotik sistemler ile şifrelenmiştir. Literatürde, kaos tabanlı şifreleme işlemleri, genellikle resim verisi üzerinden yapılmıştır [59-63, 10, 13]. Sinyal, metin, ses ve video veri türleri ile ilgili çalışmalar ise oldukça azdır [64-70, 6].

Bazı çalışmalarda ise şifreleme çalışmaları kaotik sistemlerden rasgele sayılarak üretilerek, bu sayıların anahtarlar olarak kullanılmasıyla yapılmıştır. Üretilen sayıların rasgeleliği, şifreleme uygulamalarındaki güvenilirliği doğrudan etkilemektedir. Literatürde rasgele sayıların üretilmesine yönelik kaos tabanlı olan ve olmayan birçok çalışma bulunmaktadır.

Wieczorek ve arkadaşları, FPGA ile çift kararlı flip-flop kullanarak 50 MHz çalışma frekanslı ve 5 Mbit/s bit üretim hızında RSÜ tasarımı yaparak istatistiksel testlere tabi tutarak başarılı sonuçlar elde etmişlerdir [71]. Fischer ve arkadaşları 1 Mbit/s bit üretim hızında, PLL tabanlı osilatörü FPGA kullanarak gerçekleştirmişler ve NIST testlerinden başarılı sonuçlar elde etmişlerdir [72]. István ve arkadaşları ise yine FPGA tabanlı, 50 MHz çalışma frekanslı klasik jitter osilatör yöntemi ile rasgele sayı üretimi gerçekleştirerek, NIST testlerinden başarılı sonuçlar elde etmişlerdir [73].

Kaos tabanlı gerçekleştirilen rasgele sayı üreteçlerinin tasarımında ise, Çiçek ve arkadaşları, CMOS teknolojisi ile ayrık zamanlı tek boyutlu harita kullanarak RSÜ tasarımı gerçekleştirmişler ve NIST testlerinin 11'inden başarılı sonuçlar elde etmişlerdir [74]. Yine Çiçek ve arkadaşları, FPAA (Field Programmable Analog Array) donanımı ile, tek boyutlu ayrık iki kaotik harita kullanarak 16 MHz çalışma frekanslı ve 1.5 Mbit/s bit üretim hızında RSÜ tasarımı yapmışlar ve NIST testlerinin hepsinden başarılı sonuçlar elde etmişlerdir [75]. Pareschi ve arkadaşları, CMOS

teknolojisini kullanarak 80 Mbit/s bit üretim hızında kaos tabanlı RSÜ tasarımı gerçekleştirmişlerdir [76].

Literatürde sunulan çalışmalardan görüleceği üzere, kaotik sistemler üzerine yoğun bir biçimde çalışmalar sürdürülmekte ve multimedia verileri üzerine farklı yöntemlerle şifreleme çalışmaları gerçekleştirilmektedir. Kaotik sistemler pek çok farklı alanlarda kendine uygulama alanı bulabilmektedir. Özellikle son zamanlarda, şifreleme çalışmalarındaki kullanımlara yönelik artış literatür çalışmalarında görülmektedir. Kaos tabanlı tasarlanan rasgele sayılarla şifreleme çalışmaları, kaotik sistemlerin özelliklerinden dolayı tercih edilmektedir.

Kaotik sistemlerle yapılan şifreleme çalışmaları ve rasgele sayı üreteçleri genellikle literatürde var olan kaotik sistemler kullanılarak yapıldığı için, var olan bir sistemi şifreleme çalışmalarında kullanmak dezavantaj olarak karşımıza çıkmaktadır. Literatürde, şifreleme çalışmaları için tasarlanan bazı RSÜ, FIPS-140-1 ve NIST-800-22 gibi uluslararası alanda kabul görmüş testlere tabi tutulmadığından ve yapılan birçok şifreleme çalışmasının güvenlik analizleri gerçekleştirilmediği için şifrelenen verilerin güvenilirlikleri sorun olabilmektedir. Ayrıca kaos tabanlı olan ve kaos tabanlı olmayan şifreleme yöntemlerinin gerçeklemeleri, birçok gerçek ortam uygulamalarında hız ve bellek açısından sorun olduğu için kullanılamamakta ve sınırlı belli alanlarda şifreleme çalışmaları yapılabilmektedir.

Yukarıda özetlenen çalışmalardan da gözlemlendiği gibi rasgele sayı üreteçleri ile kaos tabanlı veya kaos tabanlı olmayan yöntemlerle şifreleme işlemleri için problem olan beş önemli sorun ortaya çıkmaktadır:

1. Kaotik olmayan yöntemlerle gerçekleştirilen çalışmalarda karıştırma ve yayılma işlemi önemli problemdir. Kaotik sistemler bu özellikleri iyi sağladıklarından dolayı kaos tabanlı şifreleme işlemlerinde ön plana çıkmaktadır.

2. Kaos tabanlı şifrelemelerde genellikle literatürde var olan sistemler kullanılmaktadır. Yeni kaotik sistemler tasarlayarak şifreleme işlemleri

gerçekleştirmek bu tarz çalışmalara yenilik getirmekle beraber güvenlik düzeyini de arttırmış olacaktır. Çünkü şifreli verileri çözmek isteyen kişilerin öncelikle kaotik sistemi (denklemler, tüm parameter ve başlangıç değerleri) bulmaları gerekmektedir.

3. RSÜ, şifreleme çalışmalarında anahtarlar olarak kullanılamaktadır. RSÜ'in başarımı ise kabul görmüş bazı testlerle gerçekleştirilmektedir. Şifreleme işlemleri için bu testlerden başarıyla geçmiş RSÜ'lere ihtiyaç duyulmaktadır.

4. Bellek olarak az yer kaplayan ve süre olarak hızlı gerçeklenebilen güçlü şifreleme yöntemlerine ihtiyaç duyulmaktadır. Bu sayede FPGA, mikrodenetleyici gibi gerçek ortam uygulamalarındaki kullanım artmış olacaktır.

5. Bir yöntemin farklı multimedia verilerine (resim, metin, ses, vb.) uygulamasında sorunlar olabilmektedir. Tüm multimedia verileri için tek bir yöntem tasarım ihtiya cı bulunmaktadır. Şifreleme yöntemi ne kadar sade ve basit olursa, kullanımıda o kadar kolay olacaktır.

1.1. Tezin Amacı, Yapılacak İş

Sunulan tezin genel amacı; yeni ve farklı 3 boyutlu, sürekli zamanlı kaotik sistemleri kullanarak, FIPS-140-1 ve NIST-800-22 gibi uluslararası alanda kabul görmüş testlerden başarıyla geçmiş rasgele sayı üreteci tasar; metin, resim, ses, video gibi multimedia verilerini güvenli olarak şifrelemektir.

1.2. Tezde İzlenecek Yol

Bu tez çalışması, yeni ve farklı, 3 boyutlu, sürekli zamanlı kaotik sistemler ile, tüm multimedia veri çeşitlerinin, FIPS-140-1 ve NIST-800-22 testlerinden başarıyla geçmiş rasgele sayı üreteçlerinin yardımıyla, kaos tabanlı olarak şifrelenmesi amacı ile sonuç ve öneriler bölümüyle birlikte yedi bölüme ayrılmıştır. Bu amaçla, ikinci

bölümde; şifreleme bilimi, şifreleme teknikleri, istatistiksel rasgelelilik testleri, güvenlik analizleri ve gerçek ortamdaki hız ve bellek analizi için kullanılan AVR Studio 5.1 programından bahsedilerek, temel bilgiler verilmiştir.

Üçüncü ve dördüncü bölümde; ayrık ve sürekli zamanlı kaotik sistemler tanıtılarak, kaotik sistemlerin analizlerinin nasıl yapılacağı, hangi yöntemlerin kullanılabileceği hakkında bilgiler verilmiş, 2 adet, yeni geliştirilen; üç boyutlu, sürekli zamanlı kaotik sistemin analizleri yapılmış, elektronik devre uygulamaları gerçekleştirilmiştir. Tasarlanan yeni kaotik sistemlerin dinamik davranışlarını belirlemek amacıyla, Matlab programı kullanılarak, kaotik sistemlerin zaman serileri, faz portreleri, çatallaşma diyagramları, boyut analizleri, denge noktaları ve Lyapunov spektrumu analizleri yapılmıştır. Bu sayede tasarlanan sistemlerin kaotik olup olmadıkları kanıtlanmış ve dinamik özellikleri ortaya çıkarılmıştır. Daha sonra, tasarlanan kaotik sistemler, OrCAD-PSpice elektronik devre simüslasyon programında, elektronik elemanlar ile modellenerek, benzetime tabi tutulup, faz portre çıkışları, nümerik analiz sonuçlarındaki faz portre çıkışları ile karşılaştırılmıştır. Ayrca elektronik devre modellemeleri gerçek ortamda gerçekleştirilmiş ve osilaskop çıktıları, nümerik analiz ve elektronik simülasyon sonuçları ile karşılaştırılmıştır.

Beşinci bölümde; kaos tabanlı olarak, multimedia veri çeşitlerine ait şifrelemelerin gerçekleştirilebilmesi için tasarlanan, yeni 3 boyutlu sürekli zamanlı sistemler için gerekli olan ayrıklaştırma algoritmaları ve bu tezde hangi algoritmanın kullanıldığı hakkında temel bilgiler verilmiştir. Ayrıca ayrıklaştırılan kaotik sistemler ile RSÜ tasarım aşamaları anlatılmış ve elde edilen RSÜ'lerin istatistiksel testleri yapılarak, bu rasgele sayıların şifreleme çalışmaları için güvenli bir şekilde kullanılabilecekleri gösterilmiştir.

Altıncı bölümde, testlerden başarıyla geçmiş rasgele sayılar ile, multimedya veri çeşitlerini (sinyal, metin, ses, resim ve video) şifreleme uygulamaları, basit mantıksal operatörler yardımıyla ayrı ayrı olarak Matlab programında gerçekleştirilmiştir. Bu uygulamalar ile, tezde geliştirilen yöntemin, tüm multimedia veri çeşitleri üzerinde uygulanabilir olduğu gösterilmiştir. Şifrelenen verilerin uygulamaları ile birlikte güvenlik analizleri de verilmiştir. Ayrıca güvenlik önlemlerini arttırmak için kaotik

sistemdeki başlangıç değerleri ve parametrelerine yönelik şifreleme işlemleri de simetrik ve asimetrik olarak gerçekleştirilmiştir. Son olarak ise, gerçekleştirilen yöntemin gerçek ortam uygulamalarında kullanılabilirliğini göstermek için, AVR Studio 5.1 programı ile bellek ve hız değerlendirmesi yapılmış ve diğer şifreleme yöntemleri ile karşılaştırılarak avantajları sunulmuştur.

Son bölümde ise; tez çalışmasında gerçekleştirilen kaos tabanlı rasgele sayı üreteçleri ile yapılan şifreleme işlemlerinin sonuçlarından bahsedilerek, ileride yapılabilecek çalışmalar hakkında öneriler sunulup, değerlendirmeler yapılmıştır.

BÖLÜM 2. TEMEL KAVRAMLAR

İnsanlık tarihinde güvenlik her zaman önemli unsurlar arasında yer almıştır. Eskiden insanlar, önemli bilgilerin güvenliğini sağlamak için duvarlar örmüş, hendekler kazmış, giriş çıkış kontolleri için nöbetçileri kullanmışlardır. Her ne kadar tedbir alınmış olsada güvenlik önlemlerinin zayıf yönleri olmuştur. Bu zayıf yönleri önlemek için ise farkı yöntemler geliştirilmiştir. Taşlara kazınarak, derilere ve kağıtlara yazılarak saklanan önemli bilgiler teknolojinin gelişmesiyle birlikte dijital ortamlarda saklanmaya başlamıştır. Dijital ortamda veri kullanımı arttıkça, önemli bilgilerin korunması amacıyla bilgisayar ve ağ güvenliği gibi yeni kavramlar ortaya çıkmıştır. Yeterli güvenlik önlemleri alınmazsa istenmeyen kişiler gizli verilere erişebilmektedir. Özellikle bu gizli veriler ağ üzerinde kullanıldığında tehditler ve güvenlik açıkları artmakta ve dolayısıyla başka önlemlerin alınması gerekmektedir. Bu açıklar önlenemezse gizli veriler ifşa edilip, değiştirilerek tehditler oluşturabilmektedir [77].

Tehditleri önlemek için kullanılan yöntemlerden birisi şifrelemedir. Şifreleme, özel bir bilginin herhangi bir yöntemle değiştirilmesi veya gizlenmesi olarak tanımlanabilmektedir. Mesajı şifrelemek için genellikle stenografi ve kriptografi olarak adlandırılan iki farklı yöntem kullanılmaktadır. Stenografi bir verinin başka bir veri içerisine anlaşılamayacak şekilde şifrelenerek saklanması, kriptografi ise belli yöntemlerle mesajın anlaşılamaz hale getirilerek şifrelenmesidir. Stenografide gizlenmek istenen veri başka bir veri içerisinde (taşıyıcı veri) örneğin resim içerisinde saklandığı için, gizlenmiş verinin farkedilememe olasılığı önemli bir avantajdır. Fakat gizlenmek istenen veri çok fazla işleme tabi tutulursa taşıyıcı veride bozukluk olma ihtimali önemli bir problemdir. Kriptografide gizlenen veri açık olduğu için şifrelenmiş verinin farkedilmesi kolaydır. Veri üzerinde çok fazla işlem yapılması farkedilmeme sıkıntısı olmadığından dolayı sorun değildir. Çok karmaşık şifreleme yöntemleri, algoritmaları kullanmak mümkündür.

Şifreleme işlemleri için birçok farklı algoritma geliştirilmiştir. Kullanım alanlarına göre geliştirilen algoritmalar kullanılabilir veya uygun bir yeni algoritma tasarımı yapılabilir. Şifreleme tasarımında dikkat edilmesi gereken en önemli nokta şifrelenen verinin tekrardan orjinal haline dönüştürülebilmesidir. Şifreleme algoritmaları genellikle orjinal veri, şifrelenmiş veri, çözülmüş veri ve algoritmalarda kullanılan anahtarlardan oluşmaktadır. Şifreleme algoritmaları simetrik ve asimetrik algoritmalar iki başlık altında toplanabilir. Simetrik algoritma yönteminde ortak bir anahtar kullanılırken, asimetrik algoritmada şifreleme için aynı, şifre çözme için farklı anahtarlar kullanılmaktadır. Ayrıca simetrik algoritma yöntemi kendi içerisinde şifrelenecek veri türüne göre blok şifreleme (bloklar halinde) ve akış (bit veya bayt olarak) şifreleme olarak ikiye ayrılır. Şifreleme uygulamalarında blok şifreleme algoritmaları yaygın bir şekilde kullanılmaktadırlar. Blok şifreleme algoritmalarında S kutuları, döngü sayısı, şifrelenecek veriye matematiksel ve mantıksal işlemler uygulanması, blok uzunluğu, anahtarın uzunluğu ve anahtarın rastlantısal yani iyi dağıtılmış olması büyük önem taşımaktadır. Ayrıca kullanılacak anahtarın rastlantısal yani iyi dağıtılmış olması da gerekir [78]. Günümüzde bu işlem için kaos tabanlı yöntemler tercih edilir olmuştur. Anahtar üretimini sağlamak için kaotik bir sistem seçilerek şifreleme işlemlerinde yaygın olarak kullanılmaya başlamıştır. Geliştirilen şifreleme algoritmaları yapılan saldırılara karşı olabildiğince dayanıklı olmalıdır. Bu yüzden dolayı dayanıklılık günümüz algoritmalarının gücünün ölçülmesinde önemli bir kıstas olmuştur. Dayanıklık ölçümü için ise veri türüne göre birçok güvenlik analiz önlemleri bulunmaktadır. Bu analizlere bilgi entropi, kaos etkisi, anahtar boyutu, anahtar duyarlığı, histogram, korelasyon ve diferansiyel saldırılar örnek olarak verilebilir.

Ayrıca gerçekleştirilen güvenlik analizlerinin yanında şifreleme algoritmalarının başarımı; şifrelenmiş verilerin kırılabilme süresinin uzunluğuna, şifreleme ile şifre çözme işlemlerinde harcanan zamana, bellek miktarına, algoritma esnekliğine ve algoritmanın kullanım alanlarındaki performansına bağlıdır [79]. Özellikle kaos tabanlı gerçekleştirilen şifreleme işlemlerinde süre, bellek gereksinimi, dağıtım ve uygulamadaki avantajlar nedeniyle tercih edilir hale gelmiştir. Güvenlik analizlerinin çok iyi olması bazı durumlarda dezavantaj bir durum oluşturabilir. Örneğin güvenlik

analizleri çok iyi bir yöntem süre ve bellek açısından sıkıntı oluşturuyorsa uygulama esnekliği açısından sıkıntı olabilmektedir.

2.1. Şifreleme Bilimi

Şifreleme bilimi bilgi güvenliğinin sağlanması ile ilgili bir bilim dalıdır. Şifreleme işlemlerinde kullanılan algoritmaların tasarımı, geliştirilmesi ve analiz edilmesi şifreleme biliminin temel unsurlarındandır. Algoritma tasarımında güvenliğinin üst düzeyde sağlanabilmesi için dikkat edilmesi gereken birçok unsur vardır. Gizlilik, bütünlük ve erişilebilirlik yanında bunlara ek olarak giriş kontrolü, emniyet, inkâr edememe, güvenirlik, kayıt tutma, kimlik tespiti gibi diğer şartların sağlanması da güvenliği arttırmak için oldukça önemlidir [80]. Yukarıda bahsedilen ana unsurlardan gizlilik; bilgilerin üçüncü kişilerden korunarak sadece yetkili kişiler tarafından elde edilebilmesi, bütünlük; bilginin orjinal halinin korunması yani karşı tarafa bilginin bozulmadan, tahrip olmadan ulaşabilmesi, erişebilirlik ise bilgiye istenilen zamanda kullanıcıların yetkisi dahilinde ulaşabilmesi ve bunun için gerekli önlemlerin alınması olarak tanımlanabilir [81]. Çok karmaşık algoritma tasarımları hem performans açısından tatmin edici olmayabilir hem de istenmeyen hatalara sebep olabilmektedir. Şifre çözme işlemlerindeki küçük bir hata orjinal verinin elde edilememesine neden olacağı için tasarım aşamasında yukarıda bahsedilen unsurlara dikkat edilmelidir. Şifrelemede yapılan hatalar sonraki adımları etkilememeli ve mesajı bozmayarak bütünlük sağlamalıdır. Şifreleme de kullanılan algoritmanın karıştırma ve dağıtma özelliklerinin olabildiğince iyi olmalıdır. Kaos tabanlı yöntemlerle bu özellikler oldukça iyi sağlanmaktadır. Karıştırma ve dağıtma özelliği çok iyi olursa mesajın şifrelenmiş hali ile orjinal hali arasında ilişki kurulması oldukça zor olacaktır [82].

Şekil 2.1'de şifreleme bilimi görsel olarak özetlenerek verilmiştir. Şifreleme bilimi, kriptografi ve kriptoanaliz olarak iki başlık altında ele alınmaktadır. Şifreleme için algoritma tasarımı ve geliştirilmesine yönelik çözümler sunmaya kriptografi yani şifreleme yazılımı, bu çözümleri çürütmeye yönelik gerçekleştirilen çalışmalara ise kriptoanaliz yani şifreleme analizi denilmektedir. Kriptografide şifreleme ve protokol tasarımı yapılırken, kriptoanalizde şifreleme ve protokol analizi yapılarak orjinal veri istenmeyen kişiler tarafından elde edilmeye çalışılmaktadır. Algoritma ve protokol

tasarımı ne kadar güçlü yapılırsa, analizide o kadar uğraştırıcı ve zor olacaktır. Analiz işlemlerinde genellikle süre ön plana çıkmaktadır. Günümüzde tüm şifreleme algoritmalarının aylar belki yıllar alsada belirli bir sürede çözülebilecekleri varsayılmaktadır.

Şekil 2.1. Kriptoloji, kriptografi, kriptoanaliz

Kaotik dinamikler, şifreleme biliminin en temel unsurları olan karıştırma ve yayılma özelliklerini çok iyi sağladıklarından dolayı son zamanlarda şifreleme çalışmalarında artarak kullanılmaya başlamıştır. Kaos tabanlı ilk bilgi gizleme çalışmaları analog olarak gerçekleştirilmiştir. Kaotik sistemlerin çok hassas özellikler göstermesinden dolayı gürültü gibi küçük bozucu etkiler haberleşme sırasında problem çıkabiliyordu. En önemli problemlerden birisi şifrelemenin temel unsurlarından olan bütünlük gibi bazı unsurların sağlanamaması olabiliyordu. Bu sorunun çözümü için senkronizasyonun işlemleri yapılmaktaydı. Teknolojinin gelişmesi, yüksek işlem kapasitesine sahip cihazların artmasıyla birlikte kaos tabanlı sayısal şifreleme uygulamalarında da artış görülmeye başlamıştır.

Kaos tabanlı sayısal şifreleme uygulamalarının analog şifreleme uygulamalarına göre en büyük avantajı senkronizasyon işlemlerine gerek duyulmamasıdır. Bu sayede metin, ses, resim, video gibi tüm multimedia verilerini daha kolay bir şekilde kaos tabanlı olarak şifrelemek mümkündür. Ayrıca geliştirilen kaos tabanlı şifreleme yöntemine göre tüm multimedia verileri tek bir yöntemle şifrelenebilmektedir.

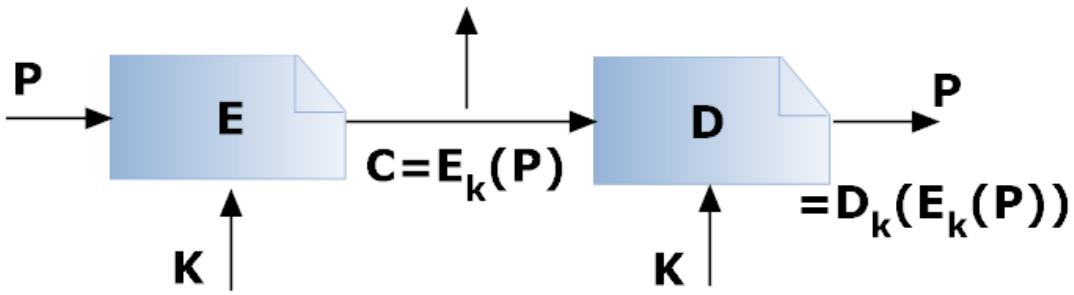
Literatürde gerçekleştirilmiş kaos tabanlı birçok şifreleme çalışmaları bulunmaktadır. Sayısal şifreleme çalışmalarında kaotik dinamikler genellikle anahtar üretimi için kullanılmaktadır. Bunun en önemli nedeni ise anahtarların rasgele üretilmesidir. Üretilen anahtar ile geliştirilen yönteme göre kaos tabanlı şifreleme işlemleri yapılmaktadır. Kaotik sistemler rasgele sayı üretmelerin yanında, gerçek ortam uygulamalarındada süre ve bellek açısından da avantaj sağlamaktadır. Bunun en önemli nedeni ise kaos tabanlı şifreleme yöntemlerinde karıştırma ve hassasiyet özellikleri için çok fazla mantıksal ve matematiksel işlemelere gerek duyulmamasıdır.

Literatürde ayrık ve sürekli zamanlı olarak tek ve çok boyutlu olarak birçok kaotik sistem bulunmaktadır. Yeni kaotik sistem buluşları halen devam etmekte ve değişik uygulamalarda kullanıma sunulmaktadır. Yeni keşfedilen bir kaotik sistemi özellikle kaos tabanlı yeni bir şifreleme yönteminde kullanmak literatürde var olanları kullanmaya göre daha avantajlı olacaktır.

2.2. Şifreleme Teknikleri

Önemli verileri şifrelemek, anlaşılmayacak hale getirmek için günümüze kadar birçok teknik kullanılmıştır. Şifreleme için uygulanacak alana göre genellikle matematiksel ve mantıksal ifadelerden oluşan farklı şifreleme algoritmaları kullanılmaktadır. Birçok algoritma şifreleme işlemleri için anahtar diye ifade edilen bir değer kullanır. Anahtar uzunlukları arttıkça, genellikle şifreleme ve şifre çözme süreleri artmakta, fakat şifreli verinin çözülmeside zorlaşmaktadır. Anahtar uzunluğu fazla olduğu için tahmin edilecek sayı olasılığıda artmış olacak ve üçüncü kişilerin şifreli verileri çözmesi daha fazla süre gerektirebilecektir. Kaos tabanlı olmayan tekniklerde anahtar uzunluğu artma durumunda hem donanımsal hemde yazılımsal gerçeklemelerde sorun olabilmektedir [83]. Kaos tabanlı tekniklerle gerçekleştirilen şifreleme çalışmalarında uygun yöntem kullanılırsa anahtar uzunluğu çok fazla olmuş olsada süre açısından kaos tabanlı olmayan sistemlere göre önemli avantaj sağlamaktadır. Süre ve bellek harcanımı genellikle doğru orantılı olduğu için, süre konusundaki avantaj ve dezavantajlar bellek kullanımı içinde geçerli olacaktır. Şifreleme teknikleri genel olarak Şekil 2.2'de gösterilen orjinal veri (P), anahtar (K), şifrelenmiş veri (C), şifreleme algoritması (E) ve şifre çözme algoritmasından (D) oluşmaktadır.

Haberleşme ortamında şifreli veriler elde edilebilir, okunabilir, fakat şifre çözme algoritması olmadan, uygun çözücü kullanmadan şifrelenen verinin elde edilmesi oldukça zordur, hatta imkansızdır [8]. Bu yüzden şifreleme algoritmaları oldukça önemlidir. Algoritma ve protokol tasarımı üçüncü kişiler için olabildiğince uğraştırıcı olmalıdır.



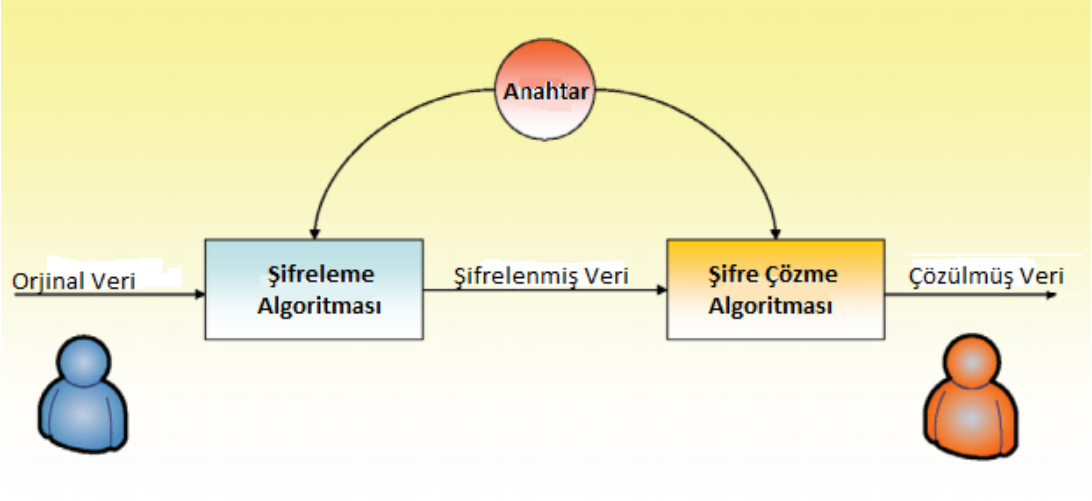
Şekil 2.2. Şifreleme ve şifre çözme işleminin blok şeması

Şifreleme teknikleri anahtarın dağıtımına göre genel olarak simetrik anahtarlı sistemler ve asimetrik anahtarlı sistemler olarak ikiye ayrılmaktadır. Kullanılan anahtarlar gizli veya özel (private) ve açık (public) anahtar dağıtımı olarakta ifade edilebilir.

2.2.1. Simetrik anahtarlı şifreleme yöntemi

Simetrik anahtarlı şifreleme yönteminde, şifreleme ve şifre çözme algoritmalarında aynı anahtarlar kullanılmaktadır. Algoritmada kullanılan anahtarlar mutlaka gizli tutulmalıdır. Anahtar gizli olması gerektiğinden dolayı simetrik anahtarlı şifreleme yöntemi, gizli anahtarlı şifreleme olarakta ifade edilmektedir. Simetrik anahtarlı şifreleme yöntemi için genel blok diyagram Şekil 2.3'de gösterildiği gibidir [84]. Şekilden de görüldüğü üzere orjinal veri bir şifreleme algoritması ile şifrelenmekte ve şifrelenen veri şifre çözme algoritması ile çözülerek tekrar orjinal veri elde edilmektedir. Dikkat edilirse iki işlem içinde ortak bir anahtar kullanılmaktadır. Şifreli veriyi çözmek isteyen kişi ortak anahtarı mutlaka bilmek ve üçüncü kişilerden korumak için gizlemek zorundadır. Günümüzde yaygın olarak kullanılan simetrik şifreleme algoritmaları, asimetrik şifreleme algoritmalarına nazaran hızlıdırlar, donanımla gerçeklemeleri kolaydır, fakat şifreli verilere karşı gerçekleştirilen

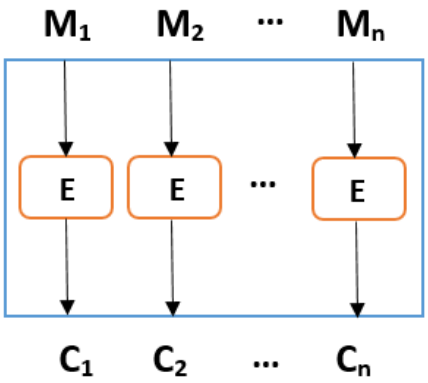
saldırılara karşı daha az dayanıklıdır [8]. Ayrıca anahtar dağıtımı, bütünlük ve kimlik denetimi gibi gereksinimler konusunda zayıftırlar [85]. AES, TEA, SEA, DES, Blowfish, IDEA ve RC4 gibi algoritmalar simetrik anahtarlı yöntemlere örnek olarak verilebilir.



Şekil 2.3. Simetrik anahtarlı şifreleme (Gizli anahtarlı şifreleme)

Şifrelenecek verinin türüne göre simetrik şifreleme algoritmaları blok ve akış (dizi, stream) şifreleme algoritmaları olarak iki başlık altında ele alınabilir. Blok şifreleme algoritmalarında veriler bloklar halinde şifrelenirken, akış şifreleme algoritmalarında veriler bit bit veya baytlar halinde şifrelenmektedir.

Blok şifreleme algoritmalarına AES, DES, IDEA, Serpent, Blowfish, Skipjack, RC5, MD5, TEA, SEA algoritmaları örnek olarak verilebilir. Bu algoritmalar şifrelenecek veriyi bloklar halinde şifreleme algoritmasına alarak sonuçta aynı uzunlukta şifrelenmiş veri blokları üretirler. Şifreli veriler çözülürkende aynı şekilde şifrelenmiş veriler bloklar halinde ele alınır ve orjinal veriler bloklar halinde elde edilir. Her şifreleme algoritmasında olduğu gibi bu algoritma türünde de dağıtım ve karmaşıklık en iyi düzeyde sağlanmış olmalıdır.



Şekil 2.4. Blok şifre sistemlerinde şifreleme

Örnek olarak Şekil 2.4'de görüldüğü gibi blok şifreleme algoritma sistemini, M, E ve C terimleri üzerinden anlatacak olursak. M1; M2; : : ; Mn şeklinde bloklara ayrılan orjinal veri, E şifreleme işlemi sonunda C1; C2; : : : ; Cn olarak şifreli bloklar haline dönüşmektedir [85]. Şifreleme işlemine başlamadan önce belirli bir sıra ve düzen belirlenmeli, karıştırma işlemi için blok dağıtımları iyi yapılmalıdır. Blok şifreleme algoritmalarında bloklar ayrı olarak kullanılabildiği gibi, belirli bir düzen içerisinde birbirlerine bağımlı bloklar halinde de şifreleme işlemleri yapılabilmektedir.

Akış şifreleme algoritmalarına ise RC4, A5/1, A5/2, Panama algoritmaları örnek olarak verilebilir. Akış şifreleme algoritmalarında veriler bit veya baytlar halinde şifrelendiği için bit katarı veya dizi şifreleme algoritmaları olarakta isimlendirilebilir. Akış şifreleme algoritmaları genellikle hızın önemli olduğu uygulamalarda tercih edilen bir yöntemdir. Bit bit şifreleme işlemi yapılabildiği için her bir bit ayrı ayrı bir fonksiyon yardımıyla şifrelenebilmektedir. Şifreli verileri çözmek için sırayla şifre çözüm işlemi yapılırsa her bir bit birbirine bağımlı olabilmektedir. Bu tarz şifreleme yöntemlerinde bir bitin çözülebilmesi için bir önceki bite ihtiyacı duyulmaktadır. Yani orjinal veri bir önceki şifreli metinlerin ve anahtarın bir fonksiyonu ile elde edilir [86]. Bu durumlarda hassasiyet özellliği ön plana çıkmaktadır. Şifreli veri çözümündeki bir hata orjinal verinin elde edilememesine neden olacaktır. Bu tez çalışmasında şifreleme tekniği olarak, simetrik anahtarlı şifreleme yönteminden olan akış şifreleme algoritması kullanılmıştır. Kaos tabanlı rasgele sayı üreteçleri yeni bir şifreleme yöntemi kullanılarak multimedia verileri üzerinde şifreleme işlemleri

gerçekleştirilmiştir. Aşağıda bazı simetrik şifrelemele yöntemleri açıklamalarıyla beraber verilmiştir.

2.2.1.1. AES şifreleme yöntemi

AES algoritması; uzunluğu 128 bitte sabit olan blok ile uzunluğu ile 128, 192 ya da 256 bit olan anahtarlar kullanır [87]. 2010 yılından sonra en çok kullanılan şifreleme yöntemlerinden birisi olmuştur. Her döngüde tersi alınabilir işlemler ve farklı anahtar materyalleri kullanılarak, son döngü hariç 4 dönüşüm kullanılır (SubBytes, ShiftRows, MixColumns ve AddRoundKey) [88].

2.2.1.2. DES şifreleme yöntemi

DES algoritması, dünyada en çok kullanılan simetrik blok şifreleme algoritmalarından birisidir. DES 64 bitlik blok uzunluklu verileri, 56 bitlik anahtar kullanarak şifreler [89]. DES şifreleme yönteminin en önemli dezavantajı anahtar uzunluğunun diğer yöntemlere göre kısa olmasıdır [88].

2.2.1.3. Skipjack şifreleme yöntemi

Skipjack şifreleme yöntemi, 64 bit uzunluğundaki verileri, 80 bit anahtar kullanarak şifreleme işlemi gerçekleştirmektedir. Şifreleme işlemi 32 döngü kullanılarak yapılmaktadır. DES şifreleme algoritması ile karşılaştırıldığında, anahtar boyunun daha uzun olması, daha basit, az işlem gerçekleştirmesi ve şifreli verinin 32 bit sonunda elde edilmesi önemli avantajları olarak ön plana çıkmaktadır. Bu nedenlerden dolayı DES ile karşılaştırışdığında daha güvenli bir şifreleme yöntemi olmaktadır [79].

2.2.1.4. RC5 ve RC6 şifreleme yöntemleri

RC6 şifreleme yöntemi, 1998 yılında RC5 şifreleme yönteminin üst versiyonu olarak geliştirilmiş bir şifreleme algoritmasıdır. RC5 şifreleme yöntemi; 16, 32 ve 64 bitlik blok yapıları ile çalışabilirken, RC6 şifreleme yöntemi; 128 bitlik blok yapıları ile çalışmakta ve 128, 192 ve 256 bitlik anahtarlarla şifrelenmektedir. RC6 şifreleme

yöntemi basit ve hızlı olmasından dolayı gerçek ortam uygulamaları için ideal bir şifreleme yöntemi olarak düşünülmektedir.

2.2.1.5. XTEA şifreleme yöntemi

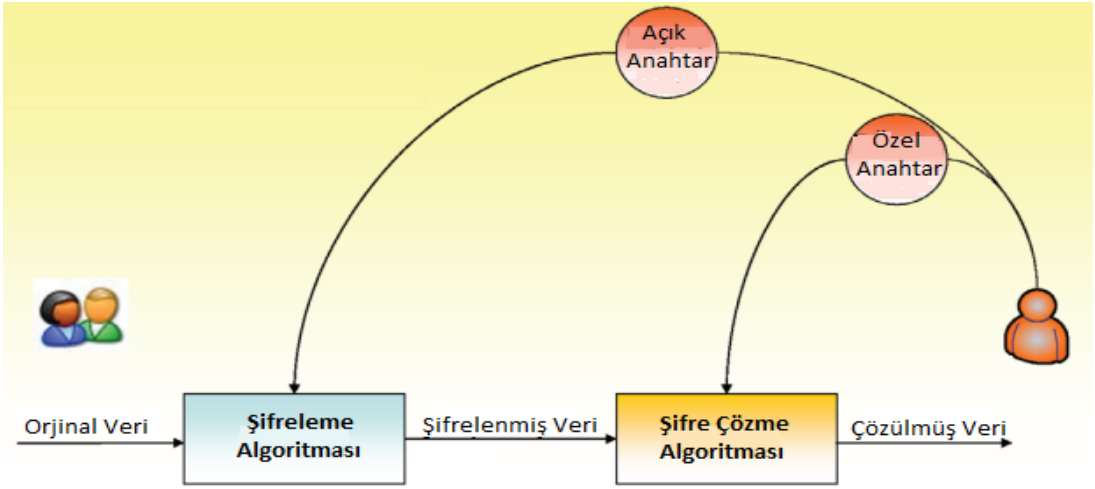
XTEA şifreleme yöntemi, 1997 yılında TEA şifreleme yönteminin zayıflıklarını düzeltmek için geliştirilmiş bir şifreleme yöntemidir. XTEA şifreleme yöntemi de, TEA şifreleme yönteminde olduğu gibi 64 bit uzunluklu blokları, 128 bitlik anahtarlarla şifrelemektedir.

2.2.1.6. CAST5 şifreleme yöntemi

Cast5 şifreleme yöntemi Cast-128 olarakta bilinmektedir. 1996 yılında geliştirilen bu şifreleme yöntemi, 64 bitlik blokları, 40 ila 128 bitlik anahtarla, 12-16 arası döngü ile şifrelemektedir. Cast5 şifreleme yönteminden sonra, 1998 yılında Cast-256 olarakta bilinen, Cast6 şifreleme yöntemi geliştirilmiştir. Cast 6 şifreleme yöntemi ile de, 128 bit uzunluklu bloklar, 128, 160, 192, 224 veya 256 bitlik anahtarlarla, 48 döngü ile şifrelenmektedir [90].

2.2.2. Asimetrik anahtarlı şifreleme yöntemi

Açık anahtarlı şifreleme yöntemi olarakta bilinen asimetrik anahtarlı şifreleme yönteminde şifreleme ve şifre çözme işlemleri simetrik anahtarlı şifreleme yönteminde olduğu gibi ortak değildir. Asimetrik anahtarlı şifrelemede açık ve özel olarak iki tür anahtar bulunmaktadır. Şekil 2.5'de görüldüğü gibi şifreleme işleminini gerçekleştiren anahtar açık anahtar, şifre çözme işlemini gerçekleştiren anahtar ise özel anahtar olarak isimlendirilebilir [84]. Açık anahtara sahip olan kişi veriyi sadece şifreleyebilir, çözemez. Özel anahtarı olan kişiler şifreli verileri çözüp, okuyabilirler. Şifre çözme işleminde kişiye özel anahtar bulunduğu için şifreleme işleminde kullanılan anahtarın açık olması, simetrik anahtarlı şifreleme yönteminde olduğu gibi herkes tarafından bilinmesi problem değildir [87].



Şekil 2.5. Asimetrik şifreleme (Açık anahtarlı şifreleme)

Asimetrik anahtarda birden fazla kullanıcı varsa her şahsın kendine ait bir özel anahtarı vardır, bu anahtar sadece o kişi içindir ve verilerin güvenliği için o anahtarın gizli tutulması gerekmektedir [83]. Asimetrik anahtarlı şifreleme yöntemi, simetrik anahtarlı şifreleme yöntemine göre daha dayanıklıdır, güvenli ve kırılması zor algoritmalardır. Fakat hız açısından karşılaştırıldıklarında asimetrik algoritmalar simetrik algoritmalara göre çok daha yavaştırlar (örneğin 1500 kat kadar). Ayrıca simetrik anahtarlı şifreleme yöntemindeki gizlilik, anahtar yönetimi, bütünlük, kimlik denetimi gibi problemler asimetrik anahtarlı şifreleme yöntemi ile giderilebilir. Şifreleme algoritmalarında güvenlik genelde anahtar uzunluklarına bağlıdır ve bu yüzden seçilen anahtar uzunluğu şifrelenmek istenen veri türüne uygun olmalıdır. Asimetrik algoritmalar bazı durumlarda anahtar uzunluk açısından kullanıma uygun değildir [82]. Simetrik algoritmalarda istenilen her türlü veri gerçekleştirilen tasarıma göre kolaylıkla şifrelenebilmektedir. Asimetrik anahtarlı şifreleme yöntemine RSA, ECC, DSA ve Elgamal algoritmaları örnek olarak verilebilir. Tez çalışmasında anahtar dağıtımı gibi konusunda sıkıntı yaşanan yerlerde anahtar şifreleme işlemleri için yeni bir kaotik sistem ve oldukça güvenilir olan asimetrik RSA şifreleme algoritması kullanılacaktır. RSA hakkındaki temel bilgiler bir sonraki bölümde verilecektir.

2.2.2.1. RSA şifreleme yöntemi

RSA (Ronald Rivest, Adi Shamir, Leonard Adleman) açık anahtarlı şifreleme yöntemi, adını bulucularının soyisimlerinin ilk harflerinden almıştır. RSA yönteminin zorluğu, tam sayıları çarpanlarına ayırman probleminin kolay olmamasına dayanmaktadır. Simetrik şifreleme olduğu gibi anahtar gizleme sorunu bulunmamaktadır. RSA kullanıcısı, iki asal sayının çarpımı sonucu bir değer seçer ve ortak anahtar oluşturur, fakat asal sayıları saklar. Anahtar açıktır ve isteyen şifreleme işlemi gerçekleştirebilir. Şifreli veriyi çözmek isteyen kişinin gizli bir anahtarı olmalıdır. Gizli anahtar için ise asal sayıların bilinmesi gerekmektedir [91, 85].

RSA şifreleme yöntemi anahtar üretimi, şifreleme ve şifre çözme olarak 3 basamaktan oluşmaktadır. İlk basamak olan anahtar üretimi için;

- Yakın uzunlukta iki farklı p ve q olarak asal sayılar seçilir.
- n=p*q hesaplanır. n değeri gizli ve açık anahtarda mod değeri olarak kullanılır.
- Φ = (p-1)(q-1) totienti değeri hesaplanır.
- e rasgele sayısı seçilir ve ortak anahtar olarak belirlenir. Fakat bu sayı $1 < e <$ Φ arasında olmalıdır ve Φ ile en büyük ortak böleni 1 olmalıdır. Yani Φ ve e aralarında asal olmalıdır.

Genişletilmiş Öklid Algoritması yardımıyla $ed \equiv 1 \pmod{\Phi}$ ve $1 < d <$ Φ koşulunu sağlayan d değeri belirlenir. d değeri gizli anahtar üssü olarak saklanır.

Gerçekleştirilen işlemlerde; ortak anahtar, mod değeri olan n ve ortak üs olan e sayısından oluşur. Gizli anahtar ise, mod değeri olan n ve gizli anahtar üssü olan gizli kalması gereken d'den oluşur. d değeri p,q ve Φ değerlerinden hesaplandığı için gizli kalmalıdır.

RSA yönteminde bir diğer basamak ise şifreleme işlemidir. Şifreleme işlemi için;

- A kişisi açık anahtarı, yani n ve e'yi B'ye gönderir, gizli anahtarı saklar.

- B mesajı A'ya göndermek istediği zaman mesajı ters çevrilebilir bir protokol ile (dolgu şeması) rassallaştırılır $0 < m < n$ olacak şekilde bir m tamsayısına dönüştürülür. Şifrelenmiş mesaj $c \equiv m^e \pmod{n}$ olarak hesaplanır.
- Son olarak B c'yi, yani şifrelenmiş mesajı A'ya göndererek şifreleme işlemi tamamlanmış olur [91].

Son basamak olan şifre çözme işlemi için ise, c şifrelenmiş metninden açık metni bulabilmek için;

- d gizli anahtarı kullanılarak, $m \equiv c^d \pmod{n}$ işleminden m elde edilir.
- m bulunduktan sonra kullanılan dolgu şemasının tersi alınarak orjinal metin elde edilebilir.

Günümüz şifreleme çalışmalarında asimetrik şifreleme algoritmaları hız olarak yavaş ve bellek olarak çok yer kapladıklarından dolayı, hybrid yöntemler içerisinde kullanılmaktadır. Yani orjinal veriyi şifreleme işlemi için simetrik, şifrelenen verinin anahtarının şifrelenmesi ve dağıtımı için asimetrik şifreleme yöntemleri kullanılmaktadır. Tez çalışmasında da buna benzer olarak hybrid şifreleme işlemleride gerçekleştirilmiştir.

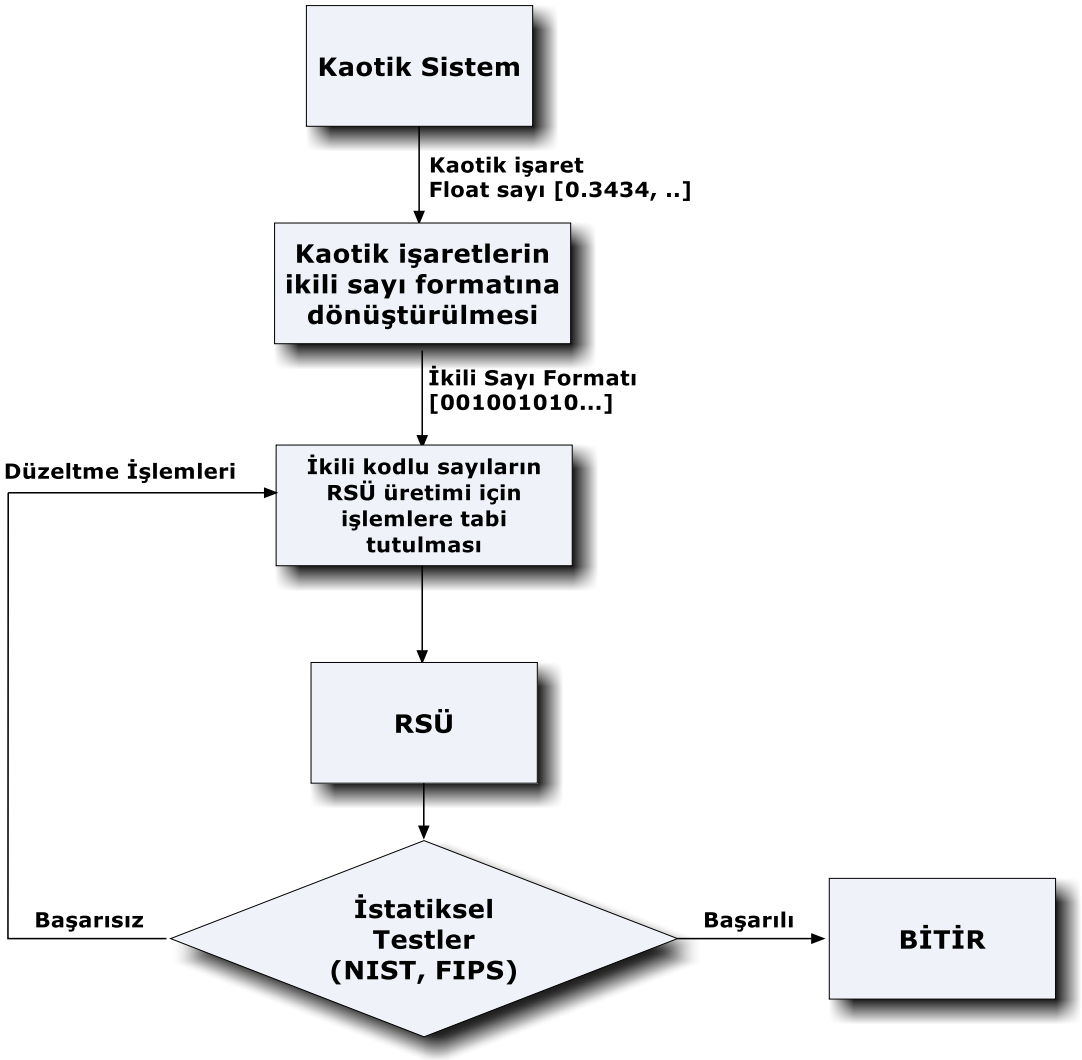
2.3. Rasgele Sayı Üreteçleri ve İstatistiksel Rasgelelik Testleri

2.3.1. Rasgele sayı üreteçleri

Şifreleme uygulamalarında kullanılan anahtarların rasgele olması en önemli unsurlardan birisidir. Günümüze kadar bu ihtiyacı gidermek için birçok çalışmalar yapılmıştır [92]. Gelişen teknoloji sayesinde kaotik sistemlerin analog veya sayısal olarak kolaylıkla gerçeklenebilmeleri, karıştırma ve yayılmayı çok iyi sağlamaları, düşük güçlerde ve yüksek frekanslarda çalışabilmeleri RSÜ olarak kullanımında avantajlı hale gelmeye başlamıştır [93]. Elde edilen RSÜleri ile birçok ortamda anahtar olarak kullanılarak şifreleme işlemleri daha güvenilir bir şekilde gerçekleştirilebilmektedir.

RSÜ'ler genel olarak Sözde RSÜ ve Gerçek RSÜler olarak iki bölüm altında incelenmektedirler [94]. Sözde RSÜler yazılımsal olarak gerçekleştirilirken, Gerçek RSÜler donanımsal olarak gerçekleştirilmektedirler. Gerçek RSÜler kendi içerisinde analog tabanlı RSÜler ve sayısal tabanlı RSÜler olarak iki başlık altında incelenebilir. Sürekli zamanlı kaotik RSÜler analog tabanlı RSÜler içerisinde ele alınırken, ayrık zamanlı kaotik RSÜler sayısal tabanlı RSÜler içerisinde ele alınmaktadırlar.

