**JEFFERSON EVANDI RICARDINI FERNANDES DE
OLIVEIRA**

**EMPARELHAMENTOS E RETICULADOS:
ESTADO-DA-ARTE EM ALGORITMOS E
PARÂMETROS PARA AS FAMÍLIAS MAIS
FLEXÍVEIS DE SISTEMAS CRIPTOGRÁFICOS.**

Dissertação apresentada à Escola Politécnica
da Universidade de São Paulo para obtenção
do Título de Mestre em Ciências.

São Paulo
2014

**JEFFERSON EVANDI RICARDINI FERNANDES DE
OLIVEIRA**

**EMPARELHAMENTOS E RETICULADOS:
ESTADO-DA-ARTE EM ALGORITMOS E
PARÂMETROS PARA AS FAMÍLIAS MAIS
FLEXÍVEIS DE SISTEMAS CRIPTOGRÁFICOS.**

Dissertação apresentada à Escola Politécnica
da Universidade de São Paulo para obtenção
do Título de Mestre em Ciências.

Área de Concentração:
Engenharia de Computação

Orientador:
Prof. Dr. Paulo Sérgio L. M. Barreto

São Paulo
2014

Este exemplar foi revisado e alterado em relação à versão original, sob responsabilidade única do autor e com a anuência de seu orientador.

São Paulo, 17 de março de 2014.

Assinatura do autor

Assinatura do orientador

FICHA CATALOGRÁFICA

Ricardini Fernandes de Oliveira, Jefferson Evandi

Emparelhamentos e reticulados: estado-da-arte em algoritmos e parâmetros para as famílias mais flexíveis de sistemas criptográficos. / J. E. Ricardini Fernandes de Oliveira. – ed. rev. – São Paulo, 2014.

107 p.

Dissertação (Mestrado) — Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Computação e Sistemas Digitais (PCS).

1. Criptologia #1. 2. Reticulados #2. 3. Algoritmos #3. I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais (PCS). II. t.

“Pairings are extremely flexible, and so are lattices. Apparently this is the reason one sometimes hears that lattices are the new pairings.” — Michael Naehrig

AGRADECIMENTOS

Aproveito a redação deste trabalho, para agradecer a todas as pessoas que contribuíram para minha formação pessoal, acadêmica e profissional, de extrema importância para a realização deste trabalho.

Primeiramente deixo o meus agradecimentos, ao meu amigo e também orientador Paulo Barreto que é um grande pesquisador e pessoa, por ter me dado as oportunidades para chegar chegar até aqui, abrindo portas e me aceitando como seu orientado.

Também agradeço aos meus colegas de trabalho e amigos do LARC, pelos momentos de descontração bem como pelas discussões empolgantes no quadro branco.

Às minhas grandes amigas Vanessa e Andressa, pelas revisões, críticas e correções ortográficas, pelas longas conversas via internet até tarde da noite e por aguentarem minhas reclamações, dando o apoio psicológico que me ajudou a suportar toda a pressão e cansaço no decorrer deste trabalho.

Aos meus pais, Valdemir e Rose, que dedicaram parte de suas vidas para me fornecer os recursos necessários pra chegar onde cheguei, mas sobretudo pela educação e exemplos de vida que me passaram, essenciais para minha formação e base da pessoa me tornei.

Por fim, aos Professores com quem tive a honra de conhecer em minha vida acadêmica pelas lições, cobranças e broncas que me deram. Coisas que vão muito além de transmitir conhecimento.

RESUMO

A criptografia de chave pública é uma área do conhecimento sujeita que é tema de intensa atividade contemporânea de pesquisa. Novos protocolos, primitivas e ataques são propostos com frequência, com semelhanças e diferenças mútuas que podem ser mais ou menos evidentes. Algumas primitivas criptográficas de chave pública mostram-se extremamente férteis em termos de flexibilidade, eficiência e segurança. Duas vertentes que se enquadram nesta categoria são os *emparelhamentos* e os *reticulados*. Por possuírem semelhanças em suas funcionalidades a despeito de possuírem naturezas completamente díspares, além de exibirem uma versatilidade rara em toda a área de criptografia de chave pública, alguns autores propuseram chamar os reticulados de “os novos emparelhamentos”, conforme a ordem cronológica em que essas primitivas passaram a atrair interesse mais vívido de pesquisa. Neste cenário, um estudo comparativo entre elas é de razoável interesse, em particular sobre vantagens e desvantagens que o estado da arte revela sobre a eficiência de cada uma delas. A pesquisa aqui relatada contempla esse estudo, e contribui técnicas de implementação eficiente de emparelhamentos (com ênfase no uso de coordenadas afins, pouco exploradas na literatura), novos parâmetros para a construção de reticulados compactos (na forma das chamadas álgebras discretas de Rojo) e uma técnica inovadora para instanciar reticulados na prática (especificamente, um algoritmo simples e natural para amostrar vetores normalmente distribuídos nos reticulados comumente adotados em sistemas criptográficos).

ABSTRACT

Public key cryptography is an area of knowledge undergoing intense research at present. New protocols, primitives and attacks are often proposed, with mutual similarities and differences that may be more or less evident. Some public key cryptographic primitives tend to be extremely prolific in terms of flexibility, efficiency and security. Two trends that fit this category are *pairings* and *lattices*. Because of their similar functionalities despite their completely disparate nature, and because of their rare versatility within the whole area of public key cryptography, some authors proposed to call lattices “the new pairings,” according to the chronological order by which these primitives began to attract more vivid research interest. In this scenario, a comparative study between them is of reasonable interest, in particular on the advantages and disadvantages that the state of the art reveals about the efficiency of each one. The research reported herein addresses this study, and also contributes efficient pairing implementation techniques (focusing on affine coordinates, which are scarcely explored in literature), new parameters for building compact lattices (in the form of the so-called discrete Rojo algebras) and an innovative technique to instantiate lattices in practical (specifically, a simple and natural algorithm for sampling normally distributed vectors in the lattices that are commonly adopted in cryptographic systems).

SUMÁRIO

Lista de Ilustrações

Lista de Tabelas

Lista de Abreviaturas 11

1 Introdução 12

1.1 Justificativa 13

1.2 Objetivos 14

1.3 Metodologia 15

1.4 Contribuições 15

1.5 Organização 17

2 Criptografia Baseada em Emparelhamentos 18

2.1 Curvas Elípticas 19

2.2 Divisores e Emparelhamentos 22

2.3 Famílias de Curvas 25

2.3.1 Curvas Supersingulares 25

2.3.2 Construções Genéricas 25

2.3.3 Famílias Esparsas de Curvas 26

2.3.4 Famílias Completas de Curvas 27

2.3.5	Famílias Holísticas	29
2.4	Algoritmos para Emparelhamentos	30
2.5	Hipóteses de Segurança em Emparelhamentos	32
2.6	Protocolos Baseados em Emparelhamentos	33
2.6.1	Encriptação Baseda em Identidade	34
2.6.2	Cifrassinatura	37
2.6.3	Assinaturas Curtas	38
2.6.4	Assinaturas Cegas	40
2.7	Sinopse	42
3	Criptografia Baseada em Reticulados	44
3.1	Definições	45
3.2	Famílias de Anéis	47
3.3	Algoritmos para Reticulados	49
3.4	Hipóteses de Segurança em Reticulados	50
3.5	Protocolos Baseados em Reticulados	53
3.5.1	NTRU	53
3.5.2	Encriptação Baseada no Problema LWE	54
3.5.3	Encriptação Homomórfica	56
3.5.4	Assinatura Digital Baseada em NTRU	57
3.5.5	Assinatura Digital sem Alçapão	58
3.5.6	Outros Protocolos Criptográficos	59

3.6	Sinopse	60
4	Implementação Eficiente de Emparelhamentos e Reticulados	62
4.1	Implementação Eficiente de Emparelhamentos	63
4.1.1	Escolha das Curvas	63
4.1.2	Aritmética em Curvas Elípticas	65
4.1.2.1	Coordenadas Afins	66
4.1.2.2	Coordenadas Projetivas Homogêneas	68
4.1.3	Contagem de Operações	68
4.1.4	Resultados	69
4.2	Implementação Eficiente de Reticulados	71
4.2.1	Transformada Rápida de Fourier	72
4.2.2	Amostragem de Vetores de Gaussianas Discretas	75
4.2.3	Álgebra de Rojo	79
4.2.4	Resultados	86
5	Discussões e conclusões	90
5.1	Comparação Sinóptica	90
5.1.1	Flexibilidade	90
5.1.2	Eficiência	91
5.1.3	Segurança	92
5.1.4	Sumário	92
5.2	Conclusões	92

LISTA DE ILUSTRAÇÕES

1	Reticulado de 2 dimensões	45
2	Matriz circulante	48
3	Matriz de Toeplitz	80
4	Matriz de Toeplitz Simétrica	81
5	Matriz de Hankel	81
6	Matriz de Hankel Centrossimétrica	82
7	geração de chaves	87
8	Encriptação	88
9	Decriptação	88

LISTA DE TABELAS

1	Implementações em plataformas de 64 bits (tempos em quilociclos). . .	70
2	Propriedades de criptossistemas baseados em reticulados	86
3	Desempenho de criptossistemas baseados em reticulados (tempos em μs)	87

LISTA DE ABREVIATURAS

BN	<i>Barreto and Naehrig</i>
CM	<i>Complex Multiplication</i>
CVP	<i>Closest Vector Problem</i>
BIDH	<i>Bilinear Inverse Diffie-Hellman</i>
BLS(1)	<i>Curvas Barreto, Lynn and Scott</i>
BLS(2)	<i>Assinaturas Boneh, Lynn and Shachan</i>
BSDH	<i>Bilinear Square Diffie-Hellman</i>
DBDH	<i>Decision Bilinear Diffie-Hellman</i>
CCA	<i>Chosen-Ciphertext Attack</i>
DCT	<i>Discrete Cosine Transform</i>
DLP	<i>Discrete Logarithm Problem</i>
DST	<i>Discrete Sine Transform</i>
ECC	<i>Elliptic Curve Cryptography</i>
ECDSA	<i>Elliptic Curve Digital Signature Algorithm</i>
EUF-CMA	<i>Existential Unforgability Under Adaptive Chosen Message Attack</i>
FFT	<i>Fast Fourier Transform</i>
FWHT	<i>Fast Walsh-Hadamard Transform</i>
GDH	<i>Gap Diffie-Hellman</i>
GGH	<i>Goldreich, Goldwasser and Halevi</i>
GIVP	<i>Gap Independent Vector Problem</i>
IBE	<i>Identity Based Encryption</i>
Inv-CDH	<i>Inverse Computational Diffie-Hellman</i>
k -BDH	<i>k-Bilinear Diffie-Hellman</i>
k -CAA	<i>Collusion Attack with k Traitors</i>
KSS	<i>Kachisa, Schaefer and Scott</i>
LLL	<i>Lenstra, Lenstra and Lovász</i>
LWE	<i>Learning with Errors</i>
MNT	<i>Miyaji, Nakabayashi and Takano</i>
NTT	<i>Number Theoretical Transform</i>
OT	<i>Oblivious Transfer</i>
PKG	<i>Public Key Generator</i>
PKI	<i>Public Key Infrastructure</i>
q -SDH	<i>q-Strong Diffie-Hellman</i>
SCDH	<i>Square Computational Diffie-Hellman</i>
SIS	<i>Short Integer Solution</i>
SIVP	<i>Shortest Independent Vector Problem</i>
SMS	<i>Short Message Service</i>
SVP	<i>shortest Vector Problem</i>
TCL	<i>Teorema Central do Limite</i>

1 INTRODUÇÃO

Emparelhamentos constituem uma primitiva de criptografia tradicional e já considerada tecnologicamente madura, possuindo implementações extremamente otimizadas em PCs (ARANHA et al., 2011; PEREIRA et al., 2011). Desse ponto de vista, a criptografia com emparelhamentos já é uma realidade, sendo usada em diversas aplicações. Ademais, a criptografia baseada em emparelhamentos atrai muita atenção graças a sua flexibilidade e versatilidade, que permitem a elaboração de diversas aplicações e protocolos, desde assinaturas digitais e encriptação comuns até aplicações mais complexas, como criptossistemas baseados em identidade (SAKAI; OHGISHI; KASAHARA, 2000), assinaturas em anéis (CHEN; ZHANG; KIM, 2006; ZHANG; KIM, 2002), assinaturas cegas (ZHANG; KIM, 2002) e protocolos de acordo de chaves autenticados (CHEN; CHENG; SMART, 2007).

Por outro lado, recentemente os reticulados têm se tornado, cada vez mais, alvo de intensa pesquisa, devido a sua grande flexibilidade e versatilidade em aplicações criptográficas, motivando alguns autores a chama-los de “os novos emparelhamentos”. No entanto, sob a perspectiva de maturidade tecnológica, ainda há muita incerteza, pois até o momento em que a presente dissertação foi escrita não foram encontradas implementações tão otimizadas e eficientes como em emparelhamentos. Em geral, criptografia baseada em reticulados ainda necessita de chaves grandes e possui um *overhead* de assinaturas e criptogramas também grandes em comparação com o que é encontrado em criptografia de curvas elípticas, tipicamente usadas em emparelhamentos. Outro problema refere-se a técnicas de amostragem de Gaussianas discretas, sendo esse o

gargalo de processamento em diversas aplicações. Em geral, esse tipo de amostragem é necessária nas operações básicas dos criptossistemas, seja geração de chaves, encriptação ou assinatura. Em muitos protocolos, essas operações exigem um vetor com variáveis normalmente amostradas, seja para a garantia de uma prova formal de segurança.

Tendo em vista o que foi citado acima, é um interesse de pesquisa o confronto entre essas técnicas a fim de verificar até que ponto os reticulados podem ser comparados aos emparelhamentos, principalmente no que tange à eficiência dos algoritmos, bem como ao espaço necessário para chaves, assinaturas e criptogramas.

Neste trabalho são analisados os estados da arte de ambas as primitivas, com o intuito de confrontar os aspectos de flexibilidade e maturidade tecnológica. Também é proposta uma alternativa de parâmetros para reticulados, parametrização esta que evita a utilização de anéis circulantes e polinômios ciclotômicos, muito utilizados em reticulados para otimizar as operações e diminuir o tamanho ocupado por chaves. Aqui ainda é proposta também uma estratégia mais eficiente para a amostragem de gaussianas discretas, necessárias em muitos protocolos baseados em reticulados.

1.1 Justificativa

Por se tratar de uma primitiva já estabelecida e possuir muitas aplicações e implementações eficientes e otimizadas para diversas plataformas, a criptografia baseada em emparelhamentos já é considerada tradicional. No entanto, ela não atende às métricas exigidas em certos cenários com dispositivos de capacidade limitada de recursos como processamento, memória ou largura de banda, como é o caso de redes de sensores sem e dispositivos normalmente utilizados em aplicações do tipo “Internet das Coisas” (MARGI et al., 2010).

No aspecto de segurança, existe ainda uma possível ameaça caso computadores

quânticos de grande porte se tornem tecnologicamente viáveis, graças ao algoritmo de Shor (SHOR, 1997), capaz de fatorar inteiros grandes e calcular logaritmos discretos em tempo polinomial. Por outro lado, primitivas criptográficas baseadas em reticulados ainda são consideradas experimentais, e não possuem a mesma maturidade tecnológica dos emparelhamentos. Mas, por se basearem em problemas computacionais diferentes, em tese não sofrem o mesmo perigo dos esquemas tradicionais. Consequentemente, parecem ser uma boa alternativa, caso se mostrem potencialmente tão eficientes quanto a criptografia baseada em emparelhamentos.

Outro fato que justifica este trabalho é o alto nível de maturidade dos emparelhamentos, o que dificulta avanços mais substanciais relativos ao estado da arte. Isto pode ser observado nos últimos avanços desta área, que em geral se tratam de otimizações mais pontuais para plataformas específicas (ACAR; LAUTER; NAEHRIG M. SHUMOW, 2013; CHEUNG et al., 2011; ARANHA et al., 2013). Em contrapartida, a criptografia baseada em reticulados ainda é uma área de pesquisa que tem muito a ser explorada, e de fato para chegar a um nível de maturidade comparável ao dos emparelhamentos ainda são necessários muitos avanços.

1.2 Objetivos

O objetivo principal deste trabalho é confrontar as primitivas criptográficas baseadas em emparelhamentos com as baseadas em reticulados, tanto de maneira qualitativa (versatilidade para a construção de protocolos) quanto quantitativa (últimos avanços e o estado da arte da implementação eficiente).

Em complemento, este trabalho contempla a apresentação de alternativas para alguns dos aspectos mais essenciais dessas primitivas. Especificamente, abordam-se aqui, por um lado, a construção de algoritmos mais eficientes para o cálculo de emparelhamentos afins (área ainda escassamente explorada na literatura), e por outro, a

proposta de uma nova álgebra (igualmente eficiente, mas livre de patentes) para representar reticulados e técnicas efetivas para a amostragem de variáveis normais em reticulados, necessária em certas operações de muitos protocolos em criptografia baseada em reticulados.

1.3 Metodologia

A metodologia deste trabalho é o confronto de duas primitivas criptográficas segundo os aspectos qualitativos de *versatilidade* e *flexibilidade*, assim como os aspectos quantitativos de *eficiência*.

Nos aspectos qualitativos, avaliam-se os tipos de serviços criptográficos oferecidos pelas duas primitivas, com o objetivo de atestar sua maturidade tecnológica de acordo com os vários protocolos encontrados na literatura, desde as assinaturas e encriptação comuns, até aplicações mais complexas como cifrassinatura, assinaturas cegas e encriptação homomórfica.

Já o aspecto quantitativo da eficiência se avalia em termos das *métricas* de ocupação de espaço (armazenamento e banda) e complexidade de processamento conforme sejam mais relevantes em criptografia, a saber, o tamanho de chaves, assinaturas e criptogramas, o espaço ocupado por código fonte, e os tempos de processamento incorridos (ou, conforme o caso, a complexidade algorítmica).

1.4 Contribuições

A presente dissertação contempla as seguintes contribuições.

- Avaliação sinóptica dos aspectos algorítmicos de duas primitivas criptográficas de naturezas diferentes, uma já bem estabelecida e tecnologicamente madura (emparelhamentos bilineares) e outra ainda considerada experimental e vista

como alternativa (reticulados), mas ambas com um potencial imenso atestado pela literatura abundante acerca de cada uma delas, a ponto de alguns pesquisadores afirmarem sugestivamente que os reticulados seriam os “novos emparelhamentos”.

- Proposta de uma nova parametrização de criptossistemas baseados em reticulados, como alternativa aos reticulados ideais utilizados atualmente (anéis polinomiais circulantes, negacíclicos e ciclotômicos em geral). Especificamente, avaliam-se aqui as denominadas álgebras discretas de Rojo, baseada nas álgebras contínuas propostas por Oscar Rojo (ROJO, 2008), e explora-se como adaptar essas álgebras ao contexto de anéis discretos e finitos, necessários para aplicações criptográficas com reticulados. Tal parametrização, além de evitar as patentes referentes aos parâmetros mais comuns adotados em criptossistemas baseados em reticulados (e.g. NTRU (HOFFSTEIN; PIPHER; SILVERMAN, 1998)) , revela-se tão compacta e quase tão eficiente quanto as outras alternativas.
- Otimização nas estratégias utilizadas para amostragem de Gaussianas discretas. Propõem-se a utilização da transformada rápida de Walsh-Hadamard (FWHT) para refinar uma amostragem normal de n variáveis criptográficas simultaneamente (conforme o cenário comum de reticulados com dimensão n), com custo total amortizado próximo da complexidade ideal, a saber, $O(\lg n)$ por variável.
- Como bônus e fruto de pesquisa realizada em parceria com Diego F. Aranha (UnB) e Patrick Longa (MSR), aperfeiçoa-se neste trabalho o estado da arte no cálculo de emparelhamentos, com uma implementação mais eficiente (recordista de desempenho na literatura) em coordenadas afins e projetivas (ARANHA; BARRETO; LONGA P.AND RICARDN, 2013).

1.5 Organização

O restante deste documento está organizado da seguinte forma:

O capítulo 2 faz uma revisão sobre criptografia baseada em emparelhamentos, com um breve histórico do tema, as principais definições matemáticas envolvidas, bem como os diversos tipos de curvas amigáveis a emparelhamentos, hipóteses de segurança e os protocolos.

Em seguida, no capítulo 3 é feita uma revisão sobre criptografia baseada em reticulados, com os principais protocolos, hipóteses de segurança e as técnicas para redução de chaves. Essas técnicas, em geral, se baseiam na utilização de matrizes circulantes ou polinômios ciclotômicos, o que também permite certas otimizações nas operações básicas.

O capítulo 4 explora o estado de arte de cada uma das primitivas sob ponto de vista de eficiência de implementação. Nesse capítulo encontram-se as contribuições deste trabalho nos aspectos de otimização do cálculos de emparelhamentos (seção: 4.1.4), da utilização de álgebras de Rojo como alternativa aos anéis comuns em reticulados (seção: 4.2.3), e da melhoria na amostragem de gaussianas discretas (Seção 4.2.2).

Finalmente, no capítulo 5 discute-se a maturidade tecnológica das duas primitivas e conclui-se a dissertação.

2 CRIPTOGRAFIA BASEADA EM EMPARELHAMENTOS

Este capítulo é dedicado à criptografia baseada em emparelhamentos. Inicialmente são apresentadas as principais definições relacionadas a emparelhamentos e curvas elípticas. Em seguida são revisados os principais tipos de curvas e os algoritmos eficientes para utilizados para calcular emparelhamentos. Por fim são apresentadas algumas das principais hipóteses de seguranças utilizadas e alguns protocolos baseados em emparelhamentos, com o intuito de demonstrar sua versatilidade.

Inicialmente, o interesse dos emparelhamentos bilineares ou *emparelhamentos* em criptografia foi para propósitos de criptoanálise, em que os emparelhamentos de Weil (WEIL, 1940) e de Tate (ou Tate-Lichtenbaum) (FREY; MÜLLER; RÜCK, 1999) foram utilizados em um ataque ao problema do logaritmo discreto em curvas elípticas (MENEZES; OKAMOTO; VANSTONE, 1993; FREY; RÜCK, 1994).

No entanto, os emparelhamentos se mostraram muito mais relevantes no contexto construtivo, ou seja, sendo usados para construir novos protocolos e aplicações com propriedades únicas, do que no contexto de criptoanálise. Isso foi percebido com o surgimento dos primeiros trabalhos utilizando emparelhamentos para fins construtivos, como o esquema de acordo de chaves autenticado não interativo baseado em identidades de Sakai, Ohgishi e Kasahara (SAKAI; OHGISHI; KASAHARA, 2000), e o protocolo de uma rodada para acordo de chaves tripartido de Joux (JOUX, 2000). Esses trabalhos levaram a uma explosão na quantidade de trabalhos que exploram diversas possibilidades dos emparelhamentos, por exemplo, criptografia baseada em identidade,

assinaturas cegas, entre outras, com funcionalidades ainda mais complexas.

Por outro lado, toda essa flexibilidade tem um preço: emparelhamentos durante muito tempo foram considerados operações extremamente custosas em termos de complexidade de implementação e tempo de processamento (e/ou espaço de memória ocupado, em uma relação entre tempo e espaço requeridos). Isto exige uma escolha muito cuidadosa de algoritmos e parâmetros para que seu uso seja realmente prático.

O trabalho pioneiro de Miller (MILLER, 1986a, 2004) mostrou que emparelhamentos poderiam ser calculados em tempo polinomial, mas ainda havia grande distância do que poderia ser considerado uma implementação verdadeiramente eficiente.

Os progressos nessa linha de pesquisa não apenas revelaram os limites teóricos de quão eficientemente um emparelhamento pode ser calculado, no sentido de ordem de complexidade computacional (VERCAUTEREN, 2010), mas também podem ser encontrados na literatura vários trabalhos de teor prático. De fato, existem implementações extremamente otimizadas, que cobrem todas as operações tipicamente encontradas em criptossistemas baseados em emparelhamentos e não apenas o cálculo do emparelhamento propriamente dito (ARANHA et al., 2011; PEREIRA et al., 2011).

2.1 Curvas Elípticas

Neal Koblitz e Victor Miller (KOBLOITZ, 1987; MILLER, 1986b) perceberam independentemente que criptossistemas baseados no Problema do Logaritmo Discreto (*Discrete Logarithm Problem* ou DLP) poderiam fornecer segurança maior quando definidos no grupo formado pelos pontos em uma curva elíptica em vez do grupo multiplicativo convencional de um corpo finito.

Desde então, foram despendidos muitos esforços em pesquisas em Criptografia Baseada em Curvas Elípticas (*Elliptic Curve Cryptography* ou ECC) e uma grande quantidade de criptossistemas foram propostos. Esse esforço se deu pelo fato de curvas

elípticas possibilitarem o uso de parâmetros mais compactos, fornecendo um nível de segurança similar ao que os métodos tidos como convencionais até então forneciam.

Outro fator que tornou possível a popularização da criptografia baseada em curvas elípticas foi o grande número de protocolos criptográficos que podem ser definidos genericamente sobre grupos cíclicos, independentemente do conjunto sobre o qual está definido o grupo. Isso possibilitou a utilização de curvas elípticas em uma série de protocolos já existentes.

Abaixo uma definição formal de curva elíptica:

Definição 1. *Seja $\mathbb{F}_q$ um corpo finito de tamanho q . Uma curva elíptica $E(\mathbb{F}_q)$ é o conjunto de soluções (x, y) sobre $\bar{\mathbb{F}}_q$ para uma equação da forma:*

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \quad (2.1)$$

onde $a_i \in \mathbb{F}_q$ e $\Delta \neq 0$, em conjunto com um ponto adicional chamado ponto no infinito, aqui denotado por ∞ (HUSEMÖLLER, 2004).

Uma curva elíptica E sobre um corpo finito $\mathbb{F}_q$ ($q \geq 3$) também pode ser definida como o conjunto de soluções em $\bar{\mathbb{F}}_q \times \bar{\mathbb{F}}_q$ da equação de Weierstrass

$$y^2 = x^3 + ax + b \quad (2.2)$$

onde $a, b \in \mathbb{F}_q$ e $\Delta = -16(4a^3 + 27b^2) \neq 0$. Assim como na Definição 1, a curva é definida pelas soluções (x, y) da equação, com $x, y \in \mathbb{F}_q$ em conjunto com ∞ .

