T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

HATALARLA ÖĞRENME TABANLI TORUS TAM HOMOMORFİK ŞİFRELEME ŞEMASININ KALAN SAYILAR SİSTEMİ VARYANTI

YÜKSEK LİSANS TEZİ

Serra SAZOĞLU

Matematik Anabilim Dalı

Cebir ve Sayılar Teorisi Bilim Dalı

TEMMUZ 2023

T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

HATALARLA ÖĞRENME TABANLI TORUS TAM HOMOMORFİK ŞİFRELEME ŞEMASININ KALAN SAYILAR SİSTEMİ VARYANTI

YÜKSEK LİSANS TEZİ

Serra SAZOĞLU

Matematik Anabilim Dalı

Cebir ve Sayılar Teorisi Bilim Dalı

Tez Danışmanı: Prof. Dr. Mehmet ÖZEN

TEMMUZ 2023

Serra Sazoğlu tarafından hazırlanan "Hatalarla Öğrenme Tabanlı Torus Tam Homomorfik Şifrelemenin Kalan Sayılar Sistemi Varyantı" adlı tez çalışması 11.07.2023 tarihinde aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Matematik Anabilim Dalı **Cebir ve Sayılar teorisi** Bilim Dalı'nda Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı : **Prof. Dr. Mehmet ÖZEN (Danışman) …….**
Sakarya Üniversitesi

Jüri Üyesi : **Doç. Dr. Ünal ÇAVUŞOĞLU ……**
Sakarya Üniversitesi

Jüri Üyesi : **Doç. Dr. Hakan ADIGÜZEL …….**
Sakarya Uygulamalı Bilimler Üniversitesi

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum "Hatalarla Öğrenme Tabanlı Torus Tam Homomorfik Şifreleme Şemasının Kalan Sayılar Sistemi Varyantı" başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete'de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi'nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(20/07/2023)

Serra Sazoğlu

TEŞEKKÜR

Yüksek Lisans eğitimim boyunca beni teşvik eden, titizlikle yönlendiren ve değerli bilgi ve deneyimlerini benimle paylaşan kıymetli danışman hocam Prof. Dr. Mehmet ÖZEN'e teşekkürlerimi sunarım. Tez çalışmalarımdaki programlamada yol gösteren ve yardımlarını benden esirgemeyen değerli hocalarım Doç. Dr. Ünal ÇAVUŞOĞLU'na ve Dr. Öğr. Üyesi Abdullah SEVİN'e teşekkürlerimi bir borç bilirim.

Ayrıca hayatım boyunca maddi ve manevi desteklerini benden esirgemeyen, her zaman yanımda olan kıymetli babama ve anneme, her türlü desteği için sevgili kardeşlerime çok teşekkür ederim.

Son olarak Yüksek Lisans eğitimim süresince aldığım BİDEB 2210-A bursu için TÜBİTAK'a teşekkürlerimi sunarım.

Serra SAZOĞLU

İÇİNDEKİLER

Sayfa

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ ..... **v**
TEŞEKKÜR ..... **vii**
İÇİNDEKİLER ..... **ix**
KISALTMALAR ..... **xi**
SİMGELER ..... **xiii**
TABLO LİSTESİ ..... **xv**
ŞEKİL LİSTESİ ..... **xvii**
SUMMARY ..... **xxi**
1. GİRİŞ **1**
1.1. Literatür Taraması ve Tezin Amacı 1
1.2. Tezin İçeriği 2
1.3. Temel Kavramlar 3
1.3.1. Cebirsel altyapı 3
1.3.2. Kriptoloji 9
1.3.2.1. Kriptolojiye giriş 9
1.3.2.2. Homomorfik şifreleme 12
1.3.3. Çin Kalan Teoremi gösterimi (CRT representation) 18
2. TORUS TAM HOMOMORFİK ŞİFRELEME (TFHE) **21**
2.1. Hatalarla Öğrenme Problemi 22
2.2. TFHE Şifreleme Şeması 24
2.2.1. Hatalarla öğrenme tabanlı 24
2.2.1.1. Şifreleme şeması 24
2.2.1.2. Kodlama-dekodlama 25
2.2.1.3. Homomorfik toplama 26
2.2.1.4. Homomorfik çarpma 26
2.2.2. Halkalarda hatalarla öğrenme tabanlı 28
2.2.2.1. Şifreleme şeması 29
2.2.2.2. Kodlama-dekodlama 29
2.2.2.3. Homomorfik toplama 30
2.2.2.4. Homomorfik çarpma 30
2.2.3. GSW tabanlı 31
2.3. Ayrıklaştırılmış Torus Üzerinde TFHE Şeması 33
2.3.1. Hatalarla öğrenme tabanlı 33
2.3.1.1. Şifreleme şeması 33
2.3.1.2. Homomorfik toplama 34
2.3.1.3. Homomorfik çarpma 34
3. TLWE ŞEMASININ RNS VARYANTI **37**
3.1. Şifreleme Şeması 37
3.2. Kodlama-Dekodlama 40
3.3. Homomorfik Toplama 41
3.4. Bilinen Bir Sabitle Homomorfik Çarpma 41

3.5. Homomorfik Çarpma 42
4. SONUÇ VE ÖNERİLER 45
KAYNAKLAR 47
ÖZGEÇMİŞ 51

KISALTMALAR

BGV : Brakerski-Gentry-Vaikuntanathan

BFV : Brakerski-Fan-Vercauteren

CKKS : Cheon-Kim-Kim Song

CRT : Chinese Remainder Theorem (Çin Kalan Teoremi)

DGHV : Djik-Gentry-Halevi-Vaikuntanathan

FHE : Fully Homomorphic Encryption (Tam Homomorfik Şifreleme)

FHEW : Fastest Homomorphic Encryption in the West (Batıdaki En Hızlı Homomorfik Şifreleme)

FV : Fan-Vercauteren

GSW : Gentry-Sahai-Waters

HE : Homomorphic Encryption (Homomorfik Şifreleme)

IoT : Internet of Things (Nesnelerin İnterneti)

LWE : Learning With Errors (Hatalarla Öğrenme)

PHE : Partially Homomorphic Encryption (Kısmi Homomorfik Şifreleme)

RLWE : Ring Learning With Errors (Halkalarda Hatalarla Öğrenme)

RNS : Residue Number System (Kalan Sayılar Teoremi)

SwHE : Somewhat Homomorphic Encryption (Yarı Homomorfik Şifreleme)

TFHE : Torus Fully Homomorphic Encryption (Torus Tam Homomorfik Şifreleme)

SİMGELER

e	: Gürültü-Hata
$\boldsymbol{\chi}$	: Dağılım
$\mathbb{Z}$	: Tam sayılar
$\mathbb{R}$	: Reel Sayılar
$\boldsymbol{T}$	: Torus
Δ	: Ölçekleme Çarpım Elemanı
$\boldsymbol{\mu}$	: Açık mesaj
m	: Mesaj
c	: Şifreli Mesaj
s	: Gizli anahtar
pk	: Açık Anahtar

TABLO LİSTESİ

Sayfa

Tablo 4.1. TFHE şemasının şifreleme ve deşifreleme sürelerinin örnekleri. 46
Tablo 4.2. RNS-TFHE şemasının şifreleme ve deşifreleme sürelerinin örnekleri. ... 46

ŞEKİL LİSTESİ

Sayfa

Şekil 1.1. Kriptoloji Alt Alanları 9
Şekil 1.2. Temel Şifreleme Modeli 10
Şekil 2.1. TFHE Sözde Kodu 28
Şekil 3.1. Torus 16 39
Şekil 4.1. RNS-TFHE Sözde Kodu 45

HATALARLA ÖĞRENME TABANLI TORUS HOMOMORFİK ŞİFRELEME ŞEMASININ KALAN SAYILAR SİSTEMİ VARYANTI

ÖZET

Homomorfik şifreleme, şifrelenmiş veriler üzerinde şifre çözmeden homomorfik işlemler gerçekleştirilmesine olanak tanıyan ve veri güvenliğini sağlayan özel bir şifreleme türüdür. Bulut sistemi kullanılarak tüm sektörler adına müşteri ve/veya hasta gizliliği için geliştirilmesi gereken en önemli sistemlerden biridir. Bulut platformlarında, veri güvenliğini sağlamak için literatürde birçok şifreleme algoritması geliştirilmiştir. Sunucu tarafında veriler üzerinde işlem yapılabilmesi için verilerin çözülmesine ihtiyaç bulunmaktadır. Bu sebeple veri gizliliğinin sağlanması konusunda tehditler oluşmaktadır. Ayrıca IoT sistemlerinin kullanımının artmasıyla birlikte bu sistemlerin kullanıldığı alanlardaki en büyük sorun yine veri koruma ve veri gizliliği olmuştur. Dahası yapay zeka uygulamalarının pek çoğunda güvenliği sağlamak için de homomorfik şifreleme son zamanların en çok tercih edilen şifreleme yöntemlerinden birisi haline gelmiştir. Homomorfik şifreleme sayesinde şifreli veriler üzerinde işlem yapılabilmekte ve verinin çözülmesine ihtiyaç kalmamaktadır. Fakat şifreleme süreleri ve kaynak kullanımı noktasında dezavantajları bulunmaktadır. Homomorfik şifrelemeden kaynaklanan uzun şifreleme süresi şemanın pratikte kullanımına engel teşkil etmektedir. Bu nedenle, homomorfik şemaları geliştirmeyi ve iyileştirmeyi amaçlayan çalışmalar büyük önem taşımaktadır. Ancak şifreleme şemalarının oluşturulması, bu şemaların maliyeti ve hesaplama boyutu nedeniyle teoride kalmıştır. Bu nedenle şifreleme şemalarını hızlandırmak ve maliyetleri düşürmek için birçok çalışma yürütülmektedir.

Bu tezde, diğer birçok homomorfik şifreleme şemasına uygulanarak şemaları geliştiren Kalan Sayı Sistemi (RNS) varyantını, yakın zamanda büyük bir atılım gerçekleştiren TFHE (Torus Tam Homomorfik Şifreleme) şemasına uygulayarak homomorfik şifrelemeye katkıda bulunulması amaçlanmaktadır. Bu kapsamda Torus Tam Homomorfik Şifreleme (TFHE) algoritması üzerinde hesaplama yükünün azaltılması ve işlem sürelerinin kısaltılması, kaynak kullanımının azaltılması hedeflenmektedir. Tez kapsamındaki bilimsel çalışmalarımızda homomorfik şifrelemenin dezavantajlarını gidermek için Çin Kalan Teoremi (CRT), literatürdeki çalışmalardan farklı olarak TFHE şemasına uygulanacaktır ve şifreleme süresi açısından iyileştirme sağlanacaktır. Kullanılacak olan CRT ile TFHE bileşenlerinin ve çalıştığı uzayın farklılığı, diğer homomorfik şifreleme şemalarından ayrılmaktadır. Çin Kalan Teoremi (CRT) yöntemi, şemaya RNS uygulamak için kullanılacaktır. Şemaya RNS uygulayarak, şemanın şifreleme ve deşifreleme sürelerinde yaklaşık 2 kat daha fazla gelişme sağlanmıştır.

RESIDUE NUMBER SYSTEM VARIANT OF LEARNING WITH ERRORS BASED TORUS FULLY HOMOMORPHİC ENCRYPTİON SCHEME

SUMMARY

Homomorphic encryption is a special type of encryption that allows performing homomorphic operations on encrypted data without decrypting and provides data security. Using the cloud system is one of the most important systems that needs to be developed for customer and/or patient confidentiality on behalf of all sectors. In cloud platforms, many encryption algorithms have been developed in the literature to ensure data security. There is a need to decode the data in order to process the data on the server side. For this reason, there are threats to ensuring data privacy. In addition, with the increasing use of IoT systems, the biggest problem in the areas where these systems are used has again been data protection and data privacy. Moreover, homomorphic encryption has become one of the most preferred encryption methods in recent times to ensure security in many of the artificial intelligence applications.

Thanks to homomorphic encryption, operations can be performed on encrypted data and there is no need to decrypt the data. However, there are disadvantages in terms of encryption times and resource usage. The long encryption time caused by homomorphic encryption is an obstacle to the practical use of the scheme. Therefore, studies aimed at developing and improving homomorphic schemes are of great importance. However, the creation of encryption schemes has remained theoretical due to the cost and computational size of these schemes. For this reason, many studies are being carried out to speed up encryption schemes and reduce costs.

Homomorphic encryption schemes are divided into three classes according to the allowed operations. If a cryptographic scheme allows for an unlimited number of only one operation (multiplication or addition), this scheme is called a partially homomorphic cryptographic (PHE) scheme. If an encryption scheme allows an unlimited number of one operation and allows a limited number of other operations, this scheme is called a somewhat homomorphic encryption (SwHE) scheme. If an encryption scheme allows an unlimited number of multiplication and addition operations, this scheme is called a fully homomorphic encryption (FHE) scheme (Özdemir and Koç, 2022).

FHE schemes can be divided into four basic categories based on problems. The first is the ideal lattice-based FHE scheme, presented by Gentry in 2009 (Gentry, 2009). In later years, Gentry's invention inspired other researchers and schemes based on the ideal lattice problem were developed. Secondly, it is a FHE scheme on integers that proposed by Van Djik et al. (Van Djik et al., 2010). Thirdly, it is a FHE scheme which based on learning with errors (LWE) problem offered by Oded Regev (Regev, 2010). Later, this problem was developed and ring learning with errors (RLWE) based

schemes were also obtained (Lyubashevsky et al., 2013). Finally, FHE schemes based on NTRU were created and developed (Hoffstein et al., 2006).

Basically, homomorphic encryption is a special type of encryption in which the ciphertext resulting from operations on ciphertext and the ciphertext resulting from the encryption of the result of the same operations performed on plaintexts are equal. From a historical point of view, the special homomorphisms used until 1978 paved the way for the use of partial homomorphic encryption for 30 years, but the most important development in this area was with the full homomorphic scheme developed by Gentry in 2009 (Gentry, 2009). Later, the DGHV (Djik-Gentry-Halevi-Vaikuntanathan) scheme (Van Djik et al., 2010) developed on integers in 2009, the BGV (Brakerski-Gentry-Vaikuntanathan) scheme based on learning problems developed in 2011 (Brakerski et al., 2014), the BFV (Brakerski-Fan-Vercauteren) (Fan and Vercauteren, 2012) study published in 2012 together with the GSW (Gentry-Sahai-Waters) scheme (Gentry et al., 2013) published in 2013, the FHEW (Fastest Homomorphic Encryption in the West) (Ducas and Micciancio, 2015) scheme introduced in 2014, the CKKS (Cheon-Kim-Kim Song) (Cheon et al., 2017) and TFHE (Torus Fully Homomorphic Encryption) (Chillotti et al., 2016) studies also introduced in 2016 have made great contributions to this field.

Recently, the full homomorphic encryption scheme based on torus, called TFHE, which proposed by Ilaria et al. (Chillotti et al., 2020), also holds promise for homomorphic encryption. Although many studies have been carried out to improve the scheme, we aim to minimize speed, time and memory problems by making the RNS variant of the scheme in this study. It has been observed that the RNS-CKKS scheme (Cheon at al., 2019), of which RNS variants were previously made, speeds up decoding, constant multiplication and homomorphic multiplication operations 17.3, 6.4, and 8.3 times faster, respectively, compared to the original scheme. Again, it has been obtained that the RNS-FV (Bajard et al., 2017) scheme increases from x5 speed to x20 speed for decryption operation and from x2 speed to x4 speed for multiplication operation between the dimensions 2^{11} and 2^{15} compared to the original FV (Fan-Vercauteren) scheme. Also RNS-BFV was studied (Halevi et al., 2017) which is RNS variant of the BFV scheme and good results were obtained.

With the CRT (Chinese Remainder Theorem) we used to create the RNS variant, we ensure that the torus structure used in the scheme is reduced to smaller modules. Even if T_q reduction was performed for $T \equiv \mathbb{R} \mod 1$ in the study of discretized torus (Joye, 2021) which conducted to develop the scheme, this reduction alone is not sufficient to apply CRT. We are planning to contribute to the shortcomings of the TFHE scheme in terms of speed, time and memory by working on the T_{q_i} structure in this thesis by reducing the ciphertexts of the TFHE scheme based on learning with errors (LWE) with the help of CRT.

In cryptology, homomorphic encryption (HE) is an encryption scheme that allows a cloud service provider to do special computional operations over data when it's encrypted (Chauhan et al., 2015). A user can not do operations on data in cloud. To do operations, firstly the data should be downloaded and decrypted. Else, private key should be shared with service provider which is not safe. Generally, while encryption schemes can't carry out operations firstly without decrypting the ciphertext,

homomorphic encryption allows carrying out operations on the ciphertext (Özdemir and Koç, 2022).

In this thesis, it is aimed to contribute to homomorphic encryption by applying the Residue Number System (RNS) variant, which improves schemes by applying it to many other homomorphic encryption schemes, to the TFHE (Torus Full Homomorphic Encryption) scheme, which has recently made a major breakthrough. In this context, it is aimed to reduce the computational load on the Torus Full Homomorphic Encryption (TFHE) algorithm, shorten processing times, and reduce resource usage.

In order to eliminate the disadvantages of homomorphic encryption in our scientific studies within the scope of the thesis, the Chinese Remainder Theorem (CRT) will be applied to the TFHE scheme, unlike the studies in the literature, and improvement will be provided in terms of encryption time. The difference between the CRT to be used and the TFHE components and the space in which it works distinguishes it from other homomorphic encryption schemes. The Chinese Remainder Theorem (CRT) method will be used to apply RNS to the scheme. By applying RNS to the scheme, about 2 times more improvement has been achieved in the encryption and decryption times of the scheme.

