


UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA



JULIANO BANDEIRA LIMA



**TRIGONOMETRIA SOBRE CORPOS
FINITOS: NOVAS DEFINIÇÕES E
CENÁRIOS DE APLICAÇÃO**



VIRTUS IMPAVIDA

RECIFE, SETEMBRO DE 2008.

JULIANO BANDEIRA LIMA

**TRIGONOMETRIA SOBRE CORPOS
FINITOS: NOVAS DEFINIÇÕES E
CENÁRIOS DE APLICAÇÃO**

Tese submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco como parte dos requisitos para obtenção do grau de **Doutor em Engenharia Elétrica**

ORIENTADOR: PROF. RICARDO MENEZES CAMPELLO DE SOUZA, PH.D.

CO-ORIENTADOR: PROF. DANIEL PANARIO, PH.D.

Recife, Setembro de 2008.

©Juliano Bandeira Lima, 2008

L732t Lima, Juliano Bandeira

Trigonometria sobre corpos finitos: novas definições e cenários de aplicação / Juliano Bandeira Lima. – Recife: O Autor, 2008.

157 f.; il., grafs.; tabs.

Tese (Doutorado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Engenharia Elétrica, 2008.

Inclui Referências Bibliográficas e Apêndices.

1. Engenharia Elétrica. 2. Trigonometria em Corpos Finitos. 3. Transformadas Digitais. I. Título.

621.3 CDD (22.ed.)

UFPE/BCTG/2008-245



Universidade Federal de Pernambuco
Pós-Graduação em Engenharia Elétrica

PARECER DA COMISSÃO EXAMINADORA DE DEFESA DE TESE DE DOUTORADO

JULIANO BANDEIRA LIMA

TÍTULO

**“TRIGONOMETRIA SOBRE CORPOS FINITOS:
NOVAS DEFINIÇÕES E CENÁRIOS DE APLICAÇÃO”**

A comissão examinadora composta pelos professores: RICARDO MENEZES CAMPELLO DE SOUZA, DES/UFPE, HÉLIO MAGALHÃES DE OLIVEIRA, DES/UFPE, VALDEMAR CARDOSO DA ROCHA JÚNIOR, DES/UFPE, FRANCISCO MADEIRO BERNARDINO JÚNIOR, DEI/UNICAP e FRANCISCO MARCOS DE ASSIS, DEE/UFCG, sob a presidência do primeiro, consideram o candidato **JULIANO BANDEIRA LIMA APROVADO.**

Recife, 30 de setembro de 2008.

EDUARDO FONTANA

Coordenador e Membro Titular Interno

RICARDO MENEZES CAMPELLO DE SOUZA

Orientador e Membro Titular Interno

FRANCISCO MADEIRO BERNARDINO

JÚNIOR

Membro Titular Externo

HÉLIO MAGALHÃES DE OLIVEIRA

Membro Titular Interno

FRANCISCO MARCOS DE ASSIS

Membro Titular Externo

VALDEMAR CARDOSO DA ROCHA JÚNIOR

Membro Titular Externo

Aos meus pais, **Beto e Elza**,
e ao meu irmão, **Renato**.

AGRADECIMENTOS

Meu sincero agradecimento a todos que, de alguma forma, contribuíram para que este trabalho fosse realizado:

A minha família, pelo suporte e pela compreensão incondicionais.

Ao Prof. Ricardo Menezes Campello de Souza, amigo, educador e orientador da tese, pela dedicação e pela confiança; ao Prof. Daniel Panario, co-orientador da tese, pelo acolhimento e pelo incentivo.

Ao Prof. Hélio Magalhães de Oliveira, pela vibração e pelo estímulo ao trabalho e às novas descobertas; aos demais professores do Grupo de Pesquisa em Comunicações, pela disponibilidade e pelos conselhos.

Aos amigos da Eng. Elétrica: André Leite, André Ricardson, Andrei Formiga, Daniel Simões, Danielle Paes Barretto, Gilson da Silva Jr., Giovanna Angelis, Humberto Vasconcelos, João Marcelo da Silva e Márcio Lima, que proporcionaram anos de agradável convivência no Laboratório de Telecomunicações.

Aos colegas do estágio no exterior: Prof. Qiang (Steven) Wang, pelas valiosas sugestões; Ariane Masuda e Burak Kantarci, pelo apoio e pelos momentos de descontração em Ottawa; Gang Li, Maximilian Dürre, Reza Naserasr, Robert Bailey, Yuanyuan Liu e Yuliya Mart-synyuk, pela partilha de experiências ao longo do estágio.

Ao CNPq, à CAPES e ao Programa de Pós-Graduação em Engenharia Elétrica.

Finalmente, agradeço a Deus pelo dom da vida e por me capacitar a aprender, criar e ser instrumento do Seu amor para outras pessoas.

Universidade Federal de Pernambuco

30 de Setembro de 2008

J. B. L.

Resumo da Tese apresentada à UFPE como parte dos requisitos necessários para a obtenção do grau de Doutor em Engenharia Elétrica

TRIGONOMETRIA SOBRE CORPOS FINITOS: NOVAS DEFINIÇÕES E CENÁRIOS DE APLICAÇÃO

Juliano Bandeira Lima

Setembro/2008

Orientador: Prof. Ricardo Menezes Campello de Souza, Ph.D.

Co-orientador: Prof. Daniel Panario, Ph.D.

Área de Concentração: Comunicações

Palavras-chave: Trigonometria em corpos finitos, transformadas digitais.

Número de páginas: 157

Nesta tese, são introduzidas novas ferramentas matemáticas relacionadas à trigonometria sobre corpos finitos e propostos alguns cenários de aplicação para as mesmas. O ponto de partida para o trabalho desenvolvido é a inédita definição das transformadas trigonométricas sobre corpos finitos (FFTT), o que inclui oito transformadas do co-seno (FFCT) e oito do seno (FFST). Estabelecidas as suas principais propriedades, propõem-se duas aplicações. A primeira delas é uma marca d'água digital frágil no domínio da FFCT; na segunda, demonstra-se o uso da propriedade de convolução simétrica das FFTT na filtragem de imagens. Em seguida, investiga-se a auto-estrutura das FFTT. Tal estudo revela alguns aspectos acerca da capacidade de formatar distribuições de probabilidade sobre os inteiros que essas transformadas possuem e cujo emprego em Criptografia é sugerido. Ainda com base nas referidas auto-estruturas, propõe-se uma técnica para separação cega de seqüências. Para isso, toma-se como referência um cenário de comunicação multiusuário, em que as informações oriundas de fontes distintas interferem de forma aditiva e são posteriormente recuperadas. Por fim, define-se a função co-seno inversa sobre corpos finitos, a qual é empregada numa nova definição para polinômios de Chebyshev em $GF(p)$. Tal definição possibilita demonstrar a segurança de criptossistemas baseados nos polinômios mencionados. Ainda nesse contexto, introduz-se um algoritmo rápido para multiplicação de polinômios na forma de Chebyshev. Ao longo de todo o trabalho, são realizadas diversas simulações e apresentados resultados que permitem avaliar as vantagens dos métodos propostos sobre alternativas convencionais. Simultaneamente, fornecem-se diretrizes que indicam a possibilidade de desenvolver outros trabalhos em que os cenários de aplicação discutidos sejam tratados de forma mais específica.

Abstract of Thesis presented to UFPE as a partial fulfillment of the requirements for the degree of Doctor in Electrical Engineering

FINITE FIELD TRIGONOMETRY: NEW DEFINITIONS AND APPLICATION SCENARIOS

Juliano Bandeira Lima

September/2008

Supervisor: Prof. Ricardo Menezes Campello de Souza, Ph.D.

Co-supervisor: Prof. Daniel Panario, Ph.D.

Area of Concentration: Communications

Keywords: Trigonometry in finite fields, digital transforms

Number of pages: 157

In this thesis, new mathematical tools related to finite field trigonometry are introduced and some application scenarios proposed. The key-point for the work is the new definition of the finite field trigonometric transforms (FFTT), which include eight cosine (FFCT) and eight sine (FFST) transforms. After establishing the properties of these transforms, two applications are proposed. The first one is a fragile digital watermarking technique in the FFCT domain; in the second one, the use of the FFTT symmetric convolution is applied to image filtering. In the sequel, the eigenstructure of the FFTT is studied. Such an investigation reveals some aspects concerning the capacity of these transforms in formatting probability distributions over the integer numbers, the use of which, in the field of Cryptography, is suggested. Also based on the referred eigenstructure, a blind sequence separation technique is proposed. In this framework, a multiuser communication scenario, where the information coming from distinct sources are recovered after being added by the channel, is used as reference. Finally, the inverse cosine function over finite fields is defined and used on a new definition of Chebyshev polynomials over $GF(p)$. Such a definition allows to evaluate the security of cryptosystems based on the mentioned polynomials. In this context, a fast algorithm for multiplying polynomials in the Chebyshev form is introduced. Throughout this work, several simulations are performed and results allowing to evaluate the advantages of the proposed methods over conventional approaches are presented. Simultaneously, directives which indicate potential further developments are given.

LISTA DE FIGURAS

2.1	Tipos de simetria de uma seqüência.	25
3.1	Exemplo mostrando que a seqüência $\mathbf{w}' = (w'_i)$, resultante de uma convolução simétrica, equivale a um trecho da seqüência $\mathbf{w} = (w_i)$, resultante de uma convolução linear com as entradas simetricamente estendidas.	50
3.2	Exemplo mostrando como a mesma $\mathbf{w} = (w_i)$ pode ser calculada tanto por uma convolução simétrica quanto por uma convolução linear, quando a seqüência de entrada é suficientemente preenchida com zeros em ambos os lados.	52
3.3	Espalhamento da marca d'água sobre uma seqüência PN, para obtenção da informação a ser inserida na imagem original ($L = 8$ bits).	56
3.4	Esquema de inserção da marca d'água no domínio da FFCT. Após um procedimento de espalhamento sobre uma seqüência PN, uma marca d'água binária é somada (módulo p) à FFCT _{2D} da imagem original. Em seguida, calcula-se a FFCT _{2D} inversa para se obter a imagem marcada.	57
3.5	Esquema de extração da marca d'água baseada na FFCT. Realiza-se um procedimento o inverso ao de inserção da marca, adicionando-se o bloco <i>Decisão</i> , cuja função é descrita mais a diante.	58
3.6	Funcionamento do bloco “Decisão” na extração da marca d'água ($L = 8$ bits).	58
3.7	(a) Imagem original, 128×128 (lenna.bmp). (b) Marca d'água, 32×32 . (c) Imagem marcada, PSNR= 38,98 dB.	59
3.8	(a) Imagem marcada com brilho acentuado em 30 %. (b) Marca d'água recuperada, $CN = 0,9796$	60
3.9	(a) Imagem marcada com 25 % de sua informação destruída. (b) Marca d'água recuperada, $CN = 0,8662$	61
3.10	(a) Marca recuperada ao se comprimir, segundo o padrão JPEG, o arquivo original em 30 % do seu tamanho, $CN = 0,6747$. (b) Marca recuperada ao se incrementar em 10 % o contraste da imagem marcada, $CN = 0,7175$	61
3.11	<i>carnival.bmp</i> com significativo conteúdo em altas freqüências.	65
3.12	<i>carnival.bmp</i> após filtragem passa-baixas implementada pela convolução simétrica das FFTT.	65
3.13	Imagem original <i>lancer.bmp</i>	66
3.14	<i>lancer.bmp</i> após filtragem passa-altas implementada pela convolução simétrica das FFTT.	66

4.1	Histogramas de uma amostra com 2^{14} números antes e após a aplicação da FFCT-2e de comprimento $N = 8$ sobre GF(127).	81
4.2	Histogramas de uma amostra com 2^{14} números antes e após a aplicação da FFCT-2e de comprimento $N = 8$ sobre GF(127). Neste caso, a transformada foi aplicada duas vezes.	82
4.3	Histogramas de uma amostra com 2^{14} números antes e após a aplicação da FFCT-4e de comprimento $N = 8$ sobre GF(127).	82
5.1	Recuperação das seqüências num esquema com 2 usuários.	86
5.2	Recuperação de seqüências num esquema com 4 usuários, baseado na auto-estrutura da FFCT-2e sobre GF(127) e com comprimento de bloco $N = 4$	89
5.3	Hierarquia com 2 níveis para comunicação multi-usuário baseada na auto-estrutura de matrizes de transformação: mapeamento extra para um corpo finito primo ($p_2 \geq p_1^2$).	91
5.4	Hierarquia com 2 níveis para comunicação multi-usuário baseada na auto-estrutura de matrizes de transformação: mapeamento extra para um corpo de extensão.	92

LISTA DE TABELAS

2.1	Transformadas do co-seno de corpo finito ($\zeta \in GI(p), p \equiv 3(\text{mod } 4)$).	32
2.2	Transformadas do seno de corpo finito ($\zeta \in GI(p), p \equiv 3(\text{mod } 4)$).	33
2.3	Transformadas do co-seno de corpo finito unitárias ($\zeta \in GI(p), p \equiv 3(\text{mod } 4)$).	34
2.4	Transformadas do seno de corpo finito unitárias ($\zeta \in GI(p), p \equiv 3(\text{mod } 4)$).	35
4.1	Multiplicidades dos autovalores da matriz de transformação da transformada de Fourier de corpo finito com dimensões $N \times N$	69
4.2	Multiplicidades dos autovalores da matriz de transformação da transformada do co-seno de corpo finito do tipo $1e$ com dimensões $N \times N$	71
4.3	Multiplicidades dos autovalores da matriz de transformação da transformada do seno de corpo finito do tipo $1e$ com dimensões $N \times N$	71
4.4	Multiplicidades dos autovalores da matriz de transformação da transformada de Fourier de corpo finito generalizada com dimensões $N \times N$	74
4.5	Somadas das multiplicidades dos autovalores $\{1, -1\}$ e $\{j, -j\}$ da matriz de transformação da transformada de Fourier de corpo finito generalizada com dimensões $N \times N$	75
4.6	Multiplicidades dos autovalores da matriz de transformação da transformada do co-seno de corpo finito do tipo $4e$ com dimensões $N \times N$	77
4.7	Multiplicidades dos autovalores da matriz de transformação da transformada do seno de corpo finito do tipo $4e$ com dimensões $N \times N$	77
4.8	Resultados do teste chi-quadrado de Pearson para verificar a aderência de uma amostra à distribuição uniforme (após a aplicação da transformada sobre corpo finito indicada).	83
5.1	Seqüências de usuário: autovetores da matriz de transformação da FFCT- $2e$, $N = 4, p = 127, \zeta = 119 + j119$. Autovalores: $\lambda_1 = 1, \lambda_2 = 20, \lambda_3 = 108$ e $\lambda_4 = 126$	94
5.2	Seqüências de usuário: autovetores da matriz de transformação da FFCT- $2e$, $N = 4, p = 127, \zeta = 119 + j119$. Autovalores: $\lambda_1 = 1, \lambda_2 = 20, \lambda_3 = 108$ e $\lambda_4 = 126$ (continuação da tabela anterior).	95
5.3	Seqüências de usuário: autovetores da matriz de transformação da FFCT- $2e$, $N = 4, p = 127, \zeta = 119 + j119$. Autovalores: $\lambda_1 = 1, \lambda_2 = 20, \lambda_3 = 108$ e $\lambda_4 = 126$ (continuação da tabela anterior).	96

5.4	Autovalores associados a matrizes de transformação da FFCT-2e de comprimento N sobre $\text{GF}(p)$, utilizando o elemento unimodular ζ	96
6.1	Todos os possíveis valores para $\cos_{\zeta_1}(x)$, onde $\zeta_1 = 2 + 2j$ é um elemento unimodular pertencente a $\text{GI}(7)$, tal que $\text{ord}(\zeta_1) = 8$	99
6.2	Todos os possíveis valores para $\cos_{\zeta_2}(x)$, onde $\zeta_2 = 3$ é um elemento pertencente a $\text{GF}(7)$, tal que $\text{ord}(\zeta_2) = 6$	100
6.3	Todos os possíveis valores de $\cos_{\zeta_1}(x)$, onde $\zeta_1 = 2 + 11j$ é um elemento unimodular de $\text{GI}(31)$ tal que $\text{ord}(\zeta_1) = 32$	106
6.4	Status de todos os termos no algoritmo de Karatsuba até $N = 8$. O número de multiplicações $m(n)$ necessário para separar os termos calculados originalmente juntos com outros termos também é apresentado.	115
6.5	Número total de multiplicações e adições para a multiplicação de polinômios na base de Chebyshev pelo método direto (resp. M_d e A_d) e pelo método de Karatsuba (resp. M_k e A_k).	120
6.6	Número total de multiplicações e adições para multiplicar polinômios na base de Chebyshev pelo método da DCT (resp. M_{DCT} e A_{DCT}) e pelo método de Karatsuba (resp. M_k e A_k).	122

SUMÁRIO

1	INTRODUÇÃO	13
1.1	Conteúdo da Tese	16
1.2	Contribuições	17
2	TRANSFORMADAS TRIGONOMÉTRICAS SOBRE CORPOS FINITOS	19
2.1	Preliminares Matemáticos	20
2.1.1	Números Complexos sobre Corpos Finitos	20
2.1.2	Trigonometria em Corpos Finitos	20
2.2	Lemas Fundamentais	21
2.3	Definição das Transformadas	25
2.3.1	Versões Unitárias das FFTTs	31
2.4	Transformadas Numéricas e Exemplos	34
3	PROPRIEDADES DAS FFTT	43
3.1	Linearidade	44
3.2	Deslocamento no tempo	44
3.3	Teorema de Parseval	46
3.4	Relação entre as FFTT e a FFFT	47
3.5	Convolução simétrica	49
3.6	Aplicações	54
3.6.1	Uma Marca D'água Digital no Domínio da Transformada do Co-seno Sobre Corpos Finitos	54
3.6.2	Simulações e resultados	59
3.6.3	Filtragem de imagens via FFTT	62
4	AUTO-ESTRUTURA DAS FFTT	67
4.1	Auto-estrutura das FFTT do tipo $1e$	68
4.2	Auto-estrutura das FFTT do tipo $4e$	72
4.2.1	A transformada de Fourier de corpo finito generalizada	72
4.2.2	Autovalores e autovetores das FFTT do tipo $4e$	76
4.3	Auto-estrutura das FFTT dos tipos $2e$ e $3e$	78
4.4	Uma Aplicação: Formatação de Distribuições de Probabilidade sobre os Inteiros	79

4.4.1	Formatação via FFCT	81
4.4.2	Aplicações em Criptosistemas	83
5	SEPARAÇÃO CEGA DE SEQÜÊNCIAS BASEADA NA AUTO-ESTRUTURA DAS FFTT	85
5.1	Esquema com 2 Usuários	86
5.2	Esquema com 4 Usuários: Um Estudo de Caso	87
5.3	Discussão	89
5.3.1	Complexidade Computacional	89
5.3.2	Uma hierarquia de comunicação multi-usuário no canal somador sobre corpos finitos	90
5.3.3	Analogia com o DS-CDMA	92
5.3.4	Requisitos de energia e análise num canal ruidoso	92
6	POLINÔMIOS DE CHEBYSHEV: UMA NOVA DEFINIÇÃO SOBRE CORPOS FINITOS E UM ALGORITMO PARA MULTIPLICAÇÃO POLINOMIAL	97
6.1	Polinômios de Chebyshev sobre Corpos Finitos Primos	98
6.1.1	A função co-seno inversa sobre corpos finitos	98
6.1.2	Polinômios de Chebyshev sobre corpos finitos primos: definição	101
6.2	Algoritmo de Cifragem de Chave Pública Baseado em Polinômios de Chebyshev sobre Corpos Finitos	102
6.2.1	Análise da segurança do algoritmo	103
6.3	Multiplificação de Polinômios na Forma de Chebyshev Baseada no Algoritmo de Karatsuba	106
6.3.1	Multiplificação de Polinômios na Forma de Chebyshev	107
6.3.2	Multiplificação de Polinômios na Forma de Chebyshev Baseada no Algoritmo de Karatsuba	108
6.3.3	Discussão	119
7	CONCLUSÕES	123
7.1	Trabalhos futuros	125
	REFERÊNCIAS	128
	Apêndice A PROVAS DOS TEOREMAS 2.1 A 2.16	138
A.1	Demonstração do Teorema 2.1 (FFCT- $1e^{-1}$)	138
A.2	Demonstração do Teorema 2.2 (FFCT- $2e^{-1}$)	140
A.3	Demonstração do Teorema 2.3 (FFCT- $3e^{-1}$)	140
A.4	Demonstração do Teorema 2.4 (FFCT- $4e^{-1}$)	141
A.5	Demonstração do Teorema 2.5 (FFCT- $1o^{-1}$)	142
A.6	Demonstração do Teorema 2.6 (FFCT- $2o^{-1}$)	143
A.7	Demonstração do Teorema 2.7 (FFCT- $3o^{-1}$)	143
A.8	Demonstração do Teorema 2.8 (FFCT- $4o^{-1}$)	145
A.9	Demonstração do Teorema 2.9 (FFST- $1e^{-1}$)	145

A.10	Demonstração do Teorema 2.10 (FFST-$2e^{-1}$)	146
A.11	Demonstração do Teorema 2.11 (FFST-$3e^{-1}$)	147
A.12	Demonstração do Teorema 2.12 (FFST-$4e^{-1}$)	148
A.13	Demonstração do Teorema 2.13 (FFST-$1o^{-1}$)	148
A.14	Demonstração do Teorema 2.14 (FFST-$2o^{-1}$)	149
A.15	Demonstração do Teorema 2.15 (FFST-$3o^{-1}$)	149
A.16	Demonstração do Teorema 2.16 (FFST-$4o^{-1}$)	150
Apêndice B PROVAS DAS PROPOSIÇÕES 4.4 E 4.9		152
B.1	Demonstração da Proposição 4.4	152
B.2	Demonstração da Proposição 4.9	155

CAPÍTULO I

INTRODUÇÃO

EM ENGENHARIA, as transformadas definidas sobre corpos finitos têm sido empregadas em diversas aplicações. Áreas como Processamento de Sinais e Códigos Corretores de Erros são beneficiadas pelo uso destas ferramentas, que proporcionam vantagens relacionadas à precisão e complexidade computacional [1], [2], [3]. Estes aspectos englobam fatores como o uso da aritmética de ponto-fixado para a realização de cálculos e a consequente precisão infinita com que os mesmos são efetuados, uma vez que, diferentemente da aritmética de ponto-flutuante, arredondamentos não são necessários [4], [5], [6].

A transformada de corpo finito mais conhecida é a de Fourier (FFFT, *finite field Fourier transform*), que foi introduzida por Pollard em 1971 em sua versão numérica [7]. As chamadas transformadas numéricas (NTTs, *number theoretic transforms*) realizam uma espécie de mapeamento de um vetor com componentes em $\text{GF}(p)$ num vetor transformado cujas componentes também estão em $\text{GF}(p)$, o campo de Galois de ordem p . Esta transformação possui propriedades análogas às da transformada discreta de Fourier (DFT, *discrete Fourier transform*), tais como linearidade, deslocamento no tempo e convolução cíclica [8]. Esta última possui uma especial importância, sendo útil para calcular o resultado de uma convolução linear entre uma determinada informação em tempo discreto e um filtro [2], [9], [10], [11], [12]. Até os dias atuais, o uso da propriedade da convolução cíclica da FFFT associado às técnicas de *overlap-add* e de *overlap-save* tem fundamentado uma grande quantidade de trabalhos relacionados à filtragem FIR de seqüências e de imagens [13], [3].

A convolução cíclica também pode ser utilizada em multiplicações numéricas que re-

queiram alta precisão (multiplicações com multi-precisão) [14]. Neste caso, cada número é escrito de uma maneira que permite que os mesmos sejam segmentados em blocos interpretados como polinômios. Cada bloco do polinômio produto pode ser obtido a partir de uma convolução linear entre seqüências cujas componentes são coeficientes dos respectivos polinômios fatores [15]. Naturalmente, para blocos de comprimento muito grande, a convolução no domínio da transformada é mais eficiente que outros algoritmos para o cálculo rápido de convoluções.

Um artifício semelhante ao utilizado na multiplicação com multi-precisão pode ser empregado na multiplicação de matrizes [14]. Uma explicação sucinta disso está no fato de que o problema de multiplicar duas matrizes $N \times N$ pode ser reformulado, sendo convertido num produto entre dois polinômios. O cálculo deste produto polinomial é realizado como descrito anteriormente.

A FFFT também tem sido parte integrante de algoritmos para solucionar de modo rápido sistemas Toeplitz [16]. Tais sistemas são regidos por equações da forma

$$Px = y, \tag{1.1}$$

em que P é uma matriz Toeplitz, ou seja, uma matriz $(n + 1) \times (n + 1)$ cujos elementos ao longo de qualquer diagonal (da esquerda para a direita) são iguais. Sua solução é importante em aplicações como predição linear, filtragem adaptativa, projeto de filtros etc. [17]. Para ilustrar a utilidade da FFFT neste cenário, é válido mencionar o fato que algoritmos para solucionar equações da forma de 1.1 baseados nesta ferramenta possuem complexidade $O(n \log_2 n)$; por outro lado, técnicas clássicas como a Gaussiana e a de Cholesky possuem complexidade $O(n^3)$ [14].

No contexto de Códigos Corretores de Erros, a transformada de Fourier de corpo finito pode ser utilizada, por exemplo, para descrever códigos de bloco [1]. Neste caso, o problema da codificação para controle de erros em sistemas de comunicação digital é analisado sob o prisma da FFFT definida apropriadamente como um mapeamento relacionando vetores com componentes em corpos finitos $GF(p^m)$, $m > 1$ [18]. Esta abordagem beneficia questões como codificação, decodificação e limites de desempenho, e fornece alternativas para a resolução de problemas de caráter teórico e de caráter prático. Com base nisso, é possível descrever famílias importantes de códigos de bloco lineares via FFFT e realizar decodificação algébrica no domínio da frequência.

Além da FFFT, outras transformadas de corpos finitos têm sido propostas, o que tem

contribuído para a ampliação do conhecimento a respeito destas ferramentas e, conseqüentemente, de suas potenciais aplicações. Um exemplo disso é a transformada de Hartley de corpo finito (FFHT, *finite field Hartley transform*), para a qual se propôs aplicações no projeto de sistemas de multiplexação digital, em sistemas de múltiplo acesso e no espalhamento espectral multinível de seqüências [8], [19], [20], [21], [22], [23], [24]. Outro exemplo é a transformada de wavelet de corpo finito (FFWT, *finite field wavelet transform*) [25]. Esta ferramenta atua no sentido de estender a aplicabilidade das wavelets, que é mais comum na representação de sinais, ao campo dos códigos corretores de erro. Mostra-se que, utilizando a FFWT, é possível implementar de modo eficiente codificadores e geradores de síndrome para códigos de bloco [26]. Além disso, têm sido propostas aplicações para outros tipos de códigos e em Criptografia [27], [28].

Uma outra importante classe de transformadas, até então definida apenas sobre o conjunto dos números reais, é a das transformadas trigonométricas discretas (DTTs, *discrete trigonometric transforms*). As funcionalidades das transformadas discretas do co-seno (DCT, *discrete cosine transform*) e do seno (DST, *discrete sine transform*) continuam sendo estudadas, fato refletido na diversidade de artigos em que são propostos sistemas e técnicas baseados nas mesmas [29], [30], [31], [32], [33], [34].

O objetivo principal do presente trabalho é introduzir a família das transformadas trigonométricas sobre corpos finitos (FFTTs, *finite field trigonometric transforms*) e estudar o seu potencial de aplicações. É relevante mencionar que, como acontece com outras transformadas sobre corpos finitos, a diversidade de problemas de Engenharia em que a DCT e a DST possuem destacada atuação servem apenas como motivação para estender o conceito das mesmas ao contexto de corpos finitos. Desta maneira, pretende-se deixar claro que a transformada do co-seno e a do seno sobre corpos finitos, denotadas respectivamente por FFCT (*finite field cosine transform*) e FFST (*finite field sine transform*), não são oferecidas como substitutas imediatas da DCT e da DST em cenários em que a importância dessas encontra-se sedimentada.

A base teórica das ferramentas matemáticas de corpos finitos, particularmente a das transformadas, apóia-se em conceitos que, normalmente, requerem um ponto de vista independente da convencional idéia de *tempo-freqüência* [8]. Pensando desta forma, busca-se enxergar com maior naturalidade as vantagens e as restrições que caracterizam tais ferramentas, e impedir que as mesmas sejam apresentadas como uma mera analogia do que já existe para os números reais.

1.1 Conteúdo da Tese

Após esta introdução, no Capítulo 2, são definidos os 16 pares de FFTTs. Inicialmente, este capítulo aborda alguns pontos que justificam a realização deste trabalho, apresentando o estado da arte em que se encontram as transformadas trigonométricas. São revisados alguns conceitos a respeito da trigonometria sobre corpos finitos e enunciados quatro lemas que fundamentam a posterior definição das FFTTs. Também são introduzidas versões normalizadas destas transformadas e apresentados exemplos das mesmas.

No Capítulo 3, são apresentadas as propriedades das FFTTs, dentre as quais se destaca a convolução simétrica. Esta operação, que provê uma maneira sistemática de convoluir a resposta ao impulso de filtros FIR de fase linear com seqüências simetricamente estendidas, é utilizada na filtragem digital de imagens, aplicação que se discute na parte final do capítulo. Com base na linearidade das transformadas propostas, uma técnica de marca d'água digital frágil no domínio da FFCT é também apresentada e discutida.

No Capítulo 4, investiga-se a auto-estrutura das matrizes de transformação das FFTT. Para isso, interpreta-se o cálculo da transformada de um vetor ou seqüência como uma multiplicação matricial. Daí, para os principais tipos de FFCT e FFST, são discutidas proposições e conjecturas que determinam o número de autovalores de cada matriz de transformação, bem como procedimentos para construir seus respectivos autovetores. Em seguida, discute-se o uso das FFTT na formatação de distribuições de probabilidade sobre os inteiros e o potencial de aplicação que esta operação possui no campo da Criptografia.

O Capítulo 5, que está fortemente relacionado ao seu antecessor, propõe um procedimento para “separação cega” de seqüências baseado na auto-estrutura das FFTT. Nesse contexto, faz-se um paralelo com a recuperação de seqüências enviadas pelos usuários simultâneos de um canal síncrono e aditivo sobre corpos finitos. No cenário proposto neste trabalho, as seqüências de usuário são autovetores de uma matriz de transformação específica, que, por estarem em auto-espacos ortogonais entre si, podem ser separados sem que se precise conhecer qualquer outra informação. Esquemas com 2 e 4 usuários baseados no procedimento proposto são apresentados e suas principais características são discutidas.

No Capítulo 6, uma nova definição para polinômios de Chebyshev sobre corpos finitos é proposta. Nesse ponto, destaca-se a introdução da função co-seno inversa sobre corpos finitos, a qual fundamenta uma inédita definição trigonométrica para os polinômios mencionados. Algumas propriedades desses polinômios são estudadas e a aplicação dos mesmos

à Criptografia de chave pública é examinada. Na parte final deste capítulo, apresenta-se um algoritmo para multiplicação de polinômios na forma de Chebyshev. O método desenvolvido baseia-se no algoritmo de Karatsuba e também é válido para polinômios sobre os números reais. Discutem-se diversos aspectos do procedimento proposto e comparam-se suas características às de outros métodos para realizar a mesma operação.

No Capítulo 7, são apresentadas as conclusões desta tese e apontados temas para investigações que dêem continuidade ao que foi desenvolvido neste trabalho.

1.2 Contribuições

A maior parte do material contido nos capítulos mencionados resultou em trabalhos já publicados ou que, atualmente, encontram-se em preparação. A seguir, são apresentadas de maneira objetiva as contribuições desta tese.

- ▷ Introdução de 3 novos lemas fundamentais envolvendo a função co-seno sobre corpos finitos [35]
- ▷ Definição de 14 novas transformadas trigonométricas sobre corpos finitos, o que tornou completa a família das FFTT, com 8 transformadas do co-seno (FFCT) e 8 transformadas do seno (FFST). Versões unitárias dessas transformadas também foram introduzidas [35].
- ▷ Estudo das propriedades das FFTT (linearidade, deslocamento no tempo, teorema de Parseval, relações com a transformada de Fourier sobre corpos finitos e convolução simétrica).
- ▷ Proposta de uma técnica de marca d'água digital frágil no domínio da transformada do co-seno sobre corpos finitos [36]
- ▷ Estudo da aplicação da propriedade de convolução simétrica das FFTT na filtragem digital de imagens [35]
- ▷ Definição da transformada de Fourier sobre corpos finitos generalizada e estudo de sua auto-estrutura [37]
- ▷ Estudo da auto-estrutura das transformadas trigonométricas sobre corpos finitos [37]
- ▷ Introdução das FFTT como ferramentas para a formatação de distribuições de probabilidade sobre os inteiros e propostas de aplicação deste procedimento em Criptosistemas [38]
- ▷ Investigação de métodos para separação cega de seqüências baseados na auto-estrutura das FFTT. Nesse contexto, fez-se um paralelo com esquemas de comunicação multiusuário, a

partir do qual diversos aspectos foram analisados [37]

- ▷ Definição da função co-seno inverso sobre corpos finitos [39]
- ▷ Introdução de uma nova definição para polinômios de Chebyshev sobre corpos finitos e estudo de algumas de suas propriedades [39]
- ▷ Análise de um algoritmo criptográfico de chave-pública baseado nos polinômios de Chebyshev sobre corpos finitos [39]
- ▷ Desenvolvimento de um método para multiplicação de polinômios na forma de Chebyshev baseado no algoritmo de Karatsuba (aplicável tanto no contexto de corpos finitos, quando no dos números reais) [40]

CAPÍTULO 2

TRANSFORMADAS TRIGONOMÉTRICAS SOBRE CORPOS FINITOS

AS TRANSFORMADAS discretas definidas sobre corpos finitos e sobre corpos infinitos são bastante conhecidas. A transformada discreta de Fourier (DFT – *discrete Fourier transform*), particularmente, tornou-se um recurso destacado em diversas aplicações da Engenharia [41]. Outra transformada discreta relevante é a do co-seno (DCT – *discrete cosine transform*), que, juntamente com a do seno (DST – *discrete sine transform*), compõe a família das transformadas trigonométricas discretas (DTT – *discrete trigonometric transforms*) e também possui importantes funcionalidades [42], [43]. Na área de Processamento de Imagem, por exemplo, a DCT é especialmente atrativa por apresentar propriedades de compactação bem mais eficientes que a DFT [44]. Por este motivo, a DCT tem sido a base para padrões de compressão, como o JPEG – *Joint Photographic Experts Group* – e o MPEG – *Moving Picture Experts Group* – e para diversas técnicas de proteção de informação digital [45], [46], [30].

Ainda que discretas no domínio do tempo, a DFT e as DTT possuem coeficientes com amplitudes pertencentes a corpos infinitos, podendo, portanto, ser interpretadas como “transformadas analógicas”. Por outro lado, conforme mencionado no capítulo introdutório desta tese, as amplitudes dos coeficientes das transformadas definidas sobre corpos finitos são discretas. Assim, as mesmas podem ser caracterizadas como legítimas “transformadas digitais”. Esse aspecto, ao qual uma série de interessantes possibilidades está atrelada, motivou a in-

rodução da transformada do co-seno sobre corpos finitos [47]. Posteriormente, foi também definida a transformada do seno sobre corpos finitos [48].

Neste capítulo, pela primeira vez na literatura, são introduzidos os 14 pares restantes de transformadas do co-seno e do seno sobre corpos finitos. Assim, constitui-se a família das chamadas transformadas trigonométricas sobre corpos finitos (FFTTs - *finite field trigonometric transforms*). Isso é feito com base nas definições e teoremas apresentados a seguir.

2.1 Preliminares Matemáticos

2.1.1 Números Complexos sobre Corpos Finitos

Definição 2.1 O conjunto de inteiros gaussianos sobre $GF(p)$ é o conjunto $GI(p) = \{a + jb, a, b \in GF(p)\}$, em que p é um número primo tal que $j^2 = -1$ é um resíduo não-quadrático sobre $GF(p)$.

Apenas números primos da forma $p \equiv 3 \pmod{4}$ atendem ao requisito exposto na definição acima [49]. O corpo de extensão $GF(p^2)$ é isomórfico à estrutura “complexa” $GI(p)$ [15], cujos elementos, denominados números complexos sobre corpos finitos, possuem uma parte “real” $a = \Re\{\zeta\}$ e uma parte “imaginária” $b = \Im\{\zeta\}$. Também é possível definir inteiros gaussianos sobre $GF(q)$, $q = p^r$, com r sendo um inteiro positivo e p um número primo ímpar. Para isso, basta selecionar um resíduo não-quadrático apropriado [8]. Nesta tese, entretanto, enfatiza-se o desenvolvimento de ferramentas sobre corpos finitos primos e suas aplicações, ficando em segundo plano a consideração de conceitos relacionados a corpos de extensão.

Definição 2.2 (conjunto unimodular) O conjunto unimodular de $GI(p)$, denotado por G_1 , é o conjunto de elementos $\zeta = (a + jb) \in GI(p)$, tais que $a^2 + b^2 \equiv 1 \pmod{p}$.

Proposição 2.1 A estrutura $\langle G_1, \bullet \rangle$ é um grupo cíclico de ordem $(p + 1)$ [50].

2.1.2 Trigonometria em Corpos Finitos

Nesta subseção, os principais conceitos relacionados à trigonometria sobre corpos finitos são revisados. Originalmente, tais conceitos foram propostos por Campello de Souza *et al.*, como requisitos para se definir a transformada de Hartley sobre corpos finitos (FFHT - *finite field Hartley transform*) [8]. No que segue, o símbolo \triangleq denota “igual por definição”.

Definição 2.3 Seja ζ um elemento não-nulo de $GI(p)$, $p \equiv 3 \pmod{4}$, com ordem multiplicativa denotada por $\text{ord}(\zeta)$. As funções trigonométricas co-seno e seno sobre corpos finitos relacionadas a ζ são

calculadas modulo p , respectivamente, por

$$\cos_{\zeta}(x) \triangleq (2^{-1} \bmod p)(\zeta^x + \zeta^{-x}) \quad (2.1)$$

e

$$\operatorname{sen}_{\zeta}(x) \triangleq (2^{-1} \bmod p)(\zeta^x - \zeta^{-x})/j, \quad (2.2)$$

$x = 0, 1, \dots, \operatorname{ord}(\zeta) - 1$ [8], [51].

Na definição acima, usa-se uma notação ligeiramente diferente daquela estabelecida em [8]¹. Entretanto, independentemente desse fato, as funções trigonométricas sobre corpos finitos conservam propriedades similares às das transformadas trigonométricas usuais, tais como *círculo unitário* e *adição de arcos* [8].

Lema 2.1 *Se $\zeta = a + jb$ é um elemento unimodular de $GF(p)$, então $\cos_{\zeta}(x) = \Re\{\zeta^x\}$ e $\operatorname{sen}_{\zeta}(x) = \Im\{\zeta^x\}$, $x = 0, 1, \dots, \operatorname{ord}(\zeta) - 1$ [51].*

Demonstração: Usando a Definição 2.3 e fazendo $\zeta^x = c + dj$, $c, d \in GF(p)$, tem-se

$$\cos_{\zeta}(x) = \frac{(c + dj) + (c + dj)^{-1}}{2}.$$

Devido à Proposição 2.1, $\zeta^x = c + dj$ é também unimodular e $(c + dj)^{-1} = (c + dj)^* = c - dj$, com $(\cdot)^*$ denotando o complexo conjugado. Portanto, a última equação pode ser reescrita como

$$\cos_{\zeta}(x) = \frac{(c + dj) + (c - dj)}{2} = c = \Re\{\zeta^x\}.$$

Aplicando argumentos análogos, verifica-se que $\operatorname{sen}_{\zeta}(x) = d = \Im\{\zeta^x\}$. ■

2.2 Lemas Fundamentais

Para definir as transformadas do co-seno e do seno usando funções trigonométricas sobre corpos finitos, são necessários os lemas apresentados nesta seção. É importante ressaltar que o Lema 2.2 foi proposto em [47] e que os Lemas 2.3, 2.4 e 2.5 são contribuições desta tese. No que segue, \mathbb{N} denota o conjunto dos números naturais.

¹Originalmente, as funções co-seno e seno sobre corpos finitos estão relacionadas a $\angle \zeta^i$, o “arco” de ζ^i , e são chamadas funções k -trigonométricas. As mesmas são computadas por $\cos_k(\angle \zeta^i) = (\zeta^{ki} + \zeta^{-ki})/2$ e $\operatorname{sen}_k(\angle \zeta^i) = (\zeta^{ki} - \zeta^{-ki})/2j$, para $i, k = 0, 1, \dots, \operatorname{ord}(\zeta) - 1$; os parâmetros i e k estão associados, respectivamente, aos domínios do “tempo” e da “frequência” da transformada de Hartley sobre corpos finitos.

Lema 2.2 Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então

$$A = \sum_{k=1}^{N-1} \cos_{\zeta}(ki) = \begin{cases} N-1, & i = t2N, t \in \mathbb{N}, \\ -1, & i \text{ par } (\neq t2N, t \in \mathbb{N}), \\ 0, & i \text{ ímpar}. \end{cases} \quad (2.3)$$

Demonstração: Por definição

$$A = \sum_{k=1}^{N-1} \cos_{\zeta}(ki) = \frac{1}{2} \sum_{k=1}^{N-1} (\zeta^{ki} + \zeta^{-ki}).$$

Observando que ζ possui ordem igual a $2N$, tem-se $\zeta^{kt2N} = 1, t \in \mathbb{N}, k = 1 \dots N-1$. Essa informação é suficiente para validar a primeira condição do lema. Caso contrário, a equação acima pode ser escrita da seguinte forma:

$$A = \frac{1}{2} \left[\frac{\zeta^i(\zeta^{i(N-1)} - 1)}{\zeta^i - 1} + \frac{\zeta^{-i}(\zeta^{-i(N-1)} - 1)}{\zeta^{-i} - 1} \right].$$

Como $\zeta^N = -1$, multiplicando o segundo termo por $(-\zeta^{-i})$, obtém-se

$$A = \frac{1}{2} \left[\frac{(-1)^i - \zeta^i}{\zeta^i - 1} + \frac{1 - (-1)^i \zeta^i}{\zeta^i - 1} \right].$$

Portanto, para i par ($\neq t2N$),

$$A = \frac{1}{2} \left[\frac{1 - \zeta^i + 1 - \zeta^i}{\zeta^i - 1} \right] = -1,$$

e, para i ímpar,

$$A = \frac{1}{2} \left[\frac{-1 - \zeta^i + 1 + \zeta^i}{\zeta^i - 1} \right] = 0. \quad \blacksquare$$

Lema 2.3 Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então

$$A = \sum_{k=1}^{N-1} \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right) = \begin{cases} (-1)^t(N-1), & i = t2N, t \in \mathbb{N}, \\ -\cos_{\zeta}(i/2), & \text{caso contrário}. \end{cases} \quad (2.0)$$

Demonstração: Por definição

$$A = \sum_{k=1}^{N-1} \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right) = \frac{1}{2} \sum_{k=1}^{N-1} (\zeta^{(k+\frac{1}{2})i} + \zeta^{-(k+\frac{1}{2})i}).$$

Observando que ζ possui ordem igual a $2N$, tem-se $\zeta^{\frac{t2N}{2}} = 1$, se t for par, e $\zeta^{\frac{t2N}{2}} = -1$, se t for ímpar. Esta informação é suficiente para validar a primeira condição do lema. Caso contrário,

$$A = \frac{1}{2} \left[\zeta^{\frac{i}{2}} \left(\frac{\zeta^i(\zeta^{i(N-1)} - 1)}{\zeta^i - 1} \right) + \zeta^{-\frac{i}{2}} \left(\frac{\zeta^{-i}(\zeta^{-i(N-1)} - 1)}{\zeta^{-i} - 1} \right) \right].$$

Como $\zeta^N = -1$, multiplicando e dividindo o segundo termo da equação acima por $(-\zeta^{-i})$, obtém-se

$$A = \frac{1}{2} \left[\zeta^{\frac{i}{2}} \left(\frac{(-1)^i - \zeta^i}{\zeta^i - 1} \right) + \zeta^{-\frac{i}{2}} \left(\frac{1 - (-1)^i \zeta^i}{\zeta^i - 1} \right) \right].$$

Efetuada a multiplicação de $\zeta^{\frac{i}{2}}$ e de $\zeta^{-\frac{i}{2}}$ pelos termos internos aos parênteses e realizando algumas manipulações, a última equação pode ser reescrita da seguinte forma:

$$A = \frac{1}{2} \left[\frac{\zeta^{\frac{i}{2}}(-1)^i - \zeta^{\frac{3i}{2}} + \zeta^{-\frac{i}{2}} - \zeta^{\frac{i}{2}}(-1)^i}{\zeta^i - 1} \right] = \frac{1}{2} \left[\frac{-\zeta^{\frac{3i}{2}} + \zeta^{-\frac{i}{2}}}{\zeta^i - 1} \right].$$

Fatorando o numerador da equação acima, obtém-se:

$$A = \frac{1}{2} \left[\frac{(-1) \left(\zeta^{\frac{i}{2}} + \zeta^{-\frac{i}{2}} \right) (\zeta^i - 1)}{\zeta^i - 1} \right] = -\cos_{\zeta}(i/2).$$

■

Lema 2.4 Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N - 1$, então

$$A = \sum_{k=1}^{N-1} \cos_{\zeta}(ki) = \begin{cases} N - 1, & i = t(2N - 1), t \in \mathbb{N}, \\ -1/2, & \text{caso contrário.} \end{cases} \quad (2.1)$$

Demonstração: Por definição

$$A = \sum_{k=1}^{N-1} \cos_{\zeta}(ki) = \frac{1}{2} \sum_{k=1}^{N-1} (\zeta^{ki} + \zeta^{-ki}).$$

Observando que ζ possui ordem igual a $2N - 1$, tem-se $\zeta^{kt(2N-1)} = 1$, $t \in \mathbb{N}$, $k = 1 \dots N - 1$. Esta informação é suficiente para validar a primeira condição do lema. Caso contrário, a equação acima pode ser escrita da seguinte forma:

$$A = \frac{1}{2} \left[\frac{\zeta^i(\zeta^{i(N-1)} - 1)}{\zeta^i - 1} + \frac{\zeta^{-i}(\zeta^{-i(N-1)} - 1)}{\zeta^{-i} - 1} \right].$$

Multiplicando e dividindo o segundo termo por $(-\zeta^{-i})$, obtém-se

$$A = \frac{1}{2} \left[\frac{\zeta^i(\zeta^{i(N-1)} - 1)}{\zeta^i - 1} + \frac{1 - \zeta^{-i(N-1)}}{\zeta^i - 1} \right].$$

Observando que $\zeta^{(2N-1)} = 1$, pode-se reescrever a última equação da seguinte forma:

$$A = \frac{1}{2} \left[\frac{\zeta^{iN} - \zeta^i + 1 - \zeta^{iN} \zeta^{-i(2N-1)}}{\zeta^i - 1} \right],$$

que se resume a

$$A = \frac{1}{2} \left[\frac{-\zeta^i + 1}{\zeta^i - 1} \right] = -\frac{1}{2}.$$

■

Lema 2.5 (lema do $k + 1/2$ -cos modificado) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N - 1$, então

$$A = \sum_{k=1}^{N-2} \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right) = \begin{cases} (-1)^t (N - 2), & i = t(2N - 1), t \in \mathbb{N}, \\ -1/2 - \cos_{\zeta} (i/2), & i \text{ par } (\neq t(2N - 1), t \text{ par}), \\ 1/2 - \cos_{\zeta} (i/2), & i \text{ ímpar } (\neq t(2N - 1), t \text{ ímpar}). \end{cases} \quad (2.2)$$

Demonstração: Por definição

$$A = \sum_{k=1}^{N-2} \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right) = \frac{1}{2} \sum_{k=1}^{N-2} \left(\zeta^{(k+\frac{1}{2})i} + \zeta^{-(k+\frac{1}{2})i} \right).$$

Observando que ζ possui ordem igual a $2N - 1$, tem-se $\zeta^{\frac{t(2N-1)}{2}} = 1$, se t for par, e $\zeta^{\frac{t(2N-1)}{2}} = -1$, se t for ímpar. Esta informação é suficiente para validar a primeira condição do lema. Caso contrário,

$$A = \frac{1}{2} \left[\zeta^{\frac{i}{2}} \left(\frac{\zeta^i (\zeta^{i(N-2)} - 1)}{\zeta^i - 1} \right) + \zeta^{-\frac{i}{2}} \left(\frac{\zeta^{-i} (\zeta^{-i(N-2)} - 1)}{\zeta^{-i} - 1} \right) \right].$$

Multiplicando e dividindo o segundo termo por $(-\zeta^{-i})$, obtém-se

$$A = \frac{1}{2} \left[\zeta^{\frac{i}{2}} \left(\frac{\zeta^i (\zeta^{i(N-2)} - 1)}{\zeta^i - 1} \right) + \zeta^{-\frac{i}{2}} \left(\frac{1 - \zeta^{-i(N-2)}}{\zeta^i - 1} \right) \right].$$

Efetuada a multiplicação de $\zeta^{\frac{i}{2}}$ e de $\zeta^{-\frac{i}{2}}$ pelos termos internos aos parênteses e realizando algumas manipulações, a equação acima pode ser reescrita da seguinte forma:

$$A = \frac{1}{2} \left[\frac{\zeta^{i(N-\frac{1}{2})} - \zeta^{\frac{3i}{2}} + \zeta^{-\frac{i}{2}} - \zeta^{-i(N-\frac{3}{2})}}{\zeta^i - 1} \right].$$

Observando que $\zeta^{N-\frac{1}{2}} = -1$, pode-se agrupar os termos do numerador da equação acima e obter o resultado a seguir:

$$A = \frac{1}{2} \left[\frac{-(-1)^i (\zeta^i - 1) - \zeta^{-\frac{i}{2}} (\zeta^i + 1) (\zeta^i - 1)}{\zeta^i - 1} \right] = \frac{1}{2} \left[(-1)^{i+1} - \left(\zeta^{\frac{i}{2}} + \zeta^{-\frac{i}{2}} \right) \right].$$

Portanto, para i par,

$$A = -1/2 - \cos_{\zeta} (i/2)$$

e, para i ímpar,

$$A = 1/2 - \cos_{\zeta} (i/2).$$

■

2.3 Definição das Transformadas

Sobre o corpo dos números reais, podem-se definir 16 versões de transformadas trigonométricas usando co-senos e senos [52]. Uma das maneiras de construir essas transformadas é considerar uma seqüência $\mathbf{x} = (x_i)$ de comprimento N , $N - 1$ ou $N + 1$, e estendê-la para a esquerda e para a direita segundo determinado critério de simetria.