A operação de soma com as soluções de uma curva elíptica possui a estrutura de grupo abeliano, que é definida sobre uma curva elíptica E pelo já bem conhecido método das secantes e tangentes (SILVERMAN, 1986).

O número de pontos de uma curva $E(\mathbb{F}_q)$ é chamado de ordem da curva, e denotado por $\#E(\mathbb{F}_q)$. Vale ressaltar que a ordem da curva é o número de soluções da sua equação no corpo subjacente. Segundo o teorema de Hasse (HASSE, 1930), a ordem $\#E(\mathbb{F}_q)$ é

limitado pelo número de elementos do corpo:

Teorema 1. *Seja E uma curva elíptica definida sobre $\mathbb{F}_q$, e seja $n := \#E(\mathbb{F}_q)$. Então:*

$$n = q + 1 - t \quad \text{com } |t| \leq 2\sqrt{q} \quad (2.3)$$

O valor t é chamado o traço de Frobenius de $E(\mathbb{F}_q)$. Reescrevendo-se a equação 2.3 de outra forma:

$$q + 1 - 2\sqrt{q} \leq n \leq q + 1 + 2\sqrt{q} \quad (2.4)$$

Logo, a ordem da curva é da ordem do tamanho do corpo.

Também se define a ordem de um ponto $P \in E(\mathbb{F}_q)$, como o menor inteiro positivo r que, multiplicado por P , resulta o ponto no infinito, ou seja, $rP = \infty$. A partir desta definição, é possível encontrar um subgrupo de r -torção de uma curva elíptica, que consiste no conjunto de pontos de ordem finita e múltipla de r , denotado por

$$E[r] = \{P \in E \mid rP = \infty\} \quad (2.5)$$

A ordem r de um ponto divide a ordem da curva n . Para um dado inteiro r , o chamado grau de mergulho de $E[r]$ é o menor inteiro positivo k tal que $r \mid q^k - 1$.

Seja $E(\mathbb{F}_p)$ uma curva elíptica definida sobre o corpo $\mathbb{F}_p$ com equação $y^2 = x^3 + ax + b$, $a, b \in \mathbb{F}_p$ e $\text{char}(\mathbb{F}_p) \neq 2, 3$.

Considerando a seguinte mudança de variáveis:

$$x_1 = \mu^2 x, \quad y_1 = \mu^3 y, \quad \mu \in \bar{\mathbb{F}}_p^*. \quad (2.6)$$

Ao aplicar a mudança 2.6 à curva $E(\mathbb{F}_p)$, obtém-se uma curva $E' : y_1^2 = x_1^3 + a_1 x_1 + b_1$, com $a_1 = \mu^4 a$, $b_1 = \mu^6 b$. O chamado *invariante j* de E é definido como

$$j = j(E) := 1728 \frac{4a^3}{4a^3 + 27b^2}$$

Aplicando-se a mudança de variáveis 2.6 calculando o invariante $j(E')$, tem-se

$$j(E') = 1728 \frac{4(\mu^4 a)^3}{4(\mu^4 a)^3 + 27(\mu^6)^2} = j(E)$$

Em outras palavras, o invariante j não se altera com essa mudança de variáveis.

É possível construir curvas com um dado número n de pontos sobre um corpo finito $\mathbb{F}_p$, utilizando o método da multiplicação complexa (*complex multiplication* ou CM). Isso é possível desde que $n = q + 1 - t$ seguindo o limite de Hasse, e de que a equação norma $DV^2 = 4q - t^2$ possa ser resolvida para valores pequenos do discriminante D .

A partir da equação norma, o invariante j da curva pode ser calculado e a equação da curva é dada por $y^2 = x^3 + b$ (para certos valores de b) quando $j = 0$, por $y^2 = x^3 + ax$ (para certos valores de a) quando $j = 1728$ e por $y^2 = x^3 - 3cx + 2c$ com $c := j/(j-1728)$ quando $j \notin \{0, 1728\}$ (BLAKE; SEROUSSI; SMART, 1999).

2.2 Divisores e Emparelhamentos

O conceito de divisores é importante para o cálculo de emparelhamentos e funções de Miller. Portanto, aqui são dadas as noções básicas de teoria de divisores de funções, e em seguida é feita uma introdução teórica a emparelhamentos.

Definição 2. *Seja $E(\mathbb{F}_{q^k})$ uma curva elíptica contendo um subgrupo de ordem prima r com grau de mergulho k . Um divisor é definido como a soma formal*

$$\mathcal{A} = \sum_P a_P(P),$$

onde $P \in E(\mathbb{F}_{q^k})$ e $a_P \in \mathbb{Z}$.

Em outras palavras, um divisor é uma associação de um coeficiente inteiro a_P a cada ponto P , ou seja, a função $\mathcal{A} : E(\mathbb{F}_{q^k}) \rightarrow \mathbb{Z}$, $P_i \mapsto a_i$, para todos os pontos $P_i \in E(\mathbb{F}_{q^k})$ e $1 \leq i \leq n$, onde n é a ordem da curva, $E(\mathbb{F}_{q^k})$.

A soma de dois divisores, $\mathcal{A} = \sum_P a_P(P)$ e $\mathcal{A}' = \sum_P a'_P(P)$, é o divisor:

$$\mathcal{A} + \mathcal{A}' = \sum_P (a_P + a'_P)(P).$$

Consequentemente há uma estrutura de grupo abeliano sobre o conjunto de divisores; em particular:

$$n\mathcal{A} = \sum_P (n a_P)(P).$$

Sejam $f : E(\mathbb{F}_{q^k}) \rightarrow \mathbb{F}_{q^k}$ uma função da curva e $\deg(\mathcal{A}) = 0$, onde $\deg(\mathcal{A})$ é o grau de um divisor $\mathcal{A}$, definido pela soma $\deg(\mathcal{A}) = \sum_P a_P$. Pode-se definir $f(\mathcal{A}) \equiv \prod_P f(P)^{a_P}$. Seja $\text{ord}_P(f)$ a multiplicidade do zero ou do polo f em P (se f não tiver zeros ou polos em P , então $\text{ord}_P(f) = 0$). O divisor de f é dado por

$$(f) := \sum_P \text{ord}_P(f)(P) \quad (2.7)$$

Um divisor $\mathcal{A}$ é chamado de principal se $\mathcal{A} = (f)$ para alguma função (f) . Um divisor $\mathcal{A}$ é principal se, e somente se $\deg(\mathcal{A}) = 0$ e $\sum_P a_P P = \infty$ (MENEZES, 1993, theorem 2.25).

Dois divisores $\mathcal{A}$ e $\mathcal{B}$ são equivalentes $\mathcal{A} \sim \mathcal{B}$, se a diferença $\mathcal{A} - \mathcal{B}$ for um divisor principal. Seja $P \in E(\mathbb{F}_q)[r]$ onde r é coprimo a q , e seja $\mathcal{A}_P$ um divisor equivalente a $(P) - (\infty)$; sob essas circunstâncias o divisor $r\mathcal{A}_P$ é principal, logo existe uma função f_P tal que:

$$(f_P) = r\mathcal{A}_P = r(P) - r(\infty)$$

Definição 3. Sejam $\mathbb{G}_1$ e $\mathbb{G}_2$ grupos escritos aditivamente e $\mathbb{G}_T$ um grupo escrito multiplicativamente. Um emparelhamento é uma função bilinear, não degenerada, efetivamente computável e $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Em outras palavras, as seguintes propriedades são válidas para a função e :

- (Bilinearidade) $e(P_1 + P_2, Q) = e(P_1, Q) \cdot e(P_2, Q)$, $e(P, Q_1 + Q_2) = e(P, Q_1) \cdot e(P, Q_2)$ para todo $P, P_1, P_2 \in \mathbb{G}_1$ e todo $Q, Q_1, Q_2 \in \mathbb{G}_2$. Segue daí que

$$e(aP, bQ) = e(P, Q)^{ab} \text{ para todo } a, b \in \mathbb{Z}.$$

- (Não-degeneração) Se $e(P, Q) = 1$ para todo $Q \in \mathbb{G}_2$, então $P = O$. Alternativamente, para todo $P \neq O$ existe $Q \in \mathbb{G}_2$ tal que $e(P, Q) \neq 1$.
- (Computabilidade) O cálculo de $e(P, Q)$ é computacionalmente tratável (tempo polinomial).

Os grupos $\mathbb{G}_1$ e $\mathbb{G}_2$ normalmente são determinados pelo autoespaço do endomorfismo de Frobenius ϕ_q sobre alguma curva elíptica $E(\mathbb{F}_q)$ com grau de mergulho $k > 1$.

Uma função de Miller $f_{i,P}$ é uma função com divisor

$$(f_{i,P}) = i(P) - ([i]P) - (i-1)(\infty).$$

Funções de Miller estão na origem, se não de todos, da maioria dos emparelhamentos para propósitos criptográficos, que por sua vez não algoritmos eficientes derivados do algoritmo de Miller (MILLER, 1986a, 2004). Uma função de Miller satisfaz:

$$f_{a+b,P}(Q) = f_{a,P}(Q) \cdot f_{b,P}(Q) \cdot g_{[a]P,[b]P}(Q) / g_{[a+b]P}(Q) \quad (2.8)$$

para um fator não nulo constante em $\mathbb{F}_q$, para todo $a, b \in \mathbb{Z}$. Na equação 2.8 as chamadas funções linha $g_{[a]P,[b]P}$ e $g_{[a+b]P}$ satisfazem:

$$(g_{[a]P,[b]P}) = ([a]P) + ([b]P) + (-[a+b]P) - 3(\infty) \quad (2.9)$$

e,

$$(g_{[a+b]P}) = ([a+b]P) + (-[a+b]P) - 2(\infty) \quad (2.10)$$

A vantagem das funções de Miller com respeito à aritmética elíptica fica clara quando se leva em consideração as relações com as funções linha das curvas. Assim, as funções de Miller, por si só, podem ser eficientemente calculadas como um resultado secundário do cálculo de $[n]P$, por meio do método de secantes e tangentes normalmente utilizado pra esse tipo de operação.

2.3 Famílias de Curvas

Nem todas as curvas elípticas são ideais para a implementação eficiente de emparelhamentos. De fato, tais curvas, chamadas *amigáveis a emparelhamentos*, são raras e exigem construções especiais. São apresentadas a seguir as principais famílias de curvas amigáveis a emparelhamentos e suas características distintivas.

2.3.1 Curvas Supersingulares

Uma curva elíptica definida sobre um corpo de característica p diz-se supersingular se, e somente se, $p \mid t$, onde t é o traço de Frobenius da curva (MENEZES, 1993).

Propostas iniciais para obter emparelhamentos eficientes dependiam da utilização de curvas supersingulares (GALBRAITH, 2001; RUBIN; SILVERBERG, 2002; DUURSMA; LEE, 2003), o que levou ao surgimento dos altamente eficientes emparelhamentos η (BARRETO et al., 2007) sobre corpos de característica pequena. Este cenário possibilitou o então chamado emparelhamento tipo I, que é definido com todos os argumentos pertencentes ao mesmo grupo (GALBRAITH; PATERSON; SMART, 2008) e facilita a descrição de muitos protocolos e a construção de provas formais de segurança.

Infelizmente, avanços recentes colocaram essa abordagem em dúvida, pois logaritmos discretos em grupos multiplicativos com corpos de extensão associados se mostraram mais fáceis de resolver do que o esperado (BARBULESCU et al., 2013). Por outro lado, certas curvas ordinárias mostraram-se resistentes a esse tipo de ataque, e ainda assim dão suporte a algoritmos muito eficientes, como pode ser visto a seguir.

2.3.2 Construções Genéricas

Métodos de construção genéricos permitem a escolha de um grau de mergulho arbitrário, limitado apenas por questões de eficiência. Abaixo dois exemplos deste tipo de construção:

- A construção Cocks-Pinch (COCKS; PINCH, 2001) permite a obtenção de curvas elípticas sobre $\mathbb{F}_q$ com um grupo de curvas amigáveis a emparelhamentos de ordem n com $\lg(q)/\lg(n) \approx 2$.
- A estratégia Dupont-Engge-Morain (DUPONT; ENGE; MORAIN, 2005) também é genérica do ponto de vista da flexibilidade do seu grau de mergulho, com a maximização do traço do endomorfismo de Frobenius. Entretanto, assim como o método Cocks-Pinch, esta estratégia apenas alcança $\lg(q)/\lg(n) \approx 2$.

Devido à menor razão possível de $\lg(q)/\lg(n)$ ser relativamente grande, esses métodos não produzem curvas de ordem prima. Estas curvas são necessárias em certas aplicações, como assinaturas curtas, e também tendem a melhorar a eficiência de processamento em geral.

2.3.3 Famílias Esparsas de Curvas

Algumas famílias de curvas podem ser obtidas com a parametrização da equação de norma:

$$4q - t^2 = 4hn - (t - 2)^2 = DV^2, \quad (2.11)$$

com polinômios $q(u)$, $t(u)$, $h(u)$, $n(u)$ e escolhendo $t(u)$ e $h(u)$ de acordo com certos critérios (por exemplo, escolhendo $h(u)$ como um polinômio pequeno e constante leva a curvas com ordem quase-prima), e encontrando soluções inteiras (em u e V) como resultado.

Na prática, isso envolve um mapeamento inteligente da equação de norma para uma equação semelhante à equação de Pell, cujas soluções levam a equações de curva por meio da multiplicação complexa (*Complex Multiplication* ou CM).

A única desvantagem que essas famílias apresentam é a relativa raridade de curvas adequadas (os únicos graus de mergulho conhecidos com soluções são $k \in \{3, 4, 6, 10\}$, e o tamanho de soluções inteiras u cresce exponencialmente, precisamente por serem

soluções de uma equação de Pell), especialmente com inteiros de ordem prima. Historicamente, as famílias esparsas de curvas são divididas em curvas Miyaji-Nakabayashi-Takano (MNT) e curvas de Freeman.

As curvas MNT foram as primeiras curvas amigáveis a emparelhamento conhecidas na literatura (MIYAJI; NAKABAYASHI; TAKANO, 2001). Dado seu número limitado de graus de mergulho admissíveis ($k \in \{3, 4, 6\}$), a finitude aparente de curvas MNT de ordem prima (LUCA; SHPARLINSKI, 2006; KARABINA; TESKE, 2008; URROZ; LUCA; SHPARLINSKI, 2012), e por questões de eficiência (veja e.g. (FREEMAN; SCOTT; TESKE, 2010)) as curvas MNT são menos úteis para níveis de segurança mais altos (2^{112} em diante).

Já as curvas de Freeman (FREEMAN, 2006), com grau de mergulho $k = 10$, são muito raras, e têm a desvantagem de não haver um *twist* de grau maior do que quadrático. Isso força o seu grupo $\mathbb{G}_2$ ser definido sobre $\mathbb{F}_{q^5}$. Essa extensão quártica não pode ser construída usando uma representação binomial.

2.3.4 Famílias Completas de Curvas

Nessa família de curvas, em vez de tentar resolver uma equação normal parcialmente parametrizada

$$4h(u)n(u) - (t(u) - 2)^2 = DV^2$$

para u e V diretamente como nas famílias esparsas, $V = V(u)$ também é parametrizado.

Haverá soluções caso os parâmetros possam ser delimitados, o que é normalmente feito considerando as propriedades do corpo numérico $\mathbb{Q}[u]/n(u)$. Mais especificamente, essa limitação se refere à necessidade desse corpo conter a k -ésima raiz da unidade, onde k é o grau de mergulho desejado.

Se $n(u)$ for escolhido de maneira que seja um polinômio ciclotômico $\Phi_\ell(u)$ com $k \mid \ell$, pode-se obter curvas de uma família adequadamente chamada família ciclotô-

mica (BARRETO; LYNN; SCOTT, 2002, 2004; BREZING; WENG, 2005; FREEMAN; SCOTT; TESKE, 2010), que permitem uma razão razoavelmente pequena $\rho := \lg(q)/\lg(n)$ (por exemplo, $\rho = (k+1)/(k-1)$ para o principal $k \equiv 3 \pmod{4}$).

Existe uma família completa de curvas que atinge $\rho \approx 1$, chamada família de curvas Barreto-Naehrig (BN) (BARRETO; NAEHRIG, 2006). As curvas BN estão entre as mais versáteis dentre as curvas amigáveis a emparelhamentos. Uma curva BN é uma curva elíptica de Bachet, ou seja, com a forma

$$E_u : y^2 = x^3 + b,$$

definida sobre um corpo finito primo $\mathbb{F}_p$ de ordem (normalmente prima também) n , onde p e n são dados por:

$$p = p(u) = 36u^4 + 36u^3 + 24u^2 + 6u + 1 \quad (2.12)$$

e

$$n = n(u) = 36u^4 + 36u^3 + 18u^2 + 6u + 1 \quad (2.13)$$

(portanto $t = t(u) = 6u + 1$) respectivamente para algum $u \in \mathbb{Z}$.

Pode-se verificar por inspeção direta que

$$\Phi_{12}(t(u) - 1) = n(u)n(-u),$$

logo

$$\Phi_{12}(p(u)) \equiv \Phi_{12}(t(u) - 1) \equiv 0 \pmod{n(u)},$$

então o grupo de ordem $n(u)$ tem grau de mergulho $k = 12$.

As curvas BN também possuem um invariante $j = 0$, então não há necessidade de se utilizar explicitamente o método CM de construção de curvas. Tudo o que se precisa fazer é escolher um inteiro u de tamanho adequado, tal que p e n , como descrito pelos polinômios acima, sejam primos. Para encontrar uma curva correspondente, deve-se

escolher $b \in \mathbb{F}_p$ para que a curva $E : y^2 = x^3 + b$ tenha ordem n .

Curvas BN admitem um *twist* sêxtico ($d = 6$), de modo que pode-se estabelecer $\mathbb{G}_2 = E'(\mathbb{F}_{p^2})[n]$. Este *twist* $E'/\mathbb{F}_{p^2}$ pode ser escolhido encontrando-se um $\xi \in \mathbb{F}_{p^2}$ não quadrático e não cúbico, e então verificando-se via multiplicação escalar se a curva $E' : y^2 = x^3 + b'$ dada por $b' = b/\xi$ ou $b' = b/\xi^5$ tem ordem divisível por n .

Entretanto, são conhecidos métodos de construção que dispensam o procedimento descrito acima, produzindo a curva certa e seu respectivo *twist* diretamente (PEREIRA et al., 2011). Por conveniência, seguindo a notação de Scott (SCOTT, 2009), aqui o *twist*:

$$E' : y^2 = x^3 + b/\xi$$

será chamado de *twist* tipo D , e o *twist*:

$$E' : y^2 = x^3 + b\xi$$

será chamado de *twist* tipo M .

2.3.5 Famílias Holísticas

Os primeiros trabalhos especificamente orientados a curvas que possuem vantagens do ponto de vista de eficiência, voltavam-se apenas a um ou alguns poucos aspectos de implementação, com foco no cálculo do próprio emparelhamento (BENGER; SCOTT, 2010; DEVEGILI; SCOTT; DAHAB, 2007; SHIRASE, 2010; BEUCHAT et al., 2010).

Abordagens mais modernas tendem a considerar, se não todos, pelo menos a maioria dos aspectos de eficiência que surgem nos esquemas baseados em emparelhamentos (COSTELLO, 2012; COSTELLO; LAUTER; NAEHRIG, 2011; PEREIRA et al., 2011). Isso significa que curvas dessas famílias buscam dar suporte não apenas a cálculos rápidos de emparelhamentos, mas também à aritmética de corpos finitos para todos os corpos finitos envolvidos, construção de curvas, construção de geradores para $\mathbb{G}_1$ e $\mathbb{G}_2$, multiplicação por escalar em ambos os grupos $\mathbb{G}_1$ e $\mathbb{G}_2$, amostragem de pontos e fun-

ções de *hash* que mapeiam valores para pontos da curva (FOUQUE; TIBOUCHI, 2012), e potencialmente em outras operações.

Curiosamente, não há uma grande diversidade entre estas famílias promissoras, que compreendem essencialmente apenas curvas BN, curvas BLS (BARRETO; LYNN; SCOTT, 2002) e curvas KSS (KACHISA; SCHAEFER; SCOTT, 2008).

2.4 Algoritmos para Emparelhamentos

Os algoritmos mais eficientes para emparelhamentos em curvas ordinárias tendem a ser variantes do emparelhamento de Tate (BARRETO et al., 2002; GALBRAITH; HARRISON; SOLDERA, 2002; HESS; SMART; VERCAUTEREN, 2006; LEE; LEE; PARK, 2009; NOGAMI et al., 2008), embora em alguns casos ainda recaiam no emparelhamento de Weil, em casos onde ele ainda é suficientemente eficiente (WEIL, 1940). Em particular, atualmente conhecem-se limites práticos de quão eficiente um emparelhamento pode vir a ser, limites esses alcançados pelos chamados emparelhamentos ótimos.

Como já dito, funções de Miller são essenciais para a definição da maioria dos emparelhamentos usados em criptografia. No entanto, todos os emparelhamentos podem ser definidos individualmente em termos formais. Deste modo, é mais construtivo apresentar essas definições aqui. Considerando uma curva subjacente $E(\mathbb{F}_q)$, que contém um grupo $E(\mathbb{F}_q)[n]$ de ordem prima n , com grau de mergulho k , e sendo $z := (q^k - 1)/n$:

- Emparelhamento de Weil:

$$w(P, Q) := (-1)^n f_{n,P}(Q) / f_{n,Q}(P).$$

- Emparelhamento de Tate:

$$\tau(P, Q) := f_{n,P}(Q)^z.$$

- Emparelhamento Eta (BARRETO et al., 2007) (chamado de *twisted Ate pairing* quando definido sobre uma curva ordinária):

$$\eta(P, Q) := f_{\lambda, P}(Q)^z$$

onde $\lambda^d \equiv 1 \pmod{n}$.

- Emparelhamento Ate (HESS; SMART; VERCAUTEREN, 2006):

$$a(P, Q) := f_{t-1, Q}(P)^z,$$

onde t é o traço de Frobenius.

- Emparelhamento Ate e emparelhamento *twisted Ate* aperfeiçoados (MATSUDA et al., 2007):

$$a_c(P, Q) := f_{(t-1)^c \bmod n, Q}(P)^z,$$

$$\eta_c(P, Q) := f_{\lambda^c \bmod n, P}(Q)^z,$$

para algum $0 < c < k$.

- Emparelhamento Ate ótimo (VERCAUTEREN, 2010):

$$a_{\text{opt}}(P, Q) := f_{\ell, Q}(P)^z$$

para um certo ℓ tal que $\lg \ell \approx (\lg n)/\varphi(k)$.

Os emparelhamentos ótimos proporcionam os laços mais curtos entre todos esses emparelhamentos. Para chegar a valores únicos, a maioria dos emparelhamentos (excetuando-se o emparelhamento de Weil) são reduzidos por meio de uma exponenciação final a z . Técnicas para o cálculo do valor de z compreendem um assunto de pesquisa por si só (SCOTT et al., 2009). Particularmente, para curvas BN com parâmetro u existe um emparelhamento Ate ótimo com laço de ordem $\ell = |6u + 2|$.

Uma tendência clara em trabalhos mais recentes, para atingir altos ganhos de desempenho, tem sido a limitação das curvas a serem usadas nos emparelhamentos a um determinado subconjunto de curvas. Em alguns casos, tais trabalhos se limitam a uma única curva para certo nível de segurança adotado (NAEHRIG; NIEDERHAGEN; SCHWABE, 2010; BEUCHAT et al., 2010; PEREIRA et al., 2011; ARANHA et al., 2011).

2.5 Hipóteses de Segurança em Emparelhamentos

A abundância de protocolos possíveis em emparelhamentos traz consigo uma abundância de hipóteses de segurança, muitas vezes adaptadas especialmente para cada protocolo em particular, embora algumas dessas hipóteses possam ser usadas de maneira mais geral e tenham se tornado “clássicas”. A seguir se encontram alguns exemplos das hipóteses de segurança mais úteis e populares em provas de segurança de protocolos baseados em emparelhamentos.

Sejam $\mathbb{G}_1$ e $\mathbb{G}_2$ de ordem n em notação multiplicativa (e $\mathbb{G}$ representando um grupo qualquer):

- **q-Strong Diffie-Hellman (q-SDH)** (BONEH; BOYEN, 2004) e muitas hipóteses relacionadas (como Diffie-Hellman computacional inverso (Inv-CDH), Diffie-Hellman computacional quadrado (SCDH), Diffie-Hellman bilinear inverso (BIDH), e Diffie-Hellman bilinear quadrado (BSDH) (ZHANG; SAFAVI-NAINI; SUSILO, 2004)): *dada uma tupla de $q + 2$ $(g_1, g_2, g_2^x, \dots, g_2^{x^q}) \in \mathbb{G}_1 \times \mathbb{G}_2^{q+1}$ como entrada, calcula o par $(c, g_1^{1/(x+c)}) \in \mathbb{Z}/n\mathbb{Z} \times \mathbb{G}_1$.*
- **Diffie-Hellman bilinear decisional (DBDH)** (BONEH; FRANKLIN, 2003) e hipóteses relacionadas (como a hipótese k -BDH (BENSON; SHACHAM; WATERS, 2013)): *Dados os geradores g_1 and g_2 dos grupos $\mathbb{G}_1$ e $\mathbb{G}_2$ respectivamente, e dados $g_1^a, g_1^b, g_1^c, g_2^a, g_2^b, g_2^c$, determine se $e(g_1, g_2)^{abc} = e(g_1, g_2)^z$.*

- Diffie-Hellman lacunar (GDH) (OKAMOTO; POINTCHEVAL, 2001): *Dados $(g, g^a, g^b) \in \mathbb{G}^3$ para um certo grupo $\mathbb{G}$ com a ajuda de um oráculo capaz de decidir se $g^{ab} = g^c$ para qualquer $g^c \in \mathbb{G}$ dado, encontre g^{ab} .*
- Função aplicada a $(k + 1)$ expoentes: *Dada uma função $f : \mathbb{Z}/n\mathbb{Z} \rightarrow \mathbb{Z}/n\mathbb{Z}$ e uma sequência $(g, g^a, g^{f(h_1+a)}, \dots, g^{f(h_k+a)}) \in \mathbb{G}_1^{k+2}$ para algum $a, h_1, \dots, h_k \in \mathbb{Z}/n\mathbb{Z}$, calcule $g^{f(h+a)}$ para algum $h \notin \{h_1, \dots, h_k\}$.*

Vale ressaltar que a última hipótese da lista é na verdade uma meta-hipótese, pois é parametrizada por uma função f aplicada aos expoentes. Essa meta-hipótese inclui, por exemplo, a hipótese de Conluio com k Traidores (k -CAA) (MITSUNARI; SAKAI; KASAHARA, 2002), em que $f(x) := 1/x$, e a hipótese de $(k + 1)$ Raízes Quadradas $((k + 1)$ -SR) (ZHANG; CHEN, 2005), onde $f(x) := \sqrt{x}$, entre outros.

Obviamente, nem todas as escolhas de f leva a uma hipótese de segurança consistente. Por exemplo, a escolha de uma função constante certamente não seria uma boa escolha. Portanto, cada instância dessa meta-hipótese de segurança deve ser cuidadosamente analisada, caso a caso.

Do ponto de vista da relação com a complexidade computacional do problema do logaritmo discreto, nem todas essas hipóteses são inteiramente satisfatórias. Em particular, o ataque de Cheon (CHEON, 2006, 2010) mostrou que, graças ao ataque Pollard- ρ (POLLARD, 1978), a hipótese q-SDH pode necessitar de chaves de até 3λ bits para chegar ao nível de segurança de 2^λ , de acordo com a escolha do valor de q . Isso difere da maioria das hipóteses do tipo logaritmo discreto, que normalmente afirmam chegar a um nível de segurança prático de 2^λ para chaves de 2λ bits.

2.6 Protocolos Baseados em Emparelhamentos

Como uma ilustração da enorme flexibilidade e maturidade tecnológica dos emparelhamentos para fins criptográficos, a seguir são apresentados alguns esquemas e

protocolos criptográficos que se utilizam de tais primitivas. Graças ao grande número de protocolos existentes, não é possível abordar e analisar todos aqui. Desta forma, o objetivo da presente sub-seção é apresentar um conjunto razoavelmente representativo de protocolos, mostrando a grande versatilidade dos emparelhamentos.

Entre os principais esquemas baseados em emparelhamentos estão, os criptosistemas baseados em identidade, dentre os quais incluem-se incluindo encriptação baseada em identidades (*Identity Based Encryption* ou IBE) (BONEH; FRANKLIN, 2001), assinaturas digitais (CHA; CHEON, 2003; SAKAI; KASAHARA, 2003), protocolos de acordo de chaves (autenticados) (CHEN; CHENG; SMART, 2007), além extensões hierárquicas de todos esses esquemas com e sem oráculos aleatórios (GENTRY; SILVERBERG, 2002; BOYEN; WATERS, 2006). Também há outros esquemas com emparelhamentos que não são baseados em identidade, mas provêm funcionalidades especiais como *handshakes* secretos (BALFANZ et al., 2003), assinaturas curtas (BONEH; LYNN; SHACHAM, 2002), assinaturas agregadas, em grupos, ou em anéis (BONEH et al., 2003; CHEN; ZHANG; KIM, 2006; ZHANG; SAFAVI-NAINI; SUSILO, 2004), assinaturas cegas (ZHANG; KIM, 2002) e protocolos de cifrassinatura (BARRETO et al., 2005; BOYEN, 2003; LIBERT; QUISQUATER, 2003).

2.6.1 Encriptação Baseda em Identidade

Encriptação assimétrica em geral necessita de uma infraestrutura de chaves públicas (*Public Key Infrastructure* ou PKI) para garantir a autenticidade das chaves publicadas. Essa necessidade advém do o risco da chave pública de um usuário do sistema ter sido substituída por uma chave falsa ou de outro usuário.

Para resolver esse problema foram criadas as PKI, em que uma autoridade certificadora confiável é responsável pela autenticidade das chaves públicas do sistema. Uma das desvantagens do modelo PKI é o armazenamento e ocupação de banda pelos certificados e atualização de tabelas de revogação de certificado, o que dificulta o uso

desse tipo de solução onde existe alguma restrição técnica que limita a banda.

Neste contexto, a encriptação baseada em identidades (*Identity Based Encryption* ou IBE) foi proposta como alternativa a PKI. Esquemas IBE não necessitam de certificados, pois a própria identidade dos usuários é usada como chave pública.

Os emparelhamentos permitiram o surgimento de diversos protocolos IBE, entre esses está o esquema de Boneh e Franklin (BONEH; FRANKLIN, 2001) que será apresentado a seguir para fins de exemplificação.

O esquema IBE básico de Boneh e Franklin é formado pela quádrupla de algoritmos: Inicializa, GeraChaves, Encripta, Decrypta descritas a seguir.

- Inicializa: Dado o parâmetro de segurança k , o algoritmo funciona em três passos.

1. Escolhe-se um primo q , e dois grupos $\mathbb{G}_1, \mathbb{G}_2$ de ordem q , e um mapa bilinear $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_2$. E escolhe-se um gerador aleatório $P \in \mathbb{G}_1$.
2. Escolhe-se $s \in \mathbb{Z}_q^*$ e calcula-se $P_{pub} = sP$.
3. Escolhe-se uma função de *hash* $H_1 : \{0, 1\}^* \rightarrow \mathbb{G}_1^*$ e uma função de *hash* $H_2 : \mathbb{G}_2 \rightarrow \{0, 1\}^n$ para um certo n . Os parâmetros gerais do sistema são:

$$\text{Params} = \langle q, \mathbb{G}_1, \mathbb{G}_2, e, n, P, P_{pub}, H_1, H_2 \rangle,$$

a chave mestra é $s \in \mathbb{Z}_q^*$.

- GeraChaves: Seja uma dada cadeia $ID \in \{0, 1\}^*$ a identidade de um usuário, o algoritmo calcula $Q_{ID} = H_1(ID) \in G_1$, e estabelece a chave privada do usuário dono da identidade ID como sendo $d_{ID} = sQ_{ID}$ onde s é a chave mestra.
- Encripta: Para encriptar uma mensagem M com a chave pública ID , calcula-se $Q_{ID} = H_1(ID) \in G_1^*$ em seguida se escolhe um $r \in \mathbb{Z}_q^*$ ao acaso, o texto encriptado

será:

$$C = \langle rP, M \oplus H_2(g_{ID}^r) \rangle$$

onde $g_{ID} = e(Q_{ID}, P_{pub})$.

- Decripta: Seja $C = \langle U, V \rangle$ um texto encriptado com a chave pública ID. Para decriptar C usando a chave privada $d_{ID} \in G_1^*$ calcula-se:

$$V \oplus H_2(e(d_{ID}, U)) = M.$$

O algoritmo é consistente, pois a encriptação é calculada como sendo o resultado da operação de *ou exclusivo* ($\oplus$) entre a mensagem, M , e o *hash* de g_{ID}^r , o que resulta no texto encriptado, V . A operação de decriçãoção é calculada com o *ou exclusivo* ($\oplus$) entre V e o *hash* de $e(d_{ID}, U)$.

As máscaras aplicadas durante a encriptação e na decriçãoção são iguais, pois:

$$e(d_{ID}, U) = e(sQ_{IS}, rP) = e(Q_{ID}, P)^{sr} = e(Q_{ID}, P_{pub})^r = g_{ID}^r.$$

Logo, a aplicação da decriçãoção após a encriptação produz a mensagem original M como esperado.

Como pode ser visto, a própria identidade do usuário é utilizada como chave pública, evitando assim a necessidade de certificados, listas de revogação de certificados e uma PKI complexa. Por outro lado, esquemas IBE têm seus próprios desafios, entre eles a revogação de uma chave que tenha sido corrompida, pois em princípio isto acarretaria na revogação da própria identidade do usuário. Embora esse seja um tópico de intensa pesquisa, tal assunto não será abordado mais profundamente aqui por estar fora do escopo geral do trabalho.

2.6.2 Cifrassinatura

O conceito de cifraassinatura surgiu em 1997 com o trabalho de Zheng (ZHENG, 1997). Com esta primitiva criptográfica é possível prover confidencialidade, autenticidade e irretratabilidade em um único protocolo. O interesse em cifraassinaturas está na possibilidade de obter todas as propriedades de segurança de encriptação e assinatura com custo menor do que pode ser obtido pela combinação dessas primitivas (i.e., primeiro a encriptação da mensagem depois a assinatura do texto encriptado, ou vice-versa).

A versatilidade dos emparelhamentos permite a implementação de uma série de protocolos de cifraassinatura. Como ilustração, a seguir é apresentado o protocolo proposto por Barreto *et al.* em (BARRETO et al., 2005).

O protocolo de cifraassinatura de Barreto *et al.* é descrito pelos algoritmos Inicializa, GeraChaves, Assina/Encripta e Verifica/Decripta, detalhados a seguir:

- **Inicializa:** Dado o parâmetro de segurança k , a autoridade geradora de chaves públicas (*public key generator* ou PKG) escolhe mapas bilineares $(\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T)$ de ordem prima $p > 2^k$ e os geradores dos grupos $Q \in \mathbb{G}_1$, $P = \psi(Q) \in \mathbb{G}_1$, $g = e(P, Q) \in \mathbb{G}_T$. Então escolhe uma chave mestra $s \xleftarrow{\$} \mathbb{Z}_p^*$, a chave pública geral do sistema é $Q_{pub} = sQ \in \mathbb{G}_2$ e as funções de *hash* são $H_1 : \{0, 1\}^* \rightarrow \mathbb{Z}_p^*$, $H_2 : \{0, 1\}^* \times \mathbb{G}_T \rightarrow \mathbb{Z}_p^*$ e $H_3 : \mathbb{G}_T \rightarrow \{0, 1\}^n$.

Os parâmetros públicos são:

$$\text{param} := \{\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, P, Q, g, Q_{pub}, e, \psi, H_1, H_2, H_3\}$$

- **GeraChaves:** Seja uma dada cadeia $ID \in \{0, 1\}^*$ a identidade de um usuário, a chave privada do usuário é dada por $S_{ID} = \frac{1}{H_1(ID)+s} Q \in \mathbb{G}_2$.
- **Assina/Encripta:** Dada uma mensagem $M \in \{0, 1\}^*$, identidade destinatário da

mensagem ID_B e a chave privada do remetente S_{ID}

1. Escolhe-se $x \xleftarrow{\$} \mathbb{Z}_p^*$, calcula-se $r = g^x$ e $c = M \oplus H_3(r) \in \{0, 1\}^n$.
2. $h = H_2(M, r) \in \mathbb{Z}_p^*$.
3. calcula-se $S = (x + h)\psi(S_{ID})$.
4. calcula-se $T = x(H_1(ID_B)P + \psi(Q_{pub}))$.

O texto cifrado é $\sigma = \langle c, S, T \rangle \in \{0, 1\}^n \times \mathbb{G}_1 \times G_1$.

- Verifica/Decrypta: Dado um texto cifrado $\sigma = \langle c, S, T \rangle$ e a identidade de algum remetente $(ID)_A$, e chave privada S_{ID} .

1. calcula-se $r = e(T, S_{ID})$, $M = c \oplus H_3(r)$ e $h = H_2(M, r)$.
2. A mensagem é aceita se, e somente se, $r = e(S, H_1(ID_A)Q + Q_{pub})g^{-h}$. Nesse caso, o algoritmo devolve a mensagem M e a assinatura $(h, S) \in \mathbb{Z}_p^* \times G_1$.

Ao analisar os algoritmos, pode-se perceber que este protocolo de cifrassinatura é também baseado em identidade, compartilhando várias características funcionais e de segurança com protocolos IBE, como o apresentado na seção 2.6.1.

2.6.3 Assinaturas Curtas

Certas aplicações possuem restrições de banda ou espaço. Assim nesse tipo de aplicação o tamanho da assinatura é um fator de extrema importância. Redes de sensores sem fio, SMS e serviços em que a assinatura deve ser impressa em meio físico com espaço limitado são exemplos desse tipo de aplicação. Um dos esquemas de assinaturas curtas baseadas em emparelhamentos mais conhecidos, são as chamadas assinaturas BLS (BONEH; LYNN; SHACHAM, 2002).

As assinaturas BLS se destacam pelo fato de, para um nível de segurança 2^k possuírem o comprimento $2k$ bits por assinatura, metade do tamanho das assinaturas produzidas por esquemas mais convencionais (ainda que baseados em curvas elípticas)

como Schnorr (SCHNORR, 1990) ou ECDSA (NIST, 2012), que em geral produzem assinaturas com tamanho $4k$ bits para um nível de segurança 2^k .

O esquema de assinaturas BLS consiste de três algoritmos: Inicializa, Assina e Verifica, detalhados a seguir.

- **Inicializa:** Dado o parâmetro de segurança k , os grupos $\mathbb{G}_1$, $\mathbb{G}_2$ e $\mathbb{G}_T$, um mapa bilinear $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$, uma função de *hash* $H : \{0, 1\}^* \rightarrow \mathbb{G}_1$ e um gerador Q do grupo $\mathbb{G}_2$. Escolhe-se um aleatório $x \xleftarrow{\$} \mathbb{Z}_p^*$, e calcula-se $V = xQ \in \mathbb{G}_2$. Onde V é a chave pública e x é a chave privada.
- **Assina:** Dada uma mensagem $M \in \{0, 1\}^*$ e uma chave privada x , para assinar M com a chave x , executam-se os seguintes passos:
 1. calcula-se o *hash* da mensagem, $R = H(M) \in \mathbb{G}_1$.
 2. $\sigma = xR$
 3. usa-se o valor da coordenada x de σ como sendo a assinatura $s \in \mathbb{F}_q$ da mensagem M .
- **Verifica:** Dada uma chave pública $V \in \mathbb{G}_2$, uma mensagem $M \in \{0, 1\}^*$ e uma assinatura $s \in \mathbb{F}_q$, executam-se os seguintes passos:
 1. Encontra-se um $y \in \mathbb{F}_q$, tal que $\sigma = (x, y) \in \mathbb{G}_1$. Caso esse y não exista, a assinatura é considerada inválida.
 2. Verifica-se se σ tem ordem prima p . Caso σ não satisfaça essa restrição, a assinatura é considerada inválida.
 3. Calcula-se $R = H(M) \in \mathbb{G}_1$,
 4. Se $e(\sigma, Q) = e(R, V)$ ou $e(\sigma, Q)^{-1} = e(R, V)$ a assinatura é considerada válida, e é aceita. Caso contrário a assinatura é considerada inválida.

Este protocolo mostra como emparelhamentos podem ser usados em ambientes onde há uma restrição de banda ou armazenamento para as assinaturas. Ou seja, emparelhamento também podem ser utilizados para aplicações de assinaturas comuns, tendo com o objetivo a otimização da utilização dos recursos disponíveis.

2.6.4 Assinaturas Cegas

As chamadas assinaturas cegas são essenciais em diversas aplicações, por exemplo, aplicações de *e-cash* (moeda eletrônica) e *e-voting* (votação eletrônica).

A principal característica desse tipo de assinatura é o fato do signatário não ser capaz de associar uma assinatura a uma mensagem ou usuário específico. Assim, a assinatura é verificável publicamente como uma assinatura comum, mas não é possível identificar a qual mensagem aquela assinatura está associada ou para qual usuário a assinatura foi gerada inicialmente.

O modelo genérico de assinatura cega foi proposto por Chaum (CHAUM, 1983) como uma possível solução para aplicações de moeda eletrônica, e é constituído de um protocolo interativo entre o signatário e o cliente.

Aqui é apresentado o esquema de assinatura cegas de Zhang e Kim (ZHANG; KIM, 2002) como exemplificação da flexibilidade dos emparelhamentos. O esquema de assinatura de Zhang e Kim é definido por um protocolo interativo entre um signatário, chamada aqui de Alice e um outro usuário, dono da mensagem a ser assinada cegamente, chamado aqui de Beto.

Seja $\mathbb{G}$ um grupo de ordem prima p e seja e um mapa bilinear $\mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$. O esquema de assinatura cega é definido pelos algoritmos:

- Inicializa: Dado um gerador P do grupo $\mathbb{G}$. Escolhe-se um aleatório $x \xleftarrow{\$} \mathbb{Z}_p^*$, e calcula-se $P_{pub} = sP$ e define-se duas funções de *hash*, $H_1 : \{0, 1\}^* \rightarrow \mathbb{Z}_q$ e

$H_2 : \{0, 1\}^* \rightarrow \mathbb{G}$. Os parâmetros públicos do sistema são

$$\text{param} = \{G, q, P, P_{pub}, H_1, H_2\}$$

e s é a chave mestra da autoridade.

- **GeraChaves:** Seja ID a identidade de um usuário, calcula-se $Q_{ID} = H_2(ID)$, o algoritmo devolve a chave privada $S_{ID} = sQ_{ID}$.

- **Protocolo interativo para assinatura:**

– **Compromisso:** Alice, escolhe um $r \xleftarrow{\$} \mathbb{Z}_p^*$ ao acaso e executa:

1. $R = rP$

– **Cega:** Dado R enviado por Alice e uma mensagem M , Beto escolhe $a, b \xleftarrow{\$} \mathbb{Z}_p^*$ ao acaso e calcula:

1. $t = e(bQ_{ID} + R + aP, P_{pub})$

2. $c = H_1(M, t) + b(\text{mod } p)$

e envia c para Alice

– **Assina:** Dado c enviado por Beto, Alice calcula:

1. $S = cS_{ID} + rP_{pub}$

e envia S para Beto

– **Desvenda:** Dado S Beto calcula

1. $S' = S + aP_{pub}$

2. $x' = c - b$

A assinatura para uma mensagem M é (S', c')

- **Verifica:** A assinatura é aceita se, e somente se,

$$c' = H_1(M, e(S', P)e(Q_{ID}, P_{pub})^{-c'}).$$

A validade da verificação da assinatura se justifica pelas equações a abaixo:

$$\begin{aligned}
& H_1(M, e(S', P)e(Q_{ID}, P_{pub})^{-c'}) \\
= & H_1(M, e(S + aP_{pub}, P)e(Q_{ID}, P_{pub})^{-c'}) \\
= & H_1(M, e(cS_{ID} + rP_{pub} + aP_{pub}, P)e(Q_{ID}, P_{pub})^{-c'}) \\
= & H_1(M, e(cS_{ID}, P)e(rP_{pub} + aP_{pub}, P)e(Q_{ID}, P_{pub})^{-c'}) \\
= & H_1(M, e(S_{ID}, P)^c e((r + a)P_{pub}, P)e(Q_{ID}, P_{pub})^{-c'}) \\
= & H_1(M, e(Q_{ID}, P_{pub})^c e((r + a)P, P_{pub})e(Q_{ID}, P_{pub})^{-c'}) \\
= & H_1(M, e(Q_{ID}, P_{pub})^{c-c'} e((r + a)P, P_{pub})) \\
= & H_1(M, e(Q_{ID}, P_{pub})^b e(R + aP, P_{pub})) \\
= & H_1(M, e(bQ_{ID} + R + aP, P_{pub})) \\
= & H_1(M, t) = c - b = c'
\end{aligned}$$

Ao analisar esse protocolo, percebe-se que ele também é baseada em identidade, possuindo as mesmas vantagens e desafios apresentados na seção 2.6.1.

2.7 Sinopse

Este capítulo apresentou um panorama geral de criptografia baseadas em curvas elípticas e emparelhamentos, desde os seus conceitos matemáticos mais básicos até os tipos de curvas elípticas onde emparelhamentos podem ser calculados de forma eficiente, alguns esquemas e protocolos criptográficos e as hipóteses em que os esses protocolos baseiam sua segurança.

Com isto mostrou-se como os emparelhamentos são ferramentas extremamente flexíveis, que permitem a implementação de diversos protocolos criptográficos, com uma grande quantidade de trabalhos nessa linha. Estes trabalhos tratam desde apli-

cações básicas como encriptação e assinaturas digitais simples, até aplicações mais complexas como criptossistemas baseados em identidade sem certificado (que evitam a troca de certificados e a necessidade de grandes estruturas de chave pública) e assinaturas cegas (utilizadas em aplicações de votação eletrônica, dinheiro eletrônico, entre outros).

Outro aspecto que demonstra a maturidade tecnológica dessa primitiva é a diversidade de plataformas para as quais existem implementações extremamente otimizadas e eficientes. A simplificação da implementação de emparelhamentos tem sido foco de intensa pesquisa desde o início do século XXI, o que deu origem a uma corrida para torná-los cada vez eficientes e práticos.

3 CRIPTOGRAFIA BASEADA EM RETICULADOS

Este capítulo é dedicado a uma revisão sobre reticulados e um panorama geral de seu uso em criptografia. É dado foco maior nos aspectos mais importantes para a criptografia, como os principais problemas computacionais relacionados e principais protocolos criptográficos baseados em reticulados.

Historicamente, reticulados vêm sendo estudados desde o final do século XVIII por matemáticos Lagrange, Gauss e posteriormente Minkowski. Mais recentemente, se tornaram um tópico de interesse e intensa pesquisa em computação, sendo utilizados como ferramenta algorítmica para a resolução de uma grande variedade de problemas (BABAI, 1986; LENSTRA, 1983; LENSTRA; LENSTRA H.W.; LOVÁSZ, 1982). Assim como os emparelhamentos, os reticulados possuem diversas aplicações em criptoanálise, sendo essa sua aplicação inicial em criptografia (COPPERSMITH, 2001; LAGARIAS; ODLYZKO, 1983; JOUX; STERN, 1998).

O uso de reticulados para fins construtivos em criptografia se iniciou com o trabalho de Ajtai (AJTAI, 1996), no qual são apresentadas propriedades dos reticulados que permitem reduções de complexidade do pior caso para o caso médio. Esse tipo de redução não é conhecido em nenhuma outra classe de problemas em NP . Posteriormente, em outro trabalho de Ajtai (AJTAI, 1998), é demonstrada a dificuldade do mais famoso problema em reticulados, o problema do vetor mais curto (*shortest vector problem* ou SVP), demonstrando esta que era um problema em aberto até então.

Esses trabalhos abriram as portas para uma série de aplicações construtivas de re-

ticulados em criptografia. De fato, uma série de esquemas e protocolos criptográficos baseados em reticulados foram propostos pouco depois dos trabalhos de Ajtai (CAI; CUSICK, 1999; GOLDREICH; GOLDWASSER; HALEVI, 1997; HOFFSTEIN; PIPHER; SILVERMAN, 1998; FISCHLIN; SEIFERT, 1999), alguns desles já criptoanalizados, outros resistindo até aos ataques mais eficientes encontrados na literatura.

3.1 Definições

Um reticulado é o conjunto de todas as combinações lineares inteiras de n vetores (não necessariamente inteiros) linearmente independentes $b_1, \dots, b_n$, e pode ser descrito como o conjunto de pontos com uma estrutura periódica em um espaço de dimensão n . A figura 1 é um exemplo de um reticulado onde $n = 2$.

Figura 1: Reticulado de 2 dimensões

Formalmente, define-se um reticulado da seguinte maneira:

Definição 4. *Seja $\mathbb{R}^n$ um espaço vetorial euclidiano, e $B = \{b_1, \dots, b_n\}$ um conjunto de n vetores linearmente independentes, o reticulado $\mathcal{L}$ em $\mathbb{R}^n$ é um subgrupo aditivo, formado por todas as combinações lineares de B com coeficientes inteiros, ou seja:*

$$\mathcal{L}(b_1, \dots, b_n) = \left\{ \sum_{i=1}^n x_i b_i : x_i \in \mathbb{Z} \right\},$$

onde os vetores $b_1, \dots, b_n$ são denominados vetores da base de $\mathcal{L}$ e o conjunto $B = \{b_1, \dots, b_n\}$ é chamado de base do reticulado $\mathcal{L}$.

Outra maneira de abordar o assunto é por meio de notação matricial, onde B é representado por uma matriz $\mathbb{R}^{n \times n}$. O reticulado gerado por B é definido como $\mathcal{L} = \{Bx \mid x \in \mathbb{Z}^n\}$ de forma que o determinante, $\det(B)$, é independente da escolha da base.

O principal problema computacional associado aos reticulados é o problema do vetor mais curto (*shortest vector problem* ou SVP. Como sugere o nome, o problema consiste em encontrar o vetor não nulo de menor norma do reticulado gerado por uma dada base B . Em criptografia, costuma ser utilizada uma versão de aproximação desse problema, onde o objetivo é encontrar um vetor não nulo cuja norma seja maior do que o vetor mais curto em no máximo um fator γ .

Outro problema importante em reticulados é o problema do vetor mais próximo ou *closest vector problem* (CVP): dados uma base B do reticulado $\mathcal{L}(B)$ e um vetor $t \in \mathbb{R}^n$, deve-se encontrar o vetor $v \in \mathcal{L}(B)$ que seja mais próximo de t . Assim como no caso do SVP, na prática em criptografia é utilizada uma aproximação do CVP, onde o vetor v deve ser mais distante em no máximo um fator γ do que o vetor mais próximo.

Existem outros problemas interessantes em reticulados, como o problema dos vetores independentes mais curtos ou *shortest independent vectors problem* (SIVP) mas, a grosso modo, o objetivo da maioria desses problemas envolve encontrar vetores curtos sob alguma definição apropriada de “curto”. O comportamento desses problemas é em geral muito similar ao do SVP, sendo que uma análise mais detalhada desses problemas pode ser encontrada em (MICCIANCIO; GOLDWASSER, 2002).

Entre os esquemas mais eficientes e que ainda possuem provas de segurança, estão os baseados no problema de Aprendizagem com Erros (*Learning with Errors*, ou

LWE) (BERNSTEIN; BUCHMANN; DAHMEN, 2008, Capítulo 5.4) e sua variante Ring-LWE ou RLWE (LYUBASHEVSKY; PEIKERT; REGEV, 2010). Esses problemas computacionais serão mais profundamente abordados na seção 3.4.

3.2 Famílias de Anéis

São apresentadas aqui as principais famílias de anéis matriciais utilizados em reticulados. Essencialmente, três famílias de anéis ocorrem na literatura para a definição de reticulados: o anel geral $\mathbb{F}_q^{n \times n}$ de matrizes de dimensão $n \times n$ sobre um corpo finito $\mathbb{F}_q$, correspondente a reticulados sem qualquer estrutura particular além da existência de um alçapão (vetor curto ou base curta) relacionado à hipótese de segurança subjacente; os anéis de matrizes circulantes (típicos do criptossistema NTRU, descrito na seção 3.5.1); e os anéis de matrizes negacíclicas (ou, de maneira mais geral, anéis ciclotômicos), relacionados em geral à hipótese LWE e variantes.

O anel geral $\mathbb{F}_q^{n \times n}$ possui desvantagens com relação aos anéis de matrizes circulantes e negacíclicas, essa desvantagem se dá pela definição das matrizes circulantes e negacíclicas que permite uma representação compacta. Uma matriz circulante C é uma matriz quadrada da seguinte forma:

$$C := \begin{bmatrix} c_0 & c_1 & \cdots & c_{N-2} & c_{N-1} \\ c_{N-1} & c_0 & c_1 & & c_{N-2} \\ \vdots & c_{N-1} & c_0 & \ddots & \vdots \\ c_2 & & \ddots & \ddots & c_1 \\ c_1 & c_2 & \cdots & c_{N-1} & c_0 \end{bmatrix}$$

Esta matriz pode ser completamente representada por um vetor c , que aparece em sua primeira linha. As linhas restantes de C são permutações cíclicas do vetor c , com deslocamento igual ao índice de linha. A estrutura da matriz circulante pode ser vista melhor na figura 2.

Figura 2: Matriz circulante

Esta representação compacta, por meio da primeira linha é o que atrai atenção para matrizes com essa estrutura, pois permite compactação das chaves criptográficas.

Uma matriz negacíclica é uma matriz de N com estrutura semelhante à matriz circulante, a principal diferença sendo que, ao circular, os elementos trocam de sinal. Esta diferença influencia na representação polinomial dessas matrizes:

$$N := \begin{bmatrix} c_0 & c_1 & \cdots & c_{n-2} & c_{n-1} \\ -c_{n-1} & c_0 & c_1 & & c_{n-2} \\ \vdots & -c_{n-1} & c_0 & \ddots & \vdots \\ -c_2 & & \ddots & \ddots & c_1 \\ -c_1 & -c_2 & \cdots & -c_{n-1} & c_0 \end{bmatrix}$$

Estas duas famílias, que são casos particulares dos chamados anéis ciclotômicos, isso significa que sua estrutura matricial tem um correspondente polinomial, onde esses polinômios são polinômios ciclotômicos. Vale ressaltar que eses anéis são instâncias populares, mas não exaustivas, de reticulados ideais.

Definição 5. O k -ésimo polinômio ciclotômico (LIDL; NIEDERREITER, 1997, definição 2.44) é o polinômio

$$\Phi_k(x) = \prod_{d|k} (x^d - 1)^{\mu(k/d)},$$

onde

$$\mu(n) = \begin{cases} 1, & \text{se } n = 1; \\ (-1)^\ell, & \text{se } n \text{ é o produto de } \ell \text{ primos distintos;} \\ 0, & \text{se } n \text{ é divisível pelo quadrado de um número primo.} \end{cases}$$

Um anel polinomial ciclotômico é um anel da forma $\mathbb{F}_q[x]/\Phi_k(x)$ para algum k . Tipicamente adota-se $k = 2^{2m}$ sobre um corpo $\mathbb{F}_q$ tal que $q \equiv 1 \pmod{2m}$, uma vez que $\Phi_{2^{2m}}(x) = x^m + 1$ fatora-se inteiramente módulo q nesse caso, proporcionando diversas propriedades interessantes do ponto de vista criptográfico (REGEV, 2005).

Anéis ciclotômicos admitem duas representações naturais, ora como matrizes com certa estrutura entre os coeficientes, ora como um anel polinomial. Em particular, o anel polinomial $\mathbb{F}_q[x]/\Phi_{2^{2m}}(x)$ corresponde naturalmente a matrizes negacíclicas de dimensão $2^m \times 2^m$ sobre $\mathbb{F}_q$.

Uma vantagem dos anéis circulantes e negacíclicos está na representação compacta de chaves em criptosistemas baseados em reticulados. Contudo, essa não é a única vantagem: especificamente, esta estrutura admite algoritmos muito mais eficientes, conforme descrito mais adiante na seção 4.2.

3.3 Algoritmos para Reticulados

Reticulados possuem uma natureza diferente de curvas elípticas utilizadas em emparelhamentos. Logo, os algoritmos e operações em reticulados são diferentes dos encontrados em emparelhamentos, tendo vantagens e desvantagens.

O primeiro contraste entre criptografia baseada em reticulados e a criptografia baseada em emparelhamentos apresentada no capítulo 2 está no tamanho dos corpos finitos envolvidos em cada uma delas. Os corpos finitos utilizados em reticulados, em geral, não muito menores do que os que aparecem em curvas elípticas e emparelha-

mentos. Isso torna as operações em corpos finitos em reticulados potencialmente mais eficientes do que em curvas elípticas.

Além das operações em corpos finitos relativamente pequenos, em reticulados estão presentes também operações com matrizes e vetores, como inversões de matrizes e multiplicações entre matrizes e vetores, operações que em geral tem complexidade cúbica, apesar existirem algoritmos ligeiramente mais eficientes. No entanto, para fins gerais pode-se considerar complexidade cúbica para operações com matrizes que se encontram no anel geral de matrizes, ou seja, não possuem qualquer tipo de estrutura própria.

Por conta dessas operações mais leves, a criptografia com reticulados, juntamente com outras primitivas ditas pós-quânticas, tem se mostrado uma boa alternativa quando o cenário envolve dispositivos com capacidade limitada (GÜNEYSU; LYUBASHEVSKY; PÖPPELMANN, 2012).

3.4 Hipóteses de Segurança em Reticulados

Um fato importante nos principais problemas envolvidos nas hipóteses de segurança em reticulados é que a real dificuldade de tais problemas ainda é um problema em aberto, apesar de se conjecturar que sejam realmente difíceis. Isso não representa um problema para esse tipo de primitiva, já que problemas base da segurança de primitivas criptográficas convencionais também não são NP-difíceis, como é o próprio caso da fatoração de números inteiros utilizada no amplamente adotado algoritmo RSA.

Conforme mencionado anteriormente o principal problema computacional associado aos reticulados é o SVP. Esse problema consiste em encontrar o vetor não nulo de menor norma no reticulado gerado por uma dada base B . Na prática, é utilizado um fator de aproximação γ para o problema SVP, ou seja, deseja-se encontrar um vetor não nulo cuja norma seja inferior ao vetor mais curto multiplicado por γ .

O problema SVP tradicional foi demonstrado NP-difícil para uma classe aleatória de reticulados (AJTAI, 1996), enquanto Micciancio (MICCIANCIO, 1998) provou que o problema é NP-difícil para um fator de aproximação inferior a $\sqrt{2}$, utilizando norma euclidiana. Posteriormente, o fator de aproximação foi aprimorado para $\gamma(n) = n^{O(1/\log \log n)}$ (KHOT, 2010). Por outro lado, para fatores superiores a $\sqrt{n/\log n}$, existem indícios fortes de que o problema SVP não seja NP-difícil (AHARONOV; REGEV, 2004). Isto coloca esquemas baseados no SVP com esse fator de aproximação num estado semelhante às primitivas tradicionais, como curvas elípticas, já que o problema de logaritmo discreto (tanto comum, quanto elíptico), também não é NP-difícil.

Entre os esquemas mais eficientes e que ainda possuem provas de segurança, estão os baseados no problema LWE (BERNSTEIN; BUCHMANN; DAHMEN, 2008, Capítulo 5.4). Nesse problema, são dados (A, v) onde $A \in \mathbb{Z}_q^{m \times n}$ é uniformemente distribuído e v pode ser um vetor escolhido uniformemente de $\mathbb{Z}_q^m$ ou calculado como $v = As + e$ para um $s \in \mathbb{Z}_q^n$ escolhido uniformemente, e $e \in \mathbb{Z}_q^m$ é um vetor escolhido de uma certa distribuição $\mathcal{D}$ sobre os $\mathbb{Z}_q$, geralmente tomada como sendo a distribuição normal. O objetivo do problema é descobrir se v é aleatório ou se foi calculado a partir de $As + e$ com probabilidade não negligenciável.

Mais recentemente, Lyubashevsky, Peikert e Regev usaram anéis polinomiais para definir o problema LWE em anel (*Ring LWE* ou RLWE) (LYUBASHEVSKY; PEIKERT; REGEV, 2010). Seja $f(x) = x^d + 1$, onde d é uma potência de 2 e dado um inteiro q e um elemento $s \in R_q = \mathbb{Z}_q[x]/f(x)$, o RLWE sobre R_q é definido equivalentemente ao LWE, ou seja, é preciso encontrar s que satisfaça as equações $s \cdot a_i + e_i = b_i \pmod{R_q}$, para $1 \leq i \leq n$, onde a_i e b_i são elementos de R_q e a redução modular em R_q é o mesmo que reduzir o polinômio resultante módulo $f(x)$ e seus coeficientes módulo q .

O problema LWE relaciona-se a um problema chamado *Short Integer Solution* (SIS): dado $A \in \mathbb{Z}^{n \times m}$ escolhido de uma distribuição aleatória, o objetivo é encontrar $z \in \mathbb{Z}^m$ tal que $Az = 0 \in \mathbb{Z}^n$ onde z é um vetor curto, de acordo com um fator β ,

de maneira que $0 < \|z\| \leq \beta$. De acordo com (MICCIANCIO; REGEV, 2007, Teorema 9.2), para quaisquer, $q, A \in \mathbb{Z}_q^{n \times m}$ e $\beta \geq \sqrt{mq}^{n/m}$, a instância do problema SIS (q, A, β) admite uma solução.

É importante notar que, no caso de reticulados construídos sobre anéis ciclotômicos em geral, e negacíclicos em particular, existem certas demonstrações de segurança com reduções do pior caso de certos problemas computacionais para o caso médio de problemas computacionais relacionados. Com isso, a segurança dos protocolos estaria baseada na dificuldade desses problemas, que são tidos por intratáveis. Contudo, estritamente falando, não se conhece uma prova formal da intratabilidade dos problemas relacionados, mas somente argumentos circunstanciais e conjecturas.

Em particular, Aharonov e Regev (AHARONOV; REGEV, 2004) demonstraram que o problema de aproximar o vetor mais curto de um reticulado de dimensão n por um fator $\sqrt{n}$ está na classe de complexidade $NP \cap coNP$, uma classe em que a maioria dos problemas tem complexidade polinomial clássica, e dentre as duas exceções conhecidas (a saber, o problema da fatoração inteira e o problema do isomorfismo de grafos), uma admite um algoritmo *quântico* de tempo polinomial. Em outras palavras, aproximar o vetor mais curto com uma precisão até maior que a típica de criptossistemas (em que se supõe, por exemplo, a intratabilidade de aproximações por fatores $O(n)$, $O(n^2)$ ou maiores) está numa classe de complexidade em que quase todos os problemas são *tratáveis* clássica ou quanticamente.

Deve-se, portanto, proceder com cuidado na adoção prática desses reticulados, e acompanhar a evolução das avaliações experimentais de segurança concreta dos parâmetros propostos na literatura.

3.5 Protocolos Baseados em Reticulados

Assim como os emparelhamentos, reticulados são extremamente versáteis, o que abre a possibilidade para diversas aplicações criptográficas, desde funções de *hash* e geradores de números pseudoaleatórios até aplicações mais complexas como protocolos de criptografia assimétrica. Tendo em vista que a principal contribuição dos emparelhamentos é na área da criptografia assimétrica, nada mais natural do que apresentar aqui os principais protocolos assimétricos baseados em reticulados para fins de comparação posterior.

3.5.1 NTRU

O criptossistema NTRU (HOFFSTEIN; PIPHER; SILVERMAN, 1998) foi construído originalmente sobre anéis polinomiais, mas também pode ser definido com base em reticulados, pois o problema subjacente pode ser interpretado como sendo o SVP e o CVP. Portanto, a resolução de algum desses problemas representaria um ataque ao criptossistema, de modo que os parâmetros devem ser ajustados para que algoritmos utilizados para esse fim não possam se usados.

O criptossistema utiliza os seguintes anéis polinomiais:

$$R := \mathbb{Z}[x]/(x^N - 1),$$

$$R_p := (\mathbb{Z}/p\mathbb{Z})[x]/(x^N - 1),$$

$$R_q := (\mathbb{Z}/q\mathbb{Z})[x]/(x^N - 1).$$

onde N , p , q são inteiros positivos.

Definição 6. *Dados os inteiros positivos d_1 e d_2 , $\mathcal{T}(d_1, d_2)$ é definido como sendo a classe de polinômios tais que existam d_1 coeficientes iguais a 1, d_2 coeficientes iguais a -1 e o restante dos coeficientes iguais a zero.*

Os parâmetros públicos são dados por (N, p, q, d) , onde N e p são primos, $\gcd(p, q) = \gcd(N, q) = 1$ e $q > (6d + 1)p$.

O criptossistema em si consiste na seguinte tripla de algoritmos GeraChaves, Encripta e Decrypta:

- **GeraChaves:** Escolhe-se $f \in \mathcal{T}(d + 1, d)$ que possua inversa em R_q , e tal que $f \equiv 1$ em R_p (basta fazer $f \leftarrow pf' + 1$ para algum $f' \in \mathcal{T}(d + 1, d)$ inversível). Escolhe-se também $g \in \mathcal{T}(d, d)$ igualmente inversível em R_q e tal que $g \equiv 0$ em R_p (basta fazer $g \leftarrow pg'$ para algum $g' \in \mathcal{T}(d, d)$ inversível). A chave pública é $h \leftarrow f^{-1} \cdot g \in R_q$.
- **Encripta:** Dado o texto claro $m \in R_p$, escolhe-se aleatoriamente $r \in \mathcal{T}(d, d)$ e calcula-se o texto cifrado $c \leftarrow r \cdot h + m \pmod{q}$, onde h é a chave pública do destinatário.
- **Decrypta:** Calcula-se $m \leftarrow f \cdot c \pmod{p}$.

3.5.2 Encriptação Baseada no Problema LWE

A Encriptação baseada no problema LWE é uma proposta eficiente e que possui demonstração de segurança com base no pior caso de certos problemas (REGEV, 2005, 2010). Inicialmente existia apenas uma demonstração com redução quântica, ou seja, uma vulnerabilidade do criptossistema implicaria em um algoritmo quântico para resolver problemas em reticulados. Mais recentemente, Peikert (PEIKERT, 2009) mostrou uma redução clássica na demonstração de segurança do problema LWE, e depois Lindner e Peikert (LINDNER; PEIKERT, 2011) propuseram o que um dos esquemas de encriptação mais eficientes conhecidos baseados no problema LWE.

O criptossistema Lindner-Peikert pode ser definido como a tripla de algoritmos GeraChaves, Encripta, Decrypta a seguir:

- **GeraChaves:** Escolhe-se $a \xleftarrow{\$} R_q$ uniformemente, onde R_q é o anel $R_q := \mathbb{Z}_q[x]/\langle x^n + 1 \rangle$ de dimensão n . Este elemento pode ser compartilhado entre vários usuários (de maneira análoga a um gerador de uma curva elíptica em sistemas baseados no problema do logaritmo discreto elíptico). Utiliza-se a distribuição normal $\mathcal{D}_{\mathbb{Z}^n, \sigma}$ sobre $\mathbb{Z}^n$ com variância σ^2 para um certo valor σ , amostram-se $f \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^n, \sigma}$ e $g \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^n, \sigma}$ até que os resultados estejam no anel R_q^* . Finalmente, calcula-se $h \leftarrow g - a \cdot f$. A chave privada é f , enquanto a chave pública é o par (a, h) , ou apenas h se a for compartilhado.
- **Encripta:** Para encriptar uma mensagem $M \in \{0, 1\}^n$ de n bits, inicialmente representa-se essa mensagem como um polinômio $m \in R_q$ da forma $m_i := ((q - 1)/2)M_i$. Portanto, um bit $M_i = 0$ continua sendo representado com $m_i = 0$, mas um bit $M_i = 1$ é mapeado no valor $(q - 1)/2$, para $0 \leq i < n$. Utilizando a distribuição normal $\mathcal{D}_{\mathbb{Z}^n, r}$ sobre $\mathbb{Z}^n$ com variância r^2 para um certo valor r , amostram-se $s \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^n, r}$, $u \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^n, r}$ e $e \xleftarrow{\$} \mathcal{D}_{\mathbb{Z}^n, r}$. Finalmente, calculam-se $c \leftarrow a \cdot s + u$ e $d \leftarrow h \cdot s + e + m$, ambos em R_q . O criptograma é o par (c, d) .
- **Decripta:** Dado o criptograma (c, d) , para recuperar a mensagem original calcula-se $m' \leftarrow c \cdot f + d \in R_q$, e decodifica-se $M_i \leftarrow 0$ se $-\lfloor q/4 \rfloor \leq m'_i < \lfloor q/4 \rfloor$, caso contrário $M_i \leftarrow 1$, para $0 \leq i < n$. Em outras palavras, se o coeficiente for mais próximo de 0 que de $q/2$, então o bit correspondente é 0, caso contrário é 1. A decritação será correta desde que $u \cdot f + g \cdot s + e$ esteja dentro do limiar de decodificação, ou seja, $|(u \cdot f + g \cdot s + e)_i| < \lfloor q/4 \rfloor$, para $0 \leq i < n$. Este requisito norteia a escolha de parâmetros.

Lindner e Peikert descrevem heurísticas para estimar o nível de segurança de escolhas concretas de n , q , σ e r , conforme o estado da arte em algoritmos de redução de reticulados do tipo BKZ (LINDNER; PEIKERT, 2011).

3.5.3 Encriptação Homomórfica

A Encriptação homomórfica é interessante no contexto de computação em nuvem, permitindo a construção de aplicações como banco de dados distribuído encriptado, encriptação de disco e mecanismos de busca sobre consultas encriptadas. Assim, a vantagem mais prominente de tal abordagem é que com ela é possível delegar a execução de algoritmos a um servidor sem perder o sigilo dos dados de entrada.

Um esquema de encriptação completamente homomórfica foi proposto por Gentry (GENTRY, 2009), resolvendo um problema que estava em aberto desde 1978, quando Rivest, Adleman e Dertouzos conjecturaram a existência de homomorfismos secretos (RIVEST; ADLEMAN; DERTOUZOS, 1978), onde a função de encriptação também seria um homomorfismo algébrico. Em outras palavras, é possível somar e multiplicar textos encriptados, de modo que, ao serem decryptados, obtém-se o resultado das mesmas operações realizadas diretamente no texto claro correspondente.

Se o espaço de texto às claras for dado por $\{0, 1\}$, então a soma de bits é equivalente a uma disjunção exclusiva lógica, enquanto a multiplicação é equivalente a uma conjunção lógica. Portanto, é possível definir qualquer circuito booleano sobre dados encriptados, o que permite a construção de máquinas de Turing. Torna-se, assim possível executar homomorficamente qualquer algoritmo com argumentos encriptados, gerando uma saída encriptada.

A construção inicial de Gentry (GENTRY, 2009) foi realizada sobre reticulados ideais, como os negacíclicos descritos na seção . O algoritmo de encriptação acrescenta um ruído ao polinômio pertencente ao ideal, de modo que a decryptação funciona desde que tal ruído não ultrapasse um determinado limiar; especificamente, a mensagem pode ser novamente obtida resolvendo o problema CVP.

Gentry apresentou o conceito de auto inicialização (*bootstrapping*) para permitir a redução do ruído utilizando apenas parâmetros públicos. Para isso, alterou a cons-

trução para que o algoritmo de decifração pudesse ser expresso por um circuito com profundidade baixa, especialmente em relação às multiplicações. Com isso, quando o ruído aproxima-se do limiar, um algoritmo de reencifração é utilizado para reduzir homomorficamente o ruído. Infelizmente, o parâmetro público que permite tal ideia torna a chave pública muito grande, da ordem de gigabytes em níveis de segurança mais altos (GRENTY; HALEVI, 2010).

Mais recentemente, Brakerski propôs o uso do problema LWE para construção de encriptação completamente homomórfica (BRAKERSKI; VAIKUNTANATHAN, 2011), reduzindo a complexidade dos algoritmos, realizando cada operação homomórfica em tempo polilogarítmico. Para isso, Brakerski utilizou uma nova estratégia para gerenciar o crescimento do ruído, tornando possível a realização de uma quantidade maior de multiplicações. Em especial, ele propôs um algoritmo para redução modular, implicitamente reduzindo a taxa de crescimento do ruído. Outro algoritmo proposto, a redução de dimensão, permitiu substituir o algoritmo de auto inicialização por um novo método, com parâmetros públicos menores. Todavia, mesmo considerando as otimizações propostas recentemente, a encriptação completamente homomórfica ainda não é prática.

3.5.4 Assinatura Digital Baseada em NTRU

O criptossistema NTRU pode ser modificado para construir esquemas de assinatura digital (HOFFSTEIN et al., 2003; HOFFSTEIN; PIPHER; SILVERMAN, 2001). No entanto, estas propostas não possuem demonstração de segurança e, de fato, existem ataques que permitem recuperar a chave privada dada uma quantidade suficiente de pares de mensagens e assinaturas (SZYDLO, 2003; NGUYEN; REGEV, 2006).

Seja R_q o anel polinomial $R_q := \mathbb{Z}[x]/(x^N - 1)$. O esquema de assinatura NTRU usa uma base “ruim” do reticulado correspondente a R_q como chave pública, representada pelo mesmo polinômio h do esquema de encriptação NTRU. Nessa base, é computa-

cionalmente difícil encontrar um vetor curto. O esquema usa também uma base “boa” desse reticulado como chave privada, representada por quatro polinômios (f, g, F, G) onde f e g são os mesmos polinômios do esquema de encriptação NTRU. Nessa base, um vetor razoavelmente curto pode ser encontrado facilmente. Para gerar essas bases, o signatário escolhe polinômios curtos $f \in R_q$ e $g \in R_q$, de maneira semelhante do esquema de encriptação NTRU. Em seguida, calcula um segundo par de polinômios curtos $F, G \in R$ tal que, $fG - gF = q$. Intuitivamente, esse tipo de equação seria solúvel com o algoritmo estendido de Euclides, mas esse algoritmo não se aplica a anéis polinomiais dessa natureza (em outras palavras, R_q não é um domínio euclidiano). A técnica de construção sugerida por Hoffstein *et al.* (HOFFSTEIN et al., 2003) recorre ao cálculo de resultantes e refinamentos heurísticos da base obtida.

O processo de assinatura envolve o uso de um oráculo aleatório que mapeia uma mensagem m para um vetor de inteiros, $H : \{0, 1\}^* \rightarrow \mathbb{Z}^{2n}$ de acordo com uma certa distribuição. Para assinar uma mensagem m , calcula-se $M = H(m) \in \mathbb{Z}^{2n}$. Hoffstein *et al.* (HOFFSTEIN et al., 2003) provam que a chave secreta (f, g, F, G) permite calcular um vetor do reticulado $(s, t) \in \mathbb{Z}^{2n}$ muito próximo de M . Esse vetor é adotado como assinatura da mensagem. Para verificar uma assinatura (s, t) de uma mensagem $M = H(m)$, verifica-se se (s, t) é suficientemente próximo de M , conforme uma métrica heurística correspondente à qualidade da base privada obtida com a construção acima.

3.5.5 Assinatura Digital sem Alçapão

Como visto na seção anterior, muitos esquemas de assinatura baseados em reticulados têm sua origem em protocolos de encriptação, ou dependem de algum alçapão em algum momento. No entanto, um esquema de assinaturas baseado em reticulados e sem alçapão foi proposto por Lyubashevsky (LYUBASHEVSKY, 2012, 2009) e aperfeiçoado por Güneysu *et al.* (GÜNEYSU; LYUBASHEVSKY; PÖPPELMANN, 2012) (numa variante aqui denominada GLP).

Esse esquema baseia-se no problema SIS e tem uma estrutura similar às assinaturas Schnorr (SCHNORR, 1990) baseadas no problema do logaritmo discreto. Definindo $R := \mathbb{Z}_q[x]/\langle x^n + 1 \rangle$, $\mathcal{R}_r := (\mathbb{Z}_q[x] \cap \{-\lfloor r/2 \rfloor \dots \lceil r/2 \rceil\})/\langle x^n + 1 \rangle$ (para qualquer inteiro r), e $\mathcal{D}_w^n := \{u \in \mathcal{R}_2 \mid \text{wt}(u) \leq w\}$, e sendo $H : \mathcal{R} \times \{0, 1\}^* \rightarrow \mathcal{D}_w^n$ um oráculo aleatório, o esquema aperfeiçoado GLP consiste em três algoritmos GeraChaves, Assina e Verifica:

- **GeraChaves:** Amostra-se $a \xleftarrow{\$} R$ (que pode ser compartilhado entre todos os usuários), e amostram-se $s_1, s_2 \xleftarrow{\$} R_2$. Calcula-se $v \leftarrow as_1 + s_2 \in R$. A chave privada é o par (s_1, s_2) , e a chave pública é o par (a, v) .
- **Assina:** Para assinar $m \in \{0, 1\}^*$ com a chave privada (s_1, s_2) , amostram-se $y_1, y_2 \xleftarrow{\$} R_{2k}$ (para um certo k , que não pode ser nem pequeno demais nem grande demais); repetem-se os cálculos $u \leftarrow ay_1 + y_2 \in R$, $c \leftarrow H(u, m) \in \mathcal{D}_w^n$, $z_1 \leftarrow s_1c + y_1$, $z_2 \leftarrow s_2c + y_2$, até que $z_1, z_2 \in R_{2(k-w)}$, e finalmente devolve-se a assinatura (z_1, z_2, c) .
- **Verifica:** Dada a chave pública v , a mensagem m e uma assinatura (z_1, z_2, c) , calcula-se $c' \leftarrow H(az_1 + z_2 - vc, m)$, e aceita-se a assinatura se, e somente se, $z_1, z_2 \in R_{2(k-w)}$ e $c' = c$.

3.5.6 Outros Protocolos Criptográficos

Além dos já citados, existem outros protocolos, mais complexos, que utilizam primitivas criptográficas baseadas em reticulados, muitos dos quais apareceram inicialmente na literatura como aplicações de curvas elípticas e emparelhamentos. Entre esses estão esquemas de assinatura cega (RÜCKERT, 2010), de encriptação baseada em identidade (GENTRY; PEIKERT; VAIKUNTANATHAN, 2008), transferência despercebida (*oblivious transfer* ou OT) (PEIKERT; VAIKUNTANATHAN; WATERS, 2008) e esquemas de identificação com prova de conhecimento zero baseada em identidade (MICCIAN-

CIO; VADHAN, 2003).

É interessante ressaltar, que muitos desses protocolos avançados são apenas provas de conceito. Por exemplo, o esquema de assinatura cega de Rückert exige que os problemas subjacentes em que se baseia sua segurança permaneçam intratáveis com fatores de aproximação muito mais grosseiros que protocolos mais simples (especificamente, $\tilde{O}(n^5)$ em vez de apenas $O(n)$ ou $O(n^2)$ para reticulados de dimensão n), além de ser extremamente ineficiente, com assinaturas de até 1.23 MiB, ocupação de banda para o protocolo de obtenção de assinatura de até 6.77 MiB, e processamento até 4 ordens de grandeza mais lento do que esquemas convencionais de assinatura cega¹.

Por brevidade, omite-se aqui a exposição de todos esses esquemas mais complexos neste trabalho, que se tornaria repetitivo. Por outro lado, encoraja-se o leitor interessado a consultar as referências acima.

3.6 Sinopse

Este capítulo apresentou um panorama geral de criptografia baseada em reticulados desde os conceitos matemáticos básicos envolvidos até sua utilização em protocolos criptográficos. Foram apresentadas também os principais problemas computacionais que fundamentam a segurança da criptografia baseada em reticulados, além de alguns esquemas e protocolos criptográficos. Este panorama mostrou que os reticulados podem ser tão flexíveis quanto os emparelhamentos. Mesmo havendo alguns tipos que aplicações que são possíveis apenas com emparelhamento e outros apenas com reticulados, em aspectos mais gerais ambos são igualmente flexíveis.

Outra característica importante é a segurança. Como visto, existem vários problemas que envolvem reticulados e podem ser utilizados como base para segurança sua aplicação na área de criptografia. Alguns desses problemas possuem reduções de pior

¹http://www.spms.ntu.edu.sg/Asiacrypt2010/AsiaCrypt_slides/8th\%20Dec\%202010/blindsig.pdf

caso para o caso médio, fato importante quando se trata de provas de segurança. Do ponto de vista prático a segurança dos reticulados é baseada testes empíricos com os ataques mais eficientes que são usados para o ajuste dos parâmetros a serem utilizados.

Com relação a eficiência, verificou-se também que existem meios eficientes de implementar protocolos baseados em reticulados, sendo que a operação básica mais usada é a multiplicação entre matrizes, ou de matriz por vetor, operações em geral mais leves que as operações típicas dos cálculos de emparelhamentos. Assim, apesar de serem consideradas mais experimentais, as primitivas baseadas em reticulados possuem grande flexibilidade e possibilidades de otimização, o que torna possível a comparação com os emparelhamentos, em certos aspectos, como a flexibilidade e versatilidade.

4 IMPLEMENTAÇÃO EFICIENTE DE EMPARELHAMENTOS E RETICULADOS

Nesse capítulo é apresentado e discutido o estado da arte de criptografia baseada em emparelhamentos e reticulados. Aqui também são descritas as contribuições resultantes deste trabalho.

Uma delas é uma contribuição complementar, resultante de pesquisas em parceria com Diego F. Aranha e Patrick Longa, e as demais são independentes

A primeira contribuição exposta trata do aperfeiçoamento do estado da arte em emparelhamentos, com uma implementação mais eficiente de emparelhamentos, sendo que a implementação apresentada aqui é a melhor para coordenadas afins e projetivas do que as encontradas na literatura. Essa implementação desmistifica a crença de que emparelhamentos com coordenadas afins são extremamente caros computacionalmente.

A contribuição seguinte é a análise de viabilidade da utilização de um novo tipo de álgebra em criptografia baseada em reticulados. Atualmente na literatura, são utilizados anéis de matrizes circulantes ou anéis de polinômios ciclotômicos, em particular os anéis negacíclicos. Aqui propõe-se a utilização de uma álgebra de natureza diferente destas, descrita na seção 4.2.3 deste capítulo. Tal parametrização tem potencial de ser tão eficiente em espaço ocupado quanto as alternativas encontradas na literatura, e por ser de uma natureza diferente dos anéis cíclicos ou ciclotômicos, mostra-se como uma boa alternativa caso esses se mostrem inseguros no futuro. Outra vantagem

desta nova parametrização está no âmbito não acadêmico, que é o fato de evitar certas patentes existentes para sistemas que utilizam anéis cíclicos e ciclotômicos, como o NTRU (HOFFSTEIN; PIPHER; SILVERMAN, 1998).

Finalmente, propõe-se aqui um método alternativo para amostrar vetores segundo a distribuição normal. O método proposto estende uma técnica pouco utilizada na literatura, e adapta-se com naturalidade às características dos esquemas criptográficos, a saber, a necessidade de amostrar n variáveis normais simultaneamente num reticulado de dimensão n (onde n é, via de regra, uma potência de 2).

4.1 Implementação Eficiente de Emparelhamentos

Existem várias técnicas para implementação eficiente de emparelhamentos, desde otimizações básicas de aritmética em corpos finitos, necessária para cálculo de emparelhamento, até melhorias e otimizações nos próprios algoritmos para emparelhamentos.

4.1.1 Escolha das Curvas

Conforme discutido na seção 2.3.4, a família de curvas BN é ideal, de um ponto de vista de implementação, para uso em esquemas baseados em emparelhamento. Possuindo um grau de mergulho $k = 12$, ela é perfeitamente adequada para o nível de segurança de 128 bits e um candidato competitivo no nível de segurança de 192 bits para protocolos envolvendo um número pequeno de cálculos de emparelhamento (ARANHA et al., 2013). Adicionalmente, o tamanho dessa família facilita a geração (PEREIRA et al., 2011) e admite muitas escolhas de parâmetros diferentes, permitindo implementações de software personalizadas para diferentes arquiteturas (GOUVÊA; LÓPEZ, 2009; GREWAL et al., 2012; ARANHA et al., 2011).

A construção do emparelhamento Ate ótimo aplicada a curvas BN genéricas proporciona uma formulação mais simples entre as outras possíveis combinações (LEE;

LEE; PARK, 2009; NOGAMI et al., 2008):

$$a_{\text{opt}} : \mathbb{G}_2 \times \mathbb{G}_1 \rightarrow \mathbb{G}_T$$

$$(Q, P) \mapsto (f_{\ell, Q}(P) \cdot g_{[\ell]Q, \phi_p(Q)}(P) \cdot g_{[\ell]Q + \phi_p(Q), -\phi_p^2(Q)}(P))^{\frac{p^{12}-1}{n}},$$

com $\ell = 6u + 2$, o mapa ϕ_p e os grupos $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ como definido anteriormente; e uma modificação eficiente do algoritmo de Miller para acumulação de todas as avaliações de linha necessárias na variável f no laço de Miller(algoritmo 1).

Algoritmo 1 Emparelhamento Ate ótimo em curvas BN gerais (ARANHA et al., 2011).

Input: $P \in \mathbb{G}_1, Q \in \mathbb{G}_2, \ell = |6u + 2| = \sum_{i=0}^{\log_2(\ell)} \ell_i 2^i$

Output: $a_{\text{opt}}(Q, P)$

```

1:  $d \leftarrow g_{Q,Q}(P), T \leftarrow 2Q, e \leftarrow 1$ 
2: if  $\ell_{\lfloor \log_2(\ell) \rfloor - 1} = 1$  then  $e \leftarrow g_{T,Q}(P), T \leftarrow T + Q$ 
3:  $f \leftarrow d \cdot e$ 
4: for  $i = \lfloor \log_2(\ell) \rfloor - 2$  downto 0 do
5:    $f \leftarrow f^2 \cdot g_{T,T}(P), T \leftarrow 2T$ 
6:   if  $\ell_i = 1$  then
7:      $f \leftarrow f \cdot g_{T,Q}(P), T \leftarrow T + Q$ 
8:   end if
9: end for
10:  $Q_1 \leftarrow \phi_p(Q), Q_2 \leftarrow \phi_p^2(Q)$ 
11: if  $u < 0$  then
12:    $T \leftarrow -T, f \leftarrow f^{p^6}$ 
13: end if
14:  $d \leftarrow g_{T,Q_1}(P), T \leftarrow T + Q_1, e \leftarrow g_{T,-Q_2}(P), T \leftarrow T - Q_2, f \leftarrow f \cdot (d \cdot e)$ 
15:  $f \leftarrow f^{(p^6-1)(p^2+1)(p^4-p^2+1)/n}$ 
16: return  $f$ 
```

A aritmética do corpo de extensão envolvendo f em todos os pontos do algoritmo 1 é uma das questões principais para o cálculo de emparelhamento, junto do algoritmo de Miller e da exponenciação final. Por isso, a implementação eficiente dessas operações é fundamental. Para tanto, recomenda-se a implementação de corpos de extensão por meio de uma construção por torre de extensões com os polinômios irredutíveis apropriados (DEVEGILI; SCOTT; DAHAB, 2007; HANKERSON; MENEZES; SCOTT, 2008; BEUCHAT et al., 2010; PEREIRA et al., 2011).

Ao permitir que valores intermediários passem para dupla precisão, e escolhendo

p como sendo um número primo levemente menor do que o limite de uma palavra do processador, pode-se adotar uma técnica de redução postergada, que pode ser eficientemente empregada em todos os níveis da torre aritmética (ARANHA et al., 2011). Um conjunto notavelmente eficiente surge da escolha da curva $E(\mathbb{F}_p) : y^2 = x^3 + 2$, com $p \equiv 3 \pmod{4}$, e $\beta = -1$, $\xi = (1 + i)$ (PEREIRA et al., 2011), otimizando simultaneamente a aritmética no corpo finito e na curva elíptica.

A estrutura particular de módulos primos tem sido explorada com sucesso, por meio de otimizações de desempenho tanto em *software* (NAEHRIG; NIEDERHAGEN; SCHWABE, 2010) quanto em *hardware* (FAN; VERCAUTEREN; VERBAUWHEDE, 2012). Implementações atuais de *software* contam com a redução Montgomery padrão (MONTGOMERY, 1985) e o estado da arte de implementações em *hardware* se baseiam na possibilidade de paralelização da Representação Numérica por Resíduos (RNS) (CHEUNG et al., 2011).

Estratégias de dividir-e-conquistar são úteis apenas para multiplicações em $\mathbb{F}_{p^2}$, $\mathbb{F}_{p^6}$ e $\mathbb{F}_{p^{12}}$, pois o método Karatsuba (KARATSUBA; OFMAN, 1962) é normalmente mais eficiente em corpos de extensão. As fórmulas completas para implementação de aritmética em corpos estendidos podem ser encontradas em (ARANHA et al., 2011), incluindo estratégias para reduzir o número de reduções de Montgomery, via reduções postergadas.

4.1.2 Aritmética em Curvas Elípticas

A seguir, a notação para os custos das operações é semelhante à notação encontrada em (ARANHA et al., 2011). As grandezas m, s, a, i denotam o custo de multiplicação, quadrado, adição e inversão em $\mathbb{F}_p$, respectivamente; $\tilde{m}, \tilde{s}, \tilde{a}, \tilde{i}$ denotam o custo de multiplicação, quadrado, adição e inversão em $\mathbb{F}_{p^2}$, respectivamente; m_u, s_u, r denotam o custo de multiplicação e quadrado sem redução e redução modular, respectivamente; $\tilde{m}_u, \tilde{s}_u, \tilde{r}$ denotam o custo de multiplicação e quadrado sem redução e redução modu-

lar em $\mathbb{F}_{p^2}$, respectivamente. Para fins de simplificação da contagem de operações, os custos de subtração, negação e divisão por dois no corpo finito serão considerados equivalentes ao custo de uma adição. O cálculo de emparelhamentos sobre curvas elípticas pode ser feito usando qualquer sistema de coordenadas, mas as escolhas mais populares têm sido as coordenadas homogêneas projetivas e coordenadas afins, dependendo da razão entre a inversão e multiplicação. Coordenadas jacobianas foram inicialmente exploradas em algumas poucas implementações (NAHRIG; NIEDERHAGEN; SCHWABE, 2010; BEUCHAT et al., 2010), mas acabaram substituídas pelas coordenadas homogêneas por conta da sua eficiência superior (COSTELLO; LANGE; NAHRIG, 2010).

A duplicação de pontos e o cálculo das funções de linha correspondentes (linhas 5 e 7 do Algoritmo 1) normalmente dominam o custo no laço de Miller, já que os parâmetros eficientes tendem a minimizar o peso de Hamming do comprimento do laço de Miller ℓ e nos números resultantes das adições de pontos.

A seguir serão revistas e ligeiramente refinadas as melhores fórmulas disponíveis para aritmética de curvas elípticas envolvidas no cálculo de emparelhamentos, tanto em coordenadas afins quanto em coordenadas projetivas homogêneas.

4.1.2.1 Coordenadas Afins

A escolha de coordenadas afins se provou mais eficiente em níveis de segurança e graus de mergulho mais altos, por ter maneiras mais simples de se calcular inversões em extensões altas (SCOTT, 2011; LAUTER; MONTGOMERY; NAHRIG, 2010).

As principais vantagens das coordenadas afins são sua simplicidade de implementação e formato das funções lineares. Isto permite uma acumulação mais rápida dentro do laço de Miller caso a dispersão adicional seja explorada.

Se $T = (x_1, y_1)$ é um ponto em $E'(\mathbb{F}_{p^2})$, pode-se calcular o ponto $2T := (x_3, y_3) = T + T$, bem como a função de linha tangente no ponto $P = (x_P, y_P)$ com o isomorfismo

de *twist* ψ , conforme necessário na linha 5 do Algoritmo 1, com o custo (GREWAL et al., 2012):

$$3\tilde{m} + 2\tilde{s} + 7\tilde{a} + \tilde{t} + 2m.$$

Contudo, pré-calculando $y'_p = 1/y_p$ e $x'_p = \bar{x}_p/y_p$ e adotando a representação $\mathbb{F}_{p^{12}} = \mathbb{F}_{p^2}[w]/(w^6 - \xi)$, pode-se calcular a linha 5 do Algoritmo 1 da seguinte maneira:

$$\begin{aligned} A &\leftarrow \frac{1}{2y_1}, \quad B \leftarrow 3x_1^2, \quad C \leftarrow AB, \quad D \leftarrow 2x_1, \quad E \leftarrow Cx_1 - y_1, \\ x_3 &\leftarrow C^2 - D, \quad y_3 \leftarrow E - Cx_3, \quad F \leftarrow Cx'_p, \quad G \leftarrow Ey'_p, \end{aligned}$$

$$y'_p \cdot g_{2\psi(T)}(P) \leftarrow 1 + (F + Gw^2)w,$$

onde o fator y'_p não altera o resultado do emparelhamento, já que exponenciação final elimina qualquer fator de um subcorpo próprio de $\mathbb{F}_{p^{12}}$, e permite uma acumulação computacionalmente mais simples:

$$f^2 \cdot g_{2\psi(T)}(P) = (f_0 + f_1w)(1 + g_1w),$$

economizando um custo líquido de $4m$ em cada execução da linha 5 do Algoritmo 1, em relação a (GREWAL et al., 2012).

Similarmente, quando se consideram pontos diferentes $T = (x_1, y_1)$ e $Q = (x_2, y_2)$, o ponto $T + Q$ e a função de linha correspondente podem ser calculados com o custo (GREWAL et al., 2012): $3\tilde{m} + \tilde{s} + 6\tilde{a} + \tilde{t} + 4m$. Porém, aplicando o mesmo truque descrito acima, se consegue o mesmo *trade-off* no desempenho (ou seja, com uma economia líquida de $4m$ em cada execução da linha 7 do Algoritmo 1) com a seguinte fórmula:

$$A = \frac{1}{x_2 - x_1}, \quad B = y_2 - y_1, \quad C = AB, \quad D = x_1 + x_2, \quad E = Cx_1 - y_1,$$

$$x_3 = C^2 - D, \quad y_3 = E - Cx_3, \quad F = Cx'_p, \quad G = Ey'_p,$$

$$y'_p \cdot g_{\psi(T), \psi(Q)}(P) = 1 + (F + Gw^2)w.$$

É possível generalizar essa técnica para outras curvas amigáveis a emparelhamen-

tos com *twists* de grau d e com grau de mergulho k , proporcionando uma economia $(k/2 - k/d)m$ multiplicações por passo do algoritmo de Miller.

4.1.2.2 Coordenadas Projetivas Homogêneas

Técnicas análogas de pré-processamento são possíveis no caso de coordenadas projetivas. O custo da linha 5 do Algoritmo 1, segundo o estado da arte (GREWAL et al., 2012), é dado por:

$$2\tilde{m} + 7\tilde{s} + 21\tilde{a} + 4m + m_{b'}.$$

Com o pré-cálculo de $\bar{y}_P = -y_P$, $x'_P = 3x_P$, e usando uma forma especial dos coeficientes b e b' (PEREIRA et al., 2011), calcular a duplicação passa a ter o custo:

$$2\tilde{m} + 7\tilde{s} + 16\tilde{a} + 4m.$$

Vale ressaltar que, como indicado por Aranha et al. (ARANHA et al., 2011), numa plataforma x64 outras opções sugerem balanços ligeiramente diferentes, mas a diferença de desempenho, além de restrita a essas plataformas, é pequena.

Ganhos semelhantes aplicam-se ao cálculo da linha 7 do Algoritmo 1, resultando no custo:

$$11\tilde{m} + 2\tilde{s} + 8\tilde{a} + 4m.$$

4.1.3 Contagem de Operações

Fazendo uma contagem detalhada das operações relevantes para o cálculo do emparelhamento em uma curva BN, considerando todas as melhorias descritas anteriormente, o laço de Miller (linhas 4 a 9 do Algoritmo 1) contribui os seguintes custos no caso afim (MLA) e no caso projetivo (MLP):

$$\text{MLA} = 1089\tilde{m}_u + 890\tilde{s}_u + 1132\tilde{r} + 9484\tilde{a} + 70\tilde{i} + i + 283m + 3a,$$

$$\text{MLP} = 1339\tilde{m}_u + 1152\tilde{s}_u + 1388\tilde{r} + 11336\tilde{a} + 282m + 6a.$$

Para o cálculo da exponenciação final (linha 15 do Algoritmo 1), foi feita uma implementação da solução estado-da-arte da literatura de Fuentes *et al* (FUENTES-CASTAÑEDA; KNAPP; RODRÍGUEZ-HENRÍQUEZ, 2011), que por sua vez acumula outras técnicas presentes na literatura (SCOTT *et al.*, 2009; BEUCHAT *et al.*, 2010; ARANHA *et al.*, 2011; SCOTT; BARRETO, 2004; NAEHRIG; BARRETO; SCHWABE, 2008). A contagem de operações para o cálculo da exponenciação final é:

$$FE = 384\tilde{m}_u + 1172\tilde{s}_u + 941\tilde{r} + 8085\tilde{a} + 4\tilde{i} + 20m + 12a.$$

4.1.4 Resultados

O custo total para emparelhamento em coordenadas projetivas homogêneas pode ser expressa como:

$$\begin{aligned} \text{MLP} + FE &= 1723\tilde{m}_u + 2324\tilde{s}_u + 2329\tilde{r} + 19421\tilde{a} + 4\tilde{i} + i + 302m + 18a \\ &= 9817m_u + 4658r + 59598a + 4\tilde{i} + i + 302m + 18a \\ &= 10119m_u + 4960r + 59616a + 4\tilde{i} + i. \end{aligned}$$

Uma comparação direta com implementações recorde anteriores (ARANHA *et al.*, 2011), considerando apenas o número de multiplicações em $\mathbb{F}_p$ gerado pela aritmética em $\mathbb{F}_{p^2}$ como métrica de eficiência, mostra que este trabalho melhora a implementação de coordenadas projetivas com uma economia de 3% de multiplicações no corpo base. Isso é reflexo da exponenciação final mais rápida adotada de (FUENTES-CASTAÑEDA; KNAPP; RODRÍGUEZ-HENRÍQUEZ, 2011) e as formulas mais eficientes para inversão e quadrado em $\mathbb{F}_{p^{12}}$. Comparando o número total de multiplicações com implementações mais recentes (ZAVATTONI *et al.*, 2013; MITSUNARI, 2013), a implementação deste trabalho reduz o número de multiplicações em 2%.

Uma implementação de referência foi feita em linguagem C, com pontos críticos de eficiência implementados em *assembly*, usando o compilador GCC versão 4.7.0.

Tabela 1: Implementações em plataformas de 64 bits (tempos em quilociclos).

Operação	N	P II	SB	IB	H	H+mulx
Laço de Miller afim	1680	1529	1365	1315	1259	1212
Laço de Miller proj.	1170	862	856	798	721	704
Exponenciação final	745	557	572	537	492	473
Emp. afim	2426	1898	1937	1852	1751	1686
Emp. proj.	1915	1419	1428	1335	1213	1177

Essa implementação é agora parte integral da biblioteca RELIC (ARANHA; GOUVÊA, 2013). Os experimentos para medição de eficiência foram feitos em um conjunto de plataformas compatíveis com Intel de 64 bits: processadores mais antigos Nehalem Core i5 540M 2.53GHz e AMD Phenom II 3.0 GHz, e processadores mais modernos Sandy Bridge Xeon E31270 3.4GHz e Ivy Bridge Core i5 3570 3.4GHz, incluindo um processador mais recente Haswell Core i7 4750 HQ 2.0GHz. Todas as máquinas tiveram seu sistema de *overclock* automático desativado para aumentar a confiabilidade dos resultados.

A tabela 1 apresenta os tempos divididos em laço de Miller e exponenciação final e expressos em quilociclos de *clock*, e tomados como a média de 10^4 repetições de cada operação. As plataformas são Intel Nehalem (N) AMD Phenom II (P II), Sandy Bridge (SB), Ivy Bridge (IB), Haswell sem e com o uso da instrução *mulx* (H, H+mulx).

Os resultados de desempenho dependentes de plataforma (por oposição à mera contagem de operações) melhoram os resultados de (ARANHA et al., 2011) em 6% a 9% nas máquinas Nehalem e Phenom II, respectivamente. Comparando com uma versão mais recente (ZAVATTONI et al., 2013) da implementação recordista anterior (BEUCHAT et al., 2010), a implementação deste trabalho em Sandy Bridge é 82.000 ciclos mais rápida, ou 5%. Considerando os resultados no Haswell com a mesma implementação em *software* proposta em (MITSUNARI, 2013), foram obtidos tempos 8% melhores sem usar a instrução *mulx*, com 3% adicionais quando essa instrução é usada explicitamente em *assembly*.

Na implementação baseada em coordenadas afins, o estado da arte para no nível de segurança de 128 bits é o descrito no trabalho de Acar *et al.* (ACAR; LAUTER; NAEHRIG M. SHUMOW, 2013). Infelizmente, apenas a latência de 15,6 milhões de ciclos em um Core 2 Duo é fornecido para arquiteturas Intel 64-bit. Isso não permite uma comparação direta, tendo em vista que a pequena melhora de desempenho entre o Core 2 Duo e Nehalem relatado em (ARANHA et al., 2011) implica que a implementação afim deveria ser pelo menos 6 vezes mais rápida que a de Acar *et al.* quando executada numa mesma máquina.

Apesar de mais lenta que a versão projetiva apresentada aqui, a implementação com coordenadas afins ainda é consideravelmente mais rápida que alguns recordes de velocidades anteriores em coordenadas projetivas (HANKERSON; MENEZES; SCOTT, 2008; NAEHRIG; NIEDERHAGEN; SCHWABE, 2010; BEUCHAT et al., 2010). Isso indica a possibilidade de que emparelhamentos em coordenadas afins podem ser melhorados ainda mais, contrariando a ideia disseminada de que a representação afim é invariavelmente pior que uma abordagem projetiva.

4.2 Implementação Eficiente de Reticulados

Em geral, conforme discutido na seção 3.3, a criptografia baseada em reticulados pode ser considerada mais eficiente e simples de implementar do que primitivas mais convencionais como emparelhamentos. Por isso, acredita-se que os reticulados sejam ideais para dispositivos de capacidade restrita, como dispositivos embarcados ou portáteis com pouca RAM, processadores simples ou com restrições com relação ao consumo de energia.

No entanto, ainda existem desafios nessa vertente criptográfica que dificultam o uso prático dela no cenário descrito acima. Os principais são: o tamanho das chaves, assinaturas e textos cifrados; e a necessidade de uma amostragem de alta qualidade de

uma Gaussiana discreta.

O primeiro desafio pode ser abordado com a utilização de matrizes estruturadas que possuem uma representação compacta, como indicado na seção 3.5.3, e uma das contribuições é a proposta de uma nova família de matrizes tão eficientes neste aspecto quanto as encontradas na literatura, descrita mais adiante na seção 4.2.3.

O segundo é em geral, tratado com otimizações de implementação de técnicas e estratégias já conhecidas, como será visto na seção 4.2.2. Existem também maneiras eficientes de calcular as operações básicas em criptossistemas baseados em reticulados (i.e. multiplicação e inversão de matrizes). Isto é feito por meio da transformada rápida de Fourier, descrita abaixo, na seção 4.2.1.

4.2.1 Transformada Rápida de Fourier

A implementação eficiente de operações básicas em reticulados é geralmente baseada na otimização das operações de multiplicação de inversão de matrizes, pois estas são as principais operações encontradas em protocolos de criptografia baseada em reticulados. Esta otimização é, muitas vezes, feita por meio da transformada rápida de Fourier (*Fast Fourier Transform* ou FFT), chamada também de transformada da teoria dos números (*Number Theoretical Transform* ou NTT) quando definida sobre corpos finitos, uma já técnica consagrada e muito conhecida em álgebra linear. Implementações dedicadas a matrizes negacíclicas para fins criptográficos confirmam sua adequação até para plataformas com recursos limitados (GÜNEYSU; LYUBASHEVSKY; PÖPPELMANN, 2012).

