

THESE DE DOCTORAT DE

L'UNIVERSITE DE RENNES 1
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies
de l'Information et de la Communication*
Spécialité : *Mathématiques et leurs Interactions*

Par

Victor Cauchois

Couches de diffusion linéaires à partir de matrices MDS

Thèse présentée et soutenue à Rennes, le 13 décembre 2018
Unité de recherche : IRMAR

Rapporteurs avant soutenance :

Joan DAEMEN Professeur, ICIS, Université de Radboud
Gilles ZEMOR Professeur, IMB, Université de Bordeaux

Composition du Jury :

Examineurs :	Daniel AUGOT	Directeur de Recherche, GRACE, INRIA Saclay
	Delphine BOUCHER	Maître de conférences, IRMAR, Université de Rennes 1
	Joan DAEMEN	Professeur, ICIS, Université de Radboud
	Pierre-Alain FOUQUE	Professeur, IRISA, Université de Rennes 1
	Maria Naya-PLASENCIA	Chargée de Recherche, SECRET, INRIA Paris
	Gilles ZEMOR	Professeur, IMB, Université de Bordeaux
Dir. de thèse :	Pierre LOIDREAU	Chercheur associé, DGA-MI et Université de Rennes 1
Co-dir. de thèse :	Henri GILBERT	Ingénieur, ANSSI.

Remerciements

Mercis

Table des matières

Remerciements	iii
Table des matières	v
Table des figures	ix
Liste des tableaux	xi
Notations	xiii
1 Introduction générale	1
1.1 Cryptologie	1
1.1.1 Cryptographie	2
1.1.2 Cryptanalyse	2
1.1.3 Cryptographie symétrique / asymétrique	3
1.2 État de l'art et contributions	3
1.2.1 Couches de diffusion linéaire légères	4
1.2.2 Attaques rebond sur permutations de type AES	5
1.2.3 Autres travaux	6
2 Préliminaires	9
2.1 Matrices	9
2.2 Polynômes	10
2.3 θ -polynômes	12
I Couches de diffusion linéaire légères à partir de matrices MDS	15
3 Présentation générale	17
3.1 Couches de diffusion linéaire	18
3.1.1 Réseau Substitution-Permutation	19
3.1.2 Attaques statistiques	20
3.1.3 Wide Trail Strategy	21
3.2 Codes et matrices	24
3.2.1 Codes linéaires et nombres de branchements	25
3.2.2 Matrices MDS	28
3.2.3 Codes et matrices MRD	31
3.3 Implémentations matérielles légères	34

3.3.1	Mesure heuristique des coûts d'implémentation matérielle	34
3.3.2	Trois tentations de réduction des coûts	36
3.3.3	Matrices de Hadamard-Cauchy	40
4	Construction de matrices récurives MDS et généralisations	41
4.1	Matrices récurives	42
4.1.1	Matrices récurives et codes cycliques	42
4.1.2	Constructions directes de matrices récurives MDS	45
4.1.3	Recherches exhaustives de matrices récurives MDS	52
4.2	Tordre la récurivité	53
4.2.1	Matrices θ -récurives et codes θ -cycliques	54
4.2.2	Construction directe de matrices θ -récurives MDS	57
4.2.3	Recherche exhaustive	63
4.3	Relâcher les contraintes	64
4.3.1	Matrices MDS involutives à partir de matrices de Vandermonde	64
4.3.2	Explorer la récurivité	67
4.3.3	Couches de diffusion linéaire parfaites récurives	68
5	Construction de matrices circulantes MDS et généralisations	71
5.1	Matrices circulantes	72
5.1.1	Matrices circulantes MDS involutives	72
5.1.2	Construction directe de matrices circulantes MDS involutives sur des corps de caractéristique impaire	76
5.1.3	Construction directe de matrices circulantes MDS sur des corps de caractéristique 2	78
5.2	Tordre la cyclicité	80
5.2.1	Matrices θ -circulantes MDS	80
5.2.2	Recherche exhaustive de matrices θ -circulantes MDS involutives	82
5.3	Relâcher les contraintes	84
5.3.1	Matrices cycliques MDS	84
5.3.2	Construction directe de matrices θ^2 -circulantes MDS quasi- θ -involutives	86
5.3.3	Couches de diffusion linéaire parfaites circulantes	89
	Conclusions et perspectives	91
II	Attaques rebond sur les permutations internes de Grøstl₅₁₂	93
6	Présentation générale	95
6.1	Fonctions de hachage	96
6.1.1	Propriétés cryptographiques	97
6.1.2	Modes d'opération - Preuves de sécurité	98
6.1.3	Grøstl	99
6.2	Algorithmes de différenciations	101
6.2.1	Modèle de l'oracle aléatoire	102
6.2.2	Paradoxe des anniversaires généralisés	102
6.2.3	Problème des anniversaires limité	102
6.3	Chemins différentiels tronqués	103
6.3.1	Principe	104
6.3.2	Attaque rebond	104

7	Attaque sur 10 tours, risques d'avalanche	107
7.1	Un chemin différentiel tronqué naturel	108
7.1.1	Motif différentiel	108
7.1.2	Plausibilité de réalisation	110
7.1.3	Instance du problème des anniversaires limité induite	110
7.2	Algorithme de différenciation	111
7.2.1	Esquisse de l'algorithme de différenciation	111
7.2.2	Étape 3 : Fusion de listes	113
8	Attaque sur 11 tours, contrôler les éléments	117
8.1	Un chemin différentiel tronqué structuré	119
8.1.1	Recherche de chemins différentiels tronqués creux	119
8.1.2	Motif différentiel	121
8.1.3	Plausibilité de réalisation	121
8.1.4	Instance du paradoxe des anniversaires généralisé induite	123
8.1.5	Description macroscopique compacte	125
8.2	Algorithme de différenciation	126
8.2.1	Esquisse de l'algorithme de différenciation	126
8.2.2	Étape 1 : Construction des bases des sous-espaces linéaires Δ_i	128
8.2.3	Étape 2 : Construction de couples de valeurs de colonnes avec différences amont/aval d'une Super-Boite S maîtrisées	129
8.2.4	Étape 3 : Construction de couples de valeurs d'état interne partiel compa- tible avec des transitions $\Delta_1 \rightarrow \delta_2$ et $\delta_2 \rightarrow \Delta_3$	130
8.2.5	Étape 4 : Premier recollement	131
8.2.6	Étape 5 : Deuxième recollement	133
8.2.7	Étape 6 : Troisième recollement	136
	Conclusions et perspectives	139

Table des figures

3.1	Réseau substitution-permutation.	19
3.2	Registre à décalage à rétroaction linéaire.	37
4.1	Registre à décalage à rétroaction linéaire tordu.	61
4.2	Registres à décalage en série	69
6.1	Mode d'opération Merkle-Damgård.	99
6.2	Mode d'opération Wide-Pipe.	99
6.3	Fonctions internes de Grøstl ₅₁₂	100
6.4	Transformations ShiftBytesWide	101
6.5	Principe de l'attaque rebond.	104
7.1	Chemin différentiel tronqué sur 10 tours de P_{1024} de Grøstl ₅₁₂	109
7.2	Espace linéaire E_1	110
7.3	Espace linéaire E_2	111
7.4	Partition selon les indices I_1, \dots, I_{16}	112
7.5	Contraintes imposées par un choix d'élément dans $R'_2 \times R'_3 \times R'_4 \times R'_5$	114
7.6	Contraintes imposées par un choix d'élément dans $R'_2 \times R'_3 \times R'_4 \times R'_5$ puis par un choix compatible dans R'_6	114
7.7	Contraintes imposées par un choix d'élément dans $R'_2 \times R'_3 \times R'_4 \times R'_5$ puis par un choix compatible dans R'_6 et enfin par un choix compatible dans L'_1	115
8.1	Chemin différentiel tronqué sur 11 tours de P_{1024} de Grøstl ₅₁₂	122
8.2	Espace linéaire E_1	123
8.3	Espace linéaire E_2	123
8.4	Chemin différentiel tronqué sur 11 tours de Q_{1024} de Grøstl ₅₁₂	124
8.5	Chemin différentiel tronqué sur 11 tours de P_{1024} en Super-Boîtes S.	125
8.6	Chemin différentiel tronqué sur 11 tours de Q_{1024} en Super-Boîtes S.	126
8.7	Chemin différentiel vérifié par les éléments de l'ensemble Δ_1	128
8.8	Transition à travers la transformation MixBytes	128
8.9	Transition différentielle tronquée à travers une opération Super-Boîte S.	129
8.10	Recollement, première édition.	131
8.11	Octets fixés et listes d'éléments.	133
8.12	Recollement, deuxième édition.	135
8.13	Fusion des listes, première édition.	135
8.14	Recollement, troisième édition.	136
8.15	Fusion de listes, deuxième édition.	137

Liste des tableaux

4.1	Dénombrement des matrices récursives MDS sur \mathbb{F}_{2^4}	52
4.2	Polynômes générateurs de matrices récursives MDS records sur \mathbb{F}_{2^4}	53
4.3	Dénombrement des matrices θ -récursives MDS sur \mathbb{F}_{2^4}	63
4.4	Polynômes générateurs de matrices θ -récursives MDS records sur \mathbb{F}_{2^4}	64
4.5	Dénombrement des matrices t -récursives MDS sur \mathbb{F}_{2^4}	67
4.6	Dénombrement des matrices t -involutives MDS sur \mathbb{F}_{2^4}	68
5.1	Dénombrement des matrices circulantes MDS sur \mathbb{F}_{2^4}	79
5.2	Polynômes générateurs de matrices circulantes MDS sur \mathbb{F}_{2^4}	80
5.3	Dénombrement des matrices θ -circulantes MDS sur \mathbb{F}_{2^4}	83
5.4	Polynômes générateurs de matrices θ -circulantes MDS records \mathbb{F}_{2^4}	83
5.5	Dénombrement des matrices cycliques MDS sur \mathbb{F}_{2^4}	86
5.6	Matrices cycliques MDS records sur \mathbb{F}_{2^4}	86
6.1	Attaques rebond sur les permutations internes de Grøstl ₅₁₂	96
6.2	Attaques génériques sur les fonctions de hachage cryptographiques	98

Notations

Soit $\delta_{i,j}$ le symbole de Kronecker, soit $\delta_{i,j} = 1$ si $i = j$ et 0 sinon.

Soit ϕ la fonction de \mathbb{N} dans \mathbb{N} qui à tout $n \in \mathbb{N}$ associe le nombre d'éléments de $\{1, \dots, n-1\}$ premiers avec n . On appelle ϕ la fonction d'Euler.

Chapitre 1

Introduction générale

Sommaire

1.1	Cryptologie	1
1.1.1	Cryptographie	2
1.1.2	Cryptanalyse	2
1.1.3	Cryptographie symétrique / asymétrique	3
1.2	État de l'art et contributions	3
1.2.1	Couches de diffusion linéaire légères	4
1.2.2	Attaques rebond sur permutations de type AES	5
1.2.3	Autres travaux	6

1.1 Cryptologie

La *cryptologie* est, comme son étymologie le suggère, la science des messages secrets [17a]. Elle se décline en deux branches duales : la conception de mécanismes de protection d'échanges de messages sur des canaux de transmission non sécurisés, la *cryptographie*, et l'étude de ces mécanismes en vue de les surpasser ou de s'en affranchir, la *cryptanalyse*. Un mécanisme cryptographique peut poursuivre plusieurs objectifs. La première propriété intuitive, la *confidentialité*, consiste à assurer une communication entre deux personnes sans qu'un tiers ne puisse prendre connaissance de la nature de leurs échanges. Pour ce faire, l'émetteur réalise une transformation d'un message, un *chiffrement*, au moyen d'une *clef de chiffrement*, pour en donner une version inintelligible, un *chiffre*. Le récepteur utilise alors une *clef de déchiffrement* pour réaliser la transformation inverse, un *déchiffrement*, et retrouver le message original, le *clair*. Un tiers qui intercepte les communications et parvient à retrouver le clair à partir du chiffré sans connaissance préalable de la *clef décrypte* le message. La menace d'un tiers malveillant sur les communications ne se limite pas aux tentatives de décryptage. Pour constater la corruption éventuelle d'un message, un émetteur va vouloir mettre en place des mécanismes cryptographiques qui permettent au récepteur de vérifier *l'intégrité* d'un message. Enfin, une troisième propriété élémentaire de protection des communications consiste à assurer l'identité des protagonistes. L'*authenticité* d'un échange assure le destinataire de l'identité de son interlocuteur.

La cryptologie s'est développée sous l'impulsion des besoins militaires pour lesquels la confidentialité est essentielle. Le chiffre de César témoigne de la longue histoire de ces pratiques et constitue une version naïve d'un chiffrement par la substitution des lettres d'un message par les lettres issues d'un décalage de positions dans l'alphabet considéré comme ensemble ordonné. 26 possibilités de chiffrements sont donc possibles. Une *recherche exhaustive*, le test des 26 décalages inverses possibles permet de retrouver le message assez rapidement. Une version améliorée de cet algorithme de chiffrement consiste à ne plus considérer seulement les substitutions induites par des décalages fixes de lettres mais n'importe quelle permutation des lettres. Cette version plus robuste, avec 26! possibilités qui la prémunissent d'attaques par recherche exhaustive, est cependant fragile car dans la plupart des langues, les lettres n'ont pas la même probabilité d'occurrence. Un échantillonnage de la fréquence d'apparition permet alors de décrypter un message dès que celui-ci est assez long. La machine Enigma, système de chiffrement allemand pendant la seconde guerre mondiale, était une version plus sophistiquée avec l'utilisation de plusieurs chiffrements de substitution indépendants successifs. De nouvelles techniques statistiques imaginées par Alan Turing, combinées à la connaissance de la structure de certains messages, ont permis le décryptage de ces machines, ce qui a constitué un avantage stratégique militaire déterminant. La recherche scientifique s'est alors emparée de ce sujet et l'essor de l'informatique a permis le perfectionnement des techniques utilisées tandis que la multiplication des besoins de mécanismes cryptographiques a soutenu cet effort dans le monde industriel. En effet, outre les applications médicales, pour lesquelles la confidentialité est aujourd'hui une obligation légale, et les applications bancaires, pour lesquelles l'authenticité est une nécessité, l'émergence et la prolifération des utilisations de données personnelles, sur les réseaux sociaux, sur les moteurs de recherches, les publicités ciblées, les maisons connectées, la vie numérique dans son ensemble ont fait des problématiques de protection de la vie privée un enjeu majeur.

1.1.1 Cryptographie

Les fondations de la cryptographie moderne ont été posées par Auguste Kerckhoffs [Ker83]. Le principe premier affirme qu'un mécanisme cryptographique se conçoit comme un algorithme public, connu, utilisé avec une clef privée, secrète. Ce principe permet de concevoir des méthodes de chiffrement communes à plusieurs utilisateurs, avec seulement des jeux de clefs différents. En cryptographie, on considère donc les algorithmes connus, les rendre secrets relevant de stratégies de défense en profondeur. À cet effet, des processus de standardisation ont fait émerger des algorithmes qui ont vocation à être propagés dans l'essentiel des applications cryptographiques pour répondre aux différents besoins. Le Data Encryption Standard (DES) puis l'Advanced Encryption Standard (AES) sont les deux standards successifs d'*algorithmes de chiffrement par bloc* qui traitent des entrées de taille fixe. La compétition eStream vise quant à elle à définir l'équivalent pour les *algorithmes de chiffrement à flot* qui produisent des suites de longueur variables à partir d'une clef et d'une valeur d'initialisation en vue de réaliser avec un message de longueur arbitraire une opération de OU EXCLUSIF (XOR) bit à bit. Des mécanismes de *chiffrement authentifié* qui assurent à la fois la confidentialité et l'authenticité des transmissions font l'objet de la compétition CAESAR. Enfin, pour les *fonctions de hachage*, qui produisent des sorties de taille fixe à partir d'entrées de tailles quelconques, les standards SHA-1, SHA-2 et SHA-3 ont été successivement définis.

1.1.2 Cryptanalyse

La sécurité d'un mécanisme cryptographique n'est pas une propriété absolue. Elle consiste en la résistance offerte face aux attaques connues et est donc toute relative. Un mécanisme est

considéré sûr tant qu'il n'existe pas de faille avérée, ces fragilités reposent la plupart du temps sur des défauts inhérents à l'architecture de l'algorithme de chiffrement. Un exemple d'une telle faille est donné par le DES *cassé* par des méthodes de cryptanalyse à partir de vulnérabilités statistiques : il existe des algorithmes pratiques de recouvrement de la clef de chiffrement ou d'un clair à partir de son chiffré. Pour la communauté des cryptologues, il n'est cependant pas nécessaire qu'il existe des attaques pratiques pour considérer qu'un schéma de chiffrement soit cassé, il suffit qu'il existe une procédure qui permette de mettre en cause une propriété élémentaire avec une méthode plus efficace que le test exhaustif sur toutes les clefs. Il est par exemple classique de supposer que l'ensemble des couples clairs/chiffrés associés est disponible. D'autres menaces pèsent sur la sécurité des mécanismes cryptographiques, comme l'ordinateur quantique, épouvantail de nombreux algorithmes de chiffrement qui reposent sur des problèmes de logarithmes discrets, de complexité exponentielle pour l'algorithmique classique mais de complexité polynomiale pour l'algorithmique quantique.

1.1.3 Cryptographie symétrique / asymétrique

Les exemples de mécanismes cryptographiques déjà mentionnés sont tous des algorithmes de chiffrement, une transformation d'un clair en chiffré, et présupposent tous la connaissance préalable d'un secret partagé. Les rôles de l'émetteur et du récepteur sont symétriques d'où la dénomination de *cryptographie symétrique*. Whitfield Diffie et Martin Hellman ont introduit l'idée qu'il est possible d'envisager deux rôles différents du chiffrement et du déchiffrement [DH76]. Un émetteur peut alors chiffrer un message à destination d'un récepteur dont il connaît la clef de chiffrement, la *clef publique*. La connaissance par le récepteur de la clef de déchiffrement associée, la *clef secrète*, permet seule le déchiffrement du message. On parle alors de *cryptographie à clef publique*, ou par opposition aux mécanismes dont les clefs sont partagées, de *cryptographie asymétrique*. En pratique, les mécanismes de cryptographie symétrique sont beaucoup plus rapides que les mécanismes de cryptographie asymétrique mais nécessitent la connaissance d'un secret partagé. Il est donc d'usage d'utiliser des *protocoles* instanciés avec des mécanismes de chiffrement asymétrique pour convenir d'une clef partagée symétrique avec laquelle on réalise la majorité des échanges.

1.2 État de l'art et contributions

La présentation des travaux rassemblés dans cette thèse partage avec la cryptologie sa structure duale. Ainsi, la première partie de cette thèse est consacrée à l'exposition de résultats sur les constructions de matrices récursives, circulantes, θ -récursives, θ -circulantes et quelques unes de leurs généralisations. Ces matrices sont dites Maximum Distance Separable lorsqu'elles présentent des propriétés de diffusion optimales sur les symboles, souhaitables pour des raisons de résistance aux attaques statistiques pour les primitives symétriques. Ces structures sont explorées parce qu'elles admettent des implémentations matérielles légères. La seconde partie recense quant à elle des résultats sur les attaques rebond, attaques visant à différencier des permutations dont l'architecture est inspirée de l'AES de permutations aléatoires. Cette architecture a été très en vogue dans les conceptions de primitives symétriques. L'utilisation de techniques de résolution de systèmes linéaires sous contraintes permet en effet de faire émerger des transitions probabilistes de valeurs de différences qui peuvent être exploitées pour différencier une permutation avec cette architecture d'un tirage aléatoire uniforme dans l'ensemble des permutations.

1.2.1 Couches de diffusion linéaire légères

Les matrices MDS appartiennent à l'intersection de la théorie des codes et de la cryptographie. Ces objets très réguliers ont été largement étudiés et de nombreux résultats sont connus.

État de l'art

Les constructions directes de matrices MDS peuvent se regrouper en deux catégories. La première catégorie contient toutes les méthodes qui se basent uniquement sur des considérations matricielles. Roth et Seroussi ont proposé d'utiliser les matrices de Cauchy, dont un sous-ensemble engendre des matrices MDS [RS85]. Les configurations de Hankel peuvent être également utilisées pour construire des matrices de Cauchy, comme l'a proposé Aydinian [Aid86]. Peyrin, Sim, Khoo et Oggier ont proposé de rajouter à ces matrices la structure de matrice de Hadamard [Sim+15]. Plusieurs autres constructions directes de matrices MDS sont connues et reposent sur des résultats de théorie des codes. Il est possible de raffiner ces constructions directes pour que les matrices MDS obtenues soient récursives, structure qui admet une implémentation matérielle légère à base de registres à rétroaction à décalage linéaire. Daniel Augot et Matthieu Finiasz ont proposé une construction directe de matrices récursives MDS à partir de codes BCH raccourcis [AF14]. La méthode de Thierry Berger engendre également des matrices récursives MDS mais cette fois à partir de codes de Gabidulin [Ber13].

La recherche académique s'est focalisée ces dernières années sur la minimisation des coûts d'implémentation matérielle pour des petits paramètres, pour lesquels des méthodes de recherche exhaustive sont possibles. L'intérêt pour les constructions directes, qui augmente avec les tailles de paramètres, a décliné. Meicheng Liu et Siang Meng Sim [LS16] et Yongqiang Li et Mingsheng Wang [LW16] ont proposé de déstructurer les matrices circulantes pour obtenir des matrices très légères à FSE 2016. Ces travaux ont été poursuivis notamment par Lijing Zhou, Licheng Wang et Yiru Sun [ZWS17]. Les dernières tendances visent les réductions des coûts d'implémentation de matrices sans structure particulière comme le proposent Jérémy Jean, Thomas Peyrin, Siang Meng Sim et Jade Tourteaux [Jea+17] ou encore Thorsten Kranz, Gregor Leander, Ko Stoffelen et Friedrich Wiemer [Kra+17].

Contributions

Les travaux présentés dans cette thèse s'inspirent des premières méthodes de construction de matrices MDS, heuristiques et non exhaustives, basées sur des résultats de théorie des codes.

On propose une nouvelle structure matricielle pour un usage en cryptographie symétrique : les matrices θ -récursives. Ces matrices s'expriment comme un produit de matrices compagnons dont les polynômes sont *tordus* par l'application d'un automorphisme. Cette structure est réalisée par une architecture matérielle presque identique à un LFSR, et est donc susceptible d'être très légère. On est alors en mesure de proposer une construction directe de matrices θ -récursives MDS θ -involutives. La θ -involutivité est également intéressante du point de vue de réduction des coûts matériels car elle signifie que l'inverse de la matrice peut être calculée avec la même architecture matérielle agrémentée d'applications de l'automorphisme. Il est heureux de parvenir à une telle construction, les matrices récursives MDS ne pouvant être involutives en caractéristique 2. Ces résultats, présentés au Chapitre 4, ont fait l'objet de la publication suivante :

- [CLM16] Victor CAUCHOIS, Pierre LOIDREAU et Nabil MERKICHE. "Direct construction of quasi-involutory recursive-like MDS matrices from 2-cyclic codes". In : *IACR Trans. Symmetric Cryptol.* 2016.2 (2016), p. 80-98.

On propose une deuxième structure matricielle pour un usage en cryptographie symétrique : les matrices θ -circulantes. Cette structure admet une architecture matérielle légère : une seule ligne est nécessaire au calcul de toute la matrice, toutes les lignes se déduisent de la précédente par une rotation des coefficients et une application d'un automorphisme aux coefficients. Les contraintes algébriques qui pèsent sur la construction de matrices θ -circulantes involutive MDS n'ont pas été levées. Cependant, on fournit une construction directe de matrices θ^2 -circulantes MDS quasi- θ -involutive. Il est heureux de parvenir à une telle construction, les matrices circulantes MDS ne pouvant être involutives en caractéristique 2 pour des tailles supérieures ou égales à 3. Ces résultats, présentés au Chapitre 5, ont fait l'objet de la publication suivante :

[CL17] Victor CAUCHOIS et Pierre LOIDREAU. “About Circulant Involutory MDS Matrices”. In : *Workshop on Coding and Cryptography* (2017).

L'inexistence de matrices circulantes MDS involutives en caractéristique 2 repose sur des propriétés algébriques valables pour tous les paramètres. L'étude de ces propriétés permet de discriminer les paramètres pour lesquels il en existe. On propose alors une construction directe de matrices circulantes MDS involutives pour des corps de caractéristique impaire. Ces matrices n'ont aucune portée pratique mais sont le premier pas vers une construction directe de matrices circulantes MDS en caractéristique 2, objets beaucoup plus recherchés. Ces résultats, présentés au Chapitre 5, n'ont pas fait l'objet de publication pour le moment.

1.2.2 Attaques rebond sur permutations de type AES

Les attaques rebond appartiennent à la catégorie des attaques différentielles, cryptanalyses qui exploitent un comportement statistique non cohérent avec un tirage aléatoire uniforme dans l'ensemble des permutations.

État de l'art

La cryptanalyse différentielle a été introduite par Eli Biham et Adi Shamir pour la cryptanalyse du DES [BS91]. Elle consiste à utiliser une déviation statistique sur les différences induites par des couples de valeurs obtenues en sortie d'une permutation à partir de couples de valeurs en entrée qui diffèrent d'une différence donnée. Les chemins différentiels tronqués, introduits par Lars Knudsen permettent une analyse simultanée pour plusieurs valeurs de différences [Knu95]. L'utilisation de primitives dont l'architecture s'inspire de l'AES, notamment pour les fonctions de hachage, publiques, a motivé l'introduction des attaques rebond par Florian Mendel, Martin Schl affer, S oren Thomsen et Christian Rechberger, cryptanalyse dont l'objet est de résoudre une instance d'un problème combinatoire plus rapidement que la méthode générique optimale [Men+09b]. Ceci contredit la non différentiabilité de cette permutation avec un tirage aléatoire uniforme dans l'ensemble des permutations. De nombreuses primitives ont été attaquées par ce biais, parmi lesquelles on en mentionne une sur Gr ostl_{512} , primitive sur laquelle nos méthodes ont été utilisées pour rendre l'attaque plus performante due notamment à J eremy Jean, Maria Naya Plasencia et Thomas Peyrin [JNP14]. L'attaque qui menace le plus grand nombre de tours, 11, est l'oeuvre de Marine Minier et Ga el Thomas et repose sur une faille exploit ee par une attaque int egrale [MT13]. Ces pr eoccupations ont  et e d esert ees ces derni eres ann ees au profit de la cryptanalyse de SHA-3. L'utilisation de la r esolution de syst emes lin eaires sous contraintes a connu cependant un v eritable essor. Si les m ethodes utilis ees dans l'attaque rebond connaissent une survivance dans le monde de la recherche acad emique, c'est gr ace aux mod eles d'attaques  a clefs reli ees, pour lesquelles l'espace des clefs fait partie des degr es de libert e de l'attaquant. Les attaques de l'AES dans ce mod ele continuent d'avoir du succ es.

Contributions

L'utilisation d'algorithmes de résolution de systèmes linéaires sous contraintes comme GUROBI [17b] permet de faire émerger des chemins différentiels tronqués structurés plus probables que les chemins différentiels tronqués issus d'avalanches libres. L'idée maîtresse consiste à remarquer que la plupart des degrés de liberté disponibles pour l'attaquant disparaissent dans l'attaque rebond classique lors du recollement de deux ensembles de valeurs de différence qui se sont propagées sur tout l'état interne. Il est possible d'en consommer nettement moins si le recollement a lieu sur un état interne partiel uniquement. Un chemin différentiel tronqué structuré itératif sur 7 colonnes apparaît lors de la minimisation du nombre de colonnes impliquées dans un motif différentiel tronqué sur plusieurs tours. Il est alors possible d'envisager simultanément trois ensembles de valeurs de différences qu'on peut recoller ensemble au moyen de trois algorithmes. Ceci permet de contrôler les profils de valeurs de différences sur 6 tours et non plus sur 4 tours. Appliquées à Grøstl₅₁₂ dont la diffusion totale par une avalanche libre est réalisée en trois tours et non en deux, il est possible d'améliorer significativement les performances de l'attaques rebond et de différencier 11 tours sur 14 de la fonction de tour de la permutation interne pour une complexité de 2^{72} quand la précédente attaque la plus performante permettait de différencier uniquement 10 tours pour une complexité de 2^{392} . Ces résultats, présentés au Chapitre 8, ont fait l'objet de la publication suivante :

[CGL17] Victor CAUCHOIS, Clément GOMEZ et Reynald LERCIER. "Grøstl Distinguishing Attacke : A New Rebound Attack of an AES-like Permutation". In : *IACR Trans. Symmetric Cryptol.* 2017.3 (2017), p. 1-23.

1.2.3 Autres travaux

Mes travaux de recherche de ces trois dernières années ne sont pas restreints aux résultats présentés dans cette thèse. Parmi les sujets divers brièvement exposés ci-après, seul le premier est l'occasion d'une soumission en cours. Les autres sujets de recherche n'ont pas donné lieu à des résultats significatifs mais conservent néanmoins leur légitimité à être mentionnés ici.

Choix de permutations pour schéma de Feistel de type-II

La conception de couches de diffusion linéaire pour les schémas de chiffrement symétrique n'est pas nécessairement inféodée à l'utilisation de matrices MDS. Une autre architecture classique est donnée par les schémas de Feistel, architecture du DES. La version la plus régulière de ceux-ci, appelée Type-II, est telle que tous les ensembles de deux k -uplets jouent un rôle identique. La détermination de la permutation utilisée dans ces schémas était jusque là basée sur une heuristique de choix et sur des méthodes de recherche exhaustive. Ces travaux ont consisté en l'amélioration des recherches par la considération de classes d'équivalences. Les permutations optimales peuvent être déterminées jusqu'à des tailles de 20 dans le cas général et de 26 dans le cas des permutations dites *pair-impair*. L'heuristique peut être poussée pour des tailles puissances de 2. On retrouve alors l'inénarrable graphe de De Bruijn, apparition des travaux de Suzuki et Minematsu [SM10].

Cryptanalyse linéaire

Des travaux sur la cryptanalyse linéaire ont été réalisés dans le but d'améliorer l'attaque linéaire proposée par Cho sur PRESENT [Cho10]. L'essentiel des résultats a consisté en un affinement du calcul des probabilités de succès de l'attaque et une amélioration de la précision des calculs de hull

linéaire. Ces travaux n'ont pas mené à une publication en raison d'une préemption douloureuse, jusqu'à l'esprit du titre original, réalisée par l'article suivant :

- [BTV18] Andrey BOGDANOV, Elmar TISCHHAUSER et Philip S. VEJRE. "Multivariate Profiling of Hulls for Linear Cryptanalysis". In : *IACR Trans. Symmetric Cryptol.* 2018.1 (2018), p. 101-125.

Chiffrement asymétrique à base de codes de Gabidulin

Ces travaux se sont focalisés sur un schéma de chiffrement asymétrique proposé par Pierre Loidreau [Loi17]. Deux objectifs étaient poursuivis : l'un d'eux consistait à déterminer des méthodes de réduction en rajoutant de la structure sur les codes utilisés dans rajouter de vulnérabilité structurelle. L'autre consistait en le développement des méthodes de cryptanalyse connues sur ces schémas. Ces travaux n'ont pas abouti à des résultats significatifs.

Décodage en liste de codes de Gabidulin

Ces travaux s'éloignent des considérations de cryptographie et s'approchent davantage de problématiques de théorie des codes, et plus précisément de théorie des codes en métrique rang. L'objectif était de déterminer des méthodes de décodage en liste pour les codes de Gabidulin qui s'inspirent des décodages en liste de Guruswami et Sudan. L'enjeu était de construire une évaluation naturelle crédible pour laquelle un décodage en liste découle. Ces travaux n'ont pas abouti à des résultats significatifs, même lorsque les paramètres étaient restreints à des paramètres d'usage comme la caractéristique 2.

Chapitre 2

Préliminaires

Les résultats suivants sont, élémentaires, à la base des travaux présentés dans cette thèse.

2.1 Matrices

Un objet classique d'algèbre linéaire est récurrent dans l'analyse des codes linéaires, notamment des codes cycliques, la matrice de Vandermonde :

Définition 1. Soient $\lambda_1, \dots, \lambda_k$ des éléments de \mathbb{F}_q . On appelle matrice de Vandermonde associée à $\{\lambda_1, \dots, \lambda_k\}$, la matrice \mathbf{V} :

$$\mathbf{V} = \begin{pmatrix} 1 & \lambda_1 & \lambda_1^2 & \cdots & \lambda_1^{k-1} \\ 1 & \lambda_2 & \lambda_2^2 & \cdots & \lambda_2^{k-1} \\ \vdots & & \ddots & & \vdots \\ 1 & \lambda_k & \lambda_k^2 & \cdots & \lambda_k^{k-1} \end{pmatrix}$$

Le déterminant d'une matrice de Vandermonde est un résultat tout aussi classique, dont la démonstration se fait par récurrence :

Proposition 1. Soit \mathbf{V} une matrice de Vandermonde associée à $\{\lambda_1, \dots, \lambda_k\}$. Son déterminant vaut :

$$\det(\mathbf{V}) = \prod_{i < j} (\lambda_j - \lambda_i)$$

Une matrice de Vandermonde est donc inversible si et seulement si son support est constitué d'éléments deux à deux distincts. Les matrices de Vandermonde admettent une généralisation immédiate :

Définition 2. Soient $\lambda_1, \dots, \lambda_k$ des éléments de \mathbb{F}_q et r_1, \dots, r_k des éléments distincts de \mathbb{N} . On appelle matrice de Vandermonde généralisée associée à $\{\lambda_1, \dots, \lambda_k\}$ d'exposants $\{r_1, \dots, r_k\}$, la matrice $\mathbf{V}^{\mathbf{G}}$:

$$\mathbf{V}^{\mathbf{G}} = \begin{pmatrix} \lambda_1^{r_1} & \lambda_1^{r_2} & \cdots & \lambda_1^{r_k} \\ \lambda_2^{r_1} & \lambda_2^{r_2} & \cdots & \lambda_2^{r_k} \\ \vdots & \ddots & \ddots & \vdots \\ \lambda_k^{r_1} & \lambda_k^{r_2} & \cdots & \lambda_k^{r_k} \end{pmatrix}$$

Le déterminant d'une matrice de Vandermonde généralisée n'admet pas de formule aussi simple que dans le cas des matrices de Vandermonde. En corps fini, il n'y a pas de discrimination simple des matrices de Vandermonde généralisées inversibles. Une telle discrimination se révélerait une plus-value certaine pour la construction directe de matrices MDS.

Un dernier résultat matriciel connexe aux matrices de Vandermonde sera plus adapté aux anneaux de polynômes non-commutatifs introduits ultérieurement, et se démontre par récurrence :

Proposition 2 ([LN97], Chapitre 3, Lemme 3.51). *Soient $\lambda_1, \dots, \lambda_k \in \mathbb{F}_{q^m}$. Alors :*

$$\begin{vmatrix} \lambda_1 & \lambda_2 & \cdots & \lambda_k \\ \lambda_1^q & \lambda_2^q & \cdots & \lambda_k^q \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_1^{q^{k-1}} & \lambda_2^{q^{k-1}} & \cdots & \lambda_k^{q^{k-1}} \end{vmatrix} = \lambda_1 \prod_{j=1}^{k-1} \prod_{c_1, \dots, c_j \in \mathbb{F}_q} (\lambda_{j+1} - \sum_{\ell=1}^j c_\ell \lambda_\ell)$$

Ce déterminant est non nul si et seulement si $\{\lambda_1, \dots, \lambda_k\}$ forme une famille \mathbb{F}_q -libre.

Ce dernier résultat suggère l'introduction d'une nouvelle définition qui adapte les matrices de Vandermonde à ces considérations de linéarité :

Définition 3. *Soit θ un élément de $\text{GAL}(\mathbb{F}_{q^m}/\mathbb{F}_q)$. Soient $\lambda_1, \dots, \lambda_k$ des éléments de \mathbb{F}_{q^m} . On appelle θ -matrice de Vandermonde associée à $\{\lambda_1, \dots, \lambda_k\}$, la matrice \mathbf{V}_θ :*

$$\mathbf{V}_\theta = \begin{pmatrix} \lambda_1 & \lambda_1^{[1]} & \cdots & \lambda_1^{[k-1]} \\ \lambda_2 & \lambda_2^{[1]} & \cdots & \lambda_2^{[k-1]} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_k & \lambda_k^{[1]} & \cdots & \lambda_k^{[k-1]} \end{pmatrix}$$

On a introduit ici la notation $a^{[1]}$ pour $\theta(a)$, plus pratique à manipuler. Elle se décline pour tout $i \in \mathbb{N}$ pour la composition i fois de θ , $a^{[i]} = \theta^i(a)$.

2.2 Polynômes

L'objet des études qui suivent est l'utilisation d'associations entre certaines structures matricielles et certains objets algébriques. On introduit la notion de division euclidienne sur les polynômes, base des démonstrations des théorèmes sur les idéaux de $\mathbb{F}_q[X]$ et de $\mathbb{F}_q[X]/(X^n - 1)$:

Théorème 1 ([LN97], Chapitre 1, Théorème 1.52). *Soit $g(X) \in \mathbb{F}_q[X]$, un polynôme non nul. Pour tout $f(X) \in \mathbb{F}_q[X]$, il existe $q(X)$ et $r(X) \in \mathbb{F}_q[X]$ tels que :*

$$f(X) = q(X)g(X) + r(X), \quad \text{avec } \text{deg}(r(X)) < \text{deg}(g(X))$$

Démonstration. Ce théorème se prouve par induction sur le degré de $f(X)$. \square

On peut alors démontrer que $\mathbb{F}_q[X]$ est un anneau principal :

Proposition 3 ([LN97], Chapitre 1, Théorème 1.54). *Tout idéal I de $\mathbb{F}_q[X]$ est principal.*

Démonstration. On prouve qu'il existe un unique polynôme unitaire $g(X) \in \mathbb{F}_q[X]$ tel que $I = (g(X))$. Pour cela, on considère $g(X)$, polynôme unitaire de degré minimal de I . Il existe du fait que le degré est une fonction à valeurs dans \mathbb{N} . Par division euclidienne de tout autre polynôme dans I et minimalité du degré de $g(X)$, on prouve son unicité et le fait qu'il divise tout élément de I . \square

La Proposition 3 induit la définition des polynômes minimaux :

Définition 4. *Soit λ un élément d'une extension de corps de \mathbb{F}_q . Le polynôme minimal de λ sur \mathbb{F}_q est l'unique polynôme unitaire qui engendre l'idéal des polynômes de \mathbb{F}_q qui s'annulent en λ .*

On est alors en mesure de prouver le théorème suivant sur les idéaux de l'anneau $\mathbb{F}_q[X]/(X^n - 1)$:

Théorème 2. *Soit I un idéal de $\mathbb{F}_q[X]/(X^n - 1)$. Alors,*

1. *Il existe un unique polynôme unitaire de plus petit degré $g(X) \in I$.*
2. *I est principal et est engendré par $g(X)$.*
3. *$g(X)$ divise $X^n - 1$ dans $\mathbb{F}_q[X]$.*

Démonstration. A l'instar de la démonstration de la Proposition 3, on définit $g(X)$ comme un polynôme unitaire de plus petit degré dans $\mathbb{F}_q[X]/(X^n - 1)$ (le degré est alors le degré du représentant dont le degré est inférieur à n). Les mêmes raisonnements issus de la division euclidienne par $g(X)$ permettent de garantir son unicité et la divisibilité de tout élément de I par $g(X)$. Alors, la division euclidienne de $X^n - 1$ par $g(X)$ assure par minimalité du degré de $g(X)$ un reste nul et la divisibilité de $X^n - 1$ par $g(X)$. \square

On considère maintenant la notion de polynômes réciproques.

Définition 5. *Le polynôme réciproque $g^*(X)$ d'un polynôme $g(X)$ de degré m est le polynôme défini par $g^*(X) = X^m g(X^{-1})$.*

On a alors la propriété immédiate :

Proposition 4. *Le polynôme réciproque d'un polynôme irréductible est irréductible.*

On peut définir sur les corps finis une notion de polynôme dérivé :

Définition 6. *Soit $f(X) = \sum_{i=0}^k f_i X^i \in \mathbb{F}_q[X]$. On appelle polynôme dérivé de $f(X)$, noté $f'(X)$, le polynôme $f'(X) = \sum_{i=1}^k i f_i X^{i-1}$.*

L'intérêt de ces polynômes se fonde sur le théorème suivant :

Théorème 3 ([LN97], Chapitre 1, Théorème 1.68). *Un élément $\lambda \in \mathbb{F}_q$ est une racine multiple de $f(X) \in \mathbb{F}_q[X]$ si et seulement si c'est une racine de $f(X)$ et de $f'(X)$.*

Démonstration. Soit $f(X) \in \mathbb{F}_q[X]$ et $\lambda \in \mathbb{F}_q$ une racine double. Alors, $f(X) = (X - \lambda)^2 g(X)$ avec $g(X) = \sum_{i=0}^{k-2} g_i X^i$. On a alors $f'(X) = \sum_{i=0}^{k-2} (i+2)g_i X^{i+1} - 2(i+1)\lambda g_i X^i + \lambda^2 g_i X^{i-1}$ et

$$f'(\lambda) = \sum_{i=0}^{k-2} (i+2 - 2i - 2 + i)g_i \lambda^{i+1} = 0$$

Réciproquement, si $f(X)$ et $f'(X)$ s'annulent en λ alors $f(X) = (X - \lambda)g(X)$ avec $g(X) = \sum_{i=0}^{k-1} g_i X^i$. On veut montrer que $g(\lambda) = 0$. Or, $f'(X) = \sum_{i=0}^{k-1} (i+1)g_i X^i - \lambda g_i X^{i-1}$ et finalement

$$f'(\lambda) = \sum_{i=0}^{k-1} g_i \lambda^i = g(\lambda) = 0$$

□

Définition 7. Le poids d'un polynôme $f(X) \in \mathbb{F}_q[X]$, noté $w_H(f(X))$ est le nombre de ses coefficients non nuls.

2.3 θ -polynômes

L'essentiel des travaux originaux présentés dans cette thèse transposent les associations entre structures matricielles et structures polynomiales aux anneaux de polynômes non commutatifs. Plus précisément, tout au long de cette thèse, on se concentre sur les anneaux de polynômes non commutatif introduits par Øystein Ore [Ore33b ; Ore34]. Ils apparaissent dans les travaux de Berlekamp sous le nom d'anneaux de polynômes linéarisés [Ber84]. Ils apparaissent également dans le domaine du calcul formel : les travaux de Bronstein et Petkovšek ont par exemple développé les méthodes de factorisation dans ces anneaux [BP94]. Nous les référençons ici sous le nom d'anneaux de θ -polynômes pour faire apparaître dans leur nomenclature θ , cœur de la non-commutativité de la structure. La construction de codes linéaires à partir de ces anneaux de polynômes a été particulièrement l'objet des travaux de Delphine Boucher et Felix Ulmer [BGU07 ; BU09 ; BU10].

Soit θ un élément de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$. On définit $\mathbb{F}_{q^m}[X; \theta]$ l'anneau construit à partir de l'ensemble des polynômes formels pour lesquels les coefficients sont écrits à gauche de l'indéterminée X , $\{\sum_{i=0}^k a_i X^i, a_i \in \mathbb{F}_q \text{ et } k \in \mathbb{N}\}$, muni des deux opérations suivantes, étendues par associativité et distributivité :

- *Addition* : addition usuelle des polynômes.
- *Multiplication par un élément de \mathbb{F}_{q^m}* : $Xa = \theta(a)X = a^{[1]}X$.

On remarque que remplacer θ par l'identité permet de retrouver la définition classique des polynômes. Pour éviter toute confusion, les θ -polynômes appartenant à un anneau $\mathbb{F}_{q^m}[X; \theta]$ seront notés $g\langle X \rangle$ plutôt que $g(X)$.

Exemple 1. Cet exemple permet d'appréhender l'arithmétique de cet anneau. Considérons $f\langle X \rangle = X + 1$ et $g\langle X \rangle = X + \alpha$, où α appartient à un certain corps de caractéristique 2. θ est ici l'automorphisme de Frobenius : $\theta : a \mapsto a^2$. On a :

- $f\langle X \rangle \cdot g\langle X \rangle = X^2 + (\alpha^2 + 1)X + \alpha$
- $g\langle X \rangle \cdot f\langle X \rangle = X^2 + (\alpha + 1)X + \alpha$

Ainsi $f\langle X \rangle \cdot g\langle X \rangle = g\langle X \rangle \cdot f\langle X \rangle$ si et seulement si $\alpha^2 = \alpha$, soit $\alpha = 0$ ou $\alpha = 1$.

La torsion de la multiplication par l'application de θ préserve certaines propriétés très utiles pour l'arithmétique des corps finis. En particulier, on a l'équivalent de la division euclidienne dans $\mathbb{F}_{q^m}[X; \theta]$, à ceci près qu'il faut différencier par non-commutativité la division euclidienne à droite et la division euclidienne à gauche :

Théorème 4 ([Ore33b]). *Soit $g\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$. Pour tout $f\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$, il existe des uniques $q_1\langle X \rangle, q_2\langle X \rangle, r_1\langle X \rangle$ et $r_2\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$ tels que :*

$$\begin{aligned} f\langle X \rangle &= g\langle X \rangle \cdot q_1\langle X \rangle + r_1\langle X \rangle && \text{où } \mathbf{deg}(r_1\langle X \rangle) < \mathbf{deg}(g\langle X \rangle) \\ f\langle X \rangle &= q_2\langle X \rangle \cdot g\langle X \rangle + r_2\langle X \rangle && \text{où } \mathbf{deg}(r_2\langle X \rangle) < \mathbf{deg}(g\langle X \rangle) \end{aligned}$$

Démonstration. Ce théorème se prouve par induction sur le degré de $f\langle X \rangle$. □

Muni de ces deux divisions euclidiennes, on retrouve des résultats similaires au cas des anneaux de polynômes classiques :

Proposition 5 ([Ore33b]). *Tout idéal à gauche, respectivement à droite, de $\mathbb{F}_{q^m}[X; \theta]$ est principal.*

Démonstration. La démonstration de ce résultat est le calque de celle de la Proposition 3. □

on écrira mod_*g l'opération qui calcule le reste de la division euclidienne à droite par $g\langle X \rangle$:

$$c\langle X \rangle \text{ mod}_*g = r\langle X \rangle \Leftrightarrow c\langle X \rangle = b\langle X \rangle \cdot g\langle X \rangle + r\langle X \rangle, \mathbf{deg}(r\langle X \rangle) < \mathbf{deg}(g\langle X \rangle)$$

On peut alors caractériser les idéaux à gauche de $\mathbb{F}_{q^m}[X; \theta]/(X^n - 1)$:

Théorème 5. *Soit I un idéal à gauche de $\mathbb{F}_{q^m}[X; \theta]/(X^n - 1)$. Alors,*

1. *Il existe un unique θ -polynôme unitaire de plus petit degré $g\langle X \rangle \in I$.*
2. *I est principal et est engendré par $g\langle X \rangle$.*
3. *$g\langle X \rangle$ divise à droite $X^n - 1$ dans $\mathbb{F}_{q^m}[X; \theta]$.*

Démonstration. La division euclidienne à droite de tout θ -polynôme de I par $g\langle X \rangle$ polynôme unitaire de degré minimal assure son unicité ainsi que la divisibilité à droite de tout élément de I par $g\langle X \rangle$. Une nouvelle division euclidienne à droite de $X^n - 1$ par $g\langle X \rangle$ assure le dernier point du théorème. □

Une notion d'évaluation associée aux θ -polynômes s'appelle l'évaluation par les opérateurs et se définit comme suit. Pour tout θ -polynôme $g\langle X \rangle = \sum_{i=0}^k g_i X^i$ et tout élément $a \in \mathbb{F}_{q^m}$, on pose :

$$g\langle a \rangle = \sum_{i=0}^k g_i a^{[i]}$$

L'évaluation correspond ainsi à une application linéaire, d'où la dénomination de polynôme linéarisés proposée par Berlekamp. Le noyau du polynôme est alors le noyau de l'application linéaire d'évaluation.

Remarque 1. *Il existe une autre évaluation associée aux θ -polynômes, appelée évaluation par les restes et se définit comme suit. Pour tout θ -polynôme $g\langle X \rangle = \sum_{i=0}^k g_i X^i$ et tout élément $a \in \mathbb{F}_{q^m}$, on pose :*

$$g\langle a \rangle = g\langle X \rangle \text{ mod}_*(X - a)$$

Théorème 6 ([Ore33a], Théorème 5). *Soit θ un automorphisme de \mathbb{F}_q , défini par $a \mapsto a^{q_0}$. \mathbb{F}_{q_0} est alors le corps fixé par θ . Il existe une extension de corps \mathbb{F}_{q^s} de \mathbb{F}_{q_0} qui contient toutes les racines de $g\langle X \rangle = \sum_{i=0}^k g_i X^i \in \mathbb{F}_q[X; \theta]$ et le \mathbb{F}_{q_0} -espace vectoriel de \mathbb{F}_{q^s} est de dimension $k - \min\{i, g_i \neq 0\}$.*

Démonstration. La preuve de ce théorème consiste à remarquer que λ racine du θ -polynôme $g\langle X \rangle = \sum_{i=0}^k g_i X^i$ est une racine du polynôme $\sum_{i=0}^k g_i X^{q_0^i}$. Ce polynôme possède q_0^k racines comptées avec multiplicité. Enfin, si $\min\{i, g_i \neq 0\} = \ell$,

$$\sum_{i=0}^k g_i X^{q_0^i} = \sum_{i=\ell}^k g_i^{q_0^\ell} X^{q_0^i} = \left(\sum_{i=\ell}^k g_i^{(q_0 - q_0)^\ell} X^{q_0^{i-\ell}} \right)^{q_0^\ell}$$

Le polynôme entre parenthèse est à racines simples puisque son polynôme dérivé égale $g_\ell^{(q_0 - q_0)^\ell} \neq 0$ et est donc premier avec lui. Les multiplicités des racines valent donc q_0^ℓ , d'où la dimension de l'espace vectoriel $k - \ell$. \square

Réciproquement, pour θ automorphisme de \mathbb{F}_q qui fixe \mathbb{F}_{q_0} , on peut définir à partir de tout \mathbb{F}_{q_0} -espace vectoriel inclus dans une extension de corps \mathbb{F}_{q^s} de dimension k un unique polynôme interpolateur unitaire de degré k dont l'espace vectoriel est le noyau de l'application linéaire d'évaluation. Pour le construire, il suffit de développer l'équation suivante :

$$\begin{vmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_k & 1 \\ \lambda_1^{[1]} & \lambda_2^{[1]} & \dots & \lambda_k^{[1]} & X \\ \vdots & & & \vdots & \vdots \\ \lambda_1^{[k]} & \lambda_2^{[k]} & \dots & \lambda_k^{[k]} & X^k \end{vmatrix} = 0$$

Définition 8. *Le poids d'un θ -polynôme $f\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$, noté $w_H(f\langle X \rangle)$ est le nombre de ses coefficients non nuls.*

Première partie

Couches de diffusion linéaire
légères à partir de matrices MDS

Chapitre 3

Présentation générale

Sommaire

3.1 Couches de diffusion linéaire	18
3.1.1 Réseau Substitution-Permutation	19
3.1.2 Attaques statistiques	20
3.1.3 Wide Trail Strategy	21
3.2 Codes et matrices	24
3.2.1 Codes linéaires et nombres de branchements	25
3.2.2 Matrices MDS	28
3.2.3 Codes et matrices MRD	31
3.3 Implémentations matérielles légères	34
3.3.1 Mesure heuristique des coûts d'implémentation matérielle	34
3.3.2 Trois tentatives de réduction des coûts	36
3.3.3 Matrices de Hadamard-Cauchy	40

La naissance de la cryptographie symétrique moderne possède sa pierre blanche, l'article séminal de Claude Shannon : *Communication theory of secrecy systems* [Sha49]. Les fondations des schémas de chiffrement symétriques modernes y sont posées et les propriétés heuristiques de sécurité y sont définies. La *confusion*, complexité des relations entre la clef de chiffrement et les messages chiffrés, la *diffusion*, l'impossibilité d'observer des relations statistiques sur un texte chiffré à partir de relations statistiques sur le texte en clair, en sont alors les caractéristiques fondamentales. Toutes les attaques contre les primitives cryptographiques symétriques, les *cryptanalyses*, reposent sur une défaillance de l'une ou l'autre de ces deux propriétés. Le réseau substitution-permutation, en anglais *Substitution-Permutation Network* ou encore SPN, est une architecture de primitive de chiffrement symétrique par bloc profondément influencée par cette idée. Il consiste en l'itération d'une routine composée de deux opérations : une opération de confusion, la *substitution*, non linéaire, et une opération de diffusion, la *permutation*, linéaire. Les codes correcteurs d'erreurs ont fait leur apparition en cryptographie conjointement pour les algorithmes de chiffrement symétriques et les algorithmes de chiffrement asymétriques. Si leur utilisation est très naturelle dans les couches de diffusion linéaire des réseaux substitution-permutation, ils sont également un candidat sérieux pour réaliser des échanges de clefs, des signatures et des chiffrements asymétriques résistants à l'algorithmique quantique. L'étude des codes MDS, codes optimaux pour la distance de Hamming, est un sujet d'étude inaltérable de la communauté des codeurs. Très réguliers, la structure de ces objets assure de nombreuses propriétés très intéressantes

pour la cryptographie. Enfin, s'ils sont très étudiés, on connaît jusqu'à la répartition des mots dans un code linéaire MDS, il conservent leur part de mystère et même pour les codes cycliques, objets standards s'il en est, il reste de très nombreuses conjectures sur leur comportement. La conjecture MDS, Graal de cette communauté, demeure d'ailleurs à l'état de conjecture malgré les avancées de Siméon Ball [Bal12; BJ12] à ce sujet. De nombreuses plateformes différentes intègrent des modules cryptographiques. Avec l'utilisation massive de l'AES et les ressources en mémoire foisonnantes des implémentations logicielles, les problématiques d'optimisation logicielle se sont focalisées sur l'accélération des calculs. L'optimisation des implémentations matérielles s'oriente essentiellement vers la réduction des surfaces nécessaires, facteur limitant pour les plus petites plateformes et les implémentations légères sont devenues un enjeu majeur des conceptions de primitive. La diversité des supports d'implémentation matérielle est telle que la détermination d'une architecture optimale pour toutes est illusoire. Cependant, il est possible de mettre en place des métriques de comparaisons des coûts matériels. Le compte de XOR, une de ces métriques, s'est peu à peu imposée comme élément de comparaison des coûts d'implémentation matérielle car simple à manipuler et relativement proche des coûts réels induits.

Ce chapitre introduit les propriétés que doivent satisfaire les couches de diffusion linéaires utilisées dans les schémas de chiffrement par bloc symétriques. La présentation des attaques statistiques sur ces schémas va motiver l'utilisation de codes correcteurs d'erreurs optimaux pour construire ces couches de diffusion linéaire. Ce chapitre se conclut alors par la présentation des problématiques d'optimisation des coûts des implémentations matérielles ainsi que de certaines structures matricielles qui visent à réduire ces coûts.

3.1 Couches de diffusion linéaire

Le rôle de l'opération de diffusion linéaire dans une primitive cryptographique symétrique est de répandre au maximum les dépendances internes : idéalement, une perturbation arbitraire d'un message doit modifier chaque bit du chiffré indépendamment avec probabilité 2^{-1} . Pour modéliser ce comportement idéal, on souhaite qu'un bit de sortie corresponde à une fonction parfaitement aléatoire en les n entrées : que tout monôme de $\mathbb{F}_2[X_1, \dots, X_n]/(X_i^2 - X_i)$ apparaisse avec probabilité 2^{-1} dans la fonction booléenne du bit de sortie en les entrées X_i . Le coût d'une implémentation matérielle d'un tel polynôme *ad hoc* est prohibitif puisqu'il possède en moyenne 2^{n-1} coefficients pour chaque bit en sortie. Une solution naïve est donc de construire une *fonction de tour* avec des propriétés de diffusion et de confusion locales et de l'itérer pour obtenir cette propriété globale. Le *phénomène d'avalanche*, description de la propagation libre d'une perturbation, le *décal de diffusion*, mesure du nombre de tours nécessaires pour que toute sortie dépende de toute entrée, évaluent l'efficacité de la diffusion. Du point de vue de la sécurité, la qualité d'une couche de diffusion linéaire est quantifiée par le *nombre de branchements* qu'elle assure, mesure de la diffusion sur deux tours. Les fortes propriétés de diffusion impliquent malheureusement des implémentations lourdes. Les stratégies de conception des couches de diffusion linéaire consistent à fonder la sécurité contre les attaques statistiques sur l'accumulation sur plusieurs tours d'effets de diffusion locale.

La structure de Réseau Substitution-Permutation apparaît en filigrane des considérations précédentes et va être maintenant présentée. Ensuite, le principe des attaques statistiques sera exposé et servira de justification à l'introduction des concepts de conception de la *Wide Trail Strategy*. C'est cette dernière qui justifie l'usage de matrices MDS dans les couches de diffusion linéaire.

3.1.1 Réseau Substitution-Permutation

Pour obtenir à coûts raisonnables le comportement idéal décrit auparavant, on profite d'un effet d'accumulation. L'itération d'une fonction de tour avec des effets de confusion et de diffusion locaux permet de dissiper les effets statistiques observables. Une stratégie de conception consiste alors à alterner une opération dite de *substitution* composée d'applications non linéaires locales, les boîtes S , qui assurent la confusion par petits ensembles de bits et une opération dite de *permutation* qui diffuse les dépendances. Cette structure s'appelle le réseau substitution-permutation, en anglais Substitution Permutation Network (SPN), et est illustrée par la Figure 3.1. On peut alors modéliser la couche de diffusion linéaire comme une transformation \mathbb{F}_2 -linéaire sur $(\mathbb{F}_2^m)^k$. Pour la plupart des couches linéaires et dans le reste de cette thèse, on considère même des transformations \mathbb{F}_{2^m} -linéaires sur $\mathbb{F}_{2^m}^k$. Le choix de cette dernière offre un des nombreux compromis possibles entre garanties de sécurité et efficacité des calculs. On appelle *état interne* le registre de n bits auquel est appliqué cette fonction de tour indifféremment pour faire référence au registre d'entrées, au $r - 1$ registres intermédiaires à chaque tour et au registre de sortie.

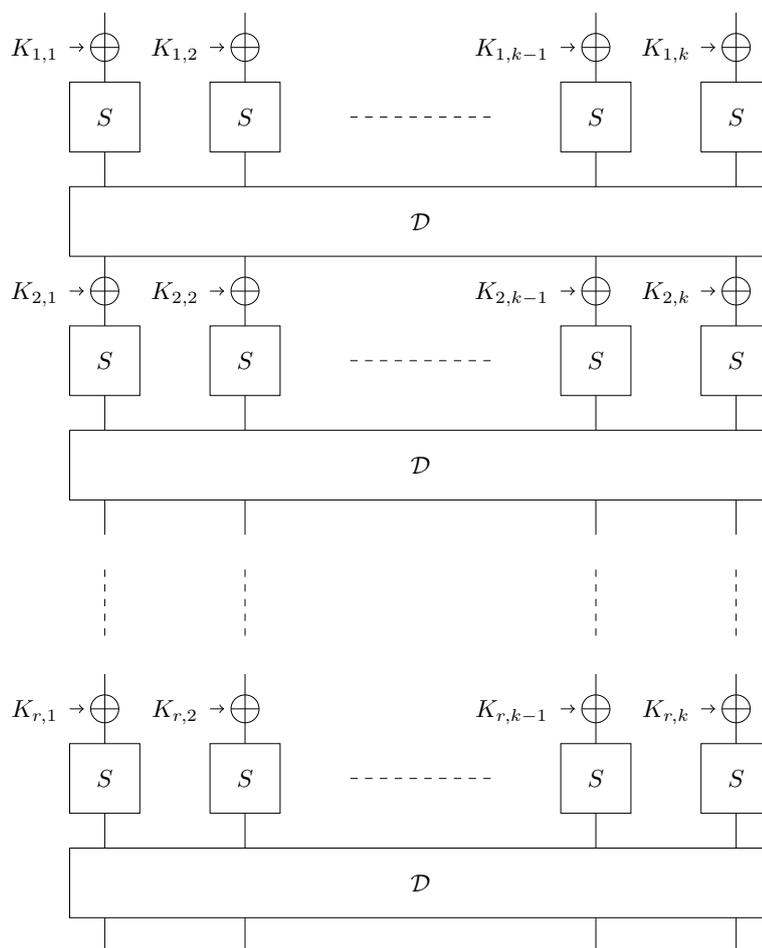


FIGURE 3.1 – Réseau substitution-permutation.

3.1.2 Attaques statistiques

Les attaques statistiques ont vu le jour au début des années 90, avec par exemple l'attaque de FEAL-8 par Henri Gilbert et Guy Chassé [GC90]. Jumelles, elles reposent sur l'utilisation de biais statistiques entre les entrées et les sorties pour l'*attaque linéaire* et entre les différences entre couples d'entrées et les différences de couples de sorties pour l'*attaque différentielle*. Un schéma de chiffrement par blocs se doit pour être pris au sérieux d'apporter des éléments de sécurité à cet égard. Hormis les attaques algébriques, dont il n'est pas question dans cette thèse, toutes les menaces connues sur les primitives symétriques dans l'arsenal dont dispose un cryptanalyste relèvent originellement de l'attaque linéaire ou de l'attaque différentielle. Elles constituent la principale menace contre les schémas de chiffrement symétriques modernes comme en témoigne la cryptanalyse du DES. Leur principe commun est l'utilisation d'une déviation statistique, *a priori* inobservable pour une fonction tirée au hasard, pour retrouver la clef secrète ou pour décrypter un message.

Attaque linéaire : Proposée par Matsui dans [Mat93], l'attaque linéaire est une attaque à clairs connus qui commence par la mise en évidence d'une déviation statistique sur les équations linéaires en les bits d'entrée et de sortie, les caractéristiques linéaires :

Définition 9. Soit E_K un schéma de chiffrement par bloc, muni d'une clef secrète K . Soit $\alpha \in \mathbb{F}_2^n$, appelé masque d'entrée. Soit $\beta \in \mathbb{F}_2^n$, appelé masque de sortie. On appelle caractéristique linéaire, notée $\mathcal{C}_{\alpha,\beta}$, l'équation :

$$\alpha \cdot X \oplus \beta \cdot E_K(X) = 0$$

Elle est vérifiée avec une probabilité $p = 2^{-1}(1 + \epsilon)$. On appelle ϵ le biais et ϵ^2 la corrélation de la caractéristique.

La première étape de l'attaque consiste à mettre en évidence un couple de masques (α, β) tel que la caractéristique linéaire associée $\mathcal{C}_{\alpha,\beta}$ soit fortement biaisée, si possible pour n'importe quelle clef. Une fois qu'une telle caractéristique a été déterminée sur presque tous les tours, la cryptanalyse consiste à évaluer le biais induit par différents choix de clefs partielles pour un grand nombre d'échantillons. On discrimine alors la vraie clef parmi toutes les possibilités en considérant les choix qui montrent la plus forte déviation.

Plusieurs hypothèses sont faites pour modéliser le comportement de ces caractéristiques linéaires et analyser les chances de succès d'une attaque linéaire, une fois un biais identifié. Une première hypothèse consiste à considérer que les sous-clefs de tour sont indépendantes et identiquement distribuées. Ceci permet pendant la cryptanalyse de considérer la propagation des masques linéaires biaisés tour par tour. La deuxième hypothèse consiste à considérer lors de la discrimination des clefs que les mauvais choix de clefs correspondent à des fonctions tirées aléatoirement parmi l'ensemble des permutations de \mathbb{F}_2^n . D'après le théorème de Parseval, on sait que la somme des corrélations associées à un masque d'entrée, respectivement à un masque de sortie, égale 1 pour n'importe quelle permutation. Alors, on suppose le comportement idéal suivant : une caractéristique linéaire associée à un couple de masques possède une corrélation de $2^{-\frac{n}{2}}$. En pratique, on considère sûre une primitive symétrique contre l'attaque linéaire, lorsque l'attaquant n'a pas les moyens de mettre en évidence une caractéristique linéaire avec une corrélation supérieure à $2^{-\frac{n}{2}}$.

Attaque différentielle : Proposée par Biham et Shamir dans [BS91], l'attaque différentielle est une attaque à clairs choisis qui commence par la mise en évidence d'une déviation statistique sur les différences en sortie, formalisée par la notion de caractéristique différentielle :

Définition 10. Soit E_K un schéma de chiffrement par bloc, muni d'une clef secrète K . Soit $\delta \in \mathbb{F}_2^n$, appelée différence d'entrée. Soit $\Delta \in \mathbb{F}_2^n$, appelée différence de sortie. On appelle caractéristique différentielle, notée $\mathcal{C}_{\delta,\Delta}$, l'équation :

$$E_K(X) \oplus E_K(X \oplus \delta) = \Delta$$

Elle est vérifiée avec probabilité $p = 2^{-1}(1 + \epsilon)$. On appelle p la probabilité et ϵ le biais de la caractéristique.

La première étape de l'attaque consiste à mettre en évidence un couple de différences (δ, Δ) tel que la caractéristique différentielle associée $\mathcal{C}_{\delta,\Delta}$ soit fortement biaisée, si possible pour n'importe quelle clef. À l'instar de la cryptanalyse linéaire, une fois qu'une telle caractéristique a été déterminée sur presque tous les tours, la cryptanalyse consiste à évaluer le biais induit par différents choix de clefs partielles pour un grand nombre d'échantillons. On discrimine alors la vraie clef parmi toutes les possibilités en considérant les choix qui montrent la plus forte déviation.

Plusieurs hypothèses sont possibles pour modéliser le comportement de ces caractéristiques différentielles et analyser la complexité et les chances de succès d'une attaque différentielle, une fois un biais identifié. Une première hypothèse consiste une nouvelle fois à considérer que les sous-clefs de tour sont indépendantes et identiquement distribuées. Ceci permet pendant la cryptanalyse de considérer la propagation des différences tour par tour. La deuxième hypothèse consiste à considérer lors de la discrimination des clefs que les mauvais choix de clefs correspondent à des fonctions tirées aléatoirement parmi l'ensemble des permutations de \mathbb{F}_2^n . La somme des probabilités qu'une différence de sortie soit obtenue à partir d'une différence d'entrée sur toutes les différences de sortie égale 1 pour n'importe quelle permutation. On remarque enfin que lorsqu'un couple d'entrées $(X, X \oplus \delta)$ engendre une différence Δ , le couple $(X \oplus \delta, X)$ aussi. Alors, on suppose le comportement idéal suivant : une caractéristique différentielle associée à un couple de différence possède une probabilité de 2^{1-n} avec probabilité 2^{-1} . En pratique, on considère sûre une primitive symétrique contre l'attaque différentielle, lorsque l'attaquant n'a pas les moyens de mettre en évidence une caractéristique différentielle avec une probabilité supérieure à 2^{1-n} .

3.1.3 Wide Trail Strategy

La représentation de la Figure 3.1 induit les réflexions qui suivent. La *Wide Trail Strategy* est une stratégie de conception de la fonction de tour d'un réseau substitution-permutation formulée pour la primitive Rijndael [DR02], algorithme choisi pour le standard AES, et que partagent de nombreux schémas de chiffrement par bloc publiés depuis. Son objectif de combiner l'efficacité du chiffrement et la résistance aux cryptanalyses linéaires et différentielles. Cette résistance est justifiée par un traitement indifférencié des attaques linéaires et différentielles, grâce à des effets de structure.

La recherche de fonctions hautement non linéaires sur un grand nombre de bits est très coûteuse et leurs réalisations pratiques, particulièrement au niveau de l'implémentation matérielle, possèdent des coûts exponentiels en le nombre des entrées. La réponse de la *Wide Trail Strategy*, comme suggérée par la représentation de la Figure 3.1 est l'utilisation de fonctions non linéaires locales optimales, les boîtes S et de considérer l'état interne selon un découpage en *mots*, partition de

l'état interne induit par les boîtes S. Pour faciliter la parallélisation et en raison de l'absence de gain de sécurité à la diversification des boîtes S, il est raisonnable qu'elles soient toutes les mêmes. Des boîtes S différentes ne modifient pas les raisonnements qui suivent, la *Wide Trail Strategy* se sert uniquement de la structure macroscopique et non des valeurs particulières de ces boîtes S pour fonder ses arguments de sécurité. L'utilisation de boîtes S différentes augmente en revanche la taille des programmes en logiciel et la surface nécessaire des implémentations matérielles.

Cette architecture de la couche de substitution non-linéaire en boîtes S locales permet d'appréhender la propagation d'un masque linéaire d'entrée ou d'une différence en entrée de cette couche de substitution. Les *transitions linéaires probabilistes*, couples de masques en amont et en aval d'une couche de substitution, et les *transitions différentielles probabilistes*, couples de différences en amont et en aval d'une couche de substitution, se considèrent séparément en entrée sur chaque boîte. Un *masque intermédiaire*, respectivement une *différence intermédiaire*, est alors une concaténation de masques locaux, respectivement de différences locales en aval d'une couche de substitution. Le calcul des corrélations, respectivement des probabilités, associées est alors le produit des corrélations, respectivement des probabilités, des transitions probabilistes locales. La couche de substitution engendre à partir d'un masque d'entrée plusieurs masques intermédiaires en sortie de cette couche de substitution avec des probabilités non nulles et pareillement à partir d'une différence d'entrée, elle engendre plusieurs différences intermédiaires. Linéaire, la couche de permutation n'induit pas, comme la couche de substitution, de multiplications des masques et des différences possibles. Les transitions entre les valeurs de masques et de différences sont déterministes.

Un paramètre d'évaluation de la sécurité consiste alors à considérer, pour une boîte S, la probabilité maximale d'une transition différentielle probabiliste et la corrélation maximale d'une transition linéaire probabiliste à travers la boîte S :

Définition 11. Soit S une boîte S, une permutation non linéaire de \mathbb{F}_2^m .

On note ϵ_S la corrélation maximale d'un couple de masques en entrée et en sortie de S :

$$\epsilon_S = \max_{\alpha, \beta \in \mathbb{F}_2^m} (\epsilon_{\alpha, \beta})$$

On note p_S la probabilité maximale d'un couple de différences en entrée et en sortie de S :

$$p_S = \max_{\delta, \Delta \in \mathbb{F}_2^m} (p_{\delta, \Delta})$$

On choisit des boîtes S qui possèdent de bonnes propriétés de non linéarité. C'est un travail délicat même pour des petites tailles. Les boîtes S optimales de taille 4×4 ont été analysées et regroupées par classes d'équivalence affine par Gregor Leander et Axel Poschmann dans [LP07] puis par classes d'équivalence raffinées par Markku-Juhani Saarinen dans [Saa11].

On appelle alors *chemin linéaire* un choix de masques intermédiaires à chaque tour.

Définition 12. Soient $\alpha_0, \dots, \alpha_r, \beta_0, \dots, \beta_r \in \mathbb{F}_2^n$. On appelle *chemin linéaire* la suite :

$$\alpha_0 \rightarrow \beta_0 \rightarrow \alpha_1 \rightarrow \dots \rightarrow \alpha_r \rightarrow \beta_r$$

$\alpha_i \rightarrow \beta_i$ est ici une transition linéaire probabiliste à travers la couche de substitution tandis que $\beta_i \rightarrow \alpha_{i+1}$ est une transition linéaire déterministe à travers la couche de permutation.

On appelle de même *chemin différentiel* un choix de différences intermédiaires à chaque tour.

Définition 13. Soient $\delta_0, \dots, \delta_r, \Delta_0, \dots, \Delta_r$, on appelle chemin linéaire la suite :

$$\delta_0 \rightarrow \Delta_0 \rightarrow \delta_1 \rightarrow \dots \rightarrow \delta_r \rightarrow \Delta_r$$

$\delta_{i0} \rightarrow \Delta_i$ est ici une transition différentielle probabiliste à travers la couche de substitution tandis que $\Delta_i \rightarrow \delta_{i+1}$ est une transition différentielle déterministe à travers la couche de permutation.

Un masque non nul, respectivement une différence non nulle, en entrée d'une boîte S contribue à la multiplication des chemins et à la diminution de leurs probabilités associées. On parle alors de *boîte active*. Il est plus efficace en termes de coûts d'implémentation de n'utiliser que des petites boîtes S, qui engendrent une moindre prolifération des chemins mais plus probables, et de concevoir des diffusions linéaires qui assurent un grand nombre de boîtes actives, ce qui va *a contrario* augmenter le nombre de chemins possibles et diminuer leurs probabilités respectives. La part relative des ressources allouées à l'étape linéaire pour fournir une grande diffusion sur plusieurs tours est donc importante. Cette stratégie a l'avantage qu'on peut la spécialiser sur un grand nombre de boîtes S différentes, avec pour seule exigence qu'elle soient optimales pour ϵ_S et p_S . Assurer au cours de leurs propagations durant les r tours, que les chemins touchent un minimum de boîtes S permet de rendre toute recherche exhaustive impossible et d'assurer qu'un chemin ne puisse servir seul à assurer un biais qui justifie l'exploitation d'une caractéristique. La cryptanalyse doit alors mettre en lumière une accumulation de chemins de faibles probabilités pour assurer qu'une caractéristique soit exploitable. La couche de diffusion doit rendre ces effets d'accumulation très dépendants de la clef et indétectable à la cryptanalyse.

La partition de l'état interne en mots induite par les boîtes S oriente la structure de la couche de diffusion linéaire. Elle doit mélanger les mots considérés comme des éléments de \mathbb{F}_{2^m} , où m représente la taille de la boîte S. Chaque mot est la sortie d'une fonction \mathbb{F}_{2^m} -linéaire des mots donc \mathbb{F}_2 -linéaire des bits. Une représentation naturelle d'une telle couche de diffusion linéaire est alors donnée par une matrice \mathbf{M} dans $\mathbb{F}_{2^m}^{k \times k}$:

$$\mathbf{y} = \mathbf{M}\mathbf{x}$$

Deux mécanismes de conception se dégagent :

- Choisir des boîtes S avec ϵ_S et p_S maximaux.
- Construire la couche de diffusion linéaire de sorte qu'il n'existe pas de chemins avec un faible nombre de boîtes actives.

L'addition de sous-clef de tour n'a pas d'impact sur le poids des boîtes actives. Une mesure pertinente de la diffusion est le nombre minimal de boîtes actives entre des entrées et les sorties d'une étape linéaire. Un minimum est donné pour une couche linéaire par son nombre de branchements, lui même borné par le nombre de boîtes actives minimal maximal sur deux tours, $k + 1$.

Définition 14. Le nombre de branchements différentiel d'une application \mathcal{D} représentée par une matrice \mathbf{D} est donné par :

$$\mathcal{B}_d(\mathcal{D}) = \min_{\mathbf{a}, \mathbf{b} \in \mathbb{F}_2^k} w_m(\mathbf{a} \oplus \mathbf{b}) + w_m(\mathbf{D}(\mathbf{a}) \oplus \mathbf{D}(\mathbf{b}))$$

Le nombre de branchements linéaire d'une application \mathcal{D} représentée par une matrice \mathbf{D} est donné par :

$$\mathcal{B}_\ell(\mathcal{D}) = \min_{\substack{\alpha, \beta \in \mathbb{F}_2^n \\ C(\alpha^T \mathbf{x}, \beta^T \mathbf{D}(\mathbf{x})) \neq 0}} w_m(\alpha) + w_m(\beta)$$

w_m représente ici le poids de Hamming en mots des masques : le nombre de mots non nuls des masques, respectivement des différences.

Pour davantage de deux tours, les propriétés désirées sur la couche linéaire sont moins triviales. Pour un chemin sur $2r$ tours, on a l'assurance que le pire cas compte davantage de boîtes actives que r séquences minimales à 2 tours. La solution privilégiée par Rijndael consiste à choisir une couche linéaire avec un nombre de branchements faible mais avec une grande diffusion locale sur des sous-ensembles de mots suivie d'une étape de dispersion qui mélange ces sous-ensembles. Une autre stratégie consiste à maximiser le nombre de branchements sur l'ensemble des mots. Qu'on privilégie l'une ou l'autre de ces deux possibilités enjoint à considérer des constructions linéaires optimales globales ou locales et appliquées en parallèle.

Remarque 2. *Le nombre de branchements linéaire d'une application spécifiée par une matrice \mathbf{D} est égal au nombre de branchements différentiel de l'application spécifiée par la matrice \mathbf{D}^T . Ces nombres de branchements respectivement à une partition ne sont pas, en toute généralité, égaux. Néanmoins, il est possible d'en construire, un exemple évident est de considérer les applications spécifiées par des matrices symétriques.*

3.2 Codes et matrices

Les codes correcteurs d'erreurs sont des objets largement étudiés, transverses à de multiples disciplines. Conçus pour assurer la viabilité des communications sur des canaux de transmissions bruités, ils sont aussi reliés aux problématiques de définition et de maximisation des nombres de branchements induits par la description de la *Wide Trail Strategy*. Parmi l'ensemble des codes, les codes dits linéaires bénéficient d'une structure adaptée à la définition de couches de diffusion linéaire. La notion de distance minimale pour la distance de Hamming d'un code linéaire est une traduction théorique dans le cadre des couches linéaires de primitives symétriques du nombre minimal de boîtes actives d'un chemin linéaire ou différentiel sur deux tours. La théorie des codes permet alors de définir, caractériser et même construire des codes optimaux du point de vue de cette distance. Ces codes optimaux possèdent des propriétés qui les destinent à leur utilisation en cryptographie symétrique. Les matrices MDS, construites à partir de ces codes optimaux, transposent leurs qualités aux couches linéaires dont elles sont les représentations matricielles. Leur utilisation en cryptographie, non marginale, a rendu la détermination de ces matrices un enjeu important des conceptions de primitives symétriques. On les retrouve en effet dans un grand nombre d'entre elles dont suit une liste non exhaustive : AES [DR02], SQUARE [DKR97], TwoFish [Sch+99], SHARK [Rij+96], Clefia [Shi+07], Grøstl [Gau+09a], PHOTON [GPP11], LED [Guo+11]. Enfin, une autre distance, la distance rang, peut être introduite dont certains codes optimaux sont également optimaux pour la distance de Hamming. Une construction directe de codes optimaux pour la distance rang, les codes de Gabidulin, peut alors être présentée en lien avec les anneaux de θ -polynômes introduits dans les préliminaires.

Les codes correcteurs d'erreurs vont à présent être introduits et la notion de distance minimale sera reliée aux problématiques de maximisation des nombres de branchements. Alors seront définies les matrices MDS, issues de codes optimaux pour la distance de Hamming. Ces résultats sont des résultats classiques et peuvent être trouvés dans le livre *The Theory of Error-Correcting Codes* de Jessie MacWilliams et Neil Sloane ou encore dans l'ouvrage *Error Correction Coding* de

Todd Moon [MS77; Moo05]. Enfin, certains codes optimaux pour la distance rang se révéleront optimaux pour la distance de Hamming et seront à leur tour définis.

3.2.1 Codes linéaires et nombres de branchements

Afin de corriger des erreurs engendrées par un canal bruité, corrompant un message en des positions aléatoires, on utilise un code correcteur d'erreur qui introduit des redondances sur un message. L'idée générale est de définir un sous-ensemble de mots dans un espace ambiant éloignés les uns des autres pour une certaine distance de sorte qu'un mot bruité puisse, quand le bruit n'est pas trop important, être associé à un mot de code sans ambiguïté.

Définition 15. *Un code en bloc \mathcal{C} de paramètres (n, M) sur un alphabet \mathcal{A}_q à q symboles est un ensemble de M vecteurs de longueur n sur \mathcal{A}_q appelés mots de code.*

La correction d'erreur est, comme déjà mentionné, liée à une notion de distance entre les éléments de l'espace ambiant. La définition la plus naïve de distance dans ce contexte consiste au nombre de coefficients différents entre deux éléments :

Définition 16. *Soit \mathcal{C} un code (n, M) .*

- *Le poids de Hamming d'un mot de code $\mathbf{c} \in \mathcal{C}$, $w_H(\mathbf{c})$, est le nombre de coefficients non nuls de \mathbf{c} .*
- *Le poids minimal de \mathcal{C} , $w_{H,\min}$, est le plus petit poids de Hamming d'un mot de code :*

$$w_{H,\min} = \min_{\mathbf{c} \in \mathcal{C} \setminus \{0\}} w_H(\mathbf{c})$$

- *La distance minimale d de \mathcal{C} est le plus petit poids de Hamming de la différence entre deux mots de code :*

$$d_{\min} = \min_{\substack{\mathbf{c}_1, \mathbf{c}_2 \in \mathcal{C} \\ \mathbf{c}_1 \neq \mathbf{c}_2}} w_H(\mathbf{c}_1 - \mathbf{c}_2)$$

Les codes linéaires ont été introduits pour accélérer par le biais de raisonnements d'algèbre linéaire la recherche du mot le plus proche. La définition des codes linéaires est donc naturelle :

Définition 17. *Un code en bloc \mathcal{C} est dit linéaire si et seulement si ses q^k mots de code forment un sous-espace vectoriel de \mathbb{F}_q^n de dimension k . Les paramètres d'un code sont usuellement donnés sous la forme $[n, k]_q$, avec n appelé la longueur du code et k appelé la dimension du code.*

Dans l'étude des codes linéaires, la distance minimale d est un paramètre incontournable, au point qu'on caractérise généralement les codes par $[n, k, d]_q$. C'est alors aussi, comme le stipule le théorème suivant, le poids minimal du code :

Théorème 7 ([MS77], Chapitre 1, Théorème 1). *La distance minimale d d'un code linéaire \mathcal{C} satisfait $d = w_{H,\min}$.*

Démonstration. $d = \min_{\substack{\mathbf{c}_i, \mathbf{c}_j \in \mathcal{C} \\ \mathbf{c}_i \neq \mathbf{c}_j}} d_H(\mathbf{c}_i, \mathbf{c}_j) = \min_{\substack{\mathbf{c}_i, \mathbf{c}_j \in \mathcal{C} \\ \mathbf{c}_i \neq \mathbf{c}_j}} d_H(\mathbf{c}_i - \mathbf{c}_j, \mathbf{c}_j - \mathbf{c}_j) = \min_{\substack{\mathbf{c} \in \mathcal{C} \\ \mathbf{c} \neq \mathbf{0}}} w_H(\mathbf{c}) = w_{H,\min}. \quad \square$

L'espace dual d'un code est lui-même un code, le code dual, dont la définition suit :

Définition 18. L'espace dual d'un code linéaire $\mathcal{C} [n, k]_q$ est appelé code dual de \mathcal{C} , noté \mathcal{C}^\perp . C'est un code $[n, n - k]_q$. Lorsque $\mathcal{C} = \mathcal{C}^\perp$, on appelle \mathcal{C} un code auto-dual.

À un code linéaire \mathcal{C} de paramètres $[n, k]_q$, sous-espace linéaire de dimension k de \mathbb{F}_q^n , on associe des matrices génératrices de \mathcal{C} , notées $\mathbf{G} \in \mathbb{F}_q^{k \times n}$, dont les lignes forment une base de \mathcal{C} . Le code dual, \mathcal{C}^\perp , code linéaire de paramètres $[n, n - k]_q$, est associé à des matrices génératrices, appelées matrices de parité de \mathcal{C} , notées $\mathbf{H} \in \mathbb{F}_q^{(n-k) \times n}$, dont les lignes forment une base du code dual. On a donc la relation matricielle suivante :

$$\mathbf{G}\mathbf{H}^T = \mathbf{0}$$

Le théorème suivant est d'un grand intérêt car il relie la distance minimale d'un code à la matrice de parité :

Théorème 8 ([MS77], Chapitre 1, Théorème 10). Soit \mathcal{C} un code en bloc linéaire de paramètres $[n, k, d]_q$ et \mathbf{H} une matrice de parité de \mathcal{C} . La distance minimale d égale le plus petit nombre positif de colonnes de \mathbf{H} qui vérifient une relation de dépendance linéaire. Ainsi, tout choix de $d - 1$ colonnes de \mathbf{H} est linéairement indépendant et il existe un choix de d colonnes de \mathbf{H} qui vérifient une relation de dépendance linéaire.

Démonstration. Notons $\mathbf{H}_1, \dots, \mathbf{H}_n$ les colonnes de \mathbf{H} . Par définition, pour tout $\mathbf{c} \in \mathcal{C}$, $\mathbf{c}\mathbf{H}^T = \mathbf{0}$. Développons cette expression :

$$\mathbf{0} = c_1\mathbf{H}_1 + \dots + c_n\mathbf{H}_n \quad (3.1)$$

Soit désormais \mathbf{c} un mot de plus petit poids $w_H(\mathbf{c}) = d$. On définit l'ensemble d'indices $\{i_1, \dots, i_d\}$ pour lesquels les coefficients correspondants de \mathbf{c} sont non nuls. Alors, l'équation (3.1) se réécrit :

$$\mathbf{0} = c_{i_1}\mathbf{H}_{i_1} + \dots + c_{i_d}\mathbf{H}_{i_d}$$

Réciproquement, par l'absurde, supposons l'existence d'une relation de dépendance linéaire vérifiée par $\ell < d$ colonnes de \mathbf{H} . L'écrivant, notant $\{i_1, \dots, i_\ell\}$ les indices des colonnes correspondantes et $c_{i_1}, \dots, c_{i_\ell}$ les coefficients de la relation de dépendance linéaire,

$$\mathbf{0} = c_{i_1}\mathbf{H}_{i_1} + \dots + c_{i_\ell}\mathbf{H}_{i_\ell},$$

on obtient un mot du code dual du code associé à la matrice génératrice \mathbf{H} , soit un mot du code \mathcal{C} de poids $\ell < d$. Absurde. \square

La distance minimale d'un code linéaire vérifie une borne fondamentale, la borne de Singleton :

Théorème 9 (Borne de Singleton, [MS77], Chapitre 1, Théorème 11). La distance minimale d'un code linéaire de paramètres $[n, k, d]_q$ est bornée par l'expression suivante :

$$d \leq n - k + 1$$

Démonstration. Un code linéaire de paramètres $[n, k, d]_q$ possède une matrice de parité \mathbf{H} avec $n - k$ lignes linéairement indépendantes. Le rang d'une matrice égale le rang de ses colonnes, \mathbf{H} est donc de rang $n - k$. Tout ensemble de $n - k + 1$ colonnes vérifie donc une relation de dépendance linéaire. Par Théorème 8, la distance minimale ne peut excéder $n - k + 1$. \square

Les codes linéaires optimaux pour la distance de Hamming sont les codes MDS, pierre angulaire de ces travaux, définis dans le cas des codes linéaires comme suit :

Définition 19. *Un code linéaire dont les paramètres $[n, k, d]_q$ vérifient le cas d'égalité de la borne de Singleton $d = n - k + 1$ est appelé code Maximum Distance Separable (MDS).*

La proposition suivante témoigne des relations entre les codes MDS et leurs codes duaux :

Proposition 6 ([MS77], Chapitre 11, Théorème 2). *Soit \mathcal{C} un code MDS $[n, k, n - k + 1]$. Alors son code dual \mathcal{C}^\perp est un code $[n, n - k, k + 1]$, également MDS.*

Démonstration. Soit \mathbf{H} une matrice de parité de \mathcal{C} , par définition matrice génératrice de \mathcal{C}^\perp . Notons $\mathbf{H}_1, \dots, \mathbf{H}_n$ ses colonnes.

Par l'absurde, supposons qu'il existe \mathbf{m} tel que $\mathbf{c} = \mathbf{m}\mathbf{H} \in \mathcal{C}^\perp$ de poids inférieur ou égal à k . \mathbf{c} possède donc des coefficients nuls en plus de $n - k$ positions, indicées par i_1, \dots, i_{n-k} . Notons alors $\tilde{\mathbf{H}}$ la matrice $[\mathbf{H}_{i_1}, \dots, \mathbf{H}_{i_{n-k}}]$, sous-matrice de \mathbf{H} non inversible. Le rang des lignes d'une matrice égale le rang de ses colonnes, il existe donc $n - k$ colonnes de $\tilde{\mathbf{H}}$ qui vérifient une relation de dépendance linéaire. Absurde.

Le poids minimal d'un mot de code de \mathcal{C}^\perp est donc supérieur à k . Par Théorème 9, sa distance minimale vaut $k + 1$. \square

Les matrices génératrice et de parité associées à des codes MDS vérifient alors certaines propriétés sur lesquelles se fondent la détermination du caractère MDS d'un code arbitraire dont les preuves sont immédiates à partir de la Proposition 6 et du Théorème 8 :

Proposition 7 ([MS77], Chapitre 11, Corollaire 3). *Les propositions suivantes sont équivalentes :*

- \mathcal{C} est un code MDS.
- Tout ensemble de $n - k$ colonnes d'une matrice de parité \mathbf{H} de \mathcal{C} forme une famille de vecteurs linéairement indépendants.
- Tout ensemble de k colonnes d'une matrice génératrice \mathbf{G} de \mathcal{C} forme une famille de vecteurs linéairement indépendants.

Des arguments de réductions linéaires permettent d'assurer pour tout code linéaire l'existence d'une matrice génératrice dite sous forme systématique :

Définition 20. *Une matrice génératrice sous forme systématique d'un code linéaire \mathcal{C} est une matrice du type $\mathbf{G}_{\text{sys}} = (\mathbf{I}_k \mid \mathbf{M})$. \mathbf{M} est alors appelée une matrice de redondance de \mathcal{C} .*

Cette définition fait le lien avec les problématiques de nombre de branchements des couches de diffusion linéaire. Un mot de code de \mathcal{C} s'écrit alors en effet $\mathbf{c} = (\mathbf{m}, \mathbf{m}\mathbf{M})$. La notion de distance minimale du code rejoint alors la notion de nombre de branchements associée à une couche de diffusion linéaire représentée par une matrice \mathbf{M} .

On conclut l'exposition des principaux résultats sur les codes par la caractérisation des isométries de la distance de Hamming. Celle-ci, initiée par MacWilliams, est généralisée dans la thèse de Sehrii Dyshko [Dys16].

Théorème 10 (Théorème d'Extension de MacWilliams, [Mac62]). *Soit \mathcal{C} un code linéaire de paramètres $[n, k, d]_q$. Toutes les isométries \mathbb{F}_q -linéaire pour la distance de Hamming se décomposent*

en la composition d'une permutation π des coefficients et d'une multiplication scalaire de sorte que :

$$\forall a \in \mathbb{F}_q^n, f((a_1, \dots, a_n)) = (c_1 a_{\pi(1)}, \dots, c_n a_{\pi(n)})$$

Ce résultat se généralise à toutes les isométries pour la distance de Hamming avec la notion d'application monomiale dont la définition suit :

Définition 21. Une application $h : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ est dite monomiale si et seulement si il existe une permutation $\pi \in \mathcal{S}_n$ et des automorphismes g_1, \dots, g_n tels que :

$$\forall a \in \mathbb{F}_q^n, f((a_1, \dots, a_n)) = (g_1(a_{\pi(1)}), \dots, g_n(a_{\pi(n)}))$$

Le Théorème d'Extension de MacWilliams admet alors la généralisation de Bonneau, formulée comme suit :

Théorème 11 ([Bon84]). Une application $h : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ est une isométrie de Hamming si et seulement si h est une application monomiale.

Remarque 3. L'intérêt de connaître ces isométries se fonde sur la réduction des coûts de recherche exhaustive qu'elles autorisent, lorsqu'on cherche à construire des matrices MDS.

Les codes MDS n'existent pas pour n'importe quels paramètres. Une conjecture conditionne cette existence de la façon suivante :

Conjecture 1. Soit $k \leq q$. Soit \mathcal{C} un code linéaire MDS $[n, k]_q$. Alors, ses paramètres vérifient :

$$\begin{aligned} n &\leq q + 2 && \text{si } q = 2^m \text{ et } k \in \{3, q - 1\} \\ n &\leq q + 1 && \text{sinon} \end{aligned}$$

Les premiers résultats sur cette conjecture sont dus à Segre [Seg55b ; Seg55a]. Un regain d'intérêt pour cette question à la fin du vingtième siècle a amené la preuve par Siméon Ball de la conjecture dans le cas des corps premiers [Bal12], puis dans le cas des extensions de degré 2 des corps premiers [BJ12]. Cette conjecture est à l'intersection d'un nombre étonnant de problèmes géométriques et de problèmes combinatoires parmi lesquels on se plaît à citer les carrés Latins. La simplicité de sa formulation en fait un défi lancé à qui la rencontre.

3.2.2 Matrices MDS

Les codes MDS possèdent une distance minimale optimale pour un paramétrage donné. Cette propriété se traduit aux matrices de redondance de ces codes, matrices des couches de diffusion linéaires qu'on cherche à construire. La définition ci-dessous réalise ladite traduction :

Définition 22. Une matrice $\mathbf{M} \in \mathbb{F}_q^{k \times (n-k)}$ est dite Maximum Distance Separable (MDS) lorsque le code \mathcal{C} associé à la matrice génératrice $\mathbf{G} = (\mathbf{I}_k | \mathbf{M})$ est MDS.

Les codes MDS sont des objets très réguliers. À tel point qu'on trouve une dénomination qui le laisse entrevoir pour les matrices MDS : matrices super régulières. Cette définition amène à une caractérisation des matrices MDS indépendante du code sous-jacent, explicitée dans la proposition suivante :

Proposition 8 ([MS77], Chapitre 11, Théorème 8). *Une matrice \mathbf{M} est MDS si et seulement si tous ses mineurs sont non nuls.*

Démonstration. Soit $\mathbf{M} \in \mathbb{F}_q^{k \times (n-k)}$ et \mathcal{C} le code engendré par $(\mathbf{I}_k | \mathbf{M})$.

\mathbf{M} n'est pas MDS

\Leftrightarrow Il existe un ensemble de k colonnes de $(\mathbf{I}_k | \mathbf{M})$ qui forme une matrice non inversible

\Leftrightarrow Il existe un ensemble de $u \leq k$ colonnes de \mathbf{I}_k et $k - u$ colonnes de \mathbf{M} qui forme une matrice non inversible

\Leftrightarrow Il existe un ensemble de $u \leq k$ colonnes de \mathbf{I}_k et $k - u$ colonnes de \mathbf{M} , tel que, à permutation des lignes près et réduction Gaussienne, on ait une matrice non inversible de la forme :

$$\left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & \\ 0 & 1 & & 0 & 0 \\ & & \ddots & & \\ 0 & 0 & 0 & 1 & \\ \hline & & & 0 & \mathbf{M}' \end{array} \right)$$

où \mathbf{M}' correspond à une sous-matrice composée de $k - u$ colonnes de \mathbf{M} après réduction Gaussienne en les u positions indexées par les colonnes issues de \mathbf{I}_k .

\Leftrightarrow Il existe une sous-matrice carrée de \mathbf{M} non inversible. \square

La proposition suivante justifie pleinement l'utilisation des matrices MDS pour constituer des couches de diffusion linéaire car elle assure l'optimalité des nombres de branchements linéaires et différentiels. Uniquement intéressé par les matrices inversibles pour assurer la possibilité du déchiffrement, on se focalise désormais sur des matrices carrées :

Proposition 9. *Si matrice \mathbf{M} carrée est MDS, alors \mathbf{M}^{-1} et \mathbf{M}^T sont MDS.*

Démonstration. La définition d'une matrice MDS assure le caractère MDS de \mathbf{M}^{-1} , comme un simple changement de variable le montre :

$$n - k + 1 = \min_{\mathbf{m} \in \mathbb{F}_q^n} (w_H(\mathbf{m}) + w_H(\mathbf{m}\mathbf{M})) = \min_{\mathbf{m} \in \mathbb{F}_q^n} (w_H(\mathbf{m}\mathbf{M}^{-1}) + w_H(\mathbf{m}))$$

Pour assurer le caractère MDS de \mathbf{M}^T , il suffit de vérifier que la matrice \mathbf{M}^T est une matrice de redondance d'une matrice génératrice sous forme systématique du code dual, donc MDS, du code MDS engendré par $(\mathbf{I}_k | \mathbf{M})$ comme le prouve l'équation suivante :

$$(\mathbf{I}_k | \mathbf{M}) \cdot (\mathbf{M}^T | \mathbf{I}_k)^T = \mathbf{0}$$

\square

Remarque 4. *La Proposition 8 suggère un algorithme de détermination du caractère MDS d'une matrice par vérification de l'inversibilité de tous ses mineurs. Il n'existe à ce jour pas de méthode générique plus efficace. La complexité de cette vérification est exponentielle et devient rapidement, même pour des tailles de matrice crédibles, 32×32 , irréalisable. Pour ces tailles de paramètres, les constructions directe de matrices MDS sont incontournables.*

Parmi les construction directes de matrices MDS, certaines, associées à des critères algébriques, sont présentées dans le corps de cette thèse. D'autres sont plus *ad hoc* et sont présentées maintenant. Une construction directe de matrices MDS à partir de matrices de Cauchy généralisées a été proposée par Roth et Lempel dans [RL89]. Des constructions directes de matrices MDS à partir de configurations de Hankel sont aussi dues à Roth et Seroussi dans [RS85] et indépendamment à Aidinyan dans [Aid86] mais s'intègrent dans la construction suivante :

Construction directe de matrices MDS à partir de matrices de Cauchy généralisées

Cette construction est remarquable parmi les constructions directes de matrices MDS car elle ne repose pas sur la théorie des codes mais utilise directement la caractérisation sur les mineurs.

Définition 23. Une matrice de Cauchy de taille $k \times k$ est une matrice construite à partir de deux ensembles d'éléments disjoints $\{a_1, \dots, a_k\}$ et $\{b_1, \dots, b_k\}$ selon le terme générique :

$$\mathbf{C}_{i,j} = \left(\frac{1}{a_i + b_j}\right)_{i,j \leq k}$$

Cette matrice est une structure classique d'algèbre linéaire. A l'instar des matrices de Vandermonde, on dispose d'une formule, obtenue par récurrence, pour calculer son déterminant.

Proposition 10 ([RS85]). Le déterminant d'une matrice de Cauchy carrée est donné par la formule :

$$\det(\mathbf{C}) = \frac{\prod_{1 \leq i < j \leq k} (a_j - a_i)(b_j - b_i)}{\prod_{1 \leq i, j \leq k} (a_i + b_j)}$$

Une nouvelle fois, à l'instar des matrices de Vandermonde, l'inversibilité d'une matrice de Cauchy est uniquement due au caractère disjoint de ses supports. Par ailleurs, la description des matrices de Cauchy par la forme de son terme général induit trivialement la proposition suivante :

Proposition 11. Toute sous-matrice d'une matrice de Cauchy est une matrice de Cauchy.

La définition des matrices de Cauchy admet plusieurs généralisations. On commence par définir les matrices de Cauchy étendues :

Définition 24. Une matrice \mathbf{C} est une matrice de Cauchy étendue si elle possède une ligne, respectivement une colonne, dont tous les coefficients égalent 1 et dont la suppression engendre une matrice de Cauchy.

La formule du déterminant d'une matrice de Cauchy étendue se déduit alors de la formule du déterminant d'une matrice de Cauchy :

Proposition 12 ([RS85]). Le déterminant d'une matrice de Cauchy étendue \mathbf{C} , avec la ligne de 1 en i -ème position, est donné par la formule suivante :

$$\det(\mathbf{C}) = (-1)^{m+i} \frac{\prod_{1 \leq i < j \leq k} (a_j - a_i)(b_j - b_i)}{\prod_{1 \leq i, j \leq k} (a_i + b_j)}$$

On en déduit immédiatement qu'une matrice de Cauchy étendue est MDS si et seulement si la matrice de Cauchy qu'elle induit est MDS. Cette structure se généralise elle-même pour définir les matrices de Cauchy généralisées :

Définition 25. Une matrice de Cauchy généralisée, respectivement matrice de Cauchy étendue généralisée, est une matrice \mathbf{C}_G qui se décompose comme suit :

$$\mathbf{C}_G = \mathbf{D}_1 \mathbf{C} \mathbf{D}_2$$

avec \mathbf{D}_1 et \mathbf{D}_2 deux matrices diagonales inversibles et \mathbf{C} une matrice de Cauchy, respectivement une matrice de Cauchy étendue.

Si la matrice de Cauchy étendue sous-jacente à une matrice de Cauchy généralisée est MDS, alors la matrice de Cauchy généralisée est MDS. On a alors de la formule du déterminant d'une matrice de Cauchy le théorème suivant :

Théorème 12. En caractéristique 2, une matrice de Cauchy définie par deux ensembles disjoints d'éléments deux à deux distincts est une matrice MDS.

Il est possible de définir des codes de Cauchy sur \mathbb{F}_q , codes qui possèdent une matrice de redondance de Cauchy, à l'exemple des travaux de Dür [Dür87]. Une présentation plus générique en a été proposée par Cary Huffman [Huf98]

3.2.3 Codes et matrices MRD

On a vu l'adéquation de la distance de Hamming avec les problématiques de nombre de branchements des couches de diffusion linéaires. Une autre notion de distance existe cependant, la distance rang.

On considère désormais des codes linéaires sur \mathbb{F}_{q^m} , extension de degré m de \mathbb{F}_q . On peut concevoir \mathbb{F}_{q^m} comme un \mathbb{F}_q -espace vectoriel de dimension m . On peut alors définir une nouvelle notion de poids, le poids rang :

Définition 26. Soit $\mathbf{c} \in \mathbb{F}_{q^m}^n$. Le poids rang $r_g(\mathbf{c})$ de \mathbf{c} est la dimension du \mathbb{F}_q -sous-espace vectoriel engendré par ses coefficients.

Il est alors possible de fonder sur cette base une nouvelle notion de distance, la distance rang :

Définition 27. La relation $d_r(\mathbf{a}, \mathbf{b}) = r_g(\mathbf{a} - \mathbf{b})$ définit une distance sur $\mathbb{F}_{q^m}^n$.

Cette distance possède des propriétés qui la relient à la distance de Hamming :

Proposition 13. Soient $\mathbf{a}, \mathbf{b} \in \mathbb{F}_{q^m}^n$. Alors,

- $d_r(a, b) \leq m$
- $d_r(a, b) \leq d_H(a, b)$

Il existe alors un équivalent de la borne de Singleton pour la distance rang :

Théorème 13. Si \mathcal{C} est un code linéaire de paramètres $[n, k, d_r]$, alors

- Si $n \leq m$, alors $k + d_r \leq n + 1$
- Si $n > m$, alors $km + nd_r \leq (m + 1)n$

Le Théorème 13 suggère alors la définition de codes optimaux pour la distance rang :

Définition 28. *Un code linéaire dont les paramètres $[n, k, d_r]$ vérifient un des cas d'égalité de la borne du Théorème 13 est appelé Maximum Rank Distance (MRD).*

Le résultat suivant est alors une conséquence immédiate de la Proposition 13

Corollaire 1. *Lorsque $n \leq m$, un code MRD est MDS.*

On peut alors transcrire cette notion aux matrices :

Définition 29. *Une matrice $\mathbf{M} \in \mathbb{F}_{q^m}^{k \times (n-k)}$ est dite Maximum Rank Distance (MRD), lorsque le code \mathcal{C} engendré par la matrice $\mathbf{G} = (\mathbf{I}_k | \mathbf{M})$ est MRD.*

Remarque 5. *Une matrice MRD possède une propriété supplémentaire aux matrices MDS. Le rang des entrées et sorties d'une matrice MRD est borné inférieurement. Toutefois, aucune cryptanalyse à ce jour n'a su tirer parti d'un défaut de diffusion à ce niveau-là.*

Par une preuve analogue à celle déployée pour la distance de Hamming, on démontre la proposition suivante :

Proposition 14. *Si une matrice carrée \mathbf{M} est MRD, alors \mathbf{M}^{-1} et \mathbf{M}^T sont MRD.*

Remarque 6. *La détermination algorithmique du caractère MRD d'une matrice générique est encore plus coûteux que pour le caractère MDS, et présente même pour des petites tailles de paramètres un coût prohibitif.*

A l'instar de ce qui a été fait pour les codes MDS, il est possible de déterminer les transformations linéaires qui sont des isométries de la distance rang, comme le prouve le résultat exposé maintenant dû à Thierry Berger [Ber03].

Définition 30. *Une isométrie pour la distance rang est un automorphisme f , \mathbb{F}_{q^m} -linéaire qui préserve le rang des éléments de $\mathbb{F}_{q^m}^n$:*

$$\forall \mathbf{m} \in \mathbb{F}_{q^m}^n, \quad r_g(\mathbf{m}) = r_g(f(\mathbf{m}))$$

Le théorème suivant caractérise alors les isométries linéaires pour la distance rang :

Théorème 14 ([Ber03], Théorème 1). *Le groupe des isométries linéaires pour la distance rang est engendré par les multiplications scalaires et les applications inversibles $GL(n, \mathbb{F}_q)$.*

Démonstration. Les multiplications scalaires comme les applications inversibles $GL(n, \mathbb{F}_q)$ sont des isométries évidentes pour la distance rang.

Soit maintenant $f \in GL(n, \mathbb{F}_{q^m})$ une transformation \mathbb{F}_{q^m} -linéaire inversible et \mathbf{M} sa matrice dans la base canonique. La i -ème ligne de \mathbf{M} correspond alors à l'image du vecteur canonique e_i par f . Supposons alors que f est une isométrie pour la distance rang. Le rang de chaque ligne de \mathbf{M} doit donc être 1. Sans perte de généralité, quitte à appliquer une multiplication scalaire, on peut supposer que les éléments de la première ligne appartiennent à \mathbb{F}_q . D'après ce qui précède, il existe $\lambda \in \mathbb{F}_{q^m}$ tel que $\lambda f(e_i) \in \mathbb{F}_q^n$.

Posons $c = e_1 + e_i$. Le poids rang de c est 1. Son image par l'isométrie f , $(m_{1,1} + m_{i,1}, \dots, m_{1,n} + m_{i,n})$, possède donc au moins un coefficient non nul, par exemple le premier. Posons alors $\nu = m_{1,1} + m_{i,1} \neq 0$.

Comme le rang de $f(c)$ est 1, il existe s tel que $m_{1,j} + m_{i,j} = s\nu$. On en déduit :

$$m_{1,j} - sm_{1,1} = -m_{i,j} + sm_{i,1}$$

Posons $t = m_{i,j} - sm_{1,j} \in \mathbb{F}_q \cap \lambda \mathbb{F}_q$. Alors soit $\lambda \in \mathbb{F}_q$, ce qui implique que les éléments de la i -ème ligne sont dans \mathbb{F}_q , soit $t = 0$, ce qui implique l'égalité $m_{1,j} = sm_{1,1}$ pour tout j , la i -ème ligne est déduite de la première ligne par multiplication par s . Comme \mathbf{M} est inversible, c'est impossible, ainsi tous les $m_{i,j}$ sont dans \mathbb{F}_q et f est dans $GL(n, \mathbb{F}_q)$. \square

L'introduction de ce paradigme de la distance rang a été motivée par les travaux de Ernst Gabidulin [Gab85] sur la base des résultats de Philippe Delsarte [Del78]. Une structure de codes linéaires a été introduite par ses soins, porte son nom et se révèle optimale pour la distance rang.

Construction directe de matrices MRD : les codes de Gabidulin

Définition 31. Soient $\lambda_1, \dots, \lambda_n$ n éléments formant une famille \mathbb{F}_q -libre de \mathbb{F}_{q^m} avec $n \leq m$. Soit θ un élément générateur de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$. Le code de Gabidulin \mathcal{G} de support $\{\lambda_1, \dots, \lambda_n\}$ et dimension k est le code engendré par la matrice :

$$\mathbf{G} = \begin{pmatrix} \lambda_1^{[0]} & \lambda_2^{[0]} & \dots & \lambda_n^{[0]} \\ \lambda_1^{[1]} & \lambda_2^{[1]} & \dots & \lambda_n^{[1]} \\ \vdots & \ddots & \ddots & \vdots \\ \lambda_1^{[k-1]} & \lambda_2^{[k-1]} & \dots & \lambda_n^{[k-1]} \end{pmatrix}$$

Le principal résultat sur ces codes est leur optimalité pour la distance rang, et incidemment pour la distance de Hamming.

Proposition 15. Les codes de Gabidulin sont MRD.

Démonstration. Par l'absurde, supposons qu'un code de Gabidulin ne soit pas MRD. Alors, il existe une combinaison linéaire non triviale des lignes dont les coefficients forment un espace vectoriel de \mathbb{F}_{q^m} de dimension inférieure à $n - k + 1$. Une combinaison linéaire des lignes correspond à un vecteur dont les coefficients égalent l'évaluation d'un θ -polynôme de degré inférieur ou égal à $k - 1$ sur les λ_i . L'image d'un espace de dimension n par l'évaluation du θ -polynôme est donc de dimension inférieure à $n - k + 1$. Le noyau de l'évaluation du θ -polynôme est donc de dimension supérieure ou égale à k en tant que \mathbb{F}_q -espace vectoriel. Soient q^a et \mathbb{F}_{q^s} issus du Théorème 6. Une relation de dépendance linéaire sur \mathbb{F}_{q^a} relie alors les k λ_i en lesquels le θ -polynôme s'annule, considérons même la relation qui relie le nombre minimal ℓ de tels λ_i :

$$\sum a_i \lambda_i = 0 = \sum a_i^{q^m} \lambda_i$$

Par minimalité de ℓ , on sait que $a_i^{q^m} = a_i$ pour tout $i \leq \ell$ et $a_i \in \mathbb{F}_{q^m}$. Comme θ est générateur de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$, a_i appartient à \mathbb{F}_q . On a donc une relation \mathbb{F}_q -linéaire qui relie les λ_i . Absurde. \square

3.3 Implémentations matérielles légères

Lorsque les environnements sur lesquels les modules cryptographiques sont déployés ne sont soumis à aucune contrainte particulière d'implémentation, les k^2 multiplications dans le corps de Galois qu'une matrice MDS implique sont implémentés. Certains environnements matériels comme les systèmes RFID ou les réseaux de capteurs disposent cependant de ressources comptées. Obtenir une estimation des coûts fine pour une pluralité de plateformes matérielles est une gageure. Il est plus illusoire encore d'obtenir une architecture qui minimise les coûts d'implémentation matérielle et logicielle. Sim, Khoo, Oggier et Peyrin se sont penchés sur ces questions [Sim+15] et sont parvenus à la conclusion que le nombre de portes XOR nécessaires pour les différentes opérations du chiffrement, en anglais *XOR count* et *compte de XOR* dans le reste de cette thèse, est une mesure de comparaison des coûts d'implémentation matérielle des couches linéaires légitime. Cette métrique permet d'effectuer des comparaisons *a priori* des implémentations et laisse les problématiques de minimisation des coûts des implémentations matérielles aux électroniciens. Les dix dernières années ont vu fleurir de nombreuses primitives symétriques qui s'orientent vers ces conceptions légères parmi lesquelles on peut citer la famille de fonctions de hachage PHOTON [GPP11]. Les matrices récursives utilisées dans cette dernière et les matrices circulantes utilisées dans l'AES ont notamment été explorées du fait qu'elles admettent des implémentations pour lesquelles la réalisation du produit matrice-vecteur demande seulement k multiplications dans le corps de Galois. L'involutivité est une autre propriété qui connaît un essor remarquable dans les algorithmes les plus récents. Cet esprit de conception se trouve poussé à son paroxysme par exemple dans le schéma de chiffrement par bloc PRINCE [Bor+12] qui utilise une fonction de tour complètement composée d'opérations involutives.

Le compte de XOR, mesure heuristique des coûts d'implémentation matérielle, va maintenant être détaillé. Les matrices récursives, circulantes et involutives seront alors présentées de manière à exhiber les réductions de coûts que leurs structures impliquent. Les matrices de Hadamard-Cauchy seront alors introduites comme construction directe de matrice MDS à la fois involutive et bénéficiant d'une implémentation sérielle.

3.3.1 Mesure heuristique des coûts d'implémentation matérielle

C'est pour comparer les différentes architectures de chiffrements par bloc que Khoongming Khoo, Thomas Peyrin, Axel Poschmann et Huihui Yap dans [Kho+14] ont proposé une métrique de comparaison qui combine les propriétés de sécurité fournies par les structures cryptographiques et les composantes avec leurs propriétés d'implémentation qu'ils nomment Figure Of Adversarial Merit (FOAM). Ce travail est précisé pour l'étude des coûts d'implémentation des couches linéaires dans [Sim+15]. Ces problématiques d'implémentations sont restreintes aux corps de caractéristique 2, seule caractéristique d'usage en cryptographie symétrique. Le compte de XOR, nombre de portes XOR nécessaires pour l'implémentation des différentes opérations arithmétiques du chiffrement, se révèle fortement corrélé avec la taille des surfaces matérielles nécessaires [Kho+14]. L'heuristique qui consiste à minimiser le poids de Hamming des coefficients multiplicatifs dans le corps de Galois mais qui néglige certains effets de réductions est alors abandonnée au profit des coefficients multiplicatifs qui possèdent le plus faible compte de XOR.

Définition 32. *Le compte de XOR d'un élément $\alpha \in \mathbb{F}_{2^m}$, noté $\chi_{\mathcal{B}}(\alpha)$ est le nombre de portes XOR requises pour implémenter la multiplication de α avec un élément arbitraire $\beta \in \mathbb{F}_{2^m}$.*

Le compte de XOR d'un élément, d'une matrice, dépend de la représentation du corps choisie, identifiée par le symbole \mathcal{B} dans la précédente définition.

Exemple 2. *Considérons \mathbb{F}_{2^4} engendré par le polynôme $T^4 + T + 1$. Considérons α racine de ce polynôme, et la multiplication par $\alpha + 1 = \alpha^4$ sa représentation binaire est $[1, 1, 0, 0]$. Soit alors β arbitraire dans \mathbb{F}_{2^4} de représentation binaire $[\beta_1, \beta_2, \beta_3, \beta_4]$. La multiplication résulte alors en*

$$[\beta_1, \beta_2, \beta_3, \beta_4] \oplus [\beta_4, \beta_1 \oplus \beta_4, \beta_2, \beta_3] = [\beta_1 \oplus \beta_4, \beta_1 \oplus \beta_2 \oplus \beta_4, \beta_3 \oplus \beta_2, \beta_4 \oplus \beta_3]$$

Ceci correspond à un compte de XOR de 5.

On en déduit le compte de XOR associé à l'implémentation d'une ligne d'une matrice. Soit $\mathbf{M} \in \mathbb{F}_{2^m}^{k \times k}$ une matrice dont on note la ligne i par \mathbf{M}_i . Le compte de XOR d'une ligne peut alors s'écrire :

$$\chi_{\mathcal{B}}(\mathbf{M}_i) = \sum \chi_{\mathcal{B}}(m_{i,j}) + (r_i - 1) \cdot m$$

r_i correspond ici au nombre de coefficients non nuls de \mathbf{M}_i .

Exemple 3. *Soit \mathbb{F}_{2^4} engendré par $T^4 + T + 1$. Considérons la matrice $\mathbf{C} \in \mathbb{F}_{2^4}^{4 \times 4}$, matrice circulante engendrée par la ligne $[1, 1, \alpha, \alpha^4]$, identification grossière de la matrice de AES dans \mathbb{F}_{2^4} . Le lecteur s'amusera à constater qu'elle est MDS :*

$$\mathbf{C} = \begin{pmatrix} 1 & 1 & \alpha & \alpha^4 \\ \alpha^4 & 1 & 1 & \alpha \\ \alpha & \alpha^4 & 1 & 1 \\ 1 & \alpha & \alpha^4 & 1 \end{pmatrix}$$

Son compte de XOR est donc $0 + 0 + 1 + 5 + 3 \cdot 4 = 6 + 3 \cdot 4$. Pour le reste de cette thèse, on omettra les termes du type $3 \cdot 4$, génériques pour les matrices MDS selon un jeu de paramètres.

Une même application linéaire peut donc se révéler plus ou moins coûteuse selon la représentation du corps. Toute démarche de minimisation des coûts impose le calcul du compte de XOR dans toutes les représentations possibles. Toutefois, certaines propriétés du compte de XOR restent indépendantes du choix de la représentation. Le théorème suivant donne la formule du compte de XOR total, la somme des comptes de XOR de tous les éléments :

Théorème 15 ([Sim+15], Théorème 1). *Soit $m \geq 2$. Le compte de XOR total pour un corps \mathbb{F}_{2^m} est constant :*

$$\sum_{m \in \mathbb{F}_{2^m}} \chi_{\mathcal{B}}(m) = m \sum_{i=2}^m 2^{i-2}(i-1)$$

Exemple 4. *Le compte de XOR total du corps \mathbb{F}_{2^4} est 68.*

Il n'existe pas de représentation de corps *a priori* moins coûteuse qu'une autre. La distribution des comptes de XOR selon les éléments varie cependant. La recommandation heuristique proposée dans [Kho+14] est de choisir le polynôme irréductible dont la représentation induite présente la plus grande déviation standard pour les comptes de XOR. Cette remarque est particulièrement pertinente pour les recherches de matrices MDS qui ne se fondent pas sur des constructions directes. L'heuristique qui présidait auparavant s'attachait à considérer des polynômes irréductibles avec un faible poids de Hamming, avec l'idée que des réductions modulaires seraient engendrées par les multiplications. Sur les petites extensions de corps, ils constatent que les fortes déviations sont souvent liées aux polynômes irréductibles de faible poids de Hamming. Toutefois, ils conjecturent que cette légimité décroît avec la croissance des degrés des extensions de corps.

Il est possible dans le cas de recherches exhaustives de réduire le nombre de représentations de corps à considérer, comme le suggère le Théorème 16. Un isomorphisme préservant l'ordre du compte de XOR assure que toute matrice MDS est envoyée sur une matrice MDS avec le même compte de XOR. On peut donc presque réduire d'un facteur 2 la recherche exhaustive des matrices au plus faible compte de XOR.

Théorème 16 ([Sim+15], Théorème 2). *Il existe une application isomorphe qui envoie un élément primitif de $\mathbb{F}_2[X]/(p(X))$ vers un élément primitif de $\mathbb{F}_2[X]/(p^*(X))$ où les comptes de XOR de α^i et β^i sont égaux pour tout $i \in \{1, \dots, 2^r - 1\}$.*

Démonstration. On a deux identifications nées des isomorphismes canoniques entre $\mathbb{F}_2[X]/(p(X))$ et $\mathbb{F}_2(\alpha)$ où α racine de $p(X)$ et entre $\mathbb{F}_2[X]/(p^*(X))$ et $\mathbb{F}_2(\beta)$ où β racine de $p^*(X)$. On définit l'isomorphisme de corps induit par l'identification canonique :

$$\begin{aligned} \varphi : \mathbb{F}_2(\alpha) &\rightarrow \mathbb{F}_2(\beta) \\ \sum a_i \alpha^i &\mapsto \sum a_i \beta^i \end{aligned}$$

Supposons que α est un élément primitif. Alors le compte de XOR d'un élément $\sum a_i \alpha^i$ revient au calcul de $(\sum a_i \alpha^i) \alpha^j$ pour tout $j \in \{1, \dots, 2^m - 1\}$. La distribution du compte de XOR est la même que calculant $(\sum a_i \alpha^{-i}) \alpha^j$.

Les calculs de α^i et α^{-i} nécessitent le même nombre de XOR puisque $p(\alpha)$ et $\alpha^{-r} p(\alpha)$ ont le même nombre de coefficients non nuls. Ainsi,

$$\varphi((\sum a_i \alpha^{-i}) \alpha^j) = (\sum a_i \varphi(\alpha^{-i}) \varphi(\alpha^j)) = (\sum a_i \beta^i) \beta^{-j}$$

Le compte de XOR d'un élément α^j est donc le même que celui d'un élément β^{-j} . Enfin, lorsque α n'est pas primitif, on reprend la même démonstration à partir de l'expression de α' primitif dans la base $\{1, \alpha, \dots, \alpha^{r-1}\}$. □

Exemple 5. *Considérons \mathbb{F}_{2^4} engendré par le polynôme $p(T) = T^4 + T + 1$ et α une racine de $p(T)$. Son polynôme réciproque est $p^*(T) = T^4 + T^3 + 1$ et on nomme β une de ses racines. L'isomorphisme induit est l'application qui envoie $\alpha \mapsto \beta^2 + 1$.*

Remarque 7. *Les coûts logiciels ne sont pas sensiblement impactés par ces choix de conceptions qui minimisent le compte de XOR. En effet, la mémoire n'est pas un facteur limitant et les implémentations en tables pré-calculées incorporent directement les multiplications par des éléments de corps dans les valeurs stockées.*

3.3.2 Trois tentations de réduction des coûts

L'involutivité est une propriété qui se traduit par une réduction des coûts d'implémentation matérielle.

Définition 33. *Une matrice $\mathbf{M} \in \mathbb{F}_q^{k \times k}$ est dite involutive lorsqu'elle vérifie l'équation :*

$$\mathbf{M}^2 = \mathbf{I}_k$$

De cette définition on apprend qu'une matrice involutive est son propre inverse. L'involutivité assure donc que les opérations de chiffrement et de déchiffrement peuvent mutualiser certains circuits logiques et jusqu'à diviser par deux les coûts d'implémentation de la couche linéaire.

Introduisons les matrices compagnons, à l'origine de la notion de récursivité :

Définition 34. Soit $g(X) = X^k + \sum_{i=0}^{k-1} g_i X^i$ un polynôme unitaire de degré k dans $\mathbb{F}_q[X]$. La matrice compagnon associée au polynôme $g(X)$, \mathbf{C}_g , est définie par :

$$\mathbf{C}_g = \begin{pmatrix} 0 & 1 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & 1 \\ -g_0 & -g_1 & \dots & -g_{k-1} \end{pmatrix}$$

On parle réciproquement de polynôme associé à une matrice compagnon.

Les matrices récursives sont construites à partir des matrices compagnons :

Définition 35. Soit \mathbf{M} une matrice de $\mathbf{F}_q^{k \times k}$. On dit que \mathbf{M} est récursive s'il existe $g(X) \in \mathbb{F}_q[X]$ unitaire de degré k tel que :

$$\mathbf{M} = \mathbf{C}_g^k$$

Outre la famille de fonctions de hachage PHOTON, les matrices récursives sont également utilisées dans le schéma de chiffrement par bloc léger LED [Guo+11] car elles bénéficient d'une implémentation matérielle naturelle peu coûteuse : le registre à décalage à rétroaction linéaire, en anglais *Linear Feedback Shift Register* (LFSR), dont une représentation est donnée par la Figure 3.2. Seul un nombre linéaire de multiplications dans le corps de Galois doit être implémenté et non un nombre quadratique comme c'est le cas pour une matrice sans structure particulière. Cette architecture est très compacte pour les implémentations matérielles car elle réutilise la mémoire existante avec un stockage temporaire presque nul ni aucun contrôle logique additionnel. Ce bénéfice s'accompagne cependant d'une contrepartie. L'utilisation d'une telle architecture induit une latence : k exécutions de la routine du LFSR sont nécessaires pour réaliser le produit matrice-vecteur avec une matrice de taille $k \times k$ quand un produit direct peut se faire en un temps d'horloge.

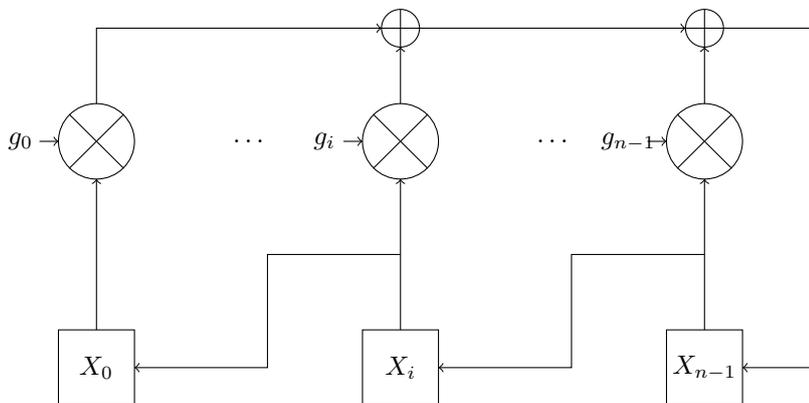


FIGURE 3.2 – Registre à décalage à rétroaction linéaire.

Remarque 8. On cherche à construire des matrices récursives pour réaliser une couche de diffusion linéaire d'une primitive de chiffrement symétrique. Dans ce contexte, il est essentiel que les opérations soient inversibles pour assurer le déchiffrement. L'inversibilité d'une matrice récursive

est inféodée à l'inversibilité de la matrice compagnon qui l'engendre elle même équivalente à $g_0 \neq 0$. Pour la suite, sauf à avoir l'hypothèse contraire explicite, on supposera que les polynômes qui engendrent des matrices récursives ne s'annulent pas en 0.

Les matrices compagnon sont très structurées. Certaines admettent une décomposition comme conjugaison d'une matrice diagonale par une matrice de Vandermonde comme le stipule le Théorème 17.

Théorème 17. Soit $g(X)$ un polynôme unitaire de degré k à racines simples $\{\lambda_1, \dots, \lambda_k\}$ non nulles. La matrice compagnon associée à $g(X)$, \mathbf{C}_g , peut alors s'écrire :

$$\mathbf{C}_g = \mathbf{VDV}^{-1} \quad (3.2)$$

où

$$\mathbf{D} = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \lambda_k \end{pmatrix} \quad \text{et} \quad \mathbf{V} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \lambda_1 & \lambda_2 & \dots & \lambda_k \\ \vdots & \ddots & \ddots & \vdots \\ \lambda_1^{k-1} & \lambda_2^{k-1} & \dots & \lambda_k^{k-1} \end{pmatrix}$$

Démonstration. L'équation (3.2) se réécrit :

$$\mathbf{C}_g \mathbf{V} = \mathbf{VD}$$

Les vecteurs suivants :

$$\begin{pmatrix} 1 \\ \lambda_1 \\ \vdots \\ \lambda_1^{k-1} \end{pmatrix}, \begin{pmatrix} 1 \\ \lambda_2 \\ \vdots \\ \lambda_2^{k-1} \end{pmatrix}, \dots, \begin{pmatrix} 1 \\ \lambda_k \\ \vdots \\ \lambda_k^{k-1} \end{pmatrix}$$

sont clairement des vecteurs propres associés aux valeurs propres λ_i , racines de $g(X)$. Comme $g(X)$ possède k racines distinctes non nulles, on sait que \mathbf{V} , concaténation des k vecteurs propres, est une matrice de Vandermonde inversible. \square

L'association entre polynômes et matrices compagnon s'étend en la proposition suivante :

Proposition 16. Soit $g(X) = X^k + \sum_{i=0}^{k-1} g_i X^i \in \mathbb{F}_q[X]$ un polynôme unitaire de degré k et \mathbf{C}_g la matrice compagnon associée à $g(X)$. Alors, \mathbf{C}_g est la matrice, dans la base canonique de $\mathbb{F}_q[X]/(g(X))$, de l'application :

$$\begin{aligned} \varphi : \mathbb{F}_q[X]/(g(X)) &\rightarrow \mathbb{F}_q[X]/(g(X)) \\ f(X) &\mapsto Xf(X) \end{aligned}$$

Démonstration. Par linéarité, il suffit de le vérifier pour les éléments $\{e_0, \dots, e_{k-1}\}$ de la base canonique de $\mathbb{F}_q[X]/(g(X))$:

$$\forall j \in \{0, \dots, k-2\}, \begin{cases} X^j X = X^{j+1} \\ e_j \mathbf{C}_g = e_{j+1} \end{cases}$$

Enfin,

$$\begin{cases} e_{k-1} \mathbf{C}_g = (-g_0, \dots, -g_{k-1}) \\ X^{k-1} X = \sum_{i=0}^{k-1} -g_i X^i \pmod{g} \end{cases}$$

□

Les matrices circulantes, comme les matrices récursives, admettent une implémentation sérialisée qui ne nécessite l'implémentation que de k multiplications dans le corps de Galois. Elles sont utilisées dans de très nombreuses primitives symétriques, à commencer par l'opération MixColumns de l'AES [DR02]. Leur architecture permet même plusieurs compromis entre surface nécessaire et nombre de temps d'horloge pour réaliser le produit matrice-vecteur suivant le nombre de lignes de la matrice qu'on implémente effectivement.

Définition 36. Une matrice $\mathbf{C} \in \mathbb{F}_q^{k \times k}$ est dite *circulante* lorsque chaque ligne est obtenue par décalage circulaire d'une position vers la droite de la ligne qui la précède, soit lorsque ses coefficients satisfont les relations $\mathbf{C}_{i,j} = \mathbf{C}_{(i-1) \bmod k, (j+1) \bmod k} = c_{i-j \bmod k}$:

$$\mathbf{C} = \begin{pmatrix} c_0 & c_1 & \dots & c_{k-1} \\ c_{k-1} & c_0 & \dots & c_{k-2} \\ \vdots & \ddots & \ddots & \vdots \\ c_1 & c_2 & \dots & c_0 \end{pmatrix}$$

Il existe une association naturelle entre polynômes unitaires de degré k et matrices circulantes.

Définition 37. Soit $h(X) = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_q[X]$ un polynôme unitaire de degré k . La matrice circulante associée à $h(X)$, \mathbf{C}_h est la matrice définie par :

$$\mathbf{C}_h = \begin{pmatrix} h_0 & h_1 & \dots & h_{k-1} \\ h_{k-1} & h_0 & \dots & h_{k-2} \\ \vdots & \ddots & \ddots & \vdots \\ h_1 & \dots & h_{k-1} & h_0 \end{pmatrix}$$

Pour toute matrice circulante, on définit réciproquement le polynôme unitaire qui lui est associé par la première ligne de la matrice suivant l'identification réalisée dans la Définition 37. Cette association tire son essence de la proposition suivante :

Proposition 17. Soit $h(X) = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_q[X]$ un polynôme unitaire de degré k et \mathbf{C}_h la matrice circulante associée à $h(X)$. Alors, \mathbf{C}_h est la matrice, dans la base canonique de $\mathbb{F}_q[X]/(X^k - 1)$, de l'application :

$$\begin{aligned} \phi : \mathbb{F}_q[X]/(X^k - 1) &\rightarrow \mathbb{F}_q[X]/(X^k - 1) \\ m(X) &\mapsto m(X)h(X) \end{aligned}$$

Démonstration. Par linéarité, il suffit de le vérifier pour les éléments $\{e_0, \dots, e_{k-1}\}$ de la base canonique de $\mathbb{F}_q[X]/(X^k - 1)$:

$$\forall j \in \{0, \dots, k-1\}, \begin{cases} X^j \cdot h(X) = \sum_{i=0}^k X^j h_i X^i = \sum_{i=0}^{k-j-1} h_i X^{i+j} + \sum_{i=1}^j h_{k-j-1+i} X^i \\ e_j \cdot \mathbf{C}_h = (h_{k-j}, \dots, h_{k-1}, h_0, \dots, h_{k-j-1}) \end{cases}$$

□

3.3.3 Matrices de Hadamard-Cauchy

Les matrices de Hadamard constituent une classe de matrices qui bénéficient de propriétés structurelles similaires aux matrices circulantes. Elles ont fait l'objet d'une étude approfondie dans [Sim+15]. Sans association algébrique naturelle, cette structure n'appelle *a priori* pas à une généralisation grâce aux anneaux non commutatifs de polynômes. Les matrices de Hadamard sont définies de façon récursive pour des tailles de matrice de la forme $2^s \times 2^s$.

Définition 38. Une matrice \mathbf{H} de taille $2^s \times 2^s$ est une matrice de Hadamard lorsqu'il existe deux matrices de Hadamard \mathbf{H}_1 et \mathbf{H}_2 de taille $2^{s-1} \times 2^{s-1}$ telles que :

$$\mathbf{H} = \begin{pmatrix} \mathbf{H}_1 & \mathbf{H}_2 \\ \mathbf{H}_2 & \mathbf{H}_1 \end{pmatrix}$$

Exemple 6. Soit \mathbb{F}_{2^4} le corps engendré par $T^4 + T + 1$. Choisissons arbitrairement une première ligne $[1, \alpha, \alpha^2, \alpha^3]$. La matrice de Hadamard définie à partir de cette première ligne est donc la matrice \mathbf{H} :

$$\mathbf{H} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 \\ \alpha & 1 & \alpha^3 & \alpha^2 \\ \alpha^2 & \alpha^3 & 1 & \alpha \\ \alpha^3 & \alpha^2 & \alpha & 1 \end{pmatrix}$$

De la définition découlent quelques propriétés immédiates. Une matrice de Hadamard est clairement symétrique et présente les mêmes nombres de branchements linéaires et différentiels. On parle alors du nombre de branchements d'une matrice de Hadamard. Par structure, sur des corps de caractéristique 2, posant c^2 la somme des carrés des coefficients de la première ligne d'une matrice de Hadamard \mathbf{H} , on vérifie aisément la relation suivante :

$$\mathbf{H}^2 = c^2 \mathbf{I}_{2^s}$$

La détermination des matrices de Hadamard involutives est alors immédiate, ce sont celles dont la somme des carrés des coefficients de la première ligne égale 1.

Deux stratégies de recherche émergent : construire des matrices de Hadamard involutives à faible compte de XOR et espérer en trouver parmi elles certaines MDS, comme réalisé par Gupta et Ray dans [CG13] ou construire des matrices de Hadamard-Cauchy, MDS et involutives, et d'espérer trouver parmi elles certaines à faible compte de XOR, comme réalisé dans [Sim+15]. Pour les petites dimensions, la première stratégie semble plus adaptée. Pour les grandes dimensions, tester le caractère MDS est prohibitif et la construction directe est plus intéressante. La recherche de matrices de Hadamard MDS non involutives présente les mêmes limites, ce faisant, on perd en revanche toute maîtrise sur les coûts d'implémentation de la matrice inverse. L'utilisation systématique de cette structure est entravée par la restriction sur les tailles de matrices qu'elle implique, des tailles puissances de 2. Pour réduire la complexité des recherches exhaustives, il est possible d'introduire des relations d'équivalence sur les matrices de Hadamard et même sur les matrices de Hadamard involutives qui préservent les nombres de branchement. Une recherche exhaustive peut donc se cantonner à l'énumération d'un représentant par classe.

Chapitre 4

Construction de matrices récurrentes MDS et généralisations

Sommaire

4.1	Matrices récurrentes	42
4.1.1	Matrices récurrentes et codes cycliques	42
4.1.2	Constructions directes de matrices récurrentes MDS	45
4.1.3	Recherches exhaustives de matrices récurrentes MDS	52
4.2	Tordre la récursivité	53
4.2.1	Matrices θ -récurrentes et codes θ -cycliques	54
4.2.2	Construction directe de matrices θ -récurrentes MDS	57
4.2.3	Recherche exhaustive	63
4.3	Relâcher les contraintes	64
4.3.1	Matrices MDS involutives à partir de matrices de Vandermonde	64
4.3.2	Explorer la récursivité	67
4.3.3	Couches de diffusion linéaire parfaites récurrentes	68

La détermination des polynômes qui engendrent des matrices récurrentes MDS n'est pas inféodée à une recherche exhaustive. Plusieurs constructions directes permettent d'obtenir de telles matrices comme celle de Daniel Augot et Matthieu Finiasz [AF14] ou celle de Thierry Berger [Ber13]. Gupta, Pandey et Venkateswarlu [GPV15a] ont tiré l'essence de ces constructions par des caractérisations algébriques des polynômes associés à ces LFSRs susceptibles d'engendrer des matrices récurrentes MDS.

Les principaux résultats originaux consistent ici en la transposition de ces résultats aux anneaux de polynômes non commutatifs de Ore. Ils conduisent à la création d'une structure cousine des LFSR et sont l'origine d'une construction directe de matrices MDS qui présente une forme de récursivité et dont l'implémentation est réalisable par un circuit proche d'un LFSR. En base normale, l'application d'un élément de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$ est réalisable par une simple permutation fixe de bits, d'un coût négligeable. Cette dernière remarque est le point de départ des travaux présentés dans cette partie. On calque alors l'association classique entre matrices récurrentes et polynômes pour construire une association entre matrices θ -récurrentes et θ -polynômes, éléments d'anneaux de polynômes non commutatifs de Ore.

Ce chapitre commence par introduire les codes cycliques à partir desquels sont construites les matrices récursives. Les constructions directes de matrices récursives MDS à partir de codes BCH raccourcis et à partir de codes de Gabidulin sont alors présentées. Cette méthode est ensuite reproduite à partir des codes θ -cycliques à partir desquels on peut construire les matrices θ -circulantes. Une construction directe de matrices θ -circulantes MDS à partir de codes de Gabidulin est alors présentée. Ce chapitre se conclut par la présentation des compromis possibles avec les rigidités des codes cycliques pour construire des couches linéaires récursives à faible coût.

4.1 Matrices récursives

Les matrices récursives sont liées à un objet classique de théorie des codes : les codes cycliques. Des constructions de codes cycliques MDS sont connues à partir desquelles on peut construire des matrices récursives MDS. Certaines propriétés comme la non-existence de matrices récursives involutives MDS sur des corps de caractéristique 2, découlent de considérations sur les polynômes.

Cette section débute par la description des liens qui unissent les matrices récursives et les codes cycliques. Deux constructions directes de matrices récursives MDS, à partir de codes BCH raccourcis ou à partir de codes de Gabidulin sont alors présentées. Les résultats des recherches exhaustives de matrices récursives MDS sont exhibés en fin de section et présentent les performances atteignables.

4.1.1 Matrices récursives et codes cycliques

Les matrices récursives dans $\mathbb{F}_q^{k \times k}$ (Définition 35) peuvent toujours être concaténées à la matrice identité, \mathbf{I}_k , pour former une matrice génératrice sous forme systématique d'un code linéaire $[2k, k]$. Déterminer la distance minimale de ce code nécessite cependant le calcul de tous les mineurs de la matrice récursive. Toutefois, il est possible d'associer les matrices récursives aux codes cycliques, comme présenté ci-après. Les codes cycliques sont un sujet classique en théorie des codes [MS77 ; PH98].

Définition 39. Soit \mathcal{C} un code linéaire de paramètres $[n, k]_q$. On dit que \mathcal{C} est cyclique si pour tout vecteur $\mathbf{c} = (c_0, \dots, c_{n-1}) \in \mathcal{C}$, le vecteur obtenu par rotation des coefficients :

$$(\mathbf{c} \ggg 1) = (c_{n-1}, c_0, \dots, c_{n-2})$$

appartient également à \mathcal{C} .

On associe les mots d'un code cyclique de paramètres $[n, k]_q$ aux éléments de $\mathbb{F}_q[X]/(X^n - 1)$:

$$\begin{aligned} \phi : \quad \mathcal{C} &\rightarrow \mathbb{F}_q[X]/(X^n - 1) \\ \mathbf{c} = (c_0, \dots, c_{n-1}) &\mapsto \sum_{i=0}^{n-1} c_i X^i \end{aligned}$$

On parlera alors indifféremment de mots de code pour les éléments de \mathcal{C} ou pour leur image par ϕ . Par abus de notation, on écrira tout aussi indifféremment \mathcal{C} pour lui-même ou pour son image par ϕ .

La rotation des coefficients correspond dans $\mathbb{F}_q[X]/(X^n - 1)$ à la multiplication par X . On a alors la proposition suivante :

Proposition 18. *Un code cyclique est un idéal de $\mathbb{F}_q[X]/(X^n - 1)$.*

Démonstration. Pour tout $\lambda \in \mathbb{F}_q$, pour tout mot de code $c(X)$, par cyclicité, $\lambda \cdot c(X), \dots, \lambda \cdot X^{n-1}c(X)$ sont des mots de code. Par linéarité, toute combinaison linéaire de ces mots de code est également un mot de code. Pour tout polynôme $a(X) \in \mathbb{F}_q[X]/(X^n - 1)$, $a(X) \cdot c(X)$ appartient donc aux mots de code et \mathcal{C} forme un idéal de $\mathbb{F}_q[X]/(X^n - 1)$. \square

Du Théorème 2, on déduit les propriétés suivantes :

Propriétés 1. *Soit \mathcal{C} un code cyclique de paramètres $[n, k]_q$. Alors :*

- \mathcal{C} possède un unique polynôme unitaire de plus petit degré $n - k$, $g(X)$, appelé polynôme générateur de \mathcal{C} .
- Tout mot de code $c(X)$ s'exprime comme un multiple du générateur : $c(X) = m(X)g(X)$, où $m(X)$ est de degré inférieur à k .
- $g(X)$ est un diviseur de $X^n - 1$ dans $\mathbb{F}_q[X]$.

De leur définition et de la Propriété 1, on dérive pour un code cyclique \mathcal{C} de paramètres $[n, k]_q$, à partir de son polynôme générateur $g(X) = \sum_{i=0}^{n-k} g_i X^i$, l'existence d'une matrice génératrice canonique de \mathcal{C} , \mathbf{G} , sous la forme :

$$\mathbf{G} = \begin{pmatrix} g_0 & \cdots & g_{n-k} & 0 & \cdots & 0 \\ 0 & g_0 & \cdots & g_{n-k} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & g_0 & \cdots & g_{n-k} \end{pmatrix} \quad (4.1)$$

On peut également définir une matrice génératrice systématique \mathbf{G}_{syst} de \mathcal{C} . Par définition d'un code cyclique, tout multiple de $g(X)$ est un mot de code et, par linéarité, le polynôme $(f(X) - (f(X) \bmod g(X)))$ appartient également au code, pour tout polynôme $f(X) \in \mathbb{F}_q[X]$ de degré inférieur à n . On utilise cette propriété spécialisée en $f(X) = X^i$, pour tout $i \in [n - k, n - 1]$, pour obtenir la matrice génératrice sous forme systématique canonique de \mathcal{C} dont la matrice de redondance suscite notre intérêt.

$$\mathbf{G}_{syst} = \left(\begin{array}{cc|ccc} -X^{n-k} & \bmod g & 1 & 0 & \cdots & 0 \\ -X^{n-k+1} & \bmod g & 0 & 1 & \ddots & \vdots \\ & \vdots & \vdots & \ddots & \ddots & 0 \\ -X^{n-1} & \bmod g & 0 & \cdots & 0 & 1 \end{array} \right)$$

Dans le cas des codes $[2k, k]$, on nomme \mathbf{R}_g l'inverse additif de la matrice de redondance du code cyclique engendré par un polynôme unitaire de degré k diviseur de $X^n - 1$:

$$\mathbf{R}_g = \begin{pmatrix} X^k & \bmod g \\ X^{k+1} & \bmod g \\ \vdots & \\ X^{2k-1} & \bmod g \end{pmatrix} \quad (4.2)$$

Cette matrice est la clef de voûte des constructions de couches linéaires à partir de LFSR, et se généralise à tout polynôme unitaire de degré k , comme le montre le théorème suivant :

Théorème 18. Soit $g(X)$ un polynôme unitaire de degré k . Alors, \mathbf{R}_g est une matrice réursive engendrée par la matrice compagnon associée à $g(X)$:

$$\mathbf{R}_g = \mathbf{C}_g^k$$

Soit $g(X)$ le polynôme générateur d'un code cyclique MDS de paramètres $[2k, k]$. Alors, \mathbf{R}_g est une matrice réursive MDS et involutive.

$$\mathbf{R}_g^2 = \mathbf{I}_k$$

Démonstration. La preuve du premier point du théorème est une imbrication des deux faits suivants :

- \mathbf{R}_g est la matrice de la multiplication par X^k dans $\mathbb{F}_q[X]/(g(X))$ sur sa base canonique.
- \mathbf{C}_g est la matrice de la multiplication par X dans $\mathbb{F}_q[X]/(g(X))$ sur sa base canonique.

Supposons désormais que $g(X)$ est le polynôme générateur d'un code cyclique MDS $[2k, k]$. Par construction, \mathbf{R}_g est l'inverse additif de la matrice de redondance d'une matrice génératrice sous forme systématique d'un code MDS et est donc MDS. La preuve de son involutivité est une imbrication des deux faits suivants :

- \mathbf{R}_g est la matrice de la multiplication par X^k dans $\mathbb{F}_q[X]/(g(X))$ sur sa base canonique.
- Pour tout polynôme $f(X) \in \mathbb{F}_q[X]/(g(X))$, $X^{2k}f(X) \bmod g = f(X) \bmod g$. Comme $g(X)$ divise $X^{2k} - 1$, la multiplication par X^k dans $\mathbb{F}_q[X]/(g(X))$ est en effet involutive. \square

Remarque 9. La relation $\mathbf{R}_g = \mathbf{C}_g^k$ permet d'implémenter le produit matrice-vecteur par un simple LFSR dont on opère la routine k fois.

La conséquence immédiate du Théorème 18 est la possibilité de construire des matrices rékursives MDS involutives à partir de codes cycliques MDS $[2k, k]$. Un écueil se présente malheureusement en caractéristique 2. Le polynôme générateur d'un code cyclique MDS de paramètres $[2k, k]$ divise en effet $X^{2k} - 1$, ses racines sont donc toutes des racines $2k$ -ième de l'unité. D'après le théorème de Lagrange, l'ordre de ces racines dans un corps de caractéristique 2 ne peut pas être pair, vu que pour tout $s \in \mathbb{N}^*$, le cardinal de $\mathbb{F}_{2^s}^*$ est impair. D'après le théorème de Gauss, toutes ces racines sont donc des racines k -ième de l'unité. On a alors :

$$\mathbf{R}_g = \mathbf{C}_g^k = \mathbf{V}\mathbf{D}^k\mathbf{V}^{-1} = \mathbf{V}\mathbf{I}_k\mathbf{V}^{-1} = \mathbf{I}_k$$

En aucun cas, \mathbf{I}_k ne peut être MDS. Il est donc impossible d'engendrer des codes cycliques MDS de paramètres $[2k, k]$ sur des corps de caractéristique 2.

Les codes cycliques font apparaître via leur matrice génératrice sous forme systématique canonique une matrice de redondance réursive. Il est donc aisé d'obtenir des matrices rékursives MDS à partir de codes cycliques MDS. La construction naturelle de matrices rékursives MDS à partir de codes cycliques MDS de paramètres $[2k, k]$ assure de plus leur involutivité. Rékursivité et involutivité présagent d'implémentations matérielles légères, avec l'exploitation de deux structures qui autorisent une réduction des coûts en termes d'aire. Cependant il n'existe pas de codes cycliques MDS de paramètres $[2k, k]$ en caractéristique 2, seule caractéristique d'usage en cryptographie symétrique. Toutefois, toutes les matrices rékursives de taille $k \times k$ ne sont pas associées à des codes cycliques de paramètres $[2k, k]$. Les constructions directes qui suivent tirent parti de ce dernier point.

4.1.2 Constructions directes de matrices récursives MDS

Il existe deux constructions directes de matrices récursives MDS. La construction qui suit est réalisée à partir de codes BCH raccourcis et est due à Daniel Augot et Matthieu Finiasz [AF14]. La construction présentée ensuite est réalisée à partir de codes de Gabidulin et a été proposée par Thierry Berger [Ber13].

Construction de matrices récursives MDS à partir de codes BCH raccourcis [AF14]

À un code cyclique de paramètres $[2k, k]$ s'associe une matrice récursive. Il est possible de réaliser l'association réciproque d'un code cyclique pour tout polynôme unitaire $g(X)$ tel que $g(0) \neq 0$. Considérons en effet $n \in \mathbb{N}$ minimal tel que $g(X)$ divise $X^n - 1$. Alors, $g(X)$ est le polynôme générateur d'un code cyclique de paramètres $[n, n - k]_q$. Bien qu'il n'existe pas de codes cycliques MDS de paramètres $[2k, k]$ en caractéristique 2, il est possible de construire des matrices récursives MDS en s'intéressant à des codes plus longs.

En toute généralité, pour s'affranchir des problématiques de détermination générique de la distance minimale d'un code linéaire cyclique, la première chose à construire est un code cyclique dont la distance minimale est prescrite. Les codes BCH répondent à cette exigence :

Définition 40. Soit β un élément d'une extension de corps \mathbb{F}_{q^m} de \mathbb{F}_q . Soient ℓ et δ deux entiers naturels. On nomme $g(X)$ le ppcm des polynômes minimaux des racines $\beta^\ell, \dots, \beta^{\ell+\delta-1}$:

$$g(X) = \text{ppcm} (\Pi_{\mathbb{F}_2}(\beta^\ell), \Pi_{\mathbb{F}_2}(\beta^{\ell+1}), \dots, \Pi_{\mathbb{F}_2}(\beta^{\ell+\delta-1}))$$

Le code cyclique engendré par $g(X)$ est appelé un code BCH.

Les paramètres d'un code BCH satisfont une borne appelée sans surprise borne BCH :

Théorème 19 ([MS77], Chapitre 7, Théorème 10). Soit \mathbb{F}_{q^m} la plus petite extension de corps de \mathbb{F}_q contenant une racine primitive n -ième de l'unité, β . Soit $g(X)$ le polynôme unitaire de degré minimal dans $\mathbb{F}_q[X]$ qui annule les δ racines consécutives de la forme $\{\beta^\ell, \dots, \beta^{\ell+\delta-1}\}$.

$$g(X) = \text{ppcm} (\Pi_{\mathbb{F}_2}(\beta^\ell), \Pi_{\mathbb{F}_2}(\beta^{\ell+1}), \dots, \Pi_{\mathbb{F}_2}(\beta^{\ell+\delta-1}))$$

Soit alors \mathcal{C} le code cyclique engendré par $g(X)$:

$$\mathcal{C} = \langle g(X) \rangle$$

\mathcal{C} est un code de paramètres $[n, n - \text{deg}(g), d]_q$ tel que $d \geq \delta + 1$.

Démonstration. Une matrice de parité de \mathcal{C} est donnée par \mathbf{H} :

$$\mathbf{H} = \begin{bmatrix} 1 & \beta^\ell & \beta^{2\ell} & \dots & \beta^{(n-1)\ell} \\ 1 & \beta^{\ell+1} & \beta^{2(\ell+1)} & \dots & \beta^{(n-1)(\ell+1)} \\ \vdots & & \ddots & & \vdots \\ 1 & \beta^{\ell+\delta-1} & \beta^{2(\ell+\delta-1)} & \dots & \beta^{(n-1)(\ell+\delta-1)} \end{bmatrix}$$

Par Théorème 8, la distance minimale de \mathcal{C} égale le plus petit nombre de colonnes de \mathbf{H} vérifiant une relation de dépendance linéaire. Par l'absurde, supposons qu'il existe un mot de code \mathbf{c} de poids $p < \delta + 1$, dont on note les indices des coefficients non nuls i_1, \dots, i_p . Par définition, on a $\mathbf{H}\mathbf{c}^T = \mathbf{0}$, soit :

$$\begin{bmatrix} \beta^{i_1 \ell} & \beta^{i_2 \ell} & \dots & \beta^{i_p \ell} \\ \beta^{i_1(\ell+1)} & \beta^{i_2(\ell+1)} & \dots & \beta^{i_p(\ell+1)} \\ \vdots & \ddots & \ddots & \vdots \\ \beta^{i_1(\ell+\delta-1)} & \beta^{i_2(\ell+\delta-1)} & \dots & \beta^{i_p(\ell+\delta-1)} \end{bmatrix} \begin{bmatrix} c_{i_1} \\ c_{i_2} \\ \vdots \\ c_{i_p} \end{bmatrix} = \mathbf{0}$$

L'extraction des p premières lignes de \mathbf{H} restreintes aux colonnes d'indices i_1, \dots, i_p engendre donc une matrice $p \times p$ non inversible. Or, son déterminant s'écrit :

$$\beta^{i_1 \ell + i_2 \ell + \dots + i_p \ell} \cdot \begin{vmatrix} 1 & 1 & \dots & 1 \\ \beta^{i_1} & \beta^{i_2} & \dots & \beta^{i_p} \\ \vdots & \vdots & \ddots & \vdots \\ \beta^{i_1(p-1)} & \beta^{i_2(p-1)} & \dots & \beta^{i_p(p-1)} \end{vmatrix}$$

On reconnaît le déterminant d'une matrice de Vandermonde, nul si et seulement si il existe j_1 et j_2 dans $\{1, \dots, p\}$ tels que $\beta^{i_{j_1}} = \beta^{i_{j_2}}$. Comme ces indices sont compris entre 1 et n et comme β est une racine primitive n -ième de l'unité, les $\beta^{i_1}, \dots, \beta^{i_p}$ sont deux à deux distincts, ce qui signifie que ce déterminant ne peut pas s'annuler. Absurde. \square

Du Théorème 19 émerge une construction de codes cycliques BCH MDS. Construire $g(X)$ de degré exactement $d-1$ assure du fait qu'il possède $d-1$ racines dans \mathbb{F}_{q^m} qu'il ne peut pas avoir d'autres racines. Les puissances successives de β , racines de $g(X)$, doivent donc être conjuguées les unes aux autres.

La construction de codes BCH MDS ne peut pas engendrer de codes cycliques MDS de paramètres $[2k, k]$ en caractéristique 2. Augot et Finiasz ont trouvé un moyen de contourner cette difficulté en construisant des codes aux longueurs et dimensions supérieures qu'ils raccourcissent ensuite [AF14].

Définition 41. Soit \mathcal{C} un code de paramètres $[n, k, d]_q$. Soit $\{i_1, \dots, i_r\}$ un ensemble d'indices inclus dans $\{1, \dots, n\}$. Le code raccourci \mathcal{C}_r de \mathcal{C} en $\{i_1, \dots, i_r\}$ est l'ensemble des mots de \mathcal{C} qui sont nuls en les positions i_1, \dots, i_r dont lesdites coordonnées ont été supprimées. Le code raccourci est donc un code de paramètres $[n', k', d']_q$ avec $n' = n - r$, $k' \geq k - r$ et $d' \geq d$.

La conséquence immédiate de cette définition appliquée aux codes MDS est le caractère MDS des codes MDS raccourcis. Ni la dimension, ni la distance minimale ne peuvent en effet croître sans contrevenir à la borne de Singleton.

On s'intéresse alors aux raccourcissements de codes BCH MDS de paramètres $[2k + r, k + r]_q$. On a l'assurance que les matrices de redondance obtenues seront MDS. Tout l'enjeu est donc de choisir les indices de raccourcissement, de manière à obtenir la propriété de récursivité souhaitée des matrices de redondance, bien que les codes ainsi obtenus perdent toute raison d'être cyclique. Un choix de positions à raccourcir se dégage de la matrice génératrice sous forme systématique canonique \mathbf{G}_{sys} d'un code cyclique \mathcal{C} de paramètres $[2k + r, k + r]_q$: les r dernières positions. La matrice de redondance devient récursive, associée au polynôme $g(X)$.

$$\mathbf{G}_{\text{sys}} = \left(\begin{array}{cc|cccccc} -X^k & \text{mod } g & 1 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ -X^{k+1} & \text{mod } g & 0 & 1 & 0 & \cdots & \cdots & \cdots & 0 \\ \vdots & & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ -X^{2k-1} & \text{mod } g & 0 & 0 & \cdots & 1 & 0 & \cdots & 0 \\ -X^{2k} & \text{mod } g & 0 & 0 & \cdots & 0 & 1 & \cdots & 0 \\ \vdots & & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ -X^{2k+r-1} & \text{mod } g & 0 & 0 & \cdots & 0 & \cdots & \cdots & 1 \end{array} \right)$$

La procédure pour construire des matrices récursives MDS à partir de raccourcissements de codes BCH MDS se résume alors à :

1. Sélectionner un élément β d'ordre $q+1$ dans \mathbb{F}_{q^2} .
Comme $q+1$ divise q^2-1 , β appartient systématiquement dans \mathbb{F}_{q^2} . Toutes les puissances non nulles de β ont donc des polynômes minimaux de degré 2 dans $\mathbb{F}_q[X]$. La racine conjuguée pour chaque racine β^i est $\beta^{qi} = \beta^{-i}$.
2. Distinguer les cas selon la parité de k :
 - Lorsque k est pair, sélectionner k puissances consécutives de β de sorte que chacune ait sa racine conjuguée contenue dans l'ensemble de ces k racines. Un ensemble apparaît :

$$\{\beta^i : i \in [\frac{q-k}{2} + 1, \frac{q+k}{2}]\}$$

- Lorsque k est impair, sélectionner k puissances consécutives de β de sorte que chacune ait sa racine conjuguée contenue dans l'ensemble de ces k racines donc que $\beta^0 = 1$ appartienne à cet ensemble de racines. Un ensemble apparaît :

$$\{\beta^i : i \in [-\frac{k-1}{2}, \frac{k-1}{2}]\}$$

3. Construire le polynôme $g(X) \in \mathbb{F}_q[X]$ unitaire de degré k qui annule l'ensemble de racines issu de l'étape 2. Ce polynôme engendre un code cyclique BCH MDS de paramètres $[q+1, q+1-k]_q$.
4. Raccourcir le code en les $q+1-2k$ dernières positions. Le code obtenu est un code MDS de paramètres $[2k, k]_q$. La matrice de redondance associée à la matrice génératrice systématique canonique de ce code est alors la matrice récursive obtenue comme puissance k -ième de la matrice compagnon associée à $g(X)$.

Exemple 7. Soit $k=4$. On veut construire des matrices à coefficients dans \mathbb{F}_{2^4} . On considère alors \mathbb{F}_{2^8} engendré par le polynôme $T^8 + T^4 + T^3 + T^2 + 1$ et α une de ses racines. Choisissons arbitrairement une racine primitive 17-ième de l'unité, α^{15} . Considérons alors le polynôme dont l'ensemble des racines est $\{\alpha^{7-15}, \dots, \alpha^{10-15}\}$, $g(X) = X^4 + \alpha^{204}X^3 + \alpha^{34}X^2 + \alpha^{204}X + 1$. Alors, $(\alpha^{204})^{16} = \alpha^{204}$ et $(\alpha^{34})^{16} = \alpha^{34}$ et le polynôme $g(X)$ appartient à $\mathbb{F}_{2^4}[X]$. La matrice \mathbf{C}_g engendre récursivement une matrice MDS, la matrice \mathbf{R}_g :

$$\mathbf{R}_g = \begin{pmatrix} 1 & \alpha^{204} & \alpha^{34} & \alpha^{204} \\ \alpha^{204} & \alpha^{119} & \alpha^{85} & \alpha^{187} \\ \alpha^{187} & \alpha^{153} & \alpha^{85} & \alpha^{68} \\ \alpha^{68} & \alpha^{102} & \alpha^{85} & \alpha^{34} \end{pmatrix}$$

Les matrices récursives issues de codes BCH raccourcis n'ont plus de raison d'être involutives. Il est néanmoins possible de composer les matrices récursives issues de la construction proposée avec une simple permutation pour construire des couches de diffusion linéaire récursives MDS et involutives. En effet, les polynômes issus de cette construction possèdent des racines conjuguées, ainsi $g(X)$ est son propre polynôme réciproque :

$$g^*(X) = X^k g(X^{-1}) = g(X)$$

Par ailleurs, en caractéristique 2, l'inverse de la matrice compagnon associée à un polynôme vérifiant la symétrie $g(X) = g^*(X)$ et $g(0) \neq 0$, est donnée par la matrice suivante :

$$\mathbf{C}_g^{-1} = \begin{pmatrix} g_0 & g_1 & \dots & g_{k-2} & g_{k-1} \\ 1 & 0 & \dots & \dots & 0 \\ & \ddots & \ddots & & \vdots \\ & & 1 & 0 & 0 \\ 0 & \dots & 0 & 1 & 0 \end{pmatrix} = \mathbf{C}_g \mathbf{P}$$

On a introduit ici la matrice \mathbf{P} :

$$\mathbf{P} = \begin{pmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & \dots & 0 & 1 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 1 & 0 & \dots & 0 \\ 1 & 0 & \dots & 0 & 0 \end{pmatrix}$$

La matrice $(\mathbf{C}_g^i \mathbf{P})$ est donc une involution pour tout $i \in \mathbb{N}$. *A fortiori*, c'est le cas pour $\mathbf{R}_g \mathbf{P}$. Il est donc très peu coûteux, une simple permutation de fils dans une implémentation matérielle, de construire des couches linéaires récursives MDS involutives.

Définition 42. Une matrice $\mathbf{M} \in \mathbb{F}_q^{k \times k}$ est dite presque involutive lorsque $(\mathbf{M}\mathbf{P})^2 = \mathbf{I}_k$.

Toutes les matrices récursives MDS ne sont pas obtenues avec cette construction directe. Cependant, lorsque les paramètres approchent les bornes de la conjecture MDS, il semble que les seules matrices récursives MDS qui existent sont celles issues de la construction précédente :

Conjecture 2. Lorsque $2k \geq q$, les seules matrices récursives MDS dans $\mathbb{F}_q^{k \times k}$ sont celles issues de codes cycliques BCH MDS de paramètres $[q+1, k+1]_q$ raccourcis.

Construction de matrices récursives MDS à partir de codes de Gabidulin [Ber13]

Soit $\alpha \in \mathbb{F}_{q^{2k}}$ une racine d'un polynôme de degré $2k$ irréductible sur $\mathbb{F}_q[X]$. Soit θ un élément générateur de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$. On construit la matrice \mathbf{G} suivante, matrice génératrice d'un code de Gabidulin \mathcal{G} :

$$\mathbf{G} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^{2k-1} \\ 1 & \alpha^{[1]} & (\alpha^2)^{[1]} & \dots & (\alpha^{2k-1})^{[1]} \\ 1 & \alpha^{[2]} & (\alpha^2)^{[2]} & \dots & (\alpha^{2k-1})^{[2]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{[k-1]} & (\alpha^2)^{[k-1]} & \dots & (\alpha^{2k-1})^{[k-1]} \end{pmatrix}$$

Cette matrice peut se réécrire $\mathbf{G} = (\mathbf{U} \mid \mathbf{D}^k \mathbf{U})$ au moyen des deux matrices \mathbf{U} et \mathbf{D} définies par :

$$\mathbf{U} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \dots & \alpha^{k-1} \\ 1 & \alpha^{[1]} & (\alpha^2)^{[1]} & \dots & (\alpha^{k-1})^{[1]} \\ 1 & \alpha^{[2]} & (\alpha^2)^{[2]} & \dots & (\alpha^{k-1})^{[2]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \alpha^{[k-1]} & (\alpha^2)^{[k-1]} & \dots & (\alpha^{k-1})^{[k-1]} \end{pmatrix} \quad \text{et} \quad \mathbf{D} = \begin{pmatrix} \alpha & 0 & \dots & 0 \\ 0 & \alpha^{[1]} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & \alpha^{[k-1]} \end{pmatrix}$$

Rappelons :

$$\begin{aligned} \alpha^i \alpha^j &= \alpha^{i+j} & \forall i, j \in \{1, \dots, 2k\} \\ (ab)^{[s]} &= a^{[s]} b^{[s]} & \forall s \in \mathbb{N}, \forall a, b \in \mathbb{F}_{q^m} \end{aligned}$$

Le lemme suivant est alors immédiat :

Lemme 1 ([Ber13], Lemme 1). *Pour tout $i \in \{1, \dots, k\}$, on a :*

$$\mathbf{D}^i \mathbf{U} = \begin{pmatrix} \alpha^i & \alpha^{i+1} & \alpha^{i+2} & \dots & \alpha^{i+k-1} \\ (\alpha^i)^{[1]} & (\alpha^{i+1})^{[1]} & (\alpha^{i+2})^{[1]} & \dots & (\alpha^{i+k-1})^{[1]} \\ (\alpha^i)^{[2]} & (\alpha^{i+1})^{[2]} & (\alpha^{i+2})^{[2]} & \dots & (\alpha^{i+k-1})^{[2]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (\alpha^i)^{[k-1]} & (\alpha^{i+1})^{[k-1]} & (\alpha^{i+2})^{[k-1]} & \dots & (\alpha^{i+k-1})^{[k-1]} \end{pmatrix}$$

La matrice $(\mathbf{I}_k \mid \mathbf{U}^{-1} \mathbf{D}^k \mathbf{U})$ est alors une matrice génératrice sous forme systématique de \mathcal{G} . Le théorème suivant prouve la récursivité de la matrice de redondance ainsi définie :

Théorème 20 ([Ber13], Théorème 1). *Il existe $g(X) \in \mathbb{F}_{q^{2k}}[X]$ tel que $\mathbf{U}^{-1} \mathbf{D}^k \mathbf{U} = \mathbf{C}_g^k$*

Démonstration. La matrice \mathbf{U} est la matrice de Vandermonde associée au même ensemble ordonné de valeurs qui forme la diagonale de \mathbf{D} : $\alpha, \alpha^{[1]}, \dots, \alpha^{[k-1]}$. D'après le Théorème 17, en posant $g(X)$ le polynôme dont les racines sont ces k valeurs distinctes, on obtient :

$$\mathbf{U} \mathbf{C}_g \mathbf{U}^{-1} = \mathbf{D}$$

Alors,

$$\mathbf{C}_g = \mathbf{U}^{-1} \mathbf{D} \mathbf{U} \quad \text{et finalement,} \quad \mathbf{C}_g^k = \mathbf{U}^{-1} \mathbf{D}^k \mathbf{U}$$

□

Par construction, \mathcal{G} est MRD donc MDS. La matrice \mathbf{C}_g^k est donc MDS en tant que matrice de redondance d'une matrice génératrice sous forme systématique d'un code MDS. Cette construction ne semble pas admettre de complément trivial qui permettrait de construire une couche de diffusion involutive à peu de frais.

Exemple 8. *Posons $k = 4$. On considère le corps \mathbb{F}_{2^8} engendré par le polynôme $T^8 + T^4 + T^3 + T^2 + 1$ et α une de ses racines. On construit alors les matrices \mathbf{U} et \mathbf{D} :*

$$\mathbf{U} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 \\ 1 & \alpha^4 & \alpha^8 & \alpha^{12} \\ 1 & \alpha^8 & \alpha^{16} & \alpha^{24} \end{pmatrix} \text{ et } \mathbf{D} = \begin{pmatrix} \alpha & 0 & 0 & 0 \\ 0 & \alpha^2 & 0 & 0 \\ 0 & 0 & \alpha^8 & 0 \\ 0 & 0 & 0 & \alpha^{24} \end{pmatrix}$$

La matrice \mathbf{C}_g induite est alors :

$$\mathbf{C}_g = \mathbf{U}^{-1} \mathbf{D} \mathbf{U}^{-1} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \alpha^{15} & \alpha^{168} & \alpha^{235} & \alpha^{238} \end{pmatrix}$$

Enfin, la matrice $\mathbf{R}_g = \mathbf{C}_g^4$ est MDS et vaut :

$$\mathbf{R}_g = \begin{pmatrix} \alpha^{15} & \alpha^{168} & \alpha^{235} & \alpha^{238} \\ \alpha^{253} & \alpha^{49} & \alpha^{170} & \alpha^{190} \\ \alpha^{205} & \alpha^{246} & \alpha^{92} & \alpha^{138} \\ \alpha^{153} & \alpha^{252} & \alpha^3 & \alpha^{18} \end{pmatrix}$$

Généralisation des constructions directes

Les deux constructions précédentes permettent de construire directement des matrices récursives MDS. Pourtant, il demeure insatisfaisant de ne pas être en mesure de dire à partir d'un polynôme générateur s'il engendre récursivement une matrice MDS à partir de considérations polynomiales. Dans la lignée de Gupta, Pandey et Venkateswarlu dans [GPV15a], les constructions précédentes se généralisent en considérant les ensembles de racines qui engendrent des polynômes dont les matrices récursives associées sont MDS.

Théorème 21 ([GPV15a], Théorème 1). *Soit $g(X) \in \mathbb{F}_q[X]$ un polynôme unitaire de degré k . La matrice compagnon associée \mathbf{C}_g engendre récursivement une matrice MDS \mathbf{C}_g^k si et seulement si le polynôme $g(X)$ ne possède pas de multiple de degré inférieur à $2k$ de poids inférieur ou égal à k .*

Démonstration. La preuve du théorème découle presque directement de la matrice génératrice sous forme systématique associée à la matrice récursive obtenue à partir de la matrice compagnon associée au polynôme $g(X)$, \mathbf{G} :

$$\mathbf{G} = \left(\begin{array}{cc|cccc} X^k & \text{mod } g & 1 & 0 & \cdots & 0 \\ X^{k+1} & \text{mod } g & 0 & 1 & \cdots & 0 \\ \vdots & & \vdots & \ddots & \ddots & \vdots \\ X^{2k-1} & \text{mod } g & 0 & 0 & \cdots & 1 \end{array} \right)$$

Tout mot de ce code s'écrit alors $(c_0, \dots, c_{k-1}, m_0, \dots, m_{k-1})$ avec $m(X) = \sum_{i=0}^{k-1} m_i X^i$ et $c(X) = \sum_{i=0}^{k-1} c_i X^i = \sum_{i=0}^{k-1} m_i X^{k+i} \text{ mod } g(X)$, de sorte qu'il existe $f(X) \in \mathbb{F}_q[X]$ tel que :

$$c(X) + m(X)X^k = f(X)g(X)$$

Un mot de code s'identifie donc à un multiple du polynôme $g(X)$ de degré inférieur ou égal à $2k - 1$ et réciproquement, tout multiple du polynôme $g(X)$ de degré inférieur ou égal à $2k - 1$ s'identifie à un mot de code.

Le caractère MDS de la matrice réursive \mathbf{C}_g^k équivaut à ce que tout mot de code soit de poids de Hamming supérieur ou égal à $k + 1$ ce qui équivaut alors à ce que tout multiple du polynôme $g(X)$ de degré inférieur ou égal à $2k - 1$ soit de poids de Hamming supérieur ou égal à $k + 1$. \square

Les résultats de Augot et Finiasz se généralisent à toutes les matrices rékursives obtenues à partir de matrices compagnon inversibles. Les matrices rékursives inversibles sont toujours la matrice de redondance d'un code cyclique de paramètres $[2k + r, k + r]$ raccourci en r positions. Malheureusement, toutes les matrices rékursives ne proviennent pas de raccourcissements de codes cycliques MDS, le raccourcissement peut en effet faire croître la distance minimale.

Exemple 9. Posons $k = 4$. On considère le corps fini \mathbb{F}_{2^4} engendré par le polynôme $T^4 + T^3 + 1$ dont on prend α une racine. Alors, le polynôme $g(X) = X^4 + \alpha^4 X^3 + \alpha^{12} X^2 + X + 1$ est associé à une matrice compagnon \mathbf{C}_g qui engendre rékursivement une matrice MDS \mathbf{R}_g . Le plus petit entier n tel que $g(X)$ divise $X^n - 1$ est 255. $g(X)$ engendre donc un code cyclique de paramètres $[255, 251]_{2^4}$ de matrice génératrice \mathbf{G} :

$$\mathbf{G} = \begin{pmatrix} g_0 & \cdots & g_3 & 1 & 0 & \cdots & 0 \\ 0 & g_0 & \cdots & g_3 & 1 & & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_0 & \cdots & g_3 & 1 \end{pmatrix}$$

Ce code présente une distance minimale de 4. Ce code n'est donc pas MDS et la matrice réursive MDS issue de la matrice compagnon associée à $g(X)$ ne peut pas être obtenue comme le raccourcissement naïf d'un code cyclique MDS.

La propriété du Théorème 21 ne permet pas de discrimination efficace des polynômes qui engendrent des matrices rékursives MDS. Il est possible d'exploiter les matrices de parité des codes engendrés par des matrices génératrices systématiques dont la matrice de redondance est une matrice réursive pour éclairer l'impact des racines d'un polynôme sur cette propriété, comme le suggère le théorème suivant :

Théorème 22 ([GPV15b], Théorème 1). Soit $\{\lambda_1, \dots, \lambda_k\}$ un ensemble de k éléments distincts non nuls de \mathbb{F}_q . Soit $g(X)$ le polynôme unitaire de degré k qui annule cet ensemble de racines. Alors, $g(X)$ engendre rékursivement une matrice MDS si et seulement si pour tous $\{r_1, \dots, r_k\} \subseteq \{0, \dots, 2k - 1\}$, la matrice de Vandermonde généralisée suivante est inversible :

$$\begin{pmatrix} \lambda_1^{r_1} & \cdots & \lambda_k^{r_1} \\ \lambda_1^{r_2} & \cdots & \lambda_k^{r_2} \\ \vdots & & \vdots \\ \lambda_1^{r_k} & \cdots & \lambda_k^{r_k} \end{pmatrix}$$

Démonstration. La démonstration de ce théorème provient du Théorème 8 et du fait que la matrice \mathbf{H} est une matrice de parité du code engendré par la matrice génératrice $\mathbf{G} = (\mathbf{C}_g^k \mid \mathbf{I}_k)$, avec :

$$\mathbf{H} = \begin{pmatrix} 1 & \lambda_1 & \lambda_1^2 & \cdots & \lambda_1^{2k-1} \\ 1 & \lambda_2 & \lambda_2^2 & \cdots & \lambda_2^{2k-1} \\ \vdots & & \ddots & & \vdots \\ 1 & \lambda_k & \lambda_k^2 & \cdots & \lambda_k^{2k-1} \end{pmatrix}$$

□

Le Théorème 22 donne la démonstration du théorème suivant, recensant trois constructions directes dont les deux premières sont les deux constructions directes présentées auparavant.

Théorème 23 ([GPV15b], Théorèmes 2,3 et 4).

- Soient λ et c dans \mathbb{F}_q . $\lambda_i = c^{i-1}\lambda$ pour tout $i \in \{1, \dots, n\}$.
Le polynôme $g(X) = \prod_{i=1}^k (X - \lambda_i)$ engendre récursivement une matrice MDS si et seulement si les éléments c, c^2, \dots, c^{2k-1} sont distincts et différents de 1.
- Soit λ un élément d'une extension de \mathbb{F}_2 . $\lambda_i = \lambda^{2^{i-1}}$ pour tout $i \in \{1, \dots, k\}$.
Le polynôme $g(X) = \prod_{i=1}^k (X - \lambda_i)$ engendre récursivement une matrice MDS si et seulement si le degré du polynôme minimal de λ sur \mathbb{F}_2 vérifie $\text{deg}(\Pi_{\mathbb{F}_2}(\lambda)) \geq 2k$
- Soient λ_1 et c dans une extension de \mathbb{F}_2 . $\lambda_i = c^{2^{i-2}}\lambda$ pour $i \in \{2, \dots, k\}$.
Le polynôme $g(X) = \prod_{i=1}^k (X - \lambda_i)$ engendre récursivement une matrice MDS si et seulement si le degré du polynôme minimal de c sur \mathbb{F}_2 vérifie $\text{deg}(\Pi_{\mathbb{F}_2}(c)) \geq 2k$

Remarque 10. Cachées dans le théorème précédent, on rappelle que les isométries pour la distance de Hamming sont données par le Théorème 10. Si un polynôme défini par un ensemble de k racines distinctes engendre récursivement une matrice MDS, il en va de même pour l'image de cet ensemble de racine par la multiplication par une constante et/ou par action d'un automorphisme sur l'ensemble des racines. La preuve de ce fait est immédiate suivant le Théorème 22. Elles expliquent l'apparition du terme λ multiplicatif des racines. On peut imaginer une deuxième généralisation qui inclut toutes les isométries et pas seulement les isométries linéaires.

Cette généralisation permet de caractériser les polynômes qui engendrent des matrices récursives MDS par des considérations sur leurs racines. Un écueil est que cette généralisation ne se décline pas directement en nouvelles constructions directes. Enfin, ce théorème porte en lui la question du cas des racines multiples.

4.1.3 Recherches exhaustives de matrices récursives MDS

Le Tableau 4.1 recense le nombre de matrices récursives MDS, presque involutives (Définition 42) ou non, ainsi que le meilleur compte de XOR obtenu pour les jeux de paramètres possibles sur \mathbb{F}_{2^4} .

Taille	Nombre de matrices MDS		Meilleur Compte de XOR	
	simples	Presque Involutives	simples	Presque Involutives
2×2	210	14	1	1
3×3	1980	12	2	2
4×4	3660	40	3	5
5×5	180	12	4	4
6×6	180	12	10	10
7×7	180	12	14	14
8×8	120	8	20	22

TABLE 4.1 – Dénombrement des matrices récursives MDS sur \mathbb{F}_{2^4}

Le Tableau 4.2 recense les polynômes qui engendrent les meilleurs comptes de XOR pour les matrices MDS sur \mathbb{F}_{2^4} . Tous ces résultats ont été obtenus lorsque \mathbb{F}_{2^4} est engendré par $T^4 + T^3 + 1$.

Taille	Polynôme record pour les matrices MDS	
	simples	Presque Involutives
2×2	$X^2 + \alpha^{14}X + 1$	<i>idem</i>
3×3	$X^3 + \alpha^{14}X^2 + \alpha^{14}X + 1$	<i>idem</i>
4×4	$X^4 + \alpha^{13}X^3 + X^2 + X + \alpha^{14}$	$X^4 + \alpha^{14}X^3 + \alpha^2X^2 + \alpha^{14}X + 1$
5×5	$X^5 + \alpha^{14}X^4 + \alpha X^3 + \alpha X^2 + \alpha^{14}X + 1$	<i>idem</i>
6×6	$X^6 + \alpha^2X^5 + \alpha X^4 + \alpha^{13}X^3 + \alpha X^2 + \alpha^2X + 1$	<i>idem</i>
7×7	$X^7 + \alpha^{13}X^6 + \alpha^{10}X^5 + X^4 + X^3 + \alpha^{10}X^2 + \alpha^{13}X + 1$	<i>idem</i>
8×8	$X^8 + \alpha^{14}X^7 + X^6 + \alpha^9X^5 + X^4 + \alpha^{13}X^3 + \alpha^8X^2 + \alpha^{11}X + \alpha$	$X^8 + \alpha^3X^7 + \alpha^{14}X^6 + \alpha^{12}X^5 + \alpha^{13}X^4 + \alpha^{12}X^3 + \alpha^{14}X^2 + \alpha^3X + 1$

TABLE 4.2 – Polynômes générateurs de matrices récursives MDS records sur \mathbb{F}_{2^4}

Remarque 11. *Ces résultats proposent de nouveaux défis :*

- *On constate sans avoir réussi à expliquer qu'il y a toujours 180 matrices dont 12 presque involutives à partir des tailles de matrices 5×5 , puis 120 matrices dont 8 presque involutives pour la taille de matrice 8×8 .*
- *Mis à part le cas des tailles 4×4 , les meilleurs comptes de XOR sont obtenus pour des matrices presque involutives. La construction de Augot et Finiasz est donc susceptible d'être très compétitive par rapport aux autres constructions directes car elles rentrent dans la catégorie des couches de diffusion linéaires involutives à moindre coût.*

4.2 Tordre la récursivité

Les matrices récursives sont reliées aux codes cycliques, idéaux de $\mathbb{F}_q[X]/(X^n - 1)$. Les deux constructions directes de matrices récursives MDS commencent par la définition d'ensembles de racines adéquats. En base normale, l'application d'un élément de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$ est peu coûteuse car correspond uniquement à une permutation de la représentation. On s'inspire alors des liens entre matrices récursives et codes cycliques pour tisser des liens similaires entre une nouvelle structure matricielles, les matrices θ -récursives et les codes θ -cycliques. Les relations entre les coefficients et les racines d'un θ -polynôme et les relations de divisibilités différent et distendent les rigidités qui interdisent l'existence de codes cycliques MDS de paramètres $[2k, k]$ sur des corps de caractéristique 2. On est dans ce nouveau paradigme en mesure de construire des codes θ -cycliques MDS, puis des matrices θ -récursives MDS, et même des matrices θ -récursives, θ^k -involutives MDS y compris sur des corps de caractéristique 2.

Nous allons introduire les matrices θ -récursives du calque de la section précédente appliqué aux codes θ -cycliques. On donne alors une construction directe de matrices θ -récursives MDS θ^k -involutives à partir de codes de Gabidulin. Nous concluons cette section par des résultats de

recherches exhaustives de matrices θ -récurives MDS.

4.2.1 Matrices θ -récurives et codes θ -cycliques

Pour la suite, on considère θ un élément de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$ et on construit l'anneau de θ -polynômes qu'il induit, $\mathbb{F}_{q^m}[X; \theta]$. La notion de code θ -cyclique, transposition naturelle de la notion de code cyclique, est précisée dans la définition suivante :

Définition 43. Soit \mathcal{C} un code linéaire de paramètres $[n, k]_{q^m}$ et θ un élément de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$. On dit que \mathcal{C} est θ -cyclique si pour tout mot de code $\mathbf{c} = (c_0, \dots, c_{n-1}) \in \mathcal{C}$, le vecteur obtenu par rotation des coefficients et application de θ coefficient par coefficient :

$$(\mathbf{c} \ggg 1)^{[1]} = (c_{n-1}^{[1]}, c_0^{[1]}, \dots, c_{n-2}^{[1]})$$

appartient également à \mathcal{C} .

On peut alors associer les mots d'un code θ -cyclique de paramètres $[n, k]_{q^m}$ et les éléments de $\mathbb{F}_{q^m}[X; \theta]/(X^n - 1)$:

$$\begin{aligned} \psi : \quad \mathcal{C} &\rightarrow \mathbb{F}_{q^m}[X; \theta]/(X^n - 1) \\ \mathbf{c} = (c_0, \dots, c_{n-1}) &\mapsto \sum_{i=0}^{n-1} c_i X^i \end{aligned}$$

On parlera alors indifféremment de mots de code pour les éléments de \mathcal{C} ou pour leur image par ψ . Par abus de notation, on écrira tout aussi indifféremment \mathcal{C} pour lui-même ou pour son image par ψ .

La rotation des coefficients suivie d'une application de θ coefficient par coefficient d'un mot de code θ -cyclique correspond dans $\mathbb{F}_{q^m}[X; \theta]/(X^n - 1)$ à la multiplication à gauche par X . On a alors la proposition suivante :

Proposition 19. Un code θ -cyclique est un idéal à gauche de $\mathbb{F}_{q^m}[X; \theta]/(X^n - 1)$.

Démonstration. Pour tout $\lambda \in \mathbb{F}_{q^m}$, pour tout mot de code $c\langle X \rangle$, par θ -cyclicité, $\lambda \cdot c\langle X \rangle, \dots, \lambda \cdot X^{n-1} \cdot c\langle X \rangle$ sont des mots de code. Par linéarité, toute combinaison linéaire de ces mots de code est également un mot de code. Pour tout polynôme $a\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]/(X^n - 1)$, $a\langle X \rangle \cdot c\langle X \rangle$ appartient donc aux mots de code et \mathcal{C} est un idéal à gauche de $\mathbb{F}_{q^m}[X; \theta]/(X^n - 1)$. \square

Du Théorème 5, on déduit les propriétés suivantes :

Propriétés 2. Soit \mathcal{C} un code θ -cyclique de paramètres $[n, k]_{q^m}$. Alors :

- \mathcal{C} possède un unique θ -polynôme unitaire de plus petit degré $n-k$, $g\langle X \rangle$, appelé θ -polynôme générateur de \mathcal{C} .
- Tout mot de code $c\langle X \rangle$ s'exprime comme un multiple à gauche du générateur $c\langle X \rangle = m\langle X \rangle \cdot g\langle X \rangle$, où $m\langle X \rangle$ est de degré inférieur à k .
- $g\langle X \rangle$ est un diviseur à droite de $X^n - 1$ dans $\mathbb{F}_{q^m}[X; \theta]$.

De leur définition et de la Propriété 2, on dérive pour un code θ -cyclique \mathcal{C} de paramètres $[n, k]_{q^m}$, à partir de son polynôme générateur $g\langle X \rangle = \sum_{i=0}^{n-k} g_i X^i$, l'existence d'une matrice génératrice canonique de \mathcal{C} , \mathbf{G} , sous la forme :

$$\mathbf{G} = \begin{pmatrix} g_0 & \cdots & g_{n-k} & 0 & 0 & \cdots & 0 \\ 0 & g_0^{[1]} & \cdots & g_{n-k}^{[1]} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0 & g_0^{[k-1]} & \cdots & g_{n-k}^{[k-1]} \end{pmatrix}$$

On peut également définir une matrice génératrice systématique \mathbf{G}_{sys} de \mathcal{C} . Par définition d'un code θ -cyclique, tout multiple à gauche de $g\langle X \rangle$ est un mot de code et, par linéarité, le θ -polynôme $(f\langle X \rangle - (f\langle X \rangle \bmod g\langle X \rangle))$ appartient également au code, pour tout θ -polynôme $f\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$ de degré inférieur à n . On utilise cette propriété spécialisée en $f\langle X \rangle = X^i$ pour tout $i \in [n-k, n-1]$ pour obtenir la matrice génératrice sous forme systématique canonique de \mathcal{C} dont la matrice de redondance suscite notre intérêt :

$$\begin{pmatrix} -X^{n-k} & \text{mod}_* g & \left| \begin{array}{ccc} 1 & 0 & \cdots & 0 \end{array} \right. \\ -X^{n-k+1} & \text{mod}_* g & \left| \begin{array}{ccc} 0 & 1 & \cdots & 0 \end{array} \right. \\ \vdots & \vdots & \left| \begin{array}{ccc} \vdots & \ddots & \ddots & \vdots \end{array} \right. \\ -X^{n-1} & \text{mod}_* g & \left| \begin{array}{ccc} 0 & 0 & \cdots & 1 \end{array} \right. \end{pmatrix}$$

Dans le cas des codes de paramètres $[2k, k]$, on nomme $\mathbf{R}_{\theta, g}$ l'inverse additif de la matrice de redondance du code θ -cyclique engendré par un polynôme unitaire de degré k diviseur à droite de $X^n - 1$:

$$\mathbf{R}_{\theta, g} = \begin{pmatrix} X^k & \text{mod}_* g \\ X^{k+1} & \text{mod}_* g \\ \vdots & \vdots \\ X^{2k-1} & \text{mod}_* g \end{pmatrix} \quad (4.3)$$

Cette matrice va être la clef de voûte des constructions de couches linéaires à partir de registre à décalage à rétroaction linéaire tordus, en anglais *Skewed Linear Feedback Shift Register* (SLFSR) présentés ultérieurement Figure 4.1, et se généralise à tout θ -polynôme unitaire de degré k , comme le laisse imaginer le théorème suivant :

Théorème 24. *Soit $g\langle X \rangle$ un θ -polynôme unitaire de degré k . Alors $\mathbf{R}_{\theta, g}$ est le produit matriciel des images successives par application de θ aux coefficients de la matrice compagnon associée à $g\langle X \rangle$.*

$$\mathbf{R}_{\theta, g} = \mathbf{C}_g^{[m-1]} \cdots \mathbf{C}_g^{[1]} \mathbf{C}_g = \mathbf{C}_{g^{[k-1]}} \cdots \mathbf{C}_{g^{[1]}} \mathbf{C}_g,$$

$$\text{où } g^{[i]}\langle X \rangle = \sum_{j=0}^{k-1} g_j^{[i]} X^j + X^k.$$

Soit $g\langle X \rangle$ le θ -polynôme générateur d'un code θ -cyclique MDS de paramètres $[2k, k]_{q^m}$. Alors, $\mathbf{R}_{\theta, g}$ est une matrice MDS qui vérifie :

$$\mathbf{R}_{\theta, g}^{[k]} \mathbf{R}_{\theta, g} = \mathbf{I}_k$$

Démonstration. La preuve du premier point du théorème repose sur la propriété \mathcal{P}_i suivante, que nous allons prouver par récurrence pour tout $i \geq 1$:

$$\mathcal{P}_i : \mathbf{C}_g^{[i-1]} \dots \mathbf{C}_g = \begin{pmatrix} X^i & \text{mod}_* g \\ \vdots & \\ X^{i+k-1} & \text{mod}_* g \end{pmatrix}$$

\mathcal{P}_0 est évidemment satisfaite puisqu'il s'agit de la matrice compagnon associée à $g(X)$:

$$\begin{pmatrix} X^1 & \text{mod}_* g \\ \vdots & \\ X^k & \text{mod}_* g \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \\ -g_0 & -g_1 & \dots & \dots & -g_{k-1} \end{pmatrix}$$

Supposons maintenant que \mathcal{P}_i soit satisfaite pour $i \geq 1$. Par calcul de $\mathbf{C}_g^{[i]} \mathbf{C}_g^{[i-1]} \dots \mathbf{C}_g$, on observe que les $(k-1)$ premières lignes sont respectivement égales à $X^{i+1} \text{mod}_* g, \dots, X^{i+k-1} \text{mod}_* g$. La dernière ligne vérifie quant à elle :

$$\begin{aligned} \sum_{j=0}^{k-1} g_j^{[i]} (X^{i+j} \text{mod}_* g) &= \left(\sum_{j=0}^{k-1} g_j^{[i]} X^{i+j} \right) \text{mod}_* g = \left(X^i \left(\sum_{j=0}^{k-1} g_j X^j \right) \right) \text{mod}_* g \\ &= (X^i (X^k + g(X))) \text{mod}_* g = X^{i+k} \text{mod}_* g \end{aligned}$$

La preuve par récurrence est conclue par cette dernière égalité et le premier point du théorème en découle directement.

Supposons désormais que $g(X)$ soit le θ -polynôme générateur d'un code θ -cyclique de paramètres $[2k, k]_{q^m}$. La propriété \mathcal{P}_{2k} s'écrit :

$$\mathbf{C}_g^{[2k-1]} \dots \mathbf{C}_g = \begin{pmatrix} X^{2k} & \text{mod}_* g \\ \vdots & \\ X^{3k-1} & \text{mod}_* g \end{pmatrix}$$

Comme $\mathbf{C}_g^{[2k-1]} \dots \mathbf{C}_g = (\mathbf{C}_g^{[k-1]} \dots \mathbf{C}_g)^{[k]} (\mathbf{C}_g^{[k-1]} \dots \mathbf{C}_g)$, on a donc :

$$\mathbf{R}_{\theta, g}^{[k]} \mathbf{R}_{\theta, g} = \begin{pmatrix} X^{2k} & \text{mod}_* g \\ \vdots & \\ X^{3k-1} & \text{mod}_* g \end{pmatrix}$$

On rappelle alors que $g(X)$ est un diviseur à droite de $X^{2k} - 1$. Ainsi, pour tout θ -polynôme $f(X) \in \mathbb{F}_{q^n} \langle X \rangle / (g(X))$, $f(X) X^{2k} \text{mod}_* g = f(X) \text{mod}_* g$. On obtient donc :

$$\mathbf{R}_{\theta, g}^{[k]} \mathbf{R}_{\theta, g} = \begin{pmatrix} X^0 & \text{mod}_* g \\ \vdots & \\ X^{k-1} & \text{mod}_* g \end{pmatrix} = \mathbf{I}_k$$

Par construction, $\mathbf{R}_{\theta,g}$ est l'inverse additif de la matrice de redondance d'une matrice génératrice sous forme systématique d'un code MDS et est donc MDS. \square

On définit alors la notion de θ -récursivité.

Définition 44. Une matrice \mathbf{M} est dite θ -récursive s'il existe $g\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$ tel que :

$$\mathbf{R}_{\theta,g} = \mathbf{C}_g^{[m-1]} \dots \mathbf{C}_g^{[1]} \mathbf{C}_g$$

On définit alors de même la notion de θ -involutive, une propriété structurelle des matrices construites à partir de codes θ -cycliques de paramètres $[2k, k]$.

Définition 45. Une matrice $\mathbf{M} \in \mathbb{F}_q^{k \times k}$ est θ^i -involutive si elle vérifie l'équation matricielle suivante :

$$\mathbf{M}^{[i]} \mathbf{M} = \mathbf{I}_k$$

Les matrices $\mathbf{R}_{\theta,g}$ issues de codes θ -cycliques $[2k, k]$ sont donc θ^k -involutives.

4.2.2 Construction directe de matrices θ -récursives MDS

Les codes de Gabidulin sont des codes MRD donc MDS. On va désormais considérer un sous-ensemble de ces codes qui sont θ -cycliques. Soit θ un élément générateur de $GAL(\mathbb{F}_{q^{2k}}/\mathbb{F}_q)$. Soit α un élément normal de $\mathbb{F}_{q^{2k}}$. Soit alors \mathcal{G} le code de Gabidulin de matrice de parité \mathbf{H} :

$$\mathbf{H} = \begin{pmatrix} \alpha^{[0]} & \alpha^{[1]} & \dots & \alpha^{[2k-1]} \\ \alpha^{[1]} & \alpha^{[2]} & \dots & \alpha^{[0]} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{[k-1]} & \alpha^{[k]} & \dots & \alpha^{[k-2]} \end{pmatrix} \quad (4.4)$$

Soit \mathbf{c} un mot de code de \mathcal{G} . Alors, $\sum_{i=0}^{2k-1} c_i \alpha^{j+i} = 0 = c(\alpha^{[j]})$, avec $c\langle X \rangle = \sum_{i=0}^{2k-1} c_i X^i$. Réciproquement, soit $c\langle X \rangle = \sum_{i=0}^{2k-1} c_i X^i$ un θ -polynôme qui s'annule en $\alpha^{[0]}, \dots, \alpha^{[k-1]}$, alors (c_0, \dots, c_{n-1}) est dans le dual du code engendré par \mathbf{H} et est donc un mot de code de \mathcal{G} . L'ensemble des θ -polynômes qui s'annulent en $\alpha^{[0]}, \dots, \alpha^{[k-1]}$ est clairement un idéal à gauche de $\mathbb{F}_{q^m}[X; \theta]/(X^{2k} - 1)$. \mathcal{G} est donc un code θ -cyclique.

On se focalise désormais sur les corps de caractéristique 2, cette construction étant motivée par une utilisation pratique en cryptographie. La procédure pour construire des matrices θ -récursives MDS se résume alors à :

1. Choisir un élément normal $\alpha \in \mathbb{F}_{2^{2k}}$.

Pour déterminer le caractère normal d'un élément, il est possible de vérifier que les images par les puissance successives de θ forment une base du \mathbb{F}_2 -espace vectoriel induit par $\mathbb{F}_{2^{2k}}$. Lorsque k est une puissance de 2, il existe 2^{2k-1} tels éléments, la moitié des éléments du corps, les éléments de trace égale à 1, [MP13] corollaire 5.2.9. Lorsque $2k$ n'est pas une puissance de 2, le nombre d'éléments normal est connu mais ne s'exprime pas aussi aisément, [MP13] corollaire 5.2.8.

2. Générer la matrice $\mathbf{H} = (\mathbf{H}_1 \mid \mathbf{H}_2)$ telle que :

$$\mathbf{H}_1 = \begin{pmatrix} \alpha^{[0]} & \alpha^{[1]} & \dots & \alpha^{[k-1]} \\ \alpha^{[1]} & \alpha^{[2]} & \dots & \alpha^{[k]} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{[k-1]} & \alpha^{[k]} & \dots & \alpha^{[2k-1]} \end{pmatrix}, \quad \mathbf{H}_2 = \begin{pmatrix} \alpha^{[k]} & \alpha^{[k+1]} & \dots & \alpha^{[2k-1]} \\ \alpha^{[k+1]} & \alpha^{[k+2]} & \dots & \alpha^{[0]} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{[2k-1]} & \alpha^{[0]} & \dots & \alpha^{[k-2]} \end{pmatrix}.$$

3. Construire $\mathbf{R}_{\theta,g} = \mathbf{H}_2 \mathbf{H}_1^{-1}$.

4. Récupérer le θ -polynôme $g\langle X \rangle$ à partir de la première ligne de $\mathbf{R}_{\theta,g}$ pour construire la matrice compagnon qui lui est associée, \mathbf{C}_g , qui engendre θ -récursivement $\mathbf{R}_{\theta,g}$.

Prouver la consistance de cette procédure revient à s'assurer que la matrice $\mathbf{H} = (\mathbf{H}_1 \mid \mathbf{H}_2)$ est une matrice de parité du code \mathcal{C} engendré par $(\mathbf{H}_2 \mathbf{H}_1^{-1} \mid \mathbf{I}_k)$. Or, on a :

$$(\mathbf{H}_2 \mathbf{H}_1^{-1} \mid \mathbf{I}_k)(\mathbf{H}_1 \mid \mathbf{H}_2)^T = \mathbf{H}_2 \mathbf{H}_1^{-1} \mathbf{H}_1^T + \mathbf{H}_2^T.$$

Comme \mathbf{H}_1 et \mathbf{H}_2 sont symétriques, $\mathbf{H}_1^T = \mathbf{H}_1$ et $\mathbf{H}_2^T = \mathbf{H}_2$. Ainsi, en caractéristique 2, on a :

$$\mathbf{H}_2 \mathbf{H}_1^{-1} \mathbf{H}_1^T + \mathbf{H}_2^T = \mathbf{H}_2 + \mathbf{H}_2 = \mathbf{0},$$

Remarque 12. Toute matrice de la forme $\mathbf{R}_{\theta,g}^{[i]}$ est également θ -récursive et θ^k -involutive.

Le théorème suivant exprime l'absence de redondance de cette construction directe :

Théorème 25. Soit θ un élément de $\text{GAL}(\mathbb{F}_{q^m}/\mathbb{F}_q)$. Soient α_1 et α_2 deux éléments normaux distincts. Alors, les deux matrices \mathbf{R}_{θ,g_1} et \mathbf{R}_{θ,g_2} obtenues respectivement canoniquement à partir des codes de Gabidulin \mathcal{G}_1 et \mathcal{G}_2 de support respectifs $\{\alpha_1, \alpha_1^{[1]}, \dots, \alpha_1^{[2k-1]}\}$ et $\{\alpha_2, \alpha_2^{[1]}, \dots, \alpha_2^{[2k-1]}\}$ sont différentes.

Démonstration. Les matrices MDS respectives \mathbf{R}_{θ,g_1} et \mathbf{R}_{θ,g_2} sont égales si et seulement si $\mathcal{G}_1 = \mathcal{G}_2$ soit si et seulement si les θ -polynômes générateurs respectifs $g_1\langle X \rangle$ et $g_2\langle X \rangle$ sont égaux. Soit $a \in \mathbb{N}$ tel que $\theta : x \mapsto x^{q^a}$.

Du point de vue des applications linéaires, comme démontré dans [Ber84], les racines de g_1 sont un \mathbb{F}_{q^a} -espace vectoriel qui contient les racines $\langle \alpha_1, \dots, \alpha_1^{[k-1]} \rangle$ tandis que les racines de g_2 sont un \mathbb{F}_{q^a} -espace vectoriel qui contient les racines $\langle \alpha_2, \dots, \alpha_2^{[k-1]} \rangle$. À l'instar du raisonnement réalisé dans la preuve de la proposition 15, la liberté sur \mathbb{F}_q de $\langle \alpha_1, \dots, \alpha_1^{[k-1]} \rangle$ implique, comme θ est générateur de $\text{GAL}(\mathbb{F}_{q^m}/\mathbb{F}_q)$ la liberté en tant que \mathbb{F}_{q^a} espace vectoriel.

Ainsi, $\mathbf{R}_{\theta,g_1} = \mathbf{R}_{\theta,g_2}$ implique :

$$\langle \alpha_1, \dots, \alpha_1^{[k-1]} \rangle = \langle \alpha_2, \dots, \alpha_2^{[k-1]} \rangle.$$

L'élément α_1 est alors une combinaison linéaire de $\alpha_2, \dots, \alpha_2^{[k-1]}$:

$$\alpha_1 = \sum_{i=0}^{k-1} a_i \alpha_2^{[i]}, \quad a_i \in \mathbb{F}_{q^m} \quad \text{et} \quad \alpha_1^{[1]} = \sum_{i=0}^{k-1} a_i \alpha_2^{[i+1]} = \sum_{i=1}^{k-1} a_{i-1} \alpha_2^{[i]} + a_{k-1} \alpha_2^{[k]}.$$

Comme $\alpha_2^{[i]} \in \langle \alpha_1, \dots, \alpha_1^{[k-1]} \rangle$ pour tout $i \in \{1, \dots, k-1\}$, le premier terme s'écrit comme une combinaison linéaire de $\alpha_1, \dots, \alpha_1^{[k-1]}$.

De même, $\alpha_2^{[k]}$ peut se réécrire $(\alpha_2^{[k-1]})^{[1]}$, où $\alpha_2^{[k-1]} = \sum_{i=0}^{k-1} b_i \alpha_1^{[i]} \in \langle \alpha_2, \dots, \alpha_2^{[k-1]} \rangle$. Ainsi, $\alpha_2^{[k]} = \sum_{i=0}^{k-1} b_{i-1} \alpha_1^{[i]} + b_{k-1} \alpha_1^{[k]}$. Nécessairement, $b_{k-1} \neq 0$ car sinon $\alpha_2^{[k]} \in \langle \alpha_2, \dots, \alpha_2^{[k-1]} \rangle$, ce qui contrevient à l'hypothèse α_2 élément normal. On a donc :

$$\alpha_1^{[1]} = \lambda + a_{k-1} \alpha_1^{[k]}.$$

Le terme λ est une combinaison linéaire de $\alpha_1, \dots, \alpha_1^{[k-1]}$. Or, $\alpha_1, \dots, \alpha_1^{[2k-1]}$ forment une famille \mathbb{F}_{q^a} -libre de $\mathbb{F}_{q^{2k}}$ et $a_{k-1} = 0$. Par induction, on montre que $a_i = 0$ pour $i \neq 0$.

Ainsi, $\langle \alpha_1, \dots, \alpha_1^{[k-1]} \rangle = \langle \alpha_2, \dots, \alpha_2^{[k-1]} \rangle \Rightarrow \alpha_1 = \alpha_2$ et $\alpha_1 \neq \alpha_2 \Rightarrow \mathbf{R}_{\theta, g_1} \neq \mathbf{R}_{\theta, g_2}$. \square

Exemple 10. *Cet exemple illustre la procédure de la construction directe dans un jeu de paramètres concret : $k = 4$. On considère alors \mathbb{F}_{2^8} engendré par le polynôme irréductible $T^8 + T^4 + T^3 + T^2 + 1$. Soit θ l'automorphisme du Frobenius. Remarquons que α n'est pas un élément normal.*

1. On choisit de considérer l'élément normal α^{21} .
2. On construit la matrice de parité sous la forme (4.4) est :

$$\left(\begin{array}{c|c|c|c|c|c|c|c} \alpha^{21} & \alpha^{42} & \alpha^{84} & \alpha^{168} & \alpha^{81} & \alpha^{162} & \alpha^{69} & \alpha^{138} \\ \alpha^{42} & \alpha^{84} & \alpha^{168} & \alpha^{81} & \alpha^{162} & \alpha^{69} & \alpha^{138} & \alpha^{21} \\ \alpha^{84} & \alpha^{168} & \alpha^{81} & \alpha^{162} & \alpha^{69} & \alpha^{138} & \alpha^{21} & \alpha^{42} \\ \alpha^{168} & \alpha^{81} & \alpha^{162} & \alpha^{69} & \alpha^{138} & \alpha^{21} & \alpha^{42} & \alpha^{84} \end{array} \right)$$

3. La matrice $\mathbf{R}_{\theta, g}$ s'écrit donc :

$$\mathbf{R}_{\theta, g} = \left(\begin{array}{c|c|c|c} \alpha^{199} & \alpha^{96} & \alpha^{52} & \alpha^{123} \\ \alpha^{190} & \alpha^{218} & \alpha^{231} & \alpha^{125} \\ \alpha^{194} & \alpha^{227} & \alpha^{224} & \alpha^{66} \\ \alpha^{76} & \alpha^{54} & \alpha^{217} & \alpha^{28} \end{array} \right)$$

La matrice inverse est alors donnée par :

$$\mathbf{R}_{\theta, g}^{-1} = \mathbf{R}_{\theta, g}^{[4]} = \left(\begin{array}{c|c|c|c} \alpha^{124} & \alpha^6 & \alpha^{67} & \alpha^{183} \\ \alpha^{235} & \alpha^{173} & \alpha^{126} & \alpha^{215} \\ \alpha^{44} & \alpha^{62} & \alpha^{14} & \alpha^{36} \\ \alpha^{196} & \alpha^{99} & \alpha^{157} & \alpha^{193} \end{array} \right)$$

4. Le produit matrice-vecteur peut être réalisé avec une implémentation de la matrice compagnon associée à $g(X)$, l'unique θ -polynôme de degré 4 satisfaisant les équations linéaires : $g(X) = \alpha^{199} + \alpha^{96} X + \alpha^{52} X^2 + \alpha^{123} X^3 + X^4$:

$$\mathbf{C}_g = \left(\begin{array}{cccc} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \alpha^{199} & \alpha^{96} & \alpha^{52} & \alpha^{123} \end{array} \right)$$

Le cœur des deux algorithmes suivant provient de la décomposition présentée dans le Théorème 24 et consiste en l'itération d'une même boucle. Une étape dans la boucle correspond dans une certaine mesure au produit matrice-vecteur avec la matrice compagnon d'un polynôme.

Algorithme 1 Produit matrice-vecteur

Entrées: $\mathbf{x} \in \mathbb{F}_{2^{2k}}^k$ un vecteur d'entrée et \mathbf{C}_g la matrice compagnon qui engendre $\mathbf{R}_{\theta,g}$.

Sortie: $\mathbf{y} = \mathbf{R}_{\theta,g}\mathbf{x}$ le résultat du produit matrice-vecteur

- 1: $\mathbf{y} \leftarrow \mathbf{x}^{[1]}$ ▷ Initialisation
 - 2: **pour** $i = 0$ à $k - 1$ **faire**
 - 3: $\mathbf{y} \leftarrow \mathbf{C}_g\mathbf{y}^{[-1]}$ ▷ Produit matrice-vecteur avec la matrice compagnon
 - 4: **fin pour**
 - 5: $\mathbf{y} \leftarrow \mathbf{y}^{[k-1]}$ ▷ Étape finale
 - 6: **retourner** \mathbf{y}
-

La consistance de l'algorithme suit la démonstration de la propriété suivante, réalisée par induction pour tout $i \geq 0$:

$$\mathcal{P}_i : \mathbf{y}^{[i]} = \mathbf{C}_g^{[i]} \dots \mathbf{C}_g \mathbf{x}$$

La propriété \mathcal{P}_0 est satisfaite attendu que $\mathbf{y} = \mathbf{C}_g \mathbf{x}$. Supposons alors \mathcal{P}_i satisfaite pour un certain $i \geq 0$. Après une $(i + 1)$ -ème étape, on a grâce à la propriété distributive de l'application de θ :

$$\mathbf{y}_{(i+1)}^{[i+1]} = \left(\mathbf{C}_g \mathbf{y}_{(i)}^{[-1]} \right)^{[i+1]} = \mathbf{C}_g^{[i+1]} \mathbf{y}_{(i)}^{[i]} = \mathbf{C}_g^{[i+1]} \mathbf{C}_g^{[i]} \dots \mathbf{C}_g \mathbf{x}$$

\mathcal{P}_{i+1} est donc satisfaite et, par induction, après k étapes :

$$\mathcal{P}_{k-1} : \mathbf{y}^{[k-1]} = \mathbf{C}_g^{[k-1]} \dots \mathbf{C}_g \mathbf{x}$$

À la fin de la boucle, on applique $k - 1$ fois θ pour obtenir le produit matrice-vecteur désiré, origine de la dernière étape.

Le produit matrice-vecteur par la matrice inverse est réalisé par un algorithme jumeau de celui pour le produit matrice-vecteur par $\mathbf{R}_{\theta,g}$:

Algorithme 2 Produit matrice-vecteur pour la matrice inverse

Entrées: $\mathbf{x} \in \mathbb{F}_{2^{2m}}^m$ un vecteur d'entrée et C_g

Sortie: $\mathbf{y} = \mathbf{R}_{\theta,g}^{-1}\mathbf{x}$ le résultat du produit matrice-vecteur pour la matrice inverse

- 1: $\mathbf{y} \leftarrow \mathbf{x}^{[-m+1]}$ ▷ Initialisation
 - 2: **pour** $i = 0$ à $m - 1$ **faire**
 - 3: $\mathbf{y} \leftarrow C_g \mathbf{y}^{[-1]}$ ▷ Produit matrice-vecteur avec la matrice compagnon
 - 4: **fin pour**
 - 5: $\mathbf{y} \leftarrow \mathbf{y}^{[-1]}$ ▷ Étape finale
 - 6: **retourner** \mathbf{y}
-

La preuve de la consistance est pratiquement la même que celle du premier algorithme puisque :

$$\mathbf{R}_{\theta,g}^{-1}\mathbf{x} = \mathbf{R}_{\theta,g}^{[k]}\mathbf{x} = \left(\mathbf{R}_{\theta,g}\mathbf{x}^{[-k]} \right)^{[k]}$$

Ainsi, calculer le produit matrice-vecteur et le produit matrice-vecteur avec la matrice inverse peuvent se factoriser facilement. Seule les étapes d'initialisation et finales sont différentes.

Le produit matrice-vecteur par une matrice $\mathbf{R}_{\theta,g}$ peut être implémenté par l'exécution de la routine d'un SLFSR, architecture non classique fortement inspirée du LFSR illustrée par la Figure 4.1. Chaque fois que la matrice est θ^k -involutive, ce qui n'est pas forcément le cas lorsqu'elle n'est pas le fruit de cette construction directe, le produit matrice-vecteur par la matrice inverse est implémenté par l'ajout d'un simple routage. Cette architecture est composée de registres, les (X_i), de modules de calcul de l'inverse de θ , les (θ^{-1}), de multiplications dans le corps de Galois par des constantes, les (g_i), et des sommes dans le corps de Galois, les (XOR).

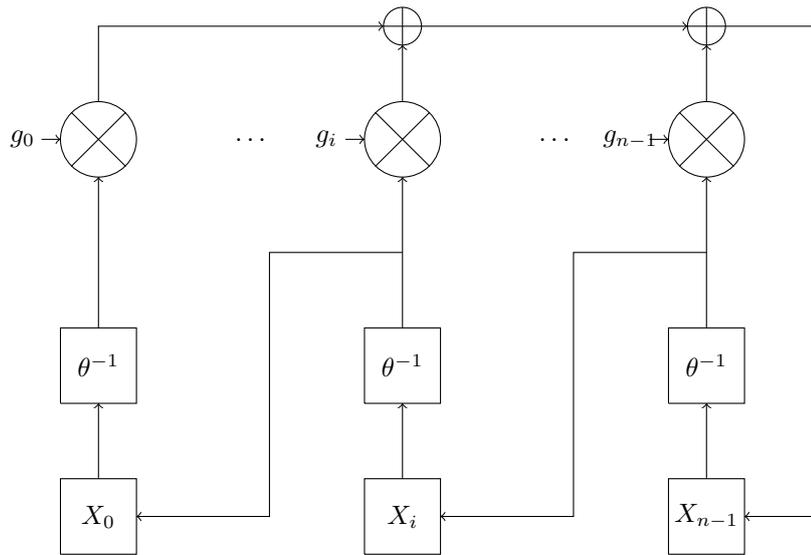


FIGURE 4.1 – Registre à décalage à rétroaction linéaire tordu.

L'utilisation de cette architecture suggère de faire tous les calculs en base normale, bien plus adaptée que les bases polynomiales pour le calcul des automorphismes. En effet, dans une base normale, l'application de θ^i est aisément réalisée par une permutation fixe de bits pour tout $i \in \mathbb{Z}$. En première approximation, il est légitime de considérer que le calcul de cette opération n'implique pas de coûts matériels supplémentaires. Par ailleurs, comme constaté par Anne Canteau et Joelle Roué, le choix de la base n'a pas d'influence sur les propriétés linéaires et différentielles du schéma de chiffrement, traduction sommaire de la complexité des attaques statistiques [CR15]. Tout coût supplémentaire concernant les transformations de bases dépend d'un choix d'implémentation et ne peut être évalué *a priori*. Si nécessaire, les transformations à partir d'une base normale vers une base polynomiale et inversement sont réalisées par des matrices linéaires binaires qui doivent alors être appliquées avant et après l'exécution de l'algorithme.

Supposons que $\mathbf{y} = (y_0, \dots, y_{m-1})$ représente l'état actuel dans une telle architecture, alors l'étape obtenue par une exécution de la routine du SLFSR est $\mathbf{C}_g \mathbf{y}^{[-1]}$. Ceci correspond bien à une étape dans la boucle *for* des algorithmes 1 et 2. Ceci implique que le produit d'une matrice $\mathbf{R}_{\theta,g}$ par un vecteur peut être implémentée simplement par k exécution d'une routine très similaire à un LFSR.

Remarque 13. Le coût du calcul d'un produit matrice-vecteur avec $\mathbf{R}_{\theta,g}$ et celui avec son inverse sont les mêmes. Cependant, pour être capable de calculer les deux avec une telle implémentation, un MUX additionnel est nécessaire pour décider quelle permutation des entrées et quelles permutations des sorties doivent être réalisées.

Généralisation de la construction directe La construction précédente se généralise en considérant les espaces de racines susceptibles d'engendrer des θ -polynômes dont les matrices θ -récurives associées sont MDS.

Théorème 26. Soit $g\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$ un θ -polynôme unitaire de degré k . La matrice compagnon associée \mathbf{C}_g engendre θ -récurivement une matrice MDS $\mathbf{R}_{\theta,g}$ si et seulement si le polynôme $g\langle X \rangle$ ne possède pas de multiple à gauche de degré inférieur à $2k$ de poids inférieur ou égal à k .

Démonstration. La preuve du théorème découle presque directement de la matrice génératrice sous forme systématique associée à la matrice θ -récurive obtenue à partir de la matrice compagnon associée au polynôme $g\langle X \rangle$, \mathbf{G} :

$$\mathbf{G} = \left(\begin{array}{cc|cccc} X^k & \text{mod}_* g & 1 & 0 & \cdots & 0 \\ X^{k+1} & \text{mod}_* g & 0 & 1 & \cdots & 0 \\ & \vdots & \vdots & \ddots & \ddots & \vdots \\ X^{2k-1} & \text{mod}_* g & 0 & 0 & \cdots & 1 \end{array} \right)$$

Tout mot de ce code s'écrit alors $(c_0, \dots, c_{k-1}, m_0, \dots, m_{k-1})$, avec $c\langle X \rangle = \sum_{i=0}^{k-1} m_i X^i$ et $c\langle X \rangle = \sum_{i=0}^{k-1} c_i X^i = \sum_{i=0}^{k-1} m_i X^{k+i} \text{ mod }_* g$, de sorte qu'il existe $f\langle X \rangle \in \mathbb{F}_{q^m}[X; \theta]$ tel que :

$$c\langle X \rangle + m\langle X \rangle \cdot X^k = f\langle X \rangle \cdot g\langle X \rangle$$

Un mot de code s'identifie donc à un multiple à gauche du θ -polynôme $g\langle X \rangle$ de degré inférieur ou égal à $2k - 1$. Réciproquement, tout multiple à gauche de $g\langle X \rangle$ de degré inférieur à $2k - 1$ s'identifie à un mot de code.

$\mathbf{R}_{\theta,g}$ MDS équivaut à ce que tout mot de code non nul soit de poids de Hamming supérieur ou égal à $k + 1$ ce qui équivaut alors à ce que tout multiple à gauche non nul de $g\langle X \rangle$ de degré inférieur ou égal à $2k - 1$ soit de poids de Hamming supérieur ou égal à $k + 1$. □

On peut généraliser la construction et considérer les matrices θ -récurives obtenues à partir de n'importe quelle matrice compagnon inversible et constater que celles-ci engendrent toujours un code θ -cyclique. De même que dans le cas classique, toutes les matrices θ -récurives MDS ne proviennent pas de codes θ -cycliques de paramètres $[2k, k]_{q^m}$ et on peut toutes les concevoir comme des raccourcissements de codes θ -cycliques de paramètres $[2k + r, k + r]_{q^m}$, non nécessairement MDS.

Exemple 11. Posons $k = 4$. Le corps fini qu'on considère est \mathbb{F}_{2^4} . Soit α une racine du polynôme $T^4 + T^3 + 1$ et θ l'automorphisme de Frobenius. Alors, le polynôme $g\langle X \rangle = X^4 + \alpha^{10} X^3 + \alpha^3 X^2 + X + 1$ est associé à une matrice compagnon \mathbf{C}_g qui engendre θ -récurivement une matrice MDS $\mathbf{R}_{\theta,g}$. Le plus petit entier n tel que $g\langle X \rangle$ divise à droite $X^n - 1$ est 16. $g\langle X \rangle$ engendre donc un code θ -cyclique de paramètres $[16, 12]_{2^4}$. Ce code présente une distance minimale de 3. Ce code n'est donc pas MDS et la matrice θ -récurive issue de la matrice compagnon associée à $g\langle X \rangle$ ne peut pas être obtenue comme le raccourcissement naïf d'un code θ -cyclique MDS.

À l'instar du Théorème 21, la propriété du Théorème 26 ne permet pas de discrimination efficace des polynômes qui engendrent des matrices θ -récurives MDS. On peut cependant préciser les contraintes qui conditionnent le caractère MDS de la matrice θ -récurive obtenue à partir d'un θ -polynôme interpolateur d'un espace racine est de dimension k :

Théorème 27. *Soit θ un élément générateur de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$. Soient $\lambda_1, \dots, \lambda_k$ des éléments \mathbb{F}_q -libres de \mathbb{F}_{q^m} . Soit $g\langle X \rangle$ le polynôme unitaire défini par cet ensemble de racines. Alors, $g\langle X \rangle$ engendre θ -récurivement une matrice MDS si et seulement si pour tous $\{r_1, \dots, r_k\} \subseteq \{0, \dots, 2k-1\}$, la matrice de Vandermonde généralisée suivante est inversible :*

$$\begin{pmatrix} \lambda_1^{[r_1]} & \dots & \lambda_k^{[r_1]} \\ \lambda_1^{[r_2]} & \dots & \lambda_k^{[r_2]} \\ \vdots & & \vdots \\ \lambda_1^{[r_k]} & \dots & \lambda_k^{[r_k]} \end{pmatrix}$$

Démonstration. La démonstration de ce théorème est le calque de celle du Théorème 22. On applique alors le Théorème 8 sur la matrice \mathbf{H}_θ , matrice de parité du code engendré par la matrice génératrice $\mathbf{G}_\theta = (\mathbf{R}_{\theta,g} \mid \mathbf{I}_k)$, avec :

$$\mathbf{H}_\theta = \begin{pmatrix} \lambda_1 & \lambda_1^{[1]} & \lambda_1^{[2]} & \dots & \lambda_1^{[2k-1]} \\ \lambda_2 & \lambda_2^{[1]} & \lambda_2^{[2]} & \dots & \lambda_2^{[2k-1]} \\ \vdots & & \ddots & & \vdots \\ \lambda_k & \lambda_k^{[1]} & \lambda_k^{[2]} & \dots & \lambda_k^{[2k-1]} \end{pmatrix}$$

□

Remarque 14. *Suivant la Remarque 10 on est capable d'engendrer à partir d'un ensemble de racines dont le polynôme associé engendre une matrice θ -récurive MDS d'autres ensembles de racines, images de l'ensemble initial par un automorphisme, dont le polynôme interpolateur engendre également une matrice θ -récurive MDS.*

4.2.3 Recherche exhaustive

Le Tableau 4.3 recense le nombre de matrices θ -récurives MDS, respectivement les matrices θ -récurives MDS θ^k -involutives, ainsi que le meilleur compte de XOR obtenu pour les jeux de paramètres possibles sur \mathbb{F}_{2^4} .

Taille	Nombre de matrices θ -récurives MDS		Meilleur Compte de XOR	
	simples	θ^k -Involutives	simples	θ^k -Involutives
2×2	624	74	3	6
3×3	6234	126	3	9
4×4	10860	720	6	8
5×5	1920	520	14	14
6×6	1120	120	17	17

TABLE 4.3 – Dénombrement des matrices θ -récurives MDS sur \mathbb{F}_{2^4}

Le Tableau 4.4 recense les θ -polynômes qui engendrent les meilleurs comptes de XOR pour les matrices θ -récurives MDS sur \mathbb{F}_{2^4} . Tous ces résultats ont été obtenus dans la base normale associée à l'élément α^3 où α racine de $T^4 + T^3 + 1$.

Taille	Polynôme record pour les matrices θ^k -récurives MDS	
	simples	θ^k -Involutives
2×2	$X^2 + \alpha^3 X + 1$	$X^2 + X + \alpha^5$
3×3	$X^3 + \alpha^3 X^2 + X + 1$	$X^3 + \alpha^6 X^2 + \alpha X + 1$ $\theta : a \mapsto a^4$
4×4	$X^4 + \alpha^{12} X^3 + X^2 + X + \alpha^3$	$X^4 + \alpha^9 X^3 + X^2 + X + \alpha^7$ $\theta : a \mapsto a^8$
5×5	$X^5 + X^4 + \alpha^4 X^3 + \alpha^7 X^2 + X + \alpha^3$ $\theta : a \mapsto a^4$	<i>idem</i>
6×6	$X^6 + \alpha^{12} X^5 + \alpha^{14} X^4 + \alpha^3 X^3 + X^2 + \alpha X + 1$	<i>idem</i>

TABLE 4.4 – Polynômes générateurs de matrices θ -récurives MDS records sur \mathbb{F}_{2^4}

Remarque 15. *Ces résultats proposent de nouveaux défis :*

- *On constate sans avoir réussi à l'expliquer qu'il n'existe pas de matrices θ -récurives de taille supérieure à 6.*
- *On constate sans avoir réussi à l'expliquer que pour la taille 6×6 , toutes les matrices MDS sont θ^6 -involutives.*
- *Aucun jeu de paramètres ne donne de meilleur compte de XOR que le cas des matrices récurives bien qu'il semble qu'avec la croissance de la taille de la matrice, cet écart soit de moins en moins significatif.*
- *On s'est assuré que les matrices ainsi obtenues n'étaient pas des matrices récurives. Prouver ce fait, s'il est général, semble non trivial.*

4.3 Relâcher les contraintes

Plusieurs raffinements sont possibles pour construire des matrices MDS en lien avec les structures précédentes. On s'applique pour commencer à présenter une construction directe de matrices MDS involutives à partir de matrices de Vandermonde dont on s'inspire pour proposer une construction directe de matrices MDS à partir de θ -matrices de Vandermonde. On s'intéresse alors aux avantages et inconvénients d'autoriser davantage de k récursions d'un LFSR ou d'un SLFSR pour construire des matrices MDS. Enfin, nous évoquerons la possibilité de modifier fortement la structure d'un LFSR pour déterminer des couches de diffusion linéaire récurives optimales en terme de diffusion.

4.3.1 Matrices MDS involutives à partir de matrices de Vandermonde

Une autre construction directe de matrices MDS, due à Sajadieh, Dakhilalian, Mala et Omoomi dans [Saj+12a], paraît à première vue *ad hoc*. Des similitudes avec les matrices de parité observées précédemment apparaissent.

Le Théorème 22 fait apparaître des matrices de Vandermonde dans les matrices de parité de codes cycliques MDS. Pour $g(X)$ scindé à racines simples, la matrice de parité \mathbf{H} associée au code cyclique raccourci de paramètres $[2k, k]$ est de la forme $(\mathbf{V}_1 \mid \mathbf{V}_2)$ avec \mathbf{V}_1 et \mathbf{V}_2 des matrices de Vandermonde. L'analogie est possible avec les θ -matrices de Vandermonde qui apparaissent dans les matrices de parité de codes θ -cycliques MDS dans le Théorème 27. Pour $g(X)$ tel que $g_0 \neq 0$, la matrice de parité \mathbf{H}_θ associée au code θ -cyclique raccourci de paramètres $[2k, k]$ est de la forme $(\mathbf{V}_{\theta,1} \mid \mathbf{V}_{\theta,2})$ avec $\mathbf{V}_{\theta,1}$ et $\mathbf{V}_{\theta,2}$ des θ -matrices de Vandermonde

La construction directe de matrices MDS à partir de matrices de Vandermonde se base également sur les liens qui les unissent aux polynômes. Plus précisément, $\mathbf{V}_1 \mathbf{V}_2^{-1}$ est MDS si et seulement

si $(\mathbf{V}_1^{-1} \mid \mathbf{V}_2^{-1})$ est MDS.

Théorème 28 ([Saj+12a], Théorème 2). *Soient \mathbf{V}_1 et \mathbf{V}_2 deux matrices de Vandermonde aux supports respectifs $\{a_0, \dots, a_{k-1}\}$ et $\{b_0, \dots, b_{k-1}\}$, disjoints et composés d'éléments deux à deux distincts :*

$$\mathbf{V}_1 = \begin{pmatrix} 1 & a_0 & a_0^2 & \dots & a_0^{k-1} \\ 1 & a_1 & a_1^2 & \dots & a_1^{k-1} \\ \vdots & & \ddots & & \vdots \\ 1 & a_{k-1} & a_{k-1}^2 & \dots & a_{k-1}^{k-1} \end{pmatrix} \quad \text{et} \quad \mathbf{V}_2 = \begin{pmatrix} 1 & b_0 & b_0^2 & \dots & b_0^{k-1} \\ 1 & b_1 & b_1^2 & \dots & b_1^{k-1} \\ \vdots & & \ddots & & \vdots \\ 1 & b_{k-1} & b_{k-1}^2 & \dots & b_{k-1}^{k-1} \end{pmatrix}$$

Alors, la matrice $\mathbf{V}_1\mathbf{V}_2^{-1}$ est une matrice MDS.

Démonstration. Par l'absurde, supposons qu'il existe \mathbf{x} et \mathbf{y} avec $w_H(\mathbf{x}) + w_H(\mathbf{y}) \leq k$ tels que $\mathbf{y} = \mathbf{V}_1\mathbf{V}_2^{-1}\mathbf{x}$. Définissons le vecteur $\mathbf{p} = \mathbf{V}_2^{-1}\mathbf{x}$. Alors :

$$\mathbf{x} = \mathbf{V}_2\mathbf{p} \Leftrightarrow \begin{cases} x_0 & = \sum_{i=0}^{k-1} b_0^i p_i \\ x_1 & = \sum_{i=0}^{k-1} b_1^i p_i \\ \dots & \\ x_{k-1} & = \sum_{i=0}^{k-1} b_{k-1}^i p_i \end{cases} \quad \text{et} \quad \mathbf{y} = \mathbf{V}_1\mathbf{p} \Leftrightarrow \begin{cases} y_0 & = \sum_{i=0}^{k-1} a_0^i p_i \\ y_1 & = \sum_{i=0}^{k-1} a_1^i p_i \\ \dots & \\ y_{k-1} & = \sum_{i=0}^{k-1} a_{k-1}^i p_i \end{cases}$$

Les $2k$ coefficients de \mathbf{x} et \mathbf{y} sont les évaluations d'un polynôme $p(X)$ de degré $k-1$ en $2k$ valeurs distinctes. Au plus $k-1$ valeurs peuvent donc être simultanément nulles pour un polynôme p non identiquement nul. Absurde. \square

Il est possible de choisir les supports des matrices de Vandermonde de sorte que la matrice MDS construite soit également involutive, comme l'indique le théorème suivant :

Théorème 29 ([Saj+12a], Théorème 3). *Soient \mathbf{V}_1 et \mathbf{V}_2 les matrices de Vandermonde définies par des éléments $(a_i)_{i \leq k}$ et Δ tels que les a_i et $a_i + \Delta$ sont tous distincts :*

$$\mathbf{V}_1 = \begin{pmatrix} 1 & a_0 & a_0^2 & \dots & a_0^{k-1} \\ 1 & a_1 & a_1^2 & \dots & a_1^{k-1} \\ \vdots & & \ddots & & \vdots \\ 1 & a_k & a_k^2 & \dots & a_k^{k-1} \end{pmatrix} \quad \text{et} \quad \mathbf{V}_2 = \begin{pmatrix} 1 & a_0 + \Delta & (a_0 + \Delta)^2 & \dots & (a_0 + \Delta)^{k-1} \\ 1 & (a_1 + \Delta) & (a_1 + \Delta)^2 & \dots & (a_1 + \Delta)^{k-1} \\ \vdots & & \ddots & & \vdots \\ 1 & (a_k + \Delta) & (a_k + \Delta)^2 & \dots & (a_k + \Delta)^{k-1} \end{pmatrix}$$

Alors, la matrice $\mathbf{V}_2\mathbf{V}_1^{-1}$ est une matrice MDS involutive.

Démonstration. On connaît déjà le caractère MDS de la matrice ainsi obtenue. Pour démontrer son caractère involutif, on va prouver la relation $\mathbf{V}_2\mathbf{V}_1^{-1}\mathbf{V}_2 = \mathbf{V}_1$. Commençons pour cela par calculer $\mathbf{V}_1^{-1}\mathbf{V}_2$. De la relation $\mathbf{V}_1^{-1}\mathbf{V}_1 = \mathbf{I}_k$, posant $\mathbf{V}_1^{-1} = (c_{i,j})_{i,j \leq k-1}$ et $\mathbf{V}_2 = (b_{i,j})_{i,j \leq k-1}$, on obtient les relations suivantes :

$$\sum_{\ell=0}^{k-1} c_{i,\ell} a_{\ell,j} = \delta_{i,\ell} \quad \text{soit} \quad \sum_{\ell=0}^{k-1} c_{i,\ell} a_{\ell}^j = \delta_{i,\ell}$$

Calculant $\mathbf{V}_1^{-1}\mathbf{V}_2$, on obtient ainsi :

$$\sum_{\ell=0}^{k-1} c_{i,\ell} b_{\ell}^j = \sum_{\ell=0}^{k-1} c_{i,\ell} (a_{\ell} + \Delta)^j$$

Le développement des binômes de Newton donne alors :

$$\sum_{\ell=0}^{k-1} \sum_{u=0}^j \binom{j}{u} c_{i,\ell} a_{\ell}^u \Delta^{j-u} = \binom{j}{u} \delta_{i,u} \Delta^{j-u}$$

Soit, sous forme matricielle :

$$\mathbf{V}_1^{-1} \mathbf{V}_2 = \begin{pmatrix} 1 & \Delta & \Delta^2 & \Delta^3 & \dots & \Delta^{n-2} & \Delta^{n-1} \\ 0 & 1 & \binom{2}{1} \Delta & \binom{3}{2} \Delta^2 & \dots & \binom{n-2}{n-3} \Delta^{n-3} & \binom{n-1}{n-2} \Delta^{n-1} \\ 0 & 0 & 1 & \binom{3}{1} \Delta & \dots & \binom{n-2}{n-4} \Delta^{n-4} & \binom{n-1}{n-3} \Delta^{n-3} \\ \vdots & & \ddots & & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 & \binom{n-1}{1} \Delta \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix}$$

Le calcul de $\mathbf{V}_2 \mathbf{V}_1^{-1} \mathbf{V}_2$ donne alors :

$$\sum_{\ell=0}^j b_i^{\ell} \binom{j}{\ell} \Delta^{j-\ell} = (b_i + \Delta)^j = a_i^j$$

Soit, sous forme matricielle :

$$\mathbf{V}_2 \mathbf{V}_1^{-1} \mathbf{V}_2 = \mathbf{V}_1$$

□

La construction tirée du Théorème 28 s'étend naturellement aux θ -polynômes pour engendrer des codes de Gabidulin. En revanche, l'involutivité n'est pas préservée par la transcription du Théorème 29 pour les θ -polynômes.

Théorème 30. *Soit θ un élément générateur de $GAL(\mathbb{F}_{q^m}/\mathbb{F}_q)$. Soient $\mathbf{V}_{\theta,1}$ et $\mathbf{V}_{\theta,2}$ les θ -matrices de Vandermonde définies par des éléments a_i et b_i formant une famille \mathbb{F}_q -libre :*

$$\mathbf{V}_{\theta,1} = \begin{pmatrix} a_0 & a_0^{[1]} & a_0^{[2]} & \dots & a_0^{[n-1]} \\ a_1 & a_1^{[1]} & a_1^{[2]} & \dots & a_1^{[n-1]} \\ \vdots & & \ddots & & \vdots \\ a_n & a_n^{[1]} & a_n^{[2]} & \dots & a_n^{[n-1]} \end{pmatrix} \text{ et } \mathbf{V}_{\theta,2} = \begin{pmatrix} b_0 & b_0^{[1]} & b_0^{[2]} & \dots & b_0^{[n-1]} \\ b_1 & b_1^{[1]} & b_1^{[2]} & \dots & b_1^{[n-1]} \\ \vdots & & \ddots & & \vdots \\ b_n & b_n^{[1]} & b_n^{[2]} & \dots & b_n^{[n-1]} \end{pmatrix}$$

Alors, la matrice $\mathbf{V}_{\theta,1} \mathbf{V}_{\theta,2}^{-1}$ est une matrice MDS.

Démonstration. La démonstration est le calque de celle du Théorème 28, à ceci près qu'on considère l'annulation d'un θ -polynôme sur un sous-espace vectoriel. Or, annuler un espace de dimension supérieure ou égale à k réclame un degré supérieur ou égal à k ou la nullité du polynôme. □

4.3.2 Explorer la récursivité

Les matrices récursives MDS involutives n'existent pas en caractéristique 2. Il est cependant possible, comme l'ont proposé Gupta, Pandey et Venkateswarlu dans [GPV17] de relâcher la définition de l'involativité pour lever l'interdiction.

Définition 46. Soit $\mathbf{M} = \mathbf{C}_g^k$ une matrice récursive MDS construite à partir d'une matrice compagnon associée à un polynôme unitaire de degré k , $g(X) \in \mathbb{F}_q[X]$. On dit que la matrice \mathbf{M} est t -involutive si $\mathbf{C}_g^t \mathbf{M}^2 = \mathbf{I}_k$.

Remarque 16. Cette piste n'est crédible que si t est petit dans un cas contraire, il est d'un intérêt faible pour l'implémentation. Il induit de plus une différence de complexité de calcul entre chiffrement et déchiffrement.

Cette généralisation ouvre la voie à la recherche de constructions directes de matrices MDS très récursives, pour lesquelles on itère plus de fois la routine du LFSR induit que la taille de la matrice. Si on relâche la condition d'involution, on peut également chercher de nouvelles matrices MDS qui pourraient être moins coûteuses à implémenter quitte à augmenter le nombre de cycles nécessaires pour les calculer.

Définition 47. Soit \mathbf{C}_g une matrice compagnon associée à un polynôme unitaire de degré k , $g(X) \in \mathbb{F}_q[X]$. On dit que la matrice \mathbf{M} est t -récursive si $\mathbf{M} = \mathbf{C}_g^{k+t}$.

Les Tableaux 4.5 et 4.6 recensent respectivement les nombres de matrices MDS t -récursives et t -involutives. Les nombres du bas correspondent aux meilleurs compte de XOR déterminés pour les paramètres respectifs.

Taille	Nombre de matrices MDS t -récursives															
	$t = 0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2×2	210 1	210 1	180 1	180 1	180 1	200 1	200 1	180 1	180 1	180 1	200 1	200 1	120 1	120 1	0	0
3×3	1980 2	1260 2	1020 2	1020 2	1020 1	900 2	1260 1	1380 1	1560 1	840 2	1080 2	840 2	960 1	1376 1	1856 1	1376 1
4×4	3660 3	840 4	600 5	900 3	900 1	600 1	1140 1	1440 4	780 3	840 1	5400 5	600 3	360 5	540 3	540 2	480 2
5×5	180 4	180 4	180 4	180 4	180 4	180 4	120 6	120 6	0	0	0	0	0	0	0	0
6×6	180 10	180 10	180 10	180 10	120 14	120 14	0	0	0	0	0	0	0	0	0	0

TABLE 4.5 – Dénombrement des matrices t -récursives MDS sur \mathbb{F}_{2^4}

Taille	Nombre de matrices MDS t -involutives															
	$t = 0$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2×2	0	10 4	0	0	0	0	10 4	0	0	0	0	90 1	0	8 1	0	0
3×3	0	0	0	0	0	0	0	4 9	0	240 2	0	8 2	0	0	0	0
4×4	0	0	0	0	0	0	0	180 4	0	8 11	0	0	0	0	0	0
5×5	0	0	0	0	0	60 4	0	8 6	0	0	0	0	0	0	0	0
6×6	0	0	0	60 10	0	8 14	0	0	0	0	0	0	0	0	0	0

TABLE 4.6 – Dénombrement des matrices t -involutives MDS sur \mathbb{F}_{2^4}

Le très grand nombre de cas recouverts par cette généralisation rend le recensement des polynômes records particulièrement fastidieux. Cette généralisation se transpose aux θ -polynômes pour engendrer un nombre significatif de matrices MDS supplémentaires. Les comptes de XOR résultants sont cependant de l'ordre de ceux des matrices θ -récursives classiques et ne compensent pas l'augmentation du nombre de cycles nécessaires pour réaliser le produit matrice-vecteur.

4.3.3 Couches de diffusion linéaire parfaites récursives

Dans la lignée des travaux de Sajadieh, Dakhilalian, Mala et Sepehrdad [Saj+12b], on peut explorer des structures récursives binaires qui engendrent des couches de diffusion linéaire avec un nombre de branchements optimal. Le terme MDS n'est alors plus légitime, on parle de couches de diffusion linéaire parfaites. Les travaux de Wu, Wang et Wu dans [WWW13] constituent la généralisation la plus globale de cette problématique. La détermination de telles couches de diffusion linéaire s'oriente vers des stratégies de recherche. Les coûts d'implémentation matérielle sont calculés comme le nombre de portes XOR nécessaire à la réalisation de l'application induite. De nombreuses généralisations sont alors possibles. Il est possible de considérer des LFSR sur \mathbb{F}_{2^m} dont les multiplications sont remplacées par des matrices inversibles quelconques. Une autre généralisation consiste à modifier plus d'un mot à chaque tour d'horloge, pour toutes les partitions de $\{1, \dots, k\}$ imaginables. La Figure 4.2 en donne un exemple pour lequel k est pair et la partition est donnée par les couples de positions consécutives. Il est toujours possible d'opérer plus de k itérations dans l'espoir de diminuer encore le compte de XOR nécessaire pour réaliser une couche de diffusion linéaire parfaite. Une dernière évolution de la structure consiste à considérer non plus des applications linéaires par mot mais un véritable registre au niveau bit. Toutes ces généralisations ouvrent la voie vers un compte de XOR minimal pour réaliser une couche de diffusion linéaire parfaite. Ces applications se limitent pour le moment à des recherches orientées et ne profitent pas de constructions directes. Elles se limitent donc à des petites tailles de paramètres. Les constructions directes ont cet avantage, pour un coût supérieur, de pouvoir générer des couches de diffusion linéaire qui maximisent sur deux tours le nombre de boîtes actives d'un schéma à plus de 16 mots.

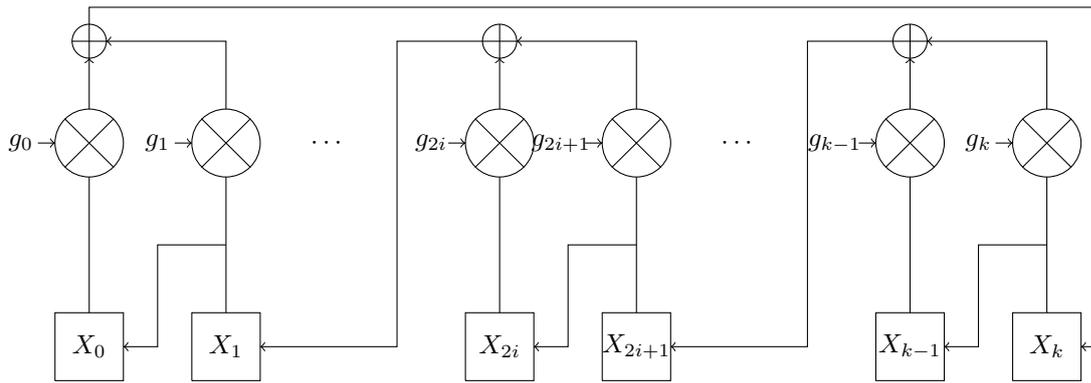


FIGURE 4.2 – Registres à décalage en série

Chapitre 5

Construction de matrices circulantes MDS et généralisations

Sommaire

5.1	Matrices circulantes	72
5.1.1	Matrices circulantes MDS involutives	72
5.1.2	Construction directe de matrices circulantes MDS involutives sur des corps de caractéristique impaire	76
5.1.3	Construction directe de matrices circulantes MDS sur des corps de caractéristique 2	78
5.2	Tordre la cyclicité	80
5.2.1	Matrices θ -circulantes MDS	80
5.2.2	Recherche exhaustive de matrices θ -circulantes MDS involutives	82
5.3	Relâcher les contraintes	84
5.3.1	Matrices cycliques MDS	84
5.3.2	Construction directe de matrices θ^2 -circulantes MDS quasi- θ -involutives	86
5.3.3	Couches de diffusion linéaire parfaites circulantes	89

L'intérêt pour les matrices circulantes est ancien et motivé par une probabilité supérieure qu'elles soient MDS à celle d'une matrice MDS sans structure, comme l'ont illustré Daemen, Knudsen et Rijmen dans [DKR97] pour des petits paramètres. Le nombre de déterminants distincts de sous-matrices carrées incluses dans une matrice circulante est en effet inférieur à celui d'une matrice sans structure particulière. Plusieurs généralisations successives sont possibles jusqu'à inclure les matrices de Hadamard. Toutefois, des matrices circulantes MDS existent avec une première ligne dont certains coefficients se répètent. Ceci peut mener à des réductions de coûts d'implémentation en comparaison des matrices de Hadamard-Cauchy qui doivent posséder au moins k entrées distinctes pour être MDS. En revanche, il a été démontré par Gupta et Ray dans [GR14] que pour les jeux de paramètres utilisés en cryptographie symétrique, il n'existe pas de matrices circulantes MDS involutives.

À l'instar des matrices θ -récursives, on peut également tisser des liens entre matrices θ -circulantes et θ -polynômes. L'absence de constructions directes de matrices circulantes MDS est contournée

pour construire astucieusement des matrices θ^2 -circulantes MDS toujours à partir des codes de Gabidulin. Grâce au canevas induit sur les polynômes qui génère les matrices circulantes, il est par ailleurs possible de préciser les conditions d'existence de matrices circulantes MDS involutives. Une construction directe émerge alors, pour les seules caractéristiques impaires. Il est toutefois possible de calquer cette méthode pour obtenir une construction directe de matrices circulantes MDS en caractéristique 2, d'un intérêt pratique avéré.

Ce chapitre introduit des relations entre matrices circulantes et polynômes dans $\mathbf{F}_q[X]/(X^n - 1)$. On peut alors déterminer les conditions d'existence des matrices circulantes involutives MDS. Une construction directe de telles matrices en caractéristique impaire est présentée, de laquelle on dérive une construction directe de matrices circulantes MDS en caractéristique 2. On dresse ensuite les mêmes relations entre les matrices θ -circulantes et les θ -polynômes. Une méthode de recherche de matrices θ -circulantes involutives MDS est présentée. Ce chapitre se conclut par la présentation des compromis possibles avec les rigidités algébriques des matrices circulantes pour construire des couches linéaires circulantes à faible coût.

5.1 Matrices circulantes

À l'instar des matrices récursives, les matrices circulantes bénéficient d'une traduction polynomiale. Cette analogie permet de caractériser les polynômes associés aux matrices circulantes MDS. Malheureusement, comme pour les matrices récursives, cette caractérisation se révèle de peu d'intérêt pour discriminer efficacement les polynômes associés aux matrices MDS. Elle permet cependant de prouver et préciser le résultat de Gupta et Ray [GR14] sur les impossibilités de trouver des matrices circulantes MDS involutives. Ces interdictions, précisées, nous mèneront, dans le cas des caractéristiques impaires, à la détermination d'un ensemble de matrices circulantes MDS involutives dont on conjecture qu'il les contient toutes. Ce travail trace un chemin vers une construction directe de matrices circulantes MDS en caractéristique 2. La caractérisation des paramètres qui l'autorisent reste vague et circonscrit une très faible part des matrices circulantes MDS.

Nous allons commencer par présenter les rigidités algébriques qui interdisent l'existence des matrices circulantes MDS involutives jusqu'à circonscire leur existence à des paramètres contraints. Nous présentons alors une construction directe de matrices circulantes involutives MDS pour certains corps de caractéristique impaire. Nous utilisons les mêmes méthodes de construction pour alors présenter une construction directe de matrices circulantes MDS en caractéristique 2.

5.1.1 Matrices circulantes MDS involutives

Gupta et Ray ont prouvé que les matrices circulantes MDS involutives n'existent pas en caractéristique 2 ou pour des tailles paires dans [GR14]. Les caractérisations de ces propriétés sur les polynômes permettent de retrouver cette interdiction et même d'en préciser les bornes. De la Proposition 17 découle la condition algébrique nécessaire et suffisante pour qu'une matrice circulante carrée soit MDS.

Proposition 20. *Soit $h(X) = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_q[X]$. Soit \mathbf{C}_h la matrice circulante associée au polynôme $h(X)$. Alors, \mathbf{C}_h est MDS si et seulement si :*

$$\forall m(X) \in \mathbb{F}_{q,k-1}[X], \quad w_H(m(X)) + w_H(c(X)) \geq k + 1.$$

où $c(X)$ le polynôme de degré inférieur à k tel que $c(X) = m(X)h(X) \bmod (X^k - 1)$.

Démonstration. Soit $h(X) = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_q[X]$ et \mathbf{C}_h sa matrice circulante associée. Alors,

$$\begin{aligned} & \mathbf{C}_h \text{ est MDS} \\ \Leftrightarrow & (\mathbf{I}_k \mid \mathbf{C}_h) \text{ est la matrice génératrice d'un code MDS} \\ \Leftrightarrow & \forall (m_0, \dots, m_{k-1}) \in \mathbb{F}_q^m, w_H((m_0, \dots, m_{k-1}) \cdot (\mathbf{I}_k \mid \mathbf{C}_h)) \geq k + 1 \\ \Leftrightarrow & \forall (m_0, \dots, m_{k-1}) \in \mathbb{F}_q^k, w_H(m_0, \dots, m_{k-1}) + w_H((m_0, \dots, m_{k-1}) \cdot \mathbf{C}_h) \geq k + 1 \end{aligned}$$

Si on considère $m(X) = \sum_{i=0}^{k-1} m_i X^i$, alors $w_H(m_0, \dots, m_{k-1}) = w_H(m(X))$. D'après la Proposition 17, \mathbf{C}_h correspond à la multiplication par $h(X)$ dans $\mathbb{F}_q[X]/(X^k - 1)$, on a alors :

$$w_H((m_0, \dots, m_{k-1}) \cdot \mathbf{C}_h) = w_H(c(X)),$$

ce qui prouve la proposition. \square

Remarque 17. Cette caractérisation est difficile à manipuler car elle ne concerne pas directement les racines mais les coefficients de tout multiple de petit degré.

Exemple 12. Soit \mathbb{F}_{2^4} défini par $T^4 + T + 1$ et α une racine de ce polynôme. La matrice \mathbf{C}_h associée à $h(X) = (X^4 + 1) + \alpha^3 X^3 + \alpha X^2 + X + 1 \in \mathbb{F}_{2^4}[X]$ est une matrice circulante MDS.

$$\mathbf{C}_h = \begin{pmatrix} 1 & 1 & \alpha & \alpha^3 \\ \alpha^3 & 1 & 1 & \alpha \\ \alpha & \alpha^3 & 1 & 1 \\ 1 & \alpha & \alpha^3 & 1 \end{pmatrix}$$

L'involutivité des matrices circulantes possède également sa transcription sur les polynômes associés, donnée par la proposition suivante :

Proposition 21. Soit $h(X) = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_q[X]$. Soit \mathbf{C}_h la matrice circulante associée à $h(X)$. Alors, \mathbf{C}_h est involutive si et seulement si $h(X)^2 = 1 \pmod{(X^k - 1)}$.

Démonstration. Cette proposition découle directement de la Proposition 17 et de la définition des matrices involutives. \square

Gupta et Ray ont prouvé l'impossibilité de trouver des matrices circulantes involutives MDS de taille paire en caractéristique 2 [GR14]. Les théorèmes suivants simplifient et étendent les preuves aux matrices circulantes MDS de tailles paires sur toute caractéristique.

Théorème 31. Soit $d \geq 2$. Il n'existe pas de matrices circulantes MDS involutives de taille $2d$ sur des corps de caractéristique $p \geq 2$.

Démonstration. Par l'absurde, supposons qu'il existe \mathbf{C}_h , une matrice circulante MDS et involutive de taille $2d$ à coefficients dans \mathbb{F}_q . Soit $h(X) \in \mathbb{F}_q[X]$ le polynôme associé à la matrice circulante \mathbf{C}_h . On va trouver un contreexemple à la Proposition 20. La preuve se scinde en deux selon la caractéristique du corps considéré :

En caractéristique 2 : On considère la caractérisation donnée par la Proposition 17. Soit $m(X) = X^d - 1$. Par hypothèse $h^2(X) - 1 = (h(X) - 1)^2 = 0 \pmod{X^{2d} - 1}$. Ainsi $h(X) - 1 = 0 \pmod{X^d - 1}$ d'où $c(X) = m(X)h(X) \pmod{X^{2d} - 1} = X^d - 1$. Alors $w_H(m(X)) + w_H(c(X)) = 4 \leq 2d + 1$. Absurde, \mathbf{C}_h n'est pas MDS.

En caractéristique impaire : Posons $b(X) = (X^{2(d-1)} + X^{2(d-2)} + \dots + 1)$ et remarquons qu'il est de poids d . Or, $X^{2d} - 1 = (X^2 - 1)b(X)$. Ainsi, $X^{2d} - 1$ est divisible par $X^2 - 1$ et on a :

$$\begin{aligned} \mathbf{C}_h \text{ involutive} &\Rightarrow (h(X))^2 = 1 \pmod{(X^{2d} - 1)} \\ &\Rightarrow (h(X))^2 = 1 \pmod{(X^2 - 1)} \\ &\Rightarrow (h(X))^2 = X^2 \pmod{(X^2 - 1)} \end{aligned}$$

On définit alors $m_1(X)$ et $m_2(X) \in \mathbb{F}_q[X]$ selon :

$$\begin{cases} (h(X) - 1)(h(X) + 1) &= m_1(X)(X^2 - 1) \\ (h(X) - X)(h(X) + X) &= m_2(X)(X^2 - 1) \end{cases}$$

On montre que $(X^2 - 1)$ divise nécessairement un des quatre polynômes suivants :

$$\{h(X) - 1, h(X) + 1, h(X) - X, h(X) + X\}$$

Supposons que $X^2 - 1$ ne divise ni $h(X) - 1$ ni $h(X) + 1$. Comme 1 et -1 sont les racines de $X^2 - 1$, on distingue deux cas :

- $h(1) = 1$ et $h(-1) = -1$.
Ceci implique que $h(X) - X$ contient les racines de $X^2 - 1$ et est ainsi divisible par $X^2 - 1$.
- $h(1) = -1$ et $h(-1) = 1$.
Ceci implique que $h(X) + X$ contient les racines de $X^2 - 1$ et est ainsi divisible par $X^2 - 1$.

Le polynôme $(X^{2d} - 1)$ divise un des quatre polynômes suivants :

$$\{b(X)(h - 1), b(X)(h + 1), b(X)(h + X), b(X)(h - X)\}$$

On sait donc qu'il existe un polynôme $m(X) \in \mathbb{F}_q[X]$ tel que :

- soit $b(X)h(X) = \pm b(X) + m(X)(X^{2d} - 1)$.
- soit $b(X)h(X) = \pm Xb(X) + m(X)(X^{2d} - 1)$

Comme le degré de $b(X)$ est inférieur à $2d - 2$, $b(X)$ ou $Xb(X)$ correspond au reste de la division de $b(X)h(X)$ par $X^{2d} - 1$. De plus, comme $b(X)$ est de poids d , $Xb(X)$ aussi, d'où :

$$w_H(b(X)) + w_H(b(X)) = w_H(b(X)) + w_H(Xb(X)) = 2d < 2d + 1.$$

Absurde, \mathbf{C}_h n'est pas MDS. □

Dans le cas particulier des caractéristique impaires, des matrices circulantes MDS involutives de taille impaire existent :

Exemple 13. *Considérons \mathbb{F}_{23} le corps à 23 éléments. Alors, \mathbf{C}_h , la matrice circulante associée au polynôme $h(X) = (X^3 - 1) + 7X^2 + 7X + 8$ est à la fois MDS et involutive :*

$$\mathbf{C}_g = \begin{pmatrix} 8 & 7 & 7 \\ 7 & 8 & 7 \\ 7 & 7 & 8 \end{pmatrix}$$

Il n'existe pas non plus de matrices circulantes de taille impaire MDS et involutives sur des corps de caractéristique 2. Ce résultat, déjà prouvé dans [GR14], reçoit ici une preuve alternative, conséquence immédiate des résultats précédents.

Théorème 32. Soit \mathbb{F}_q un corps fini de caractéristique 2. Soit $k \geq 3$ un entier impair. Il n'existe pas de matrices circulantes MDS involutives de taille k à coefficients dans \mathbb{F}_q .

Démonstration. Par l'absurde, supposons qu'il existe une matrice circulante MDS et involutive de taille k . Soit $h(X)$ le polynôme auquel cette matrice, \mathbf{C}_h , est alors associée. Soit $g'(X)$ la dérivée de $g(X) = (X^k - 1)$, alors

$$g'(X) = kX^{k-1}.$$

Le polynôme $g'(X)$ est premier avec $g(X)$. Ainsi $g(X)$ possède exactement k racines distinctes. Par hypothèse, on a $h(X)^2 = 1 \pmod{X^k - 1}$ avec $h(X)$ de degré k . Ainsi il existe un polynôme $m(X)$ tel que :

$$(h(X) + 1)^2 = m(X)(X^k - 1).$$

Ainsi, les k racines de $X^k - 1$ sont également racines de $(h(X) + 1)^2$. Racines simples, elles sont alors racines de $h(X) + 1$ et $h(X) + 1 = 0 \pmod{X^k - 1}$. Comme $h(X)$ est unitaire et de degré k , nécessairement $h(X) = X^k$. Absurde, \mathbf{C}_h n'est pas MDS. \square

En caractéristique 2, les seules matrices circulantes MDS involutives sont de taille 1 ou 2. L'exemple suivant en prouve l'existence :

Exemple 14. Soit \mathbb{F}_{2^4} défini par $T^4 + T + 1$ et α une racine de ce polynôme. La matrice \mathbf{C}_h associée à $h(X) = (X^2 + 1) + \alpha^4 X + \alpha$ est à la fois MDS et involutive.

$$\mathbf{C}_h = \begin{pmatrix} \alpha & \alpha^4 \\ \alpha^4 & \alpha \end{pmatrix}$$

On a $h(X)^2 = X^4 + \alpha^8 X^2 + \alpha^8 = X^4 \pmod{X^2 + 1} = X^2 \pmod{X^2 + 1} = 1 \pmod{X^2 + 1}$

En caractéristique impaire, de fortes contraintes sur les paramètres conditionnent l'existence de matrices circulantes MDS involutives.

Théorème 33. Soit $k = 2d + 1 \geq 3$. Soit \mathbb{F}_q un corps de caractéristique impaire. S'il existe des matrices circulantes MDS involutives de taille $k \times k$, alors il existe $a(X)$ et $b(X) \in \mathbb{F}_q[X]$ tels que $a(X)$ est de degré d , $\text{pgcd}(a(X), b(X)) = 1$ et $X^k - 1 = a(X)b(X)$.

Démonstration. Par l'absurde, supposons qu'il existe une matrice circulante MDS et involutive de taille k , dont le polynôme associé est noté $h(X)$ et que $X^k - 1$ n'admet pas de décomposition $X^k - 1 = a(X)b(X)$ avec $a(X) \in \mathbb{F}_q[X]$ de degré d .

Par hypothèse, $h^2(X) = 1 \pmod{X^k - 1}$. Ainsi, il existe un polynôme $m(X)$ tel que :

$$(h(X) - 1)(h(X) + 1) = m(X)(X^k - 1)$$

En caractéristique impaire, $(h(X)+1)$ et $(h(X)-1)$ sont premiers entre eux. $\text{pgcd}(X^k-1, h(X)-1)$ ou $\text{pgcd}(X^k-1, h(X)+1)$ est alors de degré au moins $d+2$. Il existe donc $c(X) \in \mathbb{F}_q[X]$ de degré au plus $d-1$ tel que $c(X)(h(X)+1)$ ou $c(X)(h(X)-1)$ est divisible par X^k-1 .

Pour des raisons de degré, $w_H(c(X)) \leq d$. Ainsi, $w_H(c(X)) + w_H(c(X)) \leq 2d < 2d+1$ et $c(X)h(X) = -c(X) \pmod{X^k-1}$ ou $c(X)h(X) = c(X) \pmod{X^k-1}$. Absurde, \mathbf{C}_h n'est pas MDS. \square

5.1.2 Construction directe de matrices circulantes MDS involutives sur des corps de caractéristique impaire

Le Théorème 33 a précisé les conditions d'existence de matrices circulantes MDS et involutives. Une construction directe de telles matrices peut alors être introduite. Cette construction n'a aucune portée pratique dans la mesure où les corps considérés ne peuvent être que de caractéristique impaire. Cependant, elle constitue véritablement une plongée dans les relations qu'entretiennent certaines matrices circulantes MDS avec les codes cycliques et est un premier pas vers la construction directe de matrices circulantes MDS proposée ensuite.

Soient \mathbb{F}_q et $k = 2d + 1$ tels que $X^k - 1$ soit simplement scindé. On construit à partir de α racine primitive k -ième de l'unité arbitraire et ℓ , entier arbitraire, les deux polynômes $a(X)$ et $b(X)$ de degrés respectifs d et $d + 1$ de la façon suivante :

$$\begin{cases} a(X) &= \prod_{i=0}^{d-1} (X - \alpha^{\ell+i}) \\ b(X) &= \prod_{i=0}^d (X - \alpha^{\ell+d+i}) \end{cases}$$

Construisons alors l'unique polynôme $h(X)$ unitaire de degré k qui vérifie le système :

$$\begin{cases} h(X) - 1 &= u(X)a(X) \\ h(X) + 1 &= v(X)b(X) \end{cases} \Leftrightarrow \begin{cases} 2h(X) &= u(X)a(X) + v(X)b(X) \\ 2 &= v(X)b(X) - u(X)a(x) \end{cases}$$

Ce choix de polynôme assure l'involutivité de la matrice circulante associée au polynôme $h(X)$ attendu que $(h(X) - 1)(h(X) + 1) = u(X)v(X)a(X)b(X) = u(X)v(X)(X^k - 1) = 0 \pmod{(X^k - 1)}$.

On veut alors s'assurer du caractère MDS de la matrice circulante associée au polynôme $h(X)$. Pour cela, on considère la matrice \mathbf{H}_1 suivante :

$$\mathbf{H}_1 = \left(\begin{array}{cccc|cccc} 1 & \alpha^\ell & \dots & \alpha^{(k-1)\ell} & 1 & \alpha^\ell & \dots & \alpha^{(k-1)\ell} \\ 1 & \alpha^{\ell+1} & \dots & \alpha^{(k-1)(\ell+1)} & 1 & \alpha^{\ell+1} & \dots & \alpha^{(k-1)(\ell+1)} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha^{\ell+d-1} & \dots & \alpha^{(k-1)(\ell+d-1)} & 1 & \alpha^{\ell+d-1} & \dots & \alpha^{(k-1)(\ell+d-1)} \\ \hline 1 & \alpha^{\ell+d} & \dots & \alpha^{(k-1)(\ell+d)} & -1 & -\alpha^{\ell+d} & \dots & -\alpha^{(k-1)(\ell+d)} \\ 1 & \alpha^{\ell+d+1} & \dots & \alpha^{(k-1)(\ell+d+1)} & -1 & -\alpha^{\ell+d+1} & \dots & -\alpha^{(k-1)(\ell+d+1)} \\ \vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha^{\ell+2d} & \dots & \alpha^{(k-1)(\ell+2d)} & -1 & -\alpha^{\ell+2d} & \dots & -\alpha^{(k-1)(\ell+2d)} \end{array} \right)$$

Cette matrice est une matrice de parité du code associé à la matrice génératrice sous forme systématique $(\mathbf{C}_h \mid \mathbf{I}_k)$ qui se réécrit de la façon suivante :

$$\left(\begin{array}{cc|cccc} h(X) & \text{mod } (X^k - 1) & 1 & 0 & \dots & 0 \\ Xh(X) & \text{mod } (X^k - 1) & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ X^{k-1}h(X) & \text{mod } (X^k - 1) & 0 & 0 & \dots & 1 \end{array} \right)$$

Par construction, on a en effet :

$$\begin{aligned}
& \begin{cases} h(\alpha^{\ell+i}) &= 1, \quad \forall i \in \{0, \dots, d-1\} \\ h(\alpha^{\ell+d+i}) &= -1, \quad \forall i \in \{0, \dots, d\} \end{cases} \\
\Rightarrow & \begin{cases} \sum_{j=0}^{k-1} h_j \alpha^{(\ell+i)j} &= (\alpha^{k(\ell+i)} - 1) + 1, \quad \forall i \in \{0, \dots, d-1\} \\ \sum_{j=0}^{k-1} h_j \alpha^{(\ell+d+i)j} &= (\alpha^{k(\ell+d+i)} - 1) - 1, \quad \forall i \in \{0, \dots, d\} \end{cases} \\
\Rightarrow & \begin{cases} \sum_{j=0}^{k-1} h_j \alpha^{(\ell+i)(j+u)} &= \alpha^{(\ell+i)u}, \quad \forall i \in \{0, \dots, d-1\}, \quad \forall u \in \{0, \dots, k-1\} \\ \sum_{j=0}^{k-1} h_j \alpha^{(\ell+d+i)(j+u)} &= -\alpha^{(\ell+d+i)u}, \quad \forall i \in \{0, \dots, d\}, \quad \forall u \in \{0, \dots, k-1\} \end{cases}
\end{aligned}$$

Considérons alors la matrice \mathbf{H}_2 obtenue par permutation des colonnes $2i$ et $k+2i$ de la matrice de parité \mathbf{H}_1 pour tout $i \in \{1, \dots, \frac{k-1}{2}\}$. C'est encore une matrice de parité d'un code avec la même distance minimale que le code qui admet \mathbf{H}_1 comme matrice de parité.

$$\mathbf{H}_2 = \left(\begin{array}{cccc|cccc}
1 & \alpha^\ell & \dots & \alpha^{(k-1)\ell} & 1 & \alpha^\ell & \dots & \alpha^{(k-1)\ell} \\
1 & \alpha^{\ell+1} & \dots & \alpha^{(k-1)(\ell+1)} & 1 & \alpha^{\ell+1} & \dots & \alpha^{(k-1)(\ell+1)} \\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
1 & \alpha^{\ell+d-1} & \dots & \alpha^{(k-1)(\ell+d-1)} & 1 & \alpha^{\ell+d-1} & \dots & \alpha^{(k-1)(\ell+d-1)} \\
1 & -\alpha^{\ell+d} & \dots & \alpha^{(k-1)(\ell+d)} & -1 & \alpha^{\ell+d} & \dots & -\alpha^{(k-1)(\ell+d)} \\
1 & -\alpha^{\ell+d+1} & \dots & \alpha^{(k-1)(\ell+d+1)} & -1 & \alpha^{\ell+d+1} & \dots & -\alpha^{(k-1)(\ell+d+1)} \\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots \\
1 & -\alpha^{\ell+2d} & \dots & \alpha^{(k-1)(\ell+2d)} & -1 & \alpha^{\ell+2d} & \dots & -\alpha^{(k-1)(\ell+2d)}
\end{array} \right)$$

Enfin, considérons \mathbf{H}_3 , obtenue par permutation des lignes de \mathbf{H}_2 . C'est une matrice de parité pour le même code. Sa structure interne apparaît alors :

$$\mathbf{H}_3 = \left(\begin{array}{cccccc}
1 & \alpha^\ell & \alpha^{2\ell} & \dots & \alpha^{\ell(2k-1)} \\
1 & (-\alpha^{d+1})\alpha^\ell & (-\alpha^{d+1})^2\alpha^{2\ell} & \dots & (-\alpha^{d+1})^{2k-1}\alpha^{\ell(2k-1)} \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & (-\alpha^{2d(d+1)})\alpha^\ell & (-\alpha^{2d(d+1)})^2\alpha^{2\ell} & \dots & (-\alpha^{2d(d+1)})^{2k-1}\alpha^{\ell(2k-1)}
\end{array} \right)$$

On remarque alors que \mathbf{H}_3 est la matrice de parité d'un code BCH MDS de dimension k et de longueur $2k$. \mathbf{H}_1 est alors également la matrice de parité d'un code MDS et la matrice circulante associée à $h(X)$ est donc MDS et involutive par construction.

Sur les petits paramètres, pour lesquels il reste possible de considérer par recherche exhaustive toutes les matrices circulantes MDS involutives, toutes les matrices obtenues partageaient cette structure. On formule alors la conjecture suivante :

Conjecture 3. *Les seules matrices circulantes MDS involutives sont celles obtenues par la construction précédente, chaque fois qu'il existe une factorisation $X^k - 1 = a(X)b(X)$ avec les racines de a et de b respectivement des puissances successives d'une racine primitive de l'unité.*

Cette construction directe, outre qu'elle nous rapproche d'une caractérisation complète des matrices circulantes MDS involutives, fait apparaître des liens entre certaines matrices circulantes et certains codes cycliques, ce qui était inattendu et remarquable. La détermination exacte des paramètres k et q pour lesquels une telle factorisation existe semble non triviale.

5.1.3 Construction directe de matrices circulantes MDS sur des corps de caractéristique 2

On prolonge ici les réflexions nées de la construction directe de matrices circulantes MDS involutives pour certains paramètres dans des corps de caractéristique impaire. On revient maintenant à la caractéristique 2 pour construire directement des matrices circulantes MDS pour des tailles de matrices impaires.

Soit $k = 2d + 1$. Soit m tel que $X^k - 1$ soit scindé à racines simples dans \mathbb{F}_{2^m} et soient $\{\lambda_1, \dots, \lambda_k\}$ ces racines. Considérons le code engendré par la matrice génératrice $(\mathbf{C}_h \mid \mathbf{I}_k)$, pour $h(X)$ un certain polynôme unitaire de degré k . Alors, en posant $a_i = h(\lambda_i)$, on en construit la matrice de parité \mathbf{H}_1 suivante :

$$\mathbf{H}_1 = \left(\begin{array}{cccc|cccc} 1 & \lambda_1 & \lambda_1^2 & \dots & \lambda_1^{k-1} & a_1 & \lambda_1 a_1 & \lambda_1^2 a_1 & \dots & \lambda_1^{k-1} a_1 \\ 1 & \lambda_2 & \lambda_2^2 & \dots & \lambda_2^{k-1} & a_2 & \lambda_2 a_2 & \lambda_2^2 a_2 & \dots & \lambda_2^{k-1} a_2 \\ \vdots & & \vdots & & \vdots & \vdots & & \vdots & & \vdots \\ 1 & \lambda_k & \lambda_k^2 & \dots & \lambda_k^{k-1} & a_k & \lambda_k a_k & \lambda_k^2 a_k & \dots & \lambda_k^{k-1} a_k \end{array} \right)$$

On ne peut pas reproduire exactement la procédure de la construction directe de matrices circulantes MDS involutives puisqu'en caractéristique 2, la seule racine carrée de 1 est 1. Alors, on va remarquer que le code qui possède \mathbf{H}_1 comme matrice de parité est un code raccourci du code de matrice de parité \mathbf{H}_2 :

$$\mathbf{H}_2 = \left(\begin{array}{cccc|cccc|cccc} 1 & \lambda_1 & \dots & \lambda_1^{k-1} & a_1 & \lambda_1 a_1 & \dots & \lambda_1^{k-1} a_1 & a_1^2 & \lambda_1 a_1^2 & \dots & \lambda_1^{k-1} a_1^2 \\ 1 & \lambda_2 & \dots & \lambda_2^{k-1} & a_2 & \lambda_2 a_2 & \dots & \lambda_2^{k-1} a_2 & a_2^2 & \lambda_2 a_2^2 & \dots & \lambda_2^{k-1} a_2^2 \\ \vdots & & & \vdots & \vdots & & & \vdots & \vdots & & & \vdots \\ 1 & \lambda_k & \dots & \lambda_k^{k-1} & a_k & \lambda_k a_k & \dots & \lambda_k^{k-1} a_k & a_k^2 & \lambda_k a_k^2 & \dots & \lambda_k^{k-1} a_k^2 \end{array} \right)$$

La matrice génératrice naturellement associée à ce code est donc donnée par :

$$\left(\begin{array}{cc|ccc} h(X) & \text{mod } (X^k - 1) & 1 & 0 & \dots & 0 \\ Xh(X) & \text{mod } (X^k - 1) & 0 & 1 & \dots & 0 \\ & \vdots & \vdots & \ddots & \ddots & \vdots \\ X^{2k-1}h(X) & \text{mod } (X^k - 1) & 0 & 0 & \dots & 1 \end{array} \right)$$

On peut alors directement s'inspirer de la construction précédente pour engendrer un code BCH MDS de paramètres $[3k, 2k]$ qu'on raccourcit dans les k dernières positions pour obtenir un code de paramètres $[2k, k]$ MDS. On souhaite alors que les a_i soient des racine 3-ième de l'unité. La construction directe se conçoit alors comme suit :

Soit $k = 2d + 1 \geq 5$ premier avec 3. Soit m tel que $X^k - 1$ et $X^3 - 1$ soient tous les deux scindés à racines simples dans \mathbb{F}_{2^m} . Soit α une racine primitive k -ième de l'unité et β une racine primitive 3-ième de l'unité. Alors, $\alpha\beta$ est une racine primitive $3k$ -ième de l'unité. On sait que le polynôme engendré par les racines $\{(\alpha\beta)^i : i \in [1..k]\}$ engendre un code BCH MDS de paramètres $[3k, 2k]$. Alors, le raccourcissement de ce code est également MDS et assure que la matrice de parité \mathbf{H}_1 est également MDS. Ainsi, la matrice circulante naturellement associée \mathbf{C}_h est MDS.

Exemple 15. Soit $k = 5$. Considérons \mathbb{F}_{2^4} engendré par le polynôme $T^4 + T + 1$ et soit α une de ses racines. α^3 est une racine primitive 5-ième de l'unité et α^5 est une racine primitive

3-ième de l'unité. La théorie des codes nous assure que les matrices de parité \mathbf{H}_1 et \mathbf{H}_2 engendrent respectivement des codes $[15, 10]$ et $[10, 5]$ MDS, avec respectivement

$$\mathbf{H}_1 = \begin{pmatrix} 1 & \alpha^8 & \alpha & \alpha^9 & \alpha^2 & \alpha^{10} & \alpha^3 & \alpha^{11} & \alpha^4 & \alpha^{12} & \alpha^5 & \alpha^{13} & \alpha^6 & \alpha^{14} & \alpha^7 \\ 1 & \alpha & \alpha^2 & \alpha^3 & \alpha^4 & \alpha^5 & \alpha^6 & \alpha^7 & \alpha^8 & \alpha^9 & \alpha^{10} & \alpha^{11} & \alpha^{12} & \alpha^{13} & \alpha^{14} \\ 1 & \alpha^9 & \alpha^3 & \alpha^{12} & \alpha^6 & 1 & \alpha^9 & \alpha^3 & \alpha^{12} & \alpha^6 & 1 & \alpha^9 & \alpha^3 & \alpha^{12} & \alpha^6 \\ 1 & \alpha^2 & \alpha^4 & \alpha^6 & \alpha^8 & \alpha^{10} & \alpha^{12} & \alpha^{14} & \alpha & \alpha^3 & \alpha^5 & \alpha^7 & \alpha^9 & \alpha^{11} & \alpha^{13} \\ 1 & \alpha^{10} & \alpha^5 & 1 & \alpha^{10} & \alpha^5 \end{pmatrix}$$

$$\mathbf{H}_2 = \begin{pmatrix} 1 & \alpha^3 & \alpha^6 & \alpha^9 & \alpha^{12} & \alpha^5 & \alpha^8 & \alpha^{11} & \alpha^{14} & \alpha^2 \\ 1 & \alpha^6 & \alpha^{12} & \alpha^3 & \alpha^9 & \alpha^{10} & \alpha & \alpha^7 & \alpha^{13} & \alpha^4 \\ 1 & \alpha^9 & \alpha^3 & \alpha^{12} & \alpha^6 & 1 & \alpha^9 & \alpha^3 & \alpha^{12} & \alpha^6 \\ 1 & \alpha^{12} & \alpha^9 & \alpha^6 & \alpha^3 & \alpha^5 & \alpha^2 & \alpha^{14} & \alpha^{11} & \alpha^8 \\ 1 & 1 & 1 & 1 & 1 & \alpha^{10} & \alpha^{10} & \alpha^{10} & \alpha^{10} & \alpha^{10} \end{pmatrix}$$

La matrice $\mathbf{G} = (\mathbf{C}_h \mid \mathbf{I}_5)$ est alors une matrice génératrice du code associé à la matrice de parité \mathbf{H}_2 , avec la matrice circulante \mathbf{C}_h pour matrice de redondance :

$$\mathbf{G} = \begin{pmatrix} 1 & \alpha^{14} & \alpha & \alpha^4 & \alpha^{11} & 1 & 0 & 0 & 0 & 0 \\ \alpha^{11} & 1 & \alpha^{14} & \alpha & \alpha^4 & 0 & 1 & 0 & 0 & 0 \\ \alpha^4 & \alpha^{11} & 1 & \alpha^{14} & \alpha & 0 & 0 & 1 & 0 & 0 \\ \alpha & \alpha^4 & \alpha^{11} & 1 & \alpha^{14} & 0 & 0 & 0 & 1 & 0 \\ \alpha^{14} & \alpha & \alpha^4 & \alpha^{11} & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

La matrice circulante associée au polynôme $h(X) = (X^5 - 1) + \alpha^{11}X^4 + \alpha^4X^3 + \alpha X^2 + \alpha^{14}X + 1$ est alors MDS.

Cette construction demeure circonscrite à des paramètres très restrictifs et n'est pas à l'origine des nombreuses matrices disponibles pour ces paramètres. Elle constitue cependant la première construction directe de matrice circulante MDS à notre connaissance. Pour rentrer plus avant dans ces constructions, il s'agit d'appréhender les matrices de parité qui traduisent les liens algébriques avec les polynômes lorsque $X^k - 1$ possède des racines multiples.

Le Tableau 5.1 recense le nombre de matrices circulantes MDS ainsi que le meilleur compte de XOR obtenu pour les jeux de paramètres possibles sur \mathbb{F}_{2^4} .

Taille	Nombre de matrices MDS	Meilleur Compte de XOR
2×2	210	1
3×3	2250	1
4×4	16560	3
5×5	79800	4
6×6	2160	12

TABLE 5.1 – Dénombrement des matrices circulantes MDS sur \mathbb{F}_{2^4}

Le Tableau 5.2 recense les polynômes qui engendrent les meilleurs comptes de XOR pour les matrices circulantes MDS sur \mathbb{F}_{2^4} . Tous ces résultats ont été obtenus lorsque \mathbb{F}_{2^4} est engendré par $T^4 + T^3 + 1$.

Taille	Polynôme record pour les matrices MDS
2×2	$(X^2 - 1) + \alpha^{14}X + 1$
3×3	$(X^3 - 1) + \alpha^{14}X^2 + X + 1$
4×4	$(X^4 - 1) + \alpha X^3 + \alpha^{13}X^2 + X + 1$
5×5	$(X^5 - 1) + \alpha X^4 + \alpha^{14}X^3 + \alpha^{14}X^2 + \alpha X + 1$
6×6	$(X^6 - 1) + \alpha X^5 + \alpha^9 X^4 + \alpha X^3 + \alpha^{11}X^2 + X + 1$

TABLE 5.2 – Polynômes générateurs de matrices circulantes MDS sur \mathbb{F}_{2^4}

Remarque 18. *On constate qu'il n'existe pas de matrices circulantes MDS pour des tailles de matrices égales à 7 sur \mathbb{F}_{2^4} . Une telle observation laisse imaginer qu'il existe une explication en termes de polynôme de ce phénomène sans qu'on ait été en mesure de la déterminer.*

5.2 Tordre la cyclicité

L'association naturelle entre matrices circulantes et polynômes a permis de mettre à jour les nombreuses lois qui régissent l'existence de matrices circulantes MDS et de matrices circulantes MDS et involutives. Comme cela a été fait pour les matrices récursives, on peut de nouveau considérer l'association naturelle avec les polynômes de Ore non commutatifs. Celle-ci fait apparaître des matrices θ -circulantes, structure proche de celle des matrices circulantes. Les rapports entre polynômes et racines, distendus pour les θ -polynômes vont libérer certaines contraintes qui rigidifient les conditions d'existence de matrices circulantes MDS involutives.

Nous allons introduire les matrices θ -circulantes avant de définir les propriétés algébriques qui conditionnent l'existence de matrices θ -circulantes involutives MDS. Ensuite, nous donnerons des méthodes de recherches intelligentes de ces matrices.

5.2.1 Matrices θ -circulantes MDS

Le calque de l'association entre matrices circulantes et polynômes de $\mathbb{F}_q[X]$ pour les θ -polynômes de $\mathbb{F}_{q^m}[X; \theta]$ fait naître les matrices θ -circulantes :

Définition 48. *Soit $h\langle X \rangle = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_{q^m}[X; \theta]$ un θ -polynôme unitaire de degré k . La matrice θ -circulante associée à $h\langle X \rangle$, $\mathbf{C}_{h,\theta}$ est la matrice définie par :*

$$\mathbf{C}_{h,\theta} = \begin{pmatrix} h_0 & h_1 & \dots & h_{k-1} \\ h_{k-1}^{[1]} & h_0^{[1]} & \dots & h_{k-2}^{[1]} \\ \vdots & \ddots & \ddots & \vdots \\ h_1^{[k-1]} & \dots & h_{k-1}^{[k-1]} & h_0^{[k-1]} \end{pmatrix}$$

Remarque 19. *A la différence des matrices récursives, on va se restreindre à une extension de corps de degré $m = k$ pour que les opérations, notamment les multiplications à droite par $h\langle X \rangle$, dans $\mathbb{F}_{q^m}[X; \theta]/(X^k - 1)$ soient bien définies.*

La proposition suivante est la pierre angulaire sur laquelle repose toute notre analyse des matrices θ -circulantes :

Proposition 22. Soit $h\langle X \rangle = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_{q^k}[X; \theta]$ un θ -polynôme unitaire de degré k et $\mathbf{C}_{h,\theta}$ la matrice θ -circulante associée à $h\langle X \rangle$. Alors, la matrice $\mathbf{C}_{h,\theta}$ est la matrice, dans la base canonique de $\mathbb{F}_{q^k}[X; \theta]/(X^k - 1)$, de l'application :

$$\begin{aligned} \psi : \mathbb{F}_{q^k}[X; \theta]/(X^k - 1) &\rightarrow \mathbb{F}_{q^k}[X; \theta]/(X^k - 1) \\ m\langle X \rangle &\mapsto m\langle X \rangle h\langle X \rangle \end{aligned}$$

Démonstration. Par linéarité, il suffit de le vérifier pour les éléments $\{e_1, \dots, e_k\}$ de la base canonique de $\mathbb{F}_{q^k}[X; \theta]/(X^k - 1)$:

$$\forall j \in \{0, \dots, k-1\}, \begin{cases} X^j \cdot h\langle X \rangle = \sum_{i=0}^k X^j h_i X^i = \sum_{i=0}^{k-j-1} h_i^{[j]} X^{i+j} + \sum_{i=1}^j h_{k-j-1+i}^{[j]} X^i \\ e_j \cdot \mathbf{C}_{h,\theta} = (h_{k-j}^{[j]}, \dots, h_{k-1}^{[j]}, h_0^{[j]}, \dots, h_{k-j-1}^{[j]}) \end{cases}$$

□

Comme dans le cas des polynômes classiques, on établit la condition algébrique nécessaire et suffisante sur les θ -polynômes pour caractériser ceux associés à des matrices θ -circulantes MDS.

Proposition 23. Soit $h\langle X \rangle = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_{q^k}[X; \theta]$. Soit $\mathbf{C}_{h,\theta}$ la matrice θ -circulante associée à $h\langle X \rangle$. Alors, $\mathbf{C}_{h,\theta}$ est MDS si et seulement si :

$$\forall m \in \mathbb{F}_{q^k, k-1}[X; \theta], \quad w_H(m\langle X \rangle) + w_H(c\langle X \rangle) \geq k + 1$$

où $c\langle X \rangle$ le θ -polynôme de degré inférieur à k tel que $c\langle X \rangle = m\langle X \rangle h\langle X \rangle \pmod{*(X^k - 1)}$

Démonstration. Soit $h\langle X \rangle = (X^k - 1) + \sum h_i X^i \in \mathbb{F}_{q^k}[X; \theta]$ et $\mathbf{C}_{h,\theta}$ sa matrice θ -circulante associée. Alors,

$$\begin{aligned} &\mathbf{C}_{h,\theta} \text{ est MDS} \\ \Leftrightarrow &(\mathbf{I}_k | \mathbf{C}_{h,\theta}) \text{ est la matrice génératrice d'un code MDS} \\ \Leftrightarrow &\forall (m_0, \dots, m_{k-1}) \in \mathbb{F}_{q^k}^k, w_H((m_0, \dots, m_{k-1}) \cdot (\mathbf{I}_k | \mathbf{C}_{h,\theta})) \geq k + 1 \\ \Leftrightarrow &\forall (m_0, \dots, m_{k-1}) \in \mathbb{F}_{q^k}^k, w_H(m_0, \dots, m_{k-1}) + w_H((m_0, \dots, m_{k-1}) \cdot \mathbf{C}_{h,\theta}) \geq k + 1 \end{aligned}$$

Si on considère $m\langle X \rangle = \sum_{i=0}^{k-1} m_i X^i$, alors $w_H(m_0, \dots, m_{k-1}) = w_H(m\langle X \rangle)$. D'après la Proposition 22, on sait que $\mathbf{C}_{h,\theta}$ correspond à la multiplication à droite par $h\langle X \rangle$ dans $\mathbb{F}_{q^k}[X; \theta]/(X^k - 1)$, on a donc :

$$w_H((m_0, \dots, m_{k-1}) \cdot \mathbf{C}_{h,\theta}) = w_H(c\langle X \rangle),$$

avec $c\langle X \rangle$ le θ -polynôme de degré inférieur à k tel que $c\langle X \rangle = m\langle X \rangle h\langle X \rangle \pmod{*(X^k - 1)}$, ce qui prouve la proposition. □

Exemple 16. Soit \mathbb{F}_{2^4} défini par $T^4 + T + 1$ et α une racine de ce polynôme. Soit θ l'automorphisme de Frobenius défini par $a \mapsto a^2$. La matrice $\mathbf{C}_{h,\theta}$ associée à $h\langle X \rangle = (X^4 + 1) + \alpha^{10} X^3 + \alpha X^2 + X + 1 \in \mathbb{F}_{2^4}[X; \theta]$ est une matrice θ -circulante MDS.

$$\mathbf{C}_{h,\theta} = \begin{pmatrix} 1 & 1 & \alpha & \alpha^{10} \\ \alpha^5 & 1 & 1 & \alpha^2 \\ \alpha^4 & \alpha^{10} & 1 & 1 \\ 1 & \alpha^8 & \alpha^5 & 1 \end{pmatrix}$$

Remarque 20. Cette caractérisation algébrique reste difficile à manipuler, l'annulation des coefficients d'un multiple d'un θ -polynôme reste un critère abscons.

La proposition suivante donne une condition algébrique simple nécessaire et suffisante sur les θ -polynômes dont les matrices θ -circulantes associées sont involutives.

Proposition 24. Soit $h\langle X \rangle = (X^k - 1) + \sum_{i=0}^{k-1} h_i X^i \in \mathbb{F}_{q^k}[X; \theta]$ et $\mathbf{C}_{h,\theta}$ sa matrice θ -circulante associée. Alors, $\mathbf{C}_{h,\theta}$ est involutive si et seulement si $g\langle X \rangle \cdot g\langle X \rangle = 1 \pmod{*(X^m - 1)}$.

Démonstration. La démonstration est l'application directe de la Proposition 22 et de la définition des matrices involutives. \square

Il est alors possible, à la différence des matrices circulantes, de trouver des matrices θ -circulantes MDS et involutives.

Exemple 17. Soit \mathbb{F}_{2^4} défini par $T^4 + T + 1$ et α une racine de ce polynôme. Soit θ l'automorphisme du Frobenius. La matrice $\mathbf{C}_{h,\theta}$ associée à $h\langle X \rangle = (X^4 + 1) + \alpha^7 X^3 + \alpha^{14} X^2 + X + \alpha \in \mathbb{F}_{2^4}[X; \theta]$ est une matrice θ -circulante MDS involutive.

$$\mathbf{C}_{h,\theta} = \begin{pmatrix} \alpha & 1 & \alpha^{14} & \alpha^7 \\ \alpha^{14} & \alpha^2 & 1 & \alpha^{13} \\ \alpha^{11} & \alpha^{13} & \alpha^4 & 1 \\ 1 & \alpha^7 & \alpha^{11} & \alpha^8 \end{pmatrix}$$

Les propositions 23 et 24 ne sont valables que pour l'extension de corps de degré la taille de la matrice. Elles ne sont plus valables pour les autres tailles de matrices. Cela ne traduit pourtant pas l'inexistence de telles matrices. Tout simplement, elles ne possèdent pas de description polynomiale aussi simple. Les recherches de matrices dans ces paramètres non pris en compte dans cette étude relèvent alors de la recherche exhaustive.

5.2.2 Recherche exhaustive de matrices θ -circulantes MDS involutives

Il est possible, pour des petits paramètres, de construire toutes les matrices θ -circulantes involutives dont on peut vérifier ensuite le caractère MDS. Pour ce faire, on part des θ -polynômes qui engendrent des matrices θ -circulantes involutives. La proposition suivante donne une piste de départ intéressante pour les parcourir par recherche exhaustive :

Proposition 25. Soit θ un élément générateur de $\text{GAL}(\mathbb{F}_{q^k}/\mathbb{F}_q)$. Si $g\langle X \rangle$ θ -polynôme unitaire de degré k dans \mathbb{F}_{2^k} tel que $g_0 \neq 0$ et tel qu'il engendre une matrice θ -circulante involutive, alors $g\langle X \rangle + 1$ annule un sous-espace vectoriel de dimension $\ell \geq \frac{k}{2}$.

Démonstration. On sait d'après le Théorème 6 que l'espace racine est un \mathbb{F}_q -espace vectoriel de dimension maximale k . Par l'absurde supposons que $g\langle X \rangle$ annule un sous-espace vectoriel de \mathbb{F}_{2^k} de dimension inférieure à $\frac{k}{2}$. Alors, l'image de \mathbb{F}_{q^k} par l'opération d'évaluation correspond à un sous-espace de \mathbb{F}_{2^k} de dimension supérieure ou égale à $\frac{k}{2}$. Une deuxième opération d'évaluation ne permet alors pas d'annuler cet espace de dimension trop importante et le θ -polynôme ne peut pas engendrer de matrice θ -circulante involutive. Absurde. \square

La stratégie de recherche est donc la suivante : on prend tous les sous-espaces vectoriels de dimension supérieure ou égale à $\frac{k}{2}$. On interpole le reste en l'envoyant sur une famille libre du noyau. Alors, on dispose de $g(X) + 1$. On peut alors en vérifier le caractère MDS.

$$(g_0, g_1, \dots, g_{k-1}) \cdot \begin{pmatrix} \lambda_1 & \lambda_1^{[1]} & \dots & \lambda_1^{[k-1]} \\ \lambda_2 & \lambda_2^{[1]} & \dots & \lambda_2^{[k-1]} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_k & \lambda_k^{[1]} & \dots & \lambda_k^{[k-1]} \end{pmatrix} = (0, \dots, 0, u_1, \dots, u_{k-\ell})$$

Le Tableau 5.3 recense le nombre de matrices θ -circulantes MDS ainsi que le meilleur compte de XOR obtenu pour les jeux de paramètres possibles sur \mathbb{F}_{2^4} .

Taille	Nombre de matrices θ -circulantes MDS		Meilleur Compte de XOR	
	simples	θ -Involutives	simples	θ -Involutives
2×2	510	10	3	11
3×3	5670	0	5	
4×4	24000	160 (160 θ^2 -involutives)	5	9 (et 9)
5×5	300	0	19	
6×6	5400	0	19	
7×7	0			

TABLE 5.3 – Dénombrement des matrices θ -circulantes MDS sur \mathbb{F}_{2^4}

Le Tableau 5.4 recense les polynômes qui engendrent les meilleurs comptes de XOR pour les matrices circulantes MDS sur \mathbb{F}_{2^4} . Tous ces résultats ont été obtenus lorsque \mathbb{F}_{2^4} est engendré par $T^4 + T^3 + 1$, dans la base normale associée à l'élément α^3 .

Taille	Polynôme record pour les matrices θ -circulantes MDS	
	simples	involutives
2×2	$(X^2 + 1) + \alpha^3 X + 1$	$(X^2 + 1) + \alpha^7 X + \alpha^5$ ($\theta : a \mapsto a^4$)
3×3	$(X^3 + 1) + \alpha^7 X^2 + X + 1$	
4×4	$(X^4 + 1) + \alpha^6 X^3 + \alpha^3 X^2 + X + 1$ ($\theta : a \mapsto a^4$)	$(X^4 + 1) + \alpha^5 X^3 + \alpha^3 X^2 + X + 1$ ($\theta : a \mapsto a^4$)
5×5	$(X^5 + 1) + \alpha^3 X^4 + \alpha^7 X^3 + \alpha^{14} X^2 + \alpha X + 1$ ($\theta : a \mapsto a^4$)	
6×6	$(X^6 + 1) + \alpha^{13} X^5 + \alpha^4 X^4 + \alpha^7 X^3 + \alpha^3 X^2 + X + 1$ ($\theta : a \mapsto a^4$)	

TABLE 5.4 – Polynômes générateurs de matrices θ -circulantes MDS records \mathbb{F}_{2^4}

Remarque 21.

- *Lorsqu'on considère uniquement le cas où θ est l'automorphisme du Frobenius, il semble qu'il n'existe pas de matrices θ -circulantes MDS de taille $k \times k$ sur des extensions de corps 2^s avec $s < k$ mais qu'il en existe lorsque $s \geq k$. Ceci ne semble pas vrai pour tout θ , comme l'illustre le Tableau 5.3. Ceci peut probablement s'expliquer grâce au Théorème 6 qui suggère que l'espace racine, \mathbb{F}_{q_0} -espace vectoriel sur une extension de corps avec*

$q_0 \geq q$ est de dimension suffisante quand l'espace racine d'un polynôme de degré k pour le Frobenius annule au maximum un espace de dimension k .

- Il semble qu'il n'existe pas de matrices θ^i -involutives pour des tailles différentes de $k \times k$. Si cela semble confirmer la pertinence de nos observations algébriques, il semble difficile d'envisager tous les cas de θ^i -involutivités et d'en démontrer les impossibilités.

5.3 Relâcher les contraintes

Nous allons présenter trois raffinements des structures précédemment introduites. Pour commencer, nous allons introduire les matrices cycliques, construites à l'identique des matrices circulantes mais à partir de n'importe quel cycle et plus seulement d'un décalage circulaire des coefficients. Ensuite, nous présenterons une construction directe de matrices θ^2 -circulantes MDS quasi- θ -involutives. Enfin, nous verrons qu'il est possible, au détriment de la description macroscopique de la matrice, de considérer des matrices binaires circulantes par bloc pour déterminer des couches de diffusion linéaire involutives à moindre coût.

5.3.1 Matrices cycliques MDS

L'architecture utilisée dans l'implémentation sérialisée des matrices circulantes admet une généralisation naturelle. L'utilisation d'une permutation quelconque des entrées de la ligne qui précède et plus seulement un décalage circulaire des coefficients mène à la définition des matrices cycliques, au détriment toutefois de la caractérisation algébrique précédente. Une construction directe paraît alors hors de portée, sauf méthode *ad hoc* du type construction de Hadamard-Cauchy. On détermine alors des stratégies de recherche pour obtenir des matrices MDS avec les mêmes propriétés d'implémentation que les matrices circulantes.

Cette généralisation entraîne une explosion de la complexité d'une recherche exhaustive, *a fortiori* si on autorise des permutations différentes des coefficients d'une ligne à l'autre. Largement inspiré des travaux de Liu et Sim dans [LS16], on se concentre sur les paramètres dédiés aux implémentations matérielles, et on cherche à déterminer des matrices MDS de tailles $k \leq 8$ sur \mathbb{F}_{2^4} . Comme pour les matrices de Hadamard, il est possible de raffiner les recherches par la définition d'une relation d'équivalence sur les matrices. L'intérêt de l'introduction de cette relation d'équivalence se fonde sur la proposition suivante qui permet de restreindre largement le spectre des recherches et de considérer les matrices à équivalence près.

Des résultats *ad hoc* permettent alors de réduire les recherches en caractérisant les matrices circulantes MDS de taille $k \leq 8$ en 5 catégories :

Théorème 34. *Pour les tailles de matrices $k \leq 8$, il existe au plus 5 types de matrices circulantes MDS, les matrices circulantes dont la première ligne possède k coefficients distincts, 1, 2 ou 3 paires de coefficients égaux ou 3 coefficients égaux.*

Démonstration. Commençons par remarquer qu'une matrice circulante de taille paire ne peut avoir les coefficients de sa première ligne satisfaire : $c_i = c_{i+\frac{k}{2}}$. En effet, le cas échéant, la sous-matrice carrée définie par les colonnes i et $i + \frac{k}{2}$ et les lignes 0 et $\frac{k}{2}$ donne la matrice suivante, évidemment non inversible :

$$\begin{pmatrix} c_i & c_{i+\frac{k}{2}} \\ c_{i-\frac{k}{2}} & c_i \end{pmatrix}$$

Généralisons cette idée pour remarquer qu'une matrice circulante MDS ne peut avoir les coefficients de sa première ligne satisfaisant : $c_i = c_{i+d}$ et $c_j = c_{j+d}$ avec $i \neq j$. En effet, le cas échéant, la sous-matrice carrée définie par les colonnes i et $i+d$ et les lignes 0 et $(i-j) \bmod k$ donne la matrice suivante, évidemment non inversible :

$$\begin{pmatrix} c_i & c_{i+d} \\ c_j & c_{j+d} \end{pmatrix}$$

Ainsi, les matrices circulantes MDS de taille $k \times k$ présentent au plus $\lfloor \frac{k-1}{2} \rfloor$ différences et donc possèdent au moins $\lceil \frac{k+1}{2} \rceil$ éléments différents. Pour une taille $k = 8$, en particulier, 3 distances distinctes sont autorisées et il y a donc au plus 3 paires de coefficients égaux. Si un coefficient apparaît trois fois, $c_i = c_{i+d_1} = c_{i+d_2}$, alors d_1, d_2 et $d_2 - d_1$ sont deux à deux distincts. Toute multiplicité supérieure est impossible. De la même manière, lorsque $k < 8$, il y a également au plus 3 distances possibles. D'où la classification du théorème. \square

Nous sommes maintenant en mesure de définir les matrices cycliques.

Définition 49. Soit ρ une permutation cyclique de taille k . Une matrice cyclique de taille $k \times k$ est une matrice dont chaque ligne est l'image par ρ de la ligne qui la précède.

Exemple 18. Une matrice circulante est donc une matrice cyclique associée à la permutation $\sigma_1 = (k, k-1, \dots, 1)$. Une matrice circulante à gauche est alors la matrice cyclique associée à la permutation $\sigma_2 = (1, 2, \dots, k)$. Pour une matrice de taille 4×4 sur \mathbb{F}_{2^4} engendré par le polynôme $T^4 + T + 1$, de première ligne arbitraire $[1, \alpha, \alpha^2, \alpha^3]$, on construit les matrices \mathbf{C}_{σ_1} et \mathbf{C}_{σ_2} comme suit :

$$\mathbf{C}_{\sigma_1} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 \\ \alpha^3 & 1 & \alpha & \alpha^2 \\ \alpha^2 & \alpha^3 & 1 & \alpha \\ \alpha & \alpha^2 & \alpha^3 & 1 \end{pmatrix} \text{ et } \mathbf{C}_{\sigma_2} = \begin{pmatrix} 1 & \alpha & \alpha^2 & \alpha^3 \\ \alpha & \alpha^2 & \alpha^3 & 1 \\ \alpha^2 & \alpha^3 & 1 & \alpha \\ \alpha^3 & 1 & \alpha & \alpha^2 \end{pmatrix}$$

Cette définition, non généralisée à toutes les permutations, conserve une structure aux matrices cycliques proche des matrices circulantes. Le théorème suivant permet alors de relier les matrices cycliques aux matrices circulantes et de traduire l'analyse précédente dans le contexte des matrices cycliques.

Théorème 35. Soit S un ensemble ordonné à k éléments. Toute matrice cyclique définie par cet ensemble ordonné est équivalente à une matrice circulante définie par une permutation des coefficients de S .

On a les deux corollaires immédiats :

Corollaire 2. Toute matrice cyclique est équivalente à une matrice circulante définie par le même ensemble de coefficients et avec le même nombre de branchements.

Corollaire 3. Pour les taille de matrices $k \leq 8$, il existe au plus 5 types de matrices cycliques MDS, les matrices cycliques dont la première ligne possède k coefficients distincts, 1, 2 ou 3 paires de coefficients égaux ou 3 coefficients égaux.

Les impossibilités algébriques qui interdisent aux matrices circulantes d'être involutives sont alors levées. Il est possible de mettre en place des stratégies de recherche qui permettent de tirer parti des classes d'équivalence et des propriétés comme l'involution qu'on souhaite assurer. Liu et Sim ont réalisé les recherches exhaustives sur un exemple de matrices cycliques, les matrices circulantes à gauche. Ces résultats, prometteurs, ont été prolongés à toutes les matrices cycliques possibles.

Le Tableau 5.5 recense le nombre de matrices cycliques MDS, respectivement les matrices cycliques MDS involutives, ainsi que le meilleur compte de XOR obtenu pour les jeux de paramètres possibles sur \mathbb{F}_{2^4} .

Taille	Nombre de matrices cycliques MDS		Meilleur Compte de XOR	
	simples	Involutives	simples	Involutives
3×3	4500	12	1	12
4×4	99360	0	3	
5×5	1915200	100	4	14

TABLE 5.5 – Dénombrement des matrices cycliques MDS sur \mathbb{F}_{2^4}

Le Tableau 5.6 recense les paramètres qui engendrent les meilleurs compte de XOR pour les matrices cycliques MDS sur \mathbb{F}_{2^4} .

Taille	Première ligne		permutation	
	simples	Involutives	simples	Involutives
3×3	$[1, 1, \alpha^{14}]$	$[\alpha^2, 1 + \alpha^2 + \alpha^3, \alpha^3]$	(1, 2, 3)	(1, 2, 3)
4×4	$[1, 1, \alpha^{13}, \alpha]$		(1, 2, 3, 4)	
5×5	$[1, \alpha^{14}, \alpha, \alpha, \alpha^{14}]$	$[1, \alpha^{11}, \alpha^{13}, \alpha^7, \alpha^{14}]$	(1, 2, 3, 4, 5)	(1, 2, 3, 4, 5)

TABLE 5.6 – Matrices cycliques MDS records sur \mathbb{F}_{2^4}

Remarque 22.

- *Il semble que les matrices involutives n'existent que pour les matrices cycliques circulantes à gauche.*
- *Il semble que la condition cycle des matrices cycliques soit essentielle pour obtenir des matrices MDS.*

Ceci ouvre la voie à la détermination d'autres matrices involutives éventuellement moins coûteuse. L'étape de généralisation suivante concerne des matrices dont les lignes sont toutes des permutations de la première ligne mais pas nécessairement des puissances de la même permutation. Ceci induit des coûts supplémentaire d'implémentation. Les matrices de Hadamard figurent dans cette généralisation.

5.3.2 Construction directe de matrices θ^2 -circulantes MDS quasi- θ -involutives

S'il est complexe de mettre au point une construction directe de matrices θ -circulantes MDS involutives, il est possible de relâcher la contrainte imposée par l'involutivité en θ -involutivité. Un deuxième relâchement consiste à rechercher des matrices telles que $\mathbf{M}\mathbf{M}^{[1]}$ est une permutation cyclique et plus nécessairement l'identité. On appelle une telle matrice \mathbf{M} θ -involutive à

permutation près. Son implémentation matérielle peut être très efficace en base normale avec l'ajout d'un routage supplémentaire au calcul d'une matrice θ -involutive.

Pour construire une matrice θ^2 -circulante θ -involutive à permutation près, on utilise les codes de Gabidulin. Soit donc $k \geq 2$ un entier. On considère $\mathbb{F}_{2^{2k}}$ et θ l'automorphisme de Frobenius. La procédure pour construire les matrices se résume alors à :

1. Choisir α un élément normal dans $\mathbb{F}_{2^{2k}}$.
2. Construire $\mathbf{G}_1 = (\alpha^{[2j+i]})_{i=0, j=0}^{k-1, k-1}$ et $\mathbf{G}_2 = (\alpha^{[2j+i+1]})_{i=0, j=0}^{k-1, k-1}$.
3. Calculer $\mathbf{M} = \mathbf{G}_1^{-1} \mathbf{G}_2$

Soit α un élément normal dans $\mathbb{F}_{2^{2k}}$. La matrice \mathbf{G} suivante est une matrice génératrice d'un code de Gabidulin de paramètres $[2k, k, k+1]_{2^{2k}}$.

$$\mathbf{G} = \begin{pmatrix} \alpha^{[0]} & \alpha^{[2]} & \dots & \alpha^{[2k-1]} \\ \alpha^{[1]} & \alpha^{[3]} & \dots & \alpha^{[0]} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{[k-1]} & \alpha^{[k]} & \dots & \alpha^{[k-2]} \end{pmatrix} \quad (5.1)$$

Les matrices \mathbf{G}_1 et \mathbf{G}_2 de la construction directe correspondent respectivement à la moitié gauche et à la moitié droite de \mathbf{G} . Le théorème suivant recense les caractéristiques structurelles des matrices ainsi construites.

Théorème 36. *Soit $\mathbf{M} = \mathbf{G}_1^{-1} \mathbf{G}_2$, alors*

- \mathbf{M} est une matrice θ^2 -circulante MDS
- $\mathbf{M}\mathbf{M}^{[1]}$ est une matrice de permutation binaire, i.e.

$$\mathbf{M}\mathbf{M}^{[1]} = \begin{pmatrix} 0 & \dots & 0 & 1 \\ 1 & \dots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \dots & 1 & 0 \end{pmatrix},$$

Prouvons avant tout le lemme suivant :

Lemme 2. *Soit $\beta_0, \dots, \beta_{k-1}$, la première ligne de \mathbf{G}_1^{-1} , alors*

1. $\mathbf{G}_1^{-1} = (\beta_j^{[2i]})_{i=0, j=0}^{k-1, k-1}$
2. $\sum_{u=0}^{k-1} \beta_u \alpha^{[u+2j]} = \delta_{0,j}$, pour tout $j = 0, \dots, k-1$
3. $\sum_{u=0}^{k-1} \alpha^{[2u+i]} \beta_j^{[2u]} = \delta_{i,j}$, pour tout $i, j = 0, \dots, k-1$

Démonstration. Soit $\beta_0, \dots, \beta_{k-1}$, la première ligne de \mathbf{G}_1^{-1} . Elle satisfait la deuxième condition du lemme. Élevant cette équation à la puissance [2], on obtient :

$$\begin{aligned} \sum_{u=0}^{k-1} \beta_u^{[2]} \alpha^{[u+2(j+1)]} &= \delta_{0,j}^{[2]} = \delta_{0,j}, \quad \forall j = 0, \dots, k-1 \\ \sum_{u=0}^{k-1} \beta_u^{[2]} \alpha^{[u+2j]} &= \delta_{0,j-1}^{[2]} = \delta_{1,j}, \quad \forall j = 0, \dots, k-1 \end{aligned}$$

où pour $j = k-1$, puisque $\alpha^{[2k]} = \alpha$, l'égalité vient de $\alpha^{[u+2(j+1)]} = \alpha^{[u]}$. D'où $\beta_0^{[2]}, \dots, \beta_{k-1}^{[2]}$ est la deuxième ligne de \mathbf{G}_1^{-1} . Par induction, on prouve le premier point du lemme. Le dernier point vient de la relation évidente : $\mathbf{G}_1^{-1} \mathbf{G}_1 = \mathbf{I} = \mathbf{G}_1 \mathbf{G}_1^{-1}$. \square

On peut désormais prouver le théorème.

Démonstration. Soit $\mathbf{M} = (m_{i,j}) = \mathbf{G}_1^{-1} \mathbf{G}_2$. Par construction, \mathbf{M} est la matrice de redondance d'une matrice génératrice sous forme systématique d'un code de Gabidulin de longueur $2k$ et de dimension k et est donc MDS. D'après le lemme précédent, le terme générique $m_{i,j}$ de \mathbf{M} satisfait :

$$\begin{aligned} m_{i,j} &= \sum_{u=0}^{k-1} \beta_u^{[2i]} \alpha^{[u+1+2j]}, & \text{for all } i, j = 0, \dots, k-1 \\ m_{i,j}^{[2]} &= \sum_{u=0}^{k-1} \beta_u^{[2(i+1)]} \alpha^{[u+1+2(j+1)]}, & \text{for all } i, j = 0, \dots, k-1 \end{aligned}$$

Ceci implique que $m_{i+1,j+1} = m_{i,j}^{[2]}$, ainsi \mathbf{M} est une matrice θ^2 -circulante MDS. Il reste à prouver le dernier point du théorème. Le (i, j) th terme générique de $\mathbf{M}\mathbf{M}^{[1]}$ est :

$$\left(\sum_{\ell=0}^{\ell-1} \sum_{u,u'=0}^{k-1} \beta_u^{[2i]} \alpha^{[u+2\ell]} \beta_{u'}^{[2(\ell+1)]} \alpha^{[u'+2(j+1)]} \right)_{i,j} \quad (5.2)$$

Remarquons que le seul terme dans l'équation dépendant de ℓ est $\alpha^{[u+1+2\ell]} \beta_{u'}^{[2(\ell+1)]}$. En sommant alors sur ℓ , d'après le troisième point du lemme, on obtient :

$$\sum_{\ell=0}^{k-1} \alpha^{[u+2\ell]} \beta_{u'}^{[2(\ell+1)]} = \delta_{u',u+1}$$

Ainsi, l'équation (5.2) devient :

$$\sum_{u=0}^{k-1} \beta_u^{[2i]} \alpha^{[u+2(j+1)]} = \left(\sum_{u=0}^{k-1} \beta_u \alpha^{[u+2(j-i+1)]} \right)^{[2i]} = \delta_{0,(j-i+1)}.$$

□

Le Théorème 36 établit que l'inverse de \mathbf{M}^{-1} se décompose en $\mathbf{M}^{[1]}\mathbf{P}$, où \mathbf{P} est une matrice de permutation cyclique.

Exemple 19. Soit \mathbb{F}_{2^8} défini par $X^8 + X^4 + X^3 + X^2 + 1$, et α une racine de ce polynôme. L'élément α^5 est normal et on considère le code de Gabidulin sur \mathbb{F}_{2^8} de matrice génératrice \mathbf{G} :

$$\mathbf{G} = \begin{pmatrix} \alpha^5 & \alpha^{10} & \alpha^{20} & \alpha^{40} & \alpha^{80} & \alpha^{160} & \alpha^{65} & \alpha^{130} \\ \alpha^{10} & \alpha^{20} & \alpha^{40} & \alpha^{80} & \alpha^{160} & \alpha^{65} & \alpha^{130} & \alpha^5 \\ \alpha^{20} & \alpha^{40} & \alpha^{80} & \alpha^{160} & \alpha^{65} & \alpha^{130} & \alpha^5 & \alpha^{10} \\ \alpha^{40} & \alpha^{80} & \alpha^{160} & \alpha^{65} & \alpha^{130} & \alpha^5 & \alpha^{10} & \alpha^{20} \end{pmatrix}$$

L'extraction des colonnes paires pour \mathbf{G}_1 et des colonnes impaires pour \mathbf{G}_2 donne :

$$\mathbf{M} = \mathbf{G}_1^{-1} \mathbf{G}_2 = \begin{pmatrix} \alpha^{98} & \alpha^{116} & \alpha^{132} & \alpha^{232} \\ \alpha^{163} & \alpha^{137} & \alpha^{209} & \alpha^{18} \\ \alpha^{72} & \alpha^{142} & \alpha^{38} & \alpha^{71} \\ \alpha^{29} & \alpha^{33} & \alpha^{58} & \alpha^{152} \end{pmatrix} \text{ et finalement } \mathbf{M}\mathbf{M}^{[1]} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Cette proposition permet de construire des matrices MDS avec de bonnes propriétés d'implémentation. L'intérêt de cette construction directe croît évidemment avec la taille des matrices considérées, lorsqu'une vérification du caractère MDS devient irréalisable en pratique, où trop

coûteux pour chercher des matrices au hasard. Un défaut de cette construction directe est la taille du corps sur lequel elle est employée, deux fois plus grande que la taille de la matrice, qui suggère l'utilisation de boîtes S sur un nombre de bits relativement important. La nouvelle notion d'involutivité est très éloignée de la pureté théorique de l'involutivité cependant, l'implémentation qu'elle induit est pratiquement la même que pour une involution. Il est donc possible de traduire ces propriétés en dehors de la construction directe pour lancer des recherches exhaustives en vue de diminuer les comptes de XOR.

5.3.3 Couches de diffusion linéaire parfaites circulantes

La construction de couches de diffusion linéaire avec un nombre de branchements maximal à partir de matrices circulantes MDS admet une généralisation naturelle, à l'instar de ce qui a été fait dans le paradigme des matrices récursives. On peut alors remplacer les multiplications dans un corps de Galois par des applications linéaires inversibles sur les mots. Dans la lignée des travaux de Li et Wang [LW16], les matrices binaires circulantes par bloc émergent pour appréhender la construction de couches de diffusion linéaire parfaites. La perte de commutativité des opérations, l'absence de traduction naturelle sur les polynômes associées aux matrices circulantes détruit les critères sur les polynômes qui interdisent l'existence de matrices circulantes MDS involutives en caractéristique 2. Le principal mérite de cette nouvelle structure concerne les petites tailles de paramètres pour lesquels des stratégies de recherche font émerger des matrices binaires avec un très faible compte de XOR mais qui engendrent néanmoins des couches de diffusion linéaire parfaites dont certaines sont même involutives. Idoine pour la minimisation des comptes de XOR pour des petits paramètres, cette généralisation entraîne une explosion de la complexité des recherches exhaustives, qui deviennent très vite irréalisables. Dans la lignée des travaux de Li et Wang [LW16], on se focalise ici sur les matrices binaires circulantes par bloc 4×4 pour des blocs de taille $m \in \{4, 8\}$, idéales pour une utilisation avec des boîtes S de taille $m \in \{4, 8\}$. Une telle matrice \mathbf{D} s'écrit au moyen de matrices binaires de taille $m \times m$ sur \mathbb{F}_2 :

$$\mathbf{D} = \begin{pmatrix} \mathbf{D}_1 & \mathbf{D}_2 & \dots & \mathbf{D}_k \\ \mathbf{D}_k & \mathbf{D}_1 & \dots & \mathbf{D}_{k-1} \\ \vdots & \vdots & \dots & \vdots \\ \mathbf{D}_2 & \mathbf{D}_3 & \dots & \mathbf{D}_1 \end{pmatrix}$$

Comme pour les constructions de couches de diffusion linéaires parfaites à partir de LFSRs généralisés, des stratégies de recherche peuvent être mises en place pour déterminer les minimums de compte de XOR pour un jeu de paramètres fixé. Parmi les résultats majeurs, déterminés par recherche exhaustive, on trouve que pour $m \in \{4, 8\}$, il existe \mathbf{A}, \mathbf{B} et $\mathbf{C} \in GL(m, \mathbb{F}_2)$ telles que la matrice circulante par bloc définie par la première ligne $(\mathbf{I}, \mathbf{A}, \mathbf{B}, \mathbf{C})$ est une matrice involutive qui engendre une couche de diffusion linéaire parfaite. On s'est autorisé à considérer davantage d'applications linéaires qui s'inscrivent dans une lecture macroscopique comme celle qu'offre le paradigme des codes \mathbb{F}_q^m -linéaires. Ce n'est pas une surprise d'obtenir de meilleurs résultats en termes de compte de XOR. En supprimant les barrières algébriques qui interdisent l'existence de matrices circulantes MDS involutives sur des corps de caractéristique 2 pour des tailles de matrice paires, on réussit à en faire apparaître pour ces jeux de paramètres. C'est une méthode, *ad hoc* par excellence, adaptée pour les petites tailles de paramètres pour lesquels elle donne de très bons résultats.

Exemple 20. *On donne deux exemples issus des travaux de Li et Wang [LW16].*

- La matrice \mathbf{D}_1 est la matrice d'une couche de diffusion binaire circulante par bloc de diffusion optimale et involutive :

$$\mathbf{D}_1 = \left(\begin{array}{cccc|cccc|cccc|cccc} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array} \right)$$

Son compte de XOR correspond dans l'équivalent des matrices MDS sur \mathbb{F}_{2^4} à $5 + 3 \cdot 4$ XORs par ensemble de 4 lignes.

- la matrice \mathbf{D}_2 est la matrice d'une couche de diffusion binaire Hadamard par bloc de diffusion optimale et involutive

$$\mathbf{D}_2 = \left(\begin{array}{cccc|cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array} \right)$$

Son compte de XOR correspond dans l'équivalent des matrices MDS sur \mathbb{F}_{2^4} à $6 + 3 \cdot 4$ XORs par ensemble de 4 lignes.

Conclusions et perspectives

Deux structures matricielles, destinées à une utilisation dans des implémentations matérielles légères de primitives symétriques car construite à partir d'un nombre linéaire et non quadratique de multiplications dans un corps de Galois, les matrices circulantes et les matrices récursives s'identifient à des structures polynomiales simples. De celles-ci, on caractérise sur les polynômes des propriétés d'implémentation désirables supplémentaires comme l'involutivité. De celles-ci, on extrait des raisonnements sur les racines des polynômes qui font naître des constructions directes de matrices MDS, optimales pour la diffusion sur deux tours. De celles-ci, on peut raffiner les recherches exhaustives qui les parcourent en vue des meilleurs comptes de XOR en tirant parti du Théorème de MacWilliams sur les isométries de la distance de Hamming. De celles-ci émergent enfin de nouvelles structures matricielles, les matrices θ -circulantes et les matrices θ -récursives qui s'identifient à des structures sur les θ -polynômes, dont les propriétés d'implémentation sont proches de celles des matrices circulantes et récursives et qui ouvrent le panorama des constructions directes de matrices MDS structurées. En dépit de l'extrême régularité de ces objets, de très nombreuses conjectures demeurent cependant sur les conditions d'existence de matrices MDS pour certains jeux de paramètres, d'autant plus lorsqu'on cumule les propriétés désirées. L'étude sur les structures polynomiales associées à ces objets n'a pas encore exsudé tout son jus.

La recherche aujourd'hui s'oriente davantage vers des conceptions dans lesquelles l'effet d'avalanche, la diffusion est maximale pour un nombre de tour supérieur à 2 avec des étapes de dispersion. L'optimalité de la diffusion locale est remise en question pour sa rigidité et ses coûts conséquents d'implémentation matérielle. Le standard SHA-3, basé sur la primitive KECCAK [KECCAK], en est un exemple représentatif. La visualisation de l'état interne selon la partition induite de l'utilisation de boîtes S et la \mathbb{F}_{2^m} -linéarité de l'application linéaire au lieu de la \mathbb{F}_2 -linéarité nécessaire n'apparaissent plus légitime. La diffusion, très locale, est assez faible mais compensée par des effets de dispersion, qui assurent une bonne diffusion sur plusieurs tours. Les effets de dispersion restent aujourd'hui peu analysés. La construction de matrices MDS laisse alors la place à de nouvelles constructions, binaires, qui assurent un nombre de branchements équivalent. Les constructions de couches de diffusion linéaires parfaites en sont le meilleur exemple. Restreint à des petits paramètres, des stratégies de recherches naïves permettent d'obtenir des comptes de XOR parmi les plus faibles. Cependant, les assurances de sécurité fournies par l'utilisation d'une matrice MDS demeurent solides. Les différentes généralisations des structures matricielles présentées dans cette thèse peuvent toutes se cumuler pour engendrer un grand creuset de recherche exhaustive. La multiplication par un élément dans un corps de Galois se remplace par une application linéaire inversible, l'application d'un automorphisme se remplace par une permutation arbitraire de bits, l'involutivité d'une couche de diffusion peut être considérée à permutation de bits près. Pour des grandes tailles de paramètres, il reste possible d'engendrer

directement des matrices MDS. Pas moins de trois nouvelles constructions directes de matrices MDS dans des jeux de paramètres réalistes figurent dans cette thèse.

Deuxième partie

Attaques rebond sur les permutations internes de Grøstl₅₁₂

Chapitre 6

Présentation générale

Sommaire

6.1 Fonctions de hachage	96
6.1.1 Propriétés cryptographiques	97
6.1.2 Modes d'opération - Preuves de sécurité	98
6.1.3 Grøstl	99
6.2 Algorithmes de différenciations	101
6.2.1 Modèle de l'oracle aléatoire	102
6.2.2 Paradoxe des anniversaires généralisés	102
6.2.3 Problème des anniversaires limité	102
6.3 Chemins différentiels tronqués	103
6.3.1 Principe	104
6.3.2 Attaque rebond	104

Les fonctions de hachage cryptographiques sont des primitives symétriques à la base de nombreux protocoles. Conçues pour assurer l'intégrité ou l'authentification de messages, elles interviennent par exemple dans les processus d'échange de clefs, de dérivation de clefs ou de code d'authentification de messages. Pour garantir ces services cryptographiques, des propriétés de résistance aux collisions, aux pré-images et aux secondes pré-images sont requises. Les fonctions de hachage mettent en œuvre des fonctions de compression selon des *modes d'opération*. Des preuves de sécurité assurent les propriétés de résistance mentionnées sous l'hypothèse que les fonctions de compression soient sûres. Depuis sa standardisation, vingt années de cryptanalyse n'ont pas remis en cause les revendications de sécurité de l'AES. Son architecture inspirée de la wide trail strategy a donc été copiée dans de très nombreuses primitives. Pour les seules fonctions de hachage, plusieurs reposent sur ces mêmes critères de conception : Whirlpool [BR00], GOST R [PLK06], Maelstrom [BRF06], Whirlwind [Bar+10] et évidemment PHOTON [GPP11]. La prolifération de ces architectures pour les fonctions de hachage a appelé une réponse de la cryptanalyse : l'attaque rebond, exploitation de cette structure macroscopique sans élément secret. Plusieurs raffinements de cette attaque se sont succédés jusqu'à utiliser à plein les possibilités qu'offre l'avalanche libre de différences. Grøstl₅₁₂ est également une de ces fonctions de hachage dont les permutations internes s'inspirent de l'AES. La meilleure attaque rebond connue différencie des versions réduites à 10 tours sur les 14 de ces permutations d'un tirage aléatoire dans l'ensemble des permutations pour une complexité calculatoire d'environ 2^{392} .

Contributions Le Chapitre 8 présente la première attaque rebond sur les permutations internes de Grøstl_{512} qui différencie des versions à 11 tours de celles-ci d'un tirage aléatoire dans l'ensemble des permutations pour une complexité calculatoire d'environ 2^{72} . Cette attaque repose sur un chemin différentiel tronqué structuré, qui ne provient pas d'avalanche libre, issu de techniques de programmation linéaire sous contrainte. L'introduction du formalisme des Super-Boîtes S permet d'identifier des sous-espaces linéaires dont on construit des bases, ce qui permet des énumérations rapides et peu coûteuses en mémoire. L'exploitation de la structure du chemin différentiel tronqué permet de consommer peu de degrés de liberté pour faire correspondre des différences propagées en amont et en aval. On est alors en mesure de contrôler la propagation de différences pour davantage de tours que dans le cas des attaques rebond classiques. Le Tableau 6.1, résumé des meilleurs résultats obtenus sur Grøstl_{512} , illustre la portée de ces travaux.

Nombre de tours	Attaque rebond	Mémoire	Attaque générique	Référence
7	2^{152}	2^{56}	2^{512}	[Sas+10]
8	2^{280}	2^{64}	2^{448}	[JNP14]
9	2^{328}	2^{64}	2^{384}	[JNP14]
10	2^{392}	2^{64}	2^{448}	[JNP14]
11	2^{72}	2^{56}	2^{96}	[CGL17]

TABLE 6.1 – Attaques rebond sur les permutations internes de Grøstl_{512} .

Ce chapitre introduit les fonctions de hachage et les propriétés de sécurité qu'elles doivent vérifier pour être dites *cryptographiques*, les méthodes de conception utilisées et l'exemple de Grøstl . Il introduit ensuite la notion de non différentiabilité d'un tirage aléatoire dans l'ensemble des permutations. Nous développerons alors deux problèmes combinatoires à la base desquels une telle différentiabilité peut être exhibée. Ce chapitre se conclut alors par la présentation des chemins différentiels tronqués et le principe des attaques rebond qui les exploitent.

6.1 Fonctions de hachage

Les fonctions de hachage sont des fonctions qui produisent des sorties de taille fixe, des *empreintes*, à partir d'entrées de taille quelconque. À l'origine, elles ont été conçues pour assurer un traitement efficace d'opérations comme le tri ou l'insertion d'éléments dans des bases de données. Les traitements algorithmiques sont en effet plus efficaces lorsque réalisés sur des éléments de petite taille fixe. Une description formelle des fonctions de hachage est donnée comme suit :

$$\begin{aligned}
 H : \{0,1\}^* &\rightarrow \{0,1\}^n \\
 m &\mapsto H(m)
 \end{aligned}$$

L'utilisation de ces fonctions pour la gestion d'une base de donnée suggère la propriété suivante : avec très forte probabilité, deux entrées quelconques ne doivent pas partager une même empreinte. Dans le cas contraire, ces deux entrées différentes avec une même empreinte seraient identifiées. L'involutivité d'une telle fonction est une absurdité comme l'assure le principe des tiroirs. Le comportement idéal d'une fonction de hachage se traduit donc de la façon suivante :

$$\mathbb{P}_{x \neq y}(H(x) = H(y)) = 2^{-n}$$

L'exemple classique de l'utilisation de fonctions de hachage au quotidien permet d'appréhender une nouvelle propriété souhaitable. Plutôt que de stocker des mots de passe, un serveur conservera plus volontiers en mémoire les empreintes des mots de passe. En cas de compromission de la base de donnée des mots de passe, l'attaquant ne doit pas être en mesure de posséder les couples identifiants/mot de passe. Par nature, une fonction de hachage doit être une fonction à sens unique qui ne possède pas d'algorithme d'inversion. La détermination d'antécédents d'une empreinte, la *forge* d'une pré-image, doit être difficile pour que la fonction de hachage soit dite cryptographique. Les fonctions de hachage sont publiques, la difficulté de la forge ne repose donc pas sur la connaissance d'un secret. Le modèle de sécurité qui sous-tend l'analyse des fonctions de hachage suppose qu'un attaquant peut effectuer lui-même des calculs d'empreintes. Un comportement idéal d'une fonction de hachage se traduit de la façon suivante :

$$\forall \mathbf{e} \in \{0, 1\}^n, \mathbb{P}(H(X) = \mathbf{e}) = 2^{-n}$$

L'utilisation des fonctions de hachage en cryptographie est très répandue. Pour assurer les propriétés d'intégrité, elles permettent de détecter l'existence de corruptions d'un message transmis sur un canal non sécurisé. Pour assurer l'authentification des parties, elles prouvent la connaissance d'un secret sans le divulguer. À l'instar des primitives de chiffrement, de nombreuses conceptions de fonctions de hachage dédiées à des utilisations cryptographiques ont essaimé. Une compétition organisée par le NIST s'est conclue par la standardisation de la primitive KECCAK, conçue par Bertoni, Daemen, Peeters, Van Assche, commune aujourd'hui sous le sigle SHA-3. Outre celle-ci et Grøstl déjà évoquées, d'autres fonctions de hachage cryptographiques ont été proposées, parmi lesquelles on fait mention des finalistes BLAKE [Aum+14], JH [Wu11] et Skein [Fer+11].

Nous commencerons par présenter les notions de résistance en collision, pré-image et seconde pré-image. Nous verrons alors qu'il existe des modes d'opération qui assurent cette résistance sous l'hypothèse que les permutations internes qu'ils mettent en œuvre sont non différentiable d'un tirage aléatoire. Enfin, nous présenterons plus en détail Grøstl, fonction de hachage que notre attaque cible.

6.1.1 Propriétés cryptographiques

Les fonctions de hachage sont au cœur de nombreux protocoles cryptographiques : signatures, codes d'authentification de messages, dérivation de clefs, échanges de clefs, générateurs pseudo-aléatoires. . . Cette utilisation récurrente présuppose l'impossibilité de constater une incohérence avec les comportements idéaux décrits précédemment. Pour être considérée *cryptographique*, une fonction de hachage doit offrir des garanties de sécurité qui se résument à trois propriétés fondamentales : la résistance contre les attaques en collision, les attaques en pré-image et en seconde pré-image.

Résistance en collision La recherche de collision repose sur l'heuristique du paradoxe des anniversaires : pour une fonction au comportement idéal, tout bit de sortie est tiré aléatoirement selon une loi uniforme et deux messages présentent la même empreinte avec probabilité 2^{-n} . Considérant M messages, on peut engendrer $\binom{M}{2}$ paires de messages. Supposant alors pour simplifier les calculs l'indépendance de tous les tests, une collision des empreintes est réalisée pour l'une de ces paires avec une probabilité voisine de $\binom{M}{2} \cdot 2^{-n} \simeq M^2 \cdot 2^{-(n+1)}$. Une collision existe donc avec une probabilité supérieure à 2^{-1} dès que $M \geq \sqrt{2^n}$. La résistance en collision consiste alors à assurer qu'il n'est pas possible de déterminer deux messages partageant une même

empreinte plus rapidement qu'il est possible d'engendrer une collision générique par paradoxe des anniversaires.

Résistance en pré-image La forge d'une pré-image repose sur les mêmes principes : pour une fonction au comportement idéal, le raisonnement précédent suggère une recherche probabiliste d'un message dont les sorties correspondent à l'empreinte bit à bit. Un message quelconque possède un bit fixé de l'empreinte avec probabilité 2^{-1} et la même empreinte avec probabilité 2^{-n} . La résistance en pré-image consiste à assurer qu'il n'est pas possible de déterminer un antécédent d'une empreinte donnée plus rapidement que par recherche probabiliste.

Résistance en seconde pré-image Pour une fonction au comportement idéal, la connaissance d'un antécédent d'une empreinte ne donne aucune information sur les autres antécédents. La meilleure attaque générique est donc réalisée une fois encore par une recherche probabiliste. La résistance en seconde pré-image consiste à assurer qu'il n'est pas possible de déterminer un second message présentant la même empreinte qu'un message donné plus rapidement que par recherche probabiliste.

Le Tableau 6.2 résume les complexités des attaques génériques sur les recherches de collisions, pré-images et secondes pré-images.

Recherche	Attaque générique	Complexité calculatoire
collisions	Paradoxe des anniversaires	$2^{\frac{n}{2}}$
pré-images	Recherche probabiliste	2^n
secondes pré-images	Recherche probabiliste	2^n

TABLE 6.2 – Attaques génériques sur les fonctions de hachage cryptographiques

Remarque 23. *La résistance en collision implique la résistance en seconde pré-image.*

6.1.2 Modes d'opération - Preuves de sécurité

La manipulation d'éléments de tailles variables est inefficace. Ce constat motive de transformer un message de longueur variable en un nombre variable de blocs de message de même longueur. Une telle transformation porte le nom anglais de *padding*. Sa version naïve consiste à ajouter au message un 1 suivi du nombre de 0 nécessaire pour obtenir la plus petite taille possible, multiple de la taille du bloc. Les fonctions de hachage sont alors généralement construites, dans la suite des travaux de Rabin [**Rabin:78**], à partir d'une brique élémentaire, une *fonction de compression*, qui met à jour un état interne à partir dudit état interne et d'un bloc de message. L'utilisation de cette fonction de compression est régie par un *mode opératoire*, ou encore *algorithme d'extension de domaine*. Les travaux qui suivent se focalisent sur la construction dite de Merkle-Damgård dont la Figure 6.1 illustre le fonctionnement.

La construction de Merkle-Damgård souffre d'attaques structurelles comme les attaques par extension et les attaques par multi-collisions. Pour pallier ces fragilités, la construction *wide pipe*, proposée par Stefan Lucks [**Lucks:04**], extrait l'empreinte d'un état interne deux fois plus grand. Il est donc nécessaire d'ajouter une application de compression finale. La complexité des attaques augmente alors : une collision de la fonction de compression requiert 2^n opérations. La Figure 6.2 illustre ce mode d'opération.

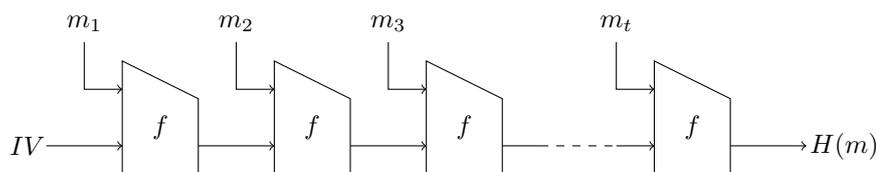


FIGURE 6.1 – Mode d’opération Merkle-Damgård.

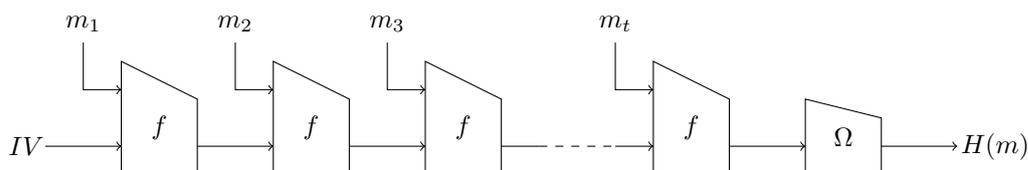


FIGURE 6.2 – Mode d’opération Wide-Pipe.

6.1.3 Grøstl

Grøstl [Gau+09b] est une des fonctions de hachage dont l’architecture des permutations internes s’inspire de l’AES. Elle a été sélectionnée comme finaliste de la compétition du NIST. Son mode d’opération suit la construction wide-pipe et sa fonction de compression comme sa transformation de sortie reposent sur deux permutations internes. Ses revendications de sécurité s’appuient sur une preuve de sécurité qui repose elle-même sur l’hypothèse que les permutations internes se comportent comme des tirages aléatoires uniformes dans l’ensemble des permutations. Deux variantes de Grøstl ont été proposées par ses auteurs, Grøstl₂₅₆ et Grøstl₅₁₂ dont les tailles d’empreintes sont respectivement de 256 et 512 bits.

On se focalise ici sur Grøstl₅₁₂. Ses permutations internes admettent en effet une représentation du registre d’état interne sous la forme d’un rectangle de 8×16 octets. À la différence de l’AES dont le registre d’état interne se représente sous la forme d’un carré, la diffusion totale par avalanche libre d’une différence nécessite trois tours et non deux. Cette divergence suggère que l’attaque rebond menace davantage de tours. L’attaque rebond par exploitation des phénomènes d’avalanche libre la plus aboutie sur Grøstl₅₁₂ est l’oeuvre de Jérémy Jean, Maria Naya Plasencia et Thomas Peyrin [JNP14]. La forme rectangulaire y est déjà mise en cause car elle permet d’étendre le chemin différentiel tronqué classique sur 9 tours des permutations de type AES pour atteindre 10 tours.

La fonction de compression de Grøstl₅₁₂, f_{1024} , et la transformation de sortie Ω sont construites à partir de deux permutations de 1024 bits P_{1024} et Q_{1024} et de Tronc₅₁₂, la troncature qui retourne les 512 derniers bits d’une entrée de longueur supérieure à 512, selon les définitions suivantes, en accord avec la Figure 6.3 :

$$\begin{aligned} f_{1024}(h, m) &= P_{1024}(h \oplus m) \oplus Q_{1024}(m) \oplus h \\ \Omega(x) &= \text{Tronc}_{512}(P_{1024}(x) \oplus x). \end{aligned}$$

La fonction de compression a été prouvée résistante aux attaques en collision et aux attaques en pré-images sous l’hypothèse que P_{1024} et Q_{1024} sont idéales [FSZ09]. Par ailleurs, la fonction

de hachage Grøstl_{512} dans sa globalité est prouvée non différentiable d'un tirage aléatoire uniforme dans l'ensemble des permutations sous l'hypothèse supplémentaire que P_{1024} et Q_{1024} sont indépendantes l'une de l'autre [AMP].

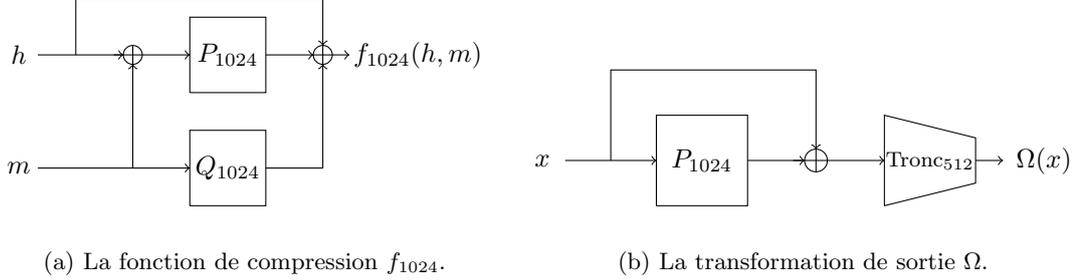


FIGURE 6.3 – Fonctions internes de Grøstl_{512} .

Les permutations P_{1024} et Q_{1024} consistent en 14 itérations de la fonction de tour suivante :

$$R := \text{MixBytes} \circ \text{ShiftBytesWide} \circ \text{SubBytes} \circ \text{AddRoundConstant},$$

où

- **AddRoundConstant (ARC)** ajoute une constante à l'état interne dont la valeur dépend du tour. Cette transformation n'affecte pas les différences et se révèle donc anecdotique dans les travaux présentés ici.
- **SubBytes (SB)** s'applique indépendamment sur les octets de la représentation matricielle. La notation **SB** fera indifféremment référence à la transformation sur l'état interne complet ou sur un ensemble quelconque d'octets.

Cette transformation consiste en la substitution de chaque octet par son image par l'application non linéaire **SBox** issue de l'algorithme Rijndael. Pour simplifier notre analyse, on supposera un comportement idéal pour cette transformation¹ :

$$\forall (\delta, \delta') \in (\mathbb{F}_{2^8}^*)^2, |\{X \in \mathbb{F}_{2^8} \mid \mathbf{SBox}(X) \oplus \mathbf{SBox}(X \oplus \delta) = \delta'\}| \in \{0, 2\}. \quad (6.1)$$

Classiquement, en raison de la non-linéarité de **SB**, on fera l'hypothèse que l'image par **SB** d'un ensemble d'octets distincts est uniformément distribuée, soit, réécrit de manière plus formelle : pour tout $Y_1 \neq Y_2 \neq \dots \neq Y_k \in \mathbb{F}_{2^8}$,

$$\mathbb{P}_{X_1 \neq X_2 \neq \dots \neq X_k} \{\mathbf{SBox}(X_1) = Y_1, \dots, \mathbf{SBox}(X_k) = Y_k\} = \frac{(2^8 - k)!}{2^8!}. \quad (6.2)$$

On suppose enfin que \mathbf{SB}^{-1} présente le même comportement que son inverse.

- **ShiftBytesWide (Sh)** réalise une permutation circulaire des octets d'une ligne vers la gauche d'un nombre de positions fixé dans la représentation matricielle :
 - Pour P_{1024} , les lignes 1 à 8 sont respectivement décalées de 0, 1, 2, 3, 4, 5, 6 et 11 positions comme l'illustre la figure 6.4a.
 - Pour Q_{1024} , les lignes 1 à 8 sont respectivement décalées de 1, 3, 5, 11, 0, 2, 4 et 6 positions comme l'illustre la figure 6.4b.

1. En réalité, la valeur 4 admet de rares occurrences

FIGURE 6.4 – Transformations **ShiftBytesWide**

- **MixBytes (MB)** s'applique indépendamment sur les colonnes dans la représentation matricielle. La notation **MB** fera indifféremment référence à la transformation sur l'état interne complet ou sur un ensemble quelconque de colonnes.

Cette transformation consiste en l'application d'une matrice MDS à coefficients dans \mathbb{F}_{2^8} sur une colonne de l'état interne, vecteur de $\mathbb{F}_{2^8}^8$. L'image par **MB** d'une colonne avec $k > 0$ octets non nuls possède alors par application de la borne de Singleton des octets non nuls en au moins $9 - k$ positions.

La matrice inverse d'une matrice MDS est MDS, \mathbf{MB}^{-1} présente donc le même comportement que son inverse.

Remarque 24. *L'usage veut que le dernier tour des structures de type AES évite l'application de la dernière transformation **MB**. La plupart des attaques rebond connues considèrent que les tours sont complets. Pour être cohérent avec la littérature, le choix a été fait ici de ne considérer que des tours complets : avec la transformation **MB**. Un regard jeté sur les chemins différentiels tronqués considérés dans les attaques rebond présentées dans cette thèse suffira à convaincre le lecteur que ceci relève du détail dans ces attaques.*

6.2 Algorithmes de différenciations

La notion de non différentiabilité est devenue très populaire parmi les concepteurs de fonctions de hachage avec l'avènement de la construction éponge [Ber+]. Cette notion fournit des preuves de sécurité pour les fonctions de hachage qui itèrent une permutation cryptographique dont l'état interne est partiellement modifié avec le message durant une phase d'absorption, puis partiellement extrait pour construire la sortie de la fonction de hachage durant une phase d'essorage. Leur validité repose sur l'hypothèse d'un comportement de la permutation sous-jacente idéal dans le modèle de l'*oracle aléatoire*. À l'instar des fonctions éponges, le mode d'opération de Grøstl₅₁₂ dispose d'une preuve de sécurité qui repose sur le caractère idéal des permutations internes de Grøstl₅₁₂ dans le modèle de l'*oracle aléatoire*.

Nous présentons le principe des algorithmes de différenciation, qui montrent les incohérences que présentent une permutation avec un tirage aléatoire dans l'ensemble des permutations. Nous présentons alors successivement deux problèmes combinatoires, le problème des anniversaires et le problème des anniversaires limités, à partir desquels sont construites nos algorithmes de différenciation.

6.2.1 Modèle de l'oracle aléatoire

Objet élémentaire du domaine des preuves de sécurité de mécanismes cryptographiques, l'oracle aléatoire modélise une fonction disponible uniquement en boîte noire, *id est* dont on ne connaît que les entrées et les sorties, dont les sorties sont indépendantes et identiquement distribuées. Dans le cadre des permutations, le modèle de l'oracle aléatoire consiste à considérer une permutation tirée au hasard dans l'ensemble des permutations. Il a été formalisé par Bellare et Rogaway [Bellare/Rogaway:1993] à partir d'idées initiées par Fiat et Shamir [Fiat/Shamir:1986]. Une permutation est donc *idéale* dans le modèle de l'oracle aléatoire s'il n'existe pas de procédure qui permette de discriminer si on a affaire à un oracle aléatoire ou à la permutation.

Les *algorithmes de différenciation* sont des algorithmes qui exhibent un caractère non cohérent de la permutation avec un tirage aléatoire. De tels algorithmes ne menacent pas directement la résistance des fonctions de hachages. Ils invalident cependant la preuve de sécurité qui les justifie et constituent un premier pas vers une cryptanalyse globale. Moralement, ils réduisent la confiance dans les primitives qu'ils attaquent. Pour montrer cette différenciabilité, une méthode consiste à construire une instance d'un problème combinatoire et de fournir un algorithme qui résout cette instance avec plus d'efficacité que la résolution générique associée.

6.2.2 Paradoxe des anniversaires généralisés

Le paradoxe des anniversaires, à l'origine de l'attaque générique sur la recherche de collisions se généralise à un problème plus global, sur lequel on construit des algorithmes de différenciation.

Problème 1. Anniversaires(P, E_1, E_2) : *Étant donné une permutation P sur n bits et deux \mathbb{F}_2 -sous-espaces linéaires E_1 et E_2 de dimensions respectives n_1 et n_2 vérifiant $n_1 \leq n_2$ et $n \leq 2n_1 + n_2$, déterminer un couple d'entrées (X, X') tel que $X \oplus X' \in E_1$ et $P(X) \oplus P(X') \in E_2$.*

L'algorithme de recherche de collisions générique se généralise alors comme suit. Deux messages possèdent une empreinte dont la différence est nulle sur le supplémentaire de E_2 avec une probabilité 2^{n_2-n} . Ainsi, considérer $2^{\frac{n-n_2}{2}}$ éléments dans E_1 , collecte possible pour $2n_1 + n_2 \geq n$, permet de construire 2^{n-n_2} couples d'entrées. L'hypothèse d'indépendance des tests permet d'assurer qu'il existe un couple dont la différence des images appartient à E_2 avec probabilité supérieure à 2^{-1} . La complexité calculatoire de l'attaque générique est donc d'environ $2^{\frac{n-n_2}{2}}$.

Remarque 25. *Les hypothèses du problème ne jouent pas le même rôle :*

- *L'hypothèse $n_1 \leq n_2$ n'est pas nécessaire et ne sert qu'à faciliter l'expression du problème. Lorsqu'elle n'est pas vérifiée, il suffit de considérer le problème symétrique pour la permutation inverse de P .*
- *L'hypothèse $n \leq 2n_1 + n_2$ est, elle, nécessaire pour assurer qu'il existe suffisamment d'échantillons pour réaliser la collision souhaitée. On peut supprimer cette hypothèse avec pour conséquence un traitement différencié selon les cas, c'est le sujet du problème des anniversaires limité suivant.*

6.2.3 Problème des anniversaires limité

Henri Gilbert et Thomas Peyrin ont introduit la notion de différenciation liée au problème des *anniversaires limité*. Ledit problème consiste en la détermination de couples d'entrées dont la

différence appartient à un sous-espace pré-défini et dont la différence des images par la permutation appartient à un autre sous-espace pré-défini, sans que ceux-ci ne soient contraints quant à leurs dimensions :

Problème 2. *Anniversaires-limité*(P, E_1, E_2) : *Étant donné une permutation P et deux \mathbb{F}_2 -sous-espaces linéaires E_1 et E_2 de dimensions respectives n_1 et n_2 , déterminer un couple d'entrées (X, X') tel que $X \oplus X' \in E_1$ et $P(X) \oplus P(X') \in E_2$.*

Le meilleur algorithme de résolution générique consiste en l'extension naïve de l'algorithme de résolution générique du problème des anniversaires. Deux entrées possèdent des images dont la différence appartient au sous-espace linéaire E_2 avec probabilité $2^{-(n-n_2)}$. Il existe 2^{n_1} valeurs de différence appartenant au sous-espace linéaire E_1 réalisables par $\binom{2^{n_1}}{2}$ couples d'entrées. L'hypothèse d'indépendance des tests nous permet d'affirmer qu'il existe alors avec probabilité $2^{2n_1-1-(n-n_2)}$ un couple parmi ceux-ci qui engendre une collision.

Il convient alors de distinguer les cas :

- Si $n \leq 2n_1 + n_2$, l'instance du problème des anniversaires limités se révèle une instance du problème des anniversaires standard. La complexité est alors de $2^{\frac{n-n_2}{2}}$.
- Si $n > 2n_1 + n_2$, reproduire la recherche de collisions pour $2^{n-2n_1-n_2}$ complétions avec des valeurs différentes sur le supplémentaire de E_1 permet d'obtenir la collision désirée avec probabilité $\frac{1}{2}$. La complexité est alors de $2^{n-2n_1-n_2} \cdot 2^{n_1} = 2^{n-n_1-n_2}$.

La complexité de cet algorithme, dont l'optimalité a été prouvée par Iwamoto, Peyrin et Sasaki dans [Iwamoto/Peyrin/Sasaki:2013], est alors résumée dans le Théorème 37 :

Théorème 37. *Pour une permutation P sur n bits, un \mathbb{F}_2 -sous-espace linéaire E_1 de dimension n_1 , un \mathbb{F}_2 -sous-espace vectoriel E_2 de dimension n_2 tels que $n_1 \leq n_2$, la complexité calculatoire \mathcal{C} de l'algorithme de résolution générique de l'instance du problème **Anniversaires-limité**(P, E_1, E_2) vérifie :*

$$\log_2(\mathcal{C}) \approx \begin{cases} (n - n_2)/2 & \text{si } n < 2n_1 + n_2, \\ n - n_1 - n_2 & \text{sinon.} \end{cases}$$

Remarque 26. *Les comportements non cohérents d'un tirage aléatoire que nous allons exhiber pour les permutations internes de Grøstl_{512} sont basés sur des instances du problème des anniversaires limité. La cryptanalyse consiste donc ici en la détermination d'une instance de ce problème et la conception d'un algorithme de différenciation, qui résout cette instance avec une complexité calculatoire plus faible que celle de l'algorithme de résolution générique. La résolution d'instances du problème des anniversaires limité n'est pas le seul type d'attaques de différenciation. [Jean/NayaPlasencia/Peyrin:2013 ; Lam+09 ; Gil14] sont autant de références d'attaques de différenciation basées sur d'autres comportements non cohérents d'un tirage aléatoire.*

6.3 Chemins différentiels tronqués

Les chemins différentiels tronqués ont été introduits par Knudsen [Knu95]. Seul le caractère actif d'une position d'octet est considéré, *id est* savoir si cette position est affectée par une valeur de différence non nulle. De nombreuses cryptanalyses de fonctions de hachage ont été analysées sous cet angle, parmi lesquelles on peut mentionner les suivantes : [Pey07 ; Mat+09 ; Men+09a ; JF11 ; JNS12 ; JNP14].

Après la présentation des principes d'un chemin différentiel tronqué, nous présenterons comment concevoir à partir de ces derniers des attaques rebond, algorithmes qui différencient une permutation d'un tirage aléatoire.

6.3.1 Principe

Le paradigme des chemins différentiels tronqués neutralise certains effets de prolifération des chemins différentiels classiques. Le chemin différentiel tronqué ne conserve pas en mémoire la valeur d'une différence intermédiaire particulière et peut, à ce titre, être considéré comme un faisceau de chemins différentiels classiques. Les chemins différentiels tronqués ont été développés pour l'analyse des primitives dont l'architecture s'inspire de l'AES. En effet, l'opération non linéaire, seule responsable de la prolifération des chemins différentiels classiques, agit de manière déterministe sur les chemins différentiels tronqués. La seule opération non déterministe de la fonction de tour consiste en l'opération de mélange des octets, **MixBytes** dans Grøstl_{512} . Les attaques rebond exploitent des failles structurelles qui apparaissent au travers de ces chemins différentiels tronqués. La compétition du NIST a été le terrain de jeu idéal pour étendre, développer et appliquer ces techniques, nombreux parmi les candidats sont en effet construits à partir de permutations de type AES. Des techniques comme le *start-from-the-middle*, introduit dans [Men+09a], les descriptions macroscopiques à base de Super-Boîtes S, dues à Daemen et Rijmen [DR06] et utilisées notamment dans [GP10; Gil14], sont autant de munitions dans l'arsenal des cryptanalistes. Des améliorations plus poussées dans [Nay11] ou [Sas+10] ont pavé le chemin vers les variantes les plus récentes.

6.3.2 Attaque rebond

Les attaques rebond, introduites dans [Men+09b], ont été une réponse de la cryptanalyse à la généralisation des permutations de type AES dans les conceptions de fonctions de hachage. Elles consistent en la détermination de couples de valeurs intermédiaire de l'état interne au milieu des tours de la permutation, dont les différences se propagent vers l'entrée et la sortie de la permutation selon un chemin différentiel tronqué de grande probabilité. La procédure d'une telle cryptanalyse comprend deux volets : chercher un chemin différentiel tronqué probable et concevoir des algorithmes efficaces pour déterminer des couples de valeurs d'état interne dont les différences intermédiaires à chaque étape du chiffrement sont compatibles avec le chemin différentiel tronqué. Lorsque l'algorithme en question nécessite une complexité inférieure à celle de l'algorithme de résolution générique de l'instance du problème des anniversaires limité induit par les valeurs de différences autorisées, compatibles avec le chemin différentiel tronqué en entrée et en sortie de la permutation, il s'agit alors d'un algorithme de différenciation.

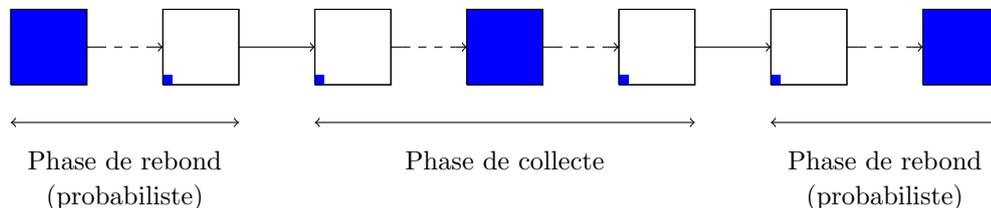


FIGURE 6.5 – Principe de l'attaque rebond.

L'attaque rebond, comme son nom l'indique, suppose un chemin différentiel tronqué sous-jacent conçu comme suit : on cherche deux différentielles tronquées dont les propagations libres respectives, les *avalanches*, en amont et en aval se rejoignent en un état interne totalement actif. Ce chemin rebondit : l'état interne actif réalise comme une symétrie centrale, les avalanches à partir des deux différentielles tronquées initiales étant identiques. Ces deux différentielles se propagent ensuite vers les premier et dernier tours, d'abord de façon probabiliste, puis selon une avalanche libre, mais toujours avec la même symétrie. Un découpage dans le traitement algorithmique se dégage naturellement de cette construction du chemin différentiel tronqué. Une étape de collecte consiste à construire de nombreux couples de valeurs d'état interne intermédiaires dont les différences intermédiaires successives lorsque propagés en amont et en aval sont compatibles avec les tours centraux du chemin différentiel tronqué, transitions de très faible probabilité. Alors survient une phase de rebond qui consiste à déterminer parmi les couples de valeurs d'état interne collectés, ceux satisfaisant simultanément les transitions probabilistes de grande probabilité restantes.

Si l'attaque rebond appartient à la famille des attaques différentielles dirigées contre les fonctions de hachage, elle est également exploitée pour réaliser des analyses de schémas de chiffrement par blocs dans le modèle dit *open-key* dans lequel l'attaquant connaît la clef de chiffrement et s'astreint à montrer le caractère incohérent dudit schéma de chiffrement d'un tirage aléatoire dans l'ensemble des permutations.

Chapitre 7

Attaque sur 10 tours, risques d’avalanche

Sommaire

7.1	Un chemin différentiel tronqué naturel	108
7.1.1	Motif différentiel	108
7.1.2	Plausibilité de réalisation	110
7.1.3	Instance du problème des anniversaires limité induite	110
7.2	Algorithme de différenciation	111
7.2.1	Esquisse de l’algorithme de différenciation	111
7.2.2	Étape 3 : Fusion de listes	113

Aucun résultat original n’est revendiqué pour ce chapitre. Présentation de la meilleure attaque connue sur Grøstl_{512} jusqu’aux travaux du chapitre suivant, il permet l’exposition de techniques mises en oeuvre qui seront réutilisées dans note attaque, comme par exemple les méthodes de fusion de liste. L’attaque présentée ici prouve qu’une version à nombre de tours réduit, 10 sur 14, des permutations internes de Grøstl_{512} est différentiable d’un tirage aléatoire. Un chemin différentiel tronqué muni d’un algorithme de différenciation va permettre de résoudre une instance du problème des anniversaires limités pour ces permutations internes. L’hypothèse du comportement idéal de ces permutations invalidée, c’est la preuve de sécurité de la fonction de hachage construite sur ces versions à nombre de tours réduit qui disparaît. On se focalisera dans la suite arbitrairement sur la permutation P_{1024} bien que tous les résultats se transposent à Q_{1024} . Comme déjà mentionné, plusieurs articles successifs ont développé les idées qui sont présentées ici. La référence incontournable pour cette attaque est la thèse de Jérémy Jean [Jea13].

Ce chapitre présente un chemin différentiel tronqué qui émerge naturellement de considérations heuristique de minimisation des transitions probabilistes réalisée par des avalanches libres. Il présente alors le détail des différentes étapes de l’algorithme de différenciation basé sur ce chemin différentiel tronqué.

7.1 Un chemin différentiel tronqué naturel

Contre la *wide trail strategy* consiste à chercher les chemins différentiels les plus probables. Chercher les chemins différentiels les plus probables est difficile de par la prolifération des chemins. Cette prolifération est réduite dans le paradigme des chemins différentiels tronqués. On cherche alors seulement les chemins différentiels tronqués qui activent le plus petit nombre possible de boîtes S. La minimisation du nombre de boîtes actives fait naturellement apparaître pour des petits nombres de tours des phénomènes d'avalanche. Ceux-ci disparaissent lorsqu'on considère des plus grands nombre de tours. Cependant, dans le paradigme des chemins différentiels tronqués, on cherche à minimiser les transitions probabilistes. Les phénomènes d'avalanche libre sont déterministes et donc avantageux dans l'optique de construire des couples qui les réalisent. L'exploitation à l'extrême de cette heuristique sur 10 tours fait apparaître le chemin différentiel tronqué de la Figure 7.1.

Après avoir présenté le chemin différentiel tronqué utilisé, et vérifié la probabilité qu'il soit réalisé, nous en déduisons une instance du problème des anniversaire limité.

7.1.1 Motif différentiel

L'avalanche libre d'une différentielle tronquée avec un seul octet actif sur trois tours active tout l'état interne. Considérer un état interne actif à la fois comme avalanche d'une différentielle tronquée avec un unique octet actif en amont et en aval suggère une attaque rebond dont les phase de collecte et phase de rebond sont alors presque induites. L'avalanche libre complète nécessite trois tours. La connaissance de la valeur d'une différence sur un unique octet disparaît cependant avec l'étape non linéaire de la fonction de tour à moins de considérer un couple de valeurs d'octet compatible avec cette valeur de différence. On préfère considérer deux colonnes actives qui par avalanche libre sur deux tours respectivement en amont et en aval, se rejoignent en un état interne actif. On construit alors des couples d'état internes satisfaisant ces deux avalanches libres et on considère les transitions d'une colonne active vers un octet actif comme des passages probabilistes, inclus dans la phase de rebond. La Figure 7.1 fait apparaître le chemin différentiel tronqué né de ces considérations heuristiques. Les cellules bleues correspondent à des octets actifs tandis que les cellules blanches correspondent à des octets non actifs.

Toutes les transitions d'une telle description de chemin différentiel tronqué sont déterministes exceptées celles induites par les transformations **MB**. Ces dernières sont en effet des transitions tronquées probabilistes : une valeur de colonne δ , non nulle en u positions d'octets fixées possède une image $\mathbf{MB}(\delta)$ nulle en v positions d'octets fixées avec probabilité β^{-v} tant que la borne de Singleton est vérifiée, et 0 sinon.

$$\mathbb{P} \left(\begin{array}{l} \mathbf{MB}(\delta) \text{ s'annule en } v \text{ positions d'octets fixées} \\ | \delta \text{ est non nul en } u \text{ positions d'octets fixées} \end{array} \right) = \begin{cases} \beta^{-v} & \text{si } u + (8 - v) \geq 9, \\ 0 & \text{sinon.} \end{cases} \quad (7.1)$$

La théorie des codes assure un comportement identique de la transformation \mathbf{MB}^{-1} .

Remarque 27. *L'octet est l'atome de notre analyse. Mesure fondamentale de la complexité des différentes étapes algorithmiques, on écrira $\beta = 2^8$. Une notation légèrement abusive sera de généraliser la notation de Landau pour notre utilisation dans laquelle β est fixe. Cependant, le lecteur tolérant se satisfera de savoir vérifié que $O(\beta^n)$ est bien inférieur à β^{n+1} .*

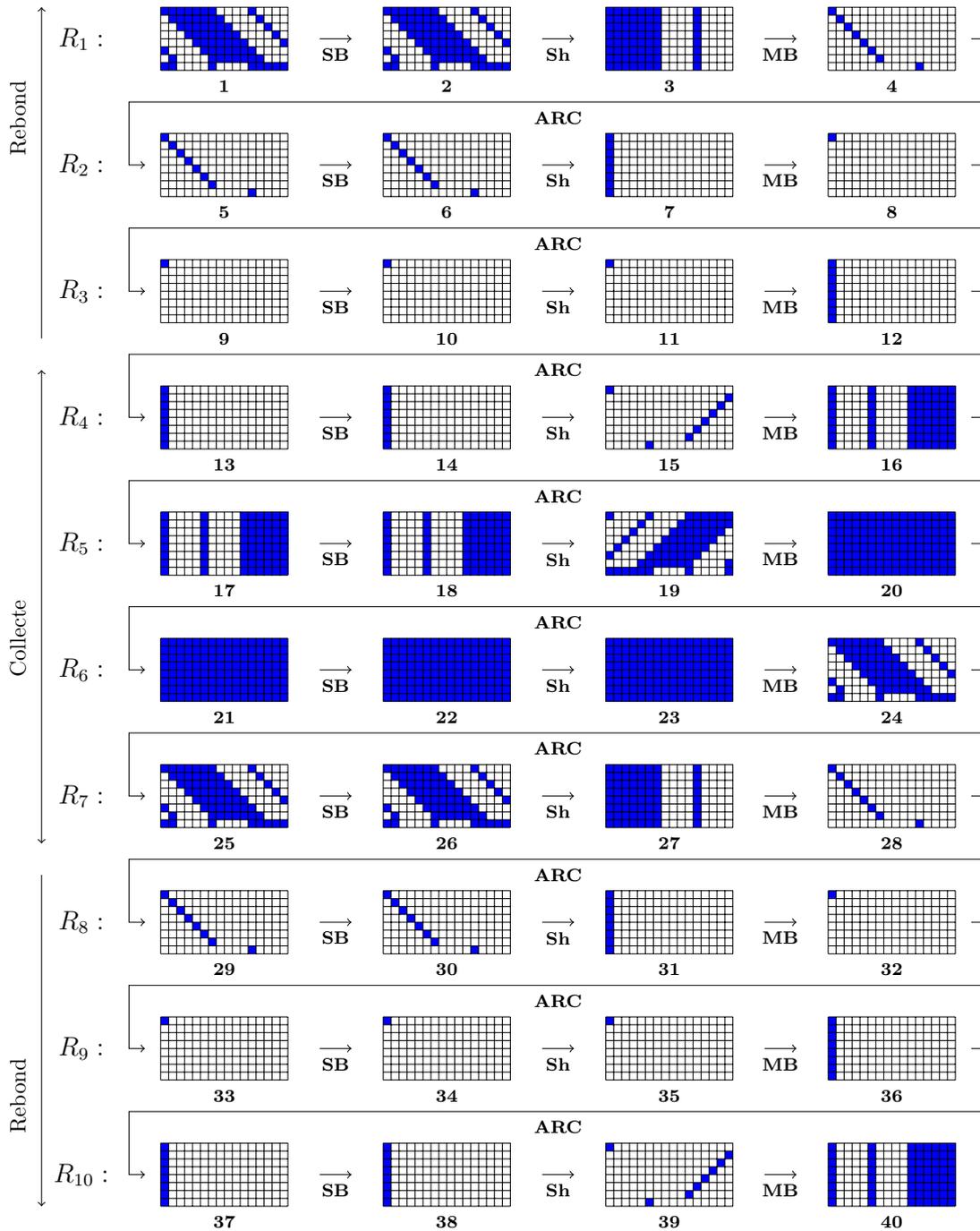


FIGURE 7.1 – Chemin différentiel tronqué sur 10 tours de P_{1024} de Grøstl₅₁₂.

7.1.2 Plausibilité de réalisation

La plupart des attaques rebond reposent sur des chemins différentiels tronqués similaires, pour lesquels une différence diffuse vers un état dont toutes les positions correspondent à des octets actifs après une propagation de trois tours en amont ou en aval. Le profil des octets actifs en début de chaque tour constitue pour ces chemins différentiels tronqués une représentation classique. Le profil associé au chemin différentiel tronqué sur 10 tours introduit dans la Figure 7.1 est alors :

$$64 \xrightarrow{R_1} 8 \xrightarrow{R_2} 1 \xrightarrow{R_3} 8 \xrightarrow{R_4} 64 \xrightarrow{R_5} 128 \xrightarrow{R_6} 64 \xrightarrow{R_7} 8 \xrightarrow{R_8} 1 \xrightarrow{R_9} 8 \xrightarrow{R_{10}} 64.$$

Une première analyse consiste à évaluer la plausibilité qu'il existe un couple de valeurs d'état interne tel que toutes les différences intermédiaires propagées à travers les transformations des fonctions de tour sont compatibles avec les motifs du chemin différentiel tronqué. Il existe $\beta^{64} \cdot \beta^{128} = \beta^{192}$ couples de valeurs d'état interne dont la différence respecte le motif d'octets actifs en entrée. La propagation d'un tel motif en accord avec le chemin différentiel tronqué pour les tours 1 et 7 est réalisée avec probabilité β^{-56} , la propagation pour les tours 2 et 8 est quant à elle réalisée avec probabilité β^{-7} . Enfin, la propagation pour le tour 6 est réalisée avec probabilité β^{-64} . Le comportement déterministe des autres transformations assure qu'il existe en moyenne $\beta^{192} \cdot \beta^{-190} = \beta^2$ couples de valeurs d'état interne compatibles avec le chemin différentiel tronqué sur 10 tours. L'enjeu est alors de les construire avec une complexité calculatoire inférieure à la complexité de l'algorithme de résolution générique de l'instance du problème des anniversaires limité induite pour obtenir un algorithme de différenciation.

7.1.3 Instance du problème des anniversaires limité induite

Le chemin différentiel sur 10 tours de la Figure 7.1 induit une instance du problème des anniversaires limité. La permutation P est ici une version à 10 tours de la permutation interne P_{1024} de Grøstl₅₁₂. L'espace linéaire E_1 correspond ici aux différences compatibles avec le motif différentiel tronqué 1 de la Figure 7.1 reproduit ci-dessous.

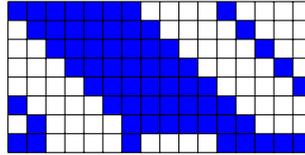


FIGURE 7.2 – Espace linéaire E_1

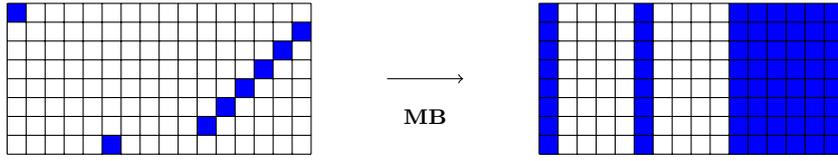
64 octets de différence sont possibles, ce qui correspond ainsi à un espace linéaire de dimension :

$$n_1 = \dim(E_1) = 64 \cdot 8$$

L'espace linéaire E_2 correspond ici aux différences compatibles avec l'image par la transformation linéaire inversible \mathbf{MB} du motif différentiel tronqué 39 de la Figure 7.1 reproduit ci-dessous.

8 octets de différence sont possibles, ce qui correspond ainsi à un espace linéaire de dimension :

$$n_2 = \dim(E_2) = 8 \cdot 8$$

FIGURE 7.3 – Espace linéaire E_2

On se retrouve donc avec une instance du problème des anniversaires limité dont l'algorithme de résolution générique présente une complexité calculatoire de :

$$\log_2(\mathcal{C}_{gen}) = (128 - 64 - 8) \cdot 8 = 56 \cdot 8 = 448$$

7.2 Algorithme de différenciation

Avant de pénétrer les entrailles de l'algorithme de différenciation, il est nécessaire d'introduire plusieurs notations. Les valeurs d'octets sont des éléments dans $\mathcal{B} = \mathbb{F}_{2^8}$. Les valeurs de colonnes sont des éléments dans $\mathcal{C} = \mathcal{B}^8$. Pour $X \in \mathcal{C}$, on écrit alors X_i la i^{eme} coordonnée de X , soit l'octet en i^{eme} position dans la colonne. Les valeurs d'état interne seront indifféremment des éléments dans \mathcal{S} , \mathcal{B}^{128} ou \mathcal{C}^{16} . Pour $Y \in \mathcal{S}$, on note $Y_{i,j}$ la valeur d'octet appartenant à la i^{eme} ligne et à la j^{eme} colonne. Pour la suite, toutes les références à des motifs sont liées à la Figure 7.1. Pour $i \in \{1, \dots, 14\}$, on note \mathcal{P}_i le \mathcal{B} -sous-espace vectoriel des valeurs différentielles compatibles avec le motif i , *id est* dont les positions d'octets actifs sont incluses dans les positions hachurées sur le motif i . On note alors I_i l'ensemble des indices de colonnes actives, colonnes qui contiennent au moins un octet actif. On écrit également δ_j la restriction d'une valeur différentielle δ à la j^{eme} colonne, plus généralement δ_I la restriction de δ aux positions d'octets incluses dans I et, par extension, on écrit $(\mathcal{P}_i)_j$ le sous-espace linéaire de \mathcal{C} compatible avec la j^{eme} colonne du motif i .

On présente en premier lieu les différentes étapes de l'algorithme avant de donner les explications précises de la seule étape non triviale. Toutes les références à des motifs dans cette esquisse se lisent sur la Figure 7.1.

7.2.1 Esquisse de l'algorithme de différenciation

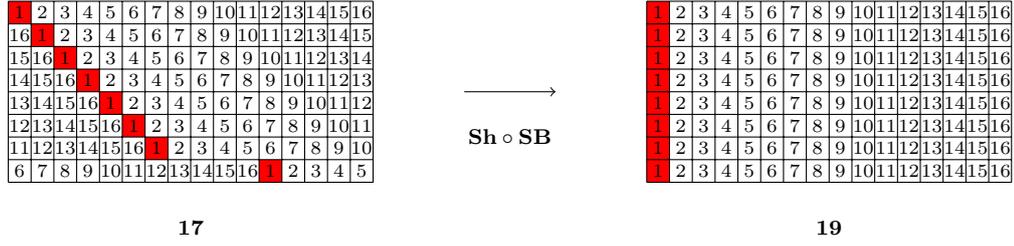
Étape 1. Choisir arbitrairement $\delta_1 \in \mathcal{P}_{14}$, une valeur de différence compatible avec le motif 14. Calculer son image δ'_1 par la transformation $\mathbf{ARC} \circ \mathbf{MB} \circ \mathbf{Sh}$:

$$\delta'_1 = (\mathbf{ARC} \circ \mathbf{MB} \circ \mathbf{Sh})(\delta_1)$$

Par symétrie, choisir arbitrairement $\delta_2 \in \mathcal{P}_{29}$, une valeur de différence compatible avec le motif 29. Calculer son image δ'_2 par la transformation $(\mathbf{ARC} \circ \mathbf{MB} \circ \mathbf{Sh})^{-1}$:

$$\delta'_2 = (\mathbf{ARC} \circ \mathbf{MB} \circ \mathbf{Sh})^{-1}(\delta_2)$$

Étape 2. Construire la partition I_1, \dots, I_{16} de l'état interne en ensembles de 8 octets dont l'image par la transformation \mathbf{Sh} correspond à la partition induite par les 16 colonnes de l'état interne : $\mathbf{Sh}(I_j) = \mathcal{C}_i$. Cette partition est rendue explicite par la Figure 7.4, l'ensemble d'indices I_1 y a été arbitrairement représenté par les cellules rouges pour plus de lisibilité.

FIGURE 7.4 – Partition selon les indices I_1, \dots, I_{16} .

Construire ensuite les 16 listes L_i contenant tous les couples de valeurs de 8 octets dont les différences égalent les restrictions de δ'_1 induites par cette partition : $(\delta'_1)_{|I_i}$. Construire alors à partir des listes L_i les 16 listes L'_i contenant les couples images des couples appartenant à L_i par la transformation $\mathbf{Sh} \circ \mathbf{SB} \circ \mathbf{ARC} \circ \mathbf{MB} \circ \mathbf{Sh}$. Ces listes sont toutes de cardinal β^8 . Cette opération nécessite une complexité calculatoire d'environ β^8 pour un coût mémoire de l'ordre de $16 \cdot \beta^8$.

Par symétrie, construire les 16 listes R_i contenant tous les couples de valeurs de colonnes dont les différences égalent les restrictions de δ'_2 sur les colonnes : $(\delta'_2)_{|i}$ puis construire alors à partir des listes R_i les 16 listes R'_i contenant les couples images par la transformation $(\mathbf{MB} \circ \mathbf{ARC} \circ \mathbf{SB})^{-1}$ des couples appartenant à R_i .

Étape 3. Toute combinaison d'éléments dans le produit cartésien des 16 listes L'_i engendre un couple de valeurs d'état interne dont les différences intermédiaires sont compatibles avec le choix de δ_1 entre les motifs 14 et 23. Par symétrie, toute combinaison d'éléments dans le produit cartésien des 16 listes R'_i engendre un couple de valeurs d'état interne dont les différences intermédiaires sont compatibles avec le choix de δ_2 entre les motifs 23 et 29. Grâce à un algorithme de type Guess and Determine, déterminer deux telles combinaisons compatibles. Cette étape résulte en un couple de valeurs d'état interne dont les différences intermédiaires sont compatibles avec le chemin différentiel tronqué entre les motifs 14 et 29 avec une complexité calculatoire d'environ β^{35} . Cette étape algorithmique fait l'objet d'explications détaillées présentées immédiatement après cette esquisse.

Étape 4. Reproduire les 3 étapes précédentes pour collecter β^{14} exemplaires de couples de valeurs d'état interne dont les différences intermédiaires sont compatibles avec le chemin différentiel tronqué entre les motifs 14 et 29. Ceci conclut la phase de collecte qui nécessite une complexité calculatoire totale d'environ 2^{49} .

Étape 5. L'énumération des β^{14} couples de valeurs d'état interne obtenus pendant la phase de collecte permet de déterminer un couple parmi ceux-ci dont les différences intermédiaires satisfont simultanément les deux transitions probabilistes indépendantes restantes : à travers la transformation \mathbf{MB}^{-1} entre les motifs 12 et 11 et à travers la transformation \mathbf{MB} entre les motifs 31 et 32. Cette phase de rebond est alors réalisée avec une complexité calculatoire d'environ $O(\beta^{14})$.

Pour résumer, cet algorithme construit un couple de valeurs d'état interne dont la propagation à travers 10 tours de P_{1024} présente des différences successives compatibles avec le chemin différentiel tronqué de la Figure 7.1. Ceci est réalisé avec une complexité calculatoire d'environ $O(\beta^{49}) \simeq 2^{392} < 2^{448}$ et une complexité en mémoire de l'ordre de $O(\beta^8) \simeq 2^{64}$. L'attaque

générique sur l'instance du problème des anniversaires limité induite par les motifs d'entrée et de sortie présente une complexité calculatoire d'environ β^{58} . Il s'agit donc bien d'un algorithme de différenciation.

7.2.2 Étape 3 : Fusion de listes

On dispose de 32 listes de β^8 éléments chacune. On souhaite trouver une combinaison de ces 32 listes qui s'accordent en 256 valeurs d'octets. Comme $\beta^{2 \cdot 128} \cdot \beta^{-2 \cdot 128} = 1$, on s'attend alors à déterminer une unique telle combinaison. L'algorithme naïf pour déterminer cette combinaison consiste à tester chaque couple de valeurs d'état interne, combinaison d'éléments du produit cartésien des L'_i pour savoir s'il appartient au produit cartésien des R'_i . Le coût d'un tel algorithme est d'environ β^{128} . Un algorithme de type Guess and Determine peut réaliser la même opération pour un coût calculatoire de l'ordre de β^{35} opérations.

On construit tous les couples de valeurs d'état interne partiel induits par une combinaison d'éléments dans les listes R'_2, R'_3, R'_4 et R'_5 . Ces listes étant de cardinal β^8 , il existe β^{32} telles combinaisons. Chacune de ces combinaisons induit un couple de valeurs sur 32 positions d'octets, illustrées par les cellules rouges de la figure 7.5

Pour chacune des listes L'_5, L'_6, L'_7 et L'_8 , 8 valeurs d'octets correspondant à 4 positions d'octets sont déterminés par le choix précédent. Ces 8 valeurs forment autant de contraintes pour chacune des 4 listes à partir desquelles on peut filtrer les candidats potentiels parmi les β^8 possibilités que contiennent chacune, indépendamment, pour déterminer un unique élément compatible par liste. Les 4 éléments ainsi déterminés de L'_5, L'_6, L'_7 et L'_8 induisent à leur tour des couples de valeurs pour 16 nouvelles positions d'octets. La Figure 7.5 représente l'état induit une fois déterminé ces uniques éléments, les cellules oranges illustrant les couples de valeurs alors déterminés.

Un coup d'oeil sur le motif 24 de la Figure 7.1 nous permet de nous rappeler que les listes R'_1 et R'_{16} ne présentent respectivement que 3 et 2 degrés de libertés pour ce qui concerne les valeurs de différences que leurs éléments induisent. Le nombre maximal de valeurs de différences indépendantes est donc atteint dans les deux cas et on peut donc déterminer les valeurs de différences pour les autres positions d'octets de ces listes, illustrées par les cellules vertes de la Figure 7.5.

Les éléments de L'_4 doivent donc satisfaire 8 contraintes déjà déterminées, à partir desquelles on filtre l'unique élément compatible que cette liste contient. 4 nouvelles positions d'octet sont alors complètement déterminées, illustrées par les cellules violettes de la Figure 7.5. On peut alors déterminer les valeurs de différence pour R'_{15} puisque le sous-espace linéaire correspondant est de dimension 2, comme l'atteste le motif 24 de la Figure 7.1. Ces contraintes apparaissent en turquoise dans la Figure 7.5

On considère alors tous les choix possibles pour R'_6 . Comme 6 octets sont déjà contraints, seulement β^2 éléments dans cette liste sont conformes aux contraintes induites par le choix précédent. Après la filtration de ces éléments compatibles, leur énumération engendre alors une complexité calculatoire globale de $\beta^{32+2} = \beta^{34}$. Toutes les positions dont les couples de valeurs sont déterminés sont illustrées par les cellules rouges sur la Figure 7.6, les positions dont uniquement les valeurs de différences sont contraintes apparaissent quant à elles vertes.

Les éléments de la liste L'_9 doivent donc vérifier 8 contraintes d'octets de sorte qu'un unique élément peut en être filtré, qui détermine à son tour les couples de valeurs pour 4 nouvelles positions d'octets, illustrées par les cellules oranges de la Figure 7.6. Les valeurs de différences

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1
3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2
4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3
5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4
6	7	8	9	10	11	12	13	14	15	16	1	2	3	4	5
7	8	9	10	11	12	13	14	15	16	1	2	3	4	5	6
12	13	14	15	16	1	2	3	4	5	6	7	8	9	10	11

23

FIGURE 7.5 – Contraintes imposées par un choix d'élément dans $R'_2 \times R'_3 \times R'_4 \times R'_5$.

pour R'_{14} sont alors déterminées et sont illustrées par les cellules vertes dans la Figure 7.6.

Il est donc possible de filtrer un unique élément compatible dans la liste L'_3 dont les éléments doivent vérifier 8 contraintes, qui détermine à son tour les couples de valeurs pour 6 nouvelles positions d'octets, illustrées par les cellules violettes de la Figure 7.6.

Les valeurs de différences des éléments compatibles dans la liste R'_1 sont alors complètement déterminées. Il reste alors $\beta^{8-3} = \beta^5$ éléments compatibles qui doivent vérifier 5 contraintes d'octets. On est donc capable de déterminer toutes les valeurs encore inconnues pour R'_1 , ce qui détermine les couples de valeurs dans les positions d'octets illustrées par des cellules roses sur la Figure 7.6.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1
3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2
4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3
5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4
6	7	8	9	10	11	12	13	14	15	16	1	2	3	4	5
7	8	9	10	11	12	13	14	15	16	1	2	3	4	5	6
12	13	14	15	16	1	2	3	4	5	6	7	8	9	10	11

23

FIGURE 7.6 – Contraintes imposées par un choix d'élément dans $R'_2 \times R'_3 \times R'_4 \times R'_5$ puis par un choix compatible dans R'_6 .

On considère alors tous les choix possibles pour L'_1 . 2 positions d'octets sont complètement déterminées et 3 positions d'octet ont leurs valeurs de différence déterminées, β éléments dans cette liste sont donc conformes aux contraintes induites par les choix précédents. Après la filtration de ces éléments compatibles, leur énumération engendre alors une complexité calculatoire globale de $\beta^{34+1} = \beta^{35}$. Toutes les positions dont les couples de valeurs sont déterminés sont illustrées par les cellules rouges sur la Figure 7.7, les positions dont uniquement les valeurs de différences sont contraintes apparaissent quant à elles vertes.

Les valeurs de différences des éléments compatibles dans la liste R'_{13} sont alors complètement déterminées, illustrées par les cellules cyan de la Figure 7.7. Cette détermination engendre une nouvelle contrainte d'octet supplémentaire qui permet alors la filtration complète de la liste L'_2 pour obtenir un unique élément compatible. Cette élément induit des contraintes pour trois nouvelles positions d'octets, illustrées par les cellules oranges de la Figure 7.7.

Les éléments de la liste R'_7 dont 4 positions d'octets sont complètement déterminées induisant 8 contraintes d'octets, comme ceux de la liste R'_{16} dont les valeurs de différences sont contraintes, tout comme 6 contraintes d'octets, doivent donc vérifier 8 contraintes d'octets de sorte qu'un unique élément peut en être filtré pour chacune. Les positions d'octets alors déterminées sont illustrées par les cellules violettes de la Figure 7.7.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1
3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2
4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3
5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4
6	7	8	9	10	11	12	13	14	15	16	1	2	3	4	5
7	8	9	10	11	12	13	14	15	16	1	2	3	4	5	6
12	13	14	15	16	1	2	3	4	5	6	7	8	9	10	11

23

FIGURE 7.7 – Contraintes imposées par un choix d'élément dans $R'_2 \times R'_3 \times R'_4 \times R'_5$ puis par un choix compatible dans R'_6 et enfin par un choix compatible dans L'_1 .

L'algorithme se conclut alors par la filtration probabiliste des listes non déjà déterminées :

- L'_{10} est surdéterminée d'un octet. Une valeur compatible existe avec probabilité β^{-1} .
- L'_{11} est soumise à 8 contraintes d'octets. Une valeur compatible existe.
- R'_8 est surdéterminée de deux octets. Une valeur compatible existe avec probabilité β^{-2}
- R'_9 est soumise à 8 contraintes d'octets. Une valeur compatible existe.
- R'_{15} est soumise à 8 contraintes d'octets. Une valeur compatible existe.

Il est alors possible de déterminer la valeur de différence pour toutes les positions d'octets de R'_{11} . Ceci implique que :

- L'_{12} est surdéterminée de trois octets. Une valeur compatible existe avec probabilité β^{-3} .
- L'_{16} est surdéterminée d'un octet. Une valeur compatible existe avec probabilité β^{-1} .

Les trois listes R'_{11} , R'_{13} et R'_{14} ont leurs valeurs de différences déjà déterminées. Respectivement 5, 5 et 6 contraintes d'octets permettent alors de filtrer ces listes dont β^5 , β^5 et β^6 éléments sont compatibles avec ces valeurs de différences. On filtre de ces trois listes un unique élément avec probabilité 1. Alors :

- R'_{10} est surdéterminée de deux octets. Une valeur compatible existe avec probabilité β^{-2} .
- R'_{12} est surdéterminée de deux octets. Une valeur compatible existe avec probabilité β^{-2} .

- L'_{13} est surdéterminée de huit octets. Une valeur compatible existe avec probabilité β^{-8} .
- L'_{14} est surdéterminée de huit octets. Une valeur compatible existe avec probabilité β^{-8} .
- L'_{15} est surdéterminée de huit octets. Une valeur compatible existe avec probabilité β^{-8} .

Lorsqu'il existe une combinaison d'éléments dans ces listes satisfaisant tous ces évènements probabilistes, les couples de valeurs pour toutes les positions d'octets ont été déterminées et un élément de chacune des 32 listes a été déterminé, compatible avec ces couples de valeurs.

Pour résumer, chaque combinaison d'éléments dans les listes $R'_2 \times R'_3 \times R'_4 \times R'_5$, R'_6 et L'_1 engendre un ensemble de contraintes donc cette procédure termine avec une combinaison d'éléments dans les 32 listes L'_i et R'_i avec une probabilité p :

$$p = \beta^{-1-2-3-1-2-2-8-8-8} = \beta^{-35}$$

β^{35} telles combinaisons sont énumérées. On détermine donc par cette énumération un élément compatible en moyenne. Cette fusion de listes est donc réalisée avec une complexité calculatoire d'environ β^{35} et un coût mémoire de l'ordre de β^8 .

Chapitre 8

Attaque sur 11 tours, contrôler les éléments

Sommaire

8.1	Un chemin différentiel tronqué structuré	119
8.1.1	Recherche de chemins différentiels tronqués creux	119
8.1.2	Motif différentiel	121
8.1.3	Plausibilité de réalisation	121
8.1.4	Instance du paradoxe des anniversaires généralisé induite	123
8.1.5	Description macroscopique compacte	125
8.2	Algorithme de différenciation	126
8.2.1	Esquisse de l'algorithme de différenciation	126
8.2.2	Étape 1 : Construction des bases des sous-espaces linéaires Δ_i	128
8.2.3	Étape 2 : Construction de couples de valeurs de colonnes avec différences amont/aval d'une Super-Boite S maîtrisées	129
8.2.4	Étape 3 : Construction de couples de valeurs d'état interne partiel compatible avec des transitions $\Delta_1 \rightarrow \delta_2$ et $\delta_2 \rightarrow \Delta_3$	130
8.2.5	Étape 4 : Premier recollement	131
8.2.6	Étape 5 : Deuxième recollement	133
8.2.7	Étape 6 : Troisième recollement	136

Ce chapitre constitue le cœur des travaux originaux sur les attaques rebond. Les algorithmes de différenciation liés à des chemins différentiels tronqués font davantage que simplement résoudre une instance du problème des anniversaires limités. Les couples de valeurs d'état interne qu'ils déterminent possèdent des différences intermédiaires compatibles avec l'intégralité du chemin différentiel tronqué, phénomène beaucoup moins probable que de vérifier seulement les motifs initiaux et finaux. Le véritable enjeu est donc de ne pas consommer lors d'une seule étape algorithmique la majorité des degrés de libertés à disposition. Dans cette optique, d'autres méthodes que l'heuristique présentée précédemment sont utilisées pour déterminer des chemins différentiels tronqués plus probables et éviter les recollements extrêmement contraints qu'une avalanche libre induit. Un tel chemin différentiel tronqué structuré peut être déterminé comme une solution d'un problème de programmation linéaire sous contraintes, en anglais Mixed-Integer Linear Programming problem (MILP). Les travaux de cryptanalyse exploitant les MILP se sont multipliés

récemment, mais ces résultats figurent parmi les premiers pour analyser des fonctions de hachage, mouvement initié par l'étude du candidat de la compétition SHA-3, SIMD [BFL].

Ce chapitre présente un chemin différentiel tronqué structuré issu d'une résolution d'un système linéaire sous contrainte. Il présente alors le détail des différentes étapes algorithmiques après en avoir donné une esquisse, la trame générale.

8.1 Un chemin différentiel tronqué structuré

L'attaque présentée sur 10 tours paraît difficile à prolonger pour plusieurs raisons. La première consiste en l'étude de plausibilité de ce chemin différentiel tronqué dont on a vu que seuls β^2 couples de valeurs d'état interne en moyenne le réalisent. Toute transition probabiliste supplémentaire risque de rendre l'existence même d'un couple de valeurs d'état interne compatible peu probable. Il s'avère de plus qu'en raison du motif différentiel central dense, 4 tours de phase de collecte contraignent complètement les valeurs d'état interne possibles. Ici, on considère un nouveau chemin différentiel qui ne s'étend que sur moins de la moitié de l'état interne. On obtient alors un nombre très important de couples de valeurs d'état interne qui réalisent ce chemin différentiel et on profite de leur caractérisation en termes de sous-espaces linéaires et de combinaisons d'éléments dans des listes de taille raisonnable pour filtrer parmi ces choix ceux compatibles avec un chemin différentiel tronqué sur 11 tours.

Nous allons décrire comment il est possible de déterminer le chemin différentiel tronqué que nous présentons ensuite. Nous en vérifions la plausibilité et en déduisons une instance du problème des anniversaires. Alors, nous concluons par une description macroscopique de ce chemin différentiel tronqué qui fait ressortir des sous-espaces linéaires, ce qui se révélera particulièrement efficace pour énumérer des éléments et conserver en mémoire des ensembles.

8.1.1 Recherche de chemins différentiels tronqués creux

La recherche de chemins différentiels tronqués adaptés pour réaliser des attaques rebond est fortement liée à la recherche des chemins différentiels les plus probables dans une primitive de chiffrement par bloc du type AES. Les méthodes de programmation linéaire sous contraintes se révèlent d'une efficacité redoutable pour la résolution de ce type de problèmes [Mou+12]. L'explication suit des méthodes utilisées pour forger les phases de collecte et de rebond.

Phase de collecte : On recherche des chemins différentiels tronqués sur plusieurs tours qui se propagent sur le plus petit nombre de colonnes possibles, dans l'espoir qu'un grand nombre de couples de valeurs d'état interne leur soient compatibles. Plus précisément, on cherche à minimiser le nombre de colonnes non nulles dans les 6 tours centraux. Un chemin différentiel tronqué avec 7 colonnes existe tandis que c'est infaisable pour 6 colonnes. Plutôt que de minimiser le nombre de boîtes S actives d'un chemin différentiel tronqué comme dans le contexte des schémas de chiffrement par bloc, on minimise ainsi le nombre de colonnes actives.

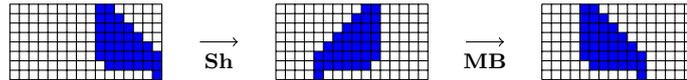
Plus formellement, on définit 7×128 variables de décision booléennes x_i , chacune desquelles représentant un octet des 7 registres de la permutation P_{1024} restreinte sur 6 tours : le registre d'entrée, les 5 registres intermédiaires et le registre de sortie. Ces variables encodent un chemin différentiel tronqué, la variable x_i égale 0 ou 1 selon que l'octet qu'elle représente est actif ou non. Les transformations **Sh** et **MB** se traduisent par les 6×16 inégalités linéaires entre les x_i suivantes :

$$\sum_{x \in \mathcal{P}} x + \sum_{x' \in \mathcal{P}'} x' \geq 9t \text{ et } t \geq v, \forall v \in \mathcal{P} \cup \mathcal{P}'.$$

Les ensembles \mathcal{P} , respectivement \mathcal{P}' , sont des sous-ensembles de 9 variables qui engendrent le registre du r^{eme} tour, respectivement le registre du $r + 1^{\text{eme}}$ tour et les variables t indiquent si une colonne est active ou non. On ajoute par ailleurs 6 inégalités, qui traduisent que la somme

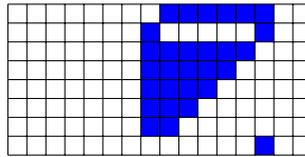
des 16 variables t définies au tour r est au plus 7. On réalise alors la minimisation de la somme des 6×16 variables t .

Un programme de résolution de tels systèmes linéaires sous contraintes, dans notre cas Gurobi, permet de trouver de nombreux tels chemins après seulement quelques minutes de calculs. La plupart sont trivialement équivalents : tout décalage d'un nombre fixé de colonnes du chemin différentiel tronqué engendre un chemin différentiel tronqué équivalent. Deux chemins itératifs retiennent finalement l'attention, celui présenté en Figure 8.1 et un second chemin itératif défini comme suit :



Ce chemin différentiel tronqué itératif alternatif ne paraît pas être meilleur pour monter l'attaque. Dans les deux cas, 5 colonnes n'atteignent pas la borne de Singleton. On peut alors trivialement en dériver des chemins plus creux. Ceux-ci ne permettent pas de réaliser de gains significatifs en complexité ou en nombre de tours.

Exemple 21. *Un exemple de porosité supérieure est donné ici Figure 8.1, strictement inclus dans le chemin différentiel tronqué précédent :*



5'

Phase de rebond : Les techniques de programmation linéaire sous contraintes permettent également la recherche des chemins différentiels tronqués les plus probables pour la phase de rebond. Pour ce faire, on définit les 128 variables x_0, \dots, x_{127} correspondant au motif différentiel issu de la phase de collecte. On ajoute après trois tours de propagation la contrainte que le registre de sortie possède plus de 16 octets non actifs et on conserve dans de nouvelles variables c le nombre d'octets nuls x_i en sortie pour chaque colonne active pour la transformation **MB**. On minimise alors la somme des variables c sous la condition que les octets d'un registre ne puissent tous être actifs.

Le meilleur chemin obtenu par cette méthode est celui de la Figure 8.1. Le registre de sortie contient alors 24 octets non actifs et la probabilité qu'un tel chemin soit réalisé vaut β^{-25} : β^{-22} pour le premier tour et β^{-3} pour le second. Un autre chemin peut être déterminé par cette méthode, avec une probabilité de réalisation légèrement plus faible, β^{-26} , et 17 octets non actifs dans le registre de sortie. Elle figure ci-dessous :



8.1.2 Motif différentiel

Bien que le chemin différentiel tronqué présenté soit spécifique à Grøstl₅₁₂, il semble que les techniques développées ici, notamment pour recoller trois ensembles de valeurs de différentielles, peuvent être appliquées à d'autres structures de type AES. La Figure 8.1 spécifie le chemin différentiel tronqué au cœur de notre attaque. Les carrés bleus correspondent toujours aux positions d'octets actifs, pour lesquels la différence peut être non nulle, les carrés blanc correspondant aux positions d'octets *non actifs*.

8.1.3 Plausibilité de réalisation

Le chemin différentiel tronqué sur 11 tours de la Figure 8.1, au delà de sa représentation classique, laisse transparaître son caractère très fortement structuré dans son profil d'octets actifs :

$$104 \xrightarrow{R_1} 53 \xrightarrow{R_2} 34 \xrightarrow{R_3} 34 \xrightarrow{R_4} 34 \xrightarrow{R_5} 34 \xrightarrow{R_6} 34 \xrightarrow{R_7} 34 \xrightarrow{R_8} 34 \xrightarrow{R_9} 53 \xrightarrow{R_{10}} 104 \xrightarrow{R_{11}} 128.$$

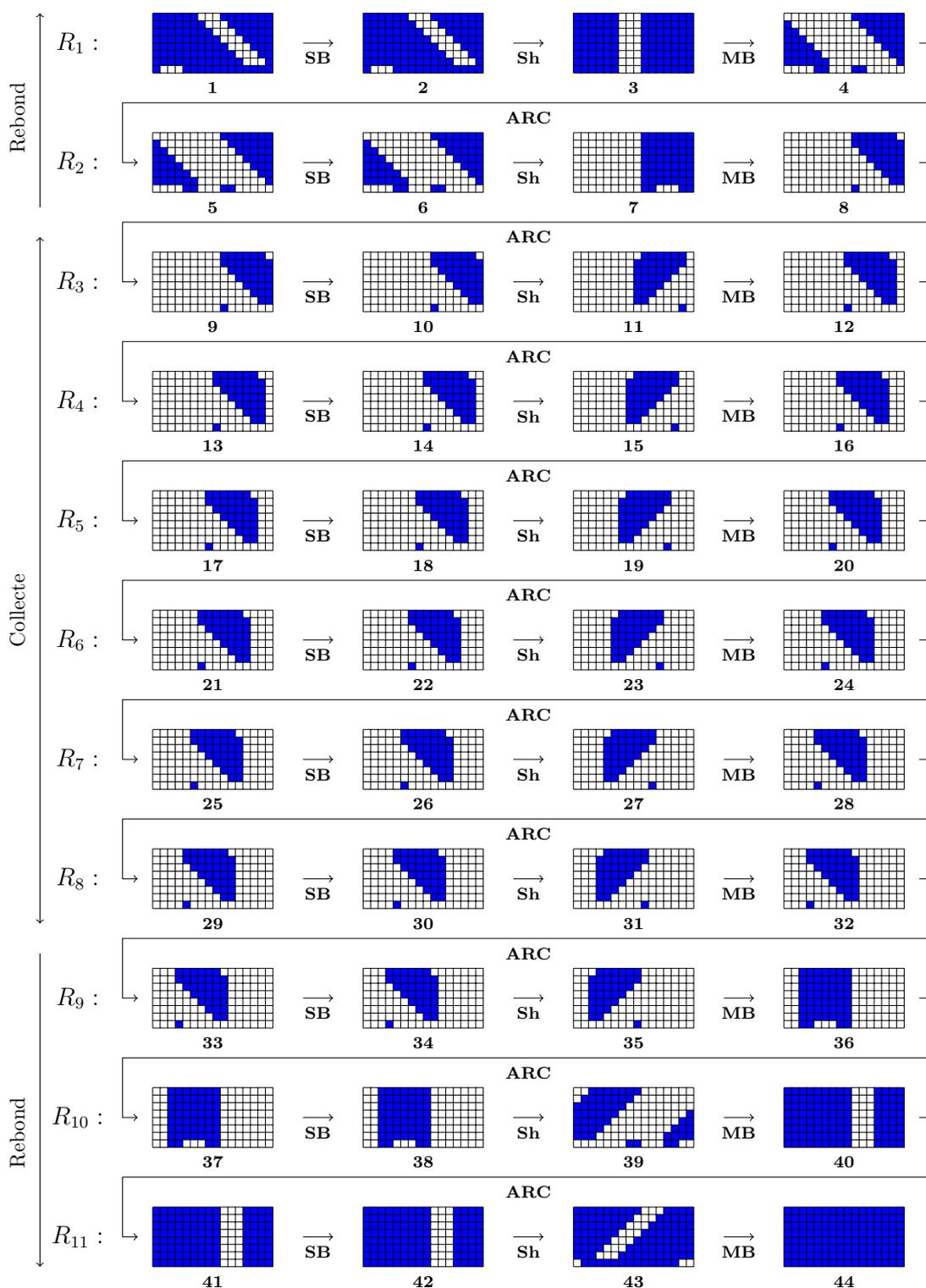
L'analyse de plausibilité associée à ce chemin différentiel tronqué en montre la pertinence. On sait qu'il existe $\beta^{128+104} = \beta^{232}$ couples ordonnés de valeurs d'état interne dont la différence est compatible avec le motif 1. À partir de l'équation (7.1) appliquée indépendamment colonne par colonne, on estime la probabilité pour un couple de valeurs d'état interne de présenter des différences intermédiaires compatibles avec l'intégralité du chemin différentiel tronqué :

- β^{-51} pour la propagation à travers la transformation **MB** du premier tour,
- β^{-22} pour la propagation à travers les transformations **MB** de chacun des tours 2 à 8
- β^{-3} pour la propagation à travers la transformation **MB** du tour 9.

Les propagations des tours 10 et 11 sont déterministes. L'hypothèse d'uniformité (6.2) nous permet alors de conclure que $\beta^{128+104} \cdot \beta^{-51+7 \cdot (-22)-3} = \beta^{232} \cdot \beta^{-208} = \beta^{24}$ couples de valeurs d'état interne en moyenne présentent des différences intermédiaires compatibles avec ce chemin différentiel tronqué.

La même analyse réalisée à partir des tours centraux laisse davantage filtrer l'essence de l'attaque rebond que le chemin différentiel tronqué de la Figure 8.1 induit. Commenant l'analyse du tour 6, on compte $\beta^{34+128} = \beta^{162}$ couples de valeurs d'état interne intermédiaire compatibles avec le motif différentiel 21. Les transitions probabilistes engendrées par les transformations **MB** des tours 6 à 11 et par les transformations **MB**⁻¹ des tours 6 à 2 sont alors réalisées respectivement avec probabilité $\beta^{-22} \cdot \beta^{-22} \cdot \beta^{-22} \cdot \beta^{-3} = \beta^{-69}$ et on retrouve les $\beta^{162} \cdot \beta^{-2 \cdot 69} = \beta^{24}$ couples de valeurs d'état interne dont les différences intermédiaires sont compatible avec l'intégralité du chemin différentiel tronqué sur 11 tours.

La phase de collecte implique ici les tours 3 à 8 et consiste à recenser des couples de valeurs d'état interne compatibles avec le chemin différentiel tronqué du motif 8 au motif 32. La phase de rebond consiste à trouver parmi ces exemplaires un couple satisfaisant les deux transitions non déterministes restantes, engendrées par la transformation **MB** du neuvième tour et par la transformation **MB**⁻¹ du deuxième tour. Chacune étant réalisée avec probabilité β^{-3} , elles sont réalisées conjointement avec probabilité β^{-6} .

FIGURE 8.1 – Chemin différentiel tronqué sur 11 tours de P_{1024} de Grøstl₅₁₂.

8.1.4 Instance du paradoxe des anniversaires généralisé induite

Le chemin différentiel sur 11 tours de la Figure 8.1 induit une instance du problème des anniversaires classique. La permutation P est ici une version à 11 tours de la permutation interne P_{1024} de Grøstl_{512} . L'espace linéaire E_1 correspond ici aux différences compatibles avec le motif différentiel tronqué 1 de la Figure 8.1 reproduit ci-dessous.

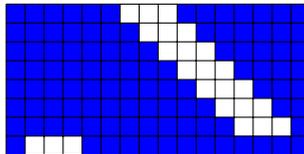


FIGURE 8.2 – Espace linéaire E_1

104 octets de différences sont possibles, ce qui correspond ainsi à un espace linéaire de dimension :

$$n_1 = \dim(E_1) = 104 \cdot 8$$

L'espace linéaire E_2 correspond ici aux différences compatibles avec l'image par la transformation linéaire inversible \mathbf{MB} du motif différentiel tronqué 43 de la Figure 8.1 reproduit ci-dessous.

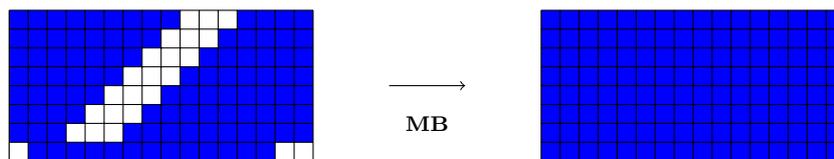


FIGURE 8.3 – Espace linéaire E_2

104 octets de différence sont possibles, ce qui correspond ainsi à un espace linéaire de dimension :

$$n_2 = \dim(E_2) = 104 \cdot 8$$

On se retrouve donc avec une instance du problème des anniversaires dont l'algorithme de résolution générique présente une complexité calculatoire de :

$$\log_2(\mathcal{C}_{gen}) = \frac{128 - 104}{2} \cdot 8 = 12 \cdot 8 = 96$$

Remarque 28. *L'approche issue de techniques de programmation linéaire sous contrainte est l'origine de chemins différentiels tronqués hautement structurés pour Q_{1024} qui partagent avec P_{1024} la même description macroscopique. Cette remarque n'est pas une surprise puisque la seule modification qui les sépare est la définition de la transformation \mathbf{Sh} , et que celle-ci demeure similaire. Le profil d'octets actifs est le même pour ces deux permutations et la Figure 8.4 donne le détail du chemin différentiel tronqué jumeau défini sur Q_{1024} .*

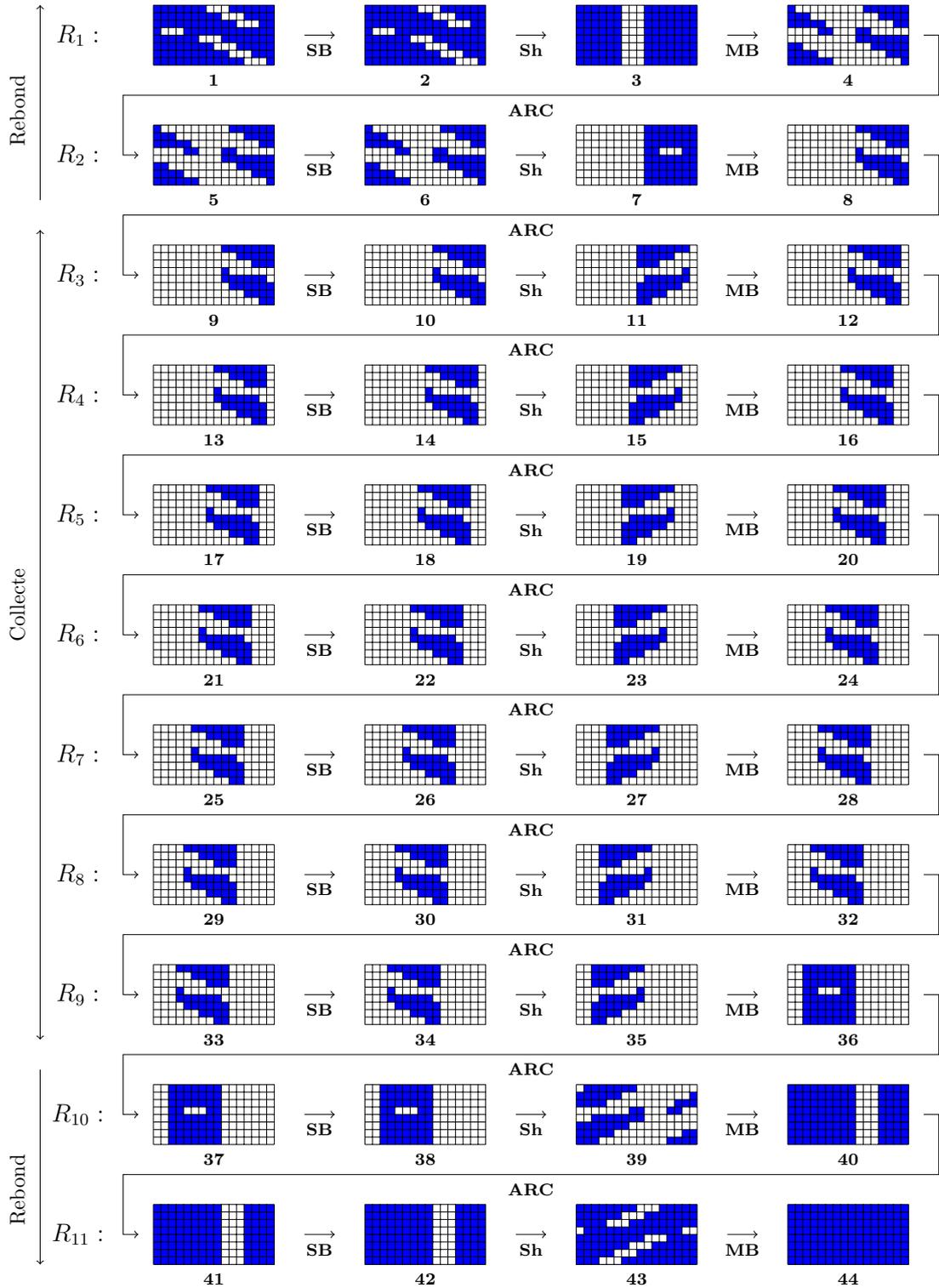


FIGURE 8.4 – Chemin différentiel tronqué sur 11 tours de Q_{1024} de $Gr\ddot{o}stl_{512}$.

8.1.5 Description macroscopique compacte

Les chemins différentiels tronqués ne sont pas affectés par les transformations **ARC**. On peut donc sans inconvénient les passer sous silence. De plus, les transformations **Sh** déplacent des valeurs de différences de positions d'octets et les transformations **SB** s'appliquent indépendamment sur les octets. Ces deux opérations commutent donc et il est indolore de permuter leurs applications. Dans la lignée de Daemen et Rijmen [DR06], on définit alors les deux transformations macroscopiques induites :

- Une transformation non linéaire **SuperSBox (SSB)** qui s'appliquent de manière indépendante sur les colonnes de l'état interne :

$$\mathbf{SuperSBox} := \mathbf{SubBytes} \circ \mathbf{MixBytes} \circ \mathbf{SubBytes}.$$

On écrira indifféremment **SuperSBox (SSB)** pour faire référence à la transformation globale ou à sa restriction sur une colonne ou un ensemble de colonnes.

- Une transformation linéaire **SuperLinear (SL)** :

$$\mathbf{SuperLinear} := \mathbf{ShiftBytesWide} \circ \mathbf{MixBytes} \circ \mathbf{ShiftBytesWide}.$$

Le chemin différentiel tronqué sur 11 tours donné en Figure 8.1 admet une écriture plus compacte grâce à ces deux opérations et résulte en la Figure 8.5.

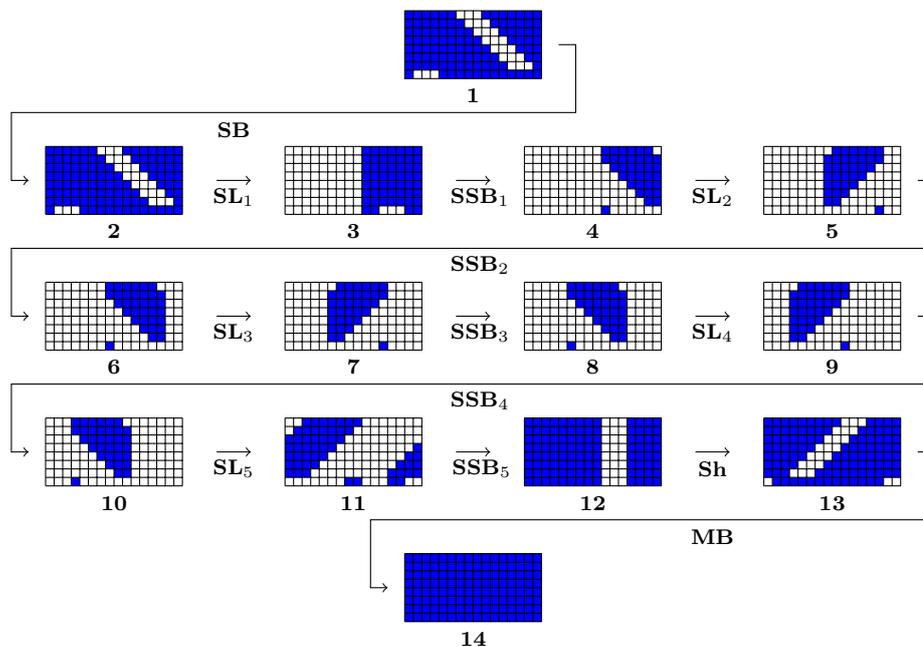


FIGURE 8.5 – Chemin différentiel tronqué sur 11 tours de P_{1024} en Super-Boîtes S.

Remarque 29. Cette réécriture n'est bien évidemment pas l'apanage du seul chemin différentiel tronqué sur P_{1024} . Son homologue sur Q_{1024} se réécrit donc quant à lui comme donné sur la Figure 8.6.

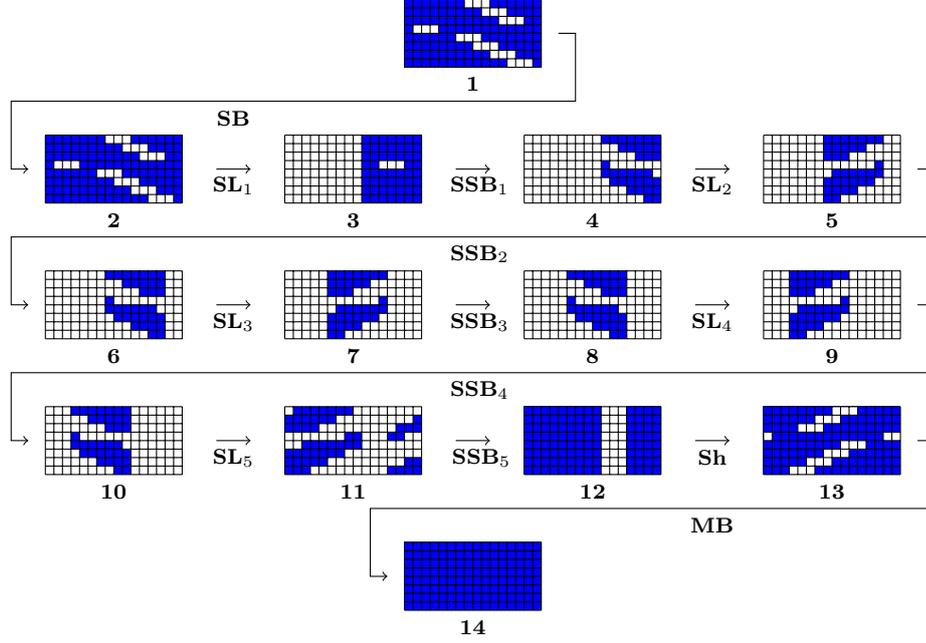


FIGURE 8.6 – Chemin différentiel tronqué sur 11 tours de Q_{1024} en Super-Boîtes S.

8.2 Algorithme de différenciation

Avant de pénétrer les entrailles de l'algorithme de différenciation, il est nécessaire d'introduire plusieurs définitions supplémentaires. On appellera complétion d'une valeur d'état interne partiel V , typiquement une valeur de colonne, une valeur d'état interne complet dont la restriction à l'ensemble de positions d'octets incluses dans V est V . Par extension, on appellera complétion d'un couple de valeurs d'état interne partiel défini sur le même ensemble de positions d'octet (V_1, V_2) un couple de complétions des V_i dont la différence dans l'ensemble des positions d'octets restantes est nulle. Pour la suite, toutes les références à des motifs sont liées à la Figure 8.5.

Pour donner une idée claire de l'enchaînement des étapes algorithmiques, on commence par en donner une esquisse. Les explications détaillées suivront cette présentation sommaire mis à part ce qui concerne la phase de rebond, triviale.

8.2.1 Esquisse de l'algorithme de différenciation

Étape 1. On construit une base de Δ_1 , le \mathcal{B} -sous-espace vectoriel de dimension 12 des éléments δ_1 dans \mathcal{S} dont les positions d'octets actifs sont incluses dans le motif 4 et dont les images par \mathbf{SL}_2 , δ'_1 , possèdent leurs positions d'octets actifs incluses dans le motif 5 :

$$\Delta_1 = \{\delta_1 \in P_4 \mid \delta'_1 = \mathbf{SL}_2(\delta_1) \in P_5\}. \quad (8.1)$$

On construit de la même façon une base de $\Delta_2 = \{\delta_2 \in P_6 \mid \delta'_2 = \mathbf{SL}_3(\delta_2) \in P_7\}$ et une base de $\Delta_3 = \{\delta_3 \in P_8 \mid \delta'_3 = \mathbf{SL}_4(\delta_3) \in P_9\}$. L'utilisation de ces bases permet d'énumérer les éléments dans ces sous-espaces vectoriels avec des complexités calculatoires et en mémoire négligeables, de l'ordre de $O(1)$.

Étape 2. On choisit un élément arbitraire δ_2 dans Δ_2 . Colonne par colonne, pour chaque colonne dont l'indice appartient à I_6 , on stocke dans des listes $(C_i)_{i \in I_6}$ tous les couples de valeurs de colonnes dont les différences sont compatibles avec δ_2 et dont les images par \mathbf{SSB}_2^{-1} possèdent des différences compatibles avec le motif 5. Par symétrie, pour chaque colonne dont l'indice appartient à I_7 , on stocke dans des listes $(C'_j)_{j \in I_7}$ tous les couples de valeurs de colonnes dont les différences sont compatibles avec δ'_2 et dont les images par \mathbf{SSB}_3 possèdent des différences compatibles avec le motif 8. On calcule donc pour tout i dans I_6 , respectivement pour tout j dans I_7 , les listes C_i , respectivement C'_j , définies par :

$$\begin{aligned} C_i &= \{ (X, Y) \in \mathcal{C}^2 \mid X \oplus Y = (\delta_2)_{|i}, \quad \mathbf{SSB}_2^{-1}(X) \oplus \mathbf{SSB}_2^{-1}(Y) \in (P_5)_{|i} \}, \\ C'_j &= \{ (X, Y) \in \mathcal{C}^2 \mid X \oplus Y = (\delta'_2)_{|j}, \quad \mathbf{SSB}_3(X) \oplus \mathbf{SSB}_3(Y) \in (P_8)_{|j} \}. \end{aligned}$$

Cette étape exige une complexité calculatoire et un coût mémoire de l'ordre de $O(\beta^7)$.

Étape 3. À partir de $O(\beta^6)$ éléments δ_1 dans Δ_1 , on calcule et stocke β^6 couples de valeurs d'état interne partiel de 7 colonnes dans une liste E , combinaisons d'éléments dans les listes $(C_i)_{i \in I_6}$, telles que toute complétion induite par une telle paire de valeurs d'état interne partiel de 7 colonnes possède une différence égale à δ_2 et dont la différence des images par $\mathbf{SL}_2^{-1} \circ \mathbf{SSB}_2^{-1}$ appartient à Δ_1 . À partir d'une énumération similaire d'éléments dans Δ_3 , dans une liste F , on stocke β^6 couples de valeurs d'état interne partiel de 7 colonnes combinaisons d'éléments dans les listes $(C'_j)_{j \in I_7}$ telles que toute complétion induite par une telle paire de valeurs d'état interne partiel de 7 colonnes possède une différence égale à δ'_2 et dont la différence des images par \mathbf{SSB}_3 appartient à Δ_3 . Cette étape est réalisable avec $O(\beta^6)$ calculs et une coût mémoire équivalent.

Étape 4. On détermine $(e, e \oplus (\delta_2)_{|I_6})$ dans E et $(f, f \oplus (\delta'_2)_{|I_7})$ dans F tel que $(f, f \oplus (\delta'_2)_{|I_7})$ admet des complétions dont les images par \mathbf{SL}_3^{-1} ne rentrent pas en conflit avec $(e, e \oplus (\delta_2)_{|I_6})$. Un choix arbitraire dans $E \times F$ engendre de telles complétions seulement avec probabilité β^{-12} . Cependant, β^{28} complétions sont disponibles chaque fois que c'est le cas. Une telle paire peut être déterminée pour une complexité calculatoire d'environ $O(\beta^7)$ et un coût mémoire de l'ordre de $O(\beta^6)$.

Étape 5. On détermine un couple de valeurs d'état interne partiel de 34 octets $(s, s \oplus \delta'_3)^1$ dont les positions sont conformes au motif 9 et dont la différence est compatible avec la valeur de différence δ'_3 induite par le choix de $(f, f \oplus (\delta'_2)_{|I_7})$. Ce couple est construit d'une telle façon que l'image de toute complétion de ce couple de valeurs d'état interne partiel de 34 octets par \mathbf{SSB}_4 possède une différence appartenant à P_{10} . Il existe alors β^{72} complétions de $(f, f \oplus (\delta'_2)_{|I_7})$ dont les images par $\mathbf{SL}_4 \circ \mathbf{SSB}_3$ ne contredisent pas ces 34 valeurs d'octets. Ce couple de valeurs d'état interne partiel de 34 octets peut être déterminé avec une complexité calculatoire d'environ $O(\beta^3)$.

Étape 6. Parmi les β^{28} complétions déterminées par l'étape 4 et les β^{72} complétions déterminées par l'étape 5, on calcule une intersection de β^6 complétions. Propagées, tous ces β^6 couples de valeurs d'état interne présentent des différences intermédiaires compatibles avec le chemin

1. Un léger abus de notation nous fait écrire ici δ'_3 indifféremment pour faire référence à la valeur différentielle sur l'état interne entier ou sa restriction aux octets actifs.

différentiel tronqué entre les motifs 4 et 10. La détermination de cette intersection nécessite une complexité calculatoire d'environ $O(\beta^9)$ et présente un coût mémoire de l'ordre de $O(\beta^7)$. Ceci conclut la phase de collecte.

Étape 7. L'énumération des β^6 couples de valeurs d'état interne collectées pendant la phase de collecte permet de déterminer un couple parmi ceux-ci satisfaisant simultanément les deux transitions probabilistes indépendantes restantes : à travers la transformation **MB** entre les motifs 10 et 11 et à travers la transformation **MB**⁻¹ entre les motifs 4 et 3. Cette phase de rebond est alors réalisée avec $O(\beta^6)$ calculs.

Pour résumer, cet algorithme construit un couple de valeurs d'état interne tel que, lorsque propagés à travers 11 tours de P_{1024} , les différences successives sont compatibles avec le chemin différentiel tronqué de la Figure 8.5. La complexité calculatoire de cet algorithme est d'environ $O(\beta^9) \simeq 2^{72} < 2^{80}$ et son coût mémoire est de l'ordre de $O(\beta^7) \simeq 2^{56}$. L'attaque générique sur l'instance du problème des anniversaires induite possède une complexité calculatoire d'environ β^{12} . Il s'agit donc bien d'un algorithme de différenciation.

8.2.2 Étape 1 : Construction des bases des sous-espaces linéaires Δ_i

On explique ici comment on peut construire une base de Δ_1 , le \mathcal{B} -sous-espace vectoriel défini par l'équation (8.1). Le détail de la transformation **SL**₂ engendre la Figure 8.7.

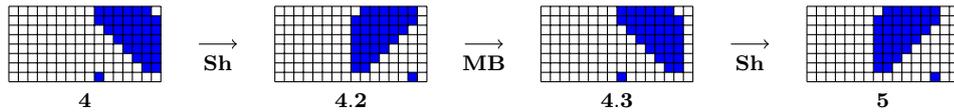


FIGURE 8.7 – Chemin différentiel vérifié par les éléments de l'ensemble Δ_1 .

Des 34 cellules bleues du motif 4, on apprend la dimension de P_4 , \mathcal{B} -sous-espace linéaire de dimension 34. Comme la transformation **MB** agit indépendamment sur les colonnes, on peut commencer arbitrairement par considérer la transition de la neuvième colonne du motif 4.2 à la neuvième colonne du motif 4.3 à travers la transformation **MB**, représentée en Figure 8.8.

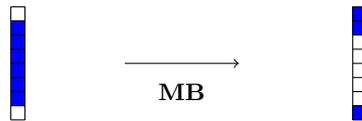


FIGURE 8.8 – Transition à travers la transformation **MixBytes**.

Comme la transformation **MB** est réalisée par une matrice MDS, on en déduit que le sous-espace suivant est de dimension 1 :

$$\{ X \in \mathcal{C} \mid X_1 = X_8 = 0 \text{ et } (\mathbf{MB}(X))_3 = \dots = (\mathbf{MB}(X))_7 = 0 \}.$$

Une élimination gaussienne permet de déterminer une base, $\{b_9\}$, de ce sous-espace. Cette procédure appliquée à la 15^{ème} colonne engendre une base, $\{b_{15}\}$, du sous-espace vectoriel

de dimension 1 induit. En l'appliquant sur les autres colonnes, on construit alors successivement $\{b_{10}, b'_{10}\}, \dots, \{b_{14}, b'_{14}\}$, des bases des sous-espaces vectoriels de dimension 2 correspondant respectivement aux colonnes 10 à 14. On construit des valeurs d'état interne $(Y_i)_{i \in I_4}$ et $(Y'_i)_{i \in \{10, \dots, 14\}}$, complétions de $(b_i)_{i \in I_4}$ et $(b'_i)_{i \in \{10, \dots, 14\}}$ avec des octets nuls dans les positions d'octet restantes. Une base de Δ_1 est alors donnée par :

$$\{ \mathbf{Sh}^{-1}(Y_9), \mathbf{Sh}^{-1}(Y_{15}), \mathbf{Sh}^{-1}(Y_{10}), \mathbf{Sh}^{-1}(Y'_{10}), \dots, \mathbf{Sh}^{-1}(Y_{14}), \mathbf{Sh}^{-1}(Y'_{14}) \} .$$

Des bases pour les sous-espaces vectoriels Δ_2 et Δ_3 sont construites par des procédures similaires.

8.2.3 Étape 2 : Construction de couples de valeurs de colonnes avec différences amont/aval d'une Super-Boite S maîtrisées

Soit δ_2 un élément arbitraire dans Δ_2 . L'objectif de cette étape algorithmique est de calculer et stocker dans des listes C_i , pour tout $i \in I_6$, et dans des listes C'_j , pour tout $j \in I_7$, les couples de valeurs de colonnes dont la différence égale les restrictions respectives $(\delta_2)_{|i}$ et $(\delta'_2)_{|j}$ et dont les images respectives par \mathbf{SSB}_2^{-1} et \mathbf{SSB}_3 sont compatibles des restrictions des motifs tronqués $(P_5)_i$ et $(P_8)_j$. Une recherche exhaustive naïve réalise cette construction pour une complexité calculatoire d'environ $14 \cdot \beta^8$. Dans l'esprit des travaux de Sasaki, Li, Wang, Sakiyama et Ohta [Sas+10], il existe une procédure plus efficace, explicitée par ce qui suit.

Cette construction peut se faire colonne par colonne. On se focalise arbitrairement sur C_8 , le traitement des autres colonnes s'en déduit immédiatement. On souhaite stocker dans la liste C_8 les éléments de \mathcal{C}^2 dont les différences égalent $(\delta_2)_{|8}$ et dont les images par \mathbf{SSB}_2^{-1} ont une différence appartenant à $(P_5)_{|8}$. La Figure 8.9 introduit à cet effet deux motifs différentiels tronqués intermédiaires induits par la décomposition de la transformation \mathbf{SSB}_2 .

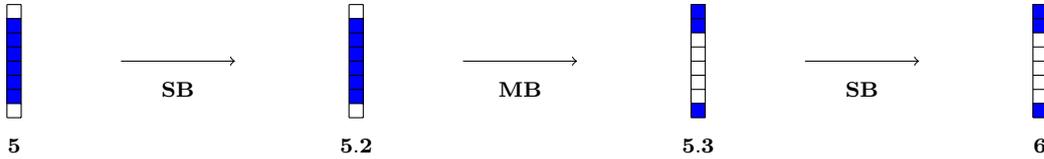


FIGURE 8.9 – Transition différentielle tronquée à travers une opération Super-Boite S.

On introduit ici D_1 , l'ensemble des valeurs de différences compatibles avec le motif 5.3 de la Figure 8.9 dont les images par la transformation \mathbf{MB}^{-1} sont compatibles avec le motif 5.2 :

$$D_1 = \{ \delta \in (P_6)_{|8} \mid \mathbf{MB}^{-1}(\delta) \in (P_5)_{|8} \} .$$

De l'équation (7.1), on déduit $|D_1| = \beta$. On introduit alors D_2 l'ensemble :

$$D_2 = \{ \delta \in \mathcal{C} \mid \exists X \in \mathcal{C} \text{ s.t. } \mathbf{SB}^{-1}(X \oplus (\delta_2)_{|8}) \oplus \mathbf{SB}^{-1}(X) = \delta \} .$$

De l'équation (6.1), on déduit $|D_2| = \beta^3 \cdot 2^{-3}$. Alors :

$$\delta \in D_2 \Rightarrow |\{ X \in \mathcal{C} \mid \mathbf{SB}^{-1}(X \oplus (\delta_2)_{|8}) \oplus \mathbf{SB}^{-1}(X) = \delta \}| = 2^3 \cdot \beta^5 .$$

L'hypothèse (6.2) assure que $|D_1 \cap D_2| = 2^{-3} \cdot \beta$. La liste C_8 est donc de cardinal β^6 :

$$C_8 = \bigcup_{\delta \in D_1 \cap D_2} \{(X, X + (\delta_2)_{|8}) \in \mathcal{C}^2 \mid \mathbf{SB}^{-1}(X \oplus (\delta_2)_{|8}) \oplus \mathbf{SB}^{-1}(X) = \delta\}.$$

On considère alors, pour construire C_8 , tous les couples $(X, X \oplus (\delta_2)_{|8}) \in \mathcal{C}^2$ tels que la restriction de X aux positions d'octets non actifs est nulle. On stocke dans une liste intermédiaire H les couples tels que la restriction de $\mathbf{MB}^{-1}(\mathbf{SB}^{-1}(X \oplus (\delta_2)_{|8}) \oplus \mathbf{SB}^{-1}(X))$ aux première et huitième position d'octets est nulle. On stocke alors dans C_8 , ordonnés selon la valeur $\delta(X) = \mathbf{SSB}_2^{-1}(X) \oplus \mathbf{SSB}_2^{-1}(X \oplus (\delta_2)_{|8})$, tous les éléments dans \mathcal{C}^2 tels que leur restriction aux première, deuxième et huitième positions d'octet égalent la restriction en ces mêmes positions d'un élément dans H et tels qu'en les positions d'octets restantes, leur différence soit nulle.

On écrira à partir de maintenant C_8 pour faire référence à la liste ainsi construite ou à la liste obtenue par le calcul des images par la transformation \mathbf{SSB}_2^{-1} .

Les autres listes sont construites avec la même procédure. La complexité calculatoire nécessaire à leur construction correspond alors à la complexité en mémoire nécessaire pour les stocker et n'est autre que le cardinal des listes :

- $C_8, C_9, C_{10}, C_{11}, C_{12}, C_{13}, C_{14}$ sont de tailles respectives $\beta^6, \beta^7, \beta^6, \beta^5, \beta^4, \beta^3, \beta^3$.
- $C'_6, C'_7, C'_8, C'_9, C'_{10}, C'_{11}, C'_{12}$ sont de tailles respectives $\beta^3, \beta^3, \beta^4, \beta^5, \beta^6, \beta^7, \beta^6$.

Remarque 30. *Le choix du chemin creux présenté dans l'exemple 21 entraîne une diminution de la complexité calculatoire et de la complexité en mémoire requises. En effet, on s'attend à ce que ce chemin soit réalisé par moins de couples de valeurs d'état interne. Le choix de ce motif réduit la dimension induite de Δ_1 de 12 à 7. C'est suffisant pour monter l'attaque et induit des listes C_i et C'_i de cardinal maximal β^6 . L'étape complète possède alors un coût $O(\beta^6)$ tant pour la complexité calculatoire que pour la complexité en mémoire.*

8.2.4 Étape 3 : Construction de couples de valeurs d'état interne partiel compatible avec des transitions $\Delta_1 \rightarrow \delta_2$ et $\delta_2 \rightarrow \Delta_3$

On choisit une valeur de différence arbitraire $\delta_1 \in \Delta_1$. On fait ici usage des bases construites à l'étape 1 des \mathcal{B} -sous-espaces vectoriels de dimension 12 Δ_i . Pour chaque telle valeur, on calcule son image δ'_1 par la transformation \mathbf{SL}_2 . Pour tout i dans I_6 , on considère tous les couples de valeurs de colonnes dans C_i calculés en étape 2 dont la différence des images desquelles par la transformation \mathbf{SSB}_2^{-1} égale $(\delta'_1)_{|i}$. Le coût calculatoire est alors simplement celui d'une recherche dans une liste triée. Chaque fois qu'un couple compatible est déterminé simultanément pour chacune de ces 7 colonnes, on stocke dans une liste E les couples de valeurs de 7 colonnes concaténations de tous les couples de valeurs de colonnes dans les listes C_i dont les différences égalent alors $(\delta_2)_{|I_6}$ et dont les images par la transformation \mathbf{SSB}_2^{-1} diffèrent en une valeur égale à $(\delta'_1)_{|I_6}$. On reproduit cette procédure jusqu'à obtenir β^6 tels couples de valeurs de 7 colonnes. L'énumération d'environ $O(\beta^6)$ éléments dans Δ_1 . permet donc de construire la liste souhaitée E avec β^6 éléments.

Une liste F avec β^6 couples de valeurs de 7 colonnes dont les différences égalent δ'_2 et dont les images par la transformation \mathbf{SSB}_3 présentent des différences égales à $(\delta_3)_{|I_7}$ pour un certain δ_3 dans Δ_3 se construit à partir des listes C'_j avec la même procédure.

Remarque 31. L'hypothèse (6.1) induit que la transformation \mathbf{SBB}_2^{-1} possède le comportement d'une boîte S idéale, appliquée indépendamment sur chaque colonne. Pour un δ'_1 arbitraire, chaque fois qu'il existe X valeur d'état interne partiel de 7 colonnes dans \mathcal{C}^7 tel que :

$$\mathbf{SBB}_2^{-1}(X \oplus (\delta_2)_{|I_6}) \oplus \mathbf{SBB}_2^{-1}(X) = (\delta'_1)_{|I_6},$$

ce qui est vrai avec probabilité 2^{-7} , on sait que 2^7 tels éléments X possèdent la même propriété.

Comme les listes C_8, \dots, C_{14} construites pendant l'étape 2 conservent tous les couples possibles de valeurs de colonnes dont les différences égalent respectivement $(\delta_2)_{|8}, \dots, (\delta_2)_{|14}$ et dont les images présentent des différences appartenant à P_5 , on détermine dans ces listes avec une probabilité de 2^{-7} pour un élément arbitraire δ_1 dans Δ_1 , 2^7 couples de valeurs d'état interne partiel défini sur les colonnes indicées par les éléments dans I_6 , dont les différences égalent $(\delta_2)_{|I_6}$ et dont les images par la transformation \mathbf{SBB}_2^{-1} présentent des différences égales à $(\delta'_1)_{|I_6}$.

8.2.5 Étape 4 : Premier recollement

On cherche ici à déterminer $(e, e \oplus (\delta_2)_{|I_6})$ dans E et $(f, f \oplus (\delta'_2)_{|I_7})$ dans F tels qu'il existe une complétion de $(f, f \oplus (\delta'_2)_{|I_7})$ dont l'image par la transformation \mathbf{SL}_3^{-1} ne rentre pas en conflit avec $(e, e \oplus (\delta_2)_{|I_6})$. On appelle une telle complétion une complétion concordante de $(f, f \oplus (\delta'_2)_{|I_7})$ et $(e, e \oplus (\delta_2)_{|I_6})$. Comme $\delta'_2 = \mathbf{SL}_3(\delta_2)$, on sait qu'une complétion concordante de f et e est également une complétion concordante de $f \oplus (\delta'_2)_{|I_7}$ et $e \oplus (\delta_2)_{|I_6}$. On montre alors que pour un choix arbitraire de $((e, e \oplus (\delta_2)_{|I_6}), (f, f \oplus (\delta'_2)_{|I_7}))$ dans $E \times F$, une complétion concordante existe avec probabilité β^{-12} . On explicite alors comment déterminer une telle paire avec une complexité calculatoire d'environ $O(\beta^7)$ et un espace mémoire de l'ordre de $O(\beta^6)$.

La Figure 8.10 introduit deux motifs intermédiaires qui apparaissent dans la décomposition de la transformation \mathbf{SL}_3 . Les cellules roses du motif 6.3 correspondent aux valeurs d'octets fixées par un choix de f alors que les cellules vertes du motif 6.2 correspondent aux valeurs d'octets fixées par un choix de e .

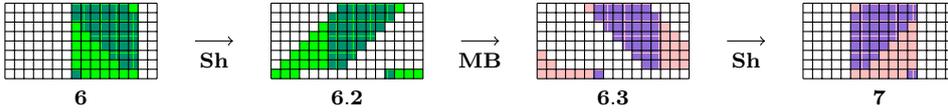


FIGURE 8.10 – Recollement, première édition.

La transformation \mathbf{MB} s'applique indépendamment sur les colonnes. Il est donc possible d'analyser les transitions locales de colonnes. Deux types de colonnes doivent alors être distingué :

- Pour les colonnes 7 à 13, les c contraintes induites par les cellules vertes sur le motif 6.2 sont trop nombreuses pour pouvoir être compensées par les d degrés de libertés induits par les cellules blanches sur le motif 6.3. Une complétion concordante sur ces colonnes est alors possible avec une probabilité de β^{d-c} .
- Pour les autres colonnes, les d degrés de libertés induits par les cellules blanches du motif 6.3 suffisent pour compenser les c contraintes induites par les cellules vertes du motif 6.2. Il existe alors β^{d-c} complétions concordantes locales pour chacune de ces colonnes.

On réalise ici l'analyse exacte sur la septième colonne, l'analyse pour les autres colonnes étant essentiellement la même. Notons $\mathbf{M} = \{m_{i,j}\}$ la matrice de la restriction de la transformation \mathbf{MB} sur une colonne. Sur la Figure 8.10, on se rend compte qu'il existe deux degrés de

liberté disponibles correspondant aux positions des octets 1 et 8 et que trois contraintes sont imposées par les valeurs fixées correspondant aux positions d'octets 1,2 et 8. On cherche alors à déterminer, étant données des valeurs d'octets x_2, \dots, x_7 et y_1, y_2, y_8 , s'il existe x_1 et x_8 satisfaisant l'équation (8.2).

$$\forall \ell \in \{1, 2, 8\}, \sum_{j=2}^7 m_{\ell,j} x_j + m_{\ell,1} x_1 + m_{\ell,8} x_8 = y_\ell. \quad (8.2)$$

Comme \mathbf{M} est MDS, son mineur $m_{1,1} \cdot m_{2,8} - m_{1,8} \cdot m_{2,1}$ est non nul. On peut alors écrire :

$$x_1 = l_{1,7}(x_2, \dots, x_7) + g_{1,7}(y_1, y_2) \quad \text{and} \quad x_8 = l_{8,7}(x_2, \dots, x_7) + g_{8,7}(y_1, y_2),$$

où $g_{1,7}, g_{8,7}, l_{1,7}$ et $l_{8,7}$ sont des formes linéaires dépendant uniquement de la transformation \mathbf{MB} et des positions des octets fixés. L'introduction des deux formes linéaires L_7 et R_7 obtenues par la substitution de x_1 et x_8 par leurs expressions, l'équation (8.2) est alors équivalente à :

$$\begin{cases} x_1 = l_{1,7}(x_2, \dots, x_7) + g_{1,7}(y_1, y_2), \\ x_8 = l_{8,7}(x_2, \dots, x_7) + g_{8,7}(y_1, y_2), \\ L_7(x_2, \dots, x_7) = R_7(y_1, y_2, y_8). \end{cases}$$

Une complétion concordante de e et f sur la septième colonne est alors possible si et seulement si $L_7(e_{2,7}, \dots, e_{7,7})$ égale $R_7(f_{2,1}, f_{2,2}, f_{2,8})$. Par souci de simplicité, on se passe de conserver dans l'expression les positions exactes des différents octets et on écrit plutôt $L_7(e)$ et $R_7(f)$.

On détermine deux formes linéaires L_{13} et R_{13} par la reproduction de la même analyse sur la treizième colonne. Pour $i \in \{8, \dots, 12\}$, on trouve des couples de formes linéaires L_i, L'_i et R_i, R'_i . On obtient en effet deux couples de formes linéaires puisque la différence entre le nombre de contraintes et le nombre de degrés de liberté n'est plus de 1 mais de 2. On reproduit ces arguments pour les colonnes restantes 1 à 6 et 14 à 16. Pour celles-ci, localement, il existe davantage de degrés de liberté que de contraintes. Chaque choix de $((e, e \oplus (\delta_2)_{|I_6}), (f, f \oplus (\delta'_2)_{|I_7}))$ dans $E \times F$ admet une complétion concordante locale. Finalement, on obtient une complétion concordante de f et e si et seulement si les douze équations $L_7(e) = R_7(f)$, $L_{13}(e) = R_{13}(f)$, $L_8(e) = R_8(f)$, $L'_8(e) = R'_8(f)$, \dots , $L_{12}(e) = R_{12}(f)$, $L'_{12}(e) = R'_{12}(f)$ sont satisfaites simultanément. Des valeurs uniformément distribuées vérifient simultanément ces équations linéaires avec probabilité β^{-12} .

On montre alors comment déterminer un couple $((e, e \oplus (\delta_2)_{|I_6}), (f, f \oplus (\delta'_2)_{|I_7}))$ dans $E \times F$ pour lequel il existe une complétion concordante de f et e avec une complexité calculatoire de l'ordre de $O(\beta^7)$: on trie la liste F selon l'ordre lexicographique défini par le calcul des formes linéaires : $(R_7(f), R_{13}(f), R_8(f), R'_8(f), \dots, R_{12}(f), R'_{12}(f))$ pour une complexité calculatoire de l'ordre de $O(\beta^7)$ et un espace mémoire d'environ $O(\beta^6)$. Pour chaque élément $(e, e \oplus (\delta_2)_{|I_6})$ de E , on calcule la valeur $(L_7(e), L_{13}(e), L_8(e), L'_8(e), \dots, L_{12}(e), L'_{12}(e))$ et on vérifie s'il appartient à la liste triée précédente. Ceci induit un coût de $O(\beta^6)$ calculs et de $O(\beta^6)$ recherches dans une liste triée. Comme β^{12} couples $((e, e \oplus (\delta_2)_{|I_6}), (f, f \oplus (\delta'_2)_{|I_7}))$ dans $E \times F$ sont disponibles et comme une complétion concordante de f et e existe avec probabilité β^{-12} , on obtient en moyenne une collision. S'il s'avère qu'un tel couple admettant une complétion concordante n'a pas été obtenu, on recommence alors à partir de l'étape 2 avec une nouvelle valeur de différence δ_2 . On peut alors supposer qu'une telle paire existe et qu'on la détermine avec une complexité calculatoire de l'ordre de $O(\beta^7)$ et un espace mémoire d'environ $O(\beta^6)$.

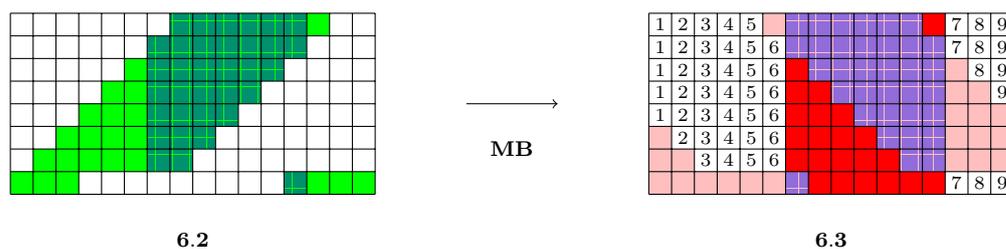


FIGURE 8.11 – Octets fixes et listes d'éléments.

Considérant fixé le couple $((e, e \oplus (\delta_2)_{|I_6}), (f, f \oplus (\delta'_2)_{|I_7}))$ dans $E \times F$, toutes les valeurs d'octets dans les colonnes 7 à 13 du motif 6.3 sont alors fixées par un tel choix, comme illustré par les cellules rouges dans la Figure 8.11).

On calcule alors toutes les complétions concordantes de e et f . On construit pour chacune des neuf colonnes restantes les listes des valeurs de colonnes compatibles avec les contraintes linéaires imposées par les octets fixés. La construction de ces listes ne diffèrent pas de ce qui a été fait avec l'utilisation de formes linéaires analogues aux $g_{i,j}$ et $f_{i,j}$. On détermine alors, comme l'illustre la Figure 8.11, des listes B_1, B_2, B_3, B_4 et B_9 , des listes de β^4 éléments dans les 1^{ère}, 2^{ème}, 3^{ème}, 4^{ème} et 16^{ème} colonnes, B_5 et B_8 , des listes de β^3 éléments dans les 5^{ème} et 15^{ème} colonnes et enfin B_6 et B_7 , des listes de β éléments dans les 6^{ème} et 14^{ème} colonnes. On obtient alors β^{28} complétions concordantes possibles comme produit direct des listes B_i . Le stockage de ce produit dans une unique liste est impossible car il nécessite des complexités outrageantes, tant en mémoire qu'en calculs.

8.2.6 Étape 5 : Deuxième recollement

Pour le moment, la valeur de différence δ_2 a été fixée. Une liste E a été construite de β^6 couples de valeurs d'état interne partiel sur 7 colonnes indicées par I_6 tels que leur différence égale $(\delta_2)_{|I_6}$ et tels que les différences des images de toute complétion par la transformation $\mathbf{SL}_2^{-1} \circ \mathbf{SSB}_2^{-1}$ appartient à Δ_1 . Symétriquement, une liste F a été construite de β^6 couples de valeurs d'état interne partiel sur 7 colonnes indicées par I_7 tels que leurs différences égalent $(\delta'_2)_{|I_7}$ et tels que les différences des images de toute complétion par la transformation \mathbf{SSB}_3 appartient à Δ_3 . Un élément $(e, e \oplus (\delta_2)_{|I_6})$ dans E et un élément $(f, f \oplus (\delta'_2)_{|I_7})$ dans F ont été déterminés de sorte qu'il existe β^{28} complétions concordantes de e et f dont les images par la transformation \mathbf{SL}_3^{-1} ne rentrent pas en conflit avec e . Les valeurs d'état interne partiel défini sur les colonnes induites par e et f sont donc désormais fixées. On appelle alors $\delta_1 \in \Delta_1$ et $\delta_3 \in \Delta_3$ les valeurs de différence induites par ces choix par propagation de $(e, e \oplus (\delta_2)_{|I_6})$ en amont par la transformation \mathbf{SSB}_2^{-1} et par propagation de $(f, f \oplus (\delta'_2)_{|I_7})$ en aval par la transformation \mathbf{SSB}_3 . On écrit alors $(f', f' \oplus (\delta_3)_{|I_7})$ l'image de $(f, f \oplus (\delta'_2)_{|I_7})$ par la transformation \mathbf{SSB}_3 .

Chacun des β^{28} couples de valeurs d'état interne tels que leur propagation en amont et en aval engendre des différences compatibles avec le chemin différentiel tronqué présenté en Figure 8.5 du motif 4 au motif 9. Pour satisfaire au chemin différentiel tronqué complet, il reste trois transitions probabilistes indépendantes, à savoir les transitions induites par la transformation \mathbf{SSB}_1^{-1} vérifiée avec probabilité β^{-3} , par la transformation \mathbf{SSB}_4 vérifiée avec probabilité β^{-22} et par la transformation \mathbf{SL}_5 vérifiée avec probabilité β^{-3} . L'énumération des β^{28} couples de valeurs d'état interne assure en moyenne la détermination parmi ceux-ci d'un couple satisfaisant

simultanément les trois transitions probabilistes indépendantes. Cependant, pour construire un algorithme de différenciation, notre algorithme doit déterminer un tel couple avec une complexité calculatoire inférieure à β^{12} .

Indépendamment des valeurs d'état interne partiel induit par le choix de $(e, e \oplus (\delta_2)_{|I_6})$ dans E , la focalisation sur le couple de valeur d'état interne partiel $(f', f' \oplus (\delta_3)_{|I_7})$ induit par le choix de $(f, f \oplus (\delta'_2)_{|I_7})$ dans F permet de montrer que fixer certains nouveaux couples de valeurs d'octets autorise d'éviter une transition probabiliste engendrée par la transformation \mathbf{SSB}_4 . On note δ'_3 l'image de δ_3 par la transformation \mathbf{SL}_4 : $\delta'_3 = \mathbf{SL}_4(\delta_3) \in P_9$. On recherche alors des couples de valeurs d'état interne partiel $(s, s \oplus \delta'_3)$ tels que $\mathbf{SSB}_4(s) \oplus \mathbf{SSB}_4(s \oplus \delta'_3) = \delta_4$ appartient à P_{10} . On ne s'intéresse pas à la valeur de δ_4 mais uniquement à son profil tronqué : savoir si $(\mathbf{MB} \circ \mathbf{SB})(s) \oplus (\mathbf{MB} \circ \mathbf{SB})(s \oplus \delta'_3)$ appartient P_{10} . Les 34 octets de s correspondant aux positions des octets actifs, illustrées par les cellules bleues du motif 9 de la Figure 8.5) suffit pour déterminer si δ_4 appartient à P_{10} .

Une étude colonne par colonne est alors réalisable, qui présente de fortes similitudes avec l'analyse réalisée pendant l'étape 2. En considérant à nouveau la Figure 8.9, qui dépeint le comportement de la colonne 4, on voit que la connaissance des valeurs de 6 octets de $s_{|4}$ est suffisant pour déterminer le comportement tronqué de leur différence. Comme les valeurs de différence sur cette colonne doivent s'annuler en 5 positions d'octets après une transformation, on calcule et on stocke dans une liste L_1 les β couples de valeurs de 6 octets dont les différences égalent la restriction de $(\delta'_3)_{|4}$ aux positions d'octets actifs et dont l'image de toute complétion présentant une différence nulle dans les positions d'octets restantes par la transformation \mathbf{SSB}_4 présente une différence appartenant à P_{10} .

On réalise les constructions similaires des six listes associées aux six colonnes restantes. On obtient ainsi :

- β couples de valeurs de 6 octets dans une liste L_1 (colonne 4)
- β^2 couples de valeurs de 7 octets dans une liste L_2 (colonne 5)
- β^2 couples de valeurs de 6 octets dans une liste L_3 (colonne 6)
- β^2 couples de valeurs de 5 octets dans une liste L_4 (colonne 7)
- β^2 couples de valeurs de 4 octets dans une liste L_5 (colonne 8),
- β^2 couples de valeurs de 3 octets dans une liste L_6 (colonne 9)
- β couples de valeurs de 3 octets dans une liste L_7 (colonne 10).

L'ensemble des combinaisons possibles de couples de valeurs d'octets dans les listes L_i fournit β^{12} couples de valeurs de 34 octets dont les différences sont compatibles avec δ'_3 et tels que leurs images par la transformation \mathbf{SSB}_4 de toute complétion avec des différences nulles dans les positions d'octets restantes appartiennent à P_{10} .

On détermine alors parmi ces possibilités, celles qui peuvent être recollées avec f . On se focalise délibérément sur les valeurs d'état interne fixées par le choix de f seul, sans les valeurs induites par le choix de e . Le recollement avec le choix de e sera considéré dans l'étape 6. La Figure 8.12 introduit deux motifs intermédiaires issus de la décomposition de la transformation \mathbf{SL}_4 . Sur le motif 9, les cellules jaunes indicées par un nombre i correspondent aux positions fixées par le choix d'un élément dans la liste L_i . Sur le motif 8, les cellules roses correspondent à l'image de f par la transformation \mathbf{SSB}_3 et de même pour la version décalée du motif 8.2.

L'analyse colonne par colonne de l'étape 4 peut être reproduite. Pour la colonne 5, les contraintes illustrées par les 3 cellules jaunes du motif 8.3 sont trop nombreuses pour être compensées par

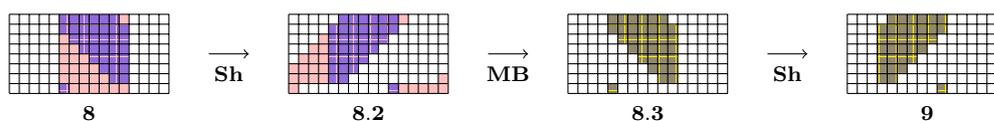


FIGURE 8.12 – Recollement, deuxième édition.

les 2 degrés de liberté illustrés par les cellules blanches du motif 8.2. Une complétion concordante locale est alors possible uniquement avec probabilité β^{-1} . Les colonnes 6 à 11 présentent un comportement similaire à la colonne 5 : une complétion concordante locale n'est également possible qu'avec une probabilité β^{-1} . L'hypothèse (6.2) assure que ces sept événements indépendants sont réalisés simultanément avec probabilité β^{-12} . On détermine donc en moyenne un élément parmi les β^{12} combinaisons possibles de couples dans les listes L_i tel qu'il existe une complétion concordante globale qui assure cette transition. On pourrait procéder à une énumération naïve gloutonne, cependant on donne ici une procédure inspiré de [JNP14] qui réalise la détermination de ce couple de valeurs de 34 octets avec environ $O(\beta^3)$ calculs.

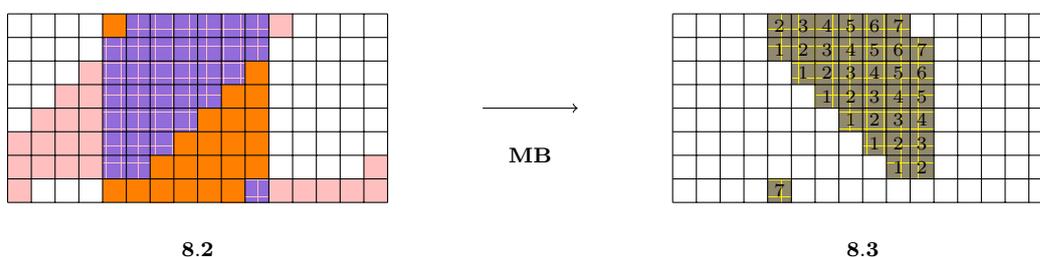


FIGURE 8.13 – Fusion des listes, première édition.

On se focalise sur la cinquième colonne du motif 8.3. Un triplet (l_1, l_2, l_7) dans $L_1 \times L_2 \times L_7$ admet une complétion concordante locale sur la cinquième colonne du motif 8.2 si et seulement si il satisfait une équation affine induite par $d = 5$ degrés de liberté pour $c = 6$ contraintes. À partir de l'énumération des éléments (l_1, l_2) dans $L_1 \times L_2$, on calcule la valeur induite pour l_7 illustrée par la cellule jaune dans le huitième octet de la cinquième colonne qui satisfait l'équation affine. L'hypothèse (6.2) assure l'existence d'un unique élément dans L_7 vérifiant cette propriété et on stocke le triplet (l_1, l_2, l_7) dans une liste L'_1 . Avec une complexité calculatoire d'environ $O(\beta^3)$, on collecte donc β^3 tels triplets.

On se focalise désormais sur la sixième colonne du motif 8.3. Un triplet (l_1, l_2, l_3) dans $L_1 \times L_2 \times L_3$ admet alors une complétion concordante locale sur la sixième colonne du motif 8.2 si et seulement si (l_1, l_2, l_3) satisfait deux équations affines induites par 5 degrés de liberté pour 7 contraintes. Une élimination gaussienne permet d'obtenir une équation affine en l_1 et l_2 mais pas en l_3 . On utilise alors cette équation pour filtrer L'_1 et construire une liste L''_1 avec β^2 éléments compatibles avec cette équation. À partir de l'énumération des éléments dans L''_1 , on calcule grâce à la deuxième équation la valeur induite pour l_3 illustrée par la cellule jaune du premier octet de la sixième colonne qui satisfait l'équation affine. L'hypothèse (6.2) assure l'existence de β éléments de L_3 pour chaque triplet (l_1, l_2, l_7) dans L''_1 et on stocke les quartets compatibles (l_1, l_2, l_3, l_7) dans une liste L'_2 . Avec une complexité calculatoire d'environ $O(\beta^3)$, on collecte donc β^3 tels quartets.

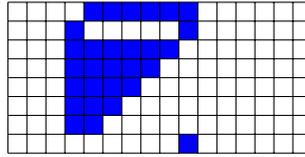
Cette procédure de filtrage suivie d'une extension reproduite pour les colonnes 7, 8 et 9 aboutit

à la construction d'une liste de 7-uplets L'_3 contenant β^3 éléments admettant des complétions concordantes locales pour les colonnes 5 à 9. Ceci peut être réalisé avec une complexité calculatoire d'environ $O(\beta^3)$.

Un élément dans L'_3 admet une complétion concordante locale sur la colonne 10 lorsqu'il satisfait deux équations affines induites par 1 degré de liberté pour 2 contraintes. On utilise ces équations pour filtrer L'_3 et obtenir une liste L'_4 de β éléments admettant des complétions concordantes pour la colonne 10. Un élément de L'_4 admet une complétion concordante locale pour la colonne 11 lorsqu'il satisfait une équation affine induite par 2 degrés de liberté pour 3 contraintes. On utilise ces équations pour filtrer L'_4 et obtenir un unique 7-uplet admettant des complétions concordantes locales sur les colonnes 5 à 11.

Une fois qu'une telle combinaison d'éléments dans les listes L_i admettant des complétions concordantes locales sur l'ensemble des colonnes 5 à 11 a été déterminée, ce qui nécessite $O(\beta^3)$ opérations, on sait qu'une unique complétion locale sur ces sept colonnes existe. Ceci signifie que les couples de valeurs correspondantes sur les 22 octets restant des colonnes 5 à 11 illustrés par les cellules oranges du motif 8.2 de la Figure 8.13 sont par là même fixés. Comme aucune contrainte ne limite les complétions possibles dans les autres colonnes, il existe donc β^{72} complétions de $(f, f \oplus (\delta'_2)_{|I_7})$ compatibles avec le choix $(s, s \oplus \delta_3)$.

Remarque 32. Comme l'exemple 21 l'énonce il est également possible de considérer un autre chemin différentiel tronqué, plus creux et strictement inclus dans le chemin de la Figure 8.1, obtenue par le remplacement du motif 9 par le motif suivant :



9'

Sélectionner ce motif réduit la dimension de Δ_3 de 12 à 7. Cela reste suffisamment faible pour monter l'attaque. Un tel motif implique l'existence de β^7 couples de valeurs de 29 octets compatibles avec le choix de $(f, f \oplus (\delta'_2)_{|I_7})$ avec une probabilité de β^{-7} . L'algorithme de fusion de listes peut être réalisé plus rapidement, avec $O(\beta^2)$ opérations.

8.2.7 Étape 6 : Troisième recollement

On dispose à ce stade de β^{28} complétions concordantes déterminées par l'étape 4 et de β^{72} complétions concordantes déterminées par l'étape 5. Celles-ci s'intersectent en β^6 complétions concordantes qui satisfont les 22 contraintes induites par la transformation **MB** qui les relie. La Figure 8.14 introduit deux motifs intermédiaires dans le but de clarifier comment se détermine cette intersection.

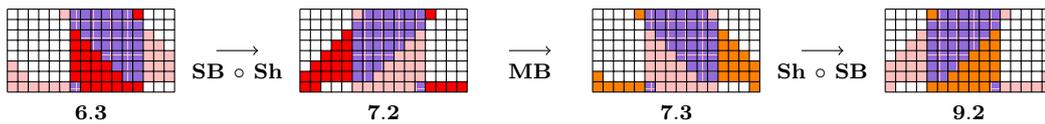


FIGURE 8.14 – Recollement, troisième édition.

Les techniques de fusion de listes de [JNP14] sont une nouvelle fois l'inspiration de la procédure suivante qui détermine avec une complexité calculatoire d'environ β^9 les β^6 couples de valeurs d'état interne attendus dont les différences intermédiaires sont compatibles avec le chemin différentiel tronqué entre les motifs 4 et 10 de la Figure 8.5.

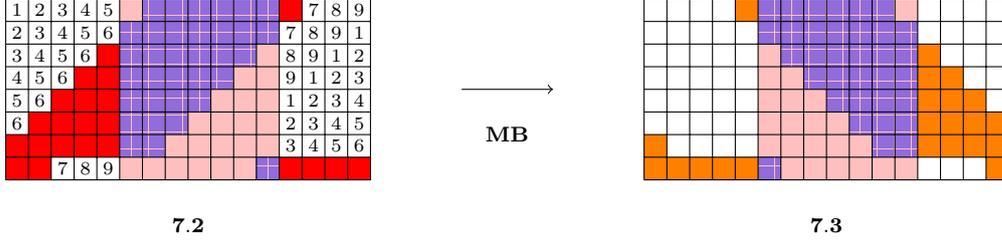


FIGURE 8.15 – Fusion de listes, deuxième édition.

Soit alors b_i un élément dans B_i , on écrit $b_{i,j}$ sa composante dans la $j^{\text{ème}}$ ligne. Par exemple, $b_{8,3}$ est la composante de b_8 dans la position d'octet référencée par l'intersection de la 13^{ème} colonne et de la 3^{ème} ligne du motif 7.2. En pré-calcul, on trie les listes B_1, B_2, B_3, B_5, B_8 et B_9 selon respectivement $(b_{1,2}, b_{1,3}, b_{1,4}, b_{1,5}), (b_{2,3}, b_{2,4}, b_{2,5}, b_{2,6}), (b_{3,7}), (b_{5,1}), (b_{8,3})$ et $(b_{9,4}, b_{9,8})$. Ceci est réalisable pour une complexité calculatoire d'environ $O(\beta^5)$:

$$\begin{aligned} B_1 &= \bigcup_{b_1 \in B_1} B'_1(b_{1,2}, b_{1,3}, b_{1,4}, b_{1,5}), & B_3 &= \bigcup_{b_3 \in B_3} B'_3(b_{3,7}), & B_8 &= \bigcup_{b_8 \in B_8} B'_8(b_{8,3}), \\ B_2 &= \bigcup_{b_2 \in B_2} B'_2(b_{2,3}, b_{2,4}, b_{2,5}, b_{2,6}), & B_5 &= \bigcup_{b_5 \in B_5} B'_5(b_{5,1}), & B_9 &= \bigcup_{b_9 \in B_9} B'_9(b_{9,4}, b_{9,8}). \end{aligned}$$

L'hypothèse (6.2) assure que $|B'_3| = \beta^3$, $|B'_5| = |B'_8| = |B'_9| = \beta^2$ et $|B'_1| = |B'_2| = 1$. Une élimination Gaussienne permet d'écrire les contraintes induites par les colonnes 3, 15 et 16 du motif 7.2, illustrées par les cellules rouges et les cellules oranges du motif 7.3, respectivement par les équations affines $l_3(b_3, b_4) = r_3(b_5, b_6, b_7)$, $l_{15}(b_3, b_4) = r_{15}(b_5, b_8, b_9)$ et $l_{16}(b_3, b_4) = r_{16}(b_5, b_6, b_9)$. où l_i et r_i sont des expressions affines.

Pour chacun des β éléments b_7 dans B_7 , on calcule les valeurs de $b_{8,3}, b_{9,4}, b_{1,5}, b_{2,6}$ et $b_{3,7}$ induites par les contraintes de la colonne 13. On calcule et on trie la liste $B'_3(b_{3,7}) \times B_4$ de β^7 éléments selon les valeurs de $(b_{3,6}, b_{4,1}, b_{4,7}, l_3(b_3, b_4), l_{15}(b_3, b_4), l_{16}(b_3, b_4))$. L'hypothèse (6.2) assure que B'_{34} possède β éléments :

$$B'_3(b_{3,7}) \times B_4 = \bigcup_{(b_3, b_4) \in B'_3(b_{3,7}) \times B_4} B'_{34}(b_{3,6}, b_{4,1}, b_{4,7}, l_3(b_3, b_4), l_{15}(b_3, b_4), l_{16}(b_3, b_4)).$$

Pour chacun des β^2 éléments b_8 dans $B'_8(b_{8,3})$ et chacun des β éléments b_6 dans B_6 , on calcule à partir de (b_6, b_7, b_8) les valeurs de $b_{5,1}$ et $b_{9,8}$ induites par les contraintes de la colonne 5.

Pour chacun des β^2 éléments b_5 dans $B'_5(b_{5,1})$ et chacun des β^2 éléments b_9 dans $B'_9(b_{9,4}, b_{9,8})$, on calcule les valeurs de $b_{3,6}$ et $b_{4,7}$ induites par les contraintes sur la colonne 14 et la valeur de $b_{4,1}$ induite par les contraintes sur la colonne 4.

Pour chacun des β éléments (b_3, b_4) dans $B'_{34}(b_{3,6}, b_{4,1}, b_{4,7}, l_3(b_3, b_4), l_{15}(b_3, b_4), l_{16}(b_3, b_4))$, on calcule les valeurs de $(b_{1,2}, b_{1,3}, b_{1,4}, b_{1,5})$ et $(b_{2,3}, b_{2,4}, b_{2,5}, b_{2,6})$ induites par les contraintes sur

les colonnes 13 à 16. On détermine les uniques valeurs attendues de b_1 dans $B'_1(b_{1,2}, b_{1,3}, b_{1,4}, b_{1,5})$ et b_2 dans $B'_2(b_{2,3}, b_{2,4}, b_{2,5}, b_{2,6})$.

On stocke dans une liste finale \mathcal{L} tous les 7-uplets satisfaisant simultanément les 3 équations indépendantes induites par les contraintes sur les colonnes 1 et 2. L'hypothèse (6.2) assure que ceci se produit avec probabilité β^{-3} . L'énumération des β^9 tels éléments fait émerger une liste \mathcal{L} de cardinal β^6 .

La complexité calculatoire globale est alors d'environ $O(\beta^9)$ et correspond au nombre d'éléments énumérés. La complexité en mémoire est de l'ordre de $O(\beta^7)$, correspondant au cardinal de la liste $B'_3 \times B_4$.

Conclusions et perspectives

Grâce à l'utilisation de nouvelles méthodes comme la programmation linéaire sous contrainte, grâce à une exploitation efficace de sous-espaces linéaires induits par la représentation macroscopique compacte, il est possible d'améliorer la cryptanalyse par attaque rebond pour les permutations internes de Grøstl_{512} . Cette variante tire en effet profit de la description macroscopique avec les transformations *SuperSbox* et *SuperLinear* pour construire un algorithme de différenciation à partir d'un chemin différentiel tronqué très structuré issu de techniques de programmation linéaire. La phase de collecte s'étale alors sur 6 tours quand, à notre connaissance, toutes les méthodes connues ne contrôlaient qu'un maximum de 5 tours. De plus, l'algorithme de différenciation ne présente une complexité calculatoire que de 2^{72} opérations. Ceci paraît un progrès substantiel, puisque les précédentes attaques sur 10 tours se faisaient avec une complexité d'environ 2^{392} calculs. Cependant, ceci ne suffit pour le moment pas à menacer la sécurité pratique de Grøstl_{512} .

Nous avons vu que le chemin différentiel tronqué sur 11 tours est réalisé en moyenne par β^{24} couples de valeurs d'état interne. Considérons l'extension naturelle de ce chemin différentiel tronqué sur 12 tours :

$$104 \xrightarrow{R_1} 53 \xrightarrow{R_2} 34 \xrightarrow{R_3} 34 \xrightarrow{R_4} 34 \xrightarrow{R_5} \dots \xrightarrow{R_8} 34 \xrightarrow{R_9} 34 \xrightarrow{R_{10}} 53 \xrightarrow{R_{11}} 104 \xrightarrow{R_{12}} 128.$$

La même analyse de plausibilité assure qu'il existe en moyenne β^2 couples de valeurs d'état interne tels que toutes les différences intermédiaires sont compatibles avec le chemin différentiel tronqué. S'il semble non trivial de construire un algorithme de différenciation avec une complexité calculatoire inférieure à la complexité de l'algorithme de résolution générique de l'instance du problème du paradoxe des anniversaires induite. Il semble pourtant qu'à la lumière de l'existence de ce chemin différentiel tronqué, les permutations de ce type devraient réaliser un minimum de 13 tours pour être considérées sûres.

Les méthodes utilisées pour l'analyse de la structure des différentielle tronquées et la conception d'un algorithme de différenciation semblent pouvoir se généraliser à n'importe quelle structure de type AES, et non réservées aux structures rectangulaires. Il n'y a aucune assurance de pouvoir construire les meilleurs algorithmes de différenciation si ce n'est qu'il semble que les principes qui conduisent à ces meilleurs performances sur les structures rectangulaires sont généraux.

Bibliographie

- [17a] *dictionnaire Le Robert maxi*. Éditions Le Robert, 2017.
- [17b] *Gurobi Optimizer Reference Manual*. Version 7.0. Gurobi Optimization, Inc. 2017. URL : <https://www.gurobi.com>.
- [AF14] D. AUGOT et M. FINIASZ. “Direct Construction of Recursive MDS Diffusion Layers Using Shortened BCH Codes”. In : *Fast Software Encryption - 21st International Workshop, FSE 2014, London, UK, March 3-5, 2014. Revised Selected Papers*. 2014, p. 3-17.
- [Aid86] A.K. AIDINYAN. “On Matrices with nondegenerate square submatrices”. In : *Problems of Information Transmission* 22.4 (1986), p. 106-108.
- [AMP] Elena ANDREEVA, Bart MENNINK et Bart PRENEEL. “On the Indifferentiability of the Grøstl Hash Function”. In : *Security and Cryptography for Networks*. T. 6280. Lecture Notes in Comput. Sci. Springer, p. 88-105.
- [Aum+14] Jean-Philippe AUMASSON et al. “The Hash Function BLAKE”. In : *Information Security and Cryptography (2014)*.
- [Bal12] Siméon BALL. “On sets of vectors of a finite vector space in which every subset of basis size is a basis”. In : *J. Eur. Math. Soc.* 14 (2012), p. 733-748.
- [Bar+10] Paulo BARRETO et al. “Whirlwind : a new cryptographic hash function”. In : *Designs, Codes and Cryptography*. T. 56. 2010, p. 141-162.
- [Ber+] Guido BERTONI et al. “On the indifferentiability of the sponge construction”. In : *Advances in cryptology—EUROCRYPT 2008*. T. 4965. Lecture Notes in Comput. Sci. Springer, p. 181-197.
- [Ber03] T.P. BERGER. In : *IEEE Transactions of Information Theory* 49 (2003), p. 3016-3019.
- [Ber13] T.P. BERGER. “Construction of Recursive MDS Diffusion Layers from Gabidulin Codes”. In : *Progress in Cryptology-INDOCRYPT 2013*. T. LNCS 8250. Springer, 2013, p. 274-285.
- [Ber84] E.R. BERLEKAMP. *Algebraic Coding Theory*. Aegean Press Part, 1984.
- [BFL] Charles BOUILLAGUET, Pierre-Alain FOUQUE et Gaëtan LEURENT. “Security analysis of SIMD”. In : *Selected Areas in Cryptography*. T. 6544. Lecture Notes in Comput. Sci. Springer, p. 351-368.
- [BGU07] D. BOUCHER, W. GEISELMANN et F. ULMER. “Skew cyclic codes”. In : *Applied Algebra in Engineering, Communication and Computing*. T. 18. 2007, p. 379-389.

- [BJ12] Siméon BALL et De Beule JAN. “On sets of vectors of a finite vector space in which every subset of basis size is a basis II”. In : *Design of Codes and Cryptography* 65 (2012), p. 5-14.
- [Bon84] P.G.A. BONNEAU. “Codes et combinatoire”. Thèse de doct. Université Pierre et Marie Curie, Paris, 1984.
- [Bor+12] Julia BORGHOFF et al. “PRINCE : A Low-latency Block Cipher for Pervasive Computing Applications”. In : *Proceedings of the 18th International Conference on The Theory and Application of Cryptology and Information Security*. ASIACRYPT’12. Beijing, China : Springer-Verlag, 2012, p. 208-225.
- [BP94] M. BRONSTEIN et M. PETKOVSEK. “On Ore rings, linear operators and factorisation”. In : *Programming and computer software* 20 (1994), p. 14-18.
- [BR00] Paulo BARRETO et Vincent RIJMEN. “The Whirlpool Hashing Function”. In : (2000).
- [BRF06] Paulo BARRETO, Vincent RIJMEN et Decio Gazzoni FILHO. “The Maelstrom-0 Hash Function”. In : (2006).
- [BS91] Eli BIHAM et Adi SHAMIR. “Differential Cryptanalysis of DES-like Cryptosystems”. In : *Proceedings of the 10th Annual International Cryptology Conference on Advances in Cryptology*. CRYPTO ’90. London, UK, UK : Springer-Verlag, 1991, p. 2-21.
- [BTV18] Andrey BOGDANOV, Elmar TISCHHAUSER et Philip S. VEJRE. “Multivariate Profiling of Hulls for Linear Cryptanalysis”. In : *IACR Trans. Symmetric Cryptol.* 2018.1 (2018), p. 101-125.
- [BU09] D. BOUCHER et F. ULMER. “Coding with skew polynomial rings”. In : *Journal of Symbolic Computation*. T. 44. 2009, p. 1644-1656.
- [BU10] D. BOUCHER et F. ULMER. “Linear codes using skew polynomials with automorphisms and derivations”. In : *Designs, Codes and Cryptography*. 2010.
- [CG13] Kishan CHAND GUPTA et Indranil GHOSH RAY. “On Constructions of Involutory MDS Matrices”. In : *Progress in Cryptology – AFRICACRYPT 2013*. Sous la dir. d’Amr YOUSSEF, Abderrahmane NITAJ et Aboul Ella HASSANIEN. Berlin, Heidelberg : Springer Berlin Heidelberg, 2013, p. 43-60.
- [CGL17] Victor CAUCHOIS, Clément GOMEZ et Reynald LERCIER. “Grøstl Distinguishing Attacke : A New Rebound Attack of an AES-like Permutation”. In : *IACR Trans. Symmetric Cryptol.* 2017.3 (2017), p. 1-23.
- [Cho10] Joo Yeon CHO. “Linear Cryptanalysis of Reduced-Round PRESENT”. In : *Topics in Cryptology - CT-RSA 2010 : The Cryptographers’ Track at the RSA Conference 2010, San Francisco, CA, USA, March 1-5, 2010. Proceedings*. Sous la dir. de Josef PIEPRZYK. Berlin, Heidelberg : Springer Berlin Heidelberg, 2010, p. 302-317.
- [CL17] Victor CAUCHOIS et Pierre LOIDREAU. “About Circulant Involutory MDS Matrices”. In : *Workshop on Coding and Cryptography* (2017).
- [CLM16] Victor CAUCHOIS, Pierre LOIDREAU et Nabil MERKICHE. “Direct construction of quasi-involutory recursive-like MDS matrices from 2-cyclic codes”. In : *IACR Trans. Symmetric Cryptol.* 2016.2 (2016), p. 80-98.

- [CR15] A. CANTEAUT et J. ROUÉ. “On the Behaviors of Affine Equivalent Sboxes Regarding Differential and Linear Attacks”. In : *Advances in Cryptology - EUROCRYPT 2015 - 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part I*. 2015, p. 45-74.
- [Del78] P. DELSARTE. “Bilinear forms over a finite field, with applications to coding theory”. In : *Journal of Combinatorial Theory* 15 (1978), p. 226-241.
- [DH76] Whitfield DIFFIE et Martin HELLMAN. “New directions in cryptography”. In : *IEEE Transactions on Information Theory* 22 (1976), p. 644-654.
- [DKR97] Joan DAEMEN, Lars KNUDSEN et Vincent RIJMEN. “The block cipher Square”. In : *Fast Software Encryption*. Sous la dir. d’Eli BIHAM. Berlin, Heidelberg : Springer Berlin Heidelberg, 1997, p. 149-165.
- [DR02] Joan DAEMEN et Vincent RIJMEN. *The design of Rijndael*. Information Security and Cryptography. Springer, 2002.
- [DR06] Joan DAEMEN et Vincent RIJMEN. “Understanding Two-Round Differentials in AES”. In : *Security and Cryptography for Networks*. T. 4116. Lecture Notes in Comput. Sci. Springer, 2006, p. 78-94.
- [Dür87] A. DÜR. “The automorphism groups of Reed-Solomon Codes”. In : *Journal of Combinatorial Theory, Series A* 44 (1987), p. 69-82.
- [Dys16] Serhii DYSHKO. “Generalizations of the MacWilliams Extension Theorem”. Thèse de doct. 2016.
- [Fer+11] Niels FERGUSON et al. “The Skein Hash Function Family”. In : (2011).
- [FSZ09] Pierre-Alain FOUQUE, Jacques STERN et Sébastien ZIMMER. “Cryptanalysis of Tweaked Versions of SMASH and Reparation”. In : *Selected Areas in Cryptography*. T. 5381. Lecture Notes in Comput. Sci. Springer, 2009, p. 136-150.
- [Gab85] E.M. GABIDULIN. “Theory of codes with maximal rank distance”. In : *Problems of Information Transmission*. 1985.
- [Gau+09a] Praveen GAURAVARAM et al. “Grøstl - a SHA-3 candidate”. In : *Symmetric Cryptography*. Sous la dir. d’Helena HANDSCHUH et al. Dagstuhl Seminar Proceedings 09031. 2009.
- [Gau+09b] Praveen GAURAVARAM et al. “Grøstl - a SHA-3 candidate”. In : *Symmetric Cryptography*. Dagstuhl Seminar Proceedings 09031. 2009.
- [GC90] Henri GILBERT et Guy CHASSÉ. “A Statistical Attack of the FEAL-8 Cryptosystem”. In : *Advances in Cryptology - CRYPTO ’90, 10th Annual International Cryptology Conference, Santa Barbara, California, USA, August 11-15, 1990, Proceedings*. 1990, p. 22-33.
- [Gil14] Henri GILBERT. “A Simplified Representation of AES”. In : *Advances in cryptology ASIACRYPT 2014*. T. 8873. Lecture Notes in Comput. Sci. Springer, 2014, p. 200-222.
- [GP10] Henri GILBERT et Thomas PEYRIN. “Super-Sbox Cryptanalysis : Improved Attacks for AES-Like Permutations”. In : *Fast Software Encryption*. T. 6147. Lecture Notes in Comput. Sci. Springer, 2010, p. 365-383.
- [GPP11] J. GUO, T. PEYRIN et A. POSCHMANN. “The PHOTON Family of Lightweight Hash Functions”. In : *Advances in Cryptology - CRYPTO 2011*. 2011, p. 222-239.

- [GPV15a] K.C. GUPTA, S.K. PANDEY et A. VENKATESWARLU. “On the direct construction of recursive MDS matrices”. In : *Proceedings of WCC 2015*. 2015.
- [GPV15b] K.C. GUPTA, S.K. PANDEY et A. VENKATESWARLU. “Towards a general construction of recursive MDS diffusion layers”. In : *Proceedings of WCC 2015*. 2015.
- [GPV17] K.C. GUPTA, S.K. PANDEY et A. VENKATESWARLU. “Almost Involutory Recursive MDS Diffusion Layers”. In : *Proceedings of WCC 2017*. 2017.
- [GR14] K.C. GUPTA et I.G. RAY. “On Constructions of Circulant MDS matrices for Lightweight Cryptography”. In : *ISPEC 2014*. 2014, p. 564-576.
- [Guo+11] J. GUO et al. “The LED block cipher”. In : *Cryptographic Hardware and Embedded Systems - CHES 2011 - 13th International Workshop, Nara, Japan, September 28 - October 1, 2011. Proceedings*. 2011, p. 326-341.
- [Huf98] W.C. HUFFMAN. “Codes and Groups”. In : *Handbook of Coding Theory*. Sous la dir. de NORTH-HOLLAND. Elsevier Science, 1998, p. 1345-1440.
- [Jea+17] Jérémy JEAN et al. “Optimizing Implementations of Lightweight Building Blocks”. In : *IACR Trans. Symmetric Cryptol.* 2017.4 (2017).
- [Jea13] Jérémy JEAN. “Cryptanalyse de primitives symétriques basées sur le chiffrement AES”. Thèse de doct. Ecole Normale Supérieure, 2013.
- [JF11] Jérémy JEAN et Pierre-Alain FOUQUE. “Practical Near-Collisions and Collisions on Round-Reduced ECHO-256 Compression Function”. In : *Fast Software Encryption*. T. 6733. Lecture Notes in Comput. Sci. Springer, 2011, p. 107-127.
- [JNP14] Jérémy JEAN, María NAYA-PLASENCIA et Thomas PEYRIN. “Improved cryptanalysis of AES-like permutations”. In : *J. Cryptology* 27.4 (2014), p. 772-798.
- [JNS12] Jérémy JEAN, María NAYA-PLASENCIA et Martin SCHLÄFFER. “Improved Analysis of ECHO-256”. In : *Selected Areas in Cryptography*. T. 6544. Lecture Notes in Comput. Sci. Springer, 2012, p. 19-36.
- [Ker83] A. KERCKHOFFS. “La cryptographie militaire”. In : *Journal des Sciences Militaires* (1883), p. 161-191.
- [Kho+14] Khoongming KHOO et al. “FOAM : Searching for Hardware-Optimal SPN Structures and Components with a Fair Comparison”. In : *Cryptographic Hardware and Embedded Systems - CHES 2014*. Sous la dir. de Lejla BATINA et Matthew ROBshaw. Berlin, Heidelberg : Springer Berlin Heidelberg, 2014, p. 433-450.
- [Knu95] Lars R. KNUDSEN. “Truncated and higher order differentials”. In : *Fast Software Encryption*. T. 1008. Lecture Notes in Comput. Sci. Springer, 1995, p. 196-211.
- [Kra+17] Thorsten KRANZ et al. “Shorter Linear Straight-Line Programs for MDS Matrices”. In : *IACR Trans. Symmetric Cryptol.* 2017.4 (2017).
- [Lam+09] Mario LAMBERGER et al. “Rebound distinguishers : results on the full whirlpool compression function”. In : *Advances in cryptology—ASIACRYPT 2009*. T. 5912. Lecture Notes in Comput. Sci. Springer, 2009, p. 126-143.
- [LN97] Rudolf LIDL et Harald NIDERREITER. *Finite Fields*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1997.
- [Loi17] Pierre LOIDREAU. “A New Rank Metric Codes Based Encryption Scheme”. In : (2017). Sous la dir. de Tsuyoshi LANGE Tanjaand Takagi, p. 3-17.

- [LP07] Gregor LEANDER et Axel POSCHMANN. “On the Classification of 4 Bit S-Boxes”. In : *Arithmetic of Finite Fields, First International Workshop, WAIFI 2007, Madrid, Spain, June 21-22, 2007, Proceedings*. 2007, p. 159-176.
- [LS16] M. LIU et S. M. SIM. “Lightweight MDS Generalized Circulant Matrices”. In : *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*. 2016, p. 101-120.
- [LW16] Y. LI et M. WANG. “On the Construction of Lightweight Circulant Involutory MDS Matrices”. In : *Fast Software Encryption - 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*. 2016, p. 121-139.
- [Mac62] F.J. MACWILLIAMS. “Combinatorial Properties of Elementary Abelian Groups”. Thèse de doct. Radcliffe College, 1962.
- [Mat+09] Krystian MATUSIEWICZ et al. “Rebound attack on the full Lane compression function”. In : *Advances in cryptology—ASIACRYPT 2009*. T. 5912. Lecture Notes in Comput. Sci. Springer, 2009, p. 106-125.
- [Mat93] Mitsuru MATSUI. “Linear Cryptanalysis Method for DES Cipher”. In : *Advances in Cryptology - EUROCRYPT '93, Workshop on the Theory and Application of Cryptographic Techniques, Lofthus, Norway, May 23-27, 1993, Proceedings*. 1993, p. 386-397.
- [Men+09a] Florian MENDEL et al. “Improved Cryptanalysis of the Reduced Grøstl Compression Function, ECHO Permutation and AES Block Cipher”. In : *Selected Areas in Cryptography*. T. 5867. Lecture Notes in Comput. Sci. Springer, 2009, p. 16-35.
- [Men+09b] Florian MENDEL et al. “The Rebound Attack : Cryptanalysis of Reduced Whirlpool and Grøstl”. In : *Fast Software Encryption*. T. 5665. Lecture Notes in Comput. Sci. Springer, 2009, p. 260-276.
- [Moo05] T.K. MOON. *Error Correction Coding*. Wiley Interscience. Hoboken, New Jersey : John Wiley et Sons, 2005.
- [Mou+12] Nicky MOUHA et al. “Differential and Linear Cryptanalysis Using Mixed-Integer Linear Programming”. In : *Information Security and Cryptology—ICISC 2012*. T. 7839. Lecture Notes in Comput. Sci. Springer, 2012, p. 57-76.
- [MP13] G.L. MULLEN et D. PANARIO. *Handbook of Finite Fields*. Chapman & Hall/CRC, 2013.
- [MS77] F. J. MACWILLIAMS et N. J. A. SLOANE. *The theory of error correcting codes*. North-Holland mathematical library. Amsterdam, New York : North-Holland Pub. Co. New York, 1977.
- [MT13] Marine MINIER et Gaël THOMAS. “An Integral Distinguisher on Grøstl—512 v3”. In : *Progress in Cryptology - INDOCRYPT 2013*. 2013, p. 50-59.
- [Nay11] María NAYA-PLASENCIA. “How to improve rebound attacks”. In : *Advances in cryptology—CRYPTO 2011*. T. 6841. Lecture Notes in Comput. Sci. Springer, 2011, p. 188-205.
- [Ore33a] Ø. ORE. “On a Special Class of Polynomials”. In : *Transactions of the American Mathematical Society* 35 (1933), p. 559-594.
- [Ore33b] Ø. ORE. “Theory of non commutative polynomials”. In : *Annals of Mathematics* (1933), p. 480-508.

- [Ore34] Ø. ORE. “Contribution to the theory of finite fields”. In : *Transactions of the American Mathematical Society* 36 (1934), p. 243-274.
- [Pey07] Thomas PEYRIN. “Cryptanalysis of GRINDAHL”. In : *Advances in cryptology—ASIACRYPT 2007*. T. 4833. Lecture Notes in Comput. Sci. Springer, 2007, p. 551-567.
- [PH98] V.S. PLESS et W.C. HUFFMAN, éd. *Handbook of Coding Theory, VOLUME I*. North-Holland, 1998.
- [PLK06] Vladimir POPOV, Serguei LEONTIEV et Igor KUREPKIN. *Additionnal Cryptographic Algorithms for Use with GOSR 28147-89, GOST R 34.10-2001, and GOST R 34.11-94 Algorithms*. 2006.
- [Rij+96] Vincent RIJMEN et al. “The cipher SHARK”. In : *Fast Software Encryption*. Sous la dir. de Dieter GOLLMANN. Berlin, Heidelberg : Springer Berlin Heidelberg, 1996, p. 99-111.
- [RL89] R.M. ROTH et A. LEMPEL. “On MDS Codes via Cauchy Matrices”. In : *IEEE transactions on information theory*. T. 35. 6. 1989, p. 1314-1319.
- [RS85] R.M. ROTH et G. SEROUSSI. “On Generator Matrices of MDS Codes”. In : *IEEE transactions on information theory*. T. 31. 6. 1985, p. 826-830.
- [Saa11] Markku-Juhani O. SAARINEN. “Cryptographic Analysis of All 4 x 4-Bit S-Boxes”. In : *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*. 2011, p. 118-133.
- [Saj+12a] Mahdi SAJADIEH et al. “On Construction of Involutory MDS Matrices from Vandermonde Matrices in $GF(2^q)$ ”. In : *Des. Codes Cryptography* 64.3 (2012), p. 287-308.
- [Saj+12b] Mahdi SAJADIEH et al. “Recursive Diffusion Layers for Block Ciphers and Hash Functions”. In : *Fast Software Encryption*. Sous la dir. d’Anne CANTEAUT. Berlin, Heidelberg : Springer Berlin Heidelberg, 2012, p. 385-401.
- [Sas+10] Yu SASAKI et al. “Non-full-active Super-Sbox Analysis : Applications to ECHO and Grøstl”. In : *Advances in cryptology—ASIACRYPT 2010*. T. 6477. Lecture Notes in Comput. Sci. Springer, 2010, p. 38-55.
- [Sch+99] Bruce SCHNEIER et al. *The Twofish Encryption Algorithm : A 128-bit Block Cipher*. New York, NY, USA : John Wiley & Sons, Inc., 1999.
- [Seg55a] B. SEGRE. “Curve razionali normali e k-archi negli spazi finiti”. In : *annali matematica pura e applicata* 39 (1955), p. 357-379.
- [Seg55b] B. SEGRE. “Ovals in a finite projective plane”. In : *Canada Journal of Mathematics* 7 (1955), p. 414-416.
- [Sha49] C. SHANNON. “Communication Theory of Secrecy Systems”. In : *Bell System Technical Journal, Vol 28, pp. 656-715* (1949).
- [Shi+07] Taizo SHIRAI et al. “The 128-bit Blockcipher CLEFIA”. In : *Proceedings of the 14th International Conference on Fast Software Encryption. FSE’07*. Luxembourg, Luxembourg : Springer-Verlag, 2007, p. 181-195.
- [Sim+15] S.M. SIM et al. “Lightweight MDS Involution Matrices”. In : *Fast Software Encryption - 21st International Workshop, FSE 2015, Istanbul, Turkey, March 8-11, 2015, Revised Selected Papers*. 2015, p. 471-493.

- [SM10] Tomoyasu SUZAKI et Kazuhiko MINEMATSU. “Improving the Generalized Feistel”. In : *Fast Software Encryption : 17th International Workshop, FSE 2010, Seoul, Korea, February 7-10, 2010, Revised Selected Papers*. Sous la dir. de Seokhie HONG et Tetsu IWATA. Berlin, Heidelberg : Springer Berlin Heidelberg, 2010, p. 19-39.
- [Wu11] Hongjun WU. “The Hash Function JH”. In : (2011).
- [WWW13] Shengbao WU, Mingsheng WANG et Wenling WU. “Recursive Diffusion Layers for (Lightweight) Block Ciphers and Hash Functions”. In : *Selected Areas in Cryptography*. Sous la dir. de Lars R. KNUDSEN et Huapeng WU. Berlin, Heidelberg : Springer Berlin Heidelberg, 2013, p. 355-371.
- [ZWS17] Lijing ZHOU, Licheng WANG et Yiru SUN. “On Efficient Constructions of Lightweight MDS Matrices”. In : *IACR Trans. Symmetric Cryptol.* 2017.4 (2017).

Titre : Couches de diffusion linéaires à partir de matrices MDS

Mots clés : Cryptographie, symétrique, diffusion linéaire, matrices MDS.

Résumé : Cette thèse s'intéresse à deux aspects de la cryptologie symétrique liés à l'utilisation de matrices MDS dans les couches de diffusion linéaires de primitives.

Une première partie se fonde sur les conceptions de couches de diffusion linéaires de schémas de chiffrement symétrique à partir de matrices MDS. Les associations entre matrices récursives, respectivement circulantes, et polynômes sont calquées pour construire de nouvelles associations entre d'autres structures de matrices et des éléments d'anneaux de polynômes non commutatifs de Ore. À l'instar des matrices récursives et circulantes, ces structures bénéficient d'implémentations matérielles légères. Des codes de Gabidulin dérivent des méthodes de construction directe de telles matrices, optimales en termes de diffusion, proches d'involutions pour l'implémentation.

La seconde partie développe une attaque par différenciation de permutations dont l'architecture s'inspire de l'AES. L'utilisation d'une couche de diffusion linéaire locale avec une matrice MDS induit une description macroscopique de la propagation de valeurs de différences à travers les étapes du chiffrement. Des chemins différentiels tronqués apparaissent, qui servent de point de départ à la conception d'attaques rebond. Les travaux présentés généralisent les attaques rebond connues à l'exploitation de chemins différentiels tronqués structurés non issus d'avalanches libres. Cette structure permet de ne pas consommer tous les degrés de libertés au cours d'une seule étape algorithmique mais de les répartir en trois étapes. Une attaque sur 11 tours d'une permutation de Grostl-512 est alors déployée.

Title : Linear diffusion layers from MDS matrices

Keywords: Cryptography, symmetric, linear diffusion, MDS matrices.

Abstract: This thesis focuses on two aspects of symmetric cryptology related to the use of MDS matrices as building blocks of linear layers for symmetric primitives.

A first part handles designs of linear layers for symmetric ciphers based upon MDS matrices. Associations between recursive, respectively circulant, matrices and polynomials are reproduced between other matrix structures and elements in non-commutative polynomial rings of Ore. As for recursive and circulant matrices, those structures come along with lightweight hardware implementations. From Gabidulin codes are derived direct constructions of MDS matrices with properties close to involution from hardware perspectives.

The second part is about distinguishing attacks on an exemple of AES-like permutations. The use of some MDS matrix to build the linear layer induces a macroscopic description of differential trails through the different steps of the algorithm computing the permutation. Truncated differential path appears, from which rebound attack are built. Original work here generalizes rebound attack applied on permutations of GROSTL-512 from structured differential path not raised from free propagations of differences. This structure allows not to consume all degrees of freedom in a simple algorithmic step but to divide this consumption into three algorithmic steps. An attack of a reduced-round version with 11 rounds of one permutation of GROSTL-512 can then be mounted.