Combinando uma simetria par ou ímpar, com uma simetria cujo eixo se localiza sobre uma amostra ou sobre o ponto médio entre duas amostras, é possível criar 4 tipos de simetria (figura 2.1): WS (*whole-sample symmetry*), WA (*whole-sample antisymmetry*), HS (*half-sample symmetry*) e HA (*half-sample antisymmetry*). O procedimento de extensão de \mathbf{x} gera uma nova seqüência $\hat{\mathbf{x}} = (\hat{x}_i)$. Os coeficientes da respectiva transformada trigonométrica discreta de \mathbf{x} são, então, obtidos a partir da DFT de um trecho de comprimento $2N$ ou $2N - 1$ de $\hat{\mathbf{x}}$.

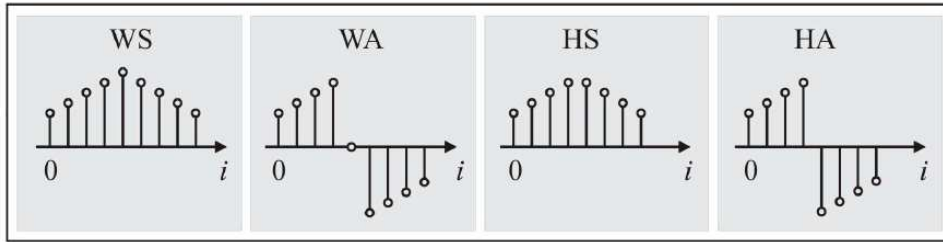


Figura 2.1: Tipos de simetria de uma seqüência.

O procedimento descrito requer o uso de núcleos de co-senos ou senos com ordens $2N$ ou $2N - 1$, de acordo com o comprimento de $\hat{\mathbf{x}}$ [47]. Este é o ponto chave para definir as FFTT. A construção de transformadas trigonométricas aplicáveis a um vetor \mathbf{x} de comprimento N , $N - 1$ ou $N + 1$, com componentes em $\text{GF}(p)$, envolve núcleos específicos de co-senos e senos sobre corpos finitos. A ordem desses núcleos é determinada pela ordem multiplicativa do número ζ , sobre o qual essas funções são calculadas.

Com base na discussão acima e usando as funções peso, dadas por

$$\beta_r \equiv \begin{cases} 2^{-1}(\text{mod } p), & r = 0 \text{ ou } N, \\ 1, & r = 1, 2, \dots, N - 1 \end{cases} \quad (2.3)$$

e

$$\gamma_r \equiv \begin{cases} 1, & r = 0, 1, \dots, N - 2, \\ 2^{-1}(\text{mod } p), & r = N - 1, \end{cases} \quad (2.4)$$

é possível introduzir as transformadas trigonométricas sobre corpos finitos. Cada transfor-

mada é identificada como sendo do co-seno (FFCT, *finite field cosine transform*) ou do seno (FFST, *finite field sine transform*), par (*e*) ou ímpar (*o*), e do tipo 1, 2, 3 ou 4.

Conforme mencionado anteriormente, a existência das FFTT é baseada nos lemas 2.2 a 2.5. De forma mais específica, cada lema é utilizado na validação de quatro dentre as dezesseis transformadas. Nas definições e teoremas apresentados a seguir, é importante observar que o inverso de N e o de $2N - 1$ são calculados (mod p). As provas dos Teoremas 2.1 a 2.16 são apresentadas no Apêndice A.

Definição 2.4 (FFCT-1e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^N 2\beta_i x_i \cos_{\zeta}(ki). \quad (2.5)$$

Teorema 2.1 (FFCT-1e⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N$, $x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=0}^N \beta_k X_k \cos_{\zeta}(ki). \quad (2.6)$$

Definição 2.5 (FFCT-2e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N - 1$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2x_i \cos_{\zeta} \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.7)$$

Teorema 2.2 (FFCT-2e⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N - 1$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=0}^{N-1} \beta_k X_k \cos_{\zeta} \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.8)$$

Definição 2.6 (FFCT-3e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N - 1$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2\beta_i x_i \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.9)$$

Teorema 2.3 (FFCT-3e⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=0}^{N-1} X_k \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.10)$$

Definição 2.7 (FFCT-4e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2x_i \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.11)$$

Teorema 2.4 (FFCT-4e⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=0}^{N-1} X_k \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.12)$$

Definição 2.8 (FFCT-1o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N-1$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N, x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N, X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^N 2\beta_i x_i \cos_{\zeta}(ki). \quad (2.13)$$

Teorema 2.5 (FFCT-1o⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N, X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N, x_i \in GF(p)$, de elementos

$$x_i = 2(2N-1)^{-1} \sum_{k=0}^N \beta_k X_k \cos_{\zeta}(ki). \quad (2.14)$$

Definição 2.9 (FFCT-2o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N-1$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2\gamma_i x_i \cos_{\zeta} \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.15)$$

Teorema 2.6 (FFCT-2o⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, de elementos

$$x_i = 2(2N-1)^{-1} \sum_{k=0}^{N-1} \beta_k X_k \cos_{\zeta} \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.16)$$

Definição 2.10 (FFCT-3o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N - 1$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N - 1$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2\beta_i x_i \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.17)$$

Teorema 2.7 (FFCT-3o⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N - 1$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in GF(p)$, de elementos

$$x_i = 2(2N - 1)^{-1} \sum_{k=0}^{N-1} \gamma_k X_k \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.18)$$

Definição 2.11 (FFCT-4o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N - 1$, então a transformada do co-seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 2$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N - 2$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-2} 2x_i \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.19)$$

Teorema 2.8 (FFCT-4o⁻¹) A transformada do co-seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N - 2$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 2$, $x_i \in GF(p)$, de elementos

$$x_i = 2(2N - 1)^{-1} \sum_{k=0}^{N-2} X_k \cos_{\zeta} \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.20)$$

Definição 2.12 (FFST-1e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 1, 2, \dots, N - 1$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 1, 2, \dots, N - 1$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=1}^{N-1} 2x_i \sen_{\zeta}(ki). \quad (2.21)$$

Teorema 2.9 (FFST-1e⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 1, 2, \dots, N - 1$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 1, 2, \dots, N - 1$, $x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=1}^{N-1} X_k \sen_{\zeta}(ki). \quad (2.22)$$

Definição 2.13 (FFST-2e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k), k = 1, 2, \dots, N, X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2x_i \text{sen}_{\zeta} \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.23)$$

Teorema 2.10 (FFST-2e⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k), k = 1, 2, \dots, N, X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=1}^N \beta_k X_k \text{sen}_{\zeta} \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.24)$$

Definição 2.14 (FFST-3e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i), i = 1, 2, \dots, N, x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=1}^N 2\beta_i x_i \text{sen}_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.25)$$

Teorema 2.11 (FFST-3e⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i), i = 1, 2, \dots, N, x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=0}^{N-1} X_k \text{sen}_{\zeta} \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.26)$$

Definição 2.15 (FFST-4e) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2x_i \text{sen}_{\zeta} \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.27)$$

Teorema 2.12 (FFST-4e⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k), k = 0, 1, \dots, N-1, X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i), i = 0, 1, \dots, N-1, x_i \in GF(p)$, de elementos

$$x_i = N^{-1} \sum_{k=0}^{N-1} X_k \text{sen}_{\zeta} \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.28)$$

Definição 2.16 (FFST-1o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N-1$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i), i = 1, 2, \dots, N-1, x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k), k = 1, 2, \dots, N-1, X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=1}^{N-1} 2x_i \text{sen}_{\zeta}(ki). \quad (2.29)$$

Teorema 2.13 (FFST-1o⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 1, 2, \dots, N-1$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 1, 2, \dots, N-1$, $x_i \in GF(p)$, de elementos

$$x_i = 2(2N-1)^{-1} \sum_{k=1}^{N-1} X_k \text{sen}_\zeta(ki). \quad (2.30)$$

Definição 2.17 (FFST-2o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N-1$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N-2$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 1, 2, \dots, N-1$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-2} 2x_i \text{sen}_\zeta \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.31)$$

Teorema 2.14 (FFST-2o⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 1, 2, \dots, N-1$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N-2$, $x_i \in GF(p)$, de elementos

$$x_i = 2(2N-1)^{-1} \sum_{k=1}^{N-1} X_k \text{sen}_\zeta \left(k \left(i + \frac{1}{2} \right) \right). \quad (2.32)$$

Definição 2.18 (FFST-3o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N-1$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 1, 2, \dots, N-1$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N-2$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=1}^{N-1} 2x_i \text{sen}_\zeta \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.33)$$

Teorema 2.15 (FFST-3o⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N-2$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 1, 2, \dots, N-1$, $x_i \in GF(p)$, de elementos

$$x_i = 2(2N-1)^{-1} \sum_{k=0}^{N-2} X_k \text{sen}_\zeta \left(\left(k + \frac{1}{2} \right) i \right). \quad (2.34)$$

Definição 2.19 (FFST-4o) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N-1$, então a transformada do seno de corpo finito da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N-1$, $x_i \in GF(p)$, é a seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N-1$, $X_k \in GI(p)$, de elementos

$$X_k \triangleq \sum_{i=0}^{N-1} 2\gamma_i x_i \text{sen}_\zeta \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.35)$$

Teorema 2.16 (FFST-4o⁻¹) A transformada do seno de corpo finito inversa da seqüência $\mathbf{X} = (X_k)$, $k = 0, 1, \dots, N-1$, $X_k \in GI(p)$, é a seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N-1$, $x_i \in GF(p)$, de elementos

$$x_i = 2(2N-1)^{-1} \sum_{k=0}^{N-1} \gamma_k X_k \text{sen}_\zeta \left(\left(k + \frac{1}{2} \right) \left(i + \frac{1}{2} \right) \right). \quad (2.36)$$

De maneira geral, qualquer FFTT de uma seqüência ou vetor linha $\mathbf{x} = (x_i)$, $x_i \in \text{GF}(p)$, é um vetor coluna $\mathbf{X} = (X_k)$, $X_k \in \text{GI}(p)$, obtido pela equação matricial

$$\mathbf{X} = \mathbf{x} \times \mathbf{M}^T, \quad (2.37)$$

em que \mathbf{M} é a matriz de transformação. Os elementos dessa matriz são computados de acordo com as tabelas 2.1 e 2.2, as quais, resumidamente, apresentam a família das FFTT. Para simplificar, as matrizes de transformação do co-seno e as do seno são denotadas por \mathbf{FC} e \mathbf{FS} , respectivamente. Na segunda coluna de cada tabela, observa-se a forma como variam os índices i (linhas) e k (colunas), que determinam as dimensões de \mathbf{x} , \mathbf{M} e \mathbf{X} .

O cálculo da FFTT inversa de uma seqüência também pode ser representado através de uma equação matricial. As relações entre cada transformada e sua inversa são dadas pelas equações

$$\mathbf{FC}_1^{-1} = N'^{-1}(\text{mod } p)\mathbf{FC}_1 \quad (2.38)$$

$$\mathbf{FC}_2^{-1} = N'^{-1}(\text{mod } p)\mathbf{FC}_3 \quad (2.39)$$

$$\mathbf{FC}_3^{-1} = N'^{-1}(\text{mod } p)\mathbf{FC}_2 \quad (2.40)$$

$$\mathbf{FC}_4^{-1} = N'^{-1}(\text{mod } p)\mathbf{FC}_4 \quad (2.41)$$

$$\mathbf{FS}_1^{-1} = N'^{-1}(\text{mod } p)\mathbf{FS}_1 \quad (2.42)$$

$$\mathbf{FS}_2^{-1} = N'^{-1}(\text{mod } p)\mathbf{FS}_3 \quad (2.43)$$

$$\mathbf{FS}_3^{-1} = N'^{-1}(\text{mod } p)\mathbf{FS}_2 \quad (2.44)$$

$$\mathbf{FS}_4^{-1} = N'^{-1}(\text{mod } p)\mathbf{FS}_4, \quad (2.45)$$

em que $N' = 2N$, para as transformadas pares, e $N' = 2N - 1$, para as transformadas ímpares.

2.3.1 Versões Unitárias das FFTTs

Em muitas abordagens, é comum incorporar às expressões que definem as transformadas trigonométricas fatores de normalização que as tornam unitárias [44]. Neste caso, além de atenderem à condição de ortonormalidade, essas transformadas satisfazem a relação

$$\sum_i x_i^2 = \sum_k X_k^2, \quad (2.46)$$

em que $\mathbf{x} = (x_i)$, $x_i \in \text{GF}(p)$, e $\mathbf{X} = (X_k)$, $X_k \in \text{GI}(p)$, correspondem, respectivamente, a uma seqüência arbitrária e sua transformada.

Tabela 2.1: Transformadas do co-seno de corpo finito ($\zeta \in GI(p)$, $p \equiv 3(\text{mod } 4)$).

Elementos da matriz de transformação	Dimensão da matriz
$[\mathbf{FC}_{1e}]_{ik} = 2 \beta_i \cos_{\zeta}(ki)$	$i, k = 0, 1, \dots, N$
$[\mathbf{FC}_{2e}]_{ik} = 2 \cos_{\zeta}(k(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\mathbf{FC}_{3e}]_{ik} = 2 \beta_i \cos_{\zeta}((k + \frac{1}{2})i)$	$i, k = 0, 1, \dots, N - 1$
$[\mathbf{FC}_{4e}]_{ik} = 2 \cos_{\zeta}((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\mathbf{FC}_{1o}]_{ik} = 2 \beta_i \cos_{\zeta}(ki)$	$i, k = 0, 1, \dots, N - 1$
$[\mathbf{FC}_{2o}]_{ik} = 2 \gamma_i \cos_{\zeta}(k(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\mathbf{FC}_{3o}]_{ik} = 2 \beta_i \cos_{\zeta}((k + \frac{1}{2})i)$	$i, k = 0, 1, \dots, N - 1$
$[\mathbf{FC}_{4o}]_{ik} = 2 \cos_{\zeta}((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 2$

As definições de cada FFTT unitária podem ser obtidas diretamente de suas correspondentes versões não unitárias. Para isso, entretanto, é necessário definir as funções peso modificadas

$$\tilde{\beta}_r \equiv \begin{cases} \sqrt{2^{-1}} \pmod{p}, & r = 0 \text{ or } N, \\ 1, & r = 1, 2, \dots, N - 1 \end{cases} \quad (2.47)$$

e

$$\tilde{\gamma}_r \equiv \begin{cases} 1, & r = 0, 1, \dots, N - 2, \\ \sqrt{2^{-1}} \pmod{p}, & r = N - 1. \end{cases} \quad (2.48)$$

De maneira similar ao que foi descrito anteriormente, os elementos das matrizes de transformação das FFTT unitárias são computados de acordo com as Tabelas 2.3 e 2.4. Para simplificar, as matrizes de transformação do co-seno e as do seno unitárias são denotadas por $\widetilde{\mathbf{FC}}$ e $\widetilde{\mathbf{FS}}$.

Tabela 2.2: Transformadas do seno de corpo finito ($\zeta \in GI(p)$, $p \equiv 3(\text{mod } 4)$).

Elementos da matriz de transformação	Dimensão da matriz
$[\mathbf{FS}_{1e}]_{ik} = 2 \text{sen}_\zeta(ki)$	$i, k = 1, 2, \dots, N - 1$
$[\mathbf{FS}_{2e}]_{ik} = 2 \text{sen}_\zeta(k(i + \frac{1}{2}))$	$i = 0, 1, \dots, N - 1$ $k = 1, 2, \dots, N$
$[\mathbf{FS}_{3e}]_{ik} = 2 \beta_i \text{sen}_\zeta((k + \frac{1}{2})i)$	$i = 1, 2, \dots, N$ $k = 0, 1, \dots, N - 1$
$[\mathbf{FS}_{4e}]_{ik} = 2 \text{sen}_\zeta((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\mathbf{FS}_{1o}]_{ik} = 2 \text{sen}_\zeta(ki)$	$i, k = 1, 2, \dots, N - 1$
$[\mathbf{FS}_{2o}]_{ik} = 2 \text{sen}_\zeta(k(i + \frac{1}{2}))$	$i = 0, 1, \dots, N - 2$ $k = 1, 2, \dots, N - 1$
$[\mathbf{FS}_{3o}]_{ik} = 2 \text{sen}_\zeta((k + \frac{1}{2})i)$	$i = 1, 2, \dots, N - 1$ $k = 0, 1, \dots, N - 2$
$[\mathbf{FS}_{4o}]_{ik} = 2 \gamma_i \text{sen}_\zeta((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$

As relações entre cada transformada unitária e sua inversa são dadas pelas equações

$$\widetilde{\mathbf{FC}}_1^{-1} = \widetilde{\mathbf{FC}}_1 \quad (2.49)$$

$$\widetilde{\mathbf{FC}}_2^{-1} = \widetilde{\mathbf{FC}}_3 \quad (2.50)$$

$$\widetilde{\mathbf{FC}}_3^{-1} = \widetilde{\mathbf{FC}}_2 \quad (2.51)$$

$$\widetilde{\mathbf{FC}}_4^{-1} = \widetilde{\mathbf{FC}}_4 \quad (2.52)$$

$$\widetilde{\mathbf{FS}}_1^{-1} = \widetilde{\mathbf{FS}}_1 \quad (2.53)$$

$$\widetilde{\mathbf{FS}}_2^{-1} = \widetilde{\mathbf{FS}}_3 \quad (2.54)$$

$$\widetilde{\mathbf{FS}}_3^{-1} = \widetilde{\mathbf{FS}}_2 \quad (2.55)$$

$$\widetilde{\mathbf{FS}}_4^{-1} = \widetilde{\mathbf{FS}}_4. \quad (2.56)$$

Tabela 2.3: Transformadas do co-seno de corpo finito unitárias ($\zeta \in GI(p)$, $p \equiv 3(\text{mod } 4)$).

Elementos da matriz de transformação	Dimensão da matriz
$[\widetilde{\mathbf{FC}}_{1e}]_{ik} = (\sqrt{2/N}) \tilde{\beta}_i \tilde{\beta}_k \cos_{\zeta}(ki)$	$i, k = 0, 1, \dots, N$
$[\widetilde{\mathbf{FC}}_{2e}]_{ik} = (\sqrt{2/N}) \tilde{\beta}_k \cos_{\zeta}(k(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FC}}_{3e}]_{ik} = (\sqrt{2/N}) \tilde{\beta}_i \cos_{\zeta}((k + \frac{1}{2})i)$	$i, k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FC}}_{4e}]_{ik} = (\sqrt{2/N}) \cos_{\zeta}((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FC}}_{1o}]_{ik} = (2/\sqrt{2N-1}) \tilde{\beta}_i \tilde{\beta}_k \cos_{\zeta}(ki)$	$i, k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FC}}_{2o}]_{ik} = (2/\sqrt{2N-1}) \tilde{\gamma}_i \tilde{\beta}_k \cos_{\zeta}(k(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FC}}_{3o}]_{ik} = (2/\sqrt{2N-1}) \tilde{\beta}_i \tilde{\gamma}_k \cos_{\zeta}((k + \frac{1}{2})i)$	$i, k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FC}}_{4o}]_{ik} = (2/\sqrt{2N-1}) \cos_{\zeta}((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 2$

2.4 Transformadas Numéricas e Exemplos

Nesta seção, são exibidos exemplos das transformadas trigonométricas sobre corpos finitos. Para implementar as FFTT, utilizou-se o *software* MATLAB^R, onde foram desenvolvidos programas com as funções básicas de buscar elementos unimodulares num corpo finito primo específico, determinar suas ordens e calcular a transformada unitária de um vetor de acordo com as definições apresentadas.

Para implementar cada FFTT, alguns aspectos precisam ser considerados. Observando as Tabelas 2.3 e 2.4, verifica-se que, num corpo finito $GF(p)$, uma condição necessária à definição das transformadas com simetria par é a existência do elemento $\sqrt{2/N} \pmod{p}$; assim como é necessária às transformadas com simetria ímpar a existência do elemento $(1/\sqrt{2N-1}) \pmod{p}$. Além disso, se alguma função peso modificada estiver presente na definição da transformada a ser calculada, é necessário que o elemento $\sqrt{2^{-1}} \pmod{p}$ exista.

Naturalmente, as ordens dos elementos ζ do corpo em questão também devem ser conhecidas, uma vez que são elas que determinam os possíveis comprimentos das transformadas que se deseja implementar. Nesta tese, conforme comentado na Seção 2.1, enfatiza-se a aplicabilidade das transformadas numéricas, isto é, aquelas cuja seqüência transformada \mathbf{X} , assim

Tabela 2.4: Transformadas do seno de corpo finito unitárias ($\zeta \in GI(p)$, $p \equiv 3(\text{mod } 4)$).

Elementos da matriz de transformação	Dimensão da matriz
$[\widetilde{\mathbf{FS}}_{1e}]_{ik} = (\sqrt{2/N}) \text{sen}_\zeta(ki)$	$i, k = 1, 2, \dots, N - 1$
$[\widetilde{\mathbf{FS}}_{2e}]_{ik} = (\sqrt{2/N}) \tilde{\beta}_k \text{sen}_\zeta(k(i + \frac{1}{2}))$	$i = 0, 1, \dots, N - 1$ $k = 1, 2, \dots, N$
$[\widetilde{\mathbf{FS}}_{3e}]_{ik} = (\sqrt{2/N}) \tilde{\beta}_i \text{sen}_\zeta((k + \frac{1}{2})i)$	$i = 1, 2, \dots, N$ $k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FS}}_{4e}]_{ik} = (\sqrt{2/N}) \text{sen}_\zeta((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$
$[\widetilde{\mathbf{FS}}_{1o}]_{ik} = (2/\sqrt{2N - 1}) \text{sen}_\zeta(ki)$	$i, k = 1, 2, \dots, N - 1$
$[\widetilde{\mathbf{FS}}_{2o}]_{ik} = (2/\sqrt{2N - 1}) \text{sen}_\zeta(k(i + \frac{1}{2}))$	$i = 0, 1, \dots, N - 2$ $k = 1, 2, \dots, N - 1$
$[\widetilde{\mathbf{FS}}_{3o}]_{ik} = (2/\sqrt{2N - 1}) \text{sen}_\zeta((k + \frac{1}{2})i)$	$i = 1, 2, \dots, N - 1$ $k = 0, 1, \dots, N - 2$
$[\widetilde{\mathbf{FS}}_{4o}]_{ik} = (2/\sqrt{2N - 1}) \tilde{\gamma}_i \tilde{\gamma}_k \text{sen}_\zeta((k + \frac{1}{2})(i + \frac{1}{2}))$	$i, k = 0, 1, \dots, N - 1$

como a seqüência \mathbf{x} , possui apenas elementos pertencentes a $\text{GF}(p)$. Transformadas com essa característica são construídas com o uso de elementos ζ unimodulares, aos quais estão associados co-senos e senos sobre $\text{GF}(p)$ (vide Lema 2.1).

Para construir uma FFTT numérica do tipo 1, elementos unimodulares com ordem multiplicativa $2N$ são necessários. A construção de FFTT numéricas dos tipos 2 e 3 requer elementos unimodulares com ordem $4N$. Isto se deve à presença do termo $1/2$ na definição dessas transformadas, o qual representa a necessidade de extração da raiz quadrada de ζ , que também precisa ser unimodular. Analogamente, FFTT numéricas do tipo 4 requerem ζ com ordem multiplicativa $8N$ (dois termos $1/2$ indicam a necessidade de obter a raiz quarta de ζ).

Diante das condições mencionadas, a implementação de uma FFTT poderia representar uma tarefa de realização bastante restrita. Entretanto, a considerável diversidade de tipos de

transformadas trigonométricas permite que, quase sempre, considerando um corpo finito e um comprimento de referência N determinados, pelo menos um tipo de FFTT seja implementado. Isso pode ser avaliado nos exemplos que seguem.

Exemplo 2.1 (FFCT-1e) Para $p = 31$, o elemento $\zeta = (7 + j13) \in GI(31)$ possui ordem $16 = 2N$. Como a FFCT-1e tem comprimento $N+1$, a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 9. Assim, a FFCT-1e de $x = (23, 12, 1, 5, 9, 28, 0, 3, 1)$ é a seqüência $X = (2, 21, 8, 29, 10, 11, 29, 11, 8)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{1e} = \begin{bmatrix} 23 & 29 & 29 & 29 & 29 & 29 & 29 & 29 & 23 \\ 29 & 12 & 29 & 22 & 0 & 9 & 2 & 19 & 2 \\ 29 & 29 & 0 & 2 & 16 & 2 & 0 & 29 & 29 \\ 29 & 22 & 2 & 19 & 0 & 12 & 29 & 9 & 2 \\ 29 & 0 & 16 & 0 & 15 & 0 & 16 & 0 & 29 \\ 29 & 9 & 2 & 12 & 0 & 19 & 29 & 22 & 2 \\ 29 & 2 & 0 & 29 & 16 & 29 & 0 & 2 & 29 \\ 29 & 19 & 29 & 9 & 0 & 22 & 2 & 12 & 2 \\ 23 & 2 & 29 & 2 & 29 & 2 & 29 & 2 & 23 \end{bmatrix}.$$

Exemplo 2.2 (FFCT-2e) Para $p = 127$, o elemento $\zeta = (106 + j103) \in GI(127)$ possui ordem $16 = 2N$. Como a FFCT-2e tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 8. Assim, a FFCT-2e de $x = (120, 11, 0, 12, 4, 67, 77, 4)$ é a seqüência $X = (90, 109, 116, 40, 67, 1, 123, 80)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{2e} = \begin{bmatrix} 123 & 51 & 74 & 93 & 4 & 107 & 12 & 112 \\ 123 & 93 & 12 & 15 & 123 & 76 & 53 & 20 \\ 123 & 107 & 115 & 76 & 123 & 112 & 74 & 93 \\ 123 & 112 & 53 & 20 & 4 & 93 & 115 & 76 \\ 123 & 15 & 53 & 107 & 4 & 34 & 115 & 51 \\ 123 & 20 & 115 & 51 & 123 & 15 & 74 & 34 \\ 123 & 34 & 12 & 112 & 123 & 51 & 53 & 107 \\ 123 & 76 & 74 & 34 & 4 & 20 & 12 & 15 \end{bmatrix}.$$

Exemplo 2.3 (FFCT-3e) Para $p = 127$, o elemento $\zeta = (106 + j103) \in GI(127)$ possui ordem $16 = 2N$. Como a FFCT-3e tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir

tamanho igual a 8. Assim, a FFCT-3e de $x = (34, 35, 12, 80, 70, 111, 123, 12)$ é a seqüência $X = (62, 74, 38, 57, 75, 28, 101, 1)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{3e} = \begin{bmatrix} 123 & 123 & 123 & 123 & 123 & 123 & 123 & 123 \\ 51 & 93 & 107 & 112 & 15 & 20 & 34 & 76 \\ 74 & 12 & 115 & 53 & 53 & 115 & 12 & 74 \\ 93 & 15 & 76 & 20 & 107 & 51 & 112 & 34 \\ 4 & 123 & 123 & 4 & 4 & 123 & 123 & 4 \\ 107 & 76 & 112 & 93 & 34 & 15 & 51 & 20 \\ 12 & 53 & 74 & 115 & 115 & 74 & 53 & 12 \\ 112 & 20 & 93 & 76 & 51 & 34 & 107 & 15 \end{bmatrix}.$$

Exemplo 2.4 (FFCT-4e) Para $p = 31$, o elemento $\zeta = (4 + j27) \in GI(31)$ possui ordem $8 = 2N$. Como a FFCT-4e tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 4. Assim, a FFCT-4e de $x = (23, 12, 1, 5)$ é a seqüência $X = (25, 12, 18, 28)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{4e} = \begin{bmatrix} 8 & 11 & 22 & 18 \\ 11 & 13 & 23 & 9 \\ 22 & 23 & 18 & 11 \\ 18 & 9 & 11 & 23 \end{bmatrix}.$$

Exemplo 2.5 (FFCT-1o) Para $p = 47$, o elemento $\zeta = (23 + j6) \in GI(47)$ possui ordem $3 = 2N - 1$. Como a FFCT-1o tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 2. Assim, a FFCT-1o de $x = (23, 3)$ é a seqüência $X = (8, 2)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{1o} = \begin{bmatrix} 4 & 19 \\ 19 & 43 \end{bmatrix}.$$

Exemplo 2.6 (FFCT-2o) Para $p = 271$, o elemento $\zeta = (67 + j173) \in GI(271)$ possui ordem $17 = 2N - 1$. Como a FFCT-2o tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 9. Assim, a FFCT-2o de $x = (234, 21, 2, 23, 120, 250, 121, 10, 2)$ é a seqüência $X = (186, 62, 72, 152, 233, 176, 210, 228, 142)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{2o} = \begin{bmatrix} 113 & 192 & 265 & 62 & 1 & 257 & 140 & 230 & 60 \\ 113 & 62 & 140 & 211 & 14 & 6 & 79 & 270 & 41 \\ 113 & 257 & 41 & 6 & 209 & 211 & 1 & 192 & 140 \\ 113 & 230 & 209 & 270 & 140 & 192 & 60 & 6 & 14 \\ 113 & 211 & 79 & 230 & 265 & 131 & 209 & 257 & 1 \\ 113 & 131 & 14 & 192 & 41 & 270 & 265 & 211 & 209 \\ 113 & 270 & 60 & 257 & 79 & 62 & 41 & 131 & 265 \\ 113 & 6 & 1 & 131 & 60 & 230 & 14 & 62 & 79 \\ 4 & 158 & 113 & 158 & 113 & 158 & 113 & 158 & 113 \end{bmatrix}.$$

Exemplo 2.7 (FFCT-3o) Para $p = 271$, o elemento $\zeta = (67 + j173) \in GI(271)$ possui ordem $17 = 2N - 1$. Como a FFCT-3o tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 9. Assim, a FFCT-3o de $x = (234, 21, 2, 23, 120, 250, 121, 10, 2)$ é a seqüência $X = (172, 160, 16, 22, 7, 1, 6, 8, 231)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{3o} = \begin{bmatrix} 113 & 113 & 113 & 113 & 113 & 113 & 113 & 113 & 4 \\ 192 & 62 & 257 & 230 & 211 & 131 & 270 & 6 & 158 \\ 265 & 140 & 41 & 209 & 79 & 14 & 60 & 1 & 113 \\ 62 & 211 & 6 & 270 & 230 & 192 & 257 & 131 & 158 \\ 1 & 14 & 209 & 140 & 265 & 41 & 79 & 60 & 113 \\ 257 & 6 & 211 & 192 & 131 & 270 & 62 & 230 & 158 \\ 140 & 79 & 1 & 60 & 209 & 265 & 41 & 14 & 113 \\ 230 & 270 & 192 & 6 & 257 & 211 & 131 & 62 & 158 \\ 60 & 41 & 140 & 14 & 1 & 209 & 265 & 79 & 113 \end{bmatrix}.$$

Exemplo 2.8 (FFCT-4o) Para $p = 43$, o elemento $\zeta = (18 + j8) \in GI(43)$ possui ordem $11 = 2N - 1$. Como a FFCT-4o tem comprimento $N - 1$, a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 5. Assim, a FFCT-4o de $x = (34, 30, 1, 23, 2)$ é a seqüência $X = (9, 25, 40, 1, 5)$. A matriz de transformação é

$$\widetilde{\mathbf{FC}}_{4o} = \begin{bmatrix} 12 & 9 & 36 & 11 & 37 \\ 9 & 37 & 32 & 31 & 7 \\ 36 & 32 & 34 & 37 & 12 \\ 11 & 31 & 37 & 36 & 34 \\ 37 & 7 & 12 & 34 & 11 \end{bmatrix}.$$

Exemplo 2.9 (FFST-1e) Para $p = 31$, o elemento $\zeta = (7 + j13) \in GI(31)$ possui ordem $16 = 2N$. Como a FFST-1e tem comprimento $N - 1$, a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 7. Assim, a FFST-1e de $x = (23, 12, 1, 5, 9, 28, 0)$ é a seqüência $X = (30, 22, 6, 0, 6, 7, 20)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{1e} = \begin{bmatrix} 9 & 2 & 19 & 16 & 19 & 2 & 9 \\ 2 & 16 & 2 & 0 & 29 & 15 & 29 \\ 19 & 2 & 22 & 15 & 22 & 2 & 19 \\ 16 & 0 & 15 & 0 & 16 & 0 & 15 \\ 19 & 29 & 22 & 16 & 22 & 29 & 19 \\ 2 & 15 & 2 & 0 & 29 & 16 & 29 \\ 9 & 29 & 19 & 15 & 19 & 29 & 9 \end{bmatrix}.$$

Exemplo 2.10 (FFST-2e) Para $p = 127$, o elemento $\zeta = (106 + j103) \in GI(127)$ possui ordem $16 = 2N$. Como a FFST-2e tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 8. Assim, a FFST-2e de $x = (120, 11, 0, 12, 4, 67, 77, 4)$ é a seqüência $X = (125, 28, 23, 55, 125, 104, 88, 80)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{2e} = \begin{bmatrix} 112 & 12 & 107 & 4 & 93 & 74 & 51 & 123 \\ 107 & 74 & 51 & 4 & 112 & 115 & 34 & 4 \\ 93 & 74 & 112 & 123 & 76 & 115 & 107 & 123 \\ 51 & 12 & 34 & 123 & 107 & 74 & 15 & 4 \\ 51 & 115 & 34 & 4 & 107 & 53 & 15 & 123 \\ 93 & 53 & 112 & 4 & 76 & 12 & 107 & 4 \\ 107 & 53 & 51 & 123 & 112 & 12 & 34 & 123 \\ 112 & 115 & 107 & 123 & 93 & 53 & 51 & 4 \end{bmatrix}.$$

Exemplo 2.11 (FFST-3e) Para $p = 127$, o elemento $\zeta = (106 + j103) \in GI(127)$ possui ordem $16 = 2N$. Como a FFST-3e tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 8. Assim, a FFST-3e de $x = (34, 35, 12, 80, 70, 111, 123, 12)$ é a seqüência $X = (111, 118, 1, 27, 67, 126, 41, 79)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{3e} = \begin{bmatrix} 112 & 107 & 93 & 51 & 51 & 93 & 107 & 112 \\ 12 & 74 & 74 & 12 & 115 & 53 & 53 & 115 \\ 107 & 51 & 112 & 34 & 34 & 112 & 51 & 107 \\ 4 & 4 & 123 & 123 & 4 & 4 & 123 & 123 \\ 93 & 112 & 76 & 107 & 107 & 76 & 112 & 93 \\ 74 & 115 & 115 & 74 & 53 & 12 & 12 & 53 \\ 51 & 34 & 107 & 15 & 15 & 107 & 34 & 51 \\ 123 & 4 & 123 & 4 & 123 & 4 & 123 & 4 \end{bmatrix}.$$

Exemplo 2.12 (FFST-4e) Para $p = 31$, o elemento $\zeta = (4 + j27) \in GI(31)$ possui ordem $8 = 2N$. Como a FFST-4e tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 4. Assim, a FFST-4e de $x = (23, 12, 1, 5)$ é a seqüência $X = (15, 24, 18, 16)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{4e} = \begin{bmatrix} 13 & 9 & 20 & 23 \\ 9 & 23 & 13 & 11 \\ 20 & 13 & 8 & 9 \\ 23 & 11 & 9 & 18 \end{bmatrix}.$$

Exemplo 2.13 (FFST-1o) Para $p = 83$, o elemento $\zeta = (5 + j15) \in GI(83)$ possui ordem $7 = 2N - 1$. Como a FFST-1o tem comprimento $N - 1$, a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 3. Assim, a FFST-1o de $x = (3, 43, 81)$ é a seqüência $X = (47, 68, 80)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{1o} = \begin{bmatrix} 50 & 2 & 53 \\ 2 & 30 & 33 \\ 53 & 33 & 2 \end{bmatrix}.$$

Exemplo 2.14 (FFST-2o) Para $p = 271$, o elemento $\zeta = (67 + j173) \in GI(271)$ possui ordem $17 = 2N - 1$. Como a FFST-2o tem comprimento $N - 1$, a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 8. Assim, a FFST-2o de $x = (234, 21, 2, 23, 120, 250, 121, 10)$ é a seqüência $X = (247, 155, 103, 37, 264, 85, 243, 5)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{2o} = \begin{bmatrix} 74 & 29 & 234 & 92 & 117 & 104 & 268 & 23 \\ 234 & 104 & 23 & 117 & 29 & 197 & 179 & 3 \\ 117 & 268 & 29 & 37 & 248 & 179 & 74 & 104 \\ 268 & 234 & 179 & 167 & 74 & 23 & 29 & 154 \\ 23 & 197 & 3 & 29 & 104 & 37 & 154 & 92 \\ 104 & 154 & 197 & 268 & 179 & 242 & 23 & 37 \\ 92 & 248 & 117 & 197 & 37 & 268 & 167 & 29 \\ 29 & 179 & 104 & 248 & 268 & 154 & 234 & 197 \end{bmatrix}.$$

Exemplo 2.15 (FFST-3o) Para $p = 271$, o elemento $\zeta = (67 + j173) \in GI(271)$ possui ordem $17 = 2N - 1$. Como a FFST-3o tem comprimento $N - 1$, a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 8. Assim, a FFST-3o de $x = (234, 21, 2, 23, 120, 250, 121, 10)$ é a seqüência $X = (254, 222, 263, 177, 257, 21, 133, 58)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{3o} = \begin{bmatrix} 74 & 234 & 117 & 268 & 23 & 104 & 92 & 29 \\ 29 & 104 & 268 & 234 & 197 & 154 & 248 & 179 \\ 234 & 23 & 29 & 179 & 3 & 197 & 117 & 104 \\ 92 & 117 & 37 & 167 & 29 & 268 & 197 & 248 \\ 117 & 29 & 248 & 74 & 104 & 179 & 37 & 268 \\ 104 & 197 & 179 & 23 & 37 & 242 & 268 & 154 \\ 268 & 179 & 74 & 29 & 154 & 23 & 167 & 234 \\ 23 & 3 & 104 & 154 & 92 & 37 & 29 & 197 \end{bmatrix}.$$

Exemplo 2.16 (FFST-4o) Para $p = 271$, o elemento $\zeta = (39 + j102) \in GI(271)$ possui ordem $17 = 2N - 1$. Como a FFST-4o tem comprimento N , a seqüência \mathbf{x} , a ser transformada, deve possuir tamanho igual a 9. Assim, a FFST-4o de $x = (221, 220, 2, 34, 12, 100, 23, 1, 2)$ é a seqüência $X = (204, 177, 176, 238, 73, 214, 167, 19, 113)$. A matriz de transformação é

$$\widetilde{\mathbf{FS}}_{4_0} = \begin{bmatrix} 131 & 79 & 270 & 60 & 62 & 265 & 230 & 14 & 158 \\ 79 & 62 & 14 & 230 & 60 & 131 & 1 & 6 & 113 \\ 270 & 14 & 62 & 140 & 6 & 41 & 192 & 60 & 158 \\ 60 & 230 & 140 & 257 & 1 & 62 & 265 & 192 & 113 \\ 62 & 60 & 6 & 1 & 230 & 79 & 257 & 140 & 158 \\ 265 & 131 & 41 & 62 & 79 & 257 & 60 & 270 & 113 \\ 230 & 1 & 192 & 265 & 257 & 60 & 131 & 209 & 158 \\ 14 & 6 & 60 & 192 & 140 & 270 & 209 & 230 & 113 \\ 158 & 113 & 158 & 113 & 158 & 113 & 158 & 113 & 267 \end{bmatrix}.$$

CAPÍTULO 3

PROPRIEDADES DAS FFTT

O ESTUDO das propriedades de uma transformada discreta possui significativa importância na caracterização e na avaliação do potencial desta ferramenta. Sendo a DFT a mais conhecida das transformadas discretas, é natural utilizá-la como referência no desenvolvimento das propriedades de outras transformadas. A transformada de Fourier de corpo finito (FFFT – *finite field Fourier transform*), particularmente, possui propriedades similares às da DFT e outras próprias da estrutura finita, devendo-se apenas considerar as diferenças entre os corpos nas quais as duas estão definidas [20].

A extensão das propriedades da DFT às transformadas trigonométricas não é uma tarefa trivial, tanto no contexto dos números reais quanto no contexto de corpos finitos. Na literatura existente, as propriedades das transformadas discretas do co-seno e do seno são, em geral, apenas mencionadas, sendo escassos os trabalhos que as abordam com uma riqueza maior de detalhes [53], [52].

Nas seções a seguir, as principais propriedades das FFTT são introduzidas. Com o intuito de simplificar a notação e tornar as demonstrações mais simples, em alguns casos, considera-se as versões não normalizadas das FFTT, introduzidas no Capítulo 2. Oportunamente, as versões unitárias das mesmas são consideradas. Como a diferença básica entre esses dois grupos de transformadas é a presença de fatores de escala específicos, as propriedades desenvolvidas com base num deles podem ser estendidas ao outro sem dificuldade. Devido à diversidade de FFTT, para cada propriedade discutida, são enfocadas apenas algumas dessas transformadas, uma vez que as demais possuem um comportamento semelhante.

Adicionalmente, como resultado do estudo das propriedades das FFTT, são apresentadas duas aplicações relacionadas às mesmas. A primeira delas consiste numa marca d'água digital frágil no domínio da transformada do co-seno sobre corpos finitos. A segunda aplicação consiste num procedimento de filtragem digital no domínio da FFTT baseado na propriedade da convolução simétrica.

3.1 Linearidade

Se duas seqüências de duração finita $\mathbf{x}_1 = (x_{1,i})$ e $\mathbf{x}_2 = (x_{2,i})$, $x_{1,i}, x_{2,i} \in GF(p)$, com FFTT dadas respectivamente por $\mathbf{X}_1 = (X_{1,k})$ e $\mathbf{X}_2 = (X_{2,k})$, $X_{1,k}, X_{2,k} \in GI(p)$, são linearmente combinadas, isto é,

$$\mathbf{x}_3 = a \mathbf{x}_1 + b \mathbf{x}_2, \quad (3.1)$$

em que $a, b \in GF(p)$, então, a FFTT de $\mathbf{x}_3 = (x_{3,i})$ é dada por

$$\mathbf{X}_3 = a \mathbf{X}_1 + b \mathbf{X}_2. \quad (3.2)$$

Naturalmente, as FFTT das três seqüências envolvidas devem ser do mesmo tipo e possuir comprimentos iguais. A demonstração desta propriedade é trivial, podendo ser diretamente realizada a partir de definições apresentadas anteriormente.

3.2 Deslocamento no tempo

Diferentemente do que acontece com a transformada de Fourier, a propriedade de deslocamento no tempo das transformadas trigonométricas não é obtida de maneira imediata [53], [54]. Como se verifica a seguir, o desenvolvimento dessa propriedade é dificultado, basicamente, pelo fato de se ter transformadas de comprimento N (ou $N + 1$, ou $N - 1$) e núcleos de ordem $2N$ (ou $2N - 1$). Para isso, considera-se a transformada FFCT-2e.

De acordo com a Definição 2.5, a FFCT-2e da seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in GF(p)$, é a seqüência $\mathbf{C} = (C_k)$, $k = 0, 1, \dots, N - 1$, $C_k \in GI(p)$, de elementos

$$C_k = \sum_{i=0}^{N-1} 2x_i \cos_{\zeta}(k(i + 1/2)). \quad (3.3)$$

Deseja-se obter uma expressão para a FFCT-2e da seqüência $\hat{\mathbf{x}} = (\hat{x}_i)$, onde, para qualquer i , $\hat{x}_i = x_{i+i_0}$, sendo i_0 um número inteiro. Da Equação 3.3, obtém-se a FFCT-2e de $\hat{\mathbf{x}}$:

$$\hat{C}_k = \sum_{i=0}^{N-1} 2\hat{x}_i \cos_{\zeta}(k(i + 1/2)). \quad (3.4)$$

Na última equação, substituindo \hat{x}_i por x_{i+i_0} e, em seguida, aplicando a fórmula do co-seno da diferença de dois arcos, tem-se

$$\begin{aligned}\hat{C}_k &= \sum_{i=0}^{N-1} 2x_{i+i_0} \cos_{\zeta}(k(i+i_0+1/2-i_0)) \\ &= \cos_{\zeta}(ki_0) \sum_{i=0}^{N-1} 2x_{i+i_0} \cos_{\zeta}(k(i+i_0+1/2)) \\ &\quad + \operatorname{sen}_{\zeta}(ki_0) \sum_{i=0}^{N-1} 2x_{i+i_0} \operatorname{sen}_{\zeta}(k(i+i_0+1/2)).\end{aligned}\quad (3.5)$$

Para simplificar o tratamento da última expressão, pode-se desmembrá-la, fazendo

$$\hat{C}'_k = \sum_{i=0}^{N-1} 2x_{i+i_0} \cos_{\zeta}(k(i+i_0+1/2)), \quad (3.6)$$

onde, substituindo $i+i_0$ por r , obtém-se

$$\hat{C}'_k = \sum_{r=i_0}^{N+i_0-1} 2x_r \cos_{\zeta}(k(r+1/2)). \quad (3.7)$$

O somatório na Equação 3.7 pode ser separado, resultando em

$$\hat{C}'_k = \sum_{r=0}^{N-1} 2x_r \cos_{\zeta}(k(r+1/2)) + \sum_{r=N}^{N+i_0-1} 2x_r \cos_{\zeta}(k(r+1/2)) \quad (3.8)$$

$$- \sum_{r=0}^{i_0-1} 2x_r \cos_{\zeta}(k(r+1/2)). \quad (3.9)$$

Assumindo que a seqüência \mathbf{x} possui período N , o deslocamento aqui considerado equivale a um deslocamento cíclico. Assim, lembrando que $\zeta^N = -1$, a equação anterior resume-se a

$$\hat{C}'_k = C_k + \sum_{r=0}^{i_0-1} 2x_r \{(-1)^k - 1\} \cos_{\zeta}(k(r+1/2)). \quad (3.10)$$

De maneira semelhante, fazendo

$$\hat{C}''_k = \sum_{i=0}^{N-1} 2x_{i+i_0} \operatorname{sen}_{\zeta}(k(i+i_0+1/2)), \quad (3.11)$$

obtém-se

$$\hat{C}''_k = S_{k+1} + \sum_{r=0}^{i_0-1} 2x_r \{(-1)^k - 1\} \operatorname{sen}_{\zeta}(k(r+1/2)). \quad (3.12)$$

Na equação anterior, o termo S_{k+1} está associado à FFST-2e da seqüência x . O índice $k+1$ explica-se pelo fato de, originalmente, esta transformada ter coeficientes com índices variando

de 1 a N (vide Equação 2.13). Como na Equação 3.3 tem-se k variando de 0 a $N - 1$, esse ajuste se faz necessário.

Substituindo os resultados obtidos nas Equações 3.10 e 3.12 na Equação 3.5, e novamente observando a fórmula do co-seno da diferença de dois arcos, chega-se à expressão

$$\hat{C}_k = \cos_\zeta(ki_0)C_k + \text{sen}_\zeta(ki_0)S_{k+1} + \sum_{r=0}^{i_0-1} 2x_r \{(-1)^k - 1\} \cos_\zeta(k(r + 1/2 - i_0)). \quad (3.13)$$

No contexto dos números reais, onde as transformadas trigonométricas discretas são bastante utilizadas, a propriedade de deslocamento no tempo desempenha, normalmente, o papel de coadjuvante. Em alguns trabalhos, entretanto, a mesma possui uma importância fundamental, como no caso de sistemas OFDM baseados nas DTT [55], [56].

3.3 Teorema de Parseval

O Teorema de Parseval relaciona a energia associada a uma determinada seqüência com a energia associada à sua transformada [44]. Para desenvolver essa propriedade, considera-se a versão unitária da FFCT-2e, cujas expressões de transformação direta e inversa são apresentadas na Tabela 2.3 e na Equação 2.50.

A energia total de uma seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in \text{GF}(p)$, é dada por

$$E \triangleq \sum_{i=0}^{N-1} x_i^2 \pmod{p}. \quad (3.14)$$

Aplicando a fórmula de inversão às componentes x_i , a última equação é reescrita como

$$E = \sum_{i=0}^{N-1} \left| \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} \tilde{\beta}_k C_k \cos_\zeta(k(i + 1/2)) \right|^2. \quad (3.15)$$

Expandindo o quadrado do co-seno e, posteriormente, trocando a posição dos somatórios, obtém-se

$$\begin{aligned} E &= \sum_{i=0}^{N-1} \left\{ \frac{2}{N} \sum_{k=0}^{N-1} \tilde{\beta}_k^2 C_k^2 \frac{1}{2} [\cos_\zeta(k(2i + 1)) + \cos_\zeta(0)] \right\} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \tilde{\beta}_k^2 C_k^2 \sum_{i=0}^{N-1} [\cos_\zeta(k(2i + 1)) + 1]. \end{aligned} \quad (3.16)$$

Utilizando a fórmula do co-seno da adição de arcos, reescreve-se a equação anterior como

$$E = \frac{1}{N} \sum_{k=0}^{N-1} \tilde{\beta}_k^2 C_k^2 \left[\cos_\zeta(k) \sum_{i=0}^{N-1} \cos_\zeta(k2i) - \text{sen}_\zeta(k) \sum_{i=0}^{N-1} \text{sen}_\zeta(k2i) + \sum_{i=0}^{N-1} 1 \right]. \quad (3.17)$$

Como o elemento ζ sobre o qual o co-seno e o seno são calculados possui ordem multiplicativa $2N$, mostra-se que as seguintes propriedades são satisfeitas [20]:

$$\sum_{k=0}^{N-1} \cos_{\zeta}(k2i) = \begin{cases} N, & i = 0, \\ 0, & i \neq 0, \end{cases} \quad (3.18)$$

e

$$\sum_{k=0}^{N-1} \sin_{\zeta}(k2i) = 0. \quad (3.19)$$

Com base nas Equações 3.18 e 3.19, é possível avaliar os termos entre colchetes na Equação 3.17. Enquanto o termo central é anulado, o último resulta em N ; o primeiro termo não é nulo apenas quando $k = 0$, o que fornece o seguinte resultado:

$$E = \frac{1}{N} \sum_{k=0}^{N-1} \frac{1}{2} C_k^2 (N + N) = \sum_{k=0}^{N-1} |C_k|^2. \quad (3.20)$$

O resultado final do Teorema de Parseval é

$$\sum_{i=0}^{N-1} |x_i|^2 = \sum_{k=0}^{N-1} |C_k|^2, \quad (3.21)$$

sendo válido para qualquer tipo de FFT unitária. As versões não unitárias das FFT também possuem esta propriedade, sendo necessário apenas adicionar fatores de escala apropriados na Equação 3.21.

3.4 Relação entre as FFT e a FFFT

Conforme descrito no Capítulo 2, a transformada trigonométrica de uma seqüência pode ser obtida a partir da transformada discreta de Fourier de uma versão simetricamente estendida da mesma. Nesta seção, este procedimento é apresentado em detalhes. O resultado é uma expressão que relaciona a transformada de Fourier de corpo finito com a FFCT-2e. Como este tipo de transformada é construído a partir de extensões simétricas do mesmo tipo (à esquerda e à direita da seqüência original), o desenvolvimento de sua relação com a FFFT é simplificado, podendo ser compreendido com mais facilidade.

Seja a seqüência $\mathbf{x} = (x_i)$, $i = 0, 1, \dots, N - 1$, $x_i \in \text{GF}(p)$. Estendendo x para a direita segundo a simetria do tipo HS, uma nova seqüência $\hat{\mathbf{x}} = (\hat{x}_i)$, $i = 0, 1, \dots, 2N - 1$, $\hat{x}_i \in \text{GF}(p)$, pode ser construída, de maneira que $\hat{x}_i = x_i + x_{2N-i-1}$ para qualquer i . A transformada de

Fourier de corpo finito da seqüência $\hat{\mathbf{x}}$ é a seqüência $\hat{\mathbf{F}} = (\hat{F}_k)$, $k = 0, 1, \dots, 2N - 1$, $\hat{F}_k \in \text{GI}(p)$, de elementos

$$\hat{F}_k = \sum_{i=0}^{2N-1} \hat{x}_i \zeta^{ki} = \sum_{i=0}^{2N-1} (x_i + x_{2N-i-1}) \zeta^{ki}, \quad (3.22)$$

em que ζ possui ordem multiplicativa igual a $2N$. Denotando por $\mathbf{F} = (F_k)$, $k = 0, 1, \dots, 2N - 1$, $F_k \in \text{GI}(p)$, a FFT de comprimento $2N$ da seqüência \mathbf{x} (que tem seu tamanho aumentado com zeros), e observando suas propriedades de reversão e deslocamento no tempo, que funcionam de maneira análoga à DFT, a última expressão pode ser reescrita como

$$\begin{aligned} \hat{F}_k &= F_k + F_k \zeta^{-k} \\ &= \zeta^{-k/2} \left(F_k \zeta^{k/2} + F_k^* \zeta^{-k/2} \right) \\ &= \zeta^{-k/2} 2\Re \left\{ F_k \zeta^{k/2} \right\}. \end{aligned} \quad (3.23)$$

Na equação anterior, o termo cuja parte real está selecionada é dado por

$$\begin{aligned} F_k \zeta^{k/2} &= \sum_{i=0}^{N-1} x_i \zeta^{ki} \zeta^{k/2} \\ &= \sum_{i=0}^{N-1} x_i \zeta^{k(i+1/2)}. \end{aligned} \quad (3.24)$$

Expandindo-se as potências de ζ em co-senos e senos, obtém-se

$$F_k \zeta^{k/2} = \sum_{i=0}^{N-1} x_i [\cos_{\zeta}(k(i+1/2)) + j \text{sen}_{\zeta}(k(i+1/2))]. \quad (3.25)$$

Da Equação 3.25, conclui-se que

$$2\Re \left\{ F_k \zeta^{k/2} \right\} = \sum_{i=0}^{N-1} 2x_i \cos_{\zeta}(k(i+1/2)) = C_k, \quad k = 0, \dots, N - 1, \quad (3.26)$$

ou

$$C_k = \zeta^{k/2} \hat{F}_k, \quad k = 0, \dots, N - 1, \quad (3.27)$$

em que a seqüência $\mathbf{C} = (C_k)$, $k = 0, 1, \dots, N - 1$, $C_k \in \text{GI}(p)$, corresponde à FFCT-2e da seqüência \mathbf{x} .

Relações entre outras FFT e a FFT podem ser obtidas de maneira análoga à que se demonstrou nessa seção. Para isso, o ponto fundamental é observar o tipo de simetria associado à construção de cada transformada, a fim de que o procedimento apresentado seja desenvolvido de maneira correta. Em casos em que é necessário estender a seqüência \mathbf{x} para a esquerda e para a direita com tipos diferentes de simetria, esta tarefa torna-se um pouco mais laboriosa.

3.5 Convolução simétrica

A convolução simétrica, que, no contexto das DTT, foi analisada pela primeira vez por Martucci [52], provê uma maneira sistemática de convoluir filtros com resposta ao impulso finita (FIR, *finite impulse response*) de fase linear com seqüências simetricamente estendidas. Esta propriedade tem sido utilizada em aplicações como filtragem de imagens sem superposição de blocos e redimensionamento de imagens no domínio da transformada [57], [29], [58]. A convolução simétrica pode ser descrita considerando a seqüência $\mathbf{h} = (h_i)$, a resposta ao impulso de um filtro FIR, obtida a partir da seqüência $\mathbf{h}^r = (h_i^r)$ simetricamente estendida para a esquerda. Considera-se, também, a seqüência $\tilde{\mathbf{x}} = (\tilde{x}_i)$, que corresponde à versão da seqüência $\mathbf{x} = (x_i)$ simetricamente estendida para ambos os lados. Sob determinadas condições, a convolução simétrica entre \mathbf{x} e \mathbf{h}^r pode equivaler à convolução linear entre $\tilde{\mathbf{x}}$ e \mathbf{h} .

O resultado da última operação mencionada é denotado por $\mathbf{w} = (w_i)$ e dado por

$$w_i = \sum_{k=-\infty}^{+\infty} h_k \tilde{x}_{i-k}, \quad (3.28)$$

e o resultado da convolução simétrica é obtido por

$$w'_i = \tau_c^{-1} \{ \tau_a \{ x_i \} \times \tau_b \{ h_i^r \} \}, \quad (3.29)$$

em que a operação \times denota o produto ponto-a-ponto entre as transformadas τ_a de \mathbf{x} e τ_b de \mathbf{h}^r ; obtém-se \mathbf{w}' pela aplicação da transformada τ_c inversa. No somatório na Equação 3.28, o índice k precisa variar apenas ao longo dos valores em que $h_k \neq 0$. A Equação 3.29 fornece somente um trecho da seqüência \mathbf{w} , o qual possui comprimento igual ao da transformada τ_c . De modo geral, tem-se

$$w'_i = w_{i-i_0}, \quad (3.30)$$

sendo que a variação do índice i é limitada segundo o comprimento de τ_c e o valor da constante i_0 é determinado em função do comprimento de \mathbf{x} e da extensão simétrica que origina $\tilde{\mathbf{x}}$. As transformadas trigonométricas τ_a , τ_b and τ_c , aqui tratadas como sendo de corpos finitos, também são selecionadas de acordo com os tipos de simetria do filtro e de $\tilde{\mathbf{x}}$.

A relação entre as expressões 3.28 e 3.29 depende unicamente das características de simetria das seqüências envolvidas. Isso pode ser demonstrado a partir das definições das FFTT e de manipulações matemáticas baseadas em substituições e mudanças de variáveis nos so-

matórios. Assim, de forma análoga ao que acontece no contexto dos números reais, é possível definir 40 tipos diferentes de convoluções simétricas utilizando os 16 tipos de FFTT [52].

A Figura 3.1 ilustra a relação entre as seqüências \mathbf{w} e \mathbf{w}' . A notação $\langle FC_{2e}^S FC_{1e} \rangle$, que informa o tipo de convolução simétrica utilizada, significa que a FFCT-2e (τ_a) de \mathbf{x} e a FFCT-1e (τ_b) de \mathbf{h}^r são calculadas. Como a escolha da transformada τ_c está atrelada aos tipos das outras duas transformadas, τ_c deve ser do tipo FFCT-2e [52]. Na figura, apresenta-se na saída a seqüência \mathbf{w}' , que, como foi dito, corresponde a apenas um trecho da convolução linear entre $\tilde{\mathbf{x}}$ e \mathbf{h} .

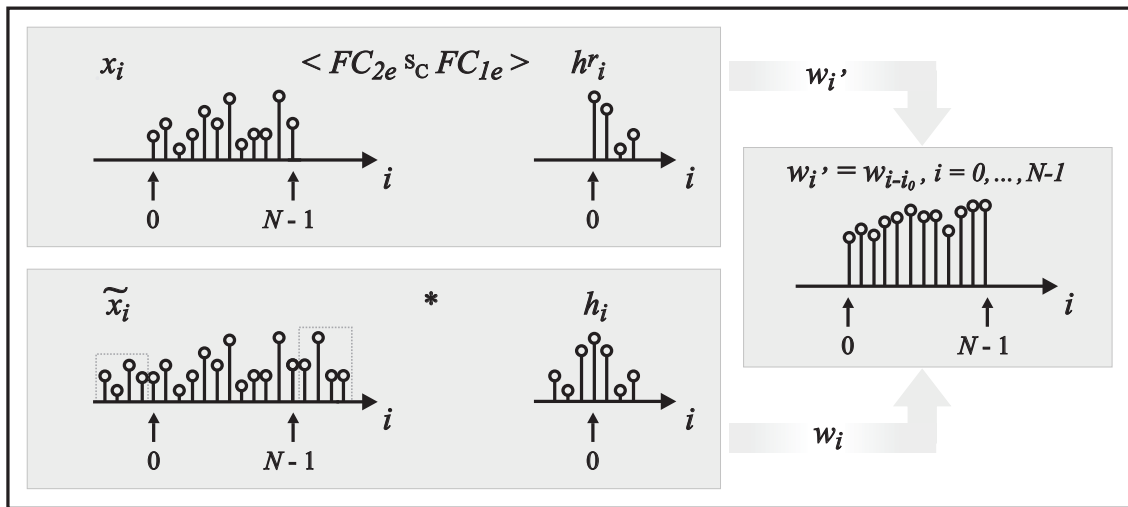


Figura 3.1: Exemplo mostrando que a seqüência $\mathbf{w}' = (w'_i)$, resultante de uma convolução simétrica, equivale a um trecho da seqüência $\mathbf{w} = (w_i)$, resultante de uma convolução linear com as entradas simetricamente estendidas.

Exemplo 3.1 O funcionamento da convolução simétrica apresentada na Figura 3.1 pode ser melhor compreendido através de um exemplo. Para isso, considera-se o filtro $\mathbf{h} = (h_i), i = -2, -1, 0, 1, 2$, com componentes sobre $GF(127)$, cuja resposta ao impulso é dada por

$$\mathbf{h} = \begin{pmatrix} 1 & 2 & 3 & 2 & 1 \end{pmatrix}, \quad (3.31)$$

e a seqüência $\mathbf{x} = (x_i), i = 0, \dots, 7$, também com componentes sobre $GF(127)$, dada por

$$\mathbf{x} = \begin{pmatrix} 2 & 0 & 1 & 0 & 1 & 2 & 3 & 1 \end{pmatrix}. \quad (3.32)$$

De acordo com o que foi descrito, a partir de \mathbf{h} , obtém-se $\mathbf{h}^r = (h_i^r), i = 0, 1, 2$, dada por

$$\mathbf{h}^r = \begin{pmatrix} 3 & 2 & 1 \end{pmatrix}. \quad (3.33)$$

Observando que \mathbf{h}^r possui comprimento igual a 3, é possível gerar a seqüência $\tilde{\mathbf{x}} = (\tilde{x}_i), i = -3, -2, \dots, 9, 10$, de comprimento 14 ($= 8 + 3 + 3$), realizando-se extensões simétricas em ambas

as extremidades de x . Neste exemplo, para utilizar a convolução simétrica, calcula-se mais à frente a FFCT-2e de \mathbf{x} , e, por este motivo, as extensões a partir das quais se compõe $\tilde{\mathbf{x}}$ são do tipo HS. Desta maneira, tem-se

$$\tilde{\mathbf{x}} = \left(1 \ 0 \ 2 \ 2 \ 0 \ 1 \ 0 \ 1 \ 2 \ 3 \ 1 \ 1 \ 3 \ 2 \right). \quad (3.34)$$

Usando a Equação 3.28, obtém-se o resultado da convolução linear entre a seqüência $\tilde{\mathbf{x}}$ e o filtro \mathbf{h} , que é a seqüência de saída $\mathbf{w} = (w_i)$, $i = -5, -4, \dots, 11, 12$, dada por

$$\mathbf{w} = \left(1 \ 2 \ 5 \ 8 \ 11 \ 11 \ 8 \ 6 \ 6 \ 11 \ 15 \ 17 \ 16 \ 16 \ 16 \ 13 \ 7 \ 2 \right). \quad (3.35)$$

Para implementar a convolução simétrica entre as seqüências \mathbf{x} e \mathbf{h}^r , deve-se computar a FFCT-2e de \mathbf{x} , como já foi mencionado, e a FFCT-1e de \mathbf{h}^r . A escolha desta última transformada justifica-se, também, pela equivalência entre as extensões simétricas que dão origem à mesma (vide Seção 2.3) e o tipo de simetria do filtro. A FFCT-2e de comprimento $N = 8$ de \mathbf{x} é a seqüência $\mathbf{C} = (C_k)$, $k = 0, 1, \dots, 7$, tal que

$$\mathbf{C} = \left(20 \ 10 \ 43 \ 65 \ 32 \ 34 \ 31 \ 87 \right) \quad (3.36)$$

e a FFCT-1e de comprimento $N + 1 = 9$ de \mathbf{h}^r é a seqüência $\mathbf{H}^r = (H_k^r)$, $k = 0, 1, \dots, 8$, tal que

$$\mathbf{H}^r = \left(9 \ 30 \ 98 \ 50 \ 1 \ 115 \ 35 \ 71 \ 1 \right). \quad (3.37)$$

De acordo com a Equação 3.29, as seqüência transformadas \mathbf{C} e \mathbf{H}^r precisam ser multiplicadas ponto-a-ponto. A fim de compatibilizar os comprimentos das mesmas e possibilitar seu produto, uma amostra com valor igual a 0 é acrescida ao final da seqüência \mathbf{C} . O resultado desta operação, que, naturalmente, é efetuada usando-se aritmética módulo 127, é a seqüência $\mathbf{W}' = (W'_k)$, $k = 0, 1, \dots, 8$, tal que

$$\mathbf{W}' = \left(53 \ 46 \ 23 \ 75 \ 32 \ 100 \ 69 \ 81 \ 0 \right). \quad (3.38)$$

Suprimindo a última amostra de \mathbf{W}' , que é nula, e calculando a sua FFCT-2e⁻¹ de comprimento $N = 8$, obtém-se o resultado final da convolução simétrica. Isso fornece a seqüência $\mathbf{w}' = (w'_i)$, $i = 0, 1, \dots, 7$, tal que

$$\mathbf{w}' = \left(11 \ 8 \ 6 \ 6 \ 11 \ 15 \ 17 \ 16 \right). \quad (3.39)$$

Comparando as seqüências \mathbf{w}' e \mathbf{w} , observa-se que

$$\mathbf{w}'_i = w_i, \quad i = 0, 1, \dots, 7. \quad (3.40)$$

No Exemplo 3.1, em que a constante $i_0 = 0$, ilustra-se a equivalência entre a seqüência resultante da convolução simétrica e um trecho da seqüência resultante da convolução linear,

como discutido anteriormente. A seguir, mostra-se que é possível modificar o procedimento exemplificado de maneira que toda a seqüência w seja obtida via convolução simétrica.

Para usar a convolução simétrica na implementação de uma convolução linear, é necessário realizar um preenchimento com zeros (*zero-padding*) em ambos os lados de x , gerando uma nova seqüência \hat{x} . A quantidade de zeros a ser inserida depende do comprimento do filtro h e dos tipos de FFTT utilizadas [52]. A Figura 3.2 ilustra esse procedimento. A notação $\langle FC_{2e} s_C FC_{1e} \rangle$ significa que a FFCT-2e de \hat{x} e a FFCT-1e de h^r devem ser calculadas. Para implementar a multiplicação ponto-a-ponto entre seqüências de comprimentos diferentes, as técnicas de *overlap-add* e *overlap-save* podem ser utilizadas [44].

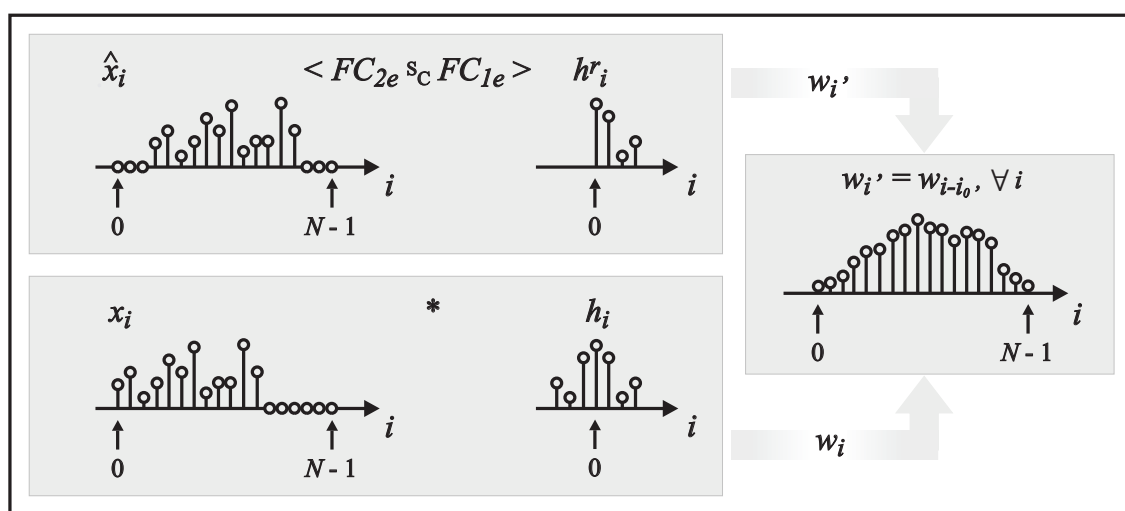


Figura 3.2: Exemplo mostrando como a mesma $w = (w_i)$ pode ser calculada tanto por uma convolução simétrica quanto por uma convolução linear, quando a seqüência de entrada é suficientemente preenchida com zeros em ambos os lados.

Exemplo 3.2 O método descrito pode ser ilustrado através deste exemplo, em que se deseja convoluir a seqüência $x = (x_i), i = 0, 1, \dots, 5$, em que

$$x = (2 \ 1 \ 3 \ 4 \ 2 \ 1), \quad (3.41)$$

com o filtro que possui resposta ao impulso $h = (h_i), i = -1, 0, 1$, em que

$$h = (1 \ 2 \ 1). \quad (3.42)$$

As componentes de ambas as seqüências estão sobre $GF(127)$. De modo análogo ao que foi feito no exemplo anterior, obtém-se $h^r = (h_i^r), i = 0, 1$, dada por

$$h^r = (2 \ 1). \quad (3.43)$$

Como o comprimento de \mathbf{h}^r é 2, deve-se realizar um preenchimento com uma amostra igual a 0 em cada extremidade da seqüência \mathbf{x} , a fim de construir a seqüência $\hat{\mathbf{x}} = (\hat{x}_i)$, $i = 0, 1, \dots, 7$. Dessa forma, tem-se

$$\hat{\mathbf{x}} = \begin{pmatrix} 0 & 2 & 1 & 3 & 4 & 2 & 1 & 0 \end{pmatrix}. \quad (3.44)$$

Usando a Equação 3.28, calcula-se de maneira direta a convolução entre \mathbf{x} e \mathbf{h} , que resulta na seqüência de saída $\mathbf{w} = (w_i)$, $i = -1, 0, \dots, 6$, tal que

$$\mathbf{w} = \begin{pmatrix} 2 & 5 & 7 & 11 & 13 & 9 & 4 & 1 \end{pmatrix}. \quad (3.45)$$

Para implementar a convolução simétrica entre $\hat{\mathbf{x}}$ e \mathbf{h}^r , deve-se computar a FFCT-2e da primeira seqüência e a FFCT-1e da segunda. A FFCT-2e de comprimento $N = 8$ de $\hat{\mathbf{x}}$ é a seqüência $\hat{\mathbf{C}} = (\hat{C}_k)$, $k = 0, 1, \dots, 7$, tal que

$$\hat{\mathbf{C}} = \begin{pmatrix} 26 & 123 & 40 & 70 & 111 & 8 & 82 & 88 \end{pmatrix} \quad (3.46)$$

e a FFCT-1e de comprimento $N + 1 = 9$ de \mathbf{h}^r é a seqüência $\mathbf{H}^r = (H_k^r)$, $k = 0, 1, \dots, 8$, em que

$$\mathbf{H}^r = \begin{pmatrix} 4 & 87 & 113 & 81 & 2 & 50 & 18 & 44 & 0 \end{pmatrix}. \quad (3.47)$$

Procedendo de modo semelhante ao primeiro exemplo, multiplica-se $\hat{\mathbf{C}}$ por \mathbf{H}^r ponto-a-ponto, obtendo-se a seqüência $\mathbf{W}' = (W'_k)$, $k = 0, 1, \dots, 8$, tal que

$$\mathbf{W}' = \begin{pmatrix} 104 & 33 & 75 & 82 & 95 & 19 & 79 & 62 & 0 \end{pmatrix}. \quad (3.48)$$

Suprimindo a última amostra de \mathbf{W}' , que é nula, e calculando a sua FFCT-2e⁻¹ de comprimento $N = 8$, obtém-se o resultado final da convolução simétrica. Isso fornece a seqüência $\mathbf{w}' = (w'_i)$, $i = 0, 1, \dots, 7$, em que

$$\mathbf{w}' = \begin{pmatrix} 2 & 5 & 7 & 11 & 13 & 9 & 4 & 1 \end{pmatrix}. \quad (3.49)$$

Comparando as seqüências \mathbf{w}' e \mathbf{w} , observa-se que

$$\mathbf{w}'_i = w_{i-1}, \quad i = 0, 1, \dots, 7. \quad (3.50)$$

Neste caso, $i_0 = 1$ e a seqüência \mathbf{w} , resultado da convolução linear, pode ser completamente obtida através da convolução simétrica.

3.6 Aplicações

3.6.1 *Uma Marca D'água Digital no Domínio da Transformada do Co-seno Sobre Corpos Finitos*

Desde a última década, o uso e a distribuição de informação multimídia digital vêm crescendo de maneira significativa. Hodiernamente, a popularização da rede mundial de computadores permite o acesso a arquivos de áudio e vídeo disponibilizados em qualquer parte do planeta. No entanto, a facilidade de comunicação e a partilha de recursos, proporcionadas pela tecnologia, possuem também aspectos negativos. A liberdade com que se pode copiar e comercializar a mídia digital tem comprometido o direito que os autores possuem sobre suas criações (fotos, desenhos, músicas, entre outras).

Diante disso, tornou-se necessário o desenvolvimento de métodos para proteger e verificar a autenticidade de uma imagem, por exemplo. Surgiu, então, a versão digital do termo “marca d'água”. Nesse contexto, marcar significa introduzir informação adicional que identifique o detentor dos direitos autorais sobre um produto ou que determine se o produto sofreu algum tipo de manipulação por parte de terceiros. Uma marca d'água digital precisa ser imperceptível, isto é, não alterar qualquer característica visível da informação original. Em alguns casos, a marca também deve ser robusta, sobrevivendo a ataques maliciosos que tenham o intuito de destruí-la [59], [60].

As marcas d'água digitais podem ser processadas no domínio espacial ou no da frequência. No domínio espacial, a marcação no bit menos significativo (LSB – *Least Significant Bit*) é uma das técnicas mais simples e conhecidas. Entretanto, a mesma apresenta limitações acerca da quantidade de informação que se pode esconder. Essa marca também é facilmente destruída, caso a imagem sofra alterações no brilho ou no contraste e compressões com perda [61]. No mesmo domínio, há, ainda, outros métodos baseados na superposição da marca à imagem original [61].

As técnicas implementadas no domínio da frequência consistem em aplicar transformadas discretas, como a de Fourier e a do co-seno, à imagem original [30], [62]. A marca d'água é inserida alterando-se os valores dos coeficientes destas transformadas. Calculando a transformada inversa, obtém-se a imagem marcada. Nestes métodos, uma das dificuldades é a implementação eficiente de algoritmos. O cálculo de transformadas reais implica o uso de aritmética de ponto-flutuante e, naturalmente, exige arredondamento, aspectos refletidos na precisão e na velocidade do processamento.

Em 2004, Aoki *et al.* propuseram uma marca d'água frágil para imagens em escala de cinza baseada na transformada de Fourier de corpo finito [63]. Nesse contexto, a inserção da marca tem o intuito principal de localizar regiões da imagem em que foram realizadas alterações, caracterizando, portanto, uma marca d'água frágil. O procedimento elimina os erros de arredondamento e simplifica os cálculos, realizando-os apenas com aritmética inteira.

Nesta seção, é apresentada uma marca d'água baseada na FFCT. Além de aproveitar a facilidade de cálculo inerente às transformadas inteiras, o esquema sugerido realiza uma espécie de “espalhamento” da marca d'água. O resultado disso é um sistema simples e robusto a algumas manipulações da imagem marcada. O esquema de inserção e o de extração da marca d'água proposta, bem como os resultados obtidos a partir das simulações realizadas, são apresentados a seguir.

A marca d'água proposta

A técnica proposta utiliza esquemas de inserção e extração da marca d'água similares àqueles que empregam transformadas reais. Inicialmente, tem-se uma imagem digital em escala de cinza, onde cada *pixel* assume valores inteiros de 0 a 255. Nesta imagem, introduz-se como marca uma imagem binária (*pixels* com valores 0 ou 1). Este procedimento faz uso de uma versão bidimensional da FFCT-2e, a qual é definida a seguir.

Definição 3.1 (FFCT_{2D-2e}) Se $\zeta \in GI(p)$ tem ordem multiplicativa $2N$, então a transformada do co-seno de corpo finito da matriz $\mathbf{x} = (x_{i_1, i_2})$, $i_1, i_2 = 0, 1, \dots, N - 1$, $x_{i_1, i_2} \in GF(p)$, é a matriz $\mathbf{C} = (C_{k_1, k_2})$, $k_1, k_2 = 0, 1, \dots, N - 1$, $C_{k_1, k_2} \in GI(p)$, de elementos

$$C_{k_1, k_2} \triangleq \sum_{i_1=0}^{N-1} \sum_{i_2=0}^{N-1} x_{i_1, i_2} 2 \cos_{\zeta}(k_1(i_1 + 1/2)) 2 \cos_{\zeta}(k_2(i_2 + 1/2)). \quad (3.51)$$

Teorema 3.1 (FFCT_{2D-2e⁻¹}) A transformada do co-seno de corpo finito inversa da matriz $\mathbf{C} = (C_{k_1, k_2})$, $k_1, k_2 = 0, 1, \dots, N - 1$, $C_{k_1, k_2} \in GI(p)$, é a seqüência $\mathbf{x} = (x_{i_1, i_2})$, $i_1, i_2 = 0, 1, \dots, N - 1$, $x_{i_1, i_2} \in GF(p)$, de elementos

$$x_{i_1, i_2} = \frac{1}{N^2} \sum_{k_1=0}^{N-1} \sum_{k_2=0}^{N-1} C_{k_1, k_2} \beta_{k_1} \cos_{\zeta}(k_1(i_1 + 1/2)) \beta_{k_2} \cos_{\zeta}(k_2(i_2 + 1/2)). \quad (3.52)$$

Para simplificar, a Definição 3.1 e o Teorema 3.1 apresentam uma FFCT_{2D} com dimensões $N \times N$. Desta forma, os co-senos sobre corpo finito que aparecem nas Equações 3.51 e 3.52 são funções do mesmo elemento unimodular ζ , que possui ordem multiplicativa $2N$. Isso significa

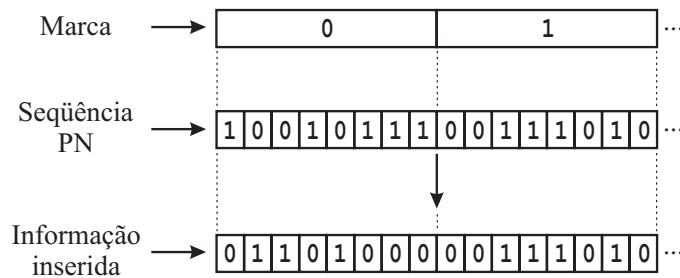


Figura 3.3: Espalhamento da marca d'água sobre uma seqüência PN, para obtenção da informação a ser inserida na imagem original ($L = 8$ bits).

que o cálculo desta transformada pode ser realizado pela mesma estrutura que calcula a FFCT- $2e$ de uma seqüência de comprimento N . Naturalmente, FFCT bidimensionais não quadradas podem ser definidas escolhendo-se dois elementos unimodulares diferentes, desde que a ordem de cada um deles seja compatível com o comprimento da respectiva dimensão.

O esquema de inserção da marca

O primeiro passo para a inserção da marca é reduzir a imagem original módulo p , o número primo associado ao corpo finito sobre o qual a FFCT $_{2D}$ é definida. Denotando por \mathbf{I} a imagem original, o procedimento descrito gera uma imagem denotada por \mathbf{I}_p . A matriz \mathbf{C} , que corresponde a FFCT $_{2D}$ de \mathbf{I}_p , é calculada e, no domínio da transformada, a marca é inserida. A informação a ser introduzida corresponde, de fato, a uma seqüência pseudo-aleatória (PN – *pseudo-noise*) na qual a marca d'água é espalhada. Cada bit da marca é associado a L bits da seqüência PN. Conforme ilustrado na Figura 3.3, se o bit da marca for 0, os L bits correspondentes são invertidos; se o bit da marca for 1, os L bits correspondentes permanecem inalterados. Esta operação, que produz a seqüência PN', tem a função de tornar mais seguro o procedimento de extração da marca, explicado mais adiante.

A Figura 3.4 detalha o esquema de inserção da marca d'água. A semente que gera a seqüência PN funciona como uma chave secreta e é definida pelo indivíduo que assina a imagem [59]. A seqüência PN' é mapeada em duas dimensões e somada a \mathbf{C} , resultando numa matriz \mathbf{C}_m . A imagem marcada, \mathbf{I}_m , é finalmente obtida calculando-se a FFCT $_{2D}^{-1}$ de \mathbf{C}_m e readicionando $\mathbf{I}' = \mathbf{I} - \mathbf{I}_p$.

Para que esta marca atenda de modo satisfatório o requisito da “invisibilidade”, mencionado na parte introdutória desta seção, é importante escolher adequadamente o valor de

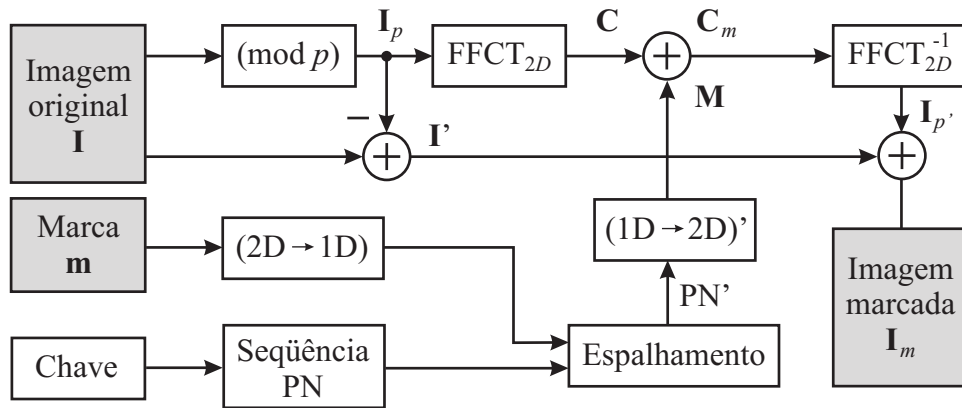


Figura 3.4: Esquema de inserção da marca d'água no domínio da FFCT. Após um procedimento de espalhamento sobre uma seqüência PN, uma marca d'água binária é somada (módulo p) à $FFCT_{2D}$ da imagem original. Em seguida, calcula-se a $FFCT_{2D}$ inversa para se obter a imagem marcada.

p . Na Figura 3.4, somar M a C significa alterar determinados coeficientes da $FFCT_{2D}$. Entretanto, numa transformada de corpo finito, adicionar apenas uma unidade a um elemento no domínio da frequência pode modificar em até $(p - 1)$ unidades os elementos de sua transformada inversa. Naturalmente, isso é explicado pelo fato de se estar usando aritmética modular. Portanto, para que o valor de cada pixel na imagem original seja alterado de maneira pouco significativa, em relação à sensibilidade do olho humano, deve-se escolher um número p que seja pequeno quando comparado a 255, que representa a máxima variação de brilho numa imagem em escala de cinza.

Com base nestas implicações, fixou-se $p = 7$ para a implementação das simulações que são apresentadas. Arelados ao valor de p estão o tamanho da transformada, 2×2 , o elemento unimodular $\zeta = 2 + j2$, cuja ordem multiplicativa é 8, e o corpo de extensão $GI(7)$.

O esquema de extração da marca

Para que a marca d'água proposta seja extraída, é necessário conhecer a semente que gera a seqüência PN utilizada em sua inserção, bem como a $FFCT_{2D}$ da imagem original reduzida módulo p . A Figura 3.5 apresenta o esquema que realiza este procedimento.

Inicialmente, calcula-se a diferença $(C' - C)$ entre a $FFCT_{2D}$ da imagem marcada e da imagem original, ambas reduzidas módulo p . A matriz resultante desta operação é mapeada numa seqüência \widetilde{PN}' de comprimento igual ao da seqüência PN. O bloco "Decisão" compara estas duas seqüências empregando uma lógica que é complementar à da operação de espalhamento realizada na inserção da marca. Para cada L pares de bits comparados, observa-

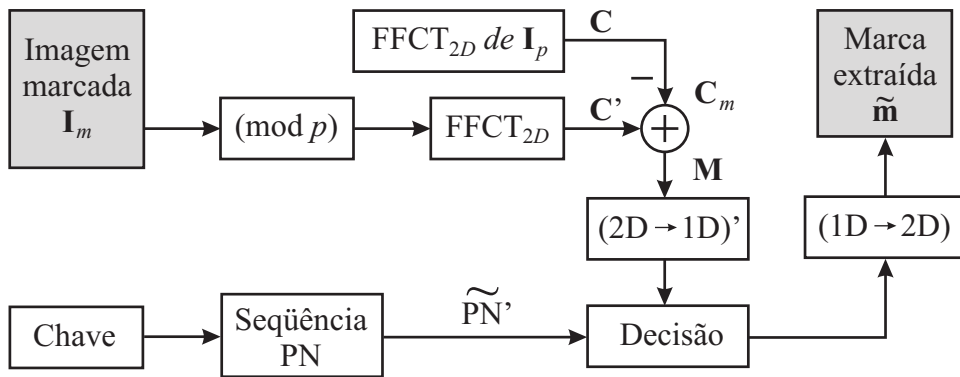


Figura 3.5: Esquema de extração da marca d'água baseada na FFCT. Realiza-se um procedimento o inverso ao de inserção da marca, adicionando-se o bloco Decisão, cuja função é descrita mais a diante.

se o número de coincidências e o de inversões. Se o número de coincidências predominar, decide-se pelo bit 1. Caso contrário, decide-se pelo bit 0 (Figura 3.6). A composição da marca é finalizada ao se realizar o mapeamento bidimensional desta nova seqüência obtida após as decisões.

Se a imagem marcada tiver sofrido alguma alteração, certamente, a seqüência \tilde{PN}' conterá valores diferentes de 0 e 1. Tais valores são tratados como indeterminados, sendo, portanto, neutros no processo de decisão de cada bit que compõe a marca. Assim, quanto maior o valor de L , maior será o número de bits válidos para auxiliar a extração da marca.

É importante ressaltar ainda que os mapeamentos dimensionais realizados na extração da marca correspondem ao inverso dos mapeamentos utilizados na inserção da mesma. Para realizar as conversões $2D \rightarrow 1D$, informações bidimensionais são lidas linha por linha, da esquerda para a direita e de cima para baixo, e escritas num vetor linha. As conversões $1D \rightarrow 2D$ são realizadas utilizando-se um processo inverso ao descrito. Qualquer erro na ordenação das informações que compõem estes esquemas compromete o método por completo.

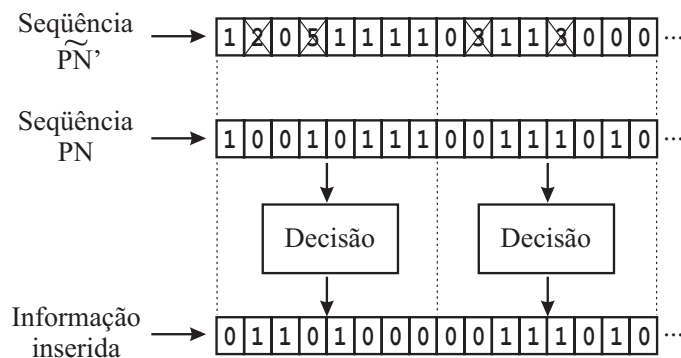


Figura 3.6: Funcionamento do bloco “Decisão” na extração da marca d'água ($L = 8$ bits).

3.6.2 Simulações e resultados

O aspecto de maior interesse numa técnica de proteção de informação digital é a avaliação de sua robustez. Esta característica mede a resistência da marca d'água a manipulações que a informação possa sofrer, determinando sob que condições é possível reconhecer a presença da mesma marca. Com o objetivo de analisar o comportamento do método proposto neste aspecto, os esquemas de inserção e extração da marca d'água foram implementados no *Matlab*. Os resultados das simulações realizadas são apresentados e discutidos a seguir.

Originalmente, considerou-se uma imagem em escala de cinza, de tamanho 128×128 pixels, livre de qualquer marca d'água e que não tenha sofrido compressão (*lenna.bmp*). O passo seguinte foi realizar a inserção de uma marca representada por uma imagem binária, de tamanho 32×32 pixels. Na Figura 3.7, são apresentadas estas duas imagens e uma terceira, obtida ao final do procedimento descrito.

Com o propósito de medir o quanto a inserção da marca d'água modifica a imagem original, é comum calcular-se a relação sinal-ruído de pico (PSNR). Para a imagem em questão, obteve-se $PSNR = 38,98$ dB, valor cujo significado visual pode-se observar na Figura 3.7.

A etapa seguinte da simulação foi recuperar a marca d'água apresentada. Como se tem enfatizado, a técnica proposta neste trabalho emprega uma ferramenta matemática baseada

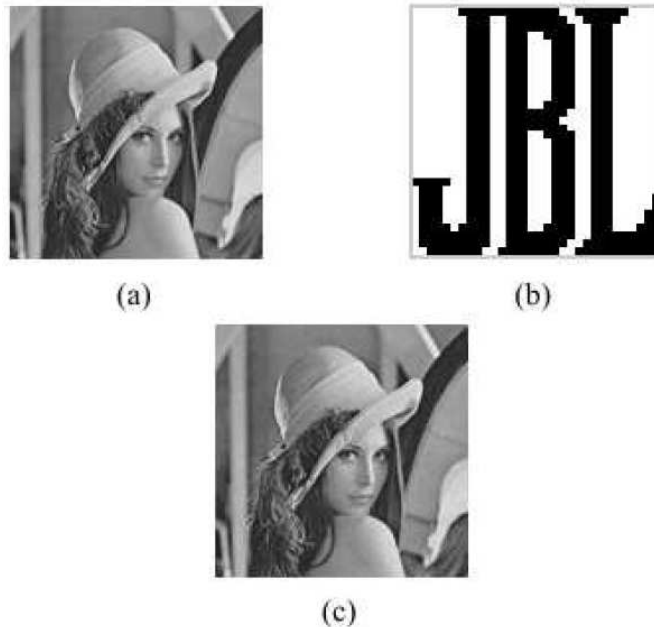


Figura 3.7: (a) Imagem original, 128×128 (*lenna.bmp*). (b) Marca d'água, 32×32 . (c) Imagem marcada, $PSNR = 38,98$ dB.



Figura 3.8: (a) Imagem marcada com brilho acentuado em 30%. (b) Marca d'água recuperada, $CN = 0,9796$.

em aritmética inteira. Por conta disso, os procedimentos de inserção e extração (Figuras 3.4 e 3.5) são precisos, proporcionando a recuperação de uma marca idêntica à que foi inserida (desde que a imagem não tenha sofrido alterações).

Finalmente, foram modificadas determinadas características da imagem marcada com o objetivo de analisar o reflexo de tais manipulações sobre a marca extraída. A medida do quanto a marca foi corrompida é dada pela correlação normalizada (CN) entre a marca recuperada e a marca inserida [60]. Este parâmetro é calculado pela equação

$$CN = \frac{\sum_i \sum_j m(i, j) \tilde{m}(i, j)}{\sum_i \sum_j [m(i, j)]^2}, \quad (3.53)$$

em que m representa a marca inserida e \tilde{m} representa a marca recuperada. Quanto mais próximo de 1 for o valor de CN , maior similaridade haverá entre as duas marcas.

A Figura 3.8.a corresponde à imagem marcada com brilho acentuado em 30%. Este tipo de alteração equivale a somar ao valor de cada pixel uma constante inteira. Na $FFCT_{2D} \ 2 \times 2$ ($p = 7$), o efeito desta manipulação é neutro. Deste modo, é possível recuperar a marca com bastante clareza (Figura 3.8.b), havendo erro apenas devido ao *overflow* (pixels que excederam 255). Para este caso, obteve-se $CN = 0,9796$.

Na Figura 3.9.a, é apresentada uma imagem marcada com uma região completamente destruída. Na parte inferior da marca recuperada (Figura 3.9.b), observa-se que linhas alternadas foram corrompidas. Este padrão é reflexo do modo como a marca foi espalhada e dos mapeamentos dimensionais utilizados no procedimento de inserção. Ainda assim, com $CN = 0,8662$, consegue-se identificar claramente a presença da marca original.

Apresenta-se na Figura 3.10 as marcas recuperadas ao se realizar outras duas manipulações na imagem. A Figura 3.10.a corresponde à marca extraída quando se comprime, segundo o padrão JPEG, a imagem marcada em 30% do seu tamanho ($CN = 0,6747$). Ao se incre-



Figura 3.9: (a) Imagem marcada com 25% de sua informação destruída. (b) Marca d'água recuperada, $CN = 0,8662$.

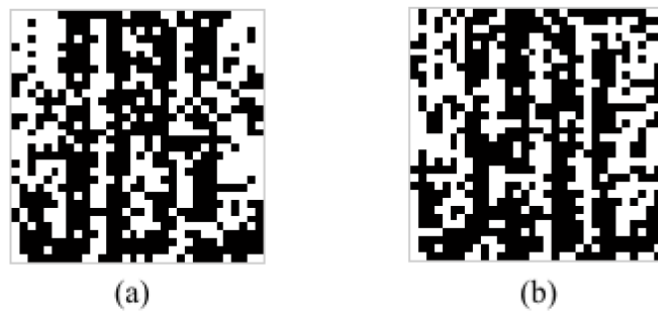


Figura 3.10: (a) Marca recuperada ao se comprimir, segundo o padrão JPEG, o arquivo original em 30% do seu tamanho, $CN = 0,6747$. (b) Marca recuperada ao se incrementar em 10% o contraste da imagem marcada, $CN = 0,7175$.

mentar em 10% o contraste da imagem marcada, extrai-se com similaridade $CN = 0,7175$ a marca apresentada na Figura 3.10.b.

A partir dos resultados apresentados nas Figuras 3.8, 3.9 e 3.10, pode-se analisar alguns aspectos do método proposto. A técnica baseada na transformada do co-seno sobre corpos finitos é pouco sensível a manipulações que alteram a imagem marcada de modo uniforme (brilho) ou esparsa (destruição total de determinadas regiões). Como a $FFCT_{2D}$ implementada possui tamanho 2×2 , é importante que o maior número possível destes blocos “sobreviva” a alterações.

Imaginando que cada manipulação sofrida pela imagem marcada seja modelada como um ruído aditivo (matriz com o mesmo tamanho da imagem), observa-se que, quanto menor a variância desse ruído, maior a similaridade entre a marca extraída e aquela originalmente inserida. Se a referida variância for elevada, menos denso deve ser o ruído, isto é, um menor número de pixels da imagem marcada deve ser alterado para que a marca seja preservada. De qualquer forma, a técnica proposta permite verificar e localizar alterações que tenham sido

realizadas na imagem.

3.6.3 Filtragem de imagens via FFTT

A segunda aplicação diretamente relacionada às propriedades das FFTT baseia-se na convolução simétrica dessas transformadas. Especificamente, discute-se o emprego desta propriedade para implementar a filtragem de imagens digitais. Nesse cenário, os sinais de entrada são matrizes de elementos inteiros com valores limitados de 0 a 255 (imagens em escala de cinza); um filtro simples é definido usando números inteiros e um fator de normalização [64]. De maneira semelhante ao que foi apresentado na Definição 3.1 e no Teorema 3.1, versões bidimensionais de outras FFTT podem ser obtidas a partir das respectivas expressões em uma dimensão.

Além dos fatores mencionados, outro aspecto a ser considerado no uso das FFTT, particularmente na filtragem digital, é a escolha criteriosa do número primo p no qual se baseará todo o procedimento. O papel fundamental das ferramentas sobre corpos finitos em aplicações de Processamento de Sinais é servir como um veículo que possibilita maior precisão e eficiência computacional. Para que isso seja conseguido, deve-se escolher um primo p e, conseqüentemente, um corpo finito $GF(p)$ em que os cálculos envolvidos na aplicação possam ser realizados sem ocasionar *overflow*. Mais adiante, essa restrição é tratada em maiores detalhes.

A proposta de filtragem de imagens via FFTT pode ser melhor ilustrada através de um exemplo. Para isso, considera-se uma imagem em escala de cinza $x = (x_{i,j})$ (sinal de entrada) a ser processada por um filtro FIR passa-baixas cuja resposta ao impulso $h = (h_{i,j})$ é

$$h_{i,j} = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix}. \quad (3.54)$$

De acordo com o que foi apresentado na Seção 3.5, para realizar a operação de filtragem utilizando-se convolução simétrica, é suficiente considerar

$$h_{i,j}^r = \frac{1}{16} \begin{pmatrix} 4 & 2 \\ 2 & 1 \end{pmatrix}, \quad (3.55)$$

de onde $h_{i,j}$ pode ser obtido a partir de uma extensão simétrica bidimensional do tipo WS. A este tipo de simetria está associada a $FFCT_{2D-1e}$, que corresponde à transformada τ_b na Equação 3.29. De fato, será calculada a transformada de $16h_{i,j}^r$. A multiplicação por 16 é

necessária para cancelar temporariamente o efeito do fator de normalização, que determina elementos não-inteiros na resposta ao impulso do filtro. Na Equação 3.29, escolhendo a $FFCT_{2D-2e}$ como a transformada τ_a , a seguinte convolução simétrica é possível [52]:

$$\hat{w}_{i,j} = FC_{2D,2e}^{-1} \{ FC_{2D,2e} \{ \hat{x}_{i,j} \} \times FC_{2D,1e} \{ 16h_{i,j}^r \} \}. \quad (3.56)$$

Na Equação 3.56, $\hat{x}_{i,j}$ corresponde a blocos de $x_{i,j}$ complementados com um número suficiente de zeros, de modo a garantir que o resultado da convolução simétrica seja o mesmo da convolução linear. A imagem filtrada $w_{i,j}$ é formada a partir dos blocos $\hat{w}_{i,j}$, superpostos organizadamente segundo o método de *overlap-add* [44]. No final deste processo, um escalonamento por um fator de 1/16 é realizado, com a finalidade de retornar os valores dos pixels para o intervalo [0, 255].

Conforme comentado, a definição do corpo finito $GF(p)$ apropriado para implementar todo o procedimento descrito está diretamente atrelada à necessidade de se evitar o *overflow* na realização da convolução. Para isso, é possível estabelecer uma condição a partir da Equação 3.28. Nessa expressão, verifica-se que o limitante superior da seqüência de saída w é dado pelo produto entre o componente máximo de \tilde{x} e a soma de todos os componentes de alguma submatriz (ou subvetor) de h . Esse raciocínio pode ser estendido ao caso bidimensional. Para o exemplo considerado, o componente máximo de $x_{i,j}$ é 255; a submatriz de $h_{i,j}$ cuja soma dos componentes é máxima, e igual a 1, é a própria matriz $h_{i,j}$. Assim, o limitante superior da matriz resultante da filtragem é 255.

Com base na informação obtida, escolheu-se utilizar $GF(8191)$. Este corpo, além de ser suficientemente grande para evitar *overflow*, permite implementar FFTT com $N = 8$ e simplificar os cálculos, uma vez que p é da forma $2^k - 1$. Usando o elemento unimodular $\zeta = 812 + j1735$, encontrado através de um procedimento de busca e que possui ordem multiplicativa $2N = 16$ sobre $GF(8191)$, define-se uma $FFCT_{2D-1e}$ e uma $FFCT_{2D-2e}$ com dimensões 8×8 .

Uma simulação computacional do procedimento descrito foi realizada. Após o processamento da imagem *carnival.bmp* por um filtro passa-baixas, o efeito da suavização em suas linhas é bastante nítido (Figuras 3.11 e 3.12). Neste resultado, um aspecto de grande relevância é a perfeita igualdade entre a imagem filtrada usando a convolução simétrica e a imagem obtida através de uma convolução linear direta com o filtro. Isso acontece porque as FFTT não requerem qualquer arredondamento, o que significa que não há introdução de ruído computacional. Usando a convolução simétrica das DTT para realizar essa filtragem, uma

PSNR = 63,26 dB é obtida quando se compara o resultado com o cálculo direto da convolução.

Um outro exemplo é apresentado nas Figuras 3.13 e 3.14. Neste caso, a imagem *lancer.bmp* é convoluída com a resposta ao impulso de um filtro dada por

$$h_{i,j} = \frac{1}{7} \begin{pmatrix} -1 & -2 & -1 \\ -2 & 19 & -2 \\ -1 & -2 & -1 \end{pmatrix}, \quad (3.57)$$

tal que

$$h_{i,j}^r = \frac{1}{7} \begin{pmatrix} 19 & -2 \\ -2 & -1 \end{pmatrix}. \quad (3.58)$$

Como no exemplo anterior, a convolução linear é implementada através da convolução simétrica dada pela Equação 3.56 e as FFTT são definidas sobre GF(8191). Naturalmente, a imagem filtrada usando a convolução simétrica e a imagem obtida através de uma convolução linear direta com o filtro também são iguais. Usando a convolução simétrica das DTT para realizar esta filtragem, uma PSNR = 66,17 dB é obtida quando se compara o resultado com o cálculo direto da convolução.

Nos exemplos apresentados, os valores da PSNR alcançados com o uso da convolução simétrica das DTT são relativamente altos. Isso significa que, visualmente, a diferença entre a imagem filtrada através desta técnica e o resultado livre de imprecisão é pouca. Entretanto, com a realização repetida deste processamento, a tendência é que o erro se propague, o que é indesejável em diversos cenários de aplicação. Esse é um dos aspectos que, no presente contexto, tornam importante a disponibilização de ferramentas de corpos finitos, particularmente, das FFTT.



Figura 3.11: *carnival.bmp* com significativo conteúdo em altas frequências.



Figura 3.12: *carnival.bmp* após filtragem passa-baixas implementada pela convolução simétrica das FFT.



Figura 3.13: *Imagem original lancer.bmp.*



Figura 3.14: *lancer.bmp após filtragem passa-altas implementada pela convolução simétrica das FFT.*

AUTO-ESTRUTURA DAS FFT

O ESTUDO das transformadas discretas sob o ponto de vista matricial tem permitido o desenvolvimento de ferramentas e a observação de uma série de propriedades relacionadas a essas transformadas. Ao se interpretar o cálculo da transformada de uma seqüência como uma multiplicação entre um vetor e uma matriz de transformação, é possível, por exemplo, derivar algoritmos rápidos baseados nas simetrias e na fatoração da referida matriz [15]. A auto-estrutura de uma matriz de transformação, isto é, a forma como se relacionam os autovalores e os autovetores da mesma, é, nesse contexto, outro aspecto de importância relevante. No caso da transformada discreta de Fourier, que foi estudado pela primeira vez em 1972, demonstra-se que sua matriz tem, no máximo, quatro autovalores distintos - $\{1, -1, j, -j\}$ - e determinam-se formas gerais para os autovetores associados [65]. Tais conclusões têm relação com o que se denomina *período* de uma transformada, o número mínimo de vezes (maior que zero) em que se precisa iterativamente calcular a transformada de uma seqüência até que se retorne à seqüência original. Para a DFT, particularmente, o período vale quatro [65], [66].

Recentemente, mostrou-se que os autovetores da DFT são necessários à definição da transformada discreta fracional de Fourier (DFRFT), para a qual diversas aplicações têm sido propostas [67], [68], [69], [70]. De forma semelhante, tem-se investigado a auto-estrutura das DCT e DST, cujos resultados têm sido empregados na definição de transformadas discretas fracionais do co-seno e do seno, respectivamente denotadas por DFRCT e DFRST [71], [72], [73]. Para as transformadas trigonométricas discretas fracionais, têm sido propostas aplicações em projetos de filtros, compressão de dados, Criptografia e marca d'água [33].

Quando se fala em transformadas sobre corpos finitos, é provável que, anteriormente ao presente trabalho, o estudo de auto-estruturas tenha sido realizado em apenas uma ocasião, quando a transformada numérica de Fourier (NTT, *number theoretic transform*) foi caracterizada segundo esse aspecto [74]. Neste capítulo, investiga-se a auto-estrutura das transformadas trigonométricas sobre corpos finitos. Para diferentes tipos de transformadas do co-seno (FFCT) e do seno (FFST), com simetria par, são derivadas proposições e discutidas conjecturas que permitem avaliar as principais características de seus autovalores e autovetores. Os resultados obtidos esclarecem alguns pontos do uso das FFTT na formatação de distribuições de probabilidade sobre os inteiros, que é apresentado na parte final do capítulo. De modo particular, as FFCT são utilizadas na obtenção de uma distribuição uniforme a partir de uma distribuição qualquer. Tal procedimento significa esconder o comportamento estatístico de uma fonte de informação, o que é desejável em muitas situações no contexto de Criptografia. Com base nisso, alguns cenários de aplicação são sugeridos [38].

4.1 Auto-estrutura das FFTT do tipo 1e

A auto-estrutura da FFCT-1e e a da FFST-1e, estudadas nesta seção, possuem uma forte ligação com a da transformada de Fourier de corpo finito. Por conta dessa relação, inicialmente, a auto-estrutura da NTT é revista [74]. Assim como no Capítulo 2, na presente abordagem, expressa-se o cálculo de uma transformada através de uma equação matricial, isto é, a transformada de uma seqüência ou vetor linha \mathbf{x} é um vetor coluna \mathbf{X} obtido por

$$\mathbf{X} = \mathbf{x} \times \mathbf{M}_N^T, \quad (4.1)$$

em que \mathbf{M}_N é uma matriz de transformação específica, com dimensões $N \times N$. Considerando a Equação 4.1, a NTT de uma seqüência ou vetor linha $\mathbf{x} = (x_i), i = 0, \dots, N - 1, x_i \in \text{GF}(p)$, é um vetor coluna $\mathbf{X} = (X_k), X_k \in \text{GF}(p), k = 0, \dots, N - 1$, obtido quando a matriz \mathbf{M} é substituída por

$$\mathbf{FF}_N = \sqrt{N^{-1}} \cdot \alpha^{i \cdot k}. \quad (4.2)$$

Na última equação, α é um elemento com ordem multiplicativa $\text{ord}(\alpha) = N$ em $\text{GF}(p)$. O fator de escala $\sqrt{N^{-1}}$ torna a matriz \mathbf{FF}_N unitária.

Proposição 4.1 *A matriz de transformação da NTT possui, no máximo, quatro autovalores distintos – $\{1, -1, j, -j\}$ – cujas multiplicidades são apresentadas na Tabela 4.1.*

Demonstração: A determinação dos autovalores da NTT baseia-se na propriedade da dualidade, derivada diretamente da definição dessa transformada. Denotando por $x_i \xleftrightarrow{NTT} X_k$ a relação entre o vetor \mathbf{x} e a sua transformada de Fourier de corpo finito \mathbf{X} , a relação $X_i \xleftrightarrow{NTT} x_{-k}$ é válida. Aplicando a NTT de forma iterativa à seqüência X_i , na última relação, e utilizando a dualidade, os seguintes pares transformados são seqüencialmente obtidos:

$$\begin{aligned} x_{-i} &\xleftrightarrow{NTT} X_{-k}, \\ X_{-i} &\xleftrightarrow{NTT} x_k. \end{aligned}$$

Do ponto de vista matricial, os pares transformados apresentados permitem concluir que $(\mathbf{FF}_N)^4 = \mathbf{I}_N$, em que \mathbf{I}_N é a matriz identidade com dimensões $N \times N$. Denotando por λ um autovalor de \mathbf{FF}_N , o respectivo autovetor \mathbf{x} satisfaz $\mathbf{FF}_N \cdot \mathbf{x}^T = \lambda \cdot \mathbf{x}^T$. Assim, pode-se escrever

$$(\mathbf{FF}_N)^4 \cdot \mathbf{x}^T = \lambda^4 \cdot \mathbf{x}^T \quad \therefore \quad \lambda^4 = 1,$$

de onde se obtêm os autovalores λ propostos. As multiplicidades apresentadas na Tabela 4.1 são calculadas usando argumentos semelhantes àqueles empregados no estudo da auto-estrutura da DFT [65]. Tal demonstração é omitida. ■

Tabela 4.1: *Multiplicidades dos autovalores da matriz de transformação da transformada de Fourier de corpo finito com dimensões $N \times N$.*

N	Mult. de 1	Mult. de -1	Mult. de j	Mult. de $-j$
$4 \cdot m$	$m + 1$	m	$m - 1$	m
$4 \cdot m + 1$	$m + 1$	m	m	m
$4 \cdot m + 2$	$m + 1$	m	m	m
$4 \cdot m + 3$	$m + 1$	$m + 1$	m	$m + 1$

Proposição 4.2 *Todos os autovetores associados à NTT possuem simetria par ou ímpar. Os autovetores pares estão relacionados aos autovalores 1 ou -1 e os autovetores ímpares estão relacionados aos autovalores j ou $-j$.*

Demonstração: Se \mathbf{x} é um autovetor da matriz da NTT, a aplicação da propriedade da dualidade ao par transformado $x_i \xleftrightarrow{NTT} X_k$ resulta em $\lambda \cdot x_i \xleftrightarrow{NTT} x_{-k}$. Portanto, $\lambda \cdot x_k = x_{-k}$ e, conseqüentemente, $x_i = x_{-i}$ (\mathbf{x} possui simetria par), caso $\lambda = \pm 1$, e $x_i = -x_{-i}$ (\mathbf{x} possui simetria ímpar), caso $\lambda = \pm j$. ■

Seqüências com simetria par e com simetria ímpar são utilizadas para construir autovetores da NTT de acordo com a proposição a seguir.

Proposição 4.3 *Os autovetores da matriz de transformação da NTT são construídos de acordo com as relações enunciadas a seguir. Se \mathbf{x} e \mathbf{X} são, respectivamente, um vetor de comprimento N com componentes em $GF(p)$ e sua NTT, então:*

▷ o vetor com simetria par $\mathbf{x} = \mathcal{E}\{x_i\} \pm \mathcal{E}\{X_i\}$ é um autovetor da matriz \mathbf{FF}_N associado ao autovalor $\lambda = \pm 1$.

▷ o vetor com simetria ímpar $\mathbf{x} = \mathcal{O}\{x_i\} \mp j \cdot \mathcal{O}\{X_i\}$ é um autovetor da matriz \mathbf{FF}_N associado ao autovalor $\lambda = \pm j$.

A proposição acima pode ser demonstrada utilizando argumentos análogos àqueles empregados nas demonstrações das Proposições 3 e 4 em [66]. Outras maneiras de construir autovetores da NTT são discutidas em [74].

Com base no conteúdo apresentado, é possível enunciar as seguintes proposições relacionadas à auto-estrutura da FFCT-1e e à da FFST-1e.

Proposição 4.4 *Os autovetores da FFCT-1e e os da FFST-1e são construídos a partir dos autovetores da NTT de acordo com as relações enunciadas a seguir.*

▷ Se $\mathbf{x} = [x_0, x_1, \dots, x_{N-2}, x_{N-1}, x_{N-2}, \dots, x_1]$ for um autovetor par da matriz \mathbf{FF}_{2N-2} , ou seja, $\mathbf{FF}_{2N-2} \cdot \mathbf{x}^T = \lambda \cdot \mathbf{x}^T$ ($\lambda = 1, -1$), então

$$\hat{\mathbf{x}} = [x_0, \sqrt{2} \cdot x_1, \dots, \sqrt{2} \cdot x_{N-2}, x_{N-1}] \quad (4.3)$$

será um autovetor da matriz $\mathbf{FC}_{N,1e}$, ou seja, $\mathbf{FC}_{N,1e} \cdot \hat{\mathbf{x}}^T = \lambda \cdot \hat{\mathbf{x}}^T$ ($\lambda = 1, -1$).

▷ Se $\mathbf{x} = [0, x_1, x_2, \dots, x_N, 0, -x_N, -x_{N-1}, \dots, -x_1]$ for um autovetor ímpar da matriz \mathbf{FF}_{2N+2} , ou seja, $\mathbf{FF}_{2N+2} \cdot \mathbf{x}^T = \lambda \cdot \mathbf{x}^T$ ($\lambda = j, -j$), então

$$\tilde{\mathbf{x}} = \sqrt{2} \cdot [x_1, x_2, \dots, x_N] \quad (4.4)$$

será um autovetor da matriz $\mathbf{FS}_{N,1e}$ com autovalor correspondente $j \cdot \lambda$, ou seja, $\mathbf{FS}_{N,1e} \cdot \tilde{\mathbf{x}}^T = j \cdot \lambda \cdot \tilde{\mathbf{x}}^T$ ($\lambda = j, -j$). Assim, os autovalores associados à FFST-1e são 1 e -1 .

Demonstração: Vide Apêndice B. ■

Proposição 4.5 *Os únicos autovalores das matrizes de transformação da FFCT-1e e da FFST-1e são 1 e -1 . Suas multiplicidades são apresentadas nas Tabelas 4.2 e 4.3, respectivamente.*

Tabela 4.2: Multiplicidades dos autovalores da matriz de transformação da transformada do co-seno de corpo finito do tipo 1e com dimensões $N \times N$.

N	Mult. de 1	Mult. de -1
ímpar	$\frac{N+1}{2}$	$\frac{N-1}{2}$
par	$\frac{N}{2}$	$\frac{N}{2}$

Tabela 4.3: Multiplicidades dos autovalores da matriz de transformação da transformada do seno de corpo finito do tipo 1e com dimensões $N \times N$.

N	Mult. de 1	Mult. de -1
ímpar	$\frac{N+1}{2}$	$\frac{N-1}{2}$
par	$\frac{N}{2}$	$\frac{N}{2}$

Demonstração: A partir da Proposição 4.4, sabe-se que os autovetores da matriz $\mathbf{FC}_{N,1e}$ estão relacionados aos autovetores com simetria par da matriz \mathbf{FF}_{2N-2} . Se N for par, pode-se escrevê-lo sob a forma $N = 2 \cdot (N' + 1)$, em que $N' \geq 0$ é um número inteiro. Assim, $2N - 2 = 4 \cdot N' + 2$. Observando a Tabela 4.1, verifica-se que, neste caso, as multiplicidades dos autovalores 1 e -1 , respectivamente denotadas por $\#(1)$ e $\#(-1)$, são iguais a $\#(1) = \#(-1) = N' + 1 = N/2$.

Se N for ímpar, pode-se escrevê-lo sob a forma $N = 2N' + 1$, em que $N' \geq 0$ é um número inteiro. Assim, $2N - 2 = 4 \cdot N'$. Daí, analogamente ao caso em que N é par, tem-se $\#(1) = N' + 1 = (N + 1)/2$ e $\#(-1) = N' - 1 = (N - 1)/2$. Os resultados para a FFST-1e são demonstrados utilizando argumentos semelhantes. ■

A verificação de que os únicos autovalores associados às matrizes de transformação da FFCT-1e e da FFST-1e são 1 e -1 também pode ser realizada a partir da propriedade de involução dessas transformadas. Um autovetor $\hat{\mathbf{x}}$ da matriz $\mathbf{FC}_{N,1e}$, por exemplo, satisfaz $\mathbf{FC}_{N,1e} \cdot \hat{\mathbf{x}}^T = \lambda \cdot \hat{\mathbf{x}}^T$. Aplicando a FFCT-1e a ambos os lados da última expressão, obtém-se

$$(\mathbf{FC}_{N,1e})^2 \cdot \hat{\mathbf{x}}^T = \lambda^2 \cdot \hat{\mathbf{x}}^T. \quad (4.5)$$

Conforme previamente comentado, uma vez que a FFCT-1e é involucionária, tem-se $(\mathbf{FC}_{N,1e})^2 = \mathbf{I}_N$. Dessa maneira, a Equação (4.5) reduz-se a $\lambda^2 = 1$, cujas únicas soluções são $\lambda = \pm 1$. Procedimento análogo aplica-se à FFST-1e.

4.2 Auto-estrutura das FFTT do tipo 4e

Nesta seção, a auto-estrutura das matrizes de transformação da FFCT e da FFST do tipo 4e são estudadas. De forma semelhante às transformadas do tipo 1e, no presente caso, também existe uma ligação com os autovalores e autovetores da transformada de Fourier de corpo finito. No entanto, conforme apresentado em [75], o estudo da auto-estrutura das transformadas trigonométricas do tipo 4e requer a definição de uma versão generalizada da transformada de Fourier. Na referência supra-citada, pelo fato de se estar considerando ferramentas sobre os números reais, introduziu-se a GDFT (transformada discreta de Fourier generalizada). Aqui, define-se a transformada de Fourier de corpo finito generalizada (considerando apenas a versão numérica), a qual é denotada por GFFFT. Inicialmente, a auto-estrutura da GFFFT é analisada e, com base na mesma, proposições acerca da auto-estrutura da FFCT e da FFST do tipo 4e são derivadas.

4.2.1 A transformada de Fourier de corpo finito generalizada

A GFFFT de uma seqüência ou vetor linha $\mathbf{x} = (x_i)$, $i = 0, \dots, N-1$, $x_i \in \text{GF}(p)$, é um vetor coluna $\mathbf{X} = (X_k)$, $X_k \in \text{GF}(p)$, $k = 0, \dots, N-1$, calculado pela Equação 4.1, em que a matriz de transformação \mathbf{M} é substituída por

$$\mathbf{FF}_{N,G} = \sqrt{N^{-1}} \cdot \alpha^{(i+\frac{1}{2}) \cdot (k+\frac{1}{2})}. \quad (4.6)$$

Na relação acima, α é um elemento com ordem multiplicativa $\text{ord}(\alpha) = N$ em $\text{GF}(p)$ e $p \equiv 3 \pmod{4}$. O fator de escala $\sqrt{N^{-1}}$ torna a matriz $\mathbf{FF}_{N,G}$ unitária. Sua inversa é dada por

$$(\mathbf{FF}_{N,G})^{-1} = \sqrt{N^{-1}} \cdot \alpha^{-(i+\frac{1}{2}) \cdot (k+\frac{1}{2})}. \quad (4.7)$$

A seguir, algumas propriedades utilizadas no estudo da auto-estrutura de $\mathbf{FF}_{N,G}$, a matriz de transformação da GFFFT, são apresentadas.

Propriedade 4.1 *Seja \mathbf{J} uma matriz com dimensões $N \times N$ composta por 1's na antidiagonal e 0's nas demais posições. Então, $(\mathbf{FF}_{N,G})^2 = -\mathbf{J}$.*

Demonstração: O elemento na $(i + 1)$ -ésima linha e na $(k + 1)$ -ésima coluna de $(\mathbf{FF}_{N,G})^2$, denotado por $(\mathbf{FF}_{N,G})_{i,k}^2$, $i, k = 0, 1, \dots, N - 1$, é computado por

$$\begin{aligned} (\mathbf{FF}_{N,G})_{i,k}^2 &= \sum_{l=0}^{N-1} (\mathbf{FF}_{N,G})_{i,l} \cdot (\mathbf{FF}_{N,G})_{l,k} \\ &= N^{-1} \cdot \sum_{l=0}^{N-1} \alpha^{(i+\frac{1}{2}) \cdot (l+\frac{1}{2})} \cdot \alpha^{(l+\frac{1}{2}) \cdot (k+\frac{1}{2})} \\ &= N^{-1} \cdot \alpha^{\frac{1}{2} \cdot (i+k+1)} \cdot \sum_{l=0}^{N-1} \alpha^{l \cdot (i+k+1)}. \end{aligned}$$

Como $1 \leq i + k + 1 \leq 2N - 1$, $\alpha^{\frac{N}{2}} = -1$, e

$$\sum_{l=0}^{N-1} \alpha^{l \cdot (i+k+1)} = \begin{cases} N, & \text{para } i + k + 1 = N, \\ 0, & \text{caso contrário,} \end{cases} \quad (4.8)$$

tem-se

$$(\mathbf{FF}_{N,G})_{i,k}^2 = \begin{cases} -1, & \text{para } i + k + 1 = N, \\ 0, & \text{caso contrário.} \end{cases} \quad (4.9)$$

■

Uma vez que a matriz $(\mathbf{FF}_{N,G})^2 = -\mathbf{J}$ está relacionada ao cálculo por duas vezes sucessivas da GFFFT de um vetor, a Propriedade 4.1 pode ser interpretada como descrito a seguir. Denotando por $x_i \xleftrightarrow{G} X_k$ a relação entre o vetor \mathbf{x} e a sua transformada de Fourier de corpo finito generalizada \mathbf{X} , a relação $X_i \xleftrightarrow{G} -x_{-k-1}$ é válida.

A propriedade seguinte observa particularidades relacionadas à GFFFT de vetores que possuem simetria. Diferentemente dos resultados apresentados para a FFFT, neste caso, considera-se que vetores \mathbf{x}_e com simetria par atendem à condição $x_{e,i} = x_{e,-i-1}$. Os mesmos podem ser construídos a partir de um vetor \mathbf{x} pela equação $x_{e,i} = 2^{-1} \cdot (x_i + x_{-i-1})$, o que equivale a dizer que $x_{e,i} = \mathcal{E}\{x_i\}$ é a parte par de \mathbf{x} ; vetores com simetria ímpar satisfazem $x_{o,i} = -x_{o,-i-1}$ e podem ser obtidos por $x_{o,i} = 2^{-1} \cdot (x_i - x_{-i-1})$. Analogamente, diz-se que $x_{o,i} = \mathcal{O}\{x_i\}$ é a parte ímpar de \mathbf{x} .

Propriedade 4.2 *Se a relação entre um vetor \mathbf{x} e a sua GFFFT, o vetor \mathbf{X} , é representada por $x_i \xleftrightarrow{G} X_k$, então, as relações $\mathcal{E}\{x_i\} \xleftrightarrow{G} \mathcal{E}\{X_k\}$ e $\mathcal{O}\{x_i\} \xleftrightarrow{G} \mathcal{O}\{X_k\}$ são válidas.*

Demonstração: Calculando a GFFFT de $\mathcal{E}\{x_i\} = x_{e,i} = 2^{-1} \cdot (x_i + x_{-i-1})$, obtém-se

$$\left(2 \cdot \sqrt{N}\right)^{-1} \cdot \left[\sum_{i=0}^{N-1} x_i \cdot \alpha^{(i+\frac{1}{2}) \cdot (k+\frac{1}{2})} + \sum_{i=0}^{N-1} x_{-i-1} \cdot \alpha^{(i+\frac{1}{2}) \cdot (k+\frac{1}{2})} \right].$$

Realizando a substituição de índices $r = -i - 1$ no segundo somatório da última expressão, a mesma pode ser reescrita como

$$\begin{aligned} & \left(2 \cdot \sqrt{N}\right)^{-1} \cdot \left[\sum_{i=0}^{N-1} x_i \cdot \alpha^{(i+\frac{1}{2}) \cdot (k+\frac{1}{2})} + \sum_{r=0}^{N-1} x_r \cdot \alpha^{(r+\frac{1}{2}) \cdot (-k-1+\frac{1}{2})} \right] \\ & = 2^{-1} \cdot (X_k + X_{-k-1}) = \mathcal{E}\{X_k\}. \end{aligned}$$

Demonstra-se a validade da relação associada à parte ímpar de \mathbf{x} de forma análoga. ■

Proposição 4.6 *A matriz de transformação da GFFFT possui, no máximo, quatro autovalores distintos $\{-1, -1, j, -j\}$ – cujas multiplicidades são apresentadas na Tabela 4.4.*

Tabela 4.4: *Multiplicidades dos autovalores da matriz de transformação da transformada de Fourier de corpo finito generalizada com dimensões $N \times N$.*

N	Mult. de 1	Mult. de -1	Mult. de j	Mult. de $-j$
$4 \cdot m$	m	m	m	m
$4 \cdot m + 1$	m	m	m	$m + 1$
$4 \cdot m + 2$	$m + 1$	m	m	$m + 1$
$4 \cdot m + 3$	$m + 1$	m	$m + 1$	$m + 1$

Demonstração: Os autovalores da matriz $\mathbf{FF}_{N,G}$ são obtidos através de um argumento análogo ao discutido após a demonstração da Proposição 4.5. Neste caso, utilizando a Propriedade 4.1, sabe-se que $(\mathbf{FF}_{N,G})^4 = \mathbf{I}_N$. Conseqüentemente, os autovalores de $\mathbf{FF}_{N,G}$ correspondem às soluções da equação $\lambda^4 = 1$, isto é, $\{1, -1, j, -j\}$. Para determinar suas multiplicidades, os mesmos são denotados por λ_n , $n = 1, 2, \dots, N$. Assim, tem-se

$$\sum_{n=1}^N \lambda_n^2 = \text{traço} \left\{ (\mathbf{FF}_{N,G})^2 \right\} = \text{traço} \{-\mathbf{J}\} = \begin{cases} -1, & N \text{ é ímpar,} \\ 0, & N \text{ é par} \end{cases} \quad (4.10)$$

e

$$\lambda_n^2 = \begin{cases} 1, & \text{se } \lambda_n = \pm 1, \\ -1, & \text{se } \lambda_n = \pm j. \end{cases} \quad (4.11)$$

Denotando por N_1 o número total de autovalores $\{1, -1\}$ e por N_2 o de $\{j, -j\}$, a partir das Equações (4.10) e (4.11), conclui-se que

$$N_2 - N_1 = \begin{cases} 0, & N \text{ é par,} \\ 1, & N \text{ é ímpar.} \end{cases}$$

Tabela 4.5: Somas das multiplicidades dos autovalores $\{1, -1\}$ e $\{j, -j\}$ da matriz de transformação da transformada de Fourier de corpo finito generalizada com dimensões $N \times N$.

N	Mult. de ± 1	Mult. de $\pm j$
$4 \cdot m$	$2 \cdot m$	$2 \cdot m$
$4 \cdot m + 1$	$2 \cdot m$	$2 \cdot m + 1$
$4 \cdot m + 2$	$2 \cdot m + 1$	$2 \cdot m + 1$
$4 \cdot m + 3$	$2 \cdot m + 1$	$2 \cdot m + 2$

Com base neste resultado, é possível construir a Tabela 4.5.

Além disso, tem-se [76]

$$\sum_{n=1}^N \lambda_n = \text{traço} \{ \mathbf{F}\mathbf{F}_{N,G} \} = \sqrt{N^{-1}} \cdot \sum_{i=0}^{N-1} \alpha^{(i+\frac{1}{2})^2} = \begin{cases} 0, & N = 4 \cdot m, \\ -j, & N = 4 \cdot m + 1, \\ 1 - j, & N = 4 \cdot m + 2, \\ 1, & N = 4 \cdot m + 3. \end{cases} \quad (4.12)$$

Combinando os resultados da Tabela 4.5 e da Equação (4.12), a Tabela 4.4 é obtida. ■

Nas proposições a seguir, descreve-se as características dos autovetores da matriz de transformação da GFFFT. Com base na Propriedade 4.2 e em [66], apresenta-se um procedimento sistemático para a construção de autovetores associados a autovalores específicos de $\mathbf{F}\mathbf{F}_{N,G}$.

Proposição 4.7 *Qualquer autovetor \mathbf{x} da matriz da GFFFT satisfaz uma das seguintes condições:*

- ▷ O vetor \mathbf{x} possui simetria par, ou seja, $\mathbf{J} \cdot \mathbf{x} = \mathbf{x}$, e o seu autovalor correspondente é j ou $-j$.
- ▷ O vetor \mathbf{x} possui simetria ímpar, ou seja, $\mathbf{J} \cdot \mathbf{x} = -\mathbf{x}$, e o seu autovalor correspondente é 1 ou -1 .

Demonstração: Como \mathbf{x} é um autovetor de $\mathbf{F}\mathbf{F}_{N,G}$, a igualdade $\mathbf{F}\mathbf{F}_{N,G} \cdot \mathbf{x} = \lambda \cdot \mathbf{x}$ é válida. Multiplicando ambos os lados da última expressão por $\mathbf{F}\mathbf{F}_{N,G}$ e usando o fato de que $(\mathbf{F}\mathbf{F}_{N,G})^2 = -\mathbf{J}$, obtém-se

$$-\mathbf{J} \cdot \mathbf{x} = \lambda^2 \cdot \mathbf{x}. \quad (4.13)$$

Caso $\lambda = \pm 1$, a Equação (4.13) reduz-se a $-\mathbf{J} \cdot \mathbf{x} = \mathbf{x}$, o que significa que o vetor \mathbf{x} possui simetria ímpar; caso $\lambda = \pm j$, a Equação (4.13) reduz-se a $\mathbf{J} \cdot \mathbf{x} = \mathbf{x}$, o que significa que o vetor \mathbf{x} possui simetria par. ■

Proposição 4.8 *Os autovetores da matriz de transformação da GFFFT são construídos de acordo com as relações enunciadas a seguir. Se \mathbf{x} e \mathbf{X} são, respectivamente, um vetor de comprimento N com componentes em $GF(p)$ e sua GFFFT, então:*

▷ o vetor com simetria par $\mathbf{x}^G = \mathcal{E}\{x_i\} \mp j \cdot \mathcal{E}\{X_i\}$ é um autovetor da matriz $\mathbf{FF}_{N,G}$ associado ao autovalor $\lambda = \pm j$.

▷ o vetor com simetria ímpar $\mathbf{x}^G = \mathcal{O}\{x_i\} \pm \mathcal{O}\{X_i\}$ é um autovetor da matriz $\mathbf{FF}_{N,G}$ associado ao autovalor $\lambda = \pm 1$.

Demonstração: Utilizando as Propriedades 4.1 e 4.2, e empregando a notação anteriormente introduzida, pode-se expressar a relação entre $\mathcal{E}\{x_i\} \mp j \cdot \mathcal{E}\{X_i\}$ e sua GFFFT da seguinte forma:

$$\mathcal{E}\{x_i\} \mp j \cdot \mathcal{E}\{X_i\} \xleftrightarrow{G} \mathcal{E}\{X_k\} \pm j \cdot \mathcal{E}\{x_{-k-1}\}.$$

Como $\mathcal{E}\{x_{-k-1}\} = \mathcal{E}\{x_k\}$, a última relação pode ser reescrita como

$$\mathcal{E}\{x_i\} \mp j \cdot \mathcal{E}\{X_i\} \xleftrightarrow{G} \pm j \cdot (\mathcal{E}\{x_k\} \mp j \cdot \mathcal{E}\{X_k\}),$$

de onde segue o resultado. De forma análoga, tem-se

$$\mathcal{O}\{x_i\} \pm \mathcal{O}\{X_i\} \xleftrightarrow{G} \mathcal{O}\{X_k\} \mp \mathcal{O}\{x_{-k-1}\}.$$

Como $\mathcal{O}\{x_{-k-1}\} = -\mathcal{O}\{x_k\}$, a última relação pode ser reescrita como

$$\mathcal{O}\{x_i\} \pm \mathcal{O}\{X_i\} \xleftrightarrow{G} \pm 1 \cdot (\mathcal{O}\{x_k\} \pm \mathcal{O}\{X_k\}),$$

de onde segue o resultado. ■

4.2.2 Autovalores e autovetores das FFTT do tipo 4e

Com base nas Proposições 4.6 e 4.7, é possível apresentar as seguintes proposições relacionadas à autoestrutura da FFCT-4e e à da FFST-4e.

Proposição 4.9 *os autovetores da FFCT-4e e os da FFST-4e são construídos a partir dos autovetores da GFFFT de acordo com as relações enunciadas a seguir.*

▷ Se $\mathbf{x} = [x_0, \dots, x_{N-1}, -x_{N-1}, \dots, -x_0]$ for um autovetor ímpar da matriz $\mathbf{FF}_{2N,G}$, ou seja, $\mathbf{FF}_{2N,G} \cdot \hat{\mathbf{x}}^T = \lambda \cdot \hat{\mathbf{x}}^T$ ($\lambda = 1, -1$), então

$$\hat{\mathbf{x}} = [x_0, \dots, x_{N-1}] \tag{4.14}$$

será um autovetor da matriz $\mathbf{FC}_{N,4e}$, ou seja, $\mathbf{FC}_{N,4e} \cdot \hat{\mathbf{x}}^T = \lambda \cdot \hat{\mathbf{x}}^T$ ($\lambda = 1, -1$).

▷ Se $\mathbf{x} = [x_0, \dots, x_{N-1}, x_{N-1}, \dots, x_0]$ for um autovetor par da matriz $\mathbf{FF}_{2N,G}$, ou seja, $\mathbf{FF}_{2N,G} \cdot \mathbf{x}^T = \lambda \cdot \mathbf{x}^T$ ($\lambda = j, -j$), então

$$\tilde{\mathbf{x}} = [x_0, \dots, x_{N-1}] \quad (4.15)$$

será um autovetor da matriz $\mathbf{FS}_{N,4e}$, ou seja, $\mathbf{FS}_{N,4e} \cdot \tilde{\mathbf{x}}^T = \lambda \cdot \tilde{\mathbf{x}}^T$ ($\lambda = j, -j$).

Demonstração: Vide Apêndice B. ■

Proposição 4.10 Os únicos autovalores das matrizes de transformação da FFCT-4e e da FFST-4e são 1 e -1 . Suas multiplicidades são apresentadas nas Tabelas 4.6 e 4.7, respectivamente.

Tabela 4.6: Multiplicidades dos autovalores da matriz de transformação da transformada do co-seno de corpo finito do tipo 4e com dimensões $N \times N$.

N	Mult. de 1	Mult. de -1
ímpar	$\frac{N+1}{2}$	$\frac{N-1}{2}$
par	$\frac{N}{2}$	$\frac{N}{2}$

Tabela 4.7: Multiplicidades dos autovalores da matriz de transformação da transformada do seno de corpo finito do tipo 4e com dimensões $N \times N$.

N	Mult. de 1	Mult. de -1
ímpar	$\frac{N+1}{2}$	$\frac{N-1}{2}$
par	$\frac{N}{2}$	$\frac{N}{2}$

Demonstração: A partir da Proposição 4.9, sabe-se que os N autovalores e autovetores da matriz da FFCT-4e podem ser obtidos a partir dos N autovetores com simetria par da matriz $\mathbf{FF}_{2N,G}$. Se $N = 2m$, a matriz da GFFFT possui m autovetores com simetria ímpar relacionados ao autovalor 1 e m autovetores com simetria ímpar relacionados ao autovalor -1 . Portanto, a matriz da FFCT-4e possui m autovetores relacionados ao autovalor 1 e m autovetores relacionados ao autovalor -1 ; se $N = 2m + 1$, a matriz da GFFFT possui $m + 1$ autovetores com simetria ímpar relacionados ao autovalor 1 e m autovetores com simetria ímpar relacionados ao autovalor -1 . Nesse caso, a matriz da FFCT-4e possui $m + 1$ autovetores relacionados ao autovalor 1 e m autovetores relacionados ao autovalor -1 . O item da Proposição 4.10 referente a matriz da FFST-4e é demonstrado de forma semelhante. ■

4.3 Auto-estrutura das FFTT dos tipos $2e$ e $3e$

Nesta seção, discutem-se aspectos relacionados às auto-estruturas das FFTT dos tipos $2e$ e $3e$. Uma vez que as matrizes de transformação de tais transformadas não são simétricas, isto é, não produzem involuções, não é possível analisar suas auto-estruturas utilizando argumentos similares aos que foram aplicados às FFTT dos tipos $1e$ e $4e$. Além disso, devido às semelhanças entre as estruturas matriciais das FFTT e das DTT, as quais têm sido evidenciadas ao longo deste capítulo, é importante observar que o estudo das auto-estruturas das DTT dos tipos $2e$ e $3e$ contém pontos pouco esclarecidos, sendo restrito a algumas conjecturas baseadas em simulações numéricas [71]. Essa abordagem é, de certa forma, empregada no desenvolvimento a ser apresentado.

O procedimento convencional para obter os autovalores de uma matriz consiste em avaliar as raízes do respectivo polinômio característico. Uma vez que as matrizes das FFTT são ortogonais, o seguinte teorema relacionado aos seus polinômios característicos é válido.

Teorema 4.1 *O polinômio característico de uma matriz ortogonal com elementos sobre o conjunto dos números inteiros módulo p é um polinômio recíproco.*

Demonstração: No que segue, todas as equações são tomadas módulo p . Seja M uma matriz ortogonal com dimensões $N \times N$ e $p(\lambda) = \det(M - \lambda I)$ o seu polinômio característico. Deseja-se demonstrar que $p(\lambda) = \pm \lambda^N p(1/\lambda)$. Uma vez que $M^{-1} = M^T$, tem-se $M - \lambda I = -\lambda M(M^T - I/\lambda)$. Tomando o determinante em ambos os lados da última equação e usando os fatos $\det(M) = \det(M^T) = \pm 1$ and $\det(\lambda M) = \lambda^N \det(M)$, tem-se

$$\det(M - \lambda I) = \pm \lambda^N \det\left(M - \frac{1}{\lambda} I\right).$$

■

Se $p(\lambda) = \lambda^N p(1/\lambda)$, $p(\lambda)$ é classificado como polinômio palindrômico; se $p(\lambda) = -\lambda^N p(1/\lambda)$, o mesmo é classificado como polinômio anti-palindrômico. O cálculo das raízes de tais polinômios é realizado de forma simplificada através de um método de substituição de variáveis que reduz o seu grau pela metade [77]. Portanto, é possível utilizar fórmulas fechadas para avaliar as raízes de polinômios palindrômicos com graus até 10. Nesse caso extremo, após excluir as raízes $\pm 1 \pmod{p}$, o grau é reduzido para 4. Se $p(\lambda)$ possuir grau maior que 10, técnicas de fatoração são empregadas na obtenção das raízes [78].

Assim, as raízes do polinômio característico da matriz $\mathbf{FC}_{N,2e}$ para diferentes valores de

N pôde ser calculado. A partir dos resultados obtidos, conjectura-se que, para qualquer comprimento N , todos os autovalores da matriz de transformação da FFCT- $2e$ são distintos. Para as transformadas FFCT- $3e$, FFST- $2e$ e FFST- $3e$, esse fato também é válido. Além disso, embora as matrizes consideradas possuam todos os elementos pertencentes a $\text{GF}(p)$, é importante observar que seus autovalores podem estar localizados em corpos de extensão de ordens mais elevadas. Uma análise probabilística acerca desse fato é encontrada em [79].

O período das matrizes $\mathbf{FC}_{N,2e}$, $\mathbf{FC}_{N,3e}$, $\mathbf{FS}_{N,2e}$, $\mathbf{FS}_{N,3e}$ pode ser investigado escrevendo-as na forma diagonal. Analisar esse aspecto torna-se interessante porque, diferentemente das FFTT dos tipos $1e$ e $4e$, as transformadas consideradas nesta seção não são involuções. Como exemplo, considera-se novamente a matriz de transformação da FFCT- $2e$, a qual é escrita como $\mathbf{FC}_{N,2e} = \mathbf{U}\Lambda\mathbf{U}^*$ (\mathbf{U} é uma matriz unitária cujas colunas são autovetores de $\mathbf{FC}_{N,2e}$ e \mathbf{U}^* é sua conjugada transposta; Λ é uma matriz diagonal cujos elementos são autovalores de $\mathbf{FC}_{N,2e}$). Uma vez que $\mathbf{U}^*\mathbf{U} = \mathbf{I}$, potências de $\mathbf{FC}_{N,2e}$ podem ser obtidas a partir de potências de Λ , as quais são computadas tomando a respectiva potência de cada elemento em sua diagonal principal. Daí, a relação $(\mathbf{FC}_{N,2e})^r = \mathbf{U}\Lambda^r\mathbf{U}^*$ é satisfeita (r é um número inteiro e positivo que corresponde ao período de $\mathbf{FC}_{N,2e}$) e o menor valor de r tal que $(\mathbf{FC}_{N,2e})^r = \mathbf{I}$ também implica $\Lambda^r = \mathbf{I}$. A partir dessa última condição, conclui-se que r é o mínimo múltiplo comum entre as ordens multiplicativas dos autovalores de $\mathbf{FC}_{N,2e}$. Para as transformadas trigonométricas dos tipos $2e$ and $3e$ sobre os números reais, tem-se conjecturado que a condição estabelecida nunca é satisfeita [71]. Por definição, considera-se que essas transformadas possuem períodos iguais a 0. Naturalmente, no caso das FFTT, r é sempre finito e diferente de 0.

Diante do conteúdo exposto, pode-se afirmar que a obtenção dos autovalores das matrizes de transformação das FFTT ds tipos $2e$ e $3e$ requer, inevitavelmente, o cálculo das raízes dos respectivos polinômios característicos. Daí, os autovetores relacionados podem ser construídos.

4.4 Uma Aplicação: Formatação de Distribuições de Probabilidade sobre os Inteiros

Em [79], a transformada de Karhunen-Loève sobre corpos finitos (I_2I -KLT, integer-to-integer Karhunen-Loève transform) foi apresentada como uma ferramenta capaz de converter a distribuição de probabilidade de determinada fonte de informação numa distribuição uni-

forme. Isso permitiria introduzir, em um sistema de comunicação digital, um bloco de interface entre o receptor, projetado segundo uma distribuição uniforme, e a informação que, muitas vezes, comporta-se segundo uma curva normal. No entanto, assim como a KLT sobre os reais, a I_2I -KLT não é prática, pois sua matriz de transformação depende dos dados que se deseja processar [79]. Esse fato motivou o estudo da propriedade de formatação de distribuições de probabilidade que outras transformadas sobre corpos finitos teriam, particularmente, a NTT e as FFTT.

A descrição dos efeitos que uma transformada sobre corpos finitos tem sobre uma distribuição de probabilidade pode ser feita considerando alguns aspectos da aritmética modular. Uma ilustração simples disso consiste em associar cada símbolo de uma fonte a um número inteiro de 0 a $q - 1$, $q \leq p$, e multiplicá-lo, módulo p , por uma constante K . O produto modular desloca as freqüências de ocorrência dos símbolos. Considerando, por exemplo, um mapeamento em que $p = q = 13$ e $K = 5$ e imaginando que a fonte possua uma distribuição similar à normal, os números 6 e 7 devem possuir altas probabilidades de ocorrência em relação aos demais. Multiplicando 6 e 7 por 5 (módulo 13), obtém-se, respectivamente, 4 e 9. As freqüências de ocorrência permanecem as mesmas, mas, agora, estão associadas a outros símbolos, de maneira que o formato da distribuição original é descaracterizado.

Quando o produto modular direto entre cada símbolo e uma constante é substituído por uma combinação linear que envolve um bloco de símbolos, o processo torna-se mais complexo. No entanto, o aspecto observado na situação apresentada como exemplo persiste. Ainda que seja considerável a diferença entre as freqüências de emissão dos símbolos de uma fonte (distribuição não-uniforme), a tendência é que a aplicação de uma transformada de corpo finito produza blocos transformados compostos por símbolos uniformemente distribuídos.

Para avaliar o efeito de uma transformada em $GF(p)$ sobre o comportamento estatístico de uma fonte, geram-se amostras de números inteiros, com valores de 0 a $p - 1$, com distribuições de probabilidade conhecidas. Neste trabalho, são apresentados resultados da aplicação das transformadas a fontes que, originalmente, atendem a uma distribuição aproximadamente normal e a uma distribuição binomial. No caso da normal, que é uma função inerentemente contínua, realizam-se procedimentos de escalonamento e aproximação, para que se tenha uma amostra com as características mencionadas. A distribuição binomial, sendo discreta, não requer tal manipulação.

A fim de ser processada, a amostra é segmentada em blocos de tamanhos iguais ao da transformada que se vai aplicar. Os resultados desse procedimento são ilustrados através de histogramas da amostra antes e após a transformação. Um teste de aderência é realizado [80], com o objetivo de quantificar a proximidade entre a distribuição obtida e a uniforme.

4.4.1 Formatação via FFCT

Exemplo 4.1 Na primeira situação considerada, gerou-se uma amostra com 2^{14} números de 0 a 126 distribuídos de modo aproximadamente normal. Utilizou-se a FFCT-2e de comprimento $N = 8$ sobre GF(127). Os histogramas da amostra antes e após a transformação são apresentados na Figura 4.1. Usando o teste chi-quadrado de Pearson, que permite verificar a aderência de dados a uma distribuição teórica, assumiu-se um valor de probabilidade $P \leq 0,05$ como justificativa para rejeitar a hipótese nula de que os dados não aderem à distribuição proposta [80]. Verificou-se que a amostra resultante adere a uma distribuição uniforme (vide Tabela 4.8).

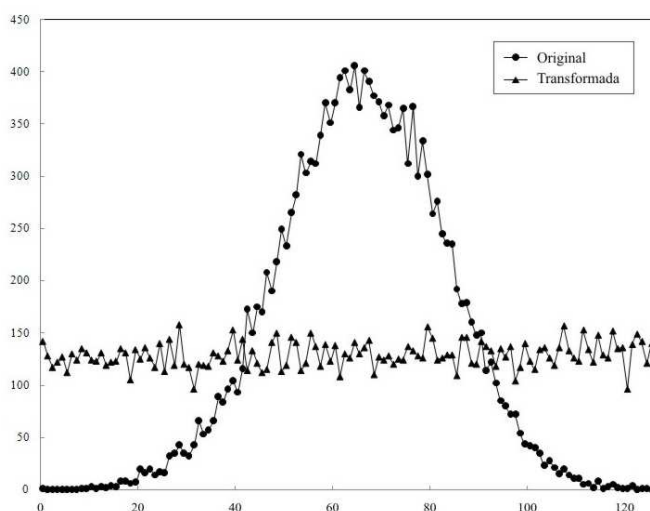


Figura 4.1: Histogramas de uma amostra com 2^{14} números antes e após a aplicação da FFCT-2e de comprimento $N = 8$ sobre GF(127).

Exemplo 4.2 Numa segunda simulação, também foi gerada uma amostra com 2^{14} números de 0 a 126, no entanto, de acordo com uma distribuição binomial. Novamente, utilizou-se a FFCT-2e de comprimento $N = 8$ sobre GF(127). Os histogramas da amostra antes e depois de duas transformações iterativas são apresentados na Figura 4.2. O cálculo repetido da transformada foi necessário, pois o fato de a distribuição binomial ser mais esparsa que a considerada no exemplo anterior não favoreceu a produção de uma distribuição uniforme na primeira iteração. Após este procedimento, a amostra obtida

foi testada, tendo sido comprovada a sua aderência à distribuição uniforme (vide Tabela 4.8).

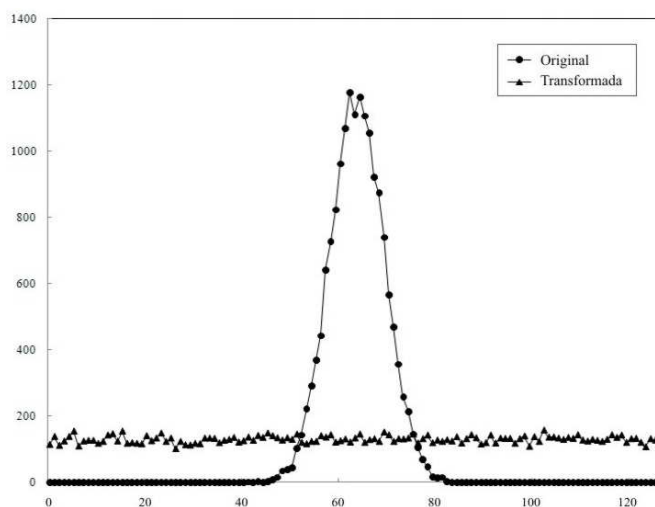


Figura 4.2: Histogramas de uma amostra com 2^{14} números antes e após a aplicação da FFCT-2e de comprimento $N = 8$ sobre GF(127). Neste caso, a transformada foi aplicada duas vezes.

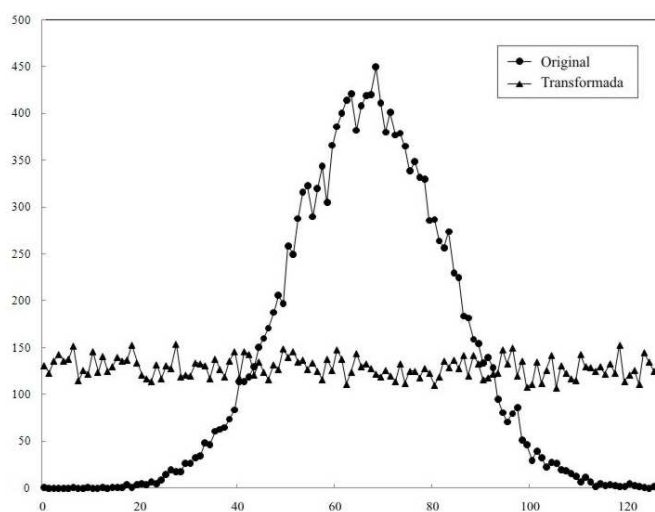


Figura 4.3: Histogramas de uma amostra com 2^{14} números antes e após a aplicação da FFCT-4e de comprimento $N = 8$ sobre GF(127).

Exemplo 4.3 Nesta situação, realizou-se um procedimento análogo ao descrito nos exemplos anteriores. No entanto, foi utilizada a FFCT-4e. Os resultados obtidos são apresentados na Figura 4.3. Após a transformação, a amostra foi testada, fornecendo forte aderência à distribuição uniforme (vide Tabela 4.8).

Nos Exemplos 4.1, 4.2 e 4.3, o aspecto relacionado ao conteúdo visto ao longo deste capítulo diz respeito aos períodos das matrizes de transformação das FFCT utilizadas. Como

esses períodos são finitos, em algum momento, o resultado das iterativas transformações é igual ao vetor original. No caso da transformada FFCT-4e que, conforme se demonstrou, possui período igual a 2, se uma amostra for processada duas vezes, obter-se-á como resultado a própria amostra. Assim, essa transformada não pode ser usada para processar distribuições que requeiram mais de uma iteração para alcançar o formato uniforme. Para a FFCT-2e, sabe-se que são possíveis períodos maiores que 2. Assim, como são necessárias diversas iterações para que se retorne à amostra original, há possibilidade de se aplicar a transformada mencionada à formatação de distribuições que se aproximem mais lentamente da distribuição uniforme. Também foram realizadas simulações em que se utilizou a FFFT. Resultados semelhantes aos dos exemplos baseados nas FFCT foram alcançados [38].

Tabela 4.8: Resultados do teste chi-quadrado de Pearson para verificar a aderência de uma amostra à distribuição uniforme (após a aplicação da transformada sobre corpo finito indicada).

Transformada	Distribuição original	$P (\leq 0,05, \text{rejeição da aderência})$
FFCT-2e	Normal	0,4482
	Binomial	0,4577
FFCT-4e	Normal	0,7792

4.4.2 Aplicações em Criptossistemas

Conforme discutido ao longo desta seção, o uso de uma transformada de corpo finito conduz à uniformização de uma distribuição de probabilidade. No campo da Criptografia, essa propriedade possui potencial aplicação, uma vez que um conhecimento prévio do comportamento estatístico de uma fonte de informação facilita a criptoanálise de textos cifrados a partir da mesma. Existem algumas ferramentas que objetivam modificar as probabilidades de ocorrência dos símbolos de uma fonte. Um exemplo é a substituição homofônica [81], [82], que tem sido parte integrante de criptossistemas em que a modificação mencionada é empregada.

A substituição homofônica visa reduzir a redundância da mensagem cifrada, aumentando, portanto, a distância de unicidade da cifra [81]. Assim, uma seqüência de símbolos da fonte com uma distribuição de freqüência arbitrária é transformada em uma seqüência unicamente decodificável de símbolos, tendo, todos, a mesma freqüência. De modo mais específico, essa técnica associa a cada símbolo s da fonte, $s \in A$, um conjunto H_s de novos símbolos, deno-

minados homófonos, pertencentes a um alfabeto maior que A , dentre os quais se escolhe um substituto para s . A cardinalidade de H_s é proporcional à frequência relativa de s no alfabeto original A e um homófono é escolhido aleatoriamente de modo a gerar uma “nova” fonte uniformemente distribuída, cujos símbolos são então cifrados.

O método de formatação de distribuições introduzida nesta seção pode ser visto como uma alternativa para a substituição clássica descrita acima. Assim, por exemplo, ao se transformar, nos moldes dos exemplos apresentados, blocos de símbolos de uma fonte binária com probabilidades p_1 e p_2 , produz-se uma fonte com distribuição aproximadamente uniforme ($p_1 = p_2 = 1/2$). Para se obter o efeito de uma substituição homofônica, é necessário não apenas uniformizar a distribuição da fonte, mas também estabelecer uma independência estatística entre os símbolos da mensagem e os produzidos pela substituição. Isso requer a expansão do alfabeto da mensagem e a introdução de um mecanismo aleatório, que não se encontra presente em transformações lineares como as consideradas aqui. Dois procedimentos que objetivam atender essa exigência são, então, sugeridos:

- ▷ Usar diferentes versões de transformadas para processar diferentes blocos de mensagem. A seqüência das transformadas a ser usada seria aleatória e funcionaria como uma chave secreta. A mesma poderia ser obtida não só dos diversos tipos possíveis de FFCT e FFST, por exemplo, mas também pelo uso de diferentes elementos unimodulares empregados na definição da transformada. Além disso, ao se usar $p > q$, a expansão do alfabeto original é obtida.
- ▷ Acrescentar a cada bloco de mensagem um bloco aleatório de comprimento fixo, o que significa uma expansão do texto claro. A transformação é então aplicada ao bloco expandido. Na decodificação, após a transformação inversa, o bloco aleatório, cuja posição é previamente conhecida, é simplesmente desconsiderado. Como anteriormente, a expansão do alfabeto original é obtida ao se usar $p > q$. Uma combinação dos dois procedimentos também é possível.

CAPÍTULO 5

SEPARAÇÃO CEGA DE SEQÜÊNCIAS BASEADA NA AUTO-ESTRUTURA DAS FFTT

NA TEORIA das Comunicações, o problema de separar informações que vêm de fontes distintas, após as mesmas serem “misturadas” segundo condições específicas, tem sido extensivamente estudado [83], [84]. Dentre diferentes técnicas para recuperar os dados originalmente transmitidos por cada usuário, um caso particularmente interessante é a separação sem o conhecimento explícito da informação relacionada a cada fonte (ou usuário), ou seja, a separação cega. Quando diferentes usuários compartilham a mesma banda de frequência ao mesmo tempo, técnicas bem estabelecidas realizam tal separação usando propriedades estatísticas de seqüências e códigos que podem ser entendidos como “portadoras digitais”.

Neste capítulo, usando como referência o cenário descrito acima, mostra-se como as auto-estruturas das FFTT podem ser usadas na separação cega de seqüências. Considera-se um canal somador sobre corpos finitos livre de erro o qual é compartilhado por diferentes usuários de modo síncrono [85], [86]. O procedimento consiste em associar um autovalor e , portanto, um conjunto de autovetores de determinada FFTT a cada usuário. A informação a ser enviada por um usuário é mapeada sobre autovetores. Uma vez que autovetores relacionados a diferentes autovalores são ortogonais, após serem somados pelo canal, os mesmos podem ser recuperados a partir da solução de um sistema de equações lineares. Esse esquema é ilustrado nas seções subseqüentes.

5.1 Esquema com 2 Usuários

Com o propósito de apresentar um esquema com 2 usuários, considera-se uma FFCT-1e de comprimento $N \geq 2$, embora qualquer outra FFTT cuja matriz de transformação tenha pelo menos 2 autovalores distintos possa ser usada. Conforme demonstrado no capítulo anterior, a matriz da FFCT-1e possui $\lambda_1 = 1$ e $\lambda_2 = -1$ como autovalores. Associa-se a esses autovalores e aos usuários 1 e 2, respectivamente, os autovetores $\mathbf{x}_1 = (x_{1,i})$ e $\mathbf{x}_2 = (x_{2,i})$, $i = 0, 1$, os quais são construídos de acordo com a Proposição 4.4.

A partir do vetor $\mathbf{y} = (y_i)$, que, devido ao efeito aditivo do canal considerado, é dado por

$$y_i = x_{1,i} + x_{2,i}, \quad (5.1)$$

em que “+” denota a adição componente por componente, é possível recuperar as seqüências dos usuários. Calculando $\mathbf{Y} = (Y_k) = \mathbf{FC}_{N,1e} \times \mathbf{y}^T$, tem-se

$$Y_k = \lambda x_{1,i} + \lambda_2 x_{2,i} = x_{1,i} - x_{2,i}. \quad (5.2)$$

Resolvendo o sistema formado pelas Equações (5.1) e (5.2), as seqüências dos usuários são recuperadas a partir das expressões $x_{1,i} = (y_i + Y_i)/2$ e $x_{2,i} = (y_i - Y_i)/2$, que, naturalmente, são avaliadas módulo p . Um diagrama de blocos ilustrando a recuperação das seqüências \mathbf{x}_1 e \mathbf{x}_2 pode ser visto na Figura 5.1.

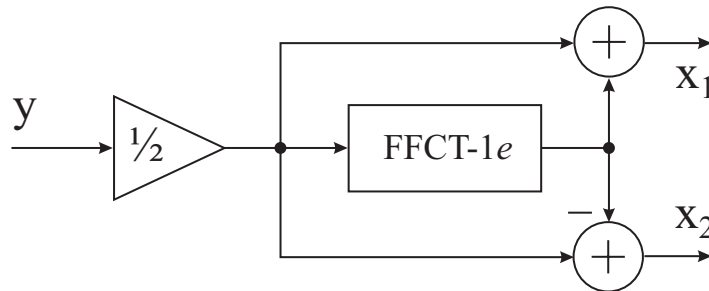


Figura 5.1: Recuperação das seqüências num esquema com 2 usuários.

Um importante aspecto a ser observado no procedimento descrito diz respeito à complexidade aritmética envolvida no mesmo. A partir do diagrama de blocos apresentado, verifica-se que o número de multiplicações e o de adições necessárias para separar os vetores \mathbf{x}_1 e \mathbf{x}_2 de comprimento N são, respectivamente, dados por

$$M_2(N) = N + M_{FC_{1e}}(N) \quad (5.3)$$

e

$$A_2(N) = 2N + A_{FC_{1e}}(N), \quad (5.4)$$

Nas duas últimas equações, o subscrito “2” indica a associação com o esquema com 2 usuários; $M_{FC_{1e}}(N)$ e $A_{FC_{1e}}(N)$ respectivamente, denotam o número de multiplicações e o de adições para calcular uma FFCT-1e de comprimento N .

5.2 Esquema com 4 Usuários: Um Estudo de Caso

Para implementar esquemas que permitam o acesso simultâneo de até 4 usuários, é necessário considerar FFTT que possuam, pelo menos, 4 autovalores distintos. Assim, de acordo com o conteúdo apresentado no capítulo anterior, um esquema com 4 usuários requer o emprego de transformadas dos tipos 2e ou 3e. Além disso, escolhendo-se uma transformada sobre $GF(p)$, os autovalores usados para implementar tal esquema também devem estar localizados em $GF(p)$, para que se evitem cálculos em corpos de extensão.

Como exemplo, considera-se a FFCT-2e com comprimento 4 sobre $GF(127)$. A matriz de transformação construída usando o elemento unimodular $\zeta = 119 + j119$ é

$$\widetilde{\mathbf{FC}}_{2e} = \begin{bmatrix} 64 & 41 & 63 & 65 \\ 64 & 65 & 64 & 86 \\ 64 & 62 & 64 & 41 \\ 64 & 86 & 63 & 62 \end{bmatrix},$$

a qual possui $\lambda_1 = 1$, $\lambda_2 = 20$, $\lambda_3 = 108$ e $\lambda_4 = 126$ como autovalores. As seqüências (ou autovetores) dos usuários são, então, denotadas por $\mathbf{x}_1 = (x_{1,i})$, $\mathbf{x}_2 = (x_{2,i})$, $\mathbf{x}_3 = (x_{3,i})$ e $\mathbf{x}_4 = (x_{4,i})$, $i = 0, 1, 2, 3$, respectivamente, e construídas de acordo com as seguintes expressões:

$$\lambda_1 : x_{1,0} = 105x_{1,3}, \quad x_{1,1} = 105x_{1,3}, \quad x_{1,2} = 126x_{1,3}, \quad x_{1,3} = x_{1,3},$$

$$\lambda_2 : x_{2,0} = 60x_{1,3}, \quad x_{2,1} = 104x_{1,3}, \quad x_{2,2} = 76x_{1,3}, \quad x_{2,3} = x_{1,3},$$

$$\lambda_3 : x_{3,0} = 91x_{1,3}, \quad x_{3,1} = 104x_{1,3}, \quad x_{3,2} = 29x_{1,3}, \quad x_{3,3} = x_{1,3},$$

$$\lambda_4 : x_{4,0} = 94x_{1,3}, \quad x_{4,1} = 36x_{1,3}, \quad x_{4,2} = 62x_{1,3}, \quad x_{4,3} = x_{1,3}.$$

Nas Tabelas 5.1, 5.2 e 5.3, que se encontram no final do capítulo, são apresentadas todas as possíveis seqüências de usuário para este exemplo.

Analogamente ao esquema com 2 usuários, o canal somador produz o vetor \mathbf{y} , cujas componentes são dadas por $y_i = x_{1,i} + x_{2,i} + x_{3,i} + x_{4,i}$, $i = 0, 1, 2, 3$. Calculando sucessivas transformadas de \mathbf{y} , tem-se $\mathbf{Y}' = (Y'_k) = \mathbf{FC}_{N,2e} \times \mathbf{y}^T$, $\mathbf{Y}'' = (Y''_k) = (\mathbf{FC}_{N,2e})^2 \times \mathbf{y}^T$ e $\mathbf{Y}''' = (Y'''_k) = (\mathbf{FC}_{N,2e})^3 \times \mathbf{y}^T$. Daí, o seguinte sistema de equações lineares é obtido:

$$\begin{cases} x_{1,i} + x_{2,i} + x_{3,i} + x_{4,i} & = y_i \\ \lambda_1 x_{1,i} + \lambda_2 x_{2,i} + \lambda_3 x_{3,i} + \lambda_4 x_{4,i} & = Y'_i \\ \lambda_1^2 x_{1,i} + \lambda_2^2 x_{2,i} + \lambda_3^2 x_{3,i} + \lambda_4^2 x_{4,i} & = Y''_i \\ \lambda_1^3 x_{1,i} + \lambda_2^3 x_{2,i} + \lambda_3^3 x_{3,i} + \lambda_4^3 x_{4,i} & = Y'''_i. \end{cases}$$

Substituindo os valores de λ_i , $i = 1, \dots, 4$, no sistema acima, tem-se

$$\begin{cases} x_{1,i} + x_{2,i} + x_{3,i} + x_{4,i} & = y_i \\ x_{1,i} + 20 x_{2,i} + 108 x_{3,i} + 126 x_{4,i} & = Y'_i \\ x_{1,i} + 19 x_{2,i} + 107 x_{3,i} + x_{4,i} & = Y''_i \\ x_{1,i} + 126 x_{2,i} + 126 x_{3,i} + 126 x_{4,i} & = Y'''_i, \end{cases}$$

cujas soluções são

$$\begin{aligned} x_{1,i} &= 64 y_i + 64 Y'''_i, \\ x_{2,i} &= 49 y_i + 36 Y'_i + 78 Y''_i + 91 Y'''_i, \\ x_{3,i} &= 36 y_i + 49 Y'_i + 91 Y''_i + 78 Y'''_i, \\ x_{4,i} &= 106 y_i + 42 Y'_i + 85 Y''_i + 21 Y'''_i. \end{aligned}$$

Portanto, a partir de \mathbf{y} , obtém-se \mathbf{Y}' , \mathbf{Y}'' and \mathbf{Y}''' e se usa as equações acima para recuperar cada seqüência de usuário. Um diagrama de blocos ilustrando esse procedimento é apresentado na Figura 5.2.

Em sistemas com 4 usuários, como o do exemplo desenvolvido, o número de multiplicações e o de adições necessárias para separar os vetores \mathbf{x}_1 , \mathbf{x}_2 , \mathbf{x}_3 e \mathbf{x}_4 de comprimento N são, respectivamente, dados por

$$M_4(N) = 7N + 3M_{FC_{2e}}(N) \quad (5.5)$$

e

$$A_4(N) = 8N + 3A_{FC_{2e}}(N). \quad (5.6)$$

Nas equações acima, $M_{FC_{2e}}(N)$ e $A_{FC_{2e}}(N)$ denotam, respectivamente, o número de multiplicações e o de adições para calcular uma FFCT-2e de comprimento N .

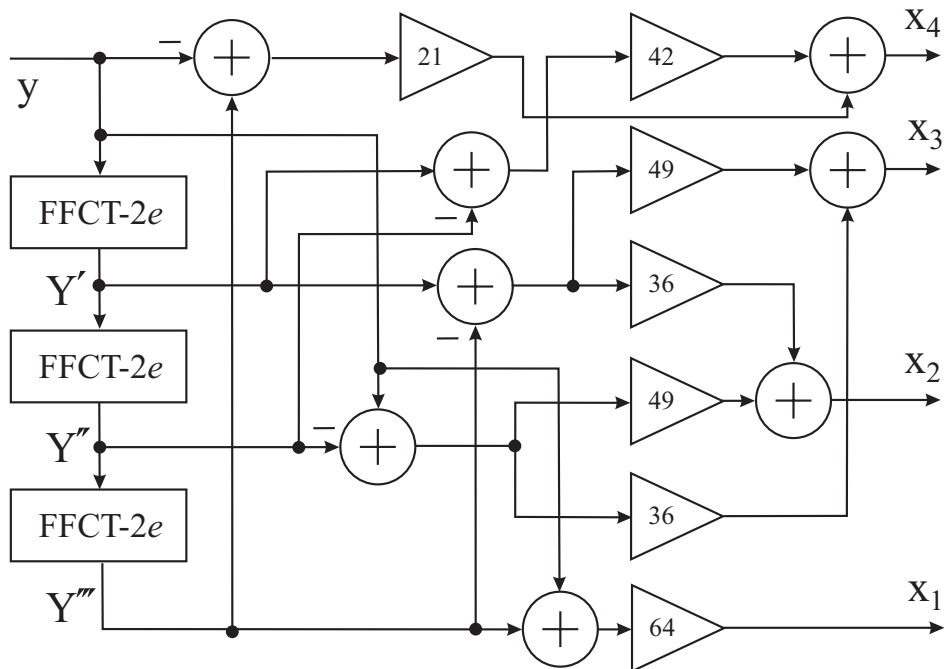


Figura 5.2: Recuperação de seqüências num esquema com 4 usuários, baseado na auto-estrutura da FFCT-2e sobre GF(127) e com comprimento de bloco $N = 4$.

Seguindo os princípios apresentados, esquemas com números maiores de usuários simultâneos podem ser projetados. Na Tabela 5.4, que se encontra no final do capítulo, algumas possibilidades para implementação de tais esquemas utilizando a FFCT-2e são apresentadas. Além do número primo p e do comprimento N , mostra-se o elemento unimodular ζ que deve ser utilizado e os autovalores da respectiva matriz de transformação que estão em GF(p).

5.3 Discussão

5.3.1 Complexidade Computacional

Uma vez que se assume que os autovalores usados num esquema específico são fixos, o sistema de equações a partir do qual as seqüências de usuário são recuperadas precisa ser resolvido apenas uma vez. De fato, é necessário aplicar a solução (que já é conhecida) cada vez que se recebe um novo vetor y . Derivar uma fórmula geral para o número exato de multiplicações e o de adições envolvidas na aplicação da referida solução é inviável, uma vez que esses valores dependem de diversos parâmetros, como a quantidade de usuários do esquema, a transformada utilizada e seus autovalores. Todavia, diante dos resultados obtidos nos exemplos desenvolvidos, pode-se esperar que $\mathcal{O}(N)$ multiplicações e adições sejam necessárias.

De acordo com o conteúdo apresentado, observa-se que, além das operações aritméticas discutidas acima, a recuperação das seqüências dos usuários requer o cálculo de transformadas do vetor y . Acerca disso, é importante ressaltar que a matriz de transformação de uma FFTT específica apresenta exatamente o mesmo tipo de simetria de sua versão equivalente sobre os números reais. Portanto, é possível calcular FFCT e FFST usando algoritmos rápidos que foram desenvolvidos para as DTT, desde que os mesmos sejam baseados nas simetrias mencionadas. Isso inclui métodos análogos ao algoritmo de Cooley-Tukey de base 2, através dos quais se calculam transformadas do co-seno e do seno com comprimento N usando $\mathcal{O}(N \log N)$ operações. Valores exatos para o número de multiplicações e o de adições envolvidos nesses métodos são encontrados em [87].

Aliando a complexidade aritmética da aplicação da solução do sistema de equações lineares à do cálculo das transformadas, conclui-se que tanto a complexidade multiplicativa quanto a aditiva, envolvidas na recuperação das seqüências dos usuários, é de $\mathcal{O}(N \log N)$. Naturalmente, esse resultado é válido apenas quando N é uma potência de 2.

5.3.2 *Uma hierarquia de comunicação multi-usuário no canal somador sobre corpos finitos*

Conforme anteriormente observado, o método para separação cega de seqüências apresentado restringe o número de usuários simultâneos de um canal ao número de autovalores distintos da FFTT usada. Entretanto, de maneira similar à multiplexação por divisão no tempo (TDM, *Time Division Multiplexing*) e à multiplexação por divisão na freqüência (FDM, *Frequency Division Multiplexing*), é possível implementar um esquema hierárquico. Isso permite o acesso de um número maior de usuários ao sistema, à medida em que novos níveis são criados pela combinação de sinais que se encontram em níveis mais baixos. Como se descreve a seguir, a implementação dessa estratégia requer uma nova camada para que a informação a ser transmitida seja mapeada em autovetores. Duas possibilidades para realizar esse procedimento são descritas.

Para ilustrar a primeira das possibilidades, assume-se que FFTT com dois autovalores distintos e com comprimento $N = 2$ são utilizadas. Assim, em cada nível hierárquico, limita-se em 2 o número de usuários que podem interferir aditivamente. Para que se implemente um esquema com 2 níveis, isto é, que reúna 4 usuários, uma transformada sobre $GF(p_1)$, que é usada nos níveis mais baixos, e outra sobre $GF(p_2)$, que é usada no nível mais alto, são

necessárias. O diagrama em blocos desse esquema é apresentado na Figura 5.3. Visualizando o diagrama, torna-se mais simples compreender que a condição $p_2 \geq (p_1)^2$ deve ser respeitada. A informação a ser enviada pelo usuário 1 é mapeada em um dos p_1 autovetores que é somado a um dos outros p_1 autovetores relacionados ao usuário 2. Isso gera $(p_1)^2$ possíveis combinações. Uma vez que as mesmas combinações são geradas pelos usuários 3 e 4, para o nível hierárquico seguinte, um novo mapeamento em que cada autovalor gere pelo menos $(p_1)^2$ autovetores é requerido. Assim, a condição previamente estabelecida é justificada. O mapeamento dos vetores y_1 e y_2 em autovetores y'_1 e y'_2 da transformada sobre $GF(p_2)$ permite a nova adição para composição de y . O procedimento de recuperação das seqüências originais é uma repetição do que foi apresentado para o esquema com 2 usuários, sendo realizado em duas etapas.

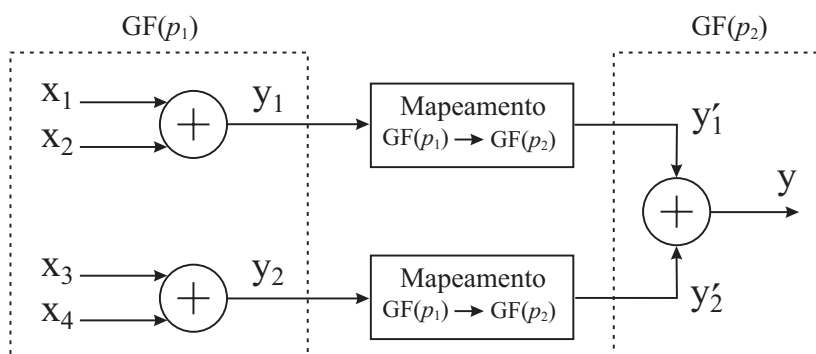


Figura 5.3: Hierarquia com 2 níveis para comunicação multi-usuário baseada na auto-estrutura de matrizes de transformação: mapeamento extra para um corpo finito primo ($p_2 \geq p_1^2$).

A segunda possibilidade utiliza uma estratégia semelhante àquela empregada na primeira. No entanto, para possibilitar a inserção de um novo nível hierárquico, em vez de se realizar um mapeamento extra para um corpo finito primo com característica maior, realiza-se um mapeamento para um corpo de extensão. Considerando que se usam os mesmos tipos de FFTT do cenário anterior, os vetores y_1 e y_2 , resultantes da adição de autovetores de uma transformada sobre $GF(p)$, são mapeados em autovetores y'_1 e y'_2 de uma transformada sobre $GF(p^2)$. Apesar de não serem o foco deste trabalho, transformadas trigonométricas sobre corpos de extensão podem ser construídas. A Figura 5.4 ilustra o esquema descrito.

Naturalmente, inúmeras variações das possibilidades discutidas podem ser feitas. Pode-se combinar diferentes tipos de transformadas, reunir seqüências obtidas pela adição de quantidades distintas de usuários, utilizar diversos tipos de mapeamentos extras na criação de novos níveis etc. Para isso, basta que se observem as condições cuja necessidade foi justificada.

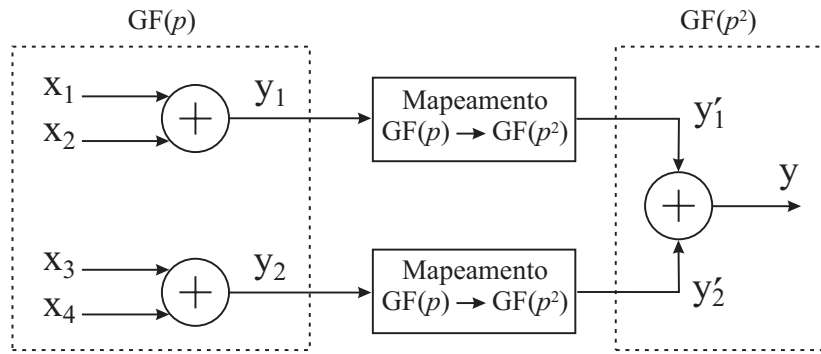


Figura 5.4: Hierarquia com 2 níveis para comunicação multi-usuário baseada na auto-estrutura de matrizes de transformação: mapeamento extra para um corpo de extensão.

5.3.3 Analogia com o DS-CDMA

Outros aspectos interessantes da comunicação multi-usuário baseada na auto-estrutura das FFTT podem ser revelados através de uma comparação com o múltiplo acesso por divisão de códigos usando o espalhamento espectral direto sobre seqüências (DS-CDMA, *direct sequence code division multiple access*) [84]. Nesse sentido, os esquemas apresentados podem ser vistos como DS-CDMA em que as seqüências de espalhamento (assinaturas dos usuários) são autovetores de uma transformada sobre corpo finito, em vez de códigos de Walsh ou seqüências pseudo-aleatórias. Diferentemente dos receptores DS-CDMA, que usam propriedades de autocorrelação e de correlação cruzada das seqüências de espalhamento, na proposta apresentada neste trabalho, o sucesso da separação de cada seqüência de usuário depende da ortogonalidade entre auto-espacos distintos.

5.3.4 Requisitos de energia e análise num canal ruidoso

Embora o objetivo deste trabalho não seja uma descrição completa de um cenário prático de comunicação multi-usuário, é relevante mencionar alguns dos seus requisitos. A energia máxima permitida no canal considerado, por exemplo, determinaria uma limitação sobre a energia de cada autovetor utilizado. Isso significa que os mesmos precisariam ser convenientemente escalonados. Além disso, uma definição clara do processo de espalhamento, o qual depende da natureza da informação a ser mapeada nos autovetores, precisaria ser realizada.

Finalmente, uma vez que, para a maioria dos canais práticos, não se pode assumir ausência de ruído, uma análise de erro seria necessária. Para realizar tal análise, pode-se tratar os autovetores (ou o vetor resultante da soma entre autovetores) como palavras de um código corretor de erros. Essa interpretação sugere estudar a susceptibilidade a erro dos esquemas apre-

sentados como um problema de decodificação de códigos de bloco lineares. Nesse contexto, a restrição a ser observada é que, para permitir a correção de erros, na transmissão da informação, não se pode utilizar todos os autovetores associados a uma matriz de transformação específica.

Para que a questão levantada fique mais clara, considera-se novamente o exemplo desenvolvido para o esquema com 4 usuários. Caso se utilizem os 4 autovalores e todos os autovetores associados a eles, a adição das seqüências dos usuários pode criar 127^4 diferentes vetores (ou palavras-código) \mathbf{y} , que, naturalmente, correspondem a todos os vetores de comprimento 4 sobre $GF(127)$. Assumindo que o vetor \mathbf{y} seja corrompido por um ruído aditivo \mathbf{n} , que seria também um vetor de comprimento 4 sobre $GF(127)$, nenhum erro seria detectado ou corrigido, uma vez que o vetor recebido $\mathbf{r} = \mathbf{y} + \mathbf{n}$ seria, necessariamente, uma das palavras do código.

Para evitar a situação supracitada, poder-se-ia introduzir uma espécie de redundância no esquema pela não utilização de alguns dos autovalores disponíveis. Se, por exemplo, apenas dois autovalores fossem utilizados, 127^2 diferentes vetores (ou palavras-código) \mathbf{y} poderiam ser criados. Assim, se a adição de um ruído resultasse num vetor $\mathbf{r} = \mathbf{y} + \mathbf{n}$ que estivesse dentre os $127^4 - 127^2$ vetores de comprimento 4 sobre $GF(127)$ que não são palavras do código, poder-se-ia detectar e, sob determinadas condições, corrigir o erro. Nesse caso, ter-se-ia o “prejuízo” de restringir a apenas 2 usuários o acesso simultâneo ao canal. Outra possibilidade que também permitiria a detecção de erros seria não utilizar todos os autovetores disponíveis para cada autovalor. Usando essa estratégia, a quantidade máxima de usuários simultâneos seria mantida; quanto menor o número de autovetores utilizados, maior seria a capacidade de detecção de erros, porém, em compensação, mais baixa seria a taxa de transmissão de informação.

Tabela 5.1: Sequências de usuário: autovetores da matriz de transformação da FFCT-2e, $N = 4$, $p = 127$, $\zeta = 119 + j119$. Autovalores: $\lambda_1 = 1$, $\lambda_2 = 20$, $\lambda_3 = 108$ e $\lambda_4 = 126$.

$x_{1,0}$	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{2,0}$	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{3,0}$	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{4,0}$	$x_{4,1}$	$x_{4,2}$	$x_{4,3}$
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
105	105	126	1	60	104	76	1	91	104	29	1	94	36	62	1
83	83	125	2	120	81	25	2	55	81	58	2	61	72	124	2
61	61	124	3	53	58	101	3	19	58	87	3	28	108	59	3
39	39	123	4	113	35	50	4	110	35	116	4	122	17	121	4
17	17	122	5	46	12	126	5	74	12	18	5	89	53	56	5
122	122	121	6	106	116	75	6	38	116	47	6	56	89	118	6
100	100	120	7	39	93	24	7	2	93	76	7	23	125	53	7
78	78	119	8	99	70	100	8	93	70	105	8	117	34	115	8
56	56	118	9	32	47	49	9	57	47	7	9	84	70	50	9
34	34	117	10	92	24	125	10	21	24	36	10	51	106	112	10
12	12	116	11	25	1	74	11	112	1	65	11	18	15	47	11
117	117	115	12	85	105	23	12	76	105	94	12	112	51	109	12
95	95	114	13	18	82	99	13	40	82	123	13	79	87	44	13
73	73	113	14	78	59	48	14	4	59	25	14	46	123	106	14
51	51	112	15	11	36	124	15	95	36	54	15	13	32	41	15
29	29	111	16	71	13	73	16	59	13	83	16	107	68	103	16
7	7	110	17	4	117	22	17	23	117	112	17	74	104	38	17
112	112	109	18	64	94	98	18	114	94	14	18	41	13	100	18
90	90	108	19	124	71	47	19	78	71	43	19	8	49	35	19
68	68	107	20	57	48	123	20	42	48	72	20	102	85	97	20
46	46	106	21	117	25	72	21	6	25	101	21	69	121	32	21
24	24	105	22	50	2	21	22	97	2	3	22	36	30	94	22
2	2	104	23	110	106	97	23	61	106	32	23	3	66	29	23
107	107	103	24	43	83	46	24	25	83	61	24	97	102	91	24
85	85	102	25	103	60	122	25	116	60	90	25	64	11	26	25
63	63	101	26	36	37	71	26	80	37	119	26	31	47	88	26
41	41	100	27	96	14	20	27	44	14	21	27	125	83	23	27
19	19	99	28	29	118	96	28	8	118	50	28	92	119	85	28
124	124	98	29	89	95	45	29	99	95	79	29	59	28	20	29
102	102	97	30	22	72	121	30	63	72	108	30	26	64	82	30
80	80	96	31	82	49	70	31	27	49	10	31	120	100	17	31
58	58	95	32	15	26	19	32	118	26	39	32	87	9	79	32
36	36	94	33	75	3	95	33	82	3	68	33	54	45	14	33
14	14	93	34	8	107	44	34	46	107	97	34	21	81	76	34
119	119	92	35	68	84	120	35	10	84	126	35	115	117	11	35
97	97	91	36	1	61	69	36	101	61	28	36	82	26	73	36
75	75	90	37	61	38	18	37	65	38	57	37	49	62	8	37
53	53	89	38	121	15	94	38	29	15	86	38	16	98	70	38
31	31	88	39	54	119	43	39	120	119	115	39	110	7	5	39
9	9	87	40	114	96	119	40	84	96	17	40	77	43	67	40
114	114	86	41	47	73	68	41	48	73	46	41	44	79	2	41
92	92	85	42	107	50	17	42	12	50	75	42	11	115	64	42
70	70	84	43	40	27	93	43	103	27	104	43	105	24	126	43
48	48	83	44	100	4	42	44	67	4	6	44	72	60	61	44
26	26	82	45	33	108	118	45	31	108	35	45	39	96	123	45
4	4	81	46	93	85	67	46	122	85	64	46	6	5	58	46
109	109	80	47	26	62	16	47	86	62	93	47	100	41	120	47
87	87	79	48	86	39	92	48	50	39	122	48	67	77	55	48
65	65	78	49	19	16	41	49	14	16	24	49	34	113	117	49
43	43	77	50	79	120	117	50	105	120	53	50	1	22	52	50
21	21	76	51	12	97	66	51	69	97	82	51	95	58	114	51
126	126	75	52	72	74	15	52	33	74	111	52	62	94	49	52

Tabela 5.2: Seqüências de usuário: autovetores da matriz de transformação da FFCT2e, $N = 4$, $p = 127$, $\zeta = 119 + j119$. Autovalores: $\lambda_1 = 1$, $\lambda_2 = 20$, $\lambda_3 = 108$ e $\lambda_4 = 126$ (continuação da tabela anterior).

$x_{1,0}$	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{2,0}$	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{3,0}$	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{4,0}$	$x_{4,1}$	$x_{4,2}$	$x_{4,3}$
104	104	74	53	5	51	91	53	124	51	13	53	29	3	111	53
82	82	73	54	65	28	40	54	88	28	42	54	123	39	46	54
60	60	72	55	125	5	116	55	52	5	71	55	90	75	108	55
38	38	71	56	58	109	65	56	16	109	100	56	57	111	43	56
16	16	70	57	118	86	14	57	107	86	2	57	24	20	105	57
121	121	69	58	51	63	90	58	71	63	31	58	118	56	40	58
99	99	68	59	111	40	39	59	35	40	60	59	85	92	102	59
77	77	67	60	44	17	115	60	126	17	89	60	52	1	37	60
55	55	66	61	104	121	64	61	90	121	118	61	19	37	99	61
33	33	65	62	37	98	13	62	54	98	20	62	113	73	34	62
11	11	64	63	97	75	89	63	18	75	49	63	80	109	96	63
116	116	63	64	30	52	38	64	109	52	78	64	47	18	31	64
94	94	62	65	90	29	114	65	73	29	107	65	14	54	93	65
72	72	61	66	23	6	63	66	37	6	9	66	108	90	28	66
50	50	60	67	83	110	12	67	1	110	38	67	75	126	90	67
28	28	59	68	16	87	88	68	92	87	67	68	42	35	25	68
6	6	58	69	76	64	37	69	56	64	96	69	9	71	87	69
111	111	57	70	9	41	113	70	20	41	125	70	103	107	22	70
89	89	56	71	69	18	62	71	111	18	27	71	70	16	84	71
67	67	55	72	2	122	11	72	75	122	56	72	37	52	19	72
45	45	54	73	62	99	87	73	39	99	85	73	4	88	81	73
23	23	53	74	122	76	36	74	3	76	114	74	98	124	16	74
1	1	52	75	55	53	112	75	94	53	16	75	65	33	78	75
106	106	51	76	115	30	61	76	58	30	45	76	32	69	13	76
84	84	50	77	48	7	10	77	22	7	74	77	126	105	75	77
62	62	49	78	108	111	86	78	113	111	103	78	93	14	10	78
40	40	48	79	41	88	35	79	77	88	5	79	60	50	72	79
18	18	47	80	101	65	111	80	41	65	34	80	27	86	7	80
123	123	46	81	34	42	60	81	5	42	63	81	121	122	69	81
101	101	45	82	94	19	9	82	96	19	92	82	88	31	4	82
79	79	44	83	27	123	85	83	60	123	121	83	55	67	66	83
57	57	43	84	87	100	34	84	24	100	23	84	22	103	1	84
35	35	42	85	20	77	110	85	115	77	52	85	116	12	63	85
13	13	41	86	80	54	59	86	79	54	81	86	83	48	125	86
118	118	40	87	13	31	8	87	43	31	110	87	50	84	60	87
96	96	39	88	73	8	84	88	7	8	12	88	17	120	122	88
74	74	38	89	6	112	33	89	98	112	41	89	111	29	57	89
52	52	37	90	66	89	109	90	62	89	70	90	78	65	119	90
30	30	36	91	126	66	58	91	26	66	99	91	45	101	54	91
8	8	35	92	59	43	7	92	117	43	1	92	12	10	116	92
113	113	34	93	119	20	83	93	81	20	30	93	106	46	51	93
91	91	33	94	52	124	32	94	45	124	59	94	73	82	113	94
69	69	32	95	112	101	108	95	9	101	88	95	40	118	48	95
47	47	31	96	45	78	57	96	100	78	117	96	7	27	110	96
25	25	30	97	105	55	6	97	64	55	19	97	101	63	45	97
3	3	29	98	38	32	82	98	28	32	48	98	68	99	107	98
108	108	28	99	98	9	31	99	119	9	77	99	35	8	42	99
86	86	27	100	31	113	107	100	83	113	106	100	2	44	104	100
64	64	26	101	91	90	56	101	47	90	8	101	96	80	39	101
42	42	25	102	24	67	5	102	11	67	37	102	63	116	101	102
20	20	24	103	84	44	81	103	102	44	66	103	30	25	36	103
125	125	23	104	17	21	30	104	66	21	95	104	124	61	98	104

Tabela 5.3: Seqüências de usuário: autovetores da matriz de transformação da FFCT-2e, $N = 4$, $p = 127$, $\zeta = 119 + j119$. Autovalores: $\lambda_1 = 1$, $\lambda_2 = 20$, $\lambda_3 = 108$ e $\lambda_4 = 126$ (continuação da tabela anterior).

$x_{1,0}$	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{2,0}$	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{3,0}$	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{4,0}$	$x_{4,1}$	$x_{4,2}$	$x_{4,3}$
103	103	22	105	77	125	106	105	30	125	124	105	91	97	33	105
81	81	21	106	10	102	55	106	121	102	26	106	58	6	95	106
59	59	20	107	70	79	4	107	85	79	55	107	25	42	30	107
37	37	19	108	3	56	80	108	49	56	84	108	119	78	92	108
15	15	18	109	63	33	29	109	13	33	113	109	86	114	27	109
120	120	17	110	123	10	105	110	104	10	15	110	53	23	89	110
98	98	16	111	56	114	54	111	68	114	44	111	20	59	24	111
76	76	15	112	116	91	3	112	32	91	73	112	114	95	86	112
54	54	14	113	49	68	79	113	123	68	102	113	81	4	21	113
32	32	13	114	109	45	28	114	87	45	4	114	48	40	83	114
10	10	12	115	42	22	104	115	51	22	33	115	15	76	18	115
115	115	11	116	102	126	53	116	15	126	62	116	109	112	80	116
93	93	10	117	35	103	2	117	106	103	91	117	76	21	15	117
71	71	9	118	95	80	78	118	70	80	120	118	43	57	77	118
49	49	8	119	28	57	27	119	34	57	22	119	10	93	12	119
27	27	7	120	88	34	103	120	125	34	51	120	104	2	74	120
5	5	6	121	21	11	52	121	89	11	80	121	71	38	9	121
110	110	5	122	81	115	1	122	53	115	109	122	38	74	71	122
88	88	4	123	14	92	77	123	17	92	11	123	5	110	6	123
66	66	3	124	74	69	26	124	108	69	40	124	99	19	68	124
44	44	2	125	7	46	102	125	72	46	69	125	66	55	3	125
22	22	1	126	67	23	51	126	36	23	98	126	33	91	65	126

Tabela 5.4: Autovalores associados a matrizes de transformação da FFCT-2e de comprimento N sobre $GF(p)$, utilizando o elemento unimodular ζ .

p	N	ζ	autovalores
257	2	$256j$	64, 253
127	2	$256j$	1, 20, 108, 126
23	2	$256j$	1, 22
127	8	$106 + 103j$	1, 47, 100, 126
47	12	$31 + 11j$	1, 46
103	13	$97 + 45j$	1, 65, 84
127	16	$97 + 25j$	1, 49, 70, 126
71	18	$53 + 23j$	1, 4, 14, 18, 59, 65, 66, 70
103	26	$7 + 40j$	24, 73
127	32	$7 + 29j$	1, 22, 28, 52, 59, 126

CAPÍTULO 6

POLINÔMIOS DE CHEBYSHEV: UMA NOVA DEFINIÇÃO SOBRE CORPOS FINITOS E UM ALGORITMO PARA MULTIPLICAÇÃO POLINOMIAL

OS POLINÔMIOS de Chebyshev têm sido uma ferramenta matemática essencial em diversos campos do conhecimento. Em Eletrônica, por exemplo, tais polinômios possuem um importante papel no projeto de filtros analógicos e digitais com características próximas às ideais [44]. Recentemente, a série de Chebyshev, isto é, a aproximação de uma função em termos de polinômios de Chebyshev, foi proposta para analisar não-linearidades de circuitos elétricos. O uso de tal expansão provê mais precisão, quando comparado ao de outras expansões, como aquela em série de Taylor [88].

Técnicas de interpolação via polinômios de Chebyshev têm sido parte de algoritmos numéricos para o cálculo do coeficiente de dispersão cromática de fibras ópticas. Isso permite traçar curvas de dispersão que descrevem o comportamento das fibras [89]. Tais técnicas também são bastante úteis na síntese digital de frequências com forma de onda arbitrária, procedimentos de reamostragem para modems multitonos discretos e em muitos outros cenários [90], [91]. De maneira geral, o uso de polinômios de Chebyshev para aproximar uma função garante

mais estabilidade que a representação monomial ou o uso de outra base. Em particular, quando um truncamento se faz necessário, o rápido decaimento dos coeficientes da expansão de Chebyshev favorece a erros de arredondamento relativamente pequenos [92], [93].

Os aspectos mencionados que, aliados a outros fatores, tornam os polinômios de Chebyshev altamente atrativos em análise numérica e, em particular, em técnicas de aproximação e interpolação, também motivam o estudo de tais polinômios sobre corpos finitos. Nessa direção, introduz-se, neste capítulo, uma nova definição para polinômios de Chebyshev sobre corpos finitos. Diferentemente de propostas anteriores, em que os polinômios de Chebyshev, por possuírem coeficientes inteiros, são simplesmente considerados módulo um número primo [94], a presente abordagem sugere uma definição mais formal, baseada na trigonometria sobre corpos finitos. À luz da teoria desenvolvida, estudam-se algumas propriedades dos respectivos polinômios.

Nas seções finais, duas aplicações são discutidas. Primeiro, apresenta-se a análise de um algoritmo de cifragem de chave pública baseado nos polinômios de Chebyshev sobre corpos finitos. De forma mais específica, demonstra-se que um criptossistema baseado no algoritmo mencionado é seguro, uma vez que a solução de um logaritmo discreto é necessária para que se recupere o texto claro a partir do texto cifrado correspondente. Finalmente, desenvolve-se um algoritmo rápido para multiplicação de polinômios na forma de Chebyshev. O procedimento, que é aplicável tanto no contexto dos números reais quanto no de corpos finitos, é baseado no algoritmo de Karatsuba [95]. Comparações entre o método proposto e outras formas de efetuar a mesma operação são realizadas.

6.1 Polinômios de Chebyshev sobre Corpos Finitos Primos

6.1.1 A função co-seno inversa sobre corpos finitos

Para que se introduza uma definição trigonométrica de polinômios de Chebyshev sobre corpos finitos, inicialmente, é necessário definir a função inversa do co-seno no mesmo contexto. Com esse objetivo, é conveniente observar que a função co-seno relacionada a um elemento específico $\zeta_a \in \text{GI}(p)$, com ordem multiplicativa $\text{ord}(\zeta_a)$, pode ser expressa como

$$\cos_{\zeta_a} : \mathbb{Z}_{\text{ord}(\zeta_a)} \rightarrow \mathbb{I}_{\zeta_a}, \quad (6.1)$$

em que $\mathbb{Z}_{\text{ord}(\zeta_a)}$ denota o conjunto dos inteiros módulo $\text{ord}(\zeta_a)$ e \mathbb{I}_{ζ_a} é o conjunto imagem relacionado à função \cos_{ζ_a} . Devido à simetria par do co-seno, a cardinalidade de \mathbb{I}_{ζ_a} é

Tabela 6.1: Todos os possíveis valores para $\cos_{\zeta_1}(x)$, onde $\zeta_1 = 2 + 2j$ é um elemento unimodular pertencente a $GI(7)$, tal que $\text{ord}(\zeta_1) = 8$.

x	$\cos_{\zeta_1}(x)$	x	$\cos_{\zeta_1}(x)$
0	1	4	6
1	2	5	5
2	0	6	0
3	5	7	2

$\#\{\mathbb{I}_{\zeta_a}\} = \lfloor \text{ord}(\zeta_a)/2 \rfloor + 1$, onde $\lfloor \cdot \rfloor$ denota o maior número inteiro menor ou igual ao argumento. Portanto, a função co-seno inversa pode ser expressa como

$$\arccos_{\zeta_a} : \mathbb{I}_{\zeta_a} \rightarrow \mathbb{Z}_{\lfloor \frac{\text{ord}(\zeta_a)}{2} \rfloor + 1}. \quad (6.2)$$

Considerando um elemento ζ_1 que seja gerador de G_1 , o conjunto unimodular de $GI(p)$, a função co-seno associada pode ser expressa como na Equação (6.1) e, de forma análoga, o respectivo conjunto imagem pode ser denotado por \mathbb{I}_{ζ_1} . A partir da Proposição 2.1 e do Lema 2.1, uma vez que ζ_1 é unimodular e $\text{ord}(\zeta_1) = p + 1$, conclui-se que \mathbb{I}_{ζ_1} é o conjunto de todos os elementos da forma $\Re\{\zeta\}$, tal que $\zeta \in GI(p)$ é unimodular. Sua cardinalidade é $\#\{\mathbb{I}_{\zeta_1}\} = (p + 1)/2 + 1$, a qual, claramente, representa a máxima cardinalidade do conjunto \mathbb{I}_{ζ} para ζ unimodular.

Exemplo 6.1 Seja $\zeta_1 = 2 + 2j$ um elemento unimodular de $GI(7)$ tal que $\text{ord}(\zeta_1) = 8$. Na Tabela 6.1, todos os possíveis valores para $\cos_{\zeta_1}(x)$ são mostrados. Nota-se que $\mathbb{I}_{\zeta_1} = \{0, 1, 2, 5, 6\}$ e, conseqüentemente, a função $\arccos_{\zeta_1}(x)$ não está definida para $x \in \{3, 4\}$.

Conforme ilustrado no Exemplo 6.1, embora $\zeta_1 \in GI(p)$ seja um elemento unimodular com máxima ordem multiplicativa $p + 1$, a função $\arccos_{\zeta_1}(x)$ não está definida para todo elemento $x \in \mathbb{Z}_p$. Assim, para que se calcule a função co-seno inversa de elementos que estão em \mathbb{Z}_p mas não estão em \mathbb{I}_{ζ_1} , é necessário selecionar um elemento $\zeta_2 (\neq \zeta_1)$ de modo que $\mathbb{I}_{\zeta_1} \cup \mathbb{I}_{\zeta_2} = \mathbb{Z}_p$. Tal elemento é especificado pelo seguinte teorema.

Teorema 6.1 Seja $\zeta_1 \in GI(p)$ um gerador do grupo G_1 e ζ_2 uma raiz primitiva de $GF(p)$. Então, $\mathbb{I}_{\zeta_1} \cup \mathbb{I}_{\zeta_2} = \mathbb{Z}_p$.

Demonstração: A partir de observações anteriores, sabe-se que $\#\{\mathbb{I}_{\zeta_1}\} = (p + 1)/2 + 1$ e $\#\{\mathbb{I}_{\zeta_2}\} = (p - 1)/2 + 1$. De acordo com o Lema 2.1, o conjunto $\mathbb{I}_{\zeta_1} \cap \mathbb{I}_{\zeta_2}$ é composto por

Tabela 6.2: Todos os possíveis valores para $\cos_{\zeta_2}(x)$, onde $\zeta_2 = 3$ é um elemento pertencente a $GF(7)$, tal que $\text{ord}(\zeta_2) = 6$.

x	$\cos_{\zeta_2}(x)$	x	$\cos_{\zeta_2}(x)$
0	1	3	6
1	4	4	3
2	3	5	4

elementos $\cos_{\zeta_2}(x) = \Re\{\zeta\}$, $x = 0, 1, \dots, p-2$, para todo ζ unimodular. Daí, com base na Definição 2.2, pode-se escrever

$$[\cos_{\zeta_2}(x)]^2 + b^2 \equiv 1 \pmod{p}, \quad b \in GF(p).$$

Aplicando a definição da função co-seno, a equação acima é reescrita como

$$\left(\frac{\zeta_2^x + \zeta_2^{-x}}{2}\right)^2 + b^2 \equiv 1 \pmod{p}.$$

Expandindo o termo do lado esquerdo da última equação, após algumas simplificações, tem-se

$$(\zeta_2^x - \zeta_2^{-x})^2 \equiv -4b^2 \pmod{p}.$$

Tomando a raiz quadrada em ambos os lados da equação acima, obtém-se

$$\zeta_2^x - \zeta_2^{-x} \equiv \pm 2bj \pmod{p}.$$

Uma vez que $\zeta_2 \in GF(p)$, a relação acima é satisfeita apenas se $b = 0$. Conseqüentemente, os possíveis valores de x são 0 e $(p-1)/2$. Então, $\#\{\mathbb{I}_{\zeta_1} \cap \mathbb{I}_{\zeta_2}\} = 2$ e

$$\begin{aligned} \#\{\mathbb{I}_{\zeta_1} \cup \mathbb{I}_{\zeta_2}\} &= \#\{\mathbb{I}_{\zeta_1}\} + \#\{\mathbb{I}_{\zeta_2}\} - \#\{\mathbb{I}_{\zeta_1} \cap \mathbb{I}_{\zeta_2}\} \\ &= \frac{p+1}{2} + 1 + \frac{p-1}{2} + 1 - 2 = p. \end{aligned}$$

Todo elemento pertencente a \mathbb{I}_{ζ_1} ou \mathbb{I}_{ζ_2} também pertence a \mathbb{Z}_p , o que conclui a demonstração. ■

Exemplo 6.2 Seja $\zeta_2 = 3$ um elemento de $GF(7)$ tal que $\text{ord}(\zeta_2) = 6$. Na Tabela 6.2, todos os possíveis valores de $\cos_{\zeta_2}(x)$ são mostrados. Nota-se que $\mathbb{I}_{\zeta_2} = \{1, 3, 4, 6\}$. Como ζ_2 foi escolhido de acordo com o Teorema 6.1, se $\mathbb{I}_{\zeta_1} = \{0, 1, 2, 5, 6\}$ é considerado (Exemplo 6.1), tem-se $\mathbb{I}_{\zeta_1} \cup \mathbb{I}_{\zeta_2} = \mathbb{Z}_p$.

6.1.2 Polinômios de Chebyshev sobre corpos finitos primos: definição

Nesta seção, apresenta-se uma nova definição para polinômios de Chebyshev sobre corpos finitos. A definição, que é baseada na função co-seno sobre corpos finitos e na sua inversa, é análoga à definição clássica de polinômios de Chebyshev sobre os números reais [92].

Definição 6.1 Os polinômios de Chebyshev do primeiro tipo sobre $GF(p)$ são definidos como

$$T_n(x) := \cos_\zeta(n \arccos_\zeta(x))(\text{mod } p), \quad (6.3)$$

em que $n \in \mathbb{N}$, $\zeta \in GI(p)$ e $x \in \mathbb{I}_\zeta$.

A Equação (6.3) corresponde ao co-seno de múltiplos de um arco. Portanto, como no caso real, pode-se realizar uma expansão usando a fórmula de De Moivre. Esse procedimento fornece polinômios de grau n em termos de co-senos do respectivo arco [92]. Entretanto, polinômios de Chebyshev para diferentes valores de n podem ser obtidos a partir de uma relação de recorrência. Para o caso sobre corpos finitos, essa relação é derivada a partir da Definição 6.1, onde se aplica a fórmula de adição de arcos [20]. Como resultado, obtém-se

$$T_{n+1}(x) = 2x T_n(x) - T_{n-1}(x)(\text{mod } p), \quad (6.4)$$

em que $x \in GF(p)$, $n \in \mathbb{N}$, $T_0(x) = 1$ e $T_1(x) = x$.

Os polinômios de Chebyshev sobre $GF(p)$ possuem a seguinte periodicidade.

Proposição 6.1 Seja ζ um elemento não-nulo em $GI(p)$ tal que $\text{ord}(\zeta) = N$. Se $x \in \mathbb{I}_\zeta$, então $T_{tN \pm n}(x) = T_n(x)$, $t \in \mathbb{Z}$.

Demonstração: A partir da definição 6.1, tem-se

$$T_{tN \pm n}(x) = \cos_\zeta((tN \pm n) \arccos_\zeta(x))(\text{mod } p).$$

Aplicando a fórmula de adição de arcos, a equação acima é escrita como

$$\begin{aligned} T_{tN \pm n}(x) &= \cos_\zeta(tN \arccos_\zeta(x)) \cos_\zeta(n \arccos_\zeta(x)) \\ &\mp \sin_\zeta(tN \arccos_\zeta(x)) \sin_\zeta(n \arccos_\zeta(x)). \end{aligned}$$

Uma vez que $\text{ord}(\zeta) = N$, aplicando a Definição 2.3, sabe-se que $\cos_\zeta(tN \arccos_\zeta(x)) = 1$ e $\sin_\zeta(tN \arccos_\zeta(x)) = 0$. Conseqüentemente, a última equação reduz-se a

$$T_{tN \pm n}(x) = \cos_\zeta(n \arccos_\zeta(x)) = T_n(x).$$

■

Embora a Definição 6.1 requeira $x \in \mathbb{I}_\zeta$, essa restrição pode ser desconsiderada, caso se deseje simplesmente avaliar $T_n(x)$ para valores particulares de n , x e um número primo p . Para isso, usa-se a Equação (6.4), a qual não depende de ζ e, portanto, não requer o cálculo explícito de valores da função co-seno e de sua inversa.

A propriedade de semi-grupo dos polinômios de Chebyshev também é válida para o caso sobre corpos finitos [96]:

$$T_r(T_s(x)) = T_s(T_r(x)) = T_{rs}(x). \quad (6.5)$$

Conforme será visto, essa propriedade desempenha um importante papel na correteza do algoritmo de cifragem discutido a seguir.

6.2 Algoritmo de Cifragem de Chave Pública Baseado em Polinômios de Chebyshev sobre Corpos Finitos

Recentemente, propôs-se um algoritmo de cifragem de chave pública baseado em polinômios de Chebyshev sobre corpos finitos [96]. A seguir, apresenta-se uma breve revisão do algoritmo, o qual é dividido em três partes: geração do par de chaves, cifragem da mensagem e decifragem da mensagem.

Geração do par de chaves

Para gerar as chaves, Alice realiza o seguinte procedimento:

- (1) Seleciona aleatoriamente os números inteiros $s \in \mathbb{Z}_p$ e $x \in \text{GF}(p)$, $s \neq 0, 1$ e $x \neq 0, 1$, e calcula $T_s(x)$.
- (2) Sua chave secreta é s e sua chave pública é $(x, T_s(x))$.

Cifragem da mensagem

Assume-se que Bob deseja enviar a mensagem $M \in \text{GF}(p)$, $M \neq 0$, para Alice. Ele realiza o seguinte procedimento:

- (1) Seleciona aleatoriamente o número inteiro $r \in \mathbb{Z}_p$, $r \neq 0, 1$.
- (2) A partir da chave pública de Alice $(x, T_s(x))$, calcula $T_r(x)$, $T_{rs}(x) = T_r(T_s(x))$ e $X = M T_{rs}(x) \pmod{p}$.
- (3) Envia o texto cifrado $C = (T_r(x), X)$ para Alice.

Decifragem da mensagem

Após receber a mensagem cifrada, para que a mesma seja decifrada, Alice realiza o seguinte procedimento:

- (1) Usa sua chave secreta s para calcular $T_{sr}(x) = T_s(T_r(x))$.
- (2) Recupera M calculando $M = X(T_{sr}(x))^{-1}(\text{mod } p)$.

Devido à propriedade de semi-grupo dos polinômios de Chebyshev sobre corpos finitos, M é recuperado de forma correta [97], [96].

6.2.1 Análise da segurança do algoritmo

Num trabalho anterior a este (ver Seção 3 de [96]), o criptossistema baseado no algoritmo apresentado foi considerado seguro devido à não-existência de uma definição trigonométrica para polinômios de Chebyshev sobre corpos finitos, necessária à aplicação do ataque descrito em [98]. Nesta seção, o referido ataque é estendido ao contexto de corpos finitos primos. Demonstra-se que sua aplicação envolve o problema do logaritmo discreto.

Recuperação do texto claro

Dados o texto cifrado $(T_r(x), X)$ e a chave pública de Alice $(x, T_s(x))$, um adversário, para obter a mensagem M , realiza o seguinte procedimento:

- (1) Calcula um número r' tal que $T_{r'}(x) = T_r(x)$.
- (2) Avalia $T_{r's}(x) = T_{r'}(T_s(x))$.
- (3) Recupera $M = X(T_{r's}(x))^{-1}(\text{mod } p)$.

Esse procedimento é sempre bem sucedido porque, se $T_{r'}(x) = T_r(x)$, então

$$T_{rs}(x) = T_{sr}(x) = T_s(T_r(x)) = T_s(T_{r'}(x)) = T_{r'}(T_s(x)).$$

O cálculo de r' é feito com base no seguinte resultado.

Lema 6.1 Para cada par $(x, T_r(x))$, o número inteiro r' satisfaz $T_{r'}(x) = T_r(x)$ se e somente se

$$r' = \pm \arccos_{\zeta}(T_r(x)) (\arccos_{\zeta}(x))^{-1}(\text{mod } N), \quad (6.6)$$

em que $N = \text{ord}(\zeta)$.

Demonstração: Assumindo que

$$r' = \pm \arccos_{\zeta}(T_r(x)) (\arccos_{\zeta}(x))^{-1} \pmod{N},$$

a partir da Definição 6.1,

$$T_{r'}(x) = \cos_{\zeta}(r' \arccos_{\zeta}(x)) = \cos_{\zeta}(\pm \arccos_{\zeta}(T_r(x))) = T_r(x).$$

Por outro lado, assumindo que $T_{r'}(x) = T_r(x)$ para algum número r' , tem-se,

$$T_{r'}(x) = \cos_{\zeta}(r' \arccos_{\zeta}(x)) = T_r(x).$$

Aplicando a função \arccos_{ζ} a ambos os lados da última igualdade, obtém-se

$$\arccos_{\zeta}(\cos_{\zeta}(r' \arccos_{\zeta}(x))) = \arccos_{\zeta}(T_r(x)). \quad (6.7)$$

Seja $y = \arccos_{\zeta}(w)$. Para todo $\beta = 0, \dots, N-1$, a relação de simetria $\cos_{\zeta}(\beta) = \cos_{\zeta}(-\beta \pmod{N})$ é válida; devido à periodicidade da função \cos_{ζ} , se $\cos_{\zeta}(\beta) = w$, tem-se $\beta = \pm y \pmod{N}$. Portanto, a Equação (6.7) é satisfeita se e somente se

$$r' \arccos_{\zeta}(x) = \pm \arccos_{\zeta}(T_r(x)) \pmod{N}.$$

Multiplicando ambos os lados da última equação por $(\arccos_{\zeta}(x))^{-1} \pmod{N}$, obtém-se

$$r' = \pm \arccos_{\zeta}(T_r(x)) (\arccos_{\zeta}(x))^{-1} \pmod{N},$$

e o lema é satisfeito. ■

Diferentemente do ataque sobre criptosistemas baseados nos polinômios de Chebyshev segundo a definição clássica, no cenário de corpos finitos, o Lema 6.1 fornece apenas dois valores de r' . Naturalmente, esses valores são inteiros, não sendo, portanto, necessário considerar aspectos de precisão numérica nem resolver sistemas de equações modulares para que os mesmos sejam obtidos [98].

O cálculo de r' pela Equação (6.6) depende da existência da função co-seno inversa. Conforme previamente discutido, o $\arccos_{\zeta}(x)$ é definido se e somente se $x \in \mathbb{I}_{\zeta}$. De acordo com o passo (1) do procedimento de geração do par de chaves, o parâmetro x pode assumir qualquer valor sobre o intervalo $[2, p-1]$. Assim, precisa-se utilizar o Teorema 6.1 para especificar elementos ζ_1 e ζ_2 tais que $\mathbb{I}_{\zeta_1} \cup \mathbb{I}_{\zeta_2} = \mathbb{Z}_p$. Isso assegura que $\arccos_{\zeta}(x)$ sempre está definido para $\zeta = \zeta_1$ ou $\zeta = \zeta_2$. Além disso, $\arccos_{\zeta}(T_r(x)) = r \arccos_{\zeta}(x)$. Assim, o cálculo de r' não requer um cálculo explícito de $(\arccos_{\zeta}(x))^{-1} \pmod{N}$.

Uma fórmula fechada para a função co-seno inverso pode ser derivada diretamente da definição da função co-seno. Após alguma manipulação de $\cos_{\zeta}(x) = (\zeta^x + \zeta^{-x})/2$, obtém-se

$$\arccos_{\zeta}(x) = \log_{\zeta} \left(x + \sqrt{x^2 - 1} \right),$$

a função \arccos_{ζ} , calculada módulo p , sob a forma de um logaritmo discreto. Aplicando a fórmula acima à Equação (6.6) e usando algumas propriedades dos logaritmos, tem-se

$$r' = \pm \left[\log_{x + \sqrt{x^2 - 1}} \left(T_r(x) + \sqrt{T_r(x)^2 - 1} \right) \right] \pmod{N}.$$

Isso significa que, dado um texto cifrado, a recuperação do texto claro correspondente através do cálculo de r' envolve o problema do logaritmo discreto. Devido à presença de raízes quadradas na equação acima, é possível que tal problema requeira a consideração do corpo $\text{GI}(p)$.

Um exemplo

Dada a chave pública de Alice $(x, T_s(x))$ e o texto cifrado $C = (T_r(x), X)$, em que $X = M T_{r_s}(x)$, neste exemplo, mostra-se como um adversário calcula o valor de $T_{r_s}(x)$ e recupera M .

Inicialmente, é preciso definir o corpo finito a ser usado na geração dos parâmetros das chaves de Alice. Seja $p = 31$, $x = 26$ e $s = 15$. A partir da Equação (6.4), calcula-se $T_s(x)$. É importante lembrar que esse cálculo pode ser feito através de uma relação de recorrência. Dessa forma, para número primos grandes, que são usados em aplicações práticas, algoritmos rápidos são aplicados como forma de prover eficiência computacional [94]. Daí, a chave pública de Alice é dada pelo par $(x, T_s(x)) = (26, 29)$. Assume-se que Bob escolhe $r = 4$, com o objetivo de cifrar a mensagem M . Analogamente a $T_s(x)$, os parâmetros $T_r(x) = 27$ e $T_r(T_s(x)) = 4$ são obtidos.

Conforme anteriormente observado, para usar o Lema 6.1 e calcular r' , um adversário deve selecionar um elemento $\zeta \in \text{GI}(31)$ tal que $\arccos_{\zeta}(26)$ exista. Como $x = 26$ é a parte real de um elemento unimodular sobre $\text{GI}(p)$, sabe-se que $26 \in \mathbb{I}_{\zeta_1}$, em que ζ_1 é unimodular e $\text{ord}(\zeta_1) = 32$. Pode-se escolher $\zeta_1 = 2 + 11j$, que possui as características mencionadas, e usá-lo para realizar o ataque. Todos os possíveis valores para $\cos_{\zeta_1}(x)$ são mostrados na Tabela 6.3.

Tabela 6.3: Todos os possíveis valores de $\cos_{\zeta_1}(x)$, onde $\zeta_1 = 2 + 11j$ é um elemento unimodular de $GI(31)$ tal que $\text{ord}(\zeta_1) = 32$.

x	$\cos_{\zeta_1}(x)$	x	$\cos_{\zeta_1}(x)$	x	$\cos_{\zeta_1}(x)$
0	1	6	18	12	27
1	2	7	20	13	5
2	7	8	0	14	24
3	26	9	11	15	29
4	4	10	13	16	30
5	21	11	10	—	—

Usando a Equação (6.6), tem-se

$$r' = \pm \arccos_{\zeta_1}(27) (\arccos_{\zeta_1}(26))^{-1} \pmod{32} = \pm 4 \pmod{32},$$

e o texto claro enviado por Bob é calculado pelo adversário como $M = X(T_4(T_s(x)))^{-1} = X(T_{28}(T_s(x)))^{-1} \pmod{p}$.

Nota-se que $\zeta_1 = 26 + 10j$, por exemplo, também poderia ter sido escolhido. Isso tornaria o cálculo de r' mais fácil, uma vez que se saberia de imediato que $\arccos_{\zeta_1}(26) = 1$. Se o parâmetro x fosse tal que $x \notin \mathbb{I}_{\zeta_1}$, o procedimento para recuperação do texto claro seria análogo. Nesse caso, de acordo com o Teorema 6.1, dever-se-ia selecionar um elemento $\zeta_2 \in \text{GF}(p)$ tal que $\text{ord}(\zeta_2) = 30$ e usá-lo para realizar o ataque. Necessariamente, ter-se-ia $x \in \mathbb{I}_{\zeta_2}$ e a existência de $\arccos_{\zeta_2}(x)$ estaria assegurada.

6.3 Multiplicação de Polinômios na Forma de Chebyshev Baseada no Algoritmo de Karatsuba

Nesta seção, discute-se a multiplicação de polinômios na forma de Chebyshev. De modo específico, dados dois polinômios $a(x)$ e $b(x)$ na forma de Chebyshev, obtém-se o polinômio $c(x) = a(x) \cdot b(x)$ também escrito na forma de Chebyshev. Em [99], duas abordagens para esse problema foram apresentadas. A primeira corresponde a uma maneira direta de realizar tal multiplicação, ao passo que a segunda é baseada na transformada discreta do co-seno (DCT).

O método aqui introduzido consiste em aplicar o algoritmo de Karatsuba [95], [100] aos polinômios $a'(x)$ e $b'(x)$, obtidos a partir de $a(x)$ e $b(x)$. Os coeficientes do produto resultante são denotados por c'_i . Daí, mostra-se que os coeficientes de $c(x)$, denotados por c_i ,

podem ser obtidos a partir dos coeficientes c'_i . Embora esse procedimento, que requer operações aritméticas extras, envolva uma complexidade computacional quadrática, o número de multiplicações é reduzido pela metade, quando comparado à multiplicação direta [99]. Sob esse aspecto, para polinômios $a(x)$ e $b(x)$ com graus pequenos, o que cobre diversas aplicações práticas da expansão de Chebyshev [88], [89], [90], o método proposto é também mais eficiente que a abordagem baseada na DCT. Além disso, algumas vantagens relacionadas à implementação e à precisão são observadas.

Conforme mencionado anteriormente, o método descrito é válido tanto no contexto dos números reais quanto no de corpos finitos. No entanto, como existe uma analogia bastante direta entre os dois casos, apresenta-se um desenvolvimento considerando apenas os polinômios de Chebyshev sobre os números reais.

6.3.1 *Multiplicação de Polinômios na Forma de Chebyshev*

A definição clássica de polinômios de Chebyshev do primeiro tipo é

$$T_i(x) := \cos(i \cdot \arccos x), \quad (6.8)$$

em que $i \in \mathbb{N}$ e $x \in [-1, 1]$. A partir da Equação (6.8), obtém-se $T_0(x) = 1$, $T_1(x) = x$, e a relação de recorrência

$$T_{i+1}(x) = 2xT_i(x) - T_{i-1}(x),$$

que, de forma simples, permite obter polinômios de Chebyshev para quaisquer valores de i .

Mostra-se que qualquer polinômio real $a(x)$ de grau $\leq N - 1$ pode ser escrito como uma combinação linear de polinômios de Chebyshev do primeiro tipo [92]. Usualmente, tal representação é chamada *expansão de Chebyshev* e dada por

$$a(x) = \frac{a_0}{2} + \sum_{i=1}^{N-1} a_i T_i(x), \quad a_i \in \mathbb{R}. \quad (6.9)$$

A partir da relação

$$T_i T_j = \frac{T_{i+j} + T_{|i-j|}}{2}, \quad i, j \in \mathbb{N},$$

a qual se verifica usando simples identidades trigonométricas, uma regra de multiplicação para polinômios na forma de Chebyshev pode ser derivada. A mesma é descrita na seguinte proposição [99].

Proposição 6.2 *Sejam $a(x)$ e $b(x)$ polinômios de grau $N - 1$ dados na forma de Chebyshev*

$$a(x) = \frac{a_0}{2} + \sum_{i=1}^{N-1} a_i T_i(x)$$

e

$$b(x) = \frac{b_0}{2} + \sum_{i=1}^{N-1} b_i T_i(x),$$

em que $a_i, b_i \in \mathbb{R}$. Então, o produto $c(x) = a(x) \cdot b(x)$ tem a forma de Chebyshev

$$c(x) = \frac{c_0}{2} + \sum_{i=1}^{2N-2} c_i T_i(x)$$

com

$$2c_i = \begin{cases} a_0 \cdot b_0 + 2 \sum_{l=1}^{N-1} a_l \cdot b_l, & i = 0; \\ \sum_{l=0}^i a_{i-l} \cdot b_l + \sum_{l=1}^{N-1-i} (a_l \cdot b_{l+i} + a_{l+i} \cdot b_l), & i = 1, \dots, N-2; \\ \sum_{l=i-N+1}^{N-1} a_{i-l} \cdot b_l, & i = N-1, \dots, 2N-2. \end{cases} \quad (6.10)$$

O cálculo de todos os coeficientes $c_i, i = 0, \dots, 2N - 2$, diretamente pela Equação (6.10) é chamado de “método direto” e envolve $\mathcal{O}(N^2)$ multiplicações reais [99]. Na mesma equação, o número de todos os possíveis produtos $a_i \cdot b_j, i, j = 0, \dots, N - 1$, e o número de produtos por $1/2$ é contado. Isso fornece $M_d(n)$, o número exato de multiplicações para calcular todos os coeficientes c_i usando esse método,

$$M_d(N) = N^2 + 2N - 1. \quad (6.11)$$

De acordo com a Equação (6.10), dados os números inteiros i_1 e i_2 tais que $1 \leq i_1 \leq N - 2$ e $i_1 < i_2 \leq 2N - 2$, qualquer termo com a forma $(a_l \cdot b_{l+i_1} + a_{l+i_1} \cdot b_l), l = 1, \dots, N - 1 - i_1$, é computado previamente na soma $\sum_{l=0}^{i_2} a_{i_2-l} \cdot b_l$ ou na soma $\sum_{l=i_2-N+1}^{N-1} a_{i_2-l} \cdot b_l$. Conseqüentemente, na segunda linha da referida equação, as adições $(a_l \cdot b_{l+i} + a_{l+i} \cdot b_l)$ não precisam ser contadas. Portanto, $A_d(N)$, o número exato de adições para obter todos os coeficientes c_i usando o método direto é

$$A_d(N) = N - 1 + \sum_{i=1}^{N-2} (N - 1) + \sum_{i=N-1}^{2N-2} (2N - 2 - i) = \frac{(N - 1)(3N - 2)}{2}. \quad (6.12)$$

6.3.2 Multiplicação de Polinômios na Forma de Chebyshev Baseada no Algoritmo de Karatsuba

Nesta seção, o algoritmo proposto, que, por simplicidade, é chamado de “método de Karatsuba”, é apresentado: usa-se o algoritmo de Karatsuba para calcular o produto de dois

polinômios cujos coeficientes de Chebyshev são dados. Os resultados intermediários do algoritmo de Karatsuba são armazenados e, então, usados para obter os coeficientes de Chebyshev c_i do polinômio produto. O ponto-chave desse procedimento é aplicar o algoritmo de Karatsuba para realizar uma multiplicação polinomial ordinária e obter os coeficientes de Chebyshev através de algumas equações.

Mais especificamente, para usar o método de Karatsuba para multiplicar $a(x)$ e $b(x)$, os coeficientes a_i e b_i são associados aos termos de grau i , $i = 0, \dots, N-1$, na representação monomial. Esse procedimento fornece os polinômios $a'(x) = \sum_{i=0}^{N-1} a_i x^i$ e $b'(x) = \sum_{i=0}^{N-1} b_i x^i$. Aplicando o algoritmo de Karatsuba, obtêm-se os polinômios $c'(x) = a'(x) \cdot b'(x) = \sum_{i=0}^{2N-2} c'_i x^i$. Por outro lado, esses coeficientes c'_i são dados por

$$c'_i = \begin{cases} a_0 \cdot b_0, & i = 0; \\ \sum_{l=0}^i a_{i-l} \cdot b_l, & i = 1, \dots, N-2; \\ \sum_{l=i-N+1}^{N-1} a_{i-l} \cdot b_l, & i = N-1, \dots, 2N-2. \end{cases} \quad (6.13)$$

Substituindo a Equação (6.13) na Equação (6.10), obtêm-se

$$2c_i = \begin{cases} c'_i + 2 \sum_{l=1}^{N-1} a_l \cdot b_l, & i = 0; \\ c'_i + \sum_{l=1}^{N-1-i} (a_l \cdot b_{l+i} + a_{l+i} \cdot b_l), & i = 1, \dots, N-2; \\ c'_i, & i = N-1, \dots, 2N-2. \end{cases} \quad (6.14)$$

Dados os coeficientes c'_i calculados pelo algoritmo de Karatsuba, os coeficientes c_i podem ser obtidos a partir da Equação (6.14) com o seguinte número de multiplicações extras: $2N-1$ devido ao fator de escala $1/2$; $N-1$ para calcular os termos $a_l \cdot b_l$, $l = 1, \dots, N-1$; $(N-2)(N-1)/2$ para calcular os termos $(a_l \cdot b_{l+i} + a_{l+i} \cdot b_l) = (a_l + a_{l+i}) \cdot (b_l + b_{l+i}) - a_l \cdot b_l - a_{l+i} \cdot b_{l+i}$, $i = 1, \dots, N-2$, $l = 1, \dots, N-1-i$. Isso implica um número total de multiplicações extras dado por $(N^2 + 3N - 2)/2$.

Os números de adições extras relacionadas à primeira e à segunda linhas da Equação (6.14) é $N-1$ e $5(N-2)(N-1)/2$, respectivamente. Então, o número total de adições extras é $(5N^2 - 13N + 8)/2$. Mostra-se como esses números de operações extras podem ser reduzidos usando os resultados intermediários do algoritmo de Karatsuba previamente aplicado.

O método proposto é dado abaixo. Sua corretude é imediata a partir das Equações (6.10), (6.13) e (6.14).

Algoritmo: Multiplicação de polinômios na forma de Chebyshev baseada no algoritmo de Karatsuba.

Entrada: polinômios $a(x) = \frac{a_0}{2} + \sum_{i=1}^{N-1} a_i T_i(x)$ e $b(x) = \frac{b_0}{2} + \sum_{i=1}^{N-1} b_i T_i(x)$ de grau $N - 1$ na forma de Chebyshev.

Saída: polinômio $c(x) = a(x) \cdot b(x) = \frac{c_0}{2} + \sum_{i=1}^{2N-2} c_i T_i(x)$ de grau $2N - 2$ na forma de Chebyshev.

Passo 1: Aplicar o algoritmo de Karatsuba aos polinômios $a'(x) = \sum_{i=0}^{N-1} a_i x^i$ e $b'(x) = \sum_{i=0}^{N-1} b_i x^i$, cujo produto é denotado por $c'(x) = a'(x) \cdot b'(x) = \sum_{i=0}^{2N-2} c'_i x^i$ e armazenar todos os cálculos intermediários.

Passo 2: Usar os termos envolvidos nas fórmulas da Equação (6.14), previamente computados, para obter $c(x) = a(x) \cdot b(x) = \frac{c_0}{2} + \sum_{i=1}^{2N-2} c_i T_i(x)$.

Detalhes acerca da execução do Passo 2 do algoritmo apresentado são fornecidos nas seções vindouras.

Algoritmo de Karatsuba

Assume-se que se deseja multiplicar dois polinômios, $a'(x)$ e $b'(x)$, com graus $N - 1$. Esses polinômios são dados na forma monomial e têm coeficientes a_i e b_i , respectivamente. Neste trabalho, considera-se a restrição $N = 2^n$, $n \in \mathbb{N}$, embora haja formas eficientes de lidar com polinômios com graus diferentes de $2^n - 1$ [100], [101]. Pode-se escrever

$$a'(x) = A_1(x) x^{N/2} + A_0(x)$$

e

$$b'(x) = B_1(x) x^{N/2} + B_0(x),$$

onde

$$A_1(x) = a_{N-1} x^{N/2-1} + \cdots + a_{N/2},$$

$$A_0(x) = a_{N/2-1} x^{N/2-1} + \cdots + a_0,$$

$$B_1(x) = b_{N-1} x^{N/2-1} + \cdots + b_{N/2},$$

$$B_0(x) = b_{N/2-1} x^{N/2-1} + \cdots + b_0.$$

Tem-se $c'(x) = a'(x) \cdot b'(x)$ dado por

$$c'(x) = [A_1(x) B_1(x)] x^N + [A_0(x) B_1(x) + A_1(x) B_0(x)] x^{N/2} + [A_0(x) B_0(x)]. \quad (6.15)$$

Na última equação, simplificando a notação e omitindo o “(x)”, o termo que multiplica $x^{N/2}$ pode ser reescrito como

$$A_0 B_1 + A_1 B_0 = (A_0 + A_1)(B_0 + B_1) - A_0 B_0 - A_1 B_1.$$

Assim, uma multiplicação é evitada, porque $A_0 B_0$ e $A_1 B_1$ foram previamente calculados. Portanto, o produto de polinômios com grau $N - 1$ pode ser computado usando três produtos de polinômios com grau $(N/2) - 1$. Como esse procedimento é recursivo, mostra-se que o algoritmo de Karatsuba para multiplicar polinômios de grau $N = 2^n$, ou seja, para obter os coeficientes c'_i , pode ser aplicado com $N^{\log_2 3}$ multiplicações e, no máximo $6 N^{\log_2 3} - 8 N + 2$ adições [5].

É importante observar que o algoritmo de Karatsuba não está sendo aplicado na forma de uma caixa preta. Ao invés disso, todos os resultados intermediários são armazenados. Também é importante enfatizar que o algoritmo possui uma estrutura de “três termos” baseada no cálculo recursivo de $A_1 B_1$, $A_0 B_0$ e $A_0 B_1 + A_1 B_0 = (A_0 + A_1)(B_0 + B_1) - A_0 B_0 - A_1 B_1$. Ao longo desta seção, os termos intermediários envolvidos no cálculo de $A_1 B_1$, $A_0 B_0$ e $A_0 B_1 + A_1 B_0$ são respectivamente associados aos símbolos $\overline{11}$, $\overline{00}$ e $\overline{01}$.

Operações Extras para o Algoritmo de Karatsuba

De acordo com a Equação (6.14), para que se obtenham os coeficientes de Chebyshev c_i do polinômio $c(x)$ a partir dos coeficientes c'_i , precisa-se considerar os fatores de escala $1/2$ e calcular os termos $a_l \cdot b_l$, $l = 1, \dots, N - 1$, e $(a_l \cdot b_{l+i} + a_{l+i} \cdot b_l)$, $i = 1, \dots, N - 2$, $l = 1, \dots, N - 1 - i$. Devido à natureza recursiva do algoritmo de Karatsuba, alguns desses termos aparecem calculados junto com outros termos. Portanto, operações aritméticas extras são necessárias para calculá-los separadamente antes de adicioná-los convenientemente aos coeficientes c'_i . Esse procedimento é referido como *separação*. De forma resumida, as operações extras para obter os coeficientes c_i a partir dos coeficientes c'_i estão relacionadas a:

- ▷ operações para separar os termos originalmente computados junto com outros termos;
- ▷ adições de termos $a_l \cdot b_l$ e $(a_l \cdot b_{l+i} + a_{l+i} \cdot b_l)$ respectivamente na primeira e segunda linhas da Equação (6.14);
- ▷ multiplicações pelo fator de escala $1/2$.

O número total de operações extras requeridas é estabelecido no teorema a seguir.

Teorema 6.2 *Sejam $a(x)$ e $b(x)$ polinômios de grau $N - 1$ cujos coeficientes de Chebyshev a_i e b_i , $i = 0, \dots, N - 1$, são dados. Sejam $a'(x) = \sum_{i=0}^{N-1} a_i x^i$ e $b'(x) = \sum_{i=0}^{N-1} b_i x^i$ polinômios cujo produto é denotado por $c'(x) = \sum_{i=0}^{2N-2} c'_i x^i$. Se o polinômio $c'(x)$ for computado usando o algoritmo de Karatsuba, então os coeficientes de Chebyshev c_i , $i = 0, \dots, 2N - 2$, do polinômio $c(x) = a(x) \cdot b(x)$ são obtidos a partir dos coeficientes c'_i , $i = 0, \dots, 2N - 2$, com*

$$M_e(N) = \frac{N^2 - 2 N^{\log_2 3} + 5 N - 2}{2} \quad (6.16)$$

multiplicações extras e

$$A_e(N) \leq \frac{5 N^2 - 6 N^{\log_2 3} + N (1 - \log_2 N)}{2} \quad (6.17)$$

adições extras.

Antes de apresentar a prova do Teorema 6.2, algumas notações são introduzidas e alguns exemplos são desenvolvidos, de modo que a derivação das Equações (6.16) e (6.17) é mais facilmente entendida. O interesse particular é observar os termos intermediários relacionados aos símbolos $\overline{11}$, $\overline{00}$ and $\overline{01}$ que são produzidos juntos. No que segue, os termos com tal característica são escritos entre $\langle \cdot \rangle$; omite-se essa notação para termos da forma $a_i \cdot b_i$.

Exemplo 6.3 *Deseja-se multiplicar os polinômios $a(x)$ e $b(x)$, $N = 2$, cujos coeficientes de Chebyshev a_i e b_i são dados. Usando o algoritmo de Karatsuba para calcular os coeficientes c'_i , tem-se*

$$\overline{11} : c'_2 = A_1 B_1 = a_1 \cdot b_1; \quad (6.18)$$

$$\overline{00} : c'_0 = A_0 B_0 = a_0 \cdot b_0; \quad (6.19)$$

$$\begin{aligned} \overline{01} : c'_1 &= (A_1 + A_0) (B_1 + B_0) - A_1 B_1 - A_0 B_0 \\ &= (a_1 + a_0) (b_1 + b_0) - a_1 b_1 - a_0 b_0 \\ &= \langle a_0 \cdot b_1 + a_1 \cdot b_0 \rangle. \end{aligned} \quad (6.20)$$

A partir da Equação (6.14), obtém-se diretamente $c_2 = c'_2/2$, $c_1 = c'_1/2$ e $c_0 = c'_0/2 + c'_2$, porque não há termos a serem separados. Nesse caso, as operações extras estão relacionadas exclusivamente ao fator de escala $1/2$ e à adição $c'_0/2 + c'_2$, o que resulta em $M_e(2) = 3$ e $A_e(2) = 1$.

Exemplo 6.4 *Neste exemplo, deseja-se multiplicar $a(x)$ e $b(x)$ em que $N = 4$. Como o algoritmo de Karatsuba é recursivo, nesse caso, os cálculos de $A_1 B_1$ e $A_0 B_0$ podem ser vistos como repetições do*

primeiro exemplo. Portanto, os termos intermediários relacionados aos símbolos $\overline{11}$ e $\overline{00}$ são

$$\overline{11} : c'_6 = a_3 \cdot b_3, \quad c'_5 = \langle a_2 \cdot b_3 + a_3 \cdot b_2 \rangle, \quad a_2 \cdot b_2; \quad (6.21)$$

$$\overline{00} : a_1 \cdot b_1, \quad c'_1 = \langle a_0 \cdot b_1 + a_1 \cdot b_0 \rangle, \quad c'_0 = a_0 \cdot b_0. \quad (6.22)$$

O cálculo de $(A_1 + A_0)(B_1 + B_0) - A_1 B_1 - A_0 B_0$ é similar, sendo necessário um cuidado especial com os termos produzidos juntos. Mais especificamente, tem-se $(A_1 + A_0) = (a_3 + a_1)x + (a_2 + a_0)$ e $(B_1 + B_0) = (b_3 + b_1)x + (b_2 + b_0)$, cujo produto produz os termos

$$\langle (a_3 + a_1) \cdot (b_3 + b_1) \rangle, \langle (a_3 + a_1) \cdot (b_2 + b_0) + (a_2 + a_0) \cdot (b_3 + b_1) \rangle \text{ e } \langle (a_2 + a_0) \cdot (b_2 + b_0) \rangle.$$

As subtrações por $A_1 B_1$ e $A_0 B_0$ vêm dos termos intermediários relacionados aos símbolos $\overline{11}$ e $\overline{00}$, respectivamente, nas Equações (6.21) e (6.22). Subtraindo $a_3 \cdot b_3$ e $a_1 \cdot b_1$ de $\langle (a_3 + a_1) \cdot (b_3 + b_1) \rangle$, obtém-se $\langle a_1 \cdot b_3 + a_3 \cdot b_1 \rangle$; subtraindo $a_2 \cdot b_2$ e $a_0 \cdot b_0$ de $\langle (a_2 + a_0) \cdot (b_2 + b_0) \rangle$, obtém-se $\langle a_0 \cdot b_2 + a_2 \cdot b_0 \rangle$; subtraindo $\langle a_2 \cdot b_3 + a_3 \cdot b_2 \rangle$ e $\langle a_0 \cdot b_1 + a_1 \cdot b_0 \rangle$ de $\langle (a_3 + a_1) \cdot (b_2 + b_0) + (a_2 + a_0) \cdot (b_3 + b_1) \rangle$, obtém-se $\langle a_1 \cdot b_2 + a_2 \cdot b_1 + a_0 \cdot b_3 + a_3 \cdot b_0 \rangle$. Portanto, o resultado final para termos intermediários relacionados ao símbolo $\overline{01}$ é

$$\overline{01} : \langle a_1 \cdot b_3 + a_3 \cdot b_1 \rangle, \quad c'_3 = \langle a_1 \cdot b_2 + a_2 \cdot b_1 + a_0 \cdot b_3 + a_3 \cdot b_0 \rangle, \quad \langle a_0 \cdot b_2 + a_2 \cdot b_0 \rangle. \quad (6.23)$$

É importante enfatizar que os coeficientes c'_i , $i = 0, \dots, 6$, são obtidos executando o algoritmo de Karatsuba até o final, após o cálculo de todos os outros termos intermediários. Entretanto, neste momento, o interesse é apenas verificar que termos são produzidos juntos, sendo suficiente realizar o primeiro passo do algoritmo. Nesse sentido, a partir da Equação (6.14), sabe-se particularmente que

$$c_1 = \frac{c'_1 + (a_1 \cdot b_2 + a_2 \cdot b_1 + a_2 \cdot b_3 + a_3 \cdot b_2)}{2}.$$

Daí, com o objetivo de avaliar c_1 , precisa-se calcular $a_1 \cdot b_2 + a_2 \cdot b_1$, uma vez que esse termo é produzido originalmente com $a_0 \cdot b_3 + a_3 \cdot b_0$, como mostra a Equação (6.23). Como se conhece $a_1 \cdot b_1$ e $a_2 \cdot b_2$, isso requer uma multiplicação e quatro adições porque

$$a_1 \cdot b_2 + a_2 \cdot b_1 = (a_1 + a_2) \cdot (b_1 + b_2) - a_1 \cdot b_1 - a_2 \cdot b_2.$$

Todos os outros coeficientes c_i podem ser obtidos de forma similar. Naturalmente, ainda é preciso contar as outras operações extras mencionadas antes do Teorema 6.2. O resultado final é $M_e(4) = 8$ e $A_e(4) = 11$.

Observação: No Exemplo 6.4, não é preciso separar $a_0 \cdot b_3 + a_3 \cdot b_0$. Entretanto, esse termo pode ser obtido com mais uma adição:

$$a_0 \cdot b_3 + a_3 \cdot b_0 = \langle a_1 \cdot b_2 + a_2 \cdot b_1 + a_0 \cdot b_3 + a_3 \cdot b_0 \rangle - \langle a_1 \cdot b_2 + a_2 \cdot b_1 \rangle. \quad (6.24)$$

O cálculo ilustrado na equação acima, que representa o último passo do procedimento de separação, é necessário em multiplicações envolvendo polinômios de graus maiores.

Exemplo 6.5 Neste exemplo, deseja-se multiplicar $a(x)$ e $b(x)$ para $N = 8$. Como no Exemplo 6.4, os cálculos de $A_1 B_1$ e $A_0 B_0$ podem ser vistos como repetições do caso $N = 4$. Os termos obtidos são

$$\begin{aligned} \overline{11} : c'_{14} &= a_7 \cdot b_7, \quad c'_{13} = \langle a_6 \cdot b_7 + a_7 \cdot b_6 \rangle, \quad a_6 \cdot b_6, \quad \langle a_5 \cdot b_7 + a_7 \cdot b_5 \rangle, \\ c'_{11} &= \langle a_5 \cdot b_6 + a_6 \cdot b_5 + a_4 \cdot b_7 + a_7 \cdot b_4 \rangle, \\ &\langle a_4 \cdot b_6 + a_6 \cdot b_4 \rangle, \quad a_5 \cdot b_5, \quad \langle a_4 \cdot b_5 + a_5 \cdot b_4 \rangle, \quad a_4 \cdot b_4; \end{aligned} \quad (6.25)$$

$$\begin{aligned} \overline{00} : a_3 \cdot b_3, \quad \langle a_2 \cdot b_3 + a_3 \cdot b_2 \rangle, \quad a_2 \cdot b_2, \quad \langle a_1 \cdot b_3 + a_3 \cdot b_1 \rangle, \\ c'_3 &= \langle a_1 \cdot b_2 + a_2 \cdot b_1 + a_0 \cdot b_3 + a_3 \cdot b_0 \rangle, \\ &\langle a_0 \cdot b_2 + a_2 \cdot b_0 \rangle, \quad a_1 \cdot b_1, \quad c'_1 = \langle a_0 \cdot b_1 + a_1 \cdot b_0 \rangle, \quad c'_0 = a_0 \cdot b_0. \end{aligned} \quad (6.26)$$

O cálculo de $(A_1 + A_0)(B_1 + B_0) - A_1 B_1 - A_0 B_0$ também é análogo. O resultado final é

$$\begin{aligned} \overline{01} : \langle a_3 \cdot b_7 + a_7 \cdot b_3 \rangle, \quad \langle a_2 \cdot b_7 + a_7 \cdot b_2 + a_3 \cdot b_6 + a_6 \cdot b_3 \rangle, \\ \langle a_2 \cdot b_6 + a_6 \cdot b_2 \rangle, \quad \langle a_1 \cdot b_7 + a_7 \cdot b_1 + a_3 \cdot b_5 + a_5 \cdot b_3 \rangle, \\ c'_7 &= \langle a_1 \cdot b_6 + a_6 \cdot b_1 + a_2 \cdot b_5 + a_5 \cdot b_2 + a_0 \cdot b_7 + a_7 \cdot b_0 + a_3 \cdot b_4 + a_4 \cdot b_3 \rangle, \\ &\langle a_0 \cdot b_6 + a_6 \cdot b_0 + a_2 \cdot b_4 + a_4 \cdot b_2 \rangle, \quad \langle a_1 \cdot b_5 + a_5 \cdot b_1 \rangle, \\ &\langle a_0 \cdot b_5 + a_5 \cdot b_0 + a_1 \cdot b_4 + a_4 \cdot b_1 \rangle, \quad \langle a_0 \cdot b_4 + a_4 \cdot b_0 \rangle. \end{aligned} \quad (6.27)$$

Considera-se o termo $c'_7 = \langle a_1 \cdot b_6 + a_6 \cdot b_1 + a_2 \cdot b_5 + a_5 \cdot b_2 + a_0 \cdot b_7 + a_7 \cdot b_0 + a_3 \cdot b_4 + a_4 \cdot b_3 \rangle$. Os termos $a_1 \cdot b_6 + a_6 \cdot b_1$, $a_2 \cdot b_5 + a_5 \cdot b_2$ e $a_3 \cdot b_4 + a_4 \cdot b_3$ precisam ser separados de c'_7 porque os mesmos precisam ser adicionados a c'_5 , c'_3 e c'_1 , para que se calcule c_5 , c_3 e c_1 , respectivamente. De modo similar ao exemplo anterior, uma multiplicação e quatro adições são necessárias para calcular cada um desses termos. A partir do termo $\langle a_2 \cdot b_7 + a_7 \cdot b_2 + a_3 \cdot b_6 + a_6 \cdot b_3 \rangle$, o qual está associado a c'_9 , precisa-se separar $a_2 \cdot b_7 + a_7 \cdot b_2$ e $a_3 \cdot b_6 + a_6 \cdot b_3$, e adicioná-los respectivamente a c'_5 e c'_3 , para calcular c_5 e c_3 . O mesmo procedimento é aplicado a todos os termos que são previamente calculados juntos. Após isso, outras operações extras precisam ser contadas para adicionar os termos separados aos coeficientes c'_i e multiplicar por $1/2$. Isso resulta em $M_e(8) = 24$ e $A_e(8) = 71$.

Com os exemplos em mente, pode-se derivar uma fórmula para o número de operações necessárias para separar os termos produzidos juntos no algoritmo de Karatsuba. Inicia-se observando os termos intermediários produzidos pelo algoritmo, isto é, antes de obter o

Tabela 6.4: Status de todos os termos no algoritmo de Karatsuba até $N = 8$. O número de multiplicações $m(n)$ necessário para separar os termos calculados originalmente juntos com outros termos também é apresentado.

$N = 2^n$	$\overline{11}$	$\overline{01}$	$\overline{00}$	$m(n)$
1	-	0	-	0
2	0	1	0	0
4	0, 1, 0	1, 2, 1	0, 1, 0	1
8	$\underbrace{0, 1, 0, 1, 2, 1, 0, 1, 0}_{m(n)^{\overline{11}}}$	$\underbrace{1, 2, 1, 2, 4, 2, 1, 2, 1}_{m(n)^{\overline{01}}}$	$\underbrace{0, 1, 0, 1, 2, 1, 0, 1, 0}_{m(n)^{\overline{00}}}$	9

resultado final dos coeficientes $c'(x)$. Associa-se os termos na forma $a_i \cdot b_i$ a **0**, $\langle a_{i_1} \cdot b_{j_1} + a_{j_1} \cdot b_{i_1} \rangle$ a **1**, $\langle a_{i_1} \cdot b_{j_1} + a_{j_1} \cdot b_{i_1} + a_{i_2} \cdot b_{j_2} + a_{j_2} \cdot b_{i_2} \rangle$ a **2**, $\langle a_{i_1} \cdot b_{j_1} + a_{j_1} \cdot b_{i_1} + a_{i_2} \cdot b_{j_2} + a_{j_2} \cdot b_{i_2} + a_{i_3} \cdot b_{j_3} + a_{j_3} \cdot b_{i_3} + a_{i_4} \cdot b_{j_4} + a_{j_4} \cdot b_{i_4} \rangle$ a **4** etc. Em geral, um termo com a forma

$$\left\langle \sum_{k=1}^{2^t} (a_{i_k} \cdot b_{j_k} + a_{j_k} \cdot b_{i_k}) \right\rangle, \quad (6.28)$$

em que $t \in \mathbb{N}$ e $i_k + j_k$ é constante para $1 \leq k \leq 2^t$, é associado ao número ou *status* $\mathbf{s} = 2^t$. Considerando que todos os termos na expressão acima precisam ser separados, $\mathbf{s} - 1$ multiplicações extras são requeridas. Conseqüentemente, no máximo $4(\mathbf{s} - 1) + 1$ adições extras são necessárias. O limite superior é justificado pela possível presença de termos na forma $\langle a_0 \cdot b_i + a_i \cdot b_0 \rangle$, $i \neq 0$, produzido junto com outros termos. Eles não precisam ser separados e, nesses casos, uma adição é evitada (ver observação após o Exemplo 6.4).

Após aplicar o procedimento de separação explicado, todos os termos possuem status no máximo **1**, isto é, têm a forma $a_i \cdot b_i$ ou $\langle a_i \cdot b_j + a_j \cdot b_i \rangle$. Tais termos são adicionados aos coeficientes c'_i de acordo com a Equação (6.14) para obter os coeficientes c_i .

Para $N = 1$, tem-se apenas $a_0 \cdot b_0$, que tem status **0** e não representa qualquer operação extra. Como esse caso é uma espécie de “estado inicial”, o mesmo é associado a $\overline{01}$. Para $N = 2$, tem-se uma repetição do caso anterior nos termos associados a $\overline{11}$ e $\overline{00}$; isso é verificado nas Equações (6.18) e (6.19). O símbolo $\overline{01}$ é também uma repetição do caso anterior, mas com o status incrementado de **0** para **1**; isso se verifica na Equação (6.20). Devido à natureza recursiva do algoritmo, um fato análogo ocorre para $N = 4, 8, \dots$. Isso pode ser verificado nas Equações (6.21)–(6.23) e nas Equações (6.25)–(6.27). Isso permite construir a Tabela 6.4, onde se mostra os status de todos os termos no algoritmo de Karatsuba até $N = 8$. A última linha enfatiza que $m(n)$, o número de multiplicações necessárias para separar os termos que o algoritmo de Karatsuba computa conjuntamente, é obtido somando-se as contribuições dos termos associados a $\overline{11}$, $\overline{01}$ e $\overline{00}$. Essas contribuições são respectivamente denotadas por

$m(n)^{\overline{11}}$, $m(n)^{\overline{01}}$ e $m(n)^{\overline{00}}$.

Se $N = 4$, por exemplo, tem-se $m(n) = m(n)^{\overline{01}} = 1$ porque apenas o termo com status **2** associado a $\overline{01}$ requer separação (ver Tabela 6.4). Especificamente, esse termo corresponde a $\langle a_1 \cdot b_2 + a_2 \cdot b_1 + a_0 \cdot b_3 + a_3 \cdot b_0 \rangle$, apresentado no Exemplo 6.3. Se $N = 8$, tem-se $m(n)^{\overline{00}} = 1$ (um termo com status **2**), $m(n)^{\overline{01}} = 7$ (quatro termos com status **2** e um termo com status **4**) e $m(n)^{\overline{11}} = 1$ (um termo com status **2**). Esses termos podem ser observados nas Equações (6.25)–(6.27). Nesse caso, $m(n) = 1 + 7 + 1 = 9$.

Além disso, comparando as linhas para $N = 4$ ($n = 2$) e $N = 8$ ($n = 3$) na Tabela 6.4, nota-se que $m(3)^{\overline{11}} = m(3)^{\overline{00}} = m(2)$; $m(3)^{\overline{01}}$ é dado por $2m(2)^{\overline{01}}$ mais a contribuição dos termos relacionados a $m(2)^{\overline{01}}$, mas com status incrementados (multiplicados por 2). Devido à recursividade do algoritmo de Karatsuba, essa situação é geral, ou seja, $m(n)^{\overline{11}} = m(n)^{\overline{00}} = m(n-1)$ e $m(n)^{\overline{01}}$ é dado por $2m(n-1)^{\overline{01}}$ mais a contribuição dos termos associados a $m(n-1)^{\overline{01}}$ com status incrementados.

Prova do Teorema 6.2: Usando as notações e as observações anteriores, o número de multiplicações necessárias para separar os termos que o algoritmo de Karatsuba calcula juntos, $m(n)$, é dado por

$$m(n) = m(n)^{\overline{11}} + m(n)^{\overline{01}} + m(n)^{\overline{00}}. \quad (6.29)$$

Sabe-se que

$$m(n)^{\overline{11}} = m(n)^{\overline{00}} = m(n-1). \quad (6.30)$$

A partir dos comentários acima, $m(n)^{\overline{01}}$ é dado por $2m(n-1)^{\overline{01}}$ mais a contribuição dos termos relacionados a $m(n-1)^{\overline{01}}$ com status incrementados (multiplicados por 2). Um termo com status $s_1 = 2^n$, $n \geq 0$, contribui com $m_{s_1} = 2^n - 1$ multiplicações extras. Conseqüentemente, um termo com status $s_2 = 2s_1 = 2^{n+1}$ contribui com $m_{s_2} = 2^{n+1} - 1 = 2(2^n - 1) + 1 = 2m_{s_1} + 1$ multiplicações extras. Então, se um conjunto com t termos contribui com m_t multiplicações extras, um novo conjunto, obtido dobrando o status de cada termo no conjunto anterior, contribui com $2m_t + t$ multiplicações extras. Nota-se que há 3^{n-2} termos associados a $m(n-1)^{\overline{01}}$ (ver Tabela 6.4 para os casos $n = 1, 2, 3$). Portanto, dobrando o status de cada um desses termos, a nova contribuição é $2m(n-1)^{\overline{01}} + 3^{n-2}$. Isso permite escrever

$$m(n)^{\overline{01}} = 2m(n-1)^{\overline{01}} + 2m(n-1)^{\overline{01}} + 3^{n-2} = 4m(n-1)^{\overline{01}} + 3^{n-2}. \quad (6.31)$$

Também se nota que $m(n-1)^{\overline{01}} = m(n-1) - 2m(n-2)$. Assim, a equação 6.31 pode ser reescrita como

$$m(n)^{\overline{01}} = 4(m(n-1) - 2m(n-2)) + 3^{n-2}. \quad (6.32)$$

Substituindo as Equações (6.30) e (6.32) na Equação (6.29), tem-se

$$\begin{aligned} m(n) &= 2m(n-1) + 4(m(n-1) - 2m(n-2)) + 3^{n-2} \\ &= 6m(n-1) - 8m(n-2) + 3^{n-2}. \end{aligned} \quad (6.33)$$

A Equação (6.33) é uma relação de recorrência¹ e pode ser resolvida utilizando a transformada z . Denotando por $M(z)$ a transformada z de $m(n)$, a Equação (6.33) é reescrita no domínio da transformada z como

$$M(z) = 6M(z)z^{-1} - 8M(z)z^{-2} + \frac{z^{-2}}{1-3z^{-1}}.$$

Na última equação, agrupando os termos em $M(z)$, tem-se

$$M(z) = \frac{z^{-2}}{(1-6z^{-1}+8z^{-2})(1-3z^{-1})} = \frac{1/2}{1-4z^{-1}} + \frac{1/2}{1-2z^{-1}} - \frac{1}{1-3z^{-1}}. \quad (6.34)$$

Aplicando a transformada z inversa, obtém-se

$$m(n) = \frac{4^n + 2^n - 2 \cdot 3^n}{2}.$$

A equação acima pode ser escrita em função de N como

$$m(N) = \frac{N^2 + N - 2N^{\log_2 3}}{2}.$$

Adicionando a $m(N)$ as multiplicações devido ao fator de escala $1/2$, computa-se $M_e(N)$, o número total de multiplicações extras para calcular os coeficientes c_i a partir dos coeficientes c'_i , por

$$M_e(N) = \frac{N^2 + N - 2N^{\log_2 3}}{2} + 2N - 1 = \frac{N^2 - 2N^{\log_2 3} + 5N - 2}{2}.$$

As adições extras vêm de duas fontes. A primeira está relacionada ao procedimento de separação. Há quatro adições por produto e no máximo mais uma adição para cada termo com status ≥ 2 ; ver comentários imediatamente após a Equação (6.28). Dado n , o número total de termos produzidos no primeiro passo do algoritmo de Karatsuba é 3^n . Denotando

¹Curiosamente, essa relação de recorrência produz uma seqüência $m(n)$, $n = 0, 1, 2, \dots$, que coincide com o número de funções Booleanas monotônicas de n variáveis com 2 *mincuts*. Tal relação também representa o número de sistemas de Sperner com 2 blocos e algumas outras seqüências arquivadas pela “Enciclopédia On-line de Seqüências Inteiras” [102].

respectivamente por $S_0(n)$ and $S_1(n)$ o número de termos com status **0** e **1** para tal n , sabe-se que $S_{\geq 2}(n)$, o número de termos com status ≥ 2 , é dado por

$$S_{\geq 2}(n) = 3^n - S_0(n) - S_1(n). \quad (6.35)$$

Nota-se que $S_0(n) = 2^n$ e

$$S_1(n) = 2S_1(n-1) + S_0(n-1) = 2S_1(n-1) + 2^{n-1}.$$

Resolvendo a equação acima usando a transformada z , obtém-se $S_1(n) = n2^{n-1}$. Daí, a Equação (6.35) pode ser escrita como $S_{\geq 2}(n) = 3^n - 2^n - n2^{n-1}$ e, conseqüentemente,

$$S_{\geq 2}(N) = N^{\log_2 3} - N - \frac{N}{2} \log_2 N = N^{\log_2 3} - N \left(1 + \frac{\log_2 N}{2}\right).$$

Assim, o número de adições extras relacionado ao procedimento de separação é no máximo

$$\begin{aligned} & 4 \frac{N^2 + N - 2N^{\log_2 3}}{2} + N^{\log_2 3} - N \left(1 + \frac{\log_2 N}{2}\right) \\ & = 2N^2 - 3N^{\log_2 3} + N \left(1 - \frac{\log_2 N}{2}\right). \end{aligned} \quad (6.36)$$

A segunda fonte de adições extras está relacionada às operações necessárias para adicionar os termos $a_l \cdot b_l$, $l = 1, \dots, N-1$, e $a_l \cdot b_{l+i} + a_{l+i} \cdot b_l$, $i = 1, \dots, N-2$, $l = 1, \dots, N-1-i$, na Equação (6.14), o que fornece

$$N-1 + \sum_{i=1}^{N-2} (N-1-i) = \frac{N(N-1)}{2}. \quad (6.37)$$

Assim, somando as Equações (6.36) e (6.37), calcula-se $A_e(N)$, o número total de adições extras para calcular os coeficientes c_i a partir dos coeficientes c'_i . Obtém-se

$$\begin{aligned} A_e(N) & \leq 2N^2 - 3N^{\log_2 3} + N \left(1 - \frac{\log_2 N}{2}\right) + \frac{N(N-1)}{2} \\ & = \frac{5N^2 - 6N^{\log_2 3} + N(1 - \log_2 N)}{2}. \end{aligned}$$

■

Complexidade aritmética total

Usando o método de Karatsuba, a complexidade aritmética total para calcular os coeficientes de Chebyshev do produto de dois polinômios na forma de Chebyshev é dada pelo seguinte teorema.

Teorema 6.3 *Sejam $a(x)$ e $b(x)$ polinômios de graus $N - 1$ cujos coeficientes de Chebyshev a_i e b_i , $i = 0, \dots, N - 1$, são dados. Usando o método proposto, baseado no algoritmo de Karatsuba, os coeficientes de Chebyshev c_i , $i = 0, \dots, 2N - 2$, do polinômio $c(x) = a(x) \cdot b(x)$ são obtidos com*

$$M_k(N) = \frac{N^2 + 5N - 2}{2} \quad (6.38)$$

multiplicações e

$$A_k(N) \leq \frac{5N^2 + 6N^{\log_2 3} - N(15 + \log_2 N) + 4}{2} \quad (6.39)$$

adições.

A prova é imediata. As Equações (6.38) e (6.39) são obtidas adicionando o número de operações necessárias para calcular os coeficientes c'_i , apresentados na Seção 6.3.2, ao número de operações extras derivadas na última subseção.

Observa-se que a aplicação convencional do algoritmo de Karatsuba para multiplicar polinômios envolve $\mathcal{O}(N^{\log_2 3})$ operações aritméticas. Aqui, devido às operações extras, o método proposto possui o custo total de $\mathcal{O}(N^2)$.

6.3.3 Discussão

A partir das Equações (6.11), (6.12), (6.38) e (6.39), constrói-se a Tabela 6.5, na qual o número total de multiplicações e o de adições para multiplicar polinômios na forma de Chebyshev pelo método direto (resp. M_d e A_d) e pelo método de Karatsuba (resp. M_k e A_k) são apresentados. Todos os valores da Tabela 6.5 foram verificados através de uma simulação computacional implementada em MATLAB^R. O programa desenvolvido conta o número de operações aritméticas requeridas pelos métodos mencionados.

Embora o método direto e o de Karatsuba envolvam $\mathcal{O}(N^2)$ multiplicações, a divisão por 2 na Equação (6.38) representa uma considerável diferença. Avaliando assintoticamente a razão $M_k(N)/M_d(N)$, conclui-se que metade das multiplicações requeridas pelo método direto são economizadas quando se aplica o algoritmo de Karatsuba. Essa tendência é observada na Tabela 6.5.

Uma vez que a estratégia fundamental do algoritmo de Karatsuba é trocar multiplicações por adições, conforme esperado, o número de adições $A_k(N)$ é maior que $A_d(N)$. Mais precisamente, a razão $A_k(N)/A_d(N)$ aproxima-se de 5/3 à medida que N aumenta. Assim, uma comparação coerente entre o método direto e o proposto depende fortemente do custo com-

Tabela 6.5: Número total de multiplicações e adições para a multiplicação de polinômios na base de Chebyshev pelo método direto (resp. M_d e A_d) e pelo método de Karatsuba (resp. M_k e A_k).

$N = 2^n$	M_d	M_k	A_d	A_k
1	2	2	0	0
2	7	6	2	5
4	23	17	15	35
8	79	51	77	171
16	287	167	345	733
32	1087	591	1457	2971
64	4223	2207	5985	11757
128	16639	8511	24257	46115

putacional de uma multiplicação em termos de adições. Considerando que uma multiplicação custa r adições, a seguinte análise é realizada.

Com $N = 2^n$, o custo computacional total $T_d(N)$ para multiplicar dois polinômios de grau $N - 1$, na forma de Chebyshev, pelo método direto é medido por

$$T_d(N) = r M_d(N) + A_d(N).$$

O custo total $T_k(n)$ usando o método de Karatsuba é

$$T_k(N) = r M_k(N) + A_k(N).$$

Uma noção geral a respeito da razão $T_d(N)/T_k(N)$ é obtida pelo seguinte cálculo:

$$\lim_{N \rightarrow \infty} \left[\frac{T_d(N)}{T_k(N)} \right] = \lim_{N \rightarrow \infty} \left[\frac{r M_d(N) + A_d(N)}{r M_k(N) + A_k(N)} \right].$$

Para encontrar a faixa de valores de r para os quais o método de Karatsuba é mais eficiente que o direto, na equação acima, substitui-se as fórmulas previamente derivadas e se obtém

$$\frac{2r + 3}{r + 5} > 1 \therefore r > 2.$$

Daí, conclui-se que o método de Karatsuba é mais “econômico” que o método direto, caso uma multiplicação custe mais que duas adições. Na maioria das aplicações, essa condição é facilmente atendida [100].

Outra alternativa para realizar a operação descrita nesta seção é considerar os polinômios dados na forma de Chebyshev e expandi-los, reescrevendo-os na forma monomial. O produto seria, então, computado pela aplicação convencional do algoritmo de Karatsuba. O último

passo seria retornar o polinômio obtido à forma de Chebyshev. Nesse caso, a conversão dos polinômios da forma de Chebyshev para a forma monomial e vice-versa, além de requerer operações aritméticas extras, necessitaria, no caso de polinômios sobre os reais, uma cautela adicional acerca dos aspectos de precisão numérica.

Também é pertinente comparar o método proposto com aquele proposto em [99], onde a multiplicação polinomial na forma de Chebyshev é calculada no domínio da transformada discreta do co-seno (DCT). Nesse caso, o produto de dois polinômios de grau $N-1$ é realizado a partir do cálculo de DCT de comprimento $2N$. Embora os autores de [99] discutam apenas aspectos assintóticos da complexidade aritmética envolvida nesse método, é possível usar fórmulas gerais e obter, de forma mais precisa, o número de multiplicações e o de adições requeridos pelo método da DCT. Tais valores são respectivamente denotados por $M_{DCT}(N)$ e $A_{DCT}(N)$ e dados em [87] por

$$M_{DCT}(N) = 3N \log_2 2N - 4N + 3$$

e

$$A_{DCT}(N) = (9N + 3) \log_2 2N - 4N + 12.$$

Observando a Tabela 6.6, em que se compara o método da DCT e o de Karatsuba, nota-se que o primeiro usa menos operações aritméticas para $N \geq 32$. Para $N = 16$, uma comparação coerente depende do custo r de uma multiplicação em termos de adições. Uma vez que a implementação da DCT requer multiplicações por co-senos de arcos, restrições de precisão devem ser também consideradas. Por outro lado, o método de Karatsuba requer, além de produtos entre os coeficientes a_i e b_i , apenas produtos por $1/2$, o que torna o aspecto mencionado menos crítico. Daí, para $N < 16$, o que cobre diversas aplicações práticas que empregam expansões de Chebyshev, o uso do método de Karatsuba é menos complexo. Por exemplo, em [88], [89] e [90], expansões de Chebyshev com $4 \leq N \leq 6$, $5 \leq N \leq 13$ e $3 \leq N \leq 5$ são usadas, respectivamente. Para valores de N mais elevados, caso a precisão não seja um problema, a DCT constitui uma melhor proposta.

Embora não se tenha focalizado implementações em *hardware* do método proposto, uma observação relevante acerca desse ponto pode ser feita. Exceto por algumas multiplicações por $1/2$, todas as operações extras necessárias para calcular os coeficientes c_i a partir dos coeficientes c'_i podem ser implementadas em paralelo ao algoritmo de Karatsuba. Usando esse fato, a velocidade do método proposto pode ser consideravelmente aumentada.

Tabela 6.6: Número total de multiplicações e adições para multiplicar polinômios na base de Chebyshev pelo método da DCT (resp. M_{DCT} e A_{DCT}) e pelo método de Karatsuba (resp. M_k e A_k).

$N = 2^n$	M_{DCT}	M_k	A_{DCT}	A_k
1	2	2	12	0
2	7	6	30	5
4	23	17	81	35
8	67	51	216	171
16	179	167	555	733
32	451	591	1374	2971
64	1091	2207	3297	11757
128	2563	8511	7716	46115

CAPÍTULO 7

CONCLUSÕES

A INTRODUÇÃO das transformadas trigonométricas sobre corpos finitos, que serviu como ponto de partida para o desenvolvimento deste trabalho, representou o preenchimento de uma lacuna que existia na teoria das transformadas sobre corpos finitos, as chamadas transformadas digitais. Aliadas, particularmente, à transformada de Fourier, à de Hartley e à Wavelet sobre corpos finitos, as FFTT passam a fazer parte de um conjunto de ferramentas cujo potencial de aplicações tem-se revelado cada vez mais atrativo em diversas áreas da Engenharia. Para isso, conforme enfatizado em alguns trechos desse trabalho, contribui de forma decisiva a simplicidade das operações aritméticas requeridas na implementação de tais ferramentas e a ausência de erros de arredondamento.

O estudo das principais propriedades das FFTT atuou de maneira direcionadora, apontando cenários em que essas transformadas são candidatas em potencial a fazer parte de técnicas que proporcionem vantagem quando comparadas a procedimentos convencionais para realização de tarefas semelhantes. Devido à grande diversidade de tipos de FFTT, certamente, seria tedioso ao leitor verificar, para todos esses casos, o funcionamento das propriedades estudadas. No entanto, ao se investigar tais propriedades para os tipos de FFTT mais usuais, foi possível vislumbrar, inicialmente, duas interessantes aplicações.

Para a primeira delas, em que foi proposta uma técnica de marca d'água frágil no domínio da FFCT, pode-se concluir que o uso dessas transformadas nesse contexto constitui uma boa alternativa para localizar regiões em que uma imagem digital tenha sido alterada. Adicionalmente, conforme sugerido através de simulações, o método desenvolvido é robusto a alguns

tipos de manipulação, o que permite a extração da marca após alterações no brilho e destruições de regiões da imagem.

Na segunda aplicação estudada, foram realizados procedimentos básicos de filtragem de imagens digitais baseados na convolução simétrica das FFTT, como o de suavização e o de realce de linhas (equivalentes à filtragem passa-baixas e passa-altas). Nesse caso, destaca-se a imunidade a erros de arredondamento, o que é necessário, por exemplo, no processamento de imagens médicas, e que não se consegue quando se usam transformadas sobre os números reais. O emprego das FFTT é particularmente interessante quando se considera um cenário em que filtrações sucessivas precisam ser realizadas. Ainda assim, não há acúmulo de erro e, adicionalmente, o efeito de “blocagem” na imagem filtrada apresenta-se diminuído quando comparado aos resultados obtidos via transformadas de Fourier [57].

A investigação das auto-estruturas das FFTT também constituiu um importante avanço no conhecimento dessas transformadas. A respeito desse ponto, foram enunciadas diversas proposições e demonstrações que permitiram esclarecer a maioria dos detalhes que caracterizam os autovalores e os autovetores das matrizes de transformação em questão. Assim como em outros aspectos, concluiu-se que existe uma forte relação entre a auto-estrutura de cada tipo de FFTT e sua transformada trigonométrica discreta equivalente. De imediato, isso permitiu descrever o comportamento das FFTT na formatação de distribuições de probabilidade sobre os inteiros e sugerir formas de como essa operação pode ser incorporada a criptossistemas.

O conhecimento das auto-estruturas das FFTT também fundamentou uma técnica para separação cega de seqüências, o que pode ser considerada a mais relevante aplicação tratada nesse trabalho. Conforme se demonstrou, é possível associar a referida separação ao procedimento de recuperação de seqüências de usuários que interfiram de forma aditiva num canal de comunicação síncrono. Uma vez que tal procedimento possui importância básica em diversos sistemas de comunicação reais, além da abordagem predominantemente teórica realizada nessa tese, o estudo de detalhes práticos do mesmo é de grande interesse, devendo, como se comenta adiante, ser objeto de pesquisas futuras.

No Capítulo 6, introduziu-se, pela primeira vez na literatura, uma definição para polinômios de Chebyshev sobre $GF(p)$ baseada na trigonometria sobre corpos finitos. Como conclusão acerca desse ponto, pode-se afirmar que a definição proposta facilita o estudo das propriedades desses polinômios e, conseqüentemente, o de suas aplicações. Como exemplo disso, demonstrou-

se a segurança de criptosistemas em que se usa um algoritmo de cifragem de chave pública baseado nos polinômios mencionados. Finalmente, destaca-se o desenvolvimento de um algoritmo rápido para multiplicação de polinômios na forma de Chebyshev baseado no algoritmo de Karatsuba (aplicável tanto no contexto dos números reais quanto no de corpos finitos). Os resultados obtidos possibilitaram determinar as condições sob as quais o uso do algoritmo proposto é vantajoso, quando comparado ao de outras alternativas que desempenham o mesmo papel.

7.1 Trabalhos futuros

Um aspecto de grande importância desta tese é a quantidade de áreas de pesquisa contempladas pela mesma, o que pode ser enxergado como uma consequência da criação de ferramentas matemáticas nunca anteriormente exploradas. Naturalmente, a partir da consistência da teoria introduzida nos capítulos iniciais deste trabalho, inúmeras possibilidades surgiram, o que pode ser comprovado pela diversidade de tópicos abordados. Em função disso, para que os resultados obtidos sejam ainda mais refinados e para que novos resultados sejam alcançados, é fundamental que se dê continuidade à exploração desses tópicos e que novos temas relacionados à pesquisa desenvolvida sejam investigados. A seguir, diversos pontos para a realização de trabalhos futuros são colocados.

- ▷ **Transformadas trigonométricas sobre corpos de extensão:** O estudo das FFTT sobre $GF(q)$, onde q é a potência de um número primo, é de grande interesse, particularmente, para a área de códigos corretores de erro. Lidar com essas transformadas em corpos de extensão e , principalmente, buscar meios para definir FFTT sobre corpos de característica 2 pode permitir descrições alternativas de famílias importantes de códigos, revelando importantes propriedades das mesmas, bem como a concepção de novos algoritmo de decodificação.
- ▷ **Algoritmos rápidos para as FFTT:** Conforme se comentou em alguns pontos do texto, as matrizes de transformação das FFTT possuem os mesmos tipos de simetria que suas equivalentes transformadas trigonométricas discretas (DTT). Por conta disso, a maior parte dos algoritmos rápidos para o cálculo das DTT pode ser utilizada para calcular as FFTT. Entretanto, é provável que se possa desenvolver algoritmos aplicáveis apenas ao contexto de corpos finitos, a partir da obtenção de vantagens computacionais baseadas em particularidades dessas estruturas (além dos aspectos de precisão e aritmética de ponto fixo).

- ▷ **Marca d'água no domínio das FFTT:** Na literatura, existe uma grande quantidade de técnicas de marca d'água no domínio de transformadas. Em trabalhos futuros, é interessante que se realize uma comparação mais completa entre o esquema proposto nesta tese e métodos usuais. Além disso, poder-se-ia integrar algoritmos criptográficos à técnica proposta, o que traria benefícios relacionados à segurança e evitaria, na extração da marca, a necessidade de conhecimento explícito de informação acerca da imagem desmarcada.
- ▷ **FFTT e Criptografia:** Em trabalhos futuros, é importante utilizar as FFTT em criptosistemas específicos, explorando a sua capacidade de formatação de distribuições de probabilidade e mensurando as vantagens que as mesmas proporcionariam. Além disso, estudar a capacidade das FFTT na promoção de difusão em criptosistemas como o SAFER ou Idea [103] também é relevante.
- ▷ **Auto-estrutura das FFTT:** O problema que continua em aberto acerca deste ponto diz respeito à auto-estrutura das transformadas dos tipos 2 e 3. Uma demonstração para a conjectura de que a essas transformadas estão associadas matrizes cujos autovalores são todos distintos é de grande interesse. Outro tema relacionado a este ponto é a possibilidade de se introduzir transformadas fracionais sobre corpos finitos. As transformadas discretas fracionais (sobre os número reais), que são definidas a partir dos autovetores de suas respectivas transformadas discretas usuais, têm sido objeto de estudos recentes, particularmente, porque possibilitam análise tempo-frequência [72], [104], [71]. Aplicações dessas transformadas em processamento de imagem, Criptografia e diversas outras áreas têm sido propostas [105], [75]. A investigação de técnicas de marca d'água de chave pública baseadas nos autovetores das FFTT constitui outro assunto com potencial de desenvolvimento [106].
- ▷ **Sistemas de comunicação multiusuário:** A aplicação dos esquemas de separação de seqüências baseada na auto-estrutura das FFTT em sistemas multiusuário constitui um tópico de grande importância para investigações futuras. A respeito disso, enfatiza-se a necessidade de se adequar as idéias desenvolvidas a cenários em que considerações práticas sejam feitas. Isso inclui aspectos relacionados a energia máxima e capacidade de um canal de comunicação, falta de sincronismo no mesmo e, principalmente, desempenho na presença de ruído. A este último item, atrela-se a interpretação dos autovetores (ou da soma de autovetores) de uma transformada como palavras de um código e a avaliação das capacidades de detecção e de correção de erros do mesmo, o que tem sido feito por trabalhos que estão em desenvolvimento atualmente.

▷ **Polinômios de Chebyshev sobre corpos finitos:** A teoria dos polinômios de Chebyshev sobre os reais é muito bem estabelecida. Nesse sentido, o conteúdo inédito apresentado nesta tese representa um ponto inicial para o desenvolvimento mais consistente dessa ferramenta sobre corpos finitos. Polinômios de Chebyshev sobre $GF(q)$ foram estudados de maneira preliminar num trabalho que se encontra em fase de finalização. Definições de polinômios dos tipos 2, 3 e 4 sobre corpos finitos e a investigação de suas propriedades ainda estão em aberto. Em paralelo ao progresso teórico, possibilidades de aplicação em contextos diferentes da Criptografia devem surgir. Quanto a isso, há um interesse particular em estudar filtros de Chebyshev sobre corpos finitos.

REFERÊNCIAS

- [1] R. E. BLAHUT, Transform techniques for error-control codes, *IBM J. Res. Dev.*, v. 23, p. 299–315, Maio 1979.
- [2] I. S. REED & T. K. TRUONG, The use of finite fields to compute convolutions, *IEEE Trans. on Information Theory*, v. 21, n. 2, p. 208–213, Março 1975.
- [3] T. TOIVONEN & J. HEIKKILÄ, Video filtering with Fermat number theoretic transforms using residue number system, *IEEE Trans. Circuits Syst. Video Tech.*, v. 16, n. 1, p. 92–101, Janeiro 2006.
- [4] K. A. WAHID, V. S. DIMITROV, & G. A. JULLIEN, Error-free arithmetic for discrete wavelet transforms using algebraic integers, In: **Proc. 16th IEEE Symposium on Computer Arithmetic**, Santiago de Compostela, Espanha, 2003, p. 238.
- [5] C. PAAR, A new architecture for a parallel finite field multiplier with low complexity based on composite fields, *IEEE Transactions on Computers*, v. 45, n. 7, p. 856–861, Julho 1996.
- [6] S. BAKTIR & B. SUNAR, Finite field polynomial multiplication in the frequency domain with application to elliptic curve cryptography, In: **Proceedings of the 21th International Symposium on Computer and Information Sciences (ISCIS 2006)**, ser. Lecture Notes in Computer Science (LNCS), v. 4263. Heidelberg: Springer, Outubro 2006, p. 991–1001.
- [7] J. M. POLLARD, The fast Fourier transform in a finite field, *Math. Comput.*, v. 25, n. 114, p. 365–374, Abril 1971.
- [8] R. M. CAMPELLO DE SOUZA, H. M. DE OLIVEIRA, & A. N. KAUFFMAN, Trigonometry in finite fields and a new Hartley transform, In: **Proc. IEEE Int. Symp. Information Theory**, Boston, MA, 1998, p. 293.

- [9] W. SHU & Y. TIANREN, Algorithm for linear convolution using number theoretic transforms, *Electronics Letters*, v. 24, n. 5, p. 249–250, Março 1988.
- [10] W. LI & M. PETERSON, FIR filtering by the modified Fermat number theoretic transform, *IEEE Trans. on Acoustics, Speech and Signal Processing*, v. 38, n. 9, p. 1641–1645, Setembro 1990.
- [11] W. LI, The modified Fermat number transform and its application, In: **Proc. IEEE Int. Symp. Circuits Syst.**, v. 3, New Orleans, LA, 1990, p. 2365–2368.
- [12] V. S. DIMITROV, T. V. COOKLEV, & B. D. DONEVSKY, Number theoretic transforms over the golden section quadratic field, *IEEE Trans. on Signal Processing*, v. 43, n. 8, p. 1790–1797, Agosto 1995.
- [13] N. S. RUBANOV, E. I. BOVBEL, P. D. KUKHARCHIK, & V. J. BODROV, The modified number theoretic transform over the direct sum of finite fields to compute the linear convolution, *IEEE Trans. on Signal Processing*, v. 46, n. 3, p. 813–817, Março 1998.
- [14] S. GUDVANGEN, Practical applications of number theoretic transforms, Dezembro 2006. [Online]. Disponível: citeseer.ist.psu.edu/235814.html
- [15] R. E. BLAHUT, **Fast Algorithms for Digital Signal Processing**. Addison-Wesley Reading, 1985.
- [16] G. CODEVICO, G. HEINIG, & M. VAN BAREL, A superfast solver for real symmetric Toeplitz systems using real trigonometric transformations, *Numerical Linear Algebra with Applications*, v. 12, n. 8, p. 699–713, Outubro 2005.
- [17] L. R. RABINER & R. W. SCHAFER, **Digital Processing of Speech Signals**, ser. Signal Processing Series. Upper Saddle River, New Jersey: Prentice Hall, 1978.
- [18] R. M. CAMPELLO DE SOUZA, Transformadas em corpos finitos para codificação de canal, *Revista da Sociedade Brasileira de Telecomunicações*, v. 5, n. 1, p. 41–57, 1990.
- [19] H. M. DE OLIVEIRA, R. M. CAMPELLO DE SOUZA, & A. N. KAUFFMAN, Efficient multiplex for band-limited channels: Galois-field multiple access, In: **Proc. of the Workshop on Coding and Criptography**, Cambridge, United Kingdom, 1999, p. 235–241.
- [20] A. N. KAUFFMAN, A transformada de Hartley em um corpo finito e aplicações, Dissertação, Universidade Federal de Pernambuco, 1999.

- [21] R. M. CAMPELLO DE SOUZA & H. M. DE OLIVEIRA, The complex Hartley transform over a finite field, P. G. FARNELL, M. DARNELL, & B. HONARY, Eds., In: **Coding, Communications and Broadcasting**, 1^a ed. Hertfordshire: Research Studies Press, Jonh Wiley, 2000, p. 267–276.
- [22] H. M. DE OLIVEIRA & R. M. CAMPELLO DE SOUZA, Orthogonal multilevel spreading sequence design, P. G. FARNELL, M. DARNELL, & B. HONARY, Eds., In: **Coding, Communications and Broadcasting**, 1^a ed. Hertfordshire: Research Studies Press, Jonh Wiley, 2000, p. 291–303.
- [23] R. M. CAMPELLO DE SOUZA, H. M. DE OLIVEIRA, L. B. ESPÍNOLA, & M. M. C. DE SOUZA, Transformadas numéricas de Hartley, In: **Anais do XVIII Simpósio Brasileiro de Telecomunicações**, Gramado, Brasil, 2000, p. 357–366.
- [24] D. SILVA, R. M. DE CAMPELLO DE SOUZA, H. M. DE OLIVEIRA, L. B. ESPÍNOLA, & M. M. C. DE SOUZA, A transformada numérica de Hartley e grupos de inteiros gaussianos, *Revista da Sociedade Brasileira de Telecomunicações*, v. 17, n. 1, p. 48–57, Junho 2002.
- [25] F. FEKRI, R. M. MERSEREAU, & R. W. SCHAFFER, Theory of wavelet transforms over finite fields, In: **Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing**, v. 3, Phoenix, AZ, 1999, p. 1213–1216.
- [26] F. FEKRI, S. W. McLAUGHLIN, R. M. MERSEREAU, & R. W. SCHAFFER, Block error correcting codes using finite-field wavelet transforms, *IEEE Trans. on Signal Processing*, v. 54, n. 3, p. 991–1004, Março 2006.
- [27] K. S. CHAN & F. FEKRI, A block cipher cryptosystem using wavelet transforms over finite fields, *IEEE Trans. on Signal Processing*, v. 52, n. 10, p. 2975–2991, Outubro 2004.
- [28] F. FEKRI, M. W. SARTIPI, R. M. MERSEREAU, & R. W. SCHAFFER, Convolutional codes using finite-field wavelets: Time-varying codes and more, *IEEE Trans. on Signal Processing*, v. 53, n. 5, p. 1881–1896, Maio 2005.
- [29] S. A. MARTUCCI, Interpolation in the DST and DCT domains, In: **Proc. of the International Conference on Image Processing**, Vancouver, Canadá, 2000, p. 339–342.
- [30] M. A. SUHAIL & M. S. OBAIDAT, Digital watermarking-based DCT and JPEG model, *IEEE Trans. on Instrumentation and Measurement*, v. 52, n. 5, p. 1640–1647, Outubro 2003.

- [31] H. PARK, Y. PARK, & S. OH, L/M-fold image resizing in block-DCT domain using symmetric convolution, *IEEE Trans. on Image Processing*, v. 12, n. 9, p. 1016–1034, September 2003.
- [32] H. PARK & Y. PARK, Design and analysis of an image resizing filter in the block-dct domain, *IEEE Trans. on Circuits and Systems for Video Technology*, v. 14, n. 2, p. 1016–1034, February 2004.
- [33] S.-C. PEI & J.-J. DING, Generalized eigenvectors and fractionalization of offset DFTs and DCTs, *IEEE Transactions on Signal Processing*, v. 52, n. 7, p. 1661–1680, Julho 2004.
- [34] N. AL-DAHIR & H. MINN, A new multicarrier transceiver based on the discrete cosine transform, In: **Proc. Wireless Communications Networking Conf.**, v. 1, Richardson, TX, 2005, p. 45–50.
- [35] J. B. LIMA & R. M. CAMPELLO DE SOUZA, New trigonometric transforms over prime finite fields for image filtering, In: **Proc. of the International Telecommunications Symposium**, Fortaleza, Brasil, 2006.
- [36] J. B. LIMA & R. M. C. DE SOUZA, Uma marca d’água digital baseada na transformada do cosseno sobre corpos finitos, In: **Anais do XXII Simpósio Brasileiro de Telecomunicações**, Campinas, Brasil, 2005.
- [37] J. B. LIMA, R. M. CAMPELLO DE SOUZA, & D. PANARIO, Blind sequence separation based on the eigenstructure of finite field transforms, In: **Anais do XXVI Simpósio Brasileiro de Telecomunicações**, Rio de Janeiro, Brasil, 2008.
- [38] J. B. LIMA, H. M. DE OLIVEIRA, & R. M. CAMPELLO DE SOUZA, Formatação de distribuições de probabilidade sobre os inteiros, In: **Anais do XXV Simpósio Brasileiro de Telecomunicações**, Recife, Brasil, 2007.
- [39] J. B. LIMA, R. M. DE CAMPELLO DE SOUZA, & D. PANARIO, Security of public-cryptosystems based on Chebyshev polynomials over prime finite fields, In: **Proc. of the International Symposium on Information Theory**, Toronto, Canadá, 2008, p. 1843–1847.
- [40] J. B. LIMA, D. PANARIO, & Q. WANG, A Karatsuba-based algorithm for polynomial multiplication in Chebyshev form, Submetido, “IEEE Transactions on Computers”, 2008.

- [41] R. BRACEWELL, **The Fourier Transform and its Applications**, 3^a ed. New York: McGraw-Hill, 1999.
- [42] N. AHMED, T. NATARAJAN, & K. R. RAO, Discrete cosine transform, *IEEE Trans. Computers*, p. 90–93, Janeiro 1974.
- [43] K. R. RAO & P. YIP, **Discrete Cosine Transform: Algorithms, Advantages, Applications**. San Diego, CA: Academic, 1990.
- [44] A. V. OPPENHEIM, R. W. SCHAFER, & J. R. BUCK, **Discrete-Time Signal Processing**, 2^a ed. Prentice Hall, 1999.
- [45] W. PENNEBAKER & B. MITCHELL, **JPEG Still Image Data Compression Standard**. New York: Van Nostrand Reinhold, 1993.
- [46] T. SIKORA, MPEG digital video-coding standards, *IEEE Signal Processing Magazine*, v. 14, n. 5, p. 82–100, Setembro 1997.
- [47] M. M. C. DE SOUZA, H. M. DE OLIVEIRA, R. M. CAMPELLO DE SOUZA, & M. M. VASCONCELOS, The discrete cosine transform over prime finite fields, In: **International Conference on Telecommunications**, ser. Lecture Notes in Computer Science, J. N. DE SOUZA, P. DINI, & P. LORENZ, Eds. Berlin: Springer, 2004, p. 482–487.
- [48] R. M. C. DE SOUZA, H. M. DE OLIVEIRA, M. M. C. DE SOUZA, & M. M. VASCONCELOS, A transformada discreta do seno em um corpo finito, In: **Anais do XXVIII Congresso Nacional de Matemática Aplicada e Computacional**, São Paulo, Brasil, 2005.
- [49] D. M. BURTON, **Elementary Number Theory**. Addison-Wesley Publishing Company, 1994.
- [50] R. M. CAMPELLO DE SOUZA, H. M. DE OLIVEIRA, & D. SILVA, The Z transform over finite fields, In: **Proc. of the International Telecommunications Symposium**, Natal, Brasil, 2002.
- [51] R. M. CAMPELLO DE SOUZA, H. M. DE OLIVEIRA, L. B. ESPÍNOLA PALMA, & M. M. CAMPELLO DE SOUZA, Hartley number theoretic transforms, In: **Proc. IEEE Int. Symp. Information Theory**, Washington, DC, 2001, p. 210.
- [52] S. A. MARTUCCI, Symmetric convolution and the discrete sine and cosine transforms, *IEEE Trans. on Signal Processing*, v. 42, n. 5, p. 1038–1051, Maio 1994.

- [53] P. YIP & K. R. RAO, On the shift property of DCT's and DST's, *IEEE Trans. on Acoustics, Speech and Signal Processing*, v. 35, n. 3, p. 404–406, Março 1987.
- [54] L. WU, Comments on on the shift property of DCT's and DST's, *IEEE Trans. on Acoustics, Speech and Signal Processing*, v. 38, n. 1, p. 90–93, Janeiro 1990.
- [55] G. D. MANDYAM, Sinusoidal transforms in OFDM systems, *IEEE Trans. Broadcast.*, v. 50, n. 2, p. 172–184, Junho 2004.
- [56] R. ABOUCHAKRA & P. KABAL, Delay estimation for transform domain acoustical echo cancellation, In: **Proc. European Conf. Speech Commun. Technol.**, Budapeste, Hungria, 1999, p. 2539–2542.
- [57] S. A. MARTUCCI & R. M. MERSEREAU, New approaches to block filtering of images using symmetric convolution and the DST or DCT, In: **Proc. 1993 IEEE Int. Symp. Circuits Syst.**, Chicago, IL, 1993, p. 259–262.
- [58] H. W. PARK, Y. S. PARK, & S. K. OH, L/M-fold image resizing in block DCT domain using symmetric convolution, *IEEE Trans. on Image Processing*, v. 12, n. 9, p. 1016–1034, Setembro 2003.
- [59] F. HARTUNG & M. KUTTER, Multimedia watermarking techniques, *Proceedings of the IEEE*, v. 87, n. 7, p. 1079–1107, Julho 1999.
- [60] W. LUO, G. L. HEILEMAN, & C. E. PIZANO, Fast and robust watermarking of JPEG files, In: **Proc. of the Fifth IEEE Southwest Symposium on Image Analysis and Interpretation**, Santa Fe, NM, 2002, p. 158–162.
- [61] N. F. JOHNSON & S. JAJODIA, Exploring the steganography: Seeing the unseen, *IEEE Computer*, v. 31, n. 2, p. 26–34, Fevereiro 1998.
- [62] J. J. K. O RUANAIDH, W. J. DOWLING, & F. M. BOLAND, Watermarking digital image for copyright processing, *IEE Proc. Vis. Image Signal Processing*, v. 143, p. 250–256, Agosto 1996.
- [63] H. TAMORI, N. AOKI, & T. YAMAMOTO, A fragile digital watermarking technique by number theoretic transform, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, v. E85-A, n. 8, p. 1902–1904, Agosto 2002.

- [64] J. S. LIM, **Two dimensional signal and image processing**. Englewood Cliffs, NJ: Prentice Hall, 1990.
- [65] J. H. McCLELLAN & T. W. PARKS, Eigenvalue and eigenvector decomposition of the discrete fourier transform, *IEEE Trans. on Audio and Electroacoustics*, v. AU-20, n. 1, p. 66–74, Março 1972.
- [66] R. M. CAMPELLO DE SOUZA & H. M. DE OLIVEIRA, Eigensequences for multiuser communication over the real adder channel, In: **Proc. International Telecommunications Symposium**, Fortaleza, Brasil, 2006, p. 711–716.
- [67] C. CANDAN, M. A. KUTAY, & H. M. OZAKTAS, The discrete fractional Fourier transform, *IEEE Trans. on Signal Processing*, v. 48, n. 5, p. 1329–1337, Maio 2000.
- [68] C. VIJAYA & J. S. BHAT, Signal compression using discrete fractional Fourier transform and set partitioning in hierarchical tree, *Signal Processing*, v. 86, n. 8, p. 1976–1983, Agosto 2006.
- [69] L. BING-ZHAO, R. TAO, & Y. WANG, Interpolation of discrete chirp-periodic signals based on fractional Fourier transform, In: **Proc. Int. Conf. on Innovative Computing, Information and Control**, v. 3, Beijing, China, 2006, p. 2–5.
- [70] R. TAO, B. DENG, W.-Q. ZHANG, & Y. WANG, Sampling and sampling rate conversion of band limited signals in the fractional Fourier transform domain, *IEEE Trans. on Signal Processing*, v. 56, n. 1, p. 1329–1337, Janeiro 2008.
- [71] G. CARIOLARO, T. ERSEGHE, & P. KRANIAUSKAS, The fractional discrete cosine transform, *IEEE Transactions on Signal Processing*, v. 50, n. 4, p. 902–911, Abril 2002.
- [72] S.-C. PEI & M.-H. YEH, The discrete fractional cosine and sine transforms, *IEEE Transactions on Signal Processing*, v. 49, n. 6, p. 1198–1207, Junho 2001.
- [73] S.-C. PEI & J.-J. DING, Fractional cosine and sine transforms, *IEEE Transactions on Signal Processing*, v. 49, n. 7, p. 1661–1680, Julho 2001.
- [74] D. T. BIRTWISTLE, The eigenstructure of the number theoretic transforms, *Signal Processing*, v. 4, n. 4, p. 287–294, Julho 1982.

- [75] C.-C. TSENG, Eigenvalues and eigenvectors of generalized DFT, generalized DHT and DST-IV matrices, *IEEE Transactions on Signal Processing*, v. 50, n. 4, p. 866–877, Abril 2002.
- [76] M. R. SCHROEDER, **Number theory in science and communication**. New York, NY: Springer-Verlag, 1983.
- [77] J. KONVALINA & V. MATACHE, Palindrome-polynomials with roots on the unit circle, *C. R. Math. Acad. Sci. Soc. R. Canada*, v. 26, n. 2, p. 39–44, 2004.
- [78] J. VON ZUR GATHEN & D. PANARIO, Factoring polynomials over finite fields: a survey, *J. Symbolic Computation*, v. 31, n. 1-2, p. 3–17, Janeiro 2001.
- [79] G. Z. KARABULUT, D. PANARIO, & A. YONGAÇOGLU, Integer to integer Karhunen-Loève transform over finite fields, In: **Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Processing**, v. 5, Quebec, Canadá, 2004, p. 213–216.
- [80] G. CASELLA & R. BERGER, **Statistical Inference**. IE-Thomson, 2002.
- [81] H. N. JENDAL, T. J. B. KUHN, & J. L. MASSEY, An information-theoretic approach to homophonic substitution, In: **Advances in Cryptology-Eurocrypt'89**, ser. Lecture Notes in Computer Science, J.-J. QUISQUATER & J. VANDERWALLE, Eds. Berlin: Springer, 1990, p. 382–394.
- [82] V. C. DA ROCHA JR., D. P. B. A. CÂMARA, & C. J. L. PIMENTEL, Homophonic substitution and random number generation, In: **Proc. International Symposium on Communication Theory and Application**, Lancaster, Reino Unido, 2007, p. 1–6.
- [83] B. P. LATHI, **Modern Digital and Analog Communication Systems**, 3^a ed., ser. The Oxford Series in Electrical and Computer Engineering. New York, New York: Oxford University Press, 1998.
- [84] H. SCHULZE & C. LÜDERS, **Theory and Applications of OFDM and CDMA**, 2^a ed. Chichester, England: John Wiley and Sons, Ltd., 2005.
- [85] B. NAZER & M. GASTPAR, Computation over multiple-access channels, *IEEE Transactions on Information Theory*, v. 53, n. 10, p. 3498–3516, Outubro 2007.

- [86] S. RAY, M. MEDARD, & J. ABOUNADI, Random coding in noise-free multiple access networks over finite fields, In: **Proc. Global Telecommunications Conference**, v. 4, San Francisco, CA, 2003, p. 1898 – 1902.
- [87] S. C. CHAN & K. L. HO, Direct method for computing sinusoidal transforms, *IEEE Proceedings*, v. 137, n. 6, p. 433–442, 1990.
- [88] I. SARKAS, D. MAVRIDIS, M. PAPAMICHAIL, & G. PAPADOPOULOS, Volterra analysis using Chebyshev series, In: **Proc. IEEE Int. Symposium on Circuits and Systems (IS-CAS'2007)**, 2007, p. 1931–1934.
- [89] P. J. CHIANG, C. P. YU, & H. C. CHANG, Robust calculation of chromatic dispersion coefficients of optical fibers from numerically determined effective indices using Chebyshev-Lagrange interpolation polynomials, *Journal of Lightwave Technology*, v. 24, n. 11, p. 4411–4416, 2006.
- [90] A. ASHRAFI, R. ADHAMI, L. JOINER, & P. KAVEH, Arbitrary waveform DDFS utilizing Chebyshev polynomials interpolation, *IEEE Transactions on Circuits and Systems–I: Regular Papers*, v. 51, n. 8, p. 1468–1475, Agosto 2004.
- [91] G. CUYPERS, G. YSEBAERT, M. MOONEN, & F. PISONI, Chebyshev interpolation for DMT modems, In: **Proc. IEEE Int. Conference on Communications (ICC'2004)**, v. 5, 2004, p. 2736–2740.
- [92] J. C. MASON & D. C. HANDSCOMB, **Chebyshev Polynomials**, 1^a ed. Boca Raton, FL: Chapman & Hall/CRC, 2003.
- [93] G. H. RAWITSCHER & I. KOLTRACHT, An efficient numerical spectral method for solving the Schrodinger equation, *Computing in Science & Engineering*, v. 7, n. 6, p. 58–66, Nov.-Dez. 2005.
- [94] G. FEE & M. MONAGAN, Cryptography using Chebyshev polynomials, In: **Proc. 2004 Maple Summer Workshop (CD-ROM)**, Disponível: <http://www.cecm.sfu.ca/CAG/products2003.shtml>, 2004.
- [95] A. KARATSUBA & Y. OFMAN, Multiplication of many-digital numbers by automatic computers, *Doklady Akad. Nauk SSSR*, v. 145, p. 293–294, 1962. Tradução em *Physics-Doklady*, no. 7, pp. 595–596, 1963.

- [96] N. HONGZHOU, L. YUN, & H. DEQUAN, Public key encryption algorithm based on Chebyshev polynomials over finite fields, In: **Proc. 8th Int. Conference on Signal Processing**, 2006.
- [97] L. KOCAREV & Z. TASEV, Public-key encryption based on Chebyshev maps, In: **Proc. IEEE Int. Symp. Circuits and Systems (ISCAS'03)**, v. 3, Bangkok, Tailândia, 2003, p. 28–31.
- [98] P. BERGAMO, P. D'ARCO, A. DE SANTIS, & L. KOCAREV, Security of public-key cryptosystems based on Chebyshev polynomials, *IEEE Trans. Circuits and Systems–I: Regular Papers*, v. 52, n. 7, p. 1382–1393, Julho 2005.
- [99] G. BASZENSKI & M. TASCHE, Fast polynomial multiplication and convolutions related to the discrete cosine transform, *Linear Algebra Appl.*, v. 252, n. 1–3, p. 1–25, 1997.
- [100] J. VON ZUR GATHEN & J. GERHARD, **Modern Computer Algebra**, 2^a ed. Cambridge, United Kingdom: Cambridge University Press, 2003.
- [101] P. L. MONTGOMERY, Five, six, and seven-term Karatsuba-like formulae, *IEEE Transactions on Computers*, v. 54, n. 3, p. 362–369, Março 2005.
- [102] N. J. A. SLOANE, The on-line encyclopedia of integer sequences, disponível: <http://www.research.att.com/~njas/sequences/A016269>.
- [103] J. L. MASSEY, SAFER K-64: A byte oriented block-ciphering algorithm, In: **Proc. Cambridge Algorithms Workshop**, Cambridge, Reino Unido, 1993.
- [104] L. B. ALMEIDA, The fractional Fourier transform and time-frequency representations, *IEEE Transactions on Signal Processing*, v. 42, n. 11, p. 3084–3091, Novembro 1994.
- [105] S.-C. PEI & W.-L. HSUE, The multiple parameter discrete fractional Fourier transform, *IEEE Signal Processing Letters*, v. 13, n. 6, p. 329–332, Junho 2006.
- [106] R. VAN SCHYNDEL, A. TIRKEL, & I. SVALBE, Key independent watermark detection, In: **Proc. IEEE International Conference on Multimedia and Computing Systems**, v. 1, Florença, Itália, 1999, p. 580–585.

APÊNDICE **A**

PROVAS DOS TEOREMAS 2.1 A 2.16

NESTE apêndice, são apresentadas as demonstrações dos teoremas 2.1 a 2.16, que correspondem às fórmulas de inversão das transformadas trigonométricas de corpo finito.

A.1 Demonstração do Teorema 2.1 (FFCT-1 e^{-1})

Substituindo a Equação 2.5 na Equação 2.6 (onde se troca o índice i por r e c por \hat{c}), obtém-se a expressão

$$\hat{c}_r = \frac{1}{N} \sum_{k=0}^N \beta_k \sum_{i=0}^N 2\beta_i c_i \cos_{\zeta}(ki) \cos_{\zeta}(kr).$$

Trocando a posição dos somatórios e expandido o produto entre os dois co-senos, a equação acima pode ser reescrita como

$$\hat{c}_r = \frac{1}{N} \sum_{i=0}^N \beta_i c_i \sum_{k=0}^N \beta_k \{ \cos_{\zeta}(k(i+r)) + \cos_{\zeta}(k(i-r)) \}.$$

Retirando do segundo somatório os termos em que $k = 0$ e $k = N$, obtém-se

$$\hat{c}_r = \frac{1}{N} \sum_{i=0}^N \beta_i c_i \left\{ 1 + (-1)^{i+r} + \sum_{k=1}^{N-1} \cos_{\zeta}(k(i+r)) + \sum_{k=1}^{N-1} \cos_{\zeta}(k(i-r)) \right\}. \quad (\text{A.1})$$

A prova consiste em mostrar que, no primeiro somatório da expressão A.1, todos os termos se anulam, com exceção daquele em que $i = r$, o que faz com que $\hat{c}_r = c_r, \forall r$. Cada um

desses termos é calculado utilizando um dos dois casos a seguir, em que se recorre ao lema 2.2 para avaliar os somatórios sobre o índice k .

a) $i = r$: devido à presença da função peso β_i na Equação A.1, três subcasos são considerados separadamente.

- $i = 0$: no somatório sobre i , o termo correspondente a este subcaso resume-se a

$$\frac{c_0}{2}\{1 + 1 + N - 1 + N - 1\} = c_0N.$$

- $i = N$: o termo correspondente a este subcaso resume-se a

$$\frac{c_N}{2}\{1 + 1 + N - 1 + N - 1\} = c_NN.$$

- $i \neq 0, N$: o termo correspondente a este subcaso resume-se a

$$c_r\{1 + 1 - 1 + N - 1\} = c_rN.$$

b) $i \neq r$: neste caso, dois outros subcasos precisam ser considerados.

- Se os termos $(i + r)$ e $(i - r)$ forem ambos pares, no somatório sobre i , o termo correspondente a este subcaso resume-se a

$$\beta_i c_i \{1 + 1 - 1 - 1\} = 0.$$

- Se os termos $(i + r)$ e $(i - r)$ forem ambos ímpares, o termo correspondente a este subcaso resume-se a

$$\beta_i c_i \{1 - 1 + 0 + 0\} = 0.$$

Observando a presença do termo $1/N$ externo ao somatório sobre i na Equação A.1, e verificando o resultado obtido em cada subcaso considerado, conclui-se que $\hat{c}_r = c_r, \forall r$.

As demonstrações dos demais teoremas relacionados ao cálculo das FFTs inversas seguem passos semelhantes aos apresentados nesta primeira prova. Portanto, para tornar a leitura mais agradável, as demonstrações a seguir são descritas de maneira mais sucinta.

A.2 Demonstração do Teorema 2.2 (FFCT- $2e^{-1}$)

Substituindo a Equação 3.3 na Equação 2.8 (onde se troca o índice i por r e c por \hat{c}), obtém-se a expressão

$$\begin{aligned}\hat{c}_r &= \frac{1}{N} \sum_{k=0}^{N-1} \beta_k \sum_{i=0}^{N-1} 2c_i \cos_\zeta(k(i+1/2)) \cos_\zeta(k(r+1/2)) \\ &= \frac{1}{N} \sum_{i=0}^{N-1} c_i \sum_{k=0}^{N-1} \beta_k \{ \cos_\zeta(k(i+r+1)) + \cos_\zeta(k(i-r)) \}.\end{aligned}\quad (\text{A.2})$$

Retirando do segundo somatório o termo em que $k=0$, obtém-se:

$$\hat{c}_r = \frac{1}{N} \sum_{i=0}^{N-1} c_i \left\{ 1 + \sum_{k=1}^{N-1} \cos_\zeta(k(i+r+1)) + \sum_{k=1}^{N-1} \cos_\zeta(k(i-r)) \right\}.\quad (\text{A.3})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.2 para avaliar os somatórios sobre o índice k .

a) $i = r$:

$$c_r \{1 + 0 + N - 1\} = c_r N.$$

b) $i \neq r$: neste caso, se $(i-r)$ for par, então $(i+r+1)$ será ímpar, e vice-versa, logo

$$c_i \{1 + 0 - 1\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.3, conclui-se que $\hat{c}_r = c_r, \forall r$.

A.3 Demonstração do Teorema 2.3 (FFCT- $3e^{-1}$)

Substituindo a Equação 2.9 na Equação 2.10 (onde se troca o índice i por r e c por \hat{c}), obtém-se a expressão

$$\begin{aligned}\hat{c}_r &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{i=0}^{N-1} 2\beta_i c_i \cos_\zeta((k+1/2)i) \cos_\zeta(k(r+1/2)) \\ &= \frac{1}{N} \sum_{i=0}^{N-1} \beta_i c_i \sum_{k=0}^{N-1} \{ \cos_\zeta((k+1/2)(i+r)) + \cos_\zeta((k+1/2)(i-r)) \}.\end{aligned}\quad (\text{A.4})$$

Retirando do segundo somatório o termo em que $k=0$, obtém-se:

$$\begin{aligned}\hat{c}_r &= \frac{1}{N} \sum_{i=0}^{N-1} \beta_i c_i \left\{ \cos_\zeta(1/2(i+r)) + \cos_\zeta(1/2(i-r)) + \sum_{k=1}^{N-1} \cos_\zeta((k+1/2)(i+r)) \right. \\ &\quad \left. + \sum_{k=1}^{N-1} \cos_\zeta((k+1/2)(i-r)) \right\}.\end{aligned}\quad (\text{A.5})$$

Deve-se considerar os dois casos a seguir, onde se recorre ao lema 2.3 para avaliar os somat6rios sobre o 6ndice k .

a) $i = r$: devido 6 presen7a da fun77o peso β_i na Equa77o A.23, dois subcasos s7o considerados separadamente.

• $i = 0$:

$$\frac{c_0}{2} \{1 + 1 + N - 1 + N - 1\} = c_0 N.$$

• $i \neq 0$:

$$c_r \{\cos_\zeta(r) + 1 - \cos_\zeta(r) + N - 1\} = c_r N.$$

b) $i \neq r$:

$$\beta_i c_i \{\cos_\zeta(1/2(i + r)) + \cos_\zeta(1/2(i - r)) - \cos_\zeta(1/2(i + r)) - \cos_\zeta(1/2(i - r))\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equa77o A.23, conclui-se que $\hat{c}_r = c_r, \forall r$.

A.4 Demonstra77o do Teorema 2.4 (FFCT-4e⁻¹)

Substituindo a Equa77o 2.11 na Equa77o 2.12 (onde se troca o 6ndice i por r e c por \hat{c}), obt6m-se a express7o

$$\begin{aligned} \hat{c}_r &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{i=0}^{N-1} 2c_i \cos_\zeta((k + 1/2)(i + 1/2)) \cos_\zeta((k + 1/2)(r + 1/2)) \\ &= \frac{1}{N} \sum_{i=0}^{N-1} c_i \sum_{k=0}^{N-1} \{\cos_\zeta((k + 1/2)(i + r + 1)) + \cos_\zeta((k + 1/2)(i - r))\}. \end{aligned} \quad (\text{A.6})$$

Retirando do segundo somat6rio o termo em que $k = 0$, obt6m-se:

$$\begin{aligned} \hat{c}_r &= \frac{1}{N} \sum_{i=0}^{N-1} c_i \left\{ \cos_\zeta(1/2(i + r + 1)) + \cos_\zeta(1/2(i - r)) + \sum_{k=1}^{N-1} \cos_\zeta((k + 1/2)(i + r + 1)) \right. \\ &\quad \left. + \sum_{k=1}^{N-1} \cos_\zeta((k + 1/2)(i - r)) \right\}. \end{aligned} \quad (\text{A.7})$$

Deve-se considerar os dois casos a seguir, onde se recorre ao lema 2.3 para avaliar os somat6rios sobre o 6ndice k .

a) $i = r$:

$$c_r \{\cos_\zeta(1/2(2r + 1)) + 1 - \cos_\zeta(1/2(2r + 1)) + N - 1\} = c_r N.$$

b) $i \neq r$:

$$c_i \{ \cos_\zeta(1/2(i+r+1)) + \cos_\zeta(1/2(i-r)) - \cos_\zeta(1/2(i+r+1)) - \cos_\zeta(1/2(i-r)) \} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.7, conclui-se que $\hat{c}_r = c_r, \forall r$.

A.5 Demonstração do Teorema 2.5 (FFCT-1 0^{-1})

Substituindo a Equação 2.13 na Equação 2.14 (onde se troca o índice i por r e c por \hat{c}), obtém-se a expressão

$$\begin{aligned} \hat{c}_r &= \frac{2}{2N-1} \sum_{k=0}^{N-1} \beta_k \sum_{i=0}^{N-1} 2\beta_i c_i \cos_\zeta(ki) \cos_\zeta(kr) \\ &= \frac{2}{2N-1} \sum_{i=0}^{N-1} \beta_i c_i \sum_{k=0}^{N-1} \beta_k \{ \cos_\zeta(k(i+r)) + \cos_\zeta(k(i-r)) \}. \end{aligned} \quad (\text{A.8})$$

Retirando do segundo somatório os termos em que $k=0$ e $k=N$, obtém-se:

$$\hat{c}_r = \frac{2}{2N-1} \sum_{i=0}^{N-1} \beta_i c_i \left\{ 1 + \sum_{k=1}^{N-1} \cos_\zeta(k(i+r)) + \sum_{k=1}^{N-1} \cos_\zeta(k(i-r)) \right\}. \quad (\text{A.9})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.4 para avaliar os somatórios sobre o índice k .

a) $i = r$: devido à presença da função peso β_i na Equação A.9, dois subcasos são considerados separadamente.

• $i = 0$:

$$\frac{c_0}{2} \{ 1 + N - 1 + N - 1 \} = c_0(2N - 1)/2.$$

• $i \neq 0$:

$$c_r \{ 1 - 1/2 + N - 1 \} = c_r(2N - 1)/2.$$

b) $i \neq r$:

$$\beta_i c_i \{ 1 - 1/2 - 1/2 \} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.9, conclui-se que $\hat{c}_r = c_r, \forall r$.

A.6 Demonstração do Teorema 2.6 (FFCT-2 o^{-1})

Substituindo a Equação 2.15 na Equação 2.16 (onde se troca o índice i por r e c por \hat{c}), obtém-se a expressão

$$\begin{aligned}\hat{c}_r &= \frac{2}{2N-1} \sum_{k=0}^{N-1} \beta_k \sum_{i=0}^{N-1} 2\gamma_i c_i \cos_\zeta(k(i+1/2)) \cos_\zeta(k(r+1/2)) \\ &= \frac{2}{2N-1} \sum_{i=0}^{N-1} \gamma_i c_i \sum_{k=0}^{N-1} \beta_k \{ \cos_\zeta(k(i+r+1)) + \cos_\zeta(k(i-r)) \}.\end{aligned}\quad (\text{A.10})$$

Retirando do segundo somatório o termo em que $k=0$, obtém-se

$$\hat{c}_r = \frac{2}{2N-1} \sum_{i=0}^{N-1} \gamma_i c_i \left\{ 1 + \sum_{k=1}^{N-1} \cos_\zeta(k(i+r+1)) + \sum_{k=1}^{N-1} \cos_\zeta(k(i-r)) \right\}.\quad (\text{A.11})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.4 para avaliar os somatórios sobre o índice k .

a) $i=r$: devido à presença da função peso γ_i na Equação A.11, dois subcasos são considerados separadamente.

- $i=N-1$:

$$\frac{c_{N-1}}{2} \{1 + N - 1 + N - 1\} = c_{N-1}(2N-1)/2.$$

- $i \neq N-1$:

$$c_r \{1 - 1/2 + N - 1\} = c_r(2N-1)/2.$$

b) $i \neq r$:

$$\gamma_i c_i \{1 - 1/2 - 1/2\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.11, conclui-se que $\hat{c}_r = c_r, \forall r$.

A.7 Demonstração do Teorema 2.7 (FFCT-3 o^{-1})

Substituindo a Equação 2.17 na Equação 2.18 (onde se troca o índice i por r e c por \hat{c}), obtém-se a expressão

$$\begin{aligned}\hat{c}_r &= \frac{2}{2N-1} \sum_{k=0}^{N-1} \gamma_k \sum_{i=0}^{N-1} 2\beta_i c_i \cos_\zeta((k+1/2)i) \cos_\zeta((k+1/2)r) \\ &= \frac{2}{2N-1} \sum_{i=0}^{N-1} \beta_i c_i \sum_{k=0}^{N-1} \gamma_k \{ \cos_\zeta((k+1/2)(i+r)) + \cos_\zeta((k+1/2)(i-r)) \}.\end{aligned}\quad (\text{A.12})$$

Retirando do segundo somatório os termos em que $k = 0$ e $k = N - 1$, obtém-se

$$\hat{c}_r = \frac{2}{2N-1} \sum_{i=0}^{N-1} \beta_i c_i \left\{ (-1)^{i+r} + \cos_\zeta(1/2(i+r)) + \cos_\zeta(1/2(i-r)) \right. \\ \left. + \sum_{k=1}^{N-2} \cos_\zeta((k+1/2)(i+r)) + \sum_{k=1}^{N-2} \cos_\zeta((k+1/2)(i-r)) \right\}. \quad (\text{A.13})$$

Deve-se considerar os dois casos a seguir, onde se recorre ao lema 2.5 para avaliar os somatórios sobre o índice k .

a) $i = r$: devido à presença da função peso β_i na Equação A.13, dois subcasos são considerados separadamente.

- $i = 0$:

$$\frac{c_0}{2} \{1 + 1 + 1/2 + 1/2 + N - 2 + N - 2\} = c_0(2N - 1)/2.$$

- $i \neq 0$:

$$c_r \{ \cos_\zeta(r) + 1 + 1/2 + 1/2 - 1/2 - \cos_\zeta(r) + N - 2 \} = c_r(2N - 1)/2.$$

b) $i \neq r$: neste caso, dois outros subcasos precisam ser considerados.

- Se os termos $(i+r)$ e $(i-r)$ forem ambos pares, no somatório sobre i , o termo correspondente a este subcaso resume-se a

$$\beta_i c_i \{ \cos_\zeta(1/2(i+r)) + \cos_\zeta(1/2(i-r)) + 1 - 1/2 - \cos_\zeta(1/2(i+r)) - 1/2 \\ - \cos_\zeta(1/2(i-r)) \} = 0. \quad (\text{A.14})$$

- Se os termos $(i+r)$ e $(i-r)$ forem ambos ímpares, o termo correspondente a este subcaso resume-se a

$$\beta_i c_i \{ \cos_\zeta(1/2(i+r)) + \cos_\zeta(1/2(i-r)) - 1 + 1/2 - \cos_\zeta(1/2(i+r)) + 1/2 \\ - \cos_\zeta(1/2(i-r)) \} = 0. \quad (\text{A.15})$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.13, conclui-se que $\hat{c}_r = c_r, \forall r$.

A.8 Demonstração do Teorema 2.8 (FFCT- $4o^{-1}$)

Substituindo a Equação 2.19 na Equação 2.20 (onde se troca o índice i por r e c por \hat{c}), obtém-se a expressão

$$\begin{aligned}\hat{c}_r &= \frac{2}{2N-1} \sum_{k=0}^{N-2} \sum_{i=0}^{N-2} 2c_i \cos_\zeta((k+1/2)(i+1/2)) \cos_\zeta((k+1/2)(r+1/2)) \\ &= \frac{2}{2N-1} \sum_{i=0}^{N-2} c_i \sum_{k=0}^{N-2} \{ \cos_\zeta((k+1/2)(i+r+1)) + \cos_\zeta((k+1/2)(i-r)) \}.\end{aligned}\quad (\text{A.16})$$

Retirando do segundo somatório o termo em que $k=0$, obtém-se

$$\begin{aligned}\hat{c}_r &= \frac{2}{2N-1} \sum_{i=0}^{N-1} c_i \left\{ \cos_\zeta(1/2(i+r+1)) + \cos_\zeta(1/2(i-r)) + \sum_{k=1}^{N-2} \cos_\zeta((k+1/2)(i+r+1)) \right. \\ &\quad \left. + \sum_{k=1}^{N-2} \cos_\zeta((k+1/2)(i-r)) \right\}.\end{aligned}\quad (\text{A.17})$$

Deve-se considerar os dois casos a seguir, onde se recorre ao lema 2.5 para avaliar os somatórios sobre o índice k .

a) $i=r$:

$$c_r \{ \cos_\zeta(1/2(2r+1)) + 1 + 1/2 - \cos_\zeta(1/2(2r+1)) + N - 2 \} = c_r(2N-1)/2.$$

b) $i \neq r$:

$$\begin{aligned}c_i \{ \cos_\zeta(1/2(i+r+1)) + \cos_\zeta(1/2(i-r)) - 1/2 - \cos_\zeta(1/2(i+r+1)) + 1/2 \\ - \cos_\zeta(1/2(i-r)) \} = 0.\end{aligned}\quad (\text{A.18})$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.17, conclui-se que $\hat{c}_r = c_r, \forall r$.

A.9 Demonstração do Teorema 2.9 (FFST- $1e^{-1}$)

Substituindo a Equação 2.21 na Equação 2.22 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned}\hat{s}_r &= \frac{1}{N} \sum_{k=1}^{N-1} \sum_{i=1}^{N-1} 2s_i \sin_\zeta(ki) \sin_\zeta(kr) \\ &= \frac{1}{N} \sum_{i=1}^{N-1} s_i \left\{ \sum_{k=1}^{N-1} \cos_\zeta(k(i-r)) - \sum_{k=1}^{N-1} \cos_\zeta(k(i+r)) \right\}.\end{aligned}\quad (\text{A.19})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.2 para avaliar os somatórios sobre o índice k .

a) $i = r$:

$$s_r\{N - 1 + 1\} = s_r N.$$

b) $i \neq r$: neste caso, os termos $(i - r)$ e $(i + r)$ serão ambos pares ou ímpares, logo

$$s_i\{1 - 1\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.19, conclui-se que $\hat{s}_r = s_r, \forall r$.

A.10 Demonstração do Teorema 2.10 (FFST- $2e^{-1}$)

Substituindo a Equação 2.23 na Equação 2.24 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned} \hat{s}_r &= \frac{1}{N} \sum_{k=1}^N \beta_k \sum_{i=0}^{N-1} 2s_i \sin_\zeta(k(i + 1/2)) \sin_\zeta(k(r + 1/2)) \\ &= \frac{1}{N} \sum_{i=0}^{N-1} s_i \sum_{k=1}^N \beta_k \{ \cos_\zeta(k(i - r)) - \cos_\zeta(k(i + r + 1)) \}. \end{aligned} \quad (\text{A.20})$$

Retirando do segundo somatório o termo em que $k = N$ (que é nulo), obtém-se

$$\hat{s}_r = \frac{1}{N} \sum_{i=0}^{N-1} s_i \left\{ (-1)^{i-r} + \sum_{k=1}^{N-1} \cos_\zeta(k(i - r)) - \sum_{k=1}^{N-1} \cos_\zeta(k(i + r + 1)) \right\}. \quad (\text{A.21})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.2 para avaliar os somatórios sobre o índice k .

a) $i = r$:

$$s_r\{1 + N - 1 - 0\} = s_r N.$$

b) $i \neq r$: neste caso, se $(i - r)$ for par, então $(i + r + 1)$ será ímpar, e vice-versa, logo

$$s_i\{1 + 0 - 1\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.21, conclui-se que $\hat{s}_r = s_r, \forall r$.

A.11 Demonstração do Teorema 2.11 (FFST-3e⁻¹)

Substituindo a Equação 2.25 na Equação 2.26 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned}\hat{s}_r &= \frac{1}{N} \sum_{k=0}^{N-1} \sum_{i=1}^N 2\beta_i s_i \sin_\zeta((k+1/2)i) \sin_\zeta((k+1/2)(r+1/2)) \\ &= \frac{1}{N} \sum_{i=1}^N \beta_i s_i \sum_{k=0}^{N-1} \{\cos_\zeta((k+1/2)(i-r)) - \cos_\zeta((k+1/2)(i+r))\}.\end{aligned}\quad (\text{A.22})$$

Retirando do segundo somatório o termo em que $k=0$, obtém-se

$$\begin{aligned}\hat{s}_r &= \frac{1}{N} \sum_{i=1}^N \beta_i s_i \left\{ \cos_\zeta(1/2(i-r)) - \cos_\zeta(1/2(i+r)) + \sum_{k=1}^{N-1} \cos_\zeta((k+1/2)(i-r)) \right. \\ &\quad \left. - \sum_{k=1}^{N-1} \cos_\zeta((k+1/2)(i+r)) \right\}.\end{aligned}\quad (\text{A.23})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.3 para avaliar os somatórios sobre o índice k .

a) $i=r$: devido à presença da função peso β_i na Equação A.23, dois subcasos são considerados separadamente.

- $i=N$:

$$\frac{s_N}{2} \{1 + 1 + N - 1 + N - 1\} = s_N N.$$

- $i \neq N$:

$$s_r \{1 - \cos_\zeta(r) + N - 1 + \cos_\zeta(r)\} = s_r N.$$

b) $i \neq r$:

$$\beta_i s_i \{\cos_\zeta(1/2(i-r)) - \cos_\zeta(1/2(i+r)) - \cos_\zeta(1/2(i-r)) + \cos_\zeta(1/2(i+r))\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.23, conclui-se que $\hat{s}_r = s_r, \forall r$.

A.12 Demonstração do Teorema 2.12 (FFST-4e⁻¹)

Substituindo a Equação 2.27 na Equação 2.28 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned}\hat{s}_r &= \frac{1}{N} \sum_{k=1}^{N-1} \sum_{i=1}^{N-1} 2s_i \cos_{\zeta}((k+1/2)(i+1/2)) \cos_{\zeta}((k+1/2)(r+1/2)) \\ &= \frac{1}{N} \sum_{i=0}^{N-1} s_i \sum_{k=0}^{N-1} \{\cos_{\zeta}((k+1/2)(i-r)) - \cos_{\zeta}((k+1/2)(i+r+1))\}.\end{aligned}\quad (\text{A.24})$$

Retirando do segundo somatório o termo em que $k=0$, obtém-se

$$\begin{aligned}\hat{s}_r &= \frac{1}{N} \sum_{i=0}^{N-1} s_i \left\{ \cos_{\zeta}(1/2(i-r)) - \cos_{\zeta}(1/2(i+r+1)) + \sum_{k=1}^{N-1} \cos_{\zeta}((k+1/2)(i-r)) \right. \\ &\quad \left. - \sum_{k=1}^{N-1} \cos_{\zeta}((k+1/2)(i+r+1)) \right\}.\end{aligned}\quad (\text{A.25})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.3 para avaliar os somatórios sobre o índice k .

a) $i=r$:

$$s_r \{1 - \cos_{\zeta}(1/2(2r+1)) + N - 1 + \cos_{\zeta}(1/2(2r+1))\} = s_r N.$$

b) $i \neq r$:

$$s_i \{\cos_{\zeta}(1/2(i-r)) - \cos_{\zeta}(1/2(i+r+1)) - \cos_{\zeta}(1/2(i-r)) + \cos_{\zeta}(1/2(i+r+1))\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.25, conclui-se que $\hat{s}_r = s_r, \forall r$.

A.13 Demonstração do Teorema 2.13 (FFST-1o⁻¹)

Substituindo a Equação 2.29 na Equação 2.30 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned}\hat{s}_r &= \frac{2}{2N-1} \sum_{k=1}^{N-1} \sum_{i=1}^{N-1} 2s_i \sin_{\zeta}(ki) \sin_{\zeta}(kr) \\ &= \frac{2}{2N-1} \sum_{i=1}^{N-1} s_i \left\{ \sum_{k=1}^{N-1} \cos_{\zeta}(k(i-r)) - \sum_{k=1}^{N-1} \cos_{\zeta}(k(i+r)) \right\}.\end{aligned}\quad (\text{A.26})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.4 para avaliar os somatórios sobre o índice k .

a) $i = r$:

$$s_r\{N - 1 + 1/2\} = s_r(2N - 1)/2.$$

b) $i \neq r$:

$$s_i\{-1/2 + 1/2\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.26, conclui-se que $\hat{s}_r = s_r, \forall r$.

A.14 Demonstração do Teorema 2.14 (FFST-2 o^{-1})

Substituindo a Equação 2.31 na Equação 2.32 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned} \hat{s}_r &= \frac{2}{2N - 1} \sum_{k=1}^{N-1} \sum_{i=0}^{N-2} 2s_i \sin_{\zeta}(k(i + 1/2)) \sin_{\zeta}(k(r + 1/2)) \\ &= \frac{2}{2N - 1} \sum_{i=0}^{N-2} s_i \sum_{k=1}^{N-1} \{\cos_{\zeta}(k(i - r)) - \cos_{\zeta}(k(i + r + 1))\}. \end{aligned} \quad (\text{A.27})$$

De maneira análoga à demonstração A.1, considera-se os dois casos a seguir, onde se recorre ao lema 2.4 para avaliar os somatórios sobre o índice k .

a) $i = r$:

$$s_r\{N - 1 + 1/2\} = s_r(2N - 1)/2.$$

b) $i \neq r$:

$$s_i\{-1/2 + 1/2\} = 0.$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.27, conclui-se que $\hat{s}_r = s_r, \forall r$.

A.15 Demonstração do Teorema 2.15 (FFST-3 o^{-1})

Substituindo a Equação 2.33 na Equação 2.34 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned} \hat{s}_r &= \frac{2}{2N - 1} \sum_{k=0}^{N-2} \sum_{i=1}^{N-1} 2s_i \sin_{\zeta}((k + 1/2)i) \sin_{\zeta}((k + 1/2)r) \\ &= \frac{2}{2N - 1} \sum_{i=1}^{N-1} s_i \sum_{k=0}^{N-2} \{\cos_{\zeta}((k + 1/2)(i - r)) - \cos_{\zeta}((k + 1/2)(i + r))\}. \end{aligned} \quad (\text{A.28})$$

Retirando do segundo somatório o termo em que $k = 0$, obtém-se

$$\hat{s}_r = \frac{2}{2N-1} \sum_{i=1}^{N-1} s_i \left\{ \cos_{\zeta}(1/2(i-r)) - \cos_{\zeta}(1/2(i+r)) + \sum_{k=1}^{N-2} \cos_{\zeta}((k+1/2)(i-r)) - \sum_{k=1}^{N-2} \cos_{\zeta}((k+1/2)(i+r)) \right\}. \quad (\text{A.29})$$

Deve-se considerar os dois casos a seguir, onde se recorre ao lema 2.5 para avaliar os somatórios sobre o índice k .

a) $i = r$:

$$s_r \{1 - \cos_{\zeta}(r) + N - 2 + 1/2 + \cos_{\zeta}(r)\} = s_r(2N - 1)/2.$$

b) $i \neq r$: neste caso, dois outros subcasos precisam ser considerados.

- Se os termos $(i+r)$ e $(i-r)$ forem ambos pares, no somatório sobre i , o termo correspondente a este subcaso resume-se a

$$s_i \{ \cos_{\zeta}(1/2(i-r)) - \cos_{\zeta}(1/2(i+r)) - 1/2 - \cos_{\zeta}(1/2(i-r)) + 1/2 + \cos_{\zeta}(1/2(i+r)) \} = 0. \quad (\text{A.30})$$

- Se os termos $(i+r)$ e $(i-r)$ forem ambos ímpares, o termo correspondente a este subcaso resume-se a

$$s_i \{ \cos_{\zeta}(1/2(i-r)) - \cos_{\zeta}(1/2(i+r)) + 1/2 - \cos_{\zeta}(1/2(i-r)) - 1/2 + \cos_{\zeta}(1/2(i+r)) \} = 0. \quad (\text{A.31})$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.29, conclui-se que $\hat{s}_r = s_r, \forall r$.

A.16 Demonstração do Teorema 2.16 (FFST-40⁻¹)

Substituindo a Equação 2.35 na Equação 2.36 (onde se troca o índice i por r e s por \hat{s}), obtém-se a expressão

$$\begin{aligned} \hat{s}_r &= \frac{2}{2N-1} \sum_{k=0}^{N-1} \gamma_k \sum_{i=0}^{N-1} 2\gamma_i s_i \sin_{\zeta}((k+1/2)(i+1/2)) \sin_{\zeta}((k+1/2)(r+1/2)) \\ &= \frac{2}{2N-1} \sum_{i=0}^{N-1} \gamma_i s_i \sum_{k=0}^{N-1} \gamma_k \{ \cos_{\zeta}((k+1/2)(i-r)) - \cos_{\zeta}((k+1/2)(i+r+1)) \}. \quad (\text{A.32}) \end{aligned}$$

Retirando do segundo somatório os termos em que $k = 0$ e $k = N - 1$, obtém-se

$$\hat{s}_r = \frac{2}{2N-1} \sum_{i=0}^{N-1} \gamma_i s_i \left\{ (-1)^{i+r} + \cos_\zeta(1/2(i-r)) - \cos_\zeta(1/2(i+r+1)) \right. \\ \left. + \sum_{k=1}^{N-2} \cos_\zeta((k+1/2)(i-r)) - \sum_{k=1}^{N-2} \cos_\zeta((k+1/2)(i+r+1)) \right\}. \quad (\text{A.33})$$

Deve-se considerar os dois casos a seguir, onde se recorre ao lema 2.5 para avaliar os somatórios sobre o índice k .

a) $i = r$: devido à presença da função peso γ_i na Equação A.33, dois subcasos são considerados separadamente.

- $i = N - 1$: no somatório sobre i , o termo correspondente a este subcaso resume-se a

$$\frac{s_{N-1}}{2} \{1 + 1 + 1 + N - 2 + N - 2\} = s_{N-1}(2N - 1)/2.$$

- $i \neq N - 1$: o termo correspondente a este subcaso resume-se a

$$s_r \{1 + 1 - \cos_\zeta(1/2(2r + 1)) + N - 2 - 1/2 + \cos_\zeta(1/2(2r + 1))\} = s_r(2N - 1)/2.$$

b) $i \neq r$: neste caso, se $(i - r)$ for par, então $(i + r + 1)$ será ímpar, e vice-versa, logo

$$\gamma_i s_i \{(-1)^{i+r} + \cos_\zeta(1/2(i-r)) - \cos_\zeta(1/2(i+r+1)) + (-1)^{i+r+1} - \cos_\zeta(1/2(i-r)) \\ + \cos_\zeta(1/2(i+r+1))\} = 0. \quad (\text{A.34})$$

Verificando o resultado obtido em cada caso considerado, com base na Equação A.33, conclui-se que $\hat{s}_r = s_r, \forall r$.

APÊNDICE **B**

PROVAS DAS PROPOSIÇÕES 4.4 E 4.9

NESTE apêndice são apresentadas as demonstrações das Proposições 4.4 e 4.9, que fazem parte do estudo da autoestrutura das transformadas trigonométricas sobre corpos finitos.

B.1 Demonstração da Proposição 4.4

No que segue, todos os cálculos são realizados módulo p , o número primo que caracteriza o corpo finito sobre o qual as transformadas são implementadas. Conforme enunciado na Proposição 4.4, considera-se o autovetor \mathbf{x} da matriz \mathbf{FF}_{2N-2} associado ao autovalor $\lambda = 1$ ou $\lambda = -1$. Equivalentemente, tem-se

$$\mathbf{FF}_{2N-2} \cdot \mathbf{x}^T = \lambda \cdot \mathbf{x}^T.$$

Usando a Equação (4.2), que define a matriz de transformação da FFT, reescreve-se a última equação da seguinte forma:

$$\sqrt{(2N-2)^{-1}} \cdot \begin{bmatrix} 1 & 1 & \cdots & \cdots & 1 \\ 1 & \alpha^1 & \cdots & \cdots & \alpha^{2N-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{N-2} & \cdots & \cdots & \alpha^{(2N-3) \cdot (N-2)} \\ 1 & \alpha^{N-1} & \cdots & \cdots & \alpha^{(2N-3) \cdot (N-1)} \\ 1 & \alpha^{N-2} & \cdots & \cdots & \alpha^{(2N-3) \cdot (N-2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{2N-3} & \cdots & \cdots & \alpha^{(2N-3)^2} \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-2} \\ x_{N-1} \\ x_{N-2} \\ \vdots \\ x_1 \end{bmatrix} = \lambda \cdot \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_{N-2} \\ x_{N-1} \\ x_{N-2} \\ \vdots \\ x_1 \end{bmatrix}$$

Efetuada a multiplicação matricial no lado esquerdo da equação acima, cada componente do vetor $\lambda \cdot \mathbf{x}^T$ é dada por

$$\begin{aligned} \lambda \cdot x_m &= \sqrt{(2N-2)^{-1}} \cdot \left(x_0 + \sum_{k=1}^{N-2} \alpha^{k \cdot m} \cdot x_k + \alpha^{m \cdot (N-1)} \cdot x_{N-1} + \sum_{k=1}^{N-2} \alpha^{m \cdot (2N-2-k)} \cdot x_k \right) \\ &= \sqrt{(2N-2)^{-1}} \cdot \left[x_0 + \alpha^{m \cdot (N-1)} \cdot x_{N-1} + \sum_{k=1}^{N-2} \left(\alpha^{k \cdot m} + \alpha^{m \cdot (2N-2-k)} \right) \cdot x_k \right] \\ &= \sqrt{(2N-2)^{-1}} \cdot \left(x_0 + (-1)^m \cdot x_{N-1} + \sum_{k=1}^{N-2} 2 \cdot \cos_\alpha(km) \right) \cdot x_k, \end{aligned}$$

para $m = 0, 1, \dots, N-1$. Observa-se que a função co-seno que aparece acima é calculada em relação ao próprio elemento α , o qual possui ordem multiplicativa $\text{ord}(\alpha) = 2N-2$ no corpo finito $\text{GF}(p)$ considerado. Portanto,

$$\lambda \cdot x_m = \sqrt{(2N-2)^{-1}} \cdot \left[\frac{1}{2} \cdot x_0 + \frac{1}{2} \cdot (-1)^m \cdot x_{N-1} + \sum_{k=1}^{N-2} x_k \cdot \cos_\alpha(km) \right]. \quad (\text{B.1})$$

O ponto-chave para finalizar a prova é observar que a Equação (B.1) também pode ser obtida a partir da equação matricial

$$\lambda \cdot \begin{bmatrix} x_0 \\ \sqrt{2} \cdot x_1 \\ \vdots \\ \sqrt{2} \cdot x_{N-2} \\ x_{N-1} \end{bmatrix} = \sqrt{\frac{2}{N-1}} \cdot \begin{bmatrix} \frac{1}{2} & \frac{1}{\sqrt{2}} & \cdots & \frac{1}{\sqrt{2}} & \frac{1}{2} \\ \frac{1}{\sqrt{2}} & \cos_\alpha(1) & \cdots & \cos_\alpha(N-2) & \frac{1}{\sqrt{2}} \cdot \cos_\alpha(N-1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{1}{\sqrt{2}} & \cos_\alpha(N-2) & \cdots & \cos_\alpha((N-2)(N-2)) & \frac{1}{\sqrt{2}} \cdot \cos_\alpha((N-2)(N-1)) \\ \frac{1}{2} & \frac{1}{\sqrt{2}} \cdot \cos_\alpha(N-1) & \cdots & \frac{1}{\sqrt{2}} \cdot \cos_\alpha((N-1)(N-2)) & \frac{1}{2} \cdot \cos_\alpha((N-1)(N-1)) \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ \sqrt{2} \cdot x_1 \\ \vdots \\ \sqrt{2} \cdot x_{N-2} \\ x_{N-1} \end{bmatrix},$$

a qual equivale a $\lambda \cdot \hat{\mathbf{x}} = \mathbf{FC}_{N,1e} \cdot \hat{\mathbf{x}}$. Portanto, λ é também um autovalor da matriz de transformação da FFCT-1e e $\hat{\mathbf{x}} = [x_0, \sqrt{2} \cdot x_1, \dots, \sqrt{2} \cdot x_{N-2}, x_{N-1}]$ o seu respectivo autovetor.

Para o caso da FFST-1e, considera-se o autovetor \mathbf{x} da matriz \mathbf{FF}_{2N+2} associado ao autovalor $\lambda = j$ ou $\lambda = -j$. Equivalentemente, tem-se

$$\mathbf{FF}_{2N+2} \cdot \mathbf{x}^T = \lambda \cdot \mathbf{x}^T.$$

Usando a Equação (4.2), que define a matriz de transformação da FFT, reescreve-se a última equação da seguinte forma:

$$\sqrt{(2N+2)^{-1}} \cdot \begin{bmatrix} 1 & 1 & \dots & \dots & 1 \\ 1 & \alpha^1 & \dots & \dots & \alpha^{2N+1} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^N & \dots & \dots & \alpha^{(2N+1) \cdot N} \\ 1 & \alpha^{N+1} & \dots & \dots & \alpha^{(2N+1) \cdot (N+1)} \\ 1 & \alpha^{N+2} & \dots & \dots & \alpha^{(2N+1) \cdot (N+2)} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & \alpha^{2N+2} & \dots & \dots & \alpha^{(2N+1)^2} \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \\ 0 \\ -x_N \\ \vdots \\ -x_1 \end{bmatrix} = \lambda \cdot \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_N \\ 0 \\ -x_N \\ \vdots \\ -x_1 \end{bmatrix}$$

Efetuada a multiplicação matricial no lado esquerdo da equação acima, cada componente do vetor $\lambda \cdot \mathbf{x}^T$ é dada por

$$\begin{aligned} \lambda \cdot x_m &= \sqrt{(2N+2)^{-1}} \cdot \left(\sum_{k=1}^N \alpha^{k \cdot m} \cdot x_k - \sum_{k=1}^N \alpha^{m \cdot (2N+2-k)} \cdot x_k \right) \\ &= \sqrt{(2N+2)^{-1}} \cdot \sum_{k=1}^N \left(\alpha^{k \cdot m} - \alpha^{m \cdot (2N+2-k)} \right) \cdot x_k \\ &= \sqrt{(2N+2)^{-1}} \cdot \left(\sum_{k=1}^N 2 \cdot j \cdot \text{sen}_\alpha(km) \right) \cdot x_k, \end{aligned}$$

para $m = 1, \dots, N$. A função seno de corpo finito que aparece acima é calculada em relação ao próprio elemento α , o qual possui ordem multiplicativa $\text{ord}(\alpha) = 2N + 2$ no corpo finito $\text{GI}(p)$ considerado. Portanto,

$$j \cdot \lambda \cdot x_m = \sqrt{(2N+2)^{-1}} \cdot \left[\sum_{k=1}^N x_k \cdot \text{sen}_\alpha(km) \right]. \quad (\text{B.2})$$

O ponto-chave para finalizar a prova é observar que a Equação (B.2) também pode ser obtida

a partir da equação matricial

$$\lambda \cdot \begin{bmatrix} \sqrt{2} \cdot x_1 \\ \sqrt{2} \cdot x_2 \\ \vdots \\ \sqrt{2} \cdot x_{N-1} \\ \sqrt{2} \cdot x_N \end{bmatrix} = \sqrt{\frac{2}{N+1}} \cdot \begin{bmatrix} \text{sen}_\alpha(1) & \text{sen}_\alpha(2) & \cdots & \text{sen}_\alpha(N-1) & \text{sen}_\alpha(N) \\ \text{sen}_\alpha(2) & \text{sen}_\alpha(2 \cdot 2) & \cdots & \text{sen}_\alpha(2(N-1)) & \text{sen}_\alpha(2N) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \text{sen}_\alpha(N-1) & \text{sen}_\alpha((N-1)2) & \cdots & \text{sen}_\alpha((N-1)(N-1)) & \text{sen}_\alpha((N-1)N) \\ \text{sen}_\alpha(N) & \text{sen}_\alpha(2N) & \cdots & \text{sen}_\alpha(N(N-1)) & \text{sen}_\alpha(N \cdot N) \end{bmatrix} \cdot \begin{bmatrix} \sqrt{2} \cdot x_1 \\ \sqrt{2} \cdot x_2 \\ \vdots \\ \sqrt{2} \cdot x_{N-1} \\ \sqrt{2} \cdot x_N \end{bmatrix},$$

a qual equivale a $j \cdot \lambda \cdot \hat{\mathbf{x}} = \mathbf{FS}_{N,1e} \cdot \hat{\mathbf{x}}$. Portanto, λ é também um autovalor da matriz de transformação da FFST-1e e $\hat{\mathbf{x}} = \sqrt{2} [x_1, x_2, \dots, x_{N-1}, x_N]$ o seu respectivo autovetor.

B.2 Demonstração da Proposição 4.9

Considerando a expressão $\lambda \mathbf{x} = \mathbf{FF}_{2N,G} \mathbf{x}$ e observando que o vetor \mathbf{x} possui simetria ímpar, pode-se escrever

$$\lambda x_k = \sqrt{(2N)^{-1}} \left(\sum_{i=0}^{N-1} x_i \alpha^{(i+\frac{1}{2})(k+\frac{1}{2})} - \sum_{i=N}^{2N-1} x_{2N-1-i} \alpha^{(i+\frac{1}{2})(k+\frac{1}{2})} \right),$$

para $0 \leq k \leq N-1$. Usando o fato de que $\alpha^{2N(k+\frac{1}{2})} = -1$, o segundo termo no lado direito da última equação pode ser reescrito como

$$\sum_{i=N}^{2N-1} x_{2N-1-i} \alpha^{(i+\frac{1}{2})(k+\frac{1}{2})} = - \sum_{l=0}^{N-1} x_l \alpha^{-(l+\frac{1}{2})(k+\frac{1}{2})}.$$

Substituindo o resultado acima na equação original, tem-se

$$\begin{aligned} \lambda x_k &= \sqrt{(2N)^{-1}} \sum_{i=0}^{N-1} x_i \left(\alpha^{(i+\frac{1}{2})(k+\frac{1}{2})} + \alpha^{-(i+\frac{1}{2})(k+\frac{1}{2})} \right) \\ &= \sqrt{2N^{-1}} \sum_{i=0}^{N-1} x_i \cos_\alpha \left(\left(i + \frac{1}{2} \right) \left(k + \frac{1}{2} \right) \right), \quad 0 \leq k \leq N-1. \end{aligned}$$

A partir da igualdade acima, conclui-se que o vetor $\hat{\mathbf{x}}$, cuja forma é apresentada na Proposição 4.9, é um autovetor da matriz da FFCT-4e ($\lambda = 1, -1$).

No caso em que o vetor \mathbf{x} possui simetria par, considerando a expressão $\lambda \mathbf{x} = \mathbf{FF}_{2N,G} \mathbf{x}$, escreve-se

$$\lambda x_k = \sqrt{(2N)^{-1}} \left(\sum_{i=0}^{N-1} x_i \alpha^{(i+\frac{1}{2})(k+\frac{1}{2})} - \sum_{i=N}^{2N-1} x_{2N-1-i} \alpha^{(i+\frac{1}{2})(k+\frac{1}{2})} \right),$$

para $0 \leq k \leq N-1$. Usando o fato de que $\alpha^{2N(k+\frac{1}{2})} = -1$, a última equação é reescrita como

$$\begin{aligned} \lambda x_k &= \sqrt{(2N)^{-1}} \sum_{i=0}^{N-1} x_i \left(\alpha^{(i+\frac{1}{2})(k+\frac{1}{2})} + \alpha^{-(i+\frac{1}{2})(k+\frac{1}{2})} \right) \\ &= j\sqrt{2N^{-1}} \sum_{i=0}^{N-1} x_i \operatorname{sen}_\alpha \left(\left(i + \frac{1}{2} \right) \left(k + \frac{1}{2} \right) \right), \quad 0 \leq k \leq N-1. \end{aligned}$$

A partir da igualdade acima, conclui-se que o vetor $\tilde{\mathbf{x}}$, cuja forma é apresentada na Proposição 4.9, é um autovetor da matriz da FFST-4e ($\lambda = j, -j$).

SOBRE O AUTOR



O autor nasceu em Recife, Pernambuco, no dia 11 de maio de 1980. Em 2002, obteve o grau de Bacharel em Engenharia Elétrica (modalidade Eletrônica) e, em 2004, o de Mestre em Engenharia Elétrica, ambos pela Universidade Federal de Pernambuco (UFPE). Durante o doutorado, participou do programa de doutorado com estágio no exterior (PDEE) na Escola de Matemática e Estatística, Universidade Carleton (Ottawa, Canadá). Entre suas áreas de interesse estão Processamento Digital de Sinais e ferramentas matemáticas sobre corpos finitos.

Endereço: Rua Dr. José Higino Ribeiro Campos, 118
Rosarinho
Recife – PE, Brasil
C.E.P.: 52.041-230

e-mail: juliano_bandeira@hotmail.com

Esta tese foi diagramada usando $\text{\LaTeX} 2_{\epsilon}$ ¹ pelo autor.

¹ $\text{\LaTeX} 2_{\epsilon}$ é uma extensão do \LaTeX . \LaTeX é uma coleção de macros criadas por Leslie Lamport para o sistema \TeX , que foi desenvolvido por Donald E. Knuth. \TeX é uma marca registrada da Sociedade Americana de Matemática (\mathcal{AMS}). O estilo usado na formatação desta tese foi escrito por Dinesh Das, Universidade do Texas. Modificado em 2001 por Renato José de Sobral Cintra, Universidade Federal de Pernambuco, e em 2005 por André Leite Wanderley.