Universidade Estadual de Campinas
Departamento de Matemática Aplicada

Doutorado em Matemática Aplicada

O Problema Inverso da Fase: Teoria e Algoritmos.

Autor: Felipe Rogério Pimentel

Orientador: Nir Cohen

O Problema Inverso da Fase: Teoria e Algoritmos

Este exemplar corresponde à redação final da tese devidamente corrigida e defendida por Felipe Rogério Pimentel e aprovada pela comissão julgadora.

Campinas, 2 de Julho de 2003.

Nir Cohen

Prof. Dr. Nir Cohen
Orientador

Banca Examinadora

Nir Cohen
José Mario Martinez
Lúcio Tunes dos Santos
Nelson Delfino D'ávila Mascarenhas
Marcos Craizer

Tese apresentada ao Instituto de Matemática, Estatística e Computação Científica, UNICAMP, como requisito parcial para obtenção do Título de DOUTOR em Matemática Aplicada.

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA DO IMECC DA UNICAMP

Pimentel, Felipe Rogério

P649p O problema inverso da fase: teoria e algoritmos / Felipe Rogério Pimentel -- Campinas, [S.P. :s.n.], 2003.

Orientador : Nir Cohen

Tese (doutorado) - Universidade Estadual de Campinas, Instituto de Matemática, Estatística e Computação Científica.

1. Fourier, Transformações de. 2. Processamento de imagens – Técnicas digitais. 3. Algoritmos. 4. Otimização matemática. I. Cohen, Nir. II. Universidade Estadual de Campinas. Instituto de Matemática, Estatística e Computação Científica. III. Título.

Tese de Doutorado defendida em 02 de julho de 2003 e aprovada

Pela Banca Examinadora composta pelos Profs. Drs.

Prof (a). Dr (a). NIR COHEN

Prof (a). Dr (a). JOSÉ MÁRIO MARTÍNEZ PEREZ

Prof (a). Dr (a). LÚCIO TUNES DOS SANTOS

Prof (a). Dr (a). NELSON DELFINO D`ÁVILA MASCARENHAS

Prof (a). Dr (a). MARCOS CRAIZER

Agradecimentos

A Deus e ao Nosso Senhor do Bonfim.

Ao Prof. Nir Cohen, meu orientador, por tanta dedicação , competência e paciência na realização deste trabalho.

Aos Profs. Robert J. Plemmons (Wake Forest University) e Moody T. Chu (North Caroline State University - NCSU) que muito me ajudaram durante o período (Setembro de 1999 a Outubro de 2000) do meu Doutorado Sanduíche na NCSU. A maior contribuição destes professores refere-se ao pacote de otimização desenvolvido pela equipe de pesquisadores do *Argonne National Laboratory Optimization Technology Center* e ao pacote FFTW, ambos fundamentais para os resultados numéricos apresentados nesta tese.

A todos os membros da Banca Examinadora, pela disposição em avaliar este trabalho que é muito importante para a minha carreira.

Ao Renato F. Cantão, pelo apoio na área de informática, que muito me ajudou na instalação e operacionalização dos pacotes (citados acima) em meu computador.

Ao pessoal da área administrativa e da secretaria do IMECC, em especial à secretaria Fátima do Depto. de Matemática Aplicada.

À Univ. Fed. de Ouro Preto (UFOP) pelo apoio em me conceder licença de minhas atividades acadêmicas, por um período de 4 anos, para a realização do meu Doutorado.

Aos colegas do DEMAT/UFOP que sempre me deram apoio para a conclusão deste trabalho.

À agência CAPES que financiou meus estudos de Doutorado através dos programas *PICDT* e *Doutorado Sanduíche*.

Por fim, quero agradecer a todas as pessoas que eu amo e que muito me ajudaram nas horas de tensão e angústia.

...cada um sabe a dor e a delícia de ser o que é...

Caetano Veloso

Dedico este livro

à memoria de meus pais

Carolina U. Pimentel e

Diamantino Pimentel, e

a todas as pessoas

que eu amo

Resumo

A Tese aborda o problema de recuperação da fase de um objeto original f (uni ou bidimensional), a partir dos dados das amplitudes de sua transformada de Fourier discreta F. O problema é abordado de maneira teórica e numérica.

Teoricamente nós tentamos encontrar condições necessárias e suficientes sobre as amplitudes de F a fim de que exista uma solução real f para o problema inverso associado. Para alguns casos particulares, damos uma descrição completa destas condições . No entanto a generalização destas condições para problemas maiores torna-se extremamente complexa, exceto a de um determinado conjunto, C, de identidades, que descrevem certas condições necessárias sobre as amplitudes de F para a solvência do problema e que foram devidamente exibidas nesta tese, tanto para o caso unidimensional quanto bidimensional. Jorge L. C. Sanz afirma em um de seus artigos que existe um certo conjunto, S, de identidades polinomiais que formam condições necessárias para o problema inverso da fase. Sanz conjeturou que encontrar tais identidades seria uma tarefa desafiadora e extremamente complicada. O conjunto C a que nos referimos anteriormente é na verdade um subconjunto de S.

Numericamente, nós propomos um novo algoritmo para recuperar as fases de F a partir de suas amplitudes. Esse algoritmo se resume em aplicar um método de otimização Quasi-Newton denominado L-BFGS-B para minimizar a função custo dada pela norma quadrada de um objeto real discreto avaliado fora do suporte. Os comentários acerca da convergência do método bem como sua comparação com outros algoritmos, já existentes, de reconstrução da fase foram devidamente registrados.

Abstract

The thesis discusses theoretical and numerical aspects of the (one or two dimensional) phase retrieval problem for an object f, from the amplitudes of its Discrete Fourier Transform F.