RSÜ devrelerinin gerçeklenmesinde kaotik sistemler, sahip oldukları özelliklerden (başlangıç şartlarına hassas bağımlılık, rasgelelik vb.) dolayı özellikle şifreleme çalışmalarındaki anahtar üretimi için kullanılabilmektedir. Şekil 2.6'da kaotik bir sistem ile RSÜ için gerekli olan adımlar anlatılmıştır. Kaotik bir sistemden elde edilen sayılar ilk olarak float sayılardır. RSÜ için bu sayıların öncelikle normalizasyon veya başka yöntemlerle ikili sayı koduna dönüştürülmeleri gerekmektedir. İkili sayı koduna dönüştürülen sayılar istatistiksel testler için belirli işlemlere tabi tutulmalıdır. İşlemler sonucu elde edilen rasgele sayıların güvenilirliğinden emin olmak için uluslararası kabul görmüş NIST-800-22 veya FIPS-140-1 gibi istatistiksel testlere tabi tutulmalıdır. Test sonucu başarılı olmayan sayı dizileri için testleri geçene kadar düzeltme işlemleri uygulanmalıdır. İstatiksel testlerle ilgili detaylı açıklamalar bir sonraki bölümde verilecektir.



Şekil 2.6. Kaotik sistemler ile rasgele sayı üretimi

2.3.2. İstatistiksel rasgelelik testleri

2.4.2.1. FIPS-140-1 testi

FIPS-140-1 testleri uluslararası düzeyde kabul görmüş olan testlerden birisidir. FIPS-140-1 testlerinin yapılabilmesi için 20.000 bitlik ikili sayı formatından oluşan veri setine ihtiyaç duyulmaktadır. FIPS-140-1 testinde, üretilen sayıların rasgele kabul edilebilmeleri için 4 farklı testden geçmesi gerekmektedir. Bu testler Monobit, Poker, Koşu ve Uzun Koşu testleridir. Testlerin başarılı koşulları aşağıda verildiği gibidir. Testlerde "0" ve "1"lerden oluşan 20.000 bitlik veri setlerinin kullanıldığı varsayılmıştır.

- Monobit Testi

Sayı dizisindeki "1" sayısının $9654<n<10346$ aralığında olması gerekmektedir [95].

- Poker Testi

m bitlik blok parçalarının, Denklem (2.1)'de verilen birbirini tekrar etme sayısı olan X değerinin $k=20000$ ve $m=4$ için, $1.03 < X < 57.4$ aralığında olması gerekmektedir [96].

$$X = \frac{2^m}{k}\left(\sum_i^{2^m} n_i^{\,2}\right) - k \tag{2.1}$$

- Koşu Testi (Run Test)

Bu testde ardarda gelen "0" ve "1" lere bakılmaktadır. Tablo 2.1'de, sayı dizilerinin koşu testinden başarılı sayılabilmesi için blok uzunluklarına göre x bit dizisinin olması gerektiği aralıklar verilmiştir [97].

Tablo 2.1. Koşu testi için blok uzunluklarına göre blok sayıları

Blok Uzunluğu	Blok sayısı aralığı
1	2267≤x≤2733
2	1079≤x≤1421
3	502≤x≤748
4	223≤x≤402
5	90≤x≤223
6 ve 6+	90≤x≤223

- Uzun Koşu Testi (Long Runs Test)

Bu testde de koşu testinde olduğu gibi ardarda gelen "0" ve "1"lere bakılmaktadır. Buradaki yeter şart ise 34 veya daha fazla aynı sayının arka arkaya gelmemesidir [98].

2.4.2.2. NIST-800-22 testi

RSÜ'leri için yapılan testlerden bir diğeri ise NIST-800-22 testidir. NIST-800-22 testi uluslararası düzeyde kabul görmüş testlerden bir diğeridir. NIST -800-22 testi için 1 milyon bit dizisi gerekmektedir. NIST-800-22 testi, FIPS-140-1 testine göre daha karmaşık ve detaylı olarak bit dizilerie testlere tabi tutulmaktadır. FIPS-140-1 testini geçen bit dizileri, NIST-800-22 testinden kalabilmektedir. Güvenilirlik bakımından FIPS-140-1 ve NIST-800-22 testleri karşılaştırıldığında, NIST-800-22 testi daha güvenilir bir test olarak kabul edilmektedir.

NIST-800-22 testi, kendi içerisinde 16 farklı testden oluşmaktadır. NIST-800-22 testine tabi tutulan bit dizisinin başarılı sayılabilmesi için bu testlerin hepsinden başarıyla geçmesi gerekmektedir. NIST-800-22 testinde, çıkan sonuçlar değiştirilebilen P-değerine göre değerlendirilmektedir. Eğer koşul olarak P-değeri 0.001 kabul edilmişse, bit testin başarılı olabilmesi için P-değeri, *0.01<P-değeri<0.001* aralıklarında olması gerekmektedir. NIST-800-22 testinde bulunan ve bit dizilerinin rasgeleliği tanımlayan 16 istatistiksel test ve detaylı açıklamaları aşağıda verildiği gibidir [93].

- Frekans testi

Frekans testi, bit dizisindeki *"1"* ve*"0"* dengesini inceler. NIST-800-22 testlerinin sonuçlarında P değerlerine bakılmaktadır. Frekans testi için P-değeri Denklem (2.2)'de verildiği gibi bulunabilir.

$$P-value = erfc\left(\frac{S_{obs}}{\sqrt{2}}\right) \tag{2.2}$$

Verilen denklemlerdeki bilinmeyeler ise aşağıdaki Denklem (2.3), (2.4) ve (2.5)'den bulunmaktadır.

$$S_{obs} = |S_n| / \sqrt{n} \tag{2.3}$$

$$erf(z) = \frac{2}{\sqrt{\pi}} \int_{z}^{\infty} e^{-u^2} du \tag{2.4}$$

$$erfc(u) = 1 - erf(u) \tag{2.5}$$

$\varepsilon = \varepsilon_1, \varepsilon_2, \ldots\ldots \varepsilon_n$ üretilen bit dizisi olsun. n değeri bit uzunluğudur. Denklem (2.3)'de verilen S_n, dizideki sayıların toplam değeridir. 0 görülen yerlere -1 değeri verilir. Örneğin bit dizisi $\varepsilon = \varepsilon_1, \varepsilon_2, \varepsilon_3 \ldots\ldots \varepsilon_{15} = 011010011100101$ olsun. Burada n=15 olmaktadır. $S_n = (-1)+1+1+(-1)+1+(-1)+(-1)+1+1+1+(-1)+(-1)+1+(-1)+1=1$ olur.

Buradan;

$$S_{obs} = |1| / \sqrt{15} = 0.2582 \tag{2.6}$$

Denklem (2.6)'daki bulunan S_{obs} gözlemlenen değer ile Denklem (2.4)'deki *erf* fonksiyon değeri, bu değer ile de Denklem (2.5)'deki *erfc* fonksiyonu değeri hesaplanarak P değeri *0.7962* olarak bulunabilir.

Sonuç olarak *P-değeri=0.7962≥ 0.001* olduğundan verilerin rasgele olduğu ve frekans testinden geçtiği söylenebilir [99, 100].

- Bir blok içerisinde frekans testi

Frekans testinden farklı olarak, genel anlamda yine bloklar halinde *"1" ve "0"* dengesi incelenir. Bit dizisi M bitlik bloklara bölünerek, bölünen blok içerisindeki *"1"* oranı incelenir. Eğer M=1 ise, bloklara ayırma olmadığı için gerçekleştirilen test, frekans testiyle aynı olmaktadır.

Test istatistiklerinin ve referans dağılımının hesaplanması için, aşağıda verilen Denklem (2.7)'deki ki-kare (χ^2) dağılımı kullanılmaktadır.

$$\pi_i = \frac{\sum_{j=1}^{M} \varepsilon_{(i-1)M+j}}{M} \tag{2.7}$$

χ^2 dağılım istatistiği olan $\chi^2(obs)$ ise, aşağıda verilen Denklem (2.8) ile hesaplanmaktadır.

$$\chi^2(\text{obs}) = 4M \sum_{i=1}^{N} \left(\pi_i - 1/2\right)^2 \tag{2.8}$$

P değerini bulmak için, yukarıdaki denklem yardımıyla bulunan $\chi^2(obs)$ değeri, aşağıda verilen Denklem (2.9)'da yerine yazılarak elde edilebilmektedir.

$$P - value = igamc\left(\frac{N}{2}, \frac{\chi^2}{2}\right) \tag{2.9}$$

Denklem (2.9)'da görülen *igamc* ifadesi, *a* ve *x* değişkenlerine bağımlı olan gama fonksiyonudur. Aşağıda verilen Denklem (2.10) ile hesaplanabilmektedir.

$$Q(a,x) \equiv \frac{\Gamma(a,x)}{\Gamma(a)} \equiv \frac{1}{\Gamma(a)} \int_x^{\infty} e^{-t} t^{a-1} dt \tag{2.10}$$

Örneğin örnek bir bit dizisi $\varepsilon=\varepsilon_1,\varepsilon_2,\ldots\ldots\ \varepsilon_{10}$= 0110011010 ve M=3 olarak ele alalım. M=3 iken sondaki bit ihmal edilmiş olacaktır.

Alınan bit dizisine göre, n=10, N=10/3=3 olursa,

π_1=2/3, π_2=1/3, π_3=2/3 ve χ^2(obs)=1 (Denklem 2.8'den) olarak bulunmaktadır. *igamc* *Q(a,0)=1* ve *Q(a,∞)=0* alınarak Denklem (2.10)'daki formül kullnılarak bulunabilir. Bulunan tüm sonuçlar *P-değeri*nde yerine konulursa *0.801* olarak elde edilir [101].

P-değeri=0.801>0,001 olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Akış testi (The Runs Test)

Bu testde, dizideki *"1" ve "0"* bloklarının osilasyonunun değişimi yavaş veya hızlı olarak incelenmektedir [93]. Bu test için öncelikle, $\tau = 2/\sqrt{n}$ olmak üzere Denklem (2.11) şartının sağlanması gerekmektedir.

$$\left|\pi - \frac{1}{2}\right| \geq \tau \tag{2.11}$$

Örnek olarak elimizde 100 bitlik bir veri dizisi olduğunu ve bunlardan 42 tanesinin "1" olduğunu düşünürsek,

Buradan *n=100* $\tau = \frac{2}{\sqrt{100}} = 0.02$ ve $\pi=0.42$ ($\pi=42/100$) değeri elde edilmektedir. Denklem (2.11)'e göre $0.42 \geq 0.02$ şartı sağlandığından akış testi yapılabilmektedir. Daha sonra bit osilasyon sayısının hesaplanması gerekmektedir. Bit osilasyon sayısı ise, aşağıdaki Denklem (2.12) yardımıyla aşağıdaki gibi hesaplanmaktadır.

$$V_n(obs) = \sum_{k=1}^{n-1} r(k) + 1 \tag{2.12}$$

Yani yukarıdaki örnek için V_n değeri *52* olarak bulunur. Son olarak ise elde edilen tüm değerler *P-değeri* için (Denklem 2.13)'de yerine konulursa;

$$P - value = erfc\left(\frac{|V_n(obs) - 2n\pi(1-\pi)|}{2\sqrt{2n}\pi(1-\pi)}\right) \tag{2.13}$$

0.5007 olarak bulunmuş olur. Sonuç olarak yine *P-değeri=0.5007≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Bir blok içerisinde en uzun birler akış testi

Testin adından da anlaşılacağı üzere n bitlik bloklar içerisindeki en uzun ardışık "1" akışına bakılmaktadır. Bu testte, en uzun koşu testi için referans olarak χ^2 dağılımı kullanılmaktadır. Testte *n* dizi uzunluğuna göre önerilen *M* blok uzunluğu değerleri; n=128 için 8, n=6272 için 128, n=750000 için 10000 olmaktadır. Yani teste tabi tutulacak bit uzunluğna göre bit dizisi belirtilen bloklara ayrılarak işlemler gerçekleştirilmektedir. Dağılım için, $\chi^2(obs)$ değeri aşağıda Denklem (2.14)'de verildiği gibi hesaplanmaktadır

$$\chi^2(\text{obs}) = \sum_{i=0}^{K} \frac{(V_i - N\pi_i)^2}{N\pi_i} \tag{2.14}$$

Denklemdeki K ve N değerleri blok uzunluğuna göre belirlenmektedir. Blok uzunluğu, yani M=8 ise; K=3 ve N=16, M=128 ise; K=5 ve N=49, M=10000 ise; K=6 ve N=75 2.28'de verilen denklemde yerlerine yazılmaktadır.

En son P değeri için ise, elde edilen $\chi^2(obs)$ ve *N* değeri aşağıdaki Denklem (2.15)'de yerlerine yazılarak *P-değeri* bulunabilmektedir.

$$P - value = igamc\left(\frac{K}{2}, \frac{\chi^2(obs)}{2}\right) \tag{2.15}$$

Örneğin dizimiz $\varepsilon=\varepsilon_1, \varepsilon_2,\ldots, \varepsilon_{128}$=11001100000101010110110001001100111000000000010010011010101000100010011111010110100000001101011111001100111001101101100010110010 olsun.

Buradan, $n=128$, $\tau = \frac{2}{\sqrt{128}} = 0.02$ olur ve $\pi=0.42$ olarak elde edilir. $n=128$ olduğu için M=8, dolayısıyla K=3 ve N=16 olur. Bu değerlerden, $\pi_0=0.2148$, $\pi_1=0.3672$, $\pi_2=0.2305$, $\pi_3=0.1875$ değerleri bulunarak, $\chi^2(obs)$ denkleminde yerine konulursa 4.882 değeri hesaplanmış olur.

Son olarak bulunan değerler, *P-değeri* için yerine konulursa *0.180* değeri elde edilmektedir. Sonuç olarak *P-değeri=0.180≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- İkili matris derece testi (The Binary Matrix Rank Test)

Bu testte, bit dizileri belirli bir düzende matris forma dönüştürülür. Oluşturulan matrisin derecesi hesaplanarak bloklar arasında doğrusal bir bağımlılığın olup olmadığına bakılmaktadır. Test istatistiği için yine, referans dağılımı olarak χ^2 dağılımı kullanılmaktadır. Matris sayısı $N=\lfloor n/MQ \rfloor$ şeklinde hesaplanmaktadır. Oluşturulan matrislerden kalan bit sayıları ihmal edilmektedir.

Örneğin dizimiz $n=20$ olmak üzere $\varepsilon=\varepsilon_1, \varepsilon_2, \ldots\ldots \varepsilon_{20}=01011001001010101101$ olsun. $M=Q=3$ ve $N=\lfloor 20/3\cdot 3 \rfloor=2$ olur. Bir matris için kullanılacak bit sayısı $M\cdot Q=3\cdot 3=9$ bit olur. Test içerisinde 2 matris oluşturulacağından $2\cdot 9=18$ bit kullanılır ve kalan 2 bit ihmal edilmektedir. Sonuç olarak aşağıdaki matrisler oluşacaktır.

$N_1 = \begin{vmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 1 & 0 \end{vmatrix}$ ve $N_2 = \begin{vmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{vmatrix}$ olmaktadır. Elde edilen değerler kullanılarak P_m, P_{m-1} ve P_{m-2} değerleri sırasıyla *0.2888, 0.5776 ve 0.1284* olarak hesaplanmaktadır.

Ki-kare dağılımının hesaplanabilmesi için Denklem (2.16) kullanılmaktadır.

$$\chi^2(\text{obs}) = \frac{(F_M - P_m \cdot N)^2}{P_m N} + \frac{(F_{M-1} - P_{m-1} \cdot \text{N})^2}{P_{m-1}\text{N}} + \frac{(N - F_M - F_{M-1} - P_{m-2}N)^2}{P_{m-2}N} \quad (2.16)$$

Bulunan değerler Denklem (2.16) kullanılarak yerine konulursa *0.597* olarak hesaplanmaktadır.

Buradan $P - value = e^{-\chi^2(obs)/2} = e^{-0.5969/2} = 0.742$ bulunmaktadır. Sonuç olarak *P-değeri=0.742≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Ayrık fourier dönüşüm testi

Bu testte dizinin periyodikliği incelenmektedir. Örnek bir bit dizisi ile açıklayacak olursak; Bit dizimiz *n=100* olmak üzere $\varepsilon=\varepsilon_1, \varepsilon_2, \ldots\ldots \varepsilon_{100}=$*1100100100001111110110101010001000100001011010001100001000110100 1100010011000110011000101000101110000* olsun. Buradan dönüşüm sonucunda *X=1, 1,-1,-1, 1,-1,-1, 1,-1,-1 ,..., -1,-1,-1* elde edilir. Yani "1" olan yerlere "1", "0" olan yerlere "-1" yazılır. Elde edilen *X* dizisine ayrık Fourier dönüşümü (AFD) uygulanarak *S=AFD(X)* hesaplanır. *S* aşağıda Denklem (2.17)'de verildiği gibi bulunmaktadır.

$$S_j = \sum_{k=1}^{n} x_k e^{(2\pi_i (k-1) j / n)} \quad (2.17)$$

Bu testde önemli kriterlerden biri tepe yüksekliği eşik değeri olan $T = \sqrt{3n} = \%95$ değerini aşmamalıdır. Verilen örnek bit dizisinde *T* değerinden daha küçük tepe yüksekliklerinin beklenen teorik değeri olan $N_0=0.95*n/2=0.95*100/2=47.5$ olarak hesaplanmaktadır. Bu örnek için *T*'den daha az gerçek gözlemlenen tepe sayısı $N_1=46$ olarak bulunmaktadır.

Bulunan N_1 ve N_0 değerlerinden, test istatistiğinin hesaplanabilmesi için $d = (N_1 - N_0) / \sqrt{n(0.95)(0.05)/4}$ ifadesi ile $d = (47.5 - 46) / \sqrt{100(0.95)(0.05)/4} = -1.3764946$ olarak bulunur. Son olarak ise, $P - value = erfc(|d| / \sqrt{2}) = erfc(1.3764964 / \sqrt{2}) = 0.168$ elde edilmektedir. Sonuç olarak, *P-değeri=0.168≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Örtüşmeyen şablon eşleştirme

Bu testte, periyodik olmayan önceden belirlenmiş örnek bir dizinin bulunma sıklığının olup olmadığı incelenmektedir [93].

Örnek bit dizisi *ε=10100100101110010110* olsun. Dolayısıyla *n=20*, *N=2* (Blok sayısı) alınırsa, *M=n/N=20/2=10* (Test edilecek alt blok) olarak hesaplanmaktadır. N=2 olduğu için M_1*=1010010010* ve M_2*=1110010110* olarak iki blok elde edilmektedir. Test için gerekli olan ifadelerden *B* istatistiksel testin içerisinde bulunan periyodik olmayan şablon örnekleri, *j=1,2,.....N* ise *Wj* blok içerisinde *m* bitlik özel *B* bloğunun kaç defa bulunduğunu ifade etmektedir.

Tablo 2.2'de, *m=3* ve *B=001* örnek şablonu için *B* şablonunun her iki blokta kaç defa bulunduğunu göstermektedir. M_1 ve M_2 blokları içerisinde *"001"*, bulunduğu takdirde W_1 değeri 1 arttırılır ve W_1*=1* olmaktadır. Aynı şekilde M_1 bloğundaki 7-9'da olduğu gibi benzer *"001"*, ile karşılaşıldığında W_1 değeri tekrar arttırılmakta ve W_1*=2* olmaktadır. Diğer durumlarda arttırma işlemleri yapılmamaktadır.

Tablo 2.2. m=3 için M1 ve M2 blokları içerisinde B=001 şablonunun incelenmesi

Bit Pozisyonları	M_1 Bloğu		M_2 Bloğu	
	Bitler	W_1	Bitler	W_2
1-3	101	0	111	0
2-4	010	0	110	0
3-5	100	0	100	0
4-6	001 (Bulundu)	Arttır 1	001 (Bulundu)	Arttır 1
5-7	Test Edilmedi	1	Test Edilmedi	1
6-8	Test Edilmedi	1	Test Edilmedi	1
7-9	001	Arttır 2	011	1
8-10	010 (Bulundu)	2	110	1

Sonuç olarak, eşleşen *"001"* sonucu, tablodan $W_1=2$ ve $W_2=1$ olarak elde edilmektedir. Testin gerçekleştirilebilmesi için, ortalama değer ve varyansın bulunması gerekmektedir.

Öncelikle rasgelelik testi için;

Ortalama değeri $\mu = (M - m + 1)/2^m = (10 - 3 + 1)/2^3 = 1.00$

Varyans değeri $\sigma^2 = M\left(1/2^m - (2m - 1/2^{2m})\right) = 10\left(1/2^3 - (2\cdot 3 - 1/2^{2\cdot 3})\right) = 0.468$ olarak hesaplanmaktadır.

Buna göre χ^2 dağılımı *2.133* olarak elde edilir. Son olarak ise *P-değeri*nin hesaplanabilmesi için bulunan değerler yerlerine yazılırsa *0.344* bulunur. *P-değeri=0.344≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Örtüşen şablon eşleştirme testi

Bu testin örtüşmeyen şablon eşleştirme testinden farkı, eğer örtüşme varsa arama işlemine bir bit sonra devam edilir. Eğer örtüşme bulunmaz ise pencere bir bit kaydırılarak arama işlemine devam edilir.

Örneğin;

$\varepsilon=\varepsilon_1, \quad \varepsilon_2, \ldots\ldots, \quad \varepsilon_{50}=10111011110010110100011100101110111110000101101001$ olsun. Verilen bit dizisinin uzunluğu 50 olduğundan; bu test için $n=50$, N bağımsız blok sayısını ($N=5$), K bağımsızlık katsayısı ($K=2$), M test edilecek olan ε bitlerinin uzunluğunu ($M=10$) olarak alınmıştır. $M=10$ olduğundan 50 adet bit dizisi 10'ar bloklar halinde alınarak, $M_1=1011101111$, $M_2=0010110100$, $M_3=0111001011$, $M_4=1011111000$, $M_5=0101101001$ blokları oluşturulmuştur. Test için, $m=2$ ve $B=11$ referans olarak alınmıştır. Tablo 2.3'de M_1 bloğu için B şablonunun bulunma durumları gösterilmiştir. Bir önceki testde olduğu gibi, aşağıdaki tabloda da görüldüğü üzere eşleşme durumlarında V_i değeri 1 arttırılmaktadır.

Tablo 2.3. M1 bloğu içerisinde B=11 özel şablonunun bulunma durumları

Bit Pozisyonları	M_1 Bloğu	
	Bitler	*Vi*
1-2	10	0
2-3	01	0
3-4	11 (Bulundu)	Arttır 1
4-5	11 (Bulundu)	Arttır 2
5-6	10	2
6-7	01	2
7-8	11 (Bulundu)	Arttır 3
8-9	11 (Bulundu)	Arttır 4
9-10	11 (Bulundu)	Arttır 5

Bir sonraki aşamada, λ değeri ve η değerinin bulunması gerekmektedir.

$\lambda = (M-m+1)/2^m = (10-2+1)/2^2 = 2.25$ ve

η değeri $\eta = \lambda/2 = 2{,}25/2 = 1.125$ olarak bulunur.

Buradan ise, $\chi^2(obs)$ dağılımı için π_i için $\pi_1=0.3246$, $\pi_2=0.1826$, $\pi_3=0.1426$, $\pi_4=0.1066$, $\pi_5=0.0771$, $\pi_6=0.1662$ değerleri olarak hesaplanır. Bulunan değerler yerlerine yazılırsa *3.1667* sonucu elde edilir. Son olarak *P-değeri* nin hesaplanması için Ki-kare dağılımı ve N değerleri test için gerekli denklemde yerlerine yazılırsa

0.274 değeri bulunmuş olur. *P-değeri=0.274≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Maurer'in "Evrensel İstatistik" Testi

Bu testte, bit dizilerinin veri kaybı olmadan sıkıştırılabilirliği incelenmektedir. Bu testin, ayrıca şifreleme uygulamalarında gizli anahtar kaynağı için bir kalite ölçütü olduğu belirtilmektedir [102]. Testdeki *L* her bir bloğun uzunluğu, *Q* başlangıç bölümü ve *K=[n/L]-Q* test bölümünü ifade etmektedir. Örnek bit dizisi, $\varepsilon=\varepsilon_1, \varepsilon_2, \ldots\ldots \varepsilon_{20}=01011010011101010111$ olsun. *n=20* uzunluklu bit dizisi için, *L=2* ve *Q=4* alınırsa, *K=[20/2]-4=6* olarak bulunur. Yani başlangıç bloğu *Q=4*2=* 8 bit, test bloğu ise *K=6*2=12* bit olur. Dolayısıyla başlangıç bloğu *Q=01011010* ve test bloğu ise *K=011101010111* olmaktadır. Bu verilere göre, Tablo 2.4'de Maurer testinde *L*-bit uzunluğundaki blokların bölümleri verilmektedir.

Tablo 2.4. Maurer testi L-bit uzunluğundaki blokların bölümleri

Blok	Blok Tipi	İçerik
1	Başlangıç Bölümü	01
2		01
3		10
4		10
5	Test Bölümü	01
6		11
7		01
8		01
9		01
10		11

Tablo 2.5'de ise, başlangıç bölümü için muhtemel *L*-bit değerleri verilmektedir.

Tablo 2.5. Dört başlangıç değeri ile oluşturulan muhtemel L-bit değerleri

	Muhtemel *L*-bit Değerleri			
Başlangıç	00 (T_0'a kaydedilmiş)	01 (T_1'a kaydedilmiş)	10 (T_2'a kaydedilmiş)	11 (T_3'a kaydedilmiş)
	0	2	4	0

Tablo 2.6'da başlangıç bölümü kullanılarak (4 nolu satır) test bölümündeki *L*-bit bloklarının aldığı değerler görülmektedir. Her *L*-bit değerleri, başlangıç bloğuyla örtüştüğünde test bloğu yine blok numarası değerini almaktadır.

Tablo 2.6. Test bölümü için L-bit değerleri

Tekrar Bloğu	Muhtemel *L*-bit Değerleri			
	00	01	10	11
4	0	2	4	0
5	0	5	4	0
6	0	5	4	6
7	0	7	4	6
8	0	8	4	6
9	0	9	4	6
10	0	9	4	10

Denklem (2.18) ile test istatistiği aşağıdaki şekilde hesaplanmaktadır.

$$f_n = \frac{1}{K}\sum_{i=Q+1}^{Q+K}\log\left(i-T_j\right) = \frac{1}{6}\sum_{i=4+1}^{10}\log\left(i-T_j\right) = \frac{1}{6}\sum_{i=4+1}^{10}\begin{pmatrix}\log(5-2)+\\ \log(6-0)+\\ \log(7-5)+\\ \log(8-7)+\\ \log(9-8)+\\ \log(10-6)\end{pmatrix} = \tag{2.18}$$

$$\frac{1}{6}\sum_{i=4+1}^{10}(1.5849+4.1699+5.1699+5.1699+5.1699+7.1699) = 1.1949$$

Beklenen değer $V_{exp}(L)$ ve varyans $var(f_n)$ değerleri Tablo 2.7'de verilmektedir.

Tablo 2.7. L değerleri için Vexp(L) ve var(fn) değerleri

L	$V_{exp}(L)$	$Var(f_n)$
6	5,2177052	2,954
7	6,1962507	3,125
8	7,1836656	3,238
9	8,1764248	3,311
10	9,1723243	3,356
11	10,1700320	3,384
12	11,1687650	3,401
13	12,1680700	3,410
14	13,1676930	3,416
15	14,1674880	3,419
16	15,1673790	3,421

Tablodan elde edilen değerler, *P-değeri* formülünde yerlerine yazılırsa *0.767* olarak bulunur. Sonuç olarak *P-değeri=0.767≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Doğrusal Karmaşıklık Testi (The Linear Complexity Test)

Bu testte, rasgele bit dizisinin doğrusal geri beslemeli kayan kaydedici uzunluğuna bakılarak dizinin karmaşıklığı incelenmektedir.

Örneğin $\varepsilon = \varepsilon_1,\varepsilon_2,.....,\varepsilon_{13}=1101011110001$ olmak üzere bir bit dizisi alınsın. Blok içerisindeki bit uzunluğu $M=13$ tür. Burada $N=1$ ve $L_i=4$ olmaktadır. T_i dağılımın rasgele değişkenini bulabilmek için, μ değeri (Denklem 2.19) yardımıyla hesaplanabilmektedir.

$$\mu = \frac{M}{2} + \frac{\left(9+(-1)^{M+1}\right)}{36} - \frac{(M/3+2/9)}{2^M} = \frac{13}{2} + \frac{\left(9+(-1)^{13+1}\right)}{36} - \frac{(13/3+2/9)}{2^{13}} = 6.7772 \quad (2.19)$$

olarak bulunur. Bulunan μ değeri T_i için yerine konulursa, (Denklem 2.20) kullanılarak;

$$T_i = (-1)^M \cdot (L_i - \mu) + 2/9 = (-1)^{13} \cdot (4 - 6.777) + 2/9 = 2.999 \quad (2.20)$$

olarak hesaplanmaktadır. π_i değerleri hesaplanarak π_0=0.0104, π_1=0.03125, π_2=0.125, π_3=0.5, π_4=0.25, π_5=0.0625 ve π_6=0.020833 olarak elde edilmektedir. Ki-kare testi için ise hesaplanan değerler (Denklem 2.21) 'de yerine konulursa;

$$\chi^2(\text{obs}) = \sum_{i=1}^{K} \frac{(v_i - N\pi_i)^2}{N\pi_i} = 47.0008 \quad (2.21)$$

bulunur ve son olarak *P-değeri* için bulunan sonuçlar (Denklem 2.22)'de yerlerine yazılırsa;

$$P - value = igamc\left(\frac{K}{2}, \frac{\chi^2(obs)}{2}\right) = igamc\left(\frac{6}{2}, \frac{47.008}{2}\right) = 0.993 \quad (2.22)$$

olur. *P-değeri=0.993≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Seri Testi (The Serial Test)

Bu testte, verilen bit dizisindeki her *m* bit örneğin dizideki diğer *m* bit örnekler ile benzer değişim ve tekdüzelilik seviyesini incelemektedir. Seri test için *m=1* olursa

frekans testi ile aynı olmaktadır. Bu testte *P-değeri 1* ve *P-değeri 2* olarak iki test sonucu elde edilmektedir.

Örnek olarak $\varepsilon = \varepsilon_1,\varepsilon_2,.....,\varepsilon_{10}=0011011101$ olmak üzere $n=10$ bitlik bir dizi verilsin.. ε' dizisi bit dizisi sonuna bit *"0"* eklenmesi ile oluşan artırım dizisidir. Örneğin $m=1$ $\varepsilon'=0011011101$ *(m)* iken, $m=2$ için $\varepsilon'=00110111010$ *(m-1)* ve $m=3$ için $\varepsilon'=001101110100$ *(m-2)* olmaktadır.

Verilen bit dizisinde $m=3$ olduğunda, $m-1=2$ ve $m-2=1$'dir. Bütün 3-bitlik blokların frekansları $v_{000}=0$, $v_{001}=1$, $v_{010}=1$, $v_{011}=2$, $v_{100}=1$, $v_{101}=2$, $v_{110}=2$, $v_{111}=1$, tüm 2-bitlik blokların frekansları $v_{00}=1$, $v_{01}=3$, $v_{10}=3$, $v_{11}=3$ ve bütün 1-bitlik blokların frekansları $v_0=4$, $v_1=6$ olmaktadır.

$\Delta\Psi_m^2(obs)$ dizideki, *m* bit örneğin gözlemlenen frekansı, $\Delta^2\Psi_m^2(obs)$ ise beklenen frekansının ne kadar iyi olduğunu gösteren ölçütlerdir. Ψ_m^2, Ψ_{m-1}^2 ve Ψ_{m-2}^2 değerleri aşağıdaki Denklem (2.23), (2.24) ve (2.25) yardımıyla hesaplanabilmektedir.

$$\Psi_m^2 = \frac{2^m}{n}\sum_{i_1..i_m}\left(v^2_{i_1..i_m} - n\right) \tag{2.23}$$

$$\Psi_{m-1}^2 = \frac{2^{m-1}}{n}\sum_{i_1..i_{m-1}}\left(v^2_{i_1..i_{m-1}} - n\right) \tag{2.24}$$

$$\Psi_{m-2}^2 = \frac{2^{m-2}}{n}\sum_{i_1..i_{m-2}}\left(v^2_{i_1..i_{m-2}} - n\right) \tag{2.25}$$

Örnek dizideki değerler denklemlerde yerlerine yazılırsa; 2.8, 1.2 ve 0.4 değerleri elde edilmektedir.

$$\Delta\Psi_m^2 = \Psi_m^2 - \Psi_{m-1}^2 \tag{2.26}$$

$$\Delta^2\Psi_m^2 = \Psi_m^2 - 2\Psi_{m-1}^2 + \Psi_{m-2}^2 \tag{2.27}$$

Hesaplanan frekans değerleri, rasgelelik testi için genelleştirilmiş seri istatistiklerinin cevabı için (Denklem 2.26 ve Denklem 2.27) kullanılıcak olursa, 1.6 ve 0.8 sonuçları elde edilmiş olur.

Bu testde iki sonuç elde ediliyordu. Referans χ^2 dağılımı için aşağıda verilen (Denklem 2.28 ve 2.29) kullanılmaktadır.

$$P-value1 = igamc\left(2^{m-2}, \nabla\Psi_m^2 / 2\right) \quad (2.28)$$

$$P-value2 = igamc\left(2^{m-3}, \nabla^2\Psi_m^2 / 2\right) \quad (2.29)$$

Bulunan sonuçlar, yukarıda verilen denklemlerdeki yerlerine yazılırsa *P-değeri* sonuçları 0.905 ve 0.880 olarak elde edilmektedir.

Sonuç olarak *P-değeri1=0.905≥0.001* ve *P-değeri2=0.880≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Yaklaşık Entropi Testi (The Aproximate Entropy Test)

Bu testte örtüşen *m* bitlik örnek dizinin frekansı incelenmektedir. Rasgele bir dizi için beklenen frekansın, iki ardışık veya bitişik uzunluktaki örtüşen blokların frekanslarını karşılaştırmaktadır.

Örnek olarak;

$\varepsilon=\varepsilon_1,\varepsilon_2,.....,\varepsilon_{100}$=1100100100001111110110101010001000100001011010001100001
0001101001100010011000110011000101000010111000

şeklinde bir bit dizisi verilsin. Burada, blok içerisindeki bit uzunluğu *n=100* ve *m=2* olmaktadır. Dizinin sonuna dizinin başından *m-1* bit eklenmektedir. Ardından *m=2* olduğundan elde edilen dizi ardışık olarak önce *m* sonra *m+1* bitlik bloklara bölünmektedir. Bir sonraki aşamada *i*, *m* bit blokların değerleri ve C_i^m muhtemel *m* bit

değerlerin sayısı olmak üzere $C_i^m=\pi_i/n$ eşitliği kullanılarak her bir i değeri için, $\pi_i=C_j^m$ ve $j=log_2 i$ olmak üzere m bit uzunluktaki bütün 2^m muhtemel blokların dizi üzerindeki ampirik dağılımın frekansı $\varphi^{(m)}$ Denklem (2.30) kullanılarak hesaplanmaktadır.

$$\varphi^{(m)} = \sum_{i=0}^{2^m-1} \pi_i \log \pi_i \tag{2.30}$$

Ki-kare test istatistiği için;

$$\begin{aligned} ApEn(m) &= \varphi^{(m)} - \varphi^{(m+1)} \\ \chi^2 &= 2n\left[\log 2 - ApEn(m)\right] \end{aligned} \tag{2.31}$$

(Denklem 2.31) ‘deki eşitlikler kullanılmaktadır. Verilen 100 bitlik örnek bit dizisi için hesaplamalar yapıldığında sırasıyla 0.6653 ve 5.550 değerleri hesaplanmaktadır.

En son olarak ise, *P-değeri* test için gerekli olan denklem kullanılarak, bulunan ifadeler yerlerine yazılırsa, 0.2353 değeri elde edilmiş olur. Buradan $P\text{-değeri}=0.2353 \geq 0{,}001$ olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Birikimli Toplamlar Testi (The Cumulative Sums Test)

Bu testdeki amaç ise, rasgele bir dizi için birikimli toplamın beklenen davranışı için kısmi alt blokların birikimli toplamının çok büyük veya çok küçük olup olmadığı incelenmektedir. Test için öncelikle bit dizisi, $X_i=2\varepsilon_i-1$ dönüşümü kullanılarak giriş dizisi 0 olan yerlere *-1* ve 1 olan yerlere *+1* yazılarak normalize edilmekte ve böylelikle yeni X_i dizisi elde edilmektedir. Sonucun 0’ a yakın çıkması rasgelelik için ana kriterdir. Örnek olarak, $\varepsilon=\varepsilon_1,\varepsilon_2,.....,\varepsilon_{10}=1011010111$ şeklindeki bir bit dizisi kullanılmıştır. Verilen bit dizisi için normalize yapıldığında elde edilen yeni dizi *X=1, -1, 1, 1, -1, 1, -1, 1, 1, 1* olmaktadır.

Tablo 2.8'de görüldüğü gibi, bu testin uygulanmasında ileri ve geri yönlü olarak, test aşamasında seçilen değere göre (0 veya 1) iki farklı metot kullanılmaktadır.

Tablo 2.8. Test için ileri ve geri yönlü metotların uygulanması

Metot1=0 (İleri yönlü)	Metot2=1 (Geri yönlü)
$S_1=X_1$	$S_1=X_n$
$S_2=X_1+X_2$	$S_2=X_n+X_{n-1}$
$S_3=X_1+X_2+X_3$	$S_3=X_n+X_{n-1}+X_{n-2}$
:	:
$S_k=X_1+X_2+X_3\cdots+X_k$	$S_k=X_n+X_{n-1}+X_{n-2}\cdots+X_{n-k+1}$
:	:
$S_n=X_1+X_2+X_3\cdots+X_k\cdots+X_n$	$S_n=X_n+X_{n-1}+X_{n-2}\cdots+X_{n-k+1}\cdots+X_1$

Verilen örnek bit dizisi için, uygulama metodunda, 1 seçilerek ileri yönlü metot uygulanmıştır.

$S_1=1$

$S_2=1+(-1)=0$

$S_3=1+(-1)+1=1$

$S_4=1+(-1)+1+1=2$

$S_5=1+(-1)+1+1+(-1)=1$

$S_6=1+(-1)+1+1+(-1)+1=2$

$S_7=1+(-1)+1+1+(-1)+1+(-1)=1$

$S_8=1+(-1)+1+1+(-1)+1+(-1)+1=2$

$S_9=1+(-1)+1+1+(-1)+1+(-1)+1+1=3$

$S_{10}=1+(-1)+1+1+(-1)+1+(-1)+1+1+1=4$

Elde edilen sonuçtan, *z* test istatistik değeri $z=max_{1\le k\le n}|4|=4$ olmaktadır. *P-değeri*nin bulunabilmesi için aşağıda verilen Denklem (2.32) kullanılmaktadır.

$$P-value=1-\sum_{k=\left(\frac{-n}{z}+1\right)/4}^{\left(\frac{n}{z}-1\right)/4}\left[\Phi\left(\frac{(4k+1)z}{\sqrt{n}}\right)-\Phi\left(\frac{(4k-1)z}{\sqrt{n}}\right)\right]+\sum_{k=\left(\frac{-n}{z}-3\right)/4}^{\left(\frac{n}{z}-1\right)/4}\left[\Phi\left(\frac{(4k+3)z}{\sqrt{n}}\right)-\Phi\left(\frac{(4k+1)z}{\sqrt{n}}\right)\right] \quad (2.32)$$

Elde edilen değerler yerlerine koyulduğunda, *P-değeri=0.411* olarak hesaplanmaktadır. *P-değeri=0.411≥0,001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Rasgele Gezinimler Testi (The Random Excursions Test)

Bu testte, birikimli toplam rasgele yürüyüşünde *K* adet döngünün sayısı tespit edilmektedir. Önceki testlerde olduğu gibi *"0"* ve*"1"* değerleri *-1* ve *+1* olarak normalize edillir. Normalize işlemi sonucu bit dizisindeki sayılar kısmi olarak toplanır. Bu test, -4, -3, -2, -1 ve +1, +2, +3, +4 olarak toplam sekiz *P-değeri*nin hesaplandığı serisel bir testtir.

Örnek olarak $\varepsilon=\varepsilon_1,\varepsilon_2,.....,\varepsilon_{10}=0110110101$ şeklinde bir test dizisi verilmiş olsun. Normalize işlemi sonrası, X_i dizisi $X=-1, 1, 1, -1, 1, 1, -1, 1, -1, 1$ olmaktadır. Bu test için verilen örneğe göre bir önceki testte olan birikimli toplamlar testinde verildiği gibi metod 1 seçilerek ileri yönlü metot uygulandığında $S_1=-1, S_2=0, S_3=1, S_4=0, S_5=1, S_6=2, S_7=1, S_8=2, S_9=1, S_{10}=2$ değerleri elde edilmektedir. Sonuç olarak, S_i kısmi toplam olmak üzere tüm değerler, $S=\{-1, 0, 1, 0, 1, 2, 1, 2, 1, 2\}$ olmaktadır.

S' kümesi yukarıda bulunan *S* kümesinin başına ve sonuna 0 elemanları eklenerek $S'=\{0, -1, 0, 1, 0, 1, 2, 1, 2, 1, 2, 0\}$ şeklinde elde edilir. Döngü sayısını ifade eden *J* sıfır elemanlarının sayısıdır. İlk "0" dikkate alınmadığı için; $J_1=\{0, -1, 0\}$, $J_2=\{0, 1, 0\}$ ve $J_3=\{0, 1, 2, 1, 2, 1, 2, 0\}$ olarak 3 tane J döngüsü bulunmaktadır. Bu döngüler kullanılarak *x* durum değerlerinin frekansları, "0" hariç $-4\leq x\leq 4$ arasındaki tüm değerler, Tablo 2.9'da verildiği gibi hesaplanmaktadır.

Tablo 2.9. Verilen ε dizisi için oluşan rasgele gezinti döngü frekansları

Durum x	Döngüler (J)		
	Döngü 1 (J_1)	Döngü 2 (J_2)	Döngü 3 (J_3)
-4	0	0	0
-3	0	0	0
-2	0	0	0
-1	1	0	0
1	0	1	3
2	0	0	3
3	0	0	0
4	0	0	0

Sonraki aşamada x değerinin 8 farklı durumu için bütün döngüler arasında k defa meydana gelen x durumundaki döngünün toplam sayısı $v_k(x)$ hesaplanmaktadır. Bulunan sonuçlar kullanılarak, yine x değerinin 8 durumu için aşağıda verilen denklem yardımıyla Denklem (2.33), Ki-kare istatistiği hesaplanmaktadır.

$$\chi^2(\text{obs}) = \sum_{k=0}^{5} \frac{\left(v_k(x) - J\pi_k(x)\right)^2}{J\pi_k(x)} \tag{2.33}$$

Yukarıda elde edilen değerler ile sadece $x=1$ durumu için verilen denklemden 4.3330 sonucu bulunmaktadır.Buradan ise P-değeri 0.502 olarak hesaplanır. Sonuç olarak *P-değeri=0.502≥0.001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

- Rasgele gezinimler değişken testi (The Random Excursions Variant Test)

Son olarak bu testte ise, birikimli toplam rasgele yürüyüşte belirli durumların meydana gelme sayısı incelenmektedir. Bu testte -9, -8, -7, -6, -5, -4, -3, -2, -1 ve +1, +2, +3, +4, +5, +6, +7, +8, +9 olmak üzere on sekiz *P-değeri* hesaplanmaktadır. Test için $\varepsilon = \varepsilon_1,\varepsilon_2,.....,\varepsilon_{10}=0110110101$ olarak bir bit dizisi verilmiş olsun. Normalize işlemi sonucu yeni X_i dizisi $X=-1, 1, 1, -1, 1, 1, -1, 1, -1, 1$ olmaktadır. Bu test için yine önceki testlerde olduğu gibi, verilen örneğe göre birikimli toplamlar testi, metod 1 seçilerek ileri yönlü metot uygulandığında $S=\{-1, 0, 1, 0, 1, 2, 1, 2, 1, 2\}$ olmaktadır. S' kümesi

ise bir önceki test örneğinde olduğu gibi *S'={0, -1, 0, 1, 0, 1, 2, 1, 2, 1, 2, 0}* elde edilmektedir.

Bu testte, *J* değeri rasgele dizideki döngü sayısını ve *ξ* bütün rasgele yürüyüşler süresince ziyaret edilen durumların toplam sayısıdır. Verilen bit dizisi için*: ξ(-1)=1, ξ(1)=4, ξ(2)=3* ve diğerleri *ξ(x)=0* olmaktadır. Her bir döngü için x değeri "0" hariç *-9≤x≤9* ve her bir *x* değerinin frekansı hesaplanmaktadır.

*P-değeri*nin hesaplanabilmesi amacıyla her bir *ξ(x)* değeri için, 18 tane *P-değeri* Denklem (2.34) yardımıyla hesaplanabilmektedir.

$$P-value = erfc\left(\frac{|\xi(x)-J|}{\sqrt{2J\left(4|x|-2\right)}}\right) \qquad (2.34)$$

Sadece *x=1* durumu için ise yukarıdaki denklem kullanıldığında 0.683 sonucu elde edilmektedir. Buradan *P-değeri=0.683≥0,001* olduğundan testi geçmiş, yani örnek olarak alınan bit dizileri rasgele olarak kabul edilmektedir [99, 100].

2.5. Güvenlik Analizleri

2.5.1. Korelasyon analizi

Korelasyon, iki değişken arasındaki doğrusal ilişkinin yönünü ve gücünü belirtir. İki değişkenin kovaryansının, yine bu değişkenlerin standart sapmalarının çarpımına bölünmesiyle elde edilir. Kovaryans ise, iki değişkenin birlikte ne kadar değiştiklerinin ölçüsüdür. Kovaryans, iki rasgele değişkenin beraber değişimlerini inceleyen bir istatistiktir. Korelasyon katsayısı, bağımsız değişkenler arasındaki ilişkinin yönü ve büyüklüğünü belirten katsayıdır. Bu katsayı, (-1) ile (+1) arasında bir değer alır. Pozitif değerler direk yönlü doğrusal ilişkiyi; negatif değerler ise ters yönlü bir doğrusal ilişkiyi belirtir. Korelasyon katsayısı 0 ise söz konusu değişkenler arasında doğrusal bir ilişki [103]. Gerekli hesaplamalar, aşağıdaki Denklem (2.35)' de verilen ifadeler ile yapılmaktadır.

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i,$$
$$D(x) = \frac{1}{N}\sum_{i=1}^{N} (x_i - E(x))^2, \quad (2.35)$$
$$\mathrm{cov}(x, y) = \frac{1}{N}(\sum_{i=1}^{N} (x_i - E(x))(y_i - E(y))),$$
$$r_{xy} = \frac{\mathrm{cov}(x, y)}{\sqrt{D(x)D(y)}}$$

Korelasyon analizi için değişkenler arası ilişkinin doğrusal olması gerekmektedir, eğer doğrusallık yoksa değişkenler arasındaki ilişkiyi ölçmek korelasyon analiz ile uygun değildir [9]. Resimler arasındaki korelasyon analizi için iki dikey, iki yatay, iki çapraz pikseller arasındaki ilişkilere bakılabilir. Orjinal resimde değişkenler arası ilişki doğrusal iken, analiz sonucu değişkenler arası ilişkinin doğrusal olmaması (dağınık olması), oldukça karmaşık dağılım göstermesi, korelasyon değerinin 0'a yakın olması analiz sonucunun iyi olduğunu göstermektedir.

2.5.2. Histogram analizi

Veri dağılımının grafiksel gösterimine histogram dağılımı diyebiliriz. Veri grubunun sıklığı histogramı oluşturmaktadır. Birçok alanda histogram analizi yapılabilmektedir. Şifrelemede ise şifrelenmiş verilerin dağılımında verilerin yaklaşık değerler çıkması şifrelemenin iyi olduğunu göstermektedir. Veriler birbirine ne kadar yakın olursa şifreli verilerin çözümü de o kadar zor olacaktır. Örneğin resim verilerinde histogram dağılımı soldan sağa koyudan renklerden açığa doğru gitmektedir. Sol tarafta fazla dağılım varsa resimde koyu renkler ağırlıkta, sağ tarafta yoğunluk varsa açık renkler ağırlıkta demektir. Bu yüzden dağılım ne kadar iyi olursa resim hakkında fikir edinme, şifreli veriyi çözebilme o oranda zor olacaktır.

2.5.3. Anahtar uzunluk analizi (Key Space)

Güçlü saldırıları etkisiz hale getirmek için anahtar uzunluğu yeterince büyük olmalıdır. Kaotiklik boyutu ve diğer değişkenler arttıkça anahtar uzunluğuda artacaktır. Bir değişken 10^{14} farklı değer alabilmektedir. Örneğin 3 boyutlu ve sadece r parametresine sahip bir sistemde anahtar uzunluğu 10^{56} (10^{42}+10^{14}) dır.

2.5.4. Anahtar duyarlılık analizi (Key Sensitivity)

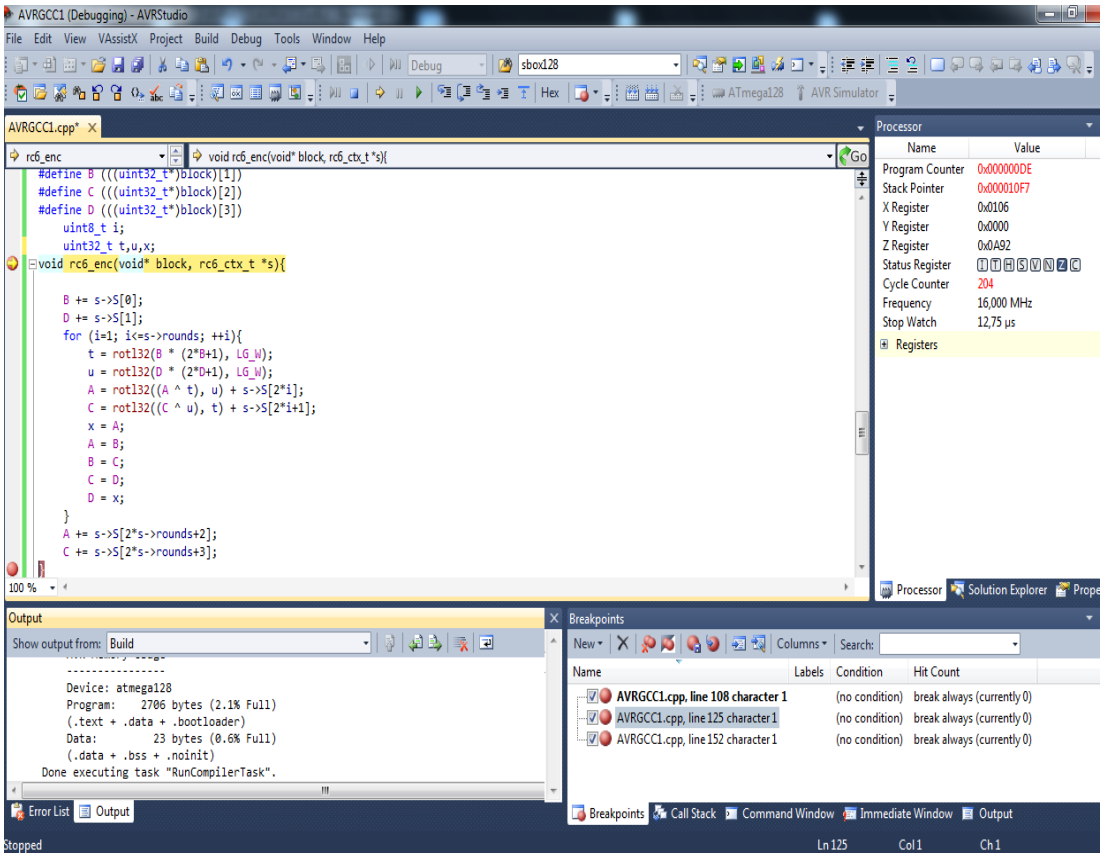
Şifrelenmiş veri çözülürken anahtar üzerindeki küçük bir değişiklik veri çözülürken farklı sonuçlara yol açmaktadır. Güvenli bir şifrelemede anahtarlardaki çok küçük bir değişim atakları engellemelidir. Kaos şifrelemenin önemi de burada ortaya çıkmaktadır. Anahtarlardan birinin değişimi sonucu direk olarak etkilemekte, yani şifrelenmiş veri bir anahtar bile değişmiş olsa çözülememektedir. Yapılan çalışmada her veri için farklı anahtarlar üretildiği için şifreli verinin çözülebilmesi için tüm anahtarların bilinmesi gerekmektedir. Ayrıca üretilen anahtarların sırasının da bilinmesi gerekmektedir. Tüm anahtar bilinse bile sırasıyla şifre çözme işlemi gerçekleştirilmezse, veri çözülemeyecektir. Bazı çalışmalarda şifreleme için birden fazla resim işleme alınarak şifreleme yapılmaktadır. Dolayısıyla herhangi bir resimdeki ufak bir değişimde hassasiyet farklı resimlerde şifrelemeye katıldığı için artabilmektedir. Analizde sonuç olarak çözülen veriye bakılmaktadır. Eğer herhangi bir küçük değişim sonucu orijinal veri elde edilememişse analiz sonucu başarıya ulaşmıştır.

2.5.5. Kaos şifreleme etkisi

Şifrelemedeki kaosun etkisidir. Şifrelenmiş verilerin kaos olması verilerin çözülemeyeceği anlamına gelmemektedir. Veriler çok karmaşık olsa bile doğru yöntemlerle şifreli veriler çözülebilmektedir.

2.6. AVR Studio 5.1

Tez çalışmasında geliştirilen güvenli şifreleme yöntemini; süre ve bellek olarak diğer yöntemlerle karşılaştırabilmek için AVR Studio 5.1 ve WinAVR programları kullanılmıştır. ATMEL firması tarafından geliştirilen program ile Assembly, C ve C++ dillerinde programlar yazılarak, derlenebilmekte ve debug işlemleri anlık olarak gerçekleştirilebilmektedir. ATMEGA 128 mikrodenetleyicisi 131072 bayt Flash, 4096 bayt EEPROM, 4096 bayt Dâhili SRAM, 65536 bayt Harici SRAM ve maximum 16 MHz hız ile yeterli özelliklerine sahip olduğu için çalışmada gerekli süre ve bellek ölçümleri için ATMEGA 128 mikrodenetleyicisi tercih edilmiştir. Süre ve bellek ölçümlerini gösteren örnek ekran görüntüsü Şekil 2.7'de verildiği gibidir [104].



Şekil 2.7. AVR Studio 5.1 örnek görünümü

BÖLÜM 3. KAOTİK SİSTEMLERİN ANALİZ YÖNTEMLERİ, MODELLENMESİ VE DEVRE GERÇEKLEMELERİ

3.1. Kaotik Sistemler

Doğrusal olmayan bir davranış türü sergileyen kaotik sistemler; sınırsız sayıda değişik periyodik salınımlar içerebilir, genlik ve frekansları tespit edilemez. Fakat sınırlı bir alanda kaotik işaretler içeren dinamiklere sahiptirler [105]. Dinamik sistemler şimdiki ve geçmiş durumunun yanında olası durumların kümesini de içermektedir. Kaotik sistemler ayrık zamanlı ve sürekli zamanlı kaotik sistemler olarak iki grup olarak incelenebilir. Dinamik sistemler kuralı gereği kaotik sistemler ayrık işaretler olarak incelenirse ayrık zamanlı, sürekli işaretler olarak incelenirse sürekli zamanlı dinamik sistemler olarak adlandırılmaktadır. Sürekli zamanlı kaotik sistemler diferansiyel denklemler kümesinden oluşmaktadır [106]. Ayrık zamanlı kaotik sistemler tek boyutlu yani bir denklemden oluşabilirken, sürekli zamanlı kaotik sistemler en az üç denklem içeren üç boyutlu kaotik ifadelerden oluşmaktadır.

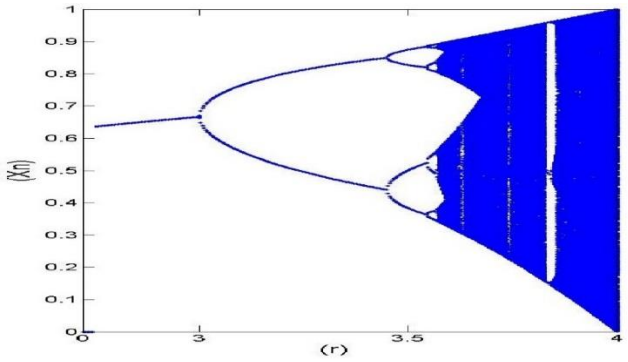
3.1.1. Ayrık zamanlı kaotik sistemler

En az üç denklemden meydana gelen sürekli kaotik sistemlerin aksine ayrık zamanlı kaotik sistemler tek denklemden oluşabilmektedir. Literatürde ayrık zamanlı kaotik sistemler genellikle tek boyutlu ve iki boyutlu olsada üç boyutlu ayrık zamanlı kaotik sistemlerde bulunmaktadır. Ayrık zamanlı kaotik sistemler dijital ortamlarda sürekli zamanlı kaotik sistemler gibi ayrıklaştırma algoritmaları ile işlemlere tabi tutulmadan doğrudan istenen sayısal uygulamalarda kullanılabilmektedir.

Literatürde yüksek seviyede etkili ve basit yapıda olan tek boyutlu birçok ayrık zamanlı kaotik sistemler bulunmaktadır [107]. Bunlardan bazıları Logistic Map, Cubic Map, Linear

Congruential Generator Map, Sine Map, Tent Map, Rickers's Population Model Map, Gauss Map, Cusp Map, Gaussian White Chaotic Map, Pinchers Map, Spence Map and Sin-Circle Map tek boyutlu ayrık zamanlı kaotik sistemleridir.

Logistic Map literatürde en çok kullanılan tek boyutlu kaos üreteçlerinden birisidir [51]. Lojistic Map; kuşlar, memeliler gibi biyolojik nüfus modelinin en basit modeli olan denklemin ayrık halidir [106]. Şekil 3.1'deki çatallaşma diyagramı dikkate alınarak Logistic Map'in kaotik olma durumu incelenmiştir. Denklem 1'de verilen kontrol parametresi olan r, 0-4 değerleri arası incelenmiştir. Şekil 3.1'de r parametresi, 0-3 arası aldığı değerler de sadece bir, 3-3.4 arası değerlerde iki, 3.5 civarı değerlerde dört, kaosa girmesini sağlayan yakın değerlerde sekiz sonuç üretmektedir. 3.5699 dan büyük değerler için ise sistemin kaosa girdiği görülmektedir. Denklemdeki r değerinin 0 ile 3.5699 arası olduğu durumlarda sistem pozitif Lyapunov üsteline sahip değildir. Lyapunov üstelinin pozitif değer olmadığından dolayı sistem kaotik davranış sergileyemez [108].



Şekil 3.1. Logistic Map için çatallaşma diyagramı

$$x_{n+1} = r x_n (1 - x_n) \tag{3.1}$$

Denklem (3.1)'deki x değeri sistem değişkeni, n ise yineleme sayısıdır. Sistem için x(0) başlangıç değeri, r değeri ise sistem parametresi olarak kabul edilmektedir [5].

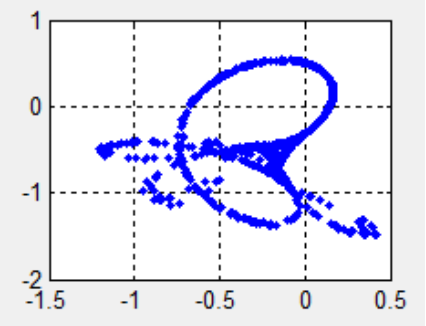
İki boyutlu kaotik sistemler iki ayrı denklemden meydana gelmektedir. Literatürde kullanılan bazı iki boyutlu kaos üreteçleri Henon Map, Lozi Map, Delayed Logistic Map, Tinkerball Map, Burgers Map, Holmes Cubic Map, Kaplan Yorke Map, Dissipative

Standard Map, Ikeda Map, Sinai Map, Discrete Predator Prey Map, Chirikov (Standard) Map, Henon Area-Preserving Quadratic Map, Arnold's Cat Map, Gingerbreadman Map and Chaotic Web Map olarak verilebilir. Örnek olarak Tinkerbell Map incelenecek olursa;

Tinkerbell Map iki boyutlu ayrık zamanlı kaotik sistemi aşağıda Denklem (3.2)'de verildiği gibidir.

$$\begin{aligned} x_{n+1} &= x_n^{\,2} - y_n^{\,2} + ax_n + by_n \\ y_{n+1} &= 2x_n y_n + cx_n + dy_n \end{aligned} \tag{3.2}$$

Denklemlerde parametreler için a=0.9, b=-0.6, c=2, d=0.5, x0 = 0, y0 = 0.5 başlangıç şartları için elde edilen x-y kaotik çekicisi Şekil 3.2'de verildiği gibidir.

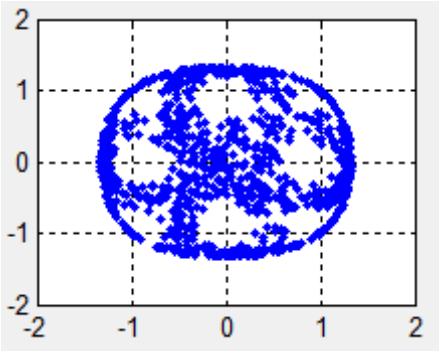


Şekil 3.2. Tinkerbell Map x-y kaotik çekicisi

Üç boyutlu ayrık zamanlı kaotik sistemler içerisinde en yaygın olarak bilinen Lorenz Chaotic Map 'dir. Bu kaotik sistemde üç ayrı denklem bulunmaktadır. Lorenz Chaotic Map kaotik sistemini Denklem (3.3)'de görüldüğü gibidir.

$$\begin{aligned} x_{n+1} &= x_n y_n - z_n \\ y_{n+1} &= x_n \\ z_{n+1} &= y_n \end{aligned} \tag{3.3}$$

Denklemlerde, x0 = 0.5, y0 = 0.5 ve z0 = -1 başlangıç şartları için elde edilen x-y kaotik çekicisi Şekil 3.3'de verildiği gibidir.



Şekil 3.3. Lorenz Chaotic Map x-y kaotik çekicisi

3.1.2. Sürekli zamanlı kaotik sistemler

Sürekli zamanlı kaotik sistemler genellikle adi diferansiyel denklemler ile ifade edilmektedir. Sürekli zamanlı *n* tane birinci dereceden adi diferansiyel denklem sistemi *i=1, 2, 3, ...,n* olmak üzere Denklem (3.4) ile verilebilir [109].

$$\left.\begin{aligned} dx^{(i)}/dt &= f_1\left(x^{(i)}, x^{(i+1)}, \cdots\cdots x^{(n)}\right) \\ dx^{(i+1)}/dt &= f_2\left(x^{(i)}, x^{(i+1)}, \cdots\cdots x^{(n)}\right) \\ &\vdots \\ dx^{(n)}/dt &= f_n\left(x^{(i)}, x^{(i+1)}, \cdots\cdots x^{(n)}\right) \end{aligned}\right\} \tag{3.4}$$

Yukarıdaki ifadeler düzenlenirse adi diferansiyel denklemler vektörel formda aşağıda Denklem (3.5)'deki gibi verilebilir.

$$\begin{aligned} dx(t)/dt &= F[x(t)] \\ x(t_0) &= x_0 \end{aligned} \tag{3.5}$$

Denklemde verilen *x*, *n* boyutlu bir vektördür. Ayrıca $x \in R^m$ durum vektörüdür. x_0 başlangıç durum vektörüdür. *t* ise zamanı ifade etmektedir.

Ayrık zamanlı kaotik sistemler doğrusal olmayan tek boyutlu olarak basit denklemlerle ifade edilebiliyordu. Fakat sürekli zamanlı kaotik sistemler en az 3 boyutludurlar. Yani *n* ifadesi

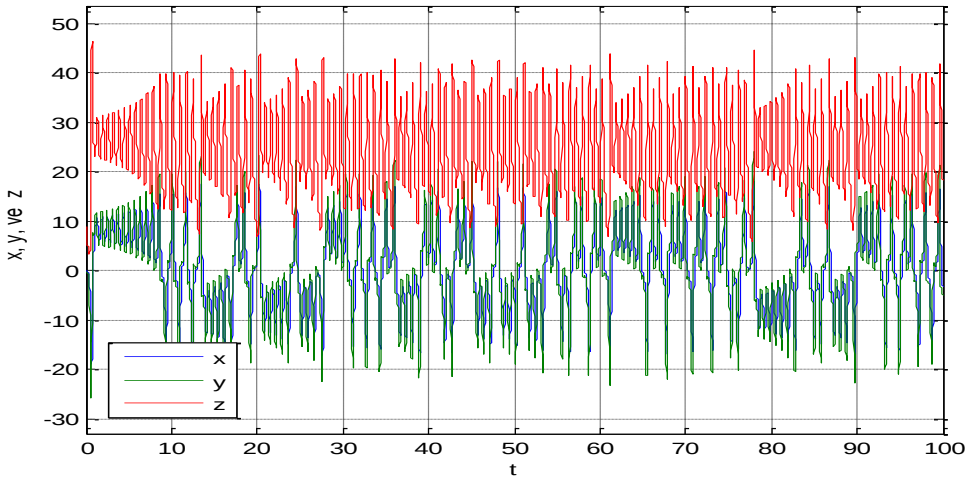
en az üç olmalıdır. Üç üzeri olan sürekli zamanlı kaotik sistemler hiperkaotik sistemler olarak ifade edilmektedir.

Günümüze kadar literatüre sunulmuş birçok sürekli zamanlı kaotik ve hiperkaotik sistemler bulunmaktadır [1,4]. Lorenz, Rössler, Duffing, Chua, Van Der Pol, Chen, Yayınımsız Lorenz, Rikikate, Rucklidge, Üç Katmanlı, Arneodo, Hindmarsh-Rose, Genelleştirilmiş Lotka-Volterra , Moore-Spiegel , Rabinovich-Fabrikant, Sprott(1994) sistemleri sürekli zamanlı kaotik sistemlere örnek olarak verilebilir.

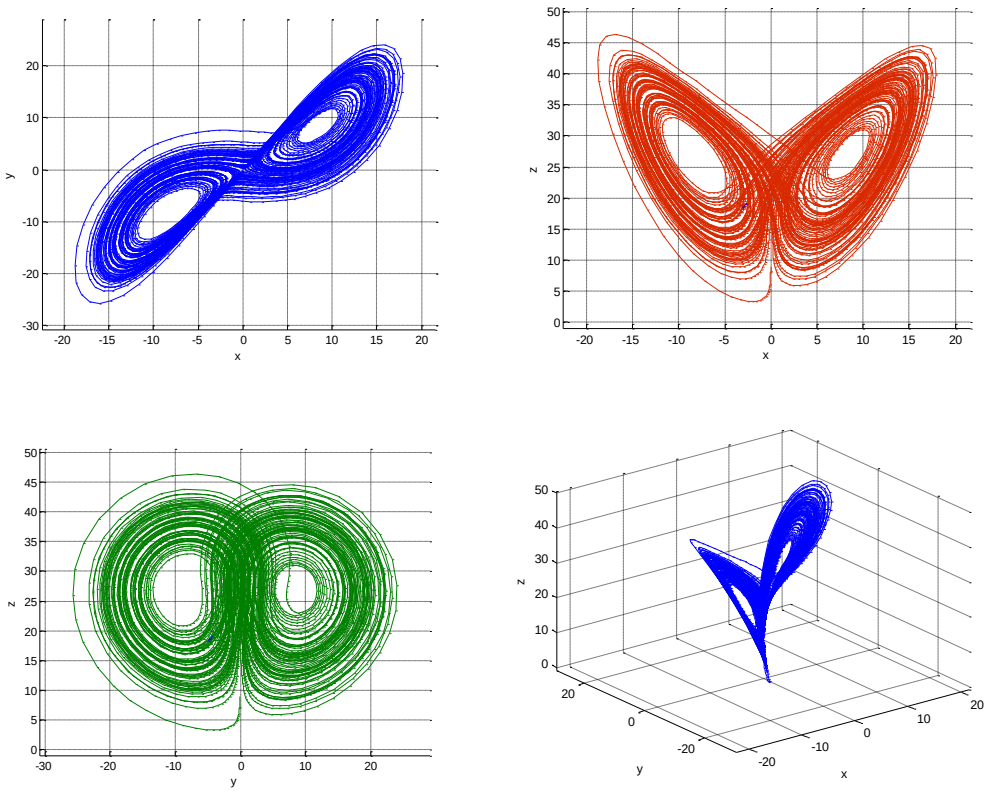
Sürekli zamanlı kaotik sistemlerden bazılarını inceleyecek olursak, ilk bulunan ve literatürde yaygın olarak kullanılan kaotik sistem Lorenz sistemidir. Denklem (3.6)'da Lorenz sisteminin diferansiyel denklemleri verilmektedir. Lorenz sisteminin diferansiyel denklem takımlarındaki α, r ve β parametreleri sistem parametreleri, $x^{(1)}$, $x^{(2)}$ ve $x^{(3)}$ ise sistemin dinamik değişkenleridir. Sistem parametreleri ve sistemin dinamik değişkenleri sistem davranışında çok önemli rol oynamaktadır. Parametrelerdeki ve başlangıç değerlerindeki çok küçük değişmeler büyük sonuçlara yol açabilmektedir.

$$\begin{aligned} dx^{(1)}/dt &= \alpha\,(x^{(2)} - x^{(1)}) \\ dx^{(2)}/dt &= -x^{(1)}\,x^{(3)} + r\cdot x^{(1)} - x^{(2)} \\ dx^{(3)}/dt &= x^{(1)}\,x^{(2)} - \beta\,x^{(3)} \end{aligned} \qquad (3.6)$$

Denklem (3.6) 'da verilen Lorenz sistemi; iki adet ikinci dereceden doğrusal olmayan (xy ve xz ile beraber) toplam yedi terim içermektedir. Sistemin kaotik olabilmesi için parametrelerden $\alpha = 10$, $r=28$ ve $\beta = 2.66$, başlangıç şartlarının ise $x^{(1)} = 0$, $x^{(2)} = -0.1$ ve $x^{(3)} = 9$ olmalıdır. Lorenz sistemi için verilen parametreler ve başlangıç değerlerine göre Matlab odesolve [110] programı ile elde edilen zaman seri ve faz portreleri Şekil 3.4 ve 3.5'de görüldüğü gibidir [106].



Şekil 3.4. Lorenz kaotik sisteminin zaman serisi



Şekil 3.5. Lorenz kaotik sisteminin x-y, x-z, y-z için faz portreleri

3.2. Kaotik sistem analiz yöntemleri

Bir sistemin kaotik olup olmadığını anlamak için kullanılan birçok analiz yöntemleri vardır. Kaotik bir sistemin belirli bir zaman içerisinde nasıl bir davranış gösterdiği (zaman serileri), kaotik çekicileri (faz portreleri), Lyapunov Üstelleri, Çatallaşma Diyagramları, Poincaré Kesiti, sistemin denge noktalarını tespit etmek bunlardan sadece bazılarıdır. Bu yöntemlerden birkaçına bakılarak sistemin kaotik olup olmadığına karar verilebilir.

3.2.1. Denge nokta analizi

Kaotik sistemler basit fonksiyonlarla ifade edilemediğinden, bu sistemler hakkında doğrudan çıkarım yapmak mümkün değildir. Bu yüzden doğrusal olmayan dinamik sistem olan kaotik sistemlerin davranışını anlamak için sistemin denge noktalarını, yani $F[x(t)]=0$ durumundaki denge noktalarının bulunarak analiz edilmesi gerekmektedir [106].

Denge nokta analizinde; kaotik sistem analizi için öncelikle sistem denge noktaları bulunur. Denge noktalarını bulmak için denklemler sıfıra eşitlenir ve çözümlenir. Denklemlerin çözümü sonucunda bulunan ifadeler reel sayılar ise sistemin denge noktaları mevcuttur denilebilir. Bazı kaotik sistemler sadece sanal denge noktalarına sahip olduklarından reel denge noktaları yoktur. Bu sistemler denge noktasız kaotik sistemler olarak isimlendirilir. Bir sonraki aşamada ise denge noktaları reel veya sanal olarak bulunduktan sonra sistemin Jacobian matrisine denge noktalarında bulunan ifadeler yazılır. Daha sonra $|J-\lambda I|=0$ karakteristik denge çözümünden özdeğerler bulunur.

Denge noktarındaki kararsızlık durumunu yani kaotik olup olmadığını özdeğerlerden anlaşılabilir. Bulunan özdeğerlerden en az birinin reel kısmının pozitif olması, denge noktasının kararsızlığına yani sistemin kaotik olduğuna işaret eder.

Örnek olarak, Denklem (3.7)'deki gibi bir denklem kümemiz olsun.

$$\begin{aligned} \dot{x} &= y - x \\ \dot{y} &= ay - xz \\ \dot{z} &= xy - a \end{aligned} \tag{3.7}$$

Verilen denklem kümesinin kaotik olup olmadığını anlamak için öncelikle $\dot{x}, \dot{y}, \dot{z} = 0$ yapılırsa,

$$\begin{aligned} 0 &= y^* - x^* \\ 0 &= ay^* - x^* z^* \\ 0 &= x^* y^* - a \end{aligned} \tag{3.8}$$

Denklem (3.8)'deki ifadeler elde edilir. x^*, y^*, z^* için denklem kümesi çözülürse denge noktaları,

$(x^*, y^*, z^*) = (\pm\sqrt{a}, \pm\sqrt{a}, a)$ olarak bulunur.

Sistemin Jacobian matrisi aşağıdaki gibidir.

$$J = \begin{bmatrix} -1 & 1 & 0 \\ -z & a & -x \\ y & x & 0 \end{bmatrix}$$

Bulunan denge noktaları sistemin Jacobian matrisinde yerine yazılırsa,

$$J = \begin{bmatrix} -1 & 1 & 0 \\ -a & a & -\sqrt{a} \\ \sqrt{a} & \sqrt{a} & 0 \end{bmatrix}$$

elde edilir.

Daha sonra ise $| J - \lambda I | = 0$ karakteristik denge nokta çözümünden özdeğerler bulunur.

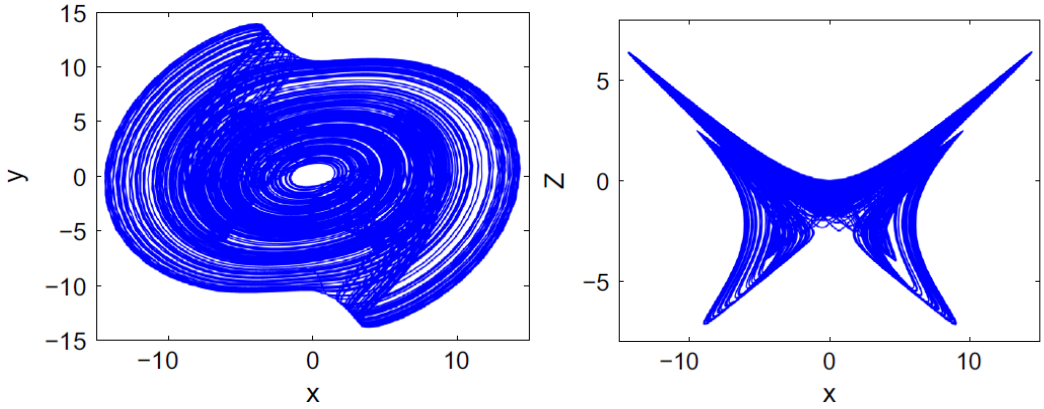
Sonuç olarak özdeğerler,

$\lambda_1 = -1, \lambda_2 = -0.968i - 0.25, \lambda_3 = 0.968i + 0.25$ olarak elde edilir.

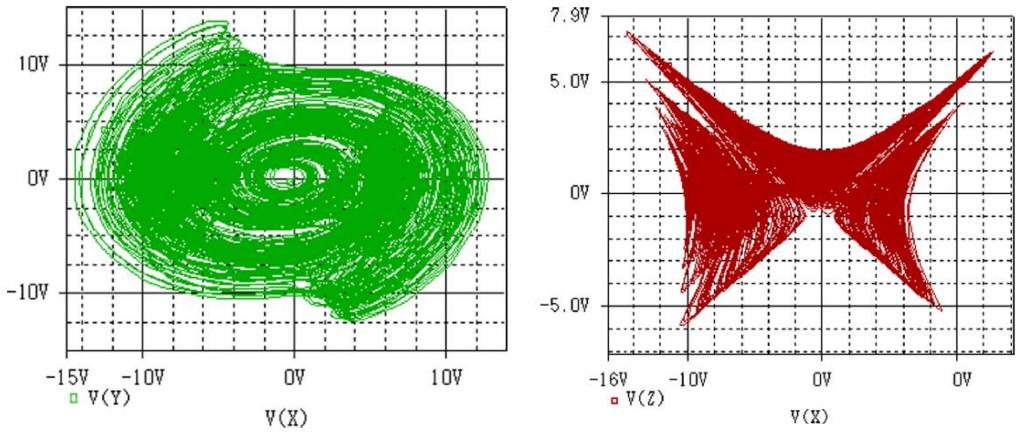
Bulunan özdeğerlerin reel kısmının en az birisi pozitif olduğu için verilen sistem kararsız yani kaotik özellik göstermektedir. Sistemin kaotik özellik sergilediğinden emin olmak için sonraki kısımlarda anlatılan diğer analizlerin yapılmasında da fayda vardır.

3.2.2. Faz portreleri

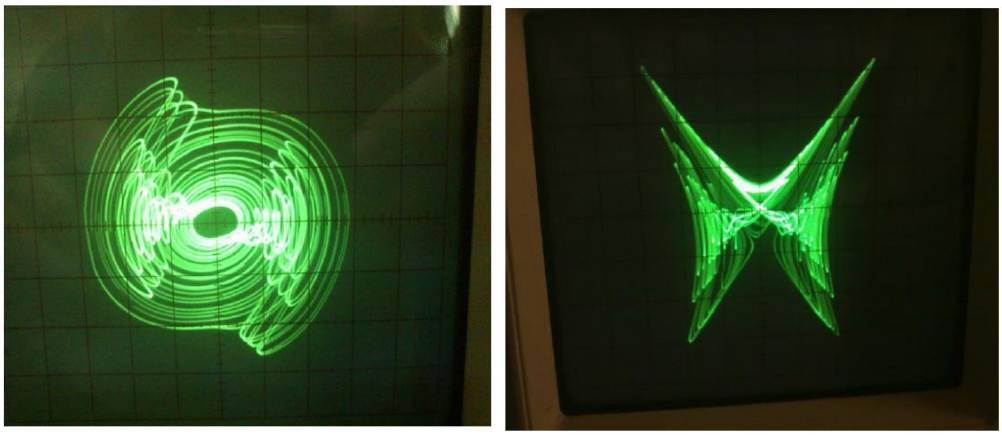
Üç boyutlu bir sistem için x-y, x-z, y-z ve x-y-z olarak dört farklı şekilde bir sistemin kaotik çekicileri yani faz portrelerine bakılabilir. Bu işlemler gelişen teknoloji sayesinde bilgisayarlar aracılığı ile kolaylıkla yapılabilmektedir. Matlab "odesolve.m" programı ile kaotik sistem verileri girilerek program çıktısında istenen faz portreleri kolaylıkla elde edilebilmektedir. Aynı işlemler Matlab Simulink, elektronik devre gerçekleme simulasyon programları veya yapılan elektronik devrelerden osilaskop çıktıları olarak elde edilebilmektedir. Şekil 3.6, 3.7 ve 3.8'de sırasıyla Matlab odesolve, PSpice ve osilaskop ile örnek faz portre çıktıları görülmektedir [111].



Şekil 3.6. Örnek Matlab faz portre görünümü



Şekil 3.7. Örnek PSpice çıkışlarının faz portre görünümü



Şekil 3.8. Örnek gerçek devre osilasop çıkışlarının faz portre görünümü

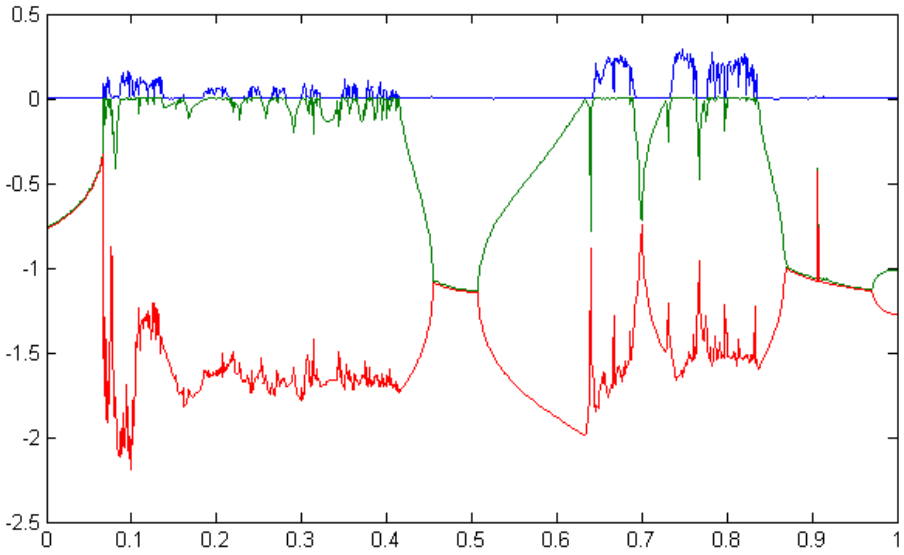
3.2.3. Lyapunov üstelleri

Lyapunov üstelleri yöntemi Aleksandr Mikhailovich Lyapunov tarafından bulunmuştur. Bu yöntem bir zaman serisinin kaotik bileşenler içerip içermediğini tespit eden bir analiz yöntemidir. Lyapunov üsteli, kısaca kaotik sistemlerdeki başlangıç şartlarına hassas bağımlılık özelliğinin sayısal olarak ifadesidir denilebilir [112].

Lyapunov üsteli λ, negatif ise farklı başlangıç şartları aynı çıkış verebilir anlamına gelmektedir ve dolayısıyla analiz edilen sistem kaotik değildir denilebilir, fakat Lyapunov üsteli λ pozitif olduğunda farklı başlangıç değerleri farklı çıkış değerleri verdiğinden dolayı sistem kaotik özellik gösterir denilebilir. Bir sistem kaç boyutlu olursa olsun en az bir pozitif

Lyapunov üsteli içeriyorsa sistem kaotik olarak tanımlanabilir [106]. Örneğin üç boyutlu bir sistemde Lyapunov üstelleri λ1, λ2 ve λ3 olsun. Bu sistemin kaotik olabilmesi için Lyapunov üstellerinden birisinin pozitif olması yeterlidir. Üç boyutlu kaotik bir sistemde Lyapunov üstelleri (λ1, λ2 ve λ3); (0,0,0) ise iki-torus, (-,-,-) ise sabit nokta, (0,-,-) ise limit döngü, (0,0,-) ise simit, (+,0,-) ise sistem kaotiktir [112]. Bu durumda üç boyutlu bir sistemde Lyapunov üstelleri için karşılaşılabilecek tek durum (+,0,-)'dır.

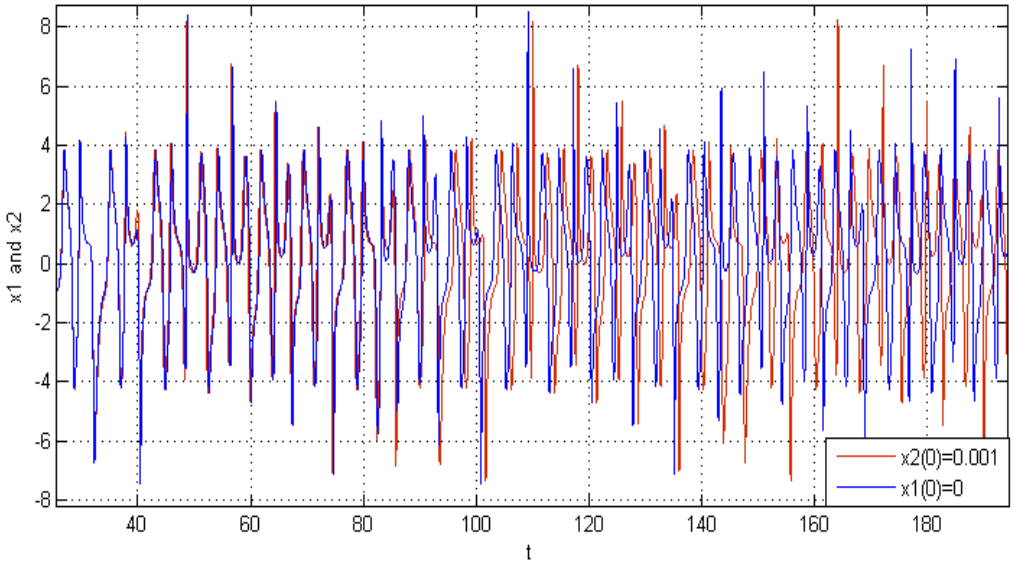
Şekil 3.9'da örnek bir Lyapunov üstel spektrum grafiği verilmektedir. Sistemin kaotik olabilmesi için Lyapunov üstellerinin (+,0,-) olması gerekmektedir. Şekil 3.9'da belli aralıklarla sistemin kaosa girip çıktığı görülmektedir.



Şekil 3.9. Örnek lyapunov üstel spektrum grafiği

3.2.4. Zaman serisinde başlangıç değerlerine hassas bağımlılık analizi

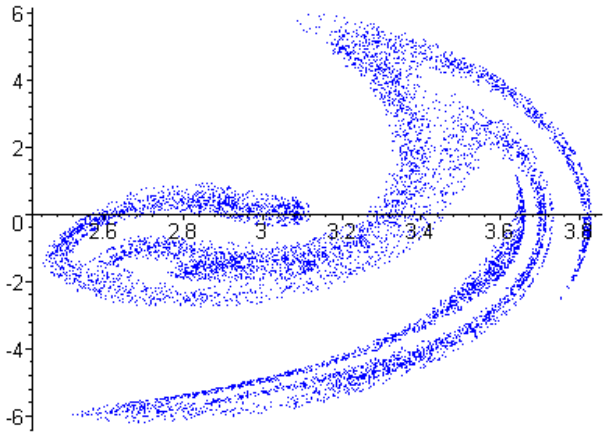
Bir sistemin kaotik olabilmesi için başlangıç şartlarına çok hassas bağımlı olması gerekmektedir. Yani sisteme verilen farklı başlangıç değerlerinin belirli bir zaman içerisinde farklı kaotik işaretler üretmesidir. Farklı başlangıç değerleri ile kaotik işaretleri en iyi gözlemleme yollarından birisi, sistemin zaman serilerini aynı ekranda incelemektir.



Şekil 3.10. Kaotik sistemlerin başlangıç şartlarına hassas bağlılığına bir örnek

3.2.5. Poincoré kesiti

Bazı kaotik sistemlerin faz portreleri çok karışık olabilir ve dolayısıyla gözlemlemek sıkıntı olabilmektedir. Poincoré kesiti yöntemiyle bu sıkıntı giderilebilmektedir. Çok karmaşık olmayan sistemler içinde bu yöntem kullanılabilir fakat, özellikle karmaşık faz portreli sistemler için kolaylık sağlamaktadır. Bu yöntem ile kaotik bir sistemin faz portresinin herhangi bir noktasından geçen kesitler alınarak, bu kesitler yorumlanıp sistemin kaotik olup olmadığına karar verilebilir. Poincoré kesitindeki noktaların dağılımı sonucu sistemin kaotik olabilmesi için noktaların belirli alanlarda yoğunlaşmış kümeler halinde olması gerekmektedir. Aksi halde sistem ya periyodik ya da yarı periyodiktir denilebilir [112]. Şekil 3.11'de örnek bir Poincore kesiti görülmektedir [113].



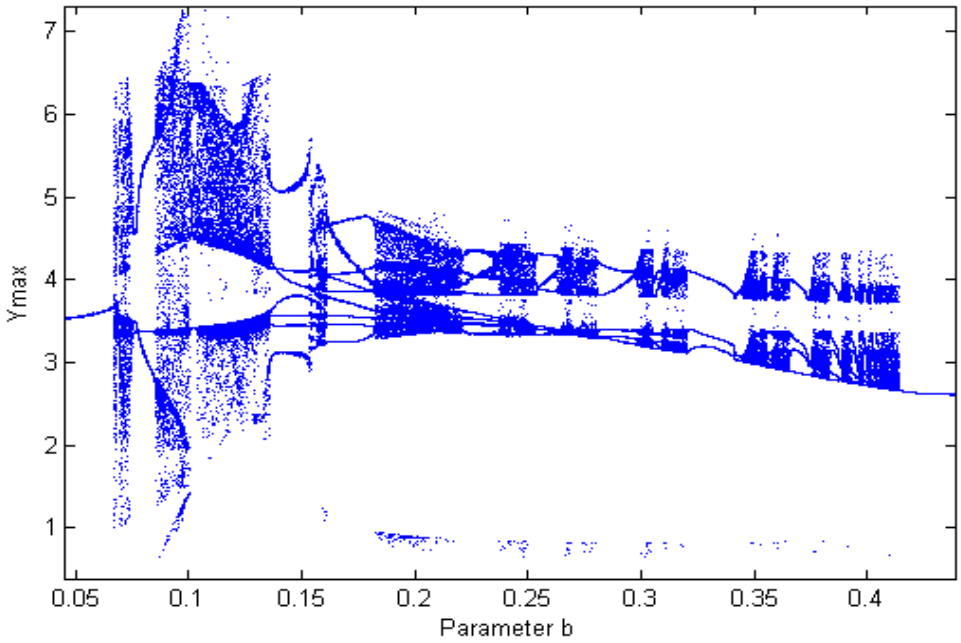
Şekil 3.11. Poincore kesit örneği

Bardak veya simitin herhangi bir kısmından alınan Poincore kesiti daire veya elips şeklinde olduğu için, Poincoré kesitlerine bakılara kaotik özellik göstermedikleri sonucuna varılabilir.

3.2.6. Çatallaşma diyagramı

Bir sistemdeki parametreler değiştiği zaman çatallaşmalar meydana gelmektedir. Bu parametrelerdeki çok küçük değişimlerin, faz uzayındaki yapısal değişimleri ise çatallaşma olarak tanımlanmaktadır. Çatallaşma diyagramı çizdirilen bir sistem incelenerek kaotik bir sistem olup olmadığına veya hangi aralıklarda kaotik özellik gösterdiğine karar verilebilir [106].

Şekil 3.12'de sistem parametresi olan b değerine göre çatallaşma diyagramı incelenmiştir. Şekildeki çatallaşma diyagramı dikkate alınarak sistemin kaotik olduğu durumlar görülebilmektedir. İlk olarak b parametresi yaklaşık 0.07' ye kadar sistem tek bir değer ürettiği için kaotik özellik göstermemektedir. Yaklaşık 0.07 den sonra üretilen değerler artmakta ve sistemin kaosa girdiği görülmektedir. Çatallaşma diyagramından görüldüğü üzere sistem belli aralıklarla kaosa girip çıkmaktadır



Şekil 3.12. Çatallaşma diyagram örneği

3.3. Kaotik Sistemlerin Modellenmesi ve Elektronik Devre Gerçeklemeleri

Doğrusal veya doğrusal olmayan tüm sistemlerin sayısal ortamlar, gerçek ortam uygulamaları vb. yerlerde kullanılabilmeleri için modellenmesi gerekmektedir. Örneğin verilen sürekli zamanlı bir kaotik sistemin denklemleri tek başlarına uygulama, simulasyon, elektronik devre gerçeklemeleri vb. için birşey ifade etmiyor olabilir. Fakat denklemler matematiksel olarak modellenir ise elde edilen model sistemler istenen yerde kullanılarak, modellenen denklemler için analizler, gözlemlemeler yapılabilir.

Sürekli zamanlı kaotik sistemlerde var olan diferansiyel denklemleri modellemek için toplama, çarpma, tersleme, integral alma gibi işlemler kullanılabilir. Tüm bu işlemleri sayısal ortamlarda blok diyagram ve elektronik uygulamalar için kullanmak mümkündür. Elektronik devre gerçeklemeleri için benzer olarak analog çarpma entegreleri, opamplar, integral alma devresi, toplama devresi, eviren ve evirmeyen yükselteç devreleri gibi temel işlemler kullanılmaktadır. Aynı denklemler için modelleme sonuçlarında elde edilen simulasyon çıktıları ve elektronik devrelerdeki osilaskop çıktı sonuçları aynı olacaktır.

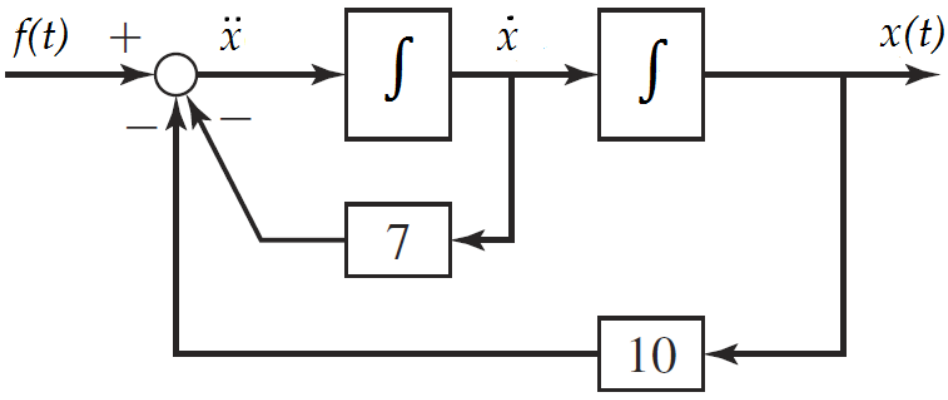
Örneğin Denklem (3.9)'daki gibi verilen bir diferansiyel denklemin blok diyagramını elde etmek için en yüksek mertebeli terimin (x'in ikinci türevi) yalnız bırakılması gerekmektedir.

$$\ddot{x} + 7\dot{x} + 10x = f(t) \tag{3.9}$$

En yüksek mertebeli x terimi yalnız bırakılarak diğer terimler karşı tarafa atılırsa aşağıdaki Denklem (3.10) elde edilmiş olur.

$$\ddot{x} = f(t) - 7\dot{x} - 10x \tag{3.10}$$

Denklem 3.10'daki ifadeyi gösteren blok diyagram Şekil 3.13'de verilmiştir. Bu şekilde bir diferansiyel denklemden blok diyagramın elde edilmesi, denklemin analog ortamlarda ve başka platformlarda kullanılmasına olanak sağlamaktadır. Matlab-Simulink gibi programlarla veya elektronik gerçeklemelerle blok diyagram sonucu istenen yerlerden çıkış alınarak istenen grafikler, analiz sonuçları gibi veriler elde edilebilir. Kaotik sistemlerdeki denklemlerde benzer şekilde modellenerek blok diyagramları çıkarılabilir ve elde edilen blok diyagram yerlerine elektronik temel işlemsel elemanlar, devreler konularak simulasyonlar ve gerçek ortam uygulamaları yapılabilir.

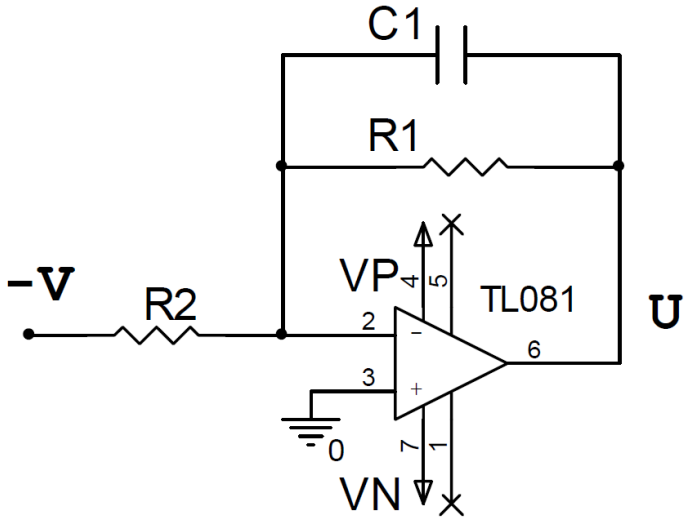


Şekil 3.13. Denklem 3.11'i modelleyen blok diyagram

Elektronik devre analizi için Denklem (3.11)'de verilen Lorenz sisteminin üç denkleminden biri olan birinci denklemi inceleyelim.

$$\dot{u} = \sigma(v - u) \tag{3.11}$$

Verilen denklem için elektronik devre tasarımı Şekil 3.14'deki gibi olmaktadır. Verilen denklemde türev işlemi olduğu için modellenen devrede bir integral devresi bulunmaktadır. Ayrıca katsayıları ayarlamak için ise dirençler kullanılmaktadır [106].



Şekil 3.14. Lorenz devre tasarımının u hesaplama devresi

Şekil 3.14'ü analiz edersek;

$u = -v\dfrac{(-1/\ jwC_1)\,//\,R_1}{R_2}$ denkleminden

$u = v\dfrac{R_1}{R_2}\dfrac{1}{1+R_1 jwC_1}$ sonucu elde edilir. Buradan da,

$ujwR_1C_1 + u = v\dfrac{R_1}{R_2}$, yani $\dot{u}R_1C_1 = v\dfrac{R_1}{R_2} - u$ olur.

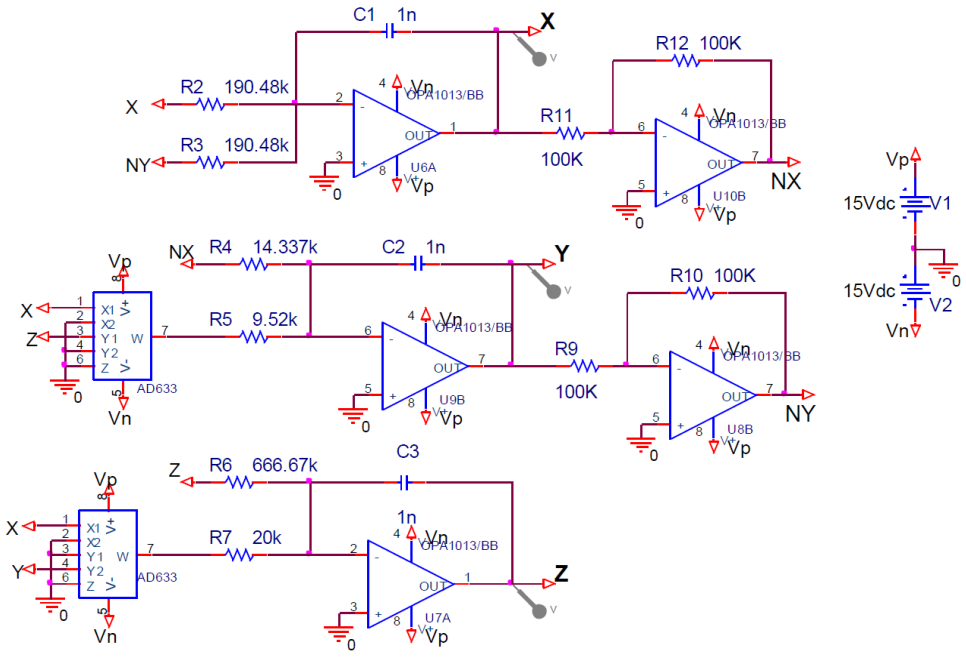
R1=R2 alınıp yerlerine R yazılırsa,

$\dot{u} = \frac{1}{RC_1}(v-u)$ sonucu elde edilmiş olur.

Verilen bir diferansiyel denklem kümesinin Matlab, PSpice ve gerçek ortamda çıkış grafiklerinin, sonuçlarının gözlemlenmesi için Tigan(T) kaotik sistemi ele alınmıştır. Tigan(T) kaotik sisteminin denklemleri Denklem (3.12)'de verilmiştir. Bu sistemde 3 farklı diferansiyel denklem bulunmaktadır. Verilen denklemler Matlab, PSPice gibi simulasyon programları ve gerçek ortamlar için analiz edilip tasarım yapılarak incelenebilir.

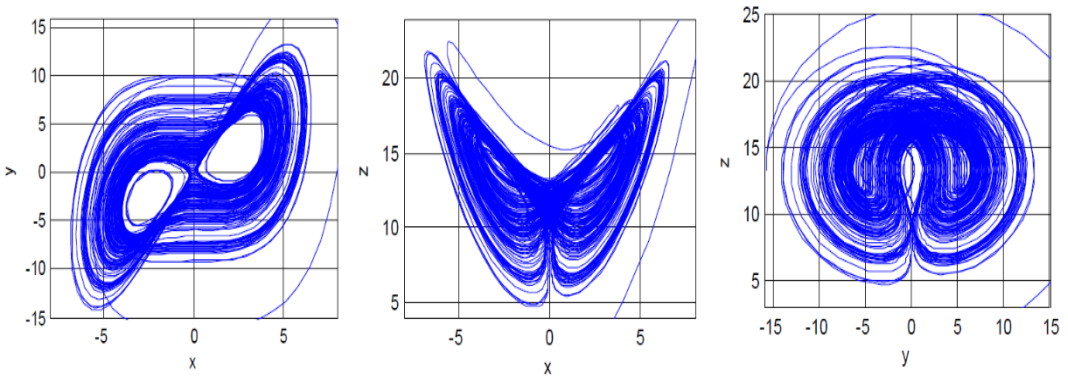
$$\begin{aligned} \dot{x} &= a(y-x) \\ \dot{y} &= (c-a)x - axz \\ \dot{z} &= xy - bz \end{aligned} \tag{3.12}$$

Tigan(T) kaotik sisteminin elektronik devre tasasımı Şekil 3.15' de verilmiştir. Şekil 3.14'de (lorenz) denklemlerinden biri ele alınmıştı. Aynı şekilde Tigan (T) kaotik sisteminin denklemleride analiz edilerek aşağıdaki elektronik devre tasarlanabilir [114].

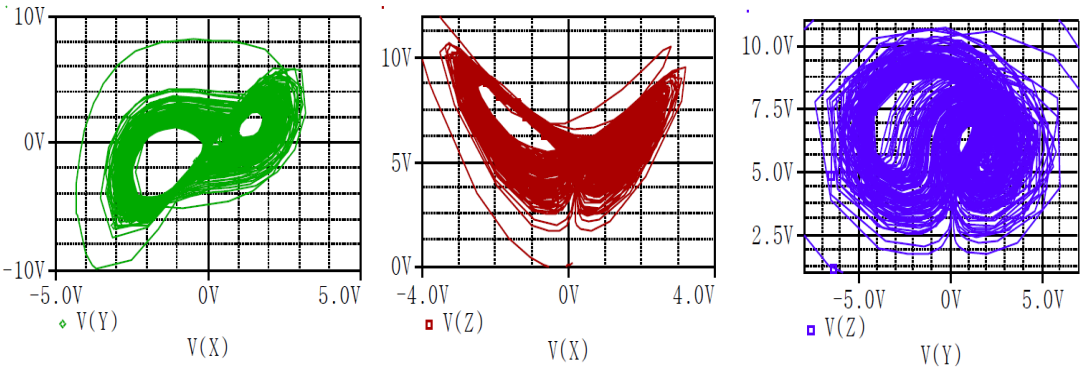


Şekil 3.15. Tigan(T) kaotik sisteminin tasarlanan elektronik devre şeması

Tigan (T) kaotik sistemlerindeki denklemler modellenerek Şekil 3.16, 3.17 ve 3.18'deki gibi sırasıyla Matlab, PSpice ve gerçek ortamdaki osilaskop çıktıları elde edilebilir. Şekillerden tüm analiz sonuçlarında aynı çıktığı görülmektedir [114]. Diğer tüm sistemlerde benzer şekilde modellenerek aşağıdaki gibi sonuçlar gözlemlenebilir.



Şekil 3.16. Tigan(T) kaotik sisteminin numerik Matlab simülasyon sonuçları



Şekil 3.17. Tigan(T) kaotik osilatörünün PSpice simülasyon sonuçları

Şekil 3.18. Gerçekleştirilen Tigan(T) kaotik osilatörünün Osiloskop çıkışları

BÖLÜM 4. YENİ BULUNAN KAOTİK SİSTEMLERİN ANALİZLERİ ve DEVRE GERÇEKLEMELERİ

4.1. Yeni Kaotik 1 Sistemi

Yeni bulunan kaotik sistem 1, sürekli zamanlı 3 boyutlu denge noktasız bir kaotik sistemdir. Sistem Denklem (4.1)'de verildiği gibi 3 ayrı diferansiyel denklemden oluşmaktadır. Sistemde üç adet durum değişkeni (x, y, z), toplam on adet terim ve "a, b, c, d, e, f" olmak üzere altı adet parametre vardır. Sistemin başlangıç şartları x(0)=0, y(0)=0, z(0)=0 iken kaotik özellik göstermektedir. Tüm başlangıç şartlarının "0" olması gerçek ortam uygulamaları için kolaylık sağlamaktadır.

$$\begin{aligned} \dot{x} &= ay - x + zy \\ \dot{y} &= -bxz - cx + yz + d \\ \dot{z} &= e - fxy - x^2 \end{aligned} \tag{4.1}$$

Sistem parametreleri $a = 2.8,\ b = 0.2,\ c = 1.4,\ d = 1,\ e = 10\ ve\ f = 2$ olduğu durumlarda kaotik özellik göstermektedir. Farklı sistem parametreleri içinde farklı kaotik sistemler elde edilebilir. Denklem (4.2)'de kaotik sistemin parametreli hali verilmiştir.

$$\begin{aligned} \dot{x} &= 2.8y - x + zy \\ \dot{y} &= -0.2xz - 1.4x + yz + 1 \\ \dot{z} &= 10 - 2xy - x^2 \end{aligned} \tag{4.2}$$

Yeni bulunan kaotik sistemde çok fazla parametre olması ve denge noktasız veya sanal denge noktalı kaotik sistem olması şifreleme çalışmaları için önemli avantaj sağlamaktadır. Şifreli verilerin çözülebilmesi için tüm parametrelerin bilinmesi gerekmektedir, yani parameter sayısı arttıkça şifreli verilerin çözülmeside zorlaşmış olacaktır. Denge noktası

sistemlerde ise bazı kaotik analiz yöntemlerinin (Shilknov metodu gibi) uygulanamaması, şifreleme işleminin yapıldığı kaotik sistemin bulunmasını zorlaştırmış olacaktır.

4.1.1. Sistem denge nokta analizi

Yeni kaotik sistem 1'de denge noktalarını bulmak için $\dot{x}=0, \dot{y}=0, \dot{z}=0$ olarak ele alınırsa,

$$
\begin{aligned}
0 &= ay - x + zy \\
0 &= -bxz - cx + yz + d \\
0 &= e - fxy - x^2
\end{aligned}
$$

elde edilir. Bu denklem sistemi çözülürse denge noktaları (Denklem 4.3)

$$
\begin{aligned}
&E_1(2.707+0.573i, 0.413-0.661i, -1.584+3.332i) \\
&E_2(2.707-0.573i, 0.413+0.661i, -1.584-3.332i) \\
&E_3(-2.962-0.739i, -0.107+0.766i, -3.215+3.924i) \\
&E_4(-2.962+0.739i, -0.107-0.766i, -3.215-3.924i)
\end{aligned}
\tag{4.3}
$$

olarak bulunur.

Denge noktaları analizi sonucunda; E_1, E_2, E_3 ve E_4 denge noktaları sanal sayılar elde edilmiştir. Kaotik sistem gerçek sayılı denge noktalarına sahip olmadığı için denge noktasız kaotik sistem olarak adlandırılır. Denge noktasız kaotik sistemler ayrıca gizli çekicili (hidden attractor) kaotik sistemler olarakta adlandırılmaktadırlar.

Denge noktalarının kararsız olup olmadığını anlamak için ayrıca sistemin özdeğerlerinin de bulunması gerekmektedir. Özdeğerleri bulmak için öncelikle sistemin Jacobian matrisinin alınması gerekmektedir. Sistemin Jacobian matrisi Denklem (4.4)'de verildiği gibidir.

$$J(x,y,z)=\begin{bmatrix} -1 & a+z & y \\ -bz-c & z & bx+y \\ -fy-2x & -fx & 0 \end{bmatrix} \tag{4.4}$$

E_1 özdeğerleri için; Denklem (4.3)'de bulunan denge noktaları, Denklem (4.4)'deki Jacobian matrisinde yerleri yazılırsa, Denklem (4.5) elde edilir.

$$J(E_1)\begin{bmatrix} -1 & 1.215+3.332i & 0.413-0.661i \\ -1.083-0.666i & -1.584+3.332i & -0.128-0.776i \\ -6.242+0.175i & -5.4151-1.147i & 0 \end{bmatrix} \tag{4.5}$$

Son olarak ise, $\left|\lambda I - J(E_1)\right| = 0$ çözümünden E_1 için karakteristik denklem bulunabilir. E_1 için karakteristik denklem,

$$\lambda^3 + (2.584-3.332i)\lambda^2 + (3.341-7.466i)\lambda - (0.280-26.838i) = 0 \tag{4.6}$$

Karakteristik denklem çözümünden, özdeğerler aşağıdaki gibi bulunmaktadır.

$$\begin{aligned} \lambda_1 &= 1.020+2.941i \\ \lambda_2 &= -1.068-2.149i \\ \lambda_3 &= -2.537+2.540i \end{aligned} \tag{4.7}$$

Sistemin kararsızlığın için özdeğerlerden en az birinin pozitif olması gerekiyordur. λ_1'de pozitif olma koşulu sağlandığı için sistem kararsızdır denilebilir. Dolayısıyla analiz edilen yeni sistem kaotik bir sistemdir. E_2, E_3 ve E_4 denge noktaları E_1 denge noktasında olduğu gibi incelenir ve Jacobian matrislerinde yerlerine yazılırsa özdeğerler,

E_2 için,

$$\begin{aligned}\lambda_1 &= 1.020 - 2.941i \\ \lambda_2 &= -1.068 + 2.149i \\ \lambda_3 &= -2.537 - 2.540i\end{aligned} \tag{4.8}$$

E_3 için,

$$\begin{aligned}\lambda_1 &= 1.289 + 3.078i \\ \lambda_2 &= -1.177 - 1.500i \\ \lambda_3 &= -4.327 + 2.346i\end{aligned} \tag{4.9}$$

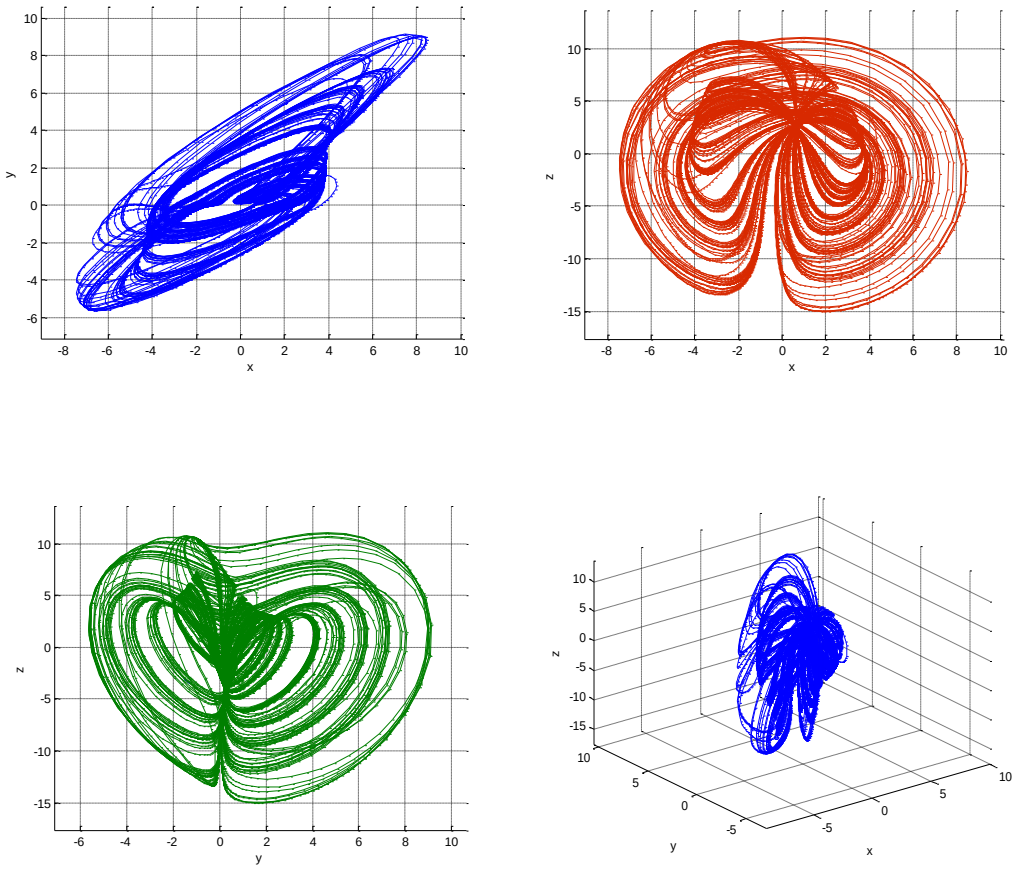
E_4 için,

$$\begin{aligned}\lambda_1 &= 1.289 - 3.078i \\ \lambda_2 &= -1.177 + 1.500i \\ \lambda_3 &= -4.327 - 2.346i\end{aligned} \tag{4.10}$$

olarak elde edilmiş olur. Sonuçlardan da görüldüğü üzere her özdeğer kendi içerisinin gerçel kısmında pozitif bir değer taşımakta ve sistemin kararsızlığını göstermektedir. Sistem kaotik olduğuna kesin olarak karar verebilmek için diğer kaotik sistem analizlerine bakılmasında fayda görülmektedir. Diğer analizler ilerleyen bölümlerde detaylı olarak verilmektedir.

4.1.2. Faz portre analizi

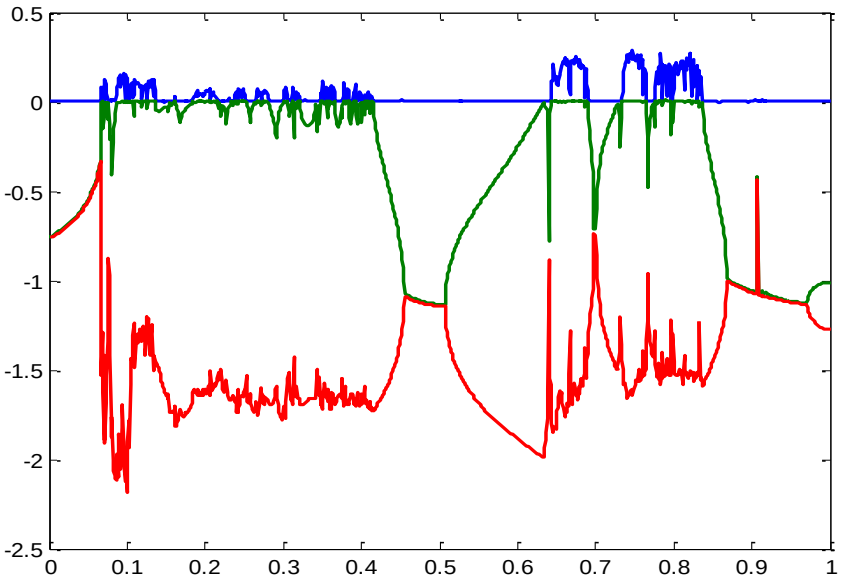
Denge noktasız yeni kaotik sistemin faz portreleri öncelikle Matlab "odesolve.m" programı ile gözlemlenmiştir. Başlangıç değerleri x=0, y=0 ve z=0 için verilen kaotik sistemin x-y, y-z, x-z ve x-y-z olarak faz portre çıktıları Şekil 4.1'de verildiği gibidir. Şekilden de görüldüğü üzere yeni kaotik sistemin zengin dinamik davranışlara sahip olduğu söylenebilir. Yeni kaotik sistem 1 için simulasyon ve gerçek devre uygulama sonucu osilaskop çıktı faz portreleri, elektronik devre uygulama konu kısmı içerisinde verilecektir.



Şekil 4.1. Yeni kaotik 1 için x-y, x-z, y-z ve xy-z için faz portreleri

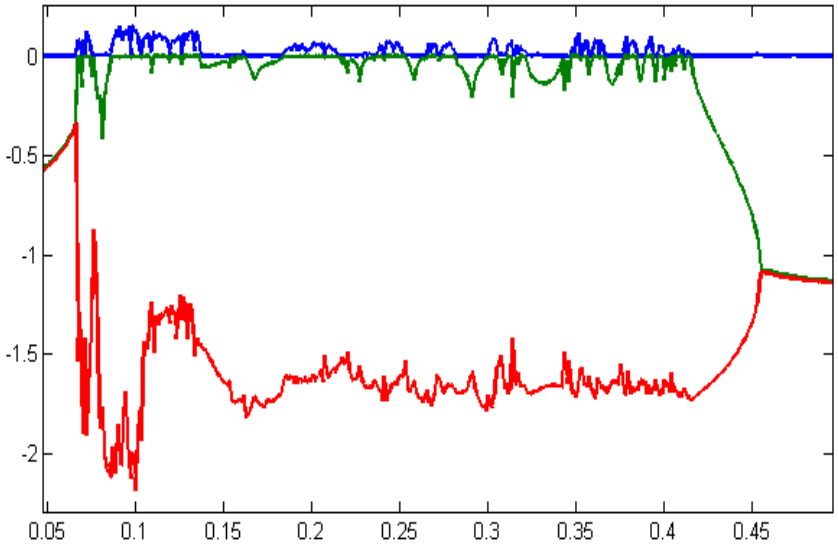
4.1.3. Lyapunov üstel spektrum analizi

Yeni Sistem 1'in kaotik bir sistem olduğu Lyapunov üstellerinin de anlaşılabilmektedir. Yeni kaotik 1 sisteminin, 'b' parametresine göre 0-1 aralığında Lyapunov üstel spektrumu Şekil 4.2'de verildiği gibidir. Lyapunov üstel analizinde sistemin kaotik olması için, değerlerin (+,0,-) olması gerekmekteydi. Şekil 4.2'den de grüldüğü üzere sistem belli aralıklarla kaosa girip çıkmaktadır.



Şekil 4.2. Yeni kaotik 1 sistem için Lyapunov üstel grafiği (b= 0-1)

Örneğin şekildeki mavi, yeşil ve kırmızı çizgiler incelendiğinde 0.05-0.47 aralığında sistem genellikle kaotiktir. Şekil 4.3'de ise sistemin kaotik olduğu aralığı gösteren detaylı Lyapunov üstel grafiği verilmektedir.



Şekil 4.3. Yeni kaotik 1 sistem için detaylı Lyapunov üstel grafiği (b= 0.05-0.47)

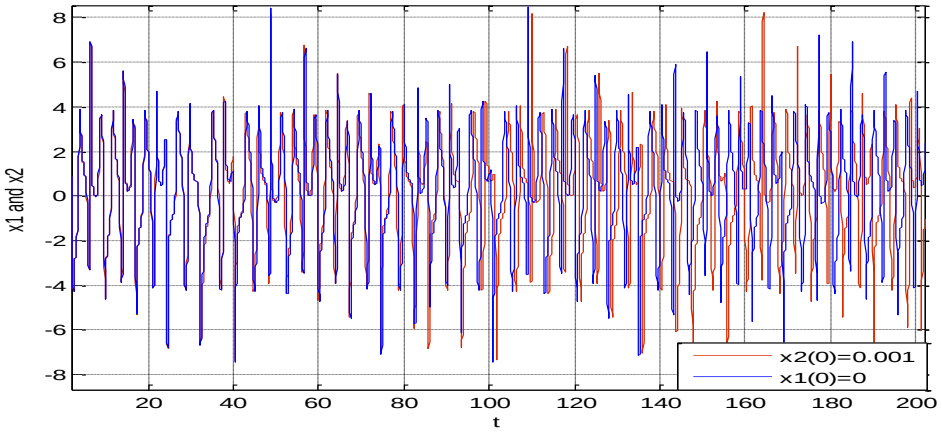
Lyapunov üstel grafiğinden ayrıca sistem boyutuda öğrenilebilmektedir. Örneğin Lyapunov üstel grafiğinin kaotik olan herhangi bir noktasından alınan değerler ile (L_1= 0.1403, L_2=0, L_3= -2.1515) aşağıdaki Denklem (4.11) yardımıyla hesap yapılacak olursa (b = 0.1 için),

$$D_L = j + \frac{1}{|L_j + 1|}\sum_{i=1}^{j} L_i = 2 + \frac{L_1 + L_2}{|L_3|} = 2.0652103183 \qquad (4.11)$$

olarak bulunur. Yani yeni kaotik sistem boyutunun hesap sonucu 2.0652103183 değeri bir üst değeri yuvarlanacak olursa 3 olmuş olur. Sonuç olarak yeni kaotik 1 sistemi 3 boyutlu sürekli zamanlı bir kaotik sistemdir denilebilir.

4.1.4. Zaman serisinde başlangıç değerlerine duyarlılık analizi

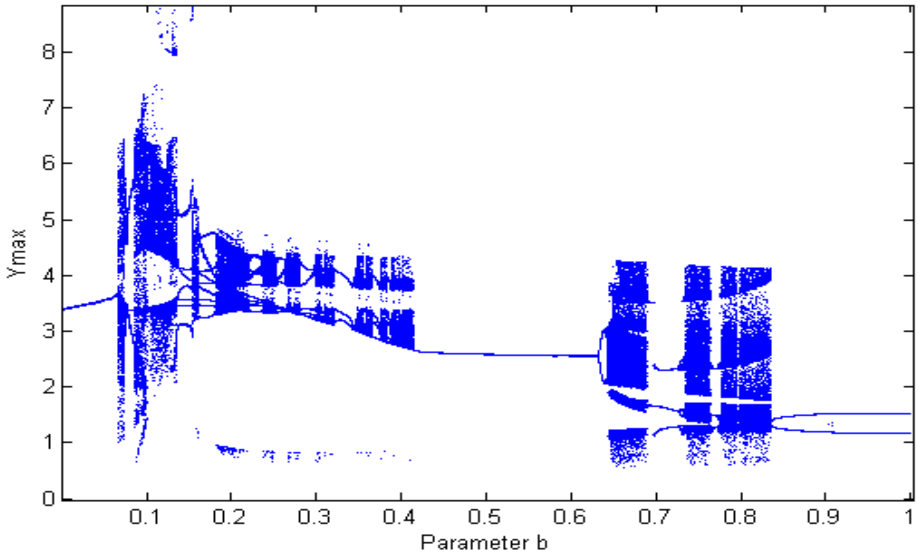
Sistemin başlangıç şart değerlerindeki çok küçük bir değişiklik sonucu farklı çıktılar vermesi kaotiklik hakkında önemli ipuçları vermektedir. Şekil 4.4'de görüldüğü üzere 'x' başlangıç şartı "0" olarak alınmış ve sonucu mavi olan eğri elde edilmiştir. Fakat x 1/1000 değiştirilerek, yani "0.001" yapılarak kırmızı eğri elde edilmiştir. İki eğri Şekil 4.4'de beraber incelendiğinde çok küçük değişimlerin yeni sistem üzerinde farklı sonuçlar verdiği, yani başlangıç şartlarına çok hassas olduğu görülebilmektedir.



Şekil 4.4. x1(0)=0 ve x2(0)=0.001 için zaman seri grafiği

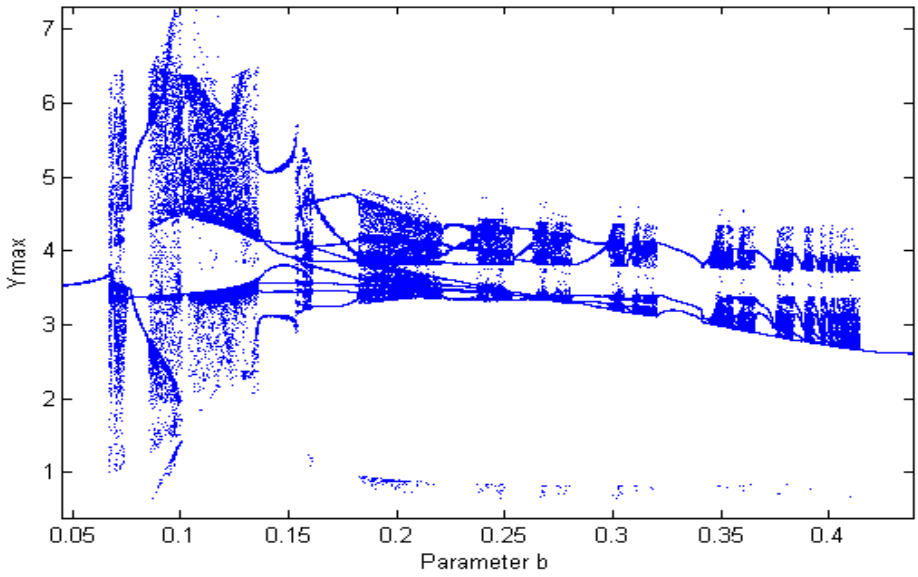
4.1.5. Çatallaşma diyagram analizi

4.1.3. bölümde, b parametresine göre Lyapunov üstel spektrum grafiği verilmşti. Bu bölümde ise yine b parametresine göre çatallaşma diyagramı çizdirilerek analiz yapılmıştır. Lyapunov üstel grafik analiz ve çatallaşma diyagramları aynı aralıklarda, aynı sonuçlar vermelidir, yani Lyapunov üstelinin kaotik özellik gösterdiği yerlerde çatallaşma diyagramı da kaotik özellik göstermelidir. Şekil 4.5'te b parametresi için 0-1 aralığında çatallaşma diyagramı çizdirilmiştir. Şekil 4.5, Şekil 4.2'deki Lyapunov üstel grafiği ile karşılaştırıldığında kaotik özelliklerin gösterildiği yerlerin aynı olduğu görülebilmektedir.



Şekil 4.5. Çatallaşma Diyagramı (b= 0-1)

Şekil 4.6'da, Şekil 4.5'de verilen çatallaşma diyagramının detaylı olarak b parametresi için 0.05-0.47 arasında detaylı gösterimi verilmektedir. Şekil 4.5 ve Şekil 4.6'da görüldüğü üzere çok fazla noktaların üretildiği yerler sistemin kaotik olduğu aralıklardır. Örneğin verilen 'b' parametresinde bir nokta için ne kadar fazla değer üretilirse, kaotiklik oranı o kadar artmış olacaktır.

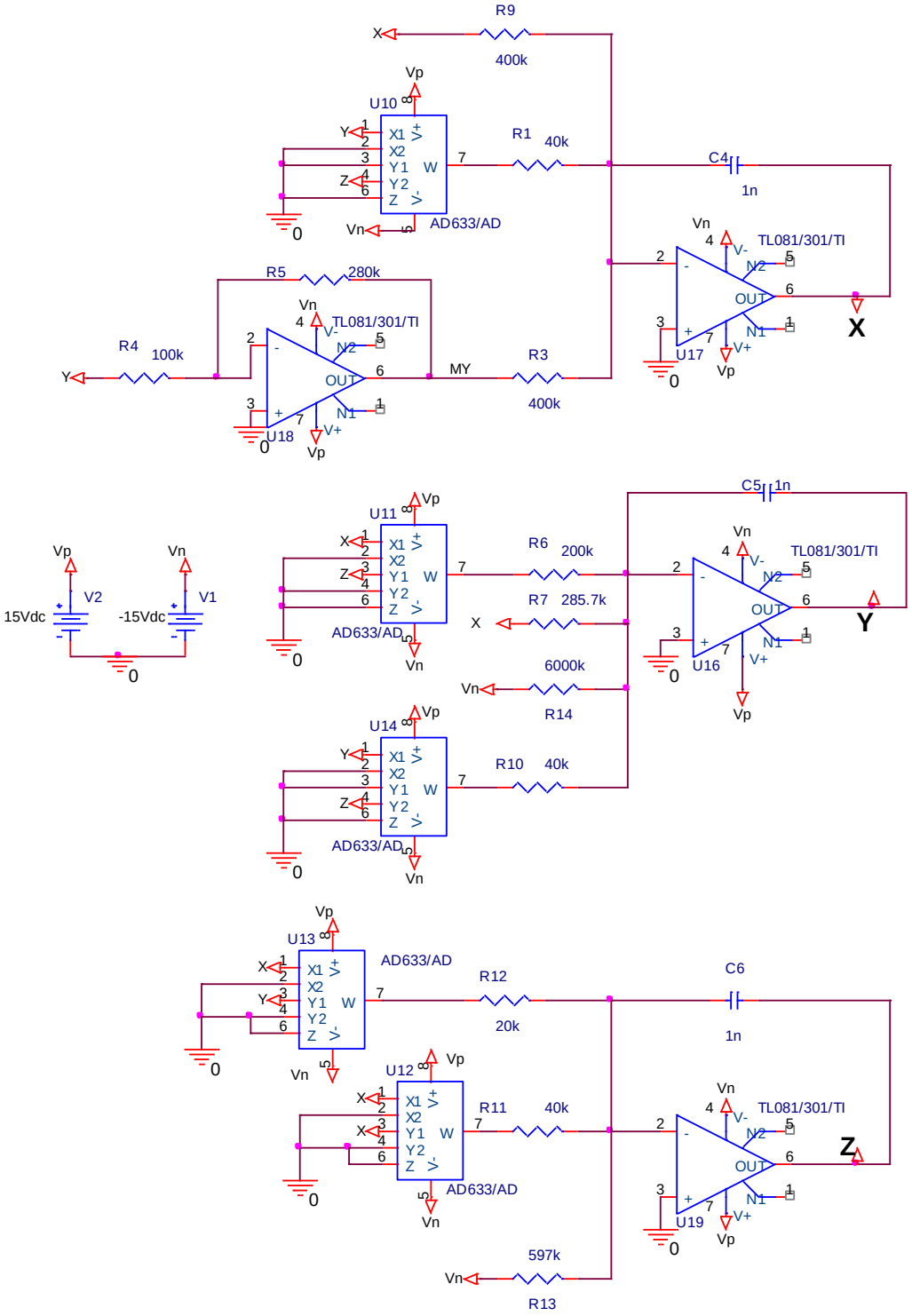


Şekil 4.6. Çatallaşma Diyagramı (b= 0.05-0.47)

4.1.6. OrCAD-PSpice'da elektronik devre simulasyon gerçeklemesi

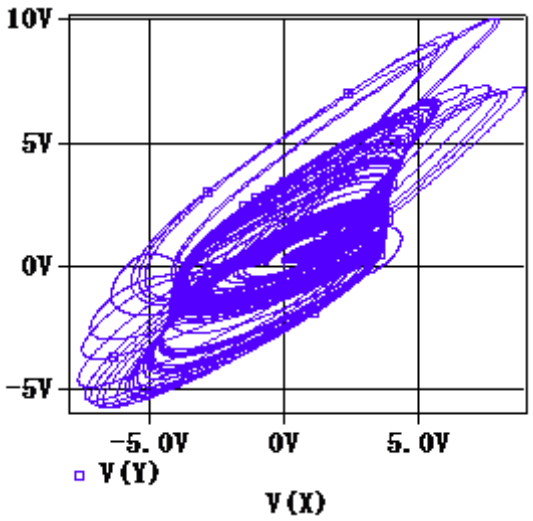
Yeni kaotik sistemi 1 için elektronik devre simulasyonu OrCAD-PSpice programı ile gerçekleştirilmiştir. Yeni kaotik sistemin elektronik devre şeması Şekil 4.7'de görüldüğü gibidir. Gerçekleştirilen elektronik devre; direnç, opamp, çarpma entegresi, kondansatör gibi temel elektronik elemanlardan meydana gelmektedir. Kaotik sistemin tasarımında başlangıç değerleri ve parametreler a =2.8, b=0.2, c=1.4, d=1, e=10, f=2 ve x=y=z=0 olarak alınmıştır. Başlangıç şartları (x,y,z) "0" olan sistemlerin elektronik devre gerçeklemeleri, başlangıç şartı "0" olmayanlara göre daha kolay gerçekleştirilebilmektedir. Yeni kaotik sistem 1, "0" başlangıç özelliğine (x,y,z) sahip olan bir sistem olduğu için gerçekleme daha kolay yapılabilmiştir. Gerçek ortam uygulamaları içinde başlangıç şartlarının "0" olması kolaylık sağlamaktadır.

Şekil 4.7'deki elektronik devre gerçeklemesinde opamp olarak TL081, çarpma entegresi olarak ise AD633 (Analog Devices) kullanılmıştır. Direnç değerleri R1=R10=R11=40K, R3=R9=400K, R4=100K, R5=280K, R6=200K, R7=285.7K, R12=20K, R13=597K, R14=6000K, kondansatör değerleri ise C1=C2=C3=1 nF olarak seçilmiştir. Bu değerler kaotik sistemin diferansiyel denklemlerinden yapılan modelleme sonucu elde edilmiştir.

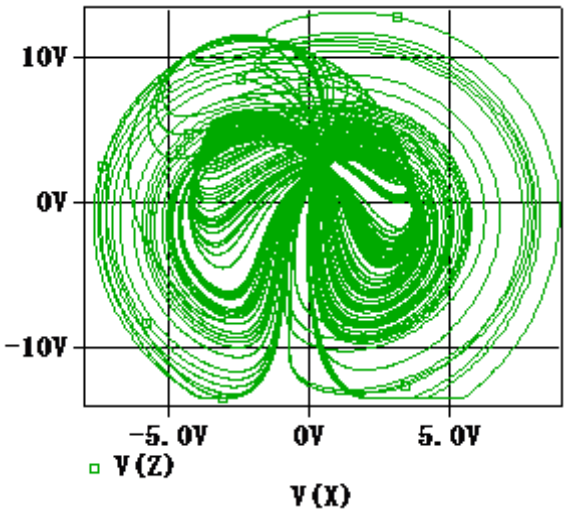


Şekil 4.7. Yeni Kaotik Sistem-1'in elektronik devre tasarımı

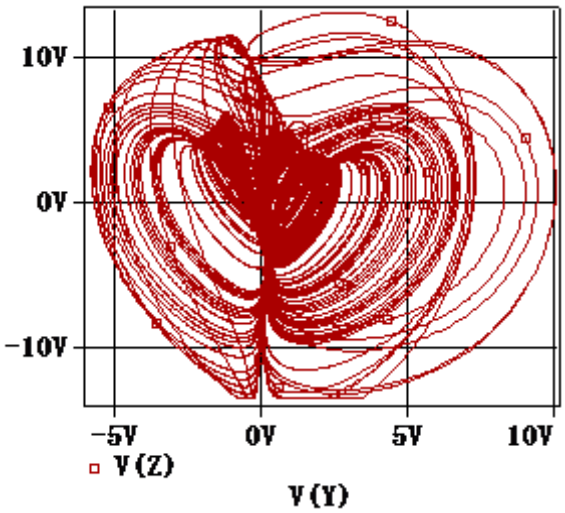
Gerçekleştirilen simulasyon sonucu yeni sistem 1 için elde edilen faz portre çıktıları ise, Şekil 4.8, 4.9 ve 4.10'da görüldüğü gibidir. Dikkat edilirse simulasyon faz portre çıktıları ve Matlab "odesolve.m" programı ile gerçekleştirilen faz portre çıktıları aynıdır. Faz portrelerinden de görüldüğü üzere yeni kaotik sistem, gerçek ortam uygulamaları için gerekli olan +15V, -15V aralıklarında olduğu için skale edilmelerine de gerek yoktur.



Şekil 4.8. OrCAD PSpice simulasyon programında çizdirilen x-y faz portre çıktısı



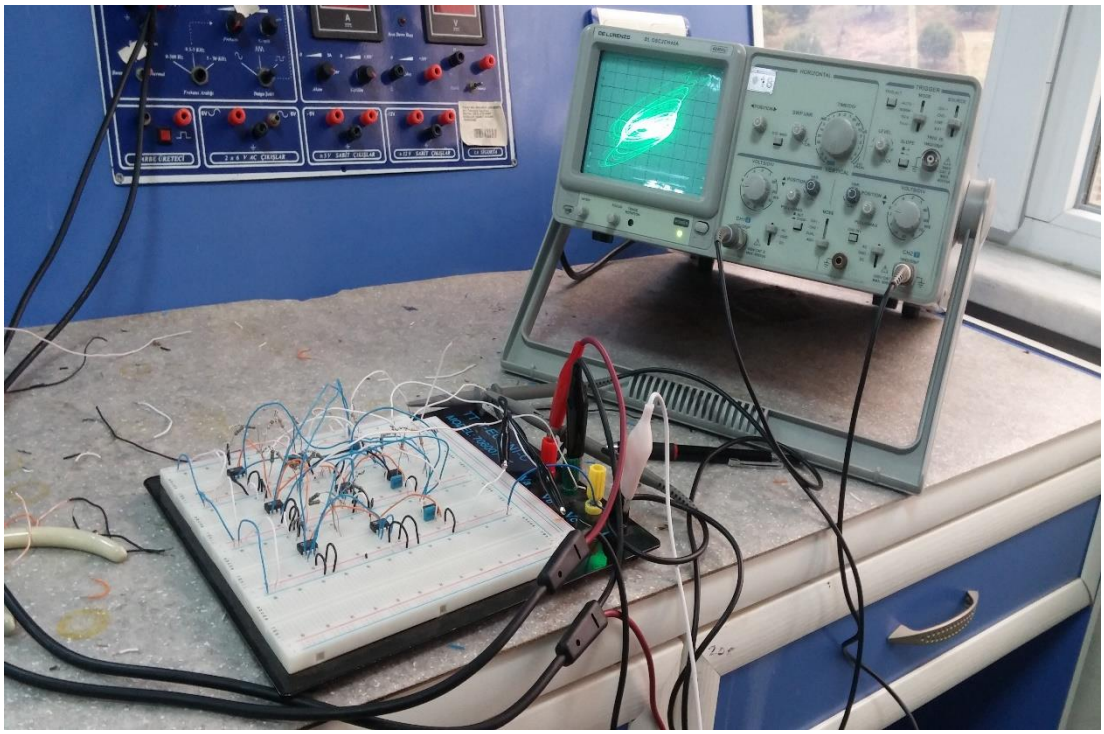
Şekil 4.9. OrCAD PSpice simulasyon programında çizdirilen x-z faz portre çıktısı



Şekil 4.10. OrCAD PSpice simulasyon programında çizdirilen y-z faz portre çıktısı

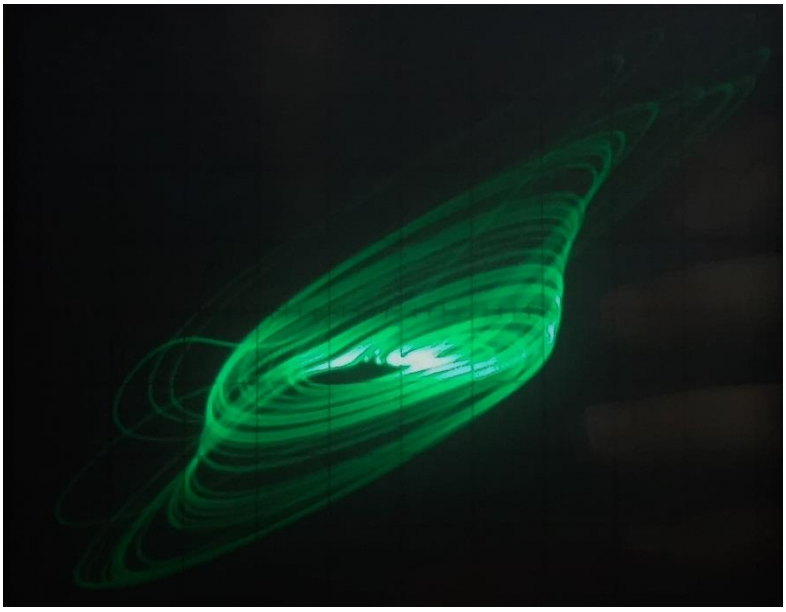
4.1.7. Gerçek ortam elektronik devre uygulaması ve osilaskop çıktıları

Şekil 4.11'de görüldüğü üzere, 4.1.6 bölümünde modellenerek gerçekleştirilen elektronik devre simulasyonunun gerçek ortam uygulaması bread-board üzerinde gerçekleştirilmiştir.

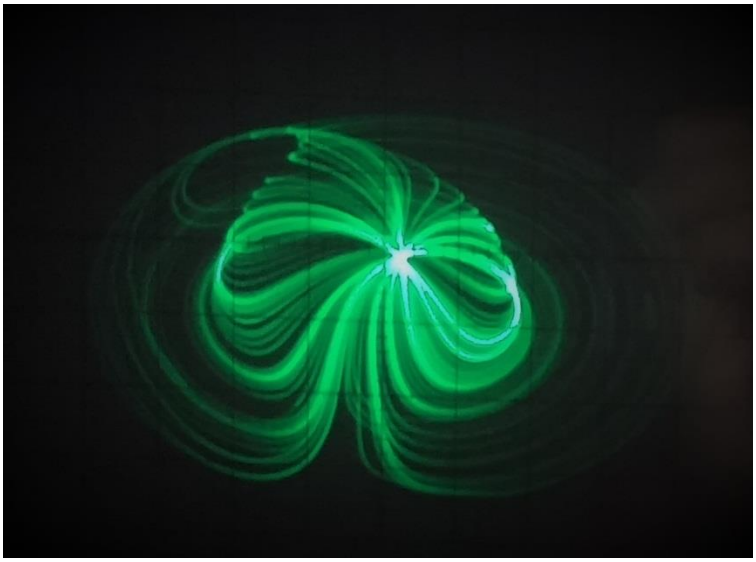


Şekil 4.11. OrCAD PSpice simulasyon programında çizdirilen y-z faz portre çıktısı

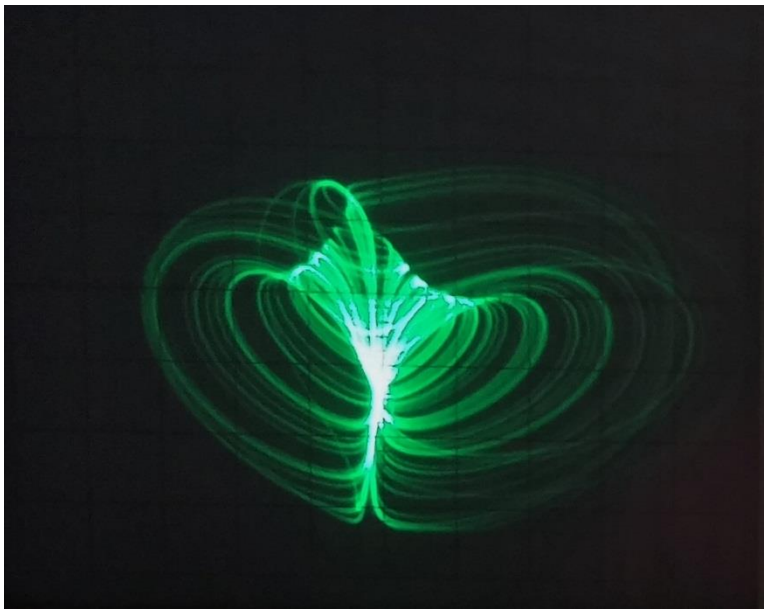
Şekil 4.11'de gerçek ortam uygulaması olarak gerçekleştirilen elektronik devre uygulamasının faz portre çıktıları Şekil 4.12, 4.13 ve 4.14'de gösterildiği gibidir.



Şekil 4.12. Osilaskop çıktı sonucu elde edilen x-y faz portre çıktısı



Şekil 4.13. Osilaskop çıktı sonucu elde edilen x-z faz portre çıktısı



Şekil 4.14. Osilaskop çıktı sonucu elde edilen y-z faz portre çıktısı

Matlab “odesolve”, OrCAD-PSpice ve gerçek ortam uygulama osilaskop çıktısından görüldüğü üzere tüm faz portre çıkışarından aynı sonuçlar elde edilmiştir.

Yapılan tüm analiz sonuçlarında (denge nokta, faz portre, Lyapunov üstel spektrum, zaman seri duyarlılık, çatallaşma, uygulamalar) tasarlanan yeni kaotik sistem 1’in, üç boyutlu sürekli zamanlı bir kaotik sistem olduğu sonucuna varılabilir.

4.2. Yeni Kaotik 2 Sistemi

Yeni kaotik sistem 2, sürekli zamanlı 3 boyutlu bir kaotik sistemdir. Denklem (4.12)’de görüldüğü üzere gibi kaotik sistem, 3 ayrı diferansiyel denklemden oluşmaktadır. Sistemde üç adet durum değişkeni (x, y, z), toplam on adet terim ve “a, b, c, d, e, f” olmak üzere altı adet parametre vardır. Sistemin başlangıç şartları x(0)=0, y(0)=0, z(0)=0’dır. Yeni Kaotik sistem 1’de olduğu gibi tüm başlangıç şartlarının “0” olması gerçek ortam uygulamaları için kolaylık sağlamaktadır.

$$\begin{aligned}\dot{x} &= y - ax + bxz \\ \dot{y} &= -cxz - dx + yz + e \\ \dot{z} &= f - y^2\end{aligned} \tag{4.12}$$

Sistem parametreleri $a = 0.7, b = 0.3, c = 4, d = 4.4, e = 0.1$ ve $f = 10$ olarak alınmıştır. Denklem (4.13)'de yeni kaotik sistemin parametreli hali verilmiştir.

$$\begin{aligned}\dot{x} &= y - 0.7x + 0.3xz \\ \dot{y} &= -4xz - 4.4x + yz + 0.1 \\ \dot{z} &= 10 - y^2\end{aligned} \tag{4.13}$$

4.2.1. Sistem denge nokta analizi

Sistem denge noktalarını bulmak için $\dot{x} = 0, \dot{y} = 0, \dot{z} = 0$ olarak ele alınırsa,

$$\begin{aligned}0 &= y - ax + bxz \\ 0 &= -cxz - dx + yz + e \\ 0 &= f - y^2\end{aligned}$$

elde edilir. Bu denklem sistemi çözülürse denge noktaları (Denklem 4.14)

$$\begin{aligned}&E_1(2.72355, 3.16222, -1.52369) \\ &E_2(-2.7018, -3.16227, -1.56808) \\ &E_3(-0.89835, -3.16227, -9.40029) \\ &E_4(0.89118, 3.16227, -9.49466)\end{aligned} \tag{4.14}$$

olarak bulunur.

Bulunan denge noktalarının kararsız olup olmadıklarını anlamak için yeni kaotik sistem 1'de olduğu gibi özdeğerlerinin bulunması gerekmektedir. Bir özdeğerin gerçel kısmının pozitif olması kaotiklik için yeter durum idi. Özdeğerleri bulmak için öncelikle sistemin Jacobian

matrisinin alınması gerekmektedir. Sistemin Jacobian matrisi Denklem (4.15)'de verildiği gibidir.

$$J(x,y,z)=\begin{bmatrix} bz-a & 1 & bx \\ -cz-d & z & -cx+y \\ 0 & -2y & 0 \end{bmatrix} \tag{4.15}$$

Sistem parametre değerleri Jacobian matrisinde yerlerine yazılırsa,

$$J(x,y,z)=\begin{bmatrix} 0.3z-0.7 & 1 & 0.3x \\ -4z-4.4 & z & -4x+y \\ 0 & -2y & 0 \end{bmatrix} \tag{4.16}$$

Denklem (4.16) elde edilmiş olur. E_1 özdeğerleri için; Denklem (4.14)'de bulunan denge noktaları, Denklem (4.16)'daki Jacobian matrisinde yerleri yazılırsa, Denklem (4.17) elde edilir.

$$J(E_1)\begin{bmatrix} -1.15710 & 1 & 0.81700 \\ 1.69476 & -1.52369 & -7.73173 \\ 0 & -6.32444 & 0 \end{bmatrix} \tag{4.17}$$

Son olarak ise, $\left|\lambda I - J(E_1)\right| = 0$ çözümünden E_1 için karakteristik denklem bulunabilir. E_1 için karakteristik denklem,

$$\lambda^3 + 2.69804\lambda^2 - 48.86425\lambda - 47.74627 = 0 \tag{4.18}$$

elde edilmiş olur. Bulunan karakteristik denklem çözümünden, özdeğerler aşağıdaki gibi bulunmaktadır.

$$\begin{aligned}\lambda_1 &= 6.285163\\ \lambda_2 &= -0.945078\\ \lambda_3 &= -8.038127\end{aligned} \tag{4.19}$$

Sistemin kararsızlığın için özdeğerlerden en az birinin pozitif olması gerekiyordu. λ_1'de pozitif olma koşulu sağlandığı için yeni kaotik sistem 2 kararsızdır. Bu yüzden yeni sistem 2 için kaotik bir sistemdir denilebilir.

E_2, E_3 ve E_4 denge noktalarıda, E_1 'de olduğıu gibi incelenir ve Jacobian matrislerinde yerlerine yazılırsa özdeğerler,

E_2 için,

$$\begin{aligned}\lambda_1 &= 8.284610\\ \lambda_2 &= -1.029108\\ \lambda_3 &= -9.994012\end{aligned} \tag{4.20}$$

E_3 için,

$$\begin{aligned}\lambda_1 &= 3.399162\\ \lambda_2 &= -1.333629\\ \lambda_3 &= -14.955914\end{aligned} \tag{4.21}$$

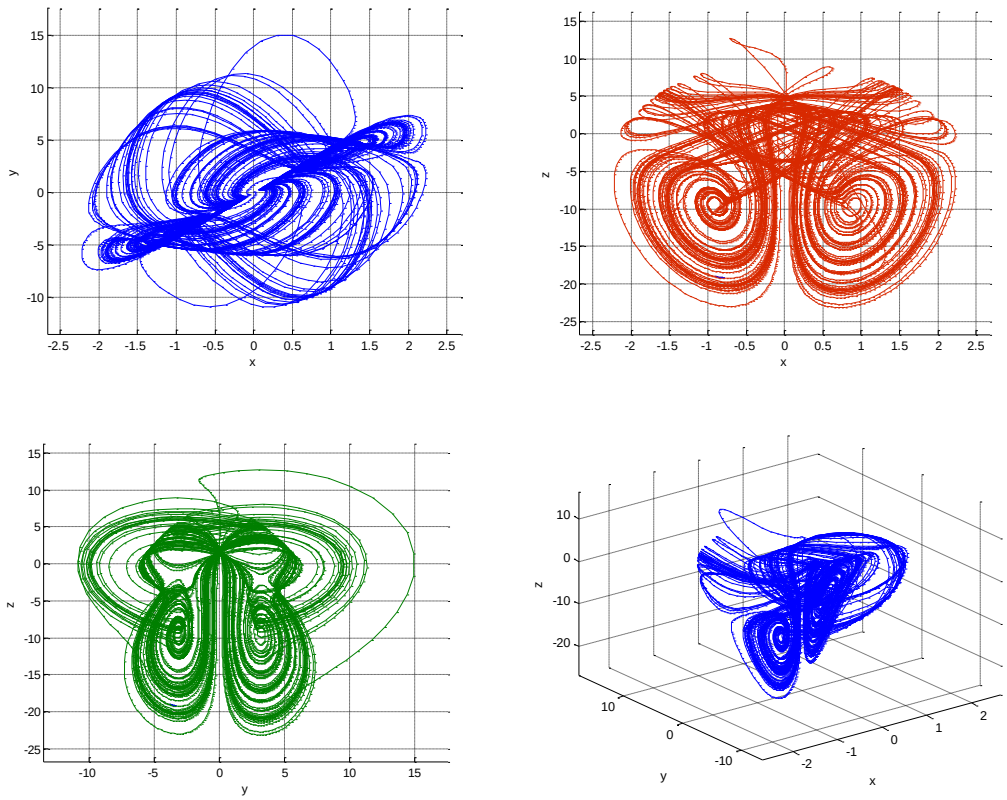
E_4 için,

$$\begin{aligned}\lambda_1 &= 0.221473+1.868522i\\ \lambda_2 &= 0.221473-1.868522i\\ \lambda_3 &= -13.486014\end{aligned} \tag{4.22}$$

olarak elde edilmiş olur. Görüldüğü üzere her denge noktasının özdeğerlerinden en az birisi pozitif değer taşımakta ve sistemin kararsızlığını, başka bir deyişle sistemin kaotikliğini göstermektedir.

4.2.2. Faz portre analizi

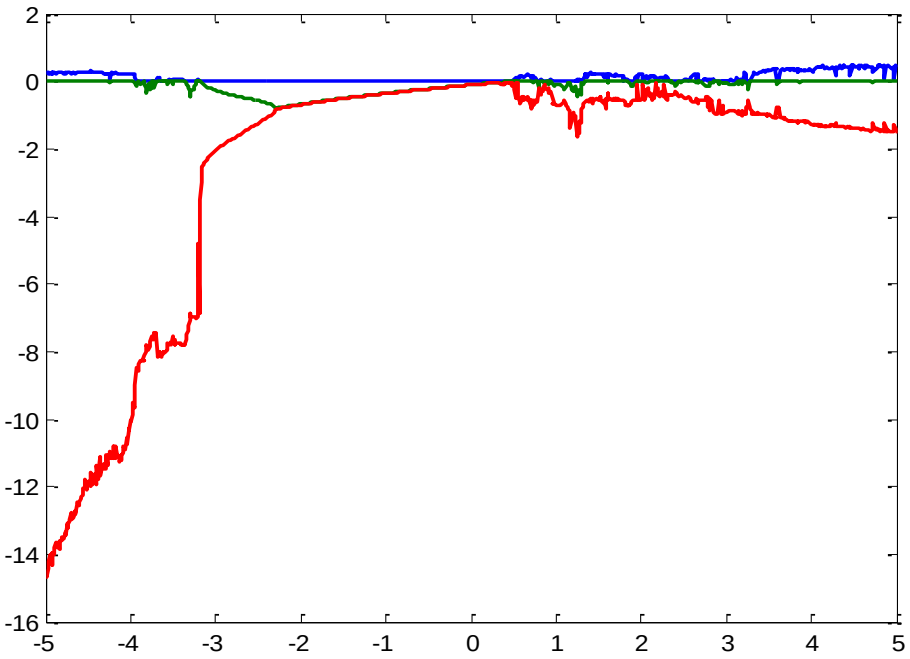
Yeni kaotik sistem 2'nin faz portreleri öncelikle Matlab "odesolve.m" programı ile incelenmiştir. Başlangıç değerleri x=0, y=0 ve z=0 için verilen kaotik sistemin x-y, y-z, x-z ve x-y-z olarak faz portre çıktıları Şekil 4.15'de verildiği gibidir. Şekilden 4.15'den de görüldüğü üzere yeni kaotik sistemin zengin dinamik davranışlara sahip olduğu söylenebilir. Yeni kaotik sistem 2 için simulasyon ve gerçek devre uygulama sonucu osilaskop çıktı faz portreleri, elektronik devre uygulama konu kısmı içerisinde verilecektir. Yeni kaotik sistem 2, faz portrelerinden de görüldüğü üzere, gerçek ortam uygulamaları için gerekli olan +15V, -15V aralıklarında olduğu için yeni kaotik sistem 1'de olduğu gibi skale edilmelerine de gerek yoktur.



Şekil 4.15. Yeni kaotik 2 için x-y, x-z, y-z ve x-y-z için faz portreleri

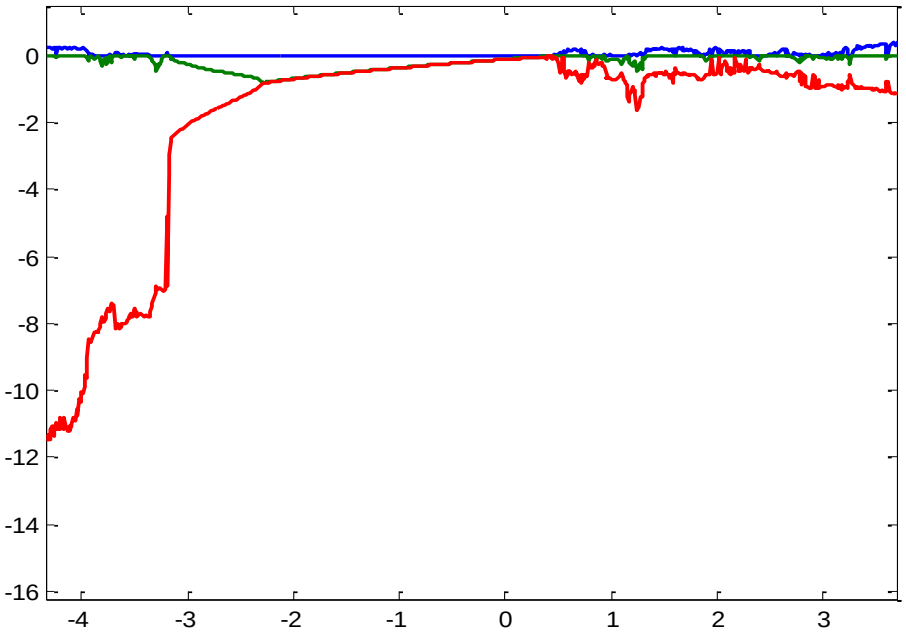
4.2.3. Lyapunov üstel spektrum analizi

Yeni Sistem 2'nin kaotik bir sistem olduğu Lyapunov üstellerinin de anlaşılabilmektedir. Yeni kaotik 2 sisteminin, 'c' parametresine göre -5 ile 5 aralığında Lyapunov üstel spektrumu Şekil 4.16'da verildiği gibidir. Lyapunov üstel analizinde sistemin kaotik olması için, değerlerin (+,0,-) olması gerekmekteydi. Şekil 4.16'dan da görüldüğü üzere sistem belli aralıklarla kaosa girip çıkmaktadır.



Şekil 4.16. Yeni kaotik 2 sistem için Lyapunov üstel grafiği (c= -5ile 5)

Şekil 4.17'de ise sistemin kaotik olduğu aralığı gösteren Lyapunov üstel grafiği yine c parametresi için -4.1 ve 3.6 arasında detaylı olarak verilmektedir.



Şekil 4.17. Yeni kaotik 2 sistem için detaylı Lyapunov üstel grafiği (c= -4.1 ile 3.6)

Lyapunov üstel grafiğinden ayrıca sistem boyutuda öğrenilebilmektedir. Örneğin Lyapunov üstel grafiğinin kaotik olan herhangi bir noktasından alınan değerler ile (L_1= 0.3107, L_2=0, L_3= -1.265) aşağıdaki Denklem (4.23) yardımıyla hesap yapılacak olursa (c = 4 için),

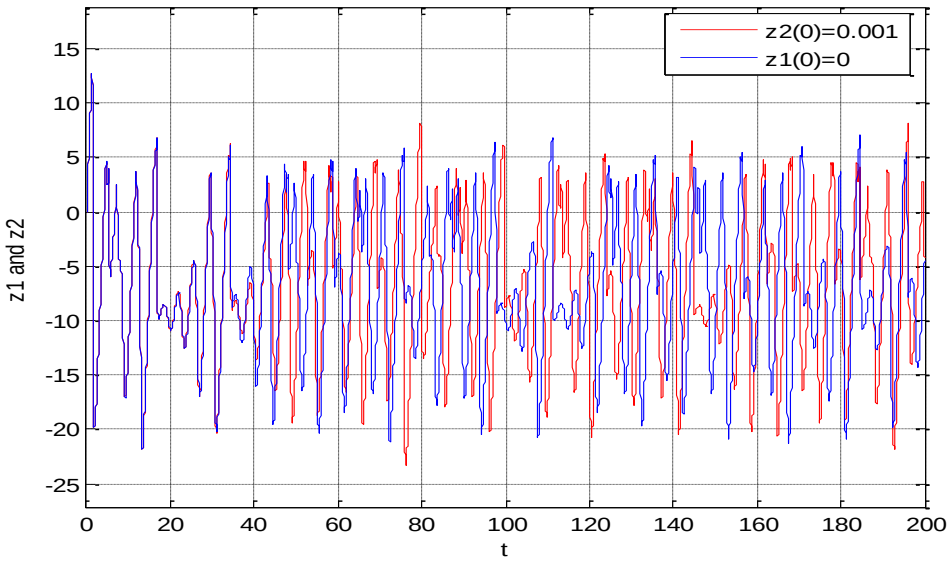
$$D_L = j + \frac{1}{\left|L_j + 1\right|}\sum_{i=1}^{j} L_i = 2 + \frac{L_1 + L_2}{\left|L_3\right|} = 2.2456126482 \qquad (4.23)$$

olarak bulunur. Yani yeni kaotik sistem boyutunun hesap sonucu 2.2456126482 değeri bir üst değeri yuvarlanacak olursa 3 olmuş olur. Sonuç olarak yeni kaotik 2 sistemi, 3 boyutlu sürekli zamanlı bir kaotik sistemdir diyebiliriz.

4.2.4. Zaman serisinde başlangıç değerlerine duyarlılık analizi

Sistemin başlangıç şartlarındaki çok küçük bir değişiklik sonucu farklı çıktılar vermesi kaotiklik hakkında önemli ipuçları vermektedir. Şekil 4.18'de görüldüğü üzere "z" başlangıç

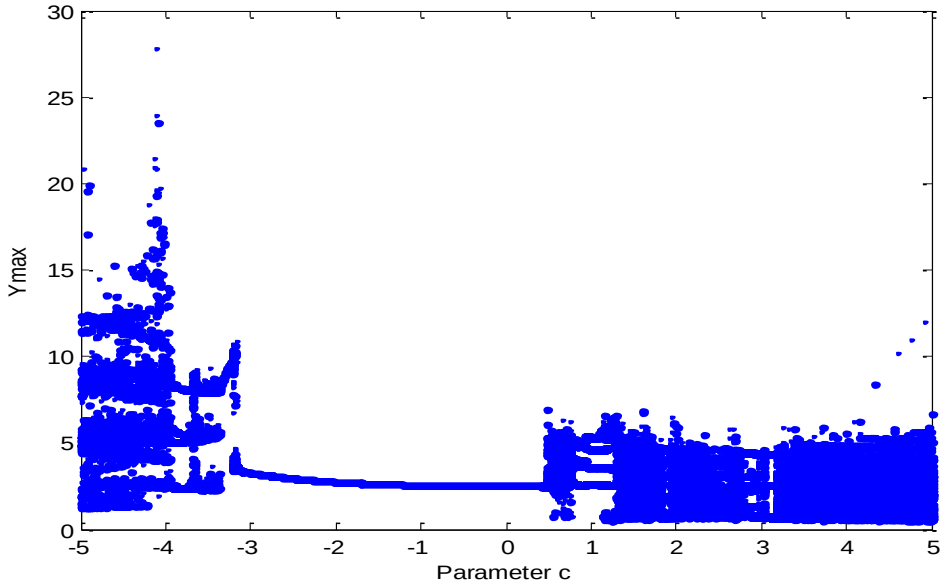
şartı “0” olarak alınmış ve sonucu mavi olan eğri elde edilmiştir. Fakat z 1/1000 değiştirilerek, yani “0.001” yapılarak kırmızı eğri elde edilmiştir. İki eğri Şekil 4.18’de beraber incelendiğinde çok küçük değişimlerin yeni sistem üzerinde farklı sonuçlar verdiği, yani başlangıç şartlarına çok hassas olduğu görülebilmektedir.



Şekil 4.18. z1(0)=0 ve z2(0)=0.001 için zaman seri grafiği

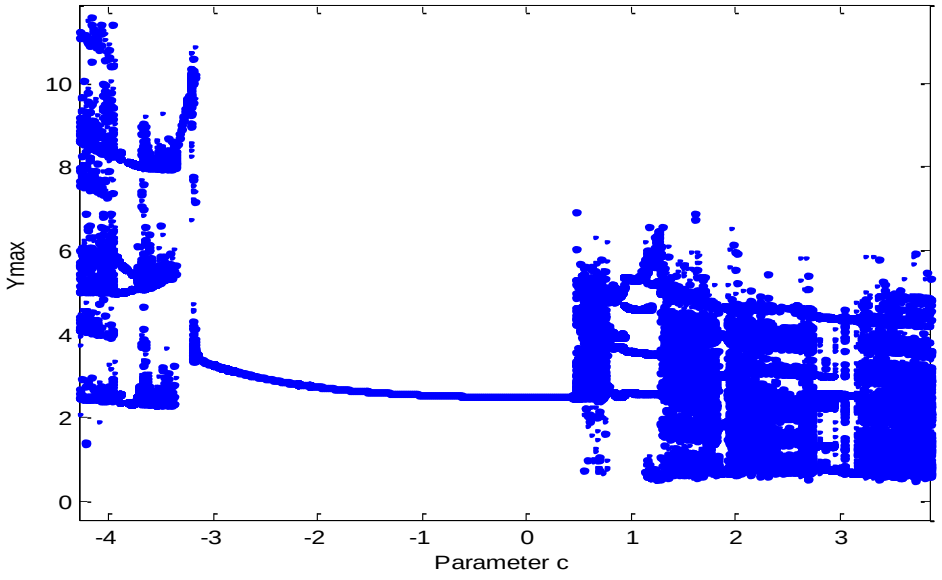
4.2.5. Çatallaşma diyagram analizi

Bu bölümde yeni kaotik sistem 2’ye ait çatallaşma diyagramları incelenmiştir. Bölüm 4.2.3‘de c parametresine göre Lyapunov üstel grafiği ele alınmıştı. Bu bölümde ise karşılaştırmak amacıyla yine c parametresine göre çatallaşma diyagramı çizdirilerek analiz yapılmıştır. Yeni kaotik sistem 1’de olduğu gibi Lyapunov üstel grafik analiz ve çatallaşma diyagramları aynı aralıklarda kaosa girip çıkmaktadırlar. Bu sayede sistemin bir nevi sağlaması yapılmış olmaktadır. Şekil 4.19’da c parametresi için -5 ve 5 aralığında çatallaşma diyagramı çizdirilmiştir.



Şekil 4.19. Yeni sistem 2 için Çatallaşma Diyagramı (c= -5 ve 5 arası)

Şekil 4.20'de ise, Şekil 4.19'da verilen çatallaşma diyagramının detaylı olarak c parametresi için -4.1 ve 3.6 arasında detaylı gösterimi verilmektedir. Şekil 4.19 ve Şekil 4.20'de görüldüğü üzere birçok noktanın üretildiği yerler sistemin kaotik olduğu kısımlardır. Lyapunov üstel grafiğinde de aynı kısımlar (+,-,0) değerler üretmekte, yani kaotik özellik göstermekte idi.



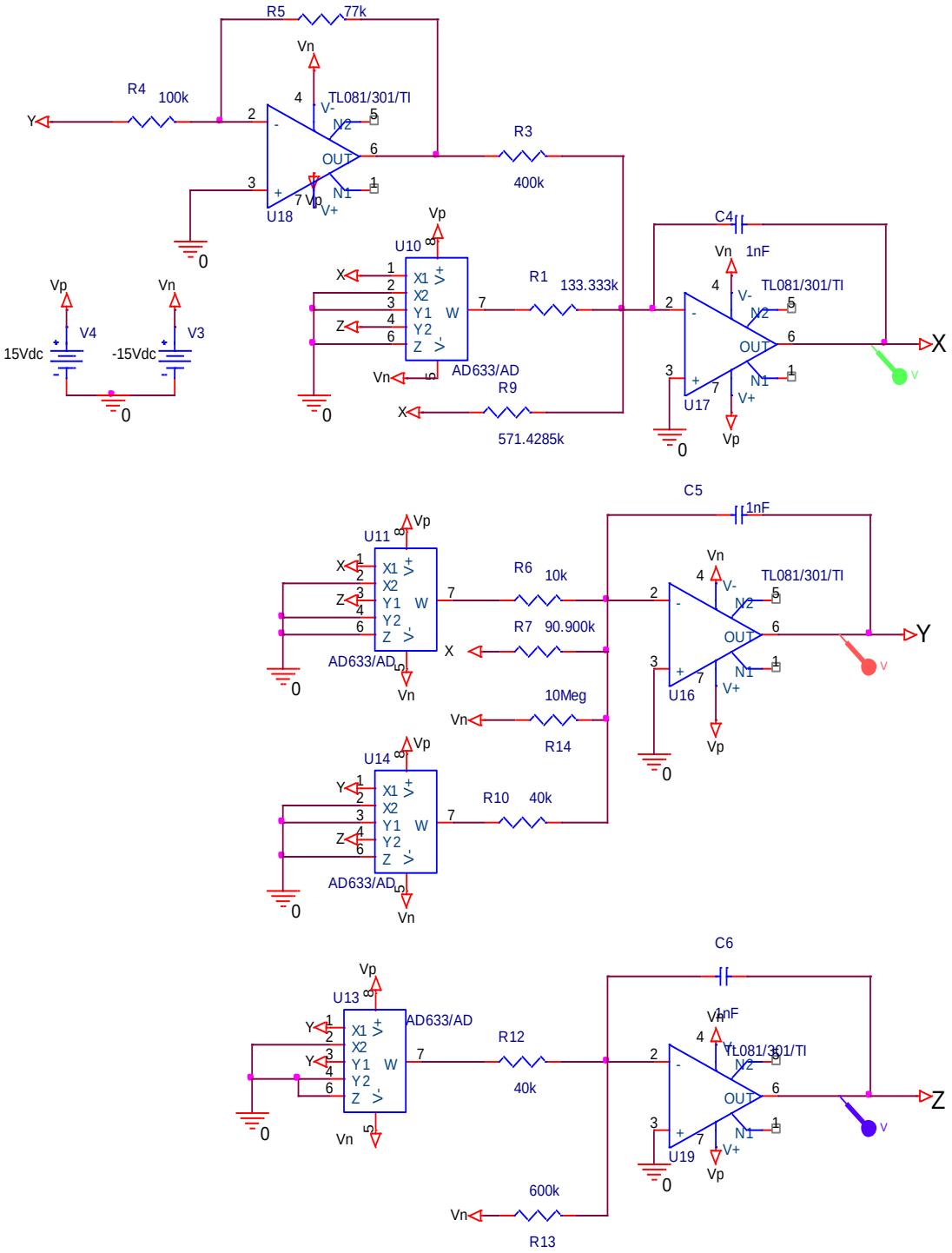
Şekil 4.20. Yeni sistem 2 için Çatallaşma Diyagramı (b= -4.1 ve 3.6 arası)

4.2.6. OrCAD-PSpice'da elektronik devre simulasyon gerçeklemesi

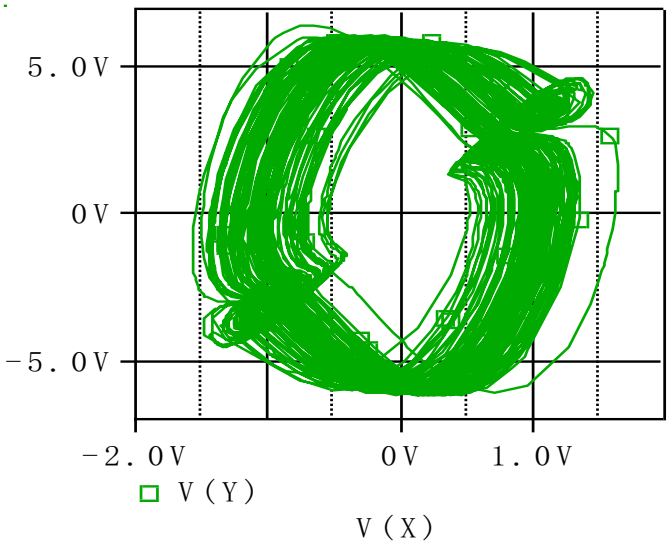
Yeni kaotik sistemi 2 için elektronik devre simulasyonu ilk sistemde olduğu gibi OrCAD-PSpice programı ile gerçekleştirilmiştir. Yeni kaotik sistem 2'nin elektronik devre şeması Şekil 4.21'de görüldüğü gibidir. Gerçekleştirilen elektronik devre, direnç, opamp, çarpma entegresi, kondansatör gibi temel elektronik elemanlardan meydana gelmektedir. Kaotik sistemin tasarımında başlangıç değerleri ve parametreler a =0.7, b=0.3, c=4, d=4.4, e=0.1, f=10 ve x=y=z=0 olarak alınmıştır. Başlangıç şartları (x,y,z) "0" olan sistemlerin elektronik devre gerçeklemeleri, başlangıç şartı "0" olmayanlara göre daha kolay gerçekleştirilebilmektedir. Yeni kaotik sistem 2, "0" başlangıç özelliğine (x,y,z) sahip olan bir sistem olduğu için gerçekleme daha kolay yapılabilmiştir. Gerçek ortam uygulamaları içinde başlangıç şartlarının "0" olması kolaylık sağlamaktadır.

Şekil 4.21'deki elektronik devre gerçeklemesinde opamp olarak TL081, çarpma entegresi olarak ise AD633 (Analog Devices) kullanılmıştır. Direnç değerleri R1=133.3, R3=400K, R4=100K, R5= 77K, R6=10K, R7=90.9K, R9=571.9, R10=R12=40K, R13=600K, R14=10M; kondansatör değerleri ise C1=C2=C3=1 nF olarak seçilmiştir. Bu değerler kaotik sistemin diferansiyel denklemlerinden yapılan modelleme sonucu elde edilmiştir.

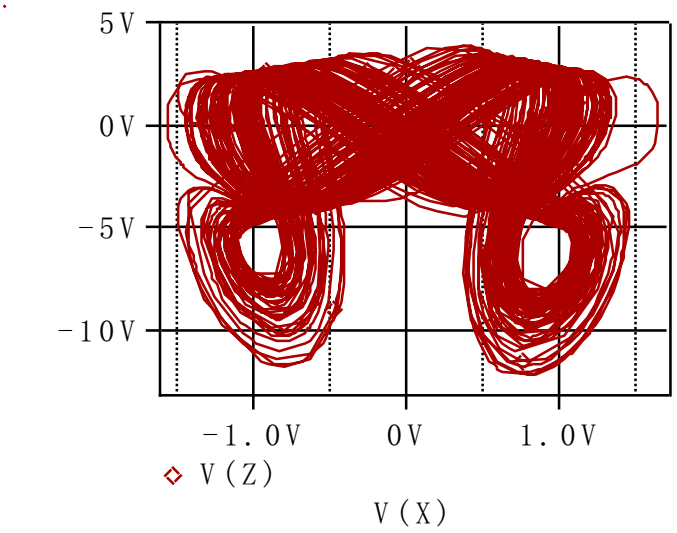
Gerçekleştirilen simulasyon sonucu yeni sistem 2 için elde edilen faz portre çıktıları ise Şekil 4.22, 4.23 ve 4.24'de görüldüğü gibidir. Faz portrelerinden de görüldüğü üzere yeni kaotik sistem, gerçek ortam uygulamaları için gerekli olan +15V, -15V aralıklarında olduğu için skale edilmelerinede gerek yoktur.



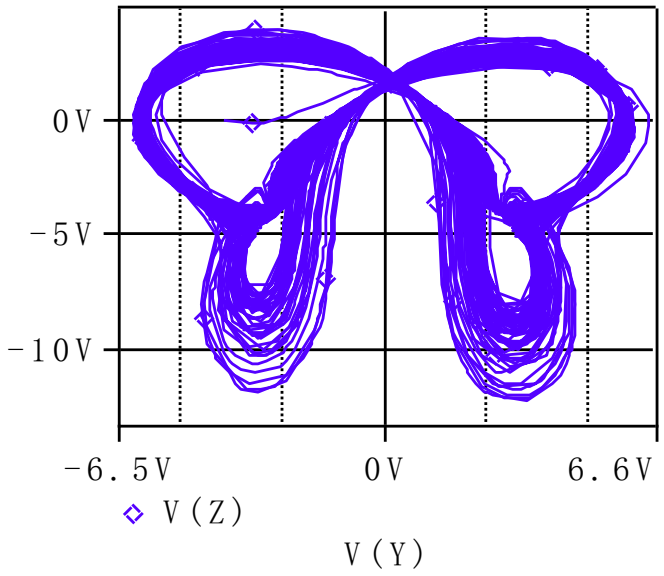
Şekil 4.21. Yeni Kaotik Sistem-2'nin elektronik devre tasarımı



Şekil 4.22. OrCAD PSpice simulasyon programında çizdirilen x-y faz portre çıktısı



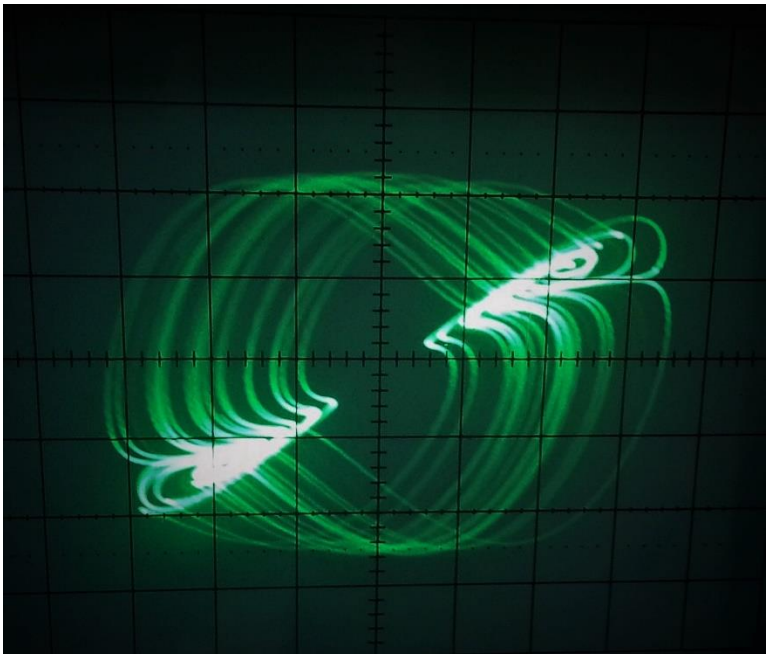
Şekil 4.23. OrCAD PSpice simulasyon programında çizdirilen x-z faz portre çıktısı



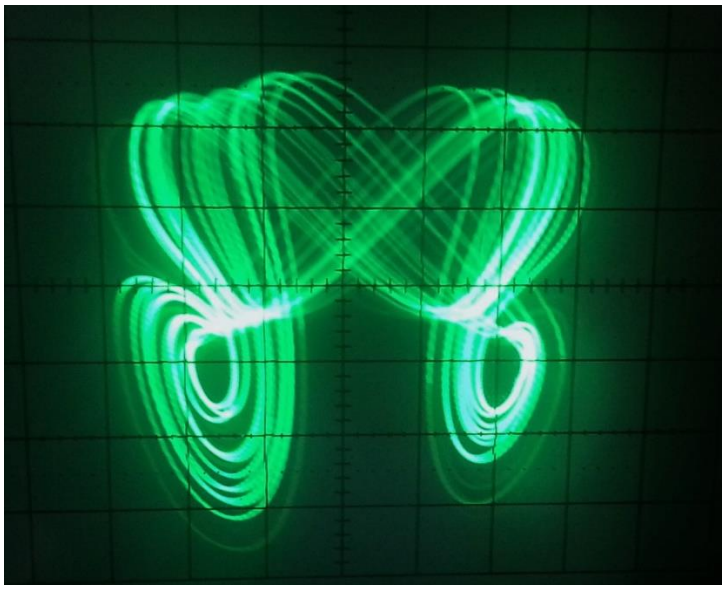
Şekil 4.24. OrCAD PSpice simulasyon programında çizdirilen y-z faz portre çıktısı

4.2.7. Gerçek ortam elektronik devre uygulaması ve osilaskop çıktıları

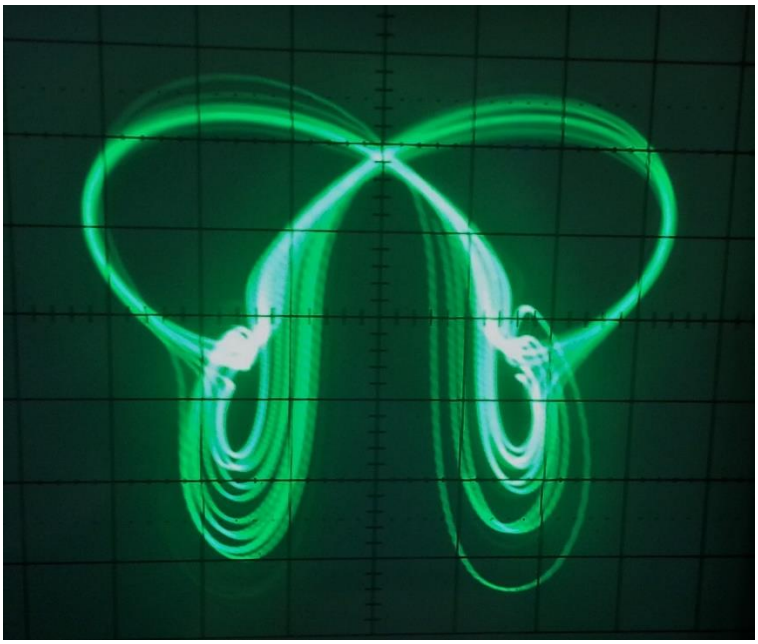
Gerçek ortam uygulaması olarak gerçekleştirilen elektronik devre uygulamasının faz portre çıktıları ise Şekil 4.25, 4.26 ve 4.27'de gösterildiği gibidir.



Şekil 4.25. Osilaskop çıktı sonucu elde edilen x-y faz portre çıktısı



Şekil 4.26. Osilaskop çıktı sonucu elde edilen x-z faz portre çıktısı



Şekil 4.27. Osilaskop çıktı sonucu elde edilen y-z faz portre çıktısı

BÖLÜM 5. YENİ BULUNAN KAOTİK SİSTEMLERLE RASGELE SAYI ÜRETECİ (RSÜ) TASARIMI, İSTATİKSEL RASGELELİK TESTLERİ ve SONUÇLARI

5.1. Yeni Kaotik Sistemlerin RSÜ Tasarımı için Ayrıklaştırılması

Euler, Heun, dördüncü dereceden Runge Kutta (RK4), beşinci dereceden Runge Kutta (RK5), Dormand-Prince [115] gibi nümerik analiz algoritmalarıyla diferansiyel denklemlerin sayısal çözümü gerçekleştirilerek sürekli zamanlı sistemler, ayrıklaştırılıp birçok sayısal uygulamada kullanılabilmektedir [93].

Euler algoritması, sürekli zamanlı sistemleri ayrıklaştırma işlemi için kullanılan en basit yöntemlerden birisidir. Euler algoritması sayısallaştırma için çok sık tercih edilmekte, fakat hassas çözümler yapamamaktadır. Kaotik sistemler hassas olduklarından dolayı euler algoritmasını kullanmak uygun değildir. Gelişmiş olan diğer bir nümerik analiz algoritması olan Heun algoritması ise, yüksek frekanslı fonksiyonlar için uygun değildir. Bu nedenlerden dolayı kaotik sistemlerin ayrıklaştırılabilmesi için daha hassas olan RK4, RK5, Dormand-Price gibi nümerik analiz yöntemlerinin kullanılması gerekmektedir. Tez çalışmasında yeni bulunan kaotik sistemlerin ayrıklaştırılmış modelleri için RK4 algoritması kullanılmıştır. RK4 algoritması oldukça iyi hassas sonuçlar üretmekte, hata oranı oldukça düşük, yazılımsal ve donanımsal olarak RK5 gibi daha gelişmiş nümerik analiz algoritmalarından daha basittir [93]. RK4 algoritması ile ayrıklaştırılmış modelleri oluşturulan yeni kaotik sistemler, rasgele sayı üreteç tasarımı ve diğer sayısal birçok alanda kullanılabilmektedir. Yeni kaotik sistemlerin ayrıklaştırılması sonucu, kaotik sistemlerden float sayılar elde edilerek, rasgele sayı üretimi için ilk basamak tamamlanmış olacaktır. Elde edilen float ifadeleri sayıların gerçek ortam uygulamalarında kullanılabilmeleri için ikili sayı sistemine dönüştürülmeleri gerekmektedir. Bu aşamalardan ilerleyen bölümlerde bahsedilecektir.

5.1.1. RK4 Nümerik analiz algoritması

Runge-Kutta veya RK4 olarak isimlendirilen RK4 nümerik analiz algoritmasına ait ifadeler denklem kümesi 1'de verildiği gibidir. En son denklemde verilen $y_{\lambda+1}$ değeri girilen sürekli zamanlı sayının ayrıklaştırılmış halidir. $y_{\lambda+1}$ değerinin bulunabilmesi için öncelikle k_1, k_2, k_3 ve k_4 değerlerinin hesaplanması gerekmektedir. Verilen denklem kümesindeki ilk ifadedeki k_1, Δh kadar aralık sonundaki başlangıç eğimi, k_2, k_3 ve k_4 değeri ise Δh aralığının orta noktasındaki sırasıyla k_1, k_2 ve k_3 değeri kullanılarak hesaplanan eğimdir. Bu şekilde devam edilerek y_λ değeri ve Δh aralık değerleri kullanılarak sistemin bir sonraki değeri olan $y_{\lambda+1}$ değeri sayısal olarak hesaplanmaktadır [109]. Aşağıda Denklem (5.1)' deki denklem kümesinde RK4 nümerik ayrıklaştırma algoritmasında bir adım için, gerekli işlemler sırasıyla verilmiştir.

$$\begin{aligned} k_1 &= f(y_\lambda) \\ k_2 &= f(y_\lambda + \frac{\Delta h}{2} k_1) \\ k_3 &= f(y_\lambda + \frac{\Delta h}{2} k_2) \\ k_4 &= f(y_\lambda + \Delta h k_3) \\ y_{\lambda+1} &= y_\lambda + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)\Delta h \end{aligned} \tag{5.1}$$

5.1.2. Bulunan yeni sistemler 1 ve 2'nin RK4 algoritması ile ayrıklaştırılması

Ayrıklaştırma işlemleri benzer şekilde gerçekleştirildiği için bu bölümde sadece yeni kaotik sistem 1'in RK4 nümerik diferansiyel denklem çözüm yöntemleri kullanılarak ayrıklaştırılmış modeli çıkarılmıştır. Yeni kaotik sistem 2'nin ayrıklaştırılmış modeli aynı yollardan elde edilebilmektedir. Denklem kümesi, Denklem (5.2)'de verilen yeni kaotik sistem 1'in, Denklem (5.3)'de f, g ve ξ fonksiyonlarına göre RK4 algoritması kullanılarak ayrıklaştırılmış matematiksel modeli verilmektedir [93].

$$\begin{aligned}\dot{x} &= f(t,x,y,z) = ay - x + zy \\ \dot{y} &= g(t,x,y,z) = -bxz - cx + yz + d \\ \dot{z} &= \delta(t,x,y,z) = e - fxy - x^2\end{aligned} \tag{5.2}$$

$$\begin{aligned}x(k+1) &= x(k) + \frac{1}{6}\Delta h\left[\kappa_1(k) + 2\kappa_2(k) + 2\kappa_3(k) + \kappa_4(k)\right] \\ y(k+1) &= y(k) + \frac{1}{6}\Delta h\left[\lambda_1(k) + 2\lambda_2(k) + 2\lambda_3(k) + \lambda_4(k)\right] \\ z(k+1) &= z(k) + \frac{1}{6}\Delta h\left[\xi_1(k) + 2\xi_2(k) + 2\xi_3(k) + \xi_4(k)\right]\end{aligned} \tag{5.3}$$

Denklem (5.3)'de bulunan κ, λ, ξ parametreleri, Denklem (5.4)'de verildiği gibi hesaplanmaktadır. Denklem (5.3)'de 1. basamaktaki tüm κ parametreleri, Denklem (5.2)'deki kaotik sistem 1'in ilk denklemine ait değerleri, 2. basamaktaki tüm λ parametreleri ikinci denkleme ait değerleri, 3. Basamaktaki tüm ξ parametreleri ise üçüncü denkleme ait değerleri Denklem (5.4)'de verildiği gibi hesaplanmaktadır. Bulunan katsayılar Denklem (5.3) deki RK4 algoritmasında yerlerine konularak, kaotik sistemin Δh kadar adım sonrası değeri olan ayrıklaştırılmış *x(k+1), y(k+1)* ve *z(k+1)* değerlerini hesaplanmaktadır. Her adım sonunda bulunan *x(k+1), y(k+1)* ve *z(k+1)* değerleri hem hesaplanan o adımda çıkış olarak, hem de bir sonraki adımda başlangıç şartı olarak kullanılabilmektedir [93].

$$\kappa_1 = f(x(k), y(k), z(k))$$
$$\lambda_1 = g(x(k), y(k), z(k))$$
$$\xi_1 = \delta(x(k), y(k), z(k))$$

$$\kappa_2 = f(x(k) + \frac{1}{2}\Delta h\kappa_1, y(k) + \frac{1}{2}\Delta h\lambda_1, z(k) + \frac{1}{2}\Delta h\xi_1)$$
$$\lambda_2 = g(x(k) + \frac{1}{2}\Delta h\kappa_1, y(k) + \frac{1}{2}\Delta h\lambda_1, z(k) + \frac{1}{2}\Delta h\xi_1)$$
$$\xi_2 = \delta(x(k) + \frac{1}{2}\Delta h\kappa_1, y(k) + \frac{1}{2}\Delta h\lambda_1, z(k) + \frac{1}{2}\Delta h\xi_1)$$

$$\kappa_3 = f(x(k) + \frac{1}{2}\Delta h\kappa_2, y(k) + \frac{1}{2}\Delta h\lambda_2, z(k) + \frac{1}{2}\Delta h\xi_2) \quad (5.4)$$
$$\lambda_3 = g(x(k) + \frac{1}{2}\Delta h\kappa_2, y(k) + \frac{1}{2}\Delta h\lambda_2, z(k) + \frac{1}{2}\Delta h\xi_2)$$
$$\xi_3 = \delta(x(k) + \frac{1}{2}\Delta h\kappa_2, y(k) + \frac{1}{2}\Delta h\lambda_2, z(k) + \frac{1}{2}\Delta h\xi_2)$$

$$\kappa_4 = f(x(k) + \Delta h\kappa_3, y(k) + \Delta h\lambda_3, z(k) + \Delta h\xi_3)$$
$$\lambda_4 = g(x(k) + \Delta h\kappa_3, y(k) + \Delta h\lambda_3, z(k) + \Delta h\xi_3)$$
$$\xi_4 = \delta(x(k) + \Delta h\kappa_3, y(k) + \Delta h\lambda_3, z(k) + \Delta h\xi_3)$$

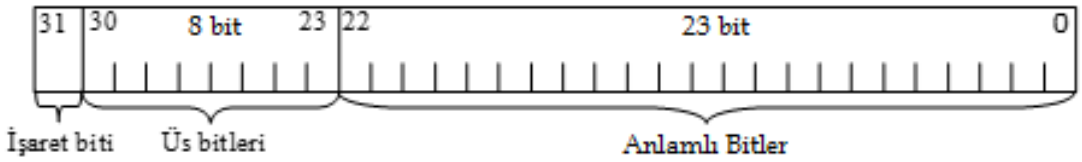
Yeni kaotik sistem 1'in ikinci denklemi (y) için örnek ayrıklaştırılmış ifadeler 10 adım için Tablo 5.1'de verilmiştir.

Tablo 5.1. Yeni kaotik 1 sisteminin RK4 ile ayrıklaştırma işlemi sonucu elde edilen sayısal ifadeler

Adım Sayısı	Ayrıklaştırma Sonucu Elde Edilen Sayısal İfadeler
1	0.003866508205357
2	0.017022109192077
3	0.042062260612412
4	0.082042775172218
5	0.140612988083349
6	0.222137744672527
7	0.331789743765176
8	0.475571084896906
9	0.660185655353679
10	0.892627200833881

5.2. Yeni Sistemler ile Rasgele Sayı Üreteç Tasarımı

RK4 nümerik analiz algoritması sonucu ayrıklaştırılan sayıların en hassas ve yüksek miktarda gösterilmesini sağlayan sayı sistemleri, tek duyarlı (32-bit_float) ve çift duyarlı (64-bit_double) olarak bilinen kayan noktalı sayı standartıdır. Bir önceki bölümde noktalı olarak elde edilen sayıların gerçek ortam uygulamalarında kullanılabilmeleri için binary sayı sistemine çevrilmeleri gerekmektedir. Şekil 5.1'de 32-bit tek duyarlı IEEE 754-1985 kayan noktalı sayı standardı gösterilmiştir. Bu standartın en yüksek değerlikli biti, yani en anlamlı olan 31. Bit, işaret biti olarak isimlendirilmektedir. İşaret bitinin değeri eğer "0" ise sayı pozitif, "1" ise sayı negatif olmaktadır. Sayının üstel kısmını belirtmek için, ikinci kısımdaki 8 adet olan üs bitleri kullanılmaktadır. Verilen gösterim tek duyarlı olduğu için üs için kaydırma değeri 2^{8-1}-1'den 127'dir. Son kısım olan üçüncü kısım ise anlamlı bitleri temsil etmektedir. Anlamlı bitler kayan noktalı sayı standardında sayının kesirli yani anlamlı kısmını göstermektedir [116].



Şekil 5.1. 32-bit IEEE 754-1985 kayan noktalı sayı standardı gösterimi

RK4 algoritması ile ayrıklaştırılan sayılar, kayan noktalı sayı standartında yani float sayılar olduğu için rasgele sayı üretimi için ikili sayı sistemine dönüştürülmesi gerekmektedir. Tablo 5.1'de, RK4 ile elde edilen sayıların ikili sayı formatına dönüştürülmüş formatı Tablo 5.2'de verilmiştir. Dikkat edilirse, tüm float sayılar pozitif olduğu için, ikili sayıya dönüştürülmüş formatta da en anlamlı yani 31. bitde "0" sonucu elde edilmiştir.

Tablo 5.2. Yeni kaotik 1 sisteminin RK4 ile ayrıklaştırma işlemi sonucu elde edilen sayısal ifadeler

Adım Sayısı	Ayrıklaştırma Sonucu Elde Edilen Sayısal İfadeler	Binary Dönüşümleri (Float to Binary)
1	0.003866508205357	00111011011111010110010100111110
2	0.017022109192077	00111100100010110111000111110011
3	0.042062260612412	00111101001011000100100101111010
4	0.082042775172218	00111101101010000000011000001011
5	0.140612988083349	00111110000011111111110011011010
6	0.222137744672527	00111110011000110111100000010100
7	0.331789743765176	00111110101010011110000001011000
8	0.475571084896906	00111110111100110111111000001110
9	0.660185655353679	00111111001010010000000111101101
10	0.892627200833881	00111111011001001000001100110111

Verilen tablolarda kaotik sistemden örnek olarak 10 bit üretilen sayıların, RSÜ için çok daha fazla üretilerek istatistiksel testlere tabi tutulması gerekmektedir. FIPS-140-1 testi için 20.000 bitlik, NIST-800-22 testi için ise en az 1.000.000 bitlik bir sayı dizisine ihtiyaç duyulmaktadır.

Bulunan yeni kaotik sistemlerden RSÜ tasarımı için yukarıdaki bölümlerde anlatılan işlemlerin gerçekleştirilmesi gerekmektedir. Bulunan yeni kaotik sistemler 3 boyutlu oldukları için RK4 ile ayrıklaştırma sonucu 3 farklı float sayı elde edilecektir. Elde edilen 3 farklı float sayının ikili sayı format dönüşümleri yapılarak RSÜ tasarımı için kullanılabilir. Ayrıklaştırma ve ikili sayı formata çevirme işlemleri sonucunda, kaotik sistemden üretilen her bir sayı sonucu “0” ve “1” lerden oluşan 32 bitlik bir sayı dizisi elde edilmiş olur. Elde edilen bu sayı dizilerindeki “0” ve “1”ler istenen duruma göre seçilebilir. RSÜ tasarımında, FIPS-140-1 ve NIST-800-22 testleri için sayıların olabildiğince rasgele olması istenen durum olduğu için aynı sayıların arka arkaya gelmemesi gerekmektedir. Tablo 5.3’de, yeni kaotik sistem 1 birinci denkleminin ürettiği ilk 30 bitlik sayılar ve bu sayıların RSÜ için ikili sayı formatına çevrilmiş durumları gösterilmiştir.

Tablo 5.3. Yeni kaotik sistem 1 (1.denklem örnek ilk 30 bitlik veri)

Üretilen Sayı No.	Yeni Sistem 1 (1.denklem_ilk 30 bitlik Binary Veri)
1	00111011011111010110010100111110
2	00111100100010110111000111110011
3	00111101001011000100100101111010
4	00111101101010000000011000001011
5	00111110000011111111110011011010
6	00111110011000110111100000010100
7	00111110101010011110000001011000
8	00111110111100110111111000001110
9	00111111001010010000000111101101
10	00111111011001001000001100110111
11	00111111100101101111001001110011
12	00111111110000110001100110011001
13	00111111111101101001101011011011
14	01000000000110000010000101110100
15	01000000001101101010101000111100
16	01000000010101001011000100010111
17	01000000011011110110110001001011
18	01000000100000100001011110101101
19	01000000100010001010001110000101
20	01000000100010110011111110100000
21	01000000100010100111111010011010
22	01000000100001110100110011000001
23	01000000100000101001100000010100
24	01000000011110100011100000010011
25	01000000011011101010100011010110
26	01000000110001100001101110100000
27	01000000101011110100110100111101
28	01000000100110010000011101101110
29	01000000100000110011010001011100
30	01000000001101101100111100101000

Tablo 5.3 dikkatli incelendiğinde, üretilen binary sayılar sıra ile alınırsa, ilk başlarda (sol taraftan) ard arda “0” ve “1”lerin olduğu gözükmektedir. Bu yüzden sayıların bit bit alınması gerekmektedir. Her binary sayının en solundaki bit alınırsa, sayıların hepsi pozitif olduğu için hep “0” elde edilmiş olacaktır. Tabloya dikkat edilirse en sağdaki bitleri alarak RSÜ tasarımı yapmak, istatistiksel testler için en uygunu olacaktır. Eğer her durumda istatistiksel testlerden başarılı sonuçlar elde edilemiyorsa, sayılar arasında XOR, AND veya OR gibi mantıksal operatörler kullanılarak rasgelelik arttırılabilir.

Yeni kaotik sistem 1 için sadece en sağdaki bitler arka arkaya sıralanıp, 20.000 ve 1.000.000'luk olarak oluşturulan bit dizisi sonucuda, gerçekleştirilen FIPS-140-1 ve NIST-800-22 istatistiksel test sonuçlarından başarıyla geçmiştir. Yeni kaotik sistem 2'de ise en sağdan bir önceki bit alındığı takdirde FIPS-140-1 ve NIST-800-22 istatistiksel testlerinden başarıyla geçtiği görülmüştür.

Rasgelelik testlerinin yapılabilmesi için üretilen sayılar için harcanan süreler iki yeni sistem için küçük farklılıklar göstermektedir. Yeni kaotik sistem 1'de, FIPS-140-1 testi için bir sayı 0.21 ms aralıklarla, 20.000 bitlik sayı dizisi ise 4.2583 snde elde edilmiştir. NIST-800-22 için gerekli olan 1.000.000 sayı dizisi 6290.1 snde üretilmiştir. Yeni kaotik sistem 2'de, FIPS-140-1 testi için bir sayı 0.22 ms aralıklarla, 20.000 bitlik sayı dizisi ise 4.5517 snde elde edilmiştir. NIST-800-22 için gerekli olan 1.000.000 sayı dizisi 6308.8 snde üretilmiştir.

5.3. Yeni Kaotik Sistemler ile Tasarlanan Rasgele Sayı Üreteçlerin İstatistiksel Rasgelelik Testleri ve Sonuçları

Yeni kaotik sistemlerle üretilen rasgele sayıların, güvenlik açısından şifreleme çalışmalarında kullanılabileceğini göstermek için bazı testlere tabi tutulmaları gerekmektedir. Bu tez çalışmasında, literatürde yaygın olarak kullanılan ve Bölüm 2.3.2'de anlatılan, uluslararası geçerliliğe sahip FIPS-140-1 ve NIST-800-22 testleri gerçekleştirilmiştir. Bölüm 6'da da görüleceği üzere, FIPS-140-1 ve NIST-800-22 testlerinden geçmiş sayı dizileriyle gerçekleştirilen (sinyal, metin, ses, resim, video gibi multimedya verilerini) şifreleme uygulamalarının güvenlik analizlerine bakıldığında oldukça iyi sonuçlar verdiği görülmektedir. Bu bölümde yeni kaotik sistem 1 ve 2 ile üretilen sayıların FIPS-140-1 ve NIST-800-22 test sonuçları verilmiştir. Her iki kaotik sistem ile elde edilen sayı dizilerinin, FIPS-140-1 ve NIST-800-22 testlerinden başarıyla geçtikleri görülmektedir.

5.3.1. Yeni kaotik sistem 1 RSÜ FIPS-140-1 ve NIST-800-22 testleri

Bu bölümde, yeni kaotik sistem 1 ile üretilen rasgele sayı dizilerinin, FIPS-140-1 ve NIST-800-22 testleri ve sonuçları verilmiştir. Yeni kaotik sistem 1 için her iki testden de başarılı sonuçlar elde edilmiştir.

5.3.1.1. Yeni kaotik sistem 1 RSÜ FIPS-140-1 testleri ve sonuçları

Yeni kaotik sistem 1 için, FIPS-140-1 test sonuçları Tablo 5.4'de görüldüğü gibidir. Test başarı kriterlerinin de anlatıldığı Bölüm 2.3.2'deki FIPS-140-1 testinde, 4 farklı test bulunmaktadır. Tablo 5.4'den de görüldüğü üzere yeni kaotik sistem 1 ile üretilen sayı dizileri FIPS-140-1 testindeki 4 testden de başarılı olmuştur. Koşu testinde, değişik blok uzunluklarına göre farklı sonuçlar elde edilmektedir. Yeni kaotik sistem ile üretilen sayı dizisi, tüm blok uzunlukları içinde testden geçmiştir. Tablo 5.4'de blok uzunluğu 6'ya kadar olan test sonuçları verilmiştir.

Tablo 5.4. Yeni kaotik sistem 1 RSÜ FIPS-140-1 testleri

FIPS-140-1 Testleri	Test başarı kriterleri	RK4-Tabanlı	Sonuç
Monobit Testi	9654< n <10346	10028	Başarılı
Poker Testi	1.03< X <57.4	51	Başarılı
Koşu Testi (Blok Uzunluğu_1)	2267≤ x ≤2733	2482	Başarılı
Koşu Testi (Blok Uzunluğu_2)	1079≤x≤1421	1226	Başarılı
Koşu Testi (Blok Uzunluğu_3)	502≤x≤748	642	Başarılı
Koşu Testi (Blok Uzunluğu_4)	223≤x≤402	344	Başarılı
Koşu Testi (Blok Uzunluğu_5)	90≤x≤223	145	Başarılı
Koşu Testi (Blok Uzunluğu_6)	90≤x≤223	150	Başarılı
Uzun Koşu Testi	34>Koşu	34>11	Başarılı

5.3.1.2. Yeni kaotik sistem 1 RSÜ NIST-800-22 testleri ve sonuçları

FIPS-140-1 testinden çok daha kapsamlı olan NIST-800-22 testi, içerisinde 16 farklı test bulunmaktadır. Bazı sayı dizileri FISP-140-1 testinden geçerken, NIST-800-22 testinden

geçememektedir. Fakat NIST-800-22 testini geçen sayı dizileri, genellikle FISP-140-1 testinden geçmektedir. Bu yüzden sadece FISP-140-1 testlerini yapmanın yanında NIST-800-22 testlerini de gerçekleştirmek çok daha sağlıklı olacaktır. Random-Excursions ve Random Excursions Variant testlerinden dolayı NIST-800-22 testi için genellikle 1.000.000'luk sayı dizisine ihtiyaç duyulmakadır. Yeni kaotik sistem 1 ile elde edilen 1.000.000 sayı dizisine yönelik gerçekleştirilen NIST-800-22 testinin sonuçları Tablo 5.5'de verilmiştir. Random-Excursions testindeki x değişkeni $-4 \leq x \leq -1$ *ve* $4 \leq x \leq 1$ arası değerler alabildiği için test sonucunda 8 adet P-değeri üretilmektedir. 8 testin sonucuda başarılı olarak gerçekleştirilmiştir, fakat tabloda örnek olması için bunlardan sadece birisi (x=-4) verilmiştir. Aynı şekilde Random Excursions Variant testide benzer olarak $-9 \leq x \leq -1$ *ve* $9 \leq x \leq 1$ arasında 18 adet P-değeri üretmektedir. Bu testdeki 18 değerin tümü için başarılı sonuçlar elde edilmiştir ve tabloda sadece x=-9 için olan değer verilmiştir [93].

Test sonuçlarının başarımı için P-değerine bakılmaktadır. Eğer $P\text{-değeri} \geq 0.001$ ise testi gerçekleştirilen sayı dizileri rasgele olarak kabul edilmekte, yani NIST-800-22 testini geçmektedir. Tablo 5.5'deki sonuçlardan da görüldüğü üzere tüm P değerleri 0.001'den büyük oldukları için tasarlanan RSÜ testi geçmiştir ve şifreleme uygulamalarında kullanmak için uygundur denilebilir.

Tablo 5.5. Yeni kaotik sistem 1 RSÜ NIST-800-22 testleri

İstatistiksel Testler	P-değeri	Sonuç
Frequency (Monobit) Test	0.5850	Başarılı
Block-Frequency Test	0.4921	Başarılı
Cumulative-Sums Test	0.7486	Başarılı
Runs Test	0.7858	Başarılı
Longest-Run Test	0.5146	Başarılı
Binary Matrix Rank Test	0.8459	Başarılı
Discrete Fourier Transform Test	0.5772	Başarılı
Non-Overlapping Templates Test	0.0114	Başarılı
Overlapping Templates Test	0.1298	Başarılı
Maurer's Universal Statistical Test	0.6092	Başarılı
Approximate Entropy Test	0.0409	Başarılı
Random-Excursions Test	0.9727	Başarılı
Random-Excursions Variant Test	0.4973	Başarılı
Serial Test-1	0.2679	Başarılı
Serial Test-2	0.5038	Başarılı
Linear-Complexity Test	0.0881	Başarılı

5.3.2. Yeni kaotik sistem 2 RSÜ FIPS-140-1 ve NIST-800-22 testleri

Bu bölümde, yeni kaotik sistem 2 ile üretilen sayı dizilerinin, FIPS-140-1 ve NIST-800-22 testleri ve sonuçları verilmiştir. Her iki testden de başarılı sonuçlar elde edilmiştir.

5.3.1.1. Yeni kaotik sistem 2 RSÜ FIPS-140-1 testleri ve sonuçları

Yeni kaotik sistem 2 için, FIPS-140-1 test sonuçları Tablo 5.6'da görüldüğü gibidir. Test başarı kriterlerinin de anlatıldığı Bölüm 2.3.2'deki FIPS-140-1 testinde, 4 farklı test bulunmaktadır. Tablo 5.6'dan da görüldüğü üzere yeni kaotik sistem 2 ile üretilen sayı dizileri FIPS-140-1 testindeki 4 testden de başarılı olmuştur. Koşu testinde, değişik blok uzunluklarına göre farklı sonuçlar elde edilmektedir. Yeni kaotik sistem ile üretilen sayı dizisi, tüm blok uzunlukları içinde testden geçmiştir. Tablo 5.6'da blok uzunluğu 6'ya kadar olan test sonuçları verilmiştir.

Tablo 5.6. Yeni kaotik sistem 2 RSÜ FIPS-140-1 testleri

FIPS-140-1 Testleri	Test başarı kriterleri	RK4-Tabanlı	Sonuç
Monobit Testi	9654< n <10346	9965	Başarılı
Poker Testi	1.03< X <57.4	51	Başarılı
Koşu Testi (Blok Uzunluğu_1)	2267≤ x ≤2733	2537	Başarılı
Koşu Testi (Blok Uzunluğu_2)	1079≤x≤1421	1215	Başarılı
Koşu Testi (Blok Uzunluğu_3)	502≤x≤748	654	Başarılı
Koşu Testi (Blok Uzunluğu_4)	223≤x≤402	325	Başarılı
Koşu Testi (Blok Uzunluğu_5)	90≤x≤223	141	Başarılı
Koşu Testi (Blok Uzunluğu_6)	90≤x≤223	159	Başarılı
Uzun Koşu Testi	34>Koşu	34>11	Başarılı

5.3.1.2. Yeni kaotik sistem 2 RSÜ NIST-800-22 testleri ve sonuçları

Yeni kaotik sistem 2 ile elde edilen 1.000.000'luk sayı dizisine yönelik gerçekleştirilen NIST-800-22 testinin sonuçları Tablo 5.7'de verilmiştir. Yeni kaotik sistem 1'de de olduğu gibi Random-Excursions testi için tabloda x=-4 değeri, Random Excursions Variant testi için

ise x=-9 değeri gösterilmiştir. Random-Excursions ve Random Excursions Variant tüm x değerleri için başarılı sonuçlar elde edilmiştir. Tablo 5.7'deki sonuçlardan görüldüğü üzere tüm P değerleri 0.001'den büyük ve eşit oldukları için tasarlanan RSÜ testi geçmiştir ve şifreleme uygulamalarında kullanmak için uygundur denilebilir.

Tablo 5.7. Yeni kaotik sistem 2 RSÜ NIST-800-22 testleri

İstatistiksel Testler	P-değeri	Sonuç
Frequency (Monobit) Test	0.2278	Başarılı
Block-Frequency Test	0.9360	Başarılı
Cumulative-Sums Test	0.3375	Başarılı
Runs Test	0.6358	Başarılı
Longest-Run Test	0.9854	Başarılı
Binary Matrix Rank Test	0.5215	Başarılı
Discrete Fourier Transform Test	0.7989	Başarılı
Non-Overlapping Templates Test	0.0013	Başarılı
Overlapping Templates Test	0.7930	Başarılı
Maurer's Universal Statistical Test	0.4614	Başarılı
Approximate Entropy Test	0.4676	Başarılı
Random-Excursions Test	0.3529	Başarılı
Random-Excursions Variant Test	0.4797	Başarılı
Serial Test-1	0.2254	Başarılı
Serial Test-2	0.0835	Başarılı
Linear-Complexity Test	0.1059	Başarılı

BÖLÜM 6. GELİŞTİRİLEN RSÜ TABANLI YENİ BİR MULTİMEDYA ŞİFRELEME YÖNTEMİ, UYGULAMALARI ve GÜVENLİK ANALİZ SONUÇLARI

Sinyal, metin, ses, resim, video gibi çoklu-ortam verilerinin şifrelenmesine yönelik birçok çalışma gerçekleştirilmiştir. Tez çalışmasında, bu bölüme kadar yeni bulunan kaotik sistemler ile rasgele sayı üreteci tasarımı yapılarak, istatistiksel testleri gerçekleştirilmiştir. Bu bölümde ise testleri başarıyla geçen RSÜ leri ile bazı çoklu-ortam verilerinin şifrelenmesine yönelik çalışmalar yapılarak, güvenlik analizleri gerçekleştirilip, diğer şifreleme yöntemleriyle performans değerlendirmeleri yapılmıştır.

İlk aşamada yeni kaotik sistem 1 ile üretilen RSÜ'lerini kullarak sadece çoklu-ortam verilerini şifreleme işlemleri gerçekleştirilmiştir. Çalışmada; sinyal, metin, ses, resim ve video verileri için ayrı ayrı şifreleme çalışmaları yapılmıştır. Yapılan şifreleme işlemlerinin ardından güvenlik analizleri verilmiştir.

İkinci aşamada, yeni kaotik sistem 1 deki sistem parametreleri (a,b,c, ..) anahtar olarak kullanılıp, sadece yeni kaotik sistem 1'in parametrelerine yönelik, yeni kaotik sistem 2 ile şifreleme işlemleri gerçekleştirilmiştir. Bu parametreler bilinmeden üretilen sayılar bilinemeyecek ve dolayısıyla şifreli veriler çözülemeyecektir.

Son aşamada ise yeni kaotik sistem 2 ile şifrelenen, yeni kaotik sistem 1'in başlangıç değer ve parametreleri, anahtar dağıtımının problem olduğu durumlarda, asimetrik şifreleme algoritmalardan RSA şifreleme yöntemi ile şifrelenerek hibrid bir sistem oluşturulup, şifreleme işlemleri gerçekleştirilmiştir.

6.1. Yeni Kaotik Sistem 1 Kullanarak Tasarlanan RSÜ ile Multimedya Verilerinin Şifreleme Uygulamaları

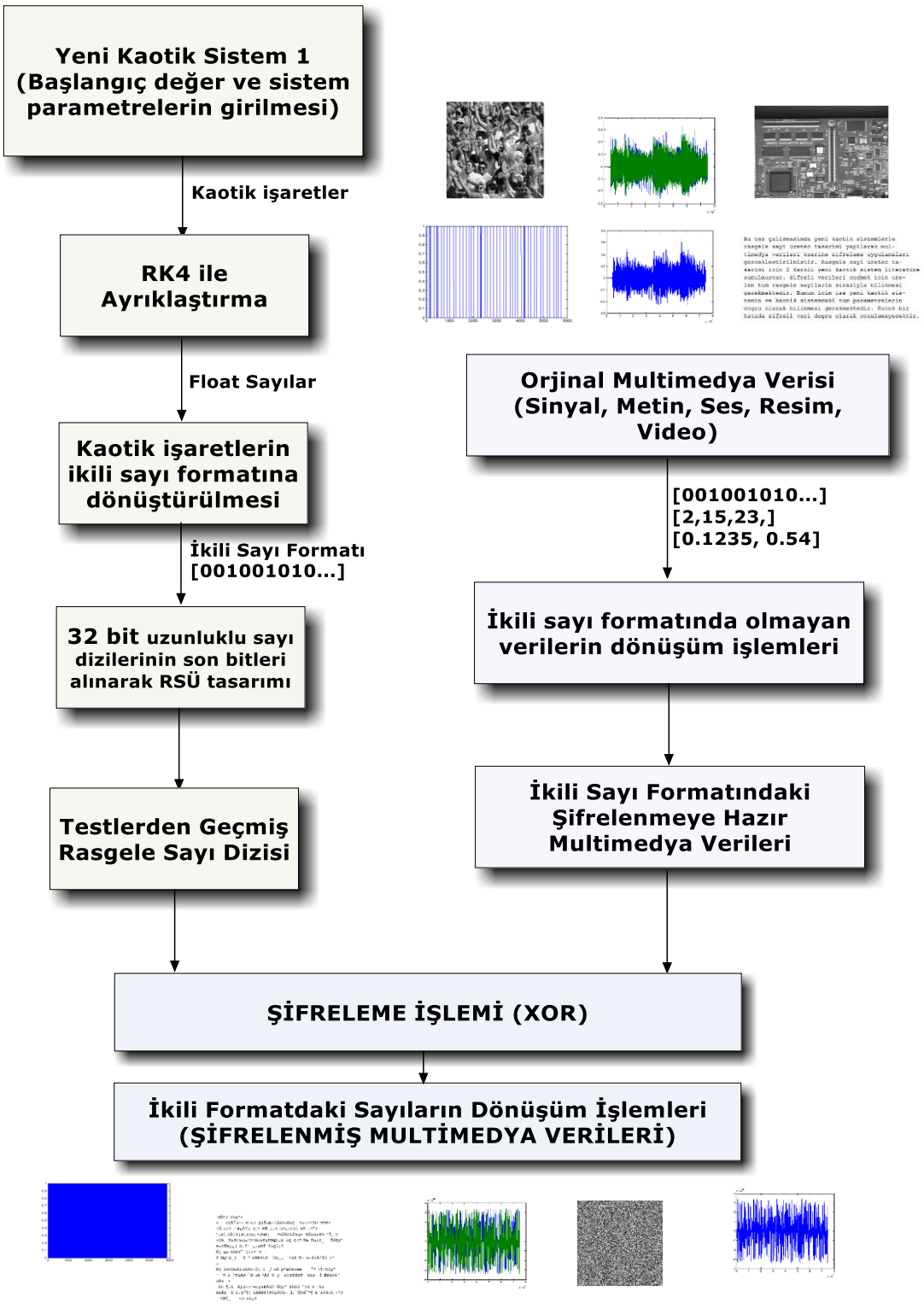
Bu bölümde; resim, metin, ses, resim ve video multimedya verileri için yeni kaotik sistem 1 Denklem (6.1) ile tasarlanan RSÜleri kullanılarak şifreleme çalışmaları gerçekleştirilmiştir.

$$\begin{aligned} \dot{x} &= ay - x + zy \\ \dot{y} &= -bxz - cx + yz + d \\ \dot{z} &= e - fxy - x^2 \end{aligned} \tag{6.1}$$

Şekil 6.1 ve Şekil 6.2'de şifreleme ve şifre çözme işlemleri için gerekli olan blok diyagram verilmiştir. Şekil 6.1'de verilen blok diyagramda, şifreleme işlemi için "0" ve "1'lerden oluşan anahtarlara ihtiyaç duyulmaktadır. Anahtarlar, yeni kaotik sistem 1'den rasgele sayılar olarak elde edilmiştir.

Şifrelenmek istenen çoklu-ortam verileri, ikili sayı formatında değilse, bazı dönüşüm işlemlerinin yapılması gerekmektedir. Float ve decimal sayı tipinde olan veriler, şifreleme işlemleri için ikili sayı formatına çevrilmeleri gerekmektedir. İkili sayı formatında elde edilen anahtarlar ve şifrelenmek istenen multimedia verileri, basit mantıksal operatörler kullanarak şifreleme işlemlerine tabi tutulmuştur. Şifrelenmiş veriler "0" ve "1'lerden oluştuğu için, gerçek ortam uygulamaları için herhangi bir problem bulunmamaktadır. Şifreleme işlemlerinde basit işlemlerin kullanılması, bellek sıkıntısı yaşanan birçok gerçek ortam uygulamaları için olanak sağlamaktadır. Testleri geçmiş rasgele sayılar ile şifreleme işlerinin yapılması, şifrelenmiş verilerdeki güvenliği arttırılmış olmaktadır. Yapılan uygulamalar ile, şifrelenmiş verilerin güvenlik analizleri de birlikte verilmiştir.

$$\dot{x} = ay - x + zy$$
$$\dot{y} = -bxz - cx + yz + d$$
$$\dot{z} = e - fxy - x^2$$



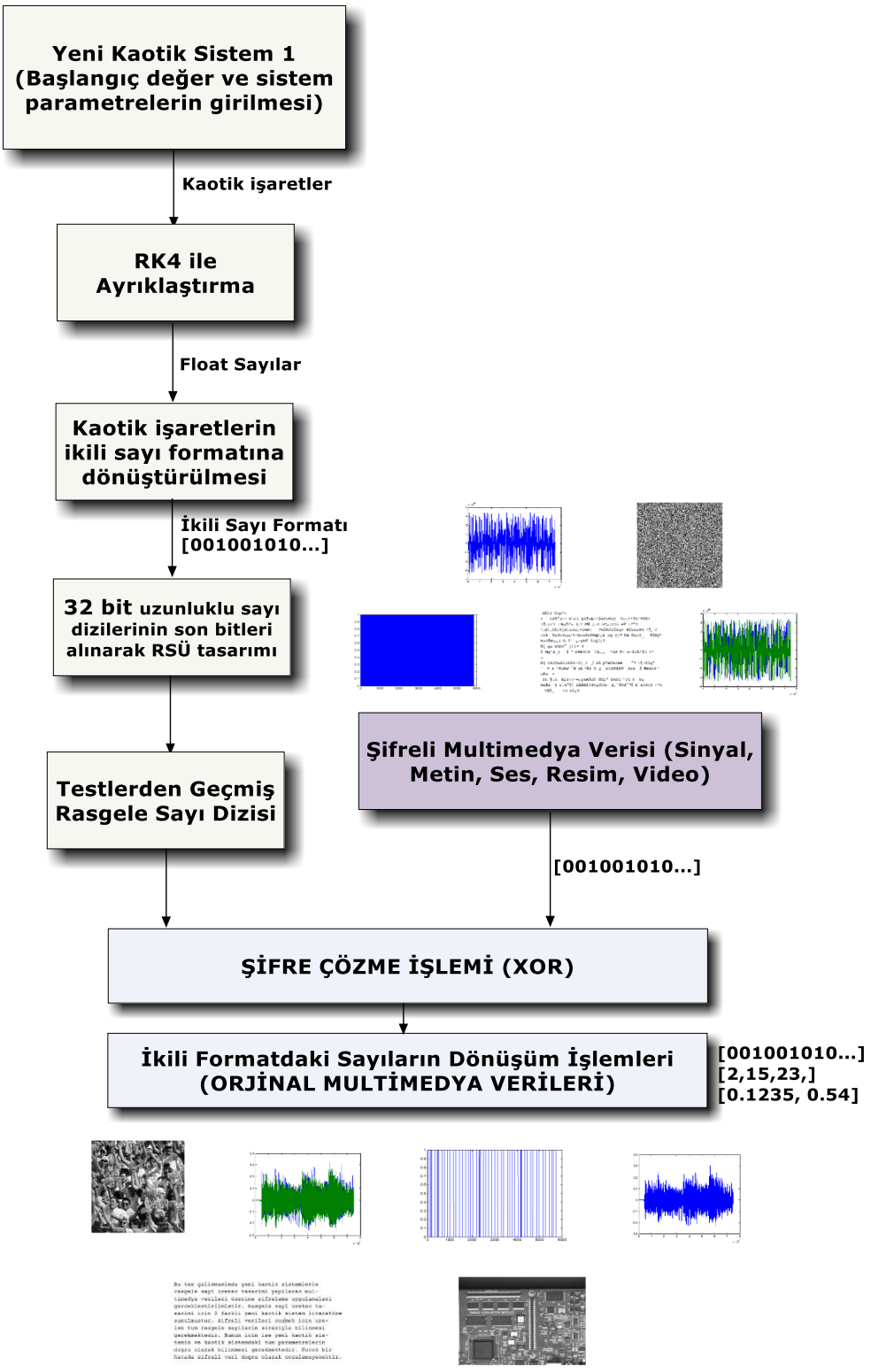
Şekil 6.1. Şifreleme blok diyagramı

Şifreli verilerin çözülebilmesi için, şifreleme işleminde kullanılan anahtarların (rasgele sayılar) tekrar üretilmesi, yani elde edilerek bilinmesi gerekmektedir. Bunun için ise, yeni kaotik sistem 1'in, denklemlerdeki başlangıç değerleri ve parametrelerin (a=2.8, b=0.2, c=1.4, d=1, e =10, f=2) doğru olarak bilinmesi gerekmektedir. Ayrıca RSÜ tasarımı için, nasıl bir yol izlenmişse, yani "0" ve "1'ler neye göre seçilerek rasgele sayılar üretilmişse, bu adımların bilinmesi gerekmektedir. Aksi takdirde şifrelenmiş veriler çözülemeyecektir. Literatürde var olan kaotik sistemlerden farklı bir kaotik sistemin bulunarak, şifreleme işlemlerinde kullanılması, şifrelenmiş verilerin çözülmesini daha da zorlaştırmıştır. Kaotik sistemler çok hassas olduklarından küçük bir hatada, orjinal veriler elde edilemeyecektir.

Şekil 6.2'de, şifre çözme işlemini gösteren blok diyagram verilmiştir. Rasgele sayıların üretilme bölümü, şifreleme işleminde gerçekleştirildiği gibidir. Elde edilen rasgele sayılar ile, şifrelenmiş multimedia verileri tekrar aynı işlemler ile çözülebilecektir. Şifreli verilerin çözülmesi sonucu ikili sayı formatında veriler elde edilmiş olacaktır. Çözülmüş verilerin, şifrelenmeden önceki durumlarının görülebilmeleri için, tekrar dönüşüm işlemlerinin yapılması gerekmektedir. Örneğin ses verisi ilk olarak float sayı tipinde olduğuı için, şifre çözme işlemi sonucunda, ikili sayı formatındaki verilerin, orjinal verinin elde edilebilmesi için float sayı formatına çevrilmeleri gerekmektedir.

İlk aşamada yeni kaotik sistem 1'de bulunan başlangıç değerleri ve parametreler şifrelenmemiştir. Anahtar dağıtımı gibi problemlerin olduğu yerlerde, yeni kaotik sistem 1'deki parametreler ve başlangıç değerleri, diğer yeni kaotik sistem 2 ve RSA asimetrik şifreleme algoritması ile şifrelenecektir. Bu parametreler ve başlangıç değerlerinin şifrelenmesine yönelik çalışmalar bir sonraki bölümde anlatılacaktır. Sinyal, metin, ses (mono ve stereo), resim ve video multimedia verilerinin, blok diyagramlara göre şifreleme ve şifre çözme işlemleri sırasıyla gerçekleştirilmiştir.

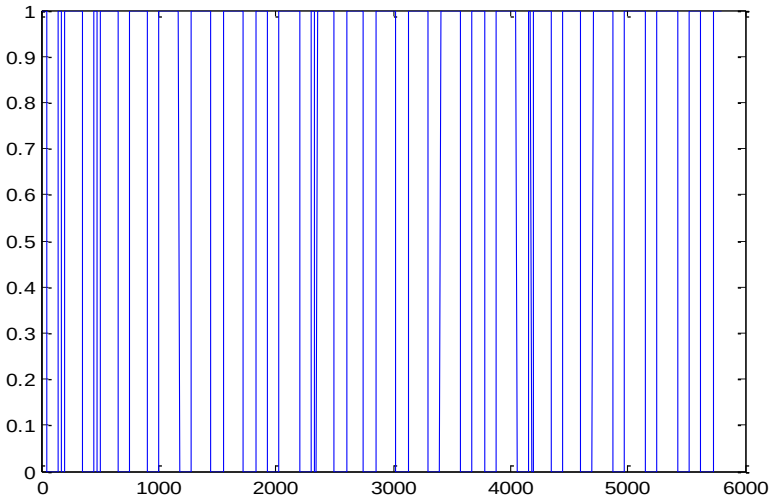
$$\dot{x} = ay - x + zy$$
$$\dot{y} = -bxz - cx + yz + d$$
$$\dot{z} = e - fxy - x^2$$



Şekil 6.2. Şifre çözme işlemi blok diyagramı

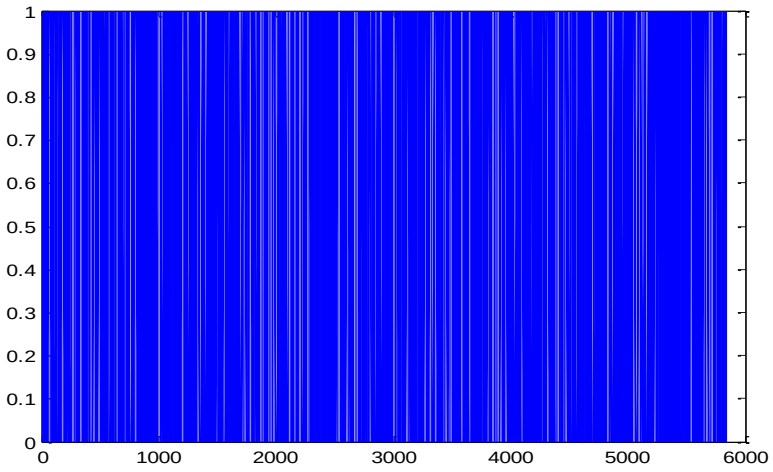
6.1.1. Sinyal şifreleme uygulaması

Bu bölümde, yeni kaotik sistem 1 kullanılarak tasarlanan RSÜ'leri ile "0" ve "1" lerden oluşan sayısal sinyallere yönelik şifreleme işlemleri gerçekleştirilmiştir. Uygulama için Şekil 6.3'de görülen 5755 bitlik veri dizisi kullanılmıştır.



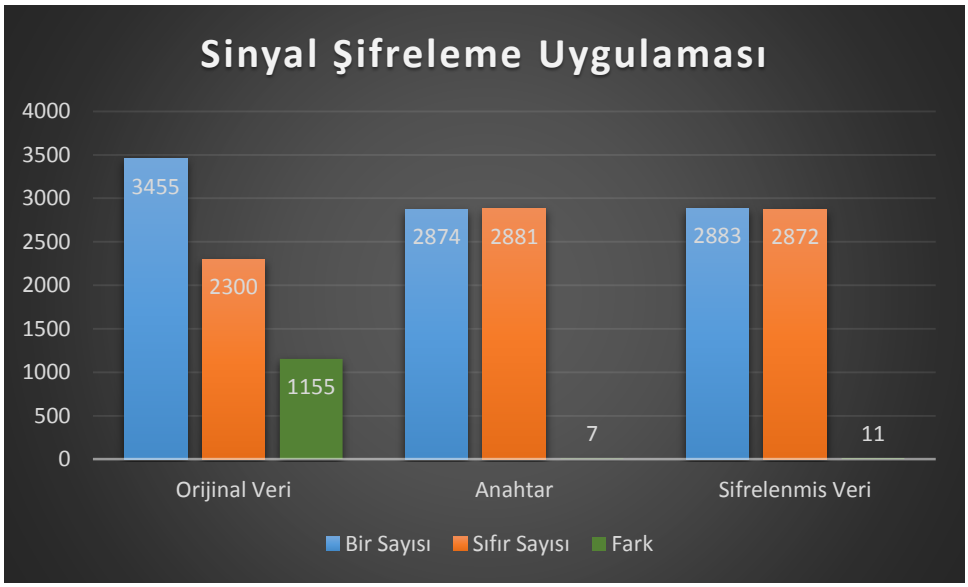
Şekil 6.3. 0 ve 1'lerden oluşan 5755 bit şifrelenecek veri dizisi

Şekil 6.3'deki 5755 bitlik sayı dizisinde, 3455 tane "1", 2300 tane "0" bulunmaktadır. Bu sayı dizileri tasarlanan RSÜler ile XOR mantıksal operatörü yardımıyla şifreleme işlemine tabi tutulmuştur. Şifreleme sonucu elde edilen 5755 bitlik şifrelenmiş veri Şekil 6.4'de görüldüğü gibidir.



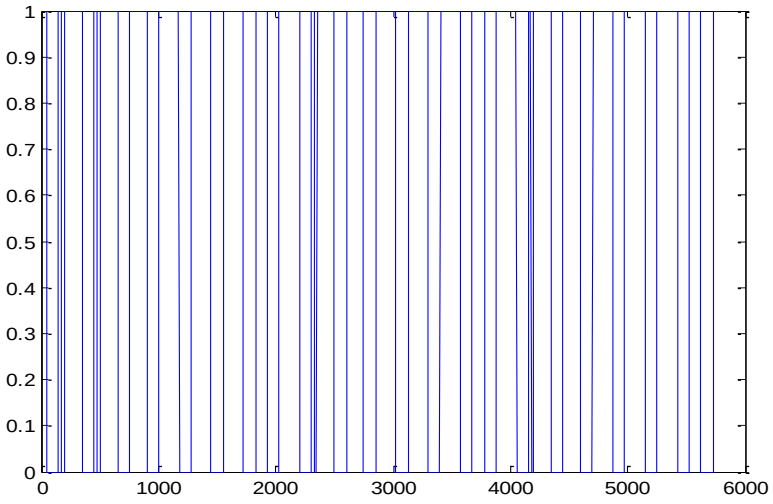
Şekil 6.4. 0 ve 1'lerden oluşan 5755 bit şifrelenmiş veri dizisi

Şifreleme işlemi sonucu; 2883 adet “1”, 2872 adet “0” üretilerek aradaki farkın 1155’den, 11’e indiği görülmüştür. Şifrelenecek veri, anahtar ve şifrelenmiş verilerdeki “1” ve “0’ların sayısı grafik 6.5’de verildiği gibidir. Aradaki farkın az olması, şifreleme işlemi sonucunun oldukça iyi olduğunu göstermektedir.



Şekil 6.5. 5755 bitlik veri dizisinin orijinal, anahtar ve şifrelenmiş veri üzerindeki 0 ve 1’lerin dağılımı

Şifrelenmiş verilerin çözülmüş hali ise Şekil 6.6’da verildiği gibidir.



Şekil 6.6. 0 ve 1’lerden oluşan 5755 bit çözülmüş veri dizisi

Şifreleme ve şifre çözme işlemlerinde basit mantıksal operatörler (XOR) kullanıldığı için bu işlemler için harcanan süreler çok kısadır. Şifreleme için harcanan süre 16.7 ms, şifre çözme için harcanan süre ise 5.8 ms'dir. Basit işlemler kullanılarak çok az sürede başarılı şifreleme işleminin gerçekleştirilebilmesinin ana nedeni yeni kaotik sistem 1 ile tasarlanan FIPS-140-1 ve NIST-800-22 testlerini geçmiş RSÜlerinin kullanımasıdır denilebilir.

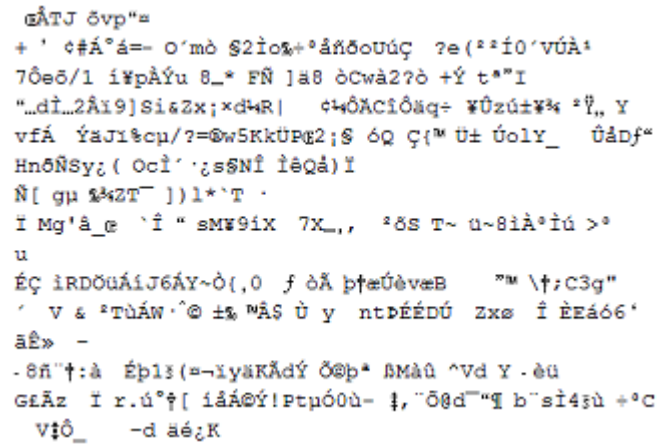
6.1.2. Metin şifreleme uygulaması

Yeni kaotik sistem 1'i kullanılarak tasarlanan RSÜ ile metin şifreleme işleminin gerçekleştirilebilmesi için aşağıdaki paragraf seçilmiştir. Şifreleme uygulamasında karakterler editörde gösterilemediği için orjinal, şifreli ve tekrar elde edilmiş paragraf yazısı resim olarak kaydedilmiştir. Şekil 6.7'de şifrelenmek istenen paragraf yazısı gösterilmiştir. Paragraf, boşluklarda dahil olmak üzere toplam 518 karakterden meydana gelmektedir.

```
Bu tez çalismasinda yeni kaotik sistemlerle
rasgele sayi üretec tasarimi yapilarak mul-
timedya verileri üzerine sifreleme uygulamalari
gerceklestirilmistir. Rasgele sayi üretec ta-
sarimi icin 2 farkli yeni kaotik sistem literatüre
sunulmustur. Sifreli verileri cozmek icin ure-
len tum rasgele sayilarin sirasiyla bilinmesi
gerekmektedir. Bunun icin ise yeni kaotik sis-
temin ve kaotik sistemdeki tum parametrelerin
dogru olarak bilinmesi gerekmektedir. Kucuk bir
hatada sifreli veri dogru olarak cozulemeyecektir.
```

Şekil 6.7. Şifrelenecek paragraf

Char veri tipi formatında olan karakterler öncelikle double veri türüne, daha sonra ise ikili sayı formatına çevrilmiştir. İkili sayı formatına çevrilen karakterleri şifrelemek için 518*8=4144 tane RSÜ'den üretilen sayı dizisine ihtiyaç duyulmaktadır. RSÜ'den gelen 4144 tane, "0" ve "1'lerden oluşan sayı dizisi, ikili sayı formatına dönüştürülmüş karakterler ile şifrelenmiştir. Şifrelenen karakter dizisi daha sonra tekrardan char veri türüne çevrilmiş ve Şekil 6.8'de görüldüğü gibi şifrelenmiş paragraf elde edilmiştir. Her bir karakter için, bir şifrelenmiş karakter elde edilmiştir.



Şekil 6.8. Şifrelenmiş paragraf

Şifrelenmiş metin verilerinin çözülmüş gösterimleri ise Şekil 6.9'da verildiği gibidir. Şekil 6.7 ve Şekil 6.9'dan görüldüğü üzere orjinal veri ile çözülmüş veri arasında hiçbir fark bulunmamaktadır. 4144 sayı dizisini XOR işlemi ile şifrelemek için harcanan süre 878 us iken çözmek için harcanan süre ise 158 us'dir. Bu kadar kısa sürede şifrelemenin olmasının nedeni, şifreleme işleminin yine basit XOR mantıksal operatörü gerçekleştirilmesinden dolayıdır. Şifrelenmiş veriler üzerindeki karmaşıklığın nedeni ise şifreleme işlemlerinde uluslararası geçerliliğe sahip olan, kabul görmüş FIPS-140-1 ve NIST-800-22 testlerinden geçmiş RSÜ sayı dizisinin kullanılmasından kaynaklanmaktadır.

```
Bu tez çalismasinda yeni kaotik sistemlerle
rasgele sayi üretec tasarimi yapilarak mul-
timedya verileri üzerine sifreleme uygulamalari
gerceklestirilmistir. Rasgele sayi üretec ta-
sarimi icin 2 farkli yeni kaotik sistem literatüre
sunulmustur. Sifreli verileri cozmek icin üre-
len tum rasgele sayilarin sirasiyla bilinmesi
gerekmektedir. Bunun icin ise yeni kaotik sis-
temin ve kaotik sistemdeki tum parametrelerin
dogru olarak bilinmesi gerekmektedir. Kucuk bir
hatada sifreli veri dogru olarak cozulemeyecektir.
```

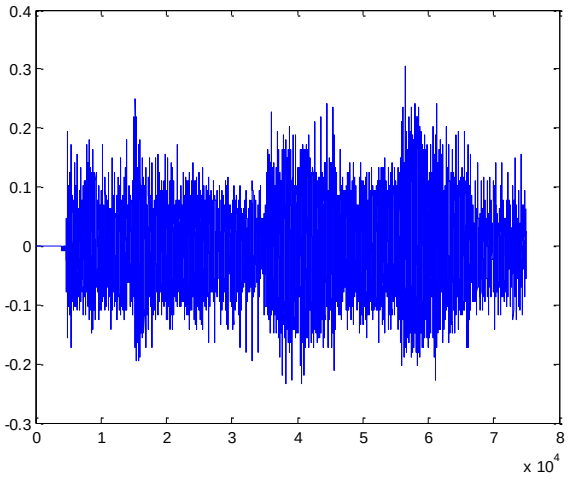
Şekil 6.9. Çözülmüş paragraf

6.1.3. Ses şifreleme uygulaması

Bu bölümde yeni kaotik sistem 1 kullanılarak mono ve stereo olarak 2 farklı ses dosyasının şifreleme uygulaması gerçekleştirilmiştir. Mono ses türü, tek kanal sinyal iken, stereo iki kanal sinyali ifade etmektedir. Yani iki hoparlör olduğu düşünülürse stereo seste, iki hoparlörden de farklı ve daha kaliteli ses duyulabilmektedir. Stereo ses için iki farklı ses çıkışı olmalıdır. İlerleyen bölümlerde de görüleceği üzere mono ses de tek sinyal bulunurken, stereo ses de iki farklı sinyal bulunmaktadır.

6.1.3.1. Mono ses şifreleme uygulaması

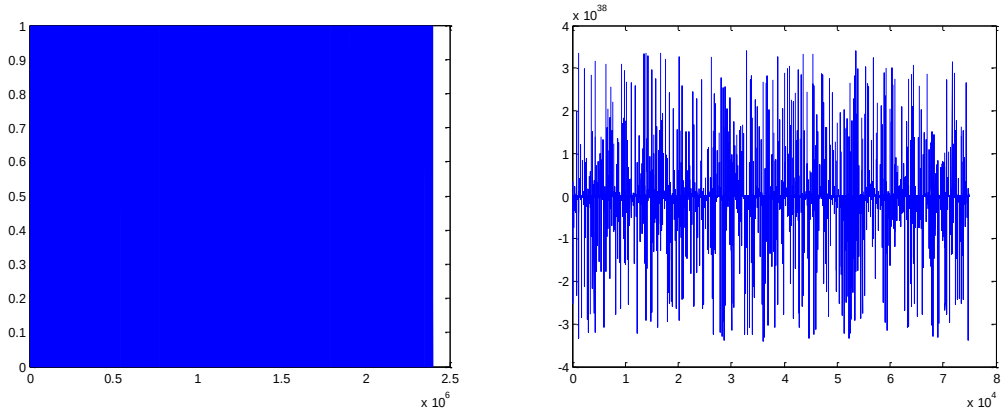
Tek (mono) kanal ses şifreleme uygulaması için Şekil 6.10'da görülen 75001 bitlik ses sinyali kullanılmıştır. Bit dizisi float sayılardan oluştuğu için, şifreleme işleminde XOR işlemine tabi tutularak RSÜ ile şifrelenebilmesi için öncelikle, ses sinyalinin ikili (binary) sayı formatına çevrilmesi gerekmektedir.



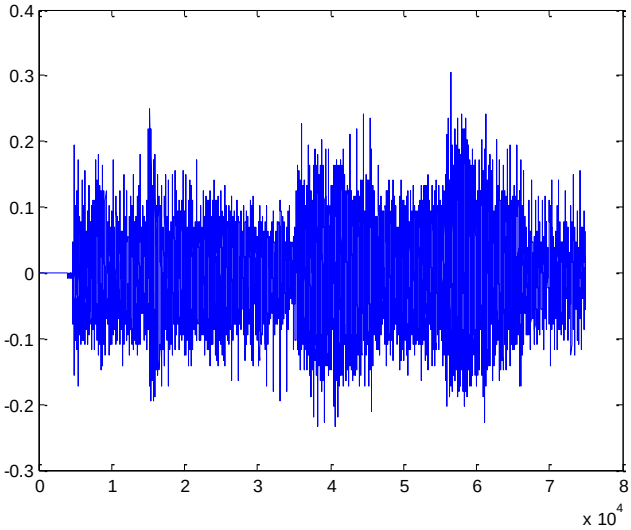
Şekil 6.10. Mono orjinal ses sinyali

75001 bitlik float formatındaki ses sinyali ikili sayı dizisine çevrilirken, her bit 32 bitlik "0" ve "1"lerden oluşan sayılara dönüşmektedir. Dolayısıyla çevrim işlemi sonucu şifrelenecek bit sayısı 75001*32'den 2400032'ye çıkmaktadır. 2400032 bit dizisi RSÜ'den gelen 2400032 rasgele sayı ile şifrelenecektir. Şifreleme işlemi sonrası ikili ve float sayı formatındaki, şifrelenmiş verilerin gösterimi Şekil 6.11'de olduğu gibidir. Şifrelenmiş ses

sinyali ikili formatında 2400032 bit dizisi varken, float formatda 75001 bit dizisi bulunmaktadır. Orjinal sinyal ile şifrelenmiş ses sinyalleri görsel olarak incelendiğinde ve işitsel olarak dinlendiğinde herhangi bir bağlantı kurulamamaktadır.



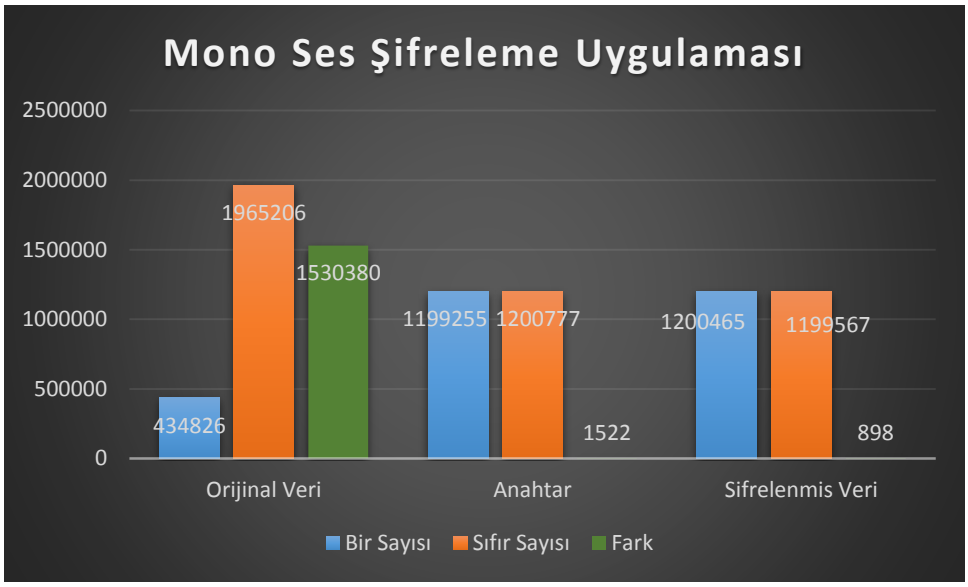
Şekil 6.11. Şifrelenmiş tek kanal ses sinyali (soldaki=ikili gösterim, sağdaki=float gösterim)



Şekil 6.12. Çözülmüş mono orjinal ses sinyali

Şekil 6.11'deki şifrelenmiş verinin çözülmüş float biçimindeki hali Şekil 6.12'de görüldüğü gibidir. İkili sayı formatında olan şifrelenmiş veriler, XOR işlemi ile çözüldüklerinde yine ikili sayı formatında olacaktır. Orjinal ses sinyalini elde edebilmek için ikili sayı formatındaki verilerin en başta olduğu gibi float sayı formatına çevrilmesine gerekmektedir. Bu uygulamada şifreleme için harcanan süre 0.0601 sn iken, şifre çözme işlemi için harcanan

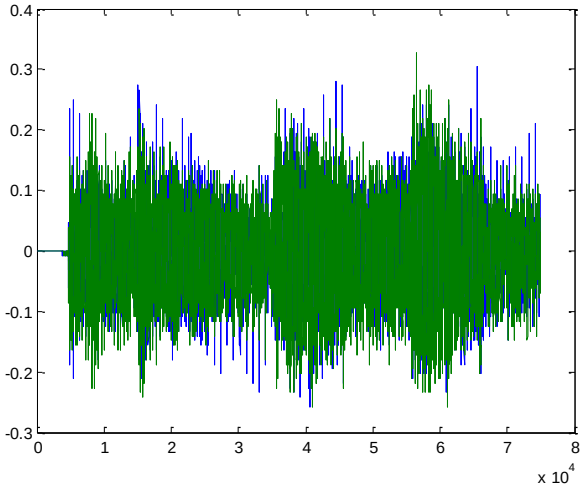
süre 0.0614 süredir. Orjinal veri, anahtar ve şifreli veriler üzerindeki “0” ve “1”lerin dağılımını gösteren grafik ise Şekil 6.13’de verilmiştir. Grafikten de görüldüğü üzere veriler şifrelendikten sonra “0” ve “1”ler arasındaki fark oldukça azalmıştır. Bu farkın olabildiğince az olması, şifreleme işleminin başarılı bir şekilde gerçekleştirilmiş olduğunu göstermektedir.



Şekil 6.13. 2400032 bitlik veri dizisinin orijinal, anahtar ve şifrelenmiş veri üzerindeki 0 ve 1’lerin dağılımı

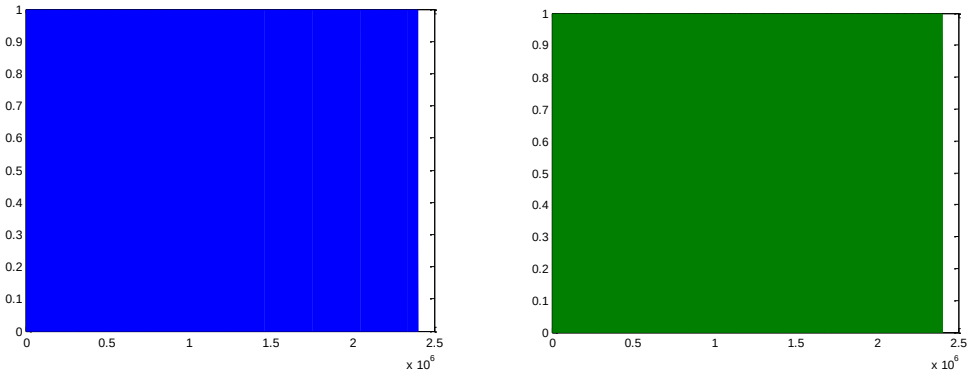
6.1.3.2. Stereo ses şifreleme uygulaması

Stereo (iki kanal) ses sinyali mono ses sinyalinden farklı olarak iki ayrı sinyalden oluşmaktadır. Şekil 6.14’de şifreleme işleminde kullanılacak örnek bir stereo ses sinyali verilmiştir. Görüldüğü üzere stereo ses sinyali mavi ve yeşil olarak iki ayrı sinyalden oluşmaktadır. Şifreleme işlemi için mono ses sinyalinde 75001’lik tek sinyal kullanılırken, stereo ses sinyali için Şekil 6.14’deki 75001 bitlik iki farklı ses sinyal kullanılmıştır.



Şekil 6.14. Stereo orjinal ses sinyali

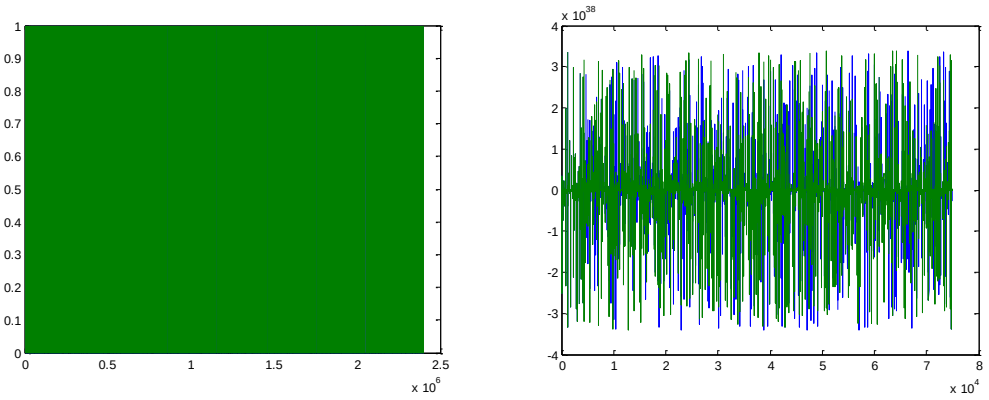
Stereo ses şifreleme uygulamasında, mono ses şifrelemesindeki uygulamaya benzer işlemler gerçekleştirilmiştir. Şifreleme işlemi için öncelikle 75001x2 float türündeki ses sinyali ikili sayı formatına çevrilerek 2400032x2'lik bit dizisi elde edilmiştir. Elde edilen bit dizileri RSÜ ile şifreleme işlemine tabi tutulduğunda Şekil 6.15 ve Şekil 6.16'daki şifrelenmiş veriler elde edilmiştir.



Şekil 6.15. Stereo şifrelenmiş ses sinyali (ayrı olarak)

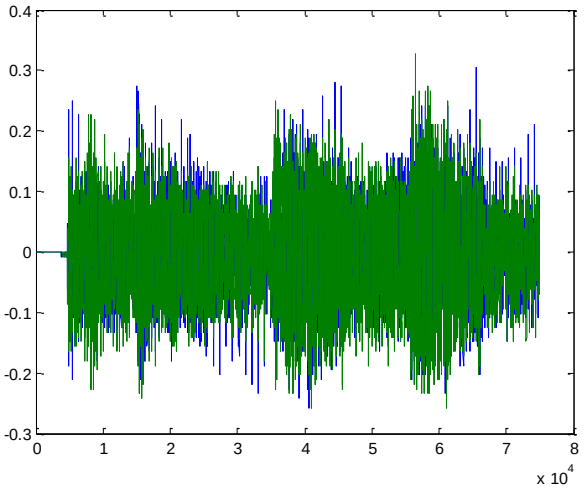
Şekil 6.15'de stereo sinyaller ayrı olarak iki farklı şekilde gösterilmiştir. Şekil 6.15'in sol tarafı, Şekil 6.14'deki orjinal ses sinyalinin mavi olan ikili formatdaki şifrelenmiş halini, sağ tarafı ise yeşil olan sinyalin ikili formatdaki şifrelenmiş halini göstermektedir. Şekil 6.16'da ise şifrelenmiş sinyaller (mavi ve yeşil) bir arada gösterilmiştir. Sol taraftaki gösterim ikili format, sağ taraftaki gösterim ise ikili formatdaki sinyallerin floata çevrilmiş durumlarını

göstermektedir. Sol tarafta, yeşil sinyal baskın gözüktüğü için, şekilde tek sinyal gibi görünmektedir.



Şekil 6.16. Şifrelenmiş iki kanal ses sinyali (soldaki=ikili gösterim, sağdaki=float gösterim)

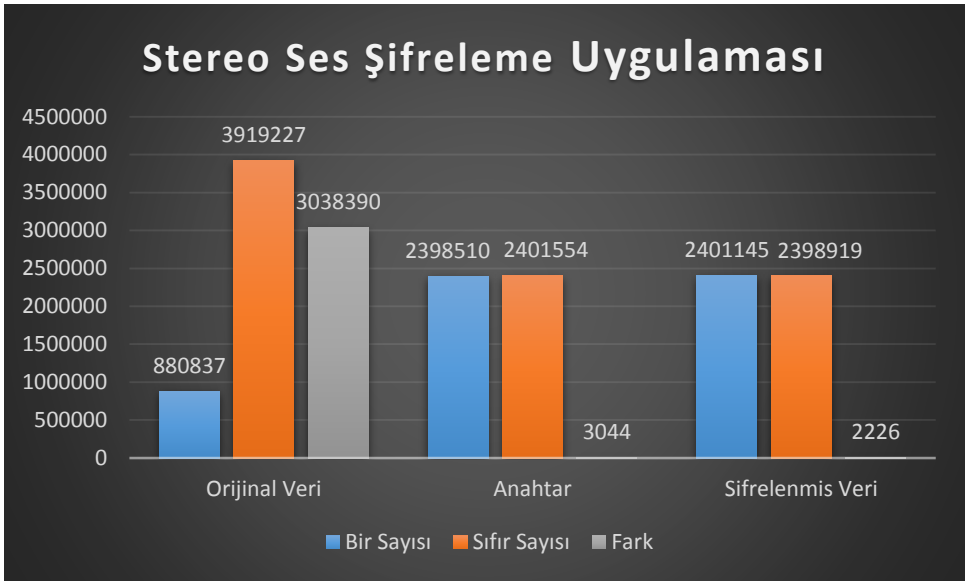
Şekil 6.16'daki şifrelenmiş verinin çözülmüş float biçimindeki hali Şekil 6.17'de görüldüğü gibidir. Orjinal ses sinyalini elde edebilmek için, mono ses şifrelemede olduğu gibi ikili sayı formatındaki verilerin float sayı formatına çevrilmesine gerekmektedir. Bu uygulamada şifreleme için harcanan süre 0.1347 sn iken, şifre çözme işlemi için harcanan süre 0.1442 süredir. Uygulamada iki ayrı ses sinyali olduğu için, mono ses sinyalindeki sürelerin yaklaşık olarak iki katı değerler elde edilmiştir.



Şekil 6.17. Çözülmüş stereo orjinal ses sinyali

Orjinal veri, anahtar ve şifreli veriler üzerindeki "0" ve "1"lerin dağılımını gösteren grafik ise Şekil 6.18'de verilmiştir. İki ses sinyali bulunduğu için değerler iki kat artmıştır.

Grafiktende görüldüğü üzere veriler şifrelendikten sonra “0” ve “1”ler arasındaki fark yine en aza inmiş ve başarılı bir şifreleme işlemi gerçekleştirilmiştir.



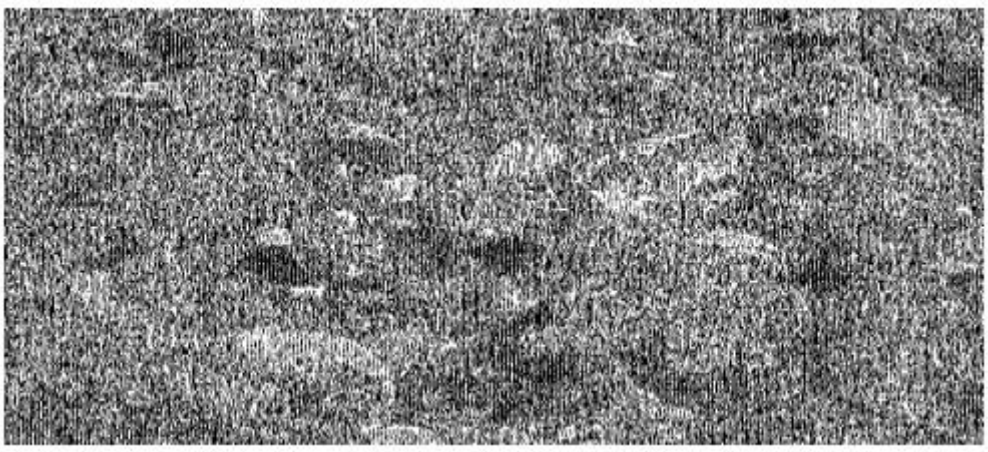
Şekil 6.18. 2400032x2 bitlik veri dizisinin orijinal, anahtar ve şifrelenmiş veri üzerindeki 0 ve 1’lerin dağılımı

6.1.4. Resim şifreleme uygulaması

Bu bölümde, Şekil 6.19’daki 256*256 matris formdaki bir resim verisinin, yeni kaotik sistem 1 kullanılarak elde edilen RSÜ’leri ile şifreleme uygulama işlemleri gerçekleştirilmiştir.

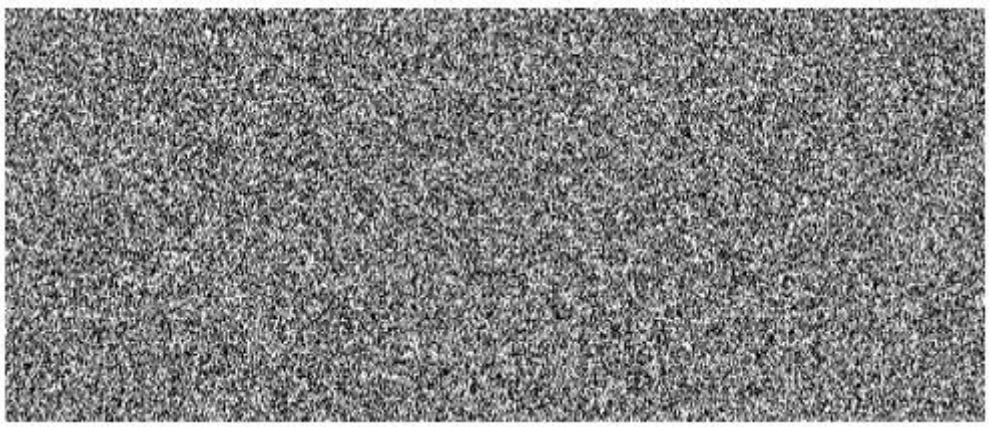
Şekil 6.19. Orjinal resim verisi

0-256 arası piksel değerleri olan resim verisinin, RSÜ ile şifreleyebilmek için onlu sayı formatından, ikili sayı formatına çevrilmesi gerekmektedir. 256*256 lık matrisden oluşan resim verisi, Şekil 6.20'de görüldüğü gibi "0" ve "1'lerden oluşan 2048*256'lık matris formatına dönüştürülmüştür.



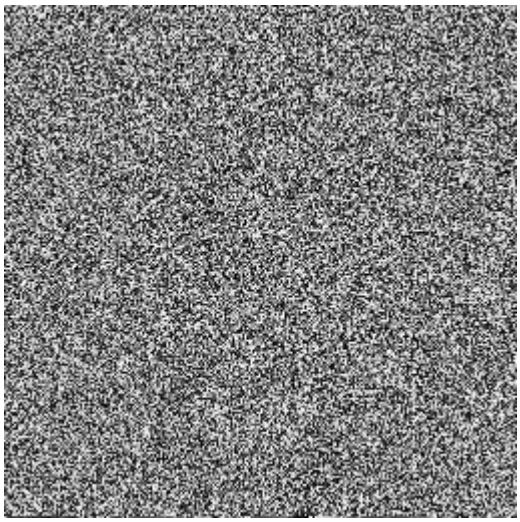
Şekil 6.20. İkili sayı formatına dönüştürülmüş orjinal resim verisi

2048*256'lık bir resim verisini şifrelemek için, RSÜ'inden elde edilen 2048*256'lı bir rasgele sayı dizisi gerekmektedir. Bunun için Şekil 6.21'deki, RSÜ'den elde edilen rasgele "0" ve "1'lerden meydana gelen, resim verisini şifrelemek için gerekli veri dizisi oluşturulmuştur.



Şekil 6.21. İkili sayı formatına dönüştürülmüş orjinal resim verisini şifrelemek için RSÜ'den oluşturulan veriler

2048*256'lık matris formunda oluşturulan şifrelenecek resim verisi ve RSÜlerden elde edilen rasgele sayılar, XOR işlemiyle şifreleme işlemine tabi tutulmuştur. Şifreleme işlemi elde edilen ikili sayı formundaki 2048*256'lık matris verisinin onluk sayı sistemine çevrilmiş, yani RSÜ'lerle şifrelenemiş resim verisi Şekil 6.22'de görüldüğü gibidir.



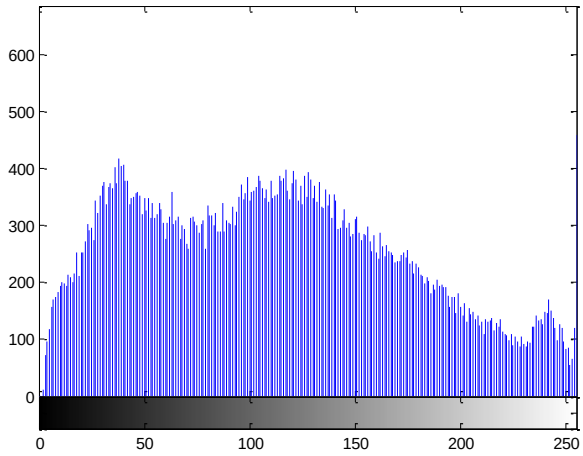
Şekil 6.22. Tasarlanan RSÜ ile şifrelenmiş resim verisi

Şifrelenmiş resim verisinin, çözülerek elde edilmiş hali ise Şekil 6.23'de verildiği gibidir. Şekil 6.19'daki orjinal resim ile Şekil 6.23'deki çözülmüş resim karşılaştırıldığında herhangi bir bozulma olmamıştır.

Şekil 6.23. Çözülmüş resim verisi

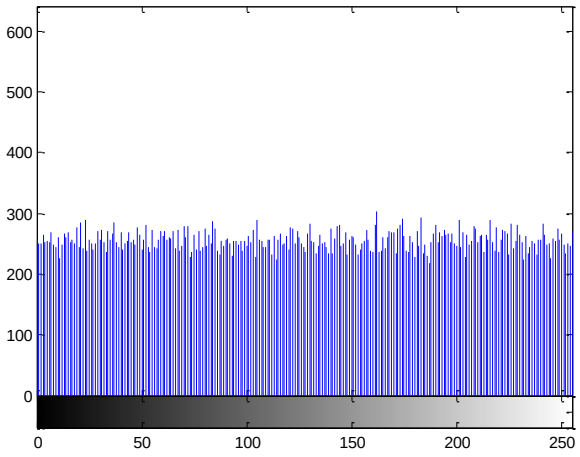
Gerçekleştirilen şifreleme ve şifre çözme işlemlerinde, şifreleme işlemi için 0.5167 sn, şifre çözme işlemi için ise 0.5293 sn harcanmıştır. Şimdiye kadar yapılan çoklu-ortam verilerinde yapılan şifreleme işlemlerinde olduğu gibi, resim verisi üzerinde gerçekleştirilen, şifreleme ve şifre çözme işlem süreleri de oldukça kısadır.

Yapılan resim şifreleme işleminde, güvenlik analizlerinden öncelikle histogram analizi gerçekleştirilmiştir. Şekil 6.24'de orjinal verinin histogramı görülmektedir. Şekil 6.25'de ise şifrelenmiş verinin histogramı gösterilmiştir.



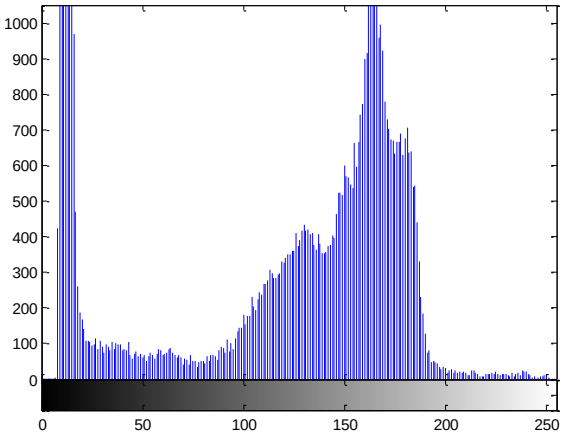
Şekil 6.24. Orjinal resim verisinin histogramı

Histogram analizinde verilerin birbirine yakın olarak dağılmış olması, kriptoanalizciler için şifreli verinin çözülmesini zorlaştıracaktır. Tüm değerler birbirine yakın olduğu için, şifreli verinin histogramından bir çıkarım yapmak mümkün olmayacaktır.



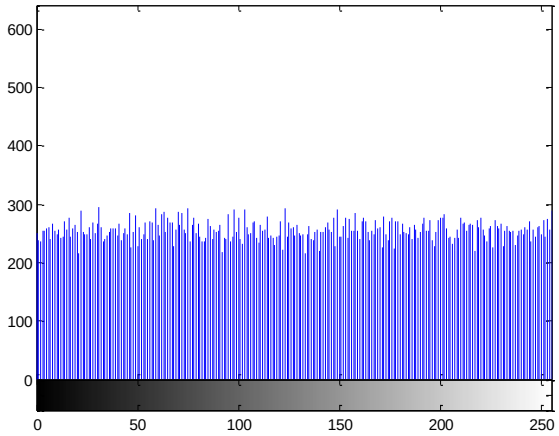
Şekil 6.25. Şifrelenmiş resim verisinin histogramı

Histogram değeri Şekil 6.26'daki gibi olan başka bir resim verisi için, şifreleme işlemi sonucu elde edilen histogram değeri ise Şekil 6.27'de görüldüğü gibidir.



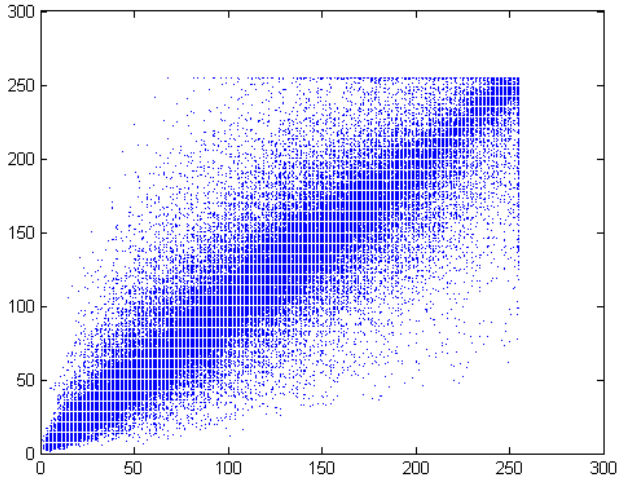
Şekil 6.26. Farklı bir orjinal resim verisinin histogramı

Resim verisi farklı iken, orjinal verinin histogramı ne kadar dağınık olsada, şifreleme işlemi sonucu elde edilen başarım benzer olarak elde edilmektedir. Şekil 6.27'de de görüldüğü üzere farklı bir şifrelenmiş resmin histogramı da, oldukça düzgün olarak dağılmıştır.



Şekil 6.27. Farklı bir şifrelenmiş resim verisinin histogramı

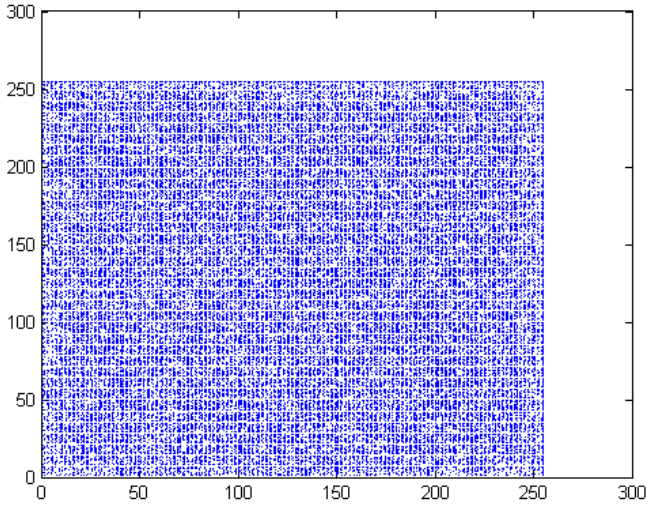
Güvenlik analizleri için gerçekleştirilen diğer bir analiz yöntemi ise korelayon analizidir. Bölüm 2.5.1'de detaylı olarak anlatılan korelasyon analizi için değişkenler arası ilişkinin doğrusal olması gerekmektedir. Eğer doğrusallık yoksa bu analizi gerçekleştirmek mümkün olmayacaktır. Şekil 6.28'de orjinal resim verisinin, Şekil 6.29'da ise şifrelenmiş resim verisinin korelasyon dağılımı gösterilmiştir.



Şekil 6.28. Orjinal resim verisinin korelasyon dağılımı

Şekil 6.28'de bir doğrusallık mevcut iken, Şekil 6.29'daki şifrelenmiş verinin korelasyonundan böyle çıkarım yapmak mümkün değildir. Doğrusallık olmadığı için,

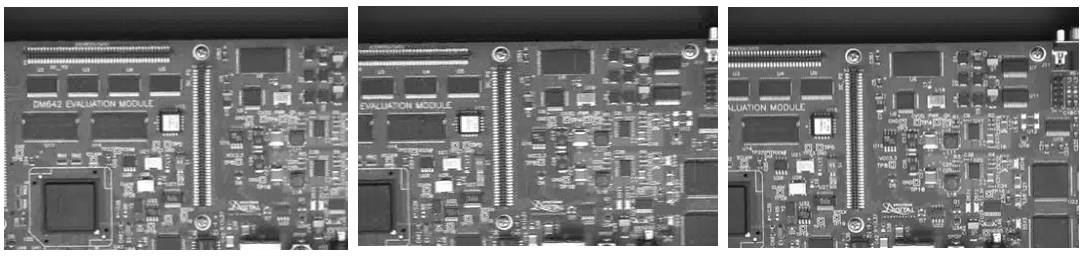
korelasyon analizi yapmak mümkün olmayacak, dolayısıyla bu istatistiksel analize kapalı olmuş olacaktır. Korelasyon değeri ise 0.9259'den, 0.5210' ya düşmüştür.



Şekil 6.29. Şifrelenmiş resim verisinin korelasyon dağılımı

6.1.5. Video şifreleme uygulaması

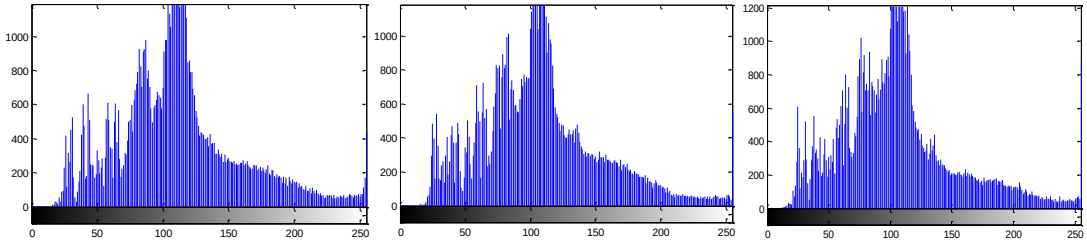
Video şifreleme uygulaması için her bir karesi 240*360 bir matrislik, toplam 340 kareden oluşan bir video kullanılmıştır. Şekil 6.30'da bu videonun 1, 20 ve 30. kareleri verilerek, şifreleme işlemleri gerçekleştirilmiştir.



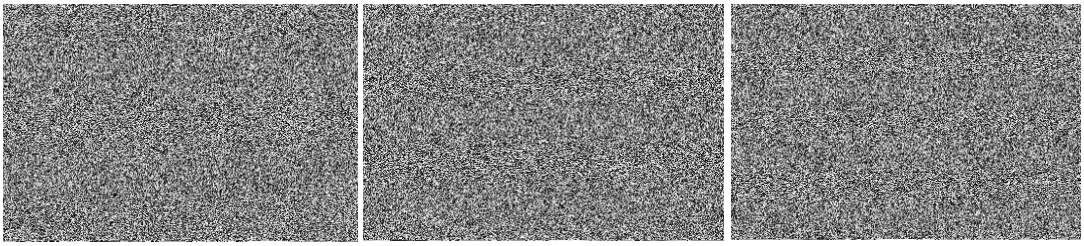
Şekil 6.30. Orjinal videonun 1, 20 ve 30. kareleri

Video şifreleme uygulaması, resim şifrelemesi uygulamasında olduğu gibi gerçekleştirilmektedir. 240*360 lık matrisdeki onlu sayı tabanındaki değerler, 240*2880'lik matris formatında ikili sayı tabanına çevrilerek, rasgele sayı üreteçleri ile şifrelenmektedir. Bu uygulamada gerçekleştirilen video da 340 kare olduğu için aynı işlemler 340 kere

gerçekleştirilmiştir. İlgili videodaki şifrelenmemiş bazı resim karelerinin histogram dağılımları Şekil 6.31'de verilmiştir.

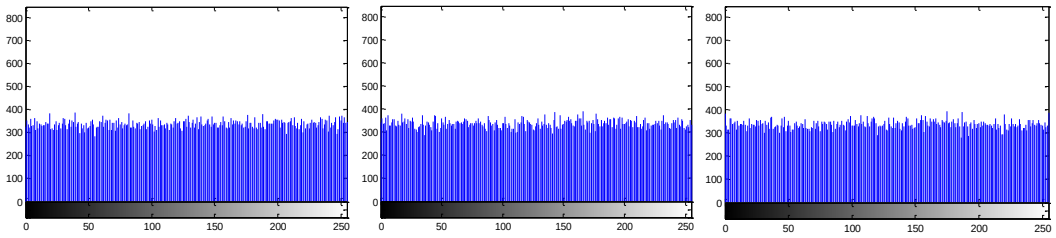


Şekil 6.31. Orjinal videonun 1, 20 ve 30. karelerinin histogram dağılımları



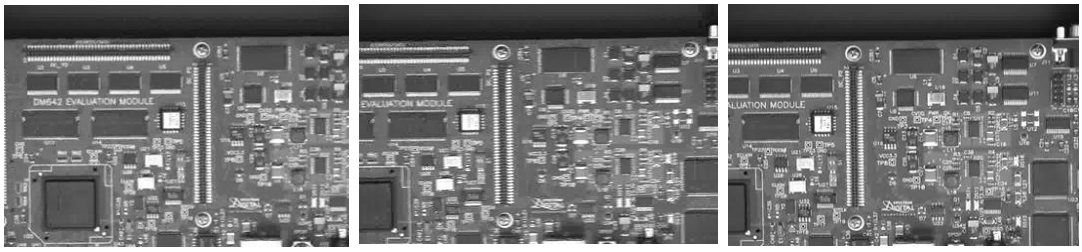
Şekil 6.32. Şifrelenmiş videonun 1, 20 ve 30. kareleri

Şekil 6.30'da verilen videonun 1, 20 ve 30. karelerinin XOR işlemiyle şifrelenmiş görünümleri ve histogram değerleri Şekil 6.32 ve Şekil 6.33'de görüldüğü gibidir. Histogram değerlerinden görüldüğü üzere, şifreleme işleminin her kare için başarılı bir şekilde gerçekleştiği görülmektedir.



Şekil 6.33. Şifrelenmiş videonun 1, 20 ve 30. karelerinin histogram dağılımları

Şekil 6.34'de şifrelenmiş video karelerinin, çözülmüş görünümleri verilmiştir. Orjinal veri ile çözülmüş veri arasında herhangi bir bozulma meydana gelmemiştir. Şifreleme işlemi için harcanan süre yaklaşık olarak 174,35 sn iken, şifre çözme işlemi için harcanan süre ise 197,302 sn'dir. Resim verisi üzerinde diğer güvenlik analizleri gösterildiği için bu bölümde sadece histogram analizi gerçekleştirilmiştir.



Şekil 6.34. Çözülmüş videonun 1, 20 ve 30. kareleri

6.1.6. Diğer güvenlik analizleri

Sadece kaotik sistemin özelliklerinden dolayı tüm uygulamalar için bazı güvenlik analizleri aynıdır. Her uygulama için aynı olan güvenlik analizleri, tekrarlar olmaması açısından bu bölümde bahsedilmiştir. Bu analizler anahtar uzunluk, anahtar duyarlılık ve kaos etkisi analizleridir.

6.1.6.1. Anahtar uzunluk analizi

Saldırılara karşı anahtar boyutlarının olabildiğince uzun olması gerekmektedir. Kaotik sistemlerde boyut, başlangıç değerleri ve sistem parametreleri arttıkça anahtar uzunluğu da artacaktır. Sadece bir değişkenin olduğu durumlarda anahtarlar, 10^{14} farklı değer alabilmektedir.

$$\begin{aligned} \dot{x} &= ay - x + zy \\ \dot{y} &= -bxz - cx + yz + d \\ \dot{z} &= e - fxy - x^2 \end{aligned} \tag{6.2}$$

Yeni kaotik sistem 1 (Denklem 6.2), anahtar uzunluk değeri olarak analiz edilirse, 3 boyutlu sürekli bir kaotik sistem olduğu için 3 (x,y ve z) 10^{42} ve 6 farklı parametreye (a,b,c,d,e ve f) sahip olduğu için 10^{84},yani toplamda 10^{126} anahtar uzunluğuna sahiptir denilebilir. Yeni kaotik sistem 1 'in, çok fazla parametre içermesi, sistemin analizini zorlaştırdığı için, şifreli verilerin çözülmesini de zorlaştırmış olacaktır. Anahtar uzunluğu, oldukça fazla olduğu için yeni kaotik sistem 1 kullanarak yapılan şifreleme işlemleri güvenilir olacaktır. Tüm

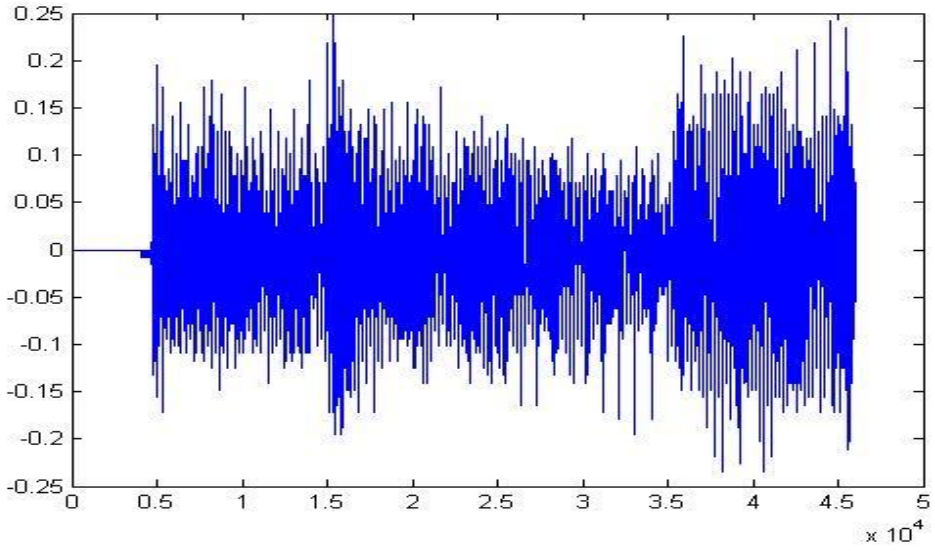
multimedya verilerini şifreleme işlemlerinde aynı kaotik sistem kullanıldığı için, anahtar uzunluk değerleri de aynı olmuş olacaktır.

6.1.6.2. Anahtar duyarlılık analizi

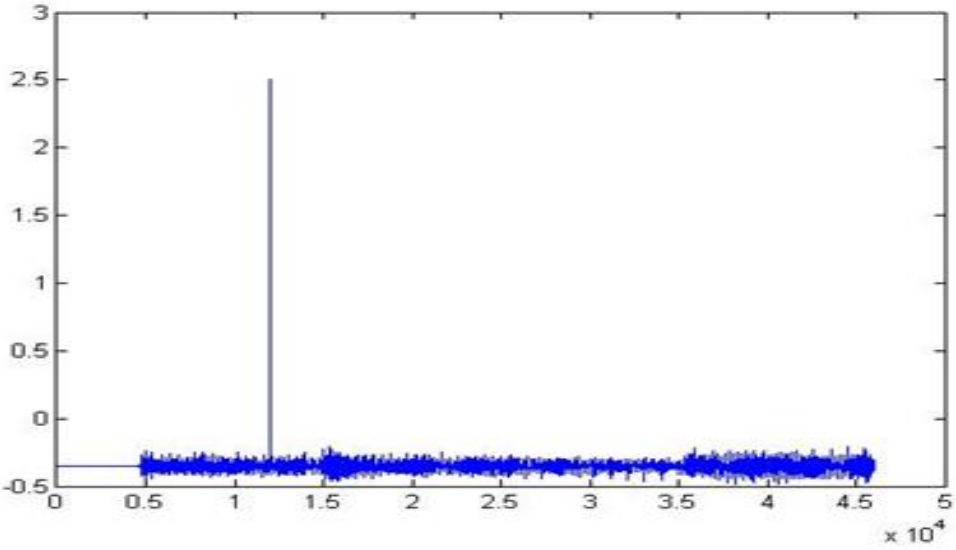
Kaotik sistemler hassasiyet özellikleri ile ön plana çıkmaktadır. Aynı değerlerin tekrar elde edilebilmesi için kullanılan kaotik sistemdeki, tüm başlangıç şartları ve parametrelerin aynı olması gerekmektedir. Küçük bir değişiklikte aynı sonuçlar elde edilemeyecektir. Kaotik sistemlerdeki duyarlılık, şifreleme uygulamaları için oldukça önemlidir. Bu tez çalışmasında RSÜ için yeni kaotik sistemler kullanılarak, rasgele sayılar elde edilmiştir. Şifreli verilerin çözülebilmesi için tekrar aynı rasgele sayılara ihtiyaç duyulmaktadır ve bu yüzden kaotik sistem ile kaotik sisteme ait tüm özelliklerin net olarak bilinmesi gerekmetedir.

RSÜ üretimi için kaotik sistemler RK4 ile ayrıklaştırma işlemlerine gerek duymaktadır. Ayrıklaştırma işleminde bir sonraki değer, öncekine bağımlı olduğu için, küçük bir hatada farklı sonuçlar, dolayısıyla sonuçta farklı rasgele sayılar elde edilecektir. Küçük bir hata diğerine neden olacağı için, neredeyse üretilen tüm rasgele sayılar hatalı olmuş olacak ve şifreli verileri çözmek mümkün olmayacaktır.

Analizde sonuç olarak çözünen veriye bakılmaktadır. Eğer herhangi bir küçük değişim sonucu orijinal veri elde edilememişse analiz sonucu başarıya ulaşmıştır. Örnek olması açısından 10000 bitlik bir mono türü ses verisi verilmiştir. Üretilen değerler birbirine bağımlı oldukları için şifre çözme esnasında Şekil 6.35'deki orijinal ses sinyali yerine, Şekil 6.36'daki bozuk ses sinyali elde edilmiştir. Tezde gerçekleştirilen uygulamalarda da, kaotik sistemdeki parametreler tam olarak doğru girilmezse, benzer şekilde bozuk çoklu-ortam verileri elde edilmiş olacaktır.



Şekil 6.35. Örnek orjinal ses sinyali



Şekil 6.36. Şifre çözme işlemi sonucu elde edilen bozuk ses sinyali

6.1.6.3. Kaos şifreleme etkisi

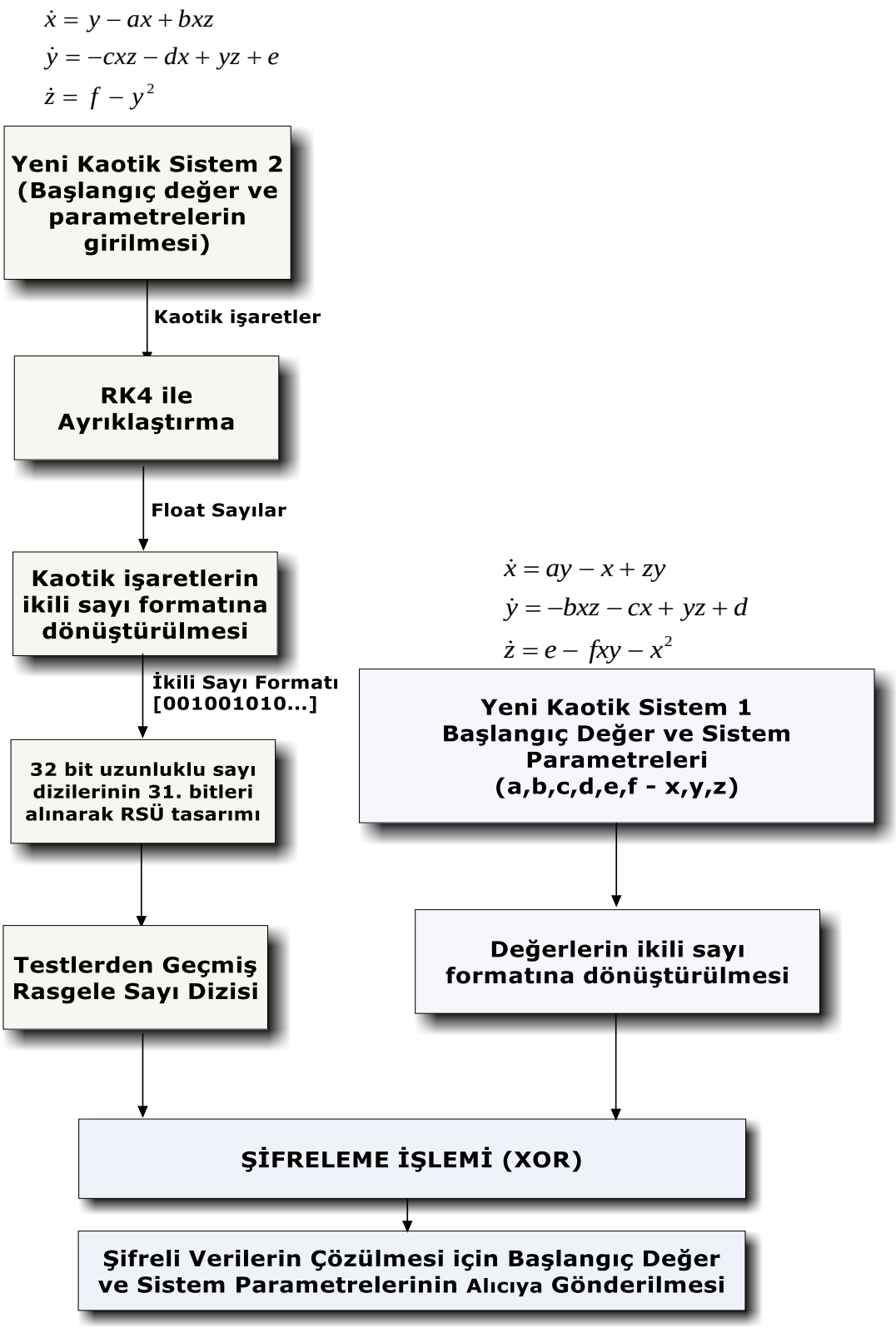
Şifreleme işlemi sonucu, şifrelenen veriler oldukça karmaşık hale gelmektedir. Bu etkiye şifrelemedeki kaosun etkisi denilmektedir. Gerçekleştirilen tüm uygulamalar incelenirse, şifrelenmiş veriler oldukça karmaşık gözükmekte, yani kaotik sistemin meydana getirdiği

etki görülebilmektedir. Bu şekildeki şifrelenmiş verileri çözmek kriptoanalizciler için kolay olmayacaktır.

6.2. Yeni Kaotik Sistem 1'deki Başlangıç Değerleri ve Parametrelerin Şifrelenmesi

Kaotik sistemler için başlangıç şart değerleri ve parametreler en önemli unsurlardır. Bu unsurlar olmadan kaotik sistemler elde edilemez. Şifreli verilerin çözülebilmesi için başlangıç değerleri ve parametrelerin mutlaka doğru olarak bilinmesi gerekmektedir. Bir önceki bölümde tüm şifreleme işlemlerinde, başlangıç değerleri ve parametreler değiştirilmeden şifreleme işlemleri gerçekleştirilmiştir. Bu değerlerin saklanması gerektiği durumlarda ikinci yeni kaotik sistem kullanılarak, birinci kaotik sistemin başlangıç değerleri ve parametreleri şifrelenmiştir. Bu şekildeki, şifreli multimedia verilerini çözmek isteyen kişilerin, öncelikle şifrelenmiş birinci kaotik sistemdeki başlangıç değerleri ve parametrelerini de çözmeleri gerekmektedir.

Yeni kaotik sistem 1'deki başlangıç değerleri ve parametrelerini şifrelemek için yeni kaotik sistem 2 ile elde edilen rasgele sayılar kullanılmıştır. Yeni kaotik sistem 2 ile elde edilen rasgele sayılar FIPS-140-1 ve NIST-800-22 testlerinden başarıyla geçmişti. Yeni kaotik sistem 1'in, sistem parametreleri $a = 2.8,\ b = 0.2,\ c = 1.4,\ d = 1,\ e = 10\ ve\ f = 2$ ve başlangıç değerleri x=y=z=0 olduğu durumlarda kaotik özellik göstermektedir. Bu bölümde kaotiklik için verilen değerlerin, şifreleme işlemleri gösterilmiştir. Şifreleme işlemine dair blok diyagram Şekil 6.37'de görüldüğü gibidir.



Şekil 6.37. Yeni kaotik sistem 1'in başlangıç değer ve parametrelerinin şifrelenmesini dair blok diyagram

Sistem parametreleri ve başlangıç değerleri decimal ve float sayılardan meydana gelmektedir. Float olan sayılar şifrelendiklerinde 32 bit, decimal sayılar ise 8 bit olarak karşı tarafa gönderılmektedir. Başlangıç değerleri ve sistem parametrelerinin şifrelenmemiş değerleri ile, float ve decimal olarak şifrelenmiş değerleri Tablo 6.1'de verildiği gibidir.

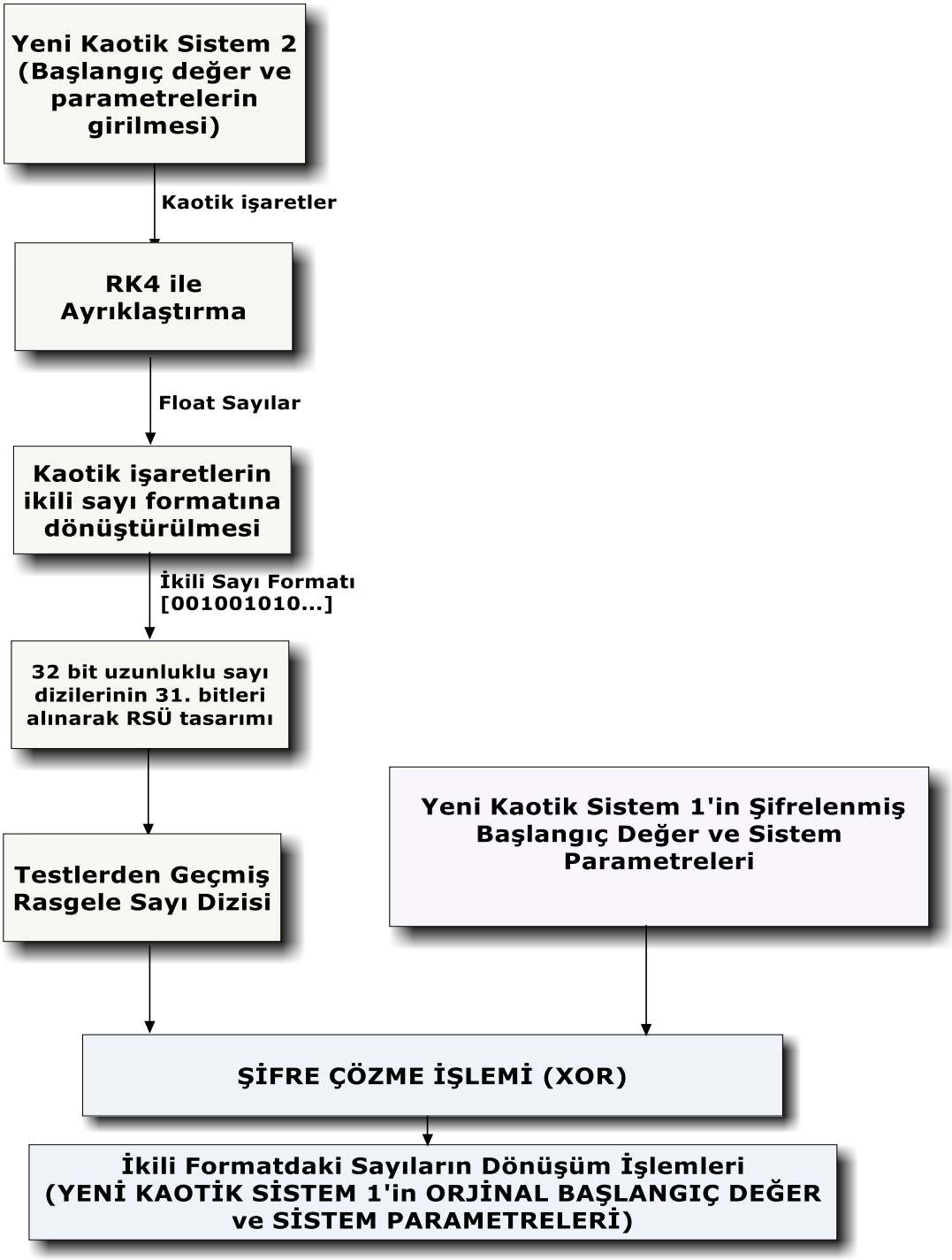
Tablo 6.1. Başlangıç değer ve parametrelerin yeni kaotik sistem 2 ile şifrelenmesi

Başlangıç Değer ve Parametreler		Şifrelenmiş Değeri (Binary)	Şifrelenmiş Değeri (Float veya Char)
a	2.8	00011010000000011010010010011111	2.68095550e -23
b	0.2	01100100011111100101101101100001	1.87682300e+22
c	1.4	01100101100000011010010010011111	7.65277500e+22
d	1	01011011	[
e	10	01010000	P
f	2	01011000	X
x y z	0	01011010	Z

Yeni kaotik sistem 1 ile şifrelenmiş multimedia verilerini çözmek isteyen kişilerin, öncelikle bu kaotik sistem için gerekli başlangıç değer ve parametrelerini bilmesi gerekmektedir. Bu şekilde, başlangıç değer ve parametreleri karşı tarafa istendiği takdirde şifreli olarak gönderilerek, multimedia verileri için daha güvenli bir iletişim sağlanabilmektedir.

Başlangıç değerleri ve sistem parametreleri yeni kaotik sistem 2 ile şifrelendikten sonra, tekrar aynı yöntemle çözülebilmektedir. Şifreleme işlemlerinde yine basit işlemlerin kullanılması hız açısından avantaj sağlamaktadır. Şifre çözme işlemine dair blok diyaram Şekil 6.38'de verildiği gibidir.

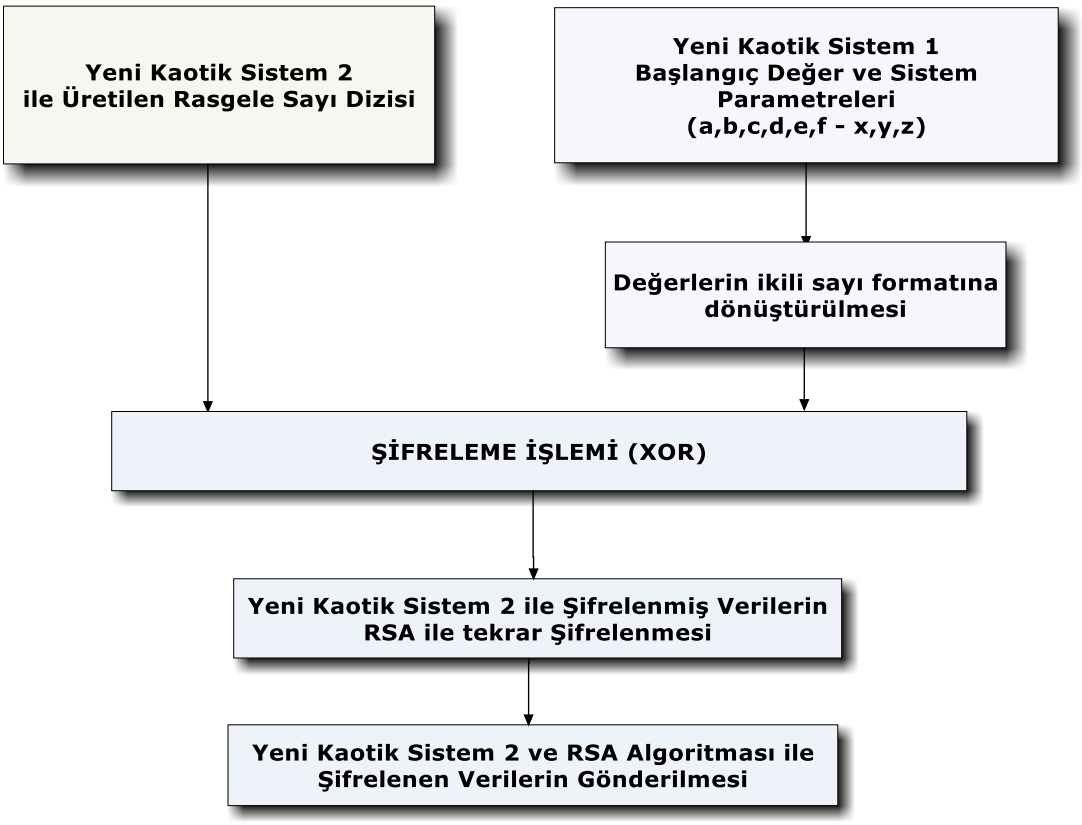
$$\dot{x} = y - ax + bxz$$
$$\dot{y} = -cxz - dx + yz + e$$
$$\dot{z} = f - y^2$$



Şekil 6.38. Yeni kaotik sistem 1'in şifrelenmiş başlangıç değer ve parametrelerinin çözülmesine dair blok diyagram

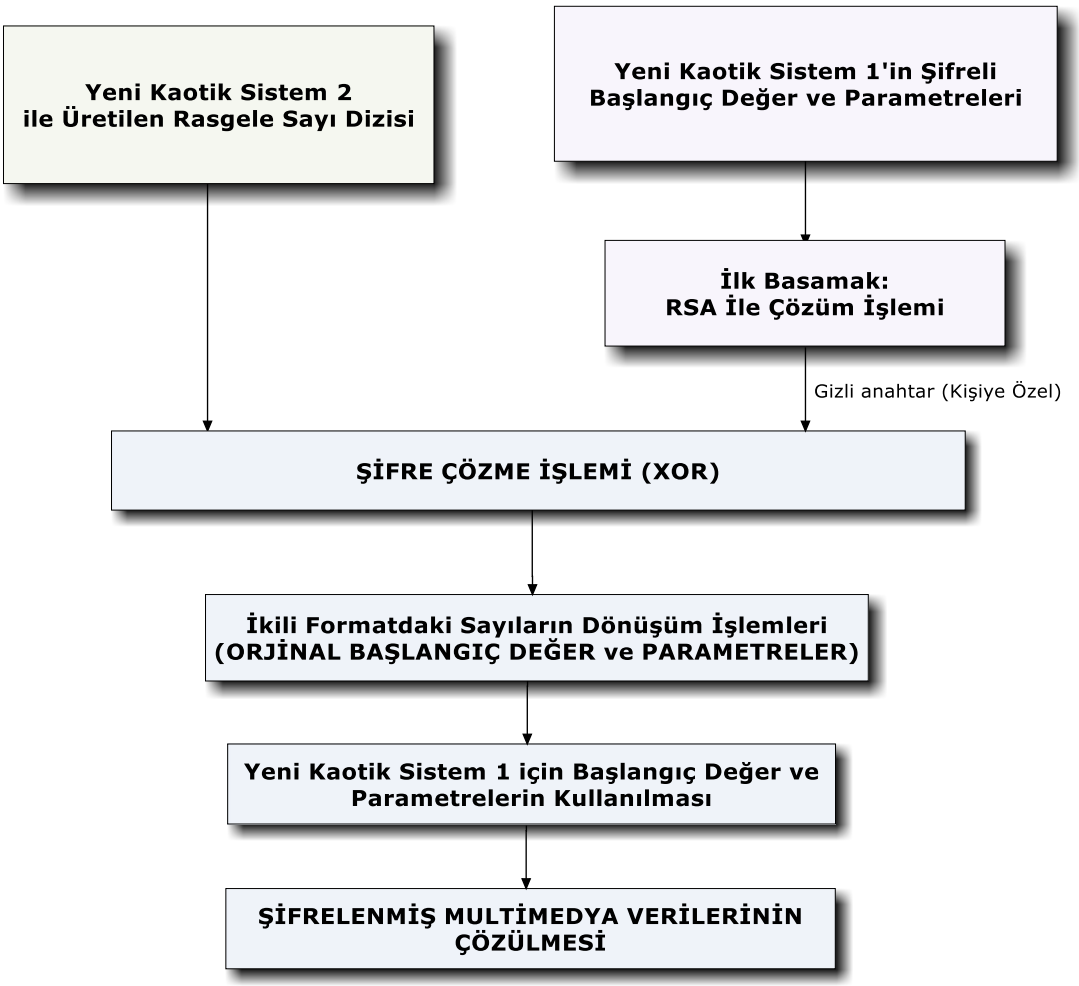
Anahtar dağıtım problemlerinin olduğu yerlerde, yeni kaotik sistem 2 ile yapılan şifreleme uygulamasına ek olarak var olan bir asimetrik şifreleme algoritmasıda kullanılabilir. En güvenilir asimetrik şifreleme algoritmalarından birisi RSA olduğu için bu tez çalışmasında asimetrik şifreleme işlemi için RSA algoritması kullanılmıştır.

Şekil 6.37'de yeni kaotik sistem 2 ile yapılan şifreleme işlemine ek olarak RSA asimetrik şifreleme algoritması ile yeni kaotik sistem 1'deki anahtarların şifreleme işlemleri gerçekleştirilmiştir. Şekil 6.39'daki blok diyagramdanda görüldüğü üzere yeni kaotik sistem 2 ile şifrelenen veriler daha sonra RSA algoritması ile şifrelenmiştir. Şifrelenen verileri başka kişilerinde çözmesi ve farklı anahtarın kullanılması isteniyorsa bu yöntem tercih edilebilir.



Şekil 6.39. RSA algoritması kullanılarak şifreleme işlemlerinin gerçekleştirilmesi

Veriler şifrelendikten sonra simetrik şifreleme yönteminden farklı olarak asimetrik şifreleme işleminde kişiye özel gizli anahtar oluşturulmuştur. Şifreli verileri çözmek isteyen kişiler, kendilerine ait anahtarı kullanarak şifrelenmiş yeni kaotik sistem 1'deki başlangıç değer ve parametrelerinin çözme kısmının ilk aşamasını gerçekleştirmiş olacaktırlar. Orjinal değerlerin elde edilebilmesi için RSA ile çözülen verilerin, yeni kaotik sistem 2 ile üretilen rasgele sayılar kullanılarak çözülmesi gerekmektedir. Çözülen başlangıç değer ve parametreler yeni kaotik sistem 1'de yerlerine konularak, şifrelenmiş multimedia verileri, bir önceki bölümde anlatılanlar gerçekleştirildiği takdirde tekrardan elde edilebilecektir.

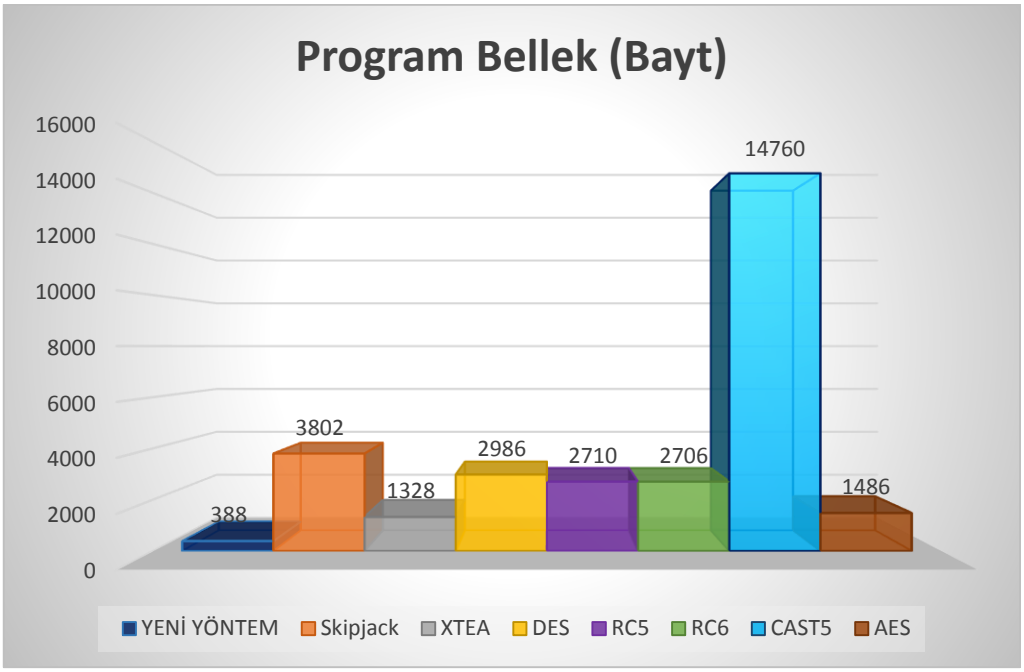


Şekil 6.40. RSA algoritması kullanılarak şifrele çözme işlemlerinin gerçekleştirilmesi

6.3. Gerçekleştirilen Kaos Tabanlı RSÜ ile Şifreleme Yönteminin AVR Studio 5.1 ile Performans Değerlendirmesi

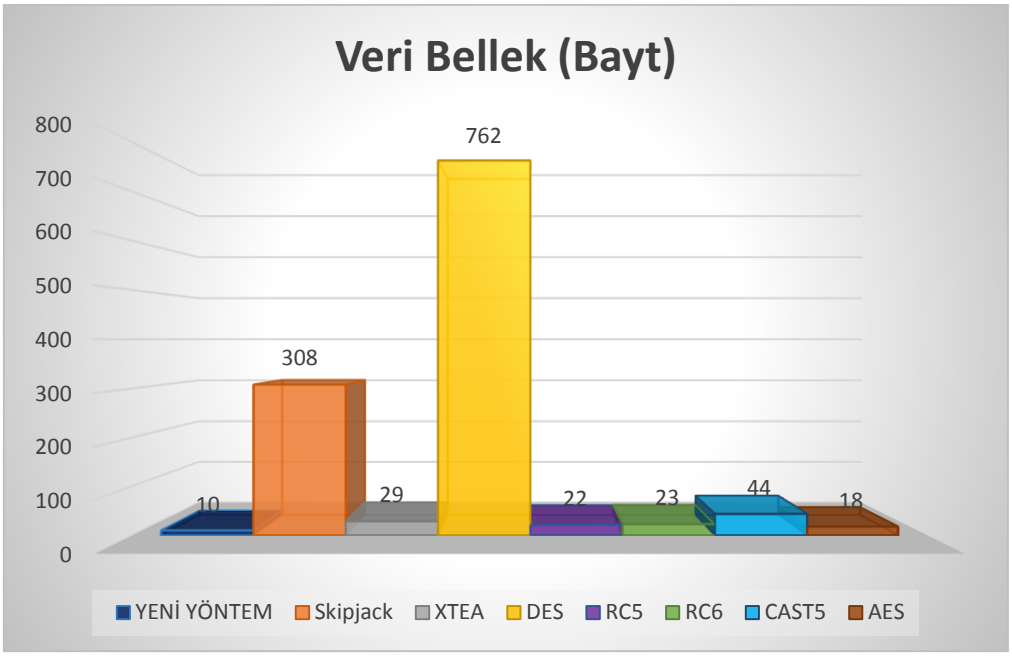
Tez çalışmasında gerçekleştirilen şifreleme uygulamaları, basit temel işlemlerle yapıldığı için (XOR), gerçekleştirilen şifreleme yöntemi, mikrodenetleyiciler gibi gerçek ortam uygulamalarında; süre, program bellek ve veri bellek olarak çok az yer kaplamaktadır. Literatürde AES, DES, XTEA, CAST5, RC5, RC6 ve Skipjack gibi birçok şifreleme yöntemi bulunmaktadır. Veri boyutu arttırıldığı zaman gerçek ortam uygulamalarında, bu tür şifreleme yöntemlerini kullanmak bazen imkansız hale gelmektedir. Performans değerlendirmesi açısından bu şifreleme yöntemleri, tez çalışmasında gerçekleştirilen yöntem ile süre, program bellek ve veri bellek olarak karşılaştırılmıştır. Uygulama platformu olarak AVR Studio 5.1 programı kullanılmıştır. Şifreleme yöntemi için gerekli olan kodlar bu program yardımıyla yazılıp, derlenmiştir. Gerçekleştirilen derleme sonuçlarında gerekli performans analizleri yapılarak, grafikler üzerinden incelemeler yapılmıştır. Değerlendirme işlemleri; RC6 ve AES şifreleme yöntemleri için 128 bit (AES ve RC6 min. 128 bitlik bloklar halinde şifreleme yapmaktadır), diğer şifreleme yöntemleri için 64 bitlik veriler şifrelenerek gerçekleştirilmiştir.

Performans değerlendirmeleri için; Bölüm 2'de açıklanan Skipjack, XTEA, DES, RC5, RC6, CAST5 ve AES şifreleme yöntemleri kullanılmıştır. Bu şifreleme yöntemleri, yeni kaotik sistem kullanılarak üretilen rasgele sayı üreteçleri ile yapılan şifreleme işlemlerindeki yöntemle süre ve bellek kriteleri dikkate alınarak karşılaştırılmıştır. Program bellek ve veri bellek, mikrodenetleyiciler gibi sınırlı donanım kaynaklarının sahip olunduğu gerçek zaman uygulamalarında oldukça önemli kriterlerdir. Çeşitli bellek boyutlarına sahip mikrodentleyiciler bulunmakla birlikte, donanım özellikleri arttıkça maliyetde artmaktadır [117]. Bellek harcamaları ne kadar az olursa gerçek ortam uygulamarındaki kullanım oranı artmakta ve maliyet azalmaktadır. Şekil 6.41 ve 6.42'de görüldüğü gibi ilk olarak program bellek ve veri bellek ölçümleri ele alınmıştır.



Şekil 6.41. AVR Studio 5.1 ile şifreleme yöntemlerinin program bellek ölçümleri

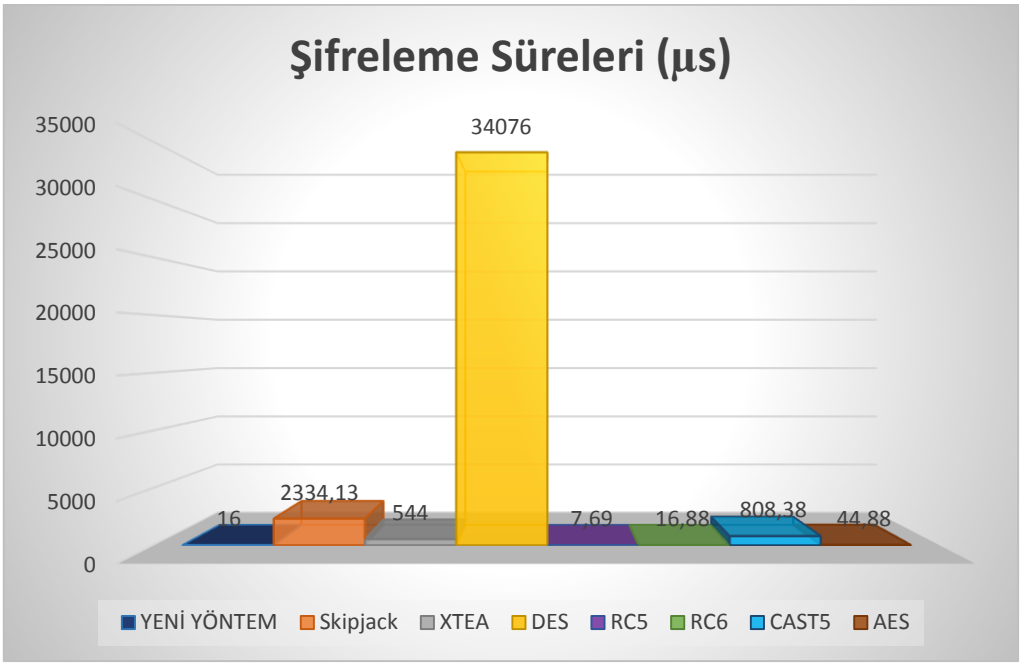
AVR Studio 5.1 programı ile yazılan kodlar derlendiğinde, program bellek ve veri bellek harcamaları, derleme çıktısı olarak verilmektedir. Tüm bellek sonuçları Şekil 6.41 ve 6.42'deki grafiklere aktarılmıştır. Grafiklerdeki sonuçlardan da görüldüğü üzere; CAST5 şifreleme yöntemi program bellekte 14760 bayt ile, DES şifreleme yöntemi ise 762 bayt ile veri bellekte en fazla yeri kaplarken, her iki bellek türünde de, yeni yöntem bellek harcamaları en azdır. Yeni yöntemle gerçekleştirilen şifreleme işlemi; 388 bayt program bellek ve 10 bayt veri bellek ile diğer şifreleme yöntemlerine göre oldukça iyidir. Şifrelenecek olan verilerin arttığı düşünüldüğünde, diğer yöntemlerle şifreleme işlemlerini gerçek ortam uygulamalarında kullanmak bellek miktarındaki sıkıntılardan dolayı mümkün olmayacaktır. Önceki bölümlerde gerçekleştirilen güvenlik analizleride dikkate alındığında, yeni yöntem ile her türlü çoklu-ortam türü veri gizleme işlemleri, bellek sıkıntısı olan gerçek ortam uygulamalarında, özellikle büyük boyutlu verilerin güvenli olarak şifrelenmesinde önemli avantajlar sağlayabilir.



Şekil 6.42. AVR Studio 5.1 ile şifreleme yöntemlerinin veri bellek ölçümleri

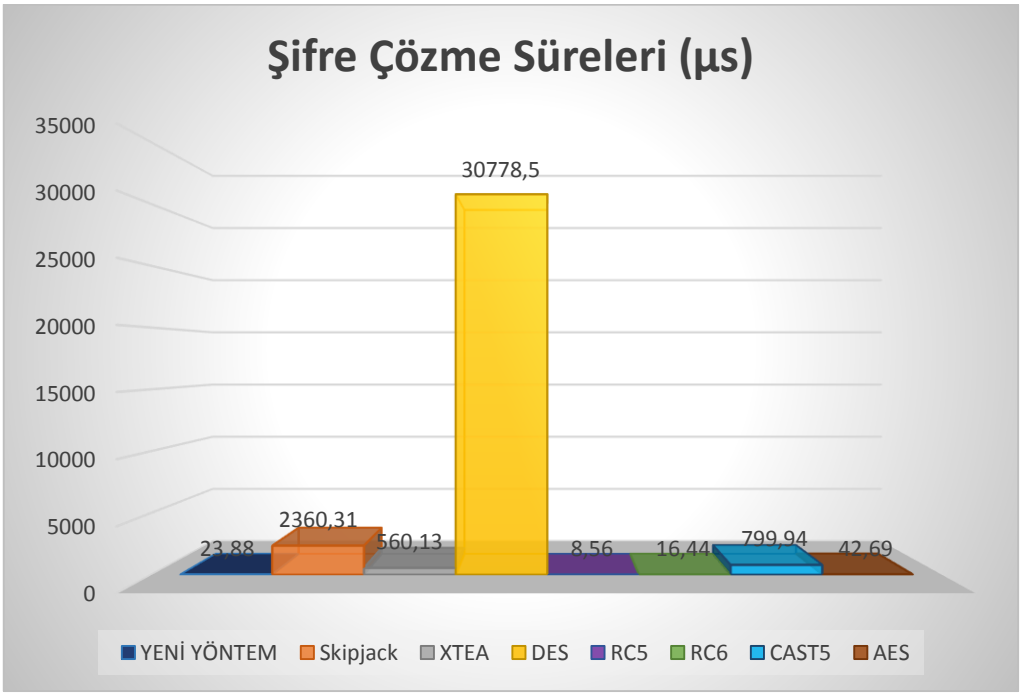
Bellek miktarlarının düşük olması ile gerçek ortam uygulamalarında kullanılabilirliğin yanında, diğer bir önemli kriterde hızdır. Bunun için bir sonraki aşamada şifreleme ve şifre çözme işlemleri için harcanan süreler ölçülmüşür. AVR Studio 5.1 programı ile süre ölçümü için, debug menüsü kullanılmıştır. Debug menüsünden adım adım çalışma izlenebildiğinden şifreleme ve şifre çözme süreleri µs cinsinden gözlemlenebilmiştir. Çalışma frekansı tüm şifreleme yöntemleri için 16 MHZ olarak ele alınmıştır.

Şekil 6.43 ve Şekil 6.44'de şifreleme ve şifre çözme süreleri verilmiştir. DES şifreleme algoritması her iki kriterde de 34076 µs ve 30778.5 µs ile en fazla süre harcamakta ve dolayısıyla kapladığı bellek miktarıda dikkate alındığında gerçek ortam uygulamaları için önemli bir dezavantaj oluşturmaktadır. RC5 şifreleme yöntemi süre kriterleri açısından daha iyi sonuçlar vermiştir. Şifreleme süresi 7.69 µs iken, şifre çözme süresi 8.56 µs'dir. Yeni yöntemle ise şifreleme süresi 16 µs iken, şifre çözme süresi 23.88 µs'dir.



Şekil 6.43. AVR Studio 5.1 ile şifreleme yöntemlerinin şifreleme sürelerinin ölçümleri

RC5 şifreleme yöntemi ile çalışma süreleri kısa olsada, harcanan program bellek miktarları dikkate alındığında; RC5 şifreleme yöntemi, yeni yönteme göre yaklaşık olarak 7 kat fazla yer kapladığı ve süre harcamalarında iki yöntemde çok fazla fark olmadığı için, harcanan süreler ihmal edilebilir bir kriter haline gelmektedir.



Şekil 6.44. AVR Studio 5.1 ile şifreleme yöntemlerinin şifre çözme sürelerinin ölçümleri

BÖLÜM 7. SONUÇLAR, ÖNERİLER ve DEĞERLENDİRMELER

Sunulan tez çalışmasında, literatürde bulunmayan yeni tasarlan kaotik sistemlerle, uluslararası en üst standart olan NIST-800-22 ve FIPS-140-1 istatistiksel testlerinden başarıyla geçirilerek tasarlanan özgün RSÜ'nin kullanıldığı, performans bakımından yüksek hızlı ve güvenlikli, bütün çoklu ortam verilerinin şifrelenebildiği (özellikle büyük boyutlu veriler), birçok gerçek ortam uygulamaları için (mikrodenetleyiciler, FPGA vb..) uygun, kaos tabanlı bir şifreleme yöntemi geliştirilmiş ve başarıyla çoklu ortam verilerinde uygulanmıştır.

Tez çalışmasının ilk kısmında yeni kaotik sistemlerin denge nokta analizi, faz portre analizi, lyapunov üstel analizi, zaman seri analizi, çatallaşma diyagram gibi detaylı analizleri gerçekleştirilmiştir. Bu analizler sonucunda yeni sistemlerin karmaşık dinamik yapıları ortaya çıkarılmış ve kaos tabanlı şifreleme için uygun yapıda oldukları gösterilmiştir. Bu analizlere ek olarak yeni kaotik sistemlerin karmaşık dinamik yapıları; elektronik devre simülasyonları ve gerçek devre uygulamaları ile de doğrulanmıştır.

Özellikle Yeni Kaotik Sistem 1'in tüm denge noktaları karmaşık sayı içerdiğinden literatürde az rastlanılan denge noktasız kaotik sistemler olarak adlandırılması ve bazı kaotik analizlerin (Örn. Shilnikov metod gibi) bu tip sistemler üzerinde yapılamaması, bu sistemin şifreleme ve benzeri uygulamalar için çok uygun olduğunu göstermektedir.

Bu tez çalışmasında yeni bulunan kaotik sistemler, kaos tabanlı şifrelemenin temelini oluşturan Rasgele Sayı Üreticini tasarlamak için kullanılmıştır. Fakat bu sistemler şifreleme uygulamaları dışında, matematik, haberleşme, görüntü işleme, bulanık mantık, kontrol uygulamaları, fizik, optimizasyon, mekatronik ve işletme, endüstri, müzik gibi sosyal içerikli alanlarda da kullanılabilme potansiyeline sahiptir.

Sürekli zamanlı olan yeni kaotik sistemleri RSÜ olarak kullanabilmek için hassas RK4 nümerik analiz yöntemiyle ayrıklaştırılarak ayrık olan float tipi ifadeler, ikili sayı formatına çevrilerek sonucunda RSÜ tasarımı yapılmıştır. RSÜ tasarımında, 32 bitlik sayı dizilerinin son ve sondan bir önceki yüksek hassasiyetli bitleri seçilerek bit dizileri oluşturulmuştur. Oluşturulan rasgele bit dizilerinin, uluslararası düzeyde kabul görmüş olan NIST-800-22 ve FIPS-140-1 rasgelelik testleri ile başarım düzeyleri ölçülmüştür. FIPS-140-1 testi için 20 Kbit veri, NIST-800-22 için 1Mbit veri alınarak testler gerçekleştirilmiştir. Üretilen rasgele diziler FIPS-140-1'deki dört testten (Tablo 5.4, Tablo 5.6) ve NIST-800-22'de ise onaltı testin hepsinden(Tablo 5.5, Tablo 5.7) başarıyla geçmiştir. Başarımı test edilen rasgele sayı dizilerinin şifreleme çalışmalarında kullanımının uygun olduğu görülmüştür.

Sunulan tez çalışmasının son aşamasında, yeni kaotik RSÜ tabanlı özgün bir şifreleme algoritması geliştirilerek ayrı ayrı sinyal, metin, ses, resim ve video gibi multimedia verilerinin şifreleme işlemleri gerçekleştirilmiştir. Şifreleme işlemlerinde karmaşık olmayan mantıksal operatörler kullanılmıştır. Gerçek ortam uygulamaları için işlem yükü az olan sade algoritmalarla güçlü şifreleme uygulamaları gerçekleştirmek oldukça önemlidir.

Şifreleme işlemleri sonrası, geliştirilen kaos tabanlı şifreleme yönteminin yüksek güvenlik seviyesinde olduğunun kanıtlanması için, veri türüne göre çeşitli güvenlik analizlerinin yapılmıştır. Kaos tabanlı şifreleme çalışmalarında görülen önemli bir eksiklik, güvenlik analizlerinin yetersiz olması ve performans değerlendirmelerinin bulunmamasıdır. Geliştirilen kaos tabanlı şifreleme yönteminin güvenlik analizleri; korelasyon, histogram, anahtar duyarlılık, anahtar uzunluk analizi gibi yöntemler ile gerçekleştirilmiştir.

Korelasyon analizi, iki değişken arasındaki doğrusal ilişkinin yönünü ve gücünü belirtir. Korelasyon analizi için değişkenler arası ilişkinin doğrusal olması gerekmektedir. Şekil 6.29'a bakıldığında, doğrusal bir ilişkiden ziyade çok iyi bir homejen dağılım görülmektedir. Analiz sonucu doğrusal bir ilişki bulunmadığından dolayı şifreli veri hakkında bir çıkarım yapmakta mümkün olmayacaktır.

Histogram analizi ile veri dağılım yoğunlukları tespit edilmekte ve bu dağılım grafiksel gösterim ile ifade edilmektedir. Şifreleme uygulamalarında, verilerin bir birine olabildiğince yakın değerler alarak dağılması şifrelemenin iyi olduğunu göstermektedir. Şekil 6.25 ve 6.33 den de görüldüğü üzere verilerin dağılımları neredeyse birbirlerine eşit ve analiz sonuçları oldukça iyidir.

Anahtar duyarlılık analizinde şifrelenmiş verinin kullanılan anahtara ne derece duyarlı olduğuna bakılmaktadır. Şifrelenmiş veri çözülürken anahtar üzerinde oluşan küçük bir değişiklik, orijinal verinin elde tekrar elde edilmesini etkilemektedir. Bu etkilenme oranına göre şifreleme işleminin anahtar duyarlılık analizi gerçekleştirilmektedir. Tezde gerliştirilen kaos tabanlı şifrelemede her veri için farklı anahtar üretildiğinden şifreli verinin çözülebilmesi için tüm anahtarların bilinmesi gerekmektedir. Üretilen anahtarlar ise bir önceki anahtar üretimine bağımlı olduğu için ufak bir değişiklik sonucunda, şifreli verilerin çözülme sürecinde Şekil 6.36'da olduğu gibi, Şekil 6.35'den çok farklı, orjinalinden uzak, bozuk veri elde edilecektir. Orijinal verinin bozulma oranının yüksekliği, şifreleme işleminin anhtara olan hassasiyetini göstermektedir.

Anahtar uzunluk analizinde anahtar seçim kümesinin yeterince büyük olup olmadığı incelenmektedir. Güçlü saldırıları etkisiz hale getirmek için anahtar uzunluğu yeterince büyük olmalıdır. Kaotiklik boyutu ve diğer değişkenler arttıkça anahtar uzunluğuda artacaktır. Bir değişken 10^{14} farklı değer alabilmektedir. Bu yüzden, tezde tasarlanan her yeni kaotik sistemde, 6 parametre ve 3 başlangıç değeri olduğu için, anahtar uzunlukları toplamda 10^{126} ($10^{84}.10^{42}$) olmaktadır. Görüldüğü gibi anahtar uzunluğunun çok fazla olması şifreli verilerin güvenliği için başka bir önemli kriterdir.

Son aşamada; yeni kaotik RSÜ tabanlı özgün şifreleme yöntemi ile güncel literatürdeki Skipjack, DES, RC5, CAST5 gibi diğer bazı şifreleme yöntemleri ile bellek ve hız bakımından AVR Studio 5.1 programı ile ATMEGA 128 mikrodenetleyicisi tercih edilerek performans değerlendirmesi de yapılmıştır. Performans değer sonuçlarından da görüldüğü üzere (Şekil 6.41 – Şekil 6.44) yeni yöntem ile şifreleme işlemleri gerçekleştirmek, bellek ve hız açısından genel olarak incelendiğinde diğer yöntemlere göre oldukça iyidir. Program bellek ve veri bellek, mikrodenetleyiciler gibi sınırlı donanım kaynaklarının sahip olunduğu

gerçek zaman uygulamalarında oldukça önemli kriterlerdir. Bellek harcamaları ne kadar az olursa gerçek ortam uygulamarındaki kullanım oranı artmakta ve maliyet azalmaktadır.

İleriki çalışmalarda; tez çalışmasında bulunan yeni kaotik sistemler, rasgele sayı üreteci ve şifreleme uygulamaları dışında; haberleşme, görüntü işleme, kontrol, fizik gibi farklı pek çok alanda kullanılabilir. Tasarlanan kaos tabanlı RSÜ kullanılarak farklı şifreleme yöntemleri geliştirilip; kriptoloji veya stegenografi gibi güvenlik uygulamalarında kullanılabilir. Ayrıca AES, RSA gibi kaos tabanlı olmayan popüler şifreleme yöntemleriyle beraber hibrid yeni şifreleme yöntemleri geliştirilebilir. Tez çalışmasında, geliştirilen yüksek hızlı ve güvenlikli şifreleme yöntemi; daha az bellek ihtiyacı gerekeceği için, mikrodenetleciler, FPGA, DSP gibi birçok farklı gerçek ortam uygulamalarında, offline veya online olarak tüm multimedia verilerini güvenli bir şekilde şifrelemede kullanılabilir. Ayrıca geliştirilen özgün kaos tabanlı şifreleme yönteminin, savunma sanayi, haberleşme, tıp, kişisel verilerin gizlenmesi gibi alanlarda bilgi güvenliği teknolojisine katkılarda bulunması beklenmektedir.

KAYNAKLAR

[1] Lu, J., Chen, G., Zhang, S., Dynamical analysis of a new chaotic attractor. International Journal of Bifurcation and Chaos, 12(5), pp:1001-1015, 2002.

[2] Pehlivan, I., Wei, Z., Analysis, Nonlinear Control and Circuit Design of an Another Strange Chaotic System. Turkish Journal of Electrical Engineering and Computer Sciences, 20(2), pp: 1229-1239, 2012.

[3] Sprott, JC., A new class of chaotic circuit. Physics Letters A, 266(1), pp: 19-23, 2000.

[4] Li, C., Pehlivan, İ., Sprott, JC., Akgul, A., A novel four-wing strange attractor born in bistability. IEICE Electronics Express, 12(4), 2015.

[5] Coskun, S., Tuncel, S., Pehlivan, İ., Akgul, A., Microcontroller-Controlled Electronic Circuit for Fast Modelling of Chaotic Equations. Electronics World, 121(1947), pp:24-25, 2015.

[6] Lian, S., Efficient image or video encryption based on spatiotemporal chaos system. Chaos, Solitons and Fractals, 40, pp:2509-2519, 2009.

[7] Wang, Z., Cang, S., Ochola, EO., Sun, Y., A hyperchaotic system without equilibrium. Nonlinear Dynamics, 69, pp. 531-537, 2012.

[8] Yildirim, K., Veri Şifrelemesinde Simetrik ve Asimetrik Anahtarlama Algoritmalarının Uygulanması (Hybrid Şifreleme) , Yüksek Lisans Tezi, Kocaeli Üniversitesi, 2006.

[9] Milani, MRA., Pehlivan, H., Pour, SH., Kaos Tabanlı Bir Şifreleme Yöntemi ve Analizi. Akademik Bilişim Konferansı Bildirileri, Malatya, pp. 487-493, 2011.

[10] Liu, H., Wang., Triple-image encryption scheme based on one-time key stream generated by chaos and plain Images. The Journal of Systems and Software, 86, pp:826-834, 2013.

[11] Wanbo, Y., Chunjian, C., Xiaopeng, W., Xuesong, Y., Image Encryption Algorithm Based on High-dimensional Chaotic Systems. International Conference on Intelligent Control and Information Processing, pp. 463–467,2010.

[12] Gupta, K., Silakari, S.,Novel Approach for fast Compressed Hybrid color image Cryptosystem. Advances in Engineering Software 49, pp. 29–42 ,2012.

[13] Volos, CK., Kyprianidis, LM., Stouboulos, IN., Image encryption process based on chaotic synchronization phenomena. Signal Processing, 93, pp:1328-1340, 2013.

[14] Munir, R., Security Analysis of Selective Image Encryption Algorithm Based on Chaos and CBC-like Mode. 7th International Conference On Telecommunication System, Services, And Application, pp. 142–146, 2012.

[15] Lorenz, EN., Deterministic nonperiodic flow. J. Atmos. Sci., 20:130–141, 1963.

[16] Rössler, OE., An equation for continuous chaos. Phys. Lett. A, 57:397–398, 1976.

[17] Lakshmanan, M., Murali, K., Chaos in Nonlinear Oscillators, Controlling and Synchronization. World Scientific, 1996.

[18] Chua, LO., Wu, C.W., Huang, A., Zhong, G., A Universal Circuit for Studying and Generating Chaos-Part I: Routes to Chaos. IEEE Trans. Circuits&Systems-I, 40:732-761, 1993.

[19] Cascais, J., Dialo, N., Costa, AN., Chaos and Reverse Bifurcation in a RCL Circuit. Physics Letters, 93A:213-216, 1983.

[20] Nakagawa, S., Saito, T., An RC OTA Hysteresis Chaos Generator. IEEE Trans. Circuits&Systems-I, 43:1019-1011, 1996.

[21] Tamasevicius, A., Namajunas, A., Cenys, A., Simple 4D Chaotic Oscillator. Electronic Letters, 32:957-958,1996.

[22] Ogorzalek, MJ., Order and Chaos in a Third Order RC Ladder Network with Nonlinear Feedback. IEEE Trans. Circuits&Systems, CA5-36:1221-1230, 1989.

[23] Matsumoto, T.,Chua, LO., Tanama, S., Simplest Chaotic Nonautonomous Circuit, Physical Rev. A, 30:1155-1157, 1984.

[24] Kawakami, H., Bifurcation of Periodic Responses in Forced Dynamic Nonlinear Circuits: Computation of Bifurcation Values of the System Parameters. IEEE Trans. Circuits&Systems., CAS-31:248-260, 1984.

[25] Saito, T., Chaotic Phenomena in a Coupled Oscillators. European Conf. on Circuit Theory and Design, pp:275-280,1987.

[26] Hamill, DC., Jeffries, DJ., Subhormonics and Chaos in a Controlled Switch-Mode Power Converters. IEEE Trans. Circuits&Systems, CAS-35:1059-1061, 1988.

[27] Poddar, G., Chakrabarty, K., Banerjee S., Control of Chaos in the Boost Converter. Electronics Letters, 31: 841-842, 1995.

[28] TSE, CK., Flip Bifurcation and Chaos in Three-State Boost Switching Regulators. IEEE Trans. Circuits&Systems-I, 41:16-23, 1994.

[29] Deschamps DD., Some Chaotic Consequences of Quantization in Digital Filters and Digital Systems, ISCAS '89 International Conference on Circuits and Systems, PortIand, pp:602-605, 1989.

[30] Chua, LO., Lin, T., Chaos and Fractals from Third-Order Digital Filters. Int. J. of Circuit Theory and Appl, 18:241-256., 1990.

[31] Chua, LO, Lin, T., Chaos in Digital Filters. IEEE Trans. Circuits&Systems, CAS-35:648-658, 1990.

[32] Chua, LO, Lin, T., Fractal Pattern of Second-Order Nonlinear Digital Filters:A Symbolic Analysis. Int. Journl. of Circuit Theory and Appl., 18:541-550, 1990.

[33] Sundarapandian, V., PehlivaN, I., Analysis, control, synchronization, and circuit design of a novel chaotic system, Mathematical and Computer Modelling. 55(7-8):1904-1915, 2012.

[34] Pehlivan, İ., Uyaroğlu, Y., Simplified Chaotic Diffusionless Lorenz Attractor and its Application to Secure Communication Systems. IET Communications, 1(5), pp:1015-1022, 2007.

[35] Pehlivan, İ., Uyaroğlu, Y., A New 3D Chaotic System with Golden Proportion Equilibria: Analysis and Electronic Circuit Realization. Computers and Electrical Engineering, 38(6), pp:1777-1784, 2012.

[36] Çiçek, S., Uyaroğlu, Y., Pehlivan, I. Simulation and Circuit Implementation Of Sprott Case H Chaotic System And Its Synchronization Application For Secure Communication Systems. Journal of Circuits, Systems and Computers, 22(4), 2013.

[37] Uyaroğlu, Y., Pehlivan, İ., Nonlinear sprott94 case a chaotic equation: Synchronization and masking communication applications. Computers and Electrical Engineering, 36(6) pp:1093-1100, 2010.

[38] Quan, SG., Hui, C., Bin, ZY., A new four-dimensional hyperchaotic Lorenz system and its adaptive control. Chinese Physics B, 20(1), 2011.

[39] Chne, G., Ueta, T., Yet another chaotic attractor. International Journal of Bifurcation and Chaos, 9(7), pp:1465-1466, 1999.

[40] Sprott, JC., Some simple chaotic flows. Physical Review-Section E-Statistical Physics Plasma Fluids Related In-terdiscpl Topics, 50(2), 1994.

[41] Sprott, JC., Simplest dissipative chaotic flow. Physics letters A, 228(4), pp:271-274, 1997.

[42] Kocamaz, UE., Uyaroğlu, Y., Kizmaz, H., Control of Rabinovich chaotic system using sliding mode control. Int. J. of Adaptive Control and Signal Proc., Wiley, 1-9, 2013.

[43] Vembarasan, V., Balasubramaniam, P., Chaotic synchronization of Rikitake system based on TS fuzzy control techniques. Nonlinear Dyn., 74(1-2):31-44, 2013.

[44] Koyuncu, I., Ozcerit, AT., Pehlivan, P., An analog circuit design and FPGA-based implementation of the Burke-Shaw chaotic system. Optoelectronics and Advanced Materials-Rapid Comm., 7(9-10):635-638, 2013.

[45] Cafagna, D., Grassi, G., Chaos in a new fractional-order system without equilibrium points. Commun Nonlinear Sci Numer Simulat., 19, pp:2919–2927,2014.

[46] Jafari, S., Sprott, JC., Hashemi GSMR., Elementary quadratic chaotic flows with no equilibria. Physics Letters A, 377, pp:699–702,2013.

[47] Leonov, G., Kuznetsov, N., Vagaitsev, V., Localization of hidden Chua's attractors. Phys. Lett. A, 375, pp. 2230-2233, 2011b.

[48] Wei, Z., Dynamical behaviors of a chaotic system with no equilibria. Phys Lett A, 376, pp. 102-108, 2011.

[49] Merah, L., Pascha, A., Said, A., Mamat, NH., Design and FPGA implementation of Lorenz chaotic system for information security issues. Appl. Math. Sci., 7(5):237-246, 2013.

[50] Sakthidasan, K., Santhosh, BV., A New Chaotic Algorithm for Image Encryption and Decryption of Digital Color Images. International Journal of Information and Education Technology, 1(2), pp: 137-141, 2011.

[51] Oğraş, H., Turk, M., Digital Image Encryption Scheme using Chaotic Sequences with a Nonlinear Function. World Academy of Science Engineering and Technology, Stockholm, 2012.

[52] Findik, O., Şifrelemede Kaotik Sistemin Kullanılması. Yüksek Lisans Tezi, Selçuk Üniversitesi, 2004.

[53] Yardim, FE., Afacan, E., Lorenz-Tabanlı Diferansiyel Kaos Kaydırmalı Anahtarlama (Dcsk) Modeli Kullanılarak Kaotik Bir Haberleşme Sisteminin Simülasyonu. Gazi Univ. Müh. Mim. Fak. Der., 25(1), pp: 101-110, 2010.

[54] Sobhy, MI., Shehata, AR., Chaotic Algorithms for Data Encryption. Acoustics, Speech, and Signal Processing. IEEE International Conference, 2001.

[55] Oğraş, H., Turk, M., Oğraş, S., Kaos Tabanlı Sayısal Csk ve Dcsk Modülasyon Tekniklerinin Matlab/Simulink Ortamında Gerçekleştirilmesi. IV. İletişim Teknolojileri Ulusal Sempozyumu, Adana, 2009.

[56] Maysaa, A., Iman, Q., A Speech Encryption Using Chaotic Map and Blowfish Algorithms. Journal of Basrah Researches, 39(2), pp:68-76, 2013.

[57] Zhang, M., A generalized Chaos Synchronization Based Encryption Algorithm For Sound Signal Communication. Circuits Systems Signal Processing, 24(5), pp:535-548, 2005.

[58] Prabu, AV., Apparao, ST., Jaganmohan, M., Babu, RK., Audio Encryption in Handsets. International Journal of Computer Applications, 40(6), pp:40-45, 2012.

[59] Gao, T., Chen, Z., Image encryption based on a new total shuffling algorithm. Chaos, Solitons and Fractals, 38, pp: 213–220, 2008.

[60] Xiao, D., Liao, X., Wei, P., Analysis and improvement of a chaos-based image encryption algorithm. Chaos, Solitons and Fractals, 40, pp:2191–2199, 2009.

[61] Chen, G., Mao, Y., Chui, CK., A symmetric image encryption scheme based on 3D chaotic cat maps. Chaos, Solitons and Fractals, 21, pp:749–761, 2004.

[62] Huang, X., A New Digital Image Encryption Algorihm Based on 4D Chaotic System. International Journal of Pure and Applied Mathematics, 80(4), pp:609-616, 2012.

[63] Wang, Y., Wong, KW., Liao, X., Chen, G., A new chaos-based fast image encryption algorithm. Applied Soft Computing, 11, pp:514-522, 2011.

[64] Dubey, AK., Shukla, CK., Chaos based Encryption and Decryption of Image and Video in Time and Frequency Domain. IJCA Special Issue on (Network Security and Cryptography), pp:35-39, 2011.

[65] Su, Z., Lian, S., Zhang, G., Jiang, J., Chaos-Based Video Encryption Algorithms. Chaos-Based Cryptography Studies in Computational Intelligence, 354, pp:205-226, 2011.

[66] Rhouma, R., Belghith, S., Cryptanalysis of a spatiotemporal chaotic image/video cryptosystem. Physics Letters A, 372, pp:5790-5794, 2008.

[67] Gnanajeyaraman, R., Prasadh, K., Ramar, D., Audio encryption using higher dimensional chaotic map. International Journal of Recent Trends in Engineering, 1(2), pp:103-107, 2009.

[68] Sobhy, IM., Shehata, A., Secure Computer Communıcatıon Usıng Chaotıc Algorıthms. International Journal of Bifurcation and Chaos, 10(12), pp:2831-2839, 2000.

[69] Akgul, A., Kacar, S., Aricioğlu, B., Pehlivan, İ., Text encryption by using one-dimensional chaos generators and nonlinear equations. Electrical and Electronics Engineering (ELECO), 2013.

[70] Akgul, A., Pehlivan, İ., An Audio Data Encryption with a Discrete - Time Chaotic System. International Science and Technology Conference, 2014.

[71] Wieczorek, PZ., Golofit, K., Dual-metastability time-competitive TRNG. IEEE Trans. on Circuits and Syst., 61(1):134-145, 2014.

[72] Fischer, V., Drutavosky, M., Simka, M., Bochard, N., High performance TRNG in sltera stratix FPLDs.Field Program. Logic and App., Springer, 555–564, 2004.

[73] Istvan, H., Suciu, A., Cret, O., FPGA based TRNG using automatic calibration. Intelligent Comp. Comm. and Proc., IEEE 5th Int. Conf. on ICCP, 373-376, 2009.

[74] Cicek, I., Pusane, AE., Dundar, G.,. A novel design method for discrete time chaos based true random number generators. Integration, the VLSI Journal, 2014, 47.1: 38-47.

[75] Cicek, I., Pusane, AE., Dundar, G., A novel dual entropy core TRNG. IEEE 8th Int. Conf. on Elec. and Electronics Eng., 332-335, 2013.

[76] Pareschi, F., Setti, G., Rovatti, R., Implementation and testing of high-speed CMOS TRNGs based on chaotic systems, IEEE Trans. on Circuits and Syst., 57(12):3124-3137, 2010.

[77] Vural, Y., Kurumsal Bilgi Güvenliği ve Sızma (Penetrasyon) Testleri , Yüksek Lisans Tezi, Gazi Üniversitesi, 2007.

[78] Şahin, A., Buluş, E., Sakalli T. M., Modern Blok Şifreleme Algortimalarının Gücünün İncelenmesi, II. Mühendislik Bilimleri Genç Araştırmacılar Kongresi, İstanbul, Kasım 2005.

[79] Bandirmali, N., Ertürk, İ., Çeken, C., Bayilmiş, C., Yüksek Riskli Kablosuz Algılayıcı Ağlarda Güvenlik ve Şifreleme Uygulaması. Ağ ve Bilgi Güvenliği Sempozyumu, Kıbrıs, 2008.

[80] Sharp, ED., Information Security in the Enterprise, Information Security Management Handbook Fifth Edition, Tipton, F. H., Krause, M., Auerbach Publications, New York, pp:1199-1200, 2004.

[81] Karadere, T., Bilgi Güvenliği, http://security.metu.edu.tr/ Documents/Bilgi%20 Guvenligi.html, 2010.

[82] Gülaçti, E., Milli Açık Anahtar Altyapısı Eğitim Kitabı, http://www.kamusm.gov.tr/tr/Bilgideposu/Belgeler/teknik/aaa/index.html, 2010.

[83] PRO-G, Bilişim Güvenliği, Sürüm 1.1, Pro-G Bilişim Güvenliği ve Araştırma Ltd., http://www.pro-g.com.tr/whitepapers/bilisim-guvenligi-v1.pdf, 2003.

[84] Alvarez, G., LI, S., Some Basic Cryptographic Requirements For Chaos-Based Cryptosystems. International Journal of Bifurcation and Chaos, 16(8): 2129–2151, 2006.

[85] Altan, K., Kaşkaloğlu, K., KindaP, N., Özakin, Ç.,Saygi, Z., Yildirim, E., Yildirim, M., Yildiz, S., Kriptolojiye Giriş, Seminer Notları. Uygulamalı Matematik Enstitüsü, Kriptografi Bölümü, ODTÜ, 2004.

[86] Sakalli, M, T., Buluş, E., Şahin, A., Büyüksaraçoğlu, F., Akış Şifrelerinde Tasarım Teknikleri Ve Güç İncelemesi. Akademik Bilişim, Dumlupınar Üniversitesi, Kütahya, 2007.

[87] Akgul, A., Yüksek Güvenlikli Kızılötesi İletişim Uygulaması. Yüksek Lisans Tezi, Sakarya Üniversitesi, 2011.

[88] https://adnankaratas.wordpress.com/2013/09/28/sifreleme-alogoritmalari/, Erişim Tarihi: 28.02.2015.

[89] Stinson D. R., Cryptography, Theory and Practice, CRC Press, 1995.

[90] http://en.wikipedia.org/wiki/CAST-128, Erişim Tarihi: 20.02.2015.

[91] http://tr.wikipedia.org/wiki/RSA, Erişim Tarihi: 24.11.2014.

[92] Menezes, AJ., Orschot PC., Vanstano, SA., Handbook of applied cryptography. CRC press, 1996.

[93] Koyuncu, İ., Kriptolojik Uygulamalar İçin Fpga Tabanlı Yeni Kaotik Osilatörlerin Ve Gerçek Rasgele Sayı Üreteçlerinin Tasarımı Ve Gerçeklenmesi. Doktora Tezi, Sakarya Üniversitesi, 2014.

[94] Zhao, L., Liao, X., Xiao, D., Xiang, T., Zhou, Q., Duan, S., TRNG from mobile telephone photo based on chaotic cryptography. Chaos, Solitons & Fract., Elsevier, 42(3):1692-1699, 2009.

[95] Demirkol, AŞ., Kaotik osilatör girişli ADC tabanlı rasgele sayı üreteci. Yüksek lisans tezi, İstanbul Teknik Üniverstesi, 2007.

[96] Güven, P., Otonom olmayan kaotik sistemlede rasgele sayı üretiminin incelenmesi. Yüksek lisans Tezi, İstanbul Teknik Üniversitesi, 2006.

[97] Federal information processing standards publication, Security requirements for cryptographic modules. FIPS PUB 140-1, 1994. http://csrc.nist.gov/publications/fips/fips1401.htm, Erişim Tarihi: 05.12.2014.

[98] http://www.csm.ornl.gov/~dunigan/fips140.txt, Erişim Tarihi: 28.02.2015.

[99] A statistical test suite for random and pseudo RNGs for cryptographic applications. National institute of stand. and tech.,NIST-800-22, 2001. http://csrc.nist.gov/publications/ nistpubs/800-22/sp-800-22-051501.pdf, Erişim Tarihi: 05.01.2015.

[100] Avaroğlu, E., Donanım tabanlı rasgele sayı üretecinin gerçekleştirilmesi. Doktora tezi, Fırat Üniverstesi, 2014.

[101] Büyüksaraçoğlu, F., Buluş, E., Sözde rastsal sayı üretiminin kriptografik açıdan incelenmesi. TMMOB Elektrik Müh. Odası IV. İletişim Tekn. Ul. Semp., Adana, 2009.

[102] Maurer, UM., A universal statistical test for random bit generators. J. of cryptology, 5(2):89-105, 1992.

[103] http://tr.wikipedia.org/wiki/Korelasyon, Erişim Tarihi: 05.03.2014.

[104] Bandirmali, N., Yeni bir kablosuz algılayıcı ağ veri bağı katmanı güvenlik protokolü tasarımı. Doktora Tezi, Kocaeli Üniversitesi, 2010.

[105] Taşkiran, A., Burke-shaw ve T (tigan) kaotik osilatörlerinin tasarımı: Güvenli haberleşme amaçlı senkronizasyon uygulamaları. Yüksek Lisans Tezi, Sakarya Üniversitesi, 2011.

[106] Pehlivan, İ., Yeni kaotik sistemler: Elektronik devre gerçeklemeleri, senkronizasyon ve güvenli haberleşme uygulamaları. Doktora Tezi, Sakarya Üniversitesi, 2007.

[107] Hongjun, L., Wang, XB., Triple-image encryption scheme based on one-time key stream generated by chaos and plain images. The Journal of Systems and Software, pp. 826-834, 2013.

[108] Yavuz, O., Kaotik Ortamlarda Güvenli Veri Transferi. Yüksek Lisans Tezi, Karadeniz Teknik Üniversitesi, 2006.

[109] Butcher, JC., Numerical methods for ordinary differential equations, 2nd ed., L. John Wiley & Sons, Ed., 2008.

[110] Polking, JC., Download Odesolve.m, Rice University, http://math.rice.edu/~dfield/, 2014.

[111] Pehlivan, İ., Moroz, IM., Vaidyanathan, S., Analysis, synchronization and circuit design of a novel butterfly attractor. Journal of Sound and Vibration, 333(20):5077–5096, 2014.

[112] http://www.evrenindili.com/component/content/article/101-dostlarimiz/263-kaos-kuram-ve-kaotik-sistemler?directory=194, Erişim Tarihi: 01.12.2014.

[113] http://minitorn.tlu.ee/~jaagup/uk/dynsys/ds2/chaos/Poincare/Poincare.html, Erişim Tarihi: 12.12.2014.

[114] Pehlivan, İ., Uyaroğlu, Y., Gün, AR., Taşkiran, A., Tigan(T) Kaotik Sisteminin Elektronik Devre Gerçeklemesi ve Senkronizasyon Uygulaması. 6. Uluslararası İleri Teknolojiler Sempozyumu (IATS'11), Elazığ, pp. 413-418, 2011.

[115] Dormand, JR., Peter, JP., A family of embedded Runge-Kutta formulae. J. of computational and applied math., 6(1):19-26, 1980.

[116] http://en.wikipedia.org/wiki/Single-precision_floating-point_format, Erişim Tarihi: 28.02.2015.

[117] Çakiroğlu, M., Software implementation and performance comparison of popular block ciphers on 8-bit low-cost microcontroller. International Journal of the Physical Sciences, 5(9):1338-1343, 2010.

EKLER

EK A: Doktora Tez Kapsamında Yapılan Bilimsel Çalışmalar

Doktora tez kapsamında yapılan bilimsel yayınlar aşağıda verilmiştir.

[1] AKGUL, A., PEHLIVAN, I., A new three-dimensional chaotic system without equilibrium points, its dynamical analyses and electronic circuit application, Technical Gazette, DOI Number: 10.17559/TV-20141212125942.

[2] CHUNBIAO, L., PEHLIVAN, I., SPROTT, JC., AKGUL, A., A novel four-wing strange attractor born in stability, IEICE Electronics Express, 12(4):1-12, 2015.

[3] COSKUN, S., TUNCEL, S., PEHLIVAN, I., AKGUL, A., Microcontroller-Controlled electronic circuit for fast modelling of chaotic equations, Electronics World, 121(1947):24–25, 2015.

[4] AKGUL, A., KACAR, S., PEHLIVAN, I., An Audio Data Encryption with Single and Double Dimension Discrete-Time Chaotic Systems, The Online Journal of Science and Technology, TOJSAT, Temmuz, 2016.

[5] BAYILMIŞ, C., ÇAVUŞOĞLU, Ü., AKGÜL, A., SEVİN, A., KAÇAR, S., Employing Chaotic Encryption for IEEE 802.15.4-based LR-WPANs, International Conference on Computer Science and Information Systems (ICSIS'2014), Dubai, October 2014.

[6] AKGUL, A., PEHLIVAN, I., An Audio Data Encryption with a Discrete - Time Chaotic System, International Science and Technology Conference, ISTEC'2014, 46-51, Katar, December 2014.

[7] AKGUL, A., KACAR, S., ARICIOĞLU, B., PEHLIVAN, I., Text Encryption by Using One-Dimensional Chaos Generators and Nonlinear Equations, IEEE 8th Internatıonal Conference On Electrıcal And Electronıcs Engıneerıng, ELECO2013, 320-323, Bursa, Kasım 2013.

Doktora tez kapsamında yapılan bilimsel projeler aşağıda verilmiştir.

[1] PEHLİVAN İ, AKGUL A, İnternet Üzerinden Kaos Tabalı Yeni Bir Güvenli Multimedya İletişim Sistemi Tasarım Ve Gerçekleştirilmesi**,** Sakarya Üniversitesi BAP, 2013-50-02-018, 2013.

ÖZGEÇMİŞ

Akif AKGÜL, 12.08.1986 tarihinde Karşıyaka'da doğdu. İlköğrenimini İzmir, Mardin ve İstanbul'daki farklı okullarda tamamladı. 2004 yılında Üsküdar Haydarpaşa Anadolu Meslek Lisesi'nden mezun oldu. 2005 yılında başladığı Kocaeli Üniversitesi Teknik Eğitim Fakültesi Elektronik Öğretmenliği Bölümü'nü 2009 yılında tamamladı. 2011 yılında Sakarya Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Bilgisayar Eğitimi Anabilim Dalı'ndaki yüksek lisansı bitirdi ve aynı yıl Elektrik-Elektronik Mühendisliği Anabilim Dalı'nda doktora eğitimine başladı. Kasım 2009'da Sakarya Üniversitesi Teknik Eğitim Fakültesi Elektronik ve Bilgisayar Eğitimi Bölümü'nde araştırma görevlisi olarak çalışma başlayan Akif AKGÜL halen aynı görevini sürdürmektedir.