T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KUANTUM ANAHTAR DAĞITIMINDA BİLGİ UZLAŞTIRMA

MUSTAFA TOYRAN

DOKTORA TEZİ

ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

GEBZE
2016

T.C.
GEBZE TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

KUANTUM ANAHTAR DAĞITIMINDA BİLGİ UZLAŞTIRMA

MUSTAFA TOYRAN
DOKTORA TEZİ
ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI
PROF. DR. AHMET ARİF ERGİN

GEBZE
2016

T.R.

GEBZE TECHNICAL UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

INFORMATION RECONCILIATION IN QUANTUM KEY DISTRIBUTION

MUSTAFA TOYRAN

A THESIS SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF ELECTRONIC ENGINEERING

THESIS SUPERVISOR

PROF. DR. AHMET ARİF ERGİN

GEBZE

2016

GEBZE TEKNİK ÜNİVERSİTESİ	**DOKTORA JÜRİ ONAY FORMU**

GTÜ Fen Bilimleri Enstitüsü Yönetim Kurulu'nun 11/05/2016 tarih ve 2016/31 sayılı kararıyla oluşturulan jüri tarafından 08/06/2016 tarihinde tez savunma sınavı yapılan Mustafa TOYRAN'ın tez çalışması Elektronik Mühendisliği Anabilim Dalında DOKTORA tezi olarak kabul edilmiştir.

JÜRİ

ÜYE
(TEZ DANIŞMANI) : Prof. Dr. A. Arif ERGİN

ÜYE : Doç. Dr. Yakup HAMEŞ

ÜYE : Doç. Dr. Koray KAYABOL

ÜYE : Prof. Dr. M. Bülent ÖRENCİK

ÜYE : Doç. Dr. Hasari ÇELEBİ

ONAY

Gebze Teknik Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun
...../...../..... tarih ve ...../..... sayılı kararı.

İMZA/MÜHÜR

ÖZET

Kuantum kriptografi camiasında çok fazla haberleşme gerektirmesinden dolayı CASCADE gibi interaktif bilgi uzlaştırma protokollerinin verimsiz olduğu yaygın olarak kabul edilir. Yerine, LDPC kodlarına ya da, daha yenice, kutupsal kodlara dayanan interaktif olmayan bilgi uzlaştırma protokolleri önerilmiştir. Bu çalışmada, modern kuantum anahtar dağıtım sistemlerinde interaktif protokollerin dikkate alınmasını tartışıyoruz. Özel olarak, doğru gerçekleme ve kullanım ile CASCADE'in performansının nasıl iyileştirilebileceğini gösteriyoruz.

Anahtar Kelimeler: Kuantum Anahtar Dağıtımı, Bilgi Uzlaştırma, Gizli Anahtar Uzlaştırma, CASCADE Protokolü.

SUMMARY

It is widely accepted in the quantum cryptography community that interactive information reconciliation protocols, such as CASCADE, are inefficient due to the communication overhead. Instead, non-interactive information reconciliation protocols based on i.e. LDPC codes or, more recently, polar codes have been proposed. In this work, we argue that interactive protocols should be taken into consideration in modern quantum key distribution systems. In particular, we demonstrate how to improve the performance of CASCADE by proper implementation and use.

Key Words: Quantum Key Distribution, Information Reconciliation, Secret Key Reconciliation, CASCADE Protocol.

TEŞEKKÜR

Doktora çalışmamı başarıyla sonuçlandırmamda büyük olumlu katkıları olan sayın Prof. Dr. Ahmet Arif ERGİN (11.06.2015'ten itibaren 27. TÜBİTAK Başkanı), Doç. Dr. Yakup HAMEŞ, Doç. Dr. Koray KAYABOL, Prof. Dr. Mehmet Bülent ÖRENCİK ve Doç. Dr. Hasari ÇELEBİ hocalarıma çok teşekkür ederim.

İÇİNDEKİLER

	Sayfa
ÖZET	v
SUMMARY	vi
TEŞEKKÜR	vii
İÇİNDEKİLER	viii
SİMGELER ve KISALTMALAR DİZİNİ	x
ŞEKİLLER DİZİNİ	xii
TABLOLAR DİZİNİ	xiv
1. GİRİŞ	1
1.1. Literatür	1
1.2. Tezin İçeriği	4
2. TEMEL KAVRAMLAR	5
2.1. Bilgi Güvenliği	5
2.2. Klasik Kriptografi	6
2.2.1. Anahtar Dağıtım Problemi	9
2.3. Kuantum Mekaniği	9
2.3.1. Heisenberg Belirsizlik İlkesi	13
2.3.2. Kuantum Kopyalanamazlık Teoremi	15
2.4. Kuantum Kriptografi	15
2.4.1. Kuantum Rasgele Sayı Üretimi	17
2.5. Bilgi Teorisi	19
3. KUANTUM ANAHTAR DAĞITIMI	22
3.1. BB84 Protokolü	22
3.1.1. Foton Polarizasyonu	23
3.1.2. Fotonun Polarize Edilmesi	25
3.1.3. Fotonun Polarizasyonunun Ölçülmesi	26
3.1.4. Polarize Edilmiş Foton için Kopyalanamazlık Teoremi	27
3.1.5. Polarize Edilmiş Foton için Heisenberg Belirsizlik İlkesi	27
3.1.6. Mesajların Polarize Edilmiş Fotonlarla Kodlanması	28
3.1.7. Polarize Edilmiş Fotonlarla Anahtar Dağıtımı	29

3.1.8. İletişimin Dinlendiğinin Anlaşılması 32
3.1.9. Güvenlik Seviyesi 37
3.2. Diğer Yaklaşımlar 40
3.3. Dezavantajlar 41
3.4. Avantajları 41
3.5. Şifreleme 42
4. BİLGİ UZLAŞTIRMA 43
4.1. Kuantum Kanal 43
4.1.1. Eşlik Kontrolü 47
4.2. Sonraki İşlemler 48
4.2.1. Bilgi Uzlaştırma 51
4.3. CASCADE Protokolü 60
4.3.1. Strateji 60
4.3.2. Parametreler 63
4.3.3. Karıştırma 63
4.3.4. BINARY 64
5. PERFORMANS 66
5.1. Hızlı CASCADE 66
5.2. Verimli CASCADE 74
6. SONUÇ 83

KAYNAKLAR 86
ÖZGEÇMİŞ 95
EKLER 96

SİMGELER ve KISALTMALAR DİZİNİ

Simgeler ve Kısaltmalar		**Açıklamalar**
ABD	:	Amerika Birleşik Devletleri
AES	:	Advanced Encryption Standard
AIT	:	Austrian Institute of Technology
ARQ	:	Automatic Repeat Request
AT&T	:	American Telephone and Telegraph Company
auth	:	Authentication
BB84	:	Bennett, Brassard, 1984
bps	:	Bits per second (bit/s, b/s)
BU	:	Bilgi uzlaştırma
CD	:	Compact Disc
CPU	:	Central Processing Unit
DES	:	Data Encryption Standard
DPS	:	Differential Phase Shift
DSS	:	Digital Signature Standard
DVD	:	Digital Versatile Disc
e	:	Gürültü
e^-	:	Elektron
FEC	:	Forward Error Correction
FPGA	:	Field Programmable Gate Array
Gbps	:	Gigabit per second (Gbit/s, Gb/s)
GHz	:	Gigahertz
GRSÜ	:	Gerçek Rasgele Sayı Üreteci
GSM	:	Global System for Mobile Communications
GTÜ	:	Gebze Teknik Üniversitesi
HP	:	Hewlett-Packard
IBM	:	International Business Machines
IDEA	:	International Data Encryption Algorithm
İSK	:	İkili Simetrik Kanal
KAD	:	Kuantum Anahtar Dağıtımı

km	:	Kilometre
KRSÜ	:	Kuantum Rasgele Sayı Üreteci
LDPC	:	Low Density Parity Check
m	:	Metre
Mbit	:	Megabit
Mbps	:	Megabit per second (Mbit/s, Mb/s)
ms	:	Milisaniye
NSA	:	National Security Agency
OBEB	:	Ortak Bölenlerin En Büyüğü
QBER	:	Quantum Bit Error Rate
RAM	:	Random Access Memory
RC4	:	Rivest Cipher 4
RSA	:	Rivest, Shamir, Adleman
sn	:	Saniye (s)
SRSÜ	:	Sözde Rasgele Sayı Üreteci
XOR	:	eXclusive OR ($\oplus$)

ŞEKİLLER DİZİNİ

Şekil No:		Sayfa
2.1:	Simetrik kriptografi.	7
2.2:	Asimetrik kriptografi.	7
2.3:	Şifreleme ve şifre çözmenin matematiği.	8
2.4:	Mekaniğin 4 büyük uğraş alanı.	10
2.5:	Filtreler deneyi.	12
2.6:	Heisenberg belirsizlik ilkesi.	13
2.7:	Kuantum rasgele sayı üretimi.	18
2.8:	Haberleşme sistemi basit bir blok diyagramı.	20
3.1:	Foton, elektrik alanı, manyetik alanı.	24
3.2:	Polarizasyon tabanları.	24
3.3:	Doğrusal polarizasyonda mevcut dört polarizasyon durumu.	25
3.4:	Fotonun dikey polarizasyon durumuna sokulması.	25
3.5:	Fotonların kristalden geçişi.	26
3.6:	Polarizasyon ölçümleri.	28
3.7:	İkili bit değerlerinin polarize edilmiş fotonlarla temsili.	29
3.8:	Polarizasyon tabanlarının ikili bitlerle temsili.	29
3.9:	Bitleri foton polarizasyon durumları ile kodlama kuralları.	30
3.10:	Tarafların ortak bir gizli anahtar üzerinde anlaşması.	30
3.11:	Elde edilen ortalama anahtar uzunlukları.	32
3.12:	Hattı dinleyen birinin varlığı.	33
3.13:	Hattı dinleyen birinin varlığının anlaşılması.	34
3.14:	Saldırganın gönderici ile aynı tabanları seçme olasılığı.	35
3.15:	Saldırganın tespit edilememe durumu.	36
3.16:	Hattın dinlenip dinlenmediğinin belirlenme olasılığı.	38
3.17:	Saldırganın belirlenme olasılığı.	39
3.18:	Test bitleri kullanılması durumunda anahtar uzunlukları.	39
4.1:	İSK blok diyagramı.	43
4.2:	KAD'da temel bileşenler.	46
4.3:	BB84 protokolü.	48
4.4:	KAD'da kimlik doğrulamanın yeri.	51

4.5: KAD'da gürültü. 56
4.6: KAD'da bilgi uzlaştırma. 57
4.7: Yan bilgi ile kaynak kodlama. 59
5.1: Bit dizilerini düzeltmek için değiş-tokuş edilen mesaj sayısı. 69
5.2: Bit dizilerini düzeltmek üzere harcanan zaman. 73
5.3: İki uzunluklu blok durumu. 77
5.4: Üç uzunluklu blok durumu. 78
5.5: Üç uzunluklu blok için sağdan dallanma. 78
5.6: Önceki küçük bloklar durumu. 79

TABLOLAR DİZİNİ

Tablo No:		**Sayfa**
3.1:	Saldırganın belirlenemediği durumda anlaşılan anahtarlar.	40
4.1:	Bitsel XOR işlemi ($\oplus$).	47
4.2:	k_1 seçimi.	63
5.1:	Verimlilik performansı karşılaştırması.	82

1. GİRİŞ

Bu bölümde konuyla ilgili literatürde yapılmış çalışmalar ve tezin konuya katkıları anlatılmaktadır.

1.1. Literatür

Kuantum anahtar dağıtımı (KAD), atanmış bir fiber hattı ile birbirine bağlı olan ya da birbirini görebilen iki taraf için uçtan uca bilgi teoriksel olarak güvenli anahtar dağıtımı olanağı sağlar.

KAD'da ilk adım, kuantum durumlarda kodlanmış bitlerin iletimidir. Heisenberg'in belirsizlik ilkesi ve kuantum kopyalanamazlık teoremi, kuantum kanaldan iletim sırasında oluşan hatalar ile saldırganlara (Eve) sızan bilgi arasında sıkı bir ilişkiyi garanti eder.

KAD'da bilgi uzlaştırma (BU), gönderici (Alice) ve alıcı (Bob) tarafların gürültülü kuantum kanalın kullanımı ile elde ettikleri bit dizileri arasındaki farklılıkları, kimlik doğrulamalı klasik kanal üzerinden herkese açık bir tartışma ile bulup düzelttikleri protokoldür.

BU, hem ayrık [1]-[3] hem de sürekli [4], [5] değişkenli birçok günümüz KAD sistemindeki darboğazlardan biridir. Bu çalışmada, ayrık değişkenli KAD sistemleri için kullanılan model olan, kuantum kanalın bir ikili simetrik kanal (İSK) ile modellendiği BU problemini ele almaktayız.

Bir BU protokolünün performansının iki ana ölçütü vardır: Verimlilik (yani, saldırganlara gerektiğinden daha fazla bilgi sızdırmadan farklılıkları düzeltme yeteneği) ve hız (yani, saniyede kaç tane giriş biti işlenebildiği). Bunlar farklı BU protokolleri için çeşitlilik gösterebilen özelliklerdir.

CASCADE [6] basit bir BU protokolüdür ve belki de şu an için KAD gerçeklemelerinde en fazla kullanılanıdır. CASCADE'de, Alice ve Bob bir bit dizisini öncelikle karıştırır ve parçalara böler. Daha sonra, her bir parçanın eşliklerini karşılaştırırlar. Bir parçanın eşliğinde uzlaşamadıklarında, parça iki yarıya bölünür ve yarıların eşlikleri karşılaştırılır. Eşliklerin uyuşmadığı yarı daha sonra hata bulunana ve düzeltilene kadar özyinelemeli olarak tekrar ikiye bölünür ve kontrol edilir. Gerekli olan iletişimler, yukarıda da belirtildiği gibi, herkese açık (ancak, saldırganın

mesajları asla değiştiremeyeceğinin garanti altına alındığı) bir klasik kanal üzerinden sağlanmaktadır. Bu prosedür, farklı karıştırmalar ve parça uzunlukları ile birkaç tur devam eder. Kolayca görülebileceği üzere, CASCADE oldukça interaktiftir ki bu da kendisini ağ gecikmelerine çok duyarlı yapar. Yaygın olarak, CASCADE'deki etkileşimin düşük hıza neden olduğuna inanılmaktadır [3], [7]-[9]. CASCADE'in aksine, LDPC (Low-Density Parity-Check) ya da kutupsal kodlar gibi ileriyönlü hata düzeltme metotlarına dayanan modern BU protokolleri interaktif değildir. Bununla birlikte, bunlar CASCADE'den daha fazla hesaplama gerektirirler.

Verimlilik, BU üzerine yapılan çalışmalarda en fazla ele alınan performans ölçütüdür (örneğin, bkz. [6], [9]-[14]). Bununla birlikte, [3] ve [7]'de de tartışıldığı gibi, BU protokolleri seçilirken haberleşme ve hesaplama giderleri arasındaki getiri-götürüler dikkatlice değerlendirilmelidir. BU, bir önceki faz olan eleme fazı tarafından kendisine girdi olarak sağlanan R_s elenmiş anahtar hızından daha yüksek bir hıza sahip olduğu sürece, verimlilik baskın olan performans kıstasıdır. Yani, gerekirse verimliliği arttırmak için hızdan aradaki fark kadar feragat edilebilir. Bununla birlikte, eğer BU fazının hızı kendisine gelen giriş hızından daha düşükse, tüm KAD sisteminin performansını hız ve verimliliğin bir birleşimi belirler. Yani, gerekirse, bu durumda da hızı arttırmak amacıyla, güvenlik sınırlarının dışına çıkmamak kaydıyla, yüksek olan verimlilikten bir miktarı feda edilebilir.

Literatürde düşük gecikmeli ağlarda hesapsal olarak basit ama interaktif BU protokollerinin hesapsal olarak karmaşık ama interaktif olmayan protokollerden daha kötü performans gösterip göstermeyeceği konusu henüz netlik kazanmamıştır. Bu çalışmanın amacı, [6], [9], [14] gibi interaktif protokollerin interaktif olmayanlardan daha kötü performansa sahip olduğu varsayımına meydan okumaktır. İddiamızı desteklemek için, özel olarak, CASCADE'in, doğru gerçekleme ve kullanım ile birçok KAD gerçeklemesi için, örneğin, çok düşük gecikmelere sahip olan fiber kanallar üzerinden KAD gerçeklemeleri gibi, en son BU protokollerinden [2], [3], [7], [8], [15] daha iyi performans gösterebildiğini kanıtlamaya çalışmaktayız.

KAD gerçeklemelerinin birçoğunda, kimlik doğrulamalı klasik kanal en fazla birkaç milisaniyelik (ms) bir gecikmeye sahip olacaktır. Fiber üzerinden bir KAD sisteminde, klasik kanalın kuantum kanal ile çoğullanmış olduğunu [16], [2] ya da aynı fiber demetindeki başka bir fiber üzerinden gönderildiğini varsaymak gerçeğe uygundur. Mevcut KAD sistemlerinin sahip olduğu mesafeler için, doğrudan bir fiber bağlantısı 1 ms'ye yakın gecikmeler verecektir (belirli bir mesafe için gecikme

süresi, mesafenin ışık hızına bölünmesiyle hesaplanabilir). Koordinasyon, eleme, hata tahmini, bilgi uzlaştırma, hata doğrulama ve gizlilik arttırma fazlarının tümü aynı kanal üzerinden yapılsa bile, Gbps hızındaki bir bağlantı gecikme üzerinde çok fazla bir etki yapmadan tüm bu haberleşme görevlerini çoğullamaya muktedirdir. Dünyaya en yakın yörüngelerde yer alan LEO (Low Earth Orbit) uydular ile olan açık-hava üzerinden KAD [17] için bile, gecikme birkaç milisaniyeden fazlasını geçmeyecektir. CASCADE'in böylesi çok düşük gecikmeli senaryolarda 80 Mbps'in üzerinde bir hızı nasıl elde edebildiğini gösteriyoruz. Sadece ağ gecikmesi 10 ms'yi geçince, CASCADE gerçeklememizin hızı en son KAD sistemleri için çok düşük olmaktadır. Gecikmenin CASCADE kullanılamayacak derecede yüksek olduğu böylesi potansiyel kurulumlara bir örnek, birkaç yüz milisaniye gecikmelerin olacağı GEO (Geostationary Earth Orbit) uydular ile yapılan KAD'dır.

Güncel BU sistemleri ve performansları ile ilgili karşılaştırmalı incelemelere [7]'den ulaşılabilir. Kutupsal ve LDPC kodların karşılaştırıldığı çalışmada rapor edilen verilere göre, mevcut BU sistemlerinde elde edilen

- en yüksek verimlilik 0.98 olup (bu değer, verimliliğin teorik limitine, teorideki ortak bilgi miktarına, hangi yüzdede ulaşıldığını ifade etmektedir) 2%'lik QBER (Quantum Bit Error Rate) için Intel Core i5 670 3.47 GHz CPU üzerinde tek çekirdek kullanarak gerçeklenen 2^{24} bit blok uzunluklu kutupsal kodlarla elde edilmiştir (bu blok uzunlukları LDPC kodlarınkilere göre çok daha yüksektir). Bu sistemin çalışma hızı ise 8.3 Mbps'dir (yani, saniyede 8.3 milyon tane kendisine gelen giriş bitini işleyebilmektedir).
- Ulaşılabilen en yüksek hız 10.9 Mbps olup yine 2%'lik bir QBER için aynı ortamda bu defa 2^{16} bit blok uzunluklu kutupsal kodlarla elde edilmiştir. Bu durumda verimlilik ise 0.94 olarak elde edilmiştir (blok uzunluğu azalınca hız artmış, verimlilik azalmıştır).

Güncel BU sistemleri ve performansları ile ilgili olarak referans [96]'da sunulan daha güncel verilere göre ise, ulaşılabilen en yüksek akış hızı 35 Mbps olup yine 2%'lik bir QBER için GPU üzerinde gerçeklenen 1944 bit blok uzunluklu LDPC kod ile elde edilmiştir. Bu durumda ulaşılan verimlilik ise 0.87 olarak verilmektedir. Aynı referansta (FPGA üzerinde) 40.8 Mbps hızlarında çalışabilecek LDPC kodlarının varlığından da söz edilmiştir.

Güncel KAD sistemleri ile ilgili karşılaştırmalı incelemelere [18]'den ulaşılabilir. Bir KAD sisteminin temel performans ölçütlerinden iki tanesi erişilebilen maksimum mesafe ve maksimum gizli anahtar dağıtım hızıdır. [18]'de tablolar halinde rapor edilen verilere göre, mevcut KAD sistemlerinde

- erişilebilen en yüksek gizli anahtar dağıtım hızı 1.65 Mbps olup sadece 5.6 kilometrelik (km) bir mesafe için elde edilebilmiştir [1].
- Ulaşılabilen en uzak mesafe ise 250 km olup bu gerçeklemede de sadece 15 bps hızında bir gizli anahtar dağıtılabilmektedir [19].

Güncel KAD sistemleri ile ilgili olarak referans [96]'da sunulan daha güncel verilere göre ise, ulaşılabilen en uzak mesafe 260 km olup bu gerçeklemede de gizli anahtar dağıtım hızı sadece 1.85 bps kadardır.

KAD'da gizli anahtar hızının belirlenmesinde etkili olan çeşitli parametreleri görmek için [3], [7] ve LDPC kodları ile CASCADE protokolünün karşılaştırıldığı [20]'deki ilgili açıklamalara bakılabilir.

1.2. Tezin İçeriği

Bu çalışma sırasıyla şu kısımlardan oluşmaktadır: 2. bölümde, tez çalışmasının dâhil olduğu alanlar ile ilgili ana konu başlıklarına değinilmektedir. 3. bölümde, BB84 (Bennett, Brassard, 1984) protokolü örneğinde, kuantum mekaniksel temelleri ve aşamaları ile KAD anlatılmaktadır. 4. bölümde, BB84 ve CASCADE protokolleri örneğinde, KAD'da BU konusuna değinilmektedir. 5. bölümde, CASCADE'in performansını arttırmaya yönelik önerilerimiz ve analizleri yer almaktadır. 6. bölüm, sonuç bölümümüzdür.

2. TEMEL KAVRAMLAR

Bu bölümde tez çalışmasının dâhil olduğu alanlar ile ilgili ana konu başlıklarına yer verilmektedir.

2.1. Bilgi Güvenliği

Bilgi güvenliği, bilginin işlenmesi, saklanması ve iletimi esnasında güvenliğinin sağlanmasıdır. Bu işlem, bilgiye yönelik başkası tarafından dinlenme, bilginin içeriğinin değiştirilmesi yanında taraflara yönelik kimlik taklidi ve inkâr etme gibi tehditlerin de ortadan kaldırılması ile sağlanır. Özellikle İnternet gibi dünyanın hemen her noktasına açık erişimin sağlandığı ağsal ortamlarda bilgi güvenliği can alıcı öneme sahiptir.

Birisinden bir mektup aldığımızda, mektubun açılıp başkaları tarafından okunup okunmadığından emin olamayız. Elektronik mektuplar ise hedeflerine ulaşmak için ağ üzerindeki onlarca bilgisayardan geçtiğine göre, mektuplarımızın gizliliği bu bilgisayarlardan sorumlu kişilerin insaflarına kalmıştır. Doğal olarak, bilgisayar ortamındaki verilerin okunup okunmadığına dair hiçbir şey bilmemiz mümkün değildir. Daha da önemlisi, aldığımız bir elektronik mektubun gerçekte kimden geldiğini ve yolda okunup okunmadığı yanında değiştirilip değiştirilmediğini öğrenmemiz de son derece zordur. Dolayısıyla, bilgilerin ve iletişimin gizliliğini ve güvenliğini sağlamak için gerekli tedbirlerin alınması gerekir. Kriptografi kullanımı bu tedbirlerin en başta gelenlerinden biridir.

Bilgi güvenliğinin sağlanması için kripto, donanım, yazılım, TEMPEST, ağ gibi kullanılan ortamlara yönelik öğeler yanında insan faktörü, doküman ve seçilen iletişim yöntemi gibi unsurların da göz önünde tutulması gerekir. Örneğin, pahalı yatırımlarla, yazılım ve donanım güvenliğininin sağlanmasına karşın personel, gizlilik konusunda yeterli eğitim ve bilinçle donatılamazsa bilgi ve haberleşme güvenliğinin sağlanması hayalden öte gidemez. Keza, her türlü önlemi almanıza ve hatta kripto kullanmanıza karşın, örneğin kriptolanan mesajların sonuna klasik "arz/rica ederim", "emirlerinizi beklerim" gibi tümceler koymak tüm güvenliği bir anda sıfırlamak anlamına gelebilir [21] (blok blok şifreleme kullanılması durumunda mesaj bloğu ve kriptolu hali bilinir).

2.2. Klasik Kriptografi

Kriptografi (aslen Yunanca olan “kryptós: gizli” ile “gráphein: yazı yazma” sözcüklerinin yan yana gelmesinden oluşmaktadır), kısaca, bilgiyi gizli tutma sanatı ve bilimidir. Matematiğin bir alt dalı olup güçlü matematiksel teknikler kullanarak bilgi güvenliğini garantilemeye yönelik çalışır. Bu amaç için tasarlanmış çeşitli algoritma ve protokollerden oluşmaktadır.

Modern kriptografinin bilgi güvenliğini sağlamaya yönelik olarak kullanıcılara sunduğu başlıca güvenlik servisleri şunlardır.

- Gizlilik

Aşağıda detaylı bir şekilde ele alınacaktır.

- Bütünlük

Bilginin saldırganlar tarafından değiştirilememesini sağlar. Doğru mesajın yerine yanlış mesaj konulmasını önler.

- Kimlik Doğrulama

Gönderici ve alıcının, birbirlerinin kimliklerini belirleyebilmelerini sağlar. Saldırganların, başkalarının kimliklerine bürünmelerini önler.

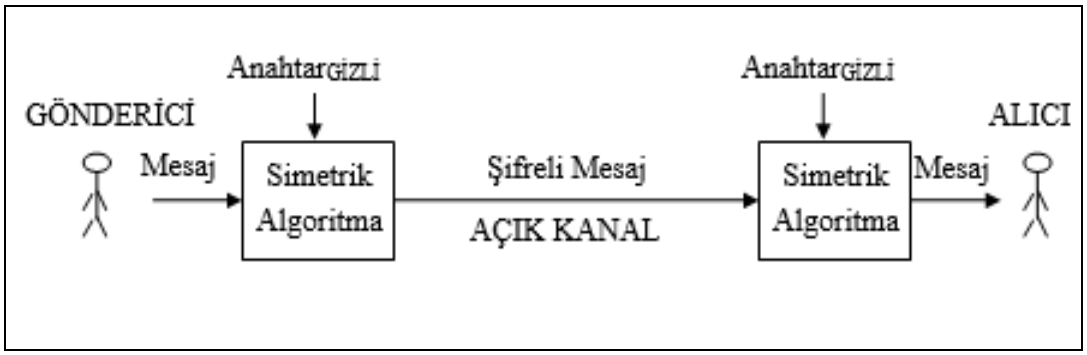
- Reddedilemezlik (İnkâr Edememe)

Bilgiyi oluşturan ya da gönderenin, daha sonra bilgiyi kendisinin oluşturduğunu veya gönderdiğini reddedememesini sağlar. Bir göndericinin bir ileti gönderdiğinde bunu daha sonrasında inkâr etmesini önler. Benzer şekilde, bir alıcının bir ileti aldığında bunu daha sonradan inkâr etmesinin önüne geçer.

İhtiyaç duyulan bilgi güvenlik düzeyine ulaşmak için pratikte bu servislerin birinden, birkaçından veya hepsinden birden yararlanmak gerekebilir.

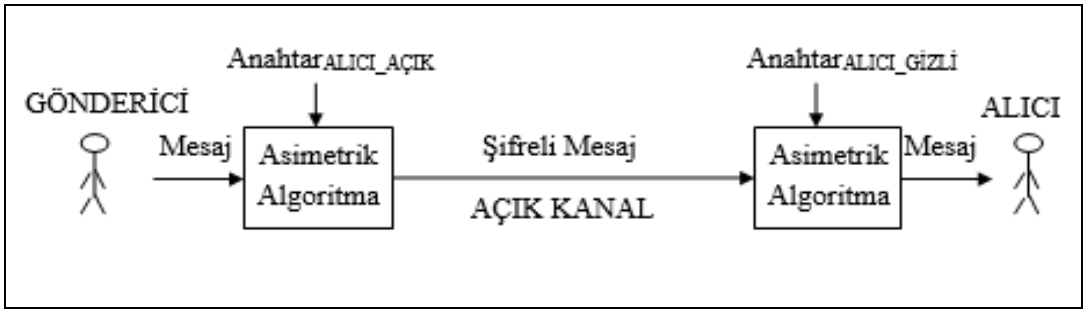
Kriptografinin gizlilik servisi bilginin gerçek alıcısı dışındaki kimseler tarafından anlaşılamamasını garantiler. Yani, örneğin, sizden başka hiçbir kimsenin size gönderilmiş olan bir elektronik mektubu asla okuyamaması gibi. Bu amaçla, kullanılan başlıca yöntem bilgiyi şifreleme (anlamsız hale getirme) ve şifre çözmedir

(tekrar anlamlı hale getirme). Günümüzde yaygın olarak kullanılan iki tür şifreleme sistemi bulunmaktadır [22]: Simetrik ve Asimetrik şifreleme sistemleri. Simetrik sistemlerde tek bir gizli anahtar vardır (bkz. Şekil 2.1). Hem gönderici hem de alıcı şifreleme ve şifre çözme için aynı gizli anahtarı kullanır. Simetrik sistemler oldukça hızlı olup şifrelemede öncelikli olarak tercih edilirler. Vernam şifresi, DES, AES, IDEA ve RC4 en çok bilinen ve kullanılan simetrik şifreleme algoritmalarıdır.



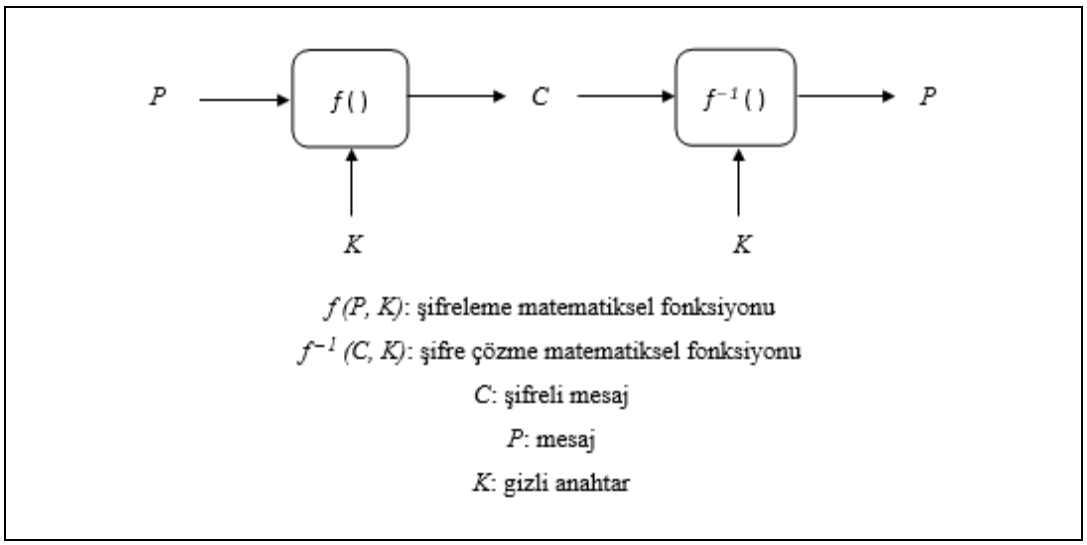
Şekil 2.1: Simetrik kriptografi.

Asimetrik sistemlerde ise açık anahtar ve gizli anahtar olarak adlandırılan iki farklı anahtar kullanılır (bkz. Şekil 2.2). Her kullanıcı bu anahtar çiftinden kendisine has olan bir tanesine sahiptir. Şifreleme için açık anahtar kullanılır ve herkese açıklanmasında bir mahsur yoktur. Şifre çözme için ise gizli anahtar kullanılır ve sahibi dışında başka hiç kimse bilmemelidir. Açık anahtarla şifrelenen bir bilgiyi sadece ilgili gizli anahtar, yani o açık ve gizli anahtar çiftinin sahibi, çözebilir. Tersi de geçerlidir; ancak, açık anahtar herkesçe bilindiğinden, gizli anahtarla şifrelenen bir bilgiyi herkes çözebilecek ve okuyabilecektir. Asimetrik sistemler yavaş olmaları nedeniyle daha çok kısa uzunluktaki mesajları şifrelemede tercih edilir. RSA, Diffie-Hellman, ElGamal ve DSS en çok kullanılan asimetrik algoritmalardır.



Şekil 2.2: Asimetrik kriptografi.

Yukarıda sözü geçen anahtar kelimeleri şifreleme ve şifre çözme işlemlerinde kullanılan elektronik bir bilgi (bit dizisi) anlamındadır. Anahtarlar olmadan şifreleme ve şifre çözme işlemlerini gerçekleştirmek mümkün değildir. Anahtarlar, şifreleme ya da şifre çözme algoritması ile birlikte kullanılarak iletilecek olan bilgiler gönderici tarafta önce şifrelenirler ve alıcı tarafta da tekrar çözülürler. Sözü geçen algoritma kelimeleri ise çeşitli matematiksel fonksiyonları ifade etmektedir. Şekil 2.1 ve Şekil 2.2'de görülen simetrik algoritma ve asimetrik algoritma kavramları aslında çeşitli matematiksel *f*() fonksiyonlarını (XOR, üs alma vd.) ifade etmektedir. Örneğin, yine yukarıda adı geçen Vernam şifresi ve RSA algoritmasının matematiksel fonksiyonlarını görmek için [23]'e bakılabilir. Dolayısıyla, şifreleme ve şifre çözme matematiksel olarak, kabaca, Şekil 2.3'teki gibi de ifade edilebilir.



Şekil 2.3: Şifreleme ve şifre çözmenin matematiği.

Algoritmaları, tanımlarını ve hatta gerçeklemelerini dahi rahatlıkla İnternet'te, kitaplarda, dergilerde vb. pek çok herkese açık ortamlarda bulmak mümkündür. Modern kriptografide algoritmaların gizli olmadığına dikkat ediniz. Algoritmalar herkese açık olup asıl gizlenen parametre, gizli anahtardır. Kerckhoff Yasası ("Gizli olması gereken sadece anahtardır." [24]) ve Shannon'a nispet edilen ünlü "Düşman, sistemi bilir." sözü de bunu ifade etmektedir [25].

Bilgi güvenliği, başkası tarafından dinlenme, bilginin içeriğinin değiştirilmesi, kimlik taklidi ve inkâr etme gibi hem bilginin hem de ilgili tarafların güvenliğini riske atan tüm tehditlerin ortadan kaldırılması ile sağlanır ve günümüzde bu amaçla kullanılan temel araç kriptografidir.

Kriptografi, konuları ve uygulamaları hakkında daha detaylı bilgi için referans [26]'ya başvurulabilir.

2.2.1. Anahtar Dağıtım Problemi

Gizli anahtarın güvenliği modern kriptosistemlerde ciddi bir meseledir. Bu sorun, anahtar dağıtım problemi olarak da bilinir. Hem modern simetrik hem de asimetrik kriptosistemler gizli anahtarların varlığına güvenirler; ancak, her iki kriptosistemde de esas problem gizli anahtarların gizliliğinin hiçbir zaman tam olarak garanti edilememesidir. Kırılamazlığı teorik olarak kanıtlanmış tek kriptosistem olan Vernam şifresi [27], [28], bilgi ile aynı uzunlukta gizli anahtarlar kullanır. Tek kullanımlık bu anahtarların her iki tarafta da olması gerekir. Anahtarın karşı tarafa güvenli olarak ulaştırılması simetrik kriptografide ciddi bir problemdir. Meraklı kişiler anahtar dağıtımı esnasında araya girerek bir şekilde anahtarın bir kopyasını ele geçirebilir. Ve bizim bunu anlamamız mümkün değildir.