1. GİRİŞ

1.1. Literatür Taraması ve Tezin Amacı

Günümüzde homomorfik şifreleme, bulut (cloud) servislerinin güvenliği açısından kriptoloji dünyasında büyük önem arz etmektedir. Temel olarak homomorfik şifreleme, şifreli metinler üzerinde yapılan işlemler sonucunda ortaya çıkan şifreli metin ile, açık metinler üzerinde yapılan aynı işlemlerin sonucunun şifrelenmesiyle ortaya çıkan şifreli metnin eşit olduğu özel bir şifreleme türüdür. Tarihsel açıdan 1978'e kadar kullanılan özel homomorfizmalar devamında 30 yıllık kısmi homomorfik şifrelemenin kullanılmasına ön ayak olsa da bu alandaki en önemli gelişme 2009'da Gentry tarafından geliştirilen tam homomorfik şema ile olmuştur (Gentry, 2009). Daha sonrasında yine 2009'da tam sayılar üzerinde geliştirilen DGHV şeması (Van Djik ve ark., 2010), 2011'de geliştirilen hatalarla öğrenme problemine dayalı BGV şeması (Brakerski ve ark., 2011), 2012'de yayınlanan BFV (Fan ve Vercauteren, 2012) çalışmasıyla birlikte 2013'te yayınlanan GSW şeması (Gentry ve ark., 2013), 2014'te tanıtılan FHEW (Ducas ve Micciancio, 2015) şeması, 2016'da da tanıtılan CKKS (Cheon ve ark., 2017) ve TFHE (Chillotti ve ark., 2016) çalışmaları bu alana büyük katkılarda bulunmuşlardır.

Homomorfik şifreleme alanında yapılan çalışmalar şifreli metin üzerinde işlem yapılabilmesini kolaylaştırsa da bazı problemleri beraberinde getirmiştir. Bu problemlerin küçültülmesi adına yapılan çözümlerden biri de Gentry ve ark. ileri sürdüğü Çin Kalan Teoremi' ne (CRT) dayalı Kalan Sayılar Sistemi (RNS) adlı teknik ile basitçe büyük sayıların küçültülerek hafıza, zaman ve hız problemlerinin en aza indirgenmesi amaçlanmıştır (Gentry ve ark., 2012). Bu çalışmanın ardından RNS-FV (Bajard ve ark., 2017), RNS-BFV (Halevi ve ark., 2017) ve RNS-CKKS (Cheon ve ark., 2019) çalışmalarında olduğu gibi şemaların RNS varyantları çalışılmış ve iyi sonuçlar elde edilmiştir.

Son zamanlarda ise İlaria ve ark. öne sürdüğü TFHE adlı torus üzerinde tam homomorfik şifreleme şeması da homomorfik şifreleme için gelecek vadetmektedir (Chillotti ve ark., 2020). Şemayı geliştirmek adına birçok çalışma yapılsa da biz bu tezde şemanın RNS varyantını yaparak hız, zaman ve hafıza problemlerini en aza indirgemeyi amaçlıyoruz. RNS varyantını oluşturmak için kullandığımız CRT ile şemada kullanılan torus yapısının daha küçük modülolara indirgenmesini sağlamaktayız. Şemanın geliştirilmesi amacıyla yapılan ayrıklaştırılmış torus (Joye, 2021) adlı çalışmada torus yapısı için T_q indirgemesi yapılmış olsa bile bu indirgeme tek başına CRT uygulamak için yeterli değildir. Bu tezde T_{q_i} yapısı üzerinde çalışılarak TFHE şemasının hatalarla öğrenmeye dayalı şifreli metinlerinin (TLWE) CRT yardımıyla indirgenerek şemanın hız, zaman ve hafıza bakımından eksikliklerine katkıda bulunulmaktadır.

1.2. Tezin İçeriği

Bu tez çalışması dört bölümden oluşmaktadır. Giriş bölümünden sonra bazı cebirsel tanımlar ve kriptoloji ile ilgili temel tanımlar verilmiştir. Bölümün devamında homomorfik şifrelemeden bahsedilmiştir.

İkinci bölümde tezde kullanılan Torus Tam Homomorfik Şifreleme şeması tanıtılarak ilgili tanımlar verilmiştir. Bölümün devamında ise tezde kullanılacak olan yönteme katkıda bulunan ayrıklaştırılmış torus yapısı tanıtılarak ilgili tanımlar verilmiştir.

Üçüncü bölümde ise tezde kullanılacak olan Çin Kalan Yöntemi, Torus Tam Homomorfik Şifreleme şemasına uygulanarak şemanın şifreleme, deşifreleme fonksiyonlarında ve homomorfik işlemler aşamasında gerekli değişiklikler yapılarak şemanın Kalan Sayılar Sistemi varyantı elde edilmiştir. Elde edilen sonuçlar orijinal şema ile karşılaştırılarak sonuçların doğruluğu ispatlanmıştır.

Dördüncü bölümde Sonuçlar ve Değerlendirme verilerek tez çalışması tamamlanmıştır.

1.3. Temel Kavramlar

1.3.1. Cebirsel altyapı

Bu başlık altında bu tezde kullanılacak olan cebirsel altyapı verilecektir. İlgili tanım ve teoremler (Hungerford, 2012), (Çallıalp, 2009, 2011), (Aydın ve ark., 2012) ve (Ling, 2004) numaralı kaynaklar esas alınarak düzenlenmiştir.

Tanım 1.3.1.1 $k \neq 0$ bir tamsayı olmak üzere $x, y \in \mathbb{Z}$ için

$$x \equiv y \pmod{k} \Rightarrow k \mid x - y \tag{1.1}$$

şeklinde tanımlanır ve x ile y $\bmod k$ denktirler denir.

Tanım 1.3.1.2 $\mathbb{Z}$ deki $\equiv$ denklik bağıntısı ile belirtilen k modülüne göre denklik sınıflarına $\bmod k$ ya göre kalan sınıflar denir ve bu sınıflar $\mathbb{Z}_k$ ile gösterilir.

Tanım 1.3.1.3 (Grup) $K \neq \varnothing$ bir küme ve $\bullet$, K de bir ikili işlem olmak üzere $(K,\bullet)$ cebirsel yapısı

G1. $\bullet$, K de bir ikili işlemdir.
G2. $\bullet$ işleminin K kümesi üzerinde birleşme özelliği vardır. Yani $\forall k,l,m \in K$ için $k\bullet(l\bullet m) = (k\bullet l)\bullet m$ dir.
G3. $\bullet$ işleminin, K de birim elemanı vardır. Yani $\forall k \in K$ için $k\bullet e = e\bullet k = k$ olacak şekilde $\exists e \in K$ vardır.
G4. K deki her elemanın $\bullet$ işlemine göre bir tersi mevcuttur, yani $k \in K$ için $k\bullet k^{-1} = k^{-1}\bullet k = e$ olacak şekilde $\exists k^{-1} \in K$ bulunabilir.

aksiyomlarını sağlıyorsa $(K,\bullet)$ cebirsel yapısı bir gruptur.

Tanım 1.3.1.4 $(K,\bullet)$ bir grup olmak üzere $\forall k,l \in K$ için $k\bullet l = l\bullet k$ değişme özelliği sağlanıyorsa K grubuna Abel (değişmeli) grup denir.

Tanım 1.3.1.5 (Grubun Mertebesi) K sonlu bir küme olmak üzere $(K,\bullet)$ grubu sonlu gruptur ve grupların eleman sayılarına da grubun mertebesi denir.

Tanım 1.3.1.6 (Alt Grup) K bir grup ve L, K nın boştan farklı bir alt kümesi olsun. Eğer K daki işleme göre L, kendi başına bir grup oluşturuyorsa L ye, K grubunun bir alt grubu denir ve $L < K$ şeklinde gösterilir.

Önerme 1.3.1.7 K bir grup ve $\varnothing \neq L \subset K$ olmak üzere L bir alt gruptur ancak ve ancak

i. $\forall k,l \in L$ için $kl \in L$

ii. $\forall k \in L$ için $k^{-1} \in L$

şartları sağlanır.

Önerme 1.3.1.8 K bir grup olmak üzere boştan farklı alt kümesi olan L nin bir alt grup olması için gerek ve yeter koşul $\forall k,l \in L$ olmak üzere $kl^{-1} \in L$ olmasıdır.

Tanım 1.3.1.9 (Üreteç) $N \subset K$ olmak üzere K grubunun, N yi kapsayan bütün alt gruplarının arakesitine N kümesinin ürettiği alt grup denir ve $< N >$ ile gösterilir. N kümesinin elemanlarına da $< N >$ grubunun üreteçleri denir.

Tanım 1.3.1.10 (Devirli Alt Grup) K bir grup olmak üzere bir $N \subset K$ alt kümesi $K =< N >$ olacak şekilde mevcutsa K ya N ile üretilmiş grup denir. Eğer N sonlu ise K grubuna sonlu üretilmiş grup denir. Eğer $N = \{k\}$ ise K ya devirli grup denir ve $K =< k >$ ile gösterilir.

Tanım 1.3.1.11 (Halka) "$\oplus$" ve "$\otimes$" ikili işlemleri, $M \neq \varnothing$ kümesi üzerinde tanımlı olmak üzere

H1. $(M,\oplus)$ grubu değişmelidir.

H2."$\otimes$" işleminin M kümesi üzerinde birleşme özelliği vardır.

H3."$\otimes$" işleminin "$\oplus$" işlemi üzerine soldan ve sağdan dağılma özellikleri vardır: $\forall k,l,m \in M$ için $k \otimes (l \oplus m) = k \otimes l \oplus k \otimes m$ ve $(k \oplus l) \otimes m = k \otimes m \oplus l \otimes m$

aksiyomlarını sağlayan $(M,\oplus,\otimes)$ cebirsel yapısına bir halka denir.

Halkanın "$\oplus$" işlemine ve "$\otimes$" işlemine göre etkisiz elemanlarına sırasıyla halkanın sıfır elemanı ve birim elemanı denir. 0_M ve 1_M ile gösterilir. Bir halkanın birim elemanı olmayabilir. Eğer bir halkanın birim elemanı mevcutsa bu halkaya birimli

halka denir. Ayrıca bir halkaya değişmeli halka denmesi için ikinci işlemine göre değişme özelliğine sahip olmalıdır.

Tanım 1.3.1.12 M halkasında $0_M \neq k \in M$ elemanı için $kl = 0_M$ veya $lk = 0_M$ olacak şekildeki $\exists 0_M \neq l \in M$ bulunabiliyorsa k elemanına halkanın sıfır böleni denir.

Tanım 1.3.1.13 (Halkanın Karakteristiği) M bir halka olsun. Her $k \in M$ için $xk = 0$ olacak şekilde bir $x > 0$ tamsayısı varsa böyle tamsayılarının en küçüğüne M halkasının karakteristiği denir. Bu şekilde hiçbir $x > 0$ tamsayısı mevcut değilse M halkasının karakteristiği sıfırdır denir.

Tanım 1.3.1.14 (Alt Halka) M bir halka ve P, M nin boş olmayan bir alt kümesi olmak üzere M halkasının işlemlerine göre P kümesi kendi başına halka şartlarını sağlıyorsa P kümesine M halkasının bir alt halkasıdır denir.

Önerme 1.3.1.15 M bir halka ve $0 \neq P \subset M$ olmak üzere P kümesi M halkasının bir alt halkasıdır ancak ve ancak $\forall k, l \in P$ için $k - l \in P$ ve $kl \in P$ sağlanır.

Önerme 1.3.1.16 Bir halkaya ait bir takım alt halkaların arakesiti de bir alt halkadır.

Tanım 1.3.1.17 $A \subset M$ olsun. M halkasının A kümesini içeren tüm alt halkalarının ara kesitine A nın ürettiği alt halka denir ve $< A >$ şeklinde gösterilir. Ayrıca A kümesinin elemanlarına da $< A >$ nın üreteçleri denir.

Tanım 1.3.1.18 (İdeal) M bir halka ve S, M nin bir alt halkası olsun.

i. Her $m \in M$ ve $s \in S$ için $ms \in S$ oluyorsa S bir sol idealdir.
ii. Her $m \in M$ ve $s \in S$ için $sm \in S$ oluyorsa S bir sağ idealdir.
iii. S hem sol hem de sağ ideal oluyorsa S, M nin bir idealidir.

Teorem 1.3.1.19 M bir halka ve S, M nin boştan farklı bir alt kümesi olmak üzere S kümesinin M nin bir sağ ideali (sol ideali) olması için gerek ve yeter koşul;

i. Her $s, t \in S$ için $s - t \in S$ ve
ii. Her $s \in S$, $m \in M$ için $sm \in S$ ($ms \in S$) olmasıdır.

Sonuç 1.3.1.20 $\{A_j \mid j \in S\}$ kümesi M halkasının (sol) ideallerinin bir ailesi olmak üzere A_j (sol) ideallerinin kesişimi olan $\bigcap_{j \in I} A_j$ de M halkasının bir (sol) idealidir.

Tanım 1.3.1.21 R, M halkasının bir alt kümesi olmak üzere M halkasının R yi içeren tüm (sol) ideallerinin ailesi $\{A_j \mid j \in S\}$ olsun. $\bigcap_{j \in I} A_j$ kesişimine R kümesi tarafından üretilen (sol) ideal denir.

R tarafından üretilen ideal $< R >$ ile gösterilir ve elemanlarına da $< R >$ idealinin üreteçleri denir. Eğer $R = \{r_1, ..., r_n\}$ ise $< R >$ ideali sonlu üretilmiştir denir. Tek bir r elemanı tarafından üretilen ideal olan $< r >$ ise temel ideal olarak adlandırılır.

Teorem 1.3.1.22 M birimli bir halka ve $k \in M$ olmak üzere k tarafından üretilen sol ideal $Mk = \{mk \mid m \in M\}$ ve sağ ideal $kM = \{km \mid m \in M\}$ şeklindedir.

Teorem 1.3.1.23 S, M halkasının bir ideali olsun. $M / S = \{k + S \mid k \in R\}$ kümesi toplama işleminin $(k + S) + (l + S) = (k + l) + S$ olarak, çarpma işleminin ise $(k + S)(l + S) = (kl) + S$ olarak tanımlanması halinde halka belirtir.

Tanım 1.3.1.24 M bir halka ve S de onun bir ideali olmak üzere M / S halkasına bölüm halkası denir.

Teorem 1.3.1.25 (Çin Kalan Teoremi) M halkasının $S_1, S_2, ..., S_n$ idealleri için aşağıdaki özellikler sağlanıyor olsun.

i. $M^2 + S_i = M$, $\forall i \in \{1, ..., n\}$

ii. $S_i + S_j = M$, $\forall i \neq j$.

Bu durumda herhangi $k_1, k_2, ..., k_n \in M$ için

$$k \equiv k_i \pmod{S_i} \quad (i = 1, 2, ..., n) \tag{1.2}$$

olacak şekilde bir $k \in M$ elemanı vardır ve bu eleman $\bmod S_1 \cap S_2 \cap ... \cap S_n$ e göre tek türlü belirlidir.

Tanım 1.3.1.26 (Halka Homomorfizması) M ve N iki halka ve $f : M \to N$ bir fonksiyon olmak üzere $\forall k,l \in M$ için

i) $f(k+l) = f(k) + f(l)$

ii) $f(kl) = f(k)\, f(l)$

ise f homomorfizmasına M halkasından N halkasına bir halka homomorfizması denir.

Tanım 1.3.1.27 Bire-bir ve örten olan bir halka homomorfizmasına halka izomorfizması denir.

Tanım 1.3.1.28 C boş olmayan bir küme olmak üzere bu kümenin elemanları arasında "$+$" ve "$\cdot$" şeklinde ikili işlemler tanımlanmış olsun. $(C,+)$ ve $(C \setminus \{0\},\cdot)$ grupları birer değişmeli grup ise $(C,+,\cdot)$ üçlüsüne cisim denir.

Teorem 1.3.1.29 Bir cismin karakteristiği ya bir asal sayıdır ya da sıfırdır.

Teorem 1.3.1.30 m pozitif bir tamsayı olmak üzere karakteristiği r olan sonlu bir cismin eleman sayısı, r^m şeklindedir.

Not 1.3.1.31 $q = r^m$ elemanlı sonlu cisim kısaca C_q olarak gösterilir.

Tanım 1.3.1.32 $C_q = \{0, \alpha, \alpha^2, ..., \alpha^{q-1}\}$ olacak şekildeki α elemanına C_q cisminin ilkel (primitive) elemanı denir. Ayrıca ilkel eleman C_q cisminin çarpımsal grubunun üretecidir, yani $C_q^* = <\alpha>$, $C_q^* = C_q \setminus \{0\}$.

Tanım 1.3.1.33 $\alpha \in C_q$ sıfırdan farklı bir eleman olmak üzere $\alpha^t = 1$ olacak şekildeki en küçük pozitif t tamsayısına α nın mertebesi denir ve $ord(\alpha) = t$ olarak gösterilir.

Önerme 1.3.1.34

i. C_q cisminin sıfırdan farklı en az bir α elemanını ilkel elemandır ancak ve ancak mertebesi $ord(\alpha) = q-1$ dir.

ii. Her sonlu cismin en az bir ilkel elemanı vardır.

Tanım 1.3.1.35 M halka olsun.

$$M[x] := \left\{ \sum_{i=0}^{n} k_i x^i : k_i \in M, n \geq 0 \right\} \tag{1.3}$$

kümesi M üzerindeki polinomlar halkası olarak adlandırılmaktadır. $M[x]$ in elemanları M üzerindeki polinomlar olarak adlandırılmaktadır. $f(x) = \sum_{i=1}^{n} k_i x^i$ polinomu için n tamsayısı eğer $a_n \neq 0$ ise $f(x)$ in derecesi olarak adlandırılmaktadır ve $\deg(f(x))$ ile gösterilmektedir. Eğer $a_n = 1$ ise n dereceli $f(x) = \sum_{i=1}^{n} k_i x^i$ polinomuna monik polinom denmektedir. Eğer $\deg(g(x)) < \deg(f(x))$, $\deg(h(x)) < \deg(f(x))$ ve $f(x) = g(x)h(x)$ olacak şekilde M üzerinde $g(x)$ ve $h(x)$ polinomları mevcut ise sıfırdan farklı pozitif dereceli $f(x)$ polinomuna indirgenebilir polinom denmektedir. Eğer yoksa sıfırdan farklı pozitif dereceli $f(x)$ polinomuna indirgenemez polinom denmektedir.

Tanım 1.3.1.36 M bir halka ve $(K,+)$ bir değişmeli grup olsun. K daki elemanların M deki elemanlarla skaler çarpımı olan $M \times K \to K$ fonksiyonu;

i. Her $m \in M$ ve her $k, k' \in K$ için $m(k+k') = mk + mk'$

ii. Her $m, m' \in M$ ve her $k \in K$ için $(m+m')k = mk + m'k$

iii. Her $m, m' \in M$ ve her $k \in K$ için $(mm')k = m(m'k)$

koşullarını sağlıyorsa K ya M üzerinde bir sol modül veya kısaca sol M -modül denir.

Not 1.3.1.37 Tanım 1.3.1.36'daki işlemler sağ taraftan tanımlandığında K bir sağ M -modüldür.

Önerme 1.3.1.38 M -modül K nin boş olmayan bir $L \subseteq K$ alt kümesinin alt modül olması için gerek ve yeter koşul her $m, m' \in M$ ve her $k, k' \in L$ için $mk + m'k' \in L$ olmasıdır.

Tanım 1.3.1.39 K bir M -modül ve $k \in K$ olsun. k elemanının ürettiği alt modül

$$< k >= Mk = \{mk \mid m \in M\} \tag{1.4}$$

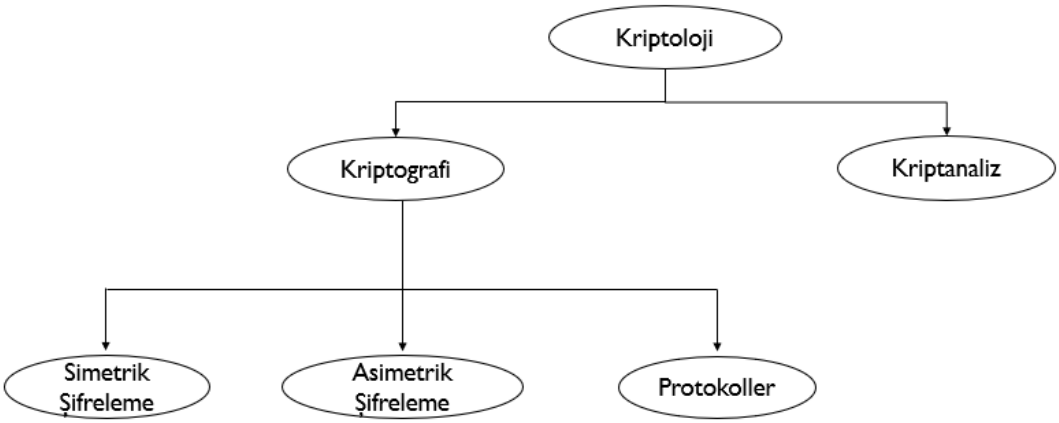
şeklinde tanımlanır. Eğer $K =< k >$ olacak şekilde bir $k \in K$ bulunabilirse K ye devirli modül denir.

1.3.2. Kriptoloji

Bu başlık altında bu tezde kullanılacak olan kriptolojik altyapı verilecektir. İlgili tanım ve teoremler (Paar, 2010), (Chauhan ve ark., 2015) ve (Özdemir ve Koç, 2022) numaralı kaynaklar esas alınarak düzenlenmiştir.

1.3.2.1. Kriptolojiye giriş

Kriptoloji, dijital bilgiyi koruma bilimidir. Yani kriptoloji, haberleşen iki veya daha fazla tarafın aralarındaki bilgi alışverişinin güvenli bir şekilde yürütülmesini sağlayan, temeli zor matematiksel problemlere dayanan teknik ve uygulama bütünüdür.



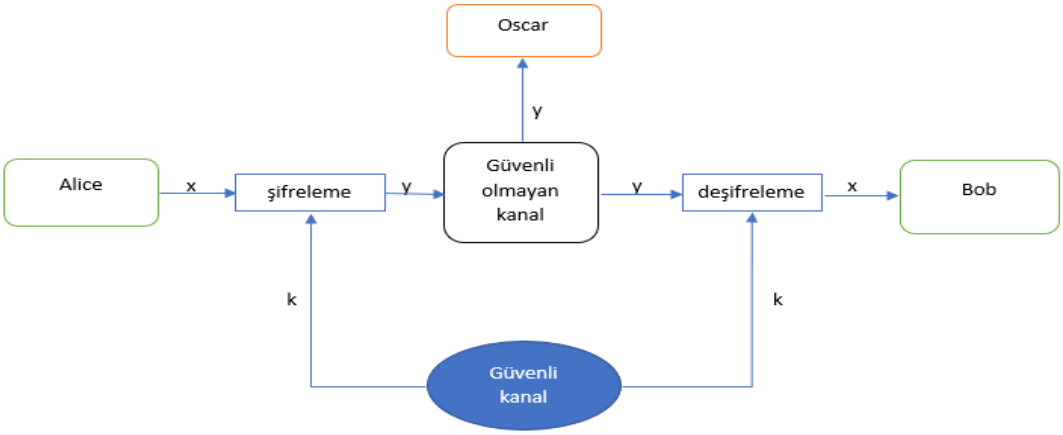
Şekil 1.1. Kriptoloji Alt Alanları

Kriptoloji iki temel kısma ayrılır:

Kriptanaliz, kripto sistemleri kırma bilimidir. Kripto sistemlerin güvenilirliğini test etmek için uygulanan yöntemler kriptanaliz altında birleşir.

Kriptografi, bir mesajın anlamını saklama amacıyla gizli yazma bilimidir. Kendi içinde simetrik şifreleme, asimetrik şifreleme ve protokoller olmak üzere üçe ayrılır.

Simetrik şifreleme tarafların aynı anahtarı kullandığı şifreleme yöntemidir. Şekil 1.1 de gösterildiği gibi bir simetrik şifreleme şemasında farklı iki taraf (Alice ve Bob) güvenli olmayan kanal üzerinden bilgi aktarımında bulunmak istendiğinde üçüncü bir kişinin (Oscar) bu bilgiye erişimini engellemek amacıyla gizli bir anahtar ile şifreleme yapılır.



Şekil 1.2. Temel Şifreleme Modeli

Şekil 1.2 de verilen x, açık metni (plaintext); y, şifreli metni (ciphertext) ve k da gizli anahtarı (secret/private key) temsil etmektedir. x açık metnindeki bilgilerin Oscar tarafından ele geçirilmesini engellemek amacıyla açık metin, k gizli anahtarı ile birlikte şifrelemeye (encryption) sokulur ve ardından güvenli olmayan kanal üzerinden Bob'a iletilir. Şifreli mesajı alan Bob, şifrelemenin tam tersi olan deşifreleme (decryption) işlemine hem şifreli mesajı hem de gizli anahtarı sokarak açık metin olan x i güvenli bir şekilde elde etmiş olur. Eğer bu mesaj Oscar tarafından ele geçirilmek istenirse açık metin Oscar'ın bilmediği bir anahtarla şifrelenmiş olduğundan ele geçirdiği şifreli mesaj açık mesajın anlamına dair herhangi bir bilgi taşımaz. Bu sayede mesajın güvenli bir şekilde iletimi gerçekleşmiş olur.

Simetrik şifreleme (symmetric encryption) kriptolojinin başladığı yıllardan günümüze kadar aktif olarak kullanılmaktadır. İlk olarak Yerine koyma Şifreleme (Substitution Cipher) ile başlayan bu süreç devamında Sezar Şifreleme (Ceaser/Shift Cipher), Afin Şifreleme (Affine Cipher) ile devam etmiştir. Günümüzde ise Blok Şifreleme (Block Cipher) ve Akış/Dizi Şifreleme (Stream Cipher) olarak ikiye ayrılan simetrik şifreleme büyük bir ilerleme kaydetmiştir. Bu ilerleme ise Ulusal Standartlar Dairesi (NBS: National Bureau of Standards şimdiki adıyla NIST: National Institute of Standards and

Technology); 1973'te "ulusal bir standart olabilecek kriptografik bir algoritma" talebinde bulunmasıyla başlamıştır. Bu talebe yönelik IBM'nin (International Business Machines: Uluslararası İşletme Makineleri) 1974'te sunduğu LUCIFER algoritması, Ulusal Güvenlik Ajansı'nın (NSA – National Security Agency) uyguladığı değişikliklerin ardından 1977'de DES (Data Encryption Standard: Veri Şifreleme Standardı) adıyla resmi bilgi şifreleme standardı olarak kabul edildi. Fakat DES'e uygulanan ataklar dolayısıyla bu şifrelemenin güvenliği kırıldı ve 2000 yılında yerini AES (Advanced Encryption Standard: Gelişmiş Şifreleme Standardı) şemasına bıraktı. Simetrik şifreleme şemaları halen büyük ölçüde çalışılan bir alan olmaya devam etmektedir.

Asimetrik şifreleme (asymmetric encryption/ public key cryptography) simetrik şifrelemedeki anahtar paylaşımındaki problemler sonucu bu problemlere çözüm olarak ortaya çıkmıştır. Asimetrik şifrelemede mesajı şifrelemek ve deşifrelemek için iki farklı anahtar kullanılmaktadır. Bunun güvenliğinin sağlanması açısından asimetrik şifrelemede zor matematik problemleri temel alınarak algoritmalar oluşturulmuştur.

1. Bunlardan ilki çarpanlarına ayırma şemalarıdır. Birkaç şema büyük tamsayıların çarpanlarına ayrılmasının zorluğuna dayanarak oluşturulmuştur. Bu algoritmanın en önemli temsilcisi 1978'de Ron Rivest, Adi Shamir ve Leonard Adleman tarafından geliştirilen RSA algoritmasıdır.

2. İkinci şema ise ayrık logaritma şemalarıdır (discrete logarithm schemes). Sonlu cisimlerde ayrık logaritma problemi olarak bilinen probleme dayalı algoritmaların en önde gelenleri 1976 yılında Whitfield Diffie ve Martin Hellman tarafından geliştirilen Diffie-Hellman anahtar değişimi (Diffie-Hellman key exchange), Taher Elgamal tarafından 1984 yılında önerilen Elgamal şifreleme ve 1991'de National Institute of Standards and Technology (NIST) tarafından tasarlanan DSA (Digital Signature Algoritm: Dijital İmza Algoritması)'dır.

3. Sonuncusu ise ayrık logaritma algoritmalarının bir genelleştirilmesi olan eliptik eğri (EC: Elliptic Curve) şemalarıdır. En bilinen örnekleri Eliptik Eğri Diffie-Hellman anahtar değişimi (Elliptic Curve Diffie–Hellman key exchange (ECDH)) ve Eliptik Eğri Dijital İmza Algoritması (Elliptic Curve Digital Signature Algorithm (ECDSA))'dır.

Kriptografik protokoller, kriptografik algoritmaların uygulamaları ile ilgilenir. Simetrik ve asimetrik algoritmalar yapıtaşları olarak görülebilir. Protokoller ise Hash (Özet) fonksiyonları gibi algoritmalar ile simetrik şifreleme özelliği taşıyan üçüncü bir algoritma sınıfı olarak düşünülebilir. Kriptografik uygulamaların çoğunda simetrik ve asimetrik algoritmalar (ve Hash fonksiyonları) birlikte kullanılmaktadır. Bu sisteme hibrit (hybrid) şemalar da denmektedir. Bu şemaların kullanım sebebi hem simetrik hem de asimetrik algoritmaların kendine özgü avantajları ve dezavantajları olmasıdır.

1.3.2.2. Homomorfik şifreleme

Kriptoloji tarihi boyunca şifreleme ve deşifreleme işlemlerinin yanında bilginin gizliliğini bozmadan bilgi üzerinde işlemler de yapılabilmesinin gerektiği yerler doğmuştur. Bu gereksinim üzerine Homomorfik Şifreleme (Homomorphic Encryption: HE) yöntemi ilk olarak 1977 yılında Rivest, Adleman ve Derouzous tarafından gündeme getirilmiştir. Kriptolojide homomorfik şifreleme bir bulut (cloud) servis sağlayıcısına veri üzerinde şifreliyken özel hesapsal fonksiyonlar yürütmesine izin veren bir şifreleme şemasıdır (Chauhan ve ark., 2015; Xu ve ark., 2022).

Bir kullanıcı buluttaki veri üzerinde hesaplamalar yapamaz. Hesaplamaları yapabilmek için veri indirilmeli ve deşifre edilmelidir ya da gizli anahtar servis sağlayıcı ile paylaşılmalıdır. Geleneksel şifreleme şemaları şifreli veriyi önce deşifre etmeksizin üzerinde işlemler gerçekleştiremezken homomorfik şifreleme, bulut sağlayıcılarının şifreli veriler üzerinde, veriyi deşifrelemeksizin üzerinde hesaplamalar yapmasına izin verir (Mattila ve ark., 2022).

Homomorfik şifreleme şifreli metinler üzerinde hesaplama yapma kabiliyeti olan ve açık mesajların üzerinde de yerine getirilen aynı hesaplamaların sonuçlarının aynı olduğu özel bir şifreleme türüdür. '$*$' işlemi üzerinde E şifreleme algoritmalı bir homomorfik şifreleme şeması M açık mesaj uzayı olmak üzere $\forall m_1, m_2 \in M$ için Denklem 1.5' i sağlar.

$$E(m_1) * E(m_2) = E(m_1 * m_2) \tag{1.5}$$

Homomorfik şifreleme şeması dört temel algoritmaya sahiptir:

1. Anahtar Üretim Algoritması (KeyGen): Girdi olarak λ güvenlik parametresini alır. Asimetrik homomorfik şifreleme şeması gizli anahtar ve açık anahtar

çiftini çıktı verirken simetrik homomorfik şifreleme şeması tek bir anahtar çıktısı verir.

2. Şifreleme Algoritması (Enc): Şifreleme algoritması, anahtar ile birlikte M mesaj uzayından m açık mesajını girdi olarak alır. C şifreli mesaj uzayından $c = E(m)$ şifreli mesajını üretir.

3. Deşifreleme Algoritması (Dec): Deşifreleme algoritması, anahtar ile birlikte c şifreli mesajını girdi olarak alır ve $D(c) = m$ açık mesajını elde eder.

4. Hesaplama (Evaluation) Algoritması (Eval): (c_1, c_2) şifreli mesajlarını girdi olarak alır ve (m_1, m_2) açık mesajlarını bilmeksizin $f(c_1, c_2) = E(f(m_1, m_2))$ hesaplanmış şifreli mesajları çıktı verebilmek için şifreli mesajlar üzerinde f fonksiyonunu yerine getirir, yani

$$D(f(c_1, c_2)) = f(m_1, m_2) \tag{1.6}$$

Homomorfik şifreleme şemaları izin verilen işlemlere göre üç sınıfa ayrılmaktadır. Bunlar Kısmi Homomorfik Şifreleme (Partially Homomorphic Encryption: PHE), Yarı Homomorfik Şifreleme (Somewhat Homomorphic Encryption: SwHE) ve Tam Homomorfik Şifreleme (Fully Homomorphic Encryption: FHE) şemalarıdır.

Kısmi homomorfik şifreleme (PHE)

Eğer bir şifreleme şeması sınırsız sayıda sadece tek bir işleme izin veriyorsa bu şema Kısmi Homomorfik Şifreleme (PHE) şeması olarak adlandırılmaktadır. Kısmi homomorfik şifreleme şemaları ya sadece toplama işlemlerini destekleyen toplamsal (additive) homomorfik şema ya da sadece çarpımsal işlemleri destekleyen çarpımsal (multiplicative) homomorfik şemalardır. PHE şemalarının önemli örnekleri RSA, Goldwasser-Micali, El-gamal, Benaloh ve Paillier şemalarıdır.

Yarı homomorfik şifreleme (SwHE)

Eğer bir şifreleme şeması bir işleme sınırsız sayıda izin verirken diğer işleme sınırlı sayıda izin veriyorsa bu şemaya Yarı Homomorfik Şifreleme (SwHE) şeması

denmektedir. Bu sınırlama şemanın şifreli mesajlarının homomorfik işlemler ile birlikte doğru bir şekilde deşifrelenebilme kabiliyetiyle tanımlanmaktadır.

Genel olarak, yarı homomorfik şifreleme şemasının şifreli mesajı gürültü (noise) parametresine sahiptir ve düzgün bir şekilde deşifre edebilmek için gürültü, spesifik bir limitten daha az olmalıdır. Bir SwHE şeması şifreli mesajlar üzerinde hem toplamsal hem de çarpımsal işlemler gerçekleştirebilir fakat bu şifreli mesajdaki gürültüyü her bir işlemle arttırır. Dolayısıyla gürültüyü küçük tutabilmek amacıyla SwHE şemaları homomorfik işlemleri sınırlı sayıda gerçekleştirebilir. SwHE şemalarının en önemli örneği Boneh-Goh-Nissim (BGN) şemasıdır.

Tam homomorfik şifreleme (FHE)

Eğer bir şifreleme şeması çarpma ve toplama işlemlerine sınırsız sayıda izin veriyorsa bu şemaya Tam Homomorfik Şifreleme (FHE) şeması denmektedir. Bu şemalar problemlere dayalı dört temel kategoriye ayrılabilir:

1) İdeal-Kafes Tabanlı (Ideal-Lattice Based) FHE
2) Tamsayılar Tabanlı (Over the Integers Based) FHE
3) (Halkalarda) Hatalarla Öğrenme Tabanlı ((Ring) Learning With Errors Based: (R)LWE) FHE
4) NTRU_Benzeri Tabanlı (NTRU_like Based) FHE

Bu kategorilerden ilki 2009 yılında Gentry tarafından sunulan ideal kafes tabanlı FHE şemasıdır (Gentry, 2009). Daha sonraki yıllarda Gentry'nin bu buluşu başka araştırmacılara da ilham olmuş ve ideal kafes problemine dayalı şemalar geliştirilmiştir. İkinci olarak Van Djik ve ark. (Van Djik ve ark., 2010) sunduğu tamsayılar üzerinde FHE şemasıdır. Üçüncü olarak Oded Regev'in sunduğu hatalarla öğrenme (LWE) (Regev, 2010) tabanlı FHE şemasıdır. Daha sonrasında bu problem geliştirilerek halkalarda hatalarla öğrenme (RLWE) tabanlı şemalar da elde edilmiştir (Lyubashevsky ve ark., 2013). Son olarak ise NTRU tabanlı FHE şemaları oluşturulmuş ve geliştirilmiştir (Hoffstein ve ark., 2006).

Tanım 1.3.2.2.1 (Olasılık Dağılımları) Pek çok FHE şeması açık mesajı Gauss gürültüsü (Gaussian noise) ile gizler. Dolayısıyla bu gürültü daima standart sapma veya varyans ile ölçülür. Bir Gauss dağılımının varyansı, boyuta bölünen ortalama

kare normuna eşittir, bu nedenle bu dağılımı kullanmak doğal olan yayılma formülüne yol açar. En önemlisi, somut uygulamalarda genellikle bir kafa karışıklığı kaynağı olan ek $\sqrt{2p}$ faktörlerinden (gürültü parametresiyle ilgili) kaçınır (Chillotti ve ark., 2020).

Tanım 1.3.2.2.2 (Gauss Dağılımları) $k \geq 1$ ve $\sigma \in \mathbb{R}^+$ olsun. Her $x, c \in \mathbb{R}^k$ için c merkezli ve σ standart sapmalı Gauss fonksiyonu $\rho_{\sigma,c}(x) = \exp\left(-\|x-c\|^2 / 2\sigma^2\right)$ ile gösterilmektedir. Eğer c ihmal edilirse dolaylı olarak 0 olarak alınır. S, $\mathbb{R}^k$ nın alt kümesi olmak üzere $\rho_{\sigma,c}(S)$, eğer S ayrık (discrete) ise $\sum_{x \in S} \rho_{\sigma,c}(x)$ olarak ya da S ölçülebilir ise $\int_{x \in S} \rho_{\sigma,c}(x)$ olarak gösterilmektedir (Chillotti ve ark., 2020).

Tüm (sürekli veya ayrık) $M \subseteq \mathbb{R}^k$ toplamsal alt grubu için $\rho_{\sigma,c}(M)$ sonludur ve M üzerindeki c merkezi ve σ standart sapması için yoğunluk fonksiyonu $D_{M,\sigma,c}(x) = \rho_{\sigma,c}(x) / \rho_{\sigma,c}(M)$ olan $D_{M,\sigma,c}$ (sınırlandırılmış) Gauss Dağılımını (Gaussian Distribution) tanımlar. L, M nin ayrık alt grubu olmak üzere M/L üzerinde $D_{M/L,\sigma,c}$ Modüler Gauss Dağılımı (Modular Gaussian Distribution) mevcuttur ve $D_{M/L,\sigma,c}(x) = D_{M,\sigma,c}(x+L)$ yoğunluk fonksiyonu ile tanımlanmaktadır (Chillotti ve ark., 2020).

Tanım 1.3.2.2.3 (Alt-Gauss Dağılımları) χ, $\mathbb{R}$ üzerinde alt-Gauss dağılımıdır ancak ve ancak χ Laplace dönüşüm sınırını (Laplace transformation bound) sağlar. Yani her $t \in \mathbb{R}$ için olasılıklar $E\left(\exp(tX)\right) \leq \exp\left(\sigma^2 t^2 / 2\right)$ eşitsizliğini sağlar (Chillotti ve ark., 2020).

Tanım 1.3.2.2.4 (Önyükleme: Bootstrapping) Kendi deşifreleme algoritma devresini hesaplayabilen şemaya önyüklenebilir (bootstrappable) denmektedir. Önyükleme temel olarak şifreli mesajlar üzerinde homomorfik işlemler gerçekleştirdikten sonra şifreli mesajın gürültüsünü azaltan bir prosedürdür. Ancak önyükleme tekniği hesaplama maliyetini arttırır, bu da şemayı gerçek hayatta kullanışsız yapar.

İlk kez Craig Gentry tarafından 2009 yılında tanıtılan FHE şeması birçok araştırmaya ilham verici olsa da hesapsal maliyetinden dolayı kullanımı zordur. Dolayısıyla bu şemaların uygulamalarını daha kullanılabilir yapmak için pek çok iyileştirme yapılmıştır (Xu ve ark., 2022; Gentry ve Halevi, 2011). Gentry'nin çalışmasını yayınlamasından beri FHE' nin gelişimi, şemalarda kullanılan tekniklere göre dört jenerasyona ayrılabilir.

1) Birinci Jenerasyon

   Gentry'nin sunduğu şema SwHE şemasından FHE şemasının nasıl elde edileceğini gösteren ilk FHE çalışmasıdır (Gentry, 2009). SwHE şemasının güvenliğini ideal kafesler üzerindeki ortalama durum karar/karmaşıklık (avarage case decision/complexity: oluşturulan algoritma tarafından kullanılan bazı hesaplama kaynaklarının -tipik olarak zaman- miktarıdır.) probleminin zorluğuna dayandırılmaktadır. Yine 2009'da Van Djik ve ark., Gentry'nin fikrine dayalı fakat basit tamsayı aritmetik işlemlerinin ideal kafes işlemlerinin yerini aldığı ikinci bir FHE şeması olan DGHV şemasını sundular (Van Djik ve ark., 2010). Daha sonrasında Smart ve Vercauteren, Gentry şemasının nispeten daha küçük anahtar ve şifreli mesaj boyutu kullanan üçüncü bir varyantını tanıttılar (Smart ve Vercauteren, 2010). Daha sonrasında ise 2011'de Gentry ve Halevi ideal kafeslere ve önyüklemeye (bootstrapping) dayanan bir şema geliştirdiler (Gentry ve Halevi, 2011). Bu ilk jenerasyon şemaları çok hızlı gürültü büyümesinden dolayı sınırlı homomorfik kapasiteye sahiptirler.

2) İkinci Jenerasyon

   İkinci jenerasyon 2011'de Brakerski ve Vaikuntanathan'ın BV adlı çalışması ile başladı (Brakerski ve Vaikuntanathan, 2011). Homomorfik çarpmalarda şifreli mesaj boyutunu kontrol etmek için "yeniden lineerleştirme" (re-linearization) tekniğini tanıttılar. Dahası boyut-modül indirgemesi (dimension-modulus reduction) adlı deşifreleme algoritmasını basitleştirmek için yeni bir metot kullanarak önyüklenebilir (bootstrappable) şemanın nasıl yapılandırılacağını gösterdiler. BV şemasının güvenliği yalnızca Regev tarafından tanıtılan "gürültü ile benzerlik öğrenme" (learning parity with noise) probleminin bir genelleştirmesi olarak tanıtılan "hatalarla öğrenme" (Learning With Errors: LWE) probleminin zorluğuna dayanmaktadır (Regev, 2010).

Sonrasında Brakerski, Gentry ve Vaikuntanathan yeni teknikler geliştirerek BGV adlı "sınırlı-FHE" (leveled-FHE) şemasını tanıttılar (Brakerski ve ark., 2014). Sınırlı-FHE, parametrelerin şemanın hesaplama kabiliyetinin olduğu devrelerin derinliğine bağlı olduğu FHE şemasıdır. Burada bahsedilen derinlik, şifreli mesajlar üzerinde yapılan ardışık çarpmaların maksimum sayısı olan çarpımsal derinliktir. BGV' de yeniden lineerleştirme ve boyut-modül indirgeme teknikleri anahtar değişimi (key switching) ve modül değişimi (modulud switching) teknikleri olarak geliştirildi. BGV şemasının güvenliği "halkalarda hatalarla öğrenme" (RLWE: Ring Learning With Errors) problemine dayanmaktadır. BGV şemasından sonra Brakerski, modül değişimi olmaksızın yeni bir FHE şeması tanıttı (Brakerski, 2012). Önceki LWE tabanlı FHE şemalarına kıyasla burada, şifreli mesajdaki gürültü homomorfik işlemlerle üsselden ziyade sadece lineer olarak büyümektedir. 2012'de Lopez-Alt, Tromer ve Vaikuntanathan tarafından umut verici etkililiği ve standartlaştırma özellikleriyle NTRU (Hoffstein ve ark., 2006) tabanlı çok anahtarlı FHE (NTRU based multikey FHE) şeması olan LTV şeması tanıtıldı (Asharov ve ark, 2012). Ancak, homomorfik işlemlere izin vermek ve güvenliği sağlamak için bu şemada da standart olmayan bir varsayım gerekliydi. İkinci jenerasyon FHE şemalarında gürültü büyümesi ilk jenerasyon FHE şemalarına kıyasla homomorfik hesaplamalar boyunca daha yavaştır. Dahası ikinci jenerasyon, ilk SwHE şemasını oluşturmaları ve daha sonrasında önyükleme (bootstrapping) kullanarak FHE şemasına dönüştürme bakımından Gentry'nin tasarımını takip etse de önyükleme olmaksızın sınırlı-FHE modunda çalıştırılabilirler ve bu onları daha etkili yapar. Ancak anahtar değişimi veya modül değişiminin karmaşık süreci pratiklik için engel teşkil eden yüksek hesaplama maliyetini gerektirir.

3) Üçüncü Jenerasyon

2013'te Gentry, Sahai ve Wators, GSW olarak biline ve maliyetli yeniden lineerleştirme tekniği yerine "yaklaşık özvektör" (approximate eigenvector) metodu kullanan yeni bir LWE tabanlı FHE şeması ileri sürdüler (Gentry ve ark., 2013). GSW'nin şifreli mesajları doğal yollarla homomorfik olarak toplanan ve çarpılan matrisler olduğu için şifreli mesaj boyutu sabit tutulmaktadır. GSW şeması önceki LWE tabanlı FHE şemalarından daha

basittir ve asimptotik olarak daha hızlıdır. Bir sonraki yıl FHEW (Ducas ve Micciancio, 2015) ve TFHE (Chillotti ve ark., 2016) olarak bilinen GSW kriptosistemin iki etkili varyantı, sırasıyla Ducas ve Micciancio tarafından ve İlaria ve ark. tarafından tanıtıldı.

4) Dördüncü Jenerasyon

Günlük yaşam uygulamalarının çoğunluğu $\mathbb{R}$ ve $\mathbb{C}$ gibi sürekli uzaylarda hesaplamalar gerektirir. Bundan dolayı Cheon ve ark. "yaklaşık sayılar" (approximate numbers) üzerinde işlemler yerine getirmek için doğal ortam sağlayan CKKS şemasını sundular (Cheon ve ark., 2017). CKKS, tahminler yürütmek ve makine öğrenmesi (machine learning) metotları için uygundur. Şemanın ismi ilk olarak HEAAN olsa da aynı isimli homomorfik şifreleme kütüphanesinden ayırt etmek için şemanın ismi CKKS olarak değiştirildi. CKKS'nin tanıtımından sonra şemanın Kalan Sayılar Sistemi (RNS) varyantı tanıtıldı (Cheon ve ark., 2019).

1.3.3. Çin Kalan Teoremi gösterimi (CRT representation)

Çin Kalan Teoreminin gösterimi Bölüm 3'te verilen torus yapısının indirgenmesi ve Torus Tam Homomorfik Şifreleme şemasının Kalan Sayılar Sistemi varyantının elde edilmesi için kullanılacaktır.

Tanım 1.3.3.1 $n \geq 2$ tam sayısı için $\mathbb{Z}_n$ halkası $\mathbb{Z} \cap [-n/2, n/2)$ simetrik aralığındaki temsili ile tanımlanmaktadır. Rastgele x reel sayısı için $[x]_n$ ile x in bu aralığa indirgenmesi gösterilmektedir. $q_1, q_2, ..., q_k$ $(q_i > 1, 1 \leq i \leq k)$ aralarında asal modüller olmak üzere $q = \prod_{i=1}^{k} q_i$ olsun. $i \in \{1, ..., k\}$ için $q_i^* = \frac{q}{q_i} \in \mathbb{Z}$ ve $\tilde{q}_i = (q_i^*)^{-1} \pmod{q_i}$ $\in \mathbb{Z}_{q_i}$ olur. Yani $\tilde{q}_i = \left[\frac{-q_i}{2}, \frac{q_i}{2} \right)$ ve $q_i^* . \tilde{q}_i = 1 \pmod{q_i}$ elde edilir. $x \in \mathbb{Z}_q$ tam sayısının $x \sim (x_1, ..., x_k)$ ile $\{q_1, ..., q_k\}$ CRT tabanına ilişkin CRT gösterimi $x_i = [x]_{q_i} \in \mathbb{Z}_{q_i}$ ile gösterilir. x i x_i lerle ifade etmenin formülü $x = \sum_{i=1}^{k} x_i \tilde{q}_i q_i^* \pmod{q}$ şeklindedir (Halevi ve ark., 2017).

$$x = \left(\sum_{i=1}^{k} \left[x_i \tilde{q}_i \right]_{q_i} q_i^* \right) - vq, \;\; v \in \mathbb{Z} \tag{1.7}$$

$$x = \left(\sum_{i=1}^{k} x_i \tilde{q}_i q_i^* \right) - v'q, \;\; v' \in \mathbb{Z} \tag{1.8}$$

2. TORUS TAM HOMOMORFİK ŞİFRELEME (TFHE)

Bu tezde incelediğimiz FHE şeması olan TFHE şeması ilgili notasyon ve tanımlar bu bölümde verilmektedir.

Torus Tam Homomorfik Şifreleme (Torus Fully Homomorphic Encryption: TFHE) şeması ilk olarak FHEW şemasının bir iyileştirmesi olarak Chillotti, Gama, Georgieva ve Izabachene tarafından 2016 yılında sunulmuştur. Daha sonrasında şema daha geniş bir yöne doğru gelişmeye başlamıştır. Şemanın güvenliği LWE problemine ve onun varyantı olan RLWE problemine dayanmaktadır. Aslında, günümüzde birçok FHE şeması LWE tabanlıdır ve gürültülü şifreli mesajlar kullanmaktadır. Fakat TFHE şeması hızlı ve paralel işlem hesaplama kabiliyetleri olan özel bir önyükleme yapısına sahip olmasıyla diğerlerinden ayrılmaktadır.

TFHE de temel olarak üç farklı şifreli mesaj tipi kullanılmaktadır. Bunlar LWE, RLWE ve RGSW şifreli mesajlarıdır. Üç farklı şifreli mesaj türünün olmasının sebebi ise her birinin homomorfik işlemlerde kullanışlı olan farklı özelliklerinin bulunmasıdır.

Tanım 2.1 Güvenlik parametresi λ olsun. B ile $\{0,1\}$ kümesi gösterilsin. T ile de reel torus yani $\mathbb{R}/\mathbb{Z}$, $\bmod 1$ e göre reel sayılar kümesi gösterilsin. $\mathbb{Z}_N[X]$ ile $\mathbb{Z}[X]/(X^N+1)$ polinomlar halkası gösterilmek üzere $T_N[X]$, $\mathbb{R}[X]/(X^N+1) \bmod 1$ i ve $B_N[X]$ de , $\mathbb{Z}_N[X]$ deki ikili (binary) katsayılı polinomları temsil eder. E^p ile E deki p boyutlu vektörler kümesi ve $M_{p,q}(E)$ ile de E deki elemanlarla oluşturulan $p \times q$ boyutlu matrislerin kümesi temsil edilmektedir (Chillotti ve ark., 2020).

Bu tezde $\mathbb{Z}$-modül olan T torus kümesi kullanılacaktır. T, $\bmod 1$ izdüşümü asıl çarpım ile uyumlu olmadığı için bir halka değildir. Örneğin, T den aldığımız 0 ve $\frac{1}{2}$ elemanlarının çarpımı olan $0 \times \frac{1}{2}$ sonucu T de tanımsızdır. Bunun yerine $\mathbb{Z}$ deki ve

T deki elemanların dış çarpımı (‘$\cdot$’) ise tanımlıdır. Örneğin $\mathbb{Z}$ den aldığımız 0 ve T den aldığımız $\frac{1}{2}$ elemanlarının dış çarpımı olan $0\cdot\frac{1}{2}$ işleminin sonucu $0\in T$ ye eşittir. Daha da önemlisi pozitif N ve k tamsayıları için $\left(T_N\left[X\right]^k,+,\cdot\right)$ bir $\mathbb{Z}_N\left[X\right]$-modüldür (Chillotti ve ark., 2020).

2.1. Hatalarla Öğrenme Problemi

Tanım 2.1.1 (Torus Üzerindeki Dağılımlar) Genel olarak torus üzerindeki dağılımların olasılıkları ya da varyansları yoktur. Örneğin, T üzerinde düzgün dağılımın olasılığını tanımlamak imkansızdır. Ancak, dağılımın desteği (support) küçük bir aralığa yoğunlaştırıldığında benzersiz notasyonlar tanımlamak mümkündür. Torus üzerindeki bir χ dağılımı yoğunlaştırılmıştır (concreated) ancak ve ancak T nin $1/4$ yarıçaplı çemberine ihmal edilebilir miktara kadar dahildir. Bu durumda, $Var(\chi)$ varyantı ve $E(\chi)$ olasılığı sırasıyla $Var(\chi)=\min_{\bar{x}\in T}\sum\chi\left|x-\bar{x}\right|^2$ ve $E(\chi)$ de bu ifadeyi minimalize eden $\bar{x}\in T$ konumu olarak tanımlanmaktadır. Bir optimizasyon formülü ile olasılığın bu dağılımı, herhangi bir gerçek uzunluk aralığı $<\frac{1}{2}$ üzerindeki dağılım kaldırıldığında ve gerçek olasılığın modül 1’ i hesapladığındaki gibi aynı sonucu verir. Yani T^n veya $T_N\left[X\right]^k$ üzerindeki χ' dağılımı yoğunlaştırılmıştır ancak ve ancak her bir katsayı torus üzerinde bağımsız yoğunlaştırılmış dağılıma sahiptir. O halde $E(\chi')$ olasılığı da her bir katsayının olasılıklarının vektörüdür ve $Var(\chi')$ de her bir katsayı varyansının maksimumunu gösterir (Chillotti ve ark., 2020).

Tanım 2.1.2 $n\geq 1$ bir tamsayı, $s\in\mathbb{Z}^n$ gizli anahtar ve ξ, $\mathbb{R}$ üzerinde bir dağılım olsun. $a\in \mathrm{T}^n$ ve $b=a.s+e$ olmak üzere $LWE_{s,\xi}$ bir (a,b) ikilisi örnekleyerek elde edilen $\mathrm{T}^n\times\mathrm{T}$ üzerinde bir dağılım olarak tanımlanır. Burada e hatası ξ dağılımından bir örnektir. S, $\mathbb{Z}^n$ üzerinde bir dağılım olsun.

1) Arama Problemi (Search Problem): Verilen pek çok bağımsız rastgele $LWE_{s,\xi}$ dan $s \leftarrow S$ nin bulunmasıdır.

2) Karar Problemi (Decision Problem): Verilen pek çok bağımsız rastgele örnekten, sabit bir $s \leftarrow S$ için $LWE_{s,\xi}$ örnekleri ve $\mathrm{T}^n \times \mathrm{T}$ den düzgün rastgele örnekleri ayırt etmektir (Regev, 2010; Chillotti ve ark., 2020).

Notasyon

$\mathrm{T}_q \coloneqq q^{-1}\mathbb{Z}/\mathbb{Z} \subset \mathrm{T}$ ile tanımlanmaktadır. Burada T_q nun gösterimi

$$\left\{\frac{i}{q} \bmod 1 \mid i \in \mathbb{Z}\right\} = \left\{\frac{i}{q} \mid i \in \mathbb{Z}/q\mathbb{Z}\right\} = \left\{0, \frac{1}{q}, ..., \frac{q-1}{q}\right\} \tag{2.1}$$

kümeleri şeklindedir. $\mathrm{T}_q \subset \mathrm{T}$ alt modülü ayrıklaştırılmış torus olarak adlandırılmaktadır. $\mathbb{Z}_q = \mathbb{Z}/q\mathbb{Z}$ olmak üzere $\frac{1}{q} \in \mathrm{T}_{N,q}[X]$ olarak düşünülürse herhangi bir $p \in \mathrm{T}_{N,q}[X]$ polinomu $p = \bar{p}.\frac{1}{q}$, $\bar{p} \in \mathbb{Z}_{N,q}[X]$ olarak da yazılabilir (Joye, 2021).

Tanım 2.1.3 (Ayrıklaştırılmış Torus Üzerinde LWE Problemi): $q, n \in \mathbb{N}$ ve $s = (s_1, s_2, ..., s_n) \leftarrow B^n$ olsun. $\hat{\chi}$, $q^{-1}\mathbb{Z}$ üzerinde hata dağılımı olsun. Ayrıklaştırılmış torus üzerinde LWE problemi Denklem 2.2 ve Denklem 2.3'te verilen dağılımları ayırt etmektir (Joye, 2021).

$$D_0 = \{(a, r) \mid a \leftarrow \mathrm{T}_q^n, r \leftarrow \mathrm{T}_q\} \tag{2.2}$$

$$D_1 = \{(a, r) \mid a = (a_1, a_2, ..., a_n) \leftarrow \mathrm{T}_q^n, \quad r = \sum_{j=1}^{n} s_j a_j + e, e \leftarrow \hat{\chi}\} \tag{2.3}$$

2.2. TFHE Şifreleme Şeması

2.2.1. Hatalarla öğrenme tabanlı

2.2.1.1. Şifreleme şeması

p ve q, $p \le q$ olacak şekilde iki pozitif tamsayı olsun. $\Delta = p/q$ tanımlansın. p ve q genellikle 2'nin kuvvetleri olarak seçilmektedir. Eğer 2'nin kuvvetleri değillerse mesajın kodlanma aşamasında yuvarlama işlemi uygulanmalıdır. Burada q şifreli metin modülü, p açık metin modülü ve Δ da ölçekleme (scaling) çarpım elemanı olarak adlandırılmaktadır. $M \in \mathrm{T}_p$ mesajının $s = (s_1, s_2, ..., s_n) \leftarrow \mathrm{T}^n$ gizli anahtarı ile şifrelenmesiyle $(a_1, a_2, ..., a_n, b) \leftarrow \mathrm{T}_q^{n+1}$ elde edilir. Burada $(a_1, a_2, ..., a_n) \leftarrow \mathrm{T}_q$ dan elde edilen rastgele düzgün dağılımdan örneklenmiştir.

$$b = \sum_{i=0}^{k-1} a_i s_i + \Delta M + e \in \mathrm{T}_q \tag{2.4}$$

olmak üzere $e \in \mathrm{T}_q$, χ Gauss dağılımından örneklenen hatadır. $c = (a_1, a_2, ..., a_n, b) \leftarrow \mathrm{T}_q^{n+1}$ şifreli metninin $(a_1, a_2, ..., a_n)$ kısmına maske, b kısmına ise vücut denmektedir. ΔM ise M mesajının kodlanması olarak adlanmaktır. M mesajının Δ faktörüyle çarpılmasındaki amaç ise mesaj uzayının T_q ya dönüştürülmesidir. Maske ve hatadan dolayı her şifreleme yapıldığında birbirinden farklı sonuçlar elde edilir. $c = (a_1, a_2, ..., a_n, b) \leftarrow \mathrm{T}_q^{n+1}$ şifreli metnin deşifrelenmesi için 2 adım gereklidir. İlk olarak

$$b - \sum_{i=0}^{k-1} a_i s_i = \Delta M + e \in \mathrm{T}_q \tag{2.5}$$

hesaplanır. İkinci olarak ise $\Delta M + e$, Δ ile bölünerek sonuç yuvarlanır. Yani $\lfloor (\Delta M + e)/\Delta \rceil = M$ ile mesaj elde edilir. Eğer $|e| < \Delta/2$ ise ikinci adım beklendiği gibi M mesajını döndürecektir (Chillotti ve ark., 2016).

2.2.1.2. Kodlama-dekodlama

Şifreleme algoritması girdi olarak P 'deki elemanları alır. Açık mesaj uzayı $q_i = 2^v$ ile $P = \mathrm{T}_p \subset \mathrm{T}_q$ dir. $Kod : M \to P$ kodlama fonksiyonu $m \in M$ yi $\mu \in P$ ye götürür. Kodlama, şifreleme işleminden önce uygulanmaktadır. $Dekod : P \to M$ dekodlama fonksiyonu deşifreleme işleminden sonra uygulanmaktadır.

Bit: Mesaj uzayı $M = \{0,1\}$ olmak üzere $b \in \{0,1\}$ biti için $Kod(b) = b/2$ şeklinde tanımlanır. Böylece 0 biti $0 = \frac{0}{q} \in \mathrm{T}_q$ torus elemanı olarak ve 1 biti $\frac{1}{2} = \frac{q/2}{q} \in \mathrm{T}_q$ torus elemanı olarak kodlanmaktadır. Ters işlem $Dekod(\mu) = \lceil 2\mu \rfloor \bmod 2$ olarak tanımlanmaktadır. Böylece eğer $\mu \in \left\{0, \frac{1}{2}\right\}$ ise $Dekod(\mu) \in \{0,1\}$ dir.

Tamsayılar mod p : $M = \{ j \bmod p \mid j \in \mathbb{Z} \} = \mathbb{Z} / p\mathbb{Z}$ ve $\Delta = q / p \in \mathbb{Z}$ olsun. Öyleyse kodlama ve dekodlama fonksiyonları Denklem 2.1.6 ve Denklem 2.1.7'deki gibidir.

$$Kod(i) = \frac{i \bmod p}{p} \left(= \frac{(i \bmod p)\Delta}{q} \right) \tag{2.6}$$

$$Dekod(\mu) = \lfloor p\mu \rceil \bmod p \tag{2.7}$$

Sabit-hassasiyet (fixed-precision) torus elemanları: $p \geq 2$, $p \mid q$ olsun. Bu durum tamsayılar mod p durumuna benzer olduğundan $t = \frac{i}{p}$, $i \in \mathbb{Z} / p\mathbb{Z}$ formundaki torus elemanlarının alt kümesini oluşturur. $x \in \mathrm{T}_p = p^{-1}\mathbb{Z} / \mathbb{Z}$ ve $\mu \in \mathrm{T}_q$ için

$$Kod(x) = x \tag{2.8}$$

$$Dekod(\mu) = \frac{\lfloor p\mu \rceil \bmod p}{p} \tag{2.9}$$

elde edilir.

Örnek 2.2.1.2.1: $p=4$, $q=64$ ve $q_i=2^4$ olmak üzere $Dekod(\mu)=\lfloor p\mu \rceil \bmod p$ için $\mu=\frac{48}{64}$ ise $Dekod(\mu)=\equiv 3\equiv -1(\bmod 4)$ elde edilir.

2.2.1.3. Homomorfik toplama

$p\le q$ olacak şekilde p ve q iki pozitif tamsayı olsun. $\Delta=p/q$ şeklinde tanımlansın. p ve q, 2'nin kuvvetleri olarak seçilsin. $m\in \mathrm{T}_p$ ve $m'\in \mathrm{T}_p$ mesajlarının $s=(s_1,s_2,...,s_n)\leftarrow \mathrm{T}^n$ gizli anahtarı ile şifrelenmesi sonucu oluşan şifreli mesajlar sırasıyla $c=(a_1,a_2,...,a_n,b)\leftarrow \mathrm{T}_q^{n+1}$ ve $c'=(a_1',a_2',...,a_n',b')\leftarrow \mathrm{T}_q^{n+1}$ olsun. Buradan iki şifreli mesajın homomorfik olarak toplanması sonucunda

$$c^*=c+c'=(a_1+a_1',a_2+a_2',...,a_n+a_n',b+b')\leftarrow \mathrm{T}_q^{n+1} \tag{2.10}$$

elde edilir. a_i ve a_i' ler $i\in\{0,...,k-1\}$ için T_q dan rastgele düzgün dağılım olduğundan ve

$$b+b'=\sum_{i=0}^{k-1}(a_i+a_i')s_i+\Delta(m+m')+(e+e')\in \mathrm{T}_q \tag{2.11}$$

olacağından bu toplam sadece hatanın biraz büyümesine sebep olur. Dolayısı ile $c^*=c+c'=E(m+m')$ olacağından homomorfik toplam sağlanır (Chillotti ve ark., 2020).

2.2.1.4. Homomorfik çarpma

$\Lambda=\sum_{i=0}^{N-1}\Lambda_i\in \mathrm{T}$ küçük bir sabit olsun. Bu sabitin $c=(a_1,a_2,...,a_n,b)\leftarrow \mathrm{T}_q^{n+1}$ şifreli mesajı ile homomorfik çarpımı sonucunda hata sabitin büyüklüğüyle orantılı olarak büyüceğinden

$$c^{(.)}=\Lambda.c=(\Lambda.a_1,\Lambda.a_2,...,\Lambda.a_n,\Lambda.b)=E(\Lambda.m)\leftarrow \mathrm{T}_q^{n+1} \tag{2.12}$$

elde edilir. Eğer homomorfik çarpım yapılan sabit küçük değilse hata büyümesinin önüne geçilmek için ayrıştırma (decomposition) yapılır. $\gamma\in \mathrm{T}_q$ büyük bir sabit olsun. Bu sabit küçük β tabanlarına ayrılarak

$$\gamma = \gamma_1 \frac{q}{\beta_1} + \gamma_2 \frac{q}{\beta_2} + ... + \gamma_l \frac{q}{\beta_l} \tag{2.13}$$

elde edilir. Burada $\gamma_1, \gamma_2, ..., \gamma_l \in \mathbb{Z}_\beta$ dır. Yani $Decomp^{\beta,l}(\gamma) = (\gamma_1, \gamma_2, ..., \gamma_l)$ dir. Burada q ve β, 2' nin kuvvetleri olarak alınır, eğer değillerse yuvarlama uygulanır. Şifreli mesajın sabitle çarpımı iki adımda gerçekleşir. İlk olarak mesajlar taban elemanlarıyla çarpılıp sonra şifrelenir. Yani

$$\bar{c} = (c_1, c_2, ...c_l) \in E\left(\frac{q}{\beta_1} m\right) \times E\left(\frac{q}{\beta_2} m\right) \times ... \times E\left(\frac{q}{\beta_l} m\right) \tag{2.14}$$

hesaplanır. İkinci olarak ise $\bar{C}$ ile γ nin bileşenleri çarpılarak toplanır. Yani

$$\left\langle Decomp^{\beta,l}(\gamma), \bar{c} \right\rangle = \sum_{j=1}^{l} \gamma_j . c_j = E(\gamma . m) \subseteq \mathrm{T}_q^{k+1} \tag{2.15}$$

elde edilir (Chillotti ve ark., 2020).

LWE tabanlı TFHE şemasının kütüphanesinde bulunan C++ diliyle yazılmış algoritmasının sözde kodu (pseudo-code) Şekil 2.1'de verilmiştir.

Algorithm 1: TFHE Sözde Kodu

Girdi:

1. $\mu_1, \mu_2 \in P := p^{-1}\mathbb{Z}/\mathbb{Z} = \mathrm{T}_p \subset \mathrm{T}_q$
2. s=$(s_1, s_2, \ldots, s_n) \leftarrow B^n$
3. pp={n, $\sigma, p, q\}$

Çıktı:

1. c$_1 = ((a_1, \ldots, a_n)_1, b_1) \in \mathrm{T}_q^{n+1}$
2. c$_2 = ((a_1, \ldots, a_n)_2, b_2) \in \mathrm{T}_q^{n+1}$
3. c$_3 := c_1 + c_2$
4. c$^* := c_1 \odot c_2$

Function Şifreleme

  for $i \leftarrow 1$ *to* 2 **do**

    $\mu_i^* = \mu_i + e_i$

    $b_i = \sum_{j=1}^{n} s_j \cdot a_{ji} + \mu_i^*$

    $c_i = ((a_1, \ldots, a_n)_i, b_i)$

return c_i

Function Deşifreleme

  for $i \leftarrow 1$ *to* 2 **do**

    $\mu_i^* = b_i - \sum_{j=1}^{n} s_j \cdot a_{ji}$

    $\mu_i = \frac{\lfloor p\mu_i^* \rceil \bmod p}{p}$

    round $\mu_i^* \rightarrow \mu_i \in P$

return μ_i

Function Homomorfik Toplama

  $(a_1, \ldots, a_n)_3 = (a_1, \ldots, a_n)_1 + (a_1, \ldots, a_n)_2$

  $\mu_3^* = \mu_1^* + \mu_2^*$

  $b_3 = b_1 + b_2$

  $c_3 := c_1 + c_2$

return c_3

Function Homomorfik Çarpma

  $\gamma \in \mathrm{T}_q$

  $\mathrm{Decomp}^{\beta,l}(\gamma) = (\gamma_1, \gamma_2, \ldots, \gamma_l)$

  $\gamma = \gamma_1 \frac{q}{\beta_1} + \gamma_2 \frac{q}{\beta_2} + \ldots + \gamma_l \frac{q}{\beta_l}, (\gamma_1, \gamma_2, \ldots, \gamma_l \in \mathbb{Z}_\beta)$

  $\bar{C} = (C_1, C_2, \ldots C_l) \in E\left(\frac{q}{\beta_1} M\right) \times E\left(\frac{q}{\beta_2} M\right) \times \ldots \times E\left(\frac{q}{\beta_l} M\right)$

  $\left\langle \mathrm{Decomp}^{\beta,l}(\gamma), \bar{C} \right\rangle = \sum_{j=1}^{l} \gamma_j . C_j = E(\gamma \cdot M) \subseteq \mathrm{T}_q^{i+1}$

return γC

Şekil 2.1. TFHE Sözde Kodu

2.2.2. Halkalarda hatalarla öğrenme tabanlı

LWE şifrelemesi, $T_{N,q}[X]$ deki torus polinomlarına kolayca genişletilebilir. T_q torus üzerindeki işlemler kolaylıkla $X^N + 1$ modülüne (ve q modülü) göre polinomlar üzerindeki işlemlerle yer değiştirilebilir. $a, b \in T_{N,q}[X]$ polinomları verilmek üzere $a + b$, a ve b polinomlarının $\left(X^N + 1, q\right)$ modülünde toplamını temsil eder. $a \in \mathbb{Z}_{N,q}[X]$ ve $b \in T_{N,q}[X]$ için $a \cdot b$, a ve b nin dış çarpımını temsil eder. Bazı p böler q için $P = T_p = p^{-1}\mathbb{Z}/\mathbb{Z}$ ile açık mesaj uzayı polinomların alt kümesidir.

$$P_N[X] := P[X]/\left(X^N + 1\right) = T_{N,p}[X] \subset T_{N,q}[X] \tag{2.16}$$

Buradaki $P_N[X]$, $T_{N,q}[X]$ in toplamsal alt grubunu oluşturur (Chillotti ve ark., 2020; Joye, 2021).

2.2.2.1. Şifreleme şeması

$KeyGen(1^\lambda)$: λ girdi güvenlik parametresi üzerinde N, 2 nin bir kuvveti ve $k \geq 1$ olmak üzere (N,k) tamsayı çifti tanımlansın. $p \mid q$ olacak şekilde p ve q pozitif tamsayıları seçilsin. $\hat{\chi}$, $q^{-1}\mathbb{Z}_N[X]$ üzerinde $\mathbb{R}_N[X]$ üzerindeki $\chi = N(0,\sigma^2)$ normal dağılımından indirgenmiş ayrıklaştırılmış hata dağılımı olsun. $s = (s_1, s_2, ..., s_n) \leftarrow B_N[X]^k$ vektörü rastgele düzgün bir şekilde örneklenmiş olsun. Açık mesaj uzayı $P_N[X] = \mathrm{T}_{N,p}[X] \subset \mathrm{T}_{N,q}[X]$ olsun. Genel parametreleri $pp = \{n, \sigma, p, q\}$ ve gizli anahtar $sk = s$ dir (Chillotti ve ark., 2016, 2020).

$Encrypt_{sk}(\mu)$: $\mu \in P_N[X]$ mesajının şifrelenmesi $c \leftarrow TRLWE_s(\mu) = (a_1, ..., a_k, b) \in \mathrm{T}_{N,q}[X]^{k+1}$ ile verilmektedir. Burada $(a_1, ..., a_k) \leftarrow \mathrm{T}_{N,q}[X]^k$ ve $e \leftarrow \hat{\chi}$ küçük hata için $b = \sum_{j=1}^{k} s_j . a_j + \mu^*$, $\mu^* = \mu + e$ şeklindedir (Chillotti ve ark., 2016, 2020).

$Decrypt_{sk}(c)$: $c = (a_1, ..., a_k, b)$ şifreli mesajını deşifrelemek için $s = (s_1, s_2, ..., s_k)$ kullanılarak $\mathrm{T}_{N,q}[X]$ da $\mu^* = b - \sum_{j=1}^{k} s_j . a_j$ hesaplanarak en yakın $\mu \in P_N[X]$ mesajı c nin deşifresi olur (Chillotti ve ark., 2016, 2020).

2.2.2.2. Kodlama-dekodlama

TRLWE şifreleme şeması rastgele $\mu \in P_N[X]$ polinomlarının şifrelenmesini destekler. Pek çok uygulamada, μ açık mesajı derecesi 0 olan polinoma kısıtlanmaktadır ve böylece P nin elemanı olarak görülmektedir. Böylece kodlama ve dekodlama fonksiyonları 2.2.1.2 bölümündeki gibi uygulanmaktadır.

N tane torus elemanı $\mu_0, ..., \mu_{N-1} \in P$ şifrelenmeye ihtiyaç duyulduğunda her biri

$$\mu(X) = \mu_0 + \mu_1 X + ... + \mu_{N-1} X^{N-1} \in P_N[X] \quad (2.17)$$

polinomunun katsayıları olarak temsil edilebilir. Böylesi bir optimizasyan katsayı paketleme (coefficient packing) olarak bilinir.

2.2.2.3. Homomorfik toplama

$P_N[X]$ deki μ_1 ve μ_2 mesajlarının TRLWE şifrelenmesi sırasıyla T_q^{n+1} de $c_1 \leftarrow TRLWE_s(\mu_1) = (a_1, ..., a_k, b) \in T_{N,q}[X]^{k+1}$ ve $c_2 \leftarrow TRLWE_s(\mu_2) = (a_1', ..., a_k', b') \in T_{N,q}[X]^{k+1}$ olsun. Eğer e_1 ve e_2 küçük gürültüleri sırasıyla c_1 ve c_2 şifreli mesajlarında mevcutsa

$$c_3 := c_1 + c_2 = (a_1 + a_1', ..., a_k + a_k', b + b') \in T_{N,q}[X]^{k+1} \quad (2.18)$$

$e_3 := e_1 + e_2$ hatası küçük olmak üzere $P_N[X]$ deki $\mu_3 := \mu_1 + \mu_2$ açık mesajlarının TRLWE şifrelemesidir (Chillotti ve ark., 2016, 2020).

2.2.2.4. Homomorfik çarpma

$\mu \in P_N[X]$ ve $K \in \mathbb{Z} \subset \mathbb{Z}_N[X]$ (yani $\mathbb{Z}_N[X]$ deki 0 dereceli polinom olarak kabul edilir) olsun. $c \leftarrow TRLWE_s(\mu)$ şifreli mesajının verilmek üzere $c' := K \cdot c$ şifreli mesajı, sonuç gürültüsünün küçük olmasını sağlayarak $P_N[X]$ deki $\mu' := K \cdot \mu$ mesajının şifrelenmesiyle elde edilir. Daha genel olarak küçük bir $k \in \mathbb{Z}_N[X]$ polinomu için sonuç gürültüsünün küçük olması sağlanarak $c' := k \cdot c$ şifreli mesajı $P_N[X]$ deki $\mu' := k \cdot \mu$ mesajının şifrelenmesiyle elde edilir (Chillotti ve ark., 2016, 2020).

$g = (1/B, .., 1/B^l) \in T_q^l$ aracı (gadget) vektörü $T_{N,q}[X]^l$ deki bir eleman olarak görülebilir. Boyutu ayarlayarak $T_{N,q}[X]$ üzerinde G gadget matrisini $G \in \mathrm{T}_{N,q}[X]^{(k+1)\times(k+1)l}$ olmak üzere Denklem 2.19'da olduğu gibi tanımlayabiliriz:

$$G^T = \mathrm{I}_{k+1} \otimes g^T = \begin{bmatrix} 1/B & & & & \\ \vdots & & & & \\ 1/B^l & & & & \\ & 1/B & & & \\ & \vdots & & & \\ & 1/B^l & & & \\ & & \ddots & & \\ & & & 1/B & \\ & & & \vdots & \\ & & & 1/B^l & \end{bmatrix} \tag{2.19}$$

İlgili $G^{-1}(\cdot)$ ters dönüşümü $\mathrm{T}_{N,q}[X]$ in $(k+1)$ polinomlarının vektörünü $\mathbb{Z}_N[X]$ in küçük katsayılı $(k+1)l$ polinomlarının vektörüne dönüştürür (yani $\left[-\lfloor \underline{B}/\underline{2} \rfloor, \lceil \overline{B}/\overline{2} \rceil\right)$ aralığına). $G^{-1}(\cdot)$ in tanımı 2.2.1.4 bölümündeki tanıma benzerdir. Tek fark T_q^{n+1} deki vektörlerin yerini $\mathrm{T}_{N,q}[X]^{(k+1)}$ deki vektörler almaktadır. Ayrıca herhangi bir $p \in \mathrm{T}_{N,q}[X]^{(k+1)}$ polinom vektörü için $G^{-1}(p).G^T \approx p$ ve $G^{-1}(p)$ nin küçük olması sağlanmaktadır (Bajard ve ark., 2017; Chillotti ve ark., 2016).

2.2.3. GSW tabanlı

GSW, Gentry, Sahai ve Waters tarafından 2013' te sunulan ve güvenliği LWE problemine dayalı olan bir şemadır (Gentry ve ark., 2013). Şema polinomlara da genişletilebilen aracı (gadget) ayrıştırma fonksiyonuna dayanmaktadır. Fakat TFHE' de TGSW, bir tahmin parametresine kadar yaklaşık araştırma fonksiyonu olarak kullanılmaktadır (Chillotti ve ark., 2016).

Tanım 2.2.3.1 Elemanları $T_N[X]$ den seçilen $d \times k+1$ boyutlu matris $M_{d\times k+1}(T_N[X])$ olmak üzere $h \in M_{d\times k+1}(T_N[X])$ olsun. $Dec_{h,\beta,\varepsilon}(v)$ nin h aracı (gadget) üzerinde β niteliği ve ε tahmini ile bir ayrıştırma algoritmasıdır ancak ve ancak herhangi bir $v \in T_N[X]^{k+1}$ TLWE örneklemi için $\|u\|_\infty \le \beta$ ve $\|u \cdot h - v\|_\infty \le \varepsilon$

olacak şekilde küçük bir $u \in \Re^d$ vektörü çıktı verir. Dahası v, $T_N[X]^{k+1}$ de düzgün bir şekilde dağıtıldığında $u \cdot h - v$ nin olasılığı 0 olmalıdır (Chillotti ve ark., 2016).

$$h = \begin{pmatrix} 1/B_g & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 1/B_g^l & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1/B_g \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1/B_g^l \end{pmatrix} \in M_{d,k+1}\left(T_N[X]\right) \tag{2.20}$$

h matrisi, her biri T deki sabit polinomları artan dizisini içeren köşegen matrislerden oluşur (Chillotti ve ark., 2016).

Tanım 2.2.3.2 (TGSW Örneklemleri) l ve $k \geq 1$ iki tamsayı, $\alpha \geq 0$ gürültü parametresi ve h de Denklem 2.20' de tanımlanan aracı (gadget) olsun. $s \in B_N[X]^k$ RLWE anahtarı olmak üzere $C \in M_{(k+1)l,k+1}\left(T_N[X]\right)$, $\mu \in \Re/h^{\perp}$ açık mesajının α gürültü parametresiyle yeni bir TGSW örneklemidir ancak ve ancak $Z \in M_{(k+1)l,k+1}\left(T_N[X]\right)$ in her bir satırı α Gauss gürültü parametresiyle bir Homojen (Homogeneous) TLWE örneklemidir. Karşılıklı olarak, $C \in M_{(k+1)l,k+1}\left(T_N[X]\right)$ elemanı bir TGSW örneklemidir ancak ve ancak $C - \mu \cdot h$ ın her bir satırı s anahtarı için 0 ın bir TLWE örneklemi olacak şekilde tek bir $\mu \in \Re/h^{\perp}$ polinomu ve tek bir s anahtarı mevcuttur (Chillotti ve ark., 2016).

Tanım 2.2.3.3 $A \in M_{(k+1)l,k+1}\left(T_N[X]\right)$, $s \in B_N[X]^k$ gizli anahtarı ve $\alpha \geq 0$ gürültü parametresi için bir TGSW örneklemi olsun. A aşaması, A nın her bir satırının $(k+1)l$ TLWE aşamalarının listesi olarak tanımlanmaktadır ve $\varphi(A) \in \left(T_N[X]\right)^{(k+1)l}$ olarak gösterilmektedir. Benzer şekilde, A nın hatası, A nın her bir satırının $(k+1)l$ TLWE hatalarının listesi olarak tanımlanmaktadır ve $Err(A)$ olarak gösterilmektedir (Chillotti ve ark., 2016).

Tanım 2.2.3.4 (Dış Çarpım) Dış çarpım $\otimes$ aşağıdaki gibi tanımlanmaktadır.

$$\otimes : TGSW \times TLWE \rightarrow TLWE$$
$$(A,b) \rightarrow A \otimes b = Dec_{h,\beta,\varepsilon}(b) \cdot A$$

Formül, sadece bir vektörün ayrıştırılması dışında orijinal GSW şemasında (Gentry ve ark., 2013) tanımlanan klasik çarpımla neredeyse aynıdır. Bu sebepten dolayı ayrıştırmadaki yaklaşık hesaplamalardan gelen ilave terimle birlikte neredeyse aynı gürültü yayım formülü elde edilmektedir (Chillotti ve ark., 2016).

2.3. Ayrıklaştırılmış Torus Üzerinde TFHE Şeması

T_q ayrıklaştırılmış torus verilsin. Açık mesaj uzayı T_q nun toplamsal alt grubudur yani p böler q için $P := p^{-1}\mathbb{Z}/\mathbb{Z} = \mathrm{T}_p \subset \mathrm{T}_q$ dur. $\hat{\chi}$, $q^{-1}\mathbb{Z}$ üzerinde $\mathbb{R}$ üzerindeki χ hata dağılımından indirgenmiş ayrıklaştırılmış dağılım olmak üzere $e \leftarrow \hat{\chi}$ hatası $e_0 \leftarrow \chi$ için $\bar{e} = round(qe_0)$ olmak üzere $e = \dfrac{\bar{e}}{q}$ ile tanımlanmaktadır. Şifreli mesajın $(a_1,...,a_n) \in \mathrm{T}_q^n$ maskesi $1 \le j \le n$ için $\bar{a}_j \leftarrow \mathbb{Z}/q\mathbb{Z}$ olmak üzere $a_j = \dfrac{\bar{a}_j}{q}$ ile verilmektedir. Böylece $e \leftarrow \hat{\chi}$ için karşılık gelen vücut $b = \sum_{j=1}^{n} s_j.a_j + \mu + e$ ile verilmektedir. Böylece $\mu \in P$ mesajının TLWE şifresi $(a_1,...,a_n,b)$ vektörüdür (Joye, 2021).

2.3.1. Hatalarla öğrenme tabanlı

2.3.1.1. Şifreleme şeması

$KeyGen(1^{\lambda})$: λ güvenlik parametresi üzerinde, $p \mid q$ olacak şekilde p ve q pozitif tamsayıları seçilsin. $\hat{\chi}$, $q^{-1}\mathbb{Z}$ üzerinde $\mathbb{R}$ üzerindeki $\chi = N(0,\sigma^2)$ normal dağılımından indirgenmiş ayrıklaştırılmış hata dağılımı olsun. $s = (s_1, s_2, ..., s_n) \leftarrow B^n$ vektörü rastgele düzgün bir şekilde örneklenmiş olsun. Açık mesaj uzayı $P = \mathrm{T}_p \subset \mathrm{T}_q$ olsun. Genel parametreleri $pp = \{n, \sigma, p, q\}$ ve gizli anahtar $sk = s$ dir (Joye, 2021).

$Encrypt_{sk}(\mu)$: $\mu \in P$ mesajının şifrelenmesi $c \leftarrow TLWE_s(\mu) = (a_1,...,a_n,b) \in \mathrm{T}_q^{n+1}$ ile verilmektedir. Burada $(a_1,...,a_n) \leftarrow \mathrm{T}_q^n$ ve $e \leftarrow \hat{\chi}$ küçük hata için $b = \sum_{j=1}^{n} s_j.a_j + \mu^*$, $\mu^* = \mu + e$ şeklindedir (Joye, 2021).

$Decrypt_{sk}(\mu)$: $c = (a_1,...,a_n,b)$ şifreli mesajını deşifrelemek için $s = (s_1, s_2,..., s_n)$ kullanılarak T_q da $\mu^* = b - \sum_{j=1}^{n} s_j.a_j$ hesaplanır ve $\mu = \dfrac{\lfloor p\mu^* \rceil \bmod p}{p}$ hesaplanarak en yakın $\mu \in P$ mesajı c nin deşifresi olur (Joye, 2021).

2.3.1.2. Homomorfik toplama

P deki μ_1 ve μ_2 mesajlarının TLWE şifrelenmesi sırasıyla T_q^{n+1} de $c_1 \leftarrow TLWE_s(\mu_1)$ ve $c_2 \leftarrow TLWE_s(\mu_2)$ olsun. e_1 ve e_2 küçük, $(a_1,...,a_n) \leftarrow \mathrm{T}_q^n$ ile $b = \sum_{j=1}^{n} s_j.a_j + \mu_1 + e_1$ ve $(a_1',...,a_n') \leftarrow \mathrm{T}_q^n$ ile $b' = \sum_{j=1}^{n} s_j.a_j' + \mu_2 + e_2$ olmak üzere T_q^{n+1} deki $c_3 := c_1 + c_2$, P deki $\mu_3 := \mu_1 + \mu_2$ mesajının şifrelenmesiyle oluşur. Yani $1 \le j \le n$ olmak üzere $a_j'' := a_j + a_j'$ ve $b_j'' := b_j + b_j'$ için $c_3 = (a_1'',...,a_n'',b'')$, $e_3 := e_1 + e_2$ hatasının küçük olmasıyla sağlanır (Joye, 2021).

2.3.1.3. Homomorfik çarpma

Şifreli metini bir sabit ile çarpmak bir dizi toplam ile elde edilebilir. Bunun bir sonucu olarak $\mu \in P$ için $c \leftarrow TLWE_s(\mu)$ TLWE şifreli mesajının bilinen bir küçük tamsayı olan $K \neq 0$ ile çarpımı eğer $K > 0$ ise $K.c = \underbrace{c + ... + c}_{k \text{ kez}}$ ile, eğer $K < 0$ ise $K.c = (-K).(-c)$ ile elde edilebilir. Bu işlem c nin her bileşeninin K ile çarpılmasıyla yapılır. Yani $c = (a_1,...,a_n,b) \in \mathrm{T}_q^{n+1}$ şifreli mesajının K ile çarpımı $K.c = (K.a_1,...,K.a_n,K.b)$ olup bu da $K.\mu \in P$ nin şifrelenmesiyle oluşan şifreli mesajdır (Joye, 2021).

Şifreli mesajların homomorfik çarpımı için ise q tamsayısı için $\mathrm{T}_q := q^{-1}\mathbb{Z}/\mathbb{Z}$ ayrıklaştırılmış torus üzerinde aracı (gadget) ayrıştırma adı verilen bir teknik sunulmaktadır (Joye, 2021). Bu teknik hatayı kontrol etmeye yardımcı olmaktadır. $B^l \mid q$ olacak şekilde B tabanı ve $l \geq 1$ tamsayısı verilsin. $G \in \mathrm{T}_q^{(n+1)\times(n+1)l}$ aracı (gadget) matrisi $g = (1/B, ..., 1/B^l) \in \mathrm{T}_q^l$ olmak üzere

$$G^T = \mathrm{I}_{n+1} \otimes g^T = diag(\underbrace{g^T, ..., g^T}_{n+1}) = \begin{bmatrix} 1/B & & & & \\ \vdots & & & & \\ 1/B^l & & & & \\ & 1/B & & & \\ & \vdots & & & \\ & 1/B^l & & & \\ & & \ddots & & \\ & & & 1/B & \\ & & & \vdots & \\ & & & 1/B^l & \end{bmatrix} \tag{2.21}$$

ile verilmektedir. $u \in \mathbb{Z}^{(n+1)l}$ giriş vektörü için $u.G^T$ çarpımı T_q^{n+1} de bir vektöre dönüşmektedir. Eğer $G^{-1} : \mathrm{T}_q^{n+1} \to \mathbb{Z}^{(n+1)l}$ ters dönüşümü göz önüne alınırsa herhangi bir $v \in \mathrm{T}_q^{n+1}$ vektörü için $G^{-1}(v).G^T \approx v$ ve $G^{-1}(v)$ küçük olacaktır.

Eğer $v_i \in \left[-\frac{1}{2}, \frac{1}{2}\right)$ için $v = (v_1, ..., v_{n+1}) \in \mathrm{T}_q^{n+1}$ ise $\overline{v_i} = \lfloor B^l v_i \rceil$ oluşturulup $u_{i,j} \in \left[-\lfloor B/2 \rfloor, \lceil B/2 \rceil\right)$ için $v_i \equiv \sum_{j=1}^{l} u_{i,j} B^{l-j} \pmod{B^l}$ yazılabilir. $g^{-1}(v_i) := (u_{i,1}, ..., u_{i,l}) \in \mathbb{Z}^l$ tanımlanır. Buradan

$$\begin{aligned} G^{-1}(v) &:= (g^{-1}(v_1), g^{-1}(v_2), ..., g^{-1}(v_{n+1})) \\ &= (u_{1,1}, ..., u_{1,l}, ..., u_{2,1}, ..., u_{2,l}, ..., u_{n+1,1}, ..., u_{n+1,l}) \in \mathbb{Z}^{(n+1)l} \end{aligned} \tag{2.22}$$

elde edilir. Eğer $B^l = q$ ise v nin her bir $v_i \in \left[-\frac{1}{2}, \frac{1}{2}\right)$ bileşeni için $\overline{v_i} = B^l v_i$ sağlanır. Böylece T_q üzerinde $G^{-1}(v).G^T = v$ tam olarak sağlanır (Joye, 2021).

3. TLWE ŞEMASININ RNS VARYANTI

3.1. Şifreleme Şeması

T_q ayrıklaştırılmış torus olmak üzere T_{q_i} nin bir toplamsal alt grubu $P_i = p_i^{-1}\mathbb{Z}/\mathbb{Z} = \mathrm{T}_{p_i} \subset \mathrm{T}_{q_i}$, $(p_i \mid q_i)$ açık mesaj uzayı olsun. $1 \le i \le k$ olmak üzere $\hat{\chi}_i$, $q_i^{-1}\mathbb{Z}$ üzerinde $\mathbb{R}$ üzerindeki χ den indirgenmiş hata dağılımı olmak üzere $e \leftarrow \hat{\chi}_i$ gürültü hatası $e = \dfrac{\overline{e}_i}{q_i}$, $\overline{e}_i = round(q_i e_0) \in \mathbb{Z}$, $e_0 \leftarrow \chi$ şeklinde tanımlansın. $1 \le j \le n$ olmak üzere $a_{j_i} = \dfrac{\overline{a}_{j_i}}{q_i}$, $\overline{a}_{j_i} \leftarrow \mathbb{Z}/q_i\mathbb{Z}$ için $p = p_1 p_2 ... p_i$ ve $q = q_1 q_2 ... q_i$ olmak üzere

$$(a_{1_i},...,a_{n_i}) \in \mathrm{T}_{q_i}^n,\ b_i = \left[\sum_{j=1}^{n} s_j \left(\sum_{i=1}^{k} a_{j_i}\right)\right] + \mu + e,\ e \leftarrow \hat{\chi} \tag{2.23}$$

şeklinde tanımlanmaktadır. $\mu \in P$ açık mesajının şifrelenmesi sonucunda $(a_{1_i},...,a_{n_i},b_i)$ elde edilir.

$KeyGen(1^{\lambda})$: λ güvenlik parametresi olmak üzere, n pozitif tamsayısı tanımlansın. $p_i \mid q_i$ olacak şekilde p_i ve q_i pozitif tamsayıları seçilsin ve $\mathbb{R}$ üzerinde $\chi = N(0,\sigma^2)$ normal dağılımından elde edilen $q_i^{-1}\mathbb{Z}$ üzerinde $\hat{\chi}_i$ ayrıklaştırılmış hata dağılımı tanımlansın. Rastgele bir $s = (s_1,...,s_n) \leftarrow B_n$ vektörü düzgün bir şekilde örneklensin. Açık mesaj uzayı $P = \mathrm{T}_{p_i} \subset \mathrm{T}_{q_i}$ olmak üzere genel parametreler $pp = (n, \sigma, p_i, q_i)$ ve gizli anahtar $sk = s$ dir. $(1 \le i \le k, 1 \le j \le n, k \le n)$.

$Encrypt_{sk}(\mu)$: $\mu \in P$ açık mesajının şifrelenmesi sonucunda $c_i \leftarrow TLWE_s(\mu) = (a_{1_i},...,a_{n_i},b_i) \in \mathrm{T}_{q_i}^{n+1}$ elde edilir. Burada $(a_{1_i},...,a_{n_i}) \leftarrow \mathrm{T}_{q_i}^n$ ve $e \leftarrow \hat{\chi}_i$ küçük gürültü olmak üzere $b_i = \left[\sum_{j=1}^{n} s_j \left(\sum_{i=1}^{k} a_{j_i}\right)\right] + \mu^*$ ve $\mu^* = \mu + e$ dir.

$Decrypt_{sk}(c_i)$: $c_i = (a_{1_i}, ..., a_{n_i}, b_i)$ şifreli mesajını deşifrelemek için $s = (s_1, ..., s_n)$ gizli anahtarı kullanılır ve T_{q_i} de $\mu^* = b_i - \sum_{j=1}^{n} s_j \left(\sum_{i=1}^{k} a_{j_i} \right)$ hesaplanır ve $\mu = \frac{\lceil p_i \mu^* \rfloor (\text{mod } p_i)}{p_i}$ elde edilir. Yani c_i nin deşifresi olarak en yakın açık mesaj $\mu \in P$ dir.

İspat: Şifreleme için $a_j = \frac{\bar{a}_j}{q} \in T_q$, $\bar{a}_j \in \mathbb{Z}/q\mathbb{Z}$ olduğundan $\frac{q_i}{q}.\bar{a}_j \in \mathbb{Z}/q_i\mathbb{Z}$ olacağından $a_j = \frac{(q/q_i).\bar{a}_j}{q}$ ifadesini $a_{j_i} = \frac{(q/q_i).\bar{a}_j}{q_i}$ ifadesine dönüştürmemiz gerektiğinden

$$\frac{(q/q_i).\bar{a}_j}{q}.\frac{q}{q_i} = \frac{(q/q_i).\bar{a}_j}{q_i} = a_{j_i} \tag{2.24}$$

olması gerekir. Dolayısıyla $\bar{a}_j \in \mathbb{Z}/q_i\mathbb{Z}$ için $a_{j_i} = \frac{(q/q_i).\bar{a}_j}{q_i} \in T_{q_i}$ olacaktır. Küçük olan e gürültüsü için de benzer şekilde işlemler uygulanabilir.

$1 \le j \le n$, $1 \le i \le k$, $k \le n$ olmak üzere $p = p_1, ..., p_i$ ve $q = q_1, ..., q_i$ için $\mu \in P_i = \left\{ 0, \frac{1}{p_i}, ..., \frac{p_i - 1}{p_i} \right\}$ açık mesajı için

$$c_i \leftarrow TLWE_s(\mu) = \left(a_{1_i}, ..., a_{n_i}, b_i \right),\ \left(a_{1_i}, ..., a_{n_i} \right) \leftarrow T_{q_i}^n \tag{2.25}$$

$$b_i = \sum_{j=1}^{n} s_j \sum_{i=1}^{k} a_{j_i} + \mu + e,\ e \leftarrow \hat{\chi} \tag{2.26}$$

elde edilir. $\mu \in P_i$ olduğu için $\mu = \frac{m}{p_i}$ olacak şekilde $m \in [0, p_i)$ tamsayısı tek türlü belirlidir. $Decrypt_{sk}(c)$ işlemi sonucunda $\mu^* = \mu + e \in \mathrm{T}_{q_i} \subset \mathrm{T}$ ile $\frac{\lceil p_i \mu^* \rfloor (\mathrm{mod}\ p_i)}{p_i}$ çıktısını verir. Eğer $|e| < \frac{1}{2p_i}$ ise bazı $\omega \in \mathbb{Z}$ için

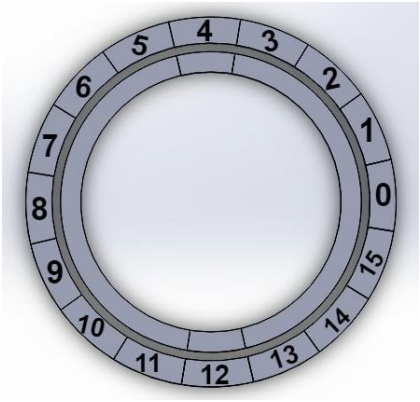
$$\begin{aligned} \lceil p_i \mu^* \rfloor &= \lfloor p_i [(\mu + e) \bmod 1] \rfloor = \lfloor p_i [(\frac{m}{p_i} + e) \bmod 1] \rfloor \\ &= \lfloor m + p_i (e + \overline{\omega}) \rfloor = m + \lfloor p_i \overline{e} \rfloor = m \end{aligned} \tag{2.27}$$

elde edilir. Bu durumda $\lfloor p_i \overline{\mu^*} \rfloor \bmod p_i = \lfloor p_i (\mu + \overline{e}) \rfloor \bmod p_i = m$ olur ve böylece

$$\frac{\lceil p_i \mu^* \rfloor (\mathrm{mod}\ p_i)}{p_i} = \frac{m}{p_i} = \mu \tag{2.28}$$

elde edilir.

Örnek 3.1.1: $p = 4$ ve $q = 64$ olsun. $p = 2.2 = p_1.p_2$ ve $q = 2^2.2^4$ olmak üzere $q_i = 2^4$ için $\mathrm{T}_{q_i} = \left\{0, \frac{1}{16}, ..., \frac{15}{16}\right\}$ ve $P = \left\{0, \frac{1}{2}\right\}$ olur.



Şekil 3.1. Torus 16

$|e| < \frac{1}{2p_i} = \frac{1}{4}$ yani $e \in \left\{\frac{-3}{16}, ..., \frac{3}{16}\right\}$ olur. $\mu^* = \mu + e$ olmak üzere $\mu \in P = \left\{0, \frac{8}{16}\right\}$ için

$\mu^* \in \left\{\frac{13}{16},\frac{14}{16},\frac{15}{16},\frac{0}{16},\frac{1}{16},\frac{2}{16},\frac{3}{16}\right\}$ ise $\mu = 0$ olur.

$\mu^* \in \left\{\frac{5}{16},\frac{6}{16},\frac{7}{16},\frac{8}{16},\frac{9}{16},\frac{10}{16},\frac{11}{16}\right\}$ ise $\mu = \frac{8}{16}$ olur.

3.2. Kodlama-Dekodlama

Şifreleme algoritması girdi olarak P 'deki elemanları alır. Açık mesaj uzayı $q_i = 2^{\varepsilon}$ ile $P = \mathrm{T}_{p_i} \subset \mathrm{T}_{q_i}$ dir. $Kod : M \to P$ kodlama fonksiyonu $m \in M$ yi $\mu \in P$ ye götürür. Kodlama, şifreleme işleminden önce uygulanmaktadır. $Dekod : P \to M$ dekodlama fonksiyonu deşifreleme işleminden sonra uygulanmaktadır.

Bit: Mesaj uzayı $M = \{0,1\}$ olmak üzere $b \in \{0,1\}$ biti için $Kod(b) = b/2$ şeklinde tanımlanır. Böylece 0 biti $0 = \frac{0}{q_i} \in \mathrm{T}_{q_i}$ torus elemanı olarak ve 1 biti $\frac{1}{2} = \frac{q_i/2}{q_i} \in \mathrm{T}_{q_i}$ torus elemanı olarak kodlanmaktadır. Ters işlem $Dekod(\mu) = \lceil 2\mu \rfloor \bmod 2$ olarak tanımlanmaktadır. Böylece eğer $\mu \in \left\{0, \frac{1}{2}\right\}$ ise $Dekod(\mu) \in \{0,1\}$ dir.

Tamsayılar $\bmod p_i$: $M = \{ j \bmod p_i \mid j \in \mathbb{Z}\} = \mathbb{Z} / p_i\mathbb{Z}$ ve $\Delta = q_i / p_i \in \mathbb{Z}$ olsun. Öyleyse kodlama ve dekodlama fonksiyonları Denklem 3.1.6 ve Denklem 3.1.7'deki gibidir.

$$Kod(j) = \frac{j \bmod p_i}{j}\left(= \frac{(j \bmod p_i)\Delta}{q_i} \right) \tag{2.29}$$

$$Dekod(\mu) = \lfloor p_i \mu \rceil \bmod p_i \tag{2.30}$$

Sabit-hassasiyet torus elemanları: $p_i \geq 2$, $p_i \mid q_i$ olsun. Bu durum tamsayılar $\bmod p_i$ durumuna benzer olduğundan $t = \frac{j}{p_i}$, $j \in \mathbb{Z} / p_i\mathbb{Z}$ formundaki torus elemanlarının alt kümesini oluşturur. $x \in \mathrm{T}_{p_i} = p_i^{-1}\mathbb{Z} / \mathbb{Z}$ ve $\mu \in \mathrm{T}_{q_i}$ için

$$Kod(x) = x \tag{2.31}$$

$$Dekod(\mu) = \frac{\lfloor \underline{p_i} \overline{\mu} \rceil \bmod p_i}{p_i} \tag{2.32}$$

elde edilir.

Örnek 3.2.1: $p = 4 = 2.2$, $q = 64 = 2^6 = 2^4 2^2$ ve $q_i = 2^4$ olmak üzere $Dekod(\mu) = \lfloor \underline{p_i} \overline{\mu} \rceil \bmod p_i$ için $\mu = \frac{8}{16}$ ise $Dekod(\mu) = \left(2.\frac{8}{16}\right) \bmod 2 \equiv 1 \equiv -1 (\bmod 2)$ elde edilir.

3.3. Homomorfik Toplama

$\mathrm{T}_{q_i}^{n+1}$ de $c_1 \leftarrow TLWE_s(\mu_1)$ ve $c_2 \leftarrow TLWE_s(\mu_2)$ sırasıyla P deki μ_1 ve μ_2 nin şifrelenmeleriyle oluşan şifreli mesajlar olmak üzere

$$c_{1_i} = \left(a_{1_i}, ..., a_{n_i}, b_i\right) \in \mathrm{T}_{q_i}^{n+} \tag{2.33}$$

$$c_{2_i} = \left(a'_{1_i}, ..., a'_{n_i}, b'_i\right) \tag{2.34}$$

şeklinde hesaplanır. Burada $\left(a_{1_i}, ..., a_{n_i}\right) \leftarrow \mathrm{T}_{q_i}^{n}$ ve $b_i = \sum_{j=1}^{n} s_j \sum_{i=1}^{k} a_{j_i} + \mu_1 + e_1$; $\left(a'_{1_i}, ..., a'_{n_i}\right) \leftarrow \mathrm{T}_{q_i}^{n}$ ve $b'_i = \sum_{j=1}^{n} s_j \sum_{i=1}^{k} a'_{j_i} + \mu_2 + e_2$ ve e_1, e_2 küçük olmak üzere $\mathrm{T}_{q_i}^{n+1}$ de $c_3 := c_1 + c_2$, P deki $\mu_3 := \mu_1 + \mu_2$ mesajının şifrelenmesi sonucu oluşan şifreli mesajdır. Yani

$$c_{3_i} = \left(a''_{1_i}, ..., a''_{n_i}, b''_i\right) \begin{Bmatrix} a''_{j_i} = a_{j_i} + a'_{j_i} \\ b''_i = b_i + b'_i \end{Bmatrix} \ (1 \le j \le n) \tag{2.35}$$

ki bu eşitlik $e_3 := e_1 + e_2$ nin küçük olma şartını sağlar.

3.4. Bilinen Bir Sabitle Homomorfik Çarpma

Şifreli mesajı bir sabit ile çarpmak bir dizi toplama ile elde edilebilir. Bunun bir sonucu olarak $\mu \in P$ için $c_i \leftarrow TLWE_s(\mu)$ TLWE şifreli mesajı verilsin. $\exists K \neq 0$ bilinen

küçük bir tamsayı için $K.\mu$; eğer $K>0$ ise $K.c_i = \underbrace{c_i+...+c_i}_{K\,\text{kez}}$, eğer $K<0$ ise $K.c_i=(-K).(-c_i)$ şeklinde elde edilir. Böylece c_i nin her vektör bileşenini K ile çarpmak gerekir. Yani $c_i=\left(a_{1_i},...,a_{n_i},b_i\right)\in \mathrm{T}_{q_i}^{n+1}$ şifreli mesajı için $K.c_i=\left(K.a_{1_i},...,K.a_{n_i},K.b_i\right)$ elde edilir. Aynı şekilde elde edilen hatanın küçük olması sağlanarak $\mathrm{T}_{q_i}^{n+1}$ deki $K.c_i$, P deki $K.\mu$ açık mesajının şifrelenmiş halidir.

3.5. Homomorfik Çarpma

Genel bir q tamsayısı için $q=q_1q_2...q_n$, $1\le i\le n$ olmak üzere $\mathrm{T}_{q_i}=q_i^{-1}\mathbb{Z}/\mathbb{Z}$ üzerinde CRT aracı (gadget) ayrıştırma tekniği sunulacaktır. B tabanı $B=B_1B_2...B_m$, $1\le j\le m$ olmak üzere $B_j^l \mid q_i$ olacak şekilde $\exists l\ge 1$ tamsayısı için karşılık gelen CRT aracı (gadget) ayrıştırma matrisi $G_i\in \mathrm{T}_{q_i}^{(n+1)\times(n+1)l}$ olmak üzere

$$G_i^T=\mathrm{I}_{n+1}\otimes g_i^T=diag(\underbrace{g_i^T,...,g_i^T}_{n+1})=\begin{bmatrix} 1/B_i & & & & \\ \vdots & & & & \\ 1/B_i^l & & & & \\ & 1/B_i & & & \\ & \vdots & & & \\ & 1/B_i^l & & & \\ & & \ddots & & \\ & & & 1/B_i & \\ & & & \vdots & \\ & & & 1/B_i^l & \end{bmatrix} \qquad (2.36)$$

ile verilmektedir. $u\in\mathbb{Z}^{(n+1)l}$ giriş vektörü için $u.G_i^T$ çarpımı $\mathrm{T}_{q_i}^{n+1}$ de bir vektör olur. Eğer $G_i^{-1}:\mathrm{T}_{q_i}^{n+1}\to\mathbb{Z}^{(n+1)l}$ ters dönüşümü göz önüne alınırsa herhangi bir $v_i\in\mathrm{T}_{q_i}^{n+1}$ vektörü için $G_i^{-1}(v_i).G_i^T\approx v_i$ ve $G_i^{-1}(v_i)$ küçük olacaktır. Eğer $v_i=(v_{i,1},...,v_{i,n+1})\in\mathrm{T}_{q_i}^{n+1}$ ise $\overline{v}_{i,k}=\lfloor B^l v_{i,k}\rceil$,$1\le i\le n$, $1\le k\le n+1$ oluşturulup $\overline{v}_{i,k}\equiv\sum_{j=1}^{l}u_{i,k,j}B_i^{l-j} \pmod{B^l}$ yazılabilir. Burada

$$\frac{1}{B} \in \mathrm{T}_q = q^{-1}\mathbb{Z}/\mathbb{Z},\ \frac{1}{B} = \frac{\overline{1/B}}{q},\ \overline{1/B} \in \mathbb{Z}_q \tag{2.37}$$

olmak üzere $\overline{1/B_i} \in \mathbb{Z}_{q_i}$ buradan da $\frac{1}{B_i} = \frac{\overline{1/B_i}}{q_i} \in \mathrm{T}_{q_i}$ elde edilir. $g_i^{-1}(v_{i,k}) := (u_{i,k,1},...,u_{i,k,l})$ tanımlansın. Öyleyse

$$\begin{aligned} G_i^{-1}(v_{i,k}) &:= (g_i^{-1}(v_{i,1}), g_i^{-1}(v_{i,2}),..., g_i^{-1}(v_{i,n+1})) \\ &= (u_{1,1,1},...,u_{1,1,l},...,u_{2,1,1},...,u_{i,n+1,l}) \in \mathbb{Z}^{(n+1)l} \end{aligned} \tag{2.38}$$

elde edilir. Eğer $B_j^l = q_i$ ise $\overline{v}_{i,k} = B_i^l v_{i,k}$ sağlanır. Böylece T_{q_i} üzerinde $G_i^{-1}(v_i).G_i^T = v_i$ tam olarak sağlanır.

Örnek 3.5.1 $n=1$, $l=2$, $B=6$ ve $q=72$ olsun. $\mathrm{T}_q = \frac{1}{72}\mathbb{Z}/\mathbb{Z}$ olur. $B = 2.3 = B_1.B_2$ ve $q = 2^3.3^2 = q_1.q_2$ olsun.

B_1 için $G_1^T = \begin{bmatrix} 1/2 & 0 \\ 1/4 & 0 \\ 0 & 1/2 \\ 0 & 1/4 \end{bmatrix}$ ve B_2 için $G_2^T = \begin{bmatrix} 1/3 & 0 \\ 1/9 & 0 \\ 0 & 1/3 \\ 0 & 1/9 \end{bmatrix}$ olur. $v = \left(\frac{43}{72}, \frac{34}{72}\right)$ olmak üzere

$$v = \left(\frac{43}{72}, \frac{34}{72}\right) = \left(\frac{3}{8} + \frac{2}{9}, \frac{2}{8} + \frac{2}{9}\right) = \left(v_{1,1} + v_{2,1}, v_{2,1} + v_{2,2}\right) \text{ elde edilir.}$$

$v_1 = \left(\frac{3}{8}, \frac{2}{8}\right)$ için $v_1 \equiv \left(\frac{-5}{8}, \frac{2}{8}\right) \bmod 1$ ve buradan $\overline{v}_{1,1} = \left\lfloor B_1.\overline{v_{1,1}} \right\rfloor = \left\lfloor 2^2.(-5/8) \right\rfloor = -3$ ve $\overline{v}_{1,2} = \left\lfloor 2^2.(2/8) \right\rfloor = 1$ olduğundan $-3 = -1.2 - 1$ ve $1 = 1.2 - 1$ bulunacağından $G^{-1}(v_1) = (-1,-1,1,-1)$, $G^{-1}(v_1).G_1^{\mathrm{T}} \approx v_1$ elde edilir.

Benzer şekilde $v_2 \equiv \left(\frac{2}{9}, \frac{2}{9}\right)$ için $v_2 \equiv \left(\frac{-7}{9}, \frac{2}{9}\right) \bmod 1$ ve buradan $\bar{v}_{2,1} = \lfloor \underline{B_2}.\overline{v_{2,1}} \rceil = \lfloor \underline{3}.\overline{(-7/9)} \rceil = -3$ ve $\bar{v}_{2,2} = \lfloor \underline{3}.\overline{(2/9)} \rceil = 1$ olduğundan $-3 = -1.2 - 1$ ve $1 = 1.2 - 1$ bulunacağından $G^{-1}(v_1) = (-1,-1,1,-1)$, $G^{-1}(v_2).G_2^{\mathrm{T}} \approx v_2$ elde edilir.

Bulduğumuz sonuçlardan da $\langle q_2 v_1, q_1 v_2 \rangle = \left(\frac{42}{72}, \frac{34}{72}\right) \approx v$ elde edilir.

4. SONUÇ VE ÖNERİLER

Tam homomorfik şifrelemede karşılaştığımız şifreleme süresinin uzunluğu, şemanın pratik kullanımının önünde bir engeldir. Buna bir çözüm olarak, önerilen Torus Tam Homomorfik Şifreleme şemasının İlaria ve ark. homomorfik şifreleme için umut vaat etmektedir. Şemayı geliştirmek için birçok çalışma yapılmış olsa da şemanın LWE problemine dayalı kısmı için bu tezde daha önce yapılmamış Kalan Sayılar Sistemi varyantı oluşturulmuştur. RNS varyantını oluşturmak için kullandığımız CRT ile şemada kullanılan torus yapısının daha küçük modüllere indirgenmesi sağlanmıştır. RNS varyantlarının şemaları iyileştirdiği göz önüne alındığında, RNS-TFHE şeması için de şifreleme ve deşifreleme sürelerinde yaklaşık 2 kat iyileşme sağlanmıştır. Bu katkılarla birlikte, şemanın geliştirilerek kullanımının basitleştirilmesine ve yaygınlaştırılmasına katkıda bulunulması amaçlanmıştır.

Algorithm 2: RNS-TFHE Sözde Kodu

Girdi:

1. $(\mu_1\mu_2...\mu_k) = \mu \in P$
2. s=$(s_1, s_2, \ldots, s_n) \leftarrow B^n$
3. pp={n, $\sigma, p, q = (q_1q_2...q_k)$}
4. c-m=1, p-m=1

Çıktı :

1. $(c_1c_2...c_k) = c \in \mathrm{T}_q^k$

Function `Şifreleme`

- **for** $i \leftarrow 1$ *to* k **do**
  - $\mu_i^* = \mu_i + e_i$
  - $b_i = \sum_{j=1}^{n} s_j \cdot a_{ji} + \mu_i^*$
  - $c_i = ((a_1, \ldots, a_n)_i, b_i)$
  - $c - m = c - m * c_i$

return $c - m$

Function `Deşifreleme`

- **for** $i \leftarrow 1$ *to* k **do**
  - $\mu_i^* = b_i - \sum_{j=1}^{n} s_j \cdot a_{ji}$
  - $\mu_i = \frac{\lfloor p\mu_i^* \rceil \bmod p}{p}$
  - round $\mu_i^* \rightarrow \mu_i \in P$
  - $p - m = p - m * mu_i$

return $p - m$

Şekil 4.1. RNS-TFHE Sözde Kodu

Bu tezde gerçekleştirilen teorik adımların sonucunda TFHE şemasına ait algoritmada değişiklikler yapılmıştır. TFHE şemasının kullandığı uzay, CRT yöntemi kullanılarak

tabanlarına bölündüğü için yeni uzayın tabanlarına ait işlemlerin ayrı ayrı şemanın algoritmasındaki şifreleme ve deşifreleme fonksiyonlarındaki açık mesajlara ve şifreli mesajlara uygulanması gerekmektedir. Bu yüzden RNS-TFHE nin sözde kodu olan Şekil 4.1'den de görüldüğü üzere CRT yöntemi öncelikle asıl veriyi tabanlarına bölerek işlemler yapmaya ve elde edilen sonuçların çarpılmasına dayanmaktadır. Algoritmalar üzerinde yapılan incelemeler sonucunda şemaya ait şifreleme ve deşifreleme süreleri Tablo 4.1 ve Tablo 4.2 arasında karşılaştırma yapılarak elde edilen yeni RNS-TFHE şemasının orijinal şemayı iyileştirdiği ve geliştirdiği gözlemlenmektedir. Yapılan işlemler şemanın TLWE tabanlı bootstrapping yapısına uygulanarak sonuçlar analiz edilebilir.

Tablo 4.1. TFHE şemasının şifreleme ve deşifreleme sürelerinin örnekleri.

	512^1		1024^2	
TFHE	1.000	10.000	1.000	10.000
Şifreleme	$\sim$ 130 ms	$\sim$ 105 ms	$\sim$ 190 ms	$\sim$ 235 ms
Deşifreleme	$\sim$ 3 ms	$\sim$ 4 ms	$\sim$ 6 ms	$\sim$ 7 ms

Not: Bu tablo TFHE şemasının 1.000 ve 10.000 örnekleme sahip örneğidir.

[1]Anahtar uzunluğu 512 bit.

[2]Anahtar uzunluğu 1024 bit.

Tablo 4.2. RNS-TFHE şemasının şifreleme ve deşifreleme sürelerinin örnekleri.

	512^1		1024^2	
RNS-TFHE	1.000	10.000	1.000	10.000
Encryption	$\sim$ 25 ms	$\sim$ 60 ms	$\sim$ 85 ms	$\sim$ 108 ms
Decryption	$\sim$ 3 ms	$\sim$ 4 ms	$\sim$ 6 ms	$\sim$ 7 ms

Not: Bu tablo RNS-TFHE şemasının 1.000 ve 10.000 örnekleme sahip örneğidir.

[1]Anahtar uzunluğu 512 bit.

[2]Anahtar uzunluğu 1024 bit.

KAYNAKLAR

Hungerford, T. W. (2012). Algebra (Vol. 73). Springer Science & Business Media. https://doi.org/10.1007/978-1-4612-6101-8

Çallıalp, F. (2011). Örneklerle soyut cebir. Birsen yayınevi.

Aydın, N., Kazım, K. ve Kandamar, H. (2012). Çözümlü Soyut Cebir Problemleri. İstanbul: Kriter Yayınevi.

Ling, S. X. (2004). Coding Theory: A First Course. New York: Cambridge University Press. https://doi.org/10.1017/cbo9780511755279

Çallıalp, F. T. (2009). Değişmeli Halkalar ve Modüller. İstanbul: Birsen Yayınevi.

Paar, C. P. (2010). Understanding Cryptography. New York: Springer. https://doi.org/10.1007/978-3-642-04101-3

Chauhan, K. K., Sanger, A. K. ve Verma, A. (2015). Homomorphic encryption for data security in cloud computing. *In 2015 International conference on information technology (ICIT)*, 206-209. IEEE. https://doi.org/10.1109/icit.2015.39

Özdemir, F. ve Koç, Ç. K. (2022). Development of Cryptography since Shannon. *Cryptology ePrint Archive*.

Gentry, C. (2009). Fully homomorphic encryption using ideal lattices. *In Proceedings of the forty-first annual ACM symposium on Theory of computing*, 169-178. https://doi.org/10.1145/1536414.1536440

Van Dijk, M., Gentry, C., Halevi, S. ve Vaikuntanathan, V. (2010). Fully homomorphic encryption over the integers. *In Advances in Cryptology–EUROCRYPT 2010: 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Proceedings 29, 24-43. https://doi.org/10.1007/978-3-642-13190-5_2

Brakerski, Z., Gentry, C. ve Vaikuntanathan, V. (2014). (Leveled) fully homomorphic encryption without bootstrapping. *ACM Transactions on Computation Theory (TOCT)*, 6(3), 1-36. https://doi.org/10.1145/2090236.2090262

Fan, J. ve Vercauteren, F. (2012). Somewhat practical fully homomorphic encryption. *Cryptology ePrint Archive*.

Gentry, C., Sahai, A. ve Waters, B. (2013). Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. *In Advances in Cryptology–CRYPTO 2013: 33rd Annual Cryptology Conference*, Proceedings, Part I, 75-92. https://doi.org/10.1007/978-3-642-40041-4_5

Ducas, L. ve Micciancio, D. (2015). FHEW: bootstrapping homomorphic encryption in less than a second. *In Advances in Cryptology--EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Proceedings, Part I 34, 617-640. https://doi.org/10.1007/978-3-662-46800-5_24

Cheon, J. H., Kim, A., Kim, M. ve Song, Y. (2017). Homomorphic encryption for arithmetic of approximate numbers. *In Advances in Cryptology–ASIACRYPT 2017: 23rd International Conference on the Theory and Applications of Cryptology and Information Security*, Proceedings, Part I 23, 409-437. https://doi.org/10.1007/978-3-319-70694-8_15

Chillotti, I., Gama, N., Georgieva, M. ve Izabachène, M. (2020). TFHE: fast fully homomorphic encryption over the torus. *Journal of Cryptology*, 33(1), 34-91. https://doi.org/10.1007/s00145-019-09319-x

Gentry, C., Halevi, S. ve Smart, N. P. (2012). Homomorphic evaluation of the AES circuit. *In Advances in Cryptology–CRYPTO 2012: 32nd Annual Cryptology Conference*, 850-867. https://doi.org/10.1007/978-3-642-32009-5_49

Bajard, J. C., Eynard, J., Hasan, M. A. ve Zucca, V. (2017). A full RNS variant of FV like somewhat homomorphic encryption schemes. *In Selected Areas in Cryptography–SAC 2016: 23rd International Conference*, 423-442. https://doi.org/10.1007/978-3-319-69453-5_23

Halevi, S., Polyakov, Y. ve Shoup, V. (2019). An improved RNS variant of the BFV homomorphic encryption scheme. *In Topics in Cryptology–CT-RSA 2019: The Cryptographers' Track at the RSA Conference 2019*, 83-105. https://doi.org/10.1007/978-3-030-12612-4_5

Cheon, J. H., Han, K., Kim, A., Kim, M. ve Song, Y. (2019). A full RNS variant of approximate homomorphic encryption. *In Selected Areas in Cryptography–SAC 2018: 25th International Conference*, 347-368. https://doi.org/10.1007/978-3-030-10970-7_16

Joye, M. (2021). Guide to fully homomorphic encryption over the [discretized] torus. *Cryptology ePrint Archive*.

Regev, O. (2010). The learning with errors problem (invited survey). *In 2010 IEEE 25th Annual Conference on Computational Complexity*, 191-204. https://doi.org/10.1109/ccc.2010.26

Lyubashevsky, V., Peikert, C. ve Regev, O. (2013). On ideal lattices and learning with errors over rings. *Journal of the ACM (JACM)*, 60(6), 1-35. https://doi.org/10.1007/978-3-642-13190-5_1

Hoffstein, J., Pipher, J. ve Silverman, J. H. (2006). NTRU: A ring-based public key cryptosystem. *In Algorithmic Number Theory: Third International Symposium*, 267-288. https://doi.org/10.1007/bfb0054868

Cheon, J. H. ve Stehlé, D. (2015). Fully homomophic encryption over the integers revisited. *In Advances in Cryptology--EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Proceedings, Part I, 513-536. https://doi.org/10.1007/978-3-662-46800-5_20

Brakerski, Z. (2012). Fully homomorphic encryption without modulus switching from classical GapSVP. *In Advances in Cryptology–CRYPTO 2012: 32nd Annual Cryptology Conference*, 868-886. https://doi.org/10.1007/978-3-642-32009-5_50

Mattila, V., Dwivedi, P., Gauri, P. ve Ahbab, M. D. (2022). Homomorphic encryption in 5ire blockchain. *International Journal of Social Sciences and Management Review*, 5, 264-276. https://doi.org/10.37602/ijssmr.2022.5219

Xu, G., Zhang, J. ve Wang, L. (2022). An Edge Computing Data Privacy-Preserving Scheme Based on Blockchain and Homomorphic Encryption. *In 2022 International Conference on Blockchain Technology and Information Security (ICBCTIS)*, 156-159. https://doi.org/10.1109/icbctis55569.2022.00044

Gentry, C. ve Halevi, S. (2011). Fully homomorphic encryption without squashing using depth-3 arithmetic circuits. *In 2011 IEEE 52nd annual symposium on foundations of computer science*, 107-109. https://doi.org/10.1109/focs.2011.94

Chillotti, I., Gama, N., Georgieva, M. ve Izabachene, M. (2016). Faster fully homomorphic encryption: Bootstrapping in less than 0.1 seconds. *In Advances in Cryptology–ASIACRYPT 2016: 22nd International Conference on the Theory and Application of Cryptology and Information Security*, Proceedings, Part I 22, 3-33. https://doi.org/10.1007/978-3-662-53887-6_1

Smart, N. P. ve Vercauteren, F. (2010). Fully homomorphic encryption with relatively small key and ciphertext sizes. *In Public Key Cryptography–PKC 2010: 13th International Conference on Practice and Theory in Public Key Cryptography*, Proceedings 13, 420-443. https://doi.org/10.1007/978-3-642-13013-7_25

Gentry, C. ve Halevi, S. (2011). Implementing gentry's fully-homomorphic encryption scheme. *In Advances in Cryptology–EUROCRYPT 2011: 30th Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Proceedings 30, 129-148. https://doi.org/10.1007/978-3-642-20465-4_9

Brakerski, Z. ve Vaikuntanathan, V. (2011). Fully homomorphic encryption from ring-LWE and security for key dependent messages. *In Advances in Cryptology–CRYPTO 2011: 31st Annual Cryptology Conference*, Proceedings 31, 505-524. https://doi.org/10.1007/978-3-642-22792-9_29

Asharov, G., Jain, A., López-Alt, A., Tromer, E., Vaikuntanathan, V. ve Wichs, D. (2012). Multiparty computation with low communication, computation and interaction via threshold FHE. *In Advances in Cryptology–EUROCRYPT 2012: 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques*, Proceedings 31, 483-501. https://doi.org/10.1007/978-3-642-29011-4_29

ÖZGEÇMİŞ

Ad-Soyad : Serra SAZOĞLU

ÖĞRENİM DURUMU:

- **Lisans :** 2020, Sakarya Üniversitesi, Fen Edebiyat Fakültesi, Matematik
- **Yüksek Lisans :** Devam ediyor, Sakarya Üniversitesi, Matematik, Cebir ve Sayılar Teorisi