Theoretically we try to find out necessary and sufficient conditions on the amplitudes of F to guarantee the solvability of the related inverse problem. We completely describe these conditions in some particular cases despite their generalization for bigger problems becoming extremely complex. In spite of these difficulties we exhibit, in both the one and two dimensional cases, a certain class, C, of identities that describes some of those necessary conditions. Jorge L. C. Sanz confirms, in one of his papers, the existence of some class, S, of polinomial identities that are necessary conditions for the phase retrieval problem. He conjectured that to find out such a set of identities would be an overwhelmingly difficult job. Our set C above actually is a subset of S.

Numerically, we suggest a new algorithm to recover the phases of F from its amplitudes. Actually this algorithm uses a Quasi-Newton optimization method called L-BFGS-B, to minimize the cost function given by the squared norm of a real discrete object, computed off the support. In addition, we comment on the convergence of the method, as well as its comparison to other phase recovery algorithms.

Índice

Lista de Símbolos

$\mathcal{F}$	;	Transformada de Fourier
(f, F)	;	Par de Fourier. f e F podem ser vetor ou matriz
Ω	;	igual a $[0, n-1] \subset \mathbb{Z}$ ou a $[0, n_1 - 1] \times [0, n_2 - 1] \subset \mathbb{Z}^2$. Os inteiros n, n_1 e n_2 são pares com $n = 2m$, $n_1 = 2m_1$ e $n_2 = 2m_2$
$\mathcal{S}$	;	Suporte de uma imagem, igual a $[0, m-1] \subset \mathbb{Z}$ ou a $[0, m_1 - 1] \times [0, m_2 - 1] \subset \mathbb{Z}^2$
f_j	;	j-ésima componente de f. O índice j pode ser uni ou bidimensional
$(f_j)_{j \in \Omega}$	;	Matriz ou vetor de componentes f_j, $j \in \Omega$
$f_{(\mathbf{1})}$	;	Subvetor formado pelas m primeiras componentes de f
$f_{(\mathbf{2})}$	;	Subvetor formado pelas m últimas componentes de f
F_u	;	u-ésima componente de F. O índice u pode ser uni ou bidimensional
α_F	;	Conjunto das amplitudes de F armazenadas na forma vetorial
α	;	Subvetor de α_F formado pelas amplitudes intermediárias de F
α_c	;	Subvetor de α_F formado pelas amplitudes dos cantos de F
ϕ_F	;	Conjunto das fases de F armazenadas na forma vetorial
ϕ	;	Subvetor de ϕ_F formado pelas fases intermediárias de F
ϕ_c	;	Subvetor de ϕ_F formado pelas fases dos cantos de F
$\|F\|$	;	Vetor ou matriz das amplitudes de F
$g(x, y) * f(x, y)$	;	Convolução contínua
$f_j * g_j$	;	Convolução discreta ou circular
$f_j \star g_j$	;	Correlação discreta ou circular
$g_{k(\mathrm{mod}\, n)}$	;	Significa $g_k = g_r$, quando r é o resto da divisão de k por n

$g_{(j,k)(\mathrm{mod}\ n_1,\mathrm{mod}\ n_2)}$	;	$g_{jk} = g_{rs}$, quando r e s são os restos respectivos das divisões de j por n_1 e k por n_2
$f^*(x,y)$	;	Conjugado complexo da função $f(x,y)$
F^*_{uv}	;	Conjugado complexo do número F_{uv}
$Z(F)$	;	Conjunto dos zeros de F
$P_F(z), F(z), F(z_1,z_2)$	;	Transformada-z
$f_{(\mathbf{jk})}$	;	Bloco (j,k) da matriz f, $j,k = 1,2$
$\mathbb{M}_{n_1\times n_2}(\mathbb{C})$	;	Espaço das matrizes complexas de ordem $n_1 \times n_2$
$\mathbf{A}_{jk}$ ou $(\mathbf{A})_{jk}$	;	Termo geral da matriz $\mathbf{A}$ da j-ésima linha e k-ésima coluna
$[\mathbf{A}]^{(k)}$	;	k-ésima coluna da matriz $\mathbf{A}$
$[\mathbf{A}]_{(j)}$	;	j-ésima linha da matriz $\mathbf{A}$
$\|\mathbf{A}\|$	;	Norma de Frobenius de $\mathbf{A}$
$\omega, \omega_1, \omega_2$	;	Números complexos dados respectivamente por $e^{\pi i/m}$, $e^{\pi i/m_1}$, $e^{\pi i/m_2}$
$\mathcal{W}$	;	Matriz de Fourier de termo geral ω^{-jk}
$\mathcal{W}_l$	;	Matriz de Fourier de termo geral ω_l^{-jk}, $l = 1,2$
$\mathcal{W}_{(\mathbf{1})}, \mathcal{W}_{(\mathbf{2})}$	;	Subblocos da matriz $\mathcal{W}$
$\overline{\mathbf{A}}$	;	Complexo conjugado da matriz $\mathbf{A}$
$\mathbf{A}^*$	;	Transposto conjugado da matriz $\mathbf{A}$
$\mathbf{A} \oplus \mathbf{B}$	;	Soma direta das matrizes $\mathbf{A}$ e $\mathbf{B}$
$\mathrm{diag}(d_0, d_1, \ldots d_{n-1})$	;	Matriz diagonal que tem os elementos $d_0, d_1, \ldots d_{n-1}$ na diagonal principal
$T_{(\phi_0,\phi_m)}, T_{\phi_c}$	;	Toróides
$\Lambda, \Lambda_1, \Lambda_2, \Lambda_3$	;	Conjunto de índices para ordenar as componentes de α_F e ϕ_F
$P_u(\alpha_F), R_v(\alpha_F)$	;	Polinômios P_u e R_v avaliados no ponto α_F
$\mathbf{f}_\mathcal{S}$	;	Vetor que armazena os pixels do suporte de um sinal, ou imagem, f
$\mathbb{R}[y]$	;	Anel dos polinômios em y com coeficientes reais
ImP	;	Conjunto imagem da função polinômial P
$Z(\mathrm{ImP})$	;	Conjunto dos polinômios de $\mathbb{R}[y]$ que se anulam em todos os pontos de ImP
$J_P(\mathbf{f}_\mathcal{S})$	;	Matriz jacobiana do polinômio P no ponto $\mathbf{f}_\mathcal{S}$
$\rho(J_P(\mathbf{f}_\mathcal{S}))$	;	Posto do jacobiano $J_P(\mathbf{f}_\mathcal{S})$
$< \cdot, \cdot >$	;	Produto interno canônico real ou complexo
P_τ, P_ϑ	;	Projeções sobre τ e ϑ respectivamente