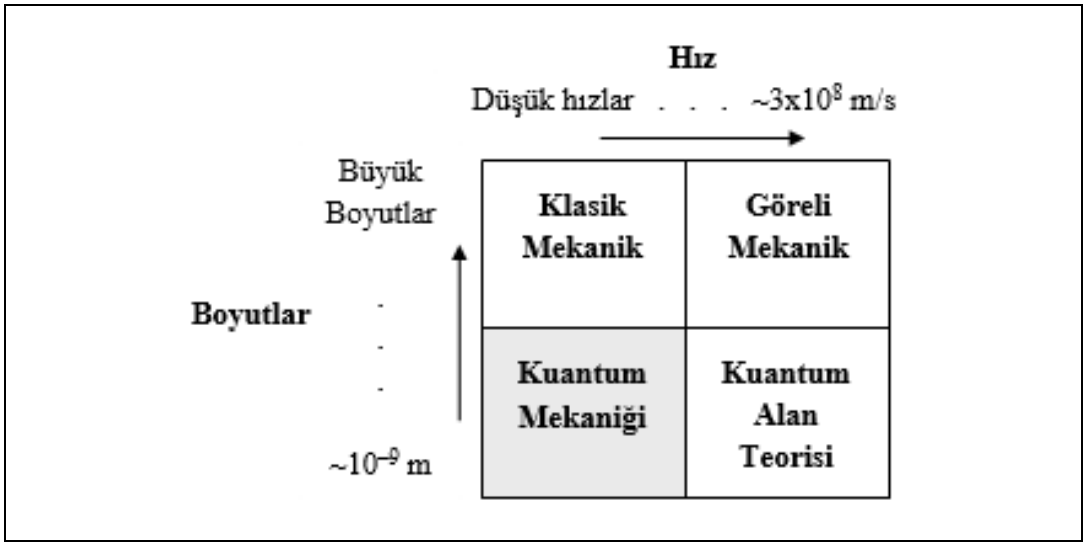
Asimetrik kriptosistemlerde ise durum daha da kötüdür. Örneğin, özellikle İnternet sayesinde dünya çapında yaygın kullanım alanı bulan RSA'de [29] açık anahtarda açıklanan bir sayıyı asal çarpanlarına ayırarak gizli anahtarı elde etmek mümkündür. Çok büyük olan bu sayıyı günümüz matematik bilgisi düzeyi ve bilgisayar hesaplama gücüyle makul bir sürede çarpanlarına ayırmak neredeyse imkânsızdır. Ancak, belki birileri bunu başarmış da olabilir. Ayrıca, eğer teknolojik gelişmelerdeki hız göz önüne alınırsa bunun oldukça riskli bir varsayım olduğu da açıktır. 10-15 yıl içinde geliştirilmesi ümit edilen kuantum bilgisayarın bu işlemi çok rahatlıkla gerçekleştirebileceği ispatlanmış durumdadır [30].

Dolayısıyla, bahsedilen anahtar dağıtım sorunlarının ve risklerinin olmadığı bir kriptosisteme ihtiyaç vardır. Bir çözüm, yepyeni bir alan olan ve kuantum mekaniği yasalarına dayanan kuantum kriptografidir.

2.3. Kuantum Mekaniği

Kuantum (aslen Latince olan "quantus: ne kadar" sözcüğünden gelmektedir) mekaniği, atomların ve atom-altı (çekirdek, elektron, foton gibi mikroskobik) parçacıkların tarifine olanak veren fizik yasalarının temelidir. Kuantum mekaniğinin

keşfi, ısıtılan cisimlerin ışımasını açıklamak için 1900 yılında önerilen Planck yasası (Max Planck: 1858-1947) ile başlar. Gelişimi, Albert Einstein (1879-1955), Niels Bohr (1885-1962), Werner Heisenberg (1901-1976), Max Born (1882-1970), John von Neumann (1902-1957), Paul Dirac (1902-1984), Wolfgang Pauli (1900-1958) gibi bilim insanlarının çalışmalarını da kapsayan 27 yıllık bir döneme yayılmıştır. 1927 yılında Erwin Schrödinger (1887-1961) tarafından Schrödinger denkleminin bulunmasıyla, esas olarak bugün öğrendiğimiz son halini almıştır. Kuantum mekaniğinin başarısı, Schrödinger denkleminin çözümlerinin doğanın özellikle mikro yapısında var olan pek çok deneysel gerçek ile tam uyuşumlu sonuçlar vermesine dayanır. Buna göre kuantum mekaniğinin temel varsayımları (postülaları), pek çok deneysel gerçeğin esasının ele alınmasıdır [31].



Şekil 2.4: Mekaniğin 4 büyük uğraş alanı.

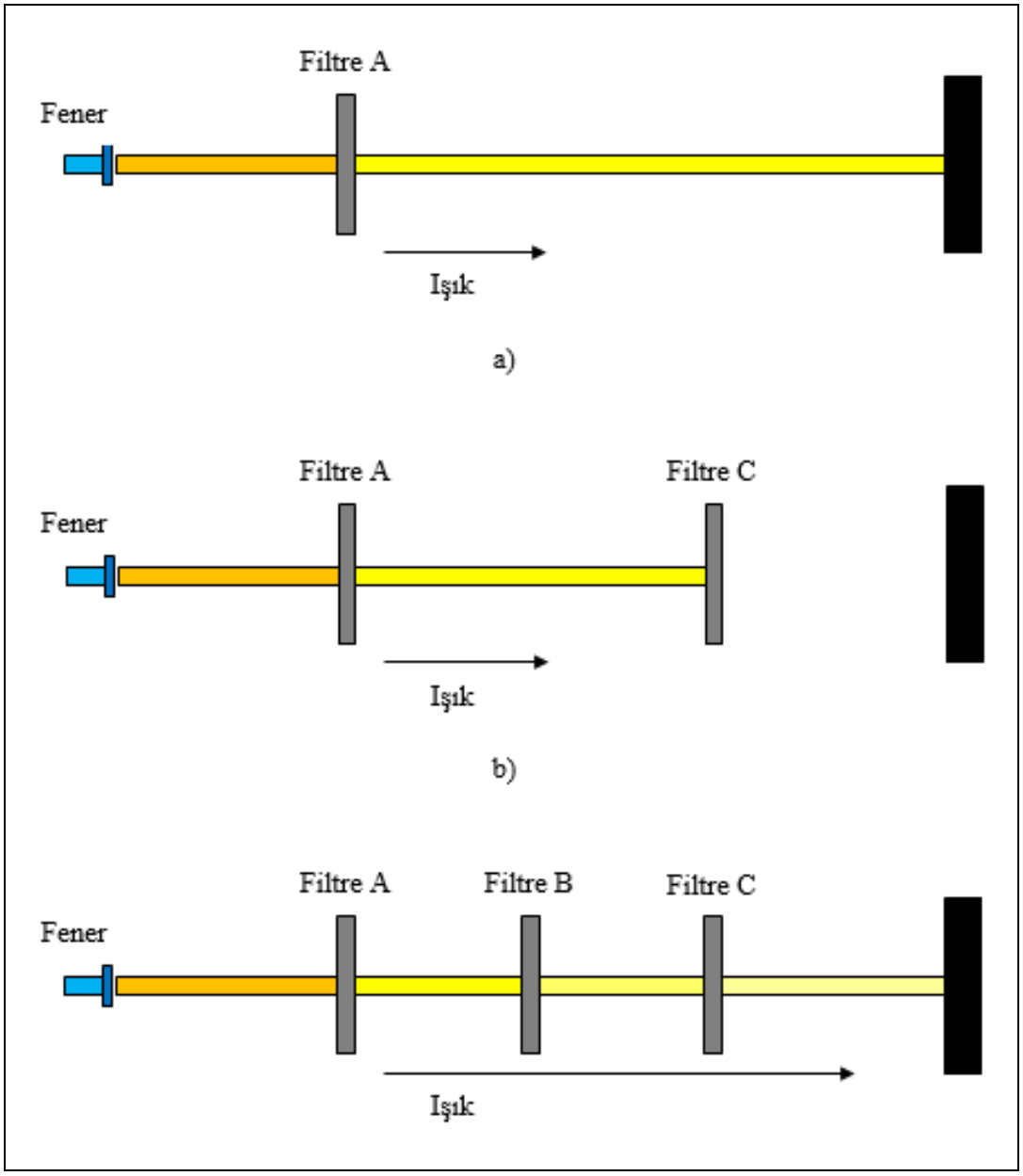
Burada hemen bir not olarak şu bilgileri paylaşmakta da yarar olabilir: "Kuantum" kelimesi ile foton, elektron, atom gibi temel tanecikler/yapıtaşları ifade edilmektedir. Ve eğer bir "olay"ın içinde bu taneciklerin, tane tane miktarlarda, kullanımları/varlığı söz konusu ise o olay da artık hemen "kuantum olay" olarak adlandırılmaktadır (örneğin, kuantum mekaniği, kuantum anahtar dağıtımı, kuantum kriptografi, kuantum kriptoloji, kuantum bilgi, kuantum bilgi teorisi, kuantum saklayıcı, kuantum bilgisayar vd.). Yine dikkat etmek gerekir ki bu taneciklerin boyutlarına inildiğinde doğa şu an bildiğimizden/alışageldiğimizden çok çok daha farklı işlemektedir. Dolayısıyla, kuantumlarla çalışırken onların dünyasının kurallarına göre oynamamız gerektiğini de unutmamalıyız.

Kuantum mekaniği, doğanın/hareketin yeni teorisidir. Bu teori atom, elektron, foton gibi mikroskobik sistemlerin davranışlarını açıklar (bkz. Şekil 2.4 [25]). 20. yüzyılın başlarına gelindiğinde artık klasik mekanik yasaları ile açıklanamayan bir dizi gözlem bulunmaktaydı. Üstelik bu gözlemlerin anlaşılması için, klasik fizikte yeri olmayan, ışığın parçacık (tanecik) özelliği, kütleli parçacıkların dalga karakteri göstermesi (maddenin dalga özelliği) ve hemen hemen tüm fiziksel niceliklerin kesikli (kuantumlu) yapısı gibi yepyeni kavramlardan söz edilmekteydi ve bunlar temel kavramlar olarak kullanılıyordu. Problemler, özellikle atom ve elektron gibi çok küçük kütleli ve çok yüksek hızlı cisimlerin işe karıştığı ve bunların ışık ve elektromanyetik alanlar ile etkileşim süreçlerinde ortaya çıkmaktaydı. Bu olayların en önemlileri şunlardır: Siyah cisim ışıması, Katıların ısı sığası, Fotoelektrik olay, Compton olayı, Elektronlarla kırınım, Atomların ışıma ve soğurma spektrumları. Başlangıçta bu olaylar, amaca uygun özel (ad hoc) ve o zamanlar garip görünen bir takım varsayımlarla açıklandı. Zamanla bunların başka olaylar için de geçerli olabileceği öngörüldü. Bu varsayımların sayıları arttıkça ve aralarındaki ilişkiler belirginleştikçe artık mekaniğin yepyeni bir formülasyonu gerekti. 1920'li yılların ikinci yarısına gelindiğinde, sayıca artmış bütün varsayımlar üzerine kurulu yeni bir hareket teorisi gerekiyordu. Sonuç, kuantum mekaniğidir.

Kuantum mekaniği, parçacık-düzeyi bir fiziktir ve her gün alışageldiğimizin dışındaki kavramlarla ilgilenmektedir. Sıradışı kuantum etkiler çok özel ekipmanlar olmadan gözlemleyemeyeceğimiz atomik ölçekte meydana gelmektedir. Bu etkileri görmek için elektron, foton gibi parçacıkların gözlenmesi gerekir.

Kuantum mekaniğinin dünyası, kuantum mekaniksel etkileri göstermede kullanılabilecek az sayıdaki örnekten biri olan Şekil 2.5'teki filtreler deneyi ile belki daha kolayca açıklanabilir [32]: Bir el fenerimiz ve A, B, C diye etiketlenmiş 3 tane de polarizasyon filtremiz olsun. Filtrelerin polarizasyonları da sırasıyla yatay, 45° ve dikey olarak ayarlanmış (döndürülmüş) olsunlar. Şimdi el fenerini duvara tutalım ve duvarla fener arasına A filtresini koyalım (bkz. Şekil 2.5.a). Filtreden sadece yatay polarizasyonda olan fotonlar (ışık tanecikleri) geçebilecek ve duvara ulaşacaklardır. Şimdi A filtresi ile duvar arasına C filtresini koyalım (bkz. Şekil 2.5.b). C filtresinden sadece dikey yönelimde olan fotonlar geçebileceği için duvara hiç ışık ulaşamayacaktır. Bu iki filtre duvara ulaşan tüm ışığı kesmiştir. Şimdi A ve C arasına B filtresini koyalım (bkz. Şekil 2.5.c). Şimdi duvarda yeniden ışık görülecektir. Klasik olarak ilave filtrenin ışık geçirmeyi daha da zorlaştırması beklenirken, garip

bir şekilde, ışığın geçmesini sağlamaktadır. İşte bu, bir tür kuantum mekaniksel davranıştır ve kuantum dünyası bunun değişik örnekleriyle doludur.



Şekil 2.5: Filtreler deneyi. a) Duvara sadece A filtresinden geçebilen ışık ulaşabilir, b) Araya farklı bir C filtresi daha eklenince duvara hiç ışık ulaşamaz, c) A ile C arasına daha farklı bir B filtresi eklenince ışık tekrar duvara ulaşıyor.

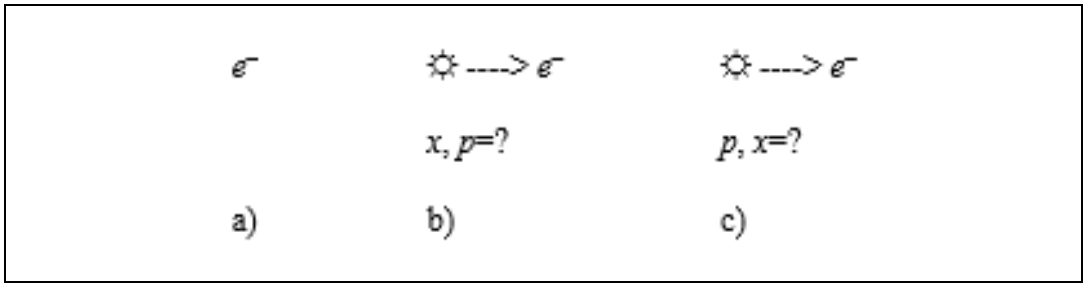
Klasik olarak bir benzeri bulunmayan hassas kuantum fenomenleri (anlaşılması güç olay, gariplik), bilginin ışınlanması, kırılamaz varsayılan kodların kırılması, gerçek rasgele sayıların üretilmesi ve arada iletişim hattını dinleyen birinin varlığını açığa çıkaran mesajlarla haberleşme gibi daha önce bir benzerleri görülmemiş işlemlerin gerçekleştirilmesine de izin vermektedir.

Kuantum mekaniği, konuları ve uygulamaları hakkında daha detaylı bilgi için referans [33]'e başvurulabilir.

2.3.1. Heisenberg Belirsizlik İlkesi

1927 yılında Alman fizikçi Werner Heisenberg, kuantum sistemlere ait belirli fiziksel büyüklük çiftlerinin aynı anda doğru olarak ölçülebilmeleri ile ilgili temel bir sınır keşfetmiştir. Heisenberg'in belirsizlik ilkesine göre, bu fiziksel büyüklük çiftinden birinin değeri ne kadar doğru olarak bilinirse, diğerinin değeri o kadar belirsiz olmaktadır [34], [35].

Heisenberg'in serbest bir elektronun (e^-) belirli bir anda konumunun (x) ve momentumunun (p) ölçülmesine ilişkin belirsizlik ilkesi şöyledir. Burada elektron sözü edilen kuantum sistem, konum ve momentum ise sözü edilen fiziksel büyüklük çifti olmaktadır. Elektronun konumunu ve momentumunu ölçmek için ona, istediğimiz bilgiyi bize geri taşıyacak bir etki uygulamamız gerekir. Farz edelim ki elektrona Şekil 2.6'daki gibi dalga boyu λ olan bir ışıkla bakıyoruz (p ve x'in elde edilmesi yeteneği λ ile karşılaştırılabilirdir).



Şekil 2.6: Heisenberg belirsizlik ilkesi. a) serbest bir elektron, b) konum bilgisinin ölçümü, c) momentum bilgisinin ölçümü.

Bu ışığın fotonlarının her birinin momentumu

$$p = h / \lambda \tag{2.1}$$

kadardır (h: Planck sabiti, bugün iyi bilinen doğanın temel sabitlerindendir). Bu fotonlardan biri elektrona çarpıp geri sıçrayınca, ki elektronu görmek için bu olmalıdır, elektronun ilk konumunu ve ilk momentumunu değiştirecektir. Çarpma

sonucu oluşan momentumdaki Δmv değişiminin kesin değeri önceden bilinemez, fakat fotonun h / λ momentumu ile aynı mertebede olacaktır. Dolayısıyla,

$$\Delta mv \approx h / \lambda \tag{2.2}$$

dır (m: elektronun kütlesi, v: elektronun hızı). İfadeye göre dalga uzunluğu ne kadar büyük olursa momentumdaki belirsizlik o kadar küçük olacaktır. Dolayısıyla, elektronun momentumunu ölçmek için uzun dalga boylu ışık kullanılabilir.

Benzer şekilde, elektronun konumu da mükemmel doğrulukta belirlenemez. Ölçüdeki ihmal edilemeyen ve yine fotonun çarpmasından kaynaklanan Δx belirsizliği için de makul tahmini bir dalga boyu olabilir. Yani,

$$\Delta x \approx \lambda \tag{2.3}$$

Buna göre, dalga boyu ne kadar küçük ise konumdaki belirsizlik o kadar küçük olmaktadır. Bu nedenle, konum ölçümünün doğruluğunu arttırmak için kısa dalga boylu ışık kullanılabilir. Ancak, böyle yapılması durumunda momentum ölçümünün doğruluğunda azalma meydana gelecektir.

Kısa dalga boylu ışık doğru konum fakat doğru olmayan bir momentum bilgisi verirken (Şekil 2.6.b), uzun dalga boylu ışık doğru momentum fakat doğru olmayan bir konum verecektir (Şekil 2.6.c). Yukarıdaki iki eşitlik,

$$\Delta x \cdot \Delta mv \approx h \tag{2.4}$$

verir ki ilk olarak Werner Heisenberg tarafından elde edilmiş olan belirsizlik ilkesinin bir şeklidir. Bu ilke bir cismin belirli bir anda konumundaki Δx belirsizliği ile aynı anda momentumundaki Δmv belirsizliğinin çarpımının yaklaşık olarak Planck sabitine eşit olduğunu ifade eder. Yani, konum ve momentumun ikisini birden aynı zamanda mükemmel bir doğrulukla ölçmek imkânsızdır.

Belirsizlik aletlerde değil doğadadır. Makroskobik ölçekte h'nin çok küçük olması nedeni ile belirsizlik ilkesi tarafından ölçüler üzerine konulan sınırlamalar ihmal edilebilir (Klasik fizikte, $h = 6.63 \times 10^{-34}$ Joule.saniye gibi değerlerin yaklaşık

olarak 0 kabul edildiğini ve dikkate alınmadığını hatırlayın). Fakat mikroskobik ölçekte belirsizlik ilkesi (ve bizim yaklaşık olarak 0 kabul ettiğimiz ve ihmal ettiğimiz değerler) pek çok olayı etkisi altına almaktadır.

Fiziksel büyüklük çiftinden birinin değeri ölçülürken, diğerinin değerinde meydana gelen bu kaçınılmaz karışıklık, böyle sistemlerin kopyalanamayacağını gösterir ve kuantum kriptografiye anahtar teşkil eder.

2.3.2. Kuantum Kopyalanamazlık Teoremi

1982 yılında Amerikalı teorik fizikçi William Kent Wootters ve yine Polonya doğumlu Amerikan vatandaşı teorik fizikçi Wojciech Hubert Żurek (1951) tarafından ifade edilen teorem [36], kuantum dünyasında kopyalama yapmanın imkânsız olduğunu söyler. Buna göre, kuantum kopyalama makineleri var olamayacağından kuantum kopyalara sahip olmak mümkün değildir.

Kopyalanamazlık teoreminin bir uygulaması olarak, bilgiyi bir kuantum sistemin (atom, elektron, foton vd.) bir durumu (spin, polarizasyon vd.) ile temsil ettiğimizi düşünelim. Teoreme göre, bilinmeyen bir kuantum durum kopyalanamayacağı için, bu kuantum durumla temsil edilen kuantum bilginin de kopyalanamayacağı anlamına gelir. Böylece, kuantum bilginin bu negatif yeteneği nedeniyle saldırganlar mükemmel bir dinleme yapamazlar.

Kuantum bilginin kopyalanamaması ve dinlenememesi gibi negatif özellikleri, bu ilkelere dayanan kuantum kriptografiyi potansiyel olarak güvenli yapması gibi olumlu bir sonuca da yol açmaktadır.

2.4. Kuantum Kriptografi

Kuantum kriptografi, bilgi güvenliğinin kuantum mekaniğine ait (belirsizlik ilkesi, kopyalanamazlık teoremi gibi fiziksel) yasalar ile garanti edildiği kriptografi tekniğidir [37]-[43]. Temel avantajı, matematik yerine fiziğe, birtakım matematiksel varsayımlar yerine kanıtlanmış evrensel kuantum mekaniği kanunlarına, dayanıyor olması, güvenliğin ispatlanabilir olmasıdır.

En bilinen ve ilk pratik uygulaması KAD'dır. Kuantum mekaniğinin temel kuramlarından olan belirsizlik prensibinin ve kopyalanamazlık teoreminin

komünikasyona uygulanmasıyla fark edilmeden dinlenilmesi imkânsız veya imkânsıza yakın iletişim hatları mümkün olmuştur [44]. Vernam şifresinin ihtiyaç duyduğu anahtarlar bu güvenli hatlar üzerinden güvenle dağıtılabilir.

Kuantum kriptografi, bugün de halen çoğunlukla anahtar dağıtımı ile ilgilidir. Bu nedenle, yakın zamana kadar kuantum kriptografi ve KAD aynı anlamda kullanılmaktaydı. Doğrusu, kuantum kriptografinin KAD'ı da içine alan daha geniş bir disiplin olduğudur. Bir genelleme yapmak gerekirse, kuantum bilgisayara karşı tüm bilgi güvenliğini sağlama çalışmaları (klasik ya da kuantum olsunlar) kuantum kriptografinin kapsamı içinde kabul edilmektedir [45].

Kuantum kriptografiyi diğer kripto sistemlerinden farklı kılan, güvenli ve devamlı anahtar dağıtımının garantilenmesidir. Mevcut kuantum kriptografi, şimdilik klasik ve kuantum kısımlardan oluşmaktadır.

- Kuantum Kısım

  KAD.

- Klasik Kısım

  Geleneksel kriptografi ile şifreleme.

Kuantum kriptografide temel prensip ise Vernam şifresi tekniğinin kullanılmasıdır. İspatlanabilir (koşulsuz, asla kırılamaz) güvenlikte bir iletişim için kuantum kriptografinin günümüzdeki temel çalışma prensibi şöyledir.

- Anahtar, taraflar arasında KAD ile dağıtılır. KAD, güvenliği ispatlı, tamamen güvenli tek anahtar dağıtım yöntemidir.
- Şifreleme, Vernam şifresi ile yapılır. Vernam şifresi kırılamazlığı, yine bilgi teorisi kullanılarak, kanıtlanmış tek şifredir.

Dolayısıyla, kuantum kriptografide anahtar dağıtım problemi KAD ile çözülmektedir. Vernam şifresinin de kullanılmasıyla, tamamen ispatlanabilir güvenlikte bir iletişim ortamı da garanti edilmiş olmaktadır.

Kuantum kriptografi, konuları ve uygulamaları hakkında daha detaylı bilgi için referans [46]'ya başvurulabilir.

2.4.1. Kuantum Rasgele Sayı Üretimi

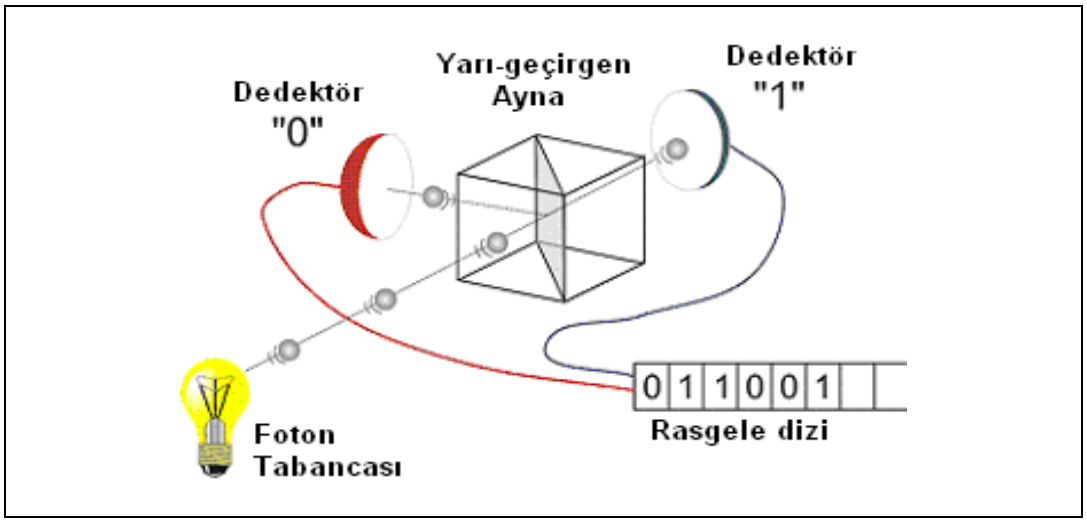
Rasgele sayı dizini üretimi klasik kriptografide olduğu gibi kuantum kriptografinin de temel öğelerindendir. Yine, KAD'ın, güvenliği için, en temel gereksinimlerinden biri kaliteli bir rasgele sayı üretecidir.

Rassal sayılar, kriptografiden istatistiğe, benzetime, örneklemeye, sayısal analize, şans oyunlarına kadar bugün pek çok önemli uygulamada yoğun olarak kullanılmaktadır. Bahsedilen uygulamaların hepsinde de uygulamanın başarımı açısından rasgele sayıların önemi büyüktür ve hepsinin kalite gereksinimleri de farklıdır. Modern kriptografide, gizliliğin doğrudan bağlı olduğu kriptografik algoritma ve protokol parametrelerinin (tohum, ilklendirme vektörü, sorgular vb.) ve gizli oturum anahtarlarının (sadece o oturuma özel olan tek kullanımlık şifreleme ve şifre çözme anahtarları) oluşturulmasında merkezi bir rol oynarlar.

Günümüzde yeterince kaliteli ve hızlı rasgele sayıları üretmek üzere 2 temel üreteç türü mevcuttur: gerçek rasgele sayı üreteci (GRSÜ) ve sözde rasgele sayı üreteci (SRSÜ). GRSÜ'ler fiziksel bir rasgelelik kaynağı kullanırlar ve daha çok da ürünlere daha sonradan eklenen ayrı bir donanım olarak tasarlanırlar. SRSÜ'ler ise yazılımsal olarak gerçeklenirler, bilgisayar ya da ürün içinde koşan ayrı bir algoritma/program olarak tasarlanırlar. SRSÜ'ler, GRSÜ'ler kadar kaliteli/güvenilir olmasalar da daha basit, ucuz, hızlı ve esnek olmaları nedeniyle daha çok tercih edilirler. Bununla birlikte, SRSÜ'nün durum parametresi fiziksel bir kaynaktan elde edilen entropi ile en başta tohumlanmalı ve daha sonra da güvenlik için periyodik olarak tekrar tekrar tohumlanmaya devam edilmelidir. SRSÜ'lerde tüm entropi bu tohum denilen parametrede yer almaktadır. Bu nedenle, SRSÜ kullanılması durumunda bile tohumu beslemek üzere GRSÜ'lere, en azından basit ama güvenilir bir GRSÜ'ye, de ihtiyaç olmaktadır. Günümüz GRSÜ'lerinin hızlarının henüz çok yüksek olmaması, çok hassas olup çalıştıkları ortamın koşullarından çok fazlaca etkilenebilmeleri, zaman zaman hata/arıza da yapabilmeleri SRSÜ kullanımını daha cazip kılan diğer başlıca nedenlerdir [47]-[50].

Yukarıdaki açıklamalardan da görüldüğü üzere, mevcut rasgelelik kaynakları, rasgeleliğin doğasına göre iki ana gruba ayrılabilir: sadece sözde rasgele bit dizileri üretebilen yazılım çözümleri ile gerçeksi rasgele bit dizileri üretebilen fiziksel kaynaklar. Burada bilinmesi gereken önemli nokta, hem klasik bilgisayarların hem

de klasik fiziğin tamamen deterministik olduğudur. Sonuç olarak, her iki rasgelelik üreteci de rasgele gibi görünen bir bit dizisi üretmek için tamamen deterministik olan klasik fiziğe dayanır. Dolayısıyla, aslında, her iki üreteç türü için de tam bir rasgelelikten söz edilemez.



Şekil 2.7: Kuantum rasgele sayı üretimi.

Bir kuantum rasgele sayı üreteci (KRSÜ), rasgele sayıları üretmek için kuantum fiziğini kullanan bir üreteçtir. Klasik fiziğin aksine kuantum fiziği deterministik değildir. Kuantum mekaniğinde tamamen rasgelelik hâkimdir. Ve KRSÜ'ler de rasgelelik kaynağı olarak kuantum dünyasının bu içsel rasgeleliğini kullanır. Rasgele sayılar üretmek için çok basit birtakım kuantum prosesleri, bunlardaki içsel rasgeleliği, bile kullanmak mümkündür. Örneğin, Şekil 2.7'de [25] şu an ticari bir ürün haline de gelmiş olan basit bir KRSÜ görülmektedir.

Quantis adı verilen bu KRSÜ, rasgelelik kaynağı olarak optiksel bir kuantum proses kullanır. Optik, ışığın bilimidir. Kuantum fiziği bakış açısından ışık, foton adı verilen temel parçacıklardan oluşur. Fotonlar belirli durumlarda rasgele bir davranış sergilerler. Örneğin, ikili rasgele sayıların üretimine de çok iyi uyan böylesi bir durum, fotonların yarı-geçirgen bir aynadan geçişleridir. Bir fotonun yarı-transparan bir aynadan geçmesi ya da yansıması tamamen rasgele bir olaydır, başka herhangi bir parametreden de etkilenmez. Şekil 2.7'deki optik sistemde de olduğu gibi yarı-gümüşlenmiş bir aynaya tek tek fotonlar göndermek ve geçtiklerini ya da yansıdıklarını ölçmek sonucunda elde edilen 0 ve 1 katarı gerçek rasgeledir. Yarı-geçirgen bir aynaya fotonlar göndermek ve geçtiklerini ya da yansıdıklarını ölçmek,

kuantum belirsizliğini kullanmanın sadece bir yoludur. Konum, momentum, elektrik alan gibi başka fiziksel büyüklüklerdeki herhangi bir belirsizlik de aynı şekilde işimizi görebilecektir. Sonuçta, elde edilen 0 ve 1 dizisi gerçekten rasgeledir.

Sonuç olarak, kuantum rasgele sayı üreteçleri tek gerçek rasgelelik üreteçleridir. Rasgele bitleri üretmek için kuantum proseslere güvenir. Klasik fiziğin aksine, kuantum fiziği tamamen rasgeledir. Genel kanı, rasgeleliğin en iyi kaynağının belirsizliğin tek geçerli olduğu yer olan kuantum dünyası olduğu yönündedir. Buradaki başlıca problemlerden biri ise gerçek rasgele sayıların üretilebileceği hızdır. Fotonlar ve yarı-geçirgen aynalarla çalışan ticari cihazlar için şimdilik 16 Mbps hıza kadar rasgele sayı üretimi yapılabilmektedir [51].

2.5. Bilgi Teorisi

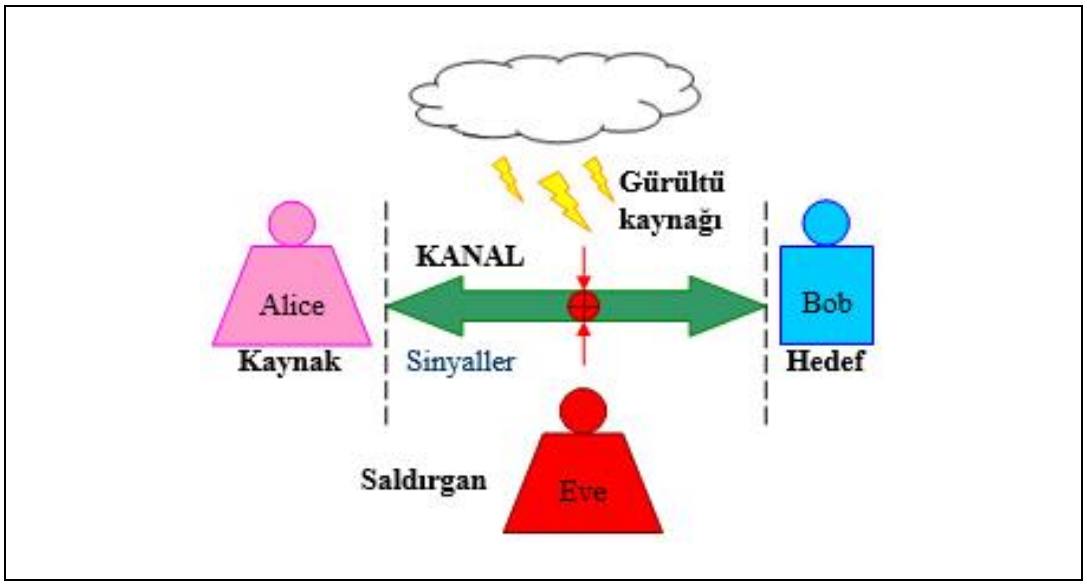
Bilgi teorisi, haberleşmenin matematiksel teorisi, haberleşmedeki (sıkıştırma, iletim ve güvenlik ile ilgili) temel limitleri belirleyebilmek üzere 1948 yılında Claude Elwood Shannon (1916-2001) tarafından geliştirilmiştir [52].

Haberleşme sistemlerinin temel amaçlarından ikisi, bilgiyi bir KANAL (iletim/taşıma ortamı) aracılığıyla Kaynak (bilgi kaynağı, gönderici, verici, 1. kişi, Alice) adı verilen bir uçtan Hedef (alıcı, 2. kişi, Bob) adı verilen başka bir uca/noktaya güvenilir ve güvenli bir şekilde taşımaktır. Burada,

- güvenilirlik ile gizli (mahrem, hassas, özel) bilginin içeriğinin dosdoğru, bozulmasız bir şekilde iletilmesi,
- güvenlik ile de gizli bilginin içeriğinin 3. kişilerin (Eve, düşman, saldırgan, meraklı) eline asla geçmemesi

kastedilmektedir. Haberleşme sistemlerinin diğer amaçlarına örnek olarak ise verimlilik, hız, maliyet vb. de verilebilir.

Şekil 2.8'de [53], pratik bir haberleşme sistemi modellenmektedir. Burada, kanal olarak CD/DVD, sabit disk, RAM vb. bellek ortamları kullanılırsa yapılan iş depolama adını alır. Depolama işlemi de bir tür haberleşme olarak düşünülebilir. Bu durumda da bilgi bir ortama yazılmakta ve daha sonra da o ortamdan tekrar okunmaktadır.



Şekil 2.8: Haberleşme sistemi basit bir blok diyagramı.

Bizler kullanıcılar olarak bir iletişimde ya da depolama işleminde güvenilirlik ve güvenlik konularından tam olarak emin olmak isteriz. Ancak, uygulamada bu amaçlara ulaşmak hiç de kolay değildir. Şekil 2.8'de de gösterilmeye çalışıldığı gibi, sisteme iki olumsuz bileşen daha dâhil olmaktadır.

- Gürültü

Mevcut bütün iletim (ve depolama) ortamları gürültülüdür. Gürültü, hatalara, bozulmalara neden olur.

- Saldırgan (Eve, Oscar, Rakip, Kriptanalizci, Hain vb. 3. Kişiler)

Mevcut bütün (klasik) iletim (ve depolama) ortamları pasif saldırganlarca dinlenebilir ve iletişimin dinlendiği anlaşılamaz.

Pratikte, güvenilirlik ve güvenlik hedeflerimize ulaşabilmek için bu olumsuzluklara karşılık biz de sisteme iki yeni olumlu bileşen ekleriz.

- Hata Sezme ve Düzeltme Teknikleri (Kanal Kodlama)

Kodlama servisi ile bilgiyi kodlarız, bozulursa anlayıp düzeltebilelim diye.

- Kriptografik Teknikler (Kriptografik Kodlama)

Şifreleme servisi ile bilgiyi şifreleriz, içeriğini başkaları asla anlayamasın diye.

Enformasyon teorisi, mesaj sinyallerinin içerdiği enformasyon miktarını belirlemek için nicel bir ölçü sağlayarak bu amaçlara ulaşılabilmesini olanaklı kılmıştır. Enformasyon teorisinin sunduğu araçlar kullanılarak,

- Veri sıkıştırmanın limiti nedir?
  - Cevap: Entropi H(veri kaynağı) bit/simge
- Haberleşmenin iletim hızının limiti nedir?
  - Cevap: Kanal kapasitesi C = maksimum ortak bilgi $I(X, Y)$ bit/simge
- Eğer Eve şifreli mesajın bir kısmını görürse, şifreleme anahtarı hakkında daha önce sahip olmadığı yeni bir bilgi elde eder mi?
  - Cevap: Koşullu entropi H(anahtar | şifreli mesaj) bit/simge
- Eğer Eve şifreli mesajın bir kısmını görürse, mesaj hakkında daha önce sahip olmadığı yeni bir bilgi elde eder mi?
  - Cevap: Koşullu entropi H(mesaj | şifreli mesaj) bit/simge

sorularına sayısal cevaplar vermek mümkün hale gelmiştir. Yukarıda, $H()$ ve $I()$ fonksiyonlarının (ortalamalarının) birimleri bit/simge olup parametrelerinin her biri ilişkili-ilişkisiz rasgele (olasılıksal) değişkenlerdir.

Bilgi teorisi, konuları ve uygulamaları hakkında daha detaylı bilgi için referans [54]'e başvurulabilir.

3. KUANTUM ANAHTAR DAĞITIMI

Bu bölümde, BB84 protokolü örneğinde, kuantum mekaniksel temelleri ve aşamaları ile kuantum anahtar dağıtımı (KAD) anlatılmaktadır.

3.1. BB84 Protokolü

KAD, atanmış bir fiber optik kablo hattı ile birbirine bağlı (kablolu) olan ya da birbirinin görüş alanında (kablosuz) olan iki taraf için uçtan uca anahtar dağıtımı imkânı sağlar.

Simetrik kriptografinin kullanımı için iki taraf arasında ortak ve gizli anahtarlara ihtiyaç vardır. Bu anahtarlar, KAD kullanılarak ispatlanabilir bir güvenlik ile elde edilebilirler.

KAD, ya da daha genel olarak kuantum kriptografi, güvenlik modelinin anahtar yönü olarak matematiğin aksine daha fazla fiziğe güvenmesi bakımından geleneksel kriptografik sistemlerden farklıdır. Bu teknikte bilgi fizik yasaları ile korunmaktadır: tek tek kuantum parçacıkların ve bunların kendilerine özgü kuantum özelliklerinin kullanımına (temel olarak, herhangi bir sistemin o sistemi rahatsız etmeden kuantum durumunu ölçmenin imkânsız olması özelliğine) dayanır. Bu yasalar fizik alanındaki ve teknolojik gelişmelerden de olumsuz yönde etkilenmezler.

Burada, daha kolay hatırda kalması açısından, kısaca özetlemek gerekirse, kuantum dünyasında

- Gözlem/Müdahale, deneyi etkiler
- Gözlem/Müdahale, gözleneni/müdahale edileni etkiler

iki ana kuralının geçerli olduğu söylenebilir. İfadelerde gözlenen/müdahale edilen ile kuantumlar (foton, elektron vd. kuantum tanecikler) kastedilmektedir. Dolayısıyla, kuantumlarla çalışırken bu hususların dikkate alınması, sistemlerin buna göre tasarlanması gerekir. KAD, böylesi uygulamaların şu an için belki de en bilineni, pratik olanıdır ve başında gelenidir.

KAD'da anahtar dağıtımı herkese açık optik kanallar (fiber optik kablo ya da açık hava) üzerinden gerçekleştirilir. Optik iletişimin aksine iletişim için tek tek

fotonlar kullanılır; bu nedenle, kuantum mekaniksel bir iletişim tekniğidir. İletişim esnasında optik kanala müdahale olup olmadığını açığa çıkarabilmekte, dinlenemez ve akan trafiği kopyalanamaz bir iletişim olanağı vermektedir.

Foton, ışığı oluşturan taneciklerinden her birisine verilen addır. Teorik olarak, fotonlardan başka kuantum parçacıkların (mesela, elektronların) kullanılması da mümkündür; ancak, fotonlar gerekli olan bütün özellikleri sağlarlar ve davranışları nispeten daha iyi anlaşılmış durumdadır (tek tek elektronları ortam ile etkileşime girmeden ve uzaklara iletmek fotonlara göre daha zor ve maliyetlidir). Ayrıca, çok yüksek bant genişlikli ve hızlı iletişimler için en umut verici ortam olan optik fiber kablolarda da en temel bilgi taşıyıcılardır (elektron hızı, foton hızı olan $c = \sim 3\times10^8$ m/sn mertebelerine kadar çıkarılabilse de daha düşüktür).

Dalga özelliği de gösterebilen fotonların bir biti kodlamada kullanılabilecek çeşitli özellikleri bulunmaktadır: polarizasyon, dalga boyu, frekans, faz gibi. KAD protokollerinde çoğunlukla polarizasyon kullanılır.

BB84 protokolü, 1984 yılında ABD'li fizikçi Charles Henry Bennett (1943) ve Kanadalı kriptocu Gilles Brassard (1955) tarafından keşfedilen ilk KAD yöntemidir [55]. Protokol, kâşiflerinin soyadlarının baş harfleri ve 1984 yılının kısaltmasından ötürü BB84 protokolü diye adlandırılmaktadır.

BB84 protokolü, 1989 yılında laboratuvar ortamında [56] sadece 30 cm'lik bir açık hava mesafesi için ve 10 bit/saniye hızında [57] gerçekleştirilebilmiş iken mevcut mesafe limitleri fiber optik kablo için 260 km [58], açık hava için 144 km [59] ve hızlar ise 1 Mbps'yi bulmuştur [60].

BB84 protokolünde gizli anahtar bitlerini güvenlice taşımak için fotonların polarizasyon özelliğinden yararlanılır.

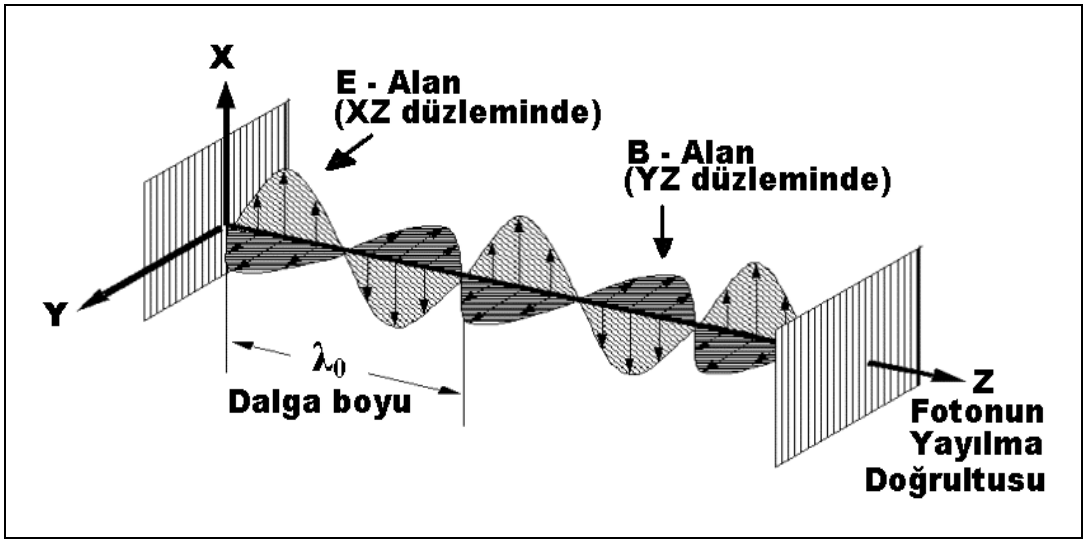
KAD, protokolleri ve yaşanan gelişmeler hakkında daha detaylı bilgi için referans [61]'e başvurulabilir.

BB84 protokolü, başlıca detaylarıyla aşağıda anlatılmaktadır [23]. Diğer KAD protokolleri de benzerdir.

3.1.1. Foton Polarizasyonu

Fotonun bir biti kodlamak için kullanılabilecek özelliklerinden biri polarizasyon durumudur. Fotonun polarizasyon özelliği, elektromanyetik alanı içinde

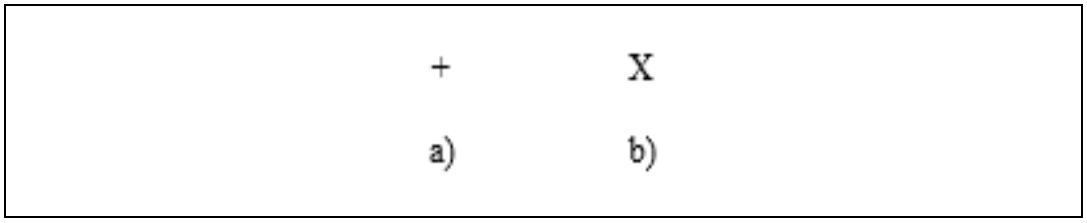
elektrik alanının (*E*) salındığı düzlem ile ilgilidir. Şekil 3.1'de foton pozitif *z*-ekseni yönünde yayılırken elektrik alanı *x*-*z* düzleminde ve manyetik alanı (*B*) ise *y*-*z* düzleminde salınmaktadır.



Şekil 3.1: Foton, elektrik alanı, manyetik alanı.

Doğrusal polarizasyon, foton yayılırken elektrik alanın hep aynı düzlemde kalması demektir. Dairesel polarize olmuş ışıkta ise, foton yayıldıkça elektrik alan belirli bir frekansta sürekli döner.

Kuantum kriptografi, örneğin KAD, doğrusal, dairesel ya da her ikisinin kombinasyonu olarak polarize edilmiş ışıkla gerçekleştirilebilir. Açıklaması daha basit olduğu için bundan sonraki incelemeler doğrusal olarak polarize edilmiş ışığı kullanan uygulamalarla sınırlanacaktır.



Şekil 3.2: Polarizasyon tabanları. a) Kenarsal taban, b) Köşegensel taban.

Doğrusal polarizasyonda bir fotonun kodlanmasında taban olarak alınan iki temel polarizasyon türü mevcuttur: Kenarsal taban ve Köşegensel taban (bkz. Şekil 3.2). İki tane taban seçilmesinin kuantum kriptografi açısından önemi büyüktür.

Anlaşılacağı üzere bu taban çifti, Heisenberg belirsizlik ilkesinde belirtilen fiziksel büyüklük çiftine karşı düşmektedir.

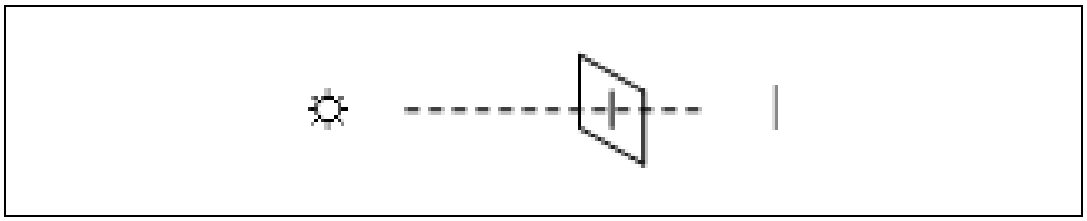
Böylesi ikili durumların, pratikte, ikili 0 ve 1 bitleri ile temsil edilebileceğine de dikkat ediniz. Aşağıda, bu ikili durumların başka örnekleri verilirken bu noktanın da akılda bulundurulması faydalı olabilir.

Şekil 3.3: Doğrusal polarizasyonda mevcut dört polarizasyon durumu. a) 0° (yatay) ve 90°'lik (dikey) kenarsal polarizasyonlar, b) 45° ve 135°'lik köşegensel polarizasyonlar.

Şekil 3.2'den de görüldüğü gibi her bir taban birbirine dik iki yönden oluşmaktadır (böylece, yanlış tabanla ölçüm durumlarında her bir yönün oluşması olasılığı eşit olur). Buna göre, her bir taban için bir fotonun sokulabileceği iki farklı polarizasyon durumu mevcuttur. Bu tabanlar için fotonun polarize edilebileceği mevcut dört polarizasyon durumu Şekil 3.3'te gösterilmiştir.

3.1.2. Fotonun Polarize Edilmesi

Bir biti, bir fotonun polarizasyon durumu ile kodlamak için, fotonu belirli bir polarizasyon durumuna sokmak gerekir. Bu ise elektrik alanı istenilen düzlemde dalgalanan bir foton yaratmakla olur. Bunu yapmanın yolu fotonu, polarizasyon ekseni istenilen açıya ayarlanmış bir polarize edici (örneğin, polarizasyon filtresi) içinden geçirmektir.



Şekil 3.4: Fotonun dikey polarizasyon durumuna sokulması (foton dikey polarize ediciden geçirilir).

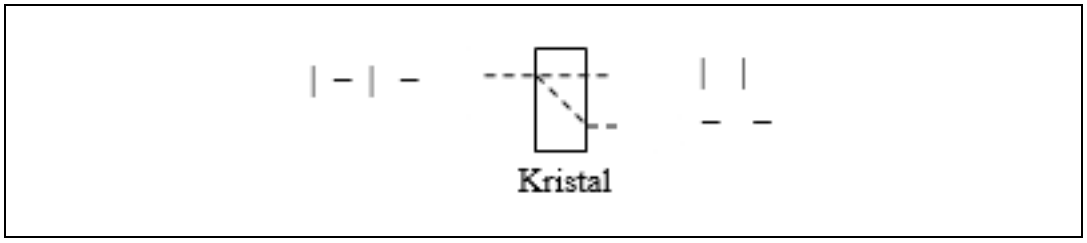
Şekil 3.4'te temsili olarak bir fotonun dikey bir polarize ediciden geçirilerek kenarsal tabandaki polarizasyon türlerinden dikey polarizasyon durumuna sokulması gösterilmektedir. Uygun polarize ediciler kullanılmak suretiyle, 0°, 45° ve 135°'lik polarizasyonlar da elde edilebilir.

Buna göre, Şekil 3.2 ve Şekil 3.3'te tanımlanan dört durumlu sistem için dört tane polarize edici gerekeceği açıktır.

3.1.3. Fotonun Polarizasyonunun Ölçülmesi

Her bir fotonu belirli bir polarizasyon yönünde polarize edilmiş olan bir foton dizisinden mesaja ait bitleri geri elde etmek için, alıcı gelen fotonların her birinin polarizasyonunu ölçebilmelidir. Bunu yapmanın yolu fotonu, uygun polarizasyon ölçücü (örneğin, Kalsit kristali) içinden geçirmektir.

Kalsit kristali içinden geçen bir foton, geçişi sırasında, kristaldeki polarizasyon ekseni ve kendi polarizasyon eksenine bağlı olarak bir elektromanyetik kuvvet hisseder. Hissettiği bu etkiyle orantılı olarak foton kristal içinden ya dosdoğru geçer veya geliş ekseninden bir miktar kaymış olarak kristalden dışarı çıkar (ya da yutularak yok olur). Örneğin, Kalsit'in polarizasyon ekseni dikey/düşey ayarlanmış ise düşey olarak polarize edilmiş fotonlar kristal içinden dosdoğru geçerler. Yatay polarizasyona sahip foton da kristal içinden geçecektir; ancak, polarizasyon düzlemi kristalinki ile paralel olmadığından kristalden dışarı, Şekil 3.5'te de görüldüğü gibi, orijinal yörüngesinden bir miktar kaymış olarak çıkacaktır.



Şekil 3.5: Fotonların kristalden geçişi.

Kalsit kristalinin bu özelliği sayesinde, verilen bir fotonun yatay mı yoksa düşey mi polarizasyona sahip olduğu anlaşılabilir.

Köşegensel olarak polarize edilmiş fotonlar için de uygun Kalsit kristali kullanılarak polarizasyonları ölçülebilir.

Bir taban için polarize edilmiş fotonları yukarıdaki gibi ayırt edebilme imkânına kavuştuktan sonra, kristal çıkışlarına foton dedektörleri yerleştirilerek bu çıkışlardan foton gelip gelmediği kolayca anlaşılabilir. Hangi detektörden çıkış alındığına göre de gelen fotonun polarizasyonu hakkında fikir sahibi olunur ve polarizasyondan da ilgili bit değerine ulaşılır.

Buna göre, her bir taban için bir kristale ve her bir kristal için de iki tane foton dedektörüne ihtiyaç olmaktadır.

3.1.4. Polarize Edilmiş Foton için Kopyalanamazlık Teoremi

Polarizasyonu bilinmeyen bir fotonun polarizasyonunu bilmemiz mümkün değildir. Örneğin, doğadaki, bizim polarize etmediğimiz herhangi bir fotonun polarizasyonunu bilebilmemizin imkânı yoktur.

Polarizasyonu bilinmeyen bir fotonun polarizasyonunu öğrenebilmemiz için onu doğru polarizasyon ölçücü ile ölçebilmemiz gerekir. Ancak, zaten polarizasyonu bilmediğimiz için, onu ölçmek için gerekli doğru polarizasyon ölçücüyü belirlemek de asla olası değildir. Yanlış polarizasyon ölçücü kullanıldığında ise bu etkileşimden dolayı fotonun gerçek polarizasyonu polarizasyon ölçücüye göre değişecek ve gerçek polarizasyonu bir daha asla yeni bir öğrenme şansı olmayacaktır.

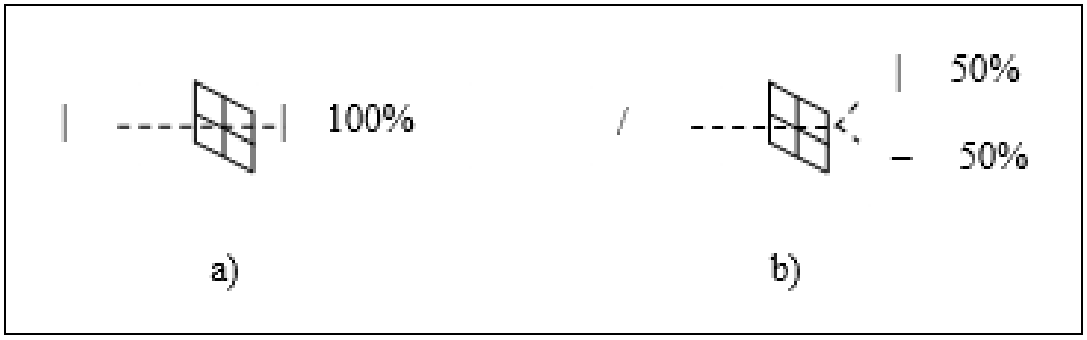
Pratikte, bir foton ölçüldüğünde o foton detektörde yok olmaktadır (ısıya vs. dönüşür). Bu nedenle, yapılan bir ölçüm sonrasında aynı fotonu ikinci bir ölçüm şansımız kalmamaktadır.

3.1.5. Polarize Edilmiş Foton için Heisenberg Belirsizlik İlkesi

Heisenberg'in belirsizlik ilkesi foton polarizasyonu için de geçerlidir. Yani, kenarsal ve köşegensel tabandaki polarizasyonlar o taban için sadece uygun Kalsit kristali kullanmak suretiyle ayrı ayrı ölçülebilirler, aynı anda tek bir ölçümle belirlenemezler. Buradaki fiziksel büyüklük çifti bir kenarsal polarizasyon ve bir köşegensel polarizasyon olmaktadır.

Polarizasyonu bilinmeyen bir foton için bir tabanda yapılan polarizasyon ölçümü bize o taban için bir bilgi verir ve yapılan ölçümle fotonun gerçek polarizasyonu artık tamamen belirsiz olur.

Gelen bir foton için uygun polarizasyon ölçücü kullanılırsa ölçüm sonucu da doğru olacaktır. Şekil 3.6.a'da kenarsal tabanda polarizasyona sahip olan bir foton kenarsal bir polarizasyon ölçücü ile ölçüldüğünden ölçüm sonucu doğru olur.



Şekil 3.6: Polarizasyon ölçümleri. a) Doğru ölçücü kullanımı, b) Yanlış ölçücü kullanımı.

Gelen bir foton için yanlış polarizasyon ölçücü kullanılırsa ölçüm sonucu da yanlış olacak ve yapılan ölçüm fotonu rahatsız edeceğinden gerçek polarizasyonu belirsiz kılacaktır (Aslında, yapılan ölçüm fotonu yok edecektir ve elimizde fotona ilişkin yanlış bir bilgi kalır). Şekil 3.6.b'de köşegensel tabanda polarizasyona sahip bir foton kenarsal bir polarizasyon ölçücü ile ölçüldüğünden ölçüm sonucu yanlış olur. Yapılan ölçüm sonucu fotonun gerçek polarizasyonu belirsiz olacağından, gerçek polarizasyonu yeni bir öğrenme imkânı olmayacaktır.

Foton polarizasyonunun ölçülmesinde açıklanan bu belirsizlik örneğin kriptografik bir anahtar değiş-tokuş işlemi sırasında, kullanılan haberleşme hattının dinlenip dinlenmediğini anlamak için kullanılabilir.

Şekil 3.6.b'de çıkışların eşit olasılıklı olmasından hareketle, böylesi bir (optik) düzeneğin/kurulumun bir (kuantum) rasgele sayı üreteci olarak kullanılabileceğine de dikkat edin (ki kullanılmaktadır da).

3.1.6. Mesajların Polarize Edilmiş Fotonlarla Kodlanması

İstenen polarizasyonda fotonları kolayca elde edebilme olanağına sahip olduktan sonra, artık 0° (kenarsal taban) ve 135° (köşegensel taban) açılarla polarize edilmiş fotonları ikili 0 değerini temsil etmek için; 90° (kenarsal taban) ve 45° (köşegensel taban) açılarla polarize edilmiş fotonları ise ikili 1 değerini temsil etmek için kullanabiliriz. Söz edilen karşı düşürme kuralı Şekil 3.7'de gösterildiği gibidir.

–	|	/	\
0	1	1	0
a)		b)	

Şekil 3.7: İkili bit değerlerinin polarize edilmiş fotonlarla temsili. a) Kenarsal polarizasyonlar, b) Köşegensel polarizasyonlar.

Hangi mesaj bitinin hangi tabandaki bir fotonla kodlanacağını belirlemek için bir kodlama kuralı da polarizasyon tabanları için Şekil 3.8'deki gibi tanımlanabilir.

+	X
1	0
a)	b)

Şekil 3.8: Polarizasyon tabanlarının ikili bitlerle temsili. a) Kenarsal taban, b) Köşegensel taban.

Buna göre mesaj bir bit dizisine dönüştürüldükten sonra mesajla aynı uzunlukta rasgele bir bit dizisi de polarize edicide, fotonları polarize etmekte kullanılacak polarizasyon tabanlarını kontrol etmek için kullanılabilir. Yani sırasıyla hangi fotonun hangi polarizasyon tabanına göre polarize edileceği rasgele olarak belirlenir (rasgelelik kaynağı olarak bir KRSÜ kullanılmalıdır).

3.1.7. Polarize Edilmiş Fotonlarla Anahtar Dağıtımı

Yukarıda verilen bilgiler doğrultusunda kriptografik bir gizli anahtarın birbirinden uzaktaki iki taraf arasında kuantum fiziği yasaları kullanılarak güvenli bir şekilde dağıtımı gerçekleştirilebilir. Kuantum kriptografide anahtar dağıtımı fotonları iletme özelliğine sahip herkese açık bir kanal üzerinde polarize edilmiş fotonlarla gerçekleştirilir.

Kuantum kriptografik anahtar dağıtım protokolünde 1 ve 0 bit dizilerinden oluşan rasgele bir mesaj her biri belirli bir polarizasyon durumuna sokulmuş

fotonlarla kodlanır ve alıcıya bu fotonlar gönderilir. Anahtar protokol sonunda bu rasgele mesajdan elde edilir. Güvenlik açısından gerekli rasgeleliklerin bir KRSÜ ile sağlanması gerekir.

+ -------->	|	–	Foton
X -------->	/	\	polarizasyonları
	1	0	Bit eşdeğeri

Şekil 3.9: Bitleri foton polarizasyon durumları ile kodlama kuralları.

Protokolü koşturmadan önce göndericinin ve alıcının Şekil 3.9'daki gibi bir kodlama kuralı üzerinde anlaşmış olması gerekir. Bu kodlama kuralının herkese açıklanmasında bir mahsur yoktur.

Göndericinin bitleri	**0**	**1**	0	0	1	**0**	**1**	1	**0**	**1**
Göndericinin polarizasyon tabanları	+	**X**	X	X	X	+	+	X	+	**X**
Gönderilen fotonlar	–	/	\	\	/	–	|	/	–	/

Alınan fotonlar	–	/	\	\	/	–	|	/	–	/
Alıcının polarizasyon tabanları	+	**X**	+	+	+	+	+	+	+	**X**
Alıcının bitleri	**0**	**1**	0	0	0	**0**	**1**	1	**0**	**1**

Gizli Anahtar: 0 1 0 1 0 1

Şekil 3.10: Tarafların ortak bir gizli anahtar üzerinde anlaşması.

Kodlama kurallarının belirlenmesinden sonra KAD yöntemi, özetle, şu şekilde çalışır (bkz. Şekil 3.10).

- Gönderici öncelikle rasgele bir bit dizisi seçer. Alıcının, göndericinin seçtiği bu bit dizisinden haberi yoktur (Güvenlik açısından gerekli rasgeleliklerin bir KRSÜ ile sağlanması gerekir).
- Gönderici her bir biti için rasgele bir polarizasyon tabanı (+ veya X) belirler, bitini bu tabanda Şekil 3.9 uyarınca uygun (|, –, / veya \ ) polarize edilmiş fotonla kodlar ve fotonu alıcıya gönderir.
- Alıcı, gelen her bir fotonun polarizasyonunu ölçer. Ancak göndericinin fotonu polarize ederken hangi polarizasyon tabanını kullandığını bilmemektedir. Bu nedenle ölçümü sırasında foton için kullanacağı polarizasyon tabanını rasgele seçer (Güvenlik açısından gerekli rasgeleliklerin alıcıda da bir KRSÜ ile sağlanması gerekir). Sonuçta, alıcı gelen fotonları rasgele ölçer ve o da Şekil 3.9 uyarınca bir bit dizisi elde eder.
- Gönderici ve alıcı sadece kullandıkları polarizasyon tabanlarını kimlik kanıtlamalı ancak gizli olması gerekmeyen bir kanal üzerinden, örneğin telefonla, birbirlerine açıklar. Aynı polarizasyon tabanlarını kullandıkları durumlar için gönderilen ve alınan bitler kesinlikle aynı olacaktır. Bu ortak, ama gizli, bitler anahtar olarak kullanılabilirler.

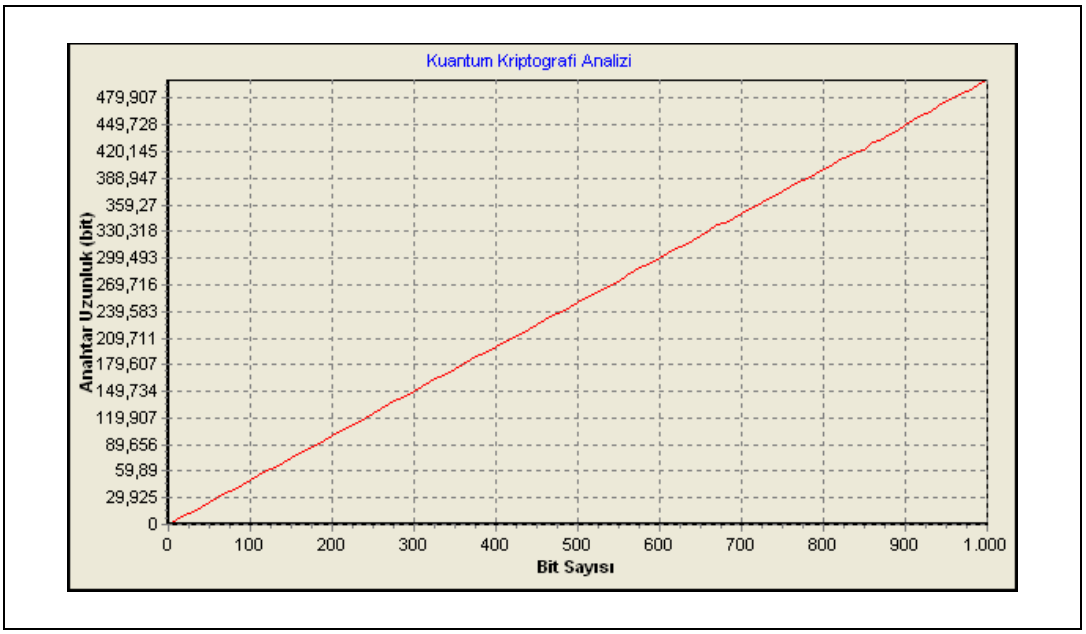
Yukarıdaki örnekte taraflar protokol sonunda 010101 gizli anahtarı üzerinde anlaşmış bulunmaktadır. Protokol sonunda elde edilen anahtar uzunluğunun protokol başındaki bit dizisinin hemen hemen yarısı olduğuna dikkat edin.

Olasılıksal olarak, gönderilen herhangi bir foton için alıcının doğru tabanı (+ veya X tabanından birini) seçmesi olasılığı,

$$p_{\substack{alıcı\ doğru \\ taban}} = \frac{1}{2} \tag{3.1}$$

dir (bu ifadenin doğru olması için bütün çıktıları eşit olasılıklı olan bir GRSÜ kullanılması gerekir ki günümüzde en güvenilir aday KRSÜ'dür). Böylece, alıcı 50% olasılıkla göndericiyle aynı tabanı kullanmayı tahmin edecektir ve bunlar gizli anahtar oluşturmak için kullanılabilecektir.

Olasılıksal olarak, gönderilen fotonların yarısı yanlış taban seçimi nedeniyle gizli anahtar oluşturmak için kullanılamayacaktır. Gerçekten, benzetim sonuçları da bunu doğrulamaktadır.



Şekil 3.11: Elde edilen ortalama anahtar uzunlukları.

Şekil 3.11'den [23] görüldüğü gibi protokol sonunda elde edilen anahtarların uzunlukları yaklaşık (istatistiki) olarak protokol başında seçilen rasgele bit dizisinin uzunluğunun yarısı kadar olmaktadır.

3.1.8. İletişimin Dinlendiğinin Anlaşılması

Yukarıdaki yöntem, anahtar dağıtımı esnasında, hatta bir müdahale olup olmadığı hakkında bilgi vermez. Şu özellik kullanılarak hatta müdahalenin varlığı da anlaşılabilir: Aynı polarizasyon tabanının kullanıldığı bir durum için gönderilen ve alınan bit aynı çıkmalıdır.

Arada birinin olması durumunda göndericinin gönderdiği fotonları önce saldırgan alır ve yaptığı ölçümler neticesinde o da bir bit dizisi belirler. Kuantum kopyalanamazlık teoremine ve Heisenberg belirsizlik ilkesine göre saldırgan fotonları kopyalayamayacağından onları yeniden oluşturur ve alıcıya bu yeni fotonları gönderir (bkz. Şekil 3.12).

Arada birinin olup olmadığını belirlemek için gönderici ile alıcı bitlerden bir alt küme için hem polarizasyon tabanlarını hem de bitlerin değerlerini karşılaştırırlar. Bu karşılaştırma, sadece polarizasyon tabanında uyuştukları bitlerin bir alt kümesi ile de yapılabilir. Polarizasyon tabanında anlaştıkları bir durum için gönderilen ve alınan bitte de kesinlikle uyuşma olmalıdır.

Göndericinin bitleri	1	1	0	0	0	0	1	1	0	1
Göndericinin polarizasyon tabanları	X	X	+	+	X	+	+	+	+	X
Gönderilen fotonlar	/	/	–	–	\	–	|	|	–	/

Alınan fotonlar	/	/	–	–	\	–	|	|	–	/
Saldırganın polarizasyon tabanları	X	X	+	+	X	+	X	+	+	+
Saldırganın bitleri	1	1	0	0	0	0	0	1	0	1

Saldırganın bitleri	1	1	0	0	0	0	0	1	0	1
Saldırganın polarizasyon tabanları	X	X	+	+	X	+	X	+	+	+
Gönderilen fotonlar	/	/	–	–	\	–	\	|	–	|

Alınan fotonlar	/	/	–	–	\	–	\	|	–	|
Alıcının polarizasyon tabanları	+	+	X	X	+	X	+	X	X	+
Alıcının bitleri	1	0	0	0	1	1	0	1	0	1

Şekil 3.12: Hattı dinleyen birinin varlığı.

Şekil 3.13'te görüldüğü gibi, test edilen koyu renkli bitlerden dördüncüsünde (soldan sağa doğru 7. bit olmakta), aynı taban kullanılmasına rağmen, saldırgandan kaynaklanan bir hata görülmektedir.

Bu (bilinçli/bilinçsizce yapılan) hata (durumu) arada hattı dinleyen biri(leri)nin bulunduğunu açığa vurmaktadır.

Göndericinin bitleri	1	**1**	**0**	0	0	**0**	**1**	1	0	1
Göndericinin polarizasyon tabanları	X	**X**	+	+	X	+	+	+	+	X
Gönderilen fotonlar	/	/	–	–	\	–	|	|	–	/

Alınan fotonlar	/	/	–	–	\	–	\	|	–	|
Alıcının polarizasyon tabanları	+	+	**X**	X	+	**X**	+	X	X	+
Alıcının bitleri	1	**0**	**0**	0	1	**1**	**0**	1	0	1

Şekil 3.13: Hattı dinleyen birinin varlığının anlaşılması.

Yapılan test sonucu hattın dinlendiği saptanamamışsa anahtar test bitleri dışındaki bitler arasından ilk yöntemde anlatıldığı gibi belirlenir.

Sonuçta, hattı dinleyen birileri varsa gönderici ile alıcı daha sonra tekrar denemek üzere görüşmelerini sonlandırmaya karar verirler.

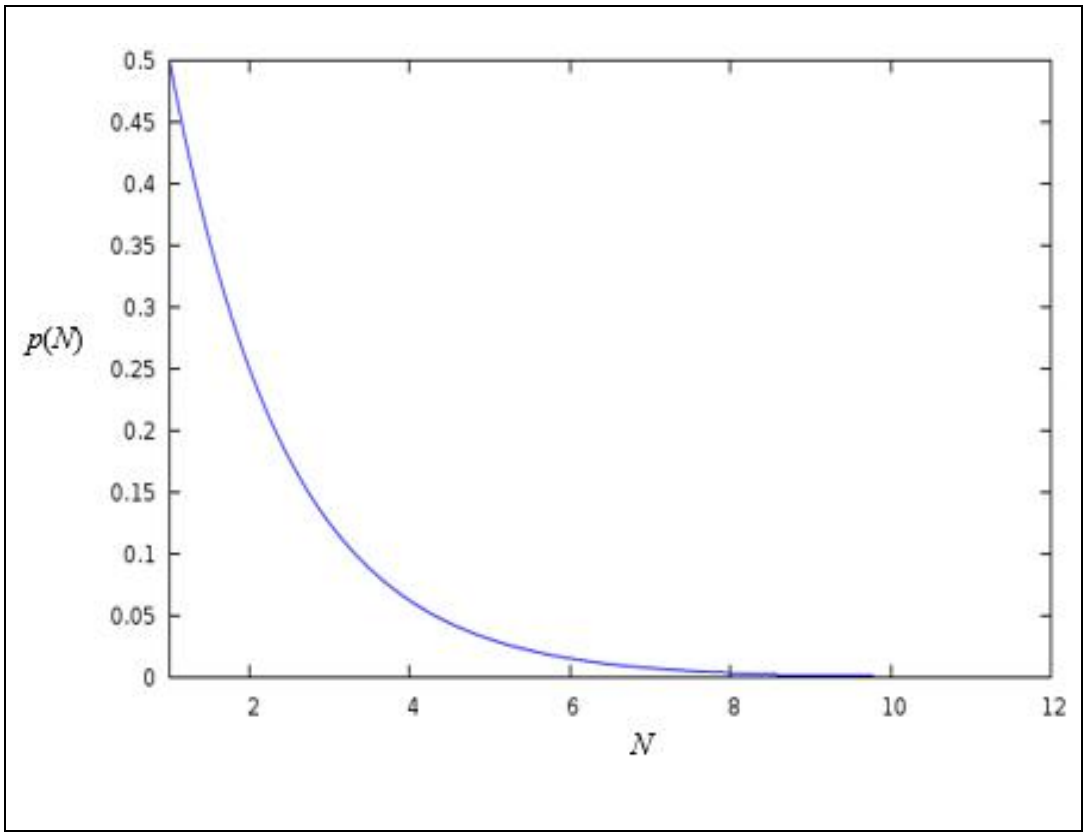
Saldırganın (varlığının) yakalanamaması için kendisine gelen fotonları aynen gönderebilmesi gerekir. Bunu yapabilmesinin tek yolu ise gönderici ile tamamen aynı tabanları kullanıyor olmasıdır. Tek bir taban için bunu yapmasının, yani, aynı tabanı tahmin etmesinin, olasılığı,

$$p_{\substack{saldırgan\ doğru \\ taban}} = \frac{1}{2} \tag{3.2}$$

dir. Toplam N adet taban için ise bu olasılık,

$$p_{\substack{doğru \\ tabanlar}} = (\frac{1}{2})^N \tag{3.3}$$

olacaktır. Bu fonksiyonun grafiği Şekil 3.14'teki gibidir.



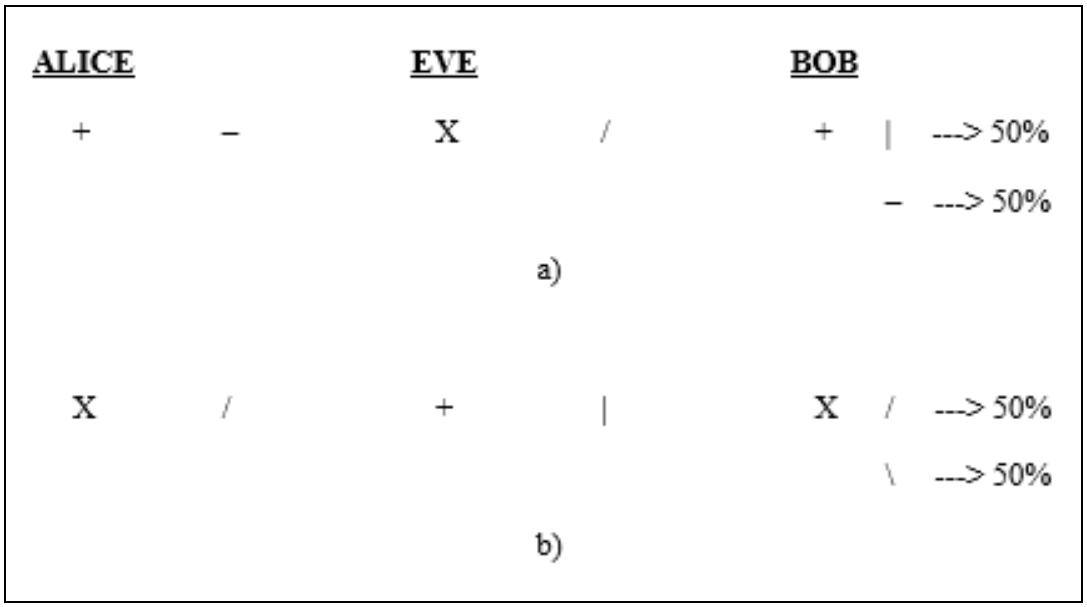
Şekil 3.14: Saldırganın gönderici ile aynı tabanları seçme olasılığı.

Gönderici ve saldırganın aynı tabanları kullanması durumunda göndericiden gelen fotonlar alıcıya aynen gideceklerinden, sanki arada bir saldırgan yokmuş gibi bir durum söz konusu olur ve saldırgan açısından herhangi bir tespit edilme vb. tehlikesi asla oluşmaz. Ancak, Şekil 3.14'ten de görüldüğü gibi $N \geq 10$ için bu durumun gerçekleşebilmesi olasılığı hayli düşüktür.

Daha önce de dikkat çekildiği gibi, bu ifadelerin doğru olması için bütün çıktıları eşit olasılıklı olan bir GRSÜ'ye (KRSÜ) ihtiyaç vardır.

Saldırganın göndericiden farklı polarizasyon tabanları seçtiği zamanlarda ise Şekil 3.15'teki gibi olasılıklar söz konusudur.

- 50% olasılıkla göndericinin gönderdiği foton alıcıya aynen ulaşır. Böylece, göndericideki ve alıcıdaki bit de kesinlikle aynı olur ve böylesi şanslı durumlarda saldırgan asla tespit edilemez.
- 50% olasılıkla göndericinin gönderdiği foton alıcıya aynen ulaşamaz. Bu nedenle, göndericideki ve alıcıdaki bit farklı olur ve ilgili bitin test biti olarak da seçilmesi halinde saldırgan tespit edilir.



Şekil 3.15: Saldırganın tespit edilememe durumu.

Görüldüğü gibi, saldırganın tespit edilebilmesi için farklı bir taban seçmesi ve buna bağlı olarak alıcının da yanlış bir bit elde etmesi gerekir.

Tek bir foton için saldırganın farklı bir taban seçmesi olasılığı,

$$p_1 = \frac{1}{2} \tag{3.4}$$

dir. Benzer şekilde, buna bağlı olarak alıcının yanlış biti elde etmesi olasılığı,

$$p_2 = \frac{1}{2} \tag{3.5}$$

dir. Böylece, tek bir bit için saldırganın tespit edilebilmesi olasılığı,

$$p_{tespit} = p_1 \cdot p_2 = \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \tag{3.6}$$

olarak belirlenir.

Göndericinin ve alıcının aynı tabanları seçmelerine rağmen, saldırgan nedeniyle değerleri farklı olan böylesi bitlerin test bitleri olarak seçilmeleri saldırganın varlığını açığa çıkaracaktır.

3.1.9. Güvenlik Seviyesi

Gizli anahtarın belirlenmesine geçmeden hemen önce, ne kadar fazla sayıda bit test edilirse bu testin araya girerek hattı dinleyen biri(leri)nin varlığını saptama olasılığı da o kadar fazla olacaktır.

Gönderici ve alıcı tarafından test edilen her bit için bu testin araya giren birilerinin varlığını açığa çıkarma olasılığı,

$$p_{tespit} = \frac{1}{4} \tag{3.7}$$

olarak belirlenmişti. Böylece, test edilen bir bit için bu testin araya giren birilerinin varlığını açığa çıkaramama olasılığı ise,

$$p'_{tespit} = 1 - p_{tespit} = \frac{3}{4} \tag{3.8}$$

olarak belirlenir. Toplam *t* test biti için ise,

$$p'_{\substack{tespit \\ edilememe}} = (\frac{3}{4})^t \tag{3.9}$$

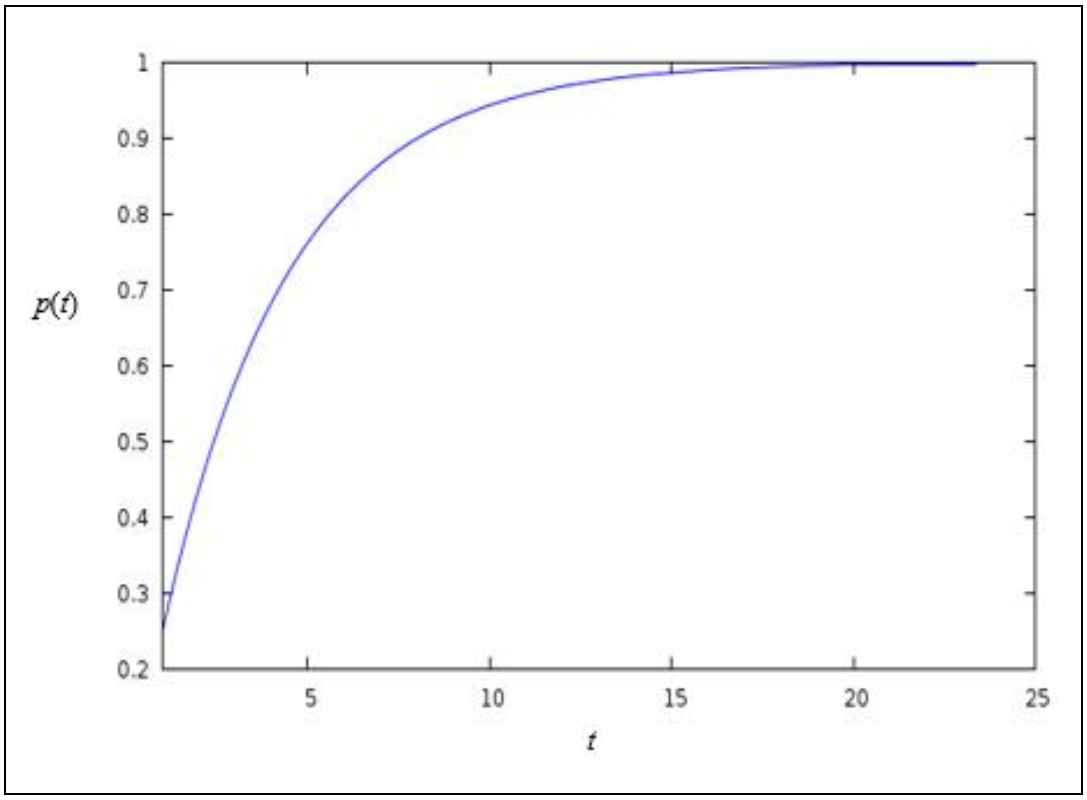
olarak elde edilir. Bu değer, eğer *t* bit test edilirse hattın dinlendiğinin açığa çıkarılamama olasılığıdır.

Sonuç olarak, eğer *t* bit test edilirse hattın dinlendiğinin açığa çıkarılma olasılığı,

$$p_{\substack{saldırgan \\ tespit}} = 1 - (\frac{3}{4})^t \tag{3.10}$$

olarak belirlenir. Bu fonksiyonun grafiği Şekil 3.16'da çizdirilmiştir.

Daha önce de dikkat çekildiği gibi, bu ifadelerin doğru olması için polarizasyon tabanları birbirine dik iki yönden oluşmalı ve rasgelelik kaynağı olarak da bir KRSÜ kullanılmalıdır.



Şekil 3.16: Hattın dinlenip dinlenmediğinin belirlenme olasılığı. Test edilen bitlerin sayısının bir fonksiyonu olarak dinlemenin anlaşılması olasılığı.

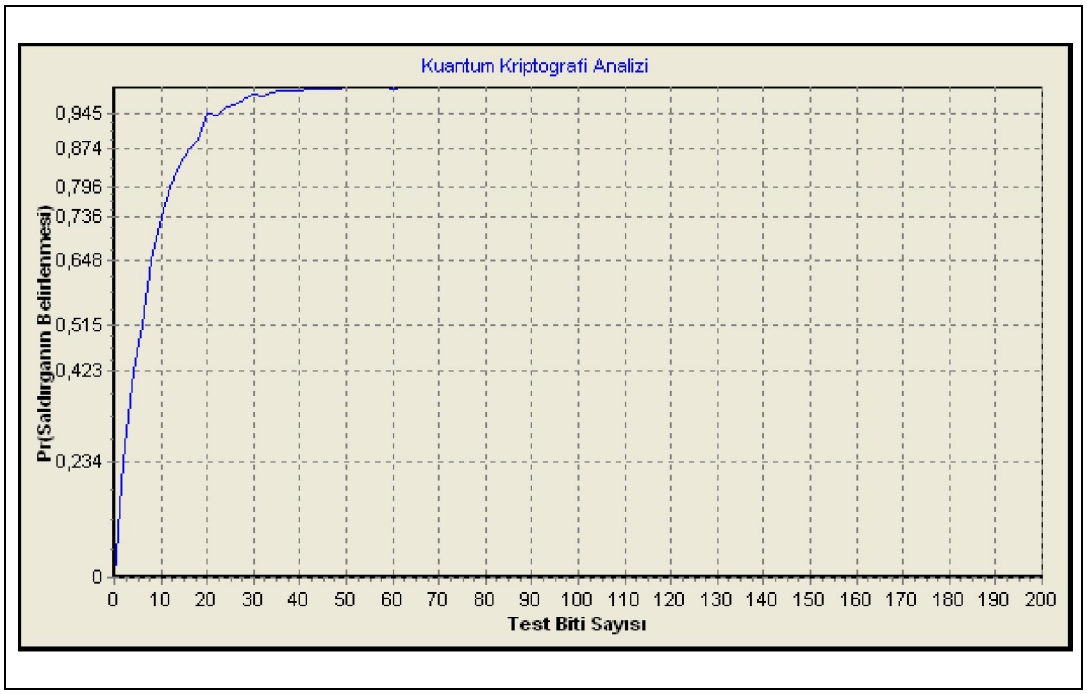
Görüldüğü gibi test edilen bitlerin sayısı sonsuza gittiği zaman hattı dinleyen birinin varlığını belirleme olasılığı da asimptotik olarak 1'e yaklaşmaktadır. Böylece, sadece test edilen bitlerin sayısı arttırılarak hattı dinleyen birilerinin varlığının belirlenme olasılığı istenilen kesinliğe kavuşturulabilir. Şekilden de görüldüğü gibi, yaklaşık 20 test bitinden sonra saldırganın tespit edilebilme olasılığı hemen hemen 1 olmaktadır (p(20) = 0.9968, p(30) = 0.9998, p(40) = 0.99998, …).

Gösterilmeye çalışıldığı gibi, KAD'ın, daha genel olarak da kuantum kriptografinin, her bir adımı olasılık, istatistik, matematik ve fizik kullanılarak açıklanabilmekte, analiz ve ispatları yapılabilmektedir.

KAD'ın kuantum aşamalarında güvenlik (burada doğa böyle çalışmaktadır denilerek) evrensel ve değişmez (kuantum) fizik kanunlarına dayandırılmaktadır. Klasik aşamalarda ise bilgi teorisi kullanılmaktadır.

Her bir adım için güvenlik seviyesinin bu şekilde kontrol altına alınabilmesi sadece kuantum kriptografiye has bir özelliktir.

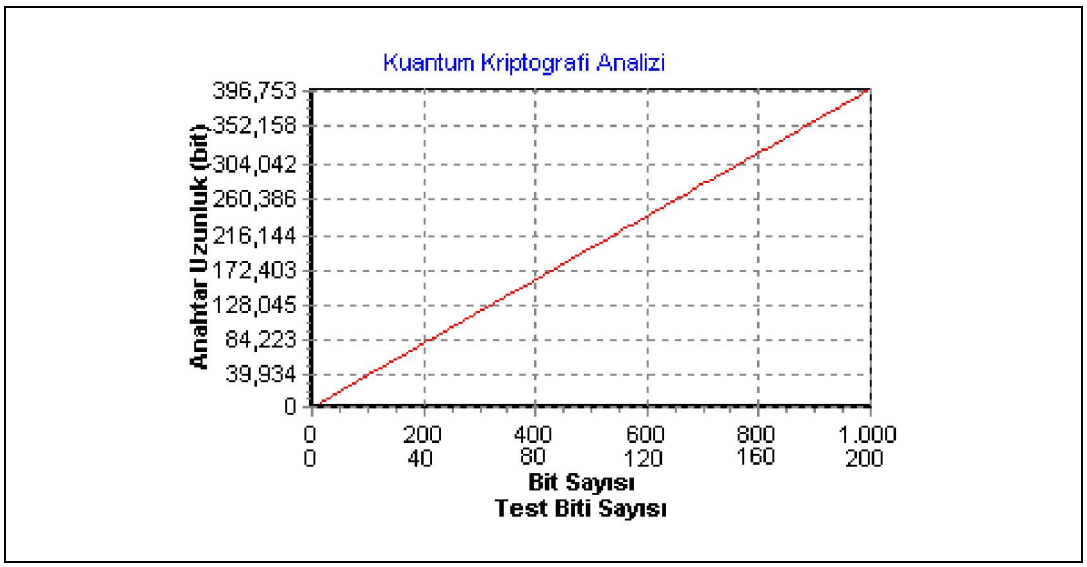
Bilgisayar benzetim sonuçları da yukarıda yapılan teorik analiz sonuçlarını doğrulamaktadır.



Şekil 3.17: Saldırganın belirlenme olasılığı.

Şekil 3.17'den [23] görüldüğü gibi yaklaşık olarak 20 tane test bitinden sonra saldırganın yakalanma olasılığı neredeyse 1 olmaktadır.

Eğer saldırgan tespit edilemezse gönderici ve alıcı normalde olduğu gibi protokol sonunda bir gizli anahtar üzerinde anlaşacaklardır. Bazı bitlerin test biti olarak kullanılmasından dolayı elde edilen anahtarlar Şekil 3.11'dekilere göre daha kısa uzunlukta olmaktadır (bkz. Şekil 3.18 [42]).



Şekil 3.18: Test bitleri kullanılması durumunda anahtar uzunlukları. Elde edilen anahtarlar daha kısa olur.

Saldırganın tespit edilemediği bir durum için gönderici, alıcı ve saldırganın anlaştığı gizli anahtarlara bir örnek Tablo 3.1'de [23] verilmektedir.

Tablo 3.1: Saldırganın belirlenemediği durumda anlaşılan anahtarlar.

Gönderici	0 0 0 1 1 0
Alıcı	0 0 0 1 1 1
Saldırgan	X 0 0 X 1 X (X: belirsiz, 0 veya 1 olabilir)

Görüldüğü gibi saldırganın belirlenemediği bu durumda gönderici ve alıcı aynı anahtar üzerinde anlaşamamıştır. Saldırgandan kaynaklanan gürültü nedeniyle iki tarafta farklı anahtarlar oluşmuştur. Ayrıca saldırgan da göndericideki ve alıcıdaki bu anahtarlara ilişkin bazı bitlere sahiptir.

Test bitlerinin sayısı yeterince uzun seçilerek bu güvensiz durumun oluşması engellenebilir.

3.2. Diğer Yaklaşımlar

Yukarıda verilen ilk KAD protokolünün keşfinden sonra güvenli anahtar dağıtımının gerçekleştirilebilmesine yönelik olarak birçok yeni öneriler de olmuştur. Bunlarda da polarizasyon gibi az bulunan başka kuantum etkilerden yararlanılır.

- Faz Kodlama

Polarizasyonu değil de fotonun faz özelliğini kullanır. Faz da polarizasyon gibi fotonun dalga oluşundan kaynaklanan bir özelliğidir [62].

- Dolaşıklık

Kuantum parçacık çiftlerinin dolaşıklık diye bilinen özelliğine dayanır. Özellik, çiftin tek bir kuantum sistemmiş gibi davranması anlamına gelmektedir. Buna göre çiftin parçacıkları eğer birbirinden ayrılırsa biri üzerinde yapılan bir etki aralarındaki mesafeden bağımsız olarak zıt yönde olmak üzere diğeri üzerinde de meydana gelmektedir. Bu özellik kuantum bilgi ışınlamasına temel teşkil etmektedir. Doğrudan bilgi ışınlamasına olanak tanıyan bu özelliğin de yine klasik sistemlerde bir eşdeğerini bulmak mümkün değildir [63].

• Diğerleri

BB84'teki dört polarizasyon durumunun aksine en az iki ve en çok altı kuantum durumunun (polarizasyon vd.) kullanıldığı başka yaklaşımlar da vardır [64]-[75].

3.3. Dezavantajlar

KAD'ın, daha genel olarak kuantum kriptografinin, başlıca dezavantajları şu şekilde özetlenebilir.

• Sınırlı Mesafe

Ulaşılabilecek maksimum doğrudan fiber optik kablo bağlantı mesafesi limiti (şimdilik) yaklaşık olarak 400 km'dir.

• Atanmış Bağlantı

Yukarıdaki mesafe limiti içinde atanmış bir fiber bağlantı hattı ya da görüş mesafesi gereklidir.

• Olgunluk

Kuantum kriptografi çok geniş ve gelişmekte olan yeni bir alandır. Optik bileşenlerin güvenli kullanımına veya güvenliğinin test edilmesine yönelik kabul edilmiş uluslararası standartlar henüz mevcut değildir, üzerinde çalışılmaya devam edilmektedir.

3.4. Avantajları

KAD'ın, ya da daha genel olarak kuantum kriptografinin, başlıca faydaları ise şu şekildedir.

• Bilgi Teoriksel Güvenlik

Her bir aşaması fizik yasaları ve/veya bilgi teorisinin sunduğu araçlar kullanılarak tam olarak analiz edilebilir ve ispatlanabilir. Böylece, güvenlik ispatlanabilirdir.

• Geleceğe Yönelik Güvenlik

Başarılı bir saldırı anahtar dağıtımı esnasında olmalıdır. O anki anahtar değiş-tokuşunun kopyalanmasının geçmişteki ya da gelecekteki herhangi bir anahtarın elde edilmesine yardımı olmaz.

3.5. Şifreleme

Anahtar dağıtımı istenilen güvenlikte gerçekleştirildikten sonra güvenle taşınan bu anahtar güçlü bir simetrik şifreleme sistemi ile birlikte kullanılarak gizli bilgilerin güvenliği de garanti altına alınabilir.

Günümüzde pek çok simetrik şifreleme yöntemi bilinmektedir. Bunlardan Vernam şifresi kırılamazlığı (Shannon tarafından) kanıtlanmış tek (simetrik) şifreleme sistemi olup yollanacak mesajlar ile aynı uzunlukta anahtarlar kullanmaktadır. Kırılamazlığına rağmen anahtar yönetiminin zorluğundan dolayı günümüze kadar pek kullanım alanı bulamamıştır.

AES [76], [77] ise günümüzün en popüler ve güvenli kabul edilen simetrik şifreleme sistemlerindendir. 128, 192 ve 256 bitlik anahtar (ve blok) uzunluklarıyla çalışabilmektedir. Bu anahtar uzunlukları dolayısı ile AES günümüz kriptanaliz ve saldırı tekniklerine karşı güvenli kabul edilmektedir. Ancak, güvenliğinin ya da kırılamazlığının bir ispatı olmadığına da dikkat edin.

Kuantum kriptografi hem kırılamaz Vernam şifresinin hem de güvenilir varsayılan AES'in ihtiyaç duyacağı güvenli anahtarları sağlayabilmektedir. Ancak Vernam şifresinin kullanımını olanaklı kılması, asla kırılamaz bilgi güvenliğini de bir bakıma olanaklı hale getirmiş olmaktadır.

Kuantum kriptografi, konuları ve uygulamaları hakkında daha detaylı bilgi için referans [78]-[82]'ye bakılabilir.

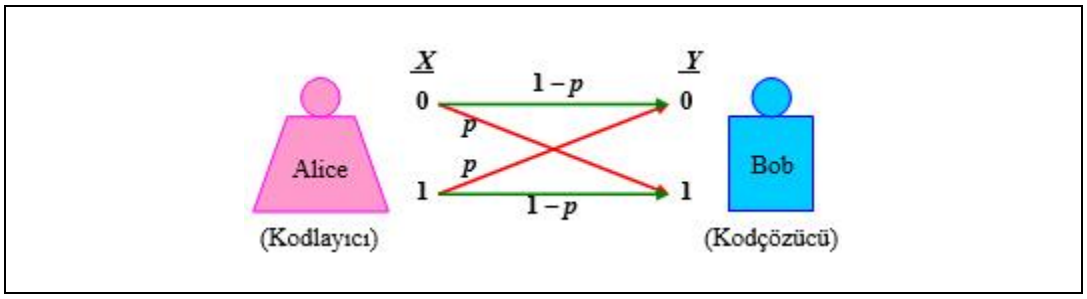
4. BİLGİ UZLAŞTIRMA

Bu bölümde, BB84 ve CASCADE protokolleri örneğinde, KAD'da bilgi uzlaştırma (BU) konusuna değinilmektedir.

4.1. Kuantum Kanal

Haberleşmede kanal kavramı ile uydu, koaksiyel kablo, fiber optik, açık hava, uzay, kurye vb. kablolu ve kablosuz iletişim ortamları ifade edilmektedir. Kanal seçiminde en önemli etmen iletilecek verinin tipidir. Ayrıca, gürültü ve tasarlanacak sistemin maliyeti de kanal seçiminde önemlidir [83].

İletişim kanalı, simgelerin (0, 1'lerin) alıcıya aktığı bir yol veya ortamdır (analog veriler Nyquist kriterine uygun örneklenerek 0, 1'lere dönüştürülebilir). En geniş anlamda modern iletişim sistemleri, elektrik ya da optik sinyaller kullanarak mesajı (içerdiği enformasyonu) bir noktadan (kaynak) diğer bir noktaya (hedef) yüksek bir verimle ve güvenilirlikle iletmeye çalışır. Bilgi (enformasyon) teorisi sayısal haberleşme sistemlerinin temelini oluşturmaktadır.



Şekil 4.1: İSK blok diyagramı.

Enformasyon kuramı, mesaj sinyallerinin içerdiği enformasyon miktarını belirlemek için bizlere nicel bir ölçü sağlar (yani, bilgiyi/bilginin içeriğini ölçebilmemizi, içerilen bilginin miktarı olarak matematiksel/sayısal bir değer/sayı sunabilmemizi bizlere olanaklı kılar) ve bu enformasyonun kaynaktan hedefe gönderilebilmesi için gerekli iletişim sistemi kapasitesinin hesaplanmasına izin verir. Örneğin, Şekil 4.1'de [53] verilen istatistiksel kanal modellerinden İSK kanalı için C ile gösterilen kanal kapasitesi aşağıdaki gibi hesaplanmaktadır [84].

$$C = 1 + p\log_2 p + (1 - p)\log_2(1 - p) \text{ bit/simge} \quad (4.1)$$

Burada, p parametresi İSK kanalın hata olasılığı olmaktadır. Ayrık değişkenli KAD analizlerinde de kuantum kanal çoğunlukla (istatistiksel kanal modellerinden biri olan) bir İSK kanalı olarak modellenmektedir.

Enformasyon teorisinin başlıca konularından biri olan kodlamanın kullanılmasıyla mesaj sinyallerindeki gereksizlik azaltılabilir ve böylece kanallar çok daha verimli bir biçimde kullanılabilir (kaynak kodlama). Ek olarak, iletilen sinyale sistematik fazlalık eklenebilir ve böylece kanallar çok daha güvenilir olarak da kullanılabilir (kanal kodlama). Kaynak ve kanal kodlama göndericideki iki temel kodlama yöntemidir. Kaynak kodlama sonucunda kaynak sinyali (ses, görüntü, veri vb.) mümkün olan en kısa uzunlukta ikili bit dizilerine çevrilir.

Kanaldan geçirilmek istenen sinyalin alıcıya en az hata ile yollanması ise kanal kodlaması ile yapılabilmektedir. Kanal kodlamada; kaynak kodlayıcısında elde edilen m bitlik ikili dizilere $n - m$ adet daha hata kontrol (denetim) biti eklenerek veri dizisinin kanalın bozucu etkilerinden (gürültü) korunması sağlanır. Kodlanan işaretin alıcıya gizlilik içinde ulaşması için kodlama işlemlerinden sonra şifreleme (kripto) gibi güvenlik arttırıcı işlemler de yapılabilir. Alıcı tarafta, gerçekleştirilen tüm bu işlemlerin tersi yapılmak suretiyle asıl veriye tekrar ulaşılır.

Kuantum kanal, göndericiden alıcıya kuantum sinyalleri (kübit: kuantum bit) göndermek için kullanılan kanaldır. Kuantum kanalın dış ortam ile etkileşimden yeterince izole edilmiş olması gerekir. Ayrıca, kanalın kendisi de kübit (örneğin, polarize edilmiş foton) ile herhangi bir etkileşime girmemelidir. Sonuç olarak, kuantum kanalın kübitleri değiştirmez olması önemlidir.

3. kişiler kuantum kanaldan gidip gelen kübitleri hem gözleyebilir hem de tekrar gönderebilirler. Fizik yasaları dışında kısıtlama olmaksızın kanala erişip istedikleri aktif ve/veya pasif müdahaleleri yapabilecekleri varsayılır. Ancak, yaptıkları ölçümler (gözlem) ilgili kübitleri (gözleneni) değiştireceğinden (hatalara sebep olacağından) varlıklarını da ele verebilecektir.

Mevcut kuantum kanallar kusursuz olmadığı için, arada bir saldırgan olmasa dahi, göndericideki ve alıcıdaki kübitler tamamen aynı olmaz. Alıcının bazı kübitleri farklı olur. Aynı şekilde, kübitleri oluşturmak, almak ve ölçmekte kullanılan tüm diğer elektro-optik ekipmanların ideal olmamasından kaynaklanan hatalar da yine

kuantum kanala mâl edilir. Sonuç olarak, tüm diğer klasik kanallarda da olduğu gibi, kuantum kanal da gürültülü bir kanal olarak kabul edilir.

Mevcut KAD sistemlerinde, göndericideki ve alıcıdaki bit dizisinin farklı olduğunun anlaşılması durumunda anahtar dağıtım işlemi hemen sonlandırılmaz. Aksi halde, mevcut kuantum kanallar hâlihazırda gürültülü olduğu için, anahtar dağıtımı asla mümkün olamaz. Bunun yerine, mevcut bileşenlerin henüz kusurlu olmasından kaynaklanan tüm sistem gürültüsü de saldırgana mâl edilerek, belirli bir hata oranına dek anahtar dağıtımı yapılmasına izin verilir.

En son güvenlik ispatları [85], göndericideki ve alıcıdaki bit dizisi arasındaki hata olasılığının $p_{eşik}$ = 20% değerlerine (bu ayrık değişkenli KAD türleri için verilen bir değerdir. Aynı kaynakta sürekli değişkenli KAD türleri için $p_{eşik}$ = 27.6% olarak verilmektedir) kadar çıkması durumunda dahi güvenli gizli anahtarların elde edilebileceğini göstermektedir. Buna göre,

- Hata miktarı bu eşikten daha yukarıda ise güvenli bir gizli anahtar elde edilemez. Bu nedenle, saldırganlar giderildikten sonra tekrar denenmek üzere, KAD protokolü derhal sonlandırılır.
- Hata miktarı bu eşikten daha aşağıda ise halen güvenli bir anahtar elde edilebilir. KAD protokolü, iki taraf arasında ortak ve gizli bir anahtarı elde etmek amacıyla işleyişine devam eder.

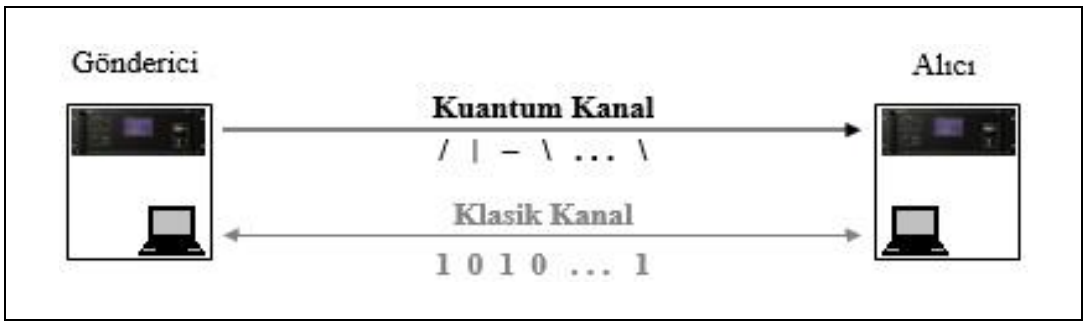
Bu bağlamda, önceki bölümde ideal durum için anlatılan (sistem gürültüsünün olmadığı, gürültünün sadece saldırgandan kaynaklandığı, bir hata sezilmesi durumunda protokolün derhal sonlandırıldığı, saldırganlar uzaklaştırıldıktan sonra anahtar dağıtımının tekrar çalıştırıldığı) BB84 protokolü, daha genelde mevcut KAD sistemleri, pratikte şu iki ana kısımdan oluşmaktadır.

- Kuantum Kanal Üzerinden Gerçekleştirilen Kuantum Kısım

Sadece göndericiden alıcıya kübitleri (örneğin, polarize edilmiş fotonlar) iletmek için kullanılır. Yani, kuantum kanal üzerindeki iletişim tek yönlüdür. İletim ortamı (örneğin, fiber optik kablo, açık hava boşluğu vb. optik ortamlar) yanında kübitleri oluşturmak, almak ve ölçmekte kullanılan tüm diğer ekipmanlar (tek foton üreteci, foton polarize edici, foton polarizasyon ölçücü, foton detektörü vd. elektro-optik bileşenler) da kuantum kanalın kapsamı içinde kabul edilir.

• Klasik Kanal Üzerinden Gerçekleşen Klasik Kısım

Göndericiden alıcıya ve alıcıdan göndericiye klasik sinyalleri (0, 1'ler) göndermekte ve almakta kullanılır. Yani, klasik kanal üzerindeki iletişim iki yönlüdür. Temel olarak, saldırganın varlığının tespit edilmesi ve ortak gizli anahtarın belirlenmesinde başlıca rol sahibidir.



Şekil 4.2: KAD'da temel bileşenler.

Bu yapıda (bkz. Şekil 4.2 [25]) her iki safhadaki tüm iletişimler için aynı optik iletişim altyapısı (aynı kanal) kullanılabilir. Klasik kanal olarak, bilinen telefon hattı, GSM, İnternet ve benzeri kablolu/kablosuz herkese açık iletişim ortamları da olabilir. Bu yapı üzerinde BB84 KAD protokolü, özetle, şu şekilde çalışır.

• Foton İletişimin Gerçekleştirildiği İlk Safha

Alice rasgele bir A bit dizisi üretir (bu amaçla bir KRSÜ kullanılabilir) ve her bir bitini tek tek fotonlarla kuantum kanaldan Bob'a gönderir (güvenlik açısından ihtiyaç duyulan rasgeleliklerin bir KRSÜ ile üretilmesi çok önem taşımaktadır). Kuantum kanal boyunca meydana gelen bozulmalar nedeniyle Bob'daki alınan bit dizisi $B = A \oplus e$ şeklinde olur (e: gürültü).

• Saldırganın Tespiti ve Anahtarı Belirlemede Kullanılan İkinci Safha

Klasik kanal üzerinden bir saldırgan testi yapılır ve koşullar yerindeyse KAD'a uygun hata sezme ve düzeltme teknikleri kullanılarak B'deki tüm hatalı bitler bulunup düzeltilir, $B = A$ yapılır.

KAD'da kuantum kanal üzerinden olan iletişimde sadece tek tek fotonlar, her bir bite karşılık yalnızca tek bir foton halinde, alıcıya gönderilir. Bu şekilde bir iletişim daha sonra iletişime bir müdahale olup olmadığının sezilebilmesi gibi klasik

iletişimde eşdeğeri olmayan bir özelliğe de olanak verir (yani, asıl iletişim öncesi taraflarca iletişimin dinlenip dinlenmediği anlaşılabilir). Ancak, kuantum kanaldaki kusurlar, gönderici ve alıcıdaki elektro-optik bileşenlerin ideal olmaması gibi sistem gürültüsü ve iletişime müdahale etmeye çalışan saldırganların da varlığı nedeniyle gönderilen ve alınan bit dizisi tamamen aynı olmaz. Aralarında biraz farklar olur. Kuantum kanal üzerinden anlamlı bilgi gönderimi mümkün olmadığından bu aşamada alıcıya herhangi bir hata sezme ve düzeltme bilgisi vb. göndermek de mümkün değildir. Dolayısıyla, bu işlemlerin başka bir aşamada ve başka bir şekilde yapılması gerekir.

KAD'da ikinci aşama herkese (saldırgana da) açık, ancak kimlik doğrulamalı, olan gürültüsüz klasik iletişimdir. İletişimin dinlenilip dinlenilmediğinin tespit edilmesi, hata sezme ve hata düzeltme gibi gerekli tüm diğer son-işleme adımları da sadece bu aşamada gerçekleştirilir. Hata sezme ve düzeltme için gerekli olan ilave bilgi alıcıya bu aşamada klasik kanal üzerinden gönderilir. Böylece, KAD protokolü sonunda sadece gönderici ve alıcı arasında şifreleme, şifre çözme için kullanılabilecek tamamen ortak (aynı) ve gizli bir bit dizisi (gizli anahtar) üzerinde anlaşılmış olur.

4.1.1. Eşlik Kontrolü

Eşlik, bir bit dizisindeki bitlerin değerlerinin toplamının modülo 2'deki karşılığıdır. İki dizinin eşliklerinin farklı olması bu iki dizinin kesinlikle farklı olduğunu gösterir ve hata sezmede işimize yarar. Eşlik yöntemi, sadece tek sayıdaki (1, 3, 5, ...) bit hatalarının sezilebilmesine olanak vermektedir.

Yukarıda bahsedilen eşlik hesaplama yöntemi (ikilik tabanda toplama) bitsel XOR işlemi ile aynıdır. Yani, bir bit dizisindeki tüm bitlerin XOR'lanması dizinin eşliğini verecektir. Bitsel XOR işleminin tablosu aşağıdaki gibidir.

Tablo 4.1: Bitsel XOR işlemi ($\oplus$).

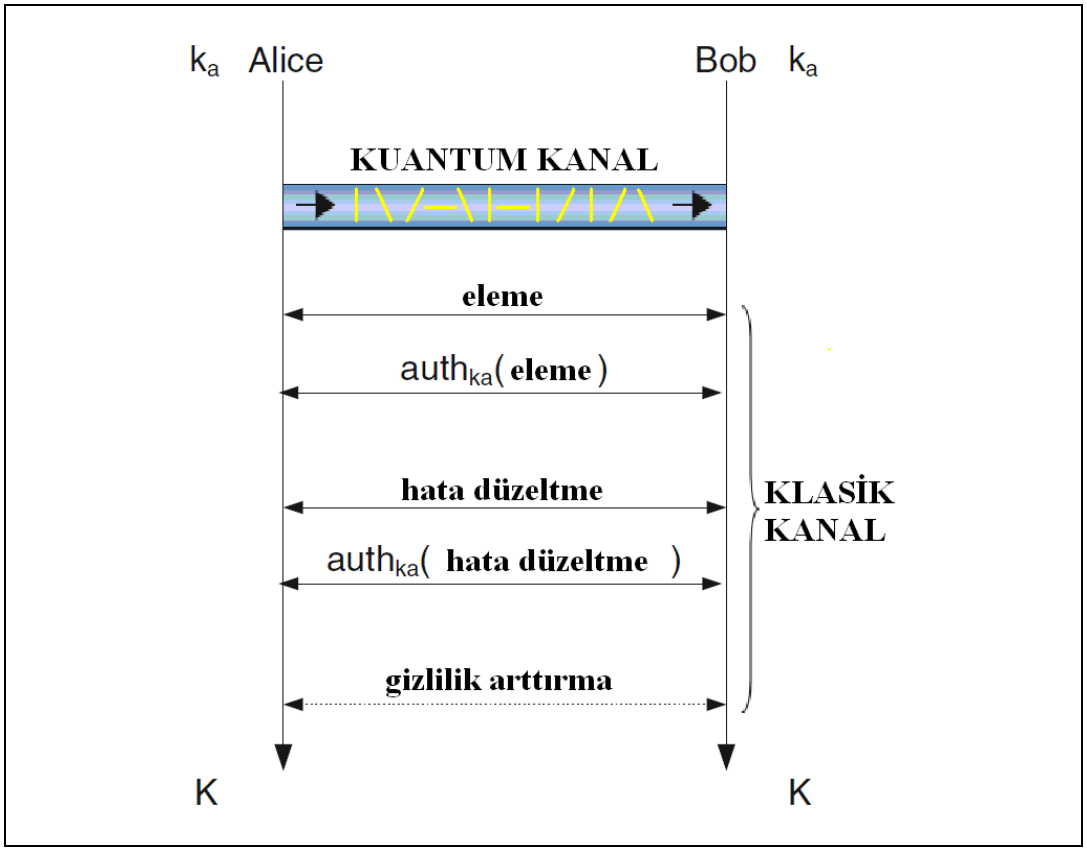
$\oplus$	0	1
0	0	1
1	1	0

Tablo 4.1'den de görüldüğü gibi, 0 ile XOR'lamanın hiçbir etkisi olmadığına, 1 ile XOR'lamanın ise (ikili) sayıyı değiştirdiğine dikkat ediniz. Ayrıca,

- XOR'lanacak iki bit aynı ise XOR sonucu kesinlikle 0 olmaktadır. Buna göre, aynı uzunlukta olan iki bit dizisi XOR'landığında sonuçta hep 0 olan bir bit dizisi (matematiksel/sayısal değer olarak da 0 değeri/sayısı) elde ediliyorsa bu iki dizi kesinlikle aynıdır demektir.
- XOR'lanacak iki bit farklı olduğunda ise XOR sonucu 1 olmaktadır.

4.2. Sonraki İşlemler

BB84 protokolünde, daha genel olarak KAD protokollerinde, iletişim kuantum kanalda başlar ve klasik kanalda sona erer.



Şekil 4.3: BB84 protokolü.

Şekil 4.3'te [86] BB84 KAD protokolünün özet akış diyagramı verilmektedir. Burada, k_a kimlik doğrulama (authentication) için gerekli önceden paylaşılmış ortak

ve gizli anahtardır. Tarafların en az bir defa bir araya gelerek bu k_a anahtarı üzerinde anlaşması gerekir. K ise BB84 protokolü çalıştırıldıktan sonra üretilen ortak ve gizli final anahtardır. Sonraki kullanımlarda K anahtarının bir kısmı k_a anahtarı olarak da kullanılabileceğinden artık tarafların bir araya gelmesine gerek kalmaz.

KAD'da kuantum olan kısmın sadece, kuantum durumların oluşturulduğu, iletildiği ve ölçüldüğü, kuantum kanal üzerinden gerçekleştirilen işlemler olduğuna dikkat edin. Özetle, gönderici tarafta üretilen aday anahtar bitlerinin tek tek foton tanecikleriyle alıcıya ulaştırılmasından oluşur. Bu kısmın sonucunda ham anahtarlar oluşur. Göndericinin ham anahtarı kuantum kanaldan gönderdiği tüm bitlerdir. Alıcının ham anahtarı ise ölçebildiği tüm bitlerdir. Gönderici ve alıcı ellerindeki bitlerinin ve kullandıkları polarizasyon tabanlarının kaydını tutarlar.

Ham anahtarlardan K gizli anahtarına götüren sonraki tüm adımlar ve iletişimler kimlik doğrulamalı klasik haberleşme kanalı üzerinden gerçekleştirilir. Bunları da 3 ana başlık altında özetlemek mümkündür.

- Eleme

Klasik kanaldaki ilk aşama eleme fazıdır. Özetle, alıcının ölçüm sonuçlarının değerlendirilmesinden ibarettir. Alice ve Bob, bu aşamada hangi bitlerin kullanılacağını hangi bitlerin de atılacağını görüşürler. Pratikte, kuantum kanaldaki kayıplardan dolayı göndericinin ve alıcının ham anahtar uzunlukları farklı olabilir. Yani, göndericinin gönderdiği her foton alıcıya ulaşamayabilir. Bu olunca, alıcı öncelikle tespit ettiği kübitlerin indeksini kimlik doğrulamalı klasik kanaldan açıklayarak göndericiyi haberdar eder. Daha sonra, bu bitlerden de yine sadece polarizasyon tabanlarının uyumlu olduğu durumlardakiler seçilir. Bu adımın sonucunda ortaya elenmiş anahtarlar çıkar. Göndericinin ve alıcının elenmiş anahtarı büyük ölçüde aynıdır ve uzunlukları da artık kesinlikle eşittir. Alıcıya ulaşamayan ve tabanların uyuşmadıkları bitler ise atılırlar (elenirler). Literatüre göre mevcut eleme fazı hızları $R_s \leq 6.29$ Mbps civarındadır [96] (bu tür çalışmaların doğası gereği açıklanmayan daha yüksek değerler de mevcut olabilir elbette).

- Uzlaştırma

Daha sonra, Alice ve Bob bilgi uzlaştırma fazına geçerler. Hata sezme ve düzeltme fazı olarak da ifade edilebilir. Kuantum kanal gürültüsüz bir kanal olmadığı için, Alice ve Bob'un tüm bitleri aynı değildir. Bob'un bitlerinde bir miktar hata

vardır ve bu hataların tümü bu fazda tespit edilir ve düzeltilir. Bu aşamanın sonunda Alice ve Bob çok yüksek bir olasılıkla tamamen aynı bit dizisine sahip olurlar. Ancak, bu dizi henüz anahtar olarak kullanılamaz. Eve'in dizi hakkındaki bilgisinin de dikkate alınması gerekir. Bilgi uzlaştırma fazı bir sonraki bölümde biraz daha detaylı olarak ele alınacaktır.

• Gizlilik Arttırma

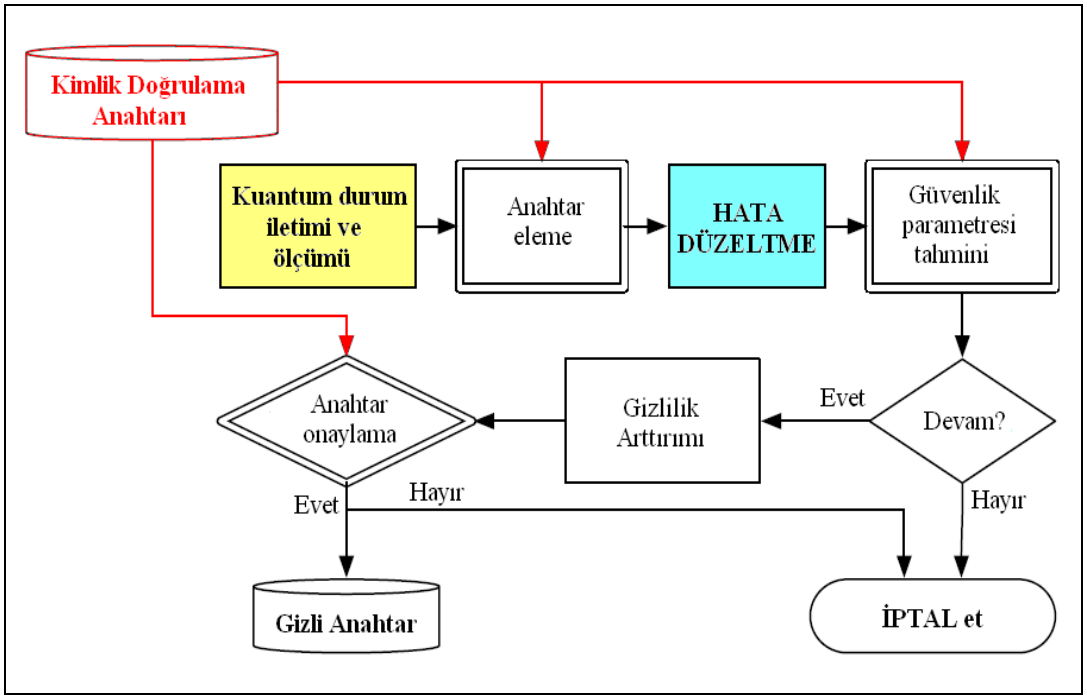
Eve, önceki aşamalarda (uzlaştırma, eleme ve belki kuantum iletişim esnasında da) bazı bilgiler edinmiş olabilir. Bu nedenle, Alice ve Bob dizilerini bir özet fonksiyonu aracılığıyla daha küçük bir kümeye karşı düşürmelidir. KAD açısından bunun güvenlik ispatlı bir fonksiyon olması gerekir (ancak, anahtar saldırganlara zaten gizli tutulduğundan tersi alınamaz olmasına gerek olmayabilir). Böylece, Eve'in bilgisi neredeyse sıfıra düşer. Gizlilik arttırma olarak adlandırılan bu aşamadan sonra ancak Alice ve Bob sadece kendilerinin bildiği ortak ve gizli bir anahtara sahip olurlar.

Güvenlik için klasik kanal üzerinden gerçekleştirilen tüm iletişimlerin kimlik doğrulamalı olması gerekir (tarafların birbirlerinin kimliklerinden emin olmaları önemlidir). Aksi halde, KAD protokolü saldırganların kanal üzerinden gidip gelen mesajları değiştirmesine ve saldırganlarca yapılacak ortadaki adam saldırısına karşı tamamen savunmasız hale gelmektedir. KAD'a uygun (güvenlik ispatı olan) bir kimlik doğrulama sistemi kullanılarak kullanılacak bitler üzerinde anlaşılmış, Eve tarafından yapılacak ortadaki adam saldırısı engellenmiş ve mesajları değiştirmediğinden de emin olunmuş olunur.

Klasik kanalın kimlik doğrulamalı olmasının zorunlu olduğuna dikkat edin. Yani, gönderici ve alıcı birbirlerinin kimliklerini mutlaka doğrulayabilmelidir. Böyle olması halinde, araya giren meraklı kişiler kimlik doğrulamalı klasik kanaldaki mesaj alış-verişini sadece pasifçe dinleyebilirler; aktif herhangi bir müdahalede (değiştirme vs.) ise bulunamazlar. Sonuç olarak, meraklıların klasik kanaldan gidip gelen bilgileri sadece gözleyebileceği varsayılır. Bununla birlikte, kuantum iletişimin aksine klasik iletişimin dinlendiği anlaşılamaz.

Şekil 4.4'ten [53] de görüldüğü gibi KAD'da kimlik doğrulamanın yeri ve önemi büyüktür. KAD'ın güvenlik düzeyi koşulsuz (ispatlanabilir) güvenliktir. KAD protokollerinin güvenliği ve bu güvenlik düzeyinin muhafaza edilebilmesi için kullanılacak kimlik doğrulama sisteminin de koşulsuz güvenlikte olması gerekir. Bir

örnek olarak, (önceden paylaşılmış bir k_a ve bir evrensel özet fonksiyonu kullanımına dayanan) Wegman-Carter kimlik doğrulama yöntemi verilebilir [87].



Şekil 4.4: KAD'da kimlik doğrulamanın yeri. Klasik kanal üzerinden gerçekleştirilen iletişimlerde kimlikler doğrulanmalıdır.

Kuantum kriptografi, KAD, güvenliği, aşamaları, analizleri ve ispatları hakkında daha fazla bilgi için [88]-[92]'ye başvurulabilir.

Tez çalışmamızın ana konusu olan bilgi uzlaştırmaya, yine BB84 protokolü örneğinde, aşağıda devam edilmektedir.

4.2.1. Bilgi Uzlaştırma

Haberleşme sistemlerinde amaçlanan, herhangi bir biçimdeki bilgiyi zaman ve uzay içinde kaynak adı verilen bir noktadan, alıcı (kullanıcı) olarak adlandırılan bir başka noktaya hatasız olarak taşımaktır [83]. Bir yerden diğerine iletilen veri, güvenilir bir şekilde taşınmalıdır. Ancak, çoğu durumda, fiziksel bağlantı iletilen bitlerin tümünün hatasız olarak taşınmasını garanti etmez [93].

Bu bölüme kadar, yer yer, deterministik sinyallerin (0, 1'ler) bir kanal üzerinden iletimini tartıştık ve iletişimde "rasgelelik" kavramının oynadığı temel rol üzerinde durmadık. Rasgele sözcüğü "öngörülemez, tahmin edilemez, önceden

bilinemez” anlamlarına gelir. Eğer bir kanalın ucundaki alıcı, üretici kaynağın mesaj çıktılarını önceden bilebilseydi, iletişime zaten hiç gerek kalmazdı. Dolayısıyla, mesaj kaynağında bir rasgelelik vardır (bu durum bilgi teorisinin neden olasılıksal bir teori olduğuna da yanıt olabilir). Ayrıca, iletilen sinyallere her zaman sistemde eklenen gürültü eşlik eder. Bu gürültü sinyalleri de öngörülemezler.

Gürültü, haberleşme kanalına verilen sinyallerde bozulmalara (hatalara) neden olur; gürültünün varlığı analog ve sayısal iletişim sistemlerinin performansını düşürür (olumsuz yönde etkiler). Gürültünün iletişim sistemlerinin performansını ne kadar etkilediği kanal çıkışındaki (hemen alıcı girişindeki) sinyal gürültü gücü oranıyla veya hata olasılığıyla ölçülür. Sinyal gürültü gücü oranı analog iletişim sistemlerinin performansını ölçmekte kullanılırken kanal hata olasılığı sayısal iletişim sistemlerinde performans ölçütü olarak kullanılır [84].

Veri iletiminde patlama (burst) ve rasgele (random) hata olarak adlandırılan iki tip hatayla karşılaşılır. Patlama hatasında, çevre koşulları nedeniyle bir süre alıcıya gerçek olmayan anlamsız bir bilgi gelir. Bu süre 1-100 ms olabilir; bu süre içinde iletim ortamından geçen tüm bitler bozulabilir. Rasgele hatada, iletim yolundaki gürültü nedeniyle veri içindeki rasgele bitlerin bozulması söz konusudur. Bu bazen tek 1 bit olabileceği gibi bazen birkaç bit de olabilir [94].

Bu çalışmada esasen, alıcı girişinde rasgele sinyalle birlikte bulunan toplanır gürültüyle ilgilenmekteyiz. Alıcı girişinde, rasgele gürültüyle karışmış rasgele bir sinyal vardır. Gürültü, kanalın bizzat kendisindeki kusurlar veya kanalda olmayan çok çeşitli (dış) nedenlerden dolayı kaynaklanmaktadır.

Hata denetimi, veriyi hatalardan korumak (hatasız iletim) için bir yol sağlar. Bilgi sistemlerinde, gönderilen veri bitlerinin hata sezme ve düzeltmeye uygun kodlamalar yapılarak iletilmesi suretiyle, alıcı tarafta hataların tespiti ve kimi zaman da belirli oranda düzeltilmesi hata denetim algoritmasının sorumluluğundadır. Böylelikle, daha üst katmanlara (görünürde) hatasız bir bağlantı sağlanmış olur. Uygulamada popüler olan iki hata denetimi stratejisi mevcuttur.

- ARQ (Otomatik Tekrar İsteği)

Hata olup olmadığını sezmeyi ve eğer hata varsa bozulan verinin göndericiden yeniden iletilmesini sağlar. Göndericiden alıcıya hata olduğunu tespit edebilmesi için asıl bitler ile birlikte bir miktar takviye bilgi (örneğin, eşlik bitleri) de gönderilir. Alıcı her hata sezdiğinde, göndericiye aynı veriyi tekrarlama isteği (ARQ) gönderir.

Hata sezilmesi, yani, kendisine her ARQ bilgisi gelmesi, halinde de gönderici ilgili veriyi alıcıya tekrar gönderir. ARQ'de hata sezme için kullanılan algoritmalara örnek olarak yukarılarda bahsedilen eşlik kontrolü yöntemi verilebilir.

- FEC (İleri Yönlü Hata Denetimi)

Hata sezmeyi ve eğer hata varsa düzeltmeyi sağlar. Bu amaçla, göndericiden alıcıya yeteri kadar takviye (fazlalık, ilave, ek) bit gönderilir. Hem hata sezme hem de hata düzeltme yapıldığı için gönderilen ek bitler ARQ'ye göre çok daha fazladır. FEC stratejisi, bilgiyi tekrar göndermenin çok güç veya imkânsız (çok pahalı ya da gecikmeli) olduğu bağlantılarda kullanılır (Kimi zaman da düzeltim işlemi, yeniden gönderimden daha çok zaman alır; bu durumlarda da verinin yeniden iletimi tercih edilir). Uygun kodlama ile (kanalda yapılacak bit hata olasılığı testleri ile yolda bozulabilecek bit sayısının üst sınırının bilindiği varsayılır) hatanın alıcıda (belirli bir bozulma ölçüsüne kadar) düzeltilmesine çalışılır, verinin tekrar gönderilmesi istenmeden hata giderilebilir (Düzeltilemeyen hata olduğu zaman da veri paketinin yeniden gönderilmesi kaçınılmaz olur). FEC algoritmalarına örnek olarak aşağıda da bahsedeceğimiz 2-boyutlu eşlik kontrolü yöntemi verilebilir. ARQ ağı, FEC ise CPU'yu birden çok kullanır ve FEC genellikle donanımda uyarlanır.

İletim ortamı çok güvenli olsa bile, kimi uygulamalar hatasız (güvenilir) iletim yapılmasını da isterler. Veriler iletilirken bazı bitlerin bozulması olasılığı çok az dahi olsa, 1 bit bile bozulsa, birçok uygulama tarafından kabul edilemez, tüm verinin yanlış anlaşılmasına sebep olabilir. Bu nedenle, iletişim yapılırken bozulma olup olmadığının anlaşılması (hata sezme teknikleri ile), bozulma olmuşsa da hatanın tamamen giderilmesi (hata düzeltme teknikleri kullanılarak) zaruridir.

Hata sezme ve düzeltme yöntemlerinin modern sayısal iletişim, depolama, hesaplama vb. tüm bilgi işleme sistemlerinin sağlıklı olarak çalışmalarında önemi büyüktür: özetle, hayati bir önemi vardır. Hata sezme ve düzeltme teknikleri olmadan bu sistemlerin hiçbirinin çalışması mümkün değildir dersek yanlış olmaz. Örneğin, modern kriptografide şifreli mesajın tek 1 biti bile değişse asıl mesaja bir daha asla ulaşılamayacak çok güçlü şifreleme ve şifre çözme teknikleri ile çalışılmaktadır. Aynı durum gizli anahtar için de söz konusudur. Dolayısıyla, şifreli mesajı ya da anahtarı iletirken oluşabilecek hataları sezmek ve düzeltmek için yöntemlere ihtiyaç vardır. Aksi takdirde, anahtarı ya da şifreli mesajı alıcıya hatasız olarak ulaştıramadıktan sonra, kriptografi kullanımının bir önemi kalmaz.

Modern hata sezme ve düzeltme yöntemlerinin temel mantığı, zekice seçilmiş sistematik bir fazlalık (ilave) bilgi de göndermek yolu ile orijinal mesajın gürültülü haberleşme ortamları üzerinden güvenilir iletimini garantilemektir. Kalabalık bir odada (gürültülü kanal) bir arkadaşımızla sohbet (iletişim) ettiğimizi düşünelim. Odadaki gürültü arttıkça birbirimizi duymamız (mesajımızı iletmemiz) daha da zorlaşacaktır. Sohbeti sürdürebilmek için ya daha yüksek sesle konuşmaya çalışırız (sinyal gücünü arttırırız) ya da söylediklerimizi tekrarlarız (mesajı tekrar göndeririz). İşte hata sezme ve düzeltme kodlarıyla yapılan da bu ikinci yaklaşımdır. Alıcının mesajı yeniden oluşturabilmesi için alıcıya mesajla birlikte, mesajla ilgili bir miktar da ilave bilgi gönderilir. Alıcı, ilave bilgi ile mesajın sağlamasını yaparak hata olup olmadığının tespitini ve hatta hata olması halinde de düzeltilmesini yapar. Bölüm 4.1.1'de eşlik kodları ile hata sezmenin yapılabileceğinden bahsetmiştik. Aşağıda da hata düzeltmenin nasıl olduğunu göstermeye çalışacağız.

FEC için kullanılan algoritmalardan biri, 2-boyutlu eşlik kontrolüdür. Eşlik kontrollerine dayanan 2-boyutlu eşlik kontrolü kodu, hata sezme yanında mesajda oluşan 1-bitlik hataları düzeltebilme olanağı da verir. Hepsi 0 bitlerinden oluşan, aşağıdaki gibi, 20 bitlik bir mesajımız olduğunu varsayalım:

00000 00000 00000 00000

(burada, yöntemin daha kolay anlaşılabilmesi açısından bitlerin hepsi 0 olarak seçilmiştir). Mesajdan aşağıdaki gibi 2-boyutlu bir dizilim oluşturulur:

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Mesajın ilk 5 biti 1. satıra, ikinci 5 biti 2. satıra, üçüncü 5 biti 3. satıra, dördüncü 5 biti de son satıra yazılmıştır. Sonra, bu 4×5'lik matrisin her bir satırlarının eşlikleri hesaplanır ve sonuç ilgili satırın sağ tarafına yazılır. Aynı şekilde, sütunlarının eşlikleri hesaplanır ve sonuç ilgili sütunun altına yazılır:

$$\begin{matrix} 0 & 0 & 0 & 0 & 0 & & \mathbf{0} \end{matrix}$$

0	0	0	0	0	**0**
0	0	0	0	0	**0**
0	0	0	0	0	**0**
0	**0**	**0**	**0**	**0**	**0**

Hesaplanan eşlikler koyu renklerle yazılmıştır. En sağ alt köşedeki eşlik biti eşliklerin eşliği hesaplanarak elde edilmektedir. Gönderici tarafta elde edilen bu son 5×6'lık matris (satırlarının yan yana eklenmesiyle oluşan 30 tane bit) kanala verilerek alıcıya gönderilir. Kanalda 3. satırın 4. sütunundaki bitin bozulduğunu varsayalım. Alıcı gelen bitlerden 5×6'lık matrisi yeniden oluşturur:

0	0	0	0	0	**0**
0	0	0	0	0	**0**
0	0	0	1	0	**0**
0	0	0	0	0	**0**
0	**0**	**0**	**0**	**0**	**0**

Bozulan bit kırmızı renkle gösterilmiştir (3. satırın soldan 4. biti, 4. sütunun üstten 3. biti, 3. satır ve 4. sütunun kesişimi olan bittir). Alıcı oluşturduğu 5×6'lık matrisin satırlarının ve sütunlarının eşliklerini hesaplar (En sağ alt köşedeki eşlik biti yeni elde edilen eşliklerin eşliği hesaplanarak elde edilmektedir). Eğer sonuçlar 0 ise, kontrol tamamlanmıştır; sonucun sıfır olmaması hataya işaret eder.

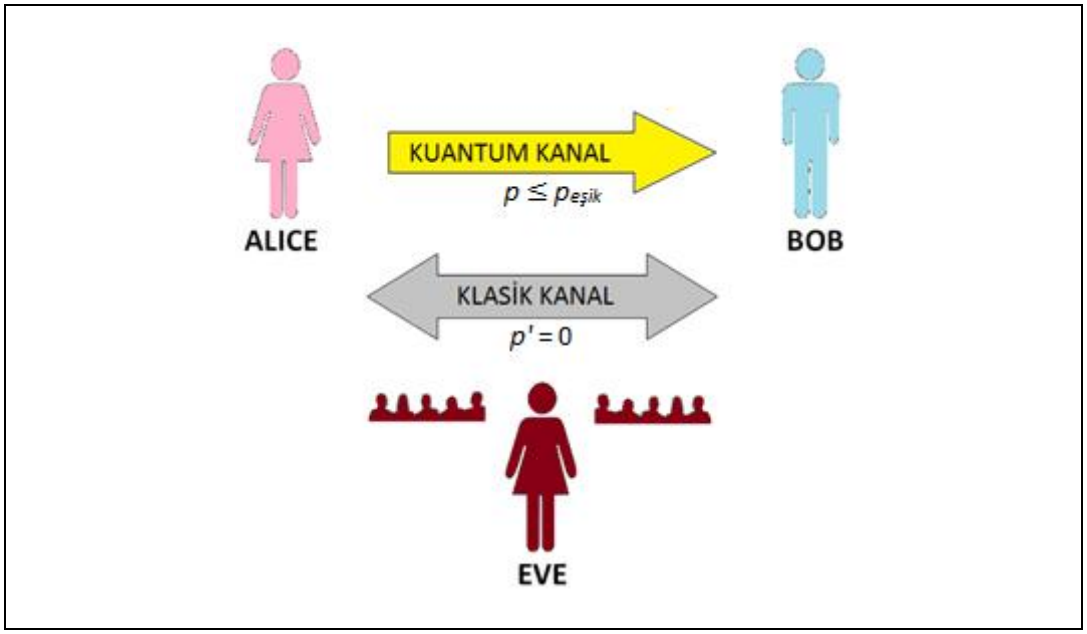
0	0	0	0	0	0	**0**
0	0	0	0	0	0	**0**
0	0	0	1	0	0	**1**
0	0	0	0	0	0	**0**
0	0	0	0	0	0	**0**
0	**0**	**0**	**1**	**0**	**0**	**1**

Hesaplanan eşlikler koyu renklerle yazılmıştır. Yukarıda, 3. satırın ve 4. sütunun eşliklerinin 1 çıkması (1 çıkan eşlikler kırmızı renklerle de gösterilmişlerdir) hem bir hata olduğunu göstermektedir hem de hatanın bulunduğu yerin koordinatlarını (yerinin 3. satır, 4. sütunda olduğunu) vermektedir (Hata olmasaydı 5×6'lık matrisin bütün eşliklerinin de 0 çıkması gerekirdi).

Alıcı belirlenen koordinattaki biti değiştirerek (1 ise 0 yaparak ya da 0 ise 1 yaparak) gerekli hata düzeltmeyi yapmış olur. Kodlama teorisinde, bozulmamış mesaj ve bozulmamış ilave bilgiden oluşan yeni veriye kod sözcüğü denilmektedir. Alıcı, aldığı veriden hata düzeltme sonucu yeniden elde ettiği kod sözcüğünden ilave bilgiyi çıkararak asıl mesaja kendisi de ulaşmış olur.

2-boyutlu eşlik kontrol kodu için yukarıda verilen örnekte, 20 bilgi bitine karşılık 10 tane de eşlik biti ilave edilerek alıcıya toplamda 30 tane bitlik bir kod sözcüğü gönderilmektedir. 2-boyutlu eşlik kontrol kodları, 2 bitlik hataları sezebilirken sadece 1 bitlik hataları düzeltebilir.

Hata sezme ve düzeltme, teknikleri, analizleri ve uygulamaları hakkında daha fazla bilgi için [95]'e de başvurulabilir.



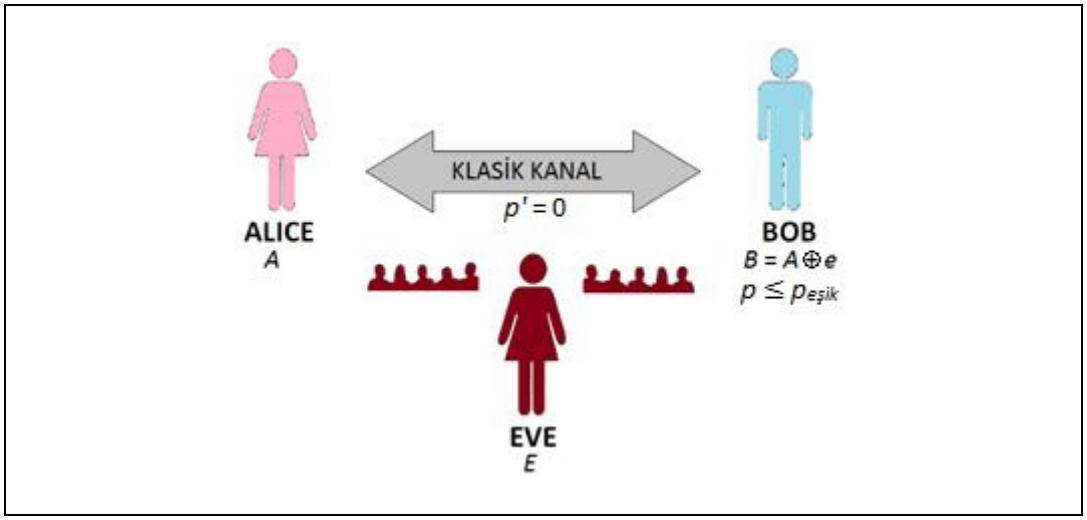
Şekil 4.5: KAD'da gürültü. Gürültü sadece kuantum kanaldan ve bu esnada saldırganın gerçekleştirdiği müdahalelerden kaynaklanmaktadır. Güvenlik açısından, tüm gürültü saldırgana mâl edilir.

Şekil 4.5'te gösterilmeye çalışıldığı gibi, KAD'da gürültülü olan kanal sadece kuantum kanaldır. Hata olasılığı, $p_{eşik}$ değerine kadar çıkabilmektedir (Güvenlik

ispatları gereği hata miktarı bu eşikten daha yukarıda ise güvenli bir anahtar elde edilemeyeceğinden KAD protokolü sonlandırılır).

KAD'da güvenlik ispatlarının onay verdiği maksimum hata miktarının $p_{eşik}$ değeri kadar olması bizim illa da $p_{eşik}$ değeri ve civarındaki ($p_{eşik}$ değerinin üstüne çıkılamaz) hata olasılıklarıyla çalışmamızı gerektirmez. Aslında, mevcut KAD sistemlerindeki kuantum kanalların p hata olasılıkları bellidir; hemen hemen en fazla 4% civarındadır [96]. $p_{eşik}$ değeri ve civarındaki hata olasılıkları ile karşılaşılması şu iki anlama gelecektir: *i*) arada bir saldırganın olduğu, *ii*) kanalın gerçekten çok gürültülü olduğu. Ancak, literatüre göre ilk durum daha olasıdır.

Gürültünün varlığı nedeniyle kuantum iletişimin sonucunda göndericideki ve alıcıdaki anahtarlar farklı olur ve bu haliyle kullanışsızdırlar. Ortak anahtarın elde edilebilmesi amacıyla, daha sonra, klasik kanal üzerinden, alıcı taraftaki anahtarın tüm hatalarının bulunup düzeltilmesine KAD literatüründe (gizli anahtar ya da bilgi) uzlaştırma denilmektedir. KAD'ın günümüzde üzerinde en çok çalışılan, darboğaz niteliğindeki önemli problemlerinden biridir. Bu konu, aslında, haberleşmecilerin çok iyi bildiği, bir tür hata sezme ve düzeltme uygulamasıdır. Bununla birlikte, aşağıda değinilecek çok hassas bazı farklılıkları da vardır.



Şekil 4.6: KAD'da bilgi uzlaştırma. Orijinal mesaj göndericideki A, bozulmuş mesaj alıcı taraftaki B ve aralarındaki kanal gürültüsüzdür. KAD protokolüne ve güvenlik ispatına bağlı olarak p hata oranı $p_{eşik}$ değerlerine kadar çıkabilmektedir.

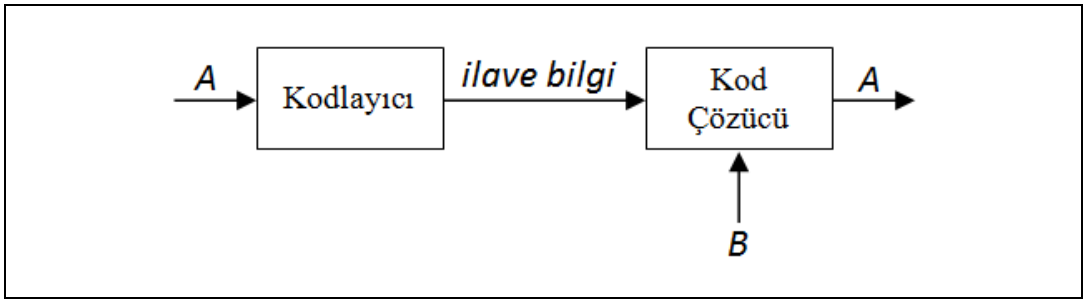
Bilgi uzlaştırma (BU), KAD'da tamamen klasik olan, herhangi bir kuantum mekaniği/fiziği bilgisi gerektirmeyen, aşamalardan bir tanesidir. Uzlaştırma fazına gelindiğinde karşılaşılan durum, özetle, şöyledir (ayrıca, bkz. Şekil 4.6): Göndericide

N uzunluklu bir rasgele bit dizisi A (orijinali), alıcıda N uzunluklu bir rasgele bit dizisi B (bozulmuş hali) vardır ve aralarındaki hata olasılığı da p kadardır, $p \leq p_{eşik}$. Bu aşamada, gereken tüm iletişimler gürültüsüz bir klasik kanal üzerinden yapılmak suretiyle $B = A$ yapılmaya, alıcı taraftaki tüm hatalar bulunup düzeltilmeye, çalışılır. Burada, dikkat çekilebilecek önemli bazı noktalar şunlardır.

- Güvenlik açısından, sadece $p \leq p_{eşik}$ değerindeki hataları düzeltebilen BU yöntemlerinin kullanılması yararlı olabilir.
- BU'nun başarıyla sonuçlanabilmesi için alıcı taraftaki tüm hataları sezebilen ve düzeltebilen yöntemler kullanılmalıdır.
- Kullanılan BU yönteminin sistemin performansı (hızı vb.) üzerinde olumsuz etkisi olmamalıdır. Örneğin, en az kendisinden bir önceki aşama olan eleme fazının R_s hızıyla aynı hızda olmalıdır.
- Hata sezme ve düzeltme gürültülü (ve pahalı) bir kuantum kanal yerine gürültüsüz bir klasik kanal üzerinden yapılmaktadır. Böylece, hatalar ve gönderilen ilave bilginin de daha az olması nedeniyle hata sezme ve düzeltme de çok daha kolay, hızlı, ucuz ve güvenli olabileceği için böylesi bir uygulama daha tercih edilebilir bir durumdur.
- Klasik kanal aynı zamanda kimlik doğrulamalı olup, bu nedenle, üzerindeki trafik sadece pasif olarak dinlenebilir (yani, değiştirme yapılamaz). Bununla birlikte, Eve'in gizli anahtarı asla elde edememesi için hata sezme ve düzeltme amacıyla gönderilen ilave bilginin de mümkün olduğu kadar az olması gerekir.

KAD'da hatalar sadece kuantum kanaldan kaynaklanır. Bu nedenle, kuantum iletim sonunda Bob'daki bit dizisi Alice'tekinin $p \leq p_{eşik}$ kadar bozulmuş halidir. Bu hata daha sonra tamamen klasik kanal üzerinden gerçekleştirilen ilave iletişimlerle düzeltilmeye çalışılır. Klasik kanal hâlihazırda güçlü hata sezme ve düzeltme teknikleri ile korunmakta olan modern bir iletişim kanalı olduğundan kanal hata olasılığı $p' = 0$ kabul edilir (ya da ihmal edilebilecek kadar küçüktür). Yani, KAD'da mesaj (gizli anahtar) bitleri ve ilave bitler (eşlik bitleri) göndericiden alıcıya farklı kanallardan gönderilmektedir. Önce gürültülü kuantum kanaldan gizli anahtar bitleri gönderilirken daha sonra gürültüsüz klasik kanaldan eşlik bitleri gönderilir. Alıcı, bu iki bilgiyi birleştirerek gizli anahtardaki tüm hataları düzeltir.

BU problemi, aslında, yine haberleşmecilerin çok iyi bildiği bir konu olan bir yan bilgi ile kaynak kodlama (asimetrik Slepian-Wolf kodlama, dağıtık kaynak kodlama) durumu (problemi) olarak da görülebilir (bkz. Şekil 4.7 [97]).



Şekil 4.7: Yan bilgi ile kaynak kodlama. Alıcı, elindeki *B* yan bilgisi ve göndericiden gelen (ilişkili) ilave bilgi ile göndericideki (ilişkili) *A* bilgisini aynen elde edebilmektedir.

Böylece, Slepian ve Wolf tarafından da gösterildiği gibi [98], [99], Bob tarafındaki bir *B* bilgisi ve Alice'teki bir *A* bilgisinin uzlaştırılabilmesi için Alice'in Bob'a göndermesi gereken minimum (en az) ilave bilgi miktarı (alt sınır),

$$I_{\text{minimum}} = H(A|B) \text{ bit/simge} \tag{4.2}$$

kadardır. Bu teorik sınır değer, Shannon limiti olarak da adlandırılmaktadır. Kuantum kanalın da modellendiği kanal olan İSK kanal için bu değer,

$$H(A|B) = h(p) \text{ bit/simge} \tag{4.3}$$

olarak hesaplanır. Bununla birlikte, pratikte elde edilen değer genellikle optimal (minimum) olmaz; daha fazla ilave bilgi gönderilir, $I_{\text{pratik}} \geq I_{\text{minimum}}$.

BU protokollerinin KAD sistemlerindeki görevi, alıcıdaki bit dizisinin tüm hatalarını bulmak ve düzeltmektir. Bunu da sistemin işleyiş hızını mümkünse hiç düşürmeden ve bit dizisi hakkında da mümkün olduğu kadar az ilave bilgi göndererek (sızdırarak) yapmaktır. Yukarıda verilen alt sınırdan daha aşağılara inilemeyeceğine göre en az bu minimumda ya da bu minimuma mümkün olduğu kadar yakın bilgi sızdırılması hedeflenmelidir.

Günümüzde KAD'da BU ihtiyacını karşılamak için en çok tercih edilen üç aday CASCADE protokolü, LDPC kodlar ve kutupsal kodlardır. Her üç yöntemin de kendine göre avantajları ve dezavantajları vardır. Haklarında daha detaylı bilgi almak için [100]-[102]'ye başvurulabilir.

Bundan sonraki bölümlerde, biz CASCADE protokolünden söz edeceğiz.

4.3. CASCADE Protokolü

CASCADE protokolü, Gilles Brassard ve Louis Salvail tarafından [6]'da sunulmuştur. Hem hata sezme hem de hata düzeltme için eşlik kontrollerinden yararlanan interaktif bir hata sezme ve düzeltme yöntemidir. Esasen,

- Birkaç çevrimden,
- Çevrimler arası karıştırma işlemlerinden ve
- Her bir çevrimde
  - Blok eşliği karşılaştırma türü hata sezme (blok eşlikleri daima Alice'ten Bob'a doğru gönderilir) ile
  - İkili-arama türü hata düzeltme (hatalı bloklar iki parçaya bölünür ve ilk parçaların eşlikleri daima Alice'ten Bob'a doğru gönderilir, bit düzeltmelerini yapan taraf da daima Bob'dur)

işlemlerinden oluşur. Protokol sunulurken çevrim sayısının belirlenmesine yönelik teorik bazı ispatlar verilirken diğer parametrelerinin nasıl belirlendiğine ve hata başarımına yönelik net bilgi verilmemiştir. Protokol, özetle, şu şekilde çalışır.

4.3.1. Strateji

- Başlangıç

  Alice ve Bob, N uzunluklu bit dizilerine sahiptir.

- Orijinal dizi Alice'tekidir: A rasgele değişkeni ile gösterilir.
- Bob'un bit dizisi Alice'teki dizinin en çok $p = p_{\text{eşik}}$ kadar bozulmuş halidir; yani, $B = A \oplus e$ ilişkili rasgele değişkeni ile gösterilir.

- Çevrim $i = 1$

  Alice ve Bob, N uzunluklu bit dizilerine sahiptir.

  - Karıştırma: Öncelikle, Bob'daki hatalı bitlerin konumlarını rasgele hale getirmek (B dizisi içinde düzgün dağılmış olmalarını sağlamak) için Alice ve Bob haberleşerek rasgele bir karıştırıcı fonksiyon üzerinde anlaşırlar ve bit dizilerini bu aynı karıştırma fonksiyonundan geçirirler (Böylece, kanalın bir İSK kanal olduğu da garanti edilmiş olur).
  - Hata Sezme: Daha sonra, Alice ve Bob bit dizilerini k_1 uzunluklu bloklara böler ve her bir bloğun eşliğini hesaplarlar. Alice, bloklarına ilişkin tüm eşlik değerlerini Bob'a gönderir. Dolayısıyla, bu adımda

$$\left\lceil \frac{N}{k_1} \right\rceil \tag{4.4}$$

  tane eşlik transferi yapılmış olur. Bob, gelen eşlikleri kendininkiler ile tek tek karşılaştırır. Karıştırma fonksiyonunun bir sonucu olarak Bob'un her bir bloğunda en fazla 1-bit hata olması beklenir. Dolayısıyla, bu 1-bitlik hatalar da eşlik kontrol yöntemi ile mutlaka sezileceklerdir.
  - İnteraktif Hata Düzeltme: Daha sonra, eşlik kontrolleri sağlanmayan tüm bu k_1 bit uzunluklu bloklardaki 1-bitlik bit hatalarının konumları interaktif BINARY metodu ile bulunur ve düzeltilir. Düzeltme daima Bob'un tarafında yapılır. Çevrim bittiğinde, Bob'un 1-bit hata içeren tüm bloklarındaki tüm 1-bitlik hatalar düzeltilmiş olur. Artık, Bob'un çevrim 1'deki bloklarında kesinlikle çift sayıda (0, 2, 4, ...) hata kalmış demektir. Ve tüm eşlik kontrolleri de sağlanır durumdadır. Sonraki çevrimlerde kullanılmak üzere tüm bloklar her iki tarafta hafızada da saklanırlar.

- Çevrim $i > 1$

  - Karıştırma: Öncelikle, Bob'daki kalan hatalı bitlerin konumlarını tekrar rasgele hale getirmek için Alice ve Bob haberleşerek yeni bir rasgele karıştırıcı fonksiyon üzerinde anlaşır ve ikisi de kendi bit dizilerini bu aynı karıştırma fonksiyonundan geçirirler.

• Hata Sezme: Daha sonra, Alice ve Bob bit dizilerini $k_i = 2 \bullet k_{i-1}$ uzunluklu bloklara böler ve her bir bloğun eşliklerini hesaplarlar. Alice, bloklarına ilişkin tüm eşlik değerlerini Bob'a gönderir. Dolayısıyla, bu adımda, çevrim başında, Alice'ten Bob'a toplamda en fazla

$$\left\lceil \frac{N}{k_i} \right\rceil \tag{4.5}$$

tane eşlik transferi yapılmış olur. Bob gelen eşlikleri kendininkiler ile tek tek karşılaştırır. Karıştırma fonksiyonunun bir sonucu olarak Bob'un her bir bloğunda en fazla 1-bit hata olması beklenir. Bu 1-bitlik hatalar ise eşlik kontrol yöntemi ile mutlaka sezilirler.

• İnteraktif Hata Düzeltme: Daha sonra, eşlik kontrolleri uyuşmayan tüm bloklardaki 1-bitlik hataların konumları interaktif BINARY metodu ile bulunur ve düzeltilir. Düzeltmeler daima Bob'un tarafında yapılır.

Çevrim $i > 1$'de 1-bit hata bulunması demek mutlaka ona karşı gelen en az 1-bitlik bir hata daha vardır anlamındadır. Çünkü, önceki çevrimlerde ve bloklarda 1-bitlik tüm hatalar tespit edilmiş ve düzeltilmişlerdi; geriye ise sadece çift sayıda (0, 2, 4, ...) bit hataları kalmıştı.

Bu nedenle, *i*. çevrimde 1-bitlik bir hata düzeltilmişse geriye doğru bakılarak düzeltilen bu bitin daha önce hangi bloklarda bulunduğu da belirlenir ("iz sürme" prosedürü). Sonra, bu blokların en küçüğü üzerinde interaktif BINARY metodu koşturularak karşı gelen 1-bitlik hata da düzeltilir. Böylece, tüm eşlik kontrollerinin tekrar uyuşması sağlanır.

Aynı şekilde, düzeltilen bu yeni 1-bit için de bulunduğu tüm (eski ve yeni) bloklar yeniden taranır ve karşı gelen 1-bitlik hata da bulunmaya ve düzeltilmeye çalışılır. Bu özyinelemeli süreç yeni bir hata bulunamayana (dolayısıyla, tüm eşlik kontrolleri sağlanana) dek sürer.

i. çevrim bittiğinde, Bob'un 1-bit hata içeren tüm bloklarındaki tüm böylesi 1-bitlik hatalar bulunmuş ve düzeltilmiş olur. Artık, Bob'un çevrim *i*'deki bloklarında kesinlikle çift sayıda (0, 2, 4, ...) hata vardır demektir. Sonraki çevrimlerde tekrar kullanabilmek üzere, Alice ve Bob tüm bloklarını hafızada saklamaya da devam ederler.

• Sonuç

$B = A$.

4.3.2. Parametreler

[6]'daki orijinal CASCADE protokolünde çevrim sayısı 4 olarak belirlenmiştir. Çevrim 1'in blok uzunluğu k_1 ise, kuantum kanalın hata olasılığına bağlı olarak (teorik ispatları verilmeden) Tablo 4.2'deki gibi seçilmiştir.

Tablo 4.2: k_1 seçimi.

p	k_1
0.01	73
0.05	14
0.10	7
0.15	5

4.3.3. Karıştırma

Amaç, bitleri rasgele karıştırarak hatalı bitlerin düzgün dağılmasını (bir yerde toplanmış olmamalarını) sağlamaktır. Bu amaçla, her i çevrimi için,

$$h_{a,b}^{i}(x) = ((ax + b) \bmod m) \bmod N, \tag{4.6}$$

evrensel özet fonksiyonları ailesine ait özet fonksiyonlarının birinden, ya da başka çok iyi karıştırıcılardan da, yararlanılabilir. Burada, $m \geq \left\lceil \frac{N}{k_i} \right\rceil$, $0 < a < m$, $0 \leq b < m$, OBEB(a, m) = 1, OBEB(b, m) = 1 ilişkileri geçerlidir.

Evrensel özet fonksiyonları rasgeleleştirme (düzensizleştirme, düzeni-periyodikliği-tekrarı ortadan kaldırma), koşulsuz güvenlikte kimlik doğrulama, karıştırma vb. işlemlerde kullanılır. Evrensel özet fonksiyonları ve uygulamaları hakkında daha fazla bilgi için [103], [104]'e başvurulabilir.

4.3.4. BINARY

İkili-arama ya da böl-ve-hallet türü denebilecek interaktif bir hata düzeltme yöntemidir. k bitlik bir bozuk blok için, sadece 1-bitlik bir hatayı göndericiden alıcıya en fazla

$$\lceil log_2 k \rceil \tag{4.7}$$

tane eşlik bilgisi transferi ile tam olarak bulabilir ve düzeltebilir. Algoritması, özetle, şu şekildedir: Alice ve Bob'un eşlik kontrol yöntemi ile tek sayıda hatayı sezdikleri her bir hatalı blok için;

- Adım 1: Alice bloğunun ilk yarısının eşliğini Bob'a gönderir. Eşlik transferi daima Alice'ten Bob'adır. Tüm blok eşliği de bilindiğinden Bob ikinci yarının eşliğini kolaylıkla hesaplar.
- Adım 2: Bob gelen ilk yarı eşliğini kendi bloğunun ilk yarısının eşliği ile karşılaştırarak 1-bit hatanın ilk yarıda mı yoksa ikinci yarıda mı olduğunu sezer ve sonucunu Alice'e de açıklar.
- Adım 3: Adım 2'de tespit edilen yarıya Adım 1'dan itibaren süreç özyinelemeli olarak tekrarlanır. Sonuçta, 1-bit hatanın konumu bulunur ve sadece Bob tarafında düzeltilir.

Görüldüğü gibi, her çevrim başında gerçekleştirilen karıştırıcı parametreleri ile blok eşliklerinin değiş-tokuşundan ve devamında koşturulan BINARY adımlarından dolayı CASCADE protokolü oldukça interaktiftir.

Hem CASCADE hem de BINARY protokollerinin ikisinin de böl ve hallet türü bir yaklaşım izlediklerine dikkat edin. Böl ve hallet taktiği, büyük bir problemi daha küçük parçalara bölerek ve her bir parçayı teker teker halletmek suretiyle problemi ortadan kaldırmaya yönelik bir yöntemdir. CASCADE protokolü hatalı bit dizisini parçalara bölerken, BINARY protokolü hatalı bloğu böler.

Kuantum kriptografi camiasında, yoğun haberleşme gerektirmesinden (dolayısıyla, ağ gecikmelerine karşı çok duyarlı-bağımlı-hassas olmasından) dolayı CASCADE gibi interaktif BU protokollerinin daha verimsiz olduğu kanısı yaygın

olarak kabul edilir. Bu nedenle, yerine LDPC kodlarına ya da daha yenice kutupsal kodlara dayanan interaktif olmayan BU protokolleri önerilmiştir.

İnteraktif olmayan yöntemlerin interaktif olanlardan en temel farkı fazlalık bilginin göndericiden alıcıya tek bir defada gönderilmesidir. Yani, ağ sadece bir defa kullanılmaktadır. Bununla birlikte, interaktif olmayan yöntemlerin kodlama ve kod çözme işlemleri hesapsal olarak daha karmaşıktır. İnteraktif olanlarda ise ağ birden çok kez kullanılırken hesapsal olarak yükleri genellikle daha azdır.

Bu çalışmanın esas amacı, [6], [9], [14] gibi interaktif protokollerin interaktif olmayanlardan daha kötü performansa sahip olduğu varsayımına meydan okumaktır. Sonraki bölümde, bu bağlamda, CASCADE protokolünün performansının daha da arttırılmasına yönelik fikirlerimizi sunacağız.

CASCADE protokolü, aşamaları ve analizleri hakkında daha detaylı bilgi için [105], [106]'ya da başvurulabilir.

5. PERFORMANS

Bu bölümde, CASCADE protokolünün performansını arttırmaya yönelik önerilerimiz ve analizleri yer almaktadır.

5.1. Hızlı CASCADE

CASCADE protokolünün, daha genel olarak hata sezme ve düzeltme kodlarının, performansını daha da arttırmaya yönelik olarak ilk yapılabilecek olan, protokolün uyarlanabilir (parametrelerinin değişik hata olasılıklarına göre ayarlanabilir) olmasıdır. Önceki bölümde görüldüğü gibi, orijinal CASCADE protokolünde geliştiricileri tarafından bu esneklik sağlanmıştır (değişik hata olasılıkları için değişik parametreler kullanılmaktadır).

BB84 KAD protokolünün belirli bir $p_{eşik}$ değerine (KAD protokolüne ve o protokole uygulanan güvenlik ispatına bağlı bir değerdir) kadar güvenli olarak çalışabileceğini ve güvenli bir gizli anahtar üretilebileceğini belirtmiştik. Hata olasılığını her zaman $p_{eşik}$ gibi yüksek bir değer kabul edip BU protokolünü de hep bu ön kabule göre çalıştırmak daima en verimlisi olmayabilir. Eğer böyle yapılırsa, hata olasılığının çok küçük bir $p < p_{eşik}$ değerinde olduğu durumlarda entropi (örneğin, saldırganın gizli anahtar hakkındaki belirsizliği azalır), zaman, işlemci, enerji gibi çok önemli kaynakların boş yere harcanması durumu meydana gelecektir. Dolayısıyla, BU protokolünün sabit bir $p_{eşik}$ olasılığı yerine her bir oturumun kendine ait $p \leq p_{eşik}$ olasılığına göre çalıştırılması en efektif yol olacaktır.

Önceki bölümde bahsetmemiş olsak da KAD protokollerinde eleme fazı ile uzlaştırma fazı arasında bir de hata tahmini ara fazı bulunmaktadır (uzlaştırma fazının bir adımı olarak kabul edildiği de olur). Bu fazda o anki KAD oturumunun p hata olasılığı belirlenmeye çalışılır. Bu amaçla, elenmiş anahtar bitlerinin bir alt kümesi yine klasik kanal üzerinden açıklanır. Alice ve Bob, birbirlerine açıkladıkları bu bitleri karşılaştırarak o oturuma ilişkin p hata olasılığını belirler ve kullandıkları BU protokolünün parametrelerini de bu değere göre güncellerler. Açıklanan bitler ise atılır, protokolün sonraki aşamalarına dâhil edilmezler.

BU protokolünün uyarlanabilir olması, sonlu kaynakların dikkatli kullanımında önemli ölçüde ilerleme sağlayacaktır. Bu sağlandıktan sonraki aşamalarda ise

kullanılan hata sezme ve düzeltme algoritmasının içerdiği temel bileşenlerin hızlandırılmasına (fazlalıklar giderilerek daha verimli hale getirilebilmelerine) yönelik girişimlerde bulunulabilir. Örneğin, CASCADE protokolü,

- karıştırma
- eşlik hesabı
- haberleşme

temel işlemlerinden oluşmaktadır. Her bir işlemin ağ, işlemci, bellek vd. kaynakları en idareli kullanan iyileştirmeleri üzerinde çalışılabilir. Aşağıda, bu işlemlerin daha verimli hale getirilmesine (haberleşmedeki, hesaplamadaki vd.deki fazlalığın giderilmesine) yönelik önerilerimiz yer almaktadır.

- Haberleşme

CASCADE protokolü denince ilk akla gelen interaktif bir yöntem olduğudur; yani, çok fazla haberleşme (klasik ağın çok fazla kullanımını, meşgul edilmesini) gerektirir. Dolayısıyla, CASCADE'in verimli gerçeklemesi için ilk doğal adım bu yoğun etkileşimin azaltılması olacaktır.

Bu amaçla, Louis Salvail tarafından da işaret edildiği gibi [96], blokların ve alt-blokların eşlik kontrolleri (hata sezme) ile hatalı olanlarının BINARY işlemleri (hata düzeltme) paralel yapılabilir. Öncelikle, her raundun başında bit dizisinin tüm bloklarının eşlikleri tek bir mesajda değiş-tokuş edilir ve hatalı olanları sezilir. Daha sonra, hatalı her bir bloğu orijinal CASCADE protokolünde yapıldığı gibi sıralı (teker teker) düzeltmek yerine, tüm bloklar ikiye bölünür ve alt-bloklarının eşlikleri tek bir mesajda gönderilir. Hata sezilen alt-bloklar üzerinde özyinelemeli BINARY süreci de aynı şekilde (paralel olarak) devam eder: hatalı alt-bloklar ikiye bölünür ve sonraki ikiye bölmeden önce eşlikleri yine tek mesaj olarak gönderilir.

Bu durumda, k bitlik bir raunt/çevrim blok uzunluğu için, blokların sayısından ve bit dizisinin uzunluğundan bağımsız olarak, BINARY boyunca değiş-tokuş edilen mesajların (klasik kanaldan yapılan haberleşmelerin) toplam sayısı

$$\lceil log_2 k \rceil \tag{5.1}$$

olarak elde edilir. Yani, k_1 bit blok uzunluklu raunt 1 açısından bakacak olursak, raunt başında yapılan ilk haberleşmenin de eklenmesiyle toplamda, sadece

$$1 + \lceil log_2 k_1 \rceil \tag{5.2}$$

adet haberleşme ile (tek sayıda hata içeren bloklardaki 1 bitlik) tüm hatalar sezilmiş ve düzeltilmiş olur (çift sayıda hata içeren bloklar ise, kullanılan eşlik kontrolü tabanlı hata sezme ve düzeltme yöntemleri nedeniyle, sonraki rauntlara kalır). Bu fikirler, 2002 yılından beri geliştirilmekte olan AIT KAD yazılım projesindeki [107] CASCADE gerçeklemesinde de kullanılmıştır.

Raunt 2'de ve sonraki rauntlarda bir hata sezildiğinde, bu hatanın önceki rauntlardaki çift sayıda hata içeren bloklarda yer aldığı anlamına gelir. Orijinal CASCADE protokolü, bu gerçekten yararlanmak için her raunttan sonra bir iz sürme adımına sahiptir: Örneğin, raunt 1'de iki tane hata içeren bir blok olduğunu düşünelim (Alice ve Bob, bu bloğun eşliğinde uzlaşacağı için hata sezilemez ve raunt 1'de bu blok üzerinde herhangi bir işlem gerçekleştirilmez). Eğer bu bloktaki hatalardan biri raunt 2'de düzeltilirse (raunt 1'deki blokta artık 1 tane hata kalır ve eşlikleri uyuşmaz hale gelir), diğer hata da raunt 1'e geri gidip düzeltilen hatayı içeren ilgili blok üzerinde BINARY protokolü çalıştırılarak düzeltilebilir (böylece, ilgili blok için eşlikler tekrar uyuşur hale gelebilir). İz sürmede, bir rauntta düzeltilen her bir hata için, o hatayı içeren önceki rauntlardaki tüm bloklar bir iz sürme listesine eklenirler. Daha sonra, BINARY protokolü en küçük bloktan başlanarak bu listedeki her bir bloğa uygulanır. İz sürme boyunca düzeltilen her bir yeni hata için, tüm önceki rauntlardaki bu hatayı içeren bloklar da iz sürme listesine eklenirler. İz sürme listesi boşalana kadar bu süreç devam eder.

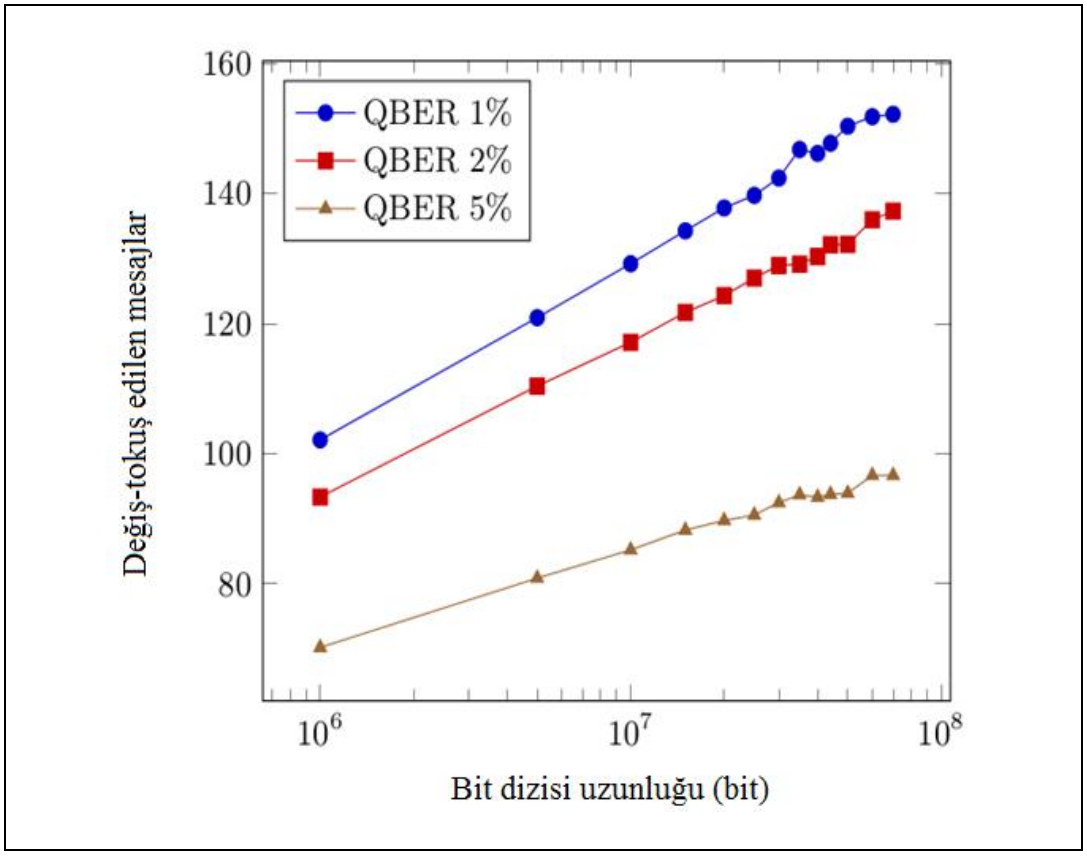
İz sürme prosedürünün orijinal formülasyonunda, bir anda sadece tek bir blok düzeltilebilir. Bu nedenle, bu sıralı çalışma da çok yüksek düzeyde bir etkileşime yol açmaktadır. Bu problemin üzerinden gelmek için, iz sürme adımına da küçük bir değişiklik öneriyoruz: BINARY protokolü, önceki rauntlardan hata içeren blokların en küçüklerine teker teker uygulanmak yerine, en eski önceki raundun hata içeren tüm bloklarına birden uygulanır. Düzeltmeler aynı raundun örtüşmeyen bloklarına yapıldığı için, BINARY protokolü yukarıda da yapıldığı gibi yine paralel olarak uygulanabilir. Böyle yapılması halinde, raunt 2'de, bu yeni iz sürme ile ziyaret edilen

bloklar ve orijinal protokoldeki eski iz sürmede ziyaret edilenler tamamen aynı bloklar olacaklardır.

Bizim yaklaşımlarımızla, Alice ve Bob arasından gönderilmesi gereken mesajların sayısı

$$r + \sum_{i=1}^{r} \lceil log_2(k_i) \rceil + l(QBER, N, k_1, \dots, k_r) \tag{5.3}$$

olup burada r parametresi rauntların sayısı, k_i parametresi i. rauntta kullanılan blok uzunluğu ve l ise iz sürme boyunca değiş-tokuş edilen tüm mesajları tanımlayan bir fonksiyondur.



Şekil 5.1: Bit dizilerini düzeltmek için değiş-tokuş edilen mesaj sayısı. Bit dizisinin uzunluğuna olan bağlılık sadece iz sürme prosedüründen gelmektedir. Bit dizisinin uzunluğuna olan logaritmik bağlılık ise büyük bit dizilerinin kullanımını teşvik etmektedir.

Bilgisayar ortamında (birbirine gigabit Ethernet bağlantısı ile bağlı iki Intel Core i7 3.4 Ghz CPU üzerinde) yapılan benzetimlere göre, bu değişikliklerden sonra

CASCADE protokolünün orijinal blok uzunlukları için Alice ve Bob arasında değiş-tokuş edilen mesajların sayısı bit dizisinin uzunluğunun bir fonksiyonu olarak Şekil 5.1'deki [96] gibidir. Buna göre,

- mesajların sayısı bit dizisinin uzunluğuna göre logaritmik olarak artar. Dolayısıyla, l değeri bit dizisinin uzunluğu N'ye göre sadece logaritmik olarak artmaktadır.
- QBER arttıkça değiş-tokuş edilen mesajların sayısı azalmaktadır. Bu, daha büyük QBER değerleri için daha küçük blok uzunluklarının (k_i) kullanılması dolayısıyladır.
- Bit dizisinin uzunluğu arttırıldığında haberleşme yükünde de bir artış olmakla birlikte, tam lineer olmayış nedeniyle bu artışta gittikçe bir düşüş olduğu da görülmektedir. Dolayısıyla, bu gerçeklemeden en büyük olası avantajı elde etmek için, bit dizisi mümkün olduğu kadar uzun olmalıdır

Bit dizisinin uzunluğu arttırıldıkça belirli bir değerden sonra, haberleşmede harcanan zamanın azalmasına rağmen, yine de hızın düşmeye bağladığı görülür. Bu düşüş artan hesaplama süresinden kaynaklanmaktadır (hesaplama süresi bit dizisinin uzunluğu ile süper lineerdir). Ağırlık o oranda protokolün haberleşme yükünden hesaplama yüküne geçmekte, zamanın çoğu hesaplamada (örneğin, eşliklerin hesaplanmasında vd.) harcanmaktadır.

Analizler, optimum hızlara milyon bitler mertebesindeki bit dizisi uzunlukları için ulaşıldığını göstermektedir. Maksimum olası bit dizisi uzunluğunu sınırlayan temel faktör ise bellektir. Gerekli bellek, bit dizisinin uzunluğu ile lineer (ve QBER ile de alt-lineer [96]) olarak artar. Bu nedenle, mümkün olduğu kadar az bellek kullanmak önemlidir. Belleği verimli kullanmak için ilk basit adım, belleğin her bir baytının 8 tane veri biti depoladığından emin olmaktır.

- Karıştırma

Bellek ile ilgili benzer bir sıkıntılı durum, iz sürme adımından kaynaklanır. CASCADE'de, her raundun başında bit dizisi karıştırılmaktadır (permütasyon). İz sürmede, şu anki rauntta düzeltilen bitin bir önceki raundun hangi bloğunda olduğunun bulunması gerekir. Bunun için de öncelikle şu anki raundun

permütasyonunun tersinin alınması (düzeltilen bitin indeksinin bulunması) ve daha sonra da (bu indekse) önceki raundun permütasyonunun uygulanması gerekir. CASCADE'in orijinal tanımında, permütasyon evrensel özet fonksiyonlarının ailesinden bir $h()$ özet fonksiyonu ile yapılmaktadır. Böylece, i indeksindeki bit $h(i)$ blok numarasına atanır. Bununla birlikte, $h()$ fonksiyonlarının bir özelliği olarak, bir bitin $h(i)$ bloğunda olduğunu bilmek bize bitin orijinal i indeksini hesaplama olanağı vermez. Basit bir çözüm, orijinal indekslerden karıştırılmış indekslere olan karşı düşürmeyi kaydetmektir. Bununla birlikte, bu karşı düşürmenin depolanması en az

$$N\log_2 N \tag{5.4}$$

bit gerektirir. Bizim amacımız olan büyük bit dizisi uzunlukları için ise depolamada harcanan bellek miktarları hayli yüksek olacaktır: Örneğin, 10 Mbit'lik bir bit dizisinin permütasyonunu depolamak için en az 232 Mbit'e ihtiyaç olur. Bit dizisi uzunluğunu 5 katına çıkarırsak bellek de 5 katına çıkacaktır.

Yukarıdaki yaklaşım yerine biz gizli anahtar bitlerini rasgele, tersi alınabilir bir fonksiyonla karıştırıyoruz. CASCADE'de permütasyon iki amaca hizmet etmektedir: Öncelikle, güvenlik için bir ilk permütasyon gereklidir. Bunun sayesinde, saldırganın yaptıklarını bir İSK olarak modelleyebilmekteyiz. Bu ilk permütasyonun tersinin alınmasına gerek yoktur. Bu nedenle, güvenlik için, burada rasgele bir permütasyon uygulamaktayız. İkinci olarak, rauntlar arasında karıştırıcı olarak kullanılır. Böylece, bir rauntta sezilemeyen hataların (çift sayıdaki hatalar) sonraki rauntta da aynı blokta yer almalarının önüne geçilir (aksi halde, yine sezilemezler). Dolayısıyla, rauntlar arasında güvenlik değil de daha çok hata sezme amaçlı karıştırıcı bir rol oynamaktadır. Tersi alınabilir bir permütasyon fonksiyonu elde etmek için tanımı aşağıdaki gibi olan 2-evrensel özet fonksiyonları ailesinden yararlanmaktayız:

$$\{\, h_{a,b}(i) = (ai+b \bmod p) \bmod N \,\}_{a,b} \tag{5.5}$$

Burada, $p \leq N$ olan sabit bir asal sayı iken $0 < a < p$ ve $b < p$ rasgele seçilen pozitif tamsayılardır. Bu fonksiyonların ana fikri, i indeksindeki bir biti $h_{a,b}(i)$ bloğuna karşı düşürmektir. Bununla birlikte, bu özet fonksiyonu bire bir değildir; yani, bir

permütasyon değildir. Aşağıdaki hesaplamayı yaparak bu özet fonksiyonunu bir permütasyona dönüştürüyoruz:

$$h_{a,b}(i) = ai+b \bmod p \tag{5.6}$$

Kullanım sırasında, eğer $j = h_{a,b}(i) \geq N$ durumu ile karşılaşılırsa, $h_{a,b}(j)$ hesaplanır ve bir $j' < N$ değeri elde edilene kadar tekrarlanır. Bu fonksiyon bire birdir ve tersi de kolayca alınabilir. $j' < N$ değerine mümkün olan en az iterasyonla ulaşılabilmesi için p asal sayısı olarak N değerinden büyük en küçük asal sayı seçilmektedir.

• Eşlik Hesabı

Büyük bit dizisi uzunlukları kullanıldığında, ve hesaplama darboğaz olduğunda, CASCADE'in hızı hesaplama hızlandırılarak yükseltilir. CASCADE'de çok fazla bir hesaplama da yoktur aslında: çoğunlukla, eşliklerin hesaplanması (eşlik hesabından sonra hesaplamaya en büyük katkı tüm zamanın 10-15% kadarını alan permütasyondan gelmektedir). BINARY boyunca, hatalı blok ikiye bölündükten sonra, alt-bloklardan birinin eşliğinin hesaplanması gerekmektedir. Bu, hatalı bloğun eşliğini hesaplamak için zaten kullanılmış olan bitlerin yarısını tekrar ziyaret etmeyi gerektirir. Böylesi tekrarların önüne geçmek için, her raundun başında, raunt başındaki blokların eşliklerinin hesaplanmasından oluşan ön eşlik listeleri oluşturuyoruz. Karıştırılmış bit dizisi olan $[d_0, ..., d_{N-1}]$ dizisinin ön eşlik listesinin i. elemanı aşağıdaki gibi hesaplanmaktadır:

$$pp_i = pp_{i-1} \oplus d_{i-1}, \quad i \in \{1, ..., N\} \tag{5.7}$$

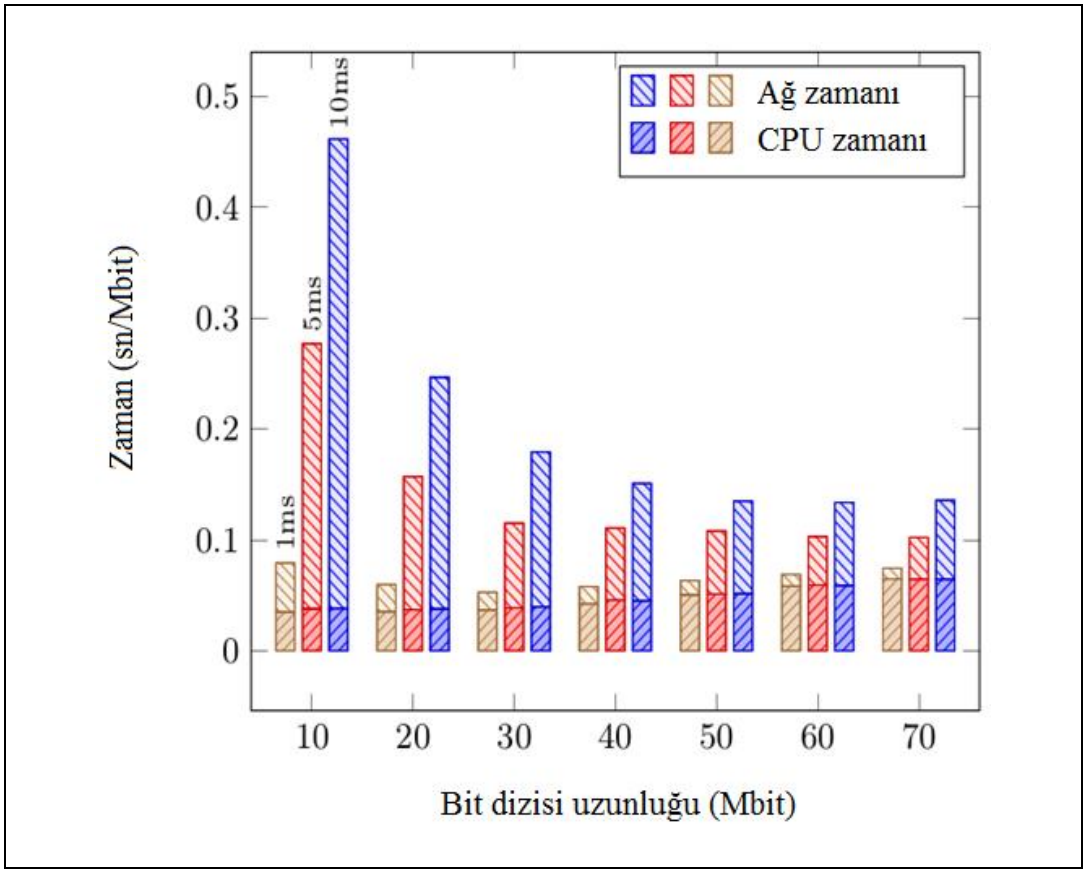
Bu hesaplamada, $pp_0 = 0$ ile başlanmaktadır. Ön eşlik listesinin hesaplanması blokların eşliklerinin hesaplanması ile aynı zamanı almaktadır; ancak, ön eşlik listesi bir kez oluşturulduğunda herhangi bir $[d_i, ..., d_{i+l}]$ veri bitleri aralığının eşliği, ön eşlik listesindeki sadece iki değere bakılarak,

$$parity([d_i, ..., d_{i+l}]) = pp_i \oplus pp_{i+l+1} \tag{5.8}$$

şeklinde hesaplanabilir. Böylece, yukarıdaki gibi $l + 1$ bit uzunluklu bir bit dizisi için eşlik hesabı toplam l adet XOR işlemi yerine sadece 1 adet XOR hesabı ile çok daha hızlıca (verimlice) yerine getirilebilmektedir.

İz sürme adımının orijinal tanımında, yenice düzeltilmiş bir bit için önceki rauntların sadece en küçük blokları araştırılır. [12]'de, yazarlar iz sürmede CASCADE boyunca görülen tüm blokların kullanıldığı bir iyileştirme önerirler. Bu, yenice düzeltilmiş biti içeren en küçük bloğun beklenen uzunluğunu düşürür; böylece, takip eden BINARY'yi daha az interaktif yapar. Bizim CASCADE gerçeklememizde bu iyileştirme de uygulanmıştır.

Şekil 5.2 [96], bir bit dizisi üzerinde harcanan toplam zamanı haberleşmeye (ağ) ve hesaplamaya (CPU) harcanan kısımlarıyla ayrı ayrı göstermektedir. Toplam zamanın üst kısmı haberleşmeye ait olan zamanı, alt kısmı ise hesaplamaya ait olan zamanı ifade etmektedir.



Şekil 5.2: Bit dizilerini düzeltmek üzere harcanan zaman. Her bir bit dizisi uzunluğu için, 1 ms (ilk sütun), 5 ms (ikinci sütun) ve 10 ms (üçüncü sütun) gecikme için zamanları listelenmektedir. Bit dizisinin uzunluğu arttıkça, haberleşmenin katkısı azalır. 1 ms gecikme için, haberleşmenin tüm harcanan zamana önemsiz bir katkısı olduğunu görüyoruz.

Şekilde verilen sonuçlar incelendiğinde, haberleşmeye harcanan zamanın üstel bir düşüşe sahip olduğu ve 40-50 milyon bitlik bir bit dizisi uzunluğundan sonra genellikle hemen hemen sabit hale geldiği görülmektedir. Diğer taraftan, hesaplama zamanı ise toplamda haberleşme zamanından daha az bir zaman almakla birlikte bit dizisinin uzunluğunun artmasıyla gittikçe bir artış göstermektedir.

Benzetimler, elde edilen 83.49 Mbps'lik optimum hıza 10 Mbit'lik bir bit dizisi ile ulaşıldığını göstermektedir (1 ms gecikme, 1% QBER ve 8 görevcikli çalışma için) [96] (şu an için literatürde sunulan en yüksek değerdir).

CASCADE protokolünün hızlı gerçeklemesi, benzetimleri ve analizleri hakkında daha detaylı bilgi için [96]'ya başvurulabilir.

5.2. Verimli CASCADE

Önceki bölümde, BU protokollerinin performansı ile ilgili önemli ölçütlerden olan hız konusunu ele almış ve interaktif BU protokollerinden CASCADE protokolü örneğinde hızı arttırmaya yönelik önerilerimizi paylaşmıştık. BU protokollerinin performansı ile ilgili çok önemli ölçütlerden bir diğeri ise verimliliktir. Bu bölümde de interaktif BU protokolleri için, yine CASCADE protokolü örneğinde, verimliliği arttırmaya yönelik önerilerimizi sunacağız.

Bir bilgilendirme notu olarak, bu bölümde anlatılan verimlilik kavramı ile önceki bölümde anlatılan, algoritmaların sınırlı sistem kaynaklarını (kanal, işlemci, bellek, zaman vd.) daha temkinli kullanmalarına yönelik gerçeklenmelerindeki verimlilik kavramının karıştırılmamasına dikkat çekmekte yarar vardır. Bu bölümde anlatılan protokol verimliliği, esas konumuz olan güvenlikle alakalı bir parametre olup saldırgana sızan bilginin oranı ile ilgili bir kavramdır. Saldırgana ne kadar az bilgi sızarsa verimlilik de o kadar yüksek olmaktadır.

Bütün klasik kanallar prensipte çok kolayca dinlenebileceğinden, klasik kanal üzerinden taşınan (ve çok gizli tutulması gereken gizli anahtar hakkındaki) fazlalık bilgiyi Eve de (pasifçe) ele geçirir. Bu nedenle, KAD'da, bu ilave bilgi, onu Eve de bildiği için, "sızan bilgi" olarak da adlandırılmaktadır. Ve daha fazla sızdırılan ilave bilgi yetkisiz Eve için anahtar hakkında o kadar az belirsizlik demektir. Fazlalık bilgi anahtarların gizliliğini düşürdüğü için, KAD sistemlerinde mümkün olduğu kadar az fazlalık bilgi sızdıran BU yöntemlerine ihtiyaç vardır.

Sızdırılan (klasik kanaldan gönderilen fazlalık) bilgi miktarı, BU protokolleri için verimlilik performansının bir ölçütüdür. Bilgi Teorisi'ne göre, tüm hataları sezmek ve düzeltmek için açıklanması gereken minimum fazlalık bilgi miktarı Shannon limiti kadardır. CASCADE protokolü gerekli bütün bu koşulları sağlayan, yani, Shannon'un limitine çok yakın miktarlarda fazlalık bilgi açıklayarak tüm hataları hızlıca sezen ve düzelten, bir BU tekniğidir.

CASCADE protokolünün verimlilik performansını arttırmaya yönelik olarak literatürdeki önemli iki versiyon [12]'de ve [13]'te önerilenlerdir. [13]'te yeni bir parametre kümesi önerilirken, [12]'de hem protokolün stratejisi değiştirilmiş hem de yeni bir parametre kümesi önerilmiştir.

Bu çalışmada, [12]'de ve [13]'te de verilen, ve aşağıda da özetlenen, verimlilik tanımı takip edilmektedir.

Kuantum kanalın modellendiği İSK(p) kanalı için, verimliliğin teorik limiti (maksimum, üst sınır) aşağıda verilen

$$\eta_{Th} = H(A) - H(A|B) = I(A; B) \text{ bit/simge} \tag{5.9}$$

değerine eşittir. Burada, bilgi teorisi, eşit olasılıklı 0, 1 simgeleri üreten bir kaynak olması nedeniyle A'daki enformasyonun ortalama değerini bize

$$H(A) = 1 \text{ bit/simge} \tag{5.10}$$

olarak verirken ve kuantum kanalın bir İSK(p) olarak modellenmesi nedeniyle sızan enformasyonun ortalama değerini de (minimum, alt sınır, Shannon limiti)

$$H(A|B) = h(p) \text{ bit/simge} \tag{5.11}$$

olarak hesaplamamızı sağlar. Burada da, yine bilgi teorisine göre,

$$h(p) = -p\log_2 p - (1-p)\log_2(1-p) \tag{5.12}$$

olan ikili entropi fonksiyonudur.

Pratikte, sızan bilgi yukarıda verilen teorik alt sınırdan daha fazlaca olur. Yani,

$$leak \geq H(A|B) \text{ bit/simge} \tag{5.13}$$

olur. Ve bu durumda, bir BU protokolünün pratik verimlilik performansı da sızma hızı *leak* değerine bağlı olarak şu formüle göre hesaplanır.

$$\eta_{Pr} = H(A) - leak \tag{5.14}$$

$H(A)$, birimi bit/simge olan ortalama bir değerdir. Ve, yine bir ortalama değer olması gereken *leak* de aşağıdaki gibi hesaplanır.

$$leak = \frac{|E|}{N} \text{ bit/simge} \tag{5.15}$$

leak, simge başına eşlik biti miktarını ifade eden bir değerdir. Burada, E, Eve'in uzlaştırma protokolü boyunca elde edebileceği sızan bit dizisidir. Yani, $|E|$, $B = A$ yapmak için klasik kanal üzerinden değiş-tokuş edilen eşlik bitleri sayılarak hesaplanır: her bir raundun başında değiş-tokuş edilen blok eşlikleri, BINARY eşlikleri ve iz sürme eşliklerinin toplamından oluşur.

Alice ve Bob tarafından değiş-tokuş edilen her bir eşlik Eve'e bir bit bilgi sızdırır (o eşliğin önceki değiş-tokuş edilmiş eşliklerden hesaplanabildiği durumlar hariç). Böylece de BU'nun verimlilik performansını düşürür.

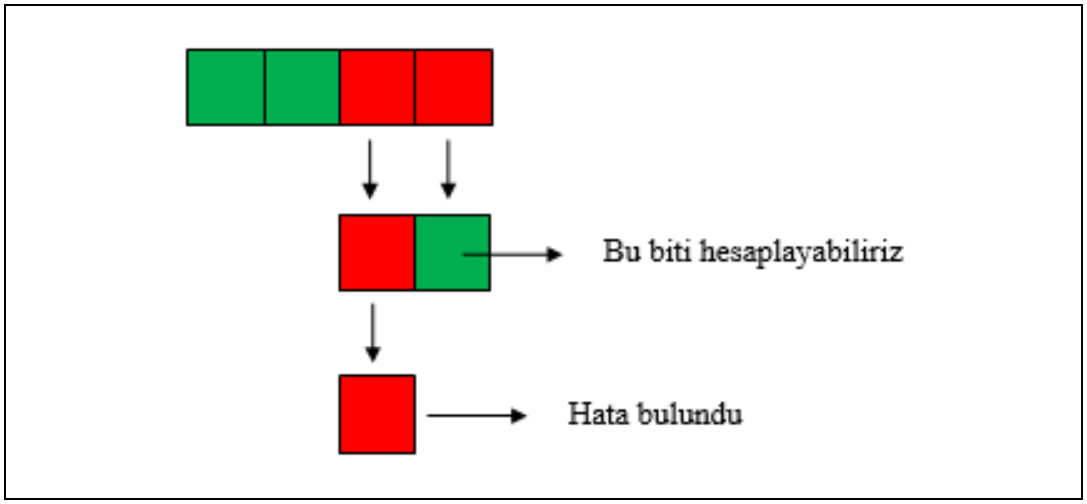
Eşitlik (5.14)'teki sızma hızını minimize etmek, ve verimliliği arttırmak, için, [12]'de de yapıldığı gibi hataları mümkün olduğu kadar küçük bloklarda aramayı hedeflemeliyiz. Bununla birlikte, o çalışmada verilene ek olarak bunu başarmak için halen başka yollar mevcuttur. Protokol içinde kullanmamız için hâlihazırda bazı başka içsel bilgiler bulunmaktadır. Böylece, biz de hataları aramak için daha küçük bloklar kullanarak değiş-tokuş edilen eşlik bitlerinin sayısını azaltmak yoluyla CASCADE protokolünün verimliliğini daha fazla iyileştirmekteyiz. BINARY sürecini daha büyükleri yerine bu küçük bloklar üzerinde koşturmakla, değiş-tokuş edilen bilgi daha az ve CASCADE protokolü de daha verimli olacaktır.

Üzerlerinde BINARY protokolünü çalıştırmadan hemen önce, hatalı blokları daha küçük yapmak (dolayısıyla, verimliliği arttırmak) üzere, bu çalışma için bizim önerdiğimiz başlıca yollar, sırasıyla, aşağıda listelenmiştir.

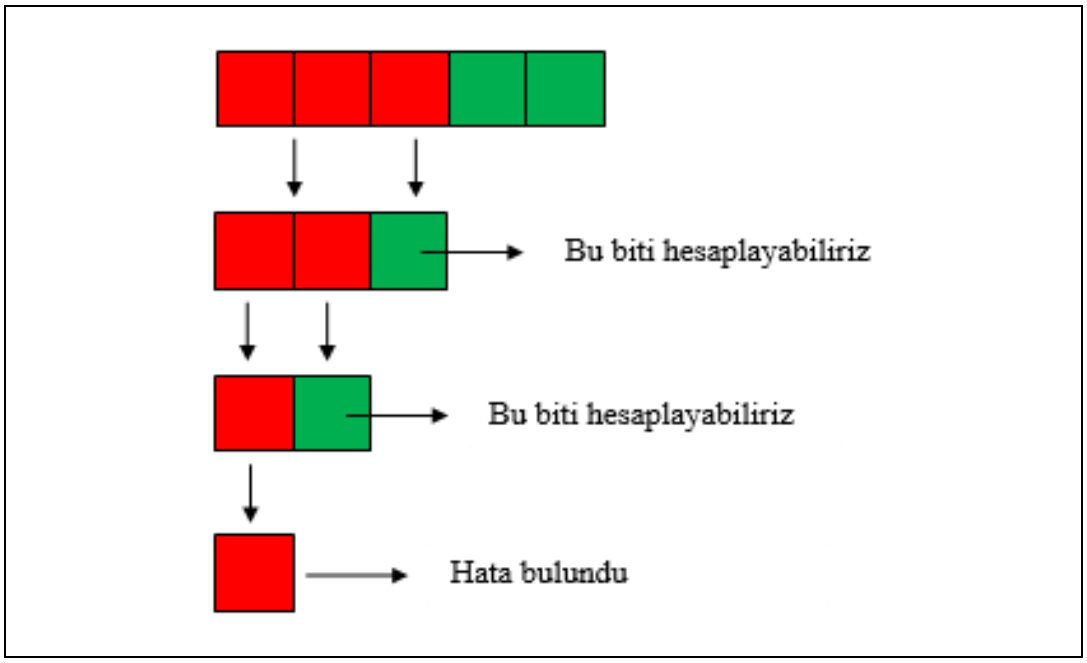
• Gözlem 1

Protokol boyunca birçok hatalı bit düzeltilir. Bununla birlikte, bir bit düzeltildiğinde, artık hatalı bit olamayacağı için, o biti gelecek BINARY uygulamalarında dâhil etmeye gerek yoktur. Düzeltilen bitlerin çıkarılması hata aradığımız bloğun uzunluğunu düşürecektir ve böylece hatayı bulmak için değiş-tokuş ettiğimiz eşliklerin sayısı azalacaktır.

Şimdi, bir hatanın nasıl bulunduğunu düşünelim: Alice ve Bob hatayı 2 bit uzunluklu bir blok içinde arıyor olsunlar. Şekil 5.3'te [109] gösterildiği gibi, 2 uzunluklu bloğun eşliğinin ve düzeltilmiş bitinin değerinin bilinmesi, bize diğer sağlam (sağlıklı) bitin değerini de söyler. Gelecekte hata ararken, hata aramalara o biti (değerini bildiğimiz ve hatalı bit de olamayacağı için) dâhil etmeye de herhangi bir gerek yoktur. Çünkü, artık hatalı bit olma olasılığı kalmamıştır. [108]'de de yapıldığı gibi, değerlerini tam olarak bildiğimiz ve doğru olduklarından emin olduğumuz için, bu iki biti "tam olarak bilinen bitler" olarak adlandırmaktayız. Bu bitleri bilmek için gereken bilginin klasik kanal üzerinden değiş-tokuş edilmesi nedeniyle, değerlerini Eve'in de bildiğine dikkat edin.

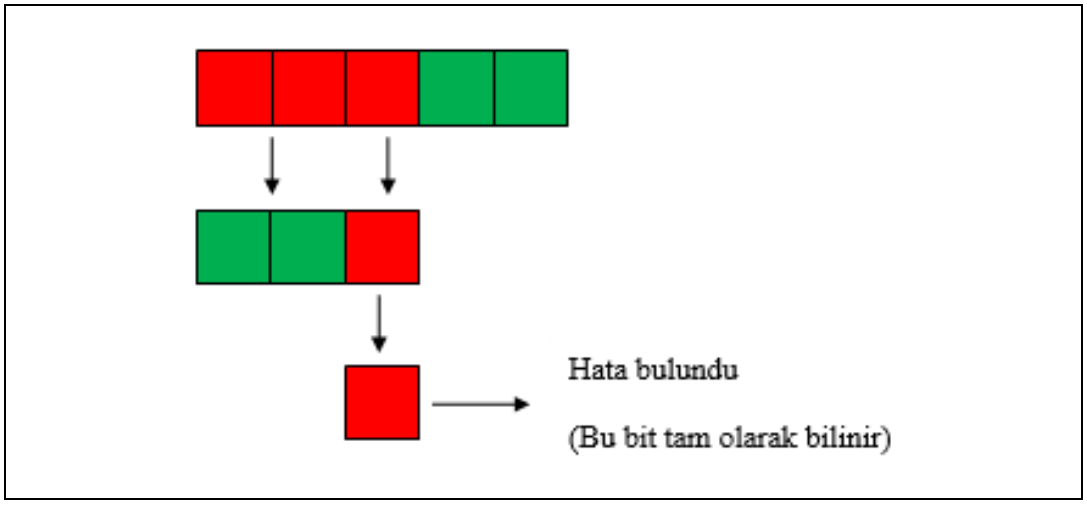


Şekil 5.3: İki uzunluklu blok durumu. Bir bit düzeltildiğinde, bir başka (doğru) bitin değerini tam olarak öğrenebilmekteyiz. İki uzunluklu bloklar için bu her zaman geçerlidir. Bu bilinen bitler daha sonra, interaktif BINARY süreci ile başka hatalar aranmaya başlamadan önce, bloklardan çıkarılacaklardır.



Şekil 5.4: Üç uzunluklu blok durumu. Gerçeklemelerimizde daha büyük yarıyı her zaman sol dal olarak aldığımız için, eğer dallanma soldan devam ederse, üç uzunluklu bloklardaki üç bitin tamamının değerini tam olarak bilebileceğiz.

Fikri, 3 bit uzunluklu bloklarla devam ettirmek de mümkündür. Şekil 5.4'te [109] görüldüğü gibi, üç bit uzunluklu bir bloktaki üç bitin tamamının değerini elde edebiliriz. Eğer hata üç bit uzunluklu bir bloğun sol dalında yer alıyorsa, üç bitin tamamı tam olarak bilinendir (hem bize hem de Eve'e). Bununla birlikte, eğer hata sağ dalda ise, bizim gerçeklemelerimizde sadece bir bit tam olarak bilinen olacaktır. Bu durum Şekil 5.5'te [110] de gösterilmiştir.

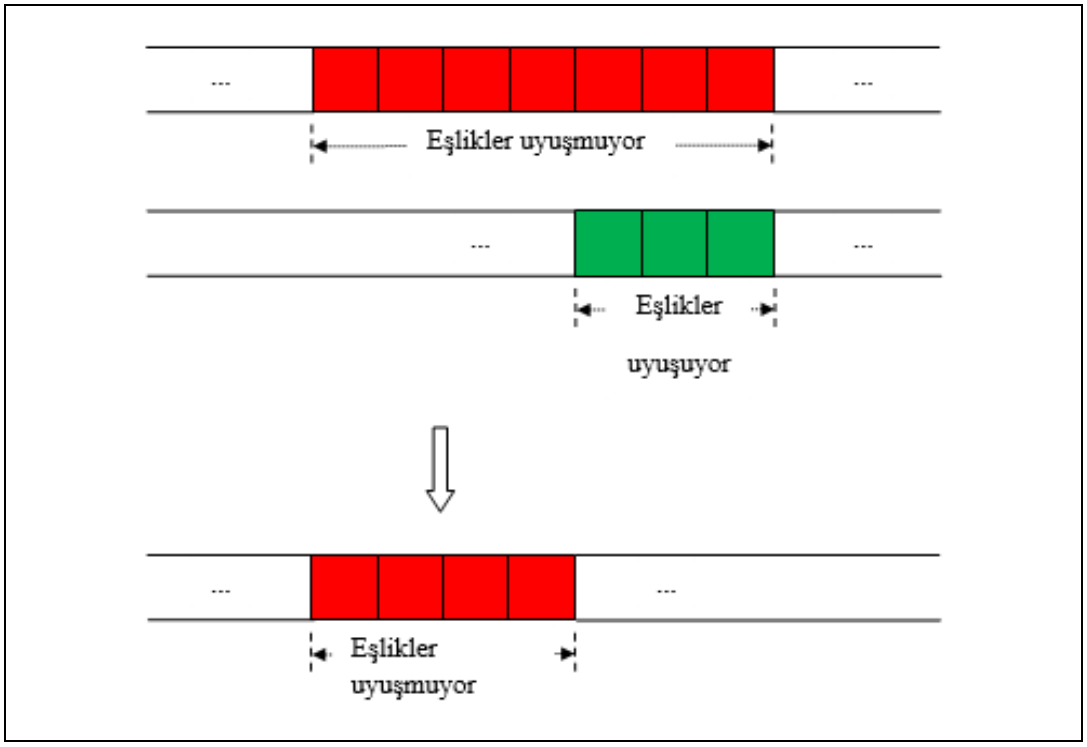


Şekil 5.5: Üç uzunluklu blok için sağdan dallanma. Eğer dallanma sağ yarıdan devam ederse, sadece bir tane tam olarak bilinen bit elde ederiz.

Bu yollarla elde edilen tam olarak bilinen bitler, BINARY uygulanmadan hemen önce bloklardan çıkarılarak hatalı bloklar daha küçük yapılabilir.

• Gözlem 2

Protokol boyunca birçok alt-bloklar da yaratılır. Bir blok üzerinde BINARY'e başlanmadan önce, Alice ve Bob, Şekil 5.6'da [109] görüldüğü gibi blok içinde yer alan ve eşliklerinde uzlaştıkları önceden hesaplanmış daha küçük blokları çıkarabilir. Böylece, bu da bloğu daha küçük yapacaktır.



Şekil 5.6: Önceki küçük bloklar durumu. Bir hatayı konumlandırmak için özyinelemeli BINARY sürecini uygulamadan önce, Alice ve Bob'un eşliği üzerinde önceden uzlaştıkları içerilen daha küçük bloklar çıkarılır.

Özetle, hata sezilen bir blok üzerinde, hata aramak üzere interaktif BINARY protokolü koşturulmadan hemen önce, blok içindeki

- tam olarak bilinen bitler ve
- eşliklerinde uyuşulan önceki daha küçük bloklar

çıkarılarak (bunun için, içerilen bu bitlerin ve küçük blokların blok içinde aranıp bulunmaları gerekir), hata aramak için, daha küçük bir blok elde edilebilir.

Daha küçük bloklar daha az sızmaya ve daha yüksek bir verimliliğe neden olacaktır. Bununla birlikte, bahsedilen fikirleri gerçeklemek için, BINARY'den hemen önce, eşliği uyuşmayan bloğun tam olarak bilinen bitlerin ve önceden yaratılmış eşliği uyuşan daha küçük blokların herhangi birini içerip içermediğine yönelik, ilave arama işlemlerine ihtiyaç vardır.

Gözlemlerimizin sonuçlarını daha iyi görmek için, CASCADE protokolünün [12] versiyonunu (literatürdeki en verimli versiyon) gerçekledik ve stratejisinde gözlemlerimize ilişkin gerekli değişiklikleri yaptık:

- Eşlik uyuşmayan bir blok için, bloğun her bir bitinin tam olarak bilinen bitlerden olup olmadığı kontrol edilir. Eğer öyle ise, ilgili bit bloktan çıkarılır.
- Eşlik uyuşmayan bir blok için, tüm önceki eşlikleri uyuşan daha küçük blokların blok içinde olup olmadığı araştırılır. Eğer öyleyse, bloktan çıkarılırlar.
- Yukarıdaki tüm çıkarmalar bittiğinde, kalan daha küçük eşlik uyuşmayan blok üzerinde BINARY protokolü çalıştırılır.

Bununla birlikte, yukarıdaki ikinci değişikliğin de eklenmesiyle benzetimlerimizde kullandığımız 20% ve 25% hata hızları (olasılıkları) için, yazılımımızdan bazı uyarılar aldık. İlginç bir şekilde, eşliği uyuşmayan (hatalı) bir blok için, önceki daha küçük blokların çıkarılmasından sonra, eşliklerin artık uyuştuğunu (uyuşur hale geldiğini) gördük: Bir şekilde hatalı bloktan hata(lar) da çıkarılmıştır! Ve, bunun gerçekten olabileceğini de fark ettik. Örneğin, $p = 0.25$ olduğunu, raunt 3'te olduğumuzu ve eşlik uyuşmayan bir blok sezdiğimizi düşünelim. Stratejimize göre, hatalı biti düzeltmeye başlamadan hemen önce, önceki rauntların, bu örnekte raunt 1 ve 2'nin, içerilen daha küçük bloklarını araştıracağız ve onları bloktan çıkaracağız. Raunt 1'de sezilemeyen iki tane hataya sahip içerilen daha küçük bir blok olduğunu düşünelim. Ve, raunt 2'de de sezilemeyen iki tane hataya sahip içerilen daha küçük bir blok olduğunu düşünelim. Bununla birlikte, bu daha küçük bloklarda da bir hatalı bitin ortak olduğunu düşünelim. Böylece, bu iki daha küçük bloğun birleşimindeki toplam hata sayısı aslında 3'tür: bir tek sayı! Bu nedenle, eğer bu daha küçük blokları daha büyük eşlik uyuşmayan bloktan çıkarırsak, gerçekten de bir eşlik değişimi yapacaktır.

Yukarıdaki problem, raunt 3'teki bloğun tek sayıda, en az 3, hataya sahip olmasıdır. Eğer blok sadece 1 hataya sahip olsaydı, problem oluşmazdı. Bu nedenle,

bu bloklarda sadece bir hataya sahip olma olasılığını arttırmalıyız. Bunu yapmanın bir yolu, her bir raundun başında blok uzunluğunu azaltmaktır. Bu nedenle, bu problemin üstesinden gelmek için, gerçeklediğimiz ve değiştirdiğimiz protokolün parametreleriyle de oynadık. Benzetimlerimize göre, yukarıdaki durum raunt 3 ve yukarısında oluşmaya başlamaktadır. Bu nedenle, protokol parametrelerini raunt 3'ten başlayarak değiştirmeye çalıştık.

• Parametrelerin Değiştirilmesi

Yukarıda gözlem/değişiklik 2 için bahsedilen problemi çözmek için, aşağıda yeni bir parametre kümesi sunmaktayız:

- Raunt 1 blok uzunluğu $k_1 = \left\lfloor \frac{0.8\ 0}{\varepsilon} \right\rfloor$,
- Raunt 2 blok uzunluğu $k_2 = 5k_1$,
- Raunt 3 blok uzunluğu $k_3 = \frac{N}{4}$,
- Raunt i blok uzunluğu $k_i = \frac{N}{2}$, $4 \leq i \leq$ ROUNDS,
- ROUNDS = 9.

Görüleceği üzere, yeni parametre setinde, [12] versiyonuna göre, sadece raunt 3 blok uzunluğunu değiştirdik (yarıya indirdik) ve toplam raunt sayısını da sadece bir azalttık. Ve bunlar denemelerimiz için yukarıda bahsedilen sorunu çözmemize yetmiştir.

Raunt 3 blok uzunluğunun bu şekilde değiştirilmesi, [12] versiyonuna göre, raunt 3'ün başında iki tane daha ilave eşlik değiş-tokuşuna neden olur. Yeni parametre kümemizde bu eklemeyi kompanze etmek için toplam raunt sayısını da bir azaltmayı tercih ettik.

Sonuç olarak, daha önceki çalışmalarda [12], [13] yapıldığı gibi, bu çalışmamızda biz de verimlilik performansını arttırmak için orijinal CASCADE protokolünün (ve aslında da [12] versiyonunun) hem stratejisini hem de parametrelerini değiştirmiş olduk.

Tüm değişiklerin (yukarıda bahsedilen Gözlem 1, Gözlem 2 ve yeni parametre kümesinin) bir arada olduğu denemelerin sonucu ise Tablo 5.1'deki gibidir. Görüldüğü gibi, bizim önerilerimizle de verimlilikte dikkate değer artış elde edilmiş ve Shannon'un limitine daha da yaklaşılmıştır.

Tablo 5.1: Verimlilik performansı karşılaştırması [110]: [12] versiyonunun ve [12] versiyonunun bizim gerçeklememizin (değişiklikler 1 & 2 ve yeni parametre kümesi hep birlikte) deneysel verimlilik performansı karşılaştırması. [12]'de $p = 0.03$ için değerler verilmediğinden, tabloda '-' ile gösterdik. Bununla birlikte, teorik limit her zaman Eşitlik (5.9) ile hesaplanabilir.

p	Orijinal CASCADE	2008 versiyonu [12]	[12]'nin bizim gerçeklememiz	Teorik limit
0.01	0.906747	0.914351	0.915918	0.919206
0.03	-	-	0.798406	-
0.05	0.662312	0.698674	0.704417	0.713603
0.10	0.413924	0.499861	0.516015	0.531004
0.15	0.218738	0.329719	0.365448	0.390159
0.20	0.068740	0.201812	0.254695	0.278071
0.25	–0.112588	0.071170	0.159215	0.188721

Yukarıdaki tüm metotlarda, gizli anahtardan hiçbir bit silinmemiştir: Alice ve Bob'un uzlaştırılmış bit dizisinin uzunluğu da başlangıçtaki gibi N bittir.

CASCADE protokolünün verimli gerçeklemesi, benzetimleri ve analizleri hakkında daha detaylı bilgi için [109], [110]'a başvurulabilir.

BU verimliliğinin başka tanımları da bulunmaktadır. Bu tanımlar hakkında da bilgi edinmek için [3], [7], [20], [96], [97]'ye bakılabilir.

Tablo 5.1'deki verimlilikleri, referans [96]'da verilen türden verimliliklere dönüştürmek için Eşitlik (5.9)'a bölmek gerekir.

Tablo 5.1'in son sütunundaki değerlerin de Eşitlik (5.9) ile hesaplanmış değerler olduğuna dikkat ediniz.

6. SONUÇ

KAD'da anahtardaki tüm hatalar düzeltildikten (BU fazı ile) ve Eve'in anahtar hakkındaki tüm bilgisi giderildikten (gizlilik arttırma fazı ile) sonra, son olarak, doğrulama fazı yer alır. Doğrulama fazında Alice ve Bob, BU fazında tüm hataların düzeltilememiş olması ihtimaline karşılık olarak, gizlilik arttırılmış (damıtılmış) anahtarlarının bir alt kümesini birbirlerine açıklarlar.

BU fazından sonra Alice ve Bob'un anahtarlarında 1 bit bile farklılık kalsa, gizlilik arttırma fazında kullanılan özet fonksiyonlarının bir özelliği olarak, Alice ve Bob'un damıtılmış anahtarları birbirinden tamamen farklı olacaktır. Dolayısıyla, doğrulama fazında açıklanan bitler (örneğin, ilk 5-10 bit) aynı ise damıtılmış anahtarlar da kesinlikle aynı olacaklardır. Karşılaştırılan bitler iki tarafta da atılırlar ve kalan her iki tarafta da ortak anahtar final anahtar adını alır.

Gizlilik arttırma fazında kullanılan özet fonksiyonunun tüm bit dizisi üzerinde değil de blok blok çalışması durumunda yukarıda bahsedilen tamamen farklı olma durumu sadece ilgili blok için geçerli olacaktır. Hatalı bloğun hangisi olduğunu bilemediğimiz böylesi durumlarda doğrulama fazı BU fazından hemen sonra gerçekleştirilir. Uzlaştırılmış anahtar kriptografik bir özet fonksiyonu ile özetlenerek özetin tamamı ya da birkaç biti karşılaştırılır. Karşılaştırma başarılı olursa evrensel özet fonksiyonlarının kullanıldığı gizlilik arttırma (son) fazına geçilir.

Bu çalışmada, interaktif BU protokollerinin (özelde, CASCADE'in) mevcut KAD sistemlerinde dikkate alınması amaçlanmıştır. Bu amaçla, dikkatli gerçekleme ve kullanım ile interaktif BU protokollerinin (özelde, CASCADE'in) en son KAD sistemleri için yeteri kadar yüksek hız ve verimliliğe ulaşabildiği gösterilmeye çalışılmıştır. Bununla birlikte, klasik kanalın çok yüksek gecikmeli olduğu durumlar için daha az interaktif olan protokollerin tercih edilebilir olacağı da açıktır.

Bizim CASCADE gerçeklememiz, fiber optik hatlar üzerinden 83.49 Mbps hızlara ulaşmıştır. Bu, bizim bildiğimiz herhangi bir BU protokolü ile daha önce gösterilmiş olandan iki kat daha hızlıdır. Bu hız, en son KAD sistemleri için ihtiyaç duyulandan da hayli daha yüksektir. Böylesi durumlarda, verimlilikten kazanmak için fazla hızın bir kısmı kurban edilebilir. Nitekim, yüksek QBER için LDPC ve kutupsal kodların standart CASCADE'den daha yüksek verimliliğe sahip olmalarına rağmen, CASCADE'in değiştirilmiş versiyonlarının hızı yeterince yüksek tutarken

daha yüksek verimliliğe ulaşabileceğini de gösterdik [96]. [109] ve [110]'da CASCADE protokolünün verimliliğini daha da arttırmaya yönelik önerilerimiz de olmuştur; ancak, tüm bu öneriler de dâhilken yeni hız performansı ölçümleri ise yapılmamıştır. Öyle ki, [109]'daki Şekil 4 durumu çok yavaş olması nedeniyle bu tez çalışmasının tamamen kapsamı dışında tutulmuştur. Dolayısıyla, gelecekte [109] ve [110]'daki tüm önerilerin daha hızlı gerçeklemeleri ve tüm bunlar dâhilken CASCADE protokolünün performansı üzerinde çalışılmaya devam edilebilir.

BU protokolünün performansı üzerinde odaklanırken, sistemin maliyeti gibi diğer tasarım kriterleri de ihmal edilmiştir. Benzetimlerimizde kullandığımız gibi 8 çekirdekli tüm bir CPU'nun BU'ya atanmasının maliyeti, bir FPGA'den daha pahalıdır. CASCADE'in FPGA'de gerçeklendiğindeki performansını bilmiyoruz. CASCADE'in FPGA gerçeklemeleri ile ilgili bir sorun, ihtiyaç duyulan yüksek miktardaki bellek olacaktır. Düşük maliyetli sistemler için, CASCADE'in donanımsal gerçeklemelerinin de daha detaylı bir çalışmasına ihtiyaç vardır.

KAD her geçen gün daha pratik hale gelmektedir. 1989 yılında IBM'de bir laboratuvar deneyinde sadece 10 bit/sn'lik bir gizli anahtar hızı ile sadece 30 cm'lik bir açık hava mesafesi üzerinden gerçeklenebilmiş iken [57], bugünlerde 50 km mertebelerinde ticari mesafelere ve 1 Mbps civarında gizli anahtar hızlarına başarıyla ulaşan uygulamalar mevcuttur [18] (IBM, AIT, Toshiba, HP, ID Quantique, MagiQ, SeQureNet gibi kurumlarla/şirketlerle irtibata geçilerek mevcut KAD sistemlerinden edinebilmek mümkündür). Artan mesafelere ve hızlara ilave olarak, bu sistemlerde kullanılmak üzere verimli ve hızlı BU yöntemlerine de ihtiyaç bulunmaktadır.

KAD, ya da daha genel olarak kuantum kriptografi, teoride kusursuz olmasına rağmen pratikte gerçeklemelerden kaynaklanan güvenlik açıkları çıkabilmektedir. Örneğin, eleme fazında rasgele seçilen tabanlar tamamen açıklanırken hata olasılığı tahmini fazında ise yine rasgele seçilen bitlerin bazılarının değerleri de yine klasik kanal üzerinden açıklanmaktadır. O halde, kullanılan rasgele sayı üreteci bir SRSÜ olmamalıdır. Aksi halde, kolayca tahmin edilebilirdir; yani, yeterince güçlü bir bilgisayar varsa ve birkaç çıkış biliniyorsa tüm diğer çıkışlar da belirlenebilir. Bu nedenle, her bir aşaması gerekli özen gösterilerek dikkatle gerçeklenmelidir.

Güvenlikle ilgili bir diğer çok önemli konu da ulusal tam bağımsızlığımız için güvenliğimizi dayandırdığımız tüm sistemleri kendimizin yapmamız gerektiğidir. Başkalarından içinde neler olduğunu bilmediğimiz karakutular almamalıyız. Aksi halde, başkalarına bağımlı kalmamız yanında, örneğin, üretici, satın aldığımız bir

KAD sisteminin içine bir SRSÜ koymuş olabilir. Ya da üretimi yaparken, belki çeşitli kleptografik teknikler [111], [112] de uygulanmış olabilir. Üretici güvenilir olsa bile bunu kötü niyetli bir çalışanı (yazılımcı, donanımcı) da yapabilir.

Sonuç olarak, riske girmemeli, kimseye güvenmemeli, kendimiz yapmalıyız. Bununla, belki bir vida, tornavida gibi ürünleri dahi kastetmesek de (onlar da olsalar hiç fena olmaz elbette) güvenliğimizle ilgili konularda işimizi şansa bırakmamalıyız. NSA, 1970'lerde DES belirlenirken kripto camiasına yardımcı olmuştu [113] ama aynı şeyi 2000'lerde AES'te yapmamış olabilir. Her ne olursa olsun, ancak kendi yaptığımız bir KAD sisteminin, ve tüm diğer (kuantum) kriptografik sistemlerin, bizi tüm NSA ve benzeri tehditlerden kesinlikle koruyacağını söyleyebiliriz.

Son söz olarak, AES'in belirlenmesinde NSA'in ortağı olan NIST de en son raporlarının sonuç bölümünde kuantum hesaplamanın geleneksel bilgi güvenlik sistemleri için ciddi bir tehdit olduğunu belirterek girişimcilere/kurumlara zamanı geldiğinde hazırlıklı olmaları için bu konudaki çalışmaları takip etmeleri ve hatta içinde dahi yer almaları yönünde tavsiyeler etmektedir [114].

Bu çalışmanın içeriğini oluşturan KAD'da BU konusunun güncel ve daha kısa bir özeti için referans [115]'e de başvurulabilir.

KAYNAKLAR

[1] Dixon A. R., Yuan Z. L., Dynes J. F., Sharpe A. W., Shields A. J., (2008), "Gigahertz decoy quantum key distribution with 1 Mbit/s secure key rate", Optics Express, 16 (23), 18790-18979.

[2] Walenta N., (2012), "1 Mbps coherent one-way QKD with dense wavelength division multiplexing and hardware key distillation", Presentation at 2nd Annual Conference on Quantum Cryptography (QCrypt 2012), Singapore, 10-14 September.

[3] Martinez-Mateo J., Elkouss D., Martin V., (2013), "Key reconciliation for high performance quantum key distribution", Scientic Reports, 3, article number 1576 | doi:10.1038/srep01576.

[4] Lodewyck J., Bloch M., García-Patrón R., Fossier S., Karpov E., Diamanti E., Debuisschert T., Cerf N. J., Tualle-Brouri R., McLaughlin S. W., Grangier P., (2007), "Quantum key distribution over 25km with an all-fiber continuous-variable system", Physical Review A, 76 (4), 042305.

[5] Jouguet P., Kunz-Jacques S., Leverrier A., (2011), "Long-distance continuous-variable quantum key distribution with a Gaussian modulation", Physical Review A, 84 (6), 062317.

[6] Brassard G., Salvail L., (1993), "Secret-key reconciliation by public discussion", In Advances in Cryptology — EUROCRYPT '93, Lecture Notes in Computer Science, 765, 410-423.

[7] Jouguet P., Kunz-Jacques S., (2014), "High performance error correction for quantum key distribution using polar codes", Quantum Information & Computation, 14 (3,4), 329-338.

[8] Mink A., Nakassis A., (2012), "LDPC for QKD reconciliation", The computing science and technology international journal, 2 (2), 6-14.

[9] Buttler W. T., Lamoreaux S. K., Torgerson J. R., Nickel G. H., Donahue C. H., Peterson C. G., (2003), "Fast, efficient error reconciliation for quantum cryptography", Physical Review A, 67 (5), 052303.

[10] Elkouss D., Martinez-Mateo J., Martin V., (2013), "Analysis of a rate-adaptive reconciliation protocol and the effect of leakage on the secret key rate", Physical Review A, 87 (4), 042334.

[11] Leverrier A., Alléaume R., Boutros J., Zémor G., Grangier P., (2008), "Multidimensional reconciliation for a continuous-variable quantum key distribution", Physical Review A, 77 (4), 042325.

[12] Yan H., Ren T., Peng X., Lin X., Jiang W., Liu T., Guo H., (2008), “Information reconciliation protocol in quantum key distribution system”, Fourth International Conference on Natural Computation (ICNC '08), 637-641, Jinan, China, 18-20 October.

[13] Sugimoto T., Yamazaki K., (2000), “A study on secret key reconciliation protocol “Cascade””. IEICE Trans. Fundamentals, E83-A (10), 1987-1991.

[14] Liu S., Tilborg H. C. A. V., Dijk M. V., (2003), “A practical protocol for advantage distillation and information reconciliation”, Designs, Codes and Cryptography, 30 (1), 39-62.

[15] Mink A., (2007), “Custom hardware to eliminate bottlenecks in QKD throughput performance”, Proc. SPIE 6780, Quantum Communications Realized, 678014, 10 September.

[16] Patel K. A., Dynes J. F., Choi I., Sharpe A. W., Dixon A. R., Yuan Z. L., Penty R. V., Shields A. J., (2012), “Coexistence of high-bit-rate quantum key distribution and data on optical fiber”, Physical Review X, 2 (4), 041010.

[17] Nauerth S., Moll F., Rau M., Fuchs C., Horwath J., Frick S., Weinfurter H., (2013), “Air-to-ground quantum communication”, Nature Photonics, 7, 382-386.

[18] Oesterling L., Hayford D., Friend G., (2012), “Comparison of commercial and next generation quantum key distribution Technologies for secure communication of information”, IEEE Conference on Technologies for Homeland Security (HST), 156 – 161, Waltham, MA, USA, 13-15 November.

[19] Stucki D., Walenta N., Vannel F., Thew R. T., Gisin N., Zbinden H., Gray S., Towery C. R., Ten S., (2009), “High rate, long-distance quantum key distribution over 250 km of ultra low loss fibres”, New Journal of Physics, 11 (7), 075003.

[20] Elkouss D., Leverrier A., Alléaume R., Boutros J. J., (2009), “Efficient reconciliation protocol for discrete-variable quantum key distribution”, IEEE International Symposium on Information Theory (ISIT 2009), 1879-1883, Seoul, South Korea, June 28-July 3.

[21] Bektaş A., (2006), “Bilgi Güvenliği ve Kriptografi”, Ocak sayısı, SPK’da Geçen Ay Dergisi.

[22] Toyran M., (2006), “Kuantum Hesaplama ve Günümüz Kriptosistemlerine Etkisi”, 3. Savunma Teknolojileri Kongresi (SAVTEK 2006), 243-252, Ankara, Türkiye, 29-30 Haziran.

[23] Toyran M., (2006), “Kuantum Kriptografi, Benzetimi ve Analizleri”, 15. İstatistik Araştırma Sempozyumu, 441-456, Ankara, Türkiye, 11-12 Mayıs.

[24] Kerckhoffs A., (1883), “La cryptographie militaire”, Journal des sciences militaires, IX, 5–83.

[25] Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Can M. A., Berber S., (2011), “Bilgi Güvenliğinde Kuantum Teknikler”, 4. Ağ ve Bilgi Güvenliği Sempozyumu: Kurumsal ve Bireysel Bilgi Güvenliği ve Kamu Politikaları (ABGS 2011), 98-107, Ankara, Türkiye, 25-26 Kasım.

[26] Schneier B., (1996), “Applied Cryptography: Protocols, Algorithms and Source Code in C”, 2nd Edition, John Wiley & Sons Inc.

[27] Vernam G., (1926), “Cipher printing telegraph systems for secret wire and radio telegraphic communications”, Journal of the American Institute of Electrical Engineers, XLV, 109-115.

[28] Shannon C. E., (1949), “Communication theory of secrecy systems”, Bell System Technical Journal, 28 (4), 656-715.

[29] Rivest R. L., Shamir A., Adleman L. M., (1978), “A Method of Obtaining Digital Signatures and Public-Key Cryptosystems”, Communications of the ACM, 21 (2), 120-126.

[30] Shor P. W., (1994), “Algoritms for quantum computation: discrete logarithms and factoring”, 35th Annual Symposium on Fundamentals of Computer Science, 124-134, Santa Fe, New Mexico, 20-22 November.

[31] Dereli T., Verçin A., (2009), “Kuantum Mekaniği Temel Kavramlar ve Uygulamaları”, Genişletilmiş İkinci Basım, TÜBA.

[32] Trappe W., Washington L. C., (2002), “Introduction to Cryptography with Coding Theory”, 1st Edition, Prentice-Hall Inc.

[33] Zettili N., (2009), “Quantum Mechanics: Concepts and Applications”, 2nd Edition, Wiley.

[34] Heisenberg W., (1927), “Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik”, Zeitschrift für Physik, 43 (3-4), 172-198.

[35] Sümer A., (1987), “Modern Teknik Fizik”, İstanbul Teknik Üniversitesi Matbaası, Gümüşsuyu.

[36] Wootters W., Zurek W., (1982), “A Single Quantum Cannot be Cloned”. Nature, 299, 802-803.

[37] Toyran M., (2003), “Kuantum Kriptografi”, Yüksek Lisans Tezi, İstanbul Teknik Üniversitesi.

[38] Toyran M., Erdem S. S., Gedikbey B., (2005), "Kuantum Kriptografi", 11. Elektrik Elektronik Bilgisayar Mühendisliği Ulusal Kongresi, 1-4, Şişli, İstanbul, Türkiye, 22-25 Eylül.

[39] Toyran M., (2006), "Optik Ağlarda Kuantum Kriptografi Kullanarak Akıllı İletişim", Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu (ASYU 2006), 1-5, İstanbul, Türkiye, 31 Mayıs - 2 Haziran.

[40] Toyran M., (2006), "EEB Mühendislikleri için Bilgisayar Destekli Eğitim: Kuantum Kriptografi Benzetim ve Eğitim Uygulaması", Elektrik, Elektronik, Bilgisayar Mühendislikleri Eğitimi 3. Ulusal Sempozyumu (EEB'06), 111-115, İstanbul, Türkiye, 16-18 Kasım.

[41] Toyran M., (2006), "EEB Mühendisliklerinde Kuantum Hesaplama Eğitimi", Elektrik, Elektronik, Bilgisayar Mühendislikleri Eğitimi 3. Ulusal Sempozyumu (EEB'06), 107-110, İstanbul, Türkiye, 16-18 Kasım.

[42] Toyran M., (2006), "Optik Ağlarda Kuantum Kriptografi Kullanarak Güvenli İletişim", Elektrik, Elektronik, Bilgisayar Mühendisliği Sempozyumu (ELECO 2006), 121-125, Bursa, Türkiye, 6-10 Aralık.

[43] Toyran M., (2007), "Kuantum Kriptografi", IEEE 15. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2007), 1-4, Eskişehir, Türkiye, 11-13 Haziran.

[44] Şahin A. B., Selçuk G., (2006), "İletişim Ağ Güvenliğinde Son Aşama: Kuantum Kriptografi ve Fiber Optik Ortamda Kuantum Temelli Rastsal Sayı Üretimi", 1. Ulusal Elektronik İmza Sempozyumu, 1-6, Ankara, Türkiye, 7-8 Aralık.

[45] Bernstein D. J., Buchmann J., Dahmen E., (2008), "Post-Quantum Cryptography", 1st edition, Springer.

[46] Kollmitzer C., Pivk M., (2010), "Applied Quantum Cryptography" , 1st edition, Springer.

[47] Toyran M., (2007), "Rasgele Sayıların Verimli Kullanımı", IEEE 15. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2007), 1-4, Eskişehir, Türkiye, 11-13 Haziran.

[48] Toyran M., Berber S., (2009), "Efficient Implementation of Diffie-Hellman (DH) Key Distribution Algorithm in Pool-Based Cryptographic Systems (PBCSs)", 6th International Conference on Electrical and Electronics Engineering (ELECO 2009), 1-5, Bursa, Turkey, 5-8 November.

[49] Toyran M., Berber S., (2010), "Eliptik Eğri Diffie-Hellman (EEDH) Anahtar Dağıtım Algoritmasının Havuz-Tabanlı Kriptografik Sistemlerde (HTKSlerde) Verimli Gerçeklemesi", IEEE 18. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2010), 780-783, Diyarbakır, Türkiye, 22-24 Nisan.

[50] Can M. A., Hacızade F., Hasekioğlu A., Pedersen T., Toyran M., (2012), "Quantum Random Number Generators", 5th International Conference on Information Security and Cryptology (ISCTurkey2012), 268-272, Ankara, Türkiye, 17-18 Mayıs.

[51] Web 1, (2015), QUANTIS True RANDOM NUMBER Generator Exploiting QUANTUM PHYSICS PCI board 16Mbits/sec, http://www.idquantique.com/random-number-generators/products/products-overview.html, (Erişim Tarihi: 05/02/2015).

[52] Shannon C. E., (1948), "A Mathematical Theory of Communication", Bell System Technical Journal, 27 (3), 379-423.

[53] Korkmaz B. A., Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Mutaf P., Can M. A., (2012), "Kuantum Anahtar Dağıtımında (KAD) Gizli Anahtar Uzlaştırma: CASCADE Protokolü ve LDPC Kodları", 5th International Conference on Information Security and Cryptology (ISCTurkey2012), 256-261, Ankara, Türkiye, 17-18 Mayıs.

[54] Cover T. M., Thomas J. A., (2006), "Elements of Information Theory 2nd Edition", 2nd Edition, Wiley-Interscience.

[55] Bennett C. H., Brassard G., (1984), "Quantum cryptography: Public key distribution and coin tossing", IEEE International Conference on Computers, Systems and Signal Processing, 175-179, Bangalore, India, 9-12 December.

[56] Bennett C. H., Brassard G., (1989), "The dawn of a new era for quantum cryptography: The experimental prototype is working!" ACM SIGACT News, 20 (4), 78–80.

[57] Bartlett B. C., (2004), "Securing Key Distribution with Quantum Cryptography", GSEC Practical Assignment, Version 1.4B, Option 1, InfoSec Reading Room, SANS Institute, USA.

[58] Wang S., Chen W., Guo J.-F., Yin Z.-Q., Li H.-W., Zhou Z., Guo G.-C., Han Z.-F., (2012), "2 ghz clock quantum key distribution over 260 km of standard telecom fiber," Optics Letters, 37 (6), 1008-1010.

[59] Schmitt-Manderbach T., Weier H., Fürst M., Ursin R., Tiefenbacher F., Scheidl T., Perdigues J., Sodnik Z., Kurtsiefer Ch., Rarity J. G., Zeilinger A., Weinfurter H., (2007), "Experimental demonstration of free-space decoy-state quantum key distribution over 144 km", Physical Review Letters, 98 (1), 010504.

[60] Dixon A. R., Sato H., (2014), "High speed and adaptable error correction for megabit/s rate quantum key distribution" Scientific Reports, 4, 7275.

[61] Cobourne S., (2011), "Quantum Key Distribution Protocols and Applications", Technical Report No: RHUL-MA-2011-05, Department of Mathematics, University of London, England.

[62] Ekert A. K., Rarity J. G., Tapster P. R., Palma G. M., (1992), "Practical Quantum Cryptography Based on Two-Photon Interferometry", Physical Review Letters, 69 (9), 1293-1295.

[63] Ekert A. K., (1991), "Quantum cryptography based on Bell's theorem", Physical Review Letters, 67 (6), 661-663.

[64] Bennett C. H., (1992), "Quantum cryptography using any two nonorthogonal states", Physical Review Letters, 68 (21), 3121.

[65] Bruss D., (1998), "Optimal eavesdropping in quantum cryptography with six states", Physical Review Letters, 81 (14), 3018-3021.

[66] Bechmann-Pasquinucci H., Gisin N., (1999), "Incoherent and Coherent Eavesdropping in the 6-state Protocol of Quantum Cryptography", Physical Review A, 59 (6), 4238-4248.

[67] Scarani V., Acín A., Ribordy G., Gisin N., (2004), "Quantum Cryptography Protocols Robust against Photon Number Splitting Attacks for Weak Laser Pulse Implementations", Physical Review Letters, 92 (5), 057901.

[68] Ralph T. C., (1999), "Continuous variable quantum cryptography", Physical Review A, 61 (1), 010303.

[69] Hillery M., (2000), "Quantum cryptography with squeezed states", Physical Review A, 61, 022309.

[70] Cerf N. J., Lévy M., Assche G. V., (2001), "Quantum distribution of Gaussian keys using squeezed states", Phys. Rev. A 63, 052311.

[71] Gottesman D., Preskill J., (2001), "Secure quantum key distribution using squeezed states", Physical Review A, 63, 022309.

[72] Grosshans F., Grangier P., (2002), "Continuous Variable Quantum Cryptography Using Coherent States", Physical Review Letters, 88 (5), 057902.

[73] Silberhorn C., Ralph T. C., Lütkenhaus N., Leuchs G., (2002), "Continuous Variable Quantum Cryptography: Beating the 3 dB Loss Limit", Physical Review Letters, 89 (16), 167901.

[74] Inoue K., Waks E., Yamamoto Y., (2002), "Differential Phase Shift Quantum Key Distribution", Physical Review Letters, 89 (3), 037902.

[75] Stucki D., Brunner N., Gisin N., Scarani V., Zbinden H., (2005), "Fast and simple one-way quantum key distribution", Applied Physics Letters, 87 (19), 194108.

[76] NIST, (2001), “Announcing the ADVANCED ENCRYPTION STANDARD (AES)”, Technical Report No: FIPS 197, National Institute of Standards and Technology (NIST), USA.

[77] Uyar A., Kılınç H. H., Erdem S. S., Toyran M., (2005), “Use of Rijndael Block Cipher on J2ME Devices for Encryption and Hashing”, 10th Nordic Workshop on Secure IT-Systems (NORSEC 2005), 144-155, Tartu, Estonia, 20-21 October.

[78] Gisin N., Ribordy G., Tittel W., Zbinden H., (2002), “Quantum Cryptography”, Reviews of Modern Physics, 74 (1), 145-195.

[79] Scarani V., Bechmann-Pasquinucci H., Cerf N. J., Dusek M., Lutkenhaus N., Peev M., (2009), “The Security of Practical Quantum Key Distribution”, Reviews of Modern Physics, 81 (3), 1301-1350.

[80] Williams C. P., Clearwater S. H., (1998), “Explorations in QUANTUM COMPUTING”, Springer-Verlag.

[81] Nielsen M. A., Chuang I. L., (2000), “Quantum Computation and Quantum Information”, Cambridge University Press.

[82] Hamitoğulları C., Sınır E. Y., (2004), “KOD KİTABI, Eski Mısır’dan Kuantum Kriptolojisine Gizlilik Bilimi”, İstanbul Klan Yayınları.

[83] Uçan O. N., Osman O., (2006), “Haberleşme Teorisi ve Mühendislik Uygulamaları”, 1. Basım, Nobel Yayın Dağıtım.

[84] Afacan A., (2003), “Schaum Serisinden Teori ve Problemlerle Analog ve Sayısal İletişim”, Nobel Yayın Dağıtım.

[85] Bae J., Acín A., (2007), “Key distillation from quantum channels using two-way communication protocols”, Physical Review A, 75 (1), 012334.

[86] Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Can M. A., Berber S., (2012), “CASCADE HATA DÜZELTME PROTOKOLÜ ve LDPC HATA DÜZELTME KODLARININ KARŞILAŞTIRILMASI”, 20th Signal Processing and Communications Applications Conference (SİU2012), 1-4, Mugla, Türkiye, 18-20 Nisan.

[87] Wegman M. N., Carter L., (1981), “New hash functions and their use in authentication and set equality”, Journal of Computer and System Sciences, 22 (3), 265-279.

[88] Weier H., (2003), “Experimental Quantum Cryptography”, Diplomarbeit, Technische Universität München.

[89] Lo H. K., Chau H. F., (1999), “Unconditional security of quantum key distribution over arbitrary long distances”, Science, 283 (5410), 2050-2056.

[90] Mayers D., (1998), “Unconditional security in quantum cryptography”, Journal of the Association for Computing Machinery, 48 (3), 351-406.

[91] Shor P. W., Preskill J., (2000), “Simple proof of security of the BB84 Quantum key distribution protocol”, Physical Review Letters, 85 (2), 441-444.

[92] Renner R., Gisin N., Kraus B., (2005), “Information-theoretic security proof for quantum-key-distribution protocols”, Physical Review A, 72 (1), 012332.

[93] Kaplan Y., (2000), “Veri Haberleşmesi Temelleri”, 1. Basım, Papatya Yayıncılık.

[94] Çölkesen R., Örencik B., (2002), “Bilgisayar Haberleşmesi ve Ağ Teknolojileri”, 1. Basım (yeniden düzenlenmiş ve genişletilmiş), Papatya Yayıncılık.

[95] Richardson T., Urbanke R., (2008), “Modern Coding Theory”, Cambridge University Press.

[96] Pedersen T. B., Toyran M., (2015), “High Performance Information Reconciliation for QKD with CASCADE”, Quantum Information & Computation, 15 (5&6), 419-434.

[97] Elkouss D., Martinez J., Lancho D., Martin V., (2010), “Rate Compatible Protocol for Information Reconciliation: An application to QKD”, IEEE Information Theory Workshop (ITW), 145-149, Cairo, Egypt, 6-8 January.

[98] Slepian D., Wolf J., (1973), “Noiseless coding of correlated information sources”, IEEE Transactions on Information Theory, 19 (4), 471-480.

[99] Assche G. V., (2006), “Quantum Cryptography and Secret-Key Distillation”, Cambridge University Press.

[100] Calver T., (2011), “An Empirical Analysis of the Cascade Secret Key Reconciliation Protocol for Quantum Key Distribution”, Master's Thesis, Air Force Institute of Technology.

[101] Gallager R. G., (1963), “Low-density parity-check codes”, MIT press.

[102] Arıkan E., (2008), “Channel polarization: A method for constructing capacity-achieving code”, IEEE International Symposium on Information Theory (ISIT 2008), 1173–1177, Toronto, Canada, 6-11 July.

[103] Carter L., Wegman M. N., (1979), “Universal classes of hash functions”, Journal of Computer and System Sciences, 18 (2), 143-154.

[104] Stinson D. R., (1992), “Universal hashing and authentication codes”, Lecture Notes in Computer Science, 576, 74-85.

[105] Johnson J. S., (2012), “An analysis of error reconciliation protocols for use in quantum key distribution”, Master's Thesis, Air Force Institute of Technology.

[106] Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Can M. A., Berber S., (2013), “CASCADE Hata Düzeltme Protokolü Üzerine Bir İnceleme”, 21th Signal Processing and Communications Applications Conference (SİU2013), 1-4, Girne, KKTC, 24-26 Nisan.

[107] Web 2, (2015), AIT QKD Software, Austrian Institute of Technology, https://sqt.ait.ac.at/software/projects/qkd-software, (Erişim Tarihi: 05/02/2015).

[108] Liu S., (2002), “Information-theoretic secret key agreement”, Ph.D. Thesis, Technische Universiteit Eindhoven.

[109] Toyran M., Pedersen T. B., (2012), “More Efficient Implementations of CASCADE”, 2nd Annual Conference on Quantum Cryptography (QCRYPT 2012), Poster presentation, Singapore, 10-14 September.

[110] Toyran M., (2016), “CASCADE Bilgi Uzlaştırma Protokolünün Daha Verimli Gerçeklemeleri”, IEEE 24. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2016), 1-4, Zonguldak, Türkiye, 16-19 Mayıs.

[111] Young A., Yung M., (1996), “The Dark Side of Black-Box Cryptography, or: Should we trust Capstone?” 16th Annual International Cryptology Conference (CRYPTO ’96), 89-103, Santa Barbara, California, USA, 18-22 August.

[112] Young A., Yung M., (2004), “Malicious Cryptography: Exposing Cryptovirology”, Wiley.

[113] Ciampa M., (2014), “CompTIA Security+ Guide to Network Security Fundamentals”, 5th Edition, Cengage Learning.

[114] ETSI, (2014), “Quantum Safe Cryptography and Security; An introduction, benefits, enablers and challenges”, V1.0.0 (2014-10), European Telecommunications Standards Institute, FRANCE.

[115] Toyran M., (2016), “Kuantum Anahtar Dağıtımında Bilgi Uzlaştırma Problemi Üzerine Bir İnceleme”, IEEE 24. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2016), 1-4, Zonguldak, Türkiye, 16-19 Mayıs.

ÖZGEÇMİŞ

Lisans derecelerini İstanbul Teknik Üniversitesi (İTÜ) Elektronik ve Haberleşme Mühendisliği ile Kontrol ve Bilgisayar Mühendisliği bölümlerinde, sırasıyla, 2000 ve 2001 yıllarında tamamlamıştır. Yüksek lisans eğitimini 2003 yılında İTÜ Bilgisayar Mühendisliği bölümünde tamamlamıştır. Doktora eğitimini 2016 yılında Gebze Teknik Üniversitesi (GTÜ) Fen Bilimleri Enstitüsü (FBE) Elektronik Mühendisliği Anabilim Dalında tamamlamıştır. 2000 yılından beri TÜBİTAK – UEKAE'de ARGE (araştırma ve geliştirme) faaliyetlerine devam etmektedir. Başlıca çalışma konuları, Klasik ve Kuantum Kriptoloji ile Bilgi ve Kodlama Teorisi ve bunların daha çok yazılımsal gerçeklemeleri üzerinedir.

EKLER

Ek A: Tez Çalışması Kapsamında Yapılan Yayınlar

[1] Toyran M., Erdem S. S., Gedikbey B., (2005), "Kuantum Kriptografi", 11. Elektrik Elektronik Bilgisayar Mühendisliği Ulusal Kongresi, 1-4, Şişli, İstanbul, Türkiye, 22-25 Eylül.

[2] Uyar A., Kılınç H. H., Erdem S. S., Toyran M., (2005), "Use of Rijndael Block Cipher on J2ME Devices for Encryption and Hashing", 10th Nordic Workshop on Secure IT-Systems (NORSEC 2005), 144-155, Tartu, Estonia, 20-21 October.

[3] Toyran M., (2006), "Kuantum Kriptografi, Benzetimi ve Analizleri", 15. İstatistik Araştırma Sempozyumu, 441-456, Ankara, Türkiye, 11-12 Mayıs.

[4] Toyran M., (2006), "Optik Ağlarda Kuantum Kriptografi Kullanarak Akıllı İletişim", Akıllı Sistemlerde Yenilikler ve Uygulamaları Sempozyumu (ASYU 2006), 1-5, İstanbul, Türkiye, 31 Mayıs - 2 Haziran.

[5] Toyran M., (2006), "Kuantum Hesaplama ve Günümüz Kriptosistemlerine Etkisi", 3. Savunma Teknolojileri Kongresi (SAVTEK 2006), 243-252, Ankara, Türkiye, 29-30 Haziran.

[6] Toyran M., (2006), "EEB Mühendisliklerinde Kuantum Hesaplama Eğitimi", Elektrik, Elektronik, Bilgisayar Mühendislikleri Eğitimi 3. Ulusal Sempozyumu (EEB'06), 107-110, İstanbul, Türkiye, 16-18 Kasım.

[7] Toyran M., (2006), "EEB Mühendislikleri için Bilgisayar Destekli Eğitim: Kuantum Kriptografi Benzetim ve Eğitim Uygulaması", Elektrik, Elektronik, Bilgisayar Mühendislikleri Eğitimi 3. Ulusal Sempozyumu (EEB'06), 111-115, İstanbul, Türkiye, 16-18 Kasım.

[8] Toyran M., (2006), "Optik Ağlarda Kuantum Kriptografi Kullanarak Güvenli İletişim", Elektrik, Elektronik, Bilgisayar Mühendisliği Sempozyumu (ELECO 2006), 121-125, Bursa, Türkiye, 6-10 Aralık.

[9] Toyran M., (2007), "Kuantum Kriptografi", IEEE 15. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2007), 1-4, Eskişehir, Türkiye, 11-13 Haziran.

[10] Toyran M., (2007), "Rasgele Sayıların Verimli Kullanımı", IEEE 15. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2007), 1-4, Eskişehir, Türkiye, 11-13 Haziran.

[11] Toyran M., Berber S., (2009), "Efficient Implementation of Diffie-Hellman (DH) Key Distribution Algorithm in Pool-Based Cryptographic Systems

(PBCSs)", 6th International Conference on Electrical and Electronics Engineering (ELECO 2009), 1-5, Bursa, Turkey, 5-8 November.

[12] Toyran M., Berber S., (2010), "Eliptik Eğri Diffie-Hellman (EEDH) Anahtar Dağıtım Algoritmasının Havuz-Tabanlı Kriptografik Sistemlerde (HTKSlerde) Verimli Gerçeklemesi", IEEE 18. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2010), 780-783, Diyarbakır, Türkiye, 22-24 Nisan.

[13] Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Can M. A., Berber S., (2011), "Bilgi Güvenliğinde Kuantum Teknikler", 4. Ağ ve Bilgi Güvenliği Sempozyumu: Kurumsal ve Bireysel Bilgi Güvenliği ve Kamu Politikaları (ABGS 2011), 98-107, Ankara, Türkiye, 25-26 Kasım.

[14] Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Can M. A., Berber S., (2012), "CASCADE HATA DÜZELTME PROTOKOLÜ ve LDPC HATA DÜZELTME KODLARININ KARŞILAŞTIRILMASI", 20th Signal Processing and Communications Applications Conference (SİU2012), 1-4, Mugla, Türkiye, 18-20 Nisan.

[15] Can M. A., Hacızade F., Hasekioğlu A., Pedersen T., Toyran M., (2012), "Quantum Random Number Generators", 5th International Conference on Information Security and Cryptology (ISCTurkey2012), 268-272, Ankara, Türkiye, 17-18 Mayıs.

[16] Korkmaz B. A., Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Mutaf P., Can M. A., (2012), "Kuantum Anahtar Dağıtımında (KAD) Gizli Anahtar Uzlaştırma: CASCADE Protokolü ve LDPC Kodları", 5th International Conference on Information Security and Cryptology (ISCTurkey2012), 256-261, Ankara, Türkiye, 17-18 Mayıs.

[17] Toyran M., Pedersen T. B., (2012), "More Efficient Implementations of CASCADE", 2nd Annual Conference on Quantum Cryptography (QCRYPT 2012), Poster presentation, Singapore, 10-14 September.

[18] Toyran M., Pedersen T. B., Hasekioğlu A. S. A., Can M. A., Berber S., (2013), "CASCADE Hata Düzeltme Protokolü Üzerine Bir İnceleme", 21th Signal Processing and Communications Applications Conference (SİU2013), 1-4, Girne, KKTC, 24-26 Nisan.

[19] Pedersen T. B., Toyran M., (2015), "High Performance Information Reconciliation for QKD with CASCADE", Quantum Information & Computation, 15 (5&6), 419-434.

[20] Toyran M., (2016), "Kuantum Anahtar Dağıtımında Bilgi Uzlaştırma Problemi Üzerine Bir İnceleme", IEEE 24. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2016), 1-4, Zonguldak, Türkiye, 16-19 Mayıs.

[21] Toyran M., (2016), "CASCADE Bilgi Uzlaştırma Protokolünün Daha Verimli Gerçeklemeleri", IEEE 24. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2016), 1-4, Zonguldak, Türkiye, 16-19 Mayıs.