A FFT é um algoritmo usado para calcular a transformada discreta de Fourier e sua inversa. No entanto, este algoritmo também pode ser usado para a diagonalização de matrizes, quando elas têm certa estrutura, o que permite o cálculo mais eficiente de multiplicação e inversão de matrizes. As estruturas que permitem a utilização da FFT mais conhecidas são justamente as matrizes que pertencem ao anel circulante e

negacíclico descritos na seção 3.5.3.

A FFT é de grande importância em uma vasta gama de aplicações, desde processamento digital de sinais para a resolução de equações diferenciais parciais até algoritmos para multiplicação de inteiros grandes. O algoritmo baseia-se no chamado método de dobramentos sucessivos, no qual se pode expressar a transformada de Fourier da seguinte forma:

$$\text{FFT}(x_0, \dots, x_{n-1}) = (y_0, \dots, y_{n-1}) \Leftrightarrow y_k = \sum_{j=0}^{n-1} x_j \omega_n^{kj}$$

onde $\omega_n = e^{-2\pi i/n}$ no caso contínuo (transformada de funções reais e/ou complexas), ou uma raiz n -ésima primitiva da unidade no caso da transformada sobre corpos finitos, ou seja, um elemento tal que $\omega_n^n = 1$ mas $\omega_n^k \neq 1$ para $0 < k < n$.

No contexto deste trabalho, a principal utilidade da FFT é no cálculo da operação de convolução. Essa operação permite a multiplicação de duas matrizes de maneira mais eficiente, desde que essas matrizes sejam simultaneamente diagonalizáveis.

A convolução para funções de domínio discreto é dada por:

$$(f * g)_k = h_k = \sum_{j=0}^k f_j \cdot g_{k-j}, \quad (4.1)$$

onde f e g são sequências de tamanho n e tal fórmula fornece o valor do k -ésimo elemento do vetor resultante. Ao observar a equação 4.1, verifica-se que a convolução discreta pode ser vista como um produto entre dois polinômios f e g de ordem $n - 1$. Por exemplo, se $f = f_0 + f_1x + f_2x^2 + \dots + f_{n-1}x^{n-1}$ e $g = g_0 + g_1x + g_2x^2 + \dots + g_{n-1}x^{n-1}$, o produto será $h = h_0 + h_1x + h_2x^2 + \dots + h_{n-1}x^{n-1}$, onde:

- $h_0 = f_0g_0$
- $h_1 = f_0g_1 + f_1g_0$
- $h_2 = f_0g_2 + f_1g_1 + f_2g_0$

e assim por diante.

A convolução pode ser efetuada aplicando a FFT a cada um dos operandos com custo $O(n \lg n)$, multiplicando as transformadas com custo $O(n)$, e em seguida aplicando a transformada inversa ao resultado com custo $O(n \lg n)$:

$$A * B = \text{FFT}^{-1}(\text{FFT}(A) \cdot \text{FFT}(B))$$

onde A e B são duas matrizes. Assim, a complexidade total é dominada pelo custo da própria FFT, ou seja, $O(n \lg n)$.

Cormen (CORMEN et al., 2009) apresenta uma descrição recursiva da FFT, e também uma versão iterativa. Será exposta aqui apenas a segunda, pois ela é mais eficiente e, assim, ideal para implementação em dispositivos com capacidade limitada. Primeiro, o algoritmo executa o procedimento BIT-REVERSE-COPY para reordenar o vetor de entrada numa ordem necessária para execução do algoritmo iterativo da FFT, descrito no algoritmo 2.

Algoritmo 2 BIT-REVERSE-COPY (CORMEN et al., 2009).

INPUT: a

OUTPUT: A , permutação de a em índices com bits revertidos.

```

1:  $n = \text{comprimento}[a]$ 
2: for  $k \leftarrow 0$  to  $n - 1$  do
3:    $A[\text{rev}(k)] \leftarrow a_k$ 
4: end for
```

No Algoritmo 2, onde $\text{rev}(k)$ é um inteiro de $\lg n$ bits formado pela inversão dos bits da representação binária de k . Assim o procedimento BIT-REVERSE-COPY insere o elemento a_k na posição $A[\text{rev}(k)]$.

Com os elementos do vetor de entrada já ordenados da maneira correta, o algoritmo iterativo para a FFT (algoritmo 3) pode ser usado.

A convolução feita por meio da FFT tem complexidade $O(n \lg n)$, muito eficiente em comparação a métodos de multiplicação de matrizes comuns, que em geral têm complexidade $O(n^3)$, ou no mínimo $O(n^2)$ no cálculo direto a partir da equação 4.1.

Algoritmo 3 ITERATIVE-FFT (CORMEN et al., 2009).

INPUT: a, ω_n

OUTPUT: $A = \text{FFT}(a)$

```

1: BIT-REVERSE-COPY( $a, A$ )
2:  $n = \text{comprimento}[a]$            //  $n$  é uma potência de 2.
3: for  $k \leftarrow 0$  to  $n$  do
4:    $m \leftarrow 2^s$ 
5:    $\omega_m \leftarrow \omega_n^{n/m}$ 
6:   for  $s \leftarrow 1$  to  $n - 1$  by  $m$  do
7:      $\omega \leftarrow 1$ 
8:     for  $j \leftarrow 0$  to  $m/2 - 1$  do
9:        $t \leftarrow \omega \cdot A[k + j + m/2]$ 
10:       $u \leftarrow A[k + j]$ 
11:       $A[k + j] \leftarrow u + t$ 
12:       $A[k + j + m/2] \leftarrow u - t$ 
13:       $\omega \leftarrow \omega \cdot \omega_m$ 
14:     end for
15:   end for
16: end for
17: return  $A$ 

```

A operação de multiplicação e inversão de matrizes, bem como a resolução de sistemas lineares, está muito presente em esquemas de criptografia baseada em reticulados. Consequentemente, é desejável que essas operações sejam as mais eficientes possíveis

Por essa razão, os parâmetros mais modernos em primitivas criptográficas baseadas em reticulados utilizam matrizes circulantes ou negacíclicas, pois elas são conhecidas como compatíveis com a FFT e com a convolução. Na seção 4.2.3, mais adiante, é apresentada uma nova alternativa de parametrização, de natureza diferente das usadas atualmente, mas ainda assim.

4.2.2 Amostragem de Vetores de Gaussianas Discretas

Um dos desafios da criptografia baseada em reticulados, é o fato de muitos esquemas necessitarem de amostragem de distribuições Gaussianas discretas em reticulados. Exemplos de esquemas que precisam desse tipo de amostragem são assinaturas com e sem alçaço (GENTRY; PEIKERT; VAIKUNTANATHAN, 2008; BOYEN, 2010) e encrip-

tação baseada em identidade (GENTRY; PEIKERT; VAIKUNTANATHAN, 2008; CASH et al., 2010). O problema é amostrar um vetor de uma distribuição Gaussiana discreta num reticulado $\mathcal{L}$ em $\mathbb{Z}^m$, tarefa que pode ser reduzida a fazer uma amostragem de m inteiros de uma distribuição Gaussiana discreta em $\mathbb{Z}$.

Apesar da amostragem de distribuições normais ser um problema recorrente em computação estatística, a implementação desse tipo de operação pode ser um inconveniente em plataformas com recursos limitados. Entre as dificuldades para a implementação desse tipo de amostragem estão a necessidade de aritmética com ponto flutuante de alta precisão ou a utilização de tabelas pré-calculadas muito grandes. Outro problema é que aplicações criptográficas exigem amostragem de alta qualidade, isto é, uma diferença estatística entre a distribuição desejada e a distribuição amostrada deve ser menor do que em aplicações de computação estatística comuns.

As principais abordagens encontradas na literatura para se fazer essas amostragens no contexto de aplicação criptográficas são: uma variante aleatória do algoritmo do plano mais próximo de Babai (BABAI, 1986)(como analisado por Gentry et al. (GENTRY; PEIKERT; VAIKUNTANATHAN, 2008)) proposta por Klein (KLEIN, 2000), e algoritmos baseados na convolução de chamados reticulados q -ários (PEIKERT, 2010; MICCIANCIO; PEIKERT, 2012). Em geral, as otimizações encontradas na literatura são implementações otimizadas, utilizando paralelismo nas operações ou fazendo pequenas alterações para deixar os algoritmos mais eficientes.

Outro método para obtenção da amostragem, não tão explorado ainda, é fazê-lo por meio do teorema central dos limites (TCL). De acordo com o TCL, é possível obter um vetor de n valores normalmente distribuídos através de n combinações lineares independentes de n valores igualmente distribuídos com média 0 e variância σ^2 . Formalmente:

Teorema 2 (TCL de Lindeberg-Lévy). *Seja $(X_1, X_2, \dots, X_n)$ um conjunto de n variáveis aleatórias amostradas de uma mesma distribuição arbitrária, com média μ e va-*

riância finita σ^2 , e seja

$$S_n := \frac{\sum_{i=1}^n X_i}{n}$$

a sua média. A distribuição amostral das variáveis aleatórias

$$\sqrt{n}(S_n - \mu)$$

aproxima-se de uma distribuição normal com média 0 e variância σ^2 à medida que n cresce, com um erro $O(1/n^2)$ em relação à normal verdadeira.

Ou seja, para fazer uma amostragem de distribuições Gaussianas discretas basta, a princípio, obter n valores iniciais com média 0 e variância $\sigma^2 = 1$ e tomar sua média. Isso pode ser feito a partir de uma distribuição uniforme no intervalo $[-\sqrt{3}\sigma, \sqrt{3}\sigma]$.

Um possível problema nessa estratégia seria o tamanho de n , caso o interesse fosse amostrar variáveis individualmente. Nesse caso, o custo de amostrar uma única variável utilizando o TCL é no mínimo n , que pode ser considerável dada a convergência apenas quadrática. No entanto, quando se necessita não apenas de uma, mas de n variáveis, como é o caso dos esquemas criptográficos, pode-se amortizar o custo total da amostragem entre todas as variáveis. Se for possível fazer n combinações lineares independentes de n variáveis, com pesos ± 1 . A maneira mais simples para atingir esse objetivo é aplicar a transformada rápida de Walsh-Hadamard (*Fast Walsh-Hadamard Transform* ou FWHT), cujo custo total é $O(n \lg n)$. Amortizando esse custo entre as n variáveis resultantes, obtém-se um custo efetivo de apenas $O(\lg n)$ por variável, que é a complexidade ótima (GALBRAITH; DWARAKANATHM, 2012).

É importante notar que o método descrito acima método já apareceu na literatura (WALLACE, 1996) em situações onde o valor de n é independente da aplicação subjacente, e refere-se apenas a uma pré-tabulação de variáveis normalmente distribuídas, mantidas num pequeno *buffer* para uso posterior sob demanda. A maioria dos esquemas criptográficos modernos baseados em reticulados, porém, depende natural-

mente de um número n considerável de variáveis normalmente distribuídas. Assim, uma contribuição da presente dissertação é atrelar o método de amostragem baseado em FWHT ao criptossistema subjacente, dispensando o tratamento de *buffer* e amortizando naturalmente o custo da amostragem.

Cumprе notar que, se a amostragem de uma distribuição uniforme no intervalo $[-\sqrt{3}\sigma, \sqrt{3}\sigma]$ acoplada à FWHT não atingir a precisão desejada da distribuição normal, pode-se recorrer a alguma outra uma distribuição mais próxima da normal, e em seguida aprimorar essa amostragem com o TCL e a FWHT.

A FWHT é um algoritmo eficiente para calcular a transformada de Walsh-Hadamard, e tem muitas semelhanças com a FFT (descrita na seção 4.2.1). O Algoritmo 4 fornece uma descrição do algoritmo da FWHT:

Algoritmo 4 Transformada Rápida de Walsh-Hadamard

INPUT: $k \in \mathbb{N}$, $\mathbf{u} \in \mathbb{K}^r$ with $r = 2^k$ and $\text{char}(\mathbb{K}) \neq 2$.

OUTPUT: $\mathbf{uH}_k$

```

1:  $d \leftarrow 1$ 
2: for  $s \leftarrow 1$  to  $k$  do
3:    $h \leftarrow d, d \leftarrow 2d$ 
4:   for  $i \leftarrow 0$  to  $r - 1$  by  $d$  do
5:     for  $j \leftarrow 0$  to  $h - 1$  do
6:        $v \leftarrow u_{i+j}, w \leftarrow u_{i+j+h}, u_{i+j} \leftarrow v + w, u_{i+j+h} \leftarrow v - w$ 
7:     end for
8:   end for
9: end for
10: return  $\mathbf{u}$ 

```

Com este algoritmo é possível acelerar o cálculo da transformação linear necessária para obter a amostragem da distribuição Gaussiana.

Vale ressaltar que, aplicando a transformada inversa FWHT^{-1} , o resultado voltaria a ser uniformemente distribuído (ou distribuído segundo uma distribuição inicial que aproxima a normal com menor qualidade). Se um criptossistema permitir esse procedimento, um agressor poderia violar as premissas de segurança, pois provas de segurança que necessitam de amostras normalmente distribuídas não seriam mais válidas. Este

problema pode ser resolvido começando de um vetor com mais de n elementos, e descartando o excesso para impossibilitar a inversão da FWHT, uma vez que essa parte descartada não seria mais conhecida. Nesse caso, o tamanho do vetor deve ser dimensionado de maneira a não possibilitar uma busca exaustiva (a solução mais simples e robusta é simplesmente dobrar o valor de n , uma vez que a FWHT exige que n seja uma potência de 2).

Outras vantagens deste método são sua simplicidade em comparação a outras estratégias de amostragem e o tempo de amostragem independente dos dados de entrada, o que faz com que amostragens feitas com esse método sejam resistentes contra ataques de canal secundário (*Side Channel Attack*).

Um ponto ainda em aberto é uma análise formal de o quão boa é uma amostragem feita com essa estratégia. Como já dito, esquemas criptográficos exigem uma amostragem de alta qualidade, ou seja, com uma diferença mínima da ordem de 2^{-100} (GALBRAITH; DWARAKANATHM, 2012) de uma amostragem de distribuição Gaussiana verdadeira.

4.2.3 Álgebra de Rojo

A chamada álgebra de Rojo (ROJO, 2008) é formada pelo conjunto das matrizes da forma:

$$A = T + H$$

onde, T é uma matriz de Toeplitz simétrica e H é uma matriz de Hankel centrossimétrica, originalmente ambas com elementos reais.

Uma matriz de Toeplitz é uma matriz de tamanho $n \times n$ em que cada diagonal descendente da esquerda para a direita possui valor constante. Em outras palavras é

uma matriz da forma:

$$T = \begin{bmatrix} t_0 & t_1 & \dots & t_{n+2} & t_{n+1} \\ t_{-1} & \ddots & \ddots & & t_{n+2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ t_{-(n+2)} & & \ddots & \ddots & t_1 \\ t_{-(n+1)} & t_{-(n+2)} & \dots & t_{-1} & t_0 \end{bmatrix}$$

A estrutura característica da matriz de Toeplitz pode ser melhor visualizada na figura 3, que apresenta uma representação visual da mesma, indicando valores idênticos por padrões idênticos.

Figura 3: Matriz de Toeplitz

A matriz de Toeplitz simétrica usada na álgebra de Rojo é uma matriz de Toeplitz em que os elementos do lado direito da diagonal principal são iguais aos elementos do lado esquerdo, ou seja:

$$T = \begin{bmatrix} t_0 & t_1 & \dots & t_{n+2} & t_{n+1} \\ t_1 & \ddots & \ddots & & t_{n+2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ t_{n+2} & & \ddots & \ddots & t_1 \\ t_{n+1} & t_{n+2} & \dots & t_1 & t_0 \end{bmatrix}$$

A figura 4 ilustra melhor a estrutura de uma matriz de Toeplitz simétrica

Qualquer matriz de Toeplitz simétrica T pode ser completamente definida pela

Figura 4: Matriz de Toeplitz Simétrica

primeira linha:

$$t = [t_0 \ t_1 \ \dots \ t_{n-2} \ t_{n-1}].$$

Uma matriz de Hankel é semelhante à matriz de Toeplitz, a diferença sendo que nas matrizes de Hankel as anti-diagonais (diagonais da direita para esquerda) são constantes. Assim, uma matriz de Hankel H é uma matriz da forma:

$$H = \begin{bmatrix} h_0 & h_1 & \dots & h_{n+2} & h_{n+1} \\ h_1 & & \ddots & \ddots & h_{-(n+2)} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ h_{n+2} & \ddots & \ddots & & h_{-1} \\ h_{n+1} & h_{-(n+2)} & \dots & h_{-1} & h_n \end{bmatrix}$$

A estrutura da matriz de Hankel pode ser melhor visualizada na figura 5

Figura 5: Matriz de Hankel

A matriz de Hankel centrossimétrica que aparece na álgebra de Rojo é da forma:

$$H = \begin{bmatrix} h_0 & h_1 & \dots & h_{n+2} & h_{n+1} \\ h_1 & & \ddots & \ddots & h_{n+2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ h_{n+2} & \ddots & \ddots & & h_1 \\ h_{n+1} & h_{n+2} & \dots & h_1 & h_0 \end{bmatrix}$$

A estrutura pode ser melhor visualizada na figura 6

Figura 6: Matriz de Hankel Centrossimétrica

Assim como a matriz de Toeplitz, ela é completamente representada apenas pela primeira linha:

$$h = [h_0 \ h_1 \ \dots \ h_{n-2} \ h_{n-1}].$$

As matrizes pertencentes à álgebra de Rojo são obtidas a partir da soma de uma matriz de Toeplitz simétrica T com uma matriz de Hankel centrossimétrica H , e essas duas matrizes devem se relacionar da seguinte maneira:

$$T = \text{Toeplitz}([t_0 \ t_1 \ \dots \ t_{n-2} \ t_{n-1}]) \quad (4.2)$$

$$H = \text{Hankel}([t_1 \ t_2 \ \dots \ t_{n-1} \ t_n]) \quad (4.3)$$

onde

$$[t_0 \ t_1 \ \dots \ t_{n-2} \ t_{n-1}]$$

é a primeira linha de T e

$$[t_1 \ t_2 \ \dots \ t_{n-1} \ t_n]$$

é a primeira linha de H . Nota-se que as matrizes pertencentes à álgebra de Rojo podem ser completamente representadas pelas primeiras linhas mais um elemento. Esta propriedade as torna tão compactas quanto matrizes circulantes, muito utilizadas atualmente em criptografia baseada em reticulados (STEHLÉ; STEINFELD, 2011; LYUBASHEVSKY; PEIKERT; REGEV, 2010).

A proposta original da álgebra de Rojo (ROJO, 2008) é contínua e infinita, ou seja, as matrizes possuem elementos definidos sobre os números reais, $\mathbb{R}$. Rojo também prova que o conjunto das matrizes dessa álgebra é simultaneamente diagonalizável. Em outras palavras, seja

$$\mathcal{A} = \{A : A = T + H\}$$

o conjunto das matrizes de Rojo, onde T e H são matrizes de Toeplitz e Hankel conforme descrito anteriormente. . Prova-se (ROJO, 2008) que qualquer matriz desse conjunto é simultaneamente diagonalizável.

Outro ponto a ser ressaltado é que a decomposição das matrizes $A \in \mathcal{A}$ não é única. Por exemplo:

$$A = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 2 & 3 \\ 3 & 2 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{bmatrix} + \begin{bmatrix} 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 4 \\ 4 & 5 & 4 & 3 \\ 5 & 4 & 3 & 2 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 \end{bmatrix} + \begin{bmatrix} 3 & 4 & 5 & 6 \\ 4 & 5 & 6 & 5 \\ 5 & 6 & 5 & 4 \\ 6 & 5 & 4 & 3 \end{bmatrix}$$

A primeira linha de A é dada por:

$$[t_0 + t_1 \quad t_1 + t_2 \quad t_2 + t_3 \quad \dots \quad t_{n-2} + t_{n-1} \quad t_{n-1} + t_n]$$

Logo a equação

$$t_j = a_{1,j} - t_{j-1}, \quad j = 1, 2, \dots, n$$

determina uma decomposição $T + H$ de A assumindo que t_0 é dado.

A prova de Rojo se baseia numa representação alternativa da álgebra, materializada na forma de matrizes ciclo-simétricas de tamanho $2n \times 2n$ definida sobre os reais e dada por:

$$C = \text{circul}(c)$$

onde c é o vetor:

$$c = [t_0 \ t_1 \ \dots \ t_{n-2} \ t_{n-1} \ t_n \ t_{n-1} \ t_{n-2} \ \dots \ t_2 \ t_1],$$

definido a partir da concatenação da primeira linha de T com a primeira linha de H escrita ao contrário. De fato, a principal contribuição de Rojo em seu trabalho é justamente a prova do seguinte lema:

Lema 1. *Se $A \in \mathcal{A}$ com autovalores $\lambda_1, \lambda_2, \dots, \lambda_n$ então existe um matriz ciclo-simétrica C com autovalores $\lambda_1, \lambda_2, \dots, \lambda_n$ cada um com multiplicidade 2*

Graças a esse lema é sempre possível utilizar a representação ciclo-simétrica da álgebra de Rojo para acelerar certas operações. Isso é possível, porque a matriz ciclo-simétrica resultante é compatível com a Transformada Discreta de Cossenos (*Discrete Cosine Transform* ou DCT). Destaca-se que existe uma representação complementar ($A = T - H$) que mantém as mesmas propriedades, caso em que é usada a Transformada Discreta de Senos (*Discrete Sine Transform* ou DST). Tanto a DCT quando a DST são operações semelhantes à FFT, sendo necessário apenas pequenas adaptações.

Neste trabalho foi analisada uma variante discreta e finita dessas álgebras, ou seja, as matrizes possuem elementos definidos sobre um corpo finito $\mathbb{F}_q$. A partir dessa análise foi verificado que a álgebra de Rojo discreta mantém as mesmas propriedades da original no que diz respeito à diagonalização simultânea. Isto faz da álgebra de Rojo

uma excelente alternativa às parametrizações já existentes em criptografia baseada em reticulados, pois é possível aplicar as mesmas técnicas de otimização utilizadas em matrizes circulantes ou negacíclicas.

Foi verificado ainda que a álgebra de Rojo discreta é compatível também com a FFT, o que torna essa nova parametrização compatível com o que já existe na literatura, descartando a necessidade de novos arcabouços de algoritmos especialmente feitos pra esse tipo de álgebra.

Outra característica própria da álgebra de Rojo é o fato dela não possuir uma representação polinomial trivial, diferente dos anéis algébricos encontrados na literatura. Isso é importante, pois comprova sua natureza diferente e deixa claro que ela é uma boa alternativa para os casos de matrizes cíclicas no advento delas se mostrarem vulneráveis a algum ataque que se aproveite da representação polinomial, além de evitar patentes que recaem sobre anéis cíclicos e negacíclicos, que possuem representação polinomial.

Finalmente, cabe notar que uma alternativa mais natural às matrizes circulantes seriam as chamadas matrizes diádicas, pois essas já são usados em códigos corretores de erros (BARRETO et al., 2011; CAYREL; HOFFMANN; PERSICHETTI, 2012; HEYSE, 2011; MISOCZKI; BARRETO, 2009). No entanto, essas matrizes possuem uma desvantagem com relação às matrizes de Rojo: elas são diagonalizáveis com a FWHT, o que não permitiria a utilização da estratégia para amostragem apresentada na seção 4.2.2. A FWHT, que só difere de sua própria inversa por um fator constante, diagonalizaria os blocos diádicos da base do reticulado ao mesmo tempo que destruiria a normalidade do ruído introduzido, levando a um sistema muito mais fraco, provavelmente quebrável em tempo polinomial. Desta maneira, as matrizes da álgebra de Rojo ainda se mostram mais vantajosas por possuírem essa opção de otimização, e possuírem uma representação compacta com possibilidade de otimização das operações básicas com a FFT, como indicado anteriormente nesta seção.

4.2.4 Resultados

Para aferir o efeito do método de amostragem via FWHT e da adoção de álgebras discretas de Rojo, implementaram-se em linguagem C as seguintes primitivas:

1. Amostragem pelo método da razão (KINDERMAN; MONAHAN, 1977), pelo método polar (MARSAGLIA; BRAY, 1964) e pelo método FWHT modificado conforme proposto neste trabalho;
2. Aritmética em anéis ideais negacíclicos $\mathbb{Z}[x]/(x^n + 1)$ e em álgebras discretas de Rojo;
3. Estimativa do nível de segurança para parâmetros dados, conforme a heurística Lindner-Peikert (LINDNER; PEIKERT, 2011);
4. Criptossistema Lindner-Peikert (LINDNER; PEIKERT, 2011), considerado na literatura um dos mais eficientes para cifração com reticulados;
5. Benchmarks para todas as combinações de método de amostragem e criptossistemas para níveis realísticos de segurança.

As propriedades dos esquemas implementados estão listadas na tabela 2 (tamanhos medidos em bits).

Os resultados obtidos são sumarizados na tabela 3 e nas figuras 7, 8 e 9

Tabela 2: Propriedades de criptossistemas baseados em reticulados

sistema	n	segurança	$ pk $	$ ct $
LP	512	2^{97}	15380	30760
LP	1024	2^{247}	33273	66546

Os resultados obtidos (Windows 7, Intel i5 2.5 GHz, compilador Code::Blocks v. 12.11) mostram que, para um dado método de amostragem de variáveis normais, o desempenho típico de criptossistemas sobre álgebras discretas de Rojo é, conforme o

Tabela 3: Desempenho de criptossistemas baseados em reticulados (tempos em μs)

sistema	n	álgebra	amostragem	GeraChaves	Encripta	Decrypta
LP	512	negacíclica	razão	174.6	256.5	25.5
LP	512	negacíclica	polar	124.4	216.1	26.1
LP	512	negacíclica	FWHT	71.6	132.8	25.6
LP	512	Rojo	razão	208.8	341.9	46.3
LP	512	Rojo	polar	171.2	305.5	46.6
LP	512	Rojo	FWHT	118.4	222.6	46.4
LP	1024	negacíclica	razão	354.0	571.5	68.1
LP	1024	negacíclica	polar	281.2	475.8	65.9
LP	1024	negacíclica	FWHT	174.5	324.1	64.4
LP	1024	Rojo	razão	468.5	783.0	119.1
LP	1024	Rojo	polar	387.1	714.8	120.3
LP	1024	Rojo	FWHT	288.7	546.0	118.0

Figura 7: geração de chaves

Figura 8: Encriptação

Figura 9: Decriptação

método adotado de amostragem normal, entre 20% e 65% mais lento para geração de chaves, entre 35% a 70% mais lento para encriptação, e entre 75% e 85% mais lento para decriptação que esses mesmos criptossistemas sobre anéis negacíclicos.

Esse comportamento não surpreende: o método adotado aqui para a implementação de aritmética de Rojo recorre diretamente à FFT sobre uma álgebra *duas vezes maior* em vez de procurar aplicar, por exemplo, a transformada discreta de cossenos ou DCT (os tempos não chegam a dobrar porque, em contraste com o caso negacíclico, as álgebras de Rojo na abordagem seguida não incorrem pré- e pós-processamento).

Note-se, contudo, que as técnicas de implementação eficiente da DCT, ao contrário da FFT, não se transportam facilmente para o caso discreto. Obter uma versão da DCT que prescindia da duplicação implícita ou explícita de tamanho é deixado aqui como problema de pesquisa.

Vale ressaltar que os parâmetros aqui adotados diferem de outros propostos na literatura. Deve-se isto ao fato de que aqueles parâmetros, embora mais compactos, causam *por projeto* (para minimizar q , e com isso reduzir os tamanhos das chaves) erros de decriptação, o que na prática é inadmissível. Com os parâmetros aqui considerados, a probabilidade de erros de decriptação pode ser mantida arbitrariamente baixa. Especificamente, adota-se aqui $q \approx 6(n \lg n)^2$ tal que $q \equiv 1 \pmod{2n}$ seja primo, além de $\sigma = 2\sqrt{n}$ para manter a equivalência entre o caso médio e o pior caso entre certos reticulados relacionados, e $r = \sqrt{2n/\pi}$ para resistir a ataques de redução de reticulados (LINDNER; PEIKERT, 2011).

A estimativa do nível concreto de segurança k em bits procede conforme o seguinte roteiro semi-empírico (LINDNER; PEIKERT, 2011): Sejam $c_\varepsilon := (q/(\pi r)) \sqrt{\ln(1/\varepsilon)/2}$, $\lg(\delta_\varepsilon) := \lg(c_\varepsilon)^2/(4n \lg(q))$, $t_\varepsilon := 2^{1.8/\lg(\delta_\varepsilon)-110}$, $T := \min\{t_\varepsilon/\varepsilon \mid 0 < \varepsilon < 1\}$. O nível de segurança de um esquema baseado em reticulados, supondo que o problema computacional subjacente é o de distinguir a amostragem efetiva (com alçapão) de uma amostragem verdadeiramente aleatória é $k \approx \lg(T) + \lg(2.3 \times 10^9) - \lg(500)$.

5 DISCUSSÕES E CONCLUSÕES

Para fins de uma comparação breve e abrangente de emparelhamentos e reticulados, procede-se aqui a uma comparação sinóptica dessas primitivas em termos de flexibilidade, eficiência e segurança. As conclusões gerais do presente trabalho seguem dessa comparação.

5.1 Comparação Sinóptica

5.1.1 Flexibilidade

Conforme foi visto, as primitivas abordadas neste trabalho são extremamente flexíveis, ocorrendo em uma série de aplicações e protocolos. Ambas as primitivas contemplam tanto aplicações elementares de criptografia de chave pública (criptação e assinaturas digitais comuns) quanto esquemas mais avançados.

No primeiro aspecto (aplicações elementares), os emparelhamentos e os reticulados podem ser considerados igualmente flexíveis, pois as aplicações e protocolos que podem ser feitos em uma primitiva também podem ser feitos com a outra.

No segundo aspecto, embora existam protocolos implementáveis com ambas as primitivas (por exemplo, IBE e cifrassinaturas), outros protocolos são implementáveis a contento somente com uma das primitivas (por exemplo, assinaturas cegas com curvas elípticas e emparelhamentos, ou criptação homomórfica com reticulados). Com isso, embora incomensuráveis, ainda se podem considerar as duas primitivas como

“equivalentes” em termos de flexibilidade.

No entanto, os protocolos avançados baseados em emparelhamentos são diretamente utilizáveis em aplicações práticas, enquanto que muitos dos protocolos avançados em reticulados são apenas provas de conceito na literatura existente, não sendo remotamente eficientes para serem usados na prática.

5.1.2 Eficiência

Sob o ponto de vista da eficiência, verificou-se que os emparelhamentos admitem uma gama de otimizações muito maior do que a criptografia baseada em reticulados, abrangendo as mais diversas plataformas computacionais. Nesse sentido, há uma grande quantidade de trabalhos recentes sobre emparelhamentos que se dedicam à implementação otimizada para uma plataforma específica. Essa mudança de foco é uma evidência de que otimizações e melhorias teóricas mais gerais já são mais difíceis de se encontrar, ou seja, que a complexidade teórica do cálculo de emparelhamentos é um assunto bem compreendido.

Em contrapartida, nos trabalhos sobre reticulados ainda são encontradas otimizações mais gerais ou assintóticas na primitiva. Em geral, essas parametrizações visam a reduzir o tamanho das chaves e o *overhead* em mensagens encriptadas e assinaturas digitais, e reduzir a complexidade teórica de processamento.

A criptografia baseada em emparelhamentos é, portanto, substancialmente mais madura tecnologicamente do que as primitivas baseadas em reticulados. Por outro lado, essa menor maturidade dos reticulados (além da intensa pesquisa e interesse crescente) sugere oportunidades inexploradas de otimização que aparentemente já foram esgotadas com emparelhamentos.

5.1.3 Segurança

As diferenças entre emparelhamentos e reticulados continuam no aspecto de segurança. Os emparelhamentos são baseados em problemas tradicionais, bem explorados e bem estabelecidos em criptografia. Ataques bem sucedidos mais recentes não afetam os parâmetros mais modernos adotados em criptossistemas baseados em emparelhamentos, desde que a escolha dos parâmetros seja adequada, devendo-se dar atenção desde a escolha dos corpos finitos até a escolha de curvas adequadas.

Por outro lado, os reticulados chamam atenção por haver reduções de segurança do caso médio em certos reticulados para o pior caso em reticulados relacionados. Essa propriedade, se for bem aplicada, oferece parâmetros em que essencialmente qualquer escolha aleatória de chaves alcança o nível desejado de segurança.

5.1.4 Sumário

Considerando os aspectos de flexibilidade, eficiência e segurança, pode-se afirmar que a sentença “os reticulados são os novos emparelhamentos” faz sentido. No entanto, os reticulados encontram-se talvez no mesmo nível de investigação que os emparelhamentos em seus primórdios, oferecendo grande potencial de evolução e exploração.

5.2 Conclusões

Este trabalho confrontou duas primitivas criptográficas diferentes. A criptografia baseada em emparelhamentos e a criptografia baseada em reticulados. Esse confronto se deu tanto nos aspectos de flexibilidade e versatilidade para a construção de protocolos, quanto nos últimos avanços e o estado da arte da implementação eficiente.

Mais especificamente, propôs-se uma nova álgebra, igualmente eficiente na métrica de tamanho de chaves, mas livre de patentes, para representar reticulados, além de técnicas efetivas para a amostragem de variáveis normais em reticulados, opera-

ção presente em diversos protocolos. Como bônus e fruto de pesquisa realizada em parceria com Diego F. Aranha (UnB) e Patrick Longa (MSR), obteve-se uma implementação recordista de desempenho em coordenadas afins, área ainda pouco explorada na literatura (seção 4.1). Este trabalho foi publicado na conferência *Selected Areas in Cryptography* (SAC 2013)(ARANHA; BARRETO; LONGA P. AND RICARDN, 2013).

A álgebra discreta de Rojo aqui proposta (seção 4.2.3) é tão eficiente quanto as encontradas na literatura na métrica de tamanho de chaves. Por outro lado, ela mostrou-se menos eficiente na métrica de tempo de processamento, pois o método adotado aqui para a implementação de aritmética de Rojo recorre diretamente à FFT sobre uma álgebra *duas vezes maior* em vez de procurar aplicar, por exemplo, a transformada discreta de cossenos ou DCT. Contudo, os tempos de execução ainda são aceitáveis, e podem representar um custo-benefício razoável em situações onde as técnicas mais eficientes estiverem cobertas por patentes.

Na seção 4.2.2 foi apresentada uma estratégia com a qual é possível obter uma distribuição normal de maneira tal que o custo total seja amortizado utilizando a FWHT. Fica como sugestão para pesquisas futuras questões como: qual distribuição seria ideal para aplicar a estratégia aqui apresentada em determinado cenário; e o quão mais eficiente que outras encontradas na literatura ela pode ser, quando se impões que a amostragem não pode vaziar informações secretas.

Finalmente, em decorrência da pesquisa relatada nesta dissertação, pode-se elencar como oportunidades de extensões e pesquisas futuras: a procura de mais álgebras alternativas às utilizadas atualmente e à álgebra discreta de Rojo aqui sugerida; uma análise formal da qualidade da amostragem normal com a FWHT; técnicas de implementação dessa amostragem para obter desempenho independente dos dados amostrados (importante para evitar ataques de canal secundário); e uma versão da DCT que dispense a duplicação implícita ou explícita do tamanho dos elementos da álgebra durante o cálculo da transformada.

REFERÊNCIAS

- ACAR, T.; LAUTER, K.; NAEHRIG M. SHUMOW, D. Affine pairings on ARM. In: ABDALLA, M.; LANGE, T. (Ed.). *Pairing-Based Cryptography – Pairing 2012*. Cologne, Germany: Springer, 2013. (Lecture Notes in Computer Science, v. 7708), p. 203–209.
- AHARONOV, D.; REGEV, O. Lattice problems in $NP \cap coNP$. In: *Foundations of Computer Science, 45th Annual IEEE Symposium on*. Washington DC, USA: IEEE Computer Society, 2004. p. 362–371. ISSN 0272-5428.
- AJTAI, M. Generating hard instances of lattice problems. In: *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*. New York, NY, USA: ACM, 1996. (STOC '96), p. 99–108. ISBN 0-89791-785-5. Disponível em: <<http://doi.acm.org/10.1145/237814.237838>>.
- _____. The shortest vector problem in L_2 is NP-hard for randomized reductions (extended abstract). In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing*. New York, NY, USA: ACM, 1998. (STOC'98), p. 10–19. ISBN 0-89791-962-9.
- ARANHA, D. F.; BARRETO, P. S. L. M.; LONGA P. AND RICARDN, J. E. The realm of the pairings. In: *Selected Areas in Cryptography – SAC 2013*. Vancouver, Canada: Springer, 2013. (Lecture Notes in Computer Science). To appear.
- ARANHA, D. F.; FUENTES-CASTAÑEDA, L.; KNAPP, E.; MENEZES, A.; RODRÍGUEZ-HENRÍQUEZ, F. Implementing pairings at the 192-bit security level. In: ABDALLA, M.; LANGE, T. (Ed.). *Pairing-Based Cryptography – Pairing 2012*. Cologne, Germany: Springer, 2013. (Lecture Notes in Computer Science, v. 7708), p. 117–195.
- ARANHA, D. F.; GOUVÊA, C. P. L. *RELIC is an Efficient Library for Cryptography*. 2013. <http://code.google.com/p/relic-toolkit/>.
- ARANHA, D. F.; KARABINA, K.; LONGA, P.; GEBOTYS, C. H.; LÓPEZ, J. Faster explicit formulas for computing pairings over ordinary curves. In: *Advances in Cryptology – Eurocrypt 2011*. Tallinn, Estonia: Springer, 2011. (Lecture Notes in Computer Science, v. 6632), p. 48–68.
- BABAI, L. On Lovász's lattice reduction and the nearest lattice point problem. *Combinatorica*, Springer-Verlag, v. 6, n. 1, p. 1–13, 1986. ISSN 0209-9683.
- BALFANZ, D.; DURFEE, G.; SHANKAR, N.; SMETTERS, D. K.; STADDON, J.; WONG, H. C. Secret handshakes from pairing-based key agreements. In: *IEEE*

Symposium on Security and Privacy – S&P 2003. Berkeley, USA: IEEE Computer Society, 2003. p. 180–196.

BARBULESCU, R.; GAUDRY, P.; JOUX, A.; THOMÉ, E. *A quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic*. 2013. Cryptology ePrint Archive, Report 2013/400. <http://eprint.iacr.org/2013/400>.

BARRETO, P.; CAYREL, P.-L.; MISOCZKI, R.; NIEBUHR, R. Quasi-dyadic CFS signatures. In: LAI, X.; YUNG, M.; LIN, D. (Ed.). *Information Security and Cryptology*. Shanghai, China: Springer Berlin Heidelberg, 2011, (Lecture Notes in Computer Science, v. 6584). p. 336–349. ISBN 978-3-642-21517-9.

BARRETO, P. S. L. M.; GALBRAITH, S. D.; HÉIGEARTAIGH, C. O.; SCOTT, M. Efficient pairing computation on supersingular abelian varieties. *Designs, Codes and Cryptography*, Kluwer Academic Publishers-Plenum Publishers, v. 42, n. 3, p. 239–271, 2007. ISSN 0925-1022.

BARRETO, P. S. L. M.; KIM, H. Y.; LYNN, B.; SCOTT, M. Efficient algorithms for pairing-based cryptosystems. In: *Advances in Cryptology – Crypto 2002*. Santa Barbara, USA: Springer, 2002. (Lecture Notes in Computer Science, v. 2442), p. 377–387.

BARRETO, P. S. L. M.; LIBERT, B.; MCCULLAGH, N.; QUISQUATER, J.-J. Efficient and provably-secure identity-based signatures and signcryption from bilinear maps. In: ROY, B. (Ed.). *Advances in Cryptology – Asiacrypt 2005*. Chennai, India: Springer, 2005, (Lecture Notes in Computer Science, v. 3788). p. 515–532. ISBN 978-3-540-30684-9.

BARRETO, P. S. L. M.; LYNN, B.; SCOTT, M. Constructing elliptic curves with prescribed embedding degrees. In: *Security in Communication Networks – SCN 2002*. Amalfi, Italy: Springer, 2002. (Lecture Notes in Computer Science, v. 2576), p. 263–273.

_____. On the selection of pairing-friendly groups. In: *Selected Areas in Cryptography – SAC 2003*. Ottawa, Canada: Springer, 2004. (Lecture Notes in Computer Science), p. 17–25.

BARRETO, P. S. L. M.; NAEHRIG, M. Pairing-friendly elliptic curves of prime order. In: PRENEEL, B.; TAVARES, S. (Ed.). *Selected Areas in Cryptography – SAC 2005*. Waterloo, Ontario, Canada: Springer, 2006, (Lecture Notes in Computer Science, v. 3897). p. 319–331. ISBN 978-3-540-33108-7.

BENGER, N.; SCOTT, M. Constructing tower extensions of finite fields for implementation of pairing-based cryptography. In: HASAN, M. A.; HELLESETH, T. (Ed.). *Arithmetic of Finite Fields – WAIFI 2010*. Istanbul, Turkey: Springer, 2010. (Lecture Notes in Computer Science, v. 6087), p. 180–195.

BENSON, K.; SHACHAM, H.; WATERS, B. The k -BDH assumption family: Bilinear map cryptography from progressively weaker assumptions. In: DAWSON, E.

(Ed.). *Topics in Cryptology – CT-RSA 2013*. California, USA: Springer, 2013, (Lecture Notes in Computer Science, v. 7779). p. 310–325. ISBN 978-3-642-36094-7.

BERNSTEIN, D. J.; BUCHMANN, J.; DAHMEN, E. *Post-Quantum Cryptography*. Heidelberg, Deutschland: Springer, 2008.

BEUCHAT, J.-L.; DÍAZ, J. E. G.; MITSUNARI, S.; OKAMOTO, E.; RODRÍGUEZ-HENRÍQUEZ, F.; TERUYA, T. High-speed software implementation of the optimal ate pairing over Barreto-Naehrig curves. In: *Pairing-Based Cryptography – Pairing 2010*. Yamanaka Onsen, Japan: Springer, 2010. (Lecture Notes in Computer Science, v. 6487), p. 21–39.

BLAKE, I.; SEROUSSI, G.; SMART, N. *Elliptic Curves in Cryptography*. New York, USA: Cambridge University Press, 1999. (Lecture note series). ISBN 9780521653749.

BONEH, D.; BOYEN, X. Short signatures without random oracles. In: CACHIN, C.; CAMENISCH, J. L. (Ed.). *Advances in Cryptology – Eurocrypt 2004*. Interlaken, Switzerland: Springer, 2004. (Lecture Notes in Computer Science, v. 3027), p. 56–73.

BONEH, D.; FRANKLIN, M. Identity-based encryption from the Weil pairing. In: *Advances in Cryptology – Crypto 2001*. Santa Barbara, USA: Springer, 2001. (Lecture Notes in Computer Science, v. 2139), p. 213–229.

_____. Identity-based encryption from the Weil pairing. *SIAM Journal of Computing*, v. 32, n. 3, p. 586–615, 2003.

BONEH, D.; GENTRY, C.; LYNN, B.; SHACHAM, H. Aggregate and verifiably encrypted signatures from bilinear maps. In: *Advances in Cryptology – Eurocrypt 2003*. Warsaw, Poland: Springer, 2003. (Lecture Notes in Computer Science), p. 416–432. ISBN 3-540-14039-5.

BONEH, D.; LYNN, B.; SHACHAM, H. Short signatures from the Weil pairing. In: *Advances in Cryptology – Asiacrypt 2001*. Gold Coast, Australia: Springer, 2002. (Lecture Notes in Computer Science, v. 2248), p. 514–532.

BOYEN, X. Multipurpose identity-based signcryption: A Swiss Army knife for identity-based cryptography. In: *Advances in Cryptology – Crypto 2003*. Santa Barbara, USA: Springer, 2003. (Lecture Notes in Computer Science, v. 2729), p. 383–399.

_____. Lattice mixing and vanishing trapdoors: A framework for fully secure short signatures and more. In: NGUYEN, P.; POINTCHEVAL, D. (Ed.). *Public Key Cryptography – PKC 2010*. Paris, France: Springer Berlin Heidelberg, 2010, (Lecture Notes in Computer Science, v. 6056). p. 499–517. ISBN 978-3-642-13012-0.

BOYEN, X.; WATERS, B. Anonymous hierarchical identity-based encryption (without random oracles). In: *Advances in Cryptology – Crypto 2006*. Santa Barbara, USA: Springer, 2006. (Lecture Notes in Computer Science, v. 4117), p. 290–307.

BRAKERSKI, Z.; VAIKUNTANATHAN, V. Efficient fully homomorphic encryption from (standard) LWE. *Electronic Colloquium on Computational Complexity (ECCC)*, v. 18, p. 109, 2011.

BREZING, F.; WENG, A. Elliptic curves suitable for pairing based cryptography. *Designs, Codes and Cryptography*, Kluwer Academic Publishers, v. 37, n. 1, p. 133–141, 2005. ISSN 0925-1022. Disponível em: <<http://dx.doi.org/10.1007/s10623-004-3808-4>>.

CAI, J.; CUSICK, T. W. A lattice-based public-key cryptosystem. *Information and Computation*, v. 151, p. 17–31, 1999.

CASH, D.; HOFHEINZ, D.; KILTZ, E.; PEIKERT, C. Bonsai trees, or how to delegate a lattice basis. In: GILBERT, H. (Ed.). *Advances in Cryptology – Eurocrypt 2010*. French Riviera: Springer Berlin Heidelberg, 2010, (Lecture Notes in Computer Science, v. 6110). p. 523–552.

CAYREL, P.; HOFFMANN, G.; PERSICHETTI, E. Efficient implementation of a CCA2-secure variant of McEliece using generalized Srivastava codes. In: FISCHLIN, M.; BUCHMANN, J.; MANULIS, M. (Ed.). *Public Key Cryptography – PKC 2012*. Darmstadt, Germany: Springer Berlin Heidelberg, 2012, (Lecture Notes in Computer Science, v. 7293). p. 138–155.

CHA, J. C.; CHEON, J. H. An identity-based signature from gap Diffie-Hellman groups. In: *Public Key Cryptography – PKC 2003*. Miami, USA: Springer, 2003. (Lecture Notes in Computer Science, v. 2567), p. 18–30.

CHAUM, D. Blind signatures for untraceable payments. In: CHAUM, D.; RIVEST, R.; SHERMAN, A. (Ed.). *Advances in Cryptology – CRYPTO '82*. Santa Barbara, CA, USA: Springer, 1983. p. 199–203.

CHEN, L.; CHENG, Z.; SMART, N. P. Identity-based key agreement protocols from pairings. *International Journal of Information Security*, Springer, v. 6, n. 4, p. 213–241, 2007.

CHEN, X.; ZHANG, F.; KIM, K. New ID-based group signature from pairings. *Journal of Electronics (China)*, Science Press, v. 23, n. 6, p. 892–900, 2006. ISSN 0217-9822.

CHEON, J. H. Security analysis of the strong Diffie-Hellman problem. In: *Advances in Cryptology – Eurocrypt 2006*. Saint Petersburg, Russia: Springer, 2006. (Lecture Notes in Computer Science, v. 4004), p. 1–11.

_____. Discrete logarithm problems with auxiliary inputs. *Journal of Cryptology*, Springer, v. 23, n. 3, p. 457–476, 2010.

CHEUNG, R. C. C.; DUQUESNE, S.; FAN, J.; GUILLERMIN, N.; VERBAUWHEDE, I.; YAO, G. X. FPGA implementation of pairings using residue number system and lazy reduction. In: PRENEEL, B.; TAKAGI, T. (Ed.). *Cryptographic Hardware and Embedded Systems – CHES 2011*. Nara, Japan: Springer, 2011. (Lecture Notes in Computer Science, v. 6917), p. 421–441.

COCKS, C.; PINCH, R. G. E. Identity-based cryptosystems based on the Weil pairing. *Unpublished manuscript*, 2001.

COPPERSMITH, D. Finding small solutions to small degree polynomials. In: *Revised Papers from the International Conference on Cryptography and Lattices*. London, UK: Springer-Verlag, 2001. (CaLC '01), p. 20–31. ISBN 3-540-42488-1.

CORMEN, T. H.; LEISERSON, C. E.; RIVEST, R. L.; STEIN, C. *Introduction to Algorithms*. 3rd. ed. Massachusetts, USA: The MIT Press, 2009. ISBN 978-0-262-03384-8.

COSTELLO, C. *Particularly Friendly Members of Family Trees*. 2012. Cryptology ePrint Archive, Report 2012/072. <http://eprint.iacr.org/2012/072>.

COSTELLO, C.; LANGE, T.; NAEHRIG, M. Faster Pairing Computations on Curves with High-Degree Twists. In: NGUYEN, P. Q.; POINTCHEVAL, D. (Ed.). *Public Key Cryptography – PKC 2010*. Paris, France: Springer, 2010. (Lecture Notes in Computer Science, v. 6056), p. 224–242.

COSTELLO, C.; LAUTER, K.; NAEHRIG, M. Attractive subfamilies of BLS curves for implementing high-security pairings. In: *Progress in Cryptology – Indocrypt 2011*. Chennai, India: Springer, 2011. (Lecture Notes in Computer Science, v. 7107), p. 320–342.

DEVEGILI, A. J.; SCOTT, M.; DAHAB, R. Implementing cryptographic pairings over Barreto-Naehrig curves. In: *Pairing-Based Cryptography – Pairing 2007*. Tokyo, Japan: Springer, 2007. (Lecture Notes in Computer Science, v. 4575), p. 197–207.

DUPONT, R.; ENGE, A.; MORAIN, F. Building curves with arbitrary small MOV degree over finite prime fields. *Journal of Cryptology*, Springer, v. 18, n. 2, p. 79–89, 2005. ISSN 0933-2790. Disponível em: <<http://dx.doi.org/10.1007/s00145-004-0219-7>>.

DUURSMA, I.; LEE, H.-S. Tate pairing implementation for hyperelliptic curves. In: LAIH, C.-S. (Ed.). *Advances in Cryptology – Asiacrypt 2003*. Springer, 2003, (Lecture Notes in Computer Science, v. 2894). p. 111–123. ISBN 978-3-540-20592-0. Disponível em: <http://dx.doi.org/10.1007/978-3-540-40061-5_7>.

FAN, J.; VERCAUTEREN, F.; VERBAUWHEDE, I. Efficient hardware implementation of $\mathbb{F}_p$ -arithmetic for pairing-friendly curves. *IEEE Transactions on Computers*, v. 61, n. 5, p. 676–685, 2012.

FISCHLIN, R.; SEIFERT, J.-P. Tensor-based trapdoors for CVP and their application to public key cryptography (extended abstract). In: WALKER, M. (Ed.). *Cryptography and Coding*. Cirencester, UK: Springer Berlin Heidelberg, 1999, (Lecture Notes in Computer Science, v. 1746). p. 244–257. ISBN 978-3-540-66887-9.

FOUQUE, P.-A.; TIBOUCHI, M. Indifferentiable hashing to Barreto-Naehrig curves. In: *Progress in Cryptology – Latincrypt 2012*. Santiago, Chile: Springer, 2012. (Lecture Notes in Computer Science, v. 7533), p. 1–17.

FREEMAN, D. Constructing pairing-friendly elliptic curves with embedding degree 10. In: HESS, F.; PAULI, S.; POHST, M. (Ed.). *Algorithmic Number Theory*. Berlin, Germany: Springer, 2006, (Lecture Notes in Computer Science, v. 4076). p. 452–465.

FREEMAN, D.; SCOTT, M.; TESKE, E. A taxonomy of pairing-friendly elliptic curves. *Journal of Cryptology*, Springer, v. 23, n. 2, p. 224–280, 2010. ISSN 0933-2790.

FREY, G.; MÜLLER, M.; RÜCK, H. The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems. *IEEE Transactions on Information Theory*, v. 45, n. 5, p. 1717–1719, 1999.

FREY, G.; RÜCK, H. G. A remark concerning m -divisibility and the discrete logarithm problem in the divisor class group of curves. *Mathematics of Computation*, v. 62, p. 865–874, 1994.

FUENTES-CASTAÑEDA, L.; KNAPP, E.; RODRÍGUEZ-HENRÍQUEZ, F. Faster Hashing to $\mathbb{G}_2$. In: MIRI, A.; VAUDENAY, S. (Ed.). *Selected Areas in Cryptography – SAC 2011*. Toronto, Canada: Springer, 2011. (Lecture Notes in Computer Science, v. 7118), p. 412–430.

GALBRAITH, S.; HARRISON, K.; SOLDERA, D. Implementing the Tate pairing. In: *Algorithm Number Theory Symposium – ANTS V*. Sydney, Australia: Springer, 2002. (Lecture Notes in Computer Science, v. 2369), p. 324–337.

GALBRAITH, S. D. Supersingular curves in cryptography. In: BOYD, C. (Ed.). *Advances in Cryptology – Asiacrypt 2001*. Gold Coast, Australia: Springer, 2001, (Lecture Notes in Computer Science, v. 2248). p. 495–513. ISBN 978-3-540-42987-6.

GALBRAITH, S. D.; DWARAKANATHM, N. C. *Efficient sampling from discrete Gaussians for lattice-based cryptography on a constrained device*. 2012. Preprint.

GALBRAITH, S. D.; PATERSON, K. G.; SMART, N. P. Pairings for cryptographers. *Discrete Applied Mathematics*, v. 156, n. 16, p. 3113–3121, 2008. ISSN 0166-218X.

GENTRY, C. *A fully homomorphic encryption scheme*. Tese (Doutorado) — Stanford University, 2009. crypto.stanford.edu/craig.

GENTRY, C.; PEIKERT, C.; VAIKUNTANATHAN, V. Trapdoors for hard lattices and new cryptographic constructions. In: *Proceedings of the 40th annual ACM symposium on Theory of computing – STOC '08*. New York, NY, USA: ACM, 2008. (Lecture Notes in Computer Science), p. 197–206. ISBN 978-1-60558-047-0. Disponível em: <<http://doi.acm.org/10.1145/1374376.1374407>>.

GENTRY, C.; SILVERBERG, A. Hierarchical ID-based cryptography. In: *Advances in Cryptology – Asiacrypt 2002*. Queenstown, New Zealand: Springer, 2002. (Lecture Notes in Computer Science, v. 2501), p. 548–566.

GOLDREICH, O.; GOLDWASSER, S.; HALEVI, S. Public-key cryptosystems from lattice reduction problems. In: *Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology – CRYPTO'97*. Santa Barbara,

CA, USA: Springer-Verlag, 1997. (Lecture Notes in Computer Science), p. 112–131. ISBN 3-540-63384-7.

GOUVÊA, C. P. L.; LÓPEZ, J. Software implementation of pairing-based cryptography on sensor networks using the MSP430 microcontroller. In: ROY, B. K.; SENDRIER, N. (Ed.). *10th International Conference on Cryptology in India – Indocrypt 2009*. New Delhi, India: Springer, 2009. (Lecture Notes in Computer Science, v. 5922), p. 248–262.

GRENTY, C.; HALEVI, S. *Implementing Gentry's Fully-Homomorphic Encryption Scheme*. 2010. Cryptology ePrint Archive, Report 2010/520. <http://eprint.iacr.org/2010/520.pdf>.

GREWAL, G.; AZARDEKAKHSH, R.; LONGA, P.; HU, S.; JAO, D. Efficient Implementation of Bilinear Pairings on ARM Processors. In: KNUDSEN, L. R.; WU, H. (Ed.). *Selected Areas in Cryptography – SAC 2012*. Ontario, Canada: Springer, 2012. (Lecture Notes in Computer Science, v. 7707), p. 149–165.

GÜNEYSU, T.; LYUBASHEVSKY, V.; PÖPPELMANN, T. Practical lattice-based cryptography: A signature scheme for embedded systems. In: PROUFF, E.; SCHAUMONT, P. (Ed.). *Cryptographic Hardware and Embedded Systems – CHES 2012*. Leuven, Belgium: Springer Berlin Heidelberg, 2012, (Lecture Notes in Computer Science, v. 7428). p. 530–547. ISBN 978-3-642-33026-1.

HANKERSON, D.; MENEZES, A.; SCOTT, M. Software implementation of pairings. IOS Press, p. 188–206, 2008.

HASSE, H. *Algebraische Theorie der Körper*. Walter de Gruyter, 1930.

HESS, F.; SMART, N.; VERCAUTEREN, F. The eta pairing revisited. *IEEE Transactions on Information Theory*, v. 52, p. 4595–4602, 2006.

HEYSE, S. Implementation of McEliece based on quasi-dyadic Goppa codes for embedded devices. In: YANG, B.-Y. (Ed.). *Post-Quantum Cryptography – PQCrypto 2011*. Taipei, Taiwan: Springer Berlin Heidelberg, 2011, (Lecture Notes in Computer Science, v. 7071). p. 143–162.

HOFFSTEIN, J.; HOWGRAVE-GRAHAM, N.; PIPHER, J.; SILVERMAN, J.; WHYTE, W. NTRUSign: Digital signatures using the NTRU lattice. In: JOYE, M. (Ed.). *Topics in Cryptology – CT-RSA 2003*. California, USA: Springer Berlin Heidelberg, 2003, (Lecture Notes in Computer Science, v. 2612). p. 122–140.

HOFFSTEIN, J.; PIPHER, J.; SILVERMAN, J. NTRU: A ring-based public key cryptosystem. In: BUHLER, J. P. (Ed.). *Algorithmic Number Theory*. Oregon, USA: Springer Berlin Heidelberg, 1998, (Lecture Notes in Computer Science, v. 1423). p. 267–288.

_____. NSS: An NTRU lattice-based signature scheme. In: PFITZMANN, B. (Ed.). *Advances in Cryptology – EUROCRYPT 2001*. Innsbruck, Austria: Springer Berlin Heidelberg, 2001, (Lecture Notes in Computer Science, v. 2045). p. 211–228. ISBN 978-3-540-42070-5.

HUSEMÖLLER, D. Elementary properties of the chord-tangent group law on a cubic curve. In: *Elliptic Curves*. New York, USA: Springer, 2004. v. 111, p. 23–43. http://dx.doi.org/10.1007/0-387-21577-8_2.

JOUX, A. A one-round protocol for tripartite Diffie-Hellman. In: *Algorithm Number Theory Symposium – ANTS IV*. Leiden, The Netherlands: Springer, 2000. (Lecture Notes in Computer Science, v. 1838), p. 385–394.

JOUX, A.; STERN, J. Lattice reduction: A toolbox for the cryptanalyst. *Journal of Cryptology*, Springer-Verlag, v. 11, n. 3, p. 161–185, 1998. ISSN 0933-2790.

KACHISA, E.; SCHAEFER, E.; SCOTT, M. Constructing Brezing-Weng pairing friendly elliptic curves using elements in the cyclotomic field. In: *Pairing-Based Cryptography – Pairing 2008*. Egham, UK: Springer, 2008. (Lecture Notes in Computer Science, v. 5209), p. 126–135.

KARABINA, K.; TESKE, E. On prime-order elliptic curves with embedding degrees $k = 3, 4$, and 6 . In: *Proceedings of the 8th international conference on Algorithmic number theory – ANTS-VIII*. Banff, Canada: Springer, 2008. (Lecture Notes in Computer Science, v. 5011), p. 102–117. ISBN 3-540-79455-7, 978-3-540-79455-4.

KARATSUBA, A.; OFMAN, Y. *Multiplication of Many-Digital Numbers by Automatic Computers*. 1962. 293–294 p. Translation in *Physics-Doklady* 7, 595–596, 1963.

KHOT, S. Inapproximability results for computational problems on lattices. In: NGUYEN, P. Q.; VALLÉ, B. (Ed.). *The LLL Algorithm*. Berlin-Heidelberg, Germany: Springer Berlin Heidelberg, 2010, (Information Security and Cryptography). p. 453–473.

KINDERMAN, A. J.; MONAHAN, J. F. Computer generation of random variables using the ratio of uniform deviates. *ACM Transactions on Mathematical Software*, ACM, New York, NY, USA, v. 3, n. 3, p. 257–260, set. 1977. ISSN 0098-3500.

KLEIN, P. Finding the closest lattice vector when it's unusually close. In: *Symposium on Discrete Algorithms - SODA 2000*. San Francisco, CA: ACM, 2000. p. 937–941.

KOBLITZ, N. Elliptic Curve Cryptosystems. *Mathematics of Computation*, v. 48, n. 177, p. 203–209, 1987. <http://www.jstor.org/stable/2007884>.

LAGARIAS, J.; ODLYZKO, A. Solving low density subset sum problems. In: *24th Annual Symposium on Foundations of Computer Science, 1983*. California, USA: Birkhäuser-Verlag, 1983. p. 1–10. ISSN 0272-5428.

LAUTER, K.; MONTGOMERY, P.; NAEHRIG, M. An analysis of affine coordinates for pairing computation. In: *Pairing-Based Cryptography – Pairing 2010*. Yamanaka Onsen, Japan: Springer, 2010. (Lecture Notes in Computer Science, v. 6487), p. 1–20.

LEE, E.; LEE, H.-S.; PARK, C.-M. Efficient and generalized pairing computation on abelian varieties. *IEEE Transactions on Information Theory*, IEEE, v. 55, n. 4, p. 1793–1803, 2009.

LENSTRA, A.; LENSTRA H.W., J.; LOVÁSZ, L. Factoring polynomials with rational coefficients. *Mathematische Annalen*, Springer-Verlag, v. 261, n. 4, p. 515–534, 1982. ISSN 0025-5831. Disponível em: <<http://dx.doi.org/10.1007/BF01457454>>.

LENSTRA, H. W. Integer programming in a fixed number of variables. *Math. Oper. Res.*, v. 8, p. 538–548, 1983.

LIBERT, B.; QUISQUATER, J.-J. New identity based signcryption schemes from pairings. In: *Information Theory Workshop – ITW 2003*. Paris, France: IEEE, 2003. p. 155–158.

LIDL, R.; NIEDERREITER, H. *Finite Fields*. 2nd. ed. Cambridge, UK: Cambridge University Press, 1997. (Encyclopedia of Mathematics and its Applications, 20).

LINDNER, R.; PEIKERT, C. Better key sizes (and attacks) for LWE-based encryption. In: *Topics in Cryptology – CT-RSA 2011*. San Francisco, CA, USA: Springer, 2011. (Lecture Notes in Computer Science, v. 6558), p. 319–339.

LUCA, F.; SHPARLINSKI, I. E. Elliptic curves with low embedding degree. *Journal of Cryptology*, Springer, v. 19, n. 4, p. 553–562, 2006. ISSN 0933-2790.

LYUBASHEVSKY, V. Fiat-shamir with aborts: Applications to lattice and factoring-based signatures. In: *Advances in Cryptology – ASIACRYPT '09*. Berlin, Heidelberg: Springer-Verlag, 2009. (Lecture Notes in Computer Science), p. 598–616.

_____. Lattice signatures without trapdoors. In: *Advances in Cryptology – EUROCRYPT 2012*. Cambridge, UK: Springer, 2012. (Lecture Notes in Computer Science, v. 7237), p. 738–755.

LYUBASHEVSKY, V.; PEIKERT, C.; REGEV, O. On ideal lattices and learning with errors over rings. In: GILBERT, H. (Ed.). *French Riviera*: Springer, 2010. v. 6110/2010, n. 015848, p. 1–23.

MARGI, C. B.; OLIVEIRA, B. T. de; SOUSA, G. T. de; JR., M. A. S.; BARRETO, P. S. L. M.; CARVALHO, T. C. M. B.; NÄSLUND, M.; GOLD, R. Impact of operating systems on wireless sensor networks (security) applications and testbeds. In: *ICCCN*. Zurich, Switzerland: IEEE, 2010. p. 1–6.

MARSAGLIA, G.; BRAY, T. A. A convenient method for generating normal variables. *SIAM Review*, v. 6, p. 260–264, 1964.

MATSUDA, S.; KANAYAMA, N.; HESS, F.; OKAMOTO, E. Optimised versions of the ate and twisted ate pairings. In: GALBRAITH, S. D. (Ed.). *Cryptography and Coding*. Cirencester, UK: Springer, 2007. (Lecture Notes in Computer Science, v. 4887). p. 302–312. ISBN 978-3-540-77271-2.

MENEZES, A. J. *Elliptic Curve Public Key Cryptosystems*. Boston, USA: Kluwer Academic Publishers, 1993.

MENEZES, A. J.; OKAMOTO, T.; VANSTONE, S. A. Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Transactions on Information Theory*, v. 39, p. 1639–1646, 1993.

MICCIANCIO, D. *On the hardness of the shortest vector problem*. Massachusetts, USA, 1998.

MICCIANCIO, D.; GOLDWASSER, S. *Complexity of Lattice Problems: a cryptographic perspective*. Boston, Massachusetts: Kluwer Academic Publishers, 2002. (The Kluwer International Series in Engineering and Computer Science, v. 671).

MICCIANCIO, D.; PEIKERT, C. Trapdoors for lattices: Simpler, tighter, faster, smaller. In: POINTCHEVAL, D.; JOHANSSON, T. (Ed.). *Advances in Cryptology – Eurocrypt 2012*. Cambridge, UK: Springer Berlin Heidelberg, 2012, (Lecture Notes in Computer Science, v. 7237). p. 700–718.

MICCIANCIO, D.; REGEV, O. Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, Society for Industrial and Applied Mathematics, Philadelphia, USA, v. 37, p. 267–302, 2007.

MICCIANCIO, D.; VADHAN, S. Statistical zero-knowledge proofs with efficient provers: Lattice problems and more. In: BONEH, D. (Ed.). *Advances in Cryptology – Crypto 2003*. Santa Barbara, CA: Springer Berlin Heidelberg, 2003, (Lecture Notes in Computer Science, v. 2729). p. 282–298. ISBN 978-3-540-40674-7.

MILLER, V. S. *Short Programs for functions on Curves*. 1986. IBM Thomas J. Watson Research Center Report. <http://crypto.stanford.edu/miller/miller.pdf>.

_____. Use of elliptic curves in cryptography. In: *Lecture notes in computer sciences; 218 on Advances in cryptology – Crypto’85*. New York, NY, USA: Springer-Verlag New York, Inc., 1986. p. 417–426. ISBN 0-387-16463-4. <http://portal.acm.org/citation.cfm?id=18262.25413>.

_____. The Weil pairing, and its efficient calculation. *Journal of Cryptology*, Springer, v. 17, n. 4, p. 235–261, 2004.

MISOCZKI, R.; BARRETO, P. Compact mceliece keys from goppa codes. In: JACOBSON MICHAELJ., J.; RIJMEN, V.; SAFAVI-NAINI, R. (Ed.). *Selected Areas in Cryptography*. Alberta, Canada: Springer Berlin Heidelberg, 2009, (Lecture Notes in Computer Science, v. 5867). p. 376–392.

MITSUNARI, S. *A Fast Implementation of the Optimal Ate Pairing over BN curve on Intel Haswell Processor*. 2013. Cryptology ePrint Archive, Report 2013/362. <http://eprint.iacr.org/>.

MITSUNARI, S.; SAKAI, R.; KASAHARA, M. A new traitor tracing. *IEICE Transactions on Fundamentals*, IEICE, E85-A, n. 2, p. 481–484, 2002.

MIYAJI, A.; NAKABAYASHI, M.; TAKANO, S. New explicit conditions of elliptic curve traces for FR-reduction. *IEICE Transactions on Fundamentals*, E84-A, n. 5, p. 1234–1243, 2001.

MONTGOMERY, P. L. Modular Multiplication Without Trial Division. *Mathematics of Computation*, American Mathematical Society, v. 44, n. 170, p. 519–521, 1985. ISSN 00255718.

NAEHRIG, M.; BARRETO, P. S. L. M.; SCHWABE, P. On compressible pairings and their computation. In: *Progress in Cryptology – Africacrypt 2008*. Casablanca, Morocco: Springer, 2008. (Lecture Notes in Computer Science, v. 5023), p. 371–388.

NAEHRIG, M.; NIEDERHAGEN, R.; SCHWABE, P. New software speed records for cryptographic pairings. In: *Progress in Cryptology – Latincrypt 2010*. Puebla, Mexico: Springer, 2010. (Lecture Notes in Computer Science, v. 6212), p. 109–123.

NGUYEN, P.; REGEV, O. Learning a parallelepiped: Cryptanalysis of ggh and ntru signatures. In: VAUDENAY, S. (Ed.). *Advances in Cryptology - EUROCRYPT 2006*. Saint Petersburg, Russia: Springer Berlin Heidelberg, 2006, (Lecture Notes in Computer Science, v. 4004). p. 271–288. ISBN 978-3-540-34546-6.

NIST. *Federal Information Processing Standard FIPS 186-3 – Digital Signature Standard (DSS) – 6. The Elliptic Curve Digital Signature Algorithm (ECDSA)*. Gaithersburg, USA: National Institute of Standards and Technology (NIST), 2012. http://csrc.nist.gov/publications/fips/fips186-3/fips_186-3.pdf.

NOGAMI, Y.; AKANE, M.; SAKEMI, Y.; KATO, H.; MORIKAWA, Y. Integer variable χ -based ate pairing. In: GALBRAITH, S. D.; PATERSON, K. G. (Ed.). *Pairing-Based Cryptography – Pairing 2008*. Egham, UK: Springer, 2008, (Lecture Notes in Computer Science, v. 5209). p. 178–191.

OKAMOTO, T.; POINTCHEVAL, D. The gap-problems: A new class of problems for the security of cryptographic schemes. In: *Public Key Cryptography – PKC 2001*. London, UK: Springer, 2001. p. 104–118. ISBN 3-540-41658-7.

PEIKERT, C. Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: *Proceedings of the 41st annual ACM symposium on Theory of computing*. New York, NY, USA: ACM, 2009. p. 333–342. Disponível em: <<http://doi.acm.org/10.1145/1536414.1536461>>.

_____. An efficient and parallel gaussian sampler for lattices. In: RABIN, T. (Ed.). *Advances in Cryptology – CRYPTO 2010*. Santa Barbara, CA: Springer Berlin Heidelberg, 2010, (Lecture Notes in Computer Science, v. 6223). p. 80–97. ISBN 978-3-642-14622-0.

PEIKERT, C.; VAIKUNTANATHAN, V.; WATERS, B. A framework for efficient and composable oblivious transfer. In: WAGNER, D. (Ed.). *Advances in Cryptology – CRYPTO 2008*. Santa Barbara, CA: Springer Berlin Heidelberg, 2008, (Lecture Notes in Computer Science, v. 5157). p. 554–571. ISBN 978-3-540-85173-8.

PEREIRA, G. C. C. F.; SIMPLÍCIO JR., M. A.; NAEHRIG, M.; BARRETO, P. S. L. M. A family of implementation-friendly BN elliptic curves. *Journal of Systems and Software*, Elsevier Science Inc., New York, NY, USA, v. 84, n. 8, p. 1319–1326, August 2011. ISSN 0164-1212.

POLLARD, J. M. Monte Carlo methods for index computation (mod p). *Mathematics of Computation*, v. 32, p. 918–924, 1978.

RÜCKERT, M. Lattice-based blind signatures. In: *Advances in Cryptology - ASIACRYPT 2010*. Singapore: Springer, 2010. (Lecture Notes in Computer Science, v. 6477), p. 413–430.

REGEV, O. On lattices, learning with errors, random linear codes, and cryptography. In: *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing*. New York, NY, USA: ACM, 2005. p. 84–93. ISBN 1-58113-960-8.

_____. The learning with errors problem (invited survey). In: *IEEE Conference on Computational Complexity*. Massachusetts, USA: IEEE Computer Society, 2010. p. 191–204. ISBN 978-0-7695-4060-3.

RIVEST, R.; ADLEMAN, L.; DERTOUZOS, M. On data banks and privacy homomorphisms. In: *Foundations of Secure Computation*. Florida, USA: Academic Press, 1978. p. 169–179.

ROJO, O. A new algebra of Toeplitz-plus-Hankel matrices and applications. *Computers and Mathematics with Applications*, v. 55, n. 12, p. 2856 – 2869, 2008. ISSN 0898-1221.

RUBIN, K.; SILVERBERG, A. Supersingular Abelian varieties in cryptology. In: YUNG, M. (Ed.). *Advances in Cryptology – CRYPTO 2002*. California, USA: Springer, 2002, (Lecture Notes in Computer Science, v. 2442). p. 336–353.

SAKAI, R.; KASAHARA, M. Cryptosystems based on pairing over elliptic curve. In: *Symposium on Cryptography and Information Security – SCIS 2003*. Hamamatsu, Japan: Springer Berlin Heidelberg, 2003. p. 8C–1.

SAKAI, R.; OHGISHI, K.; KASAHARA, M. Cryptosystems based on pairing. In: *Symposium on Cryptography and Information Security – SCIS 2000*. Okinawa, Japan: Springer, 2000.

SCHNORR, C. P. Efficient identification and signatures for smart cards. In: *Proceedings of the 9th Annual International Cryptology Conference on Advances in Cryptology*. London, UK: Springer-Verlag, 1990. (CRYPTO'89), p. 239–252.

SCOTT, M. *A note on twists for pairing friendly curves*. 2009. <ftp://ftp.computing.dcu.ie/pub/resources/crypto/twists.pdf>.

_____. On the efficient implementation of pairing-based protocols. In: *Cryptography and Coding*. Oxford, UK: Springer, 2011. (Lecture Notes in Computer Science, v. 7089), p. 296–308.

SCOTT, M.; BARRETO, P. S. L. M. Compressed pairings. In: FRANKLIN, M. (Ed.). *Advances in Cryptology – Crypto 2004*. Santa Barbara, USA: Springer, 2004. (Lecture Notes in Computer Science, v. 3152), p. 140–156.

SCOTT, M.; BENDER, N.; CHARLEMAGNE, M.; PÉREZ, L. J. D.; KACHISA, E. J. On the final exponentiation for calculating pairings on ordinary elliptic curves. In: SHACHAM, H.; WATERS, B. (Ed.). *Pairing-Based Cryptography – Pairing 2009*. Springer Berlin Heidelberg, 2009, (Lecture Notes in Computer

Science, v. 5671). p. 78–88. ISBN 978-3-642-03297-4. Disponível em: <http://dx.doi.org/10.1007/978-3-642-03298-1_6>.

SHIRASE, M. *Barreto-Naehrig Curve With Fixed Coefficient*. 2010. IACR ePrint Archive, report 2010/134. <http://eprint.iacr.org/2010/134>.

SHOR, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, v. 26, p. 1484–1509, 1997.

SILVERMAN, J. H. *The Arithmetic of Elliptic Curves*. Berlin, Germany: Springer, 1986. (Graduate Texts in Mathematics, 106).

STEHLÉ, D.; STEINFELD, R. Making ntru as secure as worst-case problems over ideal lattices. In: *Proceedings of the 30th Annual International Conference on Theory and Applications of Cryptographic Techniques: Advances in Cryptology*. Berlin, Heidelberg: Springer, 2011. (Advances in Cryptology – EUROCRYPT 2011), p. 27–47.

SZYDLO, M. Hypercubic lattice reduction and analysis of GGH and NTRU signatures. In: BIHAM, E. (Ed.). *Advances in Cryptology – EUROCRYPT 2003*. Warsaw, Poland: Springer Berlin Heidelberg, 2003, (Lecture Notes in Computer Science, v. 2656). p. 433–448.

URROZ, J. J.; LUCA, F.; SHPARLINSKI, I. On the number of isogeny classes of pairing-friendly elliptic curves and statistics of MNT curves. *Mathematics of Computation*, v. 81, n. 278, 2012.

VERCAUTEREN, F. Optimal pairings. *IEEE Transactions on Information Theory*, IEEE, v. 56, n. 1, p. 455–461, 2010.

WALLACE, C. S. Fast pseudorandom generators for normal and exponential variates. *ACM Trans. Math. Softw.*, ACM, New York, NY, USA, v. 22, n. 1, p. 119–127, 1996.

WEIL, A. Sur les fonctions algébriques à corps de constantes fini. *Comptes rendus de l'Académie des sciences*, v. 210, p. 592–594, 1940.

ZAVATTONI, E.; DOMÍNGUEZ-PÉREZ, L. J.; MITSUNARI, S.; SÁNCHEZ, A. H.; TERUYA, T.; RODRÍGUEZ-HENRÍQUEZ, F. *Software implementation of Attribute-Based Encryption*. 2013. <http://sandia.cs.cinvestav.mx/index.php?n=Site.CPABE>.

ZHANG, F.; CHEN, X. *Yet Another Short Signatures Without Random Oracles from Bilinear Pairings*. 2005. IACR Cryptology ePrint Archive, report 2005/230.

ZHANG, F.; KIM, K. ID-based blind signature and ring signature from pairings. In: ZHENG, Y. (Ed.). *Advances in Cryptology – Asiacrypt 2002*. Queenstown, New Zealand: Springer, 2002, (Lecture Notes in Computer Science, v. 2501). p. 533–547. ISBN 978-3-540-00171-3.

ZHANG, F.; SAFAVI-NAINI, R.; SUSILO, W. An efficient signature scheme from bilinear pairings and its applications. In: BAO, F.; DENG, R.; ZHOU, J. (Ed.). *Public Key Cryptography – PKC 2004*. Singapore: Springer, 2004, (Lecture Notes in Computer Science, v. 2947). p. 277–290. ISBN 978-3-540-21018-4.

ZHENG, Y. Digital signcryption or how to achieve $\text{cost}(\text{signature} \ \& \ \text{encryption}) \ll \text{cost}(\text{signature}) + \text{cost}(\text{encryption})$. In: *Advances in Cryptology – CRYPTO'97*. Santa Barbara, USA: Springer Berlin Heidelberg, 1997, (Lecture Notes in Computer Science, v. 1294). p. 165–179. ISBN 978-3-540-63384-6.