$\mathbf{E}_{jk}$	;	Matriz que tem 1 na posição (j,k) e 0 nas demais
$\mathrm{tr}(\mathbf{A}), \mathrm{Re}(\mathbf{A}), \mathrm{Im}(\mathbf{A})$	;	Traço, parte real e parte imaginária da matriz $\mathbf{A}$
$\epsilon(f,g)$	;	Erro no domínio de Fourier
$\delta(f,g)$	;	Erro no domínio do objeto
$d'(h)$	;	Derivada de d em h, calculada sobre a variedade τ.
$d''(h)$	;	Matriz Hessiana de d em $h \in \tau$.
$\nabla_\theta(d \circ f)(\theta)$	;	Gradiente de $d \circ f$ em θ
factr, pgtol	;	Critérios de parada adotado pelo método L.BFGS.B
epsmch	;	Número de precisão da máquina onde se realizam os testes numéricos
$L_k := L(\theta^{(k)})$	;	Função custo $L(\theta)$ avaliada no vetor de fases $\theta^{(k)}$
$\theta^{(0)}$	;	Condição inicial no problema de minimização de $L(\theta)$
$\mathbf{g}_k$	;	Vetor gradiente ∇L avaliado em $\theta^{(k)}$
$f^{(0)}$	;	Imagem inicial tal que as amplitudes e as fases de sua DFT, $F^{(0)}$, são formadas a partir de (α_c, α) e $(\phi_c, \theta^{(0)})$ respectivamente
$\mathrm{seed}(X)$	;	Parâmetro para indexar a variável X por algum número inteiro positivo
$\eta_{\mathbf{x}}$	;	Ruído produzido no vetor $\mathbf{x}$
$\varepsilon_{\mathbf{x}}$	;	Índice do ruído $\eta_{\mathbf{x}}$
SNR	;	Siganl to Noise Ratio. SNR e $\varepsilon_{\mathbf{x}}$ são inversamente proporcionais
$\mathbf{x}. * \mathbf{y}$	;	Vetor obtido pelo produto componente a componente dos vetores $\mathbf{x}$ e $\mathbf{y}$
$a. \wedge \mathbf{x}$	;	Vetor obtido pela potência entre o número a e cada componente do vetor $\mathbf{x}$
$\theta_k^{(0)}$	;	Condição inicial de fases de Fourier que depende
	;	do valor atribuído ao índice ε_k
N_R	;	Norma Relativa para medir a distância entre a fase original ϕ e a condição inicial $\theta^{(0)}$
$D_M\|_{\mathrm{MET}}$	;	Distância Máxima que a condição inicial $\theta^{(0)}$ deve se encontrar da fase original ϕ para garantir a convergência do método MET
$\tilde{f}\|_{\mathrm{MET}}$	;	Imagem obtida pelo método MET
$e(\tilde{f})$	;	Erro dado pela norma $\|\tilde{f}_{(2)}\|^2$.
$\delta_{\tilde{f}}(\mathrm{MET})$	;	Erro no domínio do objeto, $\delta(f,\tilde{f})$, quando $\tilde{f}$ é obtido pelo método MET
$\nabla_\beta J,\ \nabla_\lambda J$	;	Derivadas de J com respeito a β e λ respectivamente

Introdução

Neste trabalho estudaremos o problema da fase, também conhecido como o problema de decorrelação. O objetivo é reconstruir uma imagem digital f a partir dos dados da amplitude $|F|$ de sua transformada de Fourier discreta (DFT) $F = \mathcal{F}(f)$. Este problema é tema central no estudo spectral de imagens astronômicas; testes de espalhamento de raio-X de cristais, polímeros e DNA; e outras aplicações.

O problema da fase foi formulado no final dos anos 60 no contexto de imagens astronômicas [91], e a maioria dos algoritmos iterativos usados hoje em dia já eram conhecidos na década de 70[1]. Com tudo, os algoritmos existentes não são satisfatórios na presença de ruido nos dados.

Neste trabalho abordaremos principalmente dois aspectos centrais relacionados ao papel do ruído no problema da fase: as condições de consistência para dados de amplitudes que garantem a solvabilidade do problema, e um novo algoritmo numérico que visa o aumento de robustez a ruídos no processo de reconstrução.

Neste capítulo introdutório definiremos o problema da fase, no contexto geral do processamento de imagens digitais; descreveremos em breve algumas aplicações e os algoritmos existentes; discutiremos em breve os aspectos de existência e unicidade e daremos uma panorama mais completo do conteudo e inovação da tese.

