T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

EŞ ZAMANLI SIKIŞTIRICI ALGILAMA VE KRİPTOGRAFİ

ERTAN ATAR

DOKTORA TEZİ
ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ ANA BİLİM DALI
ELEKTRONİK PROGRAMI

DANIŞMAN
YRD.DOÇ.DR. LALE ÖZYILMAZ

EŞ DANIŞMAN
PROF.DR. OKAN K. ERSOY

İSTANBUL, 2016

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

EŞ ZAMANLI SIKIŞTIRICI ALGILAMA VE KRİPTOGRAFİ

Ertan ATAR tarafından hazırlanan tez çalışması 21.04.2016 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Elektronik ve Haberleşme Mühendisliği Anabilim Dalı'nda **DOKTORA TEZİ** olarak kabul edilmiştir.

Tez Danışmanı

Yrd. Doç. Dr. LALE ÖZYILMAZ

Yıldız Teknik Üniversitesi

Tez Eş Danışmanı

Prof. Dr. Okan K. ERSOY

Purdue Üniversitesi

Jüri Üyeleri

Yrd. Doç. Dr. Lale ÖZYILMAZ
Yıldız Teknik Üniversitesi _____

Yrd. Doç. Dr. M. Fatih AMASYALI
Yıldız Teknik Üniversitesi _____

Yrd. Doç. Dr. Soner ÖZGÜNEL
Haliç Üniversitesi _____

Yrd. Doç. Dr. Nihan KAHRAMAN
Yıldız Teknik Üniversitesi _____

Yrd. Doç. Dr. Banu ATAŞLAR AYYILDIZ
Kocaeli Üniversitesi _____

ÖNSÖZ

Haberleşme alanında genel olarak en önemli iki probleme, efektif/etkin/verimli işaret işleme ve bu işareti güvenli olarak iletme konularına, sıkıştırıcı algılama ile optiksel simetrik, asimetrik, hibrit kriptografi ve kriptoanaliz uygulaması sayesinde alternatif çözüm için yol gösterici olabileceğini umarım.

Bu tez çalışması sırasında bana yol gösteren, beni sürekli cesaretlendiren, yönlendiren, yakın ilgi, anlayış ve desteklerini esirgemeyen değerli danışman hocalarım, Prof.Dr.Okan K. Ersoy'a ve Yrd.Doç.Dr.Lale Özyılmaz'a sonsuz teşekkürlerimi sunarım. Bu tez çalışması sırasında devamlı yanımda olan, fikirleriyle bana yol gösteren, her konuda yardımcı olan ve cesaretlendiren tez izleme hocalarım, Yrd.Doç.Dr.M.Fatih Amasyalı'ya ve Yrd.Doç.Dr.Soner Özgünel'e çok teşekkür ederim. Ayrıca verdikleri dersler ya da öğütler olsun her zaman ufkumu açan ve bana sürekli yol gösteren değerli hocalarım Prof.Dr.Tülay Yıldırım'a, Doç.Dr.Burcu Erkmen'e, Yrd.Doç.Dr.Nihan Kahraman'a, Yrd.Doç.Dr.Tuba Kıyan'a, Prof.Dr.Etem Köklükaya'ya, Prof.Dr. Herman Sedef'e, Yrd.Doç.Dr.Banu Ayyıldız'a ve Yrd.Doç.Dr.Cenk Yavuz'a çok teşekkür ederim.

Bu çalışmamın oluşum sürecinde bana sürekli katkıları olan, Türk Telekom A.Ş. İstanbul-I Bölge Müdürlüğü ve Avcılar Telekom Müdürlüğü yönetici ve çalışanlarına, değerli yöneticilerim ve çalışma arkadaşlarım; Dr.Mehmet Beytur'a, Orhan Kıpçak'a, Kutluhan Önal'a, Nurettin Güneş'e, Yılmaz Çebi'ye, İlhan Pekdemir'e, Müjdat Demirkesen'e, Oktay Balpınar'a, Bilal Uçar'a, Doğan Ergün'e, Osman Aydın'a, Hamza Habip Tutan'a, Gökhan Öztürk'e, Sinan Yalçınkaya'ya, Ramazan Ölmez'e, Yavuz Selim Varlık'a, Erol Demirdelen'e, İbrahim Öztürk'e, Surhan Bozkurt'a, İbrahim Demirel'e, İsmail Cantürk'e, Faruk Aydın'a, Özcan Kartal'a, İbrahim Kurtoğlu'na, Mustafa Erdagöz'e, Hayati Çınar'a, Muratcan Sarı'ya, Sultan Eğitmen'e, Tuğba Nihan Atlı'ya ve Avcılar Telekom Müdürlüğü tüm Operasyon çalışanlarına teşekkür ederim.

Son olarak, hayatım boyunca beni destekleyen ve bugünlere getiren babam Vedat Atar ve annem Sayibe Atar, kardeşim Ersan Atar ile her türlü zorlukta yanımda olan ve sürekli beni destekleyen eşim Zekavet Çınar Atar'a sonsuz minnet duygularımı sunarım.

Nisan, 2016

Ertan ATAR

İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ vi

KISALTMA LİSTESİ vii

ŞEKİL LİSTESİ viii

ÇİZELGE LİSTESİ xii

ÖZET xiii

ABSTRACT xiv

BÖLÜM 1

GİRİŞ 1

1.1 Literatür Özeti 3

1.2 Tezin Amacı 4

1.3 Hipotez/Orijinal Katkı 4

BÖLÜM 2

SIKIŞTIRICI ALGILAMA 6

2.1 Nyquıst Örnekleme Kuramı 6

2.2 Seyrek Sinyaller (Seyreklik) 8

2.3 Sıkıştırılabilir Sinyal 11

2.4 Sıkıştırıcı Algılama Teorisi 12

2.5 Sıkıştırıcı Algılama Dik Eşleştirme Arayış Algoritması (DEA) 17

2.6 Ayrık Kosinüs-III Dönüşümü (AK3D) 22

BÖLÜM 3

KRİPTOGRAFİ 25

3.1 Şifrelemenin Temel Elemanları 26

3.1.1 Gizli Anahtar (Simetrik) Yöntemi 29

3.1.1.1 Veri Şifreleme Standardı (Data Encryption Standard-DES) 32

3.1.1.2 Gelişmiş Şifreleme Standardı(Advanced Encryption Standard-AES) 38

3.1.2 Açık Anahtar (Asimetrik) Yöntemi 43

3.1.2.1 RSA 44

3.1.3 Hibrit Yöntemi (simetrik+asimetrik beraber uygulaması) 45

BÖLÜM 4

SIKIŞTIRICI ALGILAMA DİK EŞLEŞTİRME ARAYIŞ YÖNTEMİ İLE GELİŞTİRDİĞİMİZ KRİPTOGRAFİ YÖNTEMLERİ 47

4.1 Sıkıştırıcı Algılama ile Simetrik Kriptografi Uygulaması 47
4.2 Sıkıştırıcı Algılama ile Hibrit(Simetrik+Asimetrik) Kriptografi (Aynı Anahtarları Kullanarak) 52
4.3 Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) Optiksel Kriptografi (Eşlenik Anahtarlar ve FFT Dönüşümü Kullanarak 58
4.4 Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) OptikselKriptografi (2Boyutlu) 65
4.5 Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) Optiksel Kriptografi (Ses İşareti) 72

BÖLÜM 5

GELİŞTİRDİĞİMİZ YÖNTEMLERİN KRİPTOANALİZ ÇALIŞMALARI..... 74

BÖLÜM 6

SONUÇLAR VE TARTIŞMA..... 90
KAYNAKLAR..... 94
ÖZGEÇMİŞ..... 99

SİMGE LİSTESİ

f_s	Örnekleme frekansı
T_s	Örnekleme periyodu
x(k)	Örneklenmiş işaret
Ω_s	Nyquist için örnekleme bant genişliği
δ	Dürtü fonksiyonu
φ	Algılama matrisi
y	Ölçüm vektörü
I	Birim matris
M	Ölçüm sayısı
N	Sinyal boyutu
K	Seyreklik derecesi
ψ	Taban
R	Rezidü vektör
S	İndis seti
t	Sayaç
y(n)	Katsayılar
p1	İlk faz+genlik kompleks, dağıtılmış ve rastgele anahtar
p2	İkinci faz+genlik kompleks, dağıtılmış ve rastgele anahtar
AAt	Kriptografik anahtarların yerleştirildiği algılama matrisinin tamamı
b	Kriptolanmış ve sıkıştırılmış (şifrelenmiş işaret) ölçüm vektörü
AA	Dönüşümler kullanarak elde edilen algılama matrisi
C	Şifreli işaret
x(t)	Sürekli sinyal
X	İlk anahtar
Y	İkinci anahtar
P	Orijinal açık işaret
fft	Fast fourier transform
x	Orijinal işaret
L	Sol
R	Sağ
K	Anahtar
s	Seyrek katsayılar vektörü
A	Genlik anahtarı
E	Şifreleme anahtarı
D	Şifre çözme anahtarı
$\tilde{x}$	Yaklaşık giriş işareti

KISALTMA LİSTESİ

SA	Sıkıştırıcı Algılama
SA DEA	Sıkıştırıcı Algılamada Dik Eşleştirme Arayış
AKD	Ayrık Kosinüs Dönüşümü
OMP	Orthogonal Matching Pursuit
MGS	Modified Gram Schmidt
DET	Dik Eşleyen Takip
HFD	Hızlı Fourier Dönüşümü
GSD	Gerçek Sinusoidal Dönüşümü
ASD	Ayrık Sinüs Dönüşümü
AK3D	Ayrık Kosinüs-III Dönüşümü
DC3T	Discrete Cosine-III Transform
DES	Data Encryption Standard
QR	QR factorization (*Q* is an orthogonal matrix, *R* is an upper triangular matrix)
DCT	Discrete Cosine Transform
IBM	International Business Machines
NASA	National Aeronautics and Space Administration
NIST	Na National Institute of Standarts and Technology
AES	Advanced Encryption Standard
RSA	Asimetrik Algoritma (ismini mucitlerinin baş harflerinden (Ronald L.Rivest, Adi Shamir ve Leonard Adleman) almıştır)
KPA	Known Plaintext Attack
FIPS	Federal Information Processing Standard
IP	Initial Permutation
ER	Error Reduction
ADC	Analog Digital Convertor
DEA	Dik Eşleştirme Arayış
CS	Compressive Sensing

ŞEKİL LİSTESİ

Sayfa

Şekil 2. 1 Örnekleme teoremi 6
Şekil 2. 2 Örneklenmiş sinyalin geri elde edilmesi 7
Şekil 2. 3 Elde edilmiş sinyalde örtüşmenin meydana gelmesi 8
Şekil 2. 4 Zaman domeni sinyali ve Fourier dönüşümü 9
Şekil 2. 5 Piksel değerleri 0 ile 255 arasında değişen (256 gri seviyeli) bir görüntü 10
Şekil 2. 6 Orijinal görüntünün dalgacık katsayıları 10
Şekil 2. 7 Dalgacık bölgesinde yaklaşık 25000 dalgacık katsayısı ile yeniden oluşturulan görüntü 11
Şekil 2. 8 Sinyal sıkıştırma ve açma 12
Şekil 2. 9 Nyquist-Shannon örnekleme teoremi 14
Şekil 2. 10 Sıkıştırıcı algılama 14
Şekil 2. 11 SA yaklaşımı 15
Şekil 2. 12 Klasik örnekleme teoremi 15
Şekil 2. 13 Sıfırdan farklı katsayılarla ilişkili kolonlar(OMP) 18
Şekil 2. 14 DEA Algoritması ile sinyali geri çatım işlemi 19
Şekil 2. 15 Dik Eşleştirme Arayış Algoritması(Orthogonal Matching Pursuit Algorithm)21
Şekil 2. 16 AKD ile OMP sonucu 24
Şekil 2. 17 AK3D ile OMP sonucu 24
Şekil 3. 1 Kriptoloji genel yapısı 25
Şekil 3. 2 Şifreleme ve şifreyi çözme işlemleri 26
Şekil 3. 3 Geleneksel şifrelemenin basit modeli 27
Şekil 3. 4 Simetrik şifreleme 29
Şekil 3. 5 Blok şifreleme 30
Şekil 3. 6 Blok şifrelemede doğru/yanlış gösterimi 31
Şekil 3. 7 Blok şifrelemede şifrelemenin gösterimi 31
Şekil 3. 8 Blok şifrelemede şifre çözmenin gösterimi 32
Şekil 3. 9 DES yapısı 33
Şekil 3. 10 DES yapısı 35
Şekil 3. 11 f fonksiyonu 36
Şekil 3. 12 DES anahtar üretimi 37
Şekil 3. 13 3DES yapısı 38
Şekil 3. 14 AES yapısı 39
Şekil 3. 15 Bayt değiştirme 40
Şekil 3. 16 Bayt değiştirme 41
Şekil 3. 17 Satır kaydırma 41

Şekil 3. 18 Sütun karıştırma 42
Şekil 3. 19 Tur anahtarıyla toplama 42
Şekil 3. 20 Anahtar oluşumu 43
Şekil 3. 21 Açık anahtar(Asimetrik) yöntemi..... 44
Şekil 4. 1 Sıkıştırıcı algılama ile simetrik kriptografi uygulama algoritması 49
Şekil 4. 2 Sıkıştırıcı algılama ile simetrik kriptografi uygulama çıktısı 50
Şekil 4. 3 Sıkıştırıcı algılama ile simetrik kriptografi uygulama sonucu, orijinal ve şifreli işaret,doğru ve yanlış anahtar sonucu 50
Şekil 4. 4 Kompleks,rastgele ve dağıtılmış 2 faz anahtarı 50
Şekil 4. 5 Sıkıştırıcı algılama ile simetrik kriptografi uygulama çıktısı 51
Şekil 4. 6 Sıkıştırıcı algılama ile simetrik kriptografi uygulama sonucu,orijinal ve şifreli işaret,doğru ve yanlış anahtar sonucu 51
Şekil 4. 7 Kompleks,rastgele ve dağıtılmış 2 faz anahtarı 51
Şekil 4. 8 Sıkıştırıcı algılama ile hibrit (simetrik+asimetrik) kriptografi (aynı anahtarları kullanarak) uygulaması program algoritması..... 54
Şekil 4. 9 Sıkıştırıcı algılama ile hibrit kriptografi (aynı anahtarları kullanarak) uygulama çıktısı..... 55
Şekil 4. 10 Sıkıştırıcı algılama ile hibrit kriptografi (aynı anahtarları kullanarak) uygulama sonucu,orjinal ve şifreli işaret,doğru ve yanlış anahtar sonucu... 55
Şekil 4. 11 Kompleks,rastgele ve dağıtılmış 2 faz anahtarı 56
Şekil 4. 12 Sıkıştırıcı algılama ile hibrit kriptografi(aynı anahtarları kullanarak) uygulama çıktısı..... 56
Şekil 4. 13 Sıkıştırıcı algılama ile hibrit kriptografi (aynı anahtarları kullanarak) uygulama sonucu,orjinal ve şifreli işaret,doğru ve yanlış anahtar sonucu... 57
Şekil 4. 14 Kompleks,rastgele ve dağıtılmış 2 faz anahtarı 57
Şekil 4. 15 Optik şifreleme 59
Şekil 4. 16 Sıkıştırıcı algılama ile hibrit (simetrik+asimetrik) optiksel kriptografi (eşlenik anahtarlar ve fft dönüşümü kullanarak) uygulaması program algoritması.. 60
Şekil 4. 17 Sıkıştırıcı algılama ile hibrit optiksel kriptografi (eşlenik anahtar ve fft kullanarak) uygulama çıktısı..... 61
Şekil 4. 18 Sıkıştırıcı algılama ile hibrit optiksel (eşlenik anahtarlar ve fft kullanarak) uygulama sonucu,orjinal ve şifreli işaret,doğru ve yanlış anahtar sonucu... 62
Şekil 4. 19 Kompleks,rastgele ve dağıtılmış 2 faz anahtarı 62
Şekil 4. 20 Sıkıştırıcı algılama ile hibrit optiksel kriptografi (eşlenik anahtar ve fft kullanarak) uygulama çıktısı..... 63
Şekil 4. 21 Sıkıştırıcı algılama ile hibrit optiksel kriptografi (eşlenik anahtarlar ve fft kullanarak) uygulama sonucu,orjinal ve şifreli işaret,doğru ve yanlış anahtar sonucu 63
Şekil 4. 22 Kompleks,rastgele ve dağıtılmış 2 faz anahtarı 64
Şekil 4. 23 Örnek 1 orijinal görüntü 66
Şekil 4. 24 Örnek 1 sıkıştırıcı algılama ile şifrelenmiş ve sıkıştırılmış görüntü 66
Şekil 4. 25 Örnek 1 sıkıştırıcı algılama ve yanlış anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü 67
Şekil 4. 26 Örnek 1 sıkıştırıcı algılama ve doğru anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü 67
Şekil 4. 27 Örnek 2 orijinal görüntü 68
Şekil 4. 28 Örnek 2 sıkıştırıcı algılama ile şifrelenmiş ve sıkıştırılmış görüntü 68

Şekil 4. 29 Örnek 2 sıkıştırıcı algılama ve yanlış anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü 69
Şekil 4. 30 Örnek 2 sıkıştırıcı algılama ve doğru anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü 69
Şekil 4. 31 Örnek 3 orijinal görüntü 70
Şekil 4. 32 Örnek 3 sıkıştırıcı algılama ile şifrelenmiş ve sıkıştırılmış görüntü 70
Şekil 4. 33 Örnek 3 sıkıştırıcı algılama ve yanlış anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü 71
Şekil 4. 34 Örnek 3 sıkıştırıcı algılama ve doğru anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü 71
Şekil 4. 35 Sıkıştırıcı Algılama İle Hibrit(Simetrik+Asimetrik) Optiksel Kriptografi (Ses İşareti) uygulama sonucu,orijinal ve şifreli işaret,doğru ve yanlış anahtar sonucu 72
Şekil 4. 36 Kompleks,rastgele ve dağıtılmış 2 faz anahtarı 73
Şekil 5. 1 ER algoritması 74
Şekil 5. 2 Zhang ve Wei tarafından [72] önerilen kriptografi yönetminin akış diyagramı 76
Şekil 5. 3 Zhang ve Wei tarafından [72] önerilen kriptografi için oluşturulan programın sonucu 77
Şekil 5. 4 Zhang ve Wei tarafından [72] önerilen kriptografi için oluşturulan programın uygulama çıktıları 77
Şekil 5. 5 Zhang ve Wei tarafından [72] önerilen kriptografi için oluşturulan programın sonucu 78
Şekil 5. 6 Zhang ve Wei tarafından [72] önerilen kriptografi için oluşturulan programın uygulama çıktıları 78
Şekil 5. 7 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın çıktısı 79
Şekil 5. 8 Kriptoanaliz için önerilen yönetimin akış diyagramı 80
Şekil 5. 9 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın orijinal işaret,şifrelenmiş ve sıkıştırılmış işaret, doğru anahtar ve yanlış anahtar sonuçları 81
Şekil 5. 10 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programında oluşturulan 2 adet rastgele ve kompleks faz+genlik anahtarları 81
Şekil 5. 11 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın çıktısı 82
Şekil 5. 12 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın orijinal işaret, şifrelenmiş ve sıkıştırılmış işaret, doğru anahtar ve yanlış anahtar sonuçları 82
Şekil 5. 13 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programında oluşturulan 2 adet rastgele ve kompleks faz+genlik anahtarları 83
Şekil 5. 14 Faz+genlik anahtarındaki sadece tek değerin hatalı olması sonucu 83
Şekil 5. 15 Faz+genlik anahtarındaki sadece tek değerin hatalı olması ile sıkıştırıcı algılama ve hibrit kriptografi programın orijinal işaret,şifrelenmiş ve sıkıştırılmış işaret,doğru anahtar ve yanlış anahtar sonuçları 84

Şekil 5. 16 Faz+genlik anahtarındaki sadece tek değerin hatalı olması ile sıkıştırıcı algılama ve hibrit kriptografi programında oluşturulan 2 adet rastgele ve kompleks Faz+genlik anahtarları.....84
Şekil 5. 17 Dirac delta fonksiyonu 87
Şekil 5. 18 Kriptoanaliz için girişe uygulanan orijinal görüntü.....88
Şekil 5. 19 Kriptoanaliz sonucu (sadece faz anahtarı ve ayrıştırılmamış toplam anahtar kullanarak.....88
Şekil 5. 20 Kriptoanaliz sonucu (Önerdiğimiz yöntem, faz+genlik anahtarı ve ayrıştırılmamış fourier ile anahtar kullanarak).....89

ÇİZELGE LİSTESİ

Sayfa

Çizelge 2. 1 AKD ve AK3D işlem karşılaştırması 23
Çizelge 3. 1 DES için ilk permütasyon örneği 34
Çizelge 3. 2 Genişletme tablosu 36
Çizelge 3. 3 AES tur sayısı 38
Çizelge 6. 1 Gerçekleştirilen uygulamaların analizi 93

ÖZET

EŞ ZAMANLI SIKIŞTIRICI ALGILAMA VE KRİPTOGRAFİ

Ertan ATAR

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Doktora Tezi

Tez Danışmanı: Yrd. Doç. Dr. Lale ÖZYILMAZ

Eş Danışman: Prof.Dr. Okan K. ERSOY

Sıkıştırıcı algılama, ilk olarak 2004 yılında Emmanuel Candes tarafından manyetik rezonans görüntüleme problemleri ile uğraşırken geliştirilmiştir. Test verilerinin, Nyquist şartlarını sağlamayacak kadar az bir veri sayısı ile yeniden oluşturulabildiğini keşfetmiştir.

Günlük hayatta ses, görüntü, radar, video vb. gibi birçok uygulamada kullanılan sinyaller genel olarak bir alanda seyrek veya sıkıştırılabilirlerdir. İlk olarak işaretler sıkıştırılmadan alınır ve seyrekleştirilmiş bir işarete dönüştürülür. Örnek vermek gerekirse, az sayıdaki dönüşüm katsayıları bilginin büyük bölümünü içermektedir. Diğer yandan, sıkıştırıcı algılama, sıkıştırılmış domende seyrekleştirilmiş işaretin nasıl ölçüleceği ve orijinal işaretin nasıl geri elde edileceği ile ilgilenir.

Sıkıştırıcı Algılama teorisi (SA), klasik veri/görüntü sıkıştırma yöntemlerinin verimli olmadığı sinyallerin veya görüntülerin istenen çözünürlüğünün çok altında bir sayıda örnek alınarak yeniden oluşturulabilmesinin mümkün olduğunu öne sürmektedir. Bu yeni veri toplama protokolü klasik yöntemler için gerekli olduğu düşünülen sensör sayısından daha az sayıda sensörle analog bilgiyi sayısal forma çevirmeyi hedeflemektedir.

Sıkıştırıcı algılama, seyrekleştirilmiş, sıkıştırılmış ve ölçülebilir bir işaret dönüşümü yapabilmek için doğrusal bir izdüşüm matrisi (genellikle rastgele) kullanmaktadır. Bu

yolla, yeni görüntü sistemleri, kameralar, algılayıcı ağları gibi analog-sayısal dönüştürücü pek çok uygulama geliştirilebilir. Seyrekleştirilmiş orijinal işaretin iyileştirilmesi sıkıştırıcı algılama teorisi ile bir uyumluluk içinde olmalıdır.

Kriptoloji bilginin şifrelenmesi ve şifrenin çözülmesine verilen yöntemlerin genel adıdır. Veri bütünlüğü, bilgi güvenliği ve bunların gerçeklenmesini sağlayan matematiksel yöntemler kriptolojinin başlıca araştırma konularıdır. Genel olarak kriptografi: simetrik, asimetrik ve hibrit kriptografi olmak üzere üçe ayrılır.

Bu çalışmada, Dik Eşleştirme Arayış Algoritması olarak isimlendirilen bir sıkıştırıcı algılama algoritması ile simetrik, asimetrik ve hibrit optiksel kriptografi uygulaması beraber kullanılmıştır. Önerilen bu çalışmada, eş zamanlı sıkıştırılmış algılama, simetrik, asimetrik kriptografi ve hibrit şifreleme ve şifre çözme başarılmıştır. Optiksel dönüşüm şifreleme ve şifre çözme metodları ile gelişmiş güvenlikli kriptografi özellikleri elde edilmiştir. Böylece, SA algılama matrisi ile çift kompleks, rastgele ve dağıtılmış faz+genlik anahtarlarının beraber kullanılması sonucunda, hem optiksel kriptografi, hem de sıkıştırıcı algılama veri boyutu düşürme (sensör sayısı azaltma) birleştirilmiştir. Kriptoanaliz sonuçları başarı ile sağlanmıştır.

Anahtar Kelimeler: Sıkıştırıcı algılama, dik eşleştirme arayış algoritması, kriptografi, kriptoanaliz, simetrik ve asimetrik kriptografi, optiksel hibrit kriptografi, ayrık kosinüs-III dönüşümü, çift rastgale faz+genlik anahtar şifrelemesi, veri boyutu ve sensör azaltma.

ABSTRACT

SIMULTANEOUS COMPRESSIVE SENSING AND CRYPTOGRAPHY

Ertan ATAR

Department of Electronics and Communications Engineering

PhD. Thesis

Adviser: Asst. Prof. Dr. Lale ÖZYILMAZ

Co-Adviser: Prof. Dr. Okan K. ERSOY

Compressive Sensing was first developed by Emmanuel Candes, Stanford University in 2004 when dealing with problems of magnetic resonance imaging. He put forward that test data could be obtained independently from Nyquist criteria.

In daily life, audio, video, and radar signals used in many applications such as video streams are sparse or compressible in general. Signals are first measured without considering compressibility, and then converted to a sparse signal, for example, by using a small number of transform coefficients containing most of the information. On the other hand, compressive sensing (CS) deals with how to measure the sparse signal in a compressed domain to start with, and how to retrieve the orginal signal by using CS theory.

According to the Compressive sensing theory, if data/image compressing methods are not efficient or resolution of the data is not too poor, original data could be rebuild. Aim of the method is increasing the number of the sensor to take signal and convert to analog data to digital data.

Compressive sensing uses a (usually random) linear projection matrix to convert the sparse signal to a measurable signal which is very much compressed. In this way, new display systems, cameras, sensor networks, and many applications such as the design of analog-digital converter can be developed. Recovery of original sparse signal requires compatibilty with the fundamental CS measurement theory.

Cryptography is the name given to methods used in encryption and decryption of information. Reliability, data integrity, studies on mathematical methods of dealing with information security, issues such as authentication are important research topics in cryptography. Overall cryptography is divided into three types: symmetric, asymmetric, and hybrid cryptography.

In this study, a compressive sensing algorithm called Orthogonal Matching Pursuit (OMP) is used together with symmetric, asymmetric and hybrid optical cryptography. In the method proposed, simultaneous compressed detection, symmetric and asymmetric cryptographic hybrid encryption / decryption is accomplished. Thus, both compressive sensing and data size reduction (reducing the number of sensors) is combined with cryptography by using CS detection matrix together with twice complex, random and distributed phase+amplitude keys as used in optical transform encryption and decryption methods with security-enhanced cryptography features. Cryptoanalysis results generated were successful.

Keywords: Compressive sensing, orthogonal matching pursuit algorithm, cryptography, cryptoanalysis, symmetric and asymmetric cryptography, optical hybrid cryptography, discrete cosine-III transform, double random phase+amplitude key encryption, data size and sensor decrease.

BÖLÜM 1

GİRİŞ

Klasik sinyal işleme tekniklerine göre, ölçüm sinyallerinin doğru geri oluşturulması için sinyalin Nyquist oranında (örneklenen sinyalin frekansının en az iki katında) örneklenmesi gerekmektedir. Örnekleme işlemi sonucunda sinyalin ham ölçümleri elde edilmektedir. Eğer ölçülen sinyal seyrek yani sıkıştırılabilir ise, algılayıcı bir dönüşüm tabanı kullanılarak sinyalin ayrıştırılması sağlanır, neticede sinyalin enerjisinin büyük kısmı az sayıda katsayı ile ifade edilebilmektedir [1-4].

Sıkıştırıcı Algılama (SA) metodunda sinyalin birçok değerinin sıfır olduğu varsayılır. Sinyal uygun uzaya taşınır, bu taşıma işleminde bir bakıma sinyal sıkıştırılır ve dönüştürülür. Alt boyuta indirgenen sinyalin sıfır olmayan katsayıları, temsil yeteneği olacak şekilde seçilir. Yeni uzayın boyutu eski sinyalin sıfır olmayan elemanlarını tarayan uzaydan daha küçük olamaz, olması durumunda eski sinyal alıcıda elde edilemez. Alt uzaya sıkıştırılan sinyal alıcıda tekrar sıkışmış uzay genişletilerek eski durumuna getirilir. Bu işlemler sürecinde bilgi kaybı yaşanmaz. Fakat ileride açıklanacak detaylı kurallara dikkat edilmesi gerekmektedir. Aksi halde bu süreç rastgele bir süreç olduğu için alıcıda çok farklı sonuçlar ortaya çıkacaktır. SA tekniğinden yararlanırken rastgele sinyal kullanılacaktır. Rastgele sinyaller sıkıştırılabilir bir sinyal olmayacağı için bu çalışmada SA yöntemi ile kurulan matrisler sıkıştırma yapmayacaktır. Bu matrisler algılama matrisi olarak kullanılacaktır. Orantılı olarak genişletme işlemi yapıldığında sinyalin alıcıdan mükemmele yakın bir şekilde alındığı gösterilecektir [5-8].

Örnek verilirse; öncelikle resim içindeki her bir pikselin değerini ölçülmektedir. Eğer resim sıkıştırılabilir ise bunu ayrık kosinüs dönüşümüne (AKD) tabi tutulursa, sıkıştırma

sonucunda az sayıda AKD katsayısı ile ölçümü yapılan resmin enerjisinin büyük kısmı ifade edilebilir. AKD katsayıları ve veri içindeki konumları kaydedilerek, geri kalan (sıfıra yakın katsayılar) katsayılar atılır. Az enerjili AKD katsayılarının atılması, resmin kalitesini ve çözünürlüğünü bozmaz.

Klasik sinyal işleme teknikleri dışında, SA teoremi yeni bir sinyal ölçüm ve geri oluşturma yöntemi sunmaktadır. SA kuramına göre sinyal eğer seyrek veya bir dönüşüm tabanında sıkıştırılabilir ise az sayıda rastgele doğrusal ölçümle geri oluşturulabilir. Sinyalin doğru geri oluşturulması için gereken ölçüm sayısı sinyalin seyreklik seviyesi ile doğrudan ilişkilidir [9-11].

Bu tez çalışmasında, SA ile yeni bir kriptografi uygulaması amaçlanmıştır. Kriptoloji bir verinin şifrelenmesi/deşifrelenmesi (kriptografi) ve şifrelenmiş metinin analizini (kriptoanaliz) içeren matematiğin bir dalıdır. Teknolojideki hızlı gelişmeler verilerin güvenli bir şekilde iletilmesinin önemini artırmıştır. Kriptoloji, ilk olarak yaklaşık 4000 yıl önce kullanılmıştır. Kripto tarihçisi David Kahn'a göre ilk kriptografik belge M.Ö. 1900'lere dayanan bir lordun hayatını anlatan bir hiyegroliftir. 2000 yıl önce Roma İmparatoru Julius Ceasar tarafından oluşturulan ve simetrik anahtar şifrelemenin ilk örneği sayılan Ceasar şifresi alfabedeki harfleri kendinden üç sonraki harfe götüren basit yerine koyma şifresidir. Şifre anlamına gelen ingilizce "cipher" ve Fransızca "chiffre" sözcükleri bu dillere Arapça "cifr" ya da "cifir" sözcüklerinden geçmiştir. Şifre analizi (kriptoanaliz) ile ilgili çalışmaları ilk olarak M.S. 600'lü yıllarda Araplar yapmıştır. Abdurrahman el-Halil ibn-i Ahmed"in M.S. 718 yılında yazdığı "Kitab-ül Muamma" adlı kitap Arapların şifreleme konusunda yazdıkları bir eser olup Bizans İmparatoru tarafından gönderilen Yunanca bir şifreli mektubun çözümünü de içermektedir. 1412 yılında Abdullah Kalkaşandî tarafından tamamlanan, "Subhu'l Aşâ" adlı 14 ciltlik ansiklopedinin de şifreleme ile ilgili bölümleri bulunmaktadır. Diplomasi ve askeri haberleşmelerde kullanılan gizli haberleşme özellikle 2. Dünya savaşında büyük rol oynamıştır. Sanayi devrimine kadar ilkel yöntemlerle oluşturulan şifreleme sistemleri, makinelerin icadı ile boyut değiştirmiş ve teknolojinin hızı ile doğru orantılı olarak kırılamayacak şifreleme sistemleri oluşturma çabaları devam etmiştir. İnternette yollanan veri paketleri birçok halka açık networklerden geçmektedir, bu da bu paketlere ulaşmayı mümkün kılmaktadır. Son derece gizli bilgiler internette nakil

olurken, bu durum önemli bir problem oluşturmaktadır. Bu tür bilgileri korumak mümkün olmadıkça, internet iş yapmak veya gizli, şahsi yazışmalarda bulunmak için asla güvenli bir yer olmayacaktır. Bilgi güvenliği başkası tarafından dinlenme, bilginin değiştirilmesi, kimlik taklidi gibi tehditlerin ortadan kaldırılması ile sağlanır ve bu amaçla kullanılan temel araç kriptografidir. İş te bu noktada, kriptografi devreye girer, bilgi güvenliğini inceler ve anlaşılabileni anlaşılamaz yapar. [12-16].

1.1 Literatür Özeti

Amir ve Esther tarafından [17] de belirtilen çalışmada, EEG işaretini ölçüm vektöründe boyut sıkıştığı için şifrelenmiş işaret olarak kullanılacağını öne sürülmektedir. Herhangi bir kriptoanaliz çalışması yapılmamıştır.

Minal ve Rajankar'ın [18] deki uygulamada, algılama matrisinin satırları ve sütunları yer değiştirilerek şifreleme yapılmıştır. Şekiller bozuk olarak geri elde edilmiştir. Kriptoanaliz uygulaması belirtilmemiştir.

Rachlin ve Baron tarafından [19] da belirtilen ve Dwork, McSherry ve Talwar tarafından [20] de belirtilen çalışmalarda sıkıştırıcı algılamanın Nyquist teoreminin aksine data güvenliği ve şifreleme konusunda da kullanabileceğini öne sürmüştür. Burada şifreleme ve şifre çözme için algılama matrisinin kullanabileceğini açıklamışlardır.

Orsdemir, Altun ve Sharma tarafından [21] deki çalışmada rastgele üretilen algılama matrisinin şifrelemedeki anahtar görevini üstlenebileceği belirtilmektedir.

Ramezani, Seyfe ve Bafghi tarafından [22] de belirtilen çalışmada, ölçüm sayısı (M) ve seyreklik derecesi (K) için $M > 2K$ özelliğini sağlaması şartıyla algılama matrisinin şifreleme ve şifre çözmede kullanabileceğini göstermişlerdir.

Zhang, Wong, Xiao ve Li tarafından [23] deki çalışmada, algılama matrisi kullanarak şifreleme ve şifre çözme uygulaması gerçekleştirilmiştir.

Zhang, Ren, Feng ve Qian tarafından [24] deki çalışmada, algılama matrisinde transform ile şifreleme ve şifre çözme işlemi uygulanmıştır.

Mo, Zhang, Zheng ve Zhou tarafından [25] de belirtilen çalışmada, algılama matrisinde hadamard matrisi kullanılarak şifreleme ve şifre çözme yapılacağını göstermişlerdir.

Peng, Zhang, Wei ve Yu tarafından [26] de yapılan çalışmada sadece faz anahtarı özelliği ile sıkıştırıcı algılama olmadan faz şifreleme uygulaması yapılmıştır.

Peng, We ve Zhang tarafından [27] de belirtilen çalışmada, Carnier, Usategui, Arcos ve Juvels tarafından [28] de belirtilen çalışmada ve son olarak da Frauel, Castro, Naughton ve Javidi tarafından [29] de belirtilen çalışmada literatürde en fazla kullanılan kriptoanaliz yöntemleri incelendiğinde sıkıştırıcı algılama uygulaması için faz geri elde yöntemi ile saldırı ve dirac delta fonksiyonu ile saldırı yöntemi bulunmaktadır.

1.2 Tezin Amacı

Bu tez çalışmasında, sıkıştırıcı algılama ile yeni bir kriptografi uygulaması ve bu sayede haberleşme alanındaki en önemli iki problem olan efektif/etkin/verimli işaret işleme ve bu işareti güvenli olarak iletme konularına alternatif çözüm araştırması amaçlanmaktadır. Bu çalışmalar esnasında 1-boyutlu ve 2-boyutlu farklı girişler için sıkıştırıcı algılama kullanarak (sinyalin uzunluğunun azaltılarak) ataklara (saldırılara) dayanıklı simetrik, asimetrik ve hibrit optiksel kriptografi uygulaması geliştirilmesi hedeflenmektedir. Ortaya çıkacak uygulamanın en çok kullanılan ataklara dayanıklı olması gerekmektedir.

1.3 Hipotez/Orijinal Katkı

Bu tez çalışmasının ilerleyen bölümleri aşağıda şekilde planlanmıştır.

İlk önce tezin temelini oluşturması için; sıkıştırıcı algılama ve dik eşleştirme arayış algoritması hakkında detaylı bilgi verilecektir, daha sonraki bölümde ise kriptografi hakkında detaylı anlatım yapılacaktır. Temel bilgilerden sonra ise sıkıştırıcı algılama ve kriptografi çalışmasında gerçekleştirdiğimiz 6 uygulamanın detayları, kriptoanalizi ve sonuçları açıklanacaktır.

Bütün uygulamalarımızda, farklılık ve geniş alanlarda kullanılabilmesi için, sıkıştırıcı algılama ile kriptografi uygulamasının giriş işareti olarak, text (tüm karakterler) işaretleri, ses işaretleri ve görüntü işaretleri üzerinde çalışılacaktır.

Bu çalışmada Sıkıştırıcı Algılama (SA) olarak, Dik Eşleştirme Arayış algoritması (Orthogonal Matching Pursuit) kullanılmıştır. Optiksel dönüşüm ile elde ettiğimiz

kriptografi sayesinde gelişmiş güvenlik özellikleri sağlanmıştır. Bu çalışma ile, sıkıştırıcı algılama, 4f optik özelliği ve çift kompleks, rastgele ve dağıtılmış faz+genlik anahtarlarının beraber kullanılması sayesinde, hem veri boyutu azaltma (sensör sayısı azaltma) hem de kriptoanaliz sonuçları başarı ile sağlanmış hibrit optiksel kriptografi elde edilmiştir. Referans [26-29] de sadece faz anahtarı kullanıldığı için faz geri elde yöntemi kullanılarak yapılan saldırılarda kriptoanaliz sonucunda başarı sağlanamamaktadır. Uygulamamızda ise anahtarların sadece faz ile değil, dağıtılmış faz+genlik anahtarları ve 4f optiksel uygulaması yapılmıştır. Gerçekleştirdiğimiz uygulamalarda sıkıştırıcı algılamada kullanılan dönüşüm matrislerinde ilave olarak orijinallik, süre ve doğruluk bakımından daha iyi sonuç için Ayrık Kosinüs-III Dönüşümü (AK3D) de kullanılmıştır. Ayrıca algoritma sonucu elde ettiğimiz *N* ve *M* boyutunda olan anahtarların bütün değerleri farklı olmasa bile sadece tek değerinin farklı olması durumunda da şifrelenmiş ve sıkıştırılmış işaretten orijinal işaretin geri elde edilememektedir. Bu durum sayesinde kaba kuvvet saldırılarına karşı çok güvenli olmaktadır.

BÖLÜM 2

SIKIŞTIRICI ALGILAMA

2.1 Nyquist Örnekleme Kuramı

Seyrek sinyal örnekleme tekniği Nyquist örnekleme tekniğinden az sayıda örnek alarak iletişimde verimliliği arttırır. İlk olarak Nyquist örnekleme sonrasında seyrek örnekleme yöntemi verilecektir. Sürekli bir sinyali ayrıklaştırarak ayrık bir sinyal elde etmek örnekleme ile gerçekleştirilir. Haberleşme sistemlerinde özellikle sayısal haberleşmede örnekleme çok önemlidir. Gerek sinyali geri elde edebilme gerek sinyalden alınan örnekler iletim hızı ve iletişimin verimliliği konusunda dikkat çekicidir. Nyquist Örnekleme sinyalden belirli aralıklarla alınan örneklerin tekrar sinyali geri elde ederken haberleşme açısından sinyalden alınacak örnekleri minimuma indirgeyen teoremdir [4], [30-33].

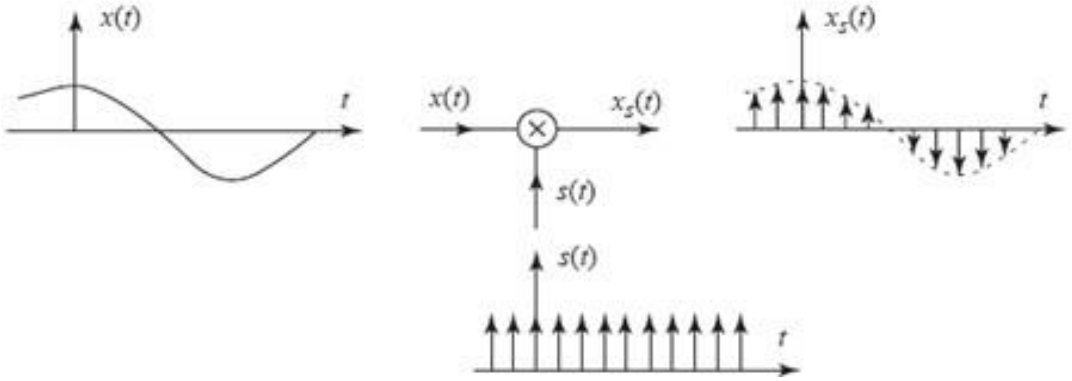
Sürekli sinyali *x(t)* maksimum frekansı *f* örneklenerek *x[k]=x(k)* örneklenmiş frekans

$$f_S=\frac{1}{Ts} \tag{2.1}$$

Nyquist teoremine göre sinyali tekrar geri elde edebilmek için gerekli koşul Denklem 2.2'de gösterilmiştir.

$$f_s \geq 2f \tag{2.2}$$

Örnekleme teoreminin grafiksel olarak gösterimi Şekil 2.1' de gösterilmiştir.



Şekil 2.1 Örnekleme teoremi [38]

$$s(t) = \sum_{k=-\infty}^{\infty} \delta(t - kT_s) \quad (2.3)$$

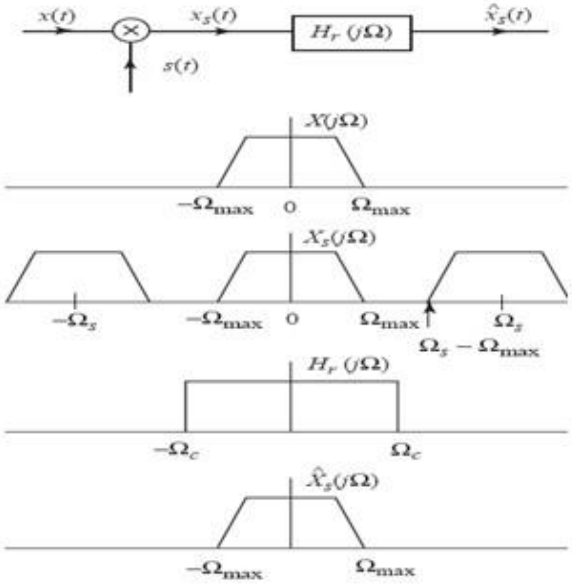
$$x_s(t) = x(t)s(t) = x(t)\sum_{k=-\infty}^{\infty} \delta(t - kT_s) \quad (2.4)$$

$$X_s(j\Omega) = \frac{1}{2\pi} X\,(j\Omega) * S(j\Omega) \quad (2.5)$$

$$\Omega_s = \frac{2\,\pi}{T_s} \quad (2.6)$$

T_s, örnekleme peryodu, Ω_s, Nyquist için örnekleme bant genişliği ve $x(t)$ sürekli sinyal olmak üzere, yukarıdaki denklemlerde sürekli sinyalin Nyquist örnekleme teoremine göre örneklenmesi gösterilmektedir.

$\Omega_s - \Omega_{max} > \Omega_{max}$ *ve* $\Omega_s > 2\Omega_{max}$ şartıyla $X_S\,(j\Omega)$ sinyalinin bant genişliği, bu sinyalin örneklenmiş hali ve sinyali geri elde etme yöntemi Şekil 2. 2'de gösterilmiştir.

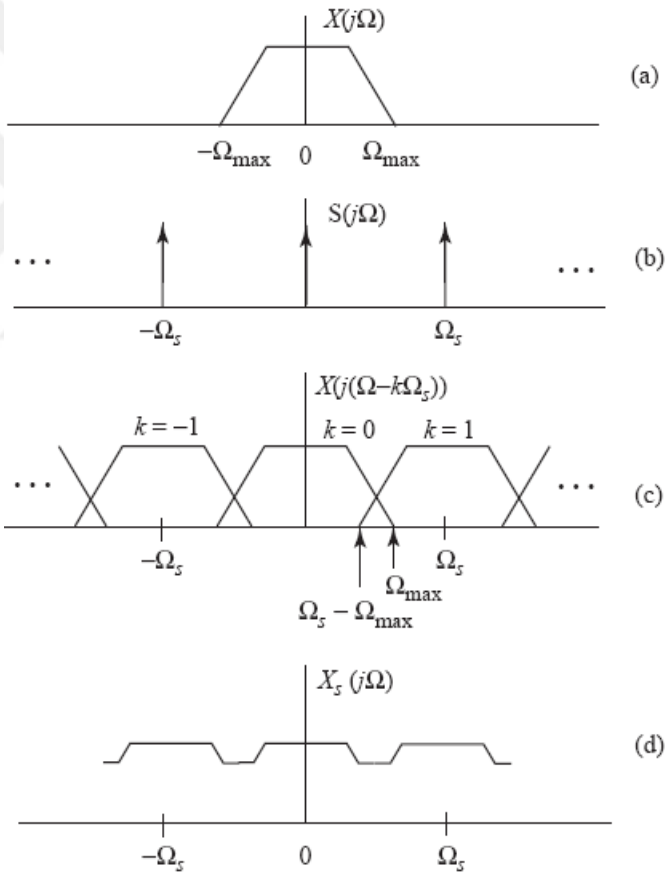


Şekil 2.2 Örneklenmiş sinyalin geri elde edilmesi [38]

Şekil 2.2'den görüldüğü gibi Nyquist hızından daha fazla örnek alınarak sinyal örneklenmiş ve alçak geçiren bir filtre ile sinyal tekrar geri ele edilmiştir. Örnekleme Nyquist hızından az olursa sinyali nasıl geri elde edilebilir mi konusu aşağıda açıklanmıştır.

$\Omega_s - \Omega_{max} < \Omega_{max}$ ve $\Omega_s < 2\,\Omega_{max}$ (2.7)

Seçilen örnekleme frekansı sinyalin maksimum frekansından büyük olabilir ancak sinyalin maksimum frekansının iki katından küçük olacak şekilde seçilirse, Şekil 2.3'te sinyali tam olarak geri elde edilemeyeceği gösterilmekte ve örtüşmenin nasıl meydana geldiği açıklanmaktadır.



Şekil 2.3 Elde edilmiş sinyalde örtüşmenin meydana gelmesi [38]

Bazı sinyaller için minimum Nyquist oranında örnek almaya gerek yoktur. Daha az örnek alınarak sinyal geri elde edilebilir. Böylece sayısal iletişimde daha az veri kullanılır. Daha az hesaplama yükü oluşur. Az maliyetle daha çok iş yapılmış olur. Bu sinyaller: seyrek ve seyrekleştirilebilir sinyallerdir [34-37].

2.2 Seyrek Sinyaller (Seyreklik)

Birçok doğal sinyal, uygun bir bölgede ifade edildiği sürece kısa bir gösterime sahiptir. Seyreklikle anlatılmak istenen, herhangi bir sürekli zaman sinyalinin içerdiği bilgi oranı, sinyalin bant genişliğinden çok daha az olması veya bir kesikli zaman sinyalinin kendisinin sonlu uzunluğuna kıyasla çok daha az bir değerle ifade edilebileceğidir.

Daha açık bir şekilde ifade edilirse, birçok doğal sinyalin, uygun bir bölgede ifade edildiğinde bulunduğu bölgeye oranla daha kısa bir gösteriminin olması gerçeği, sinyalin seyrek olması anlamına gelir.

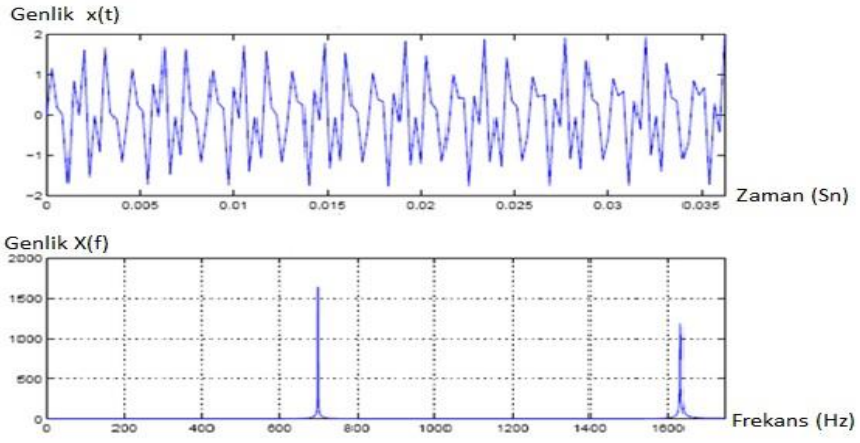
x sinyali *N* uzunluğunda bir ayrık sinyal olsun ve bu sinyalin bazı değerleri sıfır olsun eğer sinyalin *s* adet değeri sıfırdan farklı *ve s<<N* ise x ayrık sinyali seyrek sinyaldir [37-41].

Haberleşmede kullanılan sinyal, sinüs sinyali seyrektir.

Örnek vermek gerekirse;

$$x(t) = \sin(1394\pi t) + \sin(3266\pi t) \tag{2.8}$$

Yukarıdaki sinyal, Şekil 2.4'te gösterildiği üzere, bu sinyal zaman domeninde ilk 36 mili saniye boyunca çizdirilirse, bu sinyal seyrek bir sinyal mi olduğuna bakıldığında, bu sinyalin Fourier transformunu alınarak ve frekans domeninde tekrar çizdirildiğinde, zaman domeninde seyrek olmayan bir sinyal farklı domenlerde seyrek olabilir. Zaman domeninde sonsuz veri olduğu için sayısal işlem yapılamamaktadır. Veri yükünü azaltmak için farklı domene geçilir ve sayısal işlemler bu domende yapılır.

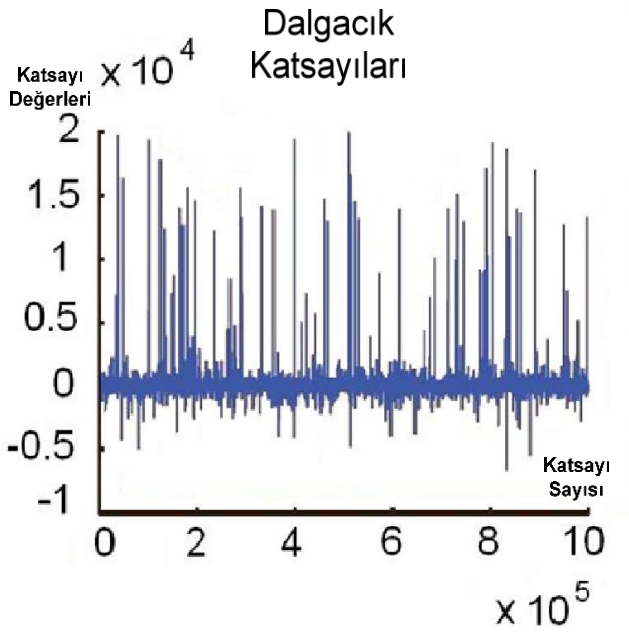


Şekil 2.4 Zaman domeni sinyali ve Fourier dönüşümü

Veri yükünü azaltmak için farklı domene geçilmesi ile ilgili farklı bir örnek vermek gerekirse; dalgacık dönüşümü kullanılabilir. Şekil 2.5'de görülen görüntünün Dalgacık (wavelet) dönüşümü katsayı değerleri Şekil 2.6'da görülmektedir.



Şekil 2.5 Piksel değerleri 0 ile 255 arasında değişen (256 gri seviyeli) bir görüntü



Şekil 2.6 Orijinal görüntünün dalgacık katsayıları

Daha genel olarak, seyreklik etkin ve temel bir sinyal işleme modelleme aracıdır. Örneğin hassas istatistiksel tahmin ve sınıflandırma, etkin veri sıkıştırma ve bunun gibi birçok dalda önemli rol oynar. Seyrekliğin veri toplama işleminde de çok etkin bir rol

aldığını söylenebilir. Seyreklik bir kişinin ne kadar etkin bir şekilde veriyi toplayabileceğini belirler [30, 42-45].

Başka bir şekilde ifade edilirse katsayıların büyük bir çoğunluğu bize herhangi bir kayıp getirmediğinden kolayca atılabilir. Şekil 2.6'da görüleceği üzere dalgacık katsayıların çoğu sıfır veya sıfıra çok yakın değerler teşkil etmektedir. Şekil 2.7'de algısal olarak kaybın anlaşılmasının neredeyse imkansız olduğu Şekil 2.5'deki resmin katsayılarının çoğunluğu atılarak yeniden oluşturulmuştur. Bu durum ile Şekil 2.6'daki katsayıların çok az bir kısmı ile orijinal görüntü tekrar geri elde edilebilmiştir [2, 30].



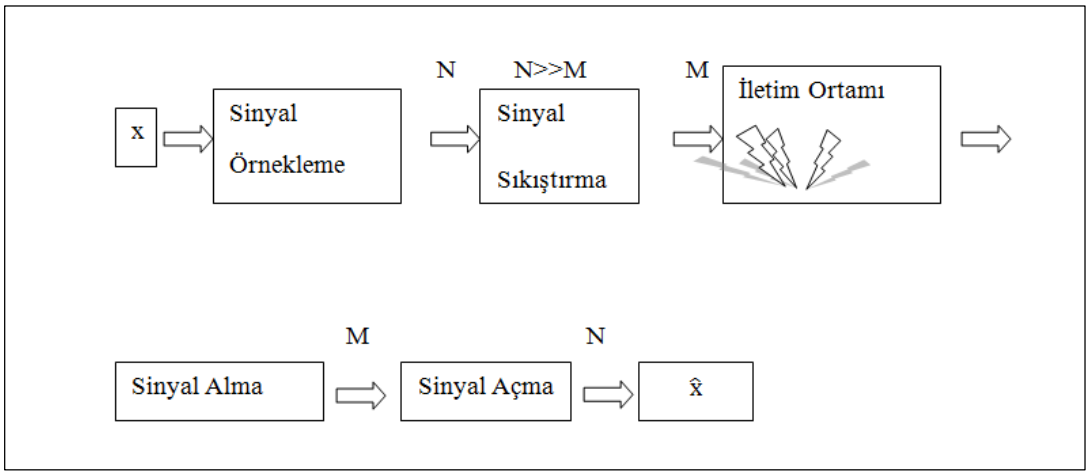
Şekil 2.7 Dalgacık bölgesinde yaklaşık 25000 dalgacık katsayısı ile yeniden oluşturulan görüntü

2.3 Sıkıştırılabilir Sinyal

Eğer bir x sinyali seyrek veya sıkıştırılabilir olarak gösterilebilen bir ψ tabanında temsil edilebiliyor ve x sinyali verilen tanıma göre seyrek bir sinyal ise bu sinyal ve buna benzer sinyallere sıkıştırılabilir sinyaller denir [46, 47].

$$x = \psi s \tag{2.9}$$

sinyal ile temsil edilir.



Şekil 2.8 Sinyal sıkıştırma ve açma

Genel olarak bir sinyalin örneklenmesi sıkıştırılması iletim ortamına verilmesi ve tekrar alıcıda alındıktan sonra geri çatılması işlemleri Şekil 2.8'de gösterilmiştir.

2.4 Sıkıştırıcı Algılama Teorisi

Sıkıştırıcı algılama, son zamanlarda data boyutunun azaltılması ve sensör sistemlerinin azaltılması gibi çok fazla uygulamayı mümkün kılmasından dolayı çok önemli bir araştırma alanı olmuştur. Sıkıştırıcı Algılamanın uygulama alanlarına tomografi, radar, haberleşme, görüntü ve işaret işleme, kablosuz sensör ağları örnek olarak verilebilir. Sıkıştırıcı algılamada ana fikir, bütün sinyalin örneklenmesi yerine dönüşüm katsayılarının az miktarının alınmasıdır. Bu az miktardan kasıt; sıfırdan farklı en büyük dönüşüm katsayılardır. Sıkıştırıcı algılamadaki bu durumun da çok az bir örnekleme ile yapılabildiği ispatlanmıştır [1, 48, 51].

Sıkıştırıcı Algılama yöntemi bir sinyalin kendi boyutundan çok daha az sayıda ölçümlerle yeniden oluşturulabileceğini öne süren, gelişmekte olan bir tekniktir. Seyrek sinyaller algılama matrisleri yardımı ile vektörler içerisine toplanır. Eğer sinyaller yeteri kadar seyrek ise orijinal sinyal başarılı bir şekilde yeniden oluşturulabilir. Örneğin görüntü verileri genelde seyrek veya seyrek hale getirilebilir sinyallerdir. SA uygulamalarında sinyal algılama kısmı basit yöntemlerle gerçekleştirilebilirken, eksik veri setinden sinyalleri yeniden oluşturma kısmında çok yüksek işlem gücü ve karmaşık istatistiksel hesaplamalar gerektirmektedir. Gözlem sonuçlarından veriyi doğru bir şekilde geri dönüştürecek yöntemlerin geliştirilmesi gerekir. Sıkıştırılmış algılamada doğru geri

dönüşümün yapılabilmesi için gerekli ölçüm sayısı *(M),* sinyalin gösterildiği alandaki seyreklik derecesine *(K),* sinyal boyutuna *(N)* ve seyreklik tabanı ile ölçüm matrisi arasındaki karşılıklı uyumuna bağlıdır. SA'nın ölçüm sayısını ne derecede azaltabildiği ve gerekli ölçüm sayısının kestirilebilmesi en önemli noktalardan biridir. Literatürde ölçüm sayısı *(M)* ile ilgili birçok bağıntı kurulmuştur. Genel olarak bu bağıntı en temel haliyle aşağıda belirtilmiştir [1, 2, 5, 30].

$$M \geq (K * log(N)) \tag{2.10}$$

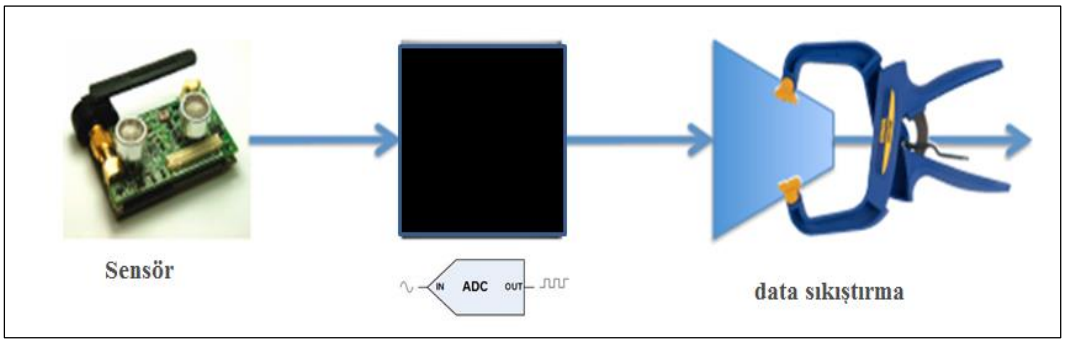
Sinyallerin veya görüntülerin frekans verilerinden toplanması ve yeniden oluşturulması konusundaki klasik yöntemlerde ve yaygın uygulamalarda temel prensip olan Nyquist örnekleme teoremi kullanılır. Nyquist teoremi, belirttiğimiz gibi, bir sinyali veya görüntüyü yeniden oluşturabilmek için sahip olunması gereken Fourier örneği sayısının, sinyalin veya görüntünün istenen çözünürlüğünü karşılaması gerektiğini ifade eder. Sıkıştırıcı Algılama (SA), klasik görüntü sıkıştırma yöntemlerinin verimli olmadığı, sinyallerin veya görüntülerin istenen çözünürlüğünün çok altında bir sayıda örnek alınarak yeniden oluşturulabilmesinin mümkün olduğunu öne sürer. Ayrıca örnekleme teorisinin altında yatan prosedür, örnekleme ve sıkıştırma işlemini eş zamanlı olarak yapmasıdır. Sıkıştırıcı Algılamanın birkaç avantajı aşağıda verilmiştir [49-52].

1- Sensör sayılarının azaltılması, böylece daha az örnek alınarak daha fazla bilgi iletilir.

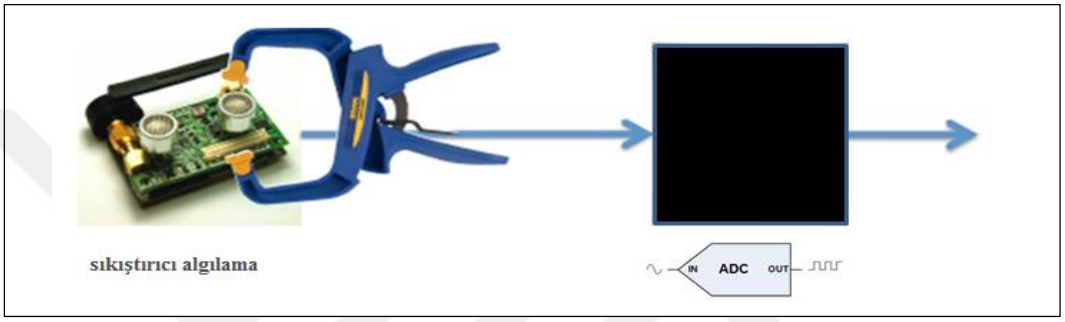
2- Görüntü işleme açısından maliyetli olan yöntemlerde veri yükünü azaltması.

3- Alıcıda algılama işlemlerinde kolaylık sağlaması; daha az veri için algılamada algoritmalara daha fazla süre tanınır.

4- Nyquist örneklemesi için gereken veri yükü fazla olduğu için analog sayısal dönüşümü işlemlerinde sınırlama olur. SA kullanılarak örnek sayısı azaltılır ve dönüşüm işlemleri rahatlar. Güç tüketimi, donanım karmaşıklığı ve süre azalır.



Şekil 2.9 Nyquist-Shannon örnekleme teoremi



Şekil 2.10 Sıkıştırıcı algılama

Şekil 2.9'da ve Şekil 2.10'da gösterildiği üzere, sıkıştırıcı algılamada datayı önce elde edip onu sonra sıkıştırmak yerine data, sıkıştırılmış formda işlem görür. Bu durum sensör sayısının azaltılmasına ve çok daha az örnekleme yapılmasına neden olur, fakat bunun için algılama ve depolamadan önce bilginin dönüşümünü gerektirir. Nyquist örnekleme teoreminde ise, SA'ya göre işaretten çok daha fazla örnek alınır, ADC'den geçirilir, daha sonra ise sıkıştırma yapılır, bu durum sebebiyle seyrek veya seyrekleştirilebilir işaretlerde gereksiz yere fazla örnekleme yapılmış olmaktadır.

SA metodunda sinyalin birçok değerinin sıfır olduğu varsayılır. Sinyal uygun uzaya taşınır bu taşıma işleminde bir nevi sinyal sıkıştırılır ve döndürülür. Alt boyuta indirgenen sinyal sıfır olmayan katsayıları temsil yeteneği olacak şekilde seçilir. Alt uzaya sıkışan sinyal alıcıda tekrar sıkışmış uzayı genişleterek eski haline getirilir. Bu işlemler sürecinde bilgi kaybı yaşanmaz. Büyük boyutlardaki bir x sinyali φ algılama matrisi yardımı ile orjinaline oranla daha küçük bir y matrisi içerisine toplanır. x sinyali, N uzunluğunda, ayrık, tek boyutlu bir sinyal (bilgi vektörü) olsun. K tane değeri sıfırdan farklıysa x sinyali K seyrekliktedir denilir. Klasik örnekleme metodunda Şekil 2.12'de x sinyali $N * N$ boyutundaki birim matris olan ölçüm matrisi ile çarpılmaktadır. Böylece

N tane ölçüm yapılarak $N*1$ boyutunda gözlem vektörü elde edilmektedir. Şekil 2.11'de SA metodunda bu φ sinyalin bütün bileşenlerini almak, yani N tane ölçüm yapmak yerine sadece $M(M << N)$ tane doğrusal ölçüm yaparak sinyali veri kaybına uğramadan geri etmek mümkündür.

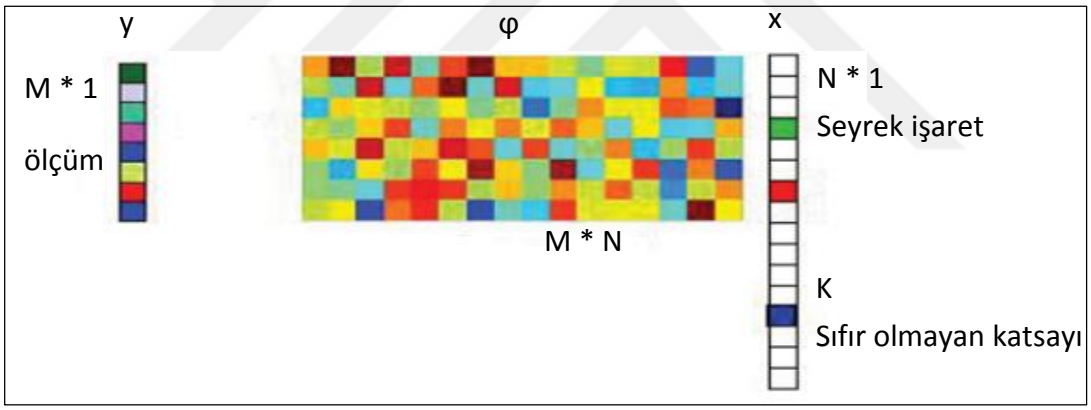
$$y = \langle x, \varphi \rangle; \tag{2.11}$$

y_k, ölçüm vektörü olarak adlandırılan y vektörünün k.elemanlarını; φ_k algılama matrisi olarak adlandırılan φ matrisinin kolon vektörlerini gösterir ise y matrisinin her bir elemanı aşağıdaki gibi hesaplanabilir.

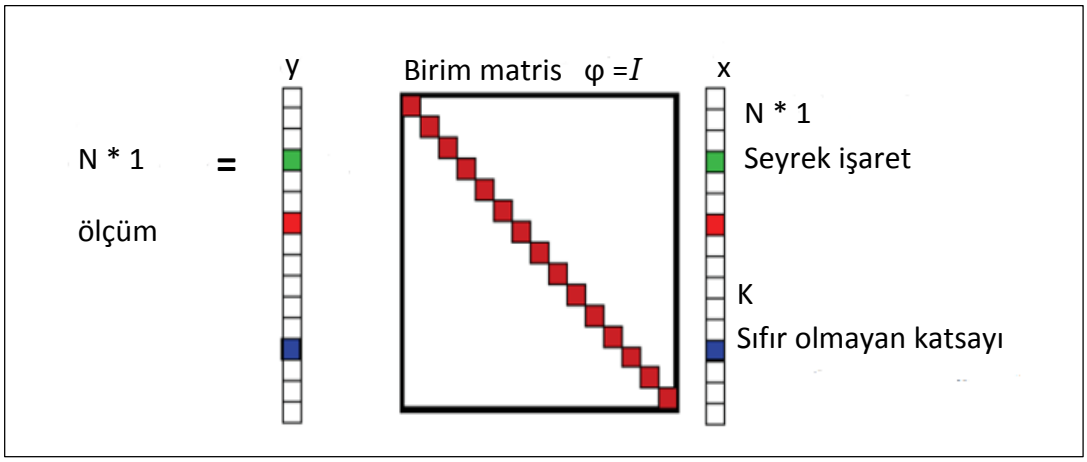
$$y_k = \langle x, \phi_k \rangle, k = 1,2, \dots, m \tag{2. 12}$$

Yukarıdaki denklemde x sinyali istenilen bir φ_k dalga biçimiyle y_k matrisine toplanmakta olup bu standart bir yapıdır. Aşağıdaki eşitlikte gösterildiği gibi y matrisinde toplanan m sayıdaki ölçümler x sinyalinin boyu olan n değerinden çok daha küçük olur.

$$m \ll n, \tag{2.13}$$



Şekil 2.11 SA yaklaşımı [8]



Şekil 2.12 Klasik örnekleme teoremi

m sayıdaki ölçüm, sinyalin sıkıştırılabilme derecesini gösterir ve sıkıştırma oranı olarak adlandırılabilir. Sıkıştırma oranı sinyalin seyrekliği ile orantılıdır ve seyrek bir görüntüde sıkıştırma oranı Nyquist oranının altında bir değere ulaşabilmektedir. m sayıda ölçüm ile orijinal sinyalin yeniden oluşturulabilmesi işleminde ise yapılması gereken bir doğrusal denklem sistemi çözümüdür. φ matrisinin m * n boyutlu, satırları $\varphi_1^*, \ldots, \varphi_m^*$ vektörlerinden oluşan bir algılama matrisi olduğunu düşünülürse; $m < n$ olduğu sürece x sinyalini Denklem 2.11 yardımı ile yeniden oluşturma işlemi çözülür.

Ayrık zamanda sinyal işleme akışında sinyalin örneklenmesi ilk adımdır. Bu adımda, zaman tabanında sürekli olan sinyal eşit aralıklarla örneklenerek ayrık zamanda ölçümler elde edilir. Ölçülen verilerden sinyallerin veya imgelerin geri elde edilmesinde dikkate alınan standart yaklaşım Nyquist-Shannon örnekleme teoremi olarak bilinir. Bu teorem, bant sınırlı bir sinyalin sürekli zamanda geri oluşturulması için gerekli olan örnekleme frekansının sinyal frekansının en az iki katı kadar seçilmesi gerektiğini söyler. Buna göre, klasik sinyal işleme yöntemlerinin kullanıldığı birçok sinyal işleme uygulamasında, sinyal ilk olarak Nyquist-Shannon örnekleme kriterine göre örneklenir. Sinyal gösteriminin ve işlemenin kolay olacağı başka bir tabana (Ör: Fourier, ayrık kosinüs donuşum tabanları) dönüşümü sağlanır. Sinyalin dönüşüm tabanına aktarımıyla sinyal normalden daha az sayıda katsayı vektörü ile ifade edilir. Bu sinyal işlemenin etkin olmayan yöntemidir, çünkü atılan ve işe yaramayan bilgi örnekleme aşamasında elde edilir. Örnekleme ile aldığımız işimize yaramayan bilgiyi en baştan ölçmek istemeyebiliriz. Fakat, örnekleme işlemlerinin yapıldığı gerçek hayattaki donanımların çoğu analog elektronik devrelerdir ve analog devre ile örnekleme yapılan donanımlara akıl ve mantık birimleri eklemek pratikte pek imkanlı değildir. Örneğin, örnekleme aşamasında eğer sinyal doğrudan sadece sıfır olmayan anlamlı katsayıları alınarak örneklenebilseydi, örnekleme daha verimli olabilirdi. Fakat, sinyal ölçümünden önce hangi dönüşüm tabanının kullanılacağı ve dönüşüm tabanı katsayılarından hangilerinin sıfır olduğu daha önceden bilinemediği için bu yaklaşım klasik yöntemlerle pek mümkün değildir.

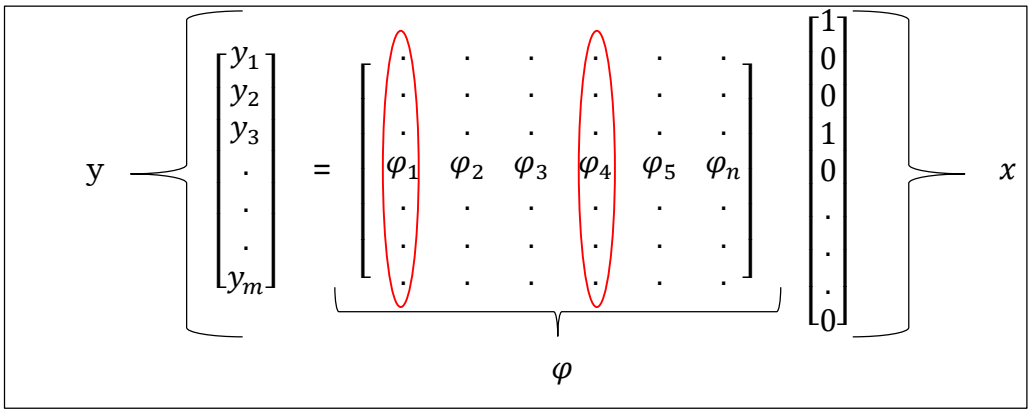
Sıkıştırıcı algılama (SA) bir x sinyalinde örnekleme ve hesaplama maliyetini büyük oranda düşüren sıkıştırmaya dayalı yeni bir sinyal ölçüm ve geri oluşturma yaklaşımıdır. SA tüm sinyali yüksek örnekleme hızı ile ölçmek yerine sinyalin sadece en fazla bilgi

içeren (en yüksek enerjili) kısmını ölçer. Böylece klasik sinyal işleme yöntemlerine göre enerji ve zamandan kazanabilir. SA sinyalin seyrek olarak ifade edilebildiği tabanı kullanır ve sinyal üzerindeki rastgele izdüşümleri kullanarak ölçümler yapar, az sayıda doğrusal ölçümle sinyalin doğru geri oluşturulabileceğini söyler. Rasgele bir x sinyali ($N*1$ boyutunda), bu sinyal seyrek olsun yada olmasın. Eğer $x = \Psi s$ şeklinde ifade edebiliyorsak sinyal seyrek olarak nitelendirilebilir. Burada Ψ, x sinyali için $N*N$ boyutunda uygun taban ve s ise $N*1$ boyutunda seyrek katsayılar vektörü olarak tanımlanmalıdır. φ algılama matrisi, x sinyalinin boyutu N, ölçülen sinyalin boyutu M ise ölçülen sinyal Denklem 2.11 şeklinde ifade edilebilir. Burada M değeri N değerinden çok küçük olacak şekilde seçilmelidir. Önemli nokta algılama matrisinin seçimi ve ne kadar ölçüme ihtiyaç duyulacağıdır. [2, 11, 30, 47].

2.5 Sıkıştırıcı Algılama Dik Eşleştirme Arayış Algoritması (DEA)

Dik Eşleştirme Arayış (Dik Eşleyen Takip) (DEA) - Orthogonal Matching Pursuit (OMP) yöntemi optimizasyon tabanlı olmayan bir yöntemdir [2, 41, 45]. Herhangi bir optimizasyon hedefini amaçlamaz, Şekil 2.13'de gösterildiği üzere örnekleme matrisinin hangi bileşenlerinin sıfırdan farklı elemanlarıyla ilişkili olduğunu belirleyerek seyrek sinyali oluşturur. DEA algoritması ölçüm sayısı olan m, k sayısına göre büyük olduğu sürece k-seyrek bir sinyalden yeniden oluşturma özelliğini başarılı bir şekilde yapabilir. [45].

DEA algoritması boş bir matris ile algoritmayı çalıştırmaya başlar. Örnekleme matrisinin sıfırdan farklı katsayılarla ilişkili olduğunu saptadığı kolonlar ile bu boş matrisi doldurmaya başlar. Her tekrarda bir tane kolon seçerek bu matrisi oluşturur. DEA algoritması açgözlü bir algoritmadır. Çünkü kolonları açgözlü bir yöntemle seçer. Açgözlülük bir sonraki adımı düşünmeden, o anda bulunan seçeneklerden en iyisini seçmektir. Algoritma her tekrarda örneklenmiş sinyal y ile en çok ilişkili kolonu seçer. Bu yöntemle oluşturulan matris Şekil 2.13'de görülmektedir. Algoritma k tekrardan sonra orjinal seyrek sinyalin sıfırdan farklı katsayılarıyla ilişkili kolonlar seçmiş olur [53-56].



Şekil 2.13 Sıfırdan farklı katsayılarla ilişkili kolonlar (OMP)

Bir x sinyali m seyrek bir sinyal ise, y ölçüm vektörüne etkisi bulunan φ örnekleme matrisinin m sayıda kolonunun bulunması gerekmektedir. Algoritmanın her tekrarında y ölçüm vektörünün kalan kısmı ile en çok ilişkili örnekleme matrisi kolonu seçilir. Modified Gram Schmidt (MGS) yöntemi kullanılarak kolonun y ölçüm vektörüne katkısı hesaplanır ve y vektöründen çıkartılır. Bir sonraki tekrarda yukarıdaki anlatılanlar kalan vektörü kullanılarak tekrar yapılır. y ölçüm vektörüne etkisi bulunan φ örnekleme matrisinin m sayıda kolonu bulunduktan sonra orjinal sinyal olan x vektörünün yaklaşık değeri, en küçük kareler denklemi çözülerek bulunur. DEA yeniden oluşturma algoritmasının işleyişi adımlar halinde açıklanmıştır:

Algoritmanın adımları Şekil 2.15'de gösterilmiştir. Algoritmanın ilk adımında kalan vektörü olarak adlandırılan R vektörüne giriş olarak y ölçüm vektörü verilir.

$$R_0 = y \tag{2.14}$$

R vektörünü kalan vektörü olarak adlandırılmasının sebebi algoritmanın tekrarlarında y ölçüm vektörüne etkisi bulunan φ örnekleme matrisinin s_t. kolonu R_t vektöründen çıkarılmasıdır. Buradaki s_t vektörü, φ algılama matrisi ile y ölçüm vektörü arasında en yüksek ilişkiye sahip vektörü temsil etmektedir. Çıkarma işlemi sonucu elde ettiğimiz kalan vektörü R vektörüne giriş olarak verilerek algoritmanın tekrar etmesi sağlanır. İndis seti olarak adlandırdığımız S vektörü ilk adımda boş kümedir.

$$S = \emptyset \tag{2.15}$$

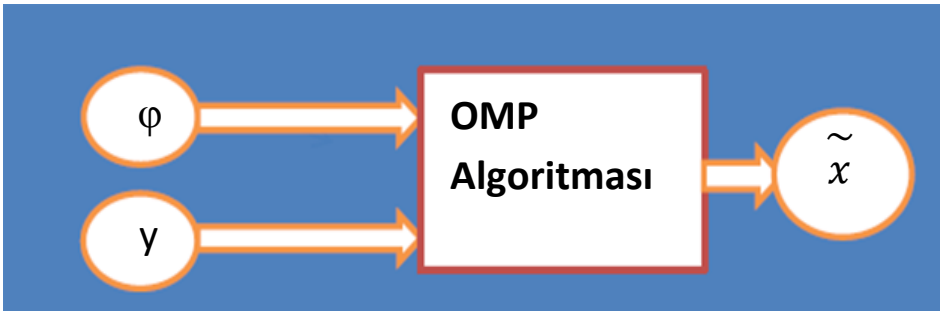
İkinci adımdaki optimizasyon problemi sonucu bulunan s_t değerleri her iterasyonda φ örnekleme matrisinin bir kolonunun indisini vererek bu vektörü dolduracaktır. Tekrar sayacı olarak adlandırdığımız t değeri algoritmanın ilk adımında 1 değerini alacaktır.

$$t = 1 \tag{2.16}$$

Algoritmanın her tekrarında bu değer 1 artırılacaktır. Bu sayacın en son aldığı değer, orjinal x sinyalinin m seyreklik sayısına eşit olmalıdır. Parametreler başlangıç değerlerine alındıktan sonra ikinci adıma geçilir. Algoritmanın ikinci adımında optimizasyon probleminin çözülmesi gerekir: Algoritmanın t. değerine göre atanan R_{t-1} vektörü ile φ örnekleme matrisinin kolonu Denklem 2.17'de tek tek iç çarpım işlemine sokulur. Bu iç çarpım işlemi sonuçları arasındaki en büyük değer seçilir.

$$S_t = argmax_{i=1,\dots,n} \left| \langle R_{t-1}, \varphi_j \rangle \right| \tag{2. 17}$$

En büyük çarpım değerini oluşturan j değeri bu problemin çözümüdür ve bu değer s_t değişkenine atanır. Orjinal x sinyali m seyrek bir sinyal olduğuna göre, bu durumda orijinal x sinyalinin m sayıda sıfır olmayan katsayısı vardır. x sinyalinin sıfır olmayan katsayısı ile φ ve R matris çarpımı işleminde sıfır olmayan katsayı ile çarpılacak φ örnekleme matrisinin kolonunun çarpımı sonucu, y vektörünün ilgili satırında büyük bir değer olarak yer alır. Sıfır olan katsayıların ise matris çarpımı işleminde bu sıfır olan katsayı ile çarpılacak φ örnekleme matrisinin kolonunun çarpımı sonucu, y vektörünün ilgili satırında 0 olarak yer alır. Dolayısıyla denklemdeki optimizasyon probleminde y vektörü ile φ örnekleme matrisinin j. kolonunun iç çarpımını en yüksek yapan j değeri, x vektörünün sıfır olmayan katsayısının satır numarasını verir. Algoritmanın üçüncü adımında, ikinci adımda bulunan s_t değerleri indis seti vektörüne eklenecektir. İndis seti vektörü algoritmanın ilerleyen adımlarında kullanılacaktır. Algoritmanın dördüncü adımında; dikleştirme yapmak ve bir önceki değerleri bir daha seçmemek için, φ örnekleme matrisinin j. kolonunun QR açılımı ile q değerleri bulunacaktır. QR açılımı yöntemi olarak MGS algoritması kullanılmıştır. MGS algoritması ile q değerleri aşağıdaki verilen denklemler yardımıyla hesaplanır [41, 57-60].



Şekil 2.14 DEA Algoritması ile sinyali geri çatım işlemi

Elde edilen q değerleri beşinci adımda kullanılarak kalan vektörlerinin hesaplanması sağlanacaktır. Algoritmanın beşinci adımında, dördüncü adımda elde ettiğimiz q değerleri ve bir önceki algoritma tekrarının beşinci adımında oluşturulan artık vektör kullanılarak Denklem 2.11'de yeni artık vektör hesaplanır.

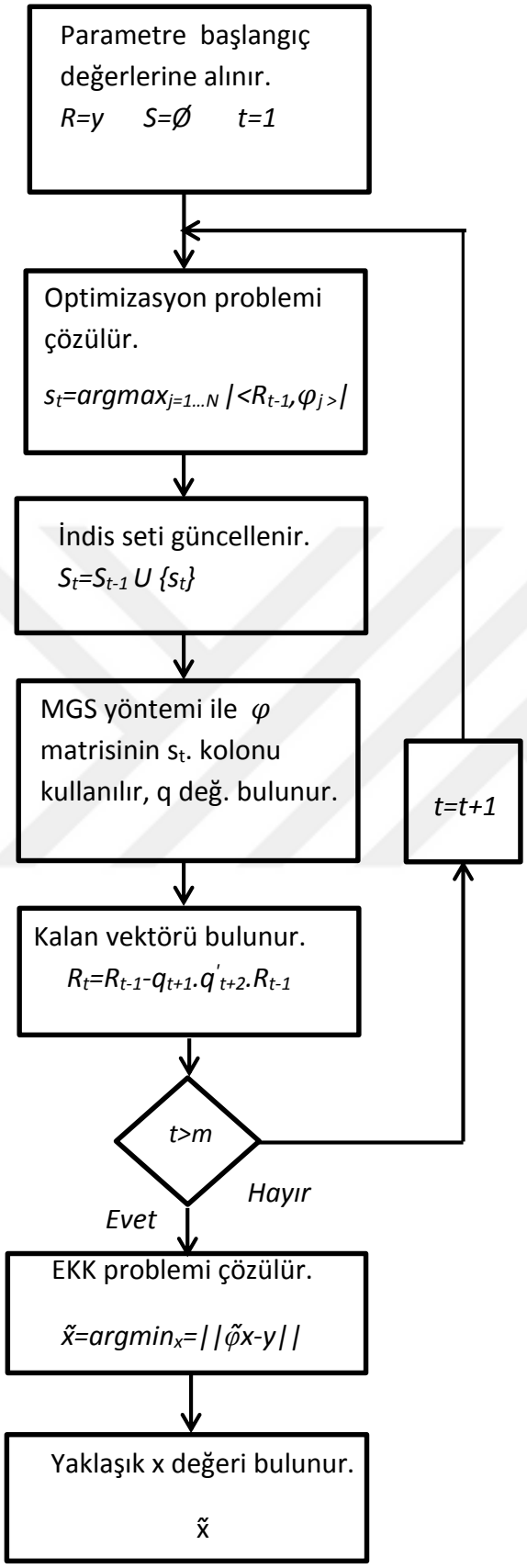
$$R_{t=}R_{t-1} - q_{t+1.}q'_{t+1}.R_{t-1} \quad (2.\ 18)$$

Bu denklem ile algoritmanın ikinci adımında bulunan s_t kolonlarının R_t matrisine katkısı çıkarılmaktadır. Dolayısıyla yeni oluşturulan R vektörü ile algoritma bir sonraki tekrarında ikinci adıma girdiğinde, yeni bir s_t değeri bulacaktır. Algoritmanın altıncı adımında tekrar sayacı t, seyreklik seviyesi m'den küçük ise tekrar sayacı (çıkar) 1 artırılır ve ikinci adıma dönülür. Tekrar sayacı m değerine eşit veya büyük ise yedinci adıma geçilir. Algoritmanın yedinci adımında S vektöründeki değerlere karşılık gelen φ matrisinin kolonları ile orijinal sinyal x'in yaklaşık değerleri aşağıdaki en küçük kareler problemi Denklem 2.19'da verilmiştir [41, 61-63].

$$\tilde{x} = argmin_x \|\tilde{\varphi} x - \quad y\| \quad (2.19)$$

Tekrarların sınırlı olması DEA algoritmasının karmaşıklığını diğer yöntemlere göre önemli ölçüde azaltır. Özellikle sinyalin seyreklik seviyesinin düşük olduğu durumlarda algoritma hem daha az adımda biter hem de karmaşıklık göreceli olarak azalır [61]. OMP algoritmasının diğer algoritmalara göre basit ve hızlı olduğundan bahsedilmiştir [62]. Ayrıca bugüne kadar yapılan denemeler sonucunda OMP algoritmasının en iyi performansa sahip olduğu kanıtlanmıştır [63].

MGS yöntemi ile algoritmanın her tekrarında yeni bir eleman seçilir. Rezidü vektörü daha önce seçilmiş kolonların y vektöründen çıkarılmış halidir. DEA algoritmasının başarısı ortogonalliğidir. Kullanılan ortogonallik ile Rezidü vektörü her zaman daha önce seçilmiş olan elemanlara ortogonaldir ve bu eleman bu sayede bir daha seçilemez. Ortogonallik karmaşıklığı azalttığı gibi, performansı da arttırır. Ortogonallik her iterasyonda yeni eleman seçilmesini sağlar. DEA programında hızlı fourier dönüşümü (HFD), ayrık kosinüs dönüşümü (AKD), gerçek sinusoidal dönüşümü (GSD), haar dalgacık dönüşümü, hadamard dönüşümü, ayrık sinus dönüşümleri(ASD) kullanılmıştır.



Şekil 2.15 Dik Eşleştirme Arayış Algoritması (Orthogonal Matching Pursuit Algorithm)

Ayrıca orijinallik olması bakımından ayrık kosinüs-III dönüşümü (AK3D) [64] de kullanılmıştır. AK3D, AKD ye göre daha az çarpma işlemi gerektirdiği için daha hızlı sonuç vermektedir.

2.6 Ayrık Kosinüs-III Dönüşümü (AK3D)

Sıkıştırıcı algılamada kullanılan AKD dönüşümü yerine orjinallik olması ve daha iyi sonuç alınması sebebiyle AK3D de kullanılmıştır. AK3D, AKD ye göre daha az çarpma işlemi gerektirdiği için daha hızlı sonuç vermektedir [64]. AK3D'nin matematiksel açıklaması, işlem farklılığı ve AKD ile AK3D 'nin bağıntısı aşağıda verilmiştir.

$x(n)(n = 0,1, \dots, N-1)$ (2.20)

$x(n) = y(n)$ (2.21)

$y(n)$ katsayılar, *N* örnek sayısı, $X(n)$ AKD dönüşümü ve $X_3(n)$ AK3D dönüşümü olmak üzere,

$$u(k) = \begin{cases} \frac{1}{2} & k = 0 \\ 0 & diğer\ durumlarda \end{cases} \quad (2.22)$$

$$v(k) = \begin{cases} \frac{1}{\sqrt{2}} & k = 0, N \\ 1 & diğer\ durumlarda \end{cases} \quad (2.23)$$

AKD:

$$X(n) = v(n)\sqrt{2/N} \sum_{k=0}^{N-1} x(k) \cos \frac{\pi(k+1/2)n}{N} \quad (2.24)$$

$$n = 0,1, \dots \dots N-1 \quad (2.25)$$

AK3D:

$$X_3(n) = v(n)\sqrt{2/N}\ \sum_{k=0}^{N} v(k)z(k) \cos \frac{\pi nk}{N} \quad (2.26)$$

$$n = 0,1, \dots \dots N-1 \quad (2.27)$$

$$z(n) = v(n)\sqrt{2/N}\ \sum_{k=0}^{N-1} v(k)X(k) \cos \frac{\pi nk}{N} \quad (2.28)$$

$$n = 0,1, \dots \dots N-1 \quad (2.29)$$

$$z(0) = \frac{1}{\sqrt{2}} x(0) \quad (2.30)$$

$$z(N) = \frac{1}{\sqrt{2}}x(N-1) \tag{2.31}$$

$$z(n) = \frac{1}{2}[x(n) + x(n-1)] \tag{2.32}$$

$$X_3(n) = v(n)\sqrt{2/N}\left\{\frac{x(0)}{2}\left[1 + cos\frac{\pi n}{N}\right].\frac{1}{2}\sum_{k=1}^{N-2}x(k)\left[\cos\frac{\pi nk}{N} + \cos\frac{\pi(k+1)n}{N}\right] + \right.$$
$$\left.\frac{x(N-1)}{2}\left[(-1)^n + cos\frac{\pi(N-1)n}{N}\right]\right\} \tag{2.33}$$

$$= v(n)\sqrt{2/N}\ \ \frac{1}{2}\sum_{k=0}^{N-1}x(k).\left[\cos\frac{\pi nk}{N} + \cos\frac{\pi(k+1)n}{N}\right] \tag{2.34}$$

$$= \cos\frac{\pi nk}{N}\left[cos\frac{\pi n}{N} + 1\right] - sin\frac{\pi nk}{N}\sin\frac{\pi n}{N} = 2\cos\frac{\pi n}{2N}cos\frac{\pi\left(k+\frac{1}{2}\right)n}{N} \tag{2.35}$$

$$X(n) = v(n)\sqrt{2/N}\ cos\frac{\pi n}{2N}\sum_{k=0}^{N-1}x(k)\ cos\frac{\pi(k+1)n}{N} \tag{2.36}$$

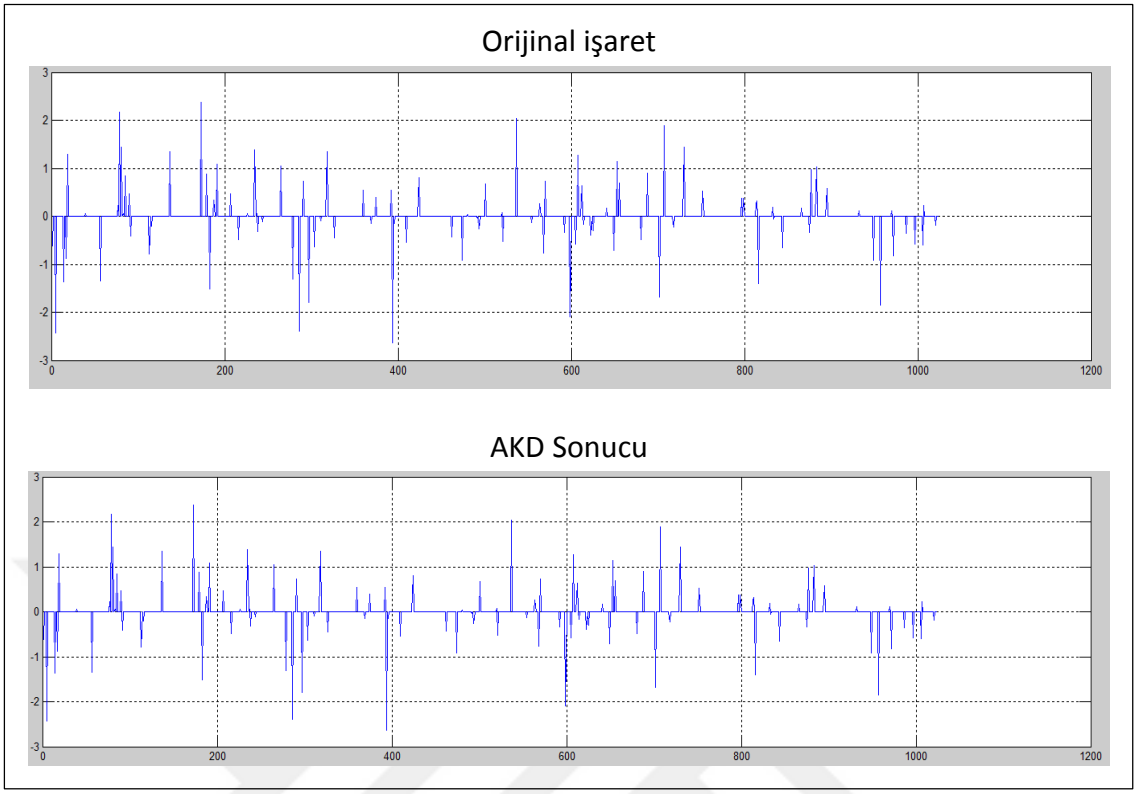
$$X_3(n) = cos\frac{\pi n}{2N}\ X(n) \qquad n = 0,1,2,\dots N-1 \tag{2.37}$$

Çizelge 2.1 AKD ve AK3D işlem karşılaştırması [64]

	Çarpma sayıları		Toplama sayıları	
N	AKD	AK3D	AKD	AK3D
8	13	7	29	34
16	33	19	81	89
32	81	51	209	222
64	193	131	513	535
128	449	323	1217	1256
256	1025	771	2817	2889
512	2305	1795	6401	6538
1024	5121	4099	14337	14603

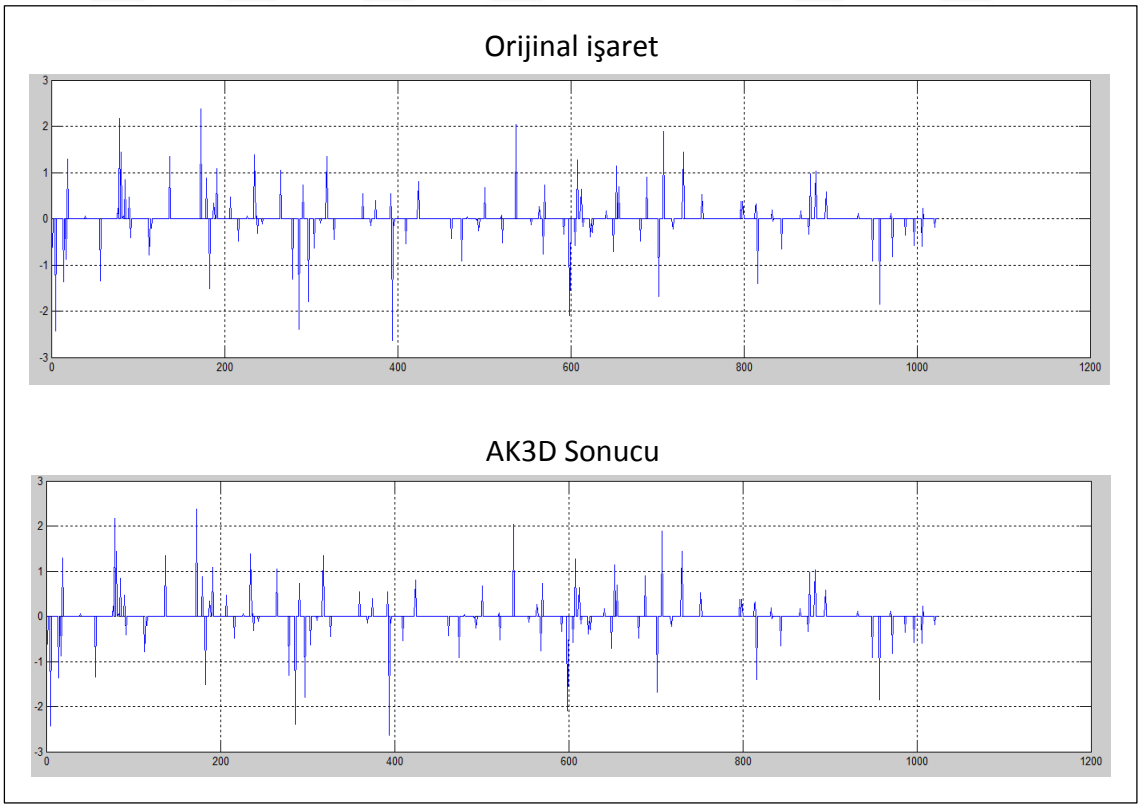
Çizelge 2.1'de AK3D'nin AKD'ye göre daha az çarpma işlemi içerdiğini göstermektedir. Bu da AK3D'nin AKD'ye göre daha hızlı olacağı anlaşılmaktadır. Bir sonraki sayfada Sıkıştırıcı Algılamada dönüşümlerde, AKD ve AK3D kullanıldığında aralarındaki fark gösterilmiştir. Şekillerde de görüleceği üzere, AK3D kullanıldığında, hem süre hem de doğruluk olarak AKD'ye göre daha iyi sonuç vermiştir. Burada yatay eksen sinyal uzunluğunu, dikey eksen ise işaretin değerini göstermektedir.

AKD ile işlem süresi: 0.915616 saniye. Hata: 2.24e-15



Şekil 2.16 AKD ile OMP sonucu

AK3D ile işlem süresi: 0.265875 saniye. Hata: 1.02e-15



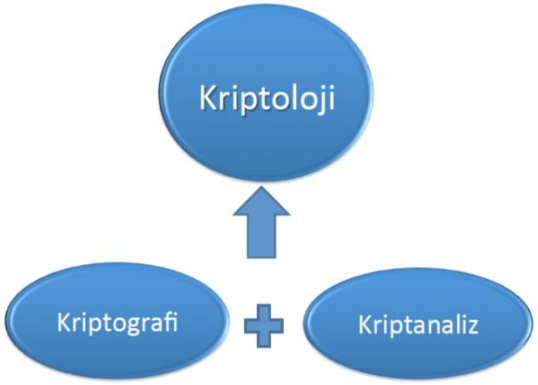
Şekil 2.17 AK3D ile OMP sonucu

BÖLÜM 3

KRİPTOGRAFİ

Günümüzde bilgi güvenliği internetin de gelişimi ile birlikte bilginin güvenliğinin sağlanması gerekliliği açısından önemli hale gelmiştir. Kriptografi, verinin güvenli bir şekilde iletilmesi ile ilgilidir. Dolayısıyla güvenli şifreleme algoritması tasarımı kriptografide çok önemli bir yer tutar. Kriptoloji, haberleşen iki veya daha fazla tarafın bilgi alışverişini emniyetli olarak yapmasını sağlayan, temeli matematiksel zor problemlere dayanan tekniklerin ve uygulamaların bütünüdür. Kriptografi, bilginin şifrelenmesi ve şifrenin çözülmesi için kullanılan yöntemlere verilen addır. Kriptoanaliz, kriptografi sistemleri tarafından ortaya konan bir şifreleme sistemini inceleyerek zayıf ve kuvvetli yönlerini ortaya koymayı amaçlayan bilim dalıdır [15, 16, 65, 66].

Crypto- veya Calypto- Latincede gizli, bilinmeyen anlamındadır. 'Cryptology' ise, bilinmeyene dair çalışma, gizlilik çalışması demektir. Dilimiz karşılığıyla 'Kriptoloji' kavramı, Şekil 3.1'de gösterildiği üzere, bünyesinde kriptografi, yani şifreleme bilimi ve kriptosistemlerin kırılması ile ilgilenen kriptanaliz çalışma alanlarını barındırmaktadır. Bununla beraber pek çok zaman kriptografi ve kriptoloji birbiri yerine geçecek şekilde kullanılmaktadır.



Şekil 3.1 Kriptoloji genel yapısı

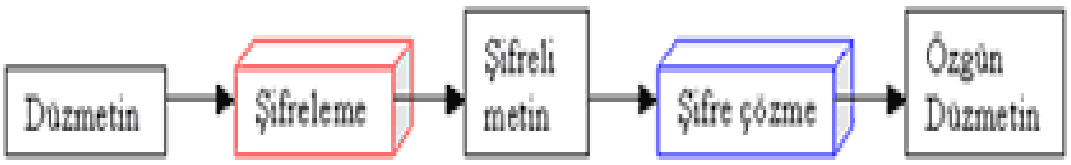
Kriptografinin tarihte görüldüğü ilk belirgin örnek olarak Julius Caesar'ın MÖ 60-50 yıllarında Roma alfabesindeki harflerin yerini değiştirerek oluşturduğu şifreleme yöntemini devlet haberleşmesinde kullanması örnek gösterilir. Bu yöntem şifrelenecek olan metindeki her harfin alfabede kendisinden 3 sonraki harfle değiştirilmesine dayanmaktadır. 730'lu yıllarda Abu Abd al-Rahman kriptografi üzerine bir kitap yazmıştır. Kitaptaki yöntemler 2. dünya savaşında kullanılan Enigma makinesi için ilham kaynağı olmuştur. Günümüzdeki modern şifre yöntemlerinin gelişmesi ise 1970'lerin ortasında DES (Data Encryption Standard) algoritmasının IBM ve NSA tarafından ortaya atılması ile başlamıştır.

3.1 Şifrelemenin Temel Elemanları

Bir gönderici bir alıcıya açık ağlar üzerinden bir ileti göndermek istediği zaman, açık ağlardan gönderilen iletiler üçüncü şahıslar tarafından dinlenme ve değiştirilme tehdidi altındadırlar. Burada söz konusu ileti düz metindir. Bazı kullanımlarda plaintext adı da verilir. Bir iletinin içeriğini saklamak üzere yapılan gizleme işlemi de şifrelemedir (encryption). Bu işlem düz metni şifreli metine dönüştürür. Bilginin içeriği başkalarının anlamayacağı hale gelir. Bu bilgi bir yere iletilmek amacıyla şifrelenen bir mesaj veya saklanmak amacıyla şifrelenen bir bilgi olabilir. Şifrelenmiş bir ileti şifreli metindir (ciphertext). Şifreli metini düz metine geri çevirme işlemi şifre çözümüdür (decrypt).

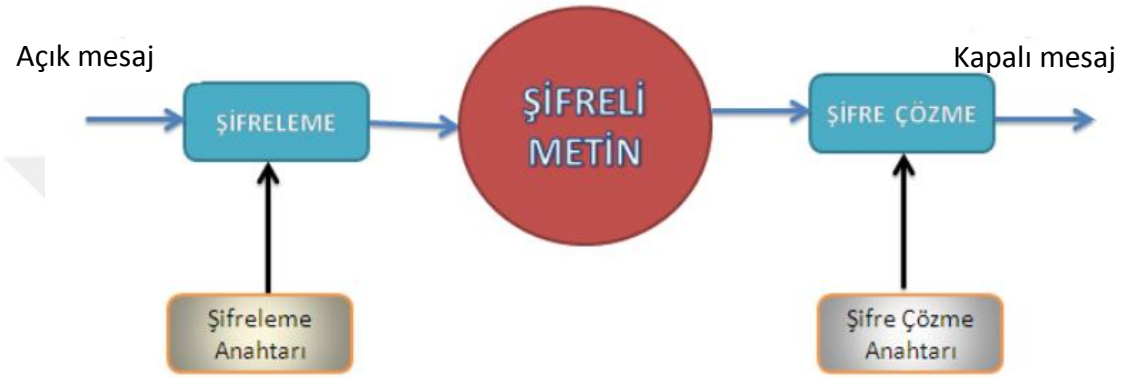
Şekil 3.2'de ve Şekil 3.3'de belirtilen bir kriptosistemin 5 elemanı vardır.

1. P=Plaintext: Şifrelenecek olan düz metin
2. K=Key(Anahtar): Şifrelemede kullanılacak olan anahtar
3. E=Encryption: Şifreleme
4. C=Ciphertext: Düz metinin şifrelenmiş hali
5. D=Decryption: Şifre çözümü



Şekil 3.2 Şifreleme ve şifreyi çözme işlemleri

Kriptografi matematiksel olarak, E_k dönüşümler ailesinden seçilen birebir ve örten bir fonksiyon olmak üzere şifreleme işlemi, anahtar ailesinin bir elemanı ile m mesajından $E_k(m) = c$ olacak şekilde bir c şifreli mesajını elde etme işlemi iken, E'nin tersi olan D fonksiyonu ile $D_k(c) = m$ olacak şekilde c şifreli metninden m mesajına ulaşmaya deşifreleme denir.



Şekil 3.3 Geleneksel Şifrelemenin Basit Modeli

Kriptoloji ne sağlar:

- Gizlilik (Confidentiality):Başkaları mesajı göremez.
- Mesaj Bütünlüğü (Integrity): Başkaları mesajı değiştiremez.
- Gönderici Kimliğini Doğrulama (Authentication): Gönderen ve alıcı, birbirlerinin kimliklerini doğrulayabilirler.
- İnkâr Edememezlik (Non-repudiation):Bilgiyi oluşturan ya da gönderen, daha sonra bilgiyi kendisinin oluşturduğunu veya gönderdiğini inkar edememeli.

Bu çalışmada kullanılacak kriptografi ile ilgili terimler aşağıda açıklanmıştır.

Düz (Açık) Metin (Plaintext): Metinlerin şifrelenmeden önceki orijinal halidir.

Şifreli Metin (Ciphertext): Orijinal metnin şifrelenmiş halidir.

Şifreleme (Encryption): Bir iletinin içeriğini saklamak için yapılan gizleme işlemidir.

Şifre Çözme (Decryption): Şifrelenmiş veriyi çözüp eski haline getirme işlemidir.

Şifre Kırma (Crack): Başkasına ait şifreyi öğrenmek için yapılan işlemdir.

Saldırgan (Adversary): Mesajın alıcısı veya göndericisi olmadığı halde mesaja ulaşmaya çalışan kişidir.

Saldırı (Attack): Şifreli sistemin şifresini kırmak için yapılan teşebbüstür.

Anahtar (Key): Şifrelemede veya şifre çözmede kullanılan algoritmanın bilinmeyen kısmını oluşturan bölümüdür.

Şifreleme Anahtarı (Encryptionkey): Şifrelemek için kullanılan anahtardır.

Şifre Çözme Anahtarı (Decryptionkey): Şifre çözmek için kullanılan anahtardır.

Açık Anahtar (PublicKey): Asimetrik Şifreleme sisteminde kullanılan ve istenildiğinde erişilebilen anahtardır.

Gizli (Özel) Anahtar (PrivateKey): Asimetrik Şifreleme sisteminde kullanılan ve şifre çözmede kullanılan ve sadece sahibi tarafından bilinen anahtardır.

Anahtar Üretimi (KeyGeneration):Asimetrik şifrelemede, açık ve gizli anahtarı elde etmek için yapılan işlemdir.

Şifreleme Algoritması: Şifreleme ve deşifreleme işlemlerinde kullanılan akış diyagramı ve matematiksel işlemlerin bütünüdür.

Gönderici (Sender): Bilgiyi gönderen kişidir.

Alıcı (Receiver): Bilgiyi alan kişidir.

Kanal (Channel): Bilgi iletiminin yapıldığı internet, telefon gibi iletişim araçlarıdır.

Kriptografi (Cryptography): Şifreleme bilimidir.

Kriptanaliz (Cryptanalysis): Şifreleme sisteminin açıklarını bularak anahtar olmaksızın orijinal verilere ulaşmayı amaçlayan analizdir.

Ayrıntılı anahtar arama yöntemi: Tüm alternatiflerin tek tek denenmesiyle anahtara ulaşmak için yapılan kriptanaliz yöntemidir.

Kriptanalist (Cryptanalist): Kriptanaliz yapan kişidir.

Kriptoloji (Cryptology): Kriptografi ve kriptanalizin bütününü içeren bilimdir. [15-16], [65-69].

Şifreleme algoritmaları anahtar kullanma yöntemlerine göre genel olarak üç kategoriye ayrılmaktadır. Bu yöntemler:

- Gizli-Anahtar (Simetrik) yöntemi

- Açık-Anahtar (Asimetrik) yöntemi

- Hibrit yöntemi (simetrik+asimetrik beraber uygulaması)

3.1.1 Gizli-Anahtar (Simetrik) Yöntemi

Simetrik şifreleme algoritmaları, şifreleme ve açma işlemleri için aynı anahtarı kullanır. Şekil 3.4'de simetrik şifreleme yöntemi gösterilmiştir. Gizli veri alışverişi yapacak kişi veya uygulamalar simetrik anahtarı kendi aralarında, emniyetli bir şekilde değiştirmelidir. Simetrik şifreleme algoritmasıyla şifrelenmiş bir verinin güvenliği, şifreleme işleminde kullanılmış olan anahtarın gizliliği ile doğrudan ilişkilidir. Bu nedenle, bu tip sistemlere gizli anahtarlı kriptografi sistemi adı da verilmektedir.



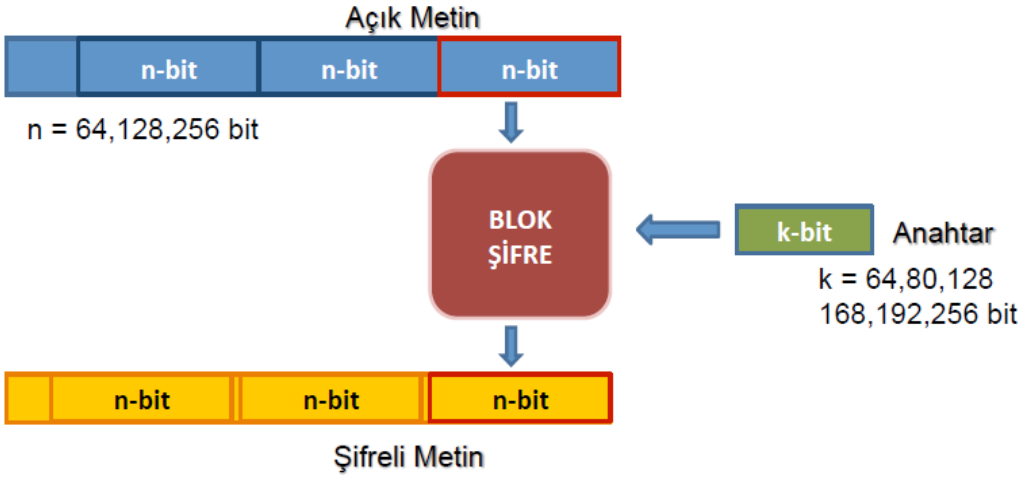
Şekil 3.4 Simetrik şifreleme

Simetrik şifrelemenin en önemli avantajlarından birisi oldukça hızlı olmasıdır. Asimetrik şifrelemeyle karşılaştırıldığında hız konusunda simetrik algoritmalar çok daha başarılıdır. Bununla birlikte simetrik algoritmayı içerdiği basit işlemlerden dolayı elektronik cihazlarda uygulamak çok daha kolaydır. Ayrıca simetrik algoritmalarda kullanılan anahtarın boyu ve dolayısıyla bit sayısı çok daha küçüktür.

Önemli gelişmeler çoğunlukla ABD'de NIST (National Institute of Standarts and Technology) öncülüğünde gerçekleşmiştir. 1975'de DES (Data Encryption Standard), ABD tarafından FIPS 46 (Federal Information Processing Standard) olarak kayda geçmiştir. 2001'de NIST'in blok şifre yarışmasını kazanan Belçikalı Daemen ve Rijmen'e

ait Rijndael algoritması, AES (Advanced Encryption Standard) adıyla standart haline getirilmiştir. Bugün Rijmen'in bağlı bulunduğu Leuven'deki kriptografi grubu, Avrupa'daki simetrik (gizli anahtarlı) algoritmalarla ilgili en önemli araştırma merkezlerinden birisi rolündedir. Öte yandan, NASA'nın DES'i ve hatta AES'i kıracak süper bilgisayar teknolojisine sahip olduğu yönünde ve hatta bu algoritmaların sadece kendilerinin bildiği açıklar içerdiği yönünde sansasyonlar mevcuttur. Bir zamanlar DES için özelleştirilmiş ve bilhassa DES'in kullandığı operasyonları hızlı çalıştırmak üzere tasarlanmış işlemcilerin yerini bugün AES kullanarak şifreleme veya deşifreleme yapabilen, bilinen tüm temel blok şifre modlarında çalışabilen ve AES standartlarında belirtilen 128-bit, 192-bit veya 256-bit anahtarları destekleyen özelleştirilmiş işlemciler almıştır.

Şekil 3.5'de belirtilen açık metin eşit uzunluktaki bloklara ayrılıp, her bir bloğu bir fonksiyon yardımı ile tek tek şifreleyerek şifreli metin oluşturulur.



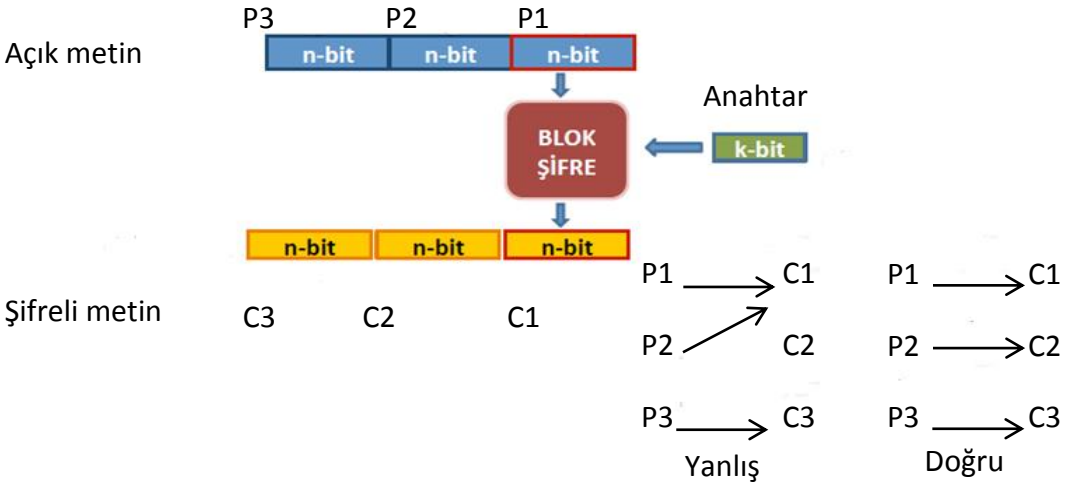
Şekil 3.5 Blok şifreleme

Matematiksel tanımı;

$$E: \{0,1\}^k \text{ x } \{0,1\}^n \rightarrow \{0,1\}^n \tag{3.1}$$

$$E(k, P) = E_k P = C. \tag{3.2}$$

Girdi olarak; k-bitlik bir anahtar ve n-bitlik açık metin bloğu yer alır, çıktı olarak n-bitlik şifreli metin bloğu oluşturulur. E fonksiyonu $\{0,1\}^n$ kümesinde bir permütasyon tanımlar. Aynı anahtar kullanılarak her girdi için farklı bir çıktı oluşmaktadır. [15-16], [65-70].



Şekil 3.6 Blok şifrelemede doğru/yanlış gösterimi

$E: \{0,1\}^k$ x $\{0,1\}^n \rightarrow \{0,1\}^n$ (3.3)

$k = 2$ bit ve $n = 3$ bit denilirse;

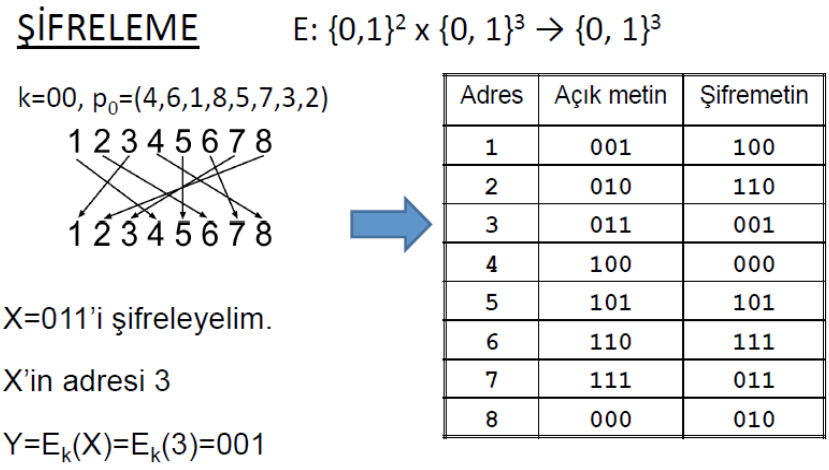
$(2n)! = 8! = 40320$ permütasyondan, sadece $2k = 4$ tanesi kullanılıyor, bu şifre dönüşümleri aşağıdaki gibi seçilirse;

$k = 00, p_0(4,6,1,8,5,7,3,2)$

$k = 01, p_1(5,2,7,1,8,6,4,3)$

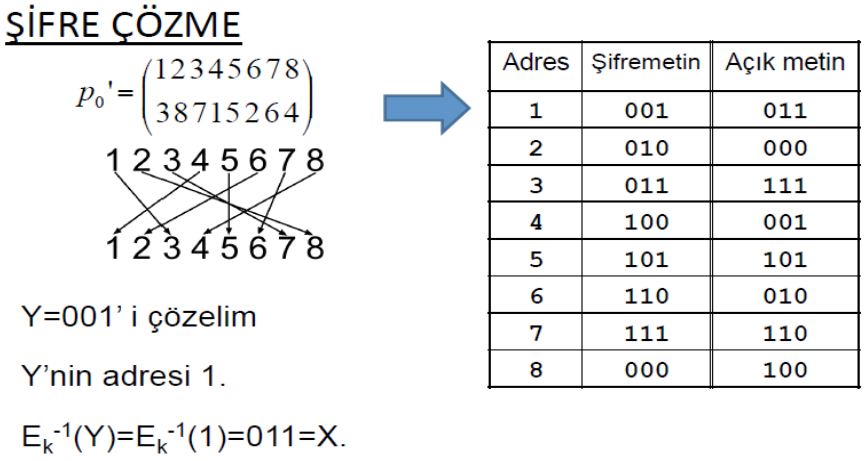
$k = 10, p_2(8,6,2,1,3,4,5,7)$

$k = 11, p_3(3,8,6,2,4,7,5,1)$



Adres	Açık metin	Şifremetin
1	001	100
2	010	110
3	011	001
4	100	000
5	101	101
6	110	111
7	111	011
8	000	010

Şekil 3.7 Blok şifrelemede şifrelemenin gösterimi

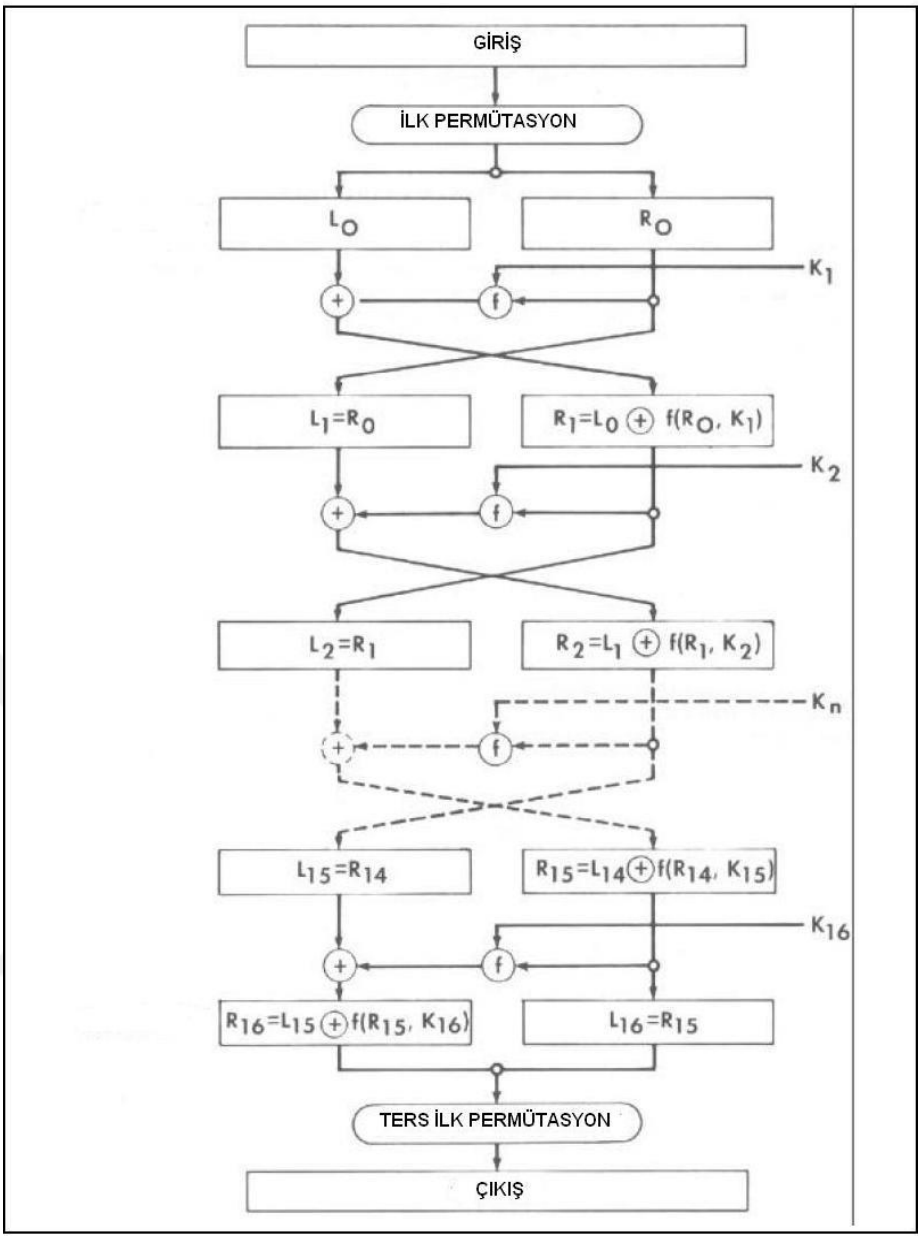


Adres	Şifremetin	Açık metin
1	001	011
2	010	000
3	011	111
4	100	001
5	101	101
6	110	010
7	111	110
8	000	100

Şekil 3.8 Blok şifrelemede şifre çözmenin gösterimi

3.1.1.1 Veri Şifreleme Standardı (Data Encryption Standard-DES)

Basitçe şifrelenecek olan açık metni parçalara bölerek (blok) her parçayı birbirinden bağımsız olarak şifreler ve şifrelenmiş metni açmak için de aynı işlemi bloklar üzerinde yapar. Bu blokların uzunluğu 64 bittir. DES, IBM tarafından geliştirilmiştir. 1975 yılında "Federal Register" tarafından yayınlanmıştır. DES 64 bitlik veriyi 56 bitlik anahtar kullanarak şifreler. DES algoritması için kullanılan 64 bitlik anahtarın 56 biti algoritma içinde işlemden geçer. Kalan 8 bit benzerlik veya hata bulmak için kullanılabilir. DES şifreleme için kullanılan teknikler yayılma ve karıştırmadır. DES algoritmasında, şifrelenecek metin öncelikle IP (Initial Permutation – ilk permütasyon)' dan geçer. Daha sonra DES işleminden geçer ve ters IP sonucunda şifreli metin elde edilir. DES algoritmasının yapısı aşağıdaki gibidir [15, 16, 65, 70]. Şekil 3.9'da DES yapısı belirtilmiştir. Bu şekilde girişte bulunan 64 bitlik data, ilk permütasyondan sonra ilk adımda L_0 ve R_0 olarak sol ve sağ şeklinde 32 bit şeklinde ikiye ayrılır. Daha sonra ise anahtar (K_1) ile şifreleme yapılır ve bu işlem 16 defa tekrarlanır.



Şekil 3.9 DES yapısı [65]

DES yapısında, IP' de permütasyon işlemi yapılmaktadır. Permütasyon işleminde, bitlerin yerleri değiştirilerek şifrelemenin ilk adımı gerçekleştirilir. Bu işlem için bir tablo kullanılmaktadır. Bu tablo çizelge 3.1'de gösterilmiştir. Bu tabloda, IP işleminden sonra girişin 58. biti ilk bit, 50. biti ise 2. bit olur ve bu şekilde devam eder. Permütasyondan geçen giriş bloğu daha sonra karmaşık anahtar bağımlı DES kutusuna girer. (DES işlemleri 16 döngüdür.) DES kutusu çıkışı ise ters ilk permütasyona (IP^{-1}) girer. Ters ilk permütasyonda da tablo kullanılmaktadır.

Çizelge 3.1 DES için ilk permütasyon örneği [65]

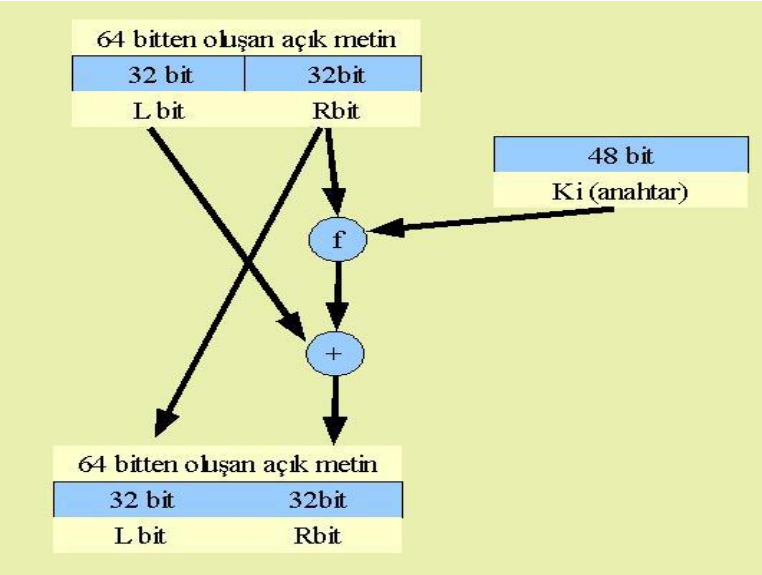
IP

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

DES'in en büyük dezavantajı anahtar uzunluğunun 56 bit olmasıdır. 1975 yılında yayınlanan bu algoritma günümüzde geliştirilen modern bilgisayarlar tarafından yapılan saldırılar (Brute Force) karşısında yetersiz kalmaktadır. 3DES, IBM tarafından geliştirilip 1977'de standart olarak kabul edilmiştir. DES algoritmasının kaba kuvvet ataklarına karşı dayanıksız olması sebebiyle geliştirilmiştir. Fakat 1997 yılında İsrail'liler tarafından kırılmıştır. Şifreleme metodunun çözülmüş olmasına rağmen günümüz bankacılık sistemlerinde kullanılmakta olan şifreleme sistemidir. Çift yönlü çalışır. Şifrelenmiş veri geri çözülebilir. Bilgisayarın donanımsal açıklarını kapatma özelliği vardır fakat algoritmanın güvenliği tamamen anahtarın gücüne bağlıdır. 3DES algoritması, DES algoritmasının şifreleme, deşifreleme, şifreleme şeklinde uygulanmasıdır. DES şifreleme yöntemine göre 3 kat daha yavaş çalışır. Şifreleme yapmak için uzunluğu 24 bayt olan bir anahtar kullanılır. Her bayt için 1 eşlik biti vardır. Dolayısıyla anahtarın uzunluğu 168 bittir. Veri, 3DES anahtarının ilk 8 baytı ile şifrelenir. Sonra veri anahtarın ortadaki 8 baytı ile çözülür. Son olarak anahtarın son 8 baytı ile şifrelenerek 8 bayt uzunluğunda blok elde edilir. DES algoritmasında şifrelenecek metin bloklar halinde şifreleme işleminden geçirilir. İleride tekrardan üstünden geçilecektir.

IP(Initial Permutation) şifreleme işleminde kullanılacak ilk permütasyondur. Verilen bir düz metin (x) üzerine ilk permütasyon uygulanarak x_0 elde edilir. $x_0 = IP(x) = L_0R_0$, $L_0 = x$'in ilk 32 biti, $R_0 = x$'in son 32 biti, ardından 16 defa tekrarlanan bir fonksiyonla yeni değerler hesaplanır. $L_iR_i(1 <= i <= 16)$ aşağıdaki kurala göre hesaplanır. En son olarak da $R_{16}L_{16}$'ya IP^{-1} uygulanarak şifreli metin elde edilir. Sonuç olarak, DES askeri amaçlı olmayan çalışmalarının ilki ve şifreleme algoritmalarının yeni bir boyut kazanmaya başladığı noktadır. DES'in algoritmasından

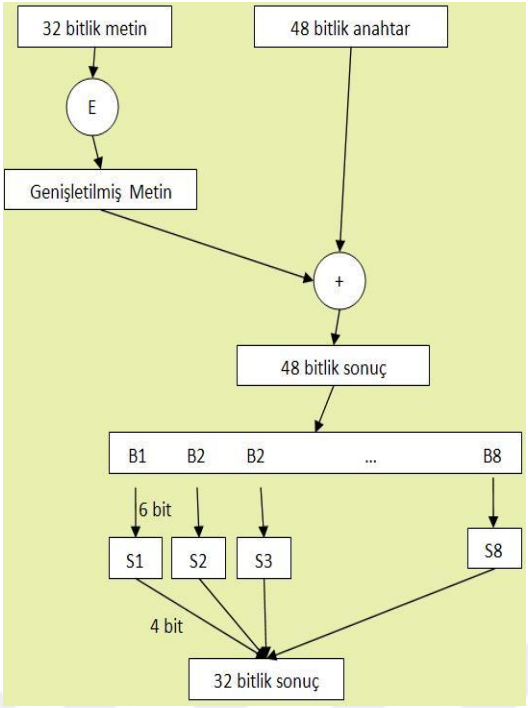
kaynaklanan bu sorunlar 'Triple DES' ya da '3DES' olarak bilinen yeni bir algoritma ile düzeltilmiştir. Günümüzde kullanılan çoğu uygulama 3DES'i kullanmaktadır.



Şekil 3.10 DES yapısı [65]

Sadece 1 adımı gösteren DES yapısı Şekil 3.10'da gösterilmektedir. Bu şekilde 64 bitten oluşan bir metnin öncelikle 2 parçaya ayrılması gerekir, soldaki ve sağdaki parça ayrı ayrı işlenmektedir. Bu işleme sağdaki 32 bitlik parçanın f fonksiyonuna anahtar ile birlikte girmesi ve parçaların yer değiştirmesi şeklinde yapılmaktadır. Yukarıdaki şekilde (+) işlemi yahut işlemidir (xor (exclusive or, özel veya)). Şekilde DES için yapılan bir geçiş çizilmiştir, DES şifrelemesinde bu işlem 16 defa tekrar edilmektedir.

f fonksiyonun detayı Şekil 3.11'de verilmiştir. f fonksiyonu bir 32bit'lik parçayı alarak 48 bit'lik anahtar ile 32 bit'lik sonuç üretmektedir. Bu üretme işlemi sırasında en kritik işlem aşağıdaki şekilde E olarak görüntülenmiş expansion işlemidir. Bu işlem basitçe aynı bit için birden fazla sonuç üretilebilmesini öngörür. Genişletilmiş (expanded) bitleri, anahtar ile xor işlemine tabî tutulur. Sonuçta 48 bitlik olarak üretilen metin 8 bloğa bölünür. Her blok 6 bitlik bir parçadan oluşmaktadır. Her 6 bitlik parça bu sefer expansion işleminin tersi olarak küçültülmekte ve 4 bit'e indirilmektedir. Sonuçta 4 bitlik 8 blok yani toplam 32 bitlik veri üretilmiştir.



Şekil 3.11 f fonksiyonu [65]

Önceki işlemde genişletme işlemi olarak adlandırılan işlem (expansion) basitçe çizelge 3.2'ye benzer bir tablo ile yapılmaktadır:

Çizelge 3.2 Genişletme tablosu

32	1	3	1	2	4	3	6
4	5	5	7	28	30	7	9
9	7	10	8	22	11	11	15
29	12	14	17	15	16	12	18
13	23	19	21	15	17	14	18
31	20	21	20	24	25	26	27

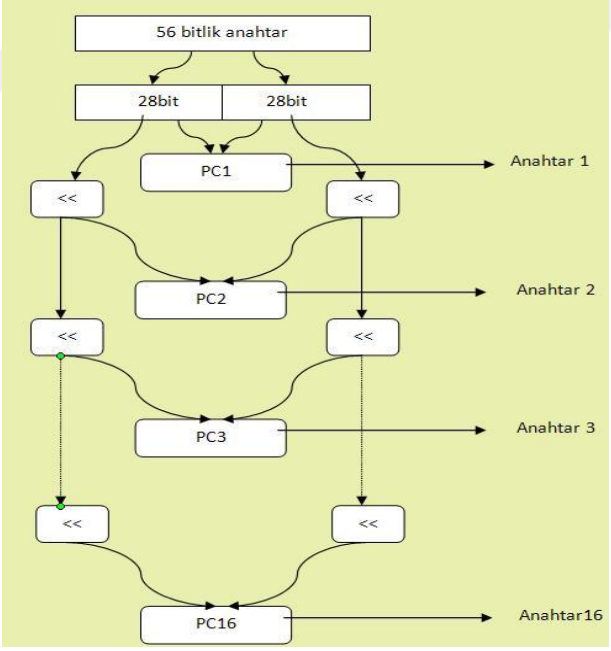
Yukarıdaki tablodaki mantık basitçe bir sayının birden fazla pozisyonda bulunmasıdır. Bu durumda örneğin 1 sayısı için hem 2. hem de 4. konumda karşılık bulunmaktadır. DES ile ilgili diğer bir ayrıntı da anahtar üretilmesidir. Dikkat edilirse DES'in 64 bitlik bir anahtar ile çalıştığı ve 8 bitlik parity kontrolü çıkarılacak olursa bu anahtarın 56 bitlik olduğu yukarıda anlatılmıştır. Ancak f fonksiyonuna giren her geçişteki anahtar boyutu

yukarıda 48 bit olarak verilmiştir. Dolayısıyla aslında her geçiş (pass) için farklı bir anahtar üretilmektedir. Bu anahtarlar 56 bitlik esas anahtardan üretilen anahtarlardır.

Aşağıda bu anahtarların nasıl üretildiği gösterilmiştir:

Aşağıdaki Şekil 3.12'de 56 bitlik giriş anahtarından her geçiş için gereken anahtarların üretilmesi gösterilmiştir. Bu işlem 16 adımda yapılmaktadır ve her adımda o adım için üretilmiş olan tablo kullanılmaktadır. Her adım öncelikle 2 adet 28 bitlik parçaya bölünmüştür. Tablolara PC1'den PC16'ya kadar isimler verilmiştir. Ayrıca her tablo girişinden önce bir kaydırma işlemi kullanılarak üretilen anahtar değiştirilmektedir. Buna göre örneğin 10. adımda orjinal verinin her iki parçası da 10 bit kaydırılmış olacaktır.

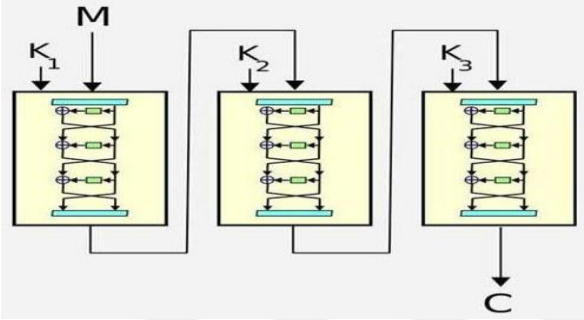
DES ile şifrelenmiş bir metni açmak için aynı algoritmaya şifreli metni (cipher text) aynı anahtar ile vermek yeterlidir. DES için zaman içinde bilgisayarların işlem hızının gelişmesi ile saldırılar kolay hale gelmiştir. DES'in daha zor saldırılır hale gelmesi için üçlü DES uygulaması geliştirilmiştir.



Şekil 3.12 DES anahtar üretimi [65]

Şekil 3.13'de belirtilen 3DES (Üçlü DES), 1977 yılında IBM tarafından geliştirilmiş olan bir şifreleme algoritmasıdır. Brute Force (kaba kuvvet) saldırılara karşı koymakta zorlanan DES (Data Encryption Standard - Veri Şifreleme Standardı) algoritmasının üzerine geliştirilmiştir. Çift yönlü çalışır. Şifrelenmiş veri geri çözülebilir. DES

şifrelemesinin 3 kere art arda yapılması şeklinde çalışır. DES şifreleme yöntemine göre 3 kat daha yavaş çalışır. Şifreleme yapmak için uzunluğu 24 bayt olan bir anahtar kullanılır. Her bayt için 1 eşlik biti vardır. Dolayısıyla anahtarın uzunluğu 168 bittir. Veri, 3DES anahtarının ilk 8 baytı ile şifrelenir. Sonra veri anahtarın ortadaki 8 baytı ile çözülür. Son olarak anahtarın son 8 baytı ile şifrelenerek 8 bayt bir blok elde edilir. [15, 16, 65, 67, 68, 70].



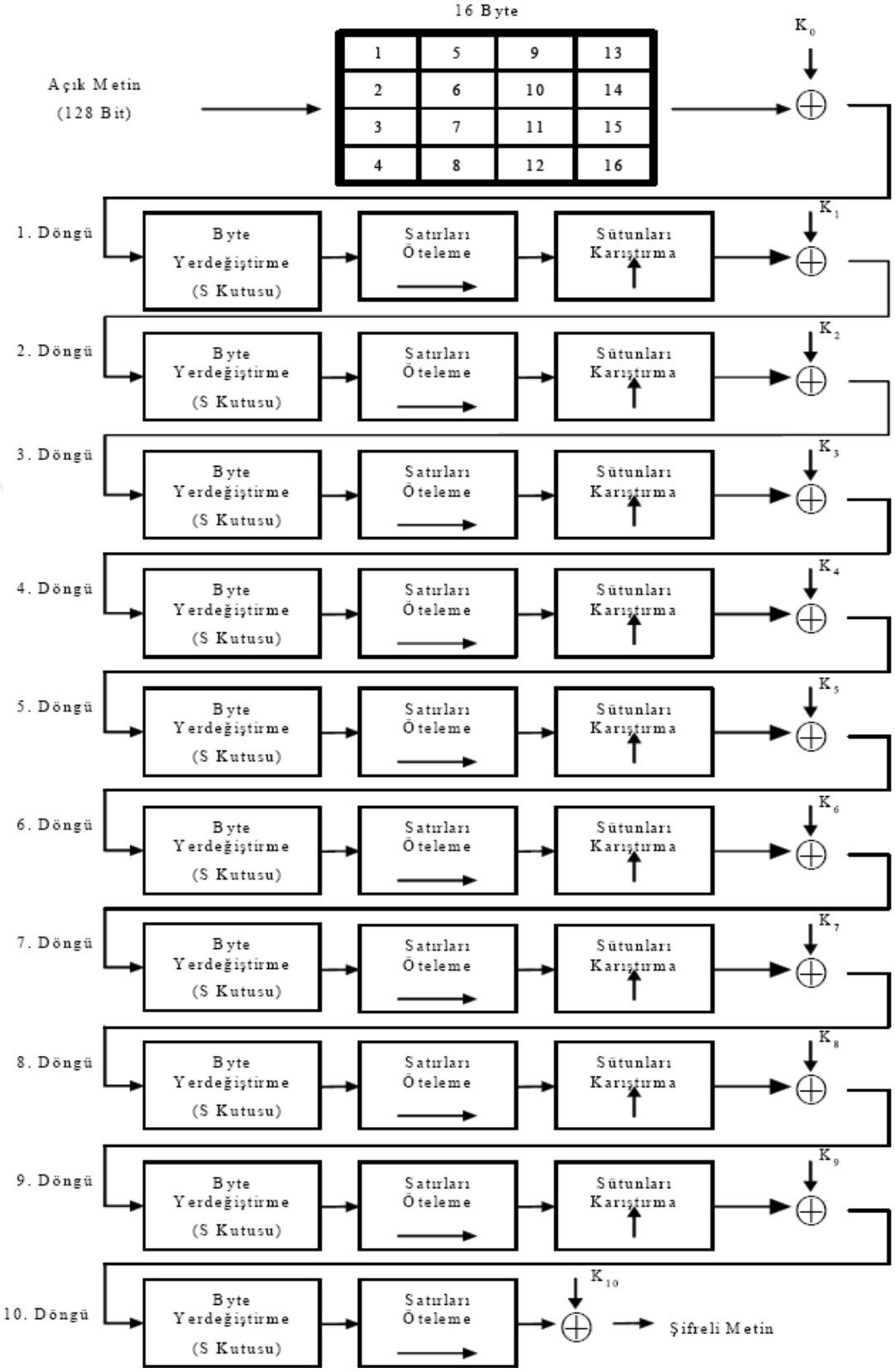
Şekil 3.13 3DES Yapısı

3.1.1.2 Gelişmiş Şifreleme Standardı (Advanced Encryption Standard-AES)

AES, John Daemen ve Vincent Rijmen tarafından Rijndael adıyla geliştirilmiş ve 2002 yılında standart haline gelmiştir. AES uzunluğu 128 bitte sabit olan blok ile uzunluğu 128, 192 ya da 256 bit olan anahtar kullanır. Şekil 3.14'de belirtilen AES yapısında kullanılan tekniklerden bazıları baytların yer değiştirmesi, 4x4' lük matrisler üzerine yayılmış metin parçalarının satırlarına uygulanan kaydırma işlemleridir. 2010 yılı itibariyle en popüler simetrik algoritmalardan biridir. Çizelge 3.3'de anahtar bit sayısına göre tur sayılarını göstermektedir.

Çizelge 3.3 AES tur sayısı

AES	**Tur Sayısı**
AES-128	10
AES-192	12
AES-256	14

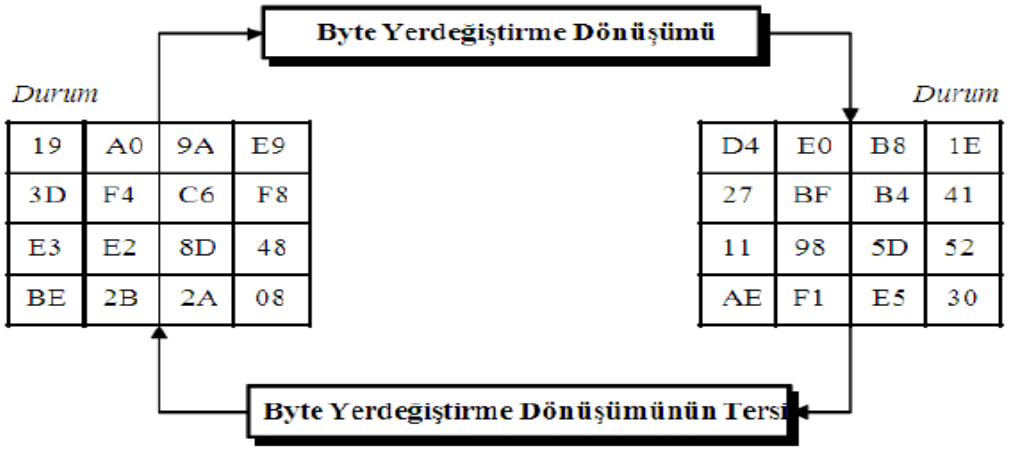


Şekil 3.14 AES Yapısı [65]

AES algoritmasında giriş, çıkış ve matrisler 128 bitliktir. Matris 4 satır, 4 sütun (4×4), 16 bölmeden oluşur. Bu matrise 'durum' denmektedir. Durumun her bölmesine bir baytlık veri düşer. Her satırda 32 bitlik bir kelimeyi meydana getirir. Anahtar uzunluğu bit sayıları arasındaki farklılık AES tur döngülerinin sayısını değiştirmektedir. Her döngü tersi alınabilir dönüşümler kullanır. Aynı zamanda her döngü, son döngü hariç, 4 dönüşüm kullanır: SubBytes (bayt değiştirme), ShiftRows (satır kaydırma), MixColumns (sütun karıştırma) ve AddRoundKey (tur anahtarı ile toplama). Son döngüde MixColumns dönüşümü göz ardı edilir. Her döngüde farklı anahtar materyali kullanılır. Farklı anahtar materyalleri anahtar planlama evresinde gelen anahtarlardır. Master anahtardan farklı anahtarlar elde edilerek şifrede kullanılır. Deşifreleme kısmında ise ters dönüşümler kullanılır.

Şifreli metnin oluşumu için girişten gelen metin 128 bitlik parçalara bölünür. Her parça durum matrisine yerleştirilir. Durum matrisi oluşturulduktan sonra, artık üzerinde tüm işlemler yapılabilir duruma gelmiş demektir. Aynı şekilde önceden alınan 128 bitlik anahtarda bu durum matrisi halinde işlem görür. Giriş metninin yazıldığı durum matrisi ilk olarak anahtar ile toplanır.

Şekil 3.15'de bayt yer değiştirme gösterilmiştir. İlk işlem algoritmanın tek doğrusal olmayan işlemi olan bayt değiştirmedir. Durum matrisinin her elemanı, değerleri önceden hesaplanarak oluşturulmuş Şekil 3.16'da belirtilen S-kutusundaki değerlerle değiştirilir.



Şekil 3.15 Bayt değiştirme

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	63	7c	77	7b	f2	6b	6f	C5	30	01	67	2b	fe	d7	ab	76
1	ca	82	c9	7d	fa	59	47	f0	ad	d4	a2	af	9c	a4	72	c0
2	b7	fd	93	26	36	3f	f7	cc	34	a5	e5	f1	71	d8	31	15
3	04	c7	23	c3	18	96	05	9a	07	12	80	e2	eb	27	b2	75
4	09	83	2c	1a	1b	6e	5a	a0	52	3b	d6	b3	29	e3	2f	84
5	53	d1	00	ed	20	fc	b1	5b	6a	cb	be	39	4a	4c	58	cf
6	d0	ef	aa	fb	43	4d	33	85	45	f9	02	7f	50	3c	9f	a8
7	51	a3	40	8f	92	9d	38	f5	bc	b6	da	21	10	ff	f3	d2
8	cd	0c	13	ec	5f	97	44	17	c4	a7	7e	3d	64	5d	19	73
9	60	81	4f	dc	22	2a	90	88	46	ee	b8	14	de	5e	0b	db
A	e0	32	3a	0a	49	06	24	5c	c2	d3	ac	62	91	95	e4	79
B	e7	c8	37	6d	8d	d5	4e	a9	6c	56	f4	ea	65	7a	ae	08
C	ba	78	25	2e	1c	a6	b4	c6	e8	dd	74	1f	4b	bd	8b	8a
D	70	3e	b5	66	48	03	f6	0e	61	35	57	b9	86	c1	1d	9e
E	e1	f8	98	11	69	d9	8e	94	9b	1e	87	e9	ce	55	28	df
F	8c	a1	89	0d	bf	e6	42	68	41	99	2d	0f	b0	54	bb	16

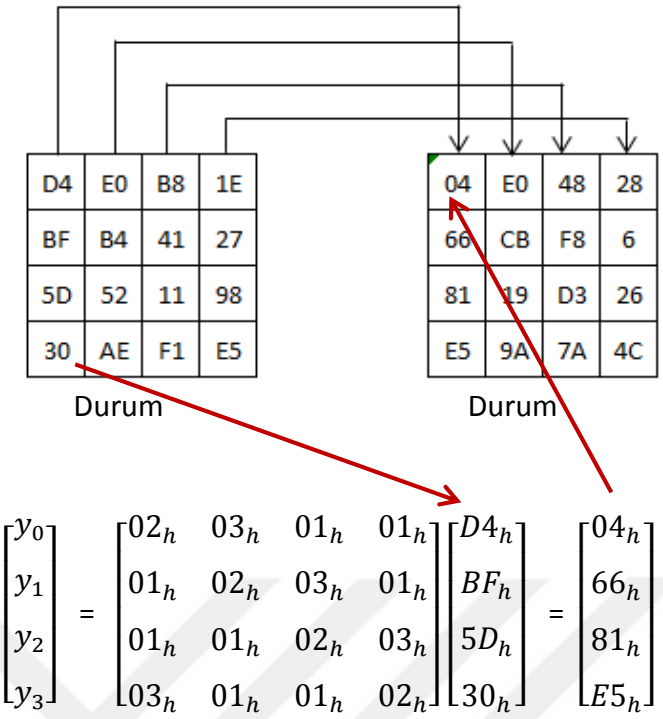
Şekil 3.16 Bayt değiştirme

Şekil 3.17'de gösterilen satır kaydırma işleminde satırlar sırasıyla çevrimsel şekilde kaydırılırlar. Yani ilk satır değiştirilmez, ikinci satır da sola 1 ötelenir, üçüncü satır sola 2 ötelenir ve son satır sola 3 ötelenir. Taşan bölmeler kaydırmanın başına eklenir.



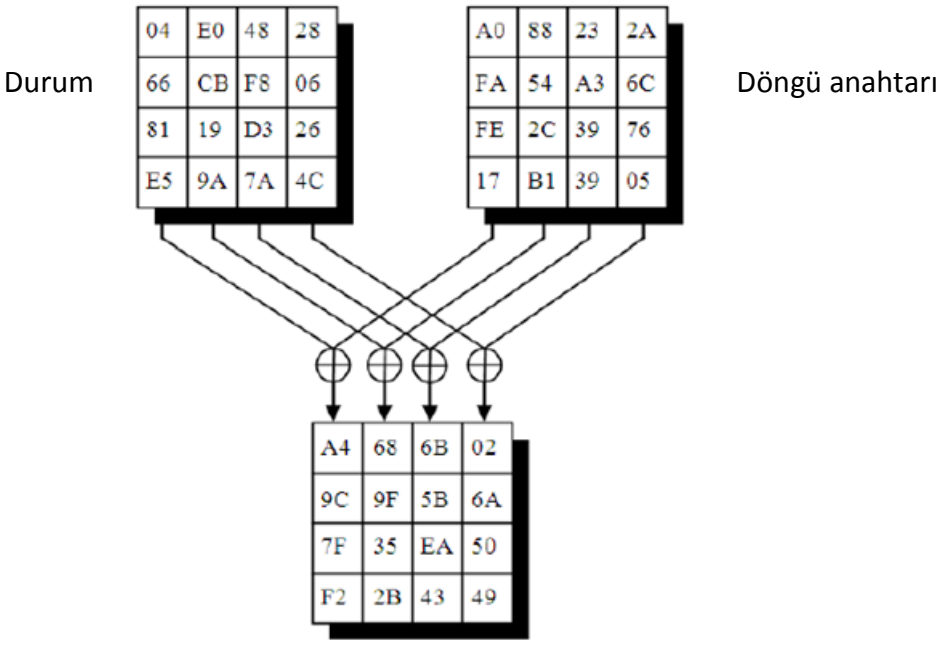
Şekil 3.17 Satır kaydırma

Şekil 3.18'de belirtilen sütun karıştırma işleminde eski sütunun elemanları kullanılarak yeni sütun elde edilmektedir. Bu yapılırken yeni sütunun elemanları eski sütunun her elemanı hesaba katılarak tek tek hesaplanır. Yapılan hesap çarpma ve toplama işleminden oluşur. Çarpma işleminde belirli bir sabit sayı $(a(x))$ kullanılır.



Şekil 3.18 Sütun karıştırma

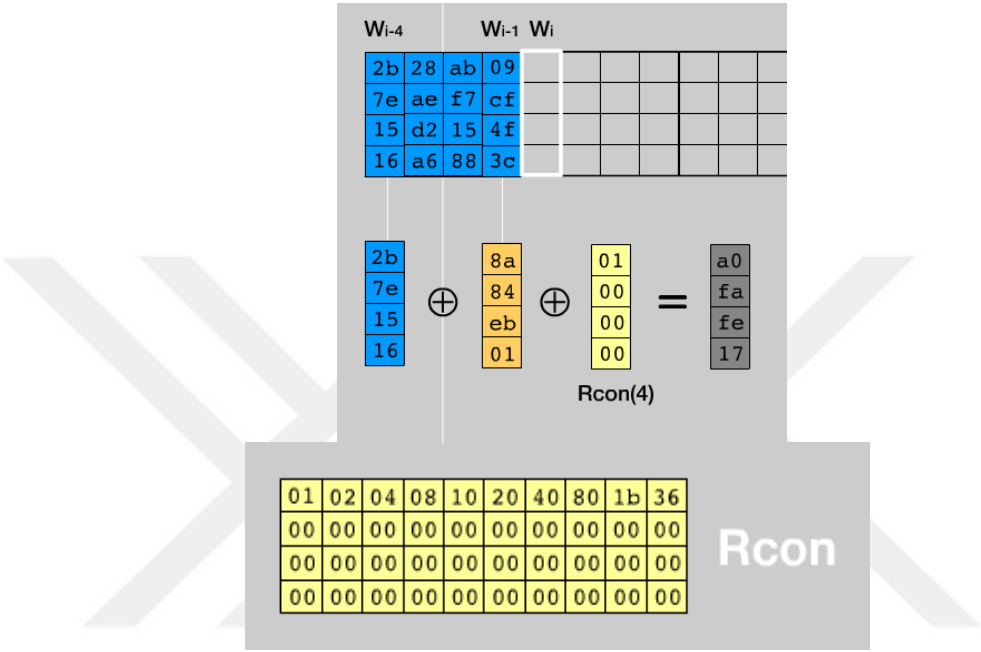
Şekil 3.19'da gösterilen tur anahtarlarıyla toplamada ise her turda daha önce saydığımız işlemlerle birlikte bir de tur anahtarı oluşturma işlemi yapılmaktadır ve her turda sonuçta oluşan durum ile o tur için hazırlanmış olan yeni anahtar toplama işlemine tabi tutulur.



Şekil 3.19 Tur anahtarıyla toplama

Bu işlem sonlu alanlarda yapılan toplama işlemidir ve bit mertebesinde özel veya (XOR) işlemine karşılık düşer. 128 bitlik durum matrisi ile 128 bitlik ara anahtar değeri bit bit özel veya elamanı ile toplanır.

Şekil 3.20'de anahtar oluşumu gösterilmiştir. Bu işlemde, bir önceki sütun ile dört önceki sütunun ve oluşturulmuş Rcon sütununun toplanmasıdır [15, 61, 69-73].

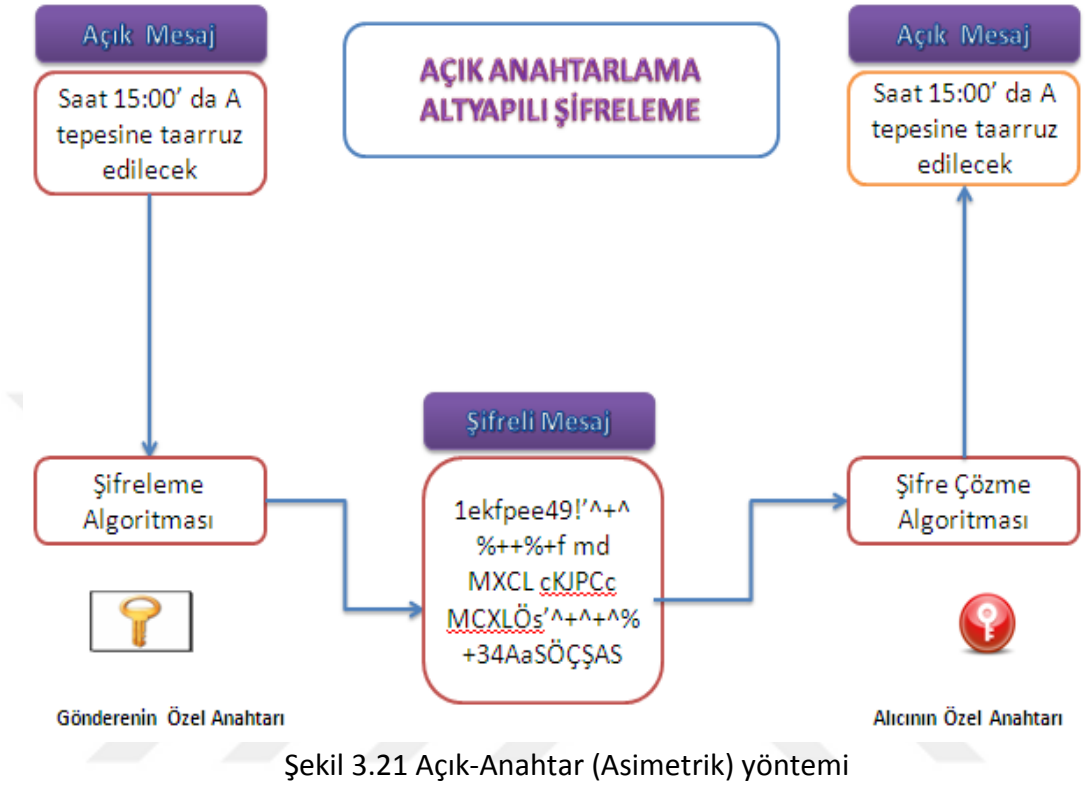


Şekil 3.20 Anahtar oluşumu

3.1.2 Açık-Anahtar (Asimetrik) yöntemi

1976 yılında Stanford Üniversitesinden Diffie ve Hellman adlı araştırmacılar iki farklı anahtara dayalı şifreleme sistemi önermiştir. Bu sistemde bir tane şifreleme için (public key) ve bundan farklı olarak bir tane de şifre çözmek için (private key) anahtar bulunur. Asimetrik şifreleme algoritmalarında çok büyük asal sayılar kullanılmaktadır. Anahtar sadece ilgili kişi tarafından kullanılabilir. Bu modele göre şifreleme anahtarı (aynı zamanda açık anahtar olarak anılır ve kamuoyuna açıktır), şifreyi çözme anahtarına (aynı zamanda gizli anahtar olarak anılır ve sadece şifreyi çözmeye yetkili kişinin bilgisi dâhilindedir) dair çok az bilgi verir ve tersi de doğrudur. Şifreleme ve çözme işlemi birbirinin simetriği olmayan algoritmalarla gerçekleştirildiğinden dolayı da asimetrik şifreleme sistemi olarak bilinir. Bu sistemde şifreleme işlemi herkes tarafından bilinen açık anahtarla yapılır. Açık anahtarlı şifrelemede, özel anahtar her zaman matematiksel olarak açık anahtara bağlıdır. Bu yüzden, açık anahtardan özel anahtar üretilerek açık

anahtarlı bir sistemin şifresinin kırılması her zaman için mümkündür. Bu duruma önlem olarak, açık anahtardan özel anahtar türetilmesinin mümkün olduğunca çok zor gerçekleşmesi sağlanmalıdır.



Şekil 3.21 Açık-Anahtar (Asimetrik) yöntemi

Açık-anahtar (asimetrik) yöntemi, iki katılımcının öncesinde herhangi bir bilgi alışverişi yapmadan güvenli olmayan bir kanal vasıtasıyla (güvenli bir şekilde) ortak bir şifrede karar kılmalarına yarayan bir protokoldür. Algoritmanın anahtar değişimi ile sağladığı asıl amaç, iki kullanıcının bir anahtarı güvenli bir şekilde birbirlerine iletmeleri ve daha sonrasında da bu anahtar yardımı ile şifreli mesajları birbirlerine gönderebilmelerini sağlamaktır. [15, 72, 74].

3.1.2.1 RSA

Dünyada en yaygın biçimde kullanılan asimetrik algoritma, ismini mucitlerinin baş harflerinden (Ronald L.Rivest, Adi Shamir ve Leonard Adleman) almıştır. Büyük sayıların modüler aritmetiğine dayalı çok basit bir prensibi vardır. Anahtarlar, iki büyük asal sayıdan üretilir. Dolayısıyla, algoritmanın güvenliği büyük sayı üretme problemine dayalıdır [74]. Anahtar oluşturma algoritması şu şekildedir:

P ve Q gibi çok büyük iki asal sayı seçilir. Bu iki asal sayının çarpımı $N = P.Q$ ve bu sayıların bir eksiklerinin çarpımı $\phi(N) = (P-1)(Q-1)$ hesaplanır. 1'den büyük $\phi(N)$'den küçük ve $\phi(N)$ ile aralarında asal bir E tamsayısı seçilir. Seçilen E tamsayısının mod $\phi(N)$ 'de tersi alınır, sonuç D gibi bir tamsayıdır. E ve N tamsayıları genel anahtarı, D ve N tamsayıları ise özel anahtarı oluşturur.

Şifreleme ve deşifreleme işlemleri ise şu şekildedir:

Herhangi bir M mesajı için şifreleme $C = M^E \ (mod\ N)$

C şifreli metni için deşifreleme $M = C^D \ (mod\ N)$

Örnek: $P = 7\ ve\ Q = 17$ gibi çok büyük iki asal sayı seçilsin.

Bu iki asal sayının çarpımı $N = P.Q = 7 * 17\ ;\ N = 119$ ve bu sayıların bir eksiklerinin çarpımı $\phi(N) = (P-1)(Q-1) = 6 * 16;\ \phi(N) = 96$ olarak hesaplanır. 1'den büyük $\phi(N)(96)$'den küçük ve 96 ile aralarında asal bir $E = 5$ tamsayısı seçilsin. Seçilen $E = 5$ tamsayısının mod 96'da tersi alınır, sonuç $D = 77$'dir. $E = 5$ ve $N = 119$ tamsayıları genel anahtarı, $D = 77$ ve $N = 119$ tamsayıları ise özel anahtarı oluşturur. (5,119) anahtarları ile şifreleme, (77,119) anahtarı ile deşifreleme yapılacaktır.

M açık metni 19 olarak seçilsin.

$C = M^E$ (mod N)→ $C = 19^5 \ (mod\ 119) \rightarrow C = 66$

$M = C^D \ (mod\ N) \rightarrow M = \ 66^{77} \ (mod\ 119) \rightarrow M = 19$ olarak bulunur.

3.1.3 Hibrit yöntemi (simetrik+asimetrik beraber uygulaması)

Saldırganlara karşı mesajların gizliliğini korumak için simetrik ve asimetrik şifreleme algoritmaları kullanılır. Hibrit şifreleme algoritmaları, simetrik ve asimetrik şifreleme algoritmalarının avantajlarını birleştirir. Hibrit şifreleme sisteminde mesajı şifrelemek için simetrik şifreleme algoritması, mesajı şifrelemek için kullanılan simetrik anahtarı şifrelemek için de asimetrik şifreleme algoritması kullanılır.

Genel olarak şifre kırmak için kullanılan 3 yöntem vardır:

Sözlük Tabanlı Şifre Kırma Yöntemi: Bu yöntemin esası sözlük formatına uygun olarak yazılmış şifrelerin bir dosyaya saklanarak, bununla şifrelerin kırılmasına dayanır. Birçok insan şifrelerini sözlüklerde bulunan kelimelerden seçtiği için bu yöntem işe yarar. Bu

yöntem genellikle şifreleri sırayla denemeye yarayan bir “script” ile kullanılır. Script dosyası şifreleri tek tek dener ve doğru şifreye ulaşmaya çalışır. Web üzerinde bu yöntem kullanıldığında her işlem 10-15 saniye sürer. Oysa lokal olarak çalışan bir makinede çok çabuk biter. Bu yöntem şifre kırmada atılan ilk adımdır, ancak etkili değildir. Çünkü sisteme defalarca girmeye çalışan ama her defasında başarısız olan bir Hacker’ın saldırısı, sistem yöneticileri tarafından kolaylıkla fark edilir. İnternette rastgele kelime üreten ve bunları bir sözlük haline getirip kullanılmasını sağlayan çok sayıda program vardır.

Zorlamayla Şifre Kırma Yöntemi: Bruteforce olarak da bilinen bu yöntemde şifrelerin tahmini için karakterlerin her tür kombinasyonu denenir. Genellikle alfanümerik karakterler (az ve 09) ile başlanır ve özel karakterlerle (@,!,?,$,vs) devam eder. Bu yöntemin süresi kullanılan şifreye bağlı olmakla birlikte bazen aylar, bazen yıllar, bazen de saniyeler mertebesindedir.

Hibrit Yöntemi: Yukarıda anlatılan 2 yöntemin birleşmesinden oluşur. Sözlük tabanlı şifre kırma yöntemi hızlı ama etkisiz, zorlamayla şifre kırma tekniği ise yavaş ama etkilidir. Hibrit yöntemde sözlükteki şifrelerle zorlama tekniğiyle yeni karakterler eklenir ve şifre kırılmaya çalışılır. Mesela sözlük listesindeki bir şifre “adam” ise, zorlama tekniği bu kelimeye “adam01”, “adam02” gibi eklemeler yapar [15, 69, 70, 74].

BÖLÜM 4

SIKIŞTIRICI ALGILAMA DİK EŞLEŞTİRME ARAYIŞ YÖNTEMİ İLE GELİŞTİRDİĞİMİZ KRİPTOGRAFİ YÖNTEMLERİ

4.1 Sıkıştırıcı Algılama ile Simetrik Kriptografi Uygulaması

Orijinallik ve uygulama açısından son zamanlarda, iki önemli konu olan sıkıştırıcı algılama ve kriptografi konusunun birleştirilmesi için bu bağlamda sıkıştırıcı algılama olarak oluşturduğumuz Dik Eşleştirme Arayış(DEA) matlab programı ile hem sıkıştırma (data boyutu azaltma) hem de simetrik şifreleme ve deşifreleme uygulaması gerçekleştirilmiştir. Bu uygulama için [75] de belirtilen internet sitesinden elde edilen harflerin decimal karşılığı ile dışarıdan kullanıcının girdiği herhangi bir kelime ve/veya cümle öncelikle decimal formata çevrilmektedir, akabinde seyrek giriş sinyali oluşturulmaktadır. Bu giriş sinyalinden SA için az sayıda örnek alınmaktadır. Seyrek sinyali oluştururken ve örnek alınırken, Sıkıştırıcı Algılama konusunda bahsedilen $M \geq K * \log(N)$ bağıntısı kullanılmıştır. K seyreklik derecesi olarak kullanıcının dışarıdan girdiği harf sayısı alınmaktadır, daha sonra da yukarıdaki eşitsizliğe göre M ve N değerleri belirlenmektedir. Literatüre bakıldığında SA uygulamalarının çoğunda bu oranlara göre işaretten çok daha fazla örnekleme yapılmaktadır. Bu önerilen yöntemde ise daha az örnek almak ve işi zorlaştırmak için sınıra yakın değerlerde çalışılmıştır.

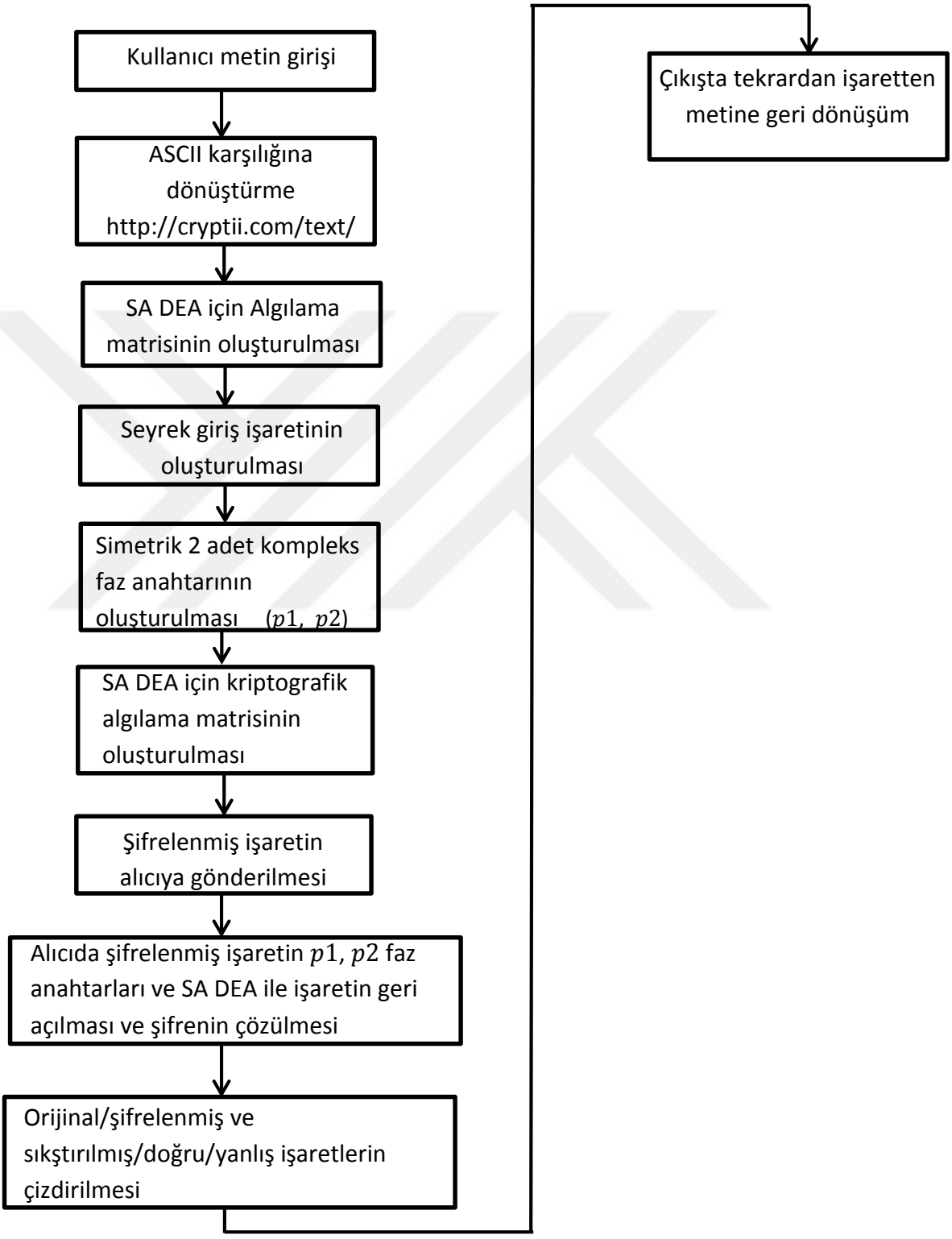
Programda, permütasyon matrisleri ile beraber algılama matrisi elde edilmektedir. Bu algılama matrisi oluşturulurken, dönüşümler olarak; hızlı fourier dönüşümü, ayrık kosinüs dönüşümü, orijinallik sağlaması bakımından ayrık kosinüs-III dönüşümü, gerçek sinüsoidal dönüşümü, haar dalgacık dönüşümü, hadamard dönüşümü ve ayrık sinüs dönüşümü kullanılmaktadır. Algılama matrisinin her çip teknolojisinde

kullanılabilmesi ve giriş işaretinin durumunun sonucu değiştirmemesi için algılama matrisi giriş işaretinden bağımsız olarak oluşturulmaktadır. Bu yöntemdeki $AAt = p2 * AA * p1$ satırında, $p2$ ve $p1$ simetrik (hem alıcı da hem de verici de aynı anahtar kullanılmaktadır) diagonal anahtarlardır. AA ise dönüşümler kullanarak elde edilen algılama matrisidir. Kriptografinin güçlü olması için $p1$ ve $p2$ faz anahtarları kompleks rastgele sayılardan oluşmaktadır, kompleks sayıların kullanılması iyi saçılma sağlamaktadır, bu saçılma ile rastgele olma durumu artmakta ve şifreleme daha da karışık olmaktadır. Ayrıca bu kompleks sayıların 360 derece tam yayılması için 2π ile çarpma işlemi yapılmaktadır. Ayrıca saçılmaya daha da artırmak ve şifrelemeyi daha da zorlaştırmak için algılama matrisi, tekrar kompleks ve 2π ile çarpılmış rastgele sayılardan oluşan ikinci bir faz anahtarı $(p2)$ ile çarpılmaktadır, böylece 2 defa şifreleme işlemi yapılmış olmaktadır. AA ise permütasyon algılama matrisi olmaktadır, AAt ise kriptografik anahtarların yerleştirildiği algılama matrisinin tamamı olmaktadır, daha sonra ise $b = AAt * x$ ile kriptolanmış ve sıkıştırılmış (şifrelenmiş işaret) ölçüm vektörü elde edilmektedir. Ölçüm vektörü ile sıkıştırıcı algılama dik eşleştirme arayış algoritması (SA DEA) kullanarak(sinyal tekrar açılarak) orijinal işaret(şifresi çözülmüş işaret) elde edilmektedir.

DEA algoritması boş bir matris ile algoritmayı çalıştırmaya başlar. Örnekleme matrisinin sıfırdan farklı katsayılarla ilişkili olduğunu saptadığı kolonlar ile bu boş matrisi doldurmaya başlar. Her tekrarda bir tane kolon seçerek bu matrisi oluşturur. Algoritma her tekrarda örneklenmiş sinyal y ile en çok ilişkili kolonu seçer. Algoritma, K tekrardan sonra orijinal seyrek sinyalin sıfırdan farklı katsayılarıyla ilişkili kolonları seçmiş olur. Programı genel olarak hem şifreleme ve hem de şifre açma olarak ikiye ayırabiliriz, DEA algoritmasının uygulamasına kadar anahtar oluşumu ile şifreleme, DEA ile beraber şifre çözme olarak çalışmaktadır.

Program sonucunda doğru anahtar ve yanlış anahtar ile hata ve sonuç, işlem süresi, SA ile kriptosu açılmış text değerleri ekrana yazdırılmaktadır, ayrıca orijinal işaret, şifrelenmiş işaret, doğru anahtar ve yanlış anahtar ile elde edilmiş SA+Kripto sonucu çizdirilmektedir. Programın sonunda doğru kriptografi ve doğru haberleşmenin kontrolü için işaretten tekrar texte dönüşüm yapılmaktadır. Bu uygulama ile sıkıştırıcı algılama dik eşleştirme arayış yöntemi ve kriptografi birbirinden bağımsız olarak değil

de, birbirinin içine geçmiş bir şekilde kullanılmıştır. Böylece hem sıkıştırıcı algılama ile data boyutu azaltma hem de SA algılama matrisinde güçlendirilmiş 2 defa faz şifrelemesi ve şifre çözme ile kriptografi elde edilmektedir. Program algoritması ve örnek program sonuçları aşağıdadır.



Şekil 4.1 Sıkıştırıcı algılama ile simetrik kriptografi uygulama algoritması

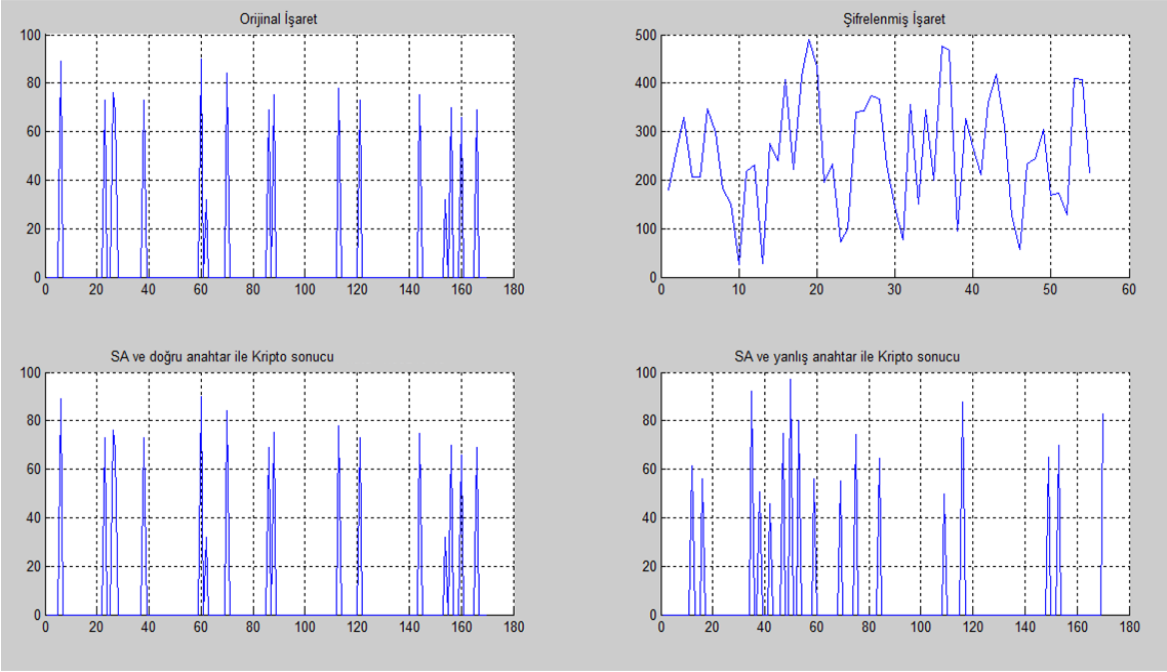
```
Kesme işareti arasına büyük harf kullanarak metin giriniz :'YILDIZ TEKNIK FBE'
Doğru anahtar ile hata 1.35e-15

Yanlış anahtar ile hata 4.05e+02

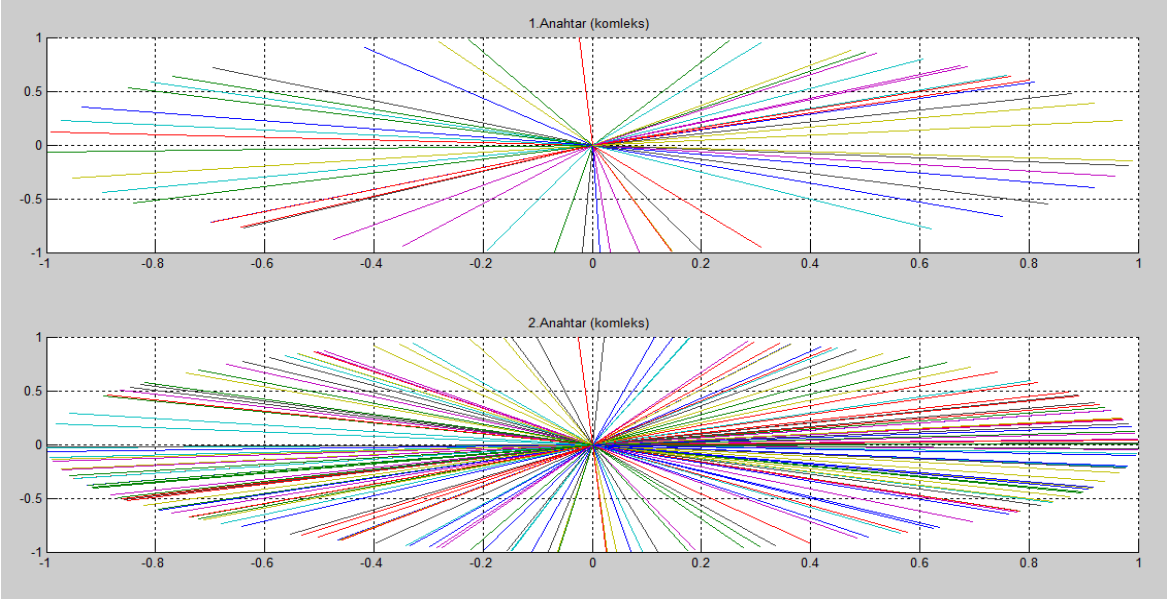
SA ile Kriptosu açılmış Text =  YILDIZ TEKNIK FBE

Elapsed time is 3.602798 seconds.
```

Şekil 4.2 Sıkıştırıcı algılama ile simetrik kriptografi uygulama çıktısı



Şekil 4.3 Sıkıştırıcı algılama ile simetrik kriptografi uygulama sonucu, orjinal ve şifreli işaret, doğru ve yanlış anahtar sonucu



Şekil 4.4 Kompleks, rastgele ve dağıtılmış 2 faz anahtarı

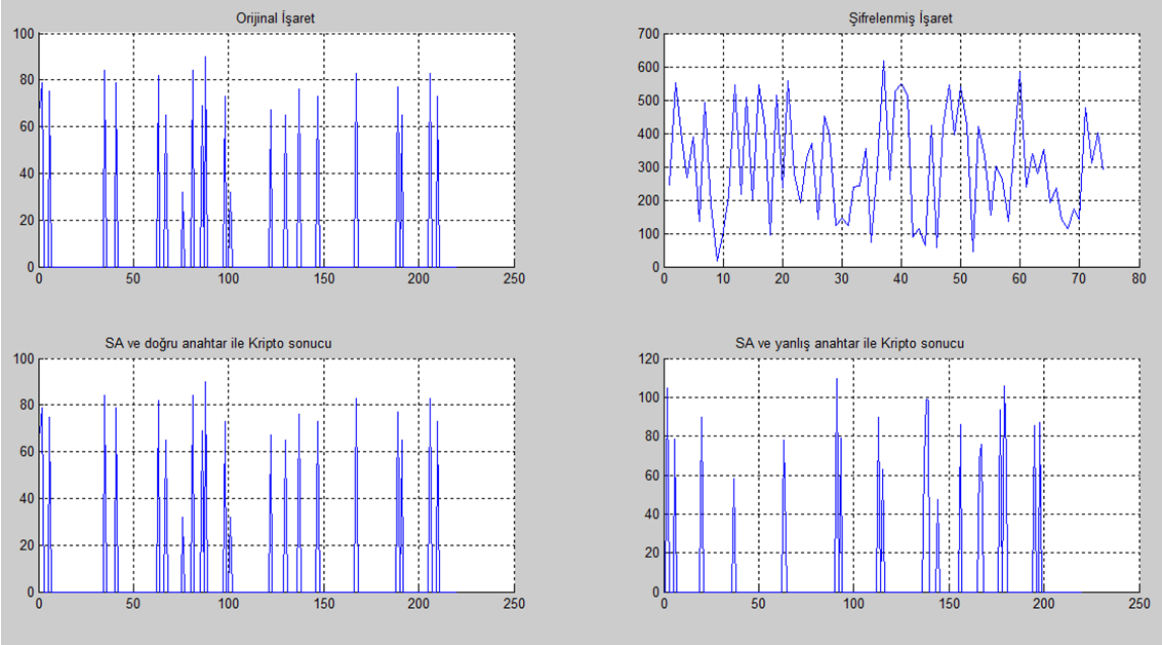
```
Kesme işareti arasına büyük harf kullanarak metin giriniz : 'DOKTORA TEZI CALISMASI'
Doğru anahtar ile hata 2.17e-15

Yanlış anahtar ile hata 5.34e+02

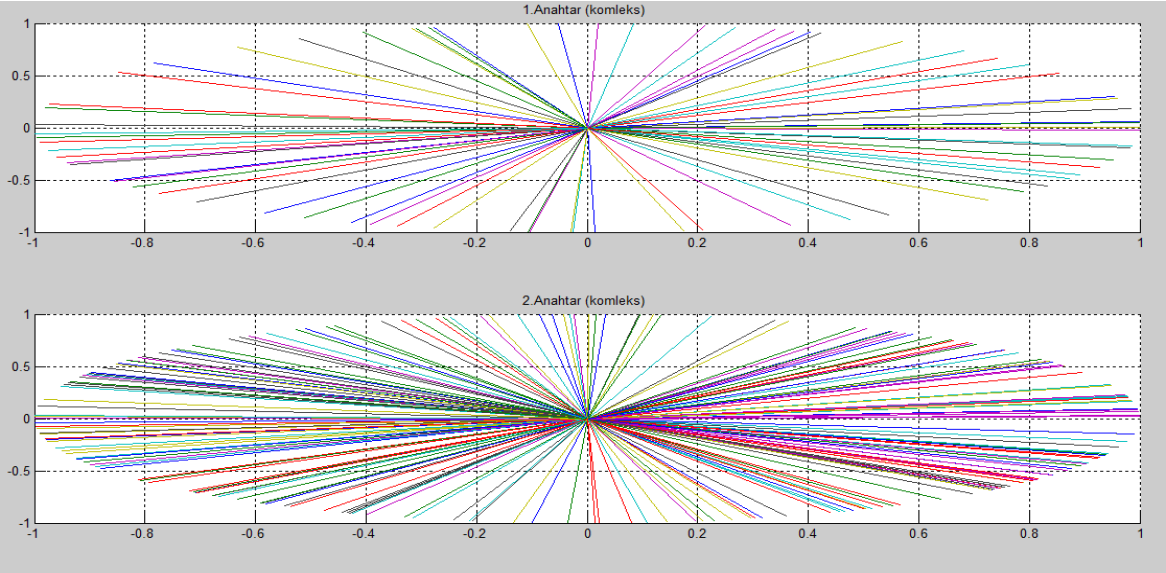
SA ile Kriptosu açılmış Text =  DOKTORA TEZI CALISMASI

Elapsed time is 2.073584 seconds.
```

Şekil 4.5 Sıkıştırıcı algılama ile simetrik kriptografi uygulama çıktısı



Şekil 4.6 Sıkıştırıcı algılama ile simetrik kriptografi uygulama sonucu, orjinal ve şifreli işaret,doğru ve yanlış anahtar sonucu



Şekil 4.7 Kompleks, rastgele ve dağıtılmış 2 faz anahtarı

Yukarıda şekil 4.3 ve şekil 4.6 da yatay eksen işaretin uzunluğunu, dikey eksen ise işaretin ASCII karşılığını göstermektedir. Şekil 4.4 ve şekil 4.7 deki eksenler, kompleks faz anahtarların i ve j eksen çizimlerini göstermektedir. Şekil 4.2 ve şekil 4.5 de giriş işareti olarak herhangi bir metin girilerek doğru anahtar ve yanlış anahtar ile kriptografi hata sonucu verilmektedir. Ayrıca alıcıda şifresi çözülmüş işaretin kontrolü ve tüm işlemlerin süre değeri gösterilmektedir. Şekil 4.3 ve şekil 4.6 da girişe göre ASCII karşılığı orijinal ve şifreli işaret, doğru ve yanlış anahtar sonuçları belirtilmiştir. Bu şekillere bakıldığında şifreli işaretin orijinal işaret boyutundan çok daha küçük olduğu ve çok karışık olduğu görülmektedir. Doğru anahtarlar ile alıcıda aynı orijinal işaret elde edilmekte, yanlış anahtarlar ise orijinal işaretten tamamen farklı bir işaret elde edilmektedir. Son olarak şekil 4.4 ve şekil 4.7 de ise kompleks, rastgele ve 360 derece tam dağıtılmış şifrelemede ve şifre çözmede kullanılan 2 faz anahtarları görülmektedir.

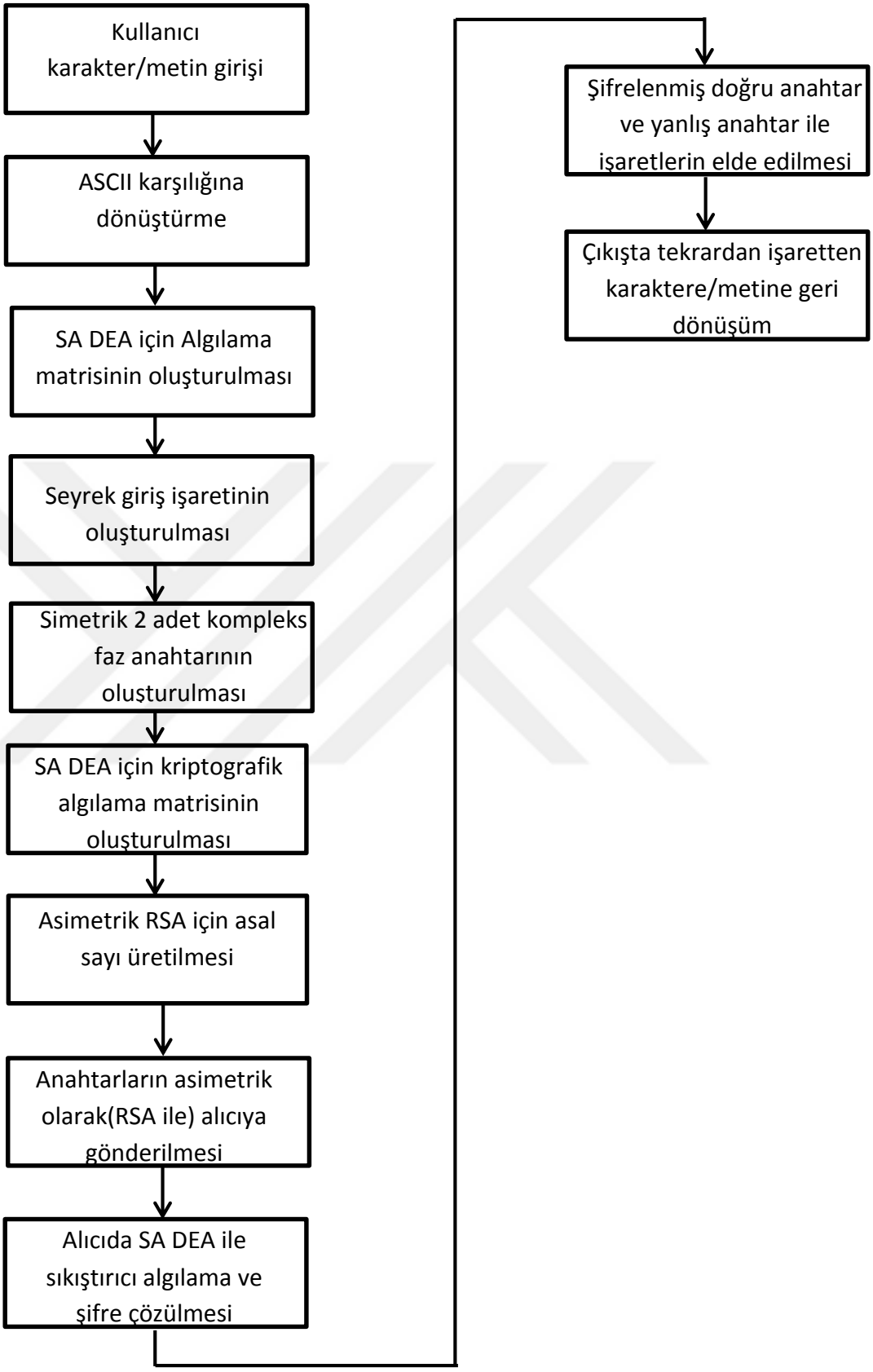
4.2 Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) Kriptografi (Aynı Anahtarları Kullanarak)

Bu bölümde, Sıkıştırıcı Algılama+Simetrik Kriptografinin üzerine, güvenliği daha da artırmak için Sıkıştırıcı Algılama+Hibrit Kriptografi uygulaması gerçekleştirilmiştir. Buradaki asıl amaç, simetrik kriptografide anahtarları karşıya daha güvenli olması için asimetrik kriptografi ile taşımaktır. Ayrıca bir önceki simetrik algoritma için oluşturulan ana program sadece büyük harfler için çalışırken, yapılan geliştirme ile büyük harf, küçük harf, rakam, özel karakter ve tüm karakterleri kapsayacak şekilde çalışır duruma getirilmiştir. Bununla beraber [75] de belirtilen internet sitesine gerek kalmadan, dönüşüm yapılmıştır. Hibrit Kriptografi uygulamasında hem simetrik kriptografi hem de asimetrik kriptografi uygulaması gerçekleştirilmiştir. Asimetrik kriptografi için RSA algoritması kullanılmıştır. RSA algoritmasının temelinde asal sayılar bulunmaktadır. RSA algoritması için asal sayı üreten ve bu asal sayılarla şifreleme/şifre çözme anahtarları elde etmek için matlab programı oluşturulmuştur. Matlabın kendi mod fonksiyonu yüksek sayılarda hata verdiği için araştırma yaparak yeni bir mod fonksiyonu geliştirilmiştir. Bu yeni mod programında iki defa mod fonksiyonu kullanılmıştır. Program, doğru bir şekilde çalışmaktadır.

Oluşturulan hibrit kriptografi programında sıkıştırıcı algılama olarak oluşturulan Dik Eşleştirme Arayış (DEA) matlab programı ile hem sıkıştırma (data boyutu azaltma) hem de simetrik ve asimetrik olarak (hibrit) şifreleme ve deşifreleme uygulaması gerçekleştirilmiştir. Dışarıdan girilen tüm karakterler giriş(orijinal) işaret olarak kabul edilmektedir. $M \geq K * \log(N)$ bağıntısına göre seyrek sinyali elde edilmektedir. K seyreklik derecesi olarak kullanıcının dışarıdan girdiği karakter sayısı alınmaktadır.

Algılama matrisini oluştururken hızlı fourier dönüşümü, ayrık kosinüs dönüşümü, orijinallik sağlaması bakımından ayrık kosinüs-III dönüşümü, gerçek sinüsoidal dönüşümü, haar dalgacık dönüşümü, hadamard dönüşümü ve ayrık sinüs dönüşümü kullanılmaktadır. Oluşturduğumuz programda $Alg_matrisi = p2 * AA * p1$ ifadesinde $p2$ ve $p1$ simetrik şifrelemede kullandığımız faz anahtarlarıdır. Asimetrik kriptografi için belirtilen bu 2 faz kompleks anahtarın reel ve imajiner kısmı ayrılmaktadır ve oluşturduğumuz asimetrik kriptografi algoritması olan RSA ile (asimetrik olarak) anahtarlar karşıya (alıcıya) gönderilmektedir, alıcı da ise bu gelen anahtarları kullanarak (simetrik olarak) Sıkıştırıcı Algılama Dik Eşleştirme Arayış Algoritmasının çözümü ile şifre çözümü ve orijinal işaret elde edilmektedir.

Hibrit kriptografi uygulaması sonucunda, işlem süresi, hata değerleri, Sıkıştırıcı Algılama Dik Eşleştirme Arayış Algoritması ve anahtarlar ile şifresi çözülmüş karakter/metin değerleri gösterilmektedir, ayrıca orijinal işaret, şifrelenmiş işaret, doğru anahtar ve yanlış anahtar ile elde edilmiş SA DEA ve Kriptografi sonuçları da belirtilmektedir. Tüm işlemlerin sonunda kriptografinin ve haberleşmenin doğruluk kontrolü yapılmaktadır. Programın akış diyagramı, programın çıktıları ve program aşağıda belirtilmiştir.



Şekil 4.8 Sıkıştırıcı algılama ile hibrit (simetrik+asimetrik) kriptografi (aynı anahtarları kullanarak) uygulaması program algoritması

```
Kesme işareti arasına açık text giriniz = 'YILDIZ TEKNIK FBE'

Açık Text = YILDIZ TEKNIK FBE

Açık Text - (ASCII olarak) ;
89 73 76 68 73 90 32 84 69 75 78 73 75 32 70 66 69

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
=======================================================
RSA başlatıldı:

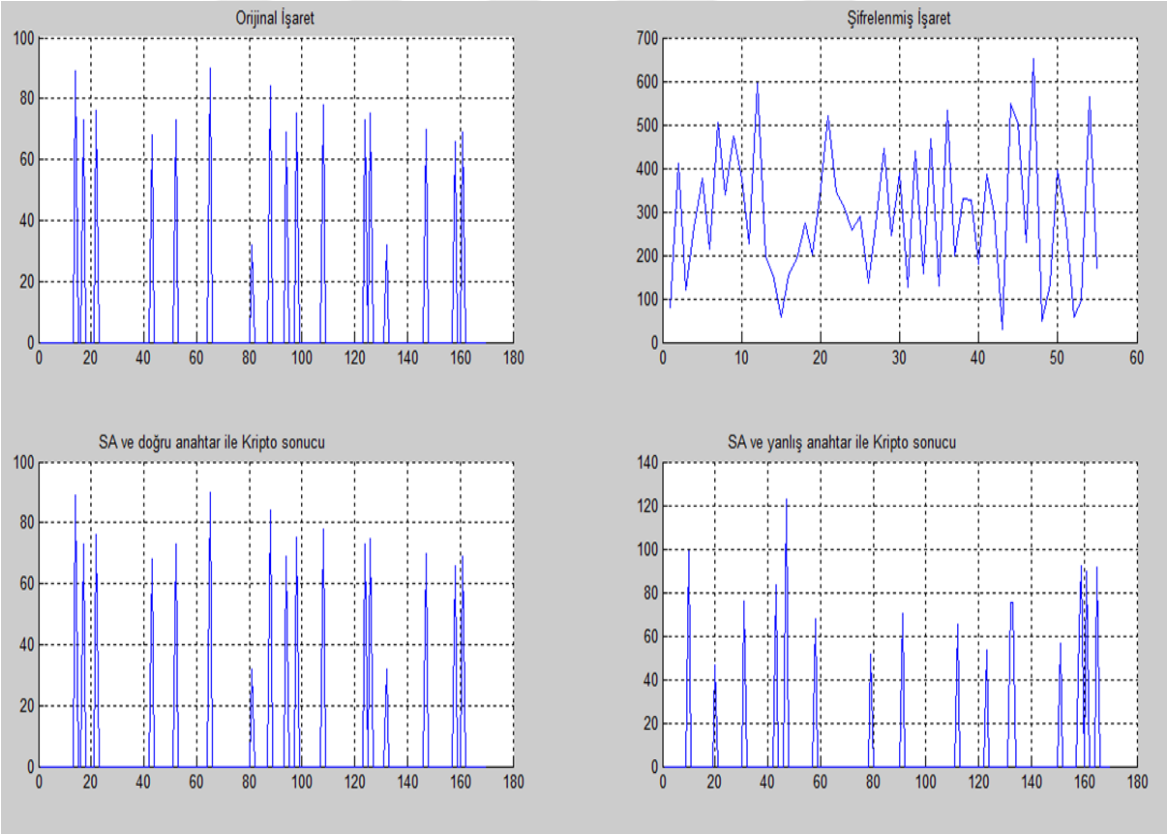
        P=499 - ilk asal değer (özel)
        Q=29 - ikinci asal değer (özel)
        N=14471 - modül (N) (genel)
        phi(N)=13944 - Euler fonksiyonunun degeri
        E=11 - kriptolama için genel anahtar (E) (genel)
        D=3803 - gizli kripto çözme anahtari (D) (özel)
=======================================================
Doğru anahtar ile hata = 1.92e-15

Yanlış anahtar ile hata = 4.35e+02

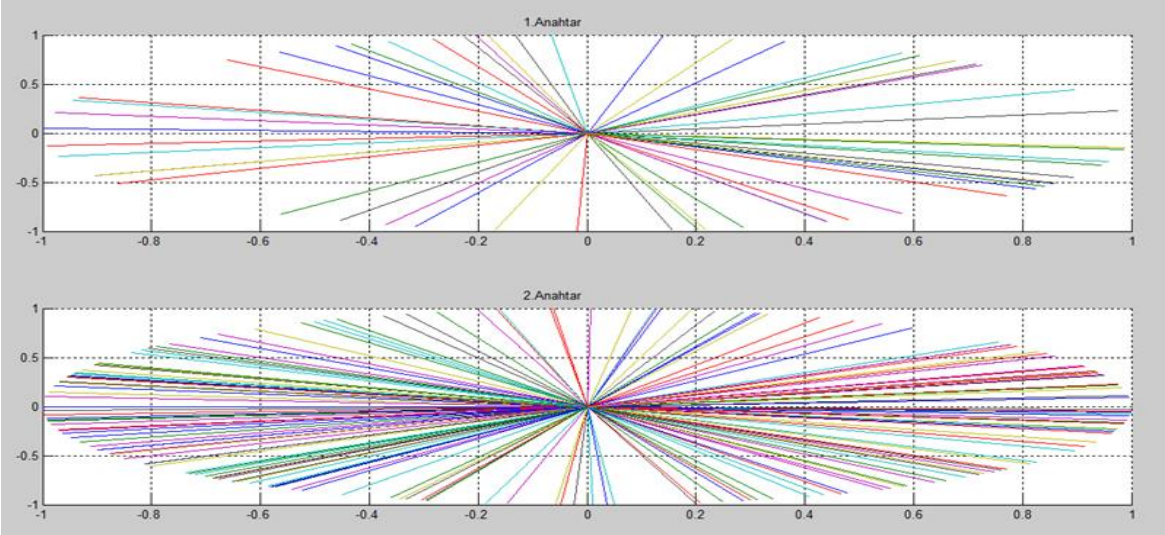
SA ile Kriptosu açılmış Text = YILDIZ TEKNIK FBE

Elapsed time is 8.211171 seconds.
```

Şekil 4.9 Sıkıştırıcı algılama ile hibrit kriptografi (aynı anahtarları kullanarak) uygulama çıktısı



Şekil 4.10 Sıkıştırıcı algılama ile hibrit kriptografi (aynı anahtarları kullanarak) uygulama sonucu, orjinal ve şifreli işaret, doğru ve yanlış anahtar sonucu



Şekil 4.11 Kompleks, rastgele ve dağıtılmış 2 faz anahtarı

```
Kesme işareti arasına açık text giriniz = 'DOKTORA Çalışması (2015)'

Açık Text = DOKTORA Çalışması (2015)

Açık Text - (ASCII olarak) ;
68 79 75 84 79 82 65 32 199 97 108 305 351 109 97 115 305 32 40 50 48 49 53 41

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
=======================================================
RSA başlatıldı:

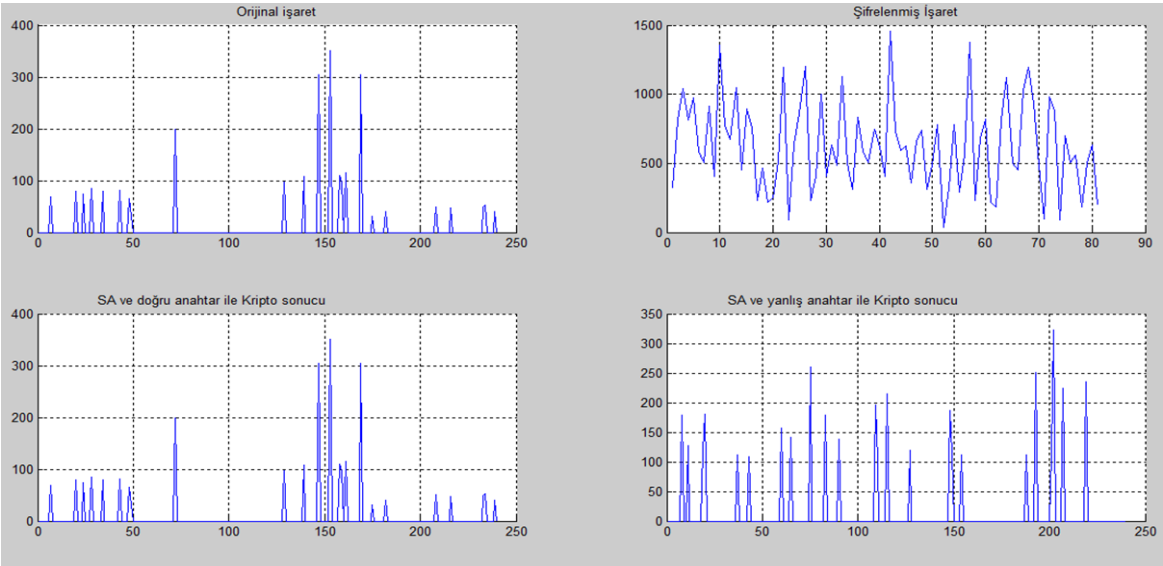
        P=743 - ilk asal değer (özel)
        Q=977 - ikinci asal değer (özel)
        N=725911 - modül (N) (genel)
        phi(N)=724192 - Euler fonksiyonunun degeri
        E=1115 - kriptolama için genel anahtar (E) (genel)
        D=1299 - gizli kripto çözme anahtari (D) (özel)
=======================================================
Doğru anahtar ile hata = 1.65e-15

Yanlış anahtar ile hata = 1.09e+03

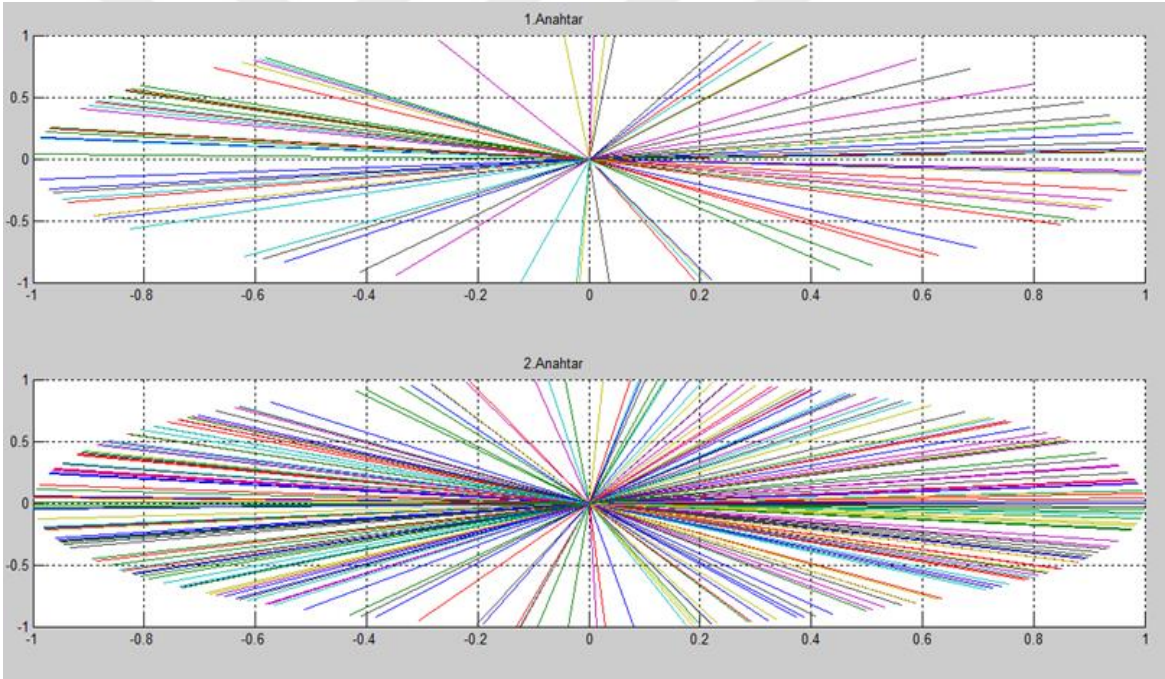
SA ile Kriptosu açılmış Text = DOKTORA Çalışması (2015)

Elapsed time is 2.126523 seconds.
```

Şekil 4.12 Sıkıştırıcı algılama ile hibrit kriptografi (aynı anahtarları kullanarak) uygulama çıktısı



Şekil 4.13 Sıkıştırıcı algılama ile hibrit kriptografi (aynı anahtarları kullanarak) uygulama sonucu, orjinal ve şifreli işaret, doğru ve yanlış anahtar sonucu



Şekil 4.14 Kompleks, rastgele ve dağıtılmış 2 faz anahtarı

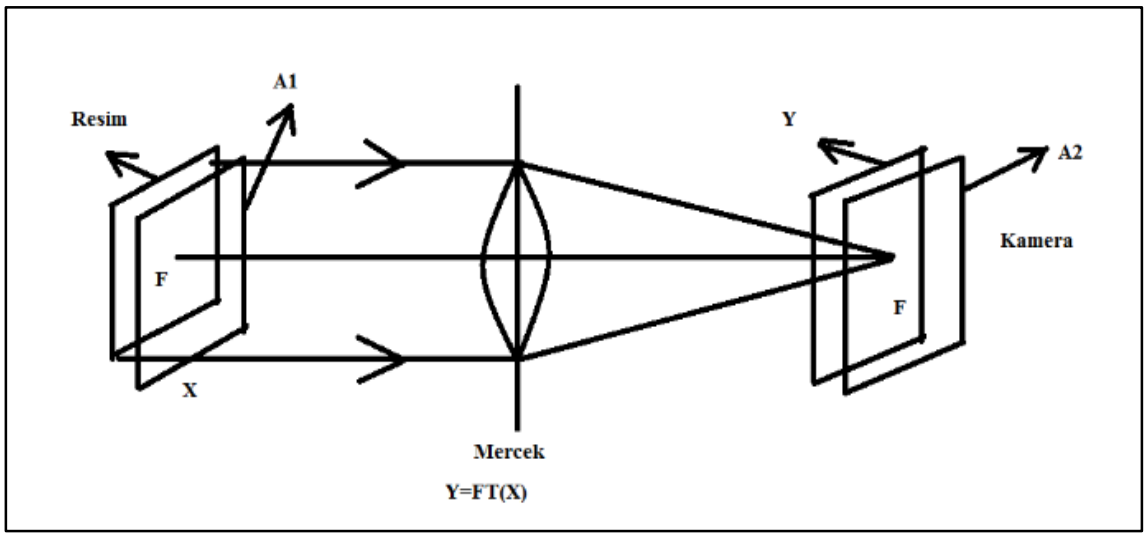
Yukarıda şekil 4.10 ve şekil 4.13 de yatay eksen işaretin uzunluğunu, dikey eksen ise işaretin ASCII karşılığını göstermektedir. Şekil 4.11 ve şekil 4.14 deki eksenler, kompleks faz anahtarların i ve j eksen çizimlerini göstermektedir. Önceki sayfalarda şekil 4.9 ve şekil 4.12 de giriş işaretinin ASCII karşılığı verilmektedir, asimetrik kriptografi için oluşturduğumuz RSA programı ile P ve Q gibi 2 rastgele asal sayı üretilmekte ve daha sonra da bu asal sayıları kullanarak E ve D ile genel ve özel anahtar değerleri elde edilmektedir. Şeklin sonunda da doğru anahtar ve yanlış anahtar ile oluşan hata

değerleri, alıcıda oluşan orijinal işaret ve işlem süresi belirtilmektedir. Şekil 4.10 ve şekil 4.13 de girişe göre oluşturulan seyrek işaretin ASCII karşılığı için orijinal ve şifreli işaret, doğru ve yanlış anahtar sonuçları gösterilmektedir. Bu şekillere bakıldığında şifreli işaretin orijinal işaret boyutundan çok daha küçük olduğu ve çok karışık olduğu görülmektedir. Doğru anahtarlar ile alıcıda aynı orijinal işaret elde edilmekte, yanlış anahtarlar ise orijinal işaretten tamamen farklı bir işaret elde edilmektedir. Son olarak şekil 4.11 ve şekil 4.14 de ise kompleks, rastgele ve 360 derece tam dağıtılmış şifrelemede ve şifre çözmede kullanılan 2 faz anahtarları görülmektedir.

4.3 Sıkıştırıcı Algılama ile Hibrit(Simetrik+Asimetrik) Optiksel Kriptografi (Eşlenik Anahtarlar ve fft Dönüşümü Kullanarak)

Bu uygulamada, optiksel kriptografi uygulaması gerçekleştirilmiştir. Optiksel kriptografinin şekli aşağıda belirtilmiştir. Önerdiğimiz uygulamada şifrelemenin daha kuvvetli olması için optik kriptografide iki mercek kullanılmıştır. Optik kriptografide, hem faz maskesi, akabinde de fourier dönüşümünün uygulanması, böylece resim (giriş işareti) tamamen gürültü olmaktadır. Oluşturduğumuz programda, optiksel kriptografideki mercek görevi için fourier dönüşümü kullanılmaktadır. $A1$ ve $A2$ faz maskeleridir.

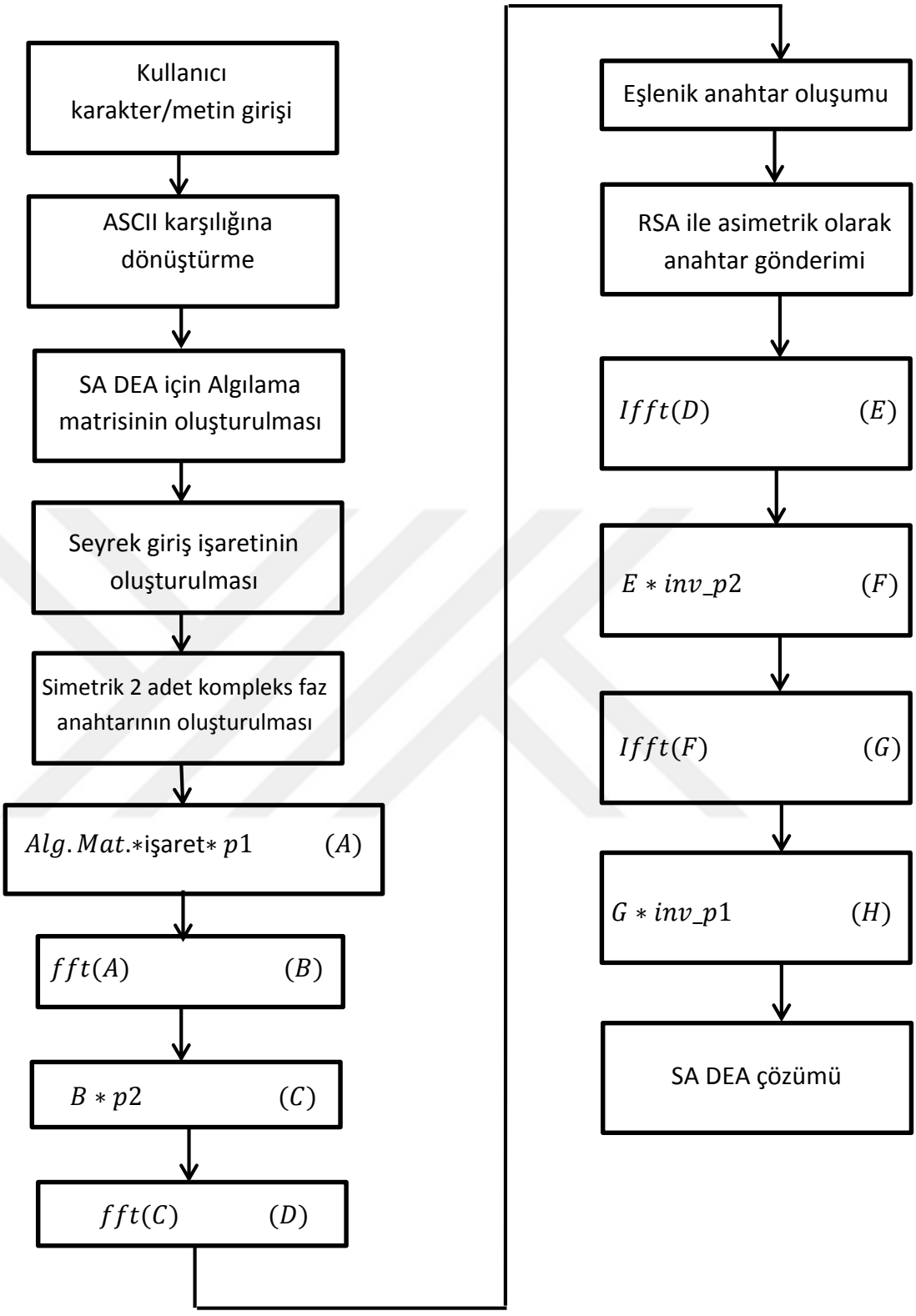
Sıkıştırılmış Algılama ile Hibrit Optiksel Kriptografide, simetrik kriptografide anahtarları karşıya daha güvenli olması için asimetrik kriptografi ile taşıma işlemi yapılmıştır. Uygulanan Hibrit Optiksel Kriptografide hem simetrik kriptografi hem de asimetrik kriptografi uygulaması gerçekleştirilmiştir. Asimetrik kriptografi için RSA algoritması kullanılmıştır. RSA algoritmasının temelinde asal sayılar bulunmaktadır. RSA algoritması için asal sayı üreten ve bu asal sayılarla şifreleme/şifre çözme anahtarları elde etmek için matlab programı oluşturulmuştur, matlabın kendi mod fonksiyonu yüksek sayılarda hata verdiği için yeni bir mod fonksiyonu organize edilmiştir.



Şekil 4.15 Optik şifreleme

Oluşturulan Hibrit Optiksel Kriptografi programında sıkıştırılmış algılama olarak, Dik Eşleştirme Arayış (DEA) algoritmasına göre yazılan matlab programı ile hem sıkıştırma (data boyutu azaltma) hem de simetrik ve asimetrik olarak (hibrit) şifreleme ve deşifreleme uygulaması gerçekleştirilmiştir. Dışarıdan girilen tüm karakterler giriş (orijinal) işaret olarak kabul edilmektedir. $M \geq K * \log(N)$ bağıntısına göre seyrek sinyal elde edilmektedir. K seyreklik derecesi olarak kullanıcının dışarıdan girdiği karakter sayısı alınmaktadır.

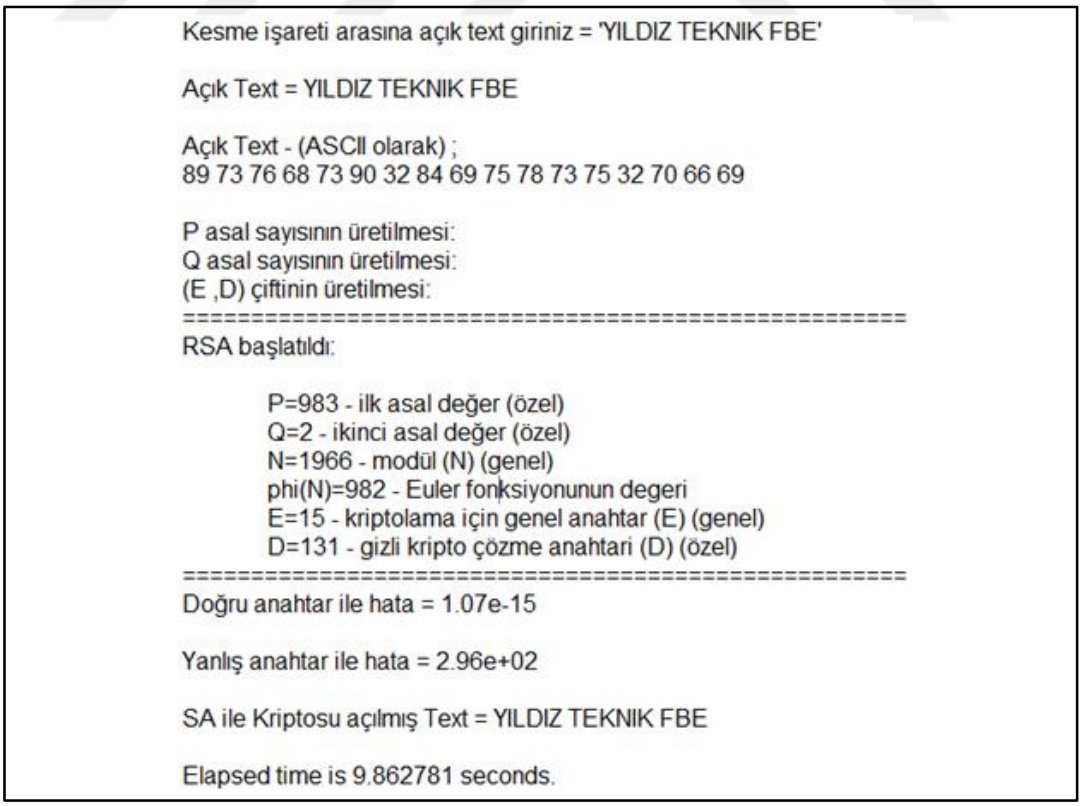
Algılama matrisinde dönüşümler olarak hızlı fourier dönüşümü, ayrık kosinüs dönüşümü, gerçek sinüsoidal dönüşümü, haar dalgacık dönüşümü, hadamard dönüşümü ve ayrık sinüs dönüşümü kullanılmaktadır. Ayrıca orijinallik olması bakımından ayrık kosinüs-III dönüşümü de kullanılmaktadır. Kriptografinin güçlü olması için; optiksel $p1$ ve $p2$ faz anahtarları kompleks rastgele sayılardan oluşmaktadır. Bu çift rastgele faz şifrelemesi ile kompleks sayıların kullanılması iyi saçılma sağlamaktadır. Saçılma sayesinde şifrelemenin güvenliği daha da artmaktadır. Ayrıca bu kompleks sayıların 360 derece tam yayılması için 2π ile çarpma işlemi yapılmaktadır, ayrıca da saçılmaya daha da arttırmak ve şifrelemeyi daha da zorlaştırmak için tekrar kompleks ve 2π ile çarpılmış rastgele sayılardan oluşan optiksel ikinci bir anahtar $(p2)$ ile çarpılmaktadır, böylece optiksel olarak çift rastgale faz şifreleme işlemi yapılmış olmaktadır.



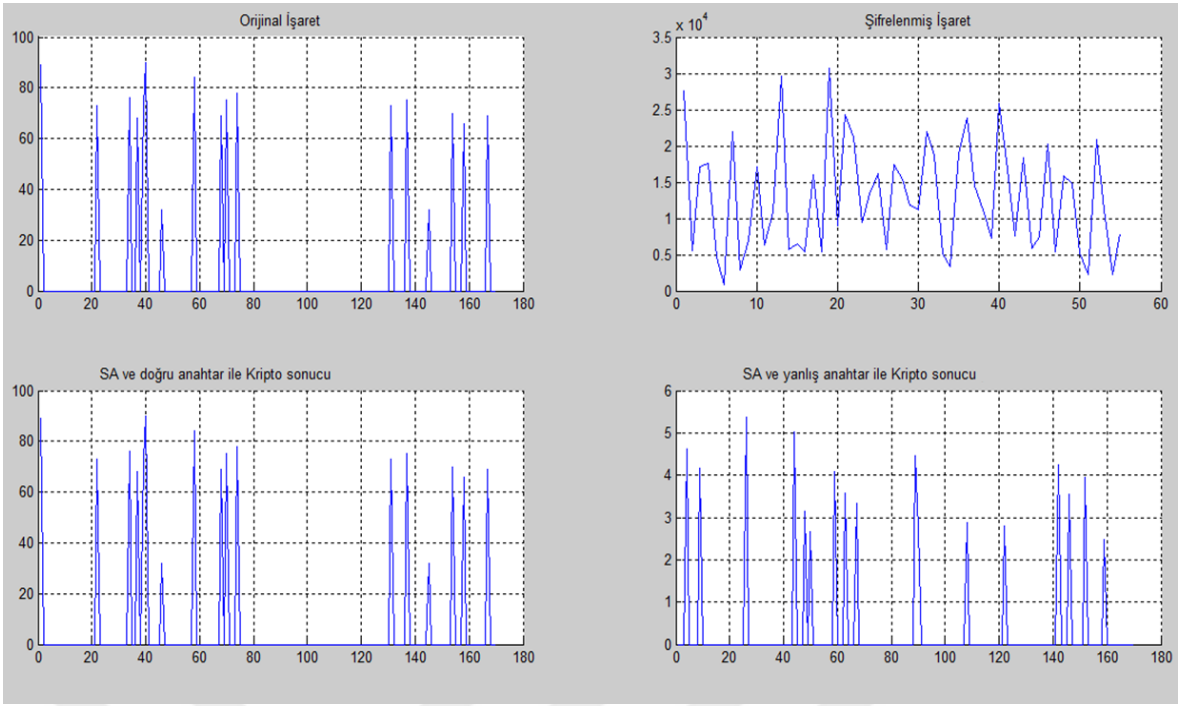
Şekil 4.16 Sıkıştırıcı algılama ile hibrit (simetrik+asimetrik) optiksel kriptografi (eşlenik anahtarlar ve fft dönüşümü kullanarak) uygulaması program algoritması

Asimetrik kriptografinin uygulanabilmesi için optiksel iki rastgale faz kompleks anahtarın reel ve imajiner kısımları ayrılmak suretiyle ve RSA ile asimetrik olarak anahtarlar alıcıya gönderilmiştir, alıcı da ise bu gelen anahtarları kullanarak simetrik olarak sıkıştırılmış algılama SA DEA'nın çözümü ile şifre çözümü ve orijinal işaret elde edilmektedir.

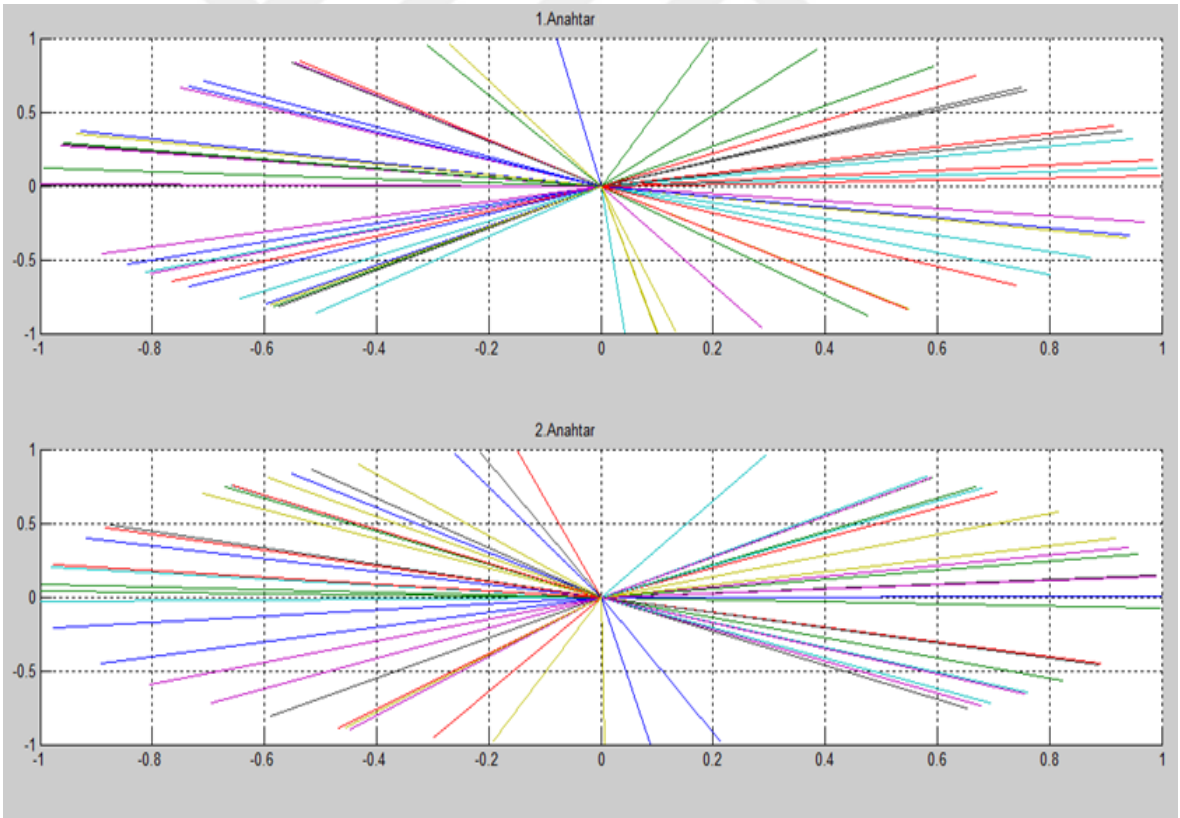
Güvenliği daha da artırmak için; Sıkıştırılmış Algılama + Hibrit Optiksel Kriptografi (eşlenik anahtarlar ve fft dönüşümü kullanarak) uygulaması yapılmıştır. Buradaki asıl amaç, saldırılara karşı daha dayanıklı olması için Hibrit Optiksel Kriptografide şifrelemede ve şifre çözmede aynı anahtarlar kullanmak yerine şifrelemede kullanılan optiksel çift rastgele faz şifreleme anahtarlarının ve fourier dönüşümlerinin eşleniklerini ve ters fourier dönüşümlerini şifre çözmede kullanılmaktadır. Ayrıca burada ilave olarak fourier dönüşümü de kullanılmaktır. Algoritma adımları aşağıda gösterilmiştir. Önerilen bu uygulama incelendiğinde şifrelenmiş işaret orijinal işarete göre çok daha fazla karışık olmaktadır. Böylece şifrelemenin güvenirliği artmaktadır. Programın sonuçları aşağıda sunulmuştur. Algoritma diğer farklı girişler için de doğru olarak çalışmaktadır.



Şekil 4.17 Sıkıştırıcı algılama ile hibrit optiksel kriptografi (eşlenik anahtar ve fft kullanarak) uygulama çıktısı



Şekil 4.18 Sıkıştırıcı algılama ile hibrit optiksel (eşlenik anahtarlar ve fft kullanarak) uygulama sonucu,orjinal ve şifreli işaret,doğru ve yanlış anahtar sonucu



Şekil 4.19 Kompleks, rastgele ve dağıtılmış 2 faz anahtarı

```
Kesme işareti arasına açık text giriniz = 'Doktora Tez Çalışmaları 2015'

Açık Text = Doktora Tez Çalışmaları 2015

Açık Text - (ASCII olarak) ;
68 111 107 116 111 114 97 32 84 101 122 32 199 97 108 305 351 109 97 108 97 114 305 32 50 48 49 53

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
=====================================================
RSA başlatıldı:

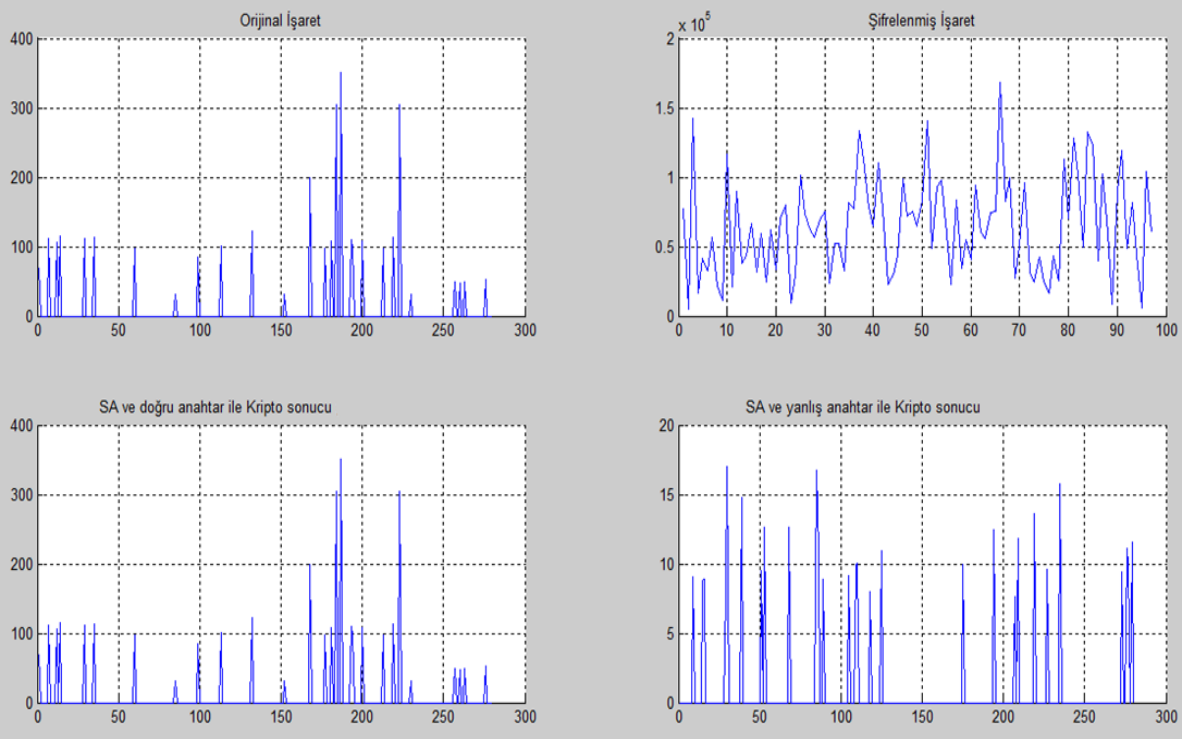
        P=733 - ilk asal değer (özel)
        Q=331 - ikinci asal değer (özel)
        N=242623 - modül (N) (genel)
        phi(N)=241560 - Euler fonksiyonunun degeri
        E=179 - kriptolama için genel anahtar (E) (genel)
        D=2699 - gizli kripto çözme anahtari (D) (özel)
=====================================================
Doğru anahtar ile hata = 1.55e-15

Yanlış anahtar ile hata = 7.38e+02

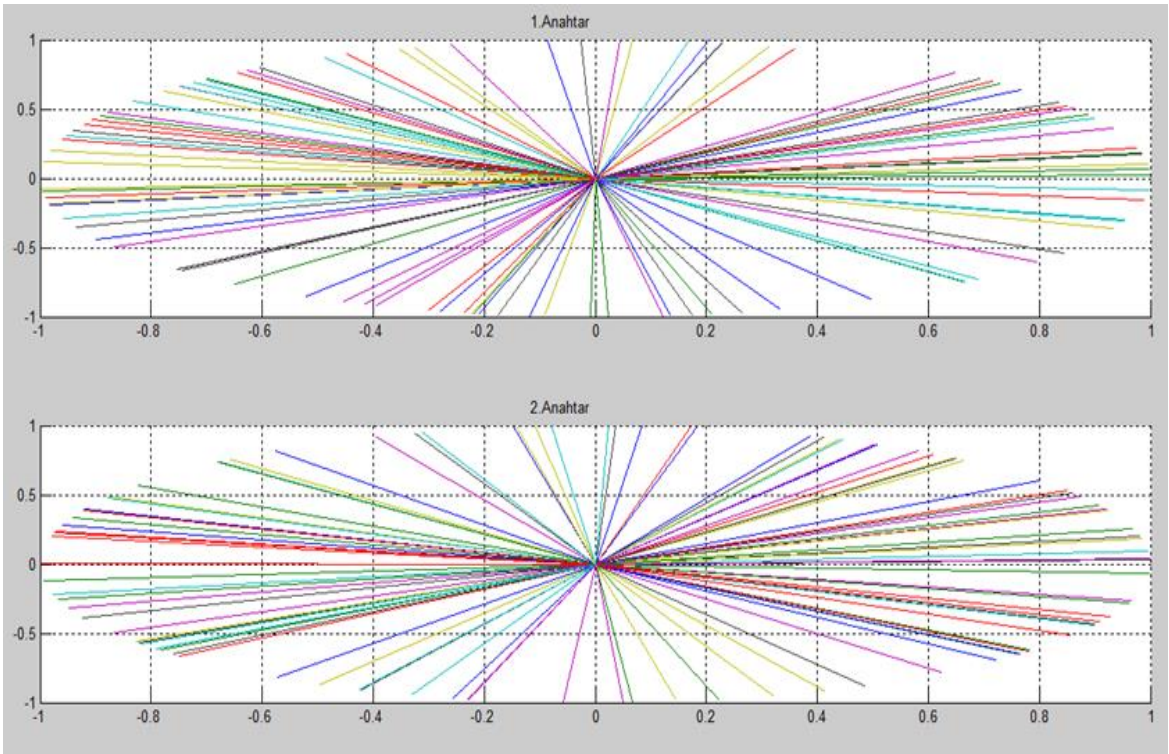
SA ile Kriptosu açılmış Text = Doktora Tez Çalışmaları 2015

Elapsed time is 1.848327 seconds.
```

Şekil 4.20 Sıkıştırıcı algılama ile hibrit optiksel kriptografi (eşlenik anahtar ve fft kullanarak) uygulama çıktısı



Şekil 4.21 Sıkıştırıcı algılama ile hibrit optiksel kriptografi (eşlenik anahtarlar ve fft kullanarak) uygulama sonucu,orjinal ve şifreli işaret,doğru ve yanlış anahtar sonucu



Şekil 4.22 Kompleks, rastgele ve dağıtılmış 2 faz anahtarı

Yukarıda şekil 4.18 ve şekil 4.21 de yatay eksen işaretin uzunluğunu, dikey eksen ise işaretin ASCII karşılığını göstermektedir. Şekil 4.19 ve şekil 4.22 deki eksenler, kompleks faz anahtarların i ve j eksen çizimlerini göstermektedir. Önceki sayfalarda şekil 4.17 ve şekil 4.20 de giriş işaretinin ASCII karşılığı verilmektedir, asimetrik kriptografi için oluşturduğumuz RSA programı ile P ve Q gibi 2 rastgele asal sayı üretilmekte ve daha sonra da bu asal sayıları kullanarak E ve D ile genel ve özel anahtar değerleri elde edilmektedir. Şeklin sonunda da doğru anahtar ve yanlış anahtar ile oluşan hata değerleri, alıcıda oluşan orijinal işaret ve işlem süresi belirtilmektedir. Şekil 4.18 ve şekil 4.21 de girişe göre oluşturulan seyrek işaretin ASCII karşılığı için orijinal ve şifreli işaret, doğru ve yanlış anahtar sonuçları gösterilmektedir. Bu şekillere bakıldığında şifreli işaretin orijinal işaret boyutundan çok daha küçük olduğu ve çok karışık olduğu görülmektedir. Doğru anahtarlar ile alıcıda aynı orijinal işaret elde edilmekte, yanlış anahtarlar ise orijinal işaretten tamamen farklı bir işaret elde edilmektedir. Ayrıca şifrelenmiş ve sıkıştırılmış işaret, önceki uygulamalara göre genliği çok daha büyük olmakta, bu da işaretin şifrelemesinin daha iyi olduğunu göstermektedir. Son olarak şekil 4.19 ve şekil 4.22 de ise kompleks, rastgele ve 360 derece tam dağıtılmış şifrelemede ve şifre çözmede kullanılan 2 faz anahtarları görülmektedir.

4.4 Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) Optiksel Kriptografi (2-Boyutlu)

Şimdiye kadar yaptığımız uygulamalarda, 1-boyutlu veriler üzerinde çalışılmıştır. Şimdi ise ilave olarak önerdiğimiz yöntemin sadece 1-boyutlu verilerde değil de görüntü verilerinde ve 2 boyutlu verilerde de doğru çalıştığının gösterilmesi amaçlanmıştır. Bunun için programda tüm dönüşümler 2-boyutlu olarak kullanılmıştır. 2-boyutlu sıkıştırıcı algılama ve hibrit optiksel kriptografi uygulamasında da; bir önceki uygulamada olduğu şekilde simetrik kriptografide anahtarları karşıya daha güvenli olması için asimetrik kriptografi ile taşınmaktadır. Hibrit Kriptografide hem simetrik kriptografi hem de asimetrik kriptografi uygulamasını gerçekleştirilmiştir. Uygulamanın gerçekleştirilmesinde asimetrik kriptografi RSA algoritması ile gerçeklenmiştir.

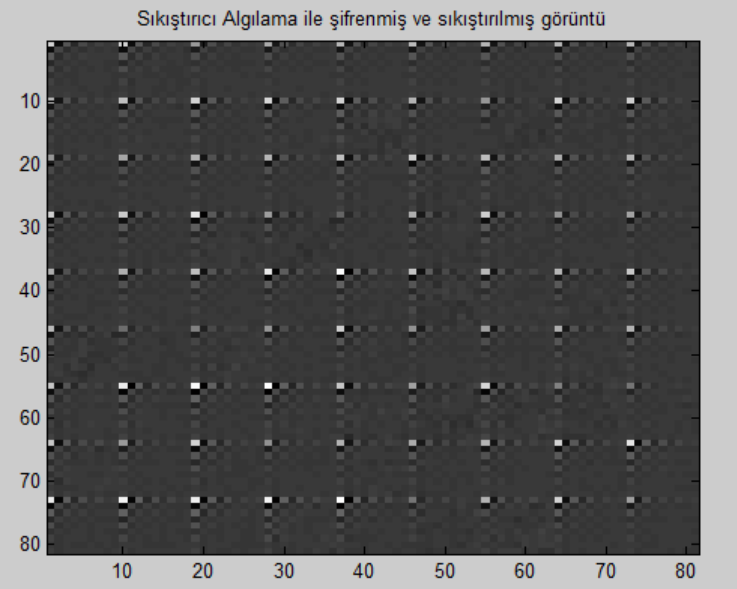
Oluşturulan 2-boyutlu hibrit kriptografi programında; sıkıştırıcı algılama olarak oluşturulan Dik Eşleştirme Arayış (DEA) matlab programı ile hem sıkıştırma (data boyutu azaltma) hem de hibrit optiksel şifreleme ve deşifreleme uygulamasını gerçekleştirilmiştir. 2-boyutlu data giriş(orijinal) işaret olarak kabul edilmektedir. $M \geq K * \log(N)$ bağıntısına göre seyrek sinyali elde edilmektedir. K seyreklik derecesi olarak kullanıcının dışarıdan girdiği 2-boyutlu data dikkate alınmaktadır. Programda, permütasyon matrisleri ile beraber algılama matrisi elde edilmektedir, bu algılama matrisi oluşturulurken, 2-boyutlu dönüşümler kullanılmıştır.

Aşağıda 3 ayrı giriş görüntüsü için uygulama çıktıları bulunmaktadır. Bu çıktılara bakıldığında şekil 4.23, şekil 4.27 ve şekil 4.31 orijinal giriş görüntüsü olmaktadır. Şekil 4.24, şekil 4.28 ve şekil 4.32 de önerdiğimiz algoritma sonucu oluşan şifrelenmiş ve sıkıştırılmış görüntüleri göstermektedir. Dikkat edildiğinde; orijinal görüntünün 256*256 boyutunda olduğu, şifrelenmiş ve sıkıştırılmış görüntünün ise 81*81 boyutunda olduğu görülmektedir. Ayrıca şifrelenmiş ve sıkıştırılmış görüntünün orijinal görüntü ile hiçbir ilişkisinin olmadığı da fark edilmektedir. Şekil 4.25, şekil 4.29 ve şekil 4.33 de yanlış anahtarlar ile şifrelenmiş ve sıkıştırılmış görüntünün açılmasını ifade etmektedir. Son olarak şekil 4.26, şekil 4.30 ve şekil 4.34 ise doğru anahtar ile şifrelenmiş ve sıkıştırılmış görüntünün SA DEA ile şifrelemenin çözülmesi ve görüntünün açılması durumlarını göstermektedir. Yanlış anahtarlar ile kesinlikle orijinal

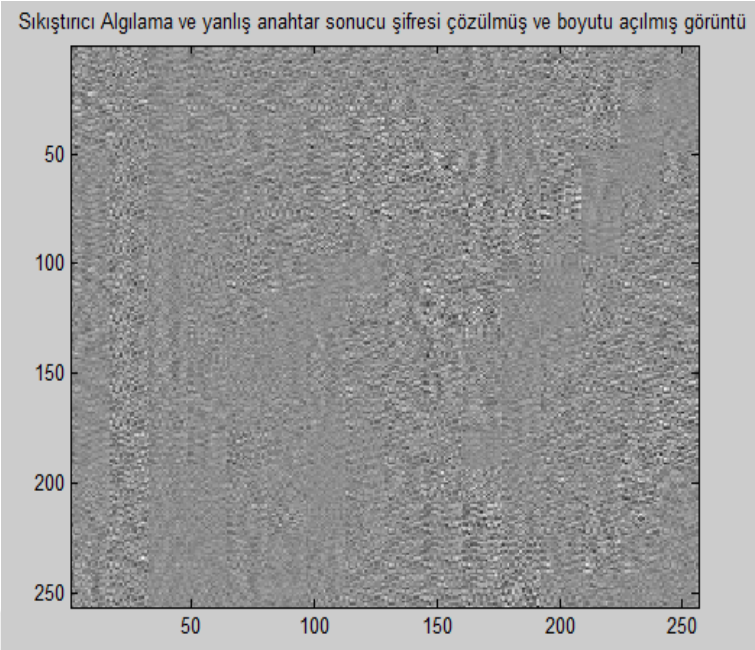
görüntü veya yakın bir görüntü elde edilememektedir. Doğru anahtarlar ile tekrardan orijinal görüntü elde edilmektedir.



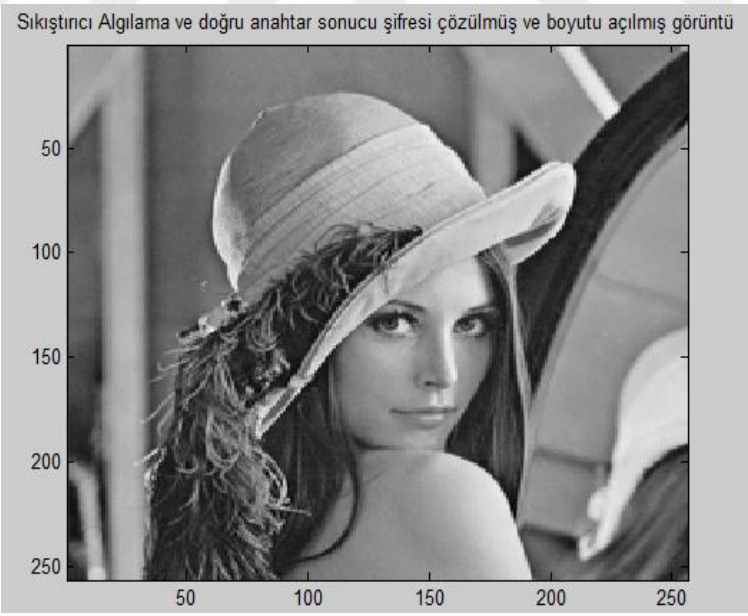
Şekil 4.23 Örnek 1 Orijinal görüntü



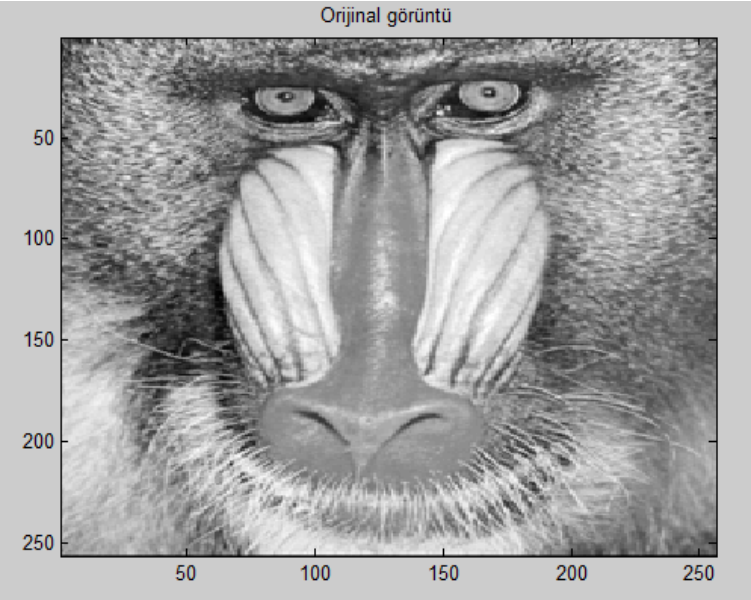
Şekil 4.24 Örnek 1 Sıkıştırıcı algılama ile şifrelenmiş ve sıkıştırılmış görüntü



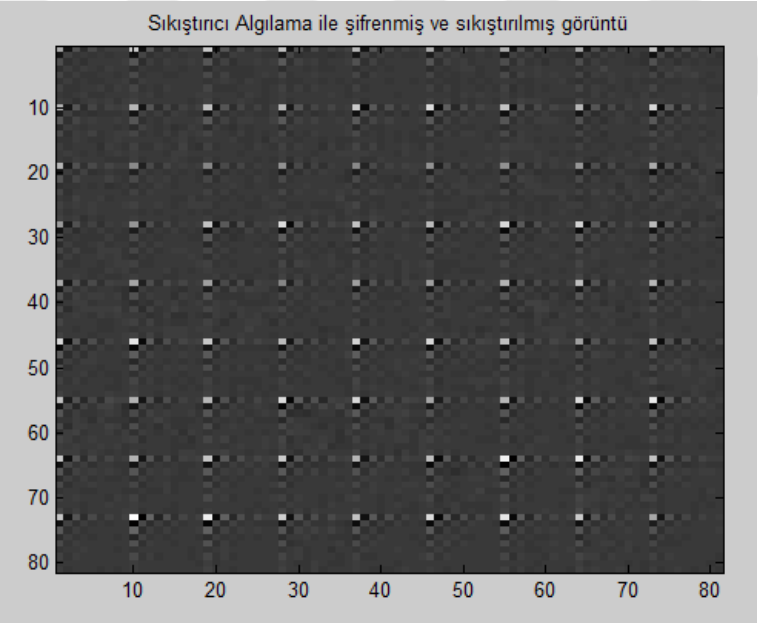
Şekil 4.25 Örnek 1 Sıkıştırıcı algılama ve yanlış anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü



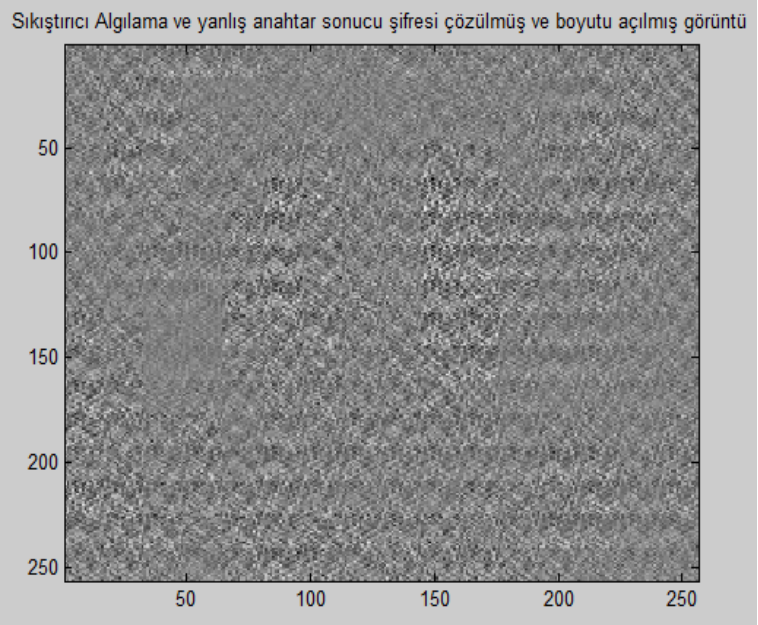
Şekil 4.26 Örnek 1 Sıkıştırıcı algılama ve doğru anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü



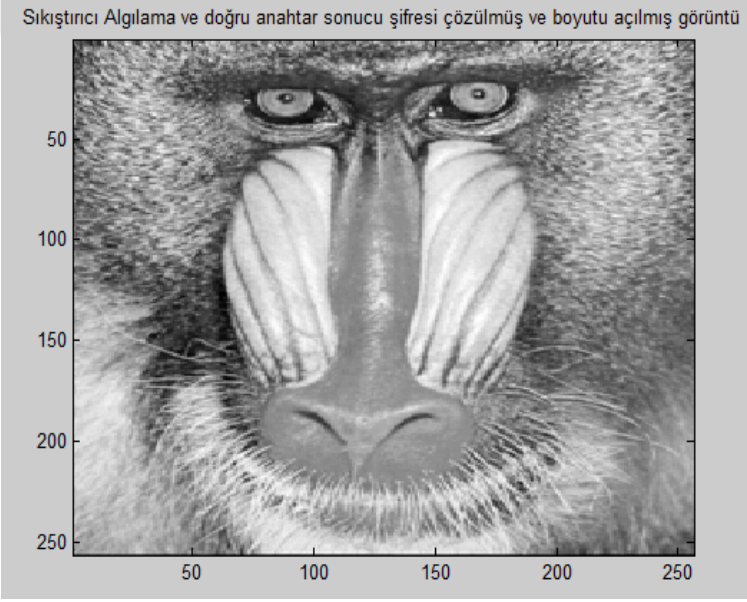
Şekil 4.27 Örnek 2 Orijinal görüntü



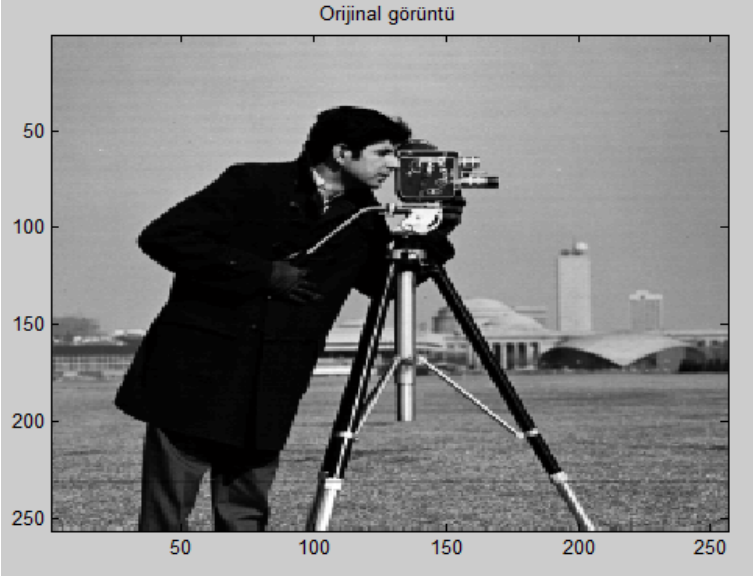
Şekil 4.28 Örnek 2 Sıkıştırıcı algılama ile şifrelenmiş ve sıkıştırılmış görüntü



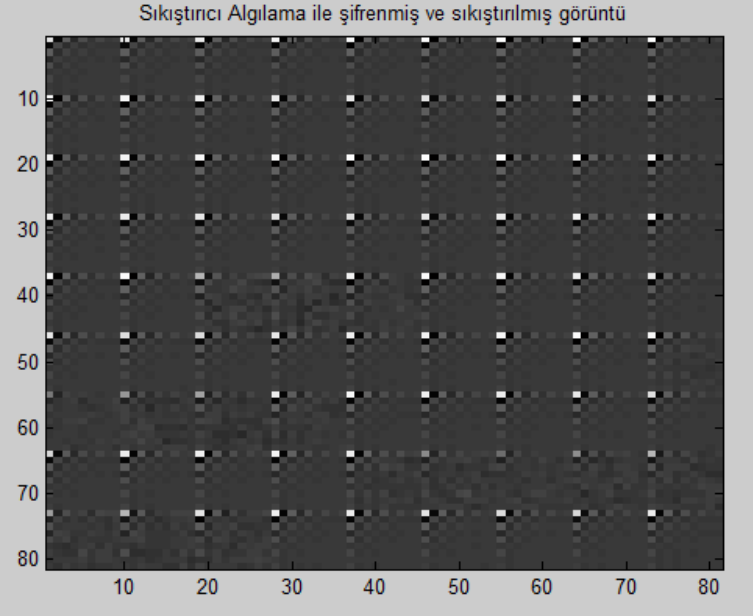
Şekil 4.29 Örnek 2 Sıkıştırıcı algılama ve yanlış anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü



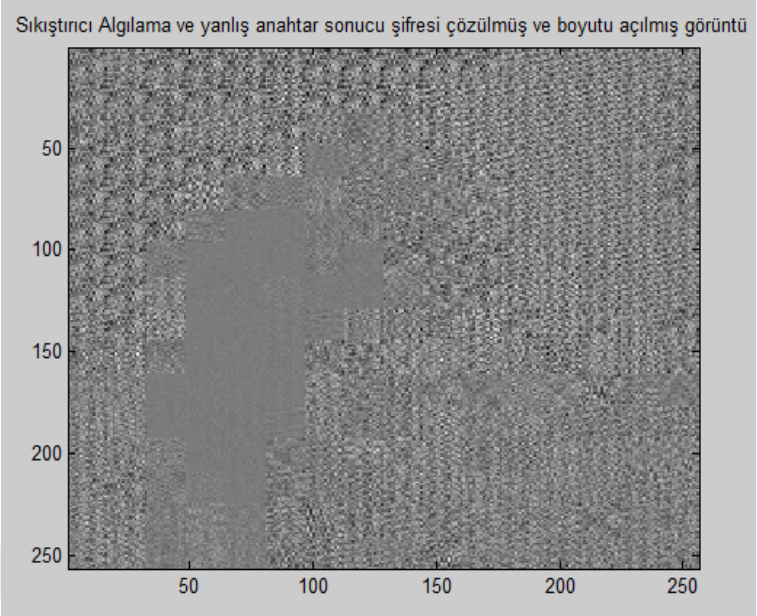
Şekil 4.30 Örnek 2 Sıkıştırıcı algılama ve doğru anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü



Şekil 4.31 Örnek 3 Orijinal görüntü



Şekil 4.32 Örnek 3 Sıkıştırıcı algılama ile şifrelenmiş ve sıkıştırılmış görüntü



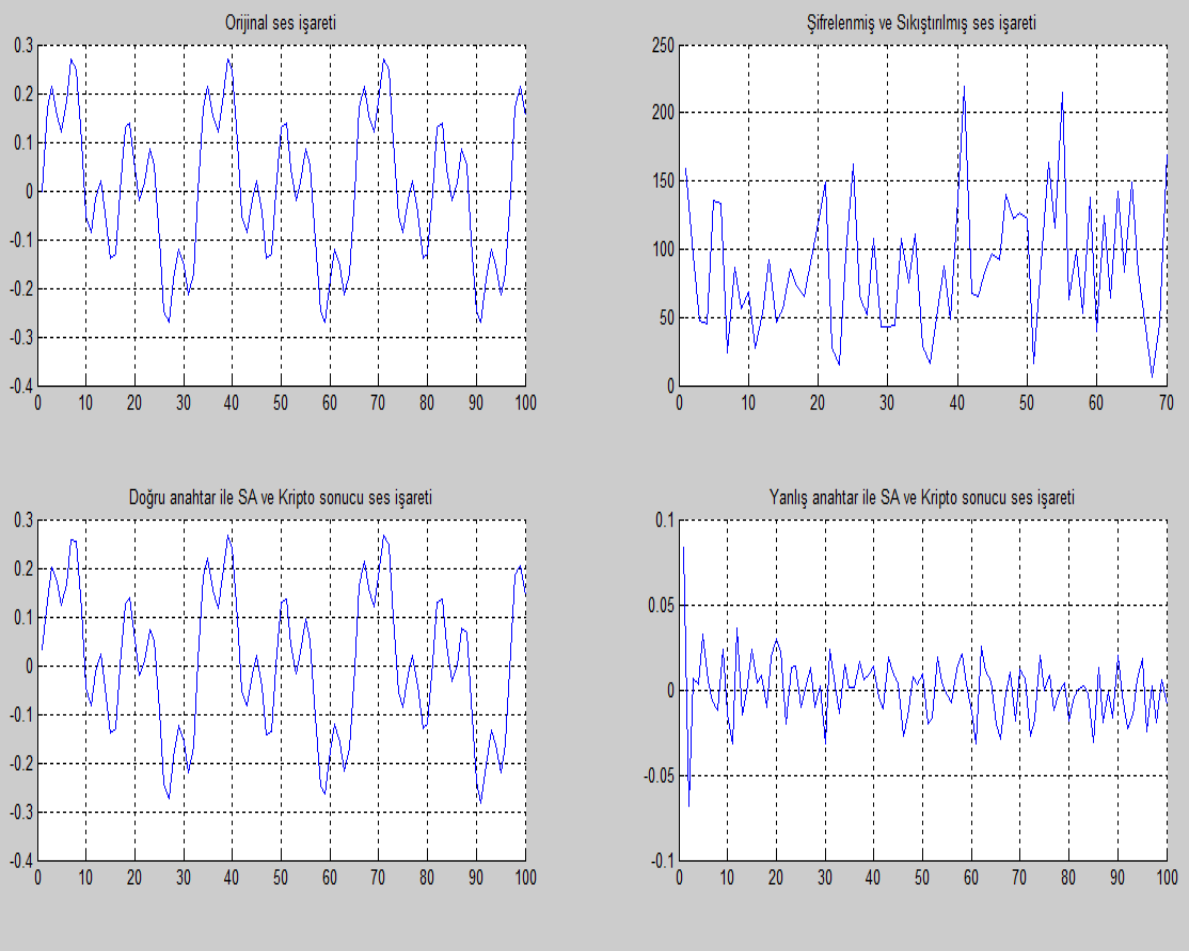
Şekil 4.33 Örnek 3 Sıkıştırıcı algılama ve yanlış anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü



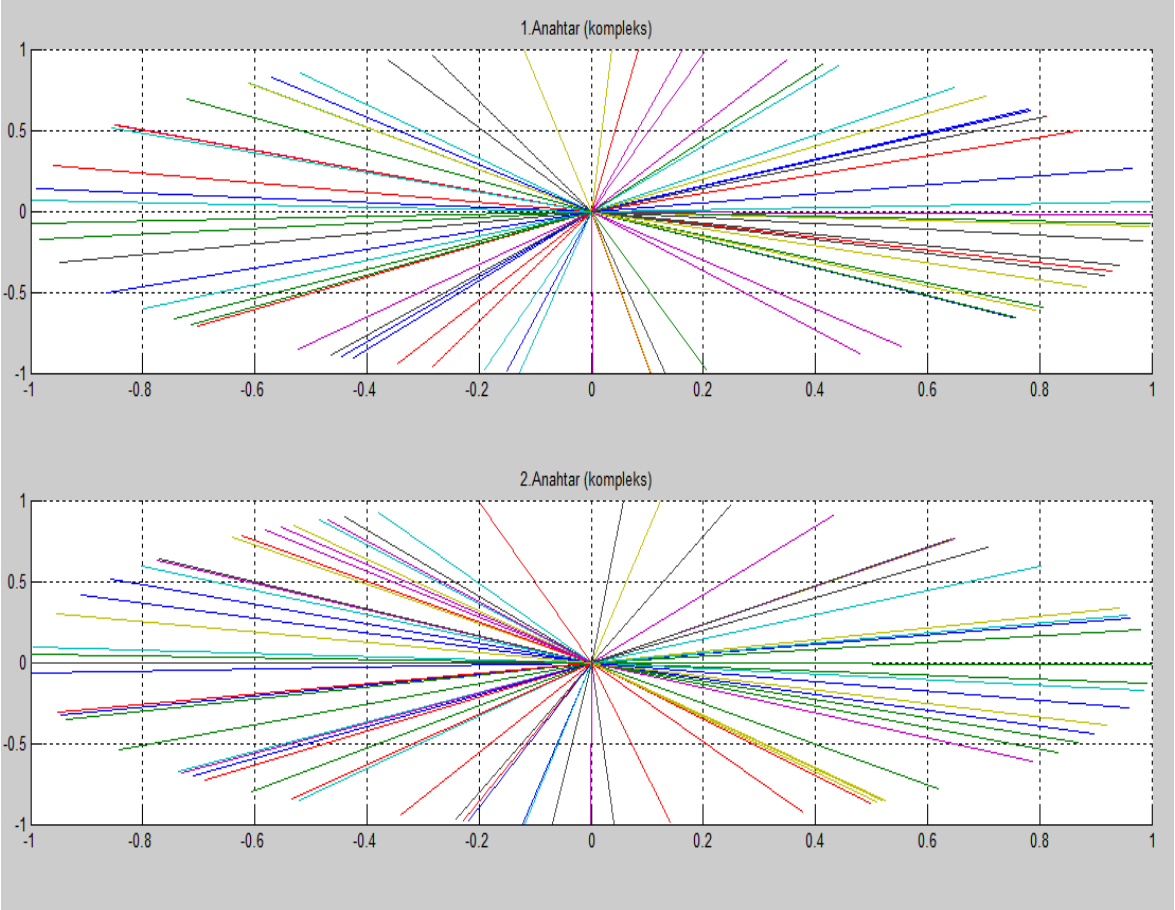
Şekil 4.34 Örnek 3 Sıkıştırıcı algılama ve doğru anahtar sonucu şifresi çözülmüş ve boyutu açılmış görüntü

4.5 Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) Optiksel Kriptografi (Ses İşareti)

Şimdiye kadar yaptığımız uygulamalarda kullanıcının dışarıdan girdiği karakterler/text ve resim üzerinde çalışılmıştır. Şimdi ise gerçek hayatta çok kullanılan ses işareti üzerinde uygulama amaçlanmıştır. Bunun için aşağıdaki örnek zil sesi üzerine çalışılmıştır. Ses giriş işareti için sıkıştırıcı algılama ve hibrit optiksel kriptografi uygulamasında simetrik kriptografi ve asimetrik kriptografi uygulaması elde edilmiştir. Aşağıda şekilde de görüleceği üzere orijinal ses işareti SA DEA sonucunda doğru anahtarla elde edilmektedir. Şifrelenmiş ve sıkıştırılmış işaret ise orijinal işarete oranla daha küçük boyutta ve çok karmaşık işaret çıkmaktadır.



Şekil 4.35 Sıkıştırıcı algılama ile hibrit optiksel kriptografi (ses işareti) uygulama sonucu, orjinal ve şifreli işaret, doğru ve yanlış anahtar sonucu



Şekil 4.36 Kompleks, rastgele ve dağıtılmış 2 faz anahtar

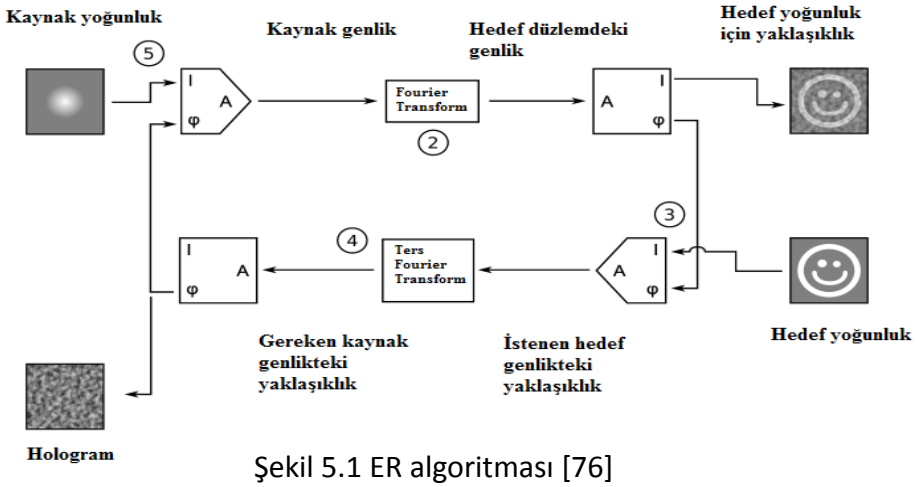
Yukarıdaki şekilde ise şifrelemede ve şifre çözmede kullanılan ve programda oluşturulan kompleks, rastgele ve 360 derece dağıtılmış 2 faz anahtarı gözükmektedir. Yukarıda şekilde eksenler, komleks faz anahtarların i ve j eksen çizimlerini göstermektedir.

BÖLÜM 5

GELİŞTİRDİĞİMİZ YÖNTEMLERİN KRİPTOANALİZ ÇALIŞMALARI

Kriptoanaliz için genellikle 2 atak (saldırı) kullanılmaktadır. Bunlar, faz geri elde yöntemleri ile ataklar ve dirac delta fonksiyon ile ataklardır. Bu ataklarda KPA yöntemi kullanılmaktadır. KPA (Known-plaintext attack) yani Bilinen Açık Metin Saldırısında, Kriptanalist sadece çeşitli düz metinlerin şifrelenmiş haline değil, bu mesajların açık metinlerine de erişebilmektedir. Yani Kriptoanalist, aynı anahtarla şifrenlenmiş bir kaç ciphertext/plaintext çiftine sahiptir.

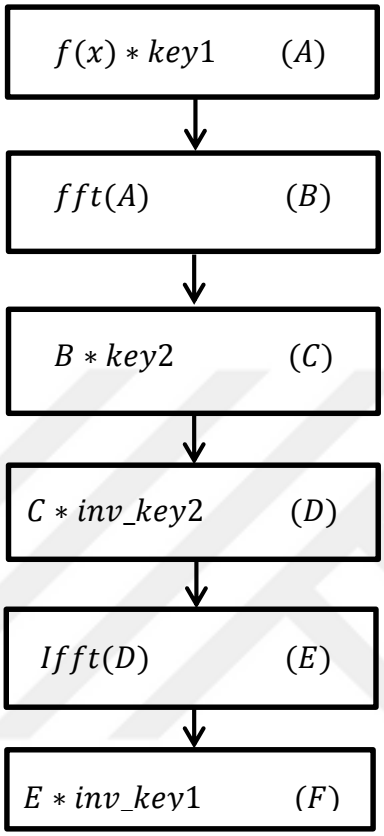
İlk önce faz geri elde yöntemleri ile saldırı incelenmiştir. Bu yöntemde genellikle ER algoritması kullanılmaktadır. ER algoritması (Error Reduction Algorithm) kısaca genliği verilen sinyalden fazın nasıl elde edileceğini açıklar. Faz geri elde etme işlemi işaretin fourier ve ters fourier dönüşümlerinin alınması ve bir sonraki adımda güncelleştirilmesi esasına dayanmaktadır. ER algoritması adımları aşağıda belirtilmiştir.
ER algoritmasında, amplitude (genlik) bilinmekte, phase (faz) ise kayıptır, faz tekrardan elde edilince işaret de elde edilmiş olur. Mercek ölçümlerinde yalnız genlik ölçülmektedir, faz ölçülememektedir, Fourier-ters Fourier dönüşümleri ile genlikten faz elde edilmektedir. Sensörler ile ışığın (görüntünün) yalnızca sinyal yoğunluğu (genliğinin karesi) ölçülmektedir, faz ölçülememektedir, bu sebeple mercek olmadan sadece genlik olmaktadır. [26-29].



Şekil 5.1 ER algoritması [76]

Zhang ve Wei tarafından [26] de belirtilen uygulamada, sıkıştırıcı algılama uygulaması ve asimetrik kriptografi kullanılmamıştır, ama uygulamamıza yakın olması için sıkıştırıcı algılama ve RSA uygulaması da ilave edilmiştir. Known-plaintext attack (KPA) olduğuna göre (plain-cipher çifti biliniyor olduğuna göre) genlik bilinip, faz geri elde etme teknikleri kullanarak (Error Reduction, ER) iki faz anahtarı elde edilmeye çalışılmaktadır. Aşağıdaki akış diyagramında [26], bu duruma göre, $f(x)$ ve C bilinmekte, $key1$ ve $key2$ bulunmaya çalışılmaktadır. Faz anahtarı olduğu için modülü 1 olmaktadır. Önerdiğimiz Sıkıştırıcı Algılama+Kriptografi uygulamalarında KPA saldırı yöntemi için öncelikle dikkat edildiğinde sıkıştırıcı algılama ile orijinal işaretten çok daha küçük boyutta şifrelenmiş işaret oluşmaktadır. Ayrıca kriptoanaliz makalelerinde [26-29] anahtar sadece faz $(e^{j\theta})$ ile oluşturulmuştur. İlk uygulamamızdaki simetrik şifrelemede de anahtarlar sadece faz $(e^{j\theta})$ ile oluşturulmuştur. Hem makalelerdeki uygulamaları hem de bu çalışmadaki Sıkıştırıcı Algılama+Kriptografinin karşılaştırmasında, anahtarın sadece faz $(e^{j\theta})$ ile değil de genlik ve faz çarpımı $(A * (e^{j\theta}))$ ile oluşturulursa faz geri elde teknikleri (örneğin; ER tekniği) de geçersiz olacaktır ve şifrelemenin sağlamlığı çok artacaktır. Bunun sonucunda önerdiğimiz algoritmalarda, hibrit uygulama için aynı anahtarları ve eşlenik anahtarları sadece faz $(e^{j\theta})$ ile değil de genlik ve faz çarpımı $(A * (e^{j\theta}))$ ile oluşturulmuştur. Anahtarların genlik ve faz çarpımı $(A * (e^{j\theta}))$ ile oluşumunda ER de şifre kırmak için yetersiz kalacaktır. Zhang ve Wei tarafından [26] de belirtilen çalışmada, aşağıdaki algoritmanın gerçekleştirilmesinde, orijinal işaretin 2 kez faz şifrelemesi ve Fourier dönüşümü kullanılmıştır, şifre çözmek için de anahtarların

eşlenikleri ve ters Fourier dönüşümü kullanılmıştır. Aşağıdaki çıktılarda da görüleceği üzere; sıkıştırıcı algılama kullanılmadığı için şifrelenmiş işaret ve orijinal işaret aynı uzunlukta olmaktadır, bu da şifrelemenin kırılmasını kolaylaştırmaktadır.



Şekil 5.2 Zhang ve Wei tarafından [26] önerilen kriptografi yönteminin akış diyagramı

Yukarıdaki algoritmayı hibrit kriptografi için uyguladığımızda aşağıdaki çıktılar alınmaktadır. İki ayrı giriş için uygulandığında, şekil 5.3 ve şekil 5.5 de giriş işaretin ASCII karşılığı, asimetrik kriptografi için RSA algoritması ile P ve Q asal sayıların üretilmesi, asimetrik E şifreleme anahtarı ve D şifre çözme anahtarının belirlenmesi, doğru ve yanlış anahtar sonucu hata değeri, süre ve alıcıda oluşan orijinal işaret gözükmektedir. Şekil 5.4 ve şekil 5.6 da ise, orijinal işaret, şifreli işaret, doğru anahtar ve yanlış anahtar sonuçları gösterilmektedir. Burada bakıldığında sıkıştırıcı algılama kullanılmadığı için, orijinal işaret ve şifrelenmiş işaret aynı boyutta olmaktadır, bu durumda şifrelemenin daha zayıf olduğunu göstermektedir. Aşağıdaki şekillerde yatay eksen işaretin uzunluğunu, dikey eksen ise işaretin ASCII karşılığını göstermektedir.

```
Kesme işareti arasına açık text giriniz = 'YILDIZ TEKNIK FBE'

Açık Text = YILDIZ TEKNIK FBE

Açık Text - (ASCII olarak) ;
89 73 76 68 73 90 32 84 69 75 78 73 75 32 70 66 69

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
==========================================================
RSA başlatıldı:

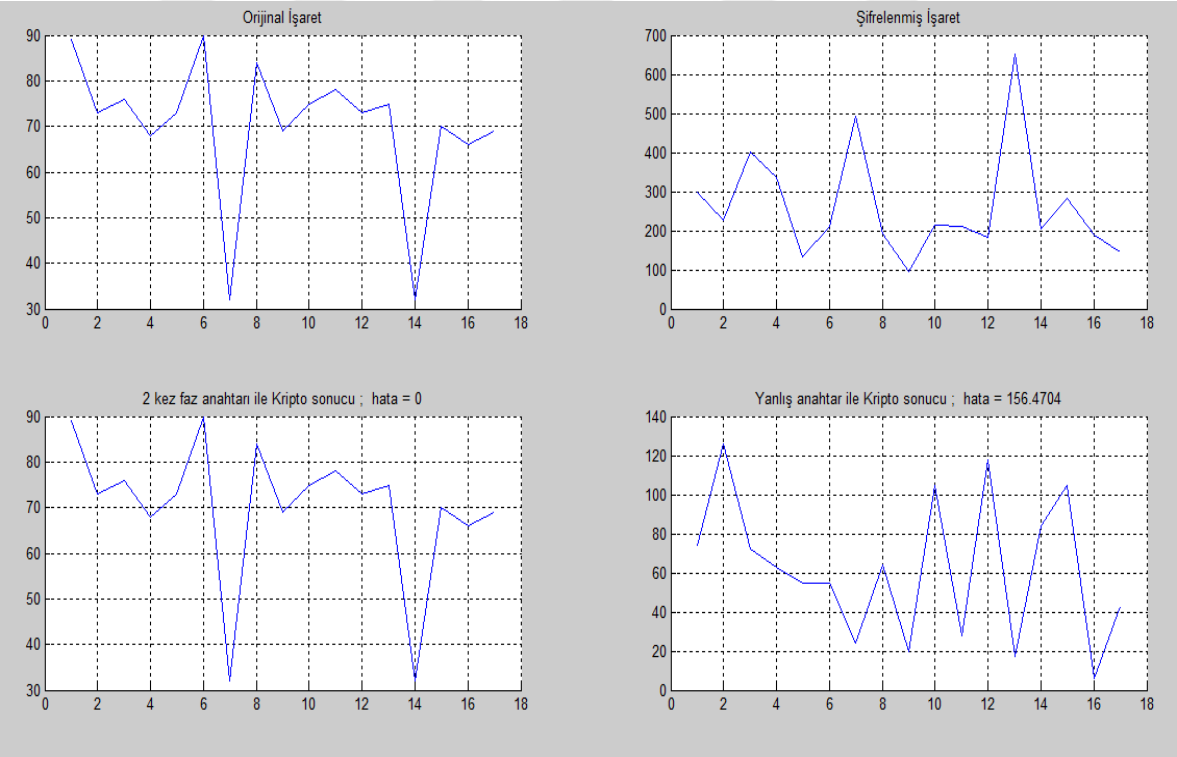
        P=173 - ilk asal değer (özel)
        Q=443 - ikinci asal değer (özel)
        N=76639 - modül (N) (genel)
        phi(N)=76024 - Euler fonksiyonunun degeri
        E=3 - kriptolama için genel anahtar (E) (genel)
        D=50683 - gizli kripto çözme anahtari (D) (özel)
==========================================================
Doğru anahtar ile hata = 0.00e+00

Yanlış anahtar ile hata = 1.56e+02

SA ile Kriptosu açılmış Text = YILDIZ TEKNIK FBE

Elapsed time is 7.494175 seconds.
```

Şekil 5.3 Zhang ve Wei tarafından [26] önerilen kriptografi için oluşturulan programın sonucu



Şekil 5.4 Zhang ve Wei tarafından [26] önerilen kriptografi için oluşturulan programın uygulama çıktıları

```
Kesme işareti arasına açık text giriniz = 'Doktora Çalışması (2015)'

Açık Text = Doktora Çalışması (2015)

Açık Text - (ASCII olarak) ;
68 111 107 116 111 114 97 32 199 97 108 305 351 109 97 115 305 32 40 50 48 49 53 41

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
=====================================================
RSA başlatıldı:

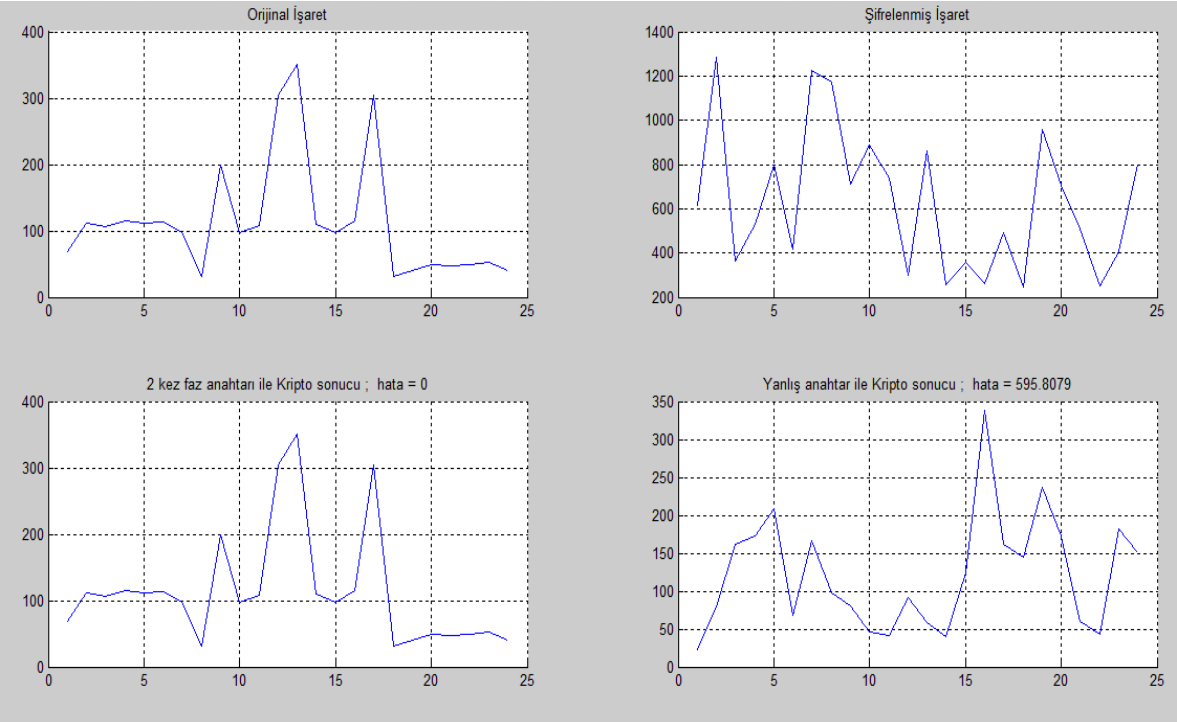
        P=149 - ilk asal değer (özel)
        Q=331 - ikinci asal değer (özel)
        N=49319 - modül (N) (genel)
        phi(N)=48840 - Euler fonksiyonunun degeri
        E=137 - kriptolama için genel anahtar (E) (genel)
        D=713 - gizli kripto çözme anahtari (D) (özel)
=====================================================
Doğru anahtar ile hata = 0.00e+00

Yanlış anahtar ile hata = 5.96e+02

SA ile Kriptosu açılmış Text = Doktora Çalışması (2015)

Elapsed time is 2.358814 seconds.
```

Şekil 5.5 Zhang ve Wei tarafından [26] önerilen kriptografi için oluşturulan programın sonucu



Şekil 5.6 Zhang ve Wei tarafından [26] önerilen kriptografi için oluşturulan programın uygulama çıktıları

Önerdiğimiz Sıkıştırıcı Algılama+Hibrit Optiksel Kriptografi uygulamasındaki simetrik, asimetrik ve hibrit şifrelemede oluşturduğumuz anahtarların ve şifrelemenin daha zor olması, faz geri elde yöntemleri ile çözülememesi için anahtar oluşumunda sadece faz $(e^{j\theta})$ ile değil de genlik ve faz çarpımı $(A * (e^{j\theta}))$ ile gerçekleştirilmiştir. Aşağıdaki şekilde algoritma ve anahtar oluşumu adımları gösterilmektedir. Bu durum sayesinde önerdiğimiz algoritma faz geri elde etme tekniği ile kırılamayacaktır. Program çıktıları da aşağıda yer almaktadır.

Kriptografinin güçlü olması için optiksel $p1$ ve $p2$ faz+genlik anahtarları kompleks rastgele sayılardan oluşmaktadır. Çift rastgele optiksel faz+genlik şifrelemesi ile kompleks sayıların kullanılması iyi saçılma sağlamaktadır. Önerdiğimiz algoritmada Asimetrik kriptografinin uygulanabilmesi için; optiksel iki rastgale faz+genlik kompleks anahtarın reel ve imajiner kısımları ayrılmak suretiyle ve RSA ile asimetrik olarak simetrik anahtarlar alıcıya gönderilmiştir, alıcı da ise bu gelen anahtarları kullanarak simetrik olarak sıkıştırılmış algılama SA DEA'nın çözümü ile şifre çözümü ve orijinal işaret elde edilmektedir. Uygulamamızda güvenliği daha da artırmak için; Sıkıştırılmış Algılama + Optiksel Hibrit Kriptografi (eşlenik anahtarlar ve fft dönüşümü kullanarak) uygulaması yapılmıştır.

```
Kesme işareti arasına açık text giriniz = 'YILDIZ TEKNIK FBE'

Açık Text = YILDIZ TEKNIK FBE

Açık Text - (ASCII olarak) ;
89 73 76 68 73 90 32 84 69 75 78 73 75 32 70 66 69

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
==================================================
RSA başlatıldı:

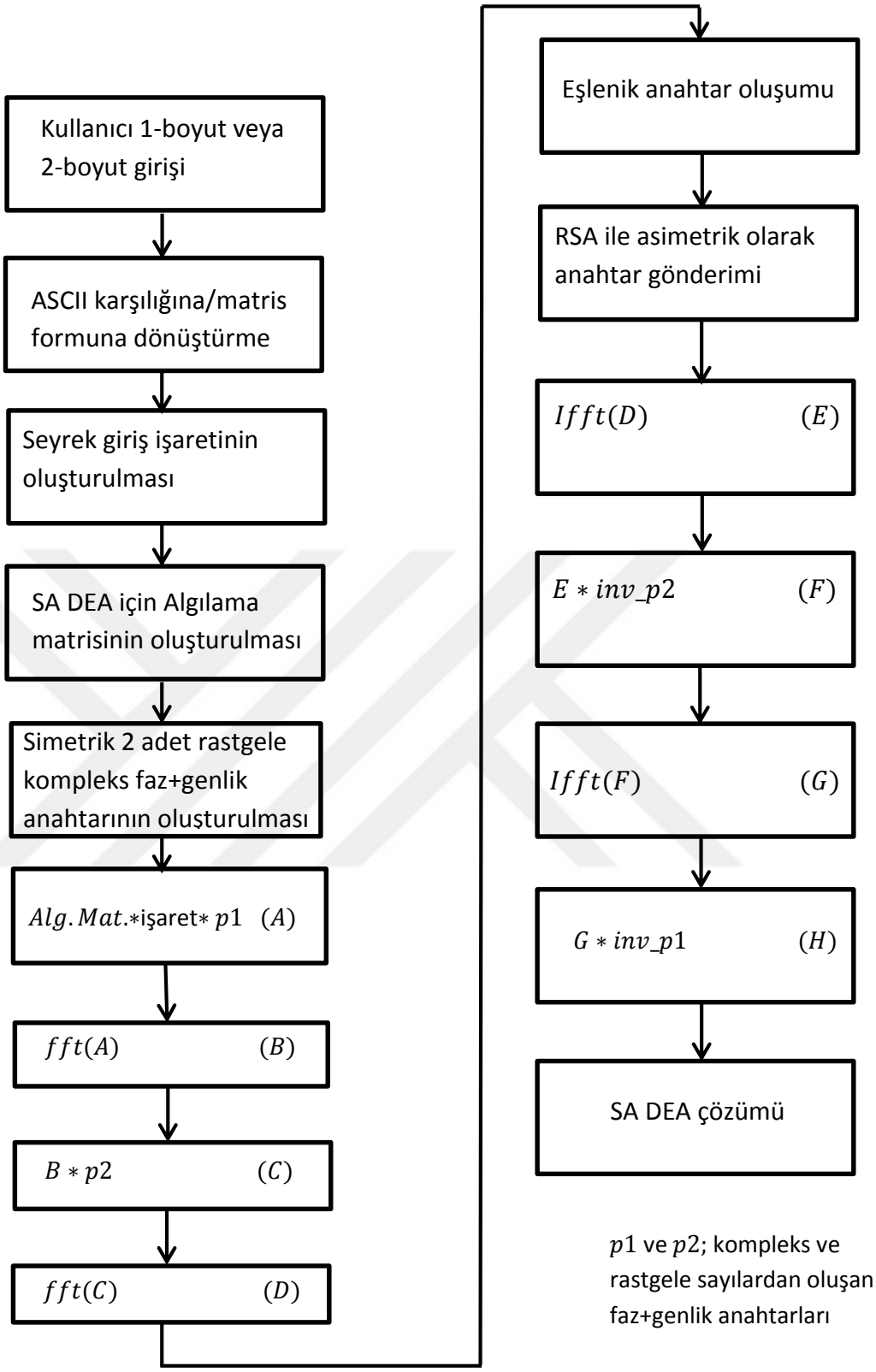
        P=251 - ilk asal değer (özel)
        Q=379 - ikinci asal değer (özel)
        N=95129 - modül (N) (genel)
        phi(N)=94500 - Euler fonksiyonunun degeri
        E=331 - kriptolama için genel anahtar (E) (genel)
        D=571 - gizli kripto çözme anahtari (D) (özel)
==================================================
Doğru anahtar ile hata = 2.34e-15

Yanlış anahtar ile hata = 2.98e+02

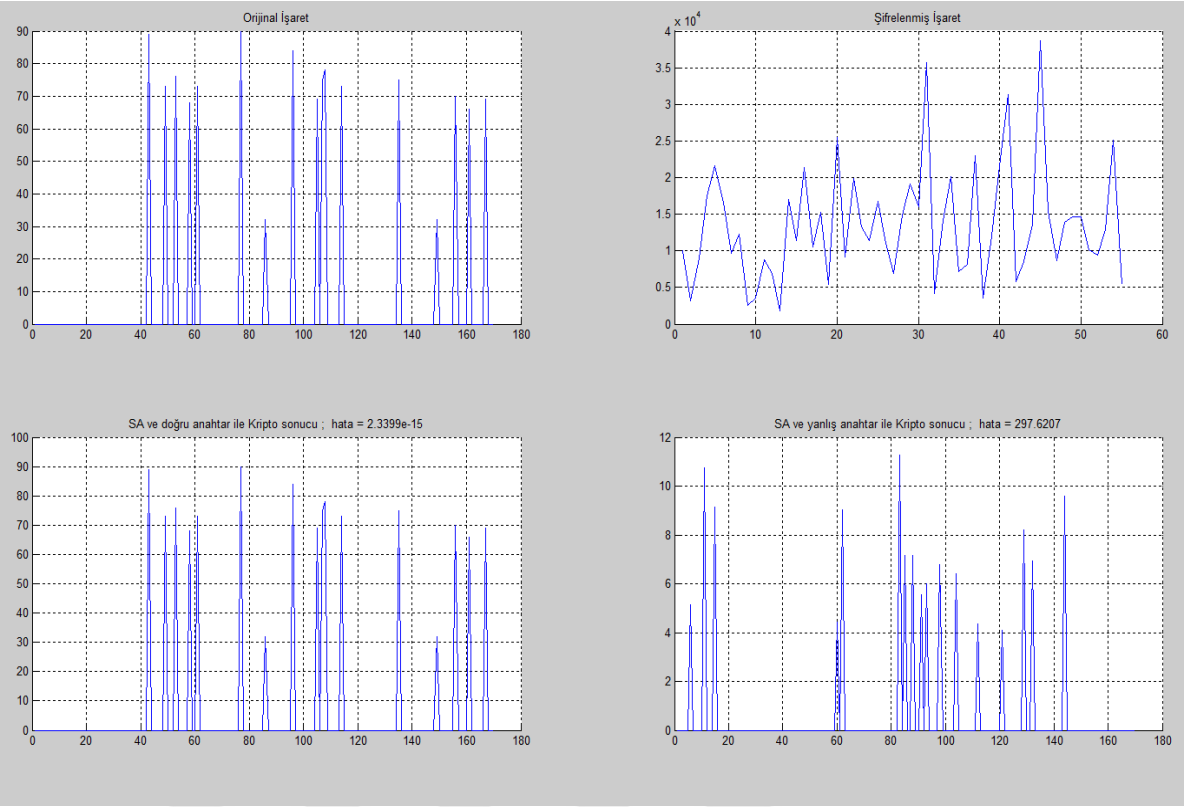
SA ile Kriptosu açılmış Text = YILDIZ TEKNIK FBE

Elapsed time is 10.065600 seconds.
```

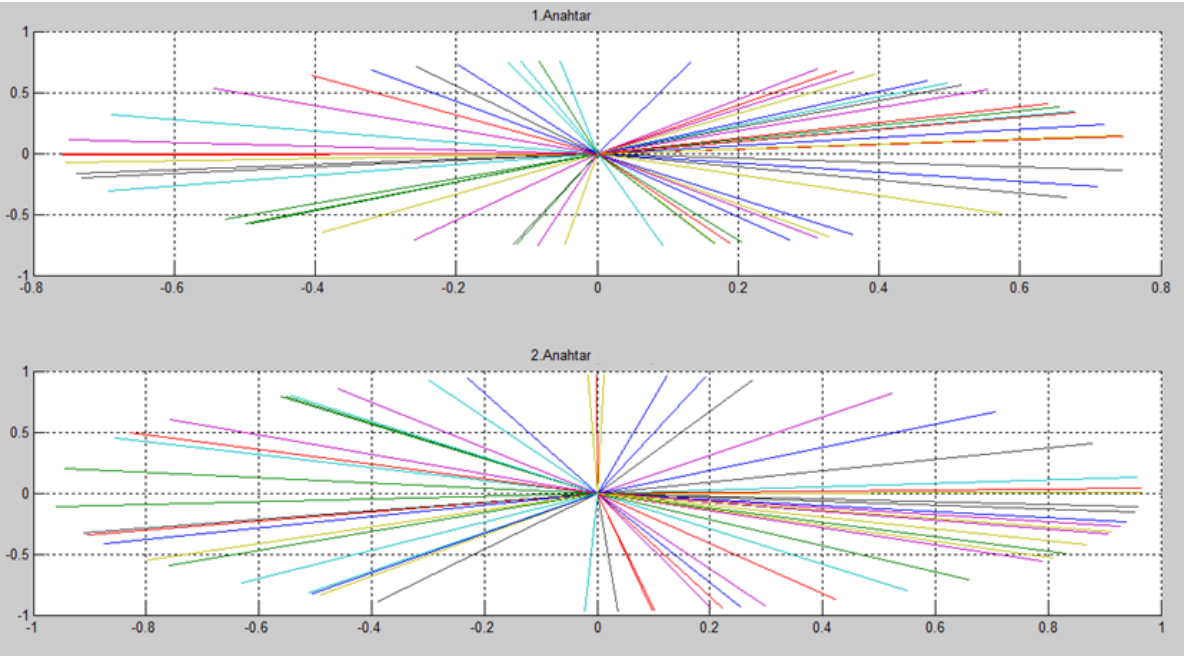
Şekil 5.7 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın çıktısı



Şekil 5.8 Kriptoanaliz için önerilen yöntemin akış diyagramı



Şekil 5.9 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın orijinal işaret, şifrelenmiş ve sıkıştırılmış işaret, doğru anahtar ve yanlış anahtar sonuçları



Şekil 5.10 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programında oluşturulan 2 adet rastgele ve kompleks faz+genlik anahtarları

```
Kesme işareti arasına açık text giriniz =  'DOKTORA ÇALIŞMASI (2015)'

Açık Text = DOKTORA ÇALIŞMASI (2015)

Açık Text - (ASCII olarak) ;
68 79 75 84 79 82 65 32 199 65 76 73 350 77 65 83 73 32 40 50 48 49 53 41

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
=====================================================
RSA başlatıldı:

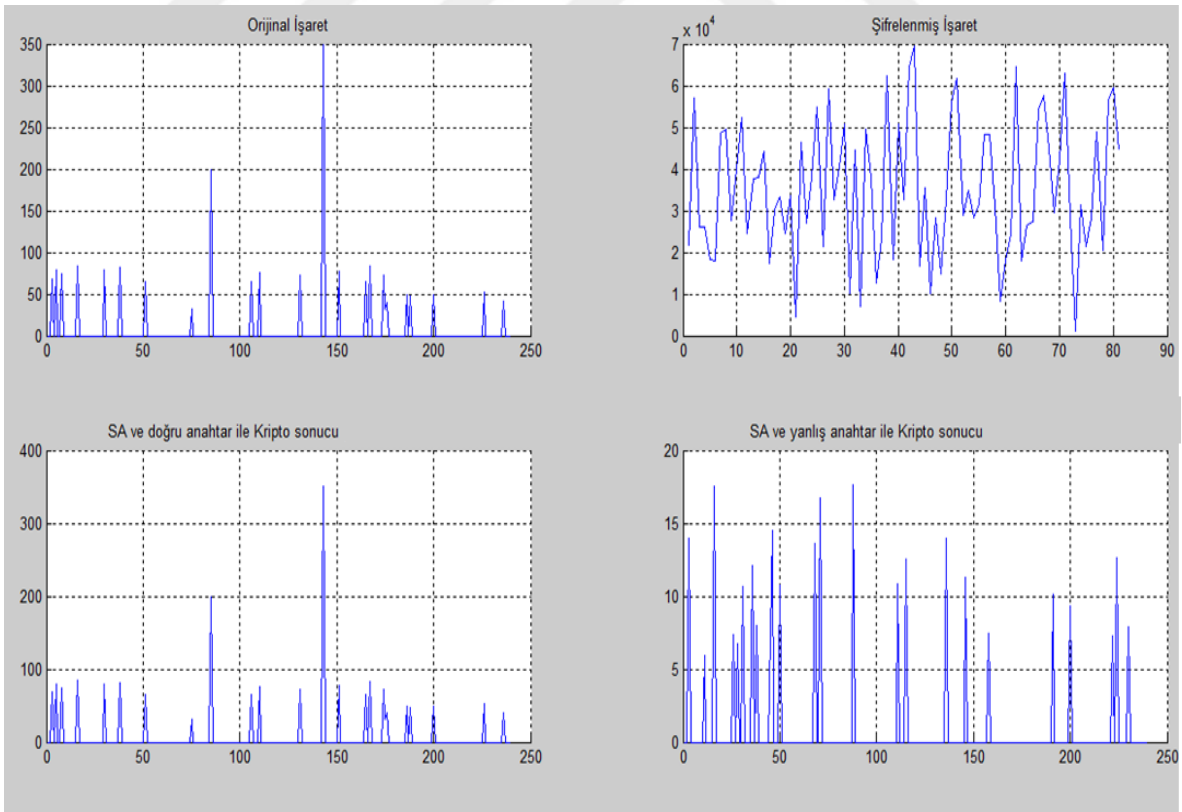
        P=113 - ilk asal değer (özel)
        Q=809 - ikinci asal değer (özel)
        N=91417 - modül (N) (genel)
        phi(N)=90496 - Euler fonksiyonunun degeri
        E=3 - kriptolama için genel anahtar (E) (genel)
        D=60331 - gizli kripto çözme anahtari (D) (özel)
=====================================================
Doğru anahtar ile hata = 2.44e-15

Yanlış anahtar ile hata = 5.07e+02

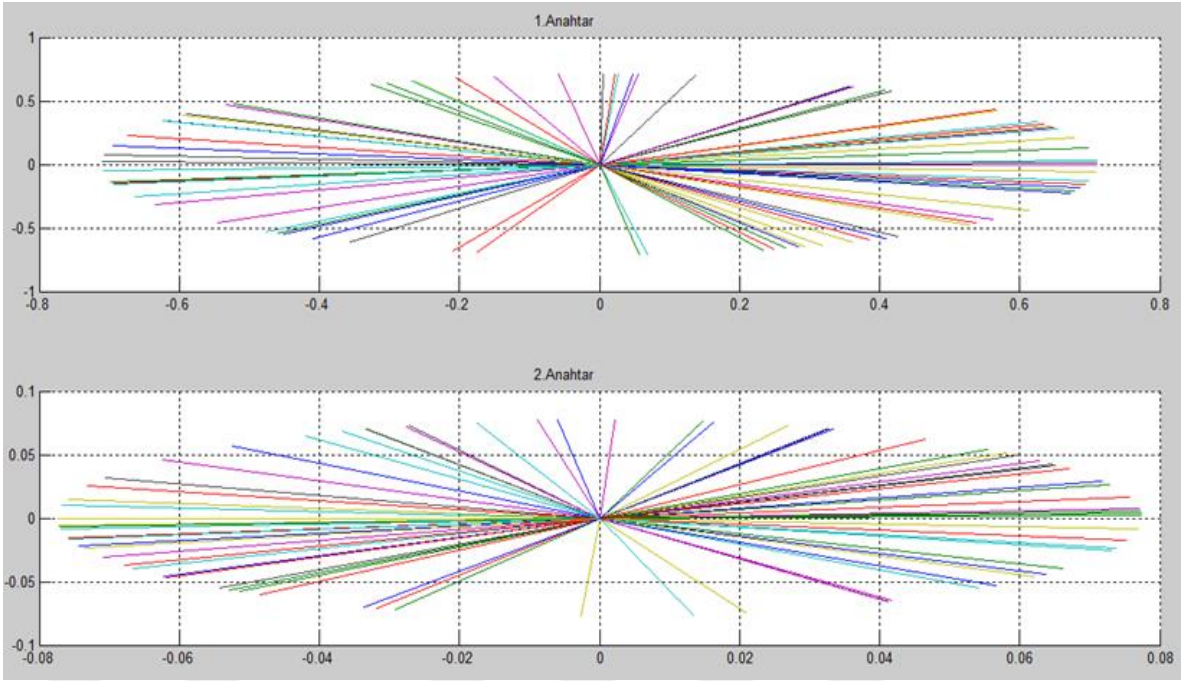
SA ile Kriptosu açılmış Text = DOKTORA ÇALIŞMASI (2015)

Elapsed time is 1.866274 seconds.
```

Şekil 5.11 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın çıktısı



Şekil 5.12 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programın orijinal işaret, şifrelenmiş ve sıkıştırılmış işaret, doğru anahtar ve yanlış anahtar sonuçları



Şekil 5.13 Faz+genlik anahtarları ile önerdiğimiz sıkıştırıcı algılama ve hibrit kriptografi programında oluşturulan 2 adet rastgele ve kompleks faz+genlik anahtarları

```
Kesme işareti arasına açık text giriniz = 'YILDIZ TEKNIK FBE'

Açık Text = YILDIZ TEKNIK FBE

Açık Text - (ASCII olarak) ;
89 73 76 68 73 90 32 84 69 75 78 73 75 32 70 66 69

P asal sayısının üretilmesi:
Q asal sayısının üretilmesi:
(E ,D) çiftinin üretilmesi:
=====================================================
RSA başlatıldı:

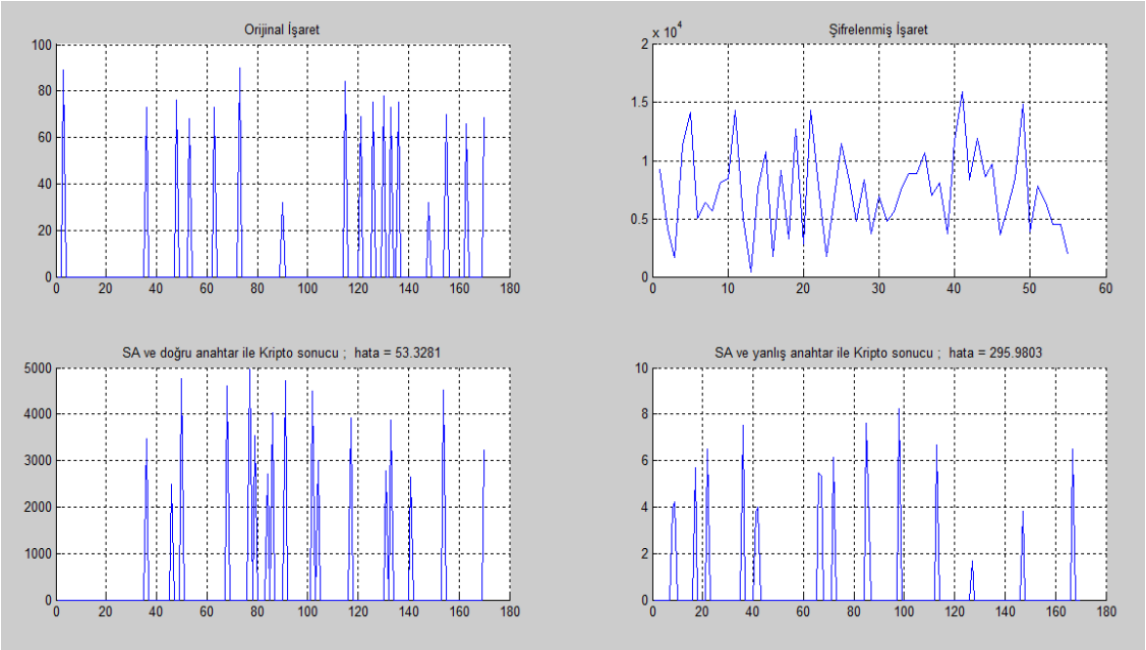
        P=647 - ilk asal değer (özel)
        Q=101 - ikinci asal değer (özel)
        N=65347 - modül (N) (genel)
        phi(N)=64600 - Euler fonksiyonunun degeri
        E=3 - kriptolama için genel anahtar (E) (genel)
        D=43067 - gizli kripto çözme anahtari (D) (özel)
=====================================================
Doğru anahtar ile hata = 5.33e+01

Yanlış anahtar ile hata = 2.96e+02

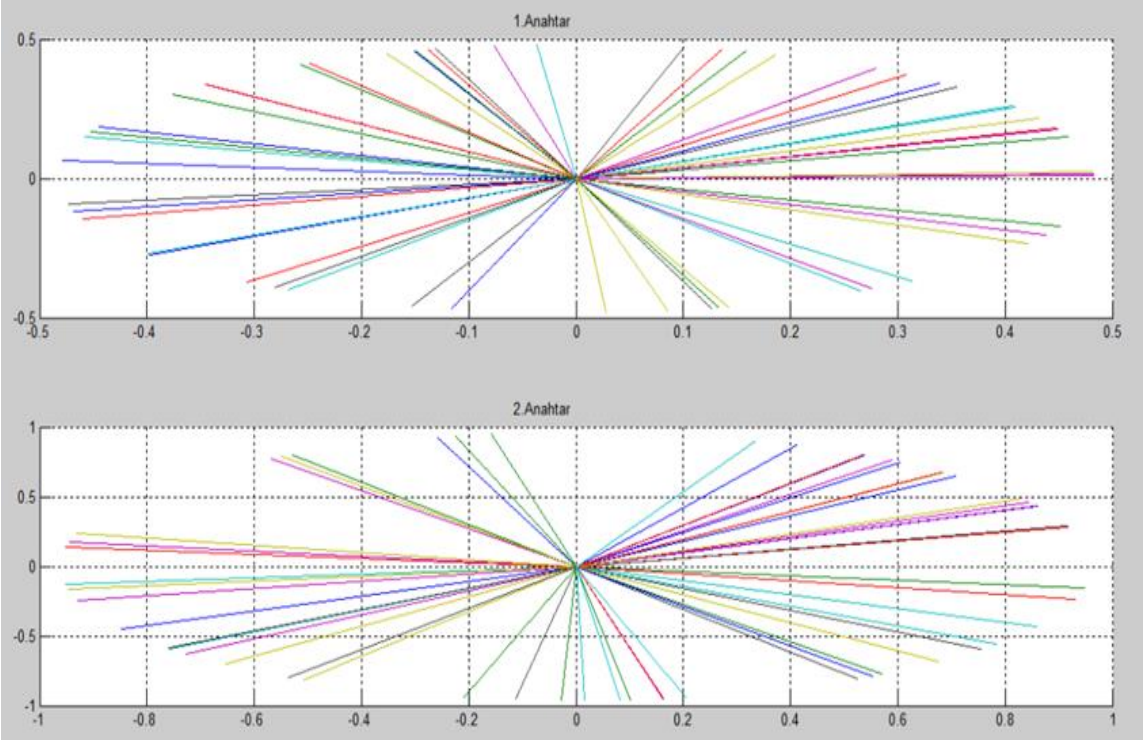
SA ile Kriptosu açılmış Text = □□□□□□□□□□□□□□□□□

Elapsed time is 2.261653 seconds.
```

Şekil 5.14 Faz+genlik anahtarındaki sadece tek değerin hatalı olması sonucu



Şekil 5.15 Faz+genlik anahtarındaki sadece tek değerin hatalı olması ile sıkıştırıcı algılama ve hibrit kriptografi programın orijinal işaret, şifrelenmiş ve sıkıştırılmış işaret, doğru anahtar ve yanlış anahtar sonuçları



Şekil 5.16 Faz+genlik anahtarındaki sadece tek değerin hatalı olması ile sıkıştırıcı algılama ve hibrit kriptografi programında oluşturulan 2 adet rastgele ve kompleks faz+genlik anahtarları

Yukarıda şekil 5.9 ve şekil 5.12 de yatay eksen işaretin uzunluğunu, dikey eksen ise işaretin ASCII karşılığını göstermektedir. Şekil 5.10 ve şekil 5.13 deki eksenler, kompleks faz anahtarların i ve j eksen çizimlerini göstermektedir. Kriptoanaliz çalışmaları için yukarıda belirtilen şekil 5.8 deki algoritmaya göre oluşturduğumuz program çıktılarına baktığımızda; şekil 5.7 ve şekil 5.11 de açık işaret, hibrit kriptografide kullanacağımız asimetrik RSA algoritması için P ve Q asal sayılar, E şifreleme anahtarı ve D şifre çözme anahtarı, doğru anahtarlar ve yanlış anahtar ile hata, işlem süresi ve alıcıda şifrelenmiş işaretin açılmış hali gösterilmiştir. Şekil 5.9 ve şekil 5.12 de orijinal işaret, faz+genlik anahtarları ile oluşturduğumuz şifreli işaret, doğru ve yanlış anahtar sonuçları bulunmaktadır. Şekil 5.10 ve şekil 5.13 de ise daha önceki uygulamalarda farklı olarak sadece faz anahtarı değil, faz+genlik anahtar sonucu bulunmaktadır. Dikkat edilirse daha önceki anahtarlar sadece faz anahtarı olduğu için modülü 1 olmaktadır, ancak buradaki anahtarlar faz+genlik şeklinde oluşturduğu için modülü 1 olmamaktadır, rastgele sayılar olarak gözükmektedir. Bu da faz geri yöntemi ile şifre kırılamayacağını göstermektedir. Şekil 5.14, şekil 5.15 ve şekil 5.16'da ise faz+genlik anahtarındaki sadece tek değerin bile farklı olması durumunda şifreli ve sıkıştırılmış işaretin geri elde edilemeyeceğini göstermektedir. Bu durum gösteriyor ki, atak durumunda çift oluşturulan M ve N boyutundaki faz+genlik anahtarlarının tüm değerlerinin doğru olarak tahmin edilmesi gerekiyor ki, doğru işaret geri elde edilebilsin, bu durum kaba kuvvet saldırısını çok zorlaştırmaktadır.

2-f optik işleme için aşağıdaki şekilde faz anahtarı ve faz+genlik kriptoanaliz uygulamasının açıklaması yapılmıştır.

$$C = Y.fft(P.X) \tag{5.1}$$

$$P.X = inv_fft(C \div Y) \tag{5.2}$$

$$P = |P.X| = |inv_fft(C \div Y)| \tag{5.3}$$

C: şifreli işaret, X: ilk anahtar, Y: ikinci anahtar, P: orijinal açık işaret

Yukarıdaki işlemlerde de görüleceği üzere, ilk işlemde faz anahtarları ile şifreleme işlemi yapılmaktadır. İkinci işlemde ise ters fourier ile şifreli işaret elde edilmeye çalışılmaktadır. Üçüncü işlemde ise her iki tarafın modülü alınmakta, anahtarlar sadece faz anahtarı olduğunda ilk anahtarın modülü alındığında 1 olacağı için ihmal

edilmektedir. KPA saldırısı olduğu için düz metin-şifreli metin çifti de bilindiğinde ikinci anahtar bulunabilir, akabinde de birinci anahtar bulunabilir, ama faz+genlik anahtarında bu yöntem işe yaramayacaktır.

4-f optik işleme için ise aşağıdaki şekilde faz anahtarı ve faz+genlik kriptoanaliz uygulamasının açıklaması yapılmıştır.

C: şifreli işaret, $p1$: ilk anahtar, $p2$: ikinci anahtar, P: orijinal açık işaret
P-C çifti biliniyor varsayılıyor (KPA atağı için)

$$fft(fft(p1.X).p2) = C \tag{5.4}$$

$$fft(p1.X).p2 = inv_fft(C) \tag{5.5}$$

$$fft(p1.X) = inv_fft(C)/p2 \tag{5.6}$$

$$p1.X = inv_fft(inv_fft(C)/p2) \tag{5.7}$$

her iki tarafın modülü alınırsa;

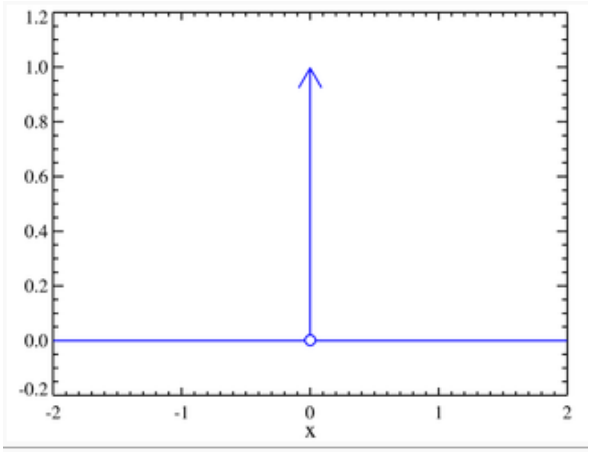
$p1$ ve $p2$ sadece faz ise $p1$ ihmal edilir.

$|X| = |inv_fft(inv_fft(C)/p2)|$ buradan $p2$ bulunur. Sonra da $p1$ bulunur.

Yukarıdaki işlemlerde de görüleceği üzere, aynı şekilde ikinci anahtar bulunabilir, akabinde de birinci anahtar bulunabilir, ama faz+genlik anahtarında bu yöntem işe yaramayacaktır.

Diğer bir saldırı yönteminde ise dirac delta fonksiyon kullanılır.

Adını Paul Dirac'tan alan Dirac delta fonksiyonu tek boyutta şekil 5.17'de tanımlanmıştır. Bu gösterime uyacak bütün matematik temsillerine delta fonksiyonu veya delta fonksiyonunun temsili denir. Delta fonksiyonu n boyuta genellenebilir. Burada x ve x_0 n boyutlu vektörlerdir.



Şekil 5. 17 Dirac delta fonksiyonu

$$\delta\,(x - x_0) = \begin{cases} 1 & x = x_0 \\ 0 & x \neq x_0 \end{cases} \tag{5.8}$$

Örnek uygulama:

$$\begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 3 \\ 4 \end{bmatrix} \qquad \begin{bmatrix} 1 & 3 \\ 2 & 4 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix}$$

Örnek uygulamada da görüleceği üzere girişe dirac delta fonksiyonu uygulandığında çıkış bilinebilmektedir. Bu kriptoanaliz, bizim uygulamalarımız için düşünüldüğünde yine KPA atağında P-C çifti bulunabilmektedir; ilk uygulamamızda $b = AAt * x$ olduğunda b şifreli işaret ve x giriş işareti bulunabilir ise bütün faz+genlik anahtarların sıkıştırıcı algılama olarak işlem gördüğü toplam anahtar AAt de bulunabilir. Bu yüzden bu yöntem ile kırılma olmaması için anahtar ve şifreli işareti aşağıdaki şekilde ayrıştırılarak değiştirilmiştir.

$$y = AA * x;\ AAA = y' * p1;\ BBB = fft(AAA);\ CCC = BBB * p2; \tag{5.9}$$

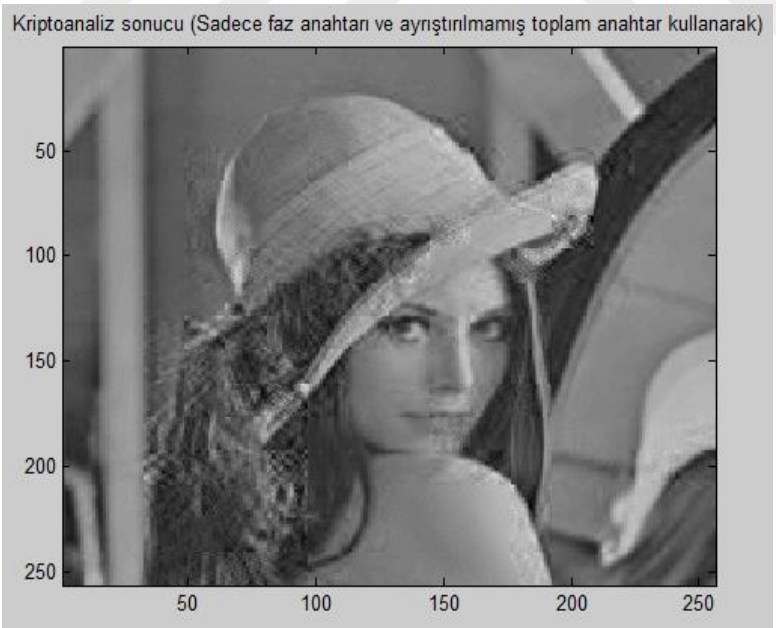
$$DDD = fft(CCC); \tag{5.10}$$

Burada y:ölçüm vektörü, AA:algılama matrisi, $p1$:birinci faz+genlik anahtarı, $p2$:ikinci faz+genlik anahtarı, DDD:şifreli işaret olmaktadır. KPA atağında, dirac delta atağı ile x ve DDD bulunabilir, ama aradaki işlemler bilinemediği için $p1$ ve $p2$ faz+genlik anahtarları bulunamaz. Bu yöntem için de kriptoanaliz başarı ile sağlanmaktadır. Oluşturduğumuz kriptoanaliz programının sonuçları aşağıda belirtilmiştir. Bu

sonuçlardan da görüleceği üzere deneysel olarak önerdiğimiz uygulama, faz geri elde yöntemleri ile ataklara ve dirac delta fonksiyon ile ataklara karşı dayanıklıdır.

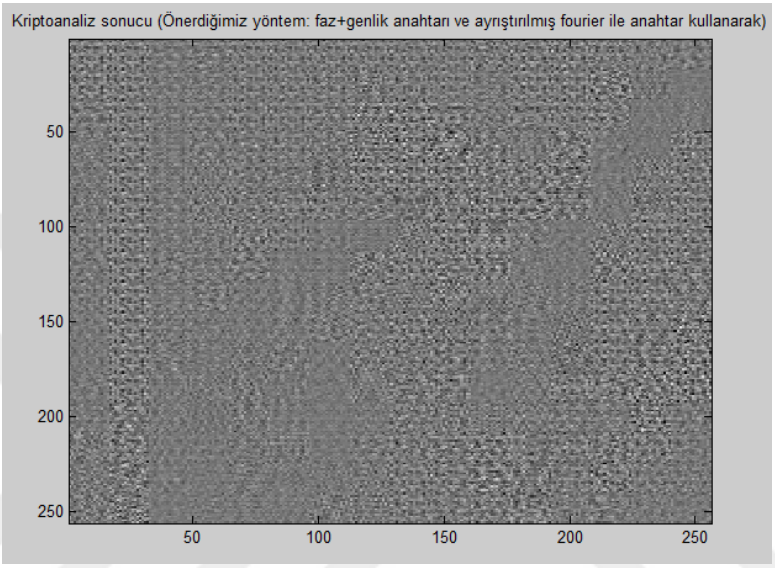


Şekil 5.18 Kriptoanaliz için girişe uygulanan orijinal görüntü



Şekil 5. 19 Kriptoanaliz sonucu (Sadece faz anahtarı ve ayrıştırılmamış toplam anahtar kullanarak)

Kriptoanaliz için yukarıdaki orijinal işareti, önerdiğimiz yönteme uygulandığında şekil 5.19'da sadece faz anahtarı ve ayrıştırılmamış toplam anahtar kullanılırsa saldırı (atak) sonucu şifreleme kırılmakta ve orijinal işaret elde edilebilmektedir. Fakat önerdiğimiz yöntem kullanıldığında Şekil 5.20'de hem faz geri elde yöntemi hem de dirac delta fonksiyonu yöntemi ataklarına karşı kırılamamaktadır. Saldırı (atak) sonucu gürültü oluşmakta ve orijinal görüntü elde edilememektedir.



Şekil 5. 20 Kriptoanaliz sonucu (Önerdiğimiz yöntem, faz+genlik anahtarı ve ayrıştırılmış fourier ile anahtar kullanarak)

BÖLÜM 6

SONUÇLAR VE TARTIŞMA

Bu çalışma ile sıkıştırıcı algılama dik eşleştirme arayış yöntemi ve çift rastgale faz+genlik anahtarları ile simetrik, asimetrik ve hibrit optiksel kriptografi uygulaması gerçekleştirilmiştir. Bu çalışmadaki 6 uygulamamız git gide zorlaşmakta ve çok daha iyi sonuç vermektedir.

Yaptığımız çalışmalara baktığımızda ana başlık sıkıştırıcı algılama+kriptografi olarak verilebilir. İlk uygulamada olarak sıkıştırıcı algılama+simetrik şifreleme gerçekleştirilmiştir. Daha sonra ise, ikinci uygulamamızda, Sıkıştırıcı Algılama ile Hibrit Kriptografi uygulaması gerçekleştirilmiştir. Bu uygulamada simetrik kriptografideki anahtarların karşıya (alıcıya) daha güvenli ulaşması için asimetrik kriptografi ile taşıma işlemi yapılmıştır. Asimetrik kriptografi olarak RSA algoritması kullanılmıştır. Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) Kriptografi (Aynı Anahtarları Kullanarak) uygulaması yapılmıştır. Diğer uygulamada üçüncü olarak ise, Sıkıştırıcı Algılama+Hibrit Kriptografinin (aynı anahtarları kullanarak) üzerine, güvenliği daha da artırmak için Sıkıştırıcı Algılama+Hibrit Optiksel Kriptografi (eşlenik anahtarlar ve fft dönüşümü kullanarak) uygulaması gerçekleştirilmiştir. Buradaki asıl amaç, hibrit kriptografide şifrelemede ve şifre çözmede kullanılan aynı anahtarlar yerine, şifrelemede kullanılan anahtarların ve Fourier dönüşümlerin eşleniklerini ve ters Fourier dönüşümlerini ve optiksel olarak şifreleme ve şifre çözmede kullanmaktır. Ayrıca burada ilave olarak optiksel mercek için, Fourier dönüşümü kullanılmaktadır. Bu uygulamamız incelendiğinde şifrelenmiş işaret çok daha fazla karışık olmaktadır, böylece şifrelemenin güvenirliği artmaktadır. Sonuç olarak, Sıkıştırıcı Algılama ile Hibrit (Simetrik+Asimetrik) Optiksel Kriptografi (Eşlenik Anahtarlar Kullanarak) uygulaması

gerçekleştirilmiştir. Dördüncü uygulamada ise, Sıkıştırıcı Algılama+Hibrit Optiksel Kriptografi uygulaması için; simetrik, asimetrik ve hibrit şifrelemesinde oluşturduğumuz anahtarların, ayrıca şifrelemenin daha zor olması, faz geri elde yöntemleri ile çözülememesi için; anahtar oluşumunda sadece faz $(e^{j\theta})$ ile değil de genlik ve faz çarpımı $(A * (e^{j\theta}))$ ile anahtar oluşumu gerçekleştirilmiştir. Bu durumda da faz geri elde etme tekniği ile de kırılamayacaktır. Yaptığımız beşinci uygulamada ise, iki anahtarın genlik ve faz çarpımı $(A * (e^{j\theta}))$ ile girişte karakter/metin yerine (1-boyut) resim girişi (2-boyut) kullanılmıştır. Yaptığımız önceki uygulamalarda kullanıcının dışarıdan girdiği karakterler ve resim üzerinde çalışılmıştır. Son uygulamamız olan altıncı çalışmada ise; en çok kullanılan kriptoanaliz (ataklar) üzerine faz geri elde tekniği ile ve dirac delta fonksiyon ile yaptığımız uygulamaların sağlamlığı deneysel olarak gösterilmiştir.

Uyguladığımız bu algoritmalarda, yüksek doğruluk elde edilmekte ve daha önce uygulanan tekniklerle kırılması imkansıza yaklaşmaktadır. Bizim uygulamamızda, sıkıştırıcı algılama kullandığımız için şifrelenmiş işaret orijinal işaretten çok daha küçük boyutta olmaktadır, böylece hem şifrelenmiş işaret orijinal işarete göre çok daha anlamsız olmakta, bu da şifrelenmiş işareti ataklara karşı çok daha fazla koruyucu yapmaktadır. Ayrıca şifrelenmiş işaretin orijinal işarete göre boyutu çok daha az olduğu için verimli ve efektif data transferi sağlanmaktadır. Bunlara ilave olarak program dönüşümler sayesinde çok hızlı çalışmaktadır. Özellikle Ersoy ve Nouira tarafından [64] de belirtilen çalışmada orijinal olan ayrık kosinüs-III dönüşümü (AK3D), ayrık kosinüs dönüşümüne(AKD) göre çok daha az çarpma işlemi içerdiğinden en hızlı sonucu vermektedir. AK3D dönüşümü sayesinde ayrıca sıkıştırılmış algılamada alışagelmiş dönüşüm değil de farklı bir dönüşüm kullanılmıştır. Bu şekilde doğruluk ve süre açısından daha iyi sonuç elde edilmiştir. Tüm programlar tarafımızdan oluşturulmuştur.

Ayrıca önerdiğimiz uygulamalar, literatürde en çok kullanılan 2 atak türü olan faz geri elde yöntemleri ile atağa ve dirac delta fonksiyon ile atağa da karşı dirençli olmaktadır. Deneysel olarak kriptoanaliz sonuçları başarılı bir şekilde sağlanmıştır. Algoritma sonucunda doğru anahtar ve yanlış anahtar ile hata ve sonuç, işlem süresi, sıkıştırıcı algılama ile kriptosu açılmış değerler ve görüntü elde edilmekte ve orijinal işaret, şifrelenmiş işaret, doğru anahtar ve yanlış anahtar ile elde edilmiş sıkıştırıcı

algılama+kripto sonucu çizdirilmektedir. Bu uygulama ile sıkıştırıcı algılama dik eşleştirme yöntemi ve optiksel kriptografi birbirinden bağımsız olarak değil de, sıkıştırılmış algılama dik eşleştirme arayış yöntemi ve çift rastgale faz+genlik anahtarlı simetrik, asimetrik ve hibrit optik kriptografi birbirinin içine geçmiş bir şekilde kullanılmıştır. Böylece hem sıkıştırıcı algılama ile data boyutu azaltma (sensör sayısı azaltma) hem de sıkıştırıcı algılama matrisinde iki defa kompleks ve dağıtılmış genlik+faz anahtarları ile güçlendirilmiş ve güvenliği artırılmış optiksel şifreleme ve şifre çözme ile simetrik, asimetrik ve hibrit kriptografi elde edilmiştir. Ayrıca faz geri elde teknikleri ve dirac delta fonksiyon atağı ile kırılamayacak hem 1-boyutlu hem de 2-boyutlu giriş işareti uygulaması gerçekleştirilmiştir.

Çizelge 6.1'de bu tezde gerçekleştirilen 6 uygulama için analiz sonuçları gösterilmiştir. Burada dikkat edildiğinde tüm uygulamalarda süre ve hata değerleri birbirine yakın ve çok düşük değerlerde seyretmektedir, ayrıca sıkıştırma oranları da birbirine yakın olmaktadır, fakat şifrelemenin zorluğu, anahtar kullanımı, atakların durumu, giriş işareti ve kriptoanaliz çalışmaları sonuçları değiştirmektedir.

Ayrıca N ve M boyutunda oluşan anahtarların bütün değerlerinin farklı olmasına gerek kalmadan sadece tek değerinin bile farklı olması durumunda şifrelenmiş ve sıkıştırılmış işaretten orijinal işaretin geri elde edilemeyeceği gösterilmiştir. Bu durum sayesinde kaba kuvvet (brute force) saldırılarına daha dayanıklı olacağı anlamına gelmektedir.

Referans [26-29] da belirtilen uygulamalarda anahtarlar sadece faz $(e^{j\theta})$ ile oluşturulmuştur. Gerçekleştirdiğimiz uygulamalarda ise hem 4f optiksel uygulaması hem faz geri elde yöntemi atağına karşı faz+genlik anahtarı $(A * (e^{j\theta}))$ hem de dirac delta atağı için dağıtılmış anahtar uygulaması gerçekleştirilmiştir. Kriptoanaliz sonuçları da başarı ile sağlanmıştır.

Dönüşümlerin kullanıldığı algılama matrisinde birbirine yakın satır ve sütunlardaki değerlerin yumuşak geçişler ve sıfıra yakın değerler olduğunda sonuçların daha iyi olduğu gözlenmiştir. Bu dönüşümler içerisinden süre ve doğruluk dikkate alındığında en iyi sonucu birbine yakın olarak, AK3D, akabinde sırasıyla AKD, HFD, GSD, ASD ve en kötü sonucu da hadamard ve haar vermektedir.

Bu çalışma, haberleşme alanındaki en önemli iki problem olan; efektif/etkin/verimli işaret işleme ve bu işareti güvenli olarak iletme konularına alternatif çözüm olarak araştırma için yol gösterici olabilecektir.

Çizelge 6.1 Gerçekleştirilen uygulamaların analizi

ANALİZ	1.Uygulama	2.Uygulama	3.Uygulama	4.Uygulama	5.Uygulama	6.Uygulama
Süre	2,23	2,17	2,1	2,11	3,81	3,75
Hata	$1{,}95.10^{-15}$	$1{,}80.10^{-15}$	$1{,}15.10^{-15}$	$2{,}01.10^{-15}$	$3{,}5.10^{-10}$	$3{,}7.10^{-15}$
Sıkıştırma	3:1	3:1	3:1	3:1	3,16:1	3,16:1
Anahtar Durumu	Çift faz anahtarı	Çift faz anahtarı	Çift optiksel anahtar	Çift faz+genlik optik anahtarı	Çift faz+genlik optik anahtarı	Çift rastgele, dağıtılmış ve kompleks faz+genlik optik anahtarı
Dirac Delta Atak Sağlam	X	X	✓	✓	✓	✓
Faz Geri Elde Atak Sağlamlığı	X	X	X	✓	✓	✓
Optiksel Kriptografi	X	X	✓	✓	✓	✓
Çıkışın/Girişe Oranı (zorluk der.)	08:01	08:01	500:01:00	500:01:00	08:01	500:01:00
Şifreleme Yöntemi	Simetrik	Hibrit	Optiksel hibrit	Optiksel hibrit	Optiksel hibrit	Optiksel hibrit
Giriş Boyutu	1-D	1-D	1-D	1-D	2-D	1-D / 2-D
Kriptoanaliz	X	X	X	X	X	✓

KAYNAKLAR

[1] Candes, E. J., (2006). “Compressive Sampling,” in International Congress of Mathematicians (ICM), Madrid, Spain , 3: 1433–1452.

[2] Donoho, D., (2006). ‘‘Compressed Sensing’’, IEEE Trans. on Information Theory, 52(4):1289 - 1306

[3] Georgia Üniversitesi, Elektrik ve Bilgisayar Mühendisliği, Compressive Sensing, http://users.ece.gatech.edu/ justin/ssp2007, 10 Ocak 2014.

[4] Gilbert, A. ve Indyk, P., (2010). “Sparse Recovery Using Sparse Matrices,” Proceedings of the IEEE, 98(6): 937–947.

[5] Donoho, D., (2006). ‘‘Compressed Sensing’’, IEEE Trans. on Information Theory, 52(4): 1289 – 1306.

[6] Shannon, C. E., (1949). “Communication in The Presence of Noise,” Proc. Institute of Radio Engineers, 37: 10–21.

[7] Gwangju Enstitüsü, Information sensing, processing, controlling, and networking, Compressive Sensing Applications, http://infonet.gist.ac.kr/, 10 Ocak 2014.

[8] Baraniuk, R., (2007). ‘‘Compressive Sensing’’, IEEE Signal- Processing Magazine, 24(4):118-121.

[9] Baron, D., Wakin, M. B., Duarte, M. F., Sarvotham, S. ve Baraniuk, R. G., (2006). “Distributed Compressed Sensing,” Tech. Rep. TREE-0612, Rice University, Department of Electrical and Computer Engineering.

[10] Haupt, J., Bajwa, W. U., Rabbat, M. ve Nowak, R., (2008). “Compressed Sensing for Networked Data,” IEEE Signal Processing Magazine, 25:92–101.

[11] Candès, E. ve Wakin, M., (2008) . “An Introduction to Compressive Sampling”, IEEE Signal Processing Mag., 25(2): 21-30.

[12] Babaoğlu, A., (2011). ‘‘Kriptolojinin Geçmişi: Bir Şifreleme Algoritması Kullanmadan Önce Son Kullanım Tarihine Bakın’’, Bilim Teknik, 500.Sayı.

[13] Apohan, U. K. ve Kara, O., (2009). ‘‘Bilgi Güvenliği Problemlerine Matematiksel Yaklaşım Getiren Bir Bilim Dalı: Kriptoloji’’, Bilim ve Teknik, 500.Sayı.

[14] Kara, O., (2009). ‘‘Kriptografinin Yapıtaşları Kriptografik Algoritmalar ve Protokoller’’, Bilim ve Teknik, 500.Sayı.

[15] Schneier, B., (1996). Applied Cryptography - Protocols, Algorithms, and Source Code in C, John Wiley & Sons, Inc., 2nd edition.

[16] Washington, L. C., (2003). Elliptic Curves, Number Theory and Cryptography, Chapman&Hall/Crc.

[17] Amir, M. A. ve Esther, R., (2010). '' Compressive Sensing: From Compressing while Sampling to Compressing ve Securing while Sampling", 32nd Annual International Conference of the IEEE EMBS, August 31 - September 4, Buenos Aires, Argentina.

[18] Minal, C. ve Rajankar, S., (2013). '' Study the Effects of Encryption on Compressive Sensed Data'', International Journal of Engineering and Advanced Technology (IJEAT), 2(Issue-5): ISSN: 2249 – 8958.

[19] Rachlin, Y. ve Baron, D., (2008). ''The Secrecy of Compressed Sensing Measurements'', Communication, Control, ve Computing, 46th Annual Allerton Conference on, Urbana-Champaign, IL,813-817.

[20] Dwork, C., McSherry, F. ve Talwar, K., (2007). "The Price of Privacy and The Limits of LP Decoding," in Symp. on Theory of Computing (STOC).

[21] Orsdemir, A., Altun, H. O. ve Sharma, G., (2008). ''On The Security and Robustness of Encryption Via Compressed Sensing'', In: Proceedings of the IEEE military communications conference MILCOM.

[22] Ramezani, M., Seyfe, B. ve Bafghi, H.G., (2013). ''Perfect Secrecy Via Compressed Sensing'', Communication ve Information Theory (IWCIT), 2013 Iran Workshop on, 8-9 May 2013, Tehran, 1-5.

[23] Zhang, Y., Wong, K., Xiao, D., Zhang, L.Y. ve Li, M., (2012). ''Embedding Cryptographic Features in Compressive Sensing'', Cryptography and Security, Information Theory.

[24] Zhang, X., Ren, Y., Feng, G. ve Qian, Z., (2011). ''Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)'', Seventh International Conference on, 14-16 Oct. 2011, Dalian ,222 – 225.

[25] Mo, Y. Zhang, A., Zheng, F. ve Nzhou, N., (2013). ''An Image Compression-Encryption Algorithm Based on 2-D Compressive Sensing'', Journal of Computational Information Systems 9(24):10057-10064.

[26] Peng, X., Zhang, P., Wei, H. ve Yu, B., (2006). ''Known-Plaintext Attack on Optical Encryption Based on Double Random Phase Keys'', Institute of Optoelectronics, Shenzhen University, 518060 Shenzhen, China, Optics Letters, 31(8).

[27] Peng, X., Wei, H. ve Zhang, P., (2006). ''Chosen-Plaintext Attack on Lensless Double-Random Phase Encoding in The Fresnel Domain'', Optics Letters, 31(22).

[28] Carnicer, A., Usategui, M. M., Arcos, S. ve Juvels, I., (2005). ''Vulnerability to Chosen-Cyphertext Attacks of Optical Encryption Schemes Based on Double Random Phase Keys'', Optics Letters, 30(13).

[29] Frauel, Y., Castro, A., Naughton, T. J. ve Javidi, B., (2007) . ''Resistance of The Double Random Phase Encryption Against Various Attacks'', Optics Letters, 15(16).

[30] Donoho, D., (2006). ''Extensions of Compressed Sensing'', Signal Proces., 86(3):549–571.

[31] Baraniuk, R. G., Davenport, M. A., Duarte M. F. ve Hegde C., (2011). An Introduction to Compressive Sensing, Connexions e-textbook, Georgia Institute of Technology.

[32] Video Lectures, An Overview of Compressed Sensing and Sparse Signal Recovery via L1 Minimization, http://videolectures.net/mlss09us_candes_ocsssrl1m/, 30 Kasım 2013.

[33] Tropp, J., (2010). ''Beyond Nyquist Efficient Sampling of Sparse Bandlimited Signals'', IEEE Trans. on, 56:520–544.

[34] Candès, E., Romberg, J. ve Tao, T., (2006) . ''Robust Uncertainty Principles: Exact Signal Reconstruction From Highly Incomplete Frequency Information'', IEEE Trans. on Information Theory, 52(2):489-509.

[35] Anila, S. B., Deepa, B., Bhai, S. ve Devi, A. S., (2012). ''Compressive Sensing for Array Signal Processing'', India Conference (INDICON), 2012 Annual IEEE,7-9 Dec. 2012, Kochi.

[36] Pope, G., (2009). Compressive Sensing A Summary of Reconstruction Algorithms, Masters Thesis, Department of Computer Science Eidgenossische Technische Hochschule, Zurich.

[37] Donoho, D. ve Huo, X., (2015). "Uncertainty Principles Near Optimal Signal Recovery from Random Projection: Universal Encoding Strategies", IEEE Trans. Inform. Theory, 52(12):5406-5425.

[38] Wu-Sheng, L., (2010). '' Compressed Sensing and Sparse Signal Processing'', Lecture Notes, University of Victoria, Canada.

[39] Elad, M., (2007). "Optimized Projections for Compressed Sensing Sampling and Reconstruction of Multiband Signals", IEEE Trans on Signal Processing , 55:(Issue:12):5695 – 5702.

[40] Hale, E., (2007). ''A Fixed-Point Continuation Method for ℓ1-Regularized Minimization with Applications to Compressed Sensing'', Technical Report TR07-07 Department of Computational and Applied Mathematics, Rice University, Houston.

[41] Donoho, D., Chen, S. ve Saunders, M., (1998). "Atomic Decomposition by Basis Pursuit", SIAM Journal on Scientific Computing, 20:33-61.

[42] Septimus, A. ve Steinberg, R., (2010), "Compressive Sampling Hardware Reconstruction",International Symposium on Circuits and Systems, Paris, 3316-3319.

[43] Borghi, A., Darbon, J., Peyronett, S., Chan, T. ve Osher, S., (2008). "Compressive Sensing Algorithm for Parallel Many-Core Architectures," CAMS Report.

[44] Donoho, D. ve Stark, P. B., (2008). "Uncertainty Principles and Signal Recovery", SIAM, J. Appl. Math., Vol. 49(3): 906-931.

[45] Tropp, J. ve Gilbert, (2007). A.C. "Signal Recovery from Partial Information Via Orthogonal Matching Pursuit", IEEE Trans. Inform. Theory, 53(12):4655-4666.

[46] Matthew, H. ve Strohmer, (2008). T., ''Compressed Sensing Radar'', IEEE International Conference on Acoustics, Speech and Signal Processing, Las Vegas, NV, USA.

[47] Becker, S. T., (2011). ‘’Practical Compressed Sensing: Modern Data Acquisition and Signal Processing Thesis’’, California Institute of Technology, Pasadena, California.

[48] Davenport, M.A., Duarte, M. F., Eldar, Y. C. ve Kutyniok G., (2011). “Introduction to Compressed Sensing,” in Compressed Sensing: Theory and Applications, Y. Eldar and G. Kutyniok, eds., Cambridge University Press.

[49] Malioutov, D., Çetin, M. ve Willsky, A. S., (2005). ‘’A Sparse Signal Reconstruction Perspective for Source Localization With Sensor Arrays’’, IEEE Transactions on Signal Processing, 53(8):3010 – 3022.

[50] Baraniuk, R., Davenport, M., Devore, R. ve Wakin, M., (2008). “A Simple Proof of The Restricted Isometry Property for Random Matrices,” Constr. Approx.

[51] Candes, E., Romberg J. ve Tao T., (2006). “Stable Signal Recovery from Incomplete and Inaccurate Measurements,” Communications on Pure and Applied Mathematics, 59:1207–1223.

[52] Lee, H. N., (2011). Introduction to Compressed Sensing, Gwangju Institute of Science andTechnology, Cheomdan Gwagiro, Gwangju, Korea,261.

[53] Candès, E. ve Romberg, J., (2007). “Sparsity ve Incoherence in Compressive Sampling”,Inverse Prob., 23(3):969-985.

[54] Feng, P. ve Bresler, Y., (1996). “Spectrum Blind Minimum Rate Sampling and Reconstruction of Multiband Signals”, IEEE Int. Conf. Acoustics Speech Signal Processing, 2:1689-1692.

[55] Tropp, J. ve Needell, D., (2008). “CoSaMP: Iterative Signal Recovery from Incomplete and Inaccurate Samples”, Arxiv preprint, arXiv:0803,2392.

[56] K. Herrity, A. Gilbert, ve J. Tropp, (2006). “Sparse Approximation Via Iterative Thresholding,” in Acoustics, Speech and Signal Processing. ICASSP 2006 Proceedings. 2006 IEEE International Conference on, vol. 3.

[57] Türeli, A. ve Gilbert, C., (2007). Signal Recovery From Random Measurements Via Orthogonal Matching Pursuit, IEEE Trans. Inform. Theory, 4655–4666.

[58] Karahanoğlu, N. B. ve Erdoğan, H., (2011). ‘’Compressed Sensing Signal Recovery via A* Orthogonal Matching Pursuit’’, IEEE International Conference on Acoustics, Speech and Signal Processing.

[59] Huang, H. ve Makur, A., (2011). ‘’Backtracking-Based Matching Pursuit Method for Sparse Signal Reconstruction’’, IEEE Signal Processing Letters, 18(7):391 – 394.

[60] Tropp, J. A., Gilbert, A. C. ve Strauss, M. J., (2006). “Algorithms for Simultaneous Sparse Approximation: Part i: Greedy Pursuit,” Signal Process, 86(3):572–588.

[61] Dai, W. ve Milenkovic, O., (2008). “Subspace Pursuit for Compressive Sensing: Closing the Gap Between Performance and Complexity”, Arxiv preprint , 803-811.

[62] Blumensath, T. ve Davies, M., (2007). “Gradient Pursuits”, IEEE Trans. Signal Processing Theory, 56(6):2370-2382.

[63] Boufounus, P., Duarte, M. ve Baraniuk, R., (2007). “Sparse Signal Reconstruction from Noisy Compressive Measurements using Cross Validation”, IEEE/SP 14th Workshop on Statistical Signal Processing,299-303.

[64] Ersoy, O.K. ve Nouira, A., (1992). ‘‘Image Coding With The Discrete Cosine-III Transform’’, IEEE Journal On Selected Areas In Communıcations, Vol. 5.

[65] Menezes, A. J., (1997). Handbook of Applied Cryptography, Boca Raton: CRC Press.

[66] Akyıldız, E., Doğanaksoy, A., Keyman, E. ve Uğuz, M., (2004). ‘‘Kriptolojiye Giriş Ders Notları’’, ODTÜ, Uygulamalı Matematik Enstitüsü, Kriptografi Bölümü.

[67] Bellovin S.M., (1998). ‘‘Cryptography and Internet’’, In Proceedings of CRYPTO '98.

[68] Yıldırım, H.M., (2014). ‘’Bilgi Güvenliği ve Kriptoloji’’, Uluslararası Adli Bilişim Sempozyumu, 31 Mayıs-02 Haziran 2014, Ankara.

[69] Stinson, D.R., (2003). Cryptography, Chapman & Hall/CRC Press Company, United States of America.

[70] Diffie, W. ve Hellman, M.E., (1976). ‘‘New Directions in Criptography’’, IEEE Transansaction on Information Theory, IT-22,644–654.

[71] FIPS 46-3, (1999). Data Encryption Standard, Federal Information Processing Standard (FIPS), Publication 46-3, National Bureau of Standards, U.S. Department of Commerce, Washington D.C.

[72] Esslinge, B., (2010). The CrypTool Script Cryptography, Mathematics, and More, 10th Edition, Frankfurt am Main, Germany, GNU Free Documentation License.

[73] FIPS 197, (2001). Advanced Encryption Standard, Federal Information Processing Standard (FIPS), Publication 197, National Bureau of Standards, U.S. Department of Commerce, Washington D.C.

[74] Rivest, R., Shamir A. ve Adleman, L., (1978). A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, Communications of the ACM, 21(2):120-126.

[75] Cyrptii, Text to Decimal, http://cryptii.com/text/select, 2 Şubat 2014.

[76] Wikipedia, Phase Retrieval, https://en.wikipedia.org/wiki/Phase_retrieval, 10 Mayıs 2014.

ÖZGEÇMİŞ

KİŞİSEL BİLGİLER

Adı Soyadı : Ertan ATAR

Doğum Tarihi ve Yeri : 28.08.1980 İstanbul

Yabancı Dili : İngilizce

E-posta : ertan.atar@turktelekom.com.tr / ertanatar@gmail.com

ÖĞRENİM DURUMU

Derece	Alan	Okul/Üniversite	Mezuniyet Yılı
Y. Lisans	Elektronik&Hab.	Yıldız Teknik Üniversitesi	2005
Lisans	Elektrik-Elektronik	Sakarya Üniversitesi	2002
Lise	Elektrik-Elektronik	Şişli Teknik Lisesi	1998

İŞ TECRÜBESİ

Yıl	Firma/Kurum	Görevi
2013	Türk Telekom	Operasyon Müdürü
2010	Türk Telekom	Uzman
2007	Türk Telekom	Uzman Yrd.
2006	BEDAŞ	Mühendis

2005	Baykal Akü	Mühendis
2003	MEB	Vekil Öğretmen
2001	Profilo	Stajyer
2000	Turkcell	Stajyer
1997	BEDAŞ	Stajyer
1996	BEDAŞ	Stajyer