

T.C.
TRAKYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

**MAKSİMUM UZAKLIKTA AYRILABİLEN YARI İNVOLUTİF MATRİSLER
İÇİN TASARIM TEKNİKLERİ**

ERCAN ŞAHİN

YÜKSEK LİSANS TEZİ

HESAPLAMALI BİLİMLER ANABİLİM DALI

Tez Danışmanı: Dr. Öğr. Üyesi Gökhan TUNCAY

EDİRNE-2025

ERCAN ŞAHİN 'ın hazırladığı “**MAKSİMUM UZAKLIKTA AYRILABİLEN YARI İNVOLUTİF MATRİSLER İÇİN TASARIM TEKNİKLERİ**” başlıklı bu tez, tarafımızca okunmuş, kapsam ve niteliği açısından Hesaplamalı Bilimler Anabilim Dalında bir **Yüksek Lisans tezi** olarak kabul edilmiştir.

Jüri Üyeleri

İmza

Dr. Öğr. Üyesi Gökhan TUNCAY

.....

Prof. Dr. Muharrem Tolga SAKALLI

.....

Dr. Öğr. Üyesi Bora ASLAN

.....

Tez Savunma Tarihi: 04.07.2025

Bu tezin Yüksek Lisans tezi olarak gerekli şartları sağladığını onaylarım.

İmza

Tez Danışmanı

Dr. Öğr. Üyesi Gökhan TUNCAY

.....

Trakya Üniversitesi Fen Bilimleri Enstitüsü onayı

.....

Doç. Dr. Filiz UMAROĞULLARI
Fen Bilimleri Enstitüsü Müdürü

T.Ü. FEN BİLİMLERİ ENSTİTÜSÜ
HESAPLAMALI BİLİMLER YÜKSEK LİSANS PROGRAMI
DOĞRULUK BEYANI

Trakya Üniversitesi Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada, tüm verilerin bilimsel ve akademik kurallar çerçevesinde elde edildiğini, kullanılan verilerde tahrifat yapılmadığını, tezin akademik ve etik kurallara uygun olarak yazıldığını, kullanılan tüm literatür bilgilerinin bilimsel normlara uygun bir şekilde kaynak gösterilerek ilgili tezde yer aldığını ve bu tezin tamamı ya da herhangi bir bölümünün daha önceden Trakya Üniversitesi ya da farklı bir üniversitede tez çalışması olarak sunulmadığını beyan ederim.

04.07.2025
Ercan ŞAHİN

İmza

Yüksek Lisans Tezi

Maksimum Uzaklıkta Ayrılabilen Yarı İnvolutif Matrisler İçin Tasarım Teknikleri

T.Ü. Fen Bilimleri Enstitüsü

Hesaplamalı Bilimler Anabilim Dalı

ÖZET

Doğrusal dönüşüm katmanları, modern blok şifreleme algoritmalarında yayılma ilkesini gerçekleştiren temel yapı taşlarıdır. Maksimum Uzaklıkta Ayrılabilen (MDS) matrisler, maksimum dal sayısı ile güçlü yayılım özellikleri sundukları için bu katmanların tasarımında tercih edilirler. Öte yandan, tersi kendisine eşit (involutif) olan doğrusal dönüşüm katmanları, şifreleme ve şifre çözme işlemlerinin aynı maliyetle uygulanmasına olanak tanıyarak avantaj sağlamaktadırlar. Bir involutif matrisin, $c \in \mathbb{F}_{2^m} \setminus \{0, 1\}$ skaleri ile çarpımına eşit olan yarı involutif MDS matrisler ise ters alma işlemlerinin yalnızca basit matris çarpımlarıyla gerçekleştirilebilmesini sağlayarak daha geniş bir tasarım alanı sunmaktadır. Bu tez çalışmasında, $\mathbb{F}_{2^3}$ ve $\mathbb{F}_{2^4}$ sonlu cisimlerinde tanımlı indirgenemez polinomlar kullanılarak involutif MDS matrislerden yarı involutif MDS matrislerin elde edilmesine olanak tanıyan cebirsel bir yöntem önerilmektedir. Ayrıca, geliştirilen matrislerin donanımsal uygulanabilirliğini artırmak amacıyla dört farklı XOR optimizasyon algoritması ile karşılaştırmalı analizleri gerçekleştirilmiştir.

Yıl : 2025

Sayfa Sayısı : 103

Anahtar Kelimeler : Yarı involutif matrisler, MDS matrisler, doğrusal dönüşüm katmanı, XOR sayısı, küresel optimizasyon teknikleri

Master of Science Thesis
Design Techniques for Maximum Distance Separable Semi Involutory Matrices
Trakya University Institute of Natural Sciences
Department of Computational Sciences

ABSTRACT

Linear transformation layers are the fundamental building blocks that implement the diffusion principle in modern block cipher algorithms. Maximum Distance Separable (MDS) matrices are preferred in the design of these layers since they offer strong diffusion properties with the maximum branch number. On the other hand, linear transformation layers that are involutory, equal to their inverses, provide an advantage by allowing encryption and decryption operations to be implemented at the same cost. Semi-involutory MDS matrices, which are equal to the multiplication of an involutory matrix by a scalar $c \in \mathbb{F}_{2^m} \setminus \{0, 1\}$, offer a broader design space by enabling the inversion process to be carried out through simple matrix multiplications. In this thesis, an algebraic method is proposed that enables the construction of semi-involutory MDS matrices from involutory MDS matrices using irreducible polynomials defined over the finite fields $\mathbb{F}_{2^3}$ and $\mathbb{F}_{2^4}$. Additionally, comparative analyses with four different XOR optimization algorithms were conducted to enhance the hardware implementability of the developed matrices.

Year : 2025

Number of Pages : 103

Keywords : Semi-involutory matrices, MDS matrices, linear transformation layer, XOR count, global optimization techniques

TEŞEKKÜR

Öğrenim sürecim boyunca desteklerini esirgemeyen, bilgi ve tecrübesiyle bana yol gösteren değerli danışman hocam, Sayın Dr. Öğr. Üyesi Gökhan TUNCAY'a, çalışmanın matematiksel altyapısının oluşturulmasında verdiği destekler için Sayın Dr. Öğr. Üyesi Gülsüm Gözde YILMAZGÜÇ BAL'a, bana her zaman destek olan ve beni bugünlere getiren aileme, her zaman yanımda olan Çağla'ya, teşekkür ederim...

Ercan ŞAHİN

İÇİNDEKİLER

ÖZET	iv
ABSTRACT	v
TEŞEKKÜR	vi
SİMGELER DİZİNİ	ix
KISALTMALAR DİZİNİ	x
ŞEKİLLER DİZİNİ	xi
ÇİZELGELER DİZİNİ	xii
1 GİRİŞ	1
1.1. Kriptoloji	1
1.2. Kriptografiye Genel Bakış	1
1.3. Kriptoanalize Genel Bakış	2
1.4. Modern Kriptografik Algoritmalar	3
1.5. Blok Şifreler	5
1.6. Tezin Kapsamı, Dayanağı ve Sağladığı Katkılar	7
2 MATEMATİKSEL TEMELLER ve TANIMLAR	9
3 OPTİMİZASYON YÖNTEMLERİ	20
3.1. Yerel Optimizasyon	23
3.1.1. Doğrudan XOR Sayısı	23
3.1.2. Sıralı XOR Sayısı	24
3.2. Küresel Optimizasyon	25
3.2.1. En Kısa Doğrusal Düz Çizgi Programı (SLP)	25

3.2.1.1.	Paar Algoritması	26
3.2.1.2.	Boyar-Peralta (BP) Algoritması	27
3.2.1.3.	Superior Boyar-Peralta (SBP) Sezgisel Algoritması	28
4	YARI İNVOLUTİF MDS MATRİSLERİN CEBİRSEL TASARIMI	30
4.1.	Hesaplamalı Cebir Sistemi: Magma	30
4.2.	İnvolutif MDS Matrislerin İnşası	31
4.2.1.	3×3 Boyutlarındaki MDS Matrisler İçin Doğrudan İnşa Yöntemi	31
4.2.2.	4×4 Boyutlarındaki MDS Matrisler için Hibrit Yöntem	33
4.2.3.	Parametrik Formüller ve Magma ile Üretim	34
4.3.	Yarı-İnvolutif Matrislere Geçiş ve Üretim	34
4.3.1.	Yarı İnvolutif Matrislerin Magma ile Üretimi ve XOR Optimizasyonu	37
4.4.	Matris Analizi ve Optimizasyonuna Yönelik Web Tabanlı Bir Yazılım Sistemi	38
4.4.1.	Kullanılan Teknolojiler	38
4.4.2.	Sistem Mimarisi Bileşenleri	39
4.4.3.	Uygulanan XOR Optimizasyon Algoritmaları	40
5	3×3 BOYUTLARINDAKİ MATRİSLERİN ANALİZİ	43
5.1.	$\mathbb{F}_{2^3}/0xB$ Cismi Üzerinde Tanımlı Matrisler	43
5.2.	$\mathbb{F}_{2^3}/0xD$ Cismi Üzerinde Tanımlı Matrisler	47
5.3.	$\mathbb{F}_{2^4}/0x13$ Cismi Üzerinde Tanımlı Matrisler	51
5.4.	$\mathbb{F}_{2^4}/0x19$ Cismi Üzerinde Tanımlı Matrisler	56
6	4×4 BOYUTLARINDAKİ MATRİSLERİN ANALİZİ	62
6.1.	$\mathbb{F}_{2^3}/0xB$ Cismi Üzerinde Tanımlı Matrisler	62
6.2.	$\mathbb{F}_{2^3}/0xD$ Cismi Üzerinde Tanımlı Matrisler	67
6.3.	$\mathbb{F}_{2^4}/0x13$ Cismi Üzerinde Tanımlı Matrisler	71
6.4.	$\mathbb{F}_{2^4}/0x19$ Cismi Üzerinde Tanımlı Matrisler	76
7	SONUÇ ve TARTIŞMA	83
	KAYNAKLAR	86
	ÖZGEÇMİŞ	91

SİMGELER DİZİNİ

β	Dal sayısı
$\mathbb{F}_2$	Elemanları 0 ve 1 olan sonlu cisim
$\mathbb{F}_{2^m}$	Elemanları 0 ve 1 olan, m elemanlı sonlu cisim
$\oplus$	Özel VEYA (eXclusive OR)
wt	Hamming ağırlığı

KISALTMALAR DİZİNİ

AES	Advanced Encryption Standard
BP	Boyar-Peralta
DES	Data Encryption Standard
DXC	Doğrudan XOR Sayısı (Direct XOR Count)
IV	Başlatma Vektörü (Initialization Vector)
MDBL	Maksimum Uzaklıkta İkili Doğrusal (Maximum Distance Binary Linear)
MDS	Maksimum Uzaklıkta Ayrılabilir (Maximum Distance Separable)
NIST	National Institute of Standards and Technology
RC4	ARCFOUR
RIM	Temsilci İnvolutif Matris (Representative Involutory Matrix)
SAT	SAT Çözücü (Satisfiability Solver)
SBP	Superior Boyar-Peralta
SLP	En Kısa Doğrusal Düz Çizgi Programı (Shortest Linear Straight Line Program)
SPN	Yer Değiştirme-Permütasyon Ağı (Substitution-Permutation Network)
SXC	Sıralı XOR Sayısı (Sequential XOR Count)

ŞEKİLLER DİZİNİ

Şekil 1.1.	Bir Simetrik Şifreleme Algoritmasının Şematik Gösterimi	3
Şekil 1.2.	Bir Asimetrik Şifreleme Algoritmasının Şematik Gösterimi	3
Şekil 1.3.	Bir Özet Fonksiyonunun Şematik Gösterimi	4
Şekil 4.1.	Web Uygulaması Ana Sayfa Görünümü	38
Şekil 4.2.	Web Uygulamasında Matris Detay Görünümü	40
Şekil 4.3.	Optimizasyon Algoritmaları Sonuçları	41
Şekil 4.4.	Örnek Bir Optimizasyon İşleminin Adımları	42

ÇİZELGELER DİZİNİ

Çizelge 2.1.	$\mathbb{F}_{2^4}/(x^4 + x + 1)$ Cismindeki Elemanların Gösterimi	12
Çizelge 2.2.	İkili $[2n, n]$ Kodlara Ait Maksimum Dal Sayıları	17
Çizelge 2.3.	XOR Kapısı Doğruluk Tablosu	18
Çizelge 3.1.	$\mathbb{F}_{2^4}$ Cisminde İndirgenemez Polinoma Göre XOR Sayıları	22
Çizelge 4.1.	Magma’da Temel Komut Örnekleri	31
Çizelge 5.1.	3×3 Boyutlarında Üretilen Yarı İnvolutif MDS Matris Sayıları	43
Çizelge 5.2.	$\mathbb{F}_{2^3}/0xB$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	47
Çizelge 5.3.	$\mathbb{F}_{2^3}/0xD$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	52
Çizelge 5.4.	$\mathbb{F}_{2^4}/0x13$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	56
Çizelge 5.5.	$\mathbb{F}_{2^4}/0x19$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	61
Çizelge 6.1.	4×4 Boyutlarında Üretilen Yarı İnvolutif MDS Matris Sayıları	62
Çizelge 6.2.	$\mathbb{F}_{2^3}/0xB$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	67
Çizelge 6.3.	$\mathbb{F}_{2^3}/0xD$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	72
Çizelge 6.4.	$\mathbb{F}_{2^4}/0x13$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	77
Çizelge 6.5.	$\mathbb{F}_{2^4}/0x19$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı	82

BÖLÜM 1

GİRİŞ

Bilgi tarih boyunca insanlığın gelişmesinde önemli bir değer olmuştur. Her değer gibi bilginin de saldırılara karşı korunması gerekliliği vardır. Bilgi güvenliği; gizlilik, bütünlük ve erişilebilirlik prensiplerine dayanır. Kişisel veriler, askeri mesajlar veya bir sektörde operasyonun devamlılığını sağlayan sektörel sırların güvenli şekilde saklanabilmesi, değiştirilebilmesi ve aktarılabilmesi gereklidir. Teknolojinin gelişiminden önce fiziksel olarak saklanan bilgi, sadece yetkili kişiler tarafından ulaşılabilir, düzenlenebilir ve aktarılabilir durumdaydı. Bilginin güvenli bir biçimde iletilebilmesi için fiziksel olarak taşınması gerekliliği mevcuttu. Ancak, bilgisayar ve internet gibi teknolojilerin ilerlemesiyle birlikte, bilginin işleme süreçlerinde önemli değişiklikler olmuştur. İnternetin sağladığı kolaylıklarla bilginin aktarımı daha basit hale gelmiş, fakat aynı zamanda güvenilirliğinin temin edilmesi için yeni yöntemlerin geliştirilmesi zorunluluğu doğmuştur. Güvenlik amaçlarına uygun olarak, kriptoloji bilimi sürekli ilerleme kaydetmiş ve iletişimde güvenliğin sağlanmasında önemli bir role sahip olmuştur.

1.1. Kriptoloji

Kriptoloji, bilginin güvenli bir biçimde iletilmesi ve saklanması amacıyla geliştirilen yöntemleri ile bu yöntemlerin analizini konu alan bir bilim alanıdır ve temel olarak kriptografi ve kriptanaliz olmak üzere iki alt disipline ayrılmaktadır.

1.2. Kriptografiye Genel Bakış

Kriptografi, belirli bilgilerin korunmasını sağlamak amacıyla matematiksel yöntemlerin kullanılmasıyla ilgilenir ve yetkisiz kişilerin bilgilere erişimini ya da bilgilerin değiştirilmesini engelleyici önlemler içerir.

Kriptografinin temel hedefleri aşağıdaki şekilde sınıflandırılabilir (Forouzan, 2007):

- **Gizlilik:** Bilginin yalnızca yetkili bireylerce okunabilmesi gerekir.
- **Bütünlük:** Veri, izinsiz olarak değiştirilememeli; ekleme, silme ya da güncelleme gibi işlemler yalnızca yetkililer tarafından yapılabilmelidir.
- **Kullanılabilirlik:** Oluşturulan ve saklanan bilgiler, yetkili kişi veya kuruluşlar tarafından sürekli erişilebilir ve kullanılabilir olmalıdır.

Bu hedeflere ulaşabilmek adına yüksek düzeyde kriptografik protokoller geliştirilmekte ve bu protokoller, belirli kriptografik bileşenlere dayandırılmaktadır. Çeşitli matematiksel yapılardan oluşan bu bileşenler; harflerden oluşan açık metin kümesi, şifrelenmiş sembollerden oluşan şifreli metin kümesi, şifreleme işleminde kullanılabilecek olası anahtarların oluşturduğu anahtar uzayı, şifre çözme sırasında kullanılacak olası anahtarların oluşturduğu şifre çözme anahtar uzayı ile etkili şifreleme ve şifre çözme algoritmalarından oluşmaktadır (Mao, 2004).

1.3. Kriptoanalize Genel Bakış

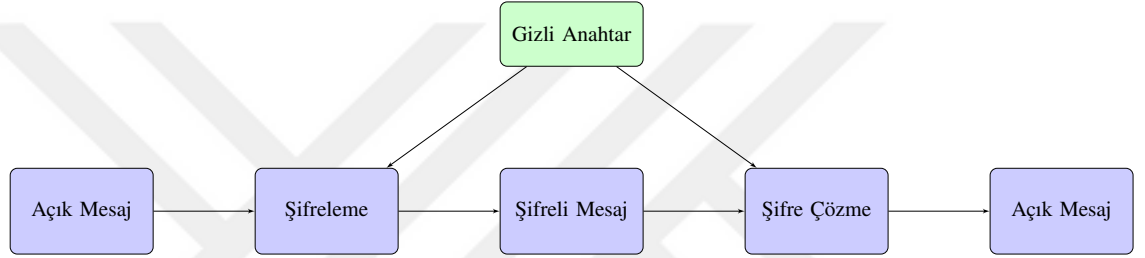
Kriptoanaliz, kullanılan kriptografik sistemlerin zayıflıklarını ortaya çıkarmayı ve güvenilirliğini test etmeyi amaçlayan bilim dalıdır. Kriptografik yapılar ne kadar güçlü olursa olsun, potansiyel saldırılara karşı test edilmeleri zorunludur. Bu bağlamda kriptoanalistler, farklı saldırı senaryoları çerçevesinde sistemleri analiz ederek kriptografik sistemlerin dayanıklılığını ölçmeyi hedeflerler. Aşağıda listelenen saldırı modelleri, hem bağımsız hem de kombinasyon halinde uygulanabilmektedir (Stinson, 2005).

- **Yalnızca Şifreli Metin Saldırısı:** Saldırgan sadece şifreli metinlere erişebilir; açık metne dair herhangi bir bilgiye sahip değildir.
- **Bilinen Açık Metin Saldırısı:** Saldırgan, bazı açık metinlerle bunlara karşılık gelen şifreli metin çiftlerine ulaşabilir.
- **Seçilmiş Açık Metin Saldırısı:** Saldırgan, seçtiği açık metinleri şifreleme cihazına göndererek karşılık gelen şifreli metinleri elde edebilir.
- **Seçilmiş Şifreli Metin Saldırısı:** Saldırgan, seçtiği şifreli metinleri şifre çözme cihazına ileterek açık metin karşılıklarını alabilir.
- **Seçilmiş Başlatma Vektörü Saldırısı:** Eğer sistem bir başlatma vektörü (IV) kullanıyorsa, saldırgan IV'yi de kontrol ederek açık metin ve IV üzerinden şifreli metinleri oluşturabilir.

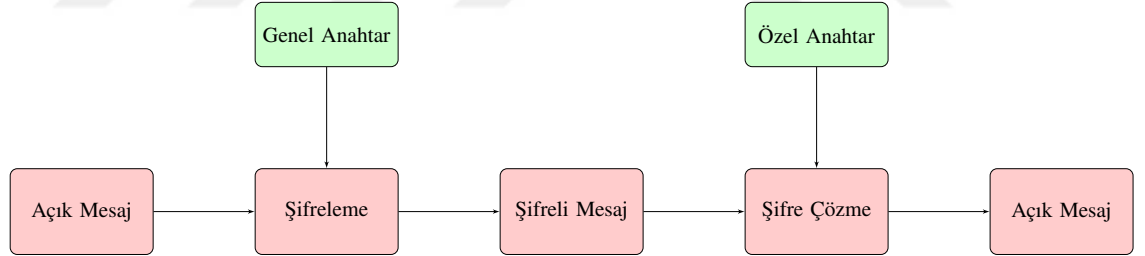
1.4. Modern Kriptografik Algoritmalar

Günümüzde kriptografik algoritmalar; Simetrik Şifreleme Algoritmaları, Asimetrik Şifreleme Algoritmaları ve Kriptografik Özet (Hash) Fonksiyonları olmak üzere üç kısma ayrılmaktadır.

Simetrik şifreleme algoritmalarında, iletişime geçen taraflar hem şifreleme hem de şifre çözme işlemlerinde aynı gizli anahtarı kullanırken; asimetrik şifreleme sistemlerinde taraflar, biri açık diğeri gizli olmak üzere bir anahtar çiftine sahiptir. Bu mimaride, verilerin şifrelenmesi açık anahtar aracılığıyla gerçekleştirilirken, şifrenin çözülmesi için yalnızca alıcıya ait olan gizli anahtar kullanılmaktadır. Şekil 1.1 ve 1.2’de, bu iki kriptografik algoritmanın temel yapısı şematik biçimde sunulmaktadır.



Şekil 1.1. Bir Simetrik Şifreleme Algoritmasının Şematik Gösterimi



Şekil 1.2. Bir Asimetrik Şifreleme Algoritmasının Şematik Gösterimi

Her iki yaklaşımda da temel güvenlik varsayımı; şifreleme sistemine dair tüm detaylar kamuya açık olsa bile, gizli anahtarın korunması koşuluyla sistemin güvenli kalması gerektiğidir. Bu düşünce, Kerckhoffs’un ilkesi olarak literatürde yer bulmuştur (Kerckhoffs, 1883).

Asimetrik kriptografi sistemleri genel olarak sayı teorisine dayalı soyut matematiksel yapılar üzerine kurulurken, simetrik yöntemler daha çok Boole fonksiyonları gibi ayrık matematiksel yapılar temelinde geliştirilmiştir. Performans açısından değerlendirildiğinde, simetrik algoritmalar asimetrik olanlara göre çok daha yüksek hızda çalışmakta ve özellikle büyük boyutlu veri kümelerinin şifrelenmesi için daha verimli çözümler sunmaktadır. Bununla birlikte, asimetrik algoritmaların en önemli avantajı,

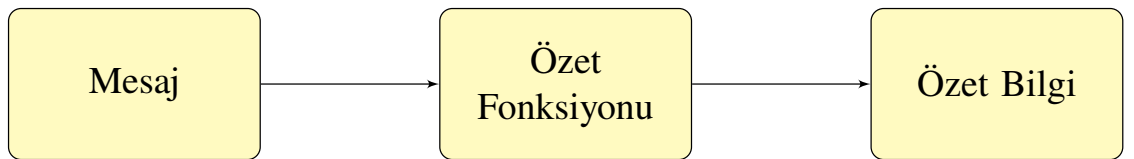
simetrik sistemlerin en büyük zaafı olan anahtar paylaşım sorununa etkili bir çözüm getirebilmeleridir.

Simetrik kriptografide kullanılan algoritmalar, şifreleme biçimlerine göre iki ana gruba ayrılmaktadır: blok şifreleme ve akış (stream) şifreleme. Blok şifreleme tekniklerinde açık metin belirli uzunluktaki bloklara ayrılarak her bir blok üzerinde işlem yapılırken, akış şifreleme yöntemleri veriyi bit ya da byte düzeyinde işler. Literatürde yaygın şekilde kullanılan blok şifreleme algoritmalarına AES (Advanced Encryption Standard) (National Institute of Standards and Technology (NIST), 2001), PRESENT (Bogdanov vd., 2007), ARIA (Kwon vd., 2003) ve Blowfish (Schneier, 1993) örnek verilebilir.

Akış şifreleme algoritmalarında özellikle kablosuz ağ güvenliğinde kullanılan RC4 (ARCFOUR) (Rivest, 1992a), HC-256 (Wu, 2004) ve Trivium (De Canniere & Preneel, 2008) en bilinen algoritmalar arasında yer almaktadır.

Asimetrik şifreleme yöntemleri arasında Diffie-Hellman anahtar değişim protokolü (Diffie & Hellman, 1976) ile RSA algoritması (Rivest, Shamir & Adleman, 1978), bu alandaki en temel örnekler olarak öne çıkmaktadır.

Özet (hash) fonksiyonları ise herhangi bir uzunluktaki bir girdiden, o girdiye özgü sabit uzunlukta ve benzersiz bir değer üretmek amacıyla kullanılan kriptografik yapılardır. Bu üretilen değer, genellikle mesajın "parmak izi" olarak nitelendirilir. Mesaj içeriğinde yapılan yalnızca 1 bitlik bir değişiklik bile, çıktı değerinin tamamen farklı olmasına yol açar. Bu fonksiyonlar tek yönlüdür; yani elde edilen özet değerinden orijinal mesajın yeniden elde edilmesi mümkün değildir. Genellikle mesajın bütünlüğünün korunup korunmadığını doğrulamak amacıyla kullanılırlar. Özet fonksiyonlarının en temel güvenlik kriteri, her farklı girdi için benzersiz bir çıktı üretme yeteneğidir. Eğer iki farklı mesaj aynı özet değerini üretirse — bu duruma çakışma (collision) denir — söz konusu özet fonksiyonu güvenli kabul edilmez ve kullanım dışı bırakılır. En çok bilinen özet fonksiyonları arasında MD5 (Rivest, 1992b), SHA-1 (Eastlake & Jones, 2001), SHA-2 (National Institute of Standards and Technology (NIST), 2015a) (özellikle SHA-256 ve SHA-512), SHA-3 (National Institute of Standards and Technology (NIST), 2015b), BLAKE2 (Aumasson, Neves, Wilcox-O'Hearn & Winnerlein, 2015) bulunmaktadır.



Şekil 1.3. Bir Özet Fonksiyonunun Şematik Gösterimi

1.5. Blok Şifreler

Claude Shannon'ın 1949 tarihli çalışmasında ortaya koyduğu karıştırma (confusion) ve yayılma (diffusion) ilkeleri, modern blok şifreleme algoritmalarının kriptografik güvenliğinin temel dayanak noktaları olarak kabul edilmektedir (Shannon, 1949). Karıştırma, şifreli metin ile kullanılan gizli anahtar arasında doğrudan bir ilişki kurulmasını zorlaştırarak, anahtarın yapısal özelliklerinin şifreli veriden türetilmesini engellemeye çalışır. Bu ilke genellikle doğrusal olmayan bileşenler olan yer değiştirme kutuları (S-boxes) aracılığıyla gerçekleştirilir. Yer değiştirme kutuları, çoğunlukla vektörel Boole fonksiyonları olarak tanımlanır. Matematiksel olarak $n - bit$ giriş ve $m - bit$ çıkışa sahip bir S-kutusu $S : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^m$ biçiminde tanımlanır ve aşağıdaki gibi gösterilir.

$$S(x) = (f_0(x), f_1(x), \dots, f_{m-1}(x))$$

Burada $f_i(x)$, S-kutusunun koordinat fonksiyonlarını temsil eder. S-kutuları, genellikle algoritmanın tek doğrusal olmayan bileşeni olduğundan, genel güvenlik düzeyini büyük ölçüde etkiler. İyi tasarlanmış bir S-kutusu, yüksek bit karışıklığı sağlar, doğrusal ve diferansiyel kriptanalize karşı direnç oluşturur.

Yayılma ise, açık metne ait istatistiksel özelliklerin şifreli metne yansımalarını önlemek amacıyla tasarlanmıştır. Bu amaçla doğrusal dönüşümler—örneğin karıştırma matrisleri, permütasyonlar veya modüler aritmetik işlemler—kullanılır. Yayılma sayesinde, açık metindeki tek bir bit değişikliği, şifreli metinde çok sayıda bitten oluşan değişime neden olur. Etkin bir doğrusal dönüşümün aşağıdaki kriterleri sağlaması beklenmektedir (Daemen & Rijmen, 2002; Feistel, 1973; Kam & Davida, 1979; Webster & Tavares, 1986).

- **Çığ Etkisi:** Girişteki tek bir bit değişikliği, çıktıdaki bitlerin yaklaşık yarısını değiştirmelidir.
- **Katı Çığ Etkisi:** Girişteki tek bir bitlik değişiklik, her çıktı bitinin %50 olasılıkla değişmesine yol açmalıdır.
- **Bütünlük Özelliği:** Her bir çıktı biti, girişin tüm bitlerine bağlı olmalıdır.
- **Dal Sayısı (Branch Number):** İkili doğrusal dönüşümlerde 0 giriş vektörü hariç giriş vektörlerinin Hamming ağırlığı ile çıkış vektörlerinin Hamming ağırlığının toplamının en düşük değeridir (Daemen & Rijmen, 2002). Bu kriter, bir blok şifrede azami yayılıma ulaşılmasını sağlayan en önemli kriptografik özelliktir. Doğrusal ve diferansiyel dal sayısı olmak üzere iki şekilde hesaplanır. Doğrusal dal sayısı bir blok

şifrenin doğrusal kriptanalize, diferansiyel dal sayısı ise diferansiyel kriptanalize karşı dayanıklılığının ölçüsüdür.

Çoğu modern algoritma, yayılım katmanında Maksimum Uzaklıkta Ayrılabilen (Maximum Distance Separable-MDS) ya da Maksimum Uzaklıkta İkili Doğrusal (Maximum Distance Binary Linear-MDBL) matrisler kullanır. Bu matrisler, şifreleme algoritmasının güvenliği, yayılım özelliği ve saldırılara karşı dayanıklılığı açısından kritik öneme sahiptir. Bu bağlamda, şifre tasarımcıları tarafından çeşitli özel matris yapıları kullanılarak güvenli ve verimli dönüşümler oluşturulmaktadır. Örneğin AES şifreleme algoritmasında kullanılan $\mathbb{F}_{2^8}$ cismi üzerinde tanımlı 4×4 dairesel MDS matris şifrenin doğrusal dönüşüm katmanlarının çekirdeğini oluşturmaktadır.

İnvolutif özellik gösteren matrislerin tersi kendisine eşit olduğundan, şifre çözme işleminin matrisin kendisiyle yapılabilmesini sağlayarak hem donanımda hem de yazılımda simetrik şifreleme için oldukça verimli çözümler sunar. Bununla birlikte, bazı uygulamalarda XOR sayısı gibi performans kriterlerine göre alternatif matris türlerinin değerlendirilmesi gerekebilir. Bu bağlamda, yayılım katmanı olarak kullanılabilen diğer yapılardan biri de semi involutif (yarı involutif) matrislerdir. Bir B matrisinin yarı involutif olarak nitelendirilebilmesi için, involutif bir A matrisinin $c \in \mathbb{F}_{2^m} \setminus \{0, 1\}$ skaleri ile çarpımına eşit olma, diğer bir deyişle $B = c \cdot A$ koşulunu sağlaması gerekmektedir.

Diğer taraftan donanımsal uygulamalarda doğrusal dönüşümlerin verimliliği yalnızca matematiksel yapılarına değil, aynı zamanda bu dönüşümlerin gerçekleştirilmesinde ihtiyaç duyulan XOR işlemi sayısına da doğrudan bağlıdır. XOR işlemi, donanımda basit ve hızlı bir mantıksal işlem olmasına rağmen, çok sayıda XOR kapısı kullanımı devre karmaşıklığını, enerji tüketimini ve gecikmeyi artırabilir. Bu nedenle, şifreleme algoritmalarında kullanılan matrislerin sahip olduğu XOR sayısının minimize edilmesi, özellikle gömülü sistemler ve düşük kaynaklı donanımlar için büyük önem taşır. Yarı involutif matrisler, uygun yapılandırmalarla hem gerekli XOR sayısını düşürebilmekte hem de ters dönüşüm hesaplamalarında avantaj sağlayabilmektedir. Bu yönüyle, performans ve güvenlik arasında denge kurmaya çalışan modern şifreleme sistemleri için dikkat çekici bir tasarım bileşeni olarak öne çıkmaktadırlar (Boyar & Peralta, 2000; Standaert & Quisquater, 2006).

Modern blok şifreleme algoritmalarının çoğu, iteratif şifreleme yapısı olarak adlandırılan bir modeli temel alır. Bu yapı, sabit sayıdaki şifreleme turunun (round) arka arkaya uygulanması prensibine dayanır. Her bir tur, belirli matematiksel işlemleri içerir ve bu işlemler arasında S-box uygulamaları, doğrusal dönüşümler, permütasyonlar ve anahtar ile XOR'lama yer alır. Bu tür yapılarda, anahtar genişletme algoritması kullanılarak gizli anahtardan her tur için farklı bir alt anahtar türetilmesi sağlanır. Bu yöntem, her turun

aynı işlemleri uygulamasına rağmen simetrik yapıların oluşmasını engeller ve şifreleme güvenliğini artırır (Ferguson, Schneier & Kohno, 2010). Güvenlik açısından güçlü bir anahtar genişletme algoritması, alt anahtarların birbirinden bağımsız ve yüksek yayılıma sahip olmasını sağlamalıdır.

Blok şifreleme algoritmalarında sıklıkla tercih edilen iki mimari yapı ise Yer Değiştirme-Permütasyon Ağı (Substitution-Permutation Network-SPN) ve Feistel Ağı olarak bilinmektedir. SPN mimarisi, tüm veri bloğu üzerinde doğrusal olmayan yer değiştirme işlemleri (S-box), ardından doğrusal yayılma işlemleri uygulayarak her turda tüm bloğu işler. Bu yapı, AES gibi şifrelerde temel alınmıştır (Daemen & Rijmen, 2002). Buna karşılık, Feistel yapısında, veri bloğu iki alt parçaya ayrılır ve her turda yalnızca bir parça işlenirken diğer parça değişmeden kalır. Daha sonra parçalar yer değiştirir ve işlem tekrarlanır. Bu yapı, DES (National Institute of Standards and Technology (NIST), 1999) gibi klasik algoritmalarda görülür ve özellikle algoritmanın tersinirliğini (şifreleme işleminin şifre çözme işlemi ile simetrik şekilde olması) garanti altına alması açısından avantaj sağlar (Menezes, van Oorschot & Vanstone, 1996).

Bu mimarilerin her biri, performans ve güvenlik gereksinimlerine göre seçilmekte ve uygulama alanına bağlı olarak özelleştirilebilmektedir. SPN yapıları genellikle donanım uygulamalarında yüksek paralelleştirilebilirlik avantajı sunarken, Feistel yapıları tersinirlik özelliği nedeniyle yazılım uygulamalarında daha esnek kullanılabilmektedir.

1.6. Tezin Kapsamı, Dayanağı ve Sağladığı Katkılar

Bu tez çalışmasında, yarı involutif matrislerin cebirsel ve yapısal özellikleri matematiksel temelleriyle incelenmekte; bu yapıların modern blok şifreleme algoritmalarındaki rolü detaylı bir şekilde değerlendirilmektedir. Çalışma kapsamında, involutif matrisler kullanılarak yarı involutif matrislerin oluşturulabildiği bir teknik önerilmiş ve bu teknik sayesinde $\mathbb{F}_{2^3}$ ile $\mathbb{F}_{2^4}$ cisimlerinde tanımlı indirgenemez polinomlar kullanılarak 3×3 ve 4×4 boyutlarındaki yarı involutif matrisler sistematik olarak elde edilebilmiştir. Elde edilen bu matrisler üzerinde, geliştirilen özel amaçlı yazılım aracılığıyla donanımsal verimliliği doğrudan etkileyen XOR sayısı açısından optimizasyon çalışmaları da gerçekleştirilmiştir.

Tez yedi bölümden oluşmaktadır. İlk bölümde, konunun anlaşılabilirliği açısından kriptolojiye dair temel kavramlara yer verilmiş; ikinci bölümde, kullanılan cebirsel yapıların ve geliştirilen tekniğin matematiksel çerçevesi tanımlanmıştır. Üçüncü bölümde, doğrusal dönüşümlerin donanımda verimli uygulanabilmesi için kullanılan XOR optimizasyon yöntemleri sunulmuştur. Dördüncü bölüm, yarı involutif matris üretiminde kullanılan tekniğin algoritmik yapısını ve teorik dayanaklarını ayrıntılı biçimde ele

almaktadır. Beşinci bölümde 3×3 , altıncı bölümde ise 4×4 boyutundaki matrislere ilişkin deneysel sonuçlara, elde edilen matrislere ve bu matrislerin optimizasyon analizlerine yer verilmiştir. Sonuç ve tartışma bölümünde ise elde edilen bulgular genel bir değerlendirme çerçevesinde ele alınarak yarı involutif matrislerin blok şifre tasarımında kullanımı tartışılmaktadır.

Bu bağlamda çalışma, yarı involutif matrislerin sistematik olarak üretilebilmesine olanak tanıyan teknikleri hem kuramsal hem de uygulamalı boyutlarıyla ortaya koymaktadır. Geliştirilen yöntem, belirli cebirsel koşullar altında 3×3 ve 4×4 boyutlu yarı involutif matrislerin yapılandırılmasını mümkün kılarken, bu matrislerin donanımda uygulanabilirliğini artırmak amacıyla doğrusal dönüşümler üzerindeki XOR optimizasyonları da ayrıntılı biçimde ele alınmıştır. Elde edilen sonuçlar, yarı involutif yapıların yalnızca teorik açıdan değil, aynı zamanda donanım verimliliği ve kriptografik özellikler bakımından da blok şifre tasarımına katkı sağlayabilecek nitelikte olduğunu göstermektedir.

BÖLÜM 2

MATEMATİKSEL TEMELLER ve TANIMLAR

Bu bölümde, tez çalışmasında geliştirilen tekniklerin anlaşılabilmesi için gerekli olan temel cebirsel yapılar ve kavramsal tanımlar sunulmaktadır. Özellikle sonlu cisimler üzerinde tanımlı matris dönüşümleri, involutif ve yarı involutif matrislerin özellikleri, bu yapıların kare alma ve skaler çarpım ilişkileriyle karakterize edilmesi gibi konular ele alınacaktır. Ayrıca, kullanılan indirgenemez polinomlar, XOR sayısı metriği ve çeşitli özel matris tanımlarına dair öncül teoremler detaylı biçimde açıklanacaktır. Burada sunulan tanım, teorem ve önermelerin kanıtlarına, ilgili literatürdeki (Barreto, 2000; Biham, 1994; Daemen & Rijmen, 2002; Davis, 1979; Forouzan, 2007; Khoo, Peyrin, Poschmann & Yap, 2014; Kwon, Sung, Song & Park, 2005; Lidl & Niederreiter, 1997; Pehlivanoglu, Sakallı, Akleyek, Duru & Rijmen, 2018; Stallings, 2013; Stinson & Paterson, 2019) kaynaklarından ulaşılabilmektedir.

Tanım 2.1 (Cisim). Bir küme $\mathbb{F}$, üzerinde iki işlem (toplama $+$ ve çarpma $\cdot$) tanımlı ve aşağıdaki koşulları sağlıyorsa, bu yapıya cisim adı verilmektedir.

1. $(\mathbb{F}, +)$ kümesi, toplama işlemi altında aşağıdaki özellikleri sağlamaktadır.

- Toplama işlemi kapalıdır: $\forall a, b \in \mathbb{F}, a + b \in \mathbb{F}$
- Toplama işlemi değişmelidir: $a + b = b + a$
- Toplama işleminin birleşme özelliği vardır: $(a + b) + c = a + (b + c)$
- Toplama işleminin birim elemanı vardır: $\exists 0 \in \mathbb{F}$ öyle ki, $a + 0 = a$
- Her elemanın toplama işlemine göre ters elemanı vardır: $\forall a \in \mathbb{F}, \exists (-a) \in \mathbb{F}$ öyle ki, $a + (-a) = 0$

2. $(\mathbb{F} \setminus \{0\}, \cdot)$ kümesi, çarpma işlemi altında aşağıdaki özellikleri sağlamaktadır.

- Çarpma işlemi kapalıdır: $\forall a, b \in \mathbb{F} \setminus \{0\}, a \cdot b \in \mathbb{F} \setminus \{0\}$
- Çarpma işlemi değişmelidir: $a \cdot b = b \cdot a$
- Çarpma işleminin birleşme özelliği vardır: $(a \cdot b) \cdot c = a \cdot (b \cdot c)$
- Çarpma işleminin birim elemanı vardır: $\exists 1 \in \mathbb{F}, 1 \neq 0$, öyle ki, $a \cdot 1 = a$
- Sıfır hariç her elemanın çarpma işlemine göre ters elemanı vardır: $\forall a \in \mathbb{F} \setminus \{0\}, \exists a^{-1} \in \mathbb{F}$ öyle ki, $a \cdot a^{-1} = 1$

3. Çarpma, toplama işlemi üzerine dağılım özelliğine sahiptir:

$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{ve} \quad (b + c) \cdot a = b \cdot a + c \cdot a, \quad \forall a, b, c \in \mathbb{F}.$$

Tanım 2.2 (Sonlu Cisim). Eleman sayısı sonlu olan her cisim yapısına sonlu cisim denir. Bu tür yapılar genellikle p asal bir sayı ve n pozitif bir tam sayı olmak üzere $\mathbb{F}_{p^n}$ ya da $\mathbb{F}_{p^n}$ (Galois Cismi) biçiminde gösterilir. Bu yapıdaki toplam eleman sayısı, cismin *mertebesi* ya da *derecesi* olarak adlandırılır.

Örnek 2.1. Aşağıda bazı sonlu cisim örnekleri verilmiştir:

- $\mathbb{F}_2$: Eleman kümesi $\{0, 1\}$ olan, iki elemanlı sonlu cisim.
- $\mathbb{F}_{2^3}$: $\{0, 1\}$ elemanlarına ve $2^3 = 8$ farklı birleşim biçimine sahip üç bitlik genişletilmiş sonlu cisim.
- $\mathbb{F}_{2^4}$: $\{0, 1\}$ elemanlarına ve $2^4 = 16$ farklı birleşim biçimine sahip dört bitlik genişletilmiş sonlu cisim.
- $\mathbb{F}_4$: Eleman kümesi $\{0, 1, 2, 3\}$ olan, dört elemanlı sonlu cisim.

Tanım 2.3 (Polinom). $n \geq 0$ tam sayısı olmak üzere, bir polinom aşağıdaki şekilde ifade edilir:

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 = \sum_{i=0}^n a_i x^i \quad (2.1)$$

Burada, a_i katsayılar S kümesine ait elemanlar olup, $a_n \neq 0$ koşulunu sağlamaktadır. Derecesi sıfır olan polinomlar sabit polinom olarak adlandırılır ve bu polinomlar, doğrudan S kümesinin bir elemanını temsil eder. Eğer en yüksek dereceli katsayı $a_n = 1$ ise, bu polinom monik polinom olarak isimlendirilir. Katsayıların oluşturduğu küme bir halka veya cisim olduğunda, sırasıyla polinom tabanlı halka veya polinom tabanlı cisim terimleri kullanılır.

Tanım 2.4 (İndirgenebilir ve İndirgenemez Polinomlar). Bir $f(x)$ polinomu, derecesi $\deg(f) > 1$ olmak üzere, eğer iki polinom $g(x)$ ve $h(x)$ var ise ve $\deg(g) < \deg(f)$, $\deg(h) < \deg(f)$ koşullarını sağlayarak $f(x) = g(x) \cdot h(x)$ biçiminde yazılabiliyorsa, $f(x)$ indirgenebilir; aksi takdirde indirgenemez polinom olarak tanımlanır.

Teorem 2.1. Bir $\mathbb{F}$ cismi üzerinde tanımlı ve derecesi birden büyük bir $f(x)$ polinomu için, $\mathbb{F}[x]/(f(x))$ kümesi toplama ve çarpma işlemleriyle bir polinom tabanlı halka oluşturur. Ayrıca, eğer $f(x)$ indirgenemez bir polinom ise, $\mathbb{F}[x]/(f(x))$ yapısı polinom tabanlı bir cisimdir.

Bu teorem uyarınca $\mathbb{F}[x]/(f(x))$, $f(x)$ polinomuna göre modüler indirgeme işlemi sonrasında elde edilen yapı olarak yorumlanabilir.

Tez çalışmasında, $\mathbb{F}_{2^m}/p(x)$ biçiminde ifade edilen sonlu cismi tanımlamak için kullanılan indirgenemez polinom, gösterim kolaylığı sağlamak amacıyla onaltılık (hexadecimal) biçimde sunulmuştur. Bu yaklaşımda, ilgili polinomun katsayıları ikili (binary) vektör dizisine dönüştürülerek gösterilir ve bu ikili gösterimin onaltılık karşılığı kullanılır. Örneğin, $\mathbb{F}_{2^3}/(x^3 + x^2 + 1)$ indirgenemez polinomu, (1101) ikili dizisi ile temsil edilir ve bu ifade onaltılık sistemde 0xD olarak yazılır. Benzer şekilde:

$$\begin{aligned}\mathbb{F}_{2^4}/(x^4 + x + 1) &\Rightarrow (00010011) \Rightarrow 0x13, \\ \mathbb{F}_{2^4}/(x^4 + x^3 + 1) &\Rightarrow (00011001) \Rightarrow 0x19\end{aligned}$$

şeklinde dönüşümler yapılmaktadır.

Bu indirgenemez polinomların yanı sıra, α ile gösterilen bir ilkel eleman kullanılarak oluşturulan $\mathbb{F}_{2^m}/p(x)$ sonlu cisminin elemanları da çalışma süresince onaltılık biçimde gösterilmektedir. Örnek olarak, Çizelge 2.1, $\mathbb{F}_{2^4}/(x^4 + x + 1)$ cismine ait elemanları ve bu elemanların α cinsinden gösterimlerini içermektedir.

Çizelge 2.1. $\mathbb{F}_{2^4}/(x^4 + x + 1)$ Cismindeki Elemanların Gösterimi

Onaltılık Gösterim	Cisim Elemanı (α cinsinden)
1_h	$\alpha^0 = 1$
2_h	$\alpha^1 = \alpha$
3_h	$\alpha^4 = \alpha + 1$
4_h	α^2
5_h	$\alpha^8 = \alpha^2 + 1$
6_h	$\alpha^5 = \alpha^2 + \alpha$
7_h	$\alpha^{10} = \alpha^2 + \alpha + 1$
8_h	α^3
9_h	$\alpha^{14} = \alpha^3 + 1$
A_h	$\alpha^9 = \alpha^3 + \alpha$
B_h	$\alpha^7 = \alpha^3 + \alpha + 1$
C_h	$\alpha^6 = \alpha^3 + \alpha^2$
D_h	$\alpha^{13} = \alpha^3 + \alpha^2 + 1$
E_h	$\alpha^{11} = \alpha^3 + \alpha^2 + \alpha$
F_h	$\alpha^{12} = \alpha^3 + \alpha^2 + \alpha + 1$

$\mathbb{F}_{2^m}/p(x)$ sonlu cisminin elemanları üzerinde toplama işlemi, mod 2 alınarak (ikili toplama) gerçekleştirilir. Çarpma işlemi ise, oluşan polinomun $p(x)$ polinomuna göre indirgenmesiyle elde edilir.

Örnek 2.2. $\mathbb{F}_{2^4}/0x13$ sonlu cisminin D_h ve E_h elemanlarının toplamı ve çarpımı aşağıdaki gibidir.

Öncelikle $D_h = \alpha^3 + \alpha^2 + 1$ ve $E_h = \alpha^3 + \alpha^2 + \alpha$ olduğundan toplama işlemi:

$$\alpha^3 + \alpha^2 + 1 + (\alpha^3 + \alpha^2 + \alpha) = \alpha^3 + \alpha^2 + 1 + \alpha^3 + \alpha^2 + \alpha = \alpha + 1$$

yani $D_h + E_h = 3_h$ olarak bulunur.

Çarpma işlemi ise şu şekilde hesaplanır:

$$(\alpha^3 + \alpha^2 + 1) \times (\alpha^3 + \alpha^2 + \alpha) = \alpha^6 + \alpha^5 + \alpha^4 + \alpha^5 + \alpha^4 + \alpha^3 + \alpha^3 + \alpha^2 + \alpha$$

Burada $\alpha^6 = \alpha^3 + \alpha^2$ olduğundan,

$$\alpha^3 + \alpha^2 + \alpha^5 + \alpha^4 + \alpha^5 + \alpha^4 + \alpha^3 + \alpha^3 + \alpha^2 + \alpha = \alpha^3 + \alpha$$

yani çarpım A_h elemanına eşittir.

Tanım 2.5 (Doğrusal Dönüşüm). Elemanları

$$A : (\{0, 1\}^m)^n \rightarrow (\{0, 1\}^m)^n$$

şeklinde tanımlanan bir fonksiyon, (2.2) ifadesindeki gibi doğrusal dönüşüm olarak adlandırılır:

$$A(x) = A \cdot x = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (2.2)$$

Burada (a_{ij}) matrisi doğrusal dönüşümün katsayılarını gösterir. Eğer $m = 1$ alınır ve bu katsayılar $\mathbb{F}_2$ cisminden seçilirse, dönüşüm ikili doğrusal dönüşüm olarak adlandırılır.

Tanım 2.6 (Hamming Ağırlığı). $wt(C)$ ifadesiyle gösterilen Hamming ağırlığı, bir C kod kelimesinde bulunan 1 değerlerinin sayısı olarak tanımlanır. Örneğin, $C = \{0, 1, 0, 1, 0, 0, 1\}$ kod kelimesi için Hamming ağırlığı 3'tür.

Tanım 2.7 (Hamming Uzaklığı). $\mathbb{F}_{2^m}$ cismi elemanlarından oluşan iki n boyutlu kod kelimesi arasındaki Hamming uzaklığı, bu kelimelerde aynı konumda yer alan fakat farklı değerlere sahip bileşenlerin sayısıdır. Örneğin, $C_1 = \{1, 0, 1, 1, 1, \mathbf{1}, \mathbf{1}\}$ ve $C_2 = \{1, 0, 1, 1, 1, \mathbf{0}, \mathbf{0}\}$ kod kelimeleri arasındaki Hamming uzaklığı 2 olarak hesaplanır.

Tanım 2.8 (Yayılım Katmanındaki Dal Sayısı). Girdi farkı vektörü $x \in \mathbb{F}_{2^m}^n \setminus \{0\}$, girdi farkı vektörünün doğrusal dönüşüm altındaki çıktısı $L(x)$ ve Hamming ağırlığı $wt(x)$ ile tanımlandığında; yayılım katmanındaki dal sayısı, diferansiyel ve doğrusal dal sayısı olmak üzere iki şekilde hesaplanır.

Tanım 2.9 (Diferansiyel Dal Sayısı). Diferansiyel dal sayısı (differential branch number), β_d ile gösterilmekte olup diferansiyel kriptanalize karşı yayılım ölçüsüdür ve girdi farkı vektörünün çıktıya etkisini ölçer.

$$\beta_d(A) = \min_{x \neq 0} (wt(x) + wt(L(x))) \quad (2.3)$$

Tanım 2.10 (Doğrusal Dal Sayısı). Doğrusal dal sayısı (linear branch number), β_l ile ifade edilmekte olup doğrusal kriptanalize karşı yayılım ölçüsüdür ve girdi maskesinin çıktıya etkisini ölçer.

$$\beta_l(A) = \min_{x \neq 0} (wt(x) + wt(L^T(x))) \quad (2.4)$$

Tanım 2.9 ve 2.10 doğrultusunda, bir ikili doğrusal dönüşümün dal sayısı (branch number), sıfırdan farklı tüm giriş vektörleri için giriş ve çıkış vektörlerinin Hamming ağırlıkları toplamının alabileceği en küçük değere karşılık gelmektedir. Bununla birlikte, bir doğrusal dönüşümün dal sayısı, birbirini takip eden iki döngüde bulunan aktif S-kutularının en düşük sınırıdır.

Tanım 2.11 (Birim Matris). Ana köşegen üzerinde yer alan elemanları 1, diğer tüm konumlardaki elemanları ise 0 olan $n \times n$ boyutundaki kare matrisler birim matris olarak adlandırılır ve genellikle I_n veya I sembolleri ile ifade edilir.

Tanım 2.12 (Köşegen Matris). Bir matrisin tüm satır ve sütun indisleri $i \neq j$ için $a_{ij} = 0$ olacak şekilde tanımlandığı durumda, bu matris köşegen (diyagonal) matris olarak adlandırılır. Bu tür matrislerde yalnızca ana köşegen üzerindeki elemanlar sıfırdan farklı olabilir.

Tanım 2.13 (Simetrik Matris). Gerçek sayılar kümesinde tanımlı bir A matrisi için $A^T = A$ eşitliği sağlanıyorsa, bu matris simetrik matris olarak adlandırılır.

Tanım 2.14 (Involutif Matris). Bir matris $A \in GL(n, \mathbb{F}_{2^k})$, kendisinin çarpımsal tersi olacak şekilde tanımlanabiliyorsa, yani

$$A^2 = I_n \quad (2.5)$$

eşitliği sağlanıyorsa, A matrisine involutif (tersi kendisine eşit) matris denir. Bu tür matrisler, doğrusal dönüşümlerin terslenebilirliğini aynı maliyetle sağlayabildikleri için blok şifreleme algoritmalarında tercih edilmektedir.

Tanım 2.15 (Semi-involutif Matris). $A \in GL(n, \mathbb{F}_{2^k})$ matrisinin involutif olmaması durumunda, eğer $c \in \mathbb{F}_{2^k} \setminus \{0, 1\}$ için aşağıdaki koşul sağlanıyorsa:

$$A^2 = c^2 \cdot I_n \quad (2.6)$$

bu durumda A matrisine semi-involutif (yarı involutif) matris denir. Yarı involutif matrisler, involutif matrislerin genelleştirilmiş bir biçimidir ve terslenebilirliğe benzer kolaylıklar sunarken, daha geniş bir tasarım uzayına olanak tanır.

Tanım 2.16 (Ortogonal Matris). $A \in \mathbb{F}^{n \times n}$ bir kare matris olmak üzere, eğer A matrisinin kendisiyle transpozunun çarpımı birim matris veriyorsa:

$$A \cdot A^T = I_n \quad (2.7)$$

bu durumda A matrisine ortogonal matris adı verilir.

Tanım 2.17 (Singüler Olmayan Matris). $A \in \mathbb{F}^{n \times n}$ matrisinin çarpımsal tersi olan bir $B \in \mathbb{F}^{n \times n}$ matrisi mevcutsa, yani:

$$AB = BA = I_n \quad (2.8)$$

eşitliği sağlanıyorsa, A matrisine singüler olmayan (ya da nonsingular) matris denir.

Tanım 2.18 (Dairesel Matris). Bir dairesel matris (circulant matrix), ilk satırındaki elemanların sağa doğru dairesel olarak kaydırılmasıyla elde edilen satırlardan oluşur. $a_{ij} = x_0, x_1, \dots, x_{n-1}$ olmak üzere, bir dairesel matris C aşağıdaki biçimde tanımlanır (Davis, 1979):

$$C = \text{circ}(x_0, x_1, \dots, x_{n-1}) = \begin{bmatrix} x_0 & x_1 & x_2 & \cdots & x_{n-1} \\ x_{n-1} & x_0 & x_1 & \cdots & x_{n-2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & x_3 & \cdots & x_0 \end{bmatrix} \quad (2.9)$$

Tanım 2.19 (Hadamard Matris). A_0 ve A_1 olmak üzere, her biri $2^{n-1} \times 2^{n-1}$ boyutlarında Hadamard matrisleri olsun. Bu durumda, $2^n \times 2^n$ boyutunda oluşturulan Hadamard matrisi H , aşağıdaki şekilde tanımlanır ve bu yapıya Hadamard matris denir (Hedayat & Wallis, 1978):

$$H = \text{had}(A_0, A_1) = \begin{bmatrix} A_0 & A_1 \\ A_1 & A_0 \end{bmatrix} \quad (2.10)$$

Tanım 2.20 (Genelleştirilmiş Hadamard Matris). $\mathbb{F}_{2^m}$ sonlu cisim üzerinde tanımlı olmak üzere, $b_1 \in \mathbb{F}_{2^m} \setminus \{0\}$ bir katsayı ve $a_0, a_1 \in \mathbb{F}_{2^m}$ olmak üzere, Tanım 2.19'deki H matrisinin 2×2 boyutundaki genelleştirilmiş versiyonu aşağıdaki gibi tanımlanır (Pehlivanoğlu vd., 2018):

$$GH = \text{Ghad}(a_0, a_1; b_1) = \begin{bmatrix} a_0 & a_1 b_1 \\ a_1 b_1^{-1} & a_0 \end{bmatrix} \quad (2.11)$$

Benzer şekilde, genelleştirilmiş Hadamard matrisinin 4×4 boyutundaki formu aşağıdaki gibidir (Pehlivanoğlu vd., 2018):

$$GH = \text{Ghad}(a_0, a_1; b_1, a_2; b_2, a_3; b_3) = \begin{bmatrix} a_0 & a_1 b_1 & a_2 b_2 & a_3 b_3 \\ a_1 b_1^{-1} & a_0 & a_3 b_1^{-1} b_2 & a_2 b_1^{-1} b_3 \\ a_2 b_2^{-1} & a_3 b_2^{-1} & a_0 & a_1 b_2^{-1} b_3 \\ a_3 b_3^{-1} & a_2 b_3^{-1} b_1 & a_1 b_3^{-1} b_2 & a_0 \end{bmatrix} \quad (2.12)$$

Tanım 2.21 (Singleton Sınırı). $\mathbb{F}_{2^m}$ cismi üzerinde tanımlı bir $[n, k, d]$ parametrelili kod için;

k : $(\mathbb{F}_{2^m})^n$ vektör uzayının boyutunu,

n : Bu uzaydan seçilen her bir vektörün bileşen sayısını,

d : Seçilen herhangi iki kod vektörü arasındaki en küçük Hamming uzaklığını ifade eder.

Bu bağlamda, doğrusal $[n, k, d]$ kodlarının sağlamlaştırmakla yükümlü olduğu $d \leq n - k + 1$ eşitsizliği, Singleton sınırı olarak adlandırılmaktadır (MacWilliams & Sloane, 1977).

Tanım 2.22 (MDS Kodlar). Bir doğrusal $[n, k, d]$ kodu olan C , eğer $d = n - k + 1$ koşulunu sağlıyorsa, bu durumda C maksimum uzaklıkta ayrılabilen (Maximum Distance Separable - MDS) bir kod olarak tanımlanır.

Tanım 2.23 (Üreteç Matrisi). $[n, k, d]$ biçiminde bir doğrusal kod için, G matrisine üreteç matrisi denir ve bu matris $k \times n$ boyutunda olup, satırları kod uzayının tabanını oluşturan vektörlerden meydana gelir.

Teorem 2.2. Eğer bir doğrusal kodun üreteç matrisi $G = [I, A]$ biçimindeyse ve bu kod $d = n - k + 1$ Singleton sınırını sağlıyorsa, bu durumda A matrisi bir MDS matristir.

Tanım 2.24 (MDS Matrisler). MDS matrisler, MDS kodlardan türetilir ve aşağıdaki temel özellikleri taşırlar (Pehlivanoglu vd., 2018):

- i) $n \times n$ boyutundaki bir matrisin tüm kare alt matrislerinin determinantları sıfırdan farklı ise, bu matris bir MDS matristir.
- ii) MDS bir matrisin tüm elemanları sıfırdan farklı olmalıdır.
- iii) Eğer bir $n \times n$ matris MDS özelliği taşıyorsa, bu matrisin diferansiyel dal sayısı ile doğrusal dal sayısı aynıdır ve dal sayısı $n + 1$ 'dir.
- iv) MDS bir matrisin transpozunu da MDS özelliğini korur.
- v) MDS bir matrisin satır veya sütunları, $\mathbb{F}_{2^m}$ cismindeki sıfırdan farklı bir eleman ile çarpıldığında, ortaya çıkan yeni matris de yine MDS matristir.

Tanım 2.25 (MDBL Kodlar). Elemanları $\mathbb{F}_2$ cismine ait olan matrisler ikili matrisler olarak adlandırılır. $n \times n$ boyutundaki bir ikili matrisin maksimum dal sayısı, ikili $[2n, n]$ doğrusal kodların elde edebileceği en yüksek uzaklığa eşittir.

Bu doğrultuda, ikili doğrusal kodların maksimum uzaklığa sahip olabilmesi için, söz konusu kodlara ait minimum ve maksimum uzaklık sınırlarının birbirine eşit olması gerekmektedir. Bu tür kodlara, maksimum uzaklığa sahip ikili doğrusal kodlar (Maximum Distance Binary Linear Codes - MDBL) adı verilir.

Çizelge 2.2. İkili $[2n, n]$ Kodlara Ait Maksimum Dal Sayıları

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

Çizelge 2.2 incelendiğinde, $n > 18$ için $n \times n$ boyutlu MDBL matrislerin maksimum dal sayısının henüz kesin olarak belirlenemediği görülmektedir. Bu nedenle, bu boyutlardaki matrisler için hem teorik sınır hem de erişilebilir değer birlikte ele alınmalıdır. Örneğin, $[64, 32]$ doğrusal kodlarının ($n = 32$) maksimum dal sayısının erişilebilir değeri 12 iken, teorik üst sınırı 16'dır. Başka bir deyişle, 32×32 boyutundaki bir ikili matrisin ulaşılabilen en yüksek dal sayısı 12 olarak bilinmektedir.

Tanım 2.26 (Sabit Nokta Sayısı). A bir doğrusal dönüşüm ve x bu dönüşümün giriş vektörü olmak üzere, $A \cdot x = x$ eşitliği sağlanıyorsa, x vektörü bu dönüşümün sabit noktası olarak adlandırılır. Yani, doğrusal dönüşüm A eylemi sonucunda x vektörü kendisine eş bir vektöre dönüşüyorsa, x sabit noktadır (Z'aba, 2010).

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (2.13)$$

A matrisinin ve birim matris I 'nin $n \times n$ boyutunda olduğu ve giriş vektörlerinin m -bitlik elemanlar içerdiği varsayılırsa, sabit noktaların tamamı aşağıdaki gibi elde edilir:

$$(A + I)x^T = 0 \quad (2.14)$$

Bu ifade matrisin rank bilgisiyle birlikte değerlendirilerek sabit nokta sayısı şu şekilde hesaplanır:

$$SabitNokta_A = 2^{m(rank(A) - rank(A+I))} = 2^{m(n - rank(A+I))} \quad (2.15)$$

Tanım 2.27 (XOR Kapısı). Özel VEYA (eXclusive OR) olarak bilinen XOR kapısı, iki giriş aynı olduğunda 0, farklı olduğunda ise 1 çıktısı üreten bir mantık kapısıdır ve $\oplus$ sembolü ile gösterilir.

Çizelge 2.3. XOR Kapısı Doğruluk Tablosu

a	b	Y
0	0	0
0	1	1
1	0	1
1	1	0

Bu kapının Boole cebri ifadesi $Y = a'b + ab'$ şeklindedir.

Tanım 2.28 (XOR Sayısı). $XOR(\alpha)$, $\mathbb{F}_{2^m}$ cismi üzerinde tanımlı bir $p(x)$ polinomunun bir elemanı olan α ile aynı cisimde yer alan başka bir β elemanının çarpımını gerçekleştirmek için gerekli XOR kapısı sayısını belirtir.

Tanım 2.29 (Matrislerde XOR Sayısı). $\mathbb{F}_{2^m}$ cismi üzerinde tanımlı $n \times n$ boyutundaki bir M matrisinin belirli bir satırındaki XOR sayısı, aşağıdaki şekilde tanımlanır (Khoo vd., 2014):

$$M \text{ matrisinin bir satırının XOR sayısı} = (\gamma_1, \gamma_2, \dots, \gamma_k) + (n-1) \cdot m \quad (2.16)$$

Burada γ_i , satırdaki i . elemanın XOR sayısını; n , satırda sıfırdan farklı eleman

sayısını; m ise sonlu cismin mertebesini temsil eder. Tüm satırlar için bu işlem tekrarlanarak, matrisin toplam XOR sayısı elde edilir.

Tanım 2.30 (Devre Derinliği). Doğrusal bir katmanın fiziksel bir devre şeklinde uygulanması sırasında, bir giriş bitinden başlayıp bir çıkış bitine ulaşan ve üzerinde en fazla sayıda ardışık XOR kapısı bulunan yol, *kritik yol* olarak tanımlanır. Bu yol, girişten çıkışa veri akışının en uzun zaman aldığı hesaplama dizisini temsil eder. Devre derinliği, bu kritik yol üzerindeki XOR işlemlerinin sayısı ile ölçülür ve birim zamanda gerçekleştirilebilecek işlem sayısını belirleyen saat frekansı üzerinde doğrudan etkilidir. Bu nedenle, devre derinliği yalnızca alan verimliliği değil, aynı zamanda zamanlama performansı açısından da temel bir metrik olarak değerlendirilir (Li, Sun, Li, Wei & Hu, 2019).



BÖLÜM 3

OPTİMİZASYON YÖNTEMLERİ

$\mathbb{R}$ herhangi bir halka olmak üzere, $\mathcal{M}_n(\mathbb{R})$ kümesi $\mathbb{R}$ halkasından seçilen girdilerle oluşturulan tüm $n \times n$ boyutlu matrisleri ifade etmektedir. Benzer şekilde, $\mathcal{M}_n(\mathcal{M}_l(\mathbb{F}_2))$ ifadesi, her bir elemanı $\mathcal{M}_l(\mathbb{F}_2)$ kümesinden seçilen tüm $n \times n$ matrislerin oluşturduğu kümeyi tanımlar. Bu kümede yer alan herhangi bir M matrisi, $nl \times nl$ boyutlarında bir ikili matris olarak temsil edilebilir.

$\mathcal{M}_l(\mathbb{F}_2)$ içerisinde yer alan bir difüzyon matrisi, sonlu sayıda XOR kapısı kullanılarak gerçekleştirilebilir. Düşük XOR sayısı, donanım alanının verimli kullanımıyla doğrudan ilişkilidir (Khoo vd., 2014). Maksimum dal sayısı kriterine sahip olmaları nedeniyle MDS matrisler ideal difüzyon tabakaları oluştursa da, uygulama açısından yüksek maliyet doğurabilirler. Yazılım tarafında, tablo tabanlı optimizasyonlar ve bellek kısıtı olmaması nedeniyle bu maliyet büyük ölçüde göz ardı edilebilir. Ancak donanım uygulamalarında, sonlu cisim çarpımlarının gerçekleştirilmesi önemli ölçüde donanım kaynağı gerektirir. Bu nedenle, minimum XOR işlemiyle gerçekleştirilebilecek matrislerin tercih edilmesi, modern kriptografik sistemlerde önemli bir tasarım ilkesi haline gelmiştir. Genel olarak düşük Hamming ağırlığına sahip elemanlar, donanımda daha az kaynak gerektirdiğinden avantajlıdır. Bu durum, bir difüzyon matrisinin seçiminde kritik bir ölçüt olarak değerlendirilir. Örneğin, AES blok şifresinin difüzyon matrisindeki katsayılar $0x1$, $0x2$ ve $0x3$ gibi düşük maliyetli elemanlardan seçilmiştir (Daemen & Rijmen, 2002). Ancak, Khoo ve arkadaşları tarafından yapılan çalışmada bu yaklaşımın her zaman geçerli olmadığı, seçilen indirgenemez polinoma ve çarpma işleminin indirgenme biçimine bağlı olarak yüksek Hamming ağırlığına sahip bazı katsayıların oldukça düşük XOR maliyetiyle uygulanabildiği gösterilmiştir.

Aşağıda verilen üç örnek, Tanım 2.28 ve Tanım 2.29 kullanılarak, belirli bir

sonlu cisim elemanının farklı indirgenemez polinomlar altında XOR maliyetini ve 4×4 boyutundaki bir matrisin toplam XOR işlemi sayısını hesaplamayı göstermektedir.

Örnek 3.1. (b_3, b_2, b_1, b_0) , $\mathbb{F}_{2^4}/0x13$ cismi üzerinde tanımlı rastgele bir β elemanının ikili gösterimi olmak üzere, aynı cisimde tanımlı $\alpha = 8_h$ elemanı ile çarpımın XOR maliyeti şu şekilde hesaplanır:

$$(1, 0, 0, 0).(b_3, b_2, b_1, b_0) = (b_0 \oplus b_3, b_2 \oplus b_3, b_1 \oplus b_2, b_1)$$

Bu işlem sonucunda, $\mathbb{F}_{2^4}/0x13$ cisminde $\alpha = 8_h$ elemanı ile yapılan çarpım için gereken XOR sayısı 3'tür.

Örnek 3.2. (b_3, b_2, b_1, b_0) , $\mathbb{F}_{2^4}/0x19$ cismi üzerinde tanımlı rastgele bir β elemanının ikili gösterimi olmak üzere, aynı cisimde tanımlı $\alpha = 8_h$ elemanı ile çarpımın XOR maliyeti şu şekilde hesaplanır:

$$(1, 0, 0, 0).(b_3, b_2, b_1, b_0) = (b_0 \oplus b_1 \oplus b_2 \oplus b_3, b_3, b_2 \oplus b_3, b_1 \oplus b_2 \oplus b_3)$$

Bu durumda, $\mathbb{F}_{2^4}/0x19$ cisminde aynı eleman ile çarpım 6 adet XOR işlemi gerektirir.

Bu örneklerden açıkça anlaşılacağı üzere, bir elemanla çarpım işleminde ortaya çıkan XOR sayısı, tanımlı olunan sonlu cismin indirgenemez polinomuna bağlı olarak önemli ölçüde değişkenlik göstermektedir. Sim ve arkadaşlarının 2015 yılında yaptığı çalışmada, indirgenemez polinom seçiminin bu varyasyon üzerindeki etkisi kapsamlı bir şekilde incelenmiş ve elde edilen sonuçlar doğrultusunda, XOR sayılarının dağılım özelliklerinin polinomdan polinoma farklılık gösterdiği ortaya konmuştur. Bu çalışmada, söz konusu dağılımlar standart sapma ile nicel olarak değerlendirilmiş ve yüksek standart sapma değerlerinin, daha fazla sayıda düşük XOR maliyetli eleman içerdiğine, dolayısıyla optimizasyon açısından daha avantajlı matrislerin oluşturulabileceğine işaret ettiği belirtilmiştir. Çizelge 3.1, $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $0x13$, $0x19$ ve $0x1f$ indirgenemez polinomları için, tüm elemanlarla gerçekleştirilen çarpım işlemlerine karşılık gelen XOR sayıları ve bu değerlerin dağılımlarına ait standart sapma ölçümlerini özetlemektedir (Sim, Khoo, Oggier & Peyrin, 2015).

İndirgenemez polinomun XOR sayısı üzerindeki etkisinin anlaşılması, özellikle kriptografik uygulamalarda kullanılan matrislerin optimizasyonu açısından kritik bir öneme sahiptir. Çünkü sonlu cisim elemanlarının çarpımı, bu tür uygulamalarda temel işlem adımlarından biri olarak yoğun hesaplama maliyetlerine yol açabilmektedir. Bu nedenle, uygun indirgenemez polinom seçimi, yalnızca çarpma işlemlerinin karmaşıklığını azaltmakla kalmayıp, aynı zamanda donanım ve yazılım tasarımında enerji verimliliği ve hız açısından da önemli avantajlar sağlamaktadır. Sim ve arkadaşlarının

analizinde ortaya çıkan dağılım farklılıkları, optimize edilmiş algoritmaların geliştirilmesi için bir rehber niteliğindedir ve bu sayede, düşük XOR sayısına sahip elemanların belirlenmesi ile daha performanslı ve güvenli kriptografik yapılar oluşturulabilmektedir. Ayrıca, farklı polinomların karşılaştırılması, uygulama gereksinimlerine göre ideal indirgenemez polinomun seçimini mümkün kılmakta ve bu da esnek ve ölçeklenebilir sonlu cisim aritmetiği çözümlerinin geliştirilmesini desteklemektedir (Sim vd., 2015).

Çizelge 3.1. $\mathbb{F}_{2^4}$ Cisminde İndirgenemez Polinoma Göre XOR Sayıları

Onaltılık Gösterim	Cisim Elemanı	XOR Sayıları		
		0x13	0x19	0x1f
0	0	0	0	0
0x1	1	0	0	0
0x2	α	1	1	3
0x3	$\alpha + 1$	5	3	5
0x4	α^2	2	3	3
0x5	$\alpha^2 + 1$	6	5	5
0x6	$\alpha^2 + \alpha$	5	2	6
0x7	$\alpha^2 + \alpha + 1$	9	6	6
0x8	α^3	3	6	3
0x9	$\alpha^3 + 1$	1	8	5
0xa	$\alpha^3 + \alpha$	8	5	6
0xb	$\alpha^3 + \alpha + 1$	6	9	6
0xc	$\alpha^3 + \alpha^2$	5	1	6
0xd	$\alpha^3 + \alpha^2 + 1$	3	5	6
0xe	$\alpha^3 + \alpha^2 + \alpha$	8	6	5
0xf	$\alpha^3 + \alpha^2 + \alpha + 1$	6	8	3
Standart Sapma		2,68	2,68	1,71

Örnek 3.3. $\mathbb{F}_{2^4}/0x13$ cismi üzerinde tanımlı $H = had(0xa, 0xb, 0x9, 0xf)$ Hadamard matrisinin toplam XOR maliyetini belirlemek için Tanım 2.29 uyarınca şu adımlar izlenir:

- Çizelge 3.1’den alınan değerlere göre, matrisin ilk satırındaki elemanların toplam XOR sayısı $(8 + 6 + 1 + 6) = 21$ ’dir.
- (Sıfır olmayan eleman sayısı $- 1$) $\times$ Cisim derecesi $= 3 \times 4 = 12$ XOR olarak hesaplanır.
- İlk satır için toplam XOR maliyeti $21 + 12 = 33$ olarak elde edilir.
- Bu hesaplama, tüm satırlar için aynı şekilde yapılır. Hadamard matris özelliği gereği satırlar eşdeğer olduğundan, toplam maliyet $33 \times 4 = 132$ olarak bulunur.

XOR sayısı, hesaplama açısından sade ve uygulanabilir bir metrik olsa da, minimum XOR sayısına sahip bir gerçekleştirme elde etmek oldukça karmaşık bir problemdir. Bunun temel nedeni, literatürde XOR sayısı ile ilişkili ölçütlerin yalnızca üst sınırları belirlemesi; buna karşılık alt sınırları ifade eden genel geçer bir ölçütün mevcut olmamasıdır. $\mathbb{F}_{2^m}$ cismi üzerinde tanımlı $n \times n$ boyutundaki matrislerin XOR maliyetini hesaplamaya yönelik genel bir yöntem ilk kez Khoo ve arkadaşları tarafından sunulmuştur (2014). Bu çalışmanın ardından, literatürde XOR işlemlerinin sayısını minimize eden difüzyon matrislerini bulmak amacıyla iki temel yaklaşımın öne çıktığı görülmektedir. İlk yaklaşım olan yerel (local) optimizasyon teknikleri, difüzyon matrisinin her bir bileşeninin XOR maliyetini ayrı ayrı değerlendirip iyileştirmeye çalışırken; ikinci yaklaşım olan küresel (global) optimizasyon teknikleri, matrisi bir bütün olarak ele almakta ve $mn \times mn$ boyutunda ikili matrisler üzerinde doğrudan optimizasyon yapmaktadır.

3.1. Yerel Optimizasyon

Yerel optimizasyon yöntemleri, XOR sayısını minimize etmeye yönelik iki farklı ölçüt sunmaktadır. Bunlardan ilki, Khoo vd. (2014) tarafından önerilen ve Jean vd. (2017) tarafından doğrudan XOR sayısı olarak adlandırılan ölçüttür. Bu ölçüt, matrisin her satırının bağımsız olarak işlendiği, doğrudan (naive) bir donanım uygulamasına karşılık gelmektedir. Literatürde en yaygın kullanılan bu yöntem, genellikle "ham XOR sayısı" ifadesiyle eşanlamlıdır.

İkinci ölçüt ise Jean vd. (2017) tarafından tanımlanan ve sıralı XOR sayısı olarak adlandırılan yaklaşımdır. Bu ölçüt, ekstra bellek kullanılmadan, yerinde (in-place) işlemlerle gerçekleştirilen sıralı uygulamalarda ortaya çıkan minimum XOR işlem sayısını ifade eder.

3.1.1. Doğrudan XOR Sayısı

Doğrudan XOR sayısı (Direct XOR Count), M matrisinin $DXC(M)$ veya $d - XOR(M)$ şeklinde gösterilen bir ölçüttür. Örnek 3.3'de hesaplanan H matrisinin XOR maliyeti aslında bu ölçüte karşılık gelmektedir. $\mathbb{F}_{2^m}$ gibi bir sonlu cisim elemanı, $\mathbb{F}_2$ cisminde tanımlı m boyutlu ikili vektörlerle temsil edilebildiğinden (bkz. Örnek 3.1 ve 3.2), bu temsil doğrultusunda, $\mathbb{F}_{2^m}$ 'de sıfırdan farklı bir α elemanı ile çarpma işlemi, $\mathbb{F}_2$ 'de tanımlı $m \times m$ boyutlu bir sol çarpım matrisi ile ifade edilebilir. Genellikle M_α ile gösterilen bu matrisin XOR maliyeti, α elemanının $d - XOR$ değeri olarak adlandırılır (Beierle, Kranz & Leander, 2016; Sarkar & Sim, 2016; Sim vd., 2015).

Bu bağlamda, $d - XOR$ ölçütünün biçimsel tanımı aşağıda verilmiştir:

Tanım 3.1 (Doğrudan XOR Sayısı). $\mathbb{F}_{2^m}/p(x)$ cisminde tanımlı bir α elemanının $d - XOR$

değeri, $\alpha : \beta \rightarrow \alpha\beta$ işlemini gerçekleştirmek için gereken XOR işlem sayısını ifade eder. Burada w , M_α çarpma matrisinin Hamming ağırlığı; r ise matrisin satır sayısı olmak üzere, doğrudan XOR sayısı aşağıdaki gibi tanımlanır:

$$d - XOR(M_\alpha) = wM_\alpha - r \quad (3.1)$$

Bu ölçüt, yalnızca XOR işlemi gereksiniminin üst sınırını ifade eder; gerçek uygulamalarda sıralı optimizasyon teknikleri sayesinde daha düşük maliyetler elde edilebilir.

3.1.2. Sıralı XOR Sayısı

Sıralı XOR sayısı (Sequential XOR Count), bir sonlu cisim elemanı ile yapılan çarpma işleminin optimize edilmiş uygulamasında ortaya çıkan en az XOR işlem sayısını temsil eder (Beierle vd., 2016). Bu ölçüt, genellikle $SXC(M)$ veya $s - XOR(M)$ şeklinde gösterilir ve doğrudan XOR sayısının azaltılmasına yönelik önerilmiştir (Jean, Peyrin, Sim & Tournateaux, 2017). Yerinde algoritmalarla gerçekleştirilen uygulamalarda, elde edilen çıktılar doğrudan girişin üzerine yazılabilir veya giriş bileşenlerinin konumlarını değiştirebilir. Bu tür algoritmaların çoğu ek bellek kullanımına ihtiyaç duymaz. $s - XOR$ ölçütü, bu tür uygulamaların ardışık bir XOR işlem dizisiyle gerçekleştirilmesini esas alır. Uygulamadaki bazı sınırlamalara rağmen, $s - XOR$ değeri birçok durumda $d - XOR$ değerinden önemli ölçüde düşük olabilir.

Aşağıda $s - XOR$ ölçütünün biçimsel tanımı verilmiştir:

Tanım 3.2 (Sıralı XOR Sayısı). $\mathbb{F}_{2^m}/p(x)$ cisminde tanımlı bir α elemanının $s - XOR$ değeri, çarpma işleminin sol çarpım matrisi olan M_α kullanılarak gerçekleştirildiğinde ortaya çıkan minimum XOR işlem sayısını belirtir. Bu minimum, tüm olası işlem sıralamaları göz önünde bulundurularak elde edilir ve $s - XOR(M_\alpha)$ ile gösterilir (Jean vd., 2017).

Örnek 3.4. $\mathbb{F}_{2^3}/0xb$ cisminde tanımlı $\alpha = 6_h$ elemanının, rastgele bir β elemanı (b_2, b_1, b_0) ile çarpılması durumunda $d - XOR$ ve $s - XOR$ değerleri aşağıdaki şekilde elde edilmektedir:

$$\begin{aligned} (1, 1, 0) \cdot (b_2, b_1, b_0) &= (b_0 \oplus b_2, b_1 \oplus b_2, b_1) \oplus (b_1, b_0 \oplus b_2, b_2) \\ &= (b_0 \oplus b_1 \oplus b_2, b_0 \oplus b_1, b_1 \oplus b_2) \end{aligned}$$

Bu işleme ait çarpım matrisi:

$$M_\alpha = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \quad M_\alpha \cdot \begin{bmatrix} b_2 \\ b_1 \\ b_0 \end{bmatrix} = \begin{bmatrix} b_0 \oplus b_1 \oplus b_2 \\ b_0 \oplus b_1 \\ b_1 \oplus b_2 \end{bmatrix}$$

M_α 'nın Hamming ağırlığı $w = 7$, satır sayısı $r = 3$ olduğundan, denklem 3.1 kullanılarak $d - \text{XOR}(M_\alpha) = 7 - 3 = 4 \text{ XOR}$ sonucu elde edilir.

$s - \text{XOR}$ değeri için ise, hesaplanan bileşenlerin bazıları birbirlerinin ara adımlarını içerdiğinden, optimizasyon yapılabilir. Örneğin, önce $b_1 \oplus b_2$ elde edilip, ardından bu değer b_0 ile XOR'lanarak ilk bileşen elde edilebilir. Bu işlem sırası ile toplamda yalnızca 2 XOR işlemi gereklidir. Böylece $s - \text{XOR}(\alpha) = 2 \text{ XOR}$ sonucuna ulaşılır.

3.2. Küresel Optimizasyon

Küresel optimizasyon teknikleri, difüzyon matrisinin bir bütün olarak ele alındığı ve her elemanın tek tek incelenmesinden ziyade tüm matrisin ikili forma dönüştürülmesiyle analiz edildiği yöntemleri kapsamaktadır. Bu yaklaşımda, özel olarak bu amaç için geliştirilmiş çeşitli yazılım araçları kullanılmaktadır (Duval & Leurent, 2018; Kranz, Leander, Stoffelen & Wiemer, 2017). Küresel optimizasyonun temel hedefi, literatürde “En Kısa Doğrusal Düz Çizgi Programı” (Shortest Linear Straight Line Program–SLP) olarak adlandırılan optimizasyon problemine etkili çözümler üretmektir.

3.2.1. En Kısa Doğrusal Düz Çizgi Programı (SLP)

Bir dizi doğrusal ifadenin hesaplanmasında gereken işlem sayısının minimize edilmesi problemi, En Kısa Doğrusal Program (SLP) olarak adlandırılır.

$A, \mathbb{F}_2$ üzerinde tanımlı $m \times n$ boyutundaki bir sabit matris ve $x, \mathbb{F}_2$ üzerinde tanımlı n değişkenli bir vektör olsun. Bu durumda SLP, belirli bir formatta olan her program satırı için $A \cdot x$ 'i hesaplayan minimum satır sayısına sahip programı bulmayı amaçlar.

$V, \{x_0, x_1, \dots, x_{n-1}\}$ giriş değişkenlerini içeren $\mathbb{F}_2$ üzerinde tanımlı bir değişken kümesi olsun. $v_i, v_j \in V$ olmak üzere, her program satırı şu formattadır:

$$v' := v_i + v_j$$

Bu program satırının çalıştırılmasıyla yeni değişken v' kümeye eklenir:

$$V := V \cup \{v'\}$$

Yeni deęişken v' artık sonraki program satırlarında kullanılabilir duruma gelir. Program, $\exists(v_1, \dots, v_m) \in V^m$ koşulu saęlandıęında $A \cdot x$ 'i hesaplar ve řu eřitlik elde edilir:

$$A \cdot x = (v_1, \dots, v_m)^T$$

En Kısa Doğrusal Program, kriptografi alanında en zorlu ve kapsamlı hesaplama problemlerinden biri olarak kabul edilmektedir. Bu problem, $\mathbb{F}_2$ cismi üzerinde yalnızca XOR kapılarının kullanıldıęı bir devre tasarımında, belirli bir doğrusal fonksiyonun en kısa hesaplama yolunu belirlemeye karřılık gelmektedir.

SLP problemine örnek olarak AES řifreleme algoritmasının MixColumns ve SubBytes ařamalarındaki doğrusal dönüşüm optimizasyonu verilebilir. Ancak bu teknik, S-kutuları gibi doğrusal olmayan işlemlerde kullanılamaz, bu durumda SAT (Satisfiability Solver) çözücüler kullanılır.

SLP probleminin temelinde, belirli bir ikili matrisin her bir çıktı bitini üretmek için gereken minimum XOR işlemi sayısını bulmak yatmaktadır. Bu doğrultuda geliştirilen çeřitli algoritmalar, doğrusal dönüşümlerin donanımsal maliyetini azaltmak amacıyla kullanılmaktadır. Paar tarafından geliştirilen yöntem, ortak alt ifadeleri belirleyerek XOR işlemlerinin sayısını düşürmeyi hedeflemekte ve böylece daha verimli çizgi programları üretmektedir (Paar, 1997). Bu yaklařımı takip eden önemli çalışmalardan biri de Boyar ve Peralta tarafından önerilen BP algoritmasıdır. Bu algoritma, çıkıřların her birini hesaplamak için yeniden kullanılabilir ara ifadeleri sistematik olarak analiz etmekte ve bu sayede toplam XOR sayısını minimize eden optimize edilmiř çizgi programları oluřturmaktadır (Boyar & Peralta, 2010). BP algoritması, özellikle büyük boyutlu doğrusal dönüşümlerde hem doğruluk hem de verimlilik aısından bařarılı sonuçlar vermekte olup, kriptografik yapıların donanımda düşük kaynak tüketimiyle gerçekleřtirilmesine olanak tanımaktadır.

3.2.1.1. Paar Algoritması

Açęözlü algoritma sınıfına dahil olan Paar Algoritması (Paar, 1997), dięer açęözlü algoritmalarda olduęu gibi çözüme en yakın seęimi yapar. Bu seęim her zaman optimal olmasa da, zaman faktörünün kritik olduęu çalışmalarda BP algoritmasına kıyasla büyük ölçekli matrisler üzerinde XOR sayısının azaltılmasında etkilidir.

XOR sayısının minimize edilmesi istenen A matrisi için Paar algoritması řu adımları takip eder:

1. Hamming aęırlıęı en yüksek olan sütunlar AND işlemi kullanılarak belirlenir.

2. Bu işlemle XOR işlemleri sırasında en çok tekrarlanan x_i ve x_j elemanları tespit edilir.
3. Elde edilen çarpım sütununa "newClmn" adı verilerek matrise eklenir ve A matrisi genişletilir. Bu şekilde yeni giriş değişkeni $v = x_i \oplus x_j$ elde edilir.
4. Mevcut sütunlara "oldClmn" adı verilir ve şu işlem uygulanır:

$$\text{oldClmn} \leftarrow \text{oldClmn} \oplus \text{newClmn}$$

Bu son işlemle gereksiz XOR işlemleri elimine edilir.

3.2.1.2. Boyar-Peralta (BP) Algoritması

Boyar-Peralta algoritması (Boyar & Peralta, 2010), $\mathbb{F}_2$ cismi üzerinde tanımlı $m \times n$ boyutlu bir A matrisi için $f(x) = A \cdot x$ değerini hesaplar ve en kısa lineer düz çizgi programını bulmayı hedefler.

Algoritma, başlangıçta sadece $(x_1, x_2, x_3, \dots, x_n)$ giriş değişkenlerinden oluşan M tabanı oluşturur. Daha sonra, A matrisinin satırlarının M kümesine olan doğrusal fonksiyonlarının Hamming uzaklıklarından oluşan bir uzaklık vektörü $\text{Uzk}[\cdot]$ tanımlanır.

Eğer f_i , A matrisinin i -inci satırı tarafından verilen doğrusal fonksiyon ise, $\text{Uzk}[i]$, M tabanından f_i 'yi elde etmek için gereken minimum fonksiyon sayısını verir.

Başlangıçta $\text{Uzk}[i]$, i -inci satırının Hamming ağırlığından 1 eksiktir. Algoritma şu döngüyü uygular:

1. M kümesinden iki temel eleman seçilir ve XOR işlemine tabi tutularak elde edilen yeni eleman M kümesine eklenir.
2. $\text{Uzk}[i]$ vektörü, M kümesine eklenen yeni elemandan dolayı güncellenir.
3. Tüm i değerleri için $\text{Uzk}[i] = 0$ elde edilinceye kadar adımlar tekrarlanır.

Herhangi bir aşamada M kümesinin boyutu t ise, yeni temel eleman seçimi için $\binom{t}{2}$ seçenek bulunur. Seçim kriterleri ise aşağıdaki gibidir:

1. Güncellenmiş $\text{Uzk}[\cdot]$ dizisinin elemanlarının toplamını minimize eden elemanın seçimi
2. Eşitlik durumunda, güncellenmiş $\text{Uzk}[\cdot]$ dizisinin Öklid normunu maksimize eden elemanın seçimi

Bu yaklaşımın mantığı, $[0, 0, 3, 1]$ gibi bir uzaklık vektörünün $[1, 1, 1, 1]$ vektörüne tercih edilmesi ilkesine dayanır. İkinci durumda 4 XOR kapısı gerekirken, birinci durumda 3 XOR kapısı ile $Uzk[\cdot] = 0$ eşitliğine ulaşılabilir.

Bu iki algoritma, bir difüzyon matrisinin uygulanmasında ihtiyaç duyulan XOR kapılarının sayısını azaltarak devre alanı üzerinde iyileştirme sağlamayı hedefleyen özel araçlara dayanmakta olsa da, bu uygulamaların değerlendirilmesinde dikkate alınması gereken bir diğer önemli kriter de gecikmedir. Gecikme, bir devrenin çalışabileceği maksimum saat frekansını belirleyen temel bir sınırlayıcıdır ve Tanım 2.30'da açıklanmış olan devre derinliği ile ifade edilir.

Li, Sun, Li, Wei ve Hu'nun çalışmasında (2019), Boyar-Peralta algoritması devre derinliği gözetilerek yeniden yapılandırılmıştır. Bu kapsamda, algoritma belirli bir derinlik sınırını aşmayan sinyallerin seçimini öncelikli hale getirecek biçimde uyarlanmıştır.

3.2.1.3. Superior Boyar-Peralta (SBP) Sezgisel Algoritması

Literatürdeki en güncel yaklaşımlardan biri olan Superior Boyar-Peralta (SBP) algoritması (Pehlivanoglu & Demir, 2024), Boyar-Peralta (BP) sezgiselinin geliştirilmiş bir versiyonudur. Bu algoritmanın temel amacı, bir lineer katmanın devresini temsil eden En Kısa Doğrusal Düz Çizgi Programları (SLP) için hem iki girişli XOR kapılarının sayısını hem de devre derinliğini minimize etmektir. Bu sayede özellikle düşük gecikmeli (low-latency) devreler elde etmede oldukça başarılıdır.

SBP, Boyar-Peralta'nın temel yapısını kullanmakla birlikte, yeni taban elemanlarını seçerken farklı ve daha kontrollü bir yapı sunar. Algoritmanın temel işleyişi ve getirdiği yenilikler şu şekilde özetlenebilir:

- **Eşik Değeri ve Aday Havuzu:** SBP, her adımda değerlendirilecek potansiyel eleman çiftlerinin sayısını sınırlamak için bir eşik değeri kullanır. Diğer algoritmaların aksine tüm olasılıkları tüketerek arama uzayını genişletmek yerine, SBP en iyi adaylardan oluşan bu havuzu dikkatlice yönetir. Bu, arama uzayını daraltarak daha verimli bir şekilde optimum sonuca ulaşmayı sağlar.
- **Rastgeleleştirme Adımı:** Aday havuzundaki en iyi çiftler arasından birini seçmek için tekdüze tamsayı dağılımı (uniform integer distribution) kullanan bir rastgeleleştirme adımı içerir. Bu yaklaşım, algoritmanın yerel optimal noktalara takılıp kalmasını önlemeye yardımcı olur.
- **Öklid Normu ve Derinlik Farkındalığı:** Yeni bir taban elemanı seçilirken, güncellenmiş uzaklık vektörünün Öklid normunu maksimize eden ve aynı zamanda

belirlenen devre derinliđi sınırını aşmayan çiftlere öncelik verir. Bu kriter, optimizasyon sürecine gecikme metriđini doğrudan dahil eder.

Tez çalışmasında, önerilen yöntemle tasarlanan yarı-involütif matrislerin donanımsal maliyet açısından optimize edilmesi de amaçlanmıştır. Bu doğrultuda, doğrusal dönüşümlerin mantıksal kapı seviyesinde en düşük kaynak kullanımıyla uygulanabilmesini sağlayan Paar algoritması ile Boyar-Peralta (BP) ve onun gecikme hassasiyeti gözetilerek geliştirilmiş hali olan Superior Boyar-Peralta (SBP) algoritmalarından yararlanılmıştır. Söz konusu algoritmalar kullanılarak, yarı-involütif matrislerin XOR sayısını minimize edecek şekilde yeniden yapılandırılması sağlanmıştır.



BÖLÜM 4

YARI INVOLUTİF MDS MATRİSLERİN CEBİRSEL TASARIMI

Bu bölümde, tez çalışmasının temelini oluşturan yarı involutif MDS matrislerin üretilmesine yönelik geliştirilen metodoloji sunulmaktadır. İzlenen yöntem, öncelikle literatürdeki mevcut involutif MDS matrislerinin Magma hesaplamalı cebir sistemi kullanılarak üretilmesini, ardından bu matrislerden faydalanarak yarı involutif yapıların türetilmesini kapsamaktadır. Son olarak, elde edilen yeni matrislerin donanım verimliliğini artırmak amacıyla uygulanan XOR optimizasyon teknikleri detaylandırılmaktadır.

4.1. Hesaplamalı Cebir Sistemi: Magma

Tez kapsamındaki yoğun cebirsel hesaplamaların gerçekleştirilmesi, doğrulanması ve analizi için Magma hesaplamalı cebir sistemi tercih edilmiştir.

Magma projesi, 1980’lerin sonunda Sydney Üniversitesi’nde Prof. John Cannon liderliğinde, Cayley sisteminin devamı olarak başlatılmıştır. Sistemin temel tasarım felsefesi, "kesin matematiksel nesneler üzerinde doğru ve verimli hesaplama yapmak" olarak özetlenebilir. Bu doğrultuda Magma; sonlu/sonsuz gruplar, halkalar, cisimler, eliptik eğriler ve kriptografik yapılar gibi matematiksel nesneler için özelleşmiş veri tipleri ve 2000’den fazla yerleşik fonksiyon sunarak hem yüksek performans hem de matematiksel doğruluk sağlar (“Magma Computational Algebra System”, 2025).

Bu çalışmada Magma, özellikle aşağıdaki alanlardaki güçlü yetenekleri nedeniyle kritik bir rol oynamıştır:

- $\mathbb{F}_{2^m}$ gibi sonlu cisimler üzerindeki polinom aritmetiği,
- MDS ve involutif/yarı involutif matrislerin parametrik formüllere dayalı olarak sistematik üretimi,

- Üretilen matrislerin özelliklerinin (tersinirlik, MDS koşulu vb.) doğrulanması.

Aşağıdaki çizelge, Magma’da sıkça kullanılan bazı temel komutları özetlemektedir.

Çizelge 4.1. Magma’da Temel Komut Örnekleri

Amaç	Komut
Oturumu başlatma	\$ magma
Sonlu cisim tanımı	$F\langle a \rangle := GF(2^8);$
Çarpanlara ayırma	$Factorization(x^{12} - 1);$
Matris ve rütbe	$M := Matrix([[1,0,1],[1,1,0],[0,1,1]]); Rank(M);$
Grup tanımı	$G := SymmetricGroup(5);$
AES alanında tersleme	$b := F!0x57; b^{(-1)};$

4.2. İnvolutif MDS Matrislerin İnşası

Bu tezde, öncelikle involutif MDS matrislerin üretimi için literatürde yer alan iki temel yöntemden faydalanılmıştır. Bunlardan ilki, 3×3 boyutundaki matrisler için (Güzel, Sakallı, Akleylek, Rijmen & Çengellenmiş, 2019) tarafından sunulan doğrudan inşa yöntemi, ikincisi ise 4×4 boyutundaki matrisler için (Tuncay vd., 2023) tarafından geliştirilen hibrit yöntemdir.

4.2.1. 3×3 Boyutlarındaki MDS Matrisler İçin Doğrudan İnşa Yöntemi

Bu yöntemde, matrisin involutif olma koşulu ($A^2 = I$) temel alınarak matris elemanları arasında cebirsel ilişkiler kurulur ve tüm 3×3 boyutlarındaki involutif MDS matrislerini üretebilen bir form sunulur.

Teorem 4.1. (3×3 İnvolutif Matris Formu) F_{2^m} üzerinde 3×3 boyutlu bir A matrisinin involutif olabilmesi için, elemanları a_{11}, a_{22} köşegen elemanları ve sıfırdan farklı $b_0, b_1 \in F_{2^m}$ parametreleri cinsinden aşağıdaki gibi ifade edilebilir (Güzel vd., 2019):

$$\begin{aligned}
 a_{12} &= (a_{11} + 1)b_0 \\
 a_{13} &= (a_{11} + 1)b_1 \\
 a_{21} &= (a_{22} + 1)b_0^{-1} \\
 a_{23} &= (a_{22} + 1)b_0^{-1}b_1 \\
 a_{31} &= (a_{11} + a_{22})b_1^{-1} \\
 a_{32} &= (a_{11} + a_{22})b_1^{-1}b_0 \\
 a_{33} &= a_{11} + a_{22} + 1
 \end{aligned}$$

Kanıt. $A = [a_{ij}]$ matrisinin involutif olması, $A^2 = I$ koşulunu gerektirir. Bu durum, A^2 'nin elemanları $c_{ij} = \sum_{k=1}^3 a_{ik}a_{kj}$ için $c_{ii} = 1$ ve $c_{ij} = 0$ ($i \neq j$) denklemlerini verir.

Köşegen Elemanları ($c_{ii} = 1$)

$$a_{11}^2 + a_{12}a_{21} + a_{13}a_{31} = 1 \quad (4.1)$$

$$a_{21}a_{12} + a_{22}^2 + a_{23}a_{32} = 1 \quad (4.2)$$

$$a_{31}a_{13} + a_{32}a_{23} + a_{33}^2 = 1 \quad (4.3)$$

Bu üç denklemin F_{2^m} cisminde taraf tarafa toplanmasıyla $(a_{11} + a_{22} + a_{33})^2 = 1$ elde edilir, bu da aşağıdaki temel ilişkiyi ortaya çıkarır:

$$a_{11} + a_{22} + a_{33} = 1 \quad (4.4)$$

Köşegen Dışı Elemanlar ($c_{ij} = 0$)

$$a_{12}(a_{11} + a_{22}) = a_{13}a_{32} \quad (4.5)$$

$$a_{13}(a_{11} + a_{33}) = a_{12}a_{23} \quad (4.6)$$

$$a_{21}(a_{11} + a_{22}) = a_{23}a_{31} \quad (4.7)$$

Denklem (4.4) kullanılarak bu sistem çözüldüğünde, 9 bilinmeyenli denklem sistemi 4 parametreye (a_{11}, a_{22}, b_0, b_1) indirgenir ve teorem kanıtlanmış olur. ■

Bu formdaki bir matrisin MDS olması için ise tüm kare alt matrislerinin determinantlarının sıfırdan farklı olması gerekir. Bu koşul da aşağıdaki kısıtlamalarla sağlanır:

$$a_{11}, a_{22} \neq 0, 1$$

$$a_{11} \neq a_{22}$$

$$a_{11} + a_{22} \neq 1$$

$$b_0, b_1 \in F_{2^m} - \{0\}$$

Söz konusu bu yöntem ile $\mathbb{F}_{2^3}$ cisminde tanımlı $0 \times b$ ve $0 \times d$ indirgenemez polinomlarının her biri için 1176'şar adet, $\mathbb{F}_{2^4}$ cisminde tanımlı 0×13 ve 0×19 indirgenemez polinomlarının her biri için ise 37800'er adet, 3×3 boyularında involutif MDS matris elde edilebilmektedir (Güzel vd., 2019).

4.2.2. 4×4 Boyutlarındaki MDS Matrisler için Hibrit Yöntem

4×4 boyutlarındaki ilişkilerin karmaşıklığı nedeniyle, doğrudan formül bulmak yerine iki aşamalı bir hibrit yöntem geliştirilmiştir. Birinci aşamada, "Temsilci Matrisler" (RIM) adı verilen özel özelliklere sahip matrisler kümesi arama yöntemi ile tespit edilmektedir. İkinci aşamada ise bulunan her RIM'den yeni parametreler kullanarak involutif MDS matrislerin tümü doğrudan tasarım yöntemi ile elde edilmektedir (Tuncay vd., 2023).

Tanım 4.1. 4×4 boyutlarındaki bir Temsilci İnvolutif Matris (RIM), aşağıdaki iki koşulu sağlar:

1. Matrisin ana köşegen elemanlarının XOR toplamı 0'dır.
2. Matrisin herhangi bir satırındaki (ve sütunundaki) elemanların XOR toplamı 1'dir.

Bu koşullar, arama uzayını önemli ölçüde daraltarak verimli bir arama yapılmasını mümkün kılar. Bulunan her bir Temsilci İnvolutif MDS matrisi $R = [r_{ij}]$ kullanılarak ve sıfırdan farklı $b_1, b_2, b_3 \in F_{2^m}$ parametreleri yardımıyla, involutif ve MDS özelliklerini koruyan yeni bir $A = [a_{ij}]$ matrisi üretilir.

$$A = [a_{ij}] = \begin{bmatrix} r_{11} & r_{12}b_1 & r_{13}b_2 & r_{14}b_3 \\ r_{21}b_1^{-1} & r_{22} & r_{23}b_1^{-1}b_2 & r_{24}b_1^{-1}b_3 \\ r_{31}b_2^{-1} & r_{32}b_2^{-1}b_1 & r_{33} & r_{34}b_2^{-1}b_3 \\ r_{41}b_3^{-1} & r_{42}b_3^{-1}b_1 & r_{43}b_3^{-1}b_2 & r_{44} \end{bmatrix}$$

Bu üretim, satır/sütun çarpımlarının bir kombinasyonudur ve bir matrisin MDS özelliğinin bu tür işlemler altında korunması ilkesine dayanır. İlgili yöntem kullanılarak, $\mathbb{F}_{2^3}$ sonlu cismi üzerinde tanımlanmış olan $0xB$ ve $0xD$ polinomlarının her biri için 48 involutif ve MDS matris temsilcisi bulunduğu, ardından b_i parametreleri involutif MDS matris temsilcilerine uygulandıktan sonra toplamda $48 \cdot (2^3 - 1)^3 = 16.464$ adet 4×4 boyutunda involutif ve MDS matris elde edildiği, $\mathbb{F}_{2^4}$ sonlu cismi üzerinde tanımlanan $0x13$ polinomu için ise 71.856 involutif ve MDS matris temsilcisi olduğu ve b_i parametreleri involutif MDS matris temsilcilerine uygulandıktan sonra toplamda $71.856 \cdot (2^4 - 1)^3 = 242.514.000 \approx 2^{27.85}$ adet 4×4 boyutunda involutif ve MDS matris elde edildiği belirtilmiştir (Tuncay vd., 2023).

4.2.3. Parametrik Formüller ve Magma ile Üretim

Yukarıda özetlenen iki yaklaşım, sırasıyla 3×3 ve 4×4 boyutundaki involutif MDS matrislerin parametrik olarak elde edilmesini sağlar. Bu parametreler, hem matrisin tersinin kolayca hesaplanmasını hem de determinantların sıfırdan farklılığının doğrudan kontrolünü mümkün kılar.

Bu matrislerin pratikte elde edilebilmesi amacıyla Magma sistemi kullanılmıştır. Süreç şu adımları içerir:

1. **Cisim ve Polinom Tanımı:** İkili cisim uzantısını ($\mathbb{F}_{2^3}, \mathbb{F}_{2^4}$ vb.) oluşturan indirgenemez polinom yüklenir. Örneğin: `P<z>:= PolynomialRing(GF(2));`
`p:= z^4+z+1; F<x>:= ext<GF(2)|p>;`
2. **Parametre Uzayı ve Döngüler:** Involutif formüllerde yer alan parametreler üzerinde iç içe for döngüleri oluşturularak tüm olası kombinasyonlar taranır.
3. **Matris İnşası ve Kontrol:** Her parametre kombinasyonu için matris, Magma'nın `A:= Matrix(...)` komutuyla oluşturulur ve $A^2 = I$ ile involutif olma, (`MinimumWeight([I|A])`) ile ise dal sayısı hesaplanıp MDS olma koşulları kontrol edilir.
4. **Sonuçların Kaydedilmesi:** Tüm koşulları sağlayan matrisler, `PrintFile` komutuyla bir dosyaya kaydedilir.

Bu prosedür, parametlerle tanımlanmış tüm involutif MDS matrislerinin üretilmesine olanak sağlar.

4.3. Yarı-Involutif Matrislere Geçiş ve Üretim

Tekil olmayan (nonsingular) bir kare matris A için A^{-1} ifadesi, A matrisinin tersini temsil etmek üzere, eğer D_1 ve D_2 diyagonal (köşegen) matrisler ise,

$$A^{-1} = D_1 A D_2$$

eşitliğini sağlayan A matrisine yarı involutif matris denilmektedir (Chatterjee & Laha, 2023; Cheon, Curtis & Kim, 2021).

Teorem 4.2. $A = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ şeklinde tanımlanan 2×2 boyutlu tekil olmayan bir ikili matris

yarı involutif ise ve

$$D_1 = \begin{bmatrix} e_1 & 0 \\ 0 & e_2 \end{bmatrix}, D_2 = \begin{bmatrix} f_1 & 0 \\ 0 & f_2 \end{bmatrix}$$

$$e_1 = \frac{1}{\Delta} \quad e_2 = \frac{1}{\Delta} \cdot \frac{a}{d} \quad f_1 = \frac{d}{a} \quad f_2 = 1$$

Kanıt. 1 Yarı involutif matris tanımına göre, $A^{-1} = D_1 A D_2$ eşitliği aşağıdaki gibi düzenlenebilir:

$$\frac{1}{\Delta} \cdot \begin{bmatrix} d & -b \\ -c & a \end{bmatrix} = \begin{bmatrix} e_1 & 0 \\ 0 & e_2 \end{bmatrix} \cdot \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} f_1 & 0 \\ 0 & f_2 \end{bmatrix} = \begin{bmatrix} e_1 f_1 a & e_1 f_2 b \\ e_2 f_1 c & e_2 f_2 d \end{bmatrix} \quad (4.8)$$

Bu matris eşitliğinden aşağıdaki dört denklem elde edilir:

$$\frac{1}{\Delta} d = e_1 f_1 a \quad (4.9)$$

$$-\frac{1}{\Delta} b = e_1 f_2 b \quad (4.10)$$

$$-\frac{1}{\Delta} c = e_2 f_1 c \quad (4.11)$$

$$\frac{1}{\Delta} a = e_2 f_2 d \quad (4.12)$$

Denklem (4.10) ve (4.11) sadeleştirildiğinde ise aşağıdaki iki denklem elde edilir:

$$-\frac{1}{\Delta} = e_1 f_2 \quad (4.13)$$

$$-\frac{1}{\Delta} = e_2 f_1 \quad (4.14)$$

Denklem (4.13) ve (4.14) çarpıldığında denklem (4.15),

$$e_1 e_2 f_1 f_2 = \frac{1}{\Delta^2} \quad (4.15)$$

Denklem (4.13) ve (4.14) taraf tarafa bölüldüğünde ise denklem (4.16) elde edilir:

$$\frac{e_1}{e_2} = \frac{f_1}{f_2} \quad (4.16)$$

Denklem (4.8) ve denklem (4.9) yardımıyla

$$\frac{d^2}{a^2} = \frac{e_1}{e_2} \cdot \frac{f_1}{f_2} \quad (4.17)$$

elde edilir ve denklem (4.16) ile (4.17) birlikte kullanılarak denklem (4.16) aşağıdaki forma dönüştürülür:

$$\left(\frac{d}{a}\right)^2 = \left(\frac{e_1}{e_2}\right)^2 \quad (4.18)$$

$\mathbb{F}_2$ sonlu cismi üzerinde çalışıldığı göz önüne alındığında, denklem (4.18) aşağıdaki şekilde gösterilebilir:

$$\frac{d}{a} = \frac{e_1}{e_2} \quad (4.19)$$

Bu durumda e_2 ve f_2 aşağıdaki şekilde yazılabilir:

$$e_2 = \frac{a}{d}e_1 \quad f_2 = \frac{a}{d}f_1 \quad (4.20)$$

Denklem (4.15) ve (4.20) birlikte kullanıldığında f_1 elemanı için son eşitlik aşağıdaki gibi elde edilebilir:

$$f_1 = \frac{d}{a} \cdot \frac{1}{e_1} \cdot \frac{1}{\Delta} \quad (4.21)$$

Son durumda A matrisinin tersi aşağıdaki matrislerin çarpımı olarak yazılabilir ve bu matrisler Teorem 4.2 koşullarını sağlar.

$$\begin{aligned} A^{-1} &= \begin{bmatrix} e_1 & 0 \\ 0 & \frac{a}{d}e_1 \end{bmatrix} \cdot A \cdot \begin{bmatrix} \frac{d}{a} \cdot \frac{1}{e_1} \cdot \frac{1}{\Delta} & 0 \\ 0 & \frac{1}{e_1} \cdot \frac{1}{\Delta} \end{bmatrix} \\ &= e_1 \cdot \frac{1}{e_1} \cdot \frac{1}{\Delta} \begin{bmatrix} 1 & 0 \\ 0 & \frac{a}{d} \end{bmatrix} \cdot A \cdot \begin{bmatrix} \frac{d}{a} & 0 \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \frac{1}{\Delta} & 0 \\ 0 & \frac{1}{\Delta} \cdot \frac{a}{d} \end{bmatrix} \cdot A \cdot \begin{bmatrix} \frac{d}{a} & 0 \\ 0 & 1 \end{bmatrix} \end{aligned}$$

■

Teorem 4.3. *A, involütif MDS bir matris ve $c \in \mathbb{F}_{2^m} - \{0, 1\}$ olsun. Bu durumda A matrisinin c skaleri ile çarpımına eşit (yani $B = c \cdot A$) olan bir B matrisi, yarı involütif bir MDS matrisi olacaktır.*

Kanıt. $B = c \cdot A$ eşitliğini düşünelim. Eşitliğin her iki tarafının karesi alınarak, $B^2 = c^2 \cdot A^2$ eşitliği elde edilebilir. A matrisi bir involütif matris olduğundan, son denklemdeki A'nın karesi birim matris ile değiştirilebilir; diğer bir deyişle $B^2 = c^2 \cdot I$ eşitliği yazılabilir. Böylece, $B = c^2 \cdot B^{-1}$ veya $B^{-1} = \frac{1}{c^2} \cdot B = (\frac{1}{c}I) \cdot B \cdot (\frac{1}{c}I)$ eşitlikleri elde edilebilir. Bu da B matrisinin yarı involütif olduğunu göstermektedir. Bununla birlikte ilgili tanımda verilen köşegen matrisler için de $D_1 = D_2 = \frac{1}{c}I$ eşitliği yazılabilir. Ayrıca, A matrisi

MDS olduğundan bu matrisin tüm alt determinantları sıfırdan farklıdır. Bu nedenle $c \cdot A$ matrisinin alt determinant değerleri $c, c^2, c^3 \dots c^m$ 'in katları olduğundan $B = c \cdot A$ matrisi de yine MDS olacaktır. Dolayısıyla Teorem 4.3'ün şartları sağlanmaktadır. ■

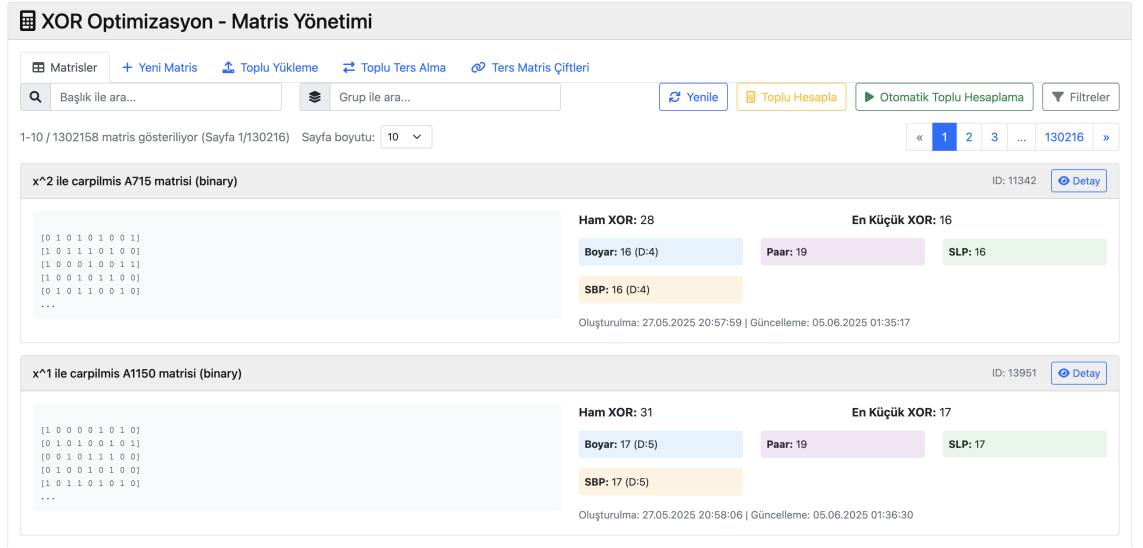
Sonuç 4.1. Yukarıdaki teoremin bir sonucu olarak, $c \in \mathbb{F}_{2^m} - \{0, 1\}$ olduğu göz önüne alındığında çalışmada kullanılan yöntem ile elde edilen yarı involutif matrislerin sayısının, involutif matrislerin sayısının $2^m - 2$ katına eşit olduğu ifade edilebilmektedir.

Yukarıda belirtilen sonuç uyarınca, bu tez çalışmasında 3×3 boyutlarında $\mathbb{F}_{2^3}$ cismi üzerinde $1.176 \cdot (2^3 - 2) = 7.056$ adet, $\mathbb{F}_{2^4}$ cismi üzerinde $37.800 \cdot (2^4 - 2) = 529.200$ adet, 4×4 boyutlarında ise $\mathbb{F}_{2^3}$ cismi üzerinde $16.464 \cdot (2^3 - 2) = 98.784$ adet, $\mathbb{F}_{2^4}$ cismi üzerinde $242.514.000 \cdot (2^4 - 2) = 3.395.196.000 \approx 2^{31.66}$ adet yarı involutif MDS matris elde edilmiştir.

4.3.1. Yarı İnvolutif Matrislerin Magma ile Üretimi ve XOR Optimizasyonu

Yarı involutif MDS matrislerin üretimi ve ön elemesi Magma'da şu adımlarla gerçekleştirilmiştir:

1. **Parametre Döngüleri ve Matris Üretimi:** Önceki adımlarda üretilen involutif MDS matrisler ve olası tüm skaler $c \in \mathbb{F}_{2^m} - \{0, 1\}$ değerleri için $B = c \cdot A$ matrisleri oluşturulur.
2. **Yarı İnvolutiflik ve MDS Kontrolü:** İlgili $\mathbb{F}_{2^m}$ cismi için $B^2 = c^2 \cdot I_{m \times m}$ eşitliği ile yarı involutif olma koşulu kontrol edilir. Ayrıca her bir B matrisinin MDS koşulunu sağladığı tekrar doğrulanır. 4×4 matrisler için bu, $\min(\text{MinimumWeight}([I_{4 \times 4} | B])) = 5$ ifadesiyle gerçekleştirilir.
3. **İkili Matrise Dönüştürme ve Ham XOR Sayısı Hesabı:** Donanım verimliliği için kritik olan XOR sayısı hesaplanması amacıyla bu çalışmada kullanılan algoritmalar optimizasyon işlemini ikili (binary) matrisler üzerinden gerçekleştirdiği için yarı involutif ve MDS koşullarını gerçekleyen her B matrisi, $\mathbb{F}_{2^m}$ cismi üzerinde tanımlı indirgenemez polinomunun bir kökü olan α elemanını temsil eden bir üreteç matris g kullanılarak $\text{GF2RepresentationMatrix}(B, g)$ fonksiyonu ile ikili forma dönüştürülür. Ardından elde edilen her ikili matrisin ham XOR sayısı hesaplanır.
4. **Ön Eleme ve Kayıt:** Özellikle $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı 4×4 boyutlarındaki matrisler için gerek çalışılan uzayın büyüklüğü, ve belirli c skaleriyle olan çarpımlarının XOR işlem maliyetinin yüksek olması nedeniyle bu matrislere $\text{hamXOR} \leq 80$ kısıtlaması eklenmiştir. Bu kısıtlamayı ve diğer tüm koşulları



Şekil 4.1. Web Uygulaması Ana Sayfa Görünümü

sağlayan matrisler, parametreleri ve ham XOR değerleriyle birlikte optimize edilmek üzere kaydedilir.

Bu yaklaşım, çalışmada kullanılan yöntemle tutarlı ve pratikte düşük XOR maliyeti gereksinimini karşılayan yarı involutif MDS matrislerinin hem etkili bir biçimde üretilmesini hem de uygun kriterlere göre filtrelenmesini mümkün kılmaktadır.

4.4. Matris Analizi ve Optimizasyonuna Yönelik Web Tabanlı Bir Yazılım Sistemi

Üretilen matrislerin yönetimi, analizi ve optimizasyonu için modern bir web uygulaması geliştirilmiştir. Bu sistem, C++ tabanlı mevcut komut satırı araçlarının sınırlılıklarını aşmayı hedefler.

4.4.1. Kullanılan Teknolojiler

- **Go (Golang):** Uygulamanın arka uç servislerinin geliştirilmesinde, Google tarafından geliştirilen açık kaynaklı bir programlama dili olan Go (Golang) tercih edilmiştir. Go dili, derlenmiş bir dil olmasının sağladığı yüksek çalışma zamanı performansı, yerleşik eşzamanlılık yapıları sayesinde paralel işlem yetenekleri ve sade sözdizimi ile sistem düzeyinde yazılım geliştirmeye elverişli bir ortam sunmaktadır (Donovan & Kernighan, 2015). Bu özellikleri dolayısıyla, özellikle yüksek hacimli veri işleme ve hızlı API yanıt sürelerinin kritik olduğu durumlarda Go dili, güvenilir ve sürdürülebilir çözümler üretmek adına uygun bir tercih olarak değerlendirilmiştir.
- **PostgreSQL:** Veri yönetimi katmanında, ilişkisel veritabanı yönetim sistemleri

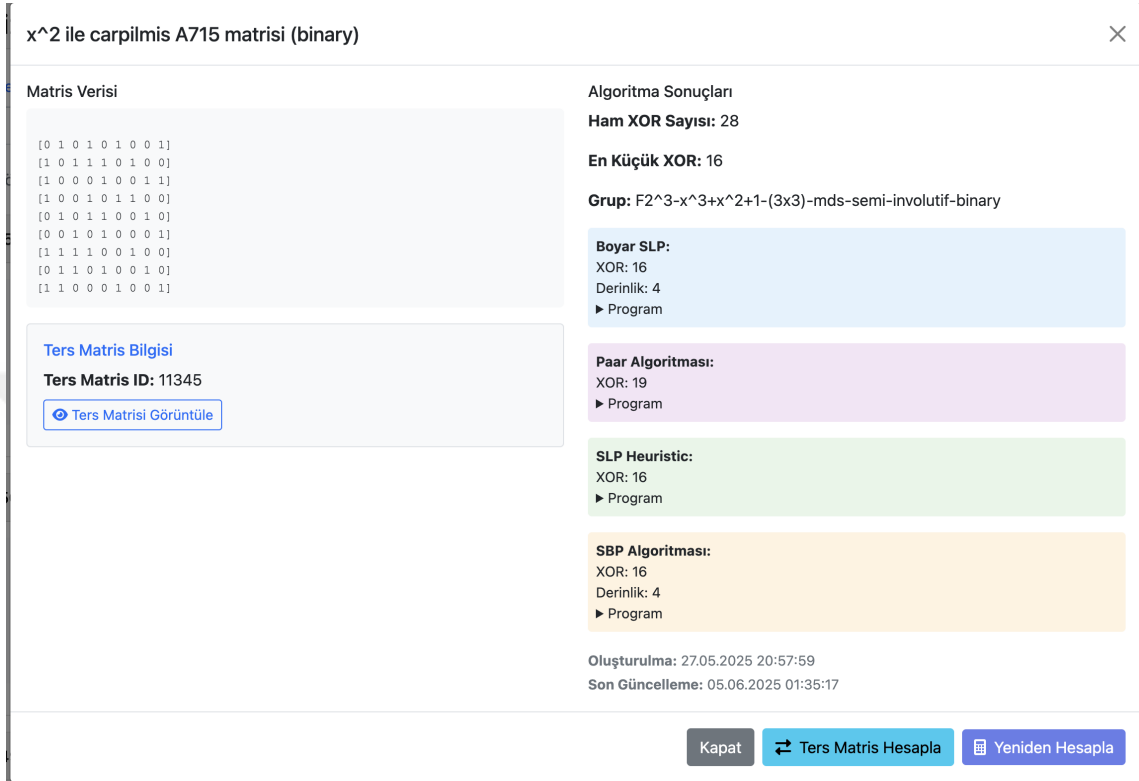
arasında açık kaynaklı ve gelişmiş özellikleriyle öne çıkan PostgreSQL tercih edilmiştir. Sistem içerisinde milyonlarca matris verisinin güvenli, kalıcı ve performanslı biçimde saklanabilmesi amacıyla PostgreSQL'in gelişmiş indeksleme, veri bütünlüğü ve sorgu optimizasyonu özelliklerinden faydalanılmıştır (Momjian, 2001). Her bir matrisin benzersiz olarak tanımlanabilmesi için özgün anahtarlar oluşturulmuş, ayrıca sorgu performansını artırmak amacıyla uygun veri alanlarında indeksleme yapılmıştır.

- **Docker:** Geliştirilen uygulamanın bağımlılıklarıyla birlikte izole ve taşınabilir bir ortamda çalıştırılabilmesi amacıyla Docker teknolojisinden yararlanılmıştır. Docker, konteyner tabanlı sanallaştırma yaklaşımı sayesinde yazılımın farklı ortamlarda tutarlı bir şekilde dağıtılmasını ve çalıştırılmasını mümkün kılar (Merkel vd., 2014). Bu sayede geliştirilen sistemin kurulum süreci basitleştirilmiş, üretim ortamına geçişte oluşabilecek çevresel farklılıkların etkisi en aza indirilmiştir.
- **JSON:** Sistem bileşenleri arasında veri alışverişi yapılırken, yaygın olarak kullanılan ve insan tarafından okunabilir bir veri serileştirme biçimi olan JavaScript Object Notation (JSON) formatı tercih edilmiştir. JSON, özellikle web tabanlı API'ler aracılığıyla istemci-sunucu iletişimde kolaylık ve esneklik sağlayan bir standart haline gelmiştir (Bray, 2014). Bu çalışmada, matris verilerinin ve işlem çıktılarının API üzerinden aktarımı JSON formatında gerçekleştirilmiş, böylece hem istemci hem de sunucu tarafında veri işleme süreçleri sadeleştirilmiştir.

4.4.2. Sistem Mimarisi Bileşenleri

- **Arka Uç Servisi (Backend API):** Uygulamanın sunucu tarafı, yüksek performanslı ve eşzamanlı işlem desteği sağlayan Go programlama dili ile geliştirilmiş RESTful bir web servisi olarak yapılandırılmıştır.
- **Veri Tabanı Katmanı:** Kalıcı, güvenli ve ölçeklenebilir veri saklama ihtiyaçları doğrultusunda, ilişkisel veritabanı yönetim sistemi olarak PostgreSQL tercih edilmiştir.
- **Ön Yüz (Frontend):** Kullanıcı etkileşimini kolaylaştırmak amacıyla, JavaScript tabanlı dinamik ve kullanıcı dostu bir web arayüzü geliştirilmiştir.
- **Otomatik Veri İşleme ve İçer Aktarma Modülü:** Uygulama başlatıldığında, belirli bir dizindeki matris dosyalarını otomatik olarak algılar, bu verileri veritabanına aktarır ve her bir matris üzerinde önceden tanımlanmış optimizasyon algoritmalarını çalıştırır.

- **Ters Matris Hesaplama Bileşeni:** İkili sonlu cisim $GF(2)$ üzerinde tanımlı matrisler için Gaussian eliminasyon yöntemi kullanılarak ters matris hesaplaması gerçekleştiren özel bir modül geliştirilmiştir.



Şekil 4.2. Web Uygulamasında Matris Detay Görünümü

4.4.3. Uygulanan XOR Optimizasyon Algoritmaları

Sistem, verilen bir matris için en düşük XOR sayısı ile gerçekleştirilebilecek doğrusal devreyi elde etmeyi amaçlayan dört farklı algoritmanın karşılaştırmalı analizini gerçekleştirmektedir. Bu algoritmalar, Bölüm 3.2.'de detayları verilen XOR tabanlı devre optimizasyonu literatüründe öne çıkan farklı yaklaşımları temsil etmektedir.

1. **Paar Algoritması:** Hamming ağırlığına dayalı açgözlü (greedy) bir algoritmadır.
2. **SLP Heuristic:** Doğrusal devre sentezinde kullanılan bu yöntem, hedef vektörlerin minimum sayıda XOR işlemi ile elde edilmesini hedefleyen klasik bir yaklaşımdır.
3. **Boyar SLP:** Matrislerdeki doğrusal devre sentezini daha pratik hale getirmek için tasarlanmış sezgisel bir algoritmadır. Geleneksel SLP Heuristic algoritmasından farklı olarak, optimizasyon sürecinde ayrıca devre derinliğini göz önünde bulundurur.

4. **SBP (Superior Boyar-Peralta):** Boyar-Peralta algoritmasının gelişmiş bir varyantı olan SBP algoritması, devre derinliğini de dikkate almasının yanı sıra doğrusal devrelerde XOR sayısını azaltmaya yönelik iyileştirilmiş stratejiler uygular.

Geliştirilen yazılımda çalıştırılan dört optimizasyon algoritmasının sonuçlarını gösteren ekran görüntüsü ise aşağıdaki şekildedir. Ayrıca her bir optimizasyon algoritmasında kullanılan adımları gösteren detaylar da yazılım üzerinden görüntülenebilmektedir.

Algoritma Sonuçları

Ham XOR Sayısı: 72

En Küçük XOR: 39

Grup: $F_2^{4-x^4+x^3+1-(4 \times 4)}$ -mds-semi-involutif-binary

Boyar SLP:

XOR: 39

Derinlik: 6

► Program

Paar Algoritması:

XOR: 48

► Program

SLP Heuristic:

XOR: 39

► Program

SBP Algoritması:

XOR: 39

Derinlik: 6

► Program

Şekil 4.3. Optimizasyon Algoritmaları Sonuçları

Bu çalışmada geliştirilen tekniğin uygulanmasıyla elde edilen 3×3 ve 4×4 yarı involutif MDS matrislerine dair deneysel veriler sırasıyla beşinci ve altıncı bölümlerde ele alınmaktadır.

Boyar SLP:

XOR: 39

Derinlik: 6

▼ Program

```
[  
    "t1 = x9 + x13 (1)",  
    "t2 = x0 + x5 (1)",  
    "t3 = x7 + x11 (1)",  
    "t4 = x8 + x12 (1)",  
    "t5 = x1 + x6 (1)",  
    "t6 = x2 + x4 (1)",  
    "t7 = x5 + t1 (2)",  
    "t8 = x3 + t7 * y11 (3)",  
    "t9 = x10 + x14 (1)",  
    "t10 = x15 + t2 (2)",  
    "t11 = x8 + t10 * y7 (3)",  
    "t12 = x1 + t3 (2)",  
    "t13 = x10 + t12 (3)",  
    "t14 = t1 + t13 * y3 (4)",  
    "t15 = x6 + t9 (2)",  
    "t16 = x0 + t15 * y8 (3)",  
    "t17 = t13 + t15 * y12 (4)",  
    "t18 = x9 + t5 (2)",  
    "t19 = x12 + t18 * y4 (3)",  
    "t20 = t2 + t18 (3)",  
    "t21 = t14 + t20 * y15 (5)",  
    "t22 = t4 + t7 (3)",  
    "t23 = t20 + t22 * y2 (4)",  
    "t24 = t6 + t22 * y10 (4)",  
    "t25 = x15 + t4 (2)",  
    "t26 = x4 + t25 (3)",  
    "t27 = t12 + t26 * y9 (4)",  
    "t28 = t8 + t11 (4)",  
    "t29 = t26 + t28 * y14 (5)",  
    "t30 = x7 + t6 (2)",  
    "t31 = t26 + t30 * y13 (4)",  
    "t32 = x11 + t22 (4)",  
    "t33 = t28 + t32 * y1 (5)",  
    "t34 = x10 + x13 (1)",  
    "t35 = t30 + t34 * y5 (3)",  
    "t36 = x11 + t6 (2)",  
    "t37 = t9 + t36 * y0 (3)",  
    "t38 = x14 + t26 (4)",  
    "t39 = t33 + t38 * y6 (6)"  
]
```

Şekil 4.4. Örnek Bir Optimizasyon İşleminin Adımları

BÖLÜM 5

3×3 BOYUTLARINDAKİ MATRİSLERİN ANALİZİ

Bu bölümde, $\mathbb{F}_{2^3}$ ve $\mathbb{F}_{2^4}$ sonlu cisimleri üzerinde, farklı indirgenemez polinomlar kullanılarak üretilen 3×3 boyutlu yarı-involutif MDS matrislerin analiz sonuçlarına yer verilmiştir. Elde edilen matris sayıları Çizelge 5.1’de özetlenmektedir.

Çizelge 5.1. 3×3 Boyutlarında Üretilen Yarı İnvolutif MDS Matris Sayıları

Sonlu Cisim	İndirgenemez Polinom	Matris Boyutu	Toplam Matris Sayısı
$\mathbb{F}_{2^3}$	$0xB$	3×3	7.056
$\mathbb{F}_{2^3}$	$0xD$	3×3	7.056
$\mathbb{F}_{2^4}$	$0x13$	3×3	529.200
$\mathbb{F}_{2^4}$	$0x19$	3×3	529.200

5.1. $\mathbb{F}_{2^3}/0xB$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0xB$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 5.1. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_1 = \begin{bmatrix} \alpha & 1 & \alpha^3 \\ \alpha^2 & \alpha^2 & \alpha^2 \\ \alpha^4 & \alpha & \alpha^5 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_1 = \begin{bmatrix} 1 & \alpha^6 & \alpha^2 \\ \alpha & \alpha & \alpha \\ \alpha^3 & 1 & \alpha^4 \end{bmatrix}$$

M_1 yarı-involutif MDS matrisinin ham XOR sayısı **32**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **18** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **17** olarak, **SBP** algoritmasına göre **5** derinlikte **17** olarak hesaplanmıştır.

M_1 matrisinin tersi:

$$M_1^{-1} = \begin{bmatrix} \alpha^2 & \alpha & \alpha^4 \\ \alpha^3 & \alpha^3 & \alpha^3 \\ \alpha^5 & \alpha^2 & \alpha^6 \end{bmatrix}$$

M_1^{-1} matrisinin ham XOR sayısı **43**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **22** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **19** olarak, **SBP** algoritmasına göre **5** derinlikte **19** olarak hesaplanmıştır.

M_1 ve M_1^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.2. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_2 = \begin{bmatrix} \alpha & 1 & \alpha^2 \\ \alpha^2 & \alpha^2 & \alpha \\ \alpha^5 & \alpha^2 & \alpha^5 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_2 = \begin{bmatrix} 1 & \alpha^6 & \alpha \\ \alpha & \alpha & 1 \\ \alpha^4 & \alpha & \alpha^4 \end{bmatrix}$$

M_2 yarı-involutif MDS matrisinin ham XOR sayısı **31**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **19** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **17** olarak, **SBP** algoritmasına göre **5** derinlikte **17** olarak hesaplanmıştır.

M_2 matrisinin tersi:

$$M_2^{-1} = \begin{bmatrix} \alpha^2 & \alpha & \alpha^3 \\ \alpha^3 & \alpha^3 & \alpha^2 \\ \alpha^6 & \alpha^3 & \alpha^6 \end{bmatrix}$$

M_2^{-1} matrisinin ham XOR sayısı **41**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **22** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **19** olarak, **SBP** algoritmasına göre **5** derinlikte **19** olarak hesaplanmıştır.

M_2 ve M_2^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.3. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_3 = \begin{bmatrix} \alpha & \alpha & \alpha^3 \\ \alpha & \alpha^2 & \alpha \\ \alpha^4 & \alpha^2 & \alpha^5 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_3 = \begin{bmatrix} 1 & 1 & \alpha^2 \\ 1 & \alpha & 1 \\ \alpha^3 & \alpha & \alpha^4 \end{bmatrix}$$

M_3 yarı-involutif MDS matrisinin ham XOR sayısı **30**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **18** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **17** olarak, **SBP** algoritmasına göre **4** derinlikte **17** olarak hesaplanmıştır.

M_3 matrisinin tersi:

$$M_3^{-1} = \begin{bmatrix} \alpha^2 & \alpha^2 & \alpha^4 \\ \alpha^2 & \alpha^3 & \alpha^2 \\ \alpha^5 & \alpha^3 & \alpha^6 \end{bmatrix}$$

M_3^{-1} matrisinin ham XOR sayısı **42**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **22** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **19** olarak, **SBP** algoritmasına

göre **4** derinlikte **19** olarak hesaplanmıştır.

M_3 ve M_3^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.4. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_4 = \begin{bmatrix} \alpha & \alpha & \alpha^4 \\ \alpha & \alpha^2 & \alpha^2 \\ \alpha^3 & \alpha & \alpha^5 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_4 = \begin{bmatrix} 1 & 1 & \alpha^3 \\ 1 & \alpha & \alpha \\ \alpha^2 & 1 & \alpha^4 \end{bmatrix}$$

M_4 yarı-involutif MDS matrisinin ham XOR sayısı **30**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **18** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **17** olarak, **SBP** algoritmasına göre **4** derinlikte **17** olarak hesaplanmıştır.

M_4 matrisinin tersi:

$$M_4^{-1} = \begin{bmatrix} \alpha^2 & \alpha^2 & \alpha^5 \\ \alpha^2 & \alpha^3 & \alpha^3 \\ \alpha^4 & \alpha^2 & \alpha^6 \end{bmatrix}$$

M_4^{-1} matrisinin ham XOR sayısı **42**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **23** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **19** olarak, **SBP** algoritmasına göre **4** derinlikte **19** olarak hesaplanmıştır.

M_4 ve M_4^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.5. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_5 = \begin{bmatrix} \alpha & \alpha & 1 \\ \alpha & \alpha^2 & \alpha^5 \\ 1 & \alpha^5 & \alpha^5 \end{bmatrix}$$

Aynı polinomun elemanı olan α^4 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_5 = \begin{bmatrix} \alpha^5 & \alpha^5 & \alpha^4 \\ \alpha^5 & \alpha^6 & \alpha^2 \\ \alpha^4 & \alpha^2 & \alpha^2 \end{bmatrix}$$

M_5 yarı-involutif MDS matrisinin ham XOR sayısı **42**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **20** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **17** olarak, **SBP** algoritmasına göre **7** derinlikte **17** olarak hesaplanmıştır.

M_5 matrisinin tersi:

$$M_5^{-1} = \begin{bmatrix} \alpha^4 & \alpha^4 & \alpha^3 \\ \alpha^4 & \alpha^5 & \alpha \\ \alpha^3 & \alpha & \alpha \end{bmatrix}$$

M_5^{-1} matrisinin ham XOR sayısı **44**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **19** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **19** olarak, **SBP** algoritmasına göre **7** derinlikte **19** olarak hesaplanmıştır.

M_5 ve M_5^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Çizelge 5.2. $\mathbb{F}_{2^3}/0xB$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
17	160
18	1680
19	2980
20	1799
21	401
22	36

5.2. $\mathbb{F}_{2^3}/0xD$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0xD$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 5.6. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_6 = \begin{bmatrix} \alpha^4 & \alpha^3 & \alpha^6 \\ \alpha^5 & \alpha^3 & \alpha^5 \\ \alpha & \alpha^5 & \alpha^5 \end{bmatrix}$$

Aynı polinomun elemanı olan α^2 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_6 = \begin{bmatrix} \alpha^6 & \alpha^5 & \alpha \\ 1 & \alpha^5 & 1 \\ \alpha^3 & 1 & 1 \end{bmatrix}$$

M_6 yarı-involutif MDS matrisinin ham XOR sayısı **28**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **19** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **16** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **16** olarak, **SBP** algoritmasına göre **4** derinlikte **16** olarak hesaplanmıştır.

M_6 matrisinin tersi:

$$M_6^{-1} = \begin{bmatrix} \alpha^2 & \alpha & \alpha^4 \\ \alpha^3 & \alpha & \alpha^3 \\ \alpha^6 & \alpha^3 & \alpha^3 \end{bmatrix}$$

M_6^{-1} matrisinin ham XOR sayısı **44**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **23** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **19** olarak, **SBP** algoritmasına göre **4** derinlikte **19** olarak hesaplanmıştır.

M_6 ve M_6^{-1} matrislerinin XOR farkı **3** olarak hesaplanmıştır.

Örnek 5.7. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_7 = \begin{bmatrix} \alpha & \alpha^5 & \alpha^6 \\ \alpha^3 & \alpha^2 & \alpha^4 \\ \alpha^5 & \alpha^5 & \alpha^4 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_7 = \begin{bmatrix} \alpha^2 & \alpha^6 & 1 \\ \alpha^4 & \alpha^3 & \alpha^5 \\ \alpha^6 & \alpha^6 & \alpha^5 \end{bmatrix}$$

M_7 yarı-involutif MDS matrisinin ham XOR sayısı **36**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **20** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **17** olarak, **SBP** algoritmasına göre **5** derinlikte **17** olarak hesaplanmıştır.

M_7 matrisinin tersi:

$$M_7^{-1} = \begin{bmatrix} 1 & \alpha^4 & \alpha^5 \\ \alpha^2 & \alpha & \alpha^3 \\ \alpha^4 & \alpha^4 & \alpha^3 \end{bmatrix}$$

M_7^{-1} matrisinin ham XOR sayısı **44**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **24** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **19** olarak, **SBP** algoritmasına göre **5** derinlikte **19** olarak hesaplanmıştır.

M_7 ve M_7^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.8. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_8 = \begin{bmatrix} \alpha & \alpha^6 & 1 \\ \alpha^2 & \alpha^2 & \alpha^4 \\ \alpha^4 & \alpha^5 & \alpha^4 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_8 = \begin{bmatrix} \alpha^2 & 1 & \alpha \\ \alpha^3 & \alpha^3 & \alpha^5 \\ \alpha^5 & \alpha^6 & \alpha^5 \end{bmatrix}$$

M_8 yarı-involutif MDS matrisinin ham XOR sayısı **37**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **20** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **17** olarak, **SBP** algoritmasına göre **6** derinlikte **17** olarak hesaplanmıştır.

M_8 matrisinin tersi:

$$M_8^{-1} = \begin{bmatrix} 1 & \alpha^5 & \alpha^6 \\ \alpha & \alpha & \alpha^3 \\ \alpha^3 & \alpha^4 & \alpha^3 \end{bmatrix}$$

M_8^{-1} matrisinin ham XOR sayısı **39**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **21** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **18** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **18** olarak, **SBP** algoritmasına göre **6** derinlikte **18** olarak hesaplanmıştır.

M_8 ve M_8^{-1} matrislerinin XOR farkı **1** olarak hesaplanmıştır.

Örnek 5.9. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_9 = \begin{bmatrix} \alpha & \alpha^6 & \alpha^3 \\ \alpha^2 & \alpha^2 & 1 \\ \alpha & \alpha^2 & \alpha^4 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_9 = \begin{bmatrix} 1 & \alpha^5 & \alpha^2 \\ \alpha & \alpha & \alpha^6 \\ 1 & \alpha & \alpha^3 \end{bmatrix}$$

M_9 yarı-involutif MDS matrisinin ham XOR sayısı **31**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **19** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **17** olarak, **SBP** algoritmasına göre **4** derinlikte **17** olarak hesaplanmıştır.

M_9 matrisinin tersi:

$$M_9^{-1} = \begin{bmatrix} \alpha^2 & 1 & \alpha^4 \\ \alpha^3 & \alpha^3 & \alpha \\ \alpha^2 & \alpha^3 & \alpha^5 \end{bmatrix}$$

M_9^{-1} matrisinin ham XOR sayısı **43**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **25** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **19** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **19** olarak, **SBP** algoritmasına

göre **4** derinlikte **19** olarak hesaplanmıştır.

M_9 ve M_9^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.10. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{10} = \begin{bmatrix} \alpha & \alpha & 1 \\ 1 & \alpha^2 & \alpha^2 \\ \alpha^4 & 1 & \alpha^4 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{10} = \begin{bmatrix} \alpha^2 & \alpha^2 & \alpha \\ \alpha & \alpha^3 & \alpha^3 \\ \alpha^5 & \alpha & \alpha^5 \end{bmatrix}$$

M_{10} yarı-involutif MDS matrisinin ham XOR sayısı **39**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **23** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **17** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **17** olarak, **SBP** algoritmasına göre **7** derinlikte **17** olarak hesaplanmıştır.

M_{10} matrisinin tersi:

$$M_{10}^{-1} = \begin{bmatrix} 1 & 1 & \alpha^6 \\ \alpha^6 & \alpha & \alpha \\ \alpha^3 & \alpha^6 & \alpha^3 \end{bmatrix}$$

M_{10}^{-1} matrisinin ham XOR sayısı **31**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **20** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **18** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **18** olarak, **SBP** algoritmasına göre **7** derinlikte **18** olarak hesaplanmıştır.

M_{10} ve M_{10}^{-1} matrislerinin XOR farkı **1** olarak hesaplanmıştır.

5.3. $\mathbb{F}_{2^4}/0x13$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0x13$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 5.11. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α

Çizelge 5.3. $\mathbb{F}_{2^3}/0xD$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
16	1
17	152
18	1644
19	3060
20	1777
21	392
22	30

kullanılarak elde edilen involutif MDS matris:

$$A_{11} = \begin{bmatrix} \alpha & \alpha^4 & \alpha^{12} \\ \alpha^{12} & \alpha^{11} & \alpha^5 \\ \alpha^{13} & \alpha^{13} & \alpha^{13} \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{11} = \begin{bmatrix} \alpha^2 & \alpha^5 & \alpha^{13} \\ \alpha^{13} & \alpha^{12} & \alpha^6 \\ \alpha^{14} & \alpha^{14} & \alpha^{14} \end{bmatrix}$$

M_{11} yarı-involutif MDS matrisinin ham XOR sayısı **51**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **26** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **22** olarak, **SBP** algoritmasına göre **6** derinlikte **22** olarak hesaplanmıştır.

M_{11} matrisinin tersi:

$$M_{11}^{-1} = \begin{bmatrix} 1 & \alpha^3 & \alpha^{11} \\ \alpha^{11} & \alpha^{10} & \alpha^4 \\ \alpha^{12} & \alpha^{12} & \alpha^{12} \end{bmatrix}$$

M_{11}^{-1} matrisinin ham XOR sayısı **75**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **35** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **26** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **108** olarak, **SBP** algoritmasına göre **6** derinlikte **108** olarak hesaplanmıştır.

M_{11} ve M_{11}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 5.12. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{12} = \begin{bmatrix} \alpha & \alpha^{12} & \alpha^4 \\ \alpha^{13} & \alpha^{13} & \alpha^{13} \\ \alpha^{12} & \alpha^5 & \alpha^{11} \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{12} = \begin{bmatrix} \alpha^2 & \alpha^{13} & \alpha^5 \\ \alpha^{14} & \alpha^{14} & \alpha^{14} \\ \alpha^{13} & \alpha^6 & \alpha^{12} \end{bmatrix}$$

M_{12} yarı-involutif MDS matrisinin ham XOR sayısı **51**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **26** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **22** olarak, **SBP** algoritmasına göre **6** derinlikte **22** olarak hesaplanmıştır.

M_{12} matrisinin tersi:

$$M_{12}^{-1} = \begin{bmatrix} 1 & \alpha^{11} & \alpha^3 \\ \alpha^{12} & \alpha^{12} & \alpha^{12} \\ \alpha^{11} & \alpha^4 & \alpha^{10} \end{bmatrix}$$

M_{12}^{-1} matrisinin ham XOR sayısı **75**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **26** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **120** olarak, **SBP** algoritmasına göre **6** derinlikte **120** olarak hesaplanmıştır.

M_{12} ve M_{12}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 5.13. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{13} = \begin{bmatrix} \alpha^2 & \alpha^{14} & \alpha^{14} \\ \alpha^8 & \alpha^3 & \alpha^{14} \\ 1 & \alpha^6 & \alpha^{13} \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{13} = \begin{bmatrix} \alpha^3 & 1 & 1 \\ \alpha^9 & \alpha^4 & 1 \\ \alpha & \alpha^7 & \alpha^{14} \end{bmatrix}$$

M_{13} yarı-involutif MDS matrisinin ham XOR sayısı **48**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **24** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **22** olarak, **SBP** algoritmasına göre **4** derinlikte **22** olarak hesaplanmıştır.

M_{13} matrisinin tersi:

$$M_{13}^{-1} = \begin{bmatrix} \alpha & \alpha^{13} & \alpha^{13} \\ \alpha^7 & \alpha^2 & \alpha^{13} \\ \alpha^{14} & \alpha^5 & \alpha^{12} \end{bmatrix}$$

M_{13}^{-1} matrisinin ham XOR sayısı **54**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **28** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **24** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **24** olarak, **SBP** algoritmasına göre **4** derinlikte **24** olarak hesaplanmıştır.

M_{13} ve M_{13}^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.14. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{14} = \begin{bmatrix} \alpha^2 & \alpha^{14} & \alpha^{14} \\ 1 & \alpha^{13} & \alpha^6 \\ \alpha^8 & \alpha^{14} & \alpha^3 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{14} = \begin{bmatrix} \alpha^3 & 1 & 1 \\ \alpha & \alpha^{14} & \alpha^7 \\ \alpha^9 & 1 & \alpha^4 \end{bmatrix}$$

M_{14} yarı-involutif MDS matrisinin ham XOR sayısı **48**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **24** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **22** olarak, **SBP** algoritmasına göre **4** derinlikte **22** olarak hesaplanmıştır.

M_{14} matrisinin tersi:

$$M_{14}^{-1} = \begin{bmatrix} \alpha & \alpha^{13} & \alpha^{13} \\ \alpha^{14} & \alpha^{12} & \alpha^5 \\ \alpha^7 & \alpha^{13} & \alpha^2 \end{bmatrix}$$

M_{14}^{-1} matrisinin ham XOR sayısı **54**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **28** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **24** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **24** olarak, **SBP** algoritmasına göre **4** derinlikte **24** olarak hesaplanmıştır.

M_{14} ve M_{14}^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.15. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{15} = \begin{bmatrix} \alpha^3 & \alpha^8 & \alpha^{14} \\ \alpha^{14} & \alpha^2 & \alpha^{14} \\ \alpha^6 & 1 & \alpha^{13} \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{15} = \begin{bmatrix} \alpha^4 & \alpha^9 & 1 \\ 1 & \alpha^3 & 1 \\ \alpha^7 & \alpha & \alpha^{14} \end{bmatrix}$$

M_{15} yarı-involutif MDS matrisinin ham XOR sayısı **48**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **24** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **22** olarak, **SBP** algoritmasına göre **4** derinlikte **22** olarak hesaplanmıştır.

M_{15} matrisinin tersi:

$$M_{15}^{-1} = \begin{bmatrix} \alpha^2 & \alpha^7 & \alpha^{13} \\ \alpha^{13} & \alpha & \alpha^{13} \\ \alpha^5 & \alpha^{14} & \alpha^{12} \end{bmatrix}$$

M_{15}^{-1} matrisinin ham XOR sayısı **54**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **28** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **24** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **24** olarak, **SBP** algoritmasına

göre 4 derinlikte 24 olarak hesaplanmıştır.

M_{15} ve M_{15}^{-1} matrislerinin XOR farkı 2 olarak hesaplanmıştır.

Çizelge 5.4. $\mathbb{F}_{2^4}/0x13$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
22	12
23	105
24	499
25	2260
26	5358
27	13919
28	31768
29	63126
30	96642
31	117895
32	104821
33	63383
34	23815
35	4916
36	647
37	34

5.4. $\mathbb{F}_{2^4}/0x19$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0x19$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 5.16. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{16} = \begin{bmatrix} \alpha^2 & \alpha^2 & \alpha^2 \\ \alpha^{10} & \alpha^4 & \alpha^3 \\ \alpha^3 & \alpha^{11} & \alpha^{14} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{16} = \begin{bmatrix} \alpha & \alpha & \alpha \\ \alpha^9 & \alpha^3 & \alpha^2 \\ \alpha^2 & \alpha^{10} & \alpha^{13} \end{bmatrix}$$

M_{16} yarı-involutif MDS matrisinin ham XOR sayısı **51**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **28** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **22** olarak, **SBP** algoritmasına göre **6** derinlikte **22** olarak hesaplanmıştır.

M_{16} matrisinin tersi:

$$M_{16}^{-1} = \begin{bmatrix} \alpha^3 & \alpha^3 & \alpha^3 \\ \alpha^{11} & \alpha^5 & \alpha^4 \\ \alpha^4 & \alpha^{12} & 1 \end{bmatrix}$$

M_{16}^{-1} matrisinin ham XOR sayısı **75**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **26** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **27** olarak, **SBP** algoritmasına göre **6** derinlikte **27** olarak hesaplanmıştır.

M_{16} ve M_{16}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 5.17. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{17} = \begin{bmatrix} \alpha^2 & \alpha^9 & 1 \\ \alpha & \alpha^{12} & \alpha^7 \\ \alpha & \alpha & \alpha^{13} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{17} = \begin{bmatrix} \alpha & \alpha^8 & \alpha^{14} \\ 1 & \alpha^{11} & \alpha^6 \\ 1 & 1 & \alpha^{12} \end{bmatrix}$$

M_{17} yarı-involutif MDS matrisinin ham XOR sayısı **48**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **24** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **22** olarak, **SBP** algoritmasına göre **4** derinlikte **22** olarak hesaplanmıştır.

M_{17} matrisinin tersi:

$$M_{17}^{-1} = \begin{bmatrix} \alpha^3 & \alpha^{10} & \alpha \\ \alpha^2 & \alpha^{13} & \alpha^8 \\ \alpha^2 & \alpha^2 & \alpha^{14} \end{bmatrix}$$

M_{17}^{-1} matrisinin ham XOR sayısı **54**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **26** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **24** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **24** olarak, **SBP** algoritmasına göre **4** derinlikte **24** olarak hesaplanmıştır.

M_{17} ve M_{17}^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.18. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{18} = \begin{bmatrix} \alpha^2 & 1 & \alpha^9 \\ \alpha & \alpha^{13} & \alpha \\ \alpha & \alpha^7 & \alpha^{12} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{18} = \begin{bmatrix} \alpha & \alpha^{14} & \alpha^8 \\ 1 & \alpha^{12} & 1 \\ 1 & \alpha^6 & \alpha^{11} \end{bmatrix}$$

M_{18} yarı-involutif MDS matrisinin ham XOR sayısı **48**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **24** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **22** olarak, **SBP** algoritmasına göre **4** derinlikte **22** olarak hesaplanmıştır.

M_{18} matrisinin tersi:

$$M_{18}^{-1} = \begin{bmatrix} \alpha^3 & \alpha & \alpha^{10} \\ \alpha^2 & \alpha^{14} & \alpha^2 \\ \alpha^2 & \alpha^8 & \alpha^{13} \end{bmatrix}$$

M_{18}^{-1} matrisinin ham XOR sayısı **54**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **26** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **24** olarak, **BP** Derinlik Öncelikli algoritmasına göre **4** derinlikte **24** olarak, **SBP** algoritmasına göre **4** derinlikte **24** olarak hesaplanmıştır.

M_{18} ve M_{18}^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 5.19. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{19} = \begin{bmatrix} \alpha^2 & \alpha^2 & \alpha^2 \\ \alpha^3 & \alpha^{14} & \alpha^{11} \\ \alpha^{10} & \alpha^3 & \alpha^4 \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{19} = \begin{bmatrix} \alpha & \alpha & \alpha \\ \alpha^2 & \alpha^{13} & \alpha^{10} \\ \alpha^9 & \alpha^2 & \alpha^3 \end{bmatrix}$$

M_{19} yarı-involutif MDS matrisinin ham XOR sayısı **51**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **26** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **22** olarak, **SBP** algoritmasına göre **6** derinlikte **22** olarak hesaplanmıştır.

M_{19} matrisinin tersi:

$$M_{19}^{-1} = \begin{bmatrix} \alpha^3 & \alpha^3 & \alpha^3 \\ \alpha^4 & 1 & \alpha^{12} \\ \alpha^{11} & \alpha^4 & \alpha^5 \end{bmatrix}$$

M_{19}^{-1} matrisinin ham XOR sayısı **75**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **26** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **27** olarak, **SBP** algoritmasına göre **6** derinlikte **27** olarak hesaplanmıştır.

M_{19} ve M_{19}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 5.20. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{20} = \begin{bmatrix} \alpha^4 & \alpha^{10} & \alpha^3 \\ \alpha^2 & \alpha^2 & \alpha^2 \\ \alpha^{11} & \alpha^3 & \alpha^{14} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{20} = \begin{bmatrix} \alpha^3 & \alpha^9 & \alpha^2 \\ \alpha & \alpha & \alpha \\ \alpha^{10} & \alpha^2 & \alpha^{13} \end{bmatrix}$$

M_{20} yarı-involutif MDS matrisinin ham XOR sayısı **51**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **28** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **22** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **22** olarak, **SBP** algoritmasına göre **6** derinlikte **22** olarak hesaplanmıştır.

M_{20} matrisinin tersi:

$$M_{20}^{-1} = \begin{bmatrix} \alpha^5 & \alpha^{11} & \alpha^4 \\ \alpha^3 & \alpha^3 & \alpha^3 \\ \alpha^{12} & \alpha^4 & 1 \end{bmatrix}$$

M_{20}^{-1} matrisinin ham XOR sayısı **75**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **33** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **26** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **27** olarak, **SBP** algoritmasına göre **6** derinlikte **27** olarak hesaplanmıştır.

M_{20} ve M_{20}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Çizelge 5.5. $\mathbb{F}_{2^4}/0x19$ Cisminde Üretilen 3×3 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
22	12
23	108
24	503
25	2264
26	5385
27	13947
28	31880
29	63069
30	97122
31	117994
32	104463
33	62797
34	23847
35	5133
36	640
37	36

BÖLÜM 6

4×4 BOYUTLARINDAKİ MATRİSLERİN ANALİZİ

Bu bölümde, $\mathbb{F}_{2^3}$ ve $\mathbb{F}_{2^4}$ sonlu cisimleri üzerinde tanımlı 4×4 boyutlu yarı-involutif MDS matrislerin analiz sonuçlarına yer verilmiştir. Elde edilen matris sayıları Çizelge 6.1’de özetlenmektedir.

Çizelge 6.1. 4×4 Boyutlarında Üretilen Yarı İnvolutif MDS Matris Sayıları

Sonlu Cisim	İndirgenemez Polinom	Matris Boyutu	Toplam Matris Sayısı
$\mathbb{F}_{2^3}$	$0xB$	4×4	98.784
$\mathbb{F}_{2^3}$	$0xD$	4×4	98.784
$\mathbb{F}_{2^4}$	$0x13$	4×4	393.792
$\mathbb{F}_{2^4}$	$0x19$	4×4	393.792

6.1. $\mathbb{F}_{2^3}/0xB$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0xB$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 6.1. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{21} = \begin{bmatrix} 1 & \alpha^2 & \alpha^2 & \alpha^6 \\ 1 & 1 & \alpha^6 & 1 \\ \alpha^6 & \alpha^5 & 1 & 1 \\ \alpha^5 & \alpha & \alpha^2 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{21} = \begin{bmatrix} \alpha & \alpha^3 & \alpha^3 & 1 \\ \alpha & \alpha & 1 & \alpha \\ 1 & \alpha^6 & \alpha & \alpha \\ \alpha^6 & \alpha^2 & \alpha^3 & \alpha \end{bmatrix}$$

M_{21} yarı-involatif MDS matrisinin ham XOR sayısı **59**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **35** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **29** olarak, **SBP** algoritmasına göre **5** derinlikte **29** olarak hesaplanmıştır.

M_{21} matrisinin tersi:

$$M_{21}^{-1} = \begin{bmatrix} \alpha^6 & \alpha & \alpha & \alpha^5 \\ \alpha^6 & \alpha^6 & \alpha^5 & \alpha^6 \\ \alpha^5 & \alpha^4 & \alpha^6 & \alpha^6 \\ \alpha^4 & 1 & \alpha & \alpha^6 \end{bmatrix}$$

M_{21}^{-1} matrisinin ham XOR sayısı **63**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **37** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **32** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **32** olarak, **SBP** algoritmasına göre **5** derinlikte **32** olarak hesaplanmıştır.

M_{21} ve M_{21}^{-1} matrislerinin XOR farkı **3** olarak hesaplanmıştır.

Örnek 6.2. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involatif MDS matris:

$$A_{22} = \begin{bmatrix} 1 & \alpha^3 & \alpha^6 & \alpha \\ \alpha^6 & 1 & \alpha^2 & \alpha \\ \alpha^2 & \alpha^2 & 1 & \alpha^5 \\ \alpha^3 & 1 & \alpha^4 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involatif MDS matris:

$$M_{22} = \begin{bmatrix} \alpha^6 & \alpha^2 & \alpha^5 & 1 \\ \alpha^5 & \alpha^6 & \alpha & 1 \\ \alpha & \alpha & \alpha^6 & \alpha^4 \\ \alpha^2 & \alpha^6 & \alpha^3 & \alpha^6 \end{bmatrix}$$

M_{22} yarı-involutif MDS matrisinin ham XOR sayısı **62**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **29** olarak, **SBP** algoritmasına göre **6** derinlikte **29** olarak hesaplanmıştır.

M_{22} matrisinin tersi:

$$M_{22}^{-1} = \begin{bmatrix} \alpha & \alpha^4 & 1 & \alpha^2 \\ 1 & \alpha & \alpha^3 & \alpha^2 \\ \alpha^3 & \alpha^3 & \alpha & \alpha^6 \\ \alpha^4 & \alpha & \alpha^5 & \alpha \end{bmatrix}$$

M_{22}^{-1} matrisinin ham XOR sayısı **69**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **38** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **34** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **34** olarak, **SBP** algoritmasına göre **6** derinlikte **34** olarak hesaplanmıştır.

M_{22} ve M_{22}^{-1} matrislerinin XOR farkı **5** olarak hesaplanmıştır.

Örnek 6.3. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{23} = \begin{bmatrix} 1 & \alpha^4 & 1 & \alpha^2 \\ \alpha^5 & 1 & \alpha^2 & \alpha \\ \alpha & \alpha^2 & 1 & \alpha^5 \\ \alpha^2 & 1 & \alpha^4 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{23} = \begin{bmatrix} \alpha^6 & \alpha^3 & \alpha^6 & \alpha \\ \alpha^4 & \alpha^6 & \alpha & 1 \\ 1 & \alpha & \alpha^6 & \alpha^4 \\ \alpha & \alpha^6 & \alpha^3 & \alpha^6 \end{bmatrix}$$

M_{23} yarı-involutif MDS matrisinin ham XOR sayısı **62**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **32** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **29** olarak, **SBP** algoritmasına göre **7** derinlikte **29** olarak hesaplanmıştır.

M_{23} matrisinin tersi:

$$M_{23}^{-1} = \begin{bmatrix} \alpha & \alpha^5 & \alpha & \alpha^3 \\ \alpha^6 & \alpha & \alpha^3 & \alpha^2 \\ \alpha^2 & \alpha^3 & \alpha & \alpha^6 \\ \alpha^3 & \alpha & \alpha^5 & \alpha \end{bmatrix}$$

M_{23}^{-1} matrisinin ham XOR sayısı **70**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **37** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **32** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **32** olarak, **SBP** algoritmasına göre **7** derinlikte **32** olarak hesaplanmıştır.

M_{23} ve M_{23}^{-1} matrislerinin XOR farkı **3** olarak hesaplanmıştır.

Örnek 6.4. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{24} = \begin{bmatrix} 1 & \alpha^5 & \alpha & \alpha^2 \\ \alpha^4 & 1 & \alpha^2 & 1 \\ 1 & \alpha^2 & 1 & \alpha^4 \\ \alpha^2 & \alpha & \alpha^5 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{24} = \begin{bmatrix} \alpha^6 & \alpha^4 & 1 & \alpha \\ \alpha^3 & \alpha^6 & \alpha & \alpha^6 \\ \alpha^6 & \alpha & \alpha^6 & \alpha^3 \\ \alpha & 1 & \alpha^4 & \alpha^6 \end{bmatrix}$$

M_{24} yarı-involutif MDS matrisinin ham XOR sayısı **62**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **32** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **29** olarak, **SBP** algoritmasına göre **7** derinlikte **29** olarak hesaplanmıştır.

M_{24} matrisinin tersi:

$$M_{24}^{-1} = \begin{bmatrix} \alpha & \alpha^6 & \alpha^2 & \alpha^3 \\ \alpha^5 & \alpha & \alpha^3 & \alpha \\ \alpha & \alpha^3 & \alpha & \alpha^5 \\ \alpha^3 & \alpha^2 & \alpha^6 & \alpha \end{bmatrix}$$

M_{24}^{-1} matrisinin ham XOR sayısı **70**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **37** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **32** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **32** olarak, **SBP** algoritmasına göre **7** derinlikte **32** olarak hesaplanmıştır.

M_{24} ve M_{24}^{-1} matrislerinin XOR farkı **3** olarak hesaplanmıştır.

Örnek 6.5. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xB$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{25} = \begin{bmatrix} 1 & 1 & 1 & \alpha^6 \\ \alpha^2 & 1 & \alpha^6 & \alpha^2 \\ \alpha & \alpha^5 & 1 & \alpha^2 \\ \alpha^5 & \alpha^6 & 1 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{25} = \begin{bmatrix} \alpha & \alpha & \alpha & 1 \\ \alpha^3 & \alpha & 1 & \alpha^3 \\ \alpha^2 & \alpha^6 & \alpha & \alpha^3 \\ \alpha^6 & 1 & \alpha & \alpha \end{bmatrix}$$

M_{25} yarı-involutif MDS matrisinin ham XOR sayısı **59**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **35** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **29** olarak, **SBP** algoritmasına göre **5** derinlikte **29** olarak hesaplanmıştır.

M_{25} matrisinin tersi:

$$M_{25}^{-1} = \begin{bmatrix} \alpha^6 & \alpha^6 & \alpha^6 & \alpha^5 \\ \alpha & \alpha^6 & \alpha^5 & \alpha \\ 1 & \alpha^4 & \alpha^6 & \alpha \\ \alpha^4 & \alpha^5 & \alpha^6 & \alpha^6 \end{bmatrix}$$

M_{25}^{-1} matrisinin ham XOR sayısı **63**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **32** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **32** olarak, **SBP** algoritmasına göre **5** derinlikte **32** olarak hesaplanmıştır.

M_{25} ve M_{25}^{-1} matrislerinin XOR farkı **3** olarak hesaplanmıştır.

Çizelge 6.2. $\mathbb{F}_{2^3}/0xB$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
29	245
30	2648
31	13814
32	30419
33	31936
34	15616
35	3573
36	521
37	12

6.2. $\mathbb{F}_{2^3}/0xD$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0xD$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 6.6. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{26} = \begin{bmatrix} 1 & \alpha^5 & \alpha^5 & \alpha^2 \\ \alpha^5 & 1 & \alpha & \alpha^6 \\ \alpha & \alpha^4 & 1 & \alpha^6 \\ \alpha^3 & 1 & \alpha^4 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{26} = \begin{bmatrix} \alpha & \alpha^6 & \alpha^6 & \alpha^3 \\ \alpha^6 & \alpha & \alpha^2 & 1 \\ \alpha^2 & \alpha^5 & \alpha & 1 \\ \alpha^4 & \alpha & \alpha^5 & \alpha \end{bmatrix}$$

M_{26} yarı-involutif MDS matrisinin ham XOR sayısı **62**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **29** olarak, **SBP** algoritmasına göre **6** derinlikte **29** olarak hesaplanmıştır.

M_{26} matrisinin tersi:

$$M_{26}^{-1} = \begin{bmatrix} \alpha^6 & \alpha^4 & \alpha^4 & \alpha \\ \alpha^4 & \alpha^6 & 1 & \alpha^5 \\ 1 & \alpha^3 & \alpha^6 & \alpha^5 \\ \alpha^2 & \alpha^6 & \alpha^3 & \alpha^6 \end{bmatrix}$$

M_{26}^{-1} matrisinin ham XOR sayısı **69**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **38** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **33** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **33** olarak, **SBP** algoritmasına göre **6** derinlikte **33** olarak hesaplanmıştır.

M_{26} ve M_{26}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 6.7. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{27} = \begin{bmatrix} 1 & \alpha^5 & \alpha^6 & \alpha \\ \alpha^5 & 1 & \alpha^2 & \alpha^5 \\ 1 & \alpha^3 & 1 & \alpha^4 \\ \alpha^4 & \alpha & \alpha^6 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{27} = \begin{bmatrix} \alpha & \alpha^6 & 1 & \alpha^2 \\ \alpha^6 & \alpha & \alpha^3 & \alpha^6 \\ \alpha & \alpha^4 & \alpha & \alpha^5 \\ \alpha^5 & \alpha^2 & 1 & \alpha \end{bmatrix}$$

M_{27} yarı-involutif MDS matrisinin ham XOR sayısı **62**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **29** olarak, **SBP** algoritmasına göre **6** derinlikte **29** olarak hesaplanmıştır.

M_{27} matrisinin tersi:

$$M_{27}^{-1} = \begin{bmatrix} \alpha^6 & \alpha^4 & \alpha^5 & 1 \\ \alpha^4 & \alpha^6 & \alpha & \alpha^4 \\ \alpha^6 & \alpha^2 & \alpha^6 & \alpha^3 \\ \alpha^3 & 1 & \alpha^5 & \alpha^6 \end{bmatrix}$$

M_{27}^{-1} matrisinin ham XOR sayısı **69**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **38** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **34** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **34** olarak, **SBP** algoritmasına göre **6** derinlikte **34** olarak hesaplanmıştır.

M_{27} ve M_{27}^{-1} matrislerinin XOR farkı **5** olarak hesaplanmıştır.

Örnek 6.8. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{28} = \begin{bmatrix} 1 & \alpha^5 & \alpha^6 & \alpha^2 \\ \alpha^5 & 1 & \alpha^2 & \alpha^6 \\ 1 & \alpha^3 & 1 & \alpha^5 \\ \alpha^3 & 1 & \alpha^5 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{28} = \begin{bmatrix} \alpha & \alpha^6 & 1 & \alpha^3 \\ \alpha^6 & \alpha & \alpha^3 & 1 \\ \alpha & \alpha^4 & \alpha & \alpha^6 \\ \alpha^4 & \alpha & \alpha^6 & \alpha \end{bmatrix}$$

M_{28} yarı-involutif MDS matrisinin ham XOR sayısı **62**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **32** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **29** olarak, **SBP** algoritmasına göre **7** derinlikte **29** olarak hesaplanmıştır.

M_{28} matrisinin tersi:

$$M_{28}^{-1} = \begin{bmatrix} \alpha^6 & \alpha^4 & \alpha^5 & \alpha \\ \alpha^4 & \alpha^6 & \alpha & \alpha^5 \\ \alpha^6 & \alpha^2 & \alpha^6 & \alpha^4 \\ \alpha^2 & \alpha^6 & \alpha^4 & \alpha^6 \end{bmatrix}$$

M_{28}^{-1} matrisinin ham XOR sayısı **70**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **35** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **32** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **32** olarak, **SBP** algoritmasına göre **7** derinlikte **32** olarak hesaplanmıştır.

M_{28} ve M_{28}^{-1} matrislerinin XOR farkı **3** olarak hesaplanmıştır.

Örnek 6.9. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{29} = \begin{bmatrix} 1 & \alpha^5 & 1 & \alpha^3 \\ \alpha^5 & 1 & \alpha^3 & 1 \\ \alpha^6 & \alpha^2 & 1 & \alpha^5 \\ \alpha^2 & \alpha^6 & \alpha^5 & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{29} = \begin{bmatrix} \alpha & \alpha^6 & \alpha & \alpha^4 \\ \alpha^6 & \alpha & \alpha^4 & \alpha \\ 1 & \alpha^3 & \alpha & \alpha^6 \\ \alpha^3 & 1 & \alpha^6 & \alpha \end{bmatrix}$$

M_{29} yarı-involutif MDS matrisinin ham XOR sayısı **62**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **34** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **29** olarak, **SBP** algoritmasına göre **7** derinlikte **29** olarak hesaplanmıştır.

M_{29} matrisinin tersi:

$$M_{29}^{-1} = \begin{bmatrix} \alpha^6 & \alpha^4 & \alpha^6 & \alpha^2 \\ \alpha^4 & \alpha^6 & \alpha^2 & \alpha^6 \\ \alpha^5 & \alpha & \alpha^6 & \alpha^4 \\ \alpha & \alpha^5 & \alpha^4 & \alpha^6 \end{bmatrix}$$

M_{29}^{-1} matrisinin ham XOR sayısı **70**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **35** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **31** olarak, **BP** Derinlik Öncelikli algoritmasına göre **7** derinlikte **31** olarak, **SBP** algoritmasına göre **7** derinlikte **31** olarak hesaplanmıştır.

M_{29} ve M_{29}^{-1} matrislerinin XOR farkı **2** olarak hesaplanmıştır.

Örnek 6.10. $\mathbb{F}_{2^3}$ cismi üzerinde tanımlı $p(\alpha) = 0xD$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{30} = \begin{bmatrix} 1 & \alpha & \alpha^5 & \alpha^5 \\ \alpha^2 & 1 & \alpha^5 & \alpha^6 \\ \alpha & 1 & 1 & \alpha^2 \\ 1 & 1 & \alpha & 1 \end{bmatrix}$$

Aynı polinomun elemanı olan α^6 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{30} = \begin{bmatrix} \alpha^6 & 1 & \alpha^4 & \alpha^4 \\ \alpha & \alpha^6 & \alpha^4 & \alpha^5 \\ 1 & \alpha^6 & \alpha^6 & \alpha \\ \alpha^6 & \alpha^6 & 1 & \alpha^6 \end{bmatrix}$$

M_{30} yarı-involutif MDS matrisinin ham XOR sayısı **59**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **35** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **29** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **29** olarak, **SBP** algoritmasına göre **5** derinlikte **29** olarak hesaplanmıştır.

M_{30} matrisinin tersi:

$$M_{30}^{-1} = \begin{bmatrix} \alpha & \alpha^2 & \alpha^6 & \alpha^6 \\ \alpha^3 & \alpha & \alpha^6 & 1 \\ \alpha^2 & \alpha & \alpha & \alpha^3 \\ \alpha & \alpha & \alpha^2 & \alpha \end{bmatrix}$$

M_{30}^{-1} matrisinin ham XOR sayısı **63**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **36** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **32** olarak, **BP** Derinlik Öncelikli algoritmasına göre **5** derinlikte **32** olarak, **SBP** algoritmasına göre **5** derinlikte **32** olarak hesaplanmıştır.

M_{30} ve M_{30}^{-1} matrislerinin XOR farkı **3** olarak hesaplanmıştır.

6.3. $\mathbb{F}_{2^4}/0x13$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0x13$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 6.11. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

Çizelge 6.3. $\mathbb{F}_{2^3}/0xD$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
29	201
30	2770
31	13731
32	30618
33	31730
34	15547
35	3657
36	510
37	19
38	1

$$A_{31} = \begin{bmatrix} \alpha & \alpha & \alpha^{14} & \alpha^{13} \\ 1 & \alpha & \alpha^{12} & \alpha^{14} \\ 1 & \alpha^{14} & \alpha & \alpha \\ \alpha^{13} & 1 & 1 & \alpha \end{bmatrix}$$

Aynı polinomun elemanı olan α^2 ile çarpılarak elde edilen yarı-ınvolutif MDS matris:

$$M_{31} = \begin{bmatrix} \alpha^3 & \alpha^3 & \alpha & 1 \\ \alpha^2 & \alpha^3 & \alpha^{14} & \alpha \\ \alpha^2 & \alpha & \alpha^3 & \alpha^3 \\ 1 & \alpha^2 & \alpha^2 & \alpha^3 \end{bmatrix}$$

M_{31} yarı-ınvolutif MDS matrisinin ham XOR sayısı **78**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **50** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **39** olarak, **SBP** algoritmasına göre **8** derinlikte **39** olarak hesaplanmıştır.

M_{31} matrisinin tersi:

$$M_{31}^{-1} = \begin{bmatrix} \alpha^{14} & \alpha^{14} & \alpha^{12} & \alpha^{11} \\ \alpha^{13} & \alpha^{14} & \alpha^{10} & \alpha^{12} \\ \alpha^{13} & \alpha^{12} & \alpha^{14} & \alpha^{14} \\ \alpha^{11} & \alpha^{13} & \alpha^{13} & \alpha^{14} \end{bmatrix}$$

M_{31}^{-1} matrisinin ham XOR sayısı **109**, optimizasyon yöntemleri ile elde edilen

XOR sayısı **Paar** algoritmasına göre **54** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **45** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **45** olarak, **SBP** algoritmasına göre **8** derinlikte **45** olarak hesaplanmıştır.

M_{31} ve M_{31}^{-1} matrislerinin XOR farkı **6** olarak hesaplanmıştır.

Örnek 6.12. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{32} = \begin{bmatrix} \alpha & 1 & \alpha^{12} & \alpha^{14} \\ \alpha & \alpha & \alpha^{14} & \alpha^{13} \\ \alpha^{14} & 1 & \alpha & \alpha \\ 1 & \alpha^{13} & 1 & \alpha \end{bmatrix}$$

Aynı polinomun elemanı olan α^2 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{32} = \begin{bmatrix} \alpha^3 & \alpha^2 & \alpha^{14} & \alpha \\ \alpha^3 & \alpha^3 & \alpha & 1 \\ \alpha & \alpha^2 & \alpha^3 & \alpha^3 \\ \alpha^2 & 1 & \alpha^2 & \alpha^3 \end{bmatrix}$$

M_{32} yarı-involutif MDS matrisinin ham XOR sayısı **78**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **53** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **39** olarak, **SBP** algoritmasına göre **8** derinlikte **39** olarak hesaplanmıştır.

M_{32} matrisinin tersi:

$$M_{32}^{-1} = \begin{bmatrix} \alpha^{14} & \alpha^{13} & \alpha^{10} & \alpha^{12} \\ \alpha^{14} & \alpha^{14} & \alpha^{12} & \alpha^{11} \\ \alpha^{12} & \alpha^{13} & \alpha^{14} & \alpha^{14} \\ \alpha^{13} & \alpha^{11} & \alpha^{13} & \alpha^{14} \end{bmatrix}$$

M_{32}^{-1} matrisinin ham XOR sayısı **109**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **53** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **43** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **43** olarak, **SBP** algoritmasına göre **8** derinlikte **43** olarak hesaplanmıştır.

M_{32} ve M_{32}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 6.13. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α

kullanılarak elde edilen involutif MDS matris:

$$A_{33} = \begin{bmatrix} \alpha & 1 & 1 & \alpha^{14} \\ \alpha^{14} & \alpha & \alpha^{12} & \alpha \\ \alpha & \alpha^{14} & \alpha & \alpha \\ \alpha^{12} & 1 & \alpha^{13} & \alpha \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{33} = \begin{bmatrix} \alpha^2 & \alpha & \alpha & 1 \\ 1 & \alpha^2 & \alpha^{13} & \alpha^2 \\ \alpha^2 & 1 & \alpha^2 & \alpha^2 \\ \alpha^{13} & \alpha & \alpha^{14} & \alpha^2 \end{bmatrix}$$

M_{33} yarı-involutif MDS matrisinin ham XOR sayısı **72**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **48** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **39** olarak, **SBP** algoritmasına göre **6** derinlikte **39** olarak hesaplanmıştır.

M_{33} matrisinin tersi:

$$M_{33}^{-1} = \begin{bmatrix} 1 & \alpha^{14} & \alpha^{14} & \alpha^{13} \\ \alpha^{13} & 1 & \alpha^{11} & 1 \\ 1 & \alpha^{13} & 1 & 1 \\ \alpha^{11} & \alpha^{14} & \alpha^{12} & 1 \end{bmatrix}$$

M_{33}^{-1} matrisinin ham XOR sayısı **82**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **48** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **45** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **45** olarak, **SBP** algoritmasına göre **6** derinlikte **45** olarak hesaplanmıştır.

M_{33} ve M_{33}^{-1} matrislerinin XOR farkı **6** olarak hesaplanmıştır.

Örnek 6.14. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{34} = \begin{bmatrix} \alpha & \alpha & \alpha^{13} & \alpha^{14} \\ 1 & \alpha & \alpha^{14} & \alpha^{12} \\ \alpha^{13} & 1 & \alpha & 1 \\ 1 & \alpha^{14} & \alpha & \alpha \end{bmatrix}$$

Aynı polinomun elemanı olan α^2 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{34} = \begin{bmatrix} \alpha^3 & \alpha^3 & 1 & \alpha \\ \alpha^2 & \alpha^3 & \alpha & \alpha^{14} \\ 1 & \alpha^2 & \alpha^3 & \alpha^2 \\ \alpha^2 & \alpha & \alpha^3 & \alpha^3 \end{bmatrix}$$

M_{34} yarı-involutif MDS matrisinin ham XOR sayısı **78**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **50** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **39** olarak, **SBP** algoritmasına göre **8** derinlikte **39** olarak hesaplanmıştır.

M_{34} matrisinin tersi:

$$M_{34}^{-1} = \begin{bmatrix} \alpha^{14} & \alpha^{14} & \alpha^{11} & \alpha^{12} \\ \alpha^{13} & \alpha^{14} & \alpha^{12} & \alpha^{10} \\ \alpha^{11} & \alpha^{13} & \alpha^{14} & \alpha^{13} \\ \alpha^{13} & \alpha^{12} & \alpha^{14} & \alpha^{14} \end{bmatrix}$$

M_{34}^{-1} matrisinin ham XOR sayısı **109**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **54** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **43** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **43** olarak, **SBP** algoritmasına göre **8** derinlikte **43** olarak hesaplanmıştır.

M_{34} ve M_{34}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 6.15. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x13$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{35} = \begin{bmatrix} \alpha & \alpha^{14} & \alpha^{12} & \alpha \\ 1 & \alpha & 1 & \alpha^{14} \\ \alpha^{14} & \alpha & \alpha & \alpha \\ 1 & \alpha^{12} & \alpha^{13} & \alpha \end{bmatrix}$$

Aynı polinomun elemanı olan α^1 ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{35} = \begin{bmatrix} \alpha^2 & 1 & \alpha^{13} & \alpha^2 \\ \alpha & \alpha^2 & \alpha & 1 \\ 1 & \alpha^2 & \alpha^2 & \alpha^2 \\ \alpha & \alpha^{13} & \alpha^{14} & \alpha^2 \end{bmatrix}$$

M_{35} yarı-involutif MDS matrisinin ham XOR sayısı **72**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **48** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **39** olarak, **SBP** algoritmasına göre **6** derinlikte **39** olarak hesaplanmıştır.

M_{35} matrisinin tersi:

$$M_{35}^{-1} = \begin{bmatrix} 1 & \alpha^{13} & \alpha^{11} & 1 \\ \alpha^{14} & 1 & \alpha^{14} & \alpha^{13} \\ \alpha^{13} & 1 & 1 & 1 \\ \alpha^{14} & \alpha^{11} & \alpha^{12} & 1 \end{bmatrix}$$

M_{35}^{-1} matrisinin ham XOR sayısı **82**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **48** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **45** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **45** olarak, **SBP** algoritmasına göre **6** derinlikte **45** olarak hesaplanmıştır.

M_{35} ve M_{35}^{-1} matrislerinin XOR farkı **6** olarak hesaplanmıştır.

6.4. $\mathbb{F}_{2^4}/0x19$ Cismi Üzerinde Tanımlı Matrisler

Bu alt bölümde, indirgenemez polinom olarak $p(\alpha) = 0x19$ kullanılarak elde edilen ve en düşük XOR sayısına sahip olan beş matris incelenmektedir.

Örnek 6.16. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{36} = \begin{bmatrix} \alpha^{14} & \alpha^3 & \alpha^{14} & \alpha \\ \alpha & \alpha^{14} & \alpha^{14} & \alpha^{14} \\ 1 & \alpha^2 & \alpha^{14} & \alpha^3 \\ 1 & 1 & \alpha & \alpha^{14} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

Çizelge 6.4. $\mathbb{F}_{2^4}/0x13$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
39	8
40	45
41	266
42	452
43	1123
44	1587
45	2402
46	2440
47	2624
48	2055
49	1471
50	944
51	411
52	150
53	31
54	22
55	2

$$M_{36} = \begin{bmatrix} \alpha^{13} & \alpha^2 & \alpha^{13} & 1 \\ 1 & \alpha^{13} & \alpha^{13} & \alpha^{13} \\ \alpha^{14} & \alpha & \alpha^{13} & \alpha^2 \\ \alpha^{14} & \alpha^{14} & 1 & \alpha^{13} \end{bmatrix}$$

M_{36} yarı-involutif MDS matrisinin ham XOR sayısı **72**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **49** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **39** olarak, **SBP** algoritmasına göre **6** derinlikte **39** olarak hesaplanmıştır.

M_{36} matrisinin tersi:

$$M_{36}^{-1} = \begin{bmatrix} 1 & \alpha^4 & 1 & \alpha^2 \\ \alpha^2 & 1 & 1 & 1 \\ \alpha^1 & \alpha^3 & 1 & \alpha^4 \\ \alpha^1 & \alpha^1 & \alpha^2 & 1 \end{bmatrix}$$

M_{36}^{-1} matrisinin ham XOR sayısı **82**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **47** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **45** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **45** olarak, **SBP** algoritmasına

göre **6** derinlikte **45** olarak hesaplanmıştır.

M_{36} ve M_{36}^{-1} matrislerinin XOR farkı **6** olarak hesaplanmıştır.

Örnek 6.17. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{37} = \begin{bmatrix} \alpha^{14} & \alpha^{14} & \alpha^2 & \alpha \\ 1 & \alpha^{14} & \alpha & \alpha^3 \\ \alpha^2 & 1 & \alpha^{14} & 1 \\ 1 & \alpha & \alpha^{14} & \alpha^{14} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{13} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{37} = \begin{bmatrix} \alpha^{12} & \alpha^{12} & 1 & \alpha^{14} \\ \alpha^{13} & \alpha^{12} & \alpha^{14} & \alpha \\ 1 & \alpha^{13} & \alpha^{12} & \alpha^{13} \\ \alpha^{13} & \alpha^{14} & \alpha^{12} & \alpha^{12} \end{bmatrix}$$

M_{37} yarı-involutif MDS matrisinin ham XOR sayısı **78**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **50** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **39** olarak, **SBP** algoritmasına göre **8** derinlikte **39** olarak hesaplanmıştır.

M_{37} matrisinin tersi:

$$M_{37}^{-1} = \begin{bmatrix} \alpha^1 & \alpha^1 & \alpha^4 & \alpha^3 \\ \alpha^2 & \alpha^1 & \alpha^3 & \alpha^5 \\ \alpha^4 & \alpha^2 & \alpha^1 & \alpha^2 \\ \alpha^2 & \alpha^3 & \alpha^1 & \alpha^1 \end{bmatrix}$$

M_{37}^{-1} matrisinin ham XOR sayısı **109**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **54** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **43** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **43** olarak, **SBP** algoritmasına göre **8** derinlikte **43** olarak hesaplanmıştır.

M_{37} ve M_{37}^{-1} matrislerinin XOR farkı **4** olarak hesaplanmıştır.

Örnek 6.18. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{38} = \begin{bmatrix} \alpha^{14} & \alpha & \alpha^3 & \alpha^{14} \\ 1 & \alpha^{14} & 1 & \alpha \\ \alpha & \alpha^{14} & \alpha^{14} & \alpha^{14} \\ 1 & \alpha^3 & \alpha^2 & \alpha^{14} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{38} = \begin{bmatrix} \alpha^{13} & 1 & \alpha^2 & \alpha^{13} \\ \alpha^{14} & \alpha^{13} & \alpha^{14} & 1 \\ 1 & \alpha^{13} & \alpha^{13} & \alpha^{13} \\ \alpha^{14} & \alpha^2 & \alpha & \alpha^{13} \end{bmatrix}$$

M_{38} yarı-involutif MDS matrisinin ham XOR sayısı **72**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **48** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **39** olarak, **SBP** algoritmasına göre **6** derinlikte **39** olarak hesaplanmıştır.

M_{38} matrisinin tersi:

$$M_{38}^{-1} = \begin{bmatrix} 1 & \alpha^2 & \alpha^4 & 1 \\ \alpha^1 & 1 & \alpha^1 & \alpha^2 \\ \alpha^2 & 1 & 1 & 1 \\ \alpha^1 & \alpha^4 & \alpha^3 & 1 \end{bmatrix}$$

M_{38}^{-1} matrisinin ham XOR sayısı **82**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **48** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **45** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **45** olarak, **SBP** algoritmasına göre **6** derinlikte **45** olarak hesaplanmıştır.

M_{38} ve M_{38}^{-1} matrislerinin XOR farkı **6** olarak hesaplanmıştır.

Örnek 6.19. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{39} = \begin{bmatrix} \alpha^{14} & \alpha^3 & \alpha & \alpha^{14} \\ \alpha & \alpha^{14} & \alpha^{14} & \alpha^{14} \\ 1 & 1 & \alpha^{14} & \alpha \\ 1 & \alpha^2 & \alpha^3 & \alpha^{14} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{14} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{39} = \begin{bmatrix} \alpha^{13} & \alpha^2 & 1 & \alpha^{13} \\ 1 & \alpha^{13} & \alpha^{13} & \alpha^{13} \\ \alpha^{14} & \alpha^{14} & \alpha^{13} & 1 \\ \alpha^{14} & \alpha & \alpha^2 & \alpha^{13} \end{bmatrix}$$

M_{39} yarı-involutif MDS matrisinin ham XOR sayısı **72**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **49** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **39** olarak, **SBP** algoritmasına göre **6** derinlikte **39** olarak hesaplanmıştır.

M_{39} matrisinin tersi:

$$M_{39}^{-1} = \begin{bmatrix} 1 & \alpha^4 & \alpha^2 & 1 \\ \alpha^2 & 1 & 1 & 1 \\ \alpha^1 & \alpha^1 & 1 & \alpha^2 \\ \alpha^1 & \alpha^3 & \alpha^4 & 1 \end{bmatrix}$$

M_{39}^{-1} matrisinin ham XOR sayısı **82**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **47** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **45** olarak, **BP** Derinlik Öncelikli algoritmasına göre **6** derinlikte **45** olarak, **SBP** algoritmasına göre **6** derinlikte **45** olarak hesaplanmıştır.

M_{39} ve M_{39}^{-1} matrislerinin XOR farkı **6** olarak hesaplanmıştır.

Örnek 6.20. $\mathbb{F}_{2^4}$ cismi üzerinde tanımlı $p(\alpha) = 0x19$ polinomunun bir kökü olan α kullanılarak elde edilen involutif MDS matris:

$$A_{40} = \begin{bmatrix} \alpha^{14} & \alpha^{14} & \alpha & \alpha^2 \\ 1 & \alpha^{14} & \alpha^3 & \alpha \\ 1 & \alpha & \alpha^{14} & \alpha^{14} \\ \alpha^2 & 1 & 1 & \alpha^{14} \end{bmatrix}$$

Aynı polinomun elemanı olan α^{13} ile çarpılarak elde edilen yarı-involutif MDS matris:

$$M_{40} = \begin{bmatrix} \alpha^{12} & \alpha^{12} & \alpha^{14} & 1 \\ \alpha^{13} & \alpha^{12} & \alpha & \alpha^{14} \\ \alpha^{13} & \alpha^{14} & \alpha^{12} & \alpha^{12} \\ 1 & \alpha^{13} & \alpha^{13} & \alpha^{12} \end{bmatrix}$$

M_{40} yarı-involatif MDS matrisinin ham XOR sayısı **78**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **50** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **39** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **39** olarak, **SBP** algoritmasına göre **8** derinlikte **39** olarak hesaplanmıştır.

M_{40} matrisinin tersi:

$$M_{40}^{-1} = \begin{bmatrix} \alpha^1 & \alpha^1 & \alpha^3 & \alpha^4 \\ \alpha^2 & \alpha^1 & \alpha^5 & \alpha^3 \\ \alpha^2 & \alpha^3 & \alpha^1 & \alpha^1 \\ \alpha^4 & \alpha^2 & \alpha^2 & \alpha^1 \end{bmatrix}$$

M_{40}^{-1} matrisinin ham XOR sayısı **109**, optimizasyon yöntemleri ile elde edilen XOR sayısı **Paar** algoritmasına göre **54** olarak, **BP** Derinlik Önceliksiz algoritmasına göre **45** olarak, **BP** Derinlik Öncelikli algoritmasına göre **8** derinlikte **45** olarak, **SBP** algoritmasına göre **8** derinlikte **45** olarak hesaplanmıştır.

M_{40} ve M_{40}^{-1} matrislerinin XOR farkı **6** olarak hesaplanmıştır.

Çizelge 6.5. $\mathbb{F}_{2^4}/0x19$ Cisminde Üretilen 4×4 Yarı İnvolutif MDS Matrislerinin En Düşük XOR Değerlerine Göre Dağılımı

En Düşük XOR	Toplam Matris Sayısı
39	6
40	63
41	220
42	503
43	1077
44	1599
45	2459
46	2447
47	2591
48	2158
49	1381
50	893
51	398
52	158
53	58
54	19
55	4

BÖLÜM 7

SONUÇ ve TARTIŞMA

Bu tez çalışmasında, $m \in \{3, 4\}$ olmak üzere $\mathbb{F}_{2^m}$ sonlu cisimleri üzerinde tanımlı maksimum uzaklıkta ayrılabilen (MDS) yarı involutif matrislerin üretimine yönelik cebirsel bir yöntem sunulmuştur. Literatürde genellikle involutif MDS matrislerin tasarım teknikleri araştırılırken, bu araştırma, yarı involutif kavramını kapsamlı bir üretimle entegre ederek önemli bir metodolojik eksikliği gidermiştir. Ayrıca, bu matrislerin donanımda daha etkin bir şekilde uygulanabilmesi için, Paar, SLP, Boyar-Peralta ve Superior Boyar-Peralta gibi çeşitli XOR optimizasyon algoritmalarıyla karşılaştırmalı analizleri gerçekleştirilmiştir.

Çalışmada önerilen üretim yöntemi, mevcut involutif MDS matrislerinin Magma hesaplamalı cebir sistemi kullanılarak parametrik olarak türetilmesi ve bu matrislerin belirli diyagonal dönüşümler yoluyla yarı involutif forma dönüştürülmesi şeklinde iki temel aşamadan oluşmaktadır. Bu yaklaşım, tüm olası parametre kombinasyonlarını sistematik olarak tarayan algoritmik yapılarla gerçekleştirilmiş ve sonuçlar Magma ortamında doğrulanmıştır.

Donanımsal uygulamalar açısından XOR sayısının minimize edilmesi de çalışmanın hedeflerinden biri olarak belirlenmiştir. Bunun için kullanılan dört optimizasyon algoritması, doğrusal devre sentezi ve devre derinliği bağlamında pratik fayda sağlamaktadır. Özellikle SBP algoritmasının derinlik kısıtlaması altında en düşük XOR sayılarını elde etmede öne çıktığı gözlemlenmiştir.

$\mathbb{F}_{2^3}$ ile $\mathbb{F}_{2^4}$ sonlu cisimleri altında tanımlı $0xB$, $0xD$, $0x13$ ve $0x19$ indirgenemez polinomları kullanılarak 3×3 ile 4×4 boyutlarında sırasıyla 1.072.512 ve 6.790.589.568 yarı involutif MDS matris üretilmiştir. Özellikle 4×4 boyutundaki matrisler için, XOR sayısı 39 gibi oldukça düşük değerlere inebildiği gözlemlenmiştir ki bu, donanımsal

etkinlik açısından dikkate değer bir başarıdır. Örneğin, $\mathbb{F}_{2^4}/0x19$ cismi üzerinde elde edilen matrislerin XOR dağılımını incelediğimizde, 39 ile 55 aralığında bir yoğunluk gözlemlenmiştir ve optimizasyon sonrasında XOR sayısının ortalamasında önemli bir düşüş kaydedilmiştir. Bu sonuçların, literatürde (Li vd., 2019), (Pehlivanoglu vd., 2018) ve (Tuncay, 2021) gibi çalışmalarda bildirilen değerlerle karşılaştırıldığında, rekabetçi bir performans sergilediği söylenebilmektedir. Bununla birlikte, yöntemin teorik yapısı incelendiğinde, daha geniş bit uzunluklarına sahip sonlu cisimlere—özellikle AES standardında kullanılan $\mathbb{F}_{2^8}$ gibi—genellenebilir nitelikte olduğu görülmektedir. Bu bağlamda, gelecekte yapılacak çalışmalarda, önerilen metodolojinin $\mathbb{F}_{2^8}$ cisminde ve bu cisim üzerinde tanımlı indirgenemez polinomlara uygulanması, çok daha fazla sayıda yarı involutif MDS matrislerin üretilmesini mümkün kılacaktır. Böylece, yalnızca teorik matris çeşitliliği artırılmakla kalmayacak, aynı zamanda gerçek dünyada kullanılan blok şifreleme algoritmalarına entegre edilebilecek yüksek yayılım kapasiteli, ters alma maliyeti düşük ve donanımına uygun matrislerin tasarımı da gerçekleştirilebilecektir.

Gelecekteki çalışmalar kapsamında, elde edilen yarı involutif MDS matrislerin kriptanalitik güvenlik düzeylerinin kapsamlı bir biçimde analiz edilmesi önemli bir gereklilik olarak öne çıkmaktadır. Ayrıca, geliştirilen matrislerin donanım düzeyindeki uygulanabilirliğinin test edilmesi amacıyla FPGA ve ASIC platformlarında sentetik devre tasarımı gerçekleştirilerek, gecikme, alan kullanımı, güç tüketimi ve frekans performansı gibi ölçütler üzerinden detaylı karşılaştırmalar yapılabilir. Bu tür uygulamalı çalışmalar, matrislerin yalnızca kuramsal düzeyde değil, gerçek zamanlı sistemlerde ne denli verimli ve güvenli olduklarını da gösterecektir. Öte yandan, matris üretim sürecine dinamik parametre seçimi, evrimsel algoritmalar veya öğrenmeye dayalı sezgisel teknikler entegre edilerek arama uzayı daha etkili biçimde taranabilir ve optimizasyon başarımı artırılabilir. Önerilen yarı involutif MDS matrislerin, AES gibi modern blok şifreleme algoritmalarının doğrusal dönüşüm katmanına entegre edilerek, hem simülasyon hem de donanım tabanlı güvenlik ve performans analizlerinin gerçekleştirilmesinin, yöntemin kapsamlı etkinliğinin değerlendirmesi açısından önemli bir adım olduğu düşünülmektedir.

Sonuç olarak bu çalışmada kullanılan tasarım teknikleri, doğrusal dönüşümlerin hem yapısal güvenlik gerekliliklerini hem de donanımsal verimlilik koşullarını eşzamanlı olarak karşılayabilecek şekilde tasarlanmıştır. Önerilen yöntem aracılığıyla yarı involutif MDS matrislerin belirli sonlu cisimler üzerinde sistematik biçimde üretilmesi ve bu yapıların XOR işlem sayısı açısından optimize edilerek donanımına uygun hale getirilebilmesi, özellikle gömülü sistemler ve kaynak kısıtlı platformlar için potansiyel uygulamalara zemin hazırlamaktadır. Elde edilen bulgular, mevcut literatürde yer alan yöntemlerle karşılaştırıldığında, hem matris üretiminde kapsayıcılık hem de donanımsal ölçütler açısından anlamlı farklar ortaya koymaktadır. Bu yönüyle tez, yarı involutif

matrislerin kriptografik sistemlerde kullanımına yönelik hem kuramsal bir çerçeve sunmakta hem de uygulamaya dönük değerlendirmelere katkıda bulunmaktadır.



KAYNAKLAR

- Aumasson, J.-P., Neves, S., Wilcox-O’Hearn, Z., & Winnerlein, C. (2015). *The BLAKE2 Cryptographic Hash and Message Authentication Code (MAC)* (Tek. Rap. No. RFC 7693). IETF. <https://www.rfc-editor.org/rfc/rfc7693.html>
- Barreto, P. S. (2000). The Anubis block cipher. *NESSIE*.
- Beierle, C., Kranz, T., & Leander, G. (2016). Lightweight multiplication in $GF(2^n)$ with applications to MDS matrices. *Annual International Cryptology Conference*, 625–653.
- Biham, E. (1994). New Types of Cryptanalytic Attacks Using Related Keys. İçinde T. Helleseeth (Ed.), *Advances in Cryptology — EUROCRYPT ’93* (ss. 398–409, C. 765). Springer. https://doi.org/10.1007/3-540-48285-7_34
- Bogdanov, A., Knudsen, L. R., Leander, G., Paar, C., Poschmann, A., Robshaw, M. J., Seurin, Y., & Vikkelse, C. (2007). PRESENT: An Ultra-Lightweight Block Cipher. *CHES 2007*.
- Boyar, J., & Peralta, R. (2010). A new combinational logic minimization technique with applications to cryptology. *Experimental Algorithms: 9th International Symposium, SEA 2010, Ischia Island, Naples, Italy, May 20-22, 2010. Proceedings* 9, 178–189.
- Boyar, J., & Peralta, R. (2000). MCM for linear transformations in cryptographic algorithms [<https://eprint.iacr.org/2000/032>]. *IACR Cryptology ePrint Archive*.
- Bray, T. (2014). *The javascript object notation (json) data interchange format* (Tek. Rap.).
- Chatterjee, T., & Laha, A. (2023). A note on semi-orthogonal (G-matrix) and semi-involutory MDS matrices. *Finite Fields and Their Applications*, 92, 102279. <https://doi.org/https://doi.org/10.1016/j.ffa.2023.102279>

- Cheon, G.-S., Curtis, B., & Kim, H. (2021). Semi-involutory matrices and signed self-inverse. *Linear Algebra and its Applications*, 622, 294–315. <https://doi.org/10.1016/j.laa.2021.03.038>
- Daemen, J., & Rijmen, V. (2002). *The Design of Rijndael: AES - The Advanced Encryption Standard*. Springer.
- Davis, P. J. (1979). *Circulant matrices*. Wiley.
- De Canniere, C., & Preneel, B. (2008). Trivium. *New Stream Cipher Designs*.
- Diffie, W., & Hellman, M. E. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, 644–654.
- Donovan, A. A., & Kernighan, B. W. (2015). *The Go programming language*. Addison-Wesley Professional.
- Duval, S., & Leurent, G. (2018). MDS matrices with lightweight circuits. *IACR Transactions on Symmetric Cryptology*, 2018(2), 48–78.
- Eastlake, D., & Jones, P. (2001). *US Secure Hash Algorithm 1 (SHA1)* (Tek. Rap. No. RFC 3174). IETF. <https://www.ietf.org/rfc/rfc3174.txt>
- Feistel, H. (1973). Cryptography and Computer Privacy. *Scientific American*, 228(5), 15–23.
- Ferguson, N., Schneier, B., & Kohno, T. (2010). *Cryptography Engineering: Design Principles and Practical Applications*. Wiley.
- Forouzan, B. A. (2007). *Cryptography & network security*. McGraw-Hill, Inc.
- Güzel, G. G., Sakallı, M. T., Akleyek, S., Rijmen, V., & Çengellenmiş, Y. (2019). A new matrix form to generate all 3×3 involutory MDS matrices over F_{2^m} . *Information Processing Letters*, 147, 61–68.
- Hedayat, A., & Wallis, W. D. (1978). Hadamard matrices and their applications. *The annals of statistics*, 1184–1238.
- Jean, J., Peyrin, T., Sim, S. M., & Tourteaux, J. (2017). Optimizing implementations of lightweight building blocks. *IACR Transactions on Symmetric Cryptology*, 130–168.
- Kam, J., & Davida, G. (1979). Structured design of substitution-permutation encryption networks. *IEEE Transactions on Computers*, 28(10), 747–753.

- Kerckhoffs, A. (1883). La cryptographie militaire. *Journal des sciences militaires*.
- Khoo, K., Peyrin, T., Poschmann, A. Y., & Yap, H. (2014). FOAM: searching for hardware-optimal SPN structures and components with a fair comparison. *International Workshop on Cryptographic Hardware and Embedded Systems*, 433–450.
- Kranz, T., Leander, G., Stoffelen, K., & Wiemer, F. (2017). Shorter linear straight-line programs for MDS matrices. *IACR Transactions on Symmetric Cryptology*, 188–211.
- Kwon, D., Kim, J., Park, S., Sung, S. H., Sohn, Y., Song, J. H., Yeom, Y., Yoon, E.-J., Lee, S., Lee, J., vd. (2003). New block cipher: ARIA. *International conference on information security and cryptology*, 432–445.
- Kwon, D., Sung, S. H., Song, J. H., & Park, S. (2005). Design of block ciphers and coding theory. *Trends in Mathematics*, 8(1), 13–20.
- Li, S., Sun, S., Li, C., Wei, Z., & Hu, L. (2019). Constructing low-latency involutory MDS matrices with lightweight circuits. *IACR Transactions on Symmetric Cryptology*, 84–117.
- Lidl, R., & Niederreiter, H. (1997). *Finite Fields*. Cambridge University Press.
- MacWilliams, F. J., & Sloane, N. J. A. (1977). *The theory of error-correcting codes* (C. 16). Elsevier.
- Magma Computational Algebra System. (2025). 13 Mayıs 2025 tarihinde <https://magma.maths.usyd.edu.au/magma/> adresinden erişildi.
- Mao, W. (2004). *Modern Cryptography: Theory and Practice*. Prentice Hall.
- Menezes, A. J., van Oorschot, P. C., & Vanstone, S. A. (1996). *Handbook of Applied Cryptography*. CRC Press.
- Merkel, D., vd. (2014). Docker: lightweight linux containers for consistent development and deployment. *Linux j*, 239(2), 2.
- Momjian, B. (2001). *PostgreSQL: introduction and concepts* (C. 192). Addison-Wesley New York.
- National Institute of Standards and Technology (NIST). (1999). *Data Encryption Standard* (Tek. Rap. No. FIPS PUB 46-3). NIST. <https://csrc.nist.gov/files/pubs/fips/46-3/final/docs/fips46-3.pdf>

- National Institute of Standards and Technology (NIST). (2001). *Advanced Encryption Standard* (Tek. Rap. No. FIPS PUB 197). NIST. <https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.197-upd1.pdf>
- National Institute of Standards and Technology (NIST). (2015a). *Secure Hash Standard (SHS)* (Tek. Rap. No. FIPS PUB 180-4). NIST. <https://csrc.nist.gov/publications/detail/fips/180/4/final>
- National Institute of Standards and Technology (NIST). (2015b). *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions* (Tek. Rap. No. FIPS PUB 202). NIST. <https://csrc.nist.gov/publications/detail/fips/202/final>
- Paar, C. (1997). Optimized arithmetic for Reed-Solomon encoders. *Proceedings of IEEE International Symposium on Information Theory*, 250.
- Pehlivanoğlu, M. K., & Demir, M. A. (2024). Optimizing implementations of linear layers using two and higher input XOR gates. *PeerJ Computer Science*, 10, e1820.
- Pehlivanoğlu, M. K., Sakalli, M. T., Akleyek, S., Duru, N., & Rijmen, V. (2018). Generalisation of Hadamard matrix to generate involutory MDS matrices for lightweight cryptography. *IET Information Security*, 12(4), 348–355.
- Rivest, R. L. (1992a). RC4 (ARCFOUR) Cipher.
- Rivest, R. L. (1992b). *The MD5 Message-Digest Algorithm* (Tek. Rap. No. RFC 1321). IETF. <https://www.ietf.org/rfc/rfc1321.txt>
- Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2), 120–126.
- Sarkar, S., & Sim, S. M. (2016). A deeper understanding of the XOR count distribution in the context of lightweight cryptography. *International Conference on Cryptology in Africa*, 167–182.
- Schneier, B. (1993). Description of a new variable-length key, 64-bit block cipher (Blowfish). *International workshop on fast software encryption*, 191–204.
- Shannon, C. E. (1949). Communication Theory of Secrecy Systems. *Bell System Technical Journal*, 28(4), 656–715.
- Sim, S. M., Khoo, K., Oggier, F., & Peyrin, T. (2015). Lightweight MDS involution matrices. *International Workshop on Fast Software Encryption*, 471–493.

- Stallings, W. (2013). *Cryptography and Network Security: Principles and Practice* (6th). Prentice Hall Press.
- Standaert, F.-X., & Quisquater, J.-J. (2006). Efficient implementation of block ciphers using threshold circuits. *Integration*, 40(1), 1–16.
- Stinson, D. R. (2005). *Cryptography: theory and practice*. Chapman; Hall/CRC.
- Stinson, D. R., & Paterson, M. B. (2019). *Cryptography: Theory and Practice* (4th). CRC Press.
- Tuncay, G. (2021). *Blok şifrelerde kullanılan ikili ve MDS matrisler için yeni tasarım teknikleri (New design techniques for binary and MDS matrices used in block ciphers)* [dok. tezi, Trakya University, Turkey].
- Tuncay, G., Sakallı, F. B., Pehlivanoglu, M. K., Yılmazgüç, G. G., Akleylek, S., & Sakallı, M. T. (2023). A new hybrid method combining search and direct based construction ideas to generate all 4×4 involutory maximum distance separable (MDS) matrices over binary field extensions. *PeerJ Computer Science*, 9, e1577.
- Webster, A. F., & Tavares, S. E. (1986). On the design of S-boxes. *Advances in Cryptology—CRYPTO'85*, 523–534.
- Wu, H. (2004). The Stream Cipher HC-256.
- Z'aba, M. R. (2010). *Analysis of linear relationships in block ciphers* [dok. tezi, Queensland University of Technology].

ÖZGEÇMİŞ

Edirne Trakya Birlik İlköğretim okulunda ilk ve orta öğrenimimi tamamladıktan sonra Edirne Anadolu Öğretmen Lisesi'nde lise öğrenimimi tamamladım. Ardından Eskişehir Anadolu Üniversitesi Bilgisayar ve Öğretim Teknolojileri Eğitimi bölümünde lisansımı tamamladım. Mezun olduktan sonra 2022 yılında Trakya Üniversitesi Fen Bilimleri Enstitüsü Hesaplamalı Bilimler Anabilim Dalı Yüksek Lisans programında halen öğrenimime devam etmekteyim. Aynı zamanda Trakya Üniversitesi Teknopark'ında bulunan bir şirkette yazılım geliştirici olarak çalışıyorum.