0.1 O problema da fase para imagens digitais

Uma *imagem digital* [33] é representada por uma matriz real não -negativa $n_1 \times n_2$ cujos elementos serão denotados por f_{ij} ou $f(i,j)$. O par (i,j) $(i = 0, \cdots, n_1 - 1, j = 0, \cdots, n_2 - 1)$ define um *pixel* da imagem, enquanto o valor f_{ij} representa o *grau de cinza* do pixel, sendo ele normalmente um número do intervalo $[0,1]$ com 0 representando preto e 1 branco.

Alternativamente, podemos considerar f_{ij} como uma função $f : \mathbb{Z}^2 \to \mathbb{R}$ *com suporte finito.* Em geral, o conjunto

$$\Omega := \{0, \cdots, n_1 - 1\} \times \{0, \cdots, n_2 - 1\} \subset \mathbb{Z}^2$$

[1]Na cristalografia, o problema ganhou fama com o prêmio Nobel em Química, conferido a H. Hauptman em 1985, baseado em uma sequência de artigos científicos no periodo 1950-1985, com os principais deles publicados na revista "Acta Crist. Sect. A".

servirá como suporte; porém, na formulação do problema da fase consideraremos a seguinte *condição de suporte reduzido:* n_1, n_2 são *pares*, ou seja $n_1 = 2m_1$ e $n_2 = 2m_2$; e

$$f_{ij} = 0 \quad \text{sempre que} \quad i > m_1 - 1 \quad \text{ou} \quad j > m_2 - 1. \tag{1}$$

Sob esta condição o conteudo de informação ocupará, no máximo, um quarto da imagem que, sem perda de generalidade, sempre será o quarto superior da esquerda. A condição (1) foi incorporada ao problema da fase por razões matemáticas: evitar *aliasing* [82] - pp. 235 - e garantir a unicidade da solução. Portanto, ao longo do trabalho referiremos ao conjunto

$$\mathcal{S} := \{0, \cdots, m_1 - 1\} \times \{0, \cdots, m_2 - 1\}$$

como o "suporte" da imagem, sempre mantendo nossa interpretação da imagem como matriz $n_1 \times n_2$, e não $m_1 \times m_2$.

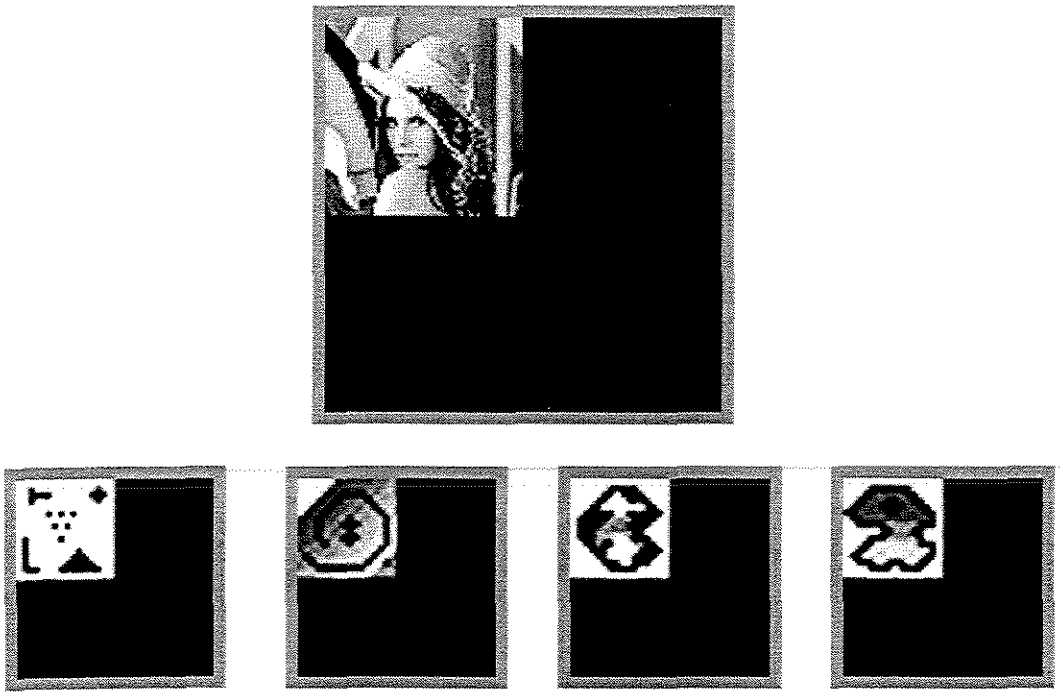

Figura 1: Exemplo de imagens digitais com a restrição de suporte. No topo temos uma imagem 128×128, com suporte 64×64. Nas imagens de baixo, o suporte é de tamanho 16×16.

A *transformada de Fourier discreta* (DFT) de f será considerada como uma imagem complexa F_{uv}, $(u, v) \in \Omega$, obtida através da relação

$$F_{uv} = \mathcal{F}(f_{jk}) = \frac{1}{n_1 n_2} \sum_{j=0}^{n_1-1} \sum_{k=0}^{n_2-1} f_{jk} \exp[-2\pi i(\frac{uj}{n_1} + \frac{vk}{n_2})], \tag{2}$$

Usando a representação polar, temos $F_{uv} = |F_{uv}| \cdot \exp(i\phi_{uv})$, com *amplitude* (ou *magnitude*, ou *módulo*) $|F_{uv}|$ e *fase* ϕ_{uv}. Portanto, podemos armazenar os dados spectrais de f em duas matrizes reais $n_1 \times n_2$, também denotadas $|F|$ e ϕ, codificando amplitudes e fases.

> *No problema da fase pede-se recuperar a fase* ϕ *da DFT de uma imagem digital* f, *sujeito à condição de suporte diminuido* (1), *a partir dos dados da amplitude* $|F|$.

Recuperando-se a fase, recupera-se portanto a imagem f.

0.2 A origem do problema

Na realidade, a imagem a ser recuperada é *analógica* [33], [82] e pode ser modelada por uma função real não -negativa com suporte compacto: $g(x,y) \quad (x,y) \in \Omega' \subset \mathbb{R}^2$. Nesse contexto, o problema da fase consiste na sua reconstrução a partir de dados da amplitude de sua *transformada de Fourier*, dada por

$$G(u,v) = \mathcal{F}[g(x,y)] = \iint_{\Omega'} g(x,y) \exp(-2\pi\, i\,(ux+vy))dx\, dy. \tag{3}$$

Observamos que neste contexto a restrição de suporte reduzido não é relevante, pois uma diminuição do objeto obtida através de contração apenas resulta em expansão proporcional de G.

Em contraste ao aspecto *analógico* do problema original, o advento de tecnologias de análise, síntese e gráfica digitais nas últimas décadas tem favorecido o processamento *digital* do problema. Para que as imagens digitais (f,F) (satisfazendo (1)) sejam aproximações fieis de (g,G), é necessário que o suporte de g seja igualmente diminuido, ou seja, com a informação ocupando apenas um quarto do campo de vista, $S' \subset \Omega,'$ analogamente à situação da Fig. 1.

De maneira análoga à definição da transformada direta, definimos a *transformada de Fourier inversa*

$$g(x,y) = \mathcal{F}^{-1}[G(u,v)] = \frac{1}{2\pi}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} G(u,v)\exp(2\pi\, i\,(ux+vy))du\, dv.$$

Será útil descrevermos a *convolução contínua* [11], [33] de duas funções $g(x,y)$ e $h(x,y)$, denotada por $g(x,y) * h(x,y)$, definida pela integral

$$g(x,y) * h(x,y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(r,s)h(x-r,y-s)dr\, ds.$$

Algumas propriedades importantes são facilmente verificadas como por exemplo

$$\begin{aligned}
\mathcal{F}^{-1}[G^*(u,v)] &= g^*(-x,-y) \text{ (Fórmula de inversão alternada)},\\
\mathcal{F}^{-1}[e^{-2\pi i(au+bv)}G(u,v)] &= h(x-a,y-b) \text{ (Time shifting)},\\
\mathcal{F}[g(x,y)*h(x,y)] &= G(u,v)H(u,v) \text{ (Teorema da convolução)}.
\end{aligned}$$

0.2.1 Sistemas óticos

Considere um raio eletromagnético atravessando um sistema de lentes. Distinguimos 2 planos importantes perpendiculares à direção de propagação do raio, chamados *plano ocular* e *plano distante* ("near field" e "far field" - veja Fig. 2), relacionados aos 2 pontos focais extremos do sistema. Em testes spectrais, a ótica faz com que a imagem G no plano ocular seja o módulo da transformada de Fourier da imagem g no plano distante. Já na tomografia, por exemplo, a relação é dada pela transformada de Radon, que também pode ser reduzida à transformada de Fourier [82], pp. 328, [58], [56].

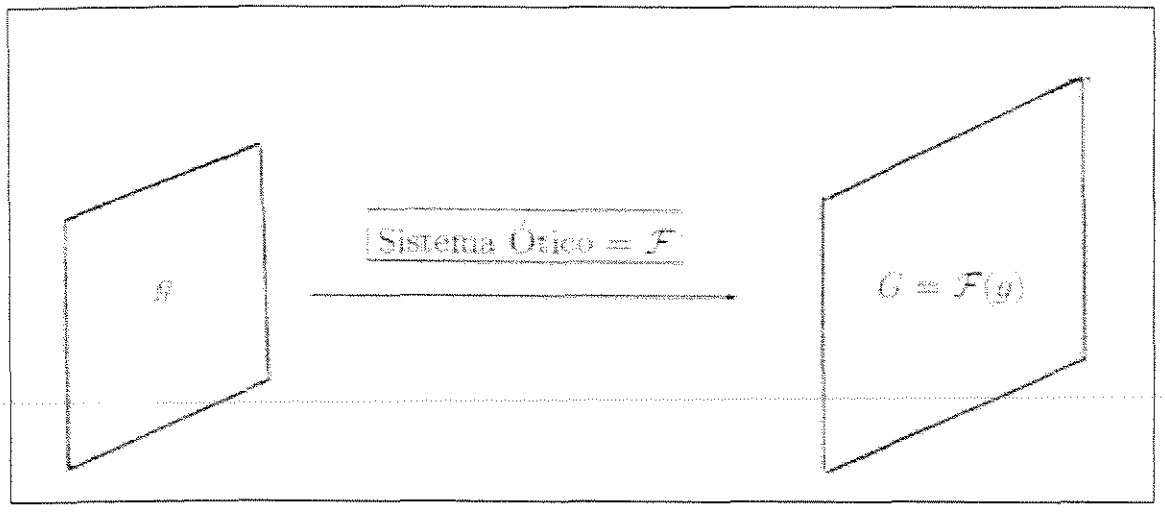

Figura 2: À esquerda temos a imagem g no plano distante ou *far field* e à direita, a imagem G no plano ocular ou *near field.*

O conteudo de frequência $\omega = (u,v)$ da onda eletromagnética pode ser descrito como $G(\omega) = |G(\omega)|\exp(i\omega(t-\phi(\omega)))$, com intensidade $|G(\omega)|$ e atraso (ou fase) $\phi(\omega)$. Apenas as intensidades podem ser medidas no plano ocular, ou seja, temos acesso apenas a $|G(u,v)|$. O problema da fase surgiu da tentativa de recuperar as fases perdidas. De particular importância estão as aplicações em *radio-astronomia e astronomia ótica* [82] (cap. 6 e 7), [4], [15], [16], [42].

Como mencionado, a restrição de suporte reduzido (1) não tem sua origem nas aplicações. Para impor esta restrição *in natura*, um sistema auxiliar de lentes e/ou

espelhos diminue duas vezes o tamanho da imagem estudada dentro do campo de vista no plano distante, obtendo um suporte reduzido S' e, dentro dele, uma imagem g, fiel à original. Nesse momento,uma imagem representando $|G|$ aparece no plano ocular. O processo de digitalização produz a partir de G uma aproximação digital $|F|$, usada como dados de amplitudes para o problema da fase.

0.2.2 Difração de Raio X

Considere um raio X de uma frequência dada atravessando um objeto, tipicamente uma molécula grande. Como na ótica, as fases dos raios espalhados são perdidas e podemos gravar apenas as amplitudes, ou intensidades. Mais precisamente, o objeto é posto no centro de uma esfera de raio suficientemente grande, e a leitura das intensidades é feita na superficie da esfera. Matematicamente, as intensidades gravadas representam a amplitude da transformada de Radon, e portanto, essencialmente em termos da transformada de Fourier.

Se a molécula estudada não é muito grande, o problema da fase que resulta seria similar ao problema da ótica. Complicações adicionais ocorrem no caso de moléculas grandes com estrutura periódica ou repetida, como cristais, polímeros e DNA.

Em espalhamento, não existe frequentemente uma maneira fácil de garantir a restrição de suporte reduzido, a fim de evitar aliasing e garantir solução única. Portanto, a resolução efetiva do problema da fase depende criticamente da estrutura de cada objeto estudado. O trabalho matemático sobre o problema da fase na *cristalografia* desenvolvido por H. Hauptman rendeu a ele um prêmio Nobel em Química, no ano de 1985. Hauptman encontrou uma maneira de fazer uma estimativa inicial da fase baseado na amplitude, que é valida para cristais com estrutura relativamente simples e dá início a um método DIRETO para a determinação da estrutura desses cristais [39], [71], [48], [42].

0.2.3 Espectroscopia

Aqui a função de densidade espectral é não -negativa e é a *transformada de Fourier discreta inversa* (veja 1.7) da função de coerência, γ, uma função complexa cuja amplitude $|\gamma|$ é facilmente medida. O problema consiste em determinar a fase de γ.

0.3 Fase versus amplitude

A questão da importância relativa da fase ϕ e da amplitude $|F|$ na reconstrução de uma imagem *real* f tem sido abordada em vários artigos, através de várias técnicas de reconstrução . No início, dados da fase e amplitude vindos de duas imagens diferentes foram usados na reconstrução , a fim de descrever a

contribuição relativa destes dois dados na formação da imagem resultante. Em seguida, problemas inversos baseados na representação polar foram abordados, tais como o problema da fase. Existem outros dois problemas desta classe os quais mencionamos a seguir.

0.3.1 O problema da amplitude

No problema da amplitude [82], pp 212-220, pede-se a reconstrução de uma imagem digital real f a partir das fases ϕ de sua DFT. Este tipo de problema inverso é abordado em *sismologia, acústica* [61], *radar, sonar*, entre outros.

O problema da amplitude é, matematicamente, um problema de minimização convexa (o que não é verdade para o problema da fase), e portanto pode ser resolvido com mais facilidade [52],[54],[37],[88],[89].

0.3.2 O problema das duas fases

Em alguns sistemas óticos, intensidades são medidas tanto no campo ocular como no campo distante. Nesse caso deve-se adotar uma visão mais ampla, considerando-se um par de imagens *complexas* $g = |g| \exp i\theta$ e $G = |G| \exp i\phi$, sendo G a transformada de Fourier de g. Pede-se a reconstrução [60] da imagem g ou, equivalentemente, de G, a partir dos dados das duas amplitudes $|g|$ e $|G|$.

A seguir comentamos algumas aplicações que abordam o problema das duas fases.

0.3.2.1 Spectrum shaping e kinoformas [29], [42]. Imaginemos um raio de laser que atravessa um sistema ótico, do tipo discutido antes, a fim de criar um holograma. Queremos sintetizar uma imagem desejada g no campo distante do sistema ótico, usando uma kinoforma: uma transparência colocada no campo *ocular* do sistema, i.e., na saída do laser.

Nossa restrição prática é a de que apenas um padrão composto de círculos abertos pequenos e idênticos, com seu lado aberto posto em vários ângulos, pode ser gravado na transparência. O padrão deve encher um disco $B(0, r)$, de raio dado r, compatível com a espessura do raio. Esse padrão pode ser considerado como uma aproximação grosseira da fase ϕ da transformada G da imagem desejada g, enquanto a amplitude $|F|$ aproxima uma função de *banda limitada* do tipo $|F| = C\chi_B$, uma constante multiplicando a função característica do disco dado. Assim obtemos uma versão particular do problema das duas fases, procurando-se a melhor escolha da fase ϕ para acomodar a imagem desejada g.

0.3.2.2 Projeto de lentes. Um problema similar foi estudado em [19] (veja também [60]). Dadas uma função real não -negativa $\rho(x)$, com $x \in \Omega \subset \mathbb{R}^2$ repre-

sentando a transparência de uma lente, e uma função $\alpha(u)$ que representa o modelo de campo de intensidades desejado, queremos encontrar uma função $\theta(x)$, $x \in \Omega$, que represente a espessura da lente, tal que a transformada de Fourier de $f = \rho e^{i\theta}$ tenha a amplitude α.

0.3.2.3 Wavefront sensing [91]. Neste tipo de problema, a imagem $|f(x)|^2$ de uma fonte pontual é gravada através de um sistema ótico. Assumindo que a aberração é uma *função de fase pura* no sentido de que seu módulo é constante e igual a 1, então $F(u)$ tem módulo $|F(u)|$ igual à abertura do diafragma do sistema ótico. O problema consiste em reconstruir a fase de $F(u)$.

0.4 Unicidade

Até finais da década de 70, havia muita dúvida de que o problema da fase para imagens digitais pudesse ser resolvido unicamente. De fato, a teoria de funções analíticas mostrava que existia um grande número de *soluções ambíguas* para o problema unidimensional (caso 1-D)[2].

As primeiras constatações de que o problema para o caso bidimensional (2-D) apresentava *solução única* vieram de resultados empíricos de tentativas de reconstrução de imagens óticas. Estes resultados deram esperanças de que o problema da fase para o caso 2-D poderia ser unicamente solúvel e foram, portanto, fonte de incentivo para se extender a teoria existente do caso 1-D ao caso 2-D. A demostração de unicidade, a partir de dados genéricos de amplitudes, apareceu pela primeira vez no artigo [12] (Bruck e Sodin 1979), mais uma vez usando a teoria de funções analíticas. A análise a seguir pode ser encontrada em [75], [74].

0.4.1 Imagens analógicas

Considere o problema de se recuperar um sinal complexo $f(x, y)$ de suporte compacto $\Omega' \subset \mathbb{R}^2$, supondo-se $|F(u, v)|$ conhecida para todas as frequências $(u, v) \in \mathbb{R}^2$. Esse problema não pode ter solução única pela seguinte razão : dado qualquer objeto $f(x, y)$, existem outros objetos com o mesmo módulo de Fourier: as *translações* do objeto, $f(x - a, y - b)$, a *imagem gêmea*, $f^*(-x, -y)$, e qualquer uma destas multiplicada por uma constante complexa de norma unitária, $\exp(i\phi_c)$ [26],[81],[75],[76]. Ambiguidades desse tipo serão chamadas *ambiguidades triviais.*

[2]Na seção 1.3 discutimos a não unicidade do problema da fase para o caso 1-D

Ambiguidades triviais diferem em posição mas tem a mesma aparência e podem ser consideradas "idênticas". Daqui em diante, diremos que o problema da fase *tem solução única* se todas suas soluções são ambíguas triviais de um único objeto.

Uma condição suficiente para a existência de ambiguidades não triviais para f é a existência de funções g, h de suporte compacto tal que vale a convolução $f = g*h$. Nesse caso, seja $\hat{g}$ a gêmea de g. Então $\hat{g} * h$ terá o mesmo módulo de Fourier de f, com suporte compacto, mas, em geral, não será uma ambiguidade trivial dela. Esta condição pode ser interpretada no domínio de frequências. Lembemos que, pelo teorema de Pólya-Plancherel [69],[70] a transformada de Fourier de um objeto de suporte compacto pode ser extendida a todo o $\mathbb{C}^2$ como uma função inteira do tipo exponencial. A existência de uma convolução $f = g * h$ (ambos g, h com suporte compacto) implica que a função analítica F fatora como um produto $F = GH$ de funções analíticas de tipo exponencial.

Podemos concluir, então , que a falta de unicidade para o problema da fase é, essencialmente, um problema de *fatorabilidade* ou *redutibilidade.*

Definição 0.1 *Dizemos que $F : \mathbb{R}^2 \longrightarrow \mathbb{C}$ é uma função banda-limitada se F pode ser unicamente extendida a $\mathbb{C}^2$ como uma função inteira do tipo exponencial, i.e., existe $A \geq 0$, $c \in \mathbb{R}$, $k \in \mathbb{Z}$ tal que*

$$|F(z_1, z_2)| \leq c(1 + ||(z_1, z_2)||)^k e^{A||(Im(z_1), Im(z_2))||}$$

para todo $z_1, z_2 \in \mathbb{C}$ (veja [40] - pp. 21, ou [75]).

Lema 0.1 *Sejam $Z(F)$ e $Z(H)$ os conjuntos dos zeros das funções analíticas $F, H : \mathbb{C}^2 \longrightarrow \mathbb{C}$. Se $Z(F) = Z(H)$ e se F é irredutível, então existe uma função inteira do tipo exponencial $G : \mathbb{C}^2 \longrightarrow \mathbb{C}$ tal que $H = FG$.*

Um resumo da prova do lema acima encontra-se em [42], pp. 149. A prova completa está em [75].

A unicidade de solução para o problema da fase no caso contínuo é uma consequência do próximo teorema (veja [75], [76] e [42] - pp 139). Apesar de nos restringirmos ao caso 2-D, esse teorema é válido para o caso n-dimensional; e será aplicado a funções banda-limitadas. O teorema foi retirado de [75], com as notações adaptadas de acordo com as adotadas neste trabalho.

Teorema 0.1 *Seja $F : \mathbb{R}^2 \longrightarrow \mathbb{C}$ uma função banda-limitada e $F(z_1, z_2)$, $z_1, z_2 \in \mathbb{C}$, sua única extensão inteira do tipo exponencial. Se $F(z_1, z_2)$ é irredutível em $\mathbb{C}^2$ então $f(x, y)$ pode ser unicamente recuperada, a menos das ambiguidades triviais, de $|F(u, v)|$, $u, v \in \mathbb{R}$.*

Resumo de demonstração do teorema 0.1 A prova do teorema repousa sobre o lema 0.1. Seja $F(z_1, z_2)$ irredutível e suponha que exista uma outra função banda-limitada $H : \mathbb{R}^2 \longrightarrow \mathbb{C}$ tal que $|H(u,v)| = |F(u,v)|$, $u, v \in \mathbb{R}$. Então $FF^* = HH^*$ e pode-se concluir que ou $Z(F) = Z(H) \cap Z(F)$ ou $Z(F) = Z(H^*) \cap Z(F)$. Mostra-se que a primeira opção implica $Z(F) = Z(H)$; e que, analogamente, a segunda implica $Z(F) = Z(H^*)$. Pelo lema acima, ou $H = FG$ ou $H^* = FG$. Assim $HH^* = FF^*GG^*$ e, por $FF^* = HH^*$, conclui-se que $1 = GG^* = |G(z_1, z_2)|^2$, $\forall (z_1, z_2) \in \mathbb{C}^2$. O resto da prova do teorema é para mostrar que $|G(z_1, z_2)| = e^{c_1 Im(z_1) + c_2 Im(z_2) + l}$, para $c_1, c_2, l \in \mathbb{R}$; ou seja G é do tipo exponencial. Em conclusão nós temos uma das seguintes alternnativas

$$H(z_1, z_2) = e^{-i(\alpha_1 z_1 + \alpha_2 z_2 + \beta)} F^*(z_1^*, z_2^*) \tag{4}$$

ou

$$H(z_1, z_2) = e^{-i(\alpha_1 z_1 + \alpha_2 z_2 + \beta)} F(z_1, z_2), \tag{5}$$

para α_1, α_2 e β constantes reais. Finalmente, restringindo as equações (4) e (5) ao caso real e em seguida aplicando a transformada inversa de fourier nos dois lados das equações , segue da "Fórmula de inversão alternada" e da propriedade "Time shifiting", que

$$h(x, y) = e^{-\beta i} f^*(-x - x_0, -y - y_0) \tag{6}$$

ou

$$h(x, y) = e^{-\beta i} f(x - x_0, y - y_0), \tag{7}$$

para algumas constantes reais x_0 e x_1. Com isto conclui-se a unicidade de $f(x, y)$ a menos das ambiguidades triviais ■

Não é nosso objetivo aprofundarmos o problema da fase em sua versão contínua. Os resultados relativos a funções analíticas enunciados nesta introdução são em geral elementares e servem apenas para dar ao leitor uma sustentação aos resultados similares da versão discreta.

0.4.2 Imagens digitais

Considere uma imagen digital *real* f e sua DFT F de acordo com (2), ignorando por enquanto qualquer restrição de suporte. Por conveniência de notação adotaremos nesta subseção o par (x, y), ao invés de (i, j), para os pixels da imagem f. Como *ambiguidades triviais* do objeto $f(x, y)$ para o problema da fase definimos as translações *periódicas* $f(x - a, y - b)$ $((a, b) \in \Omega)$, a gêmea, $f(-x, -y)$, e $-f(x, y)$. Tais funções têm o mesmo módulo de DFT como f.

Se, além disso, f satisfaz a restrição de suporte reduzido, e f_1 ocupa sua *janela não -trivial,* de tamanho $m_1 \times m_2$, com $f_1 \geq 0$, então as únicas ambiguidades

triviais possíveis de f (que mantêm a condição de suporte) são a própria $f(x,y)$ e sua gêmea[3]

$$\hat{f}(x,y) = f(m_1 - 1 - x, m_2 - 1 - y); \quad \forall\,(x,y) \in \mathcal{S}. \tag{8}$$

Figura 3: Imagem gêmea, $\hat{f}(x,y)$, dada como em (8). Note que $\hat{f}(x,y)$ mantém a mesma aparência da imagem original, $f(x,y)$, mudando apenas a posição dentro do suporte.

Isso fica válido enquanto o suporte mínimo da imagem não ocupar uma janela de tamanho $m_1' \times m_2'$ estritamente menor que $\mathcal{S}$. Caso isso ocorra, então, algumas translações da imagem também serão permitidas.

Uma condição suficiente para a quebra de unicidade na reconstrução da fase é obtida pela decomposição $f = f_1 * f_2$, onde $*$ é a *convolução circular* em (1.42). O mecanismo é idêntico ao descrito no caso analógico, e não vai ser repetido aqui.

Equivalentemente, esta condição envolve a fatorabilidade da DFT F, ou melhor, a fatorabilidade da transformada-z, que extende a DFT analiticamente da circunferência unitária ao inteiro campo complexo. Definimos a transformada-z de f como o polinômio nas variáveis complexas z_1 e z_2 dado por

$$F(z_1, z_2) = \sum_{x=0}^{m_1-1} \sum_{y=0}^{m_2-1} f(x,y) z_1^x z_2^y. \tag{9}$$

Quando $z_1 = \exp(-\pi i u_1/m_1)$ e $z_2 = \exp(-\pi i u_2/m_2)$ recupera-se a DFT.

[3]Note que por um abuso de linguagem adotamos a mesma terminologia "gêmea" para designar a imagem em (8) que é, na verdade, a translação periódica $f'(x - m_1 - 1, y - m_2 - 1)$ da imagem $f'(x,y) := f(-x,-y) = f(n_1 - x, n_2 - y)$.

A seguir citamos o principal resultado de unicidade, dada na versão discreta, para o problema da fase (veja [42] (pp 138)),[37][4] provado por Hayes:

Teorema 0.2 . *Se $F(z_1, z_2)$ em (9) é irredutível, então um objeto f com suporte reduzido é unicamente determinado, a menos de suas ambiguidades triviais, a partir da magnitude de sua transformada de Fourier, $|F(u,v)|$, $(u,v) \in \Omega$.*

A ocorrência de ambiguidades no problema da fase está associada então à fatorabilidade do polinômio associado. Isto explica a fundamental diferença entre os casos 1-D e 2-D, pois polinômios de uma variável, de grau maior ou igual a 2, são sempre fatoráveis (veja seção 1.3 para descrição completa das ambíguidades não triviais para o caso 1-D), enquanto polinômios de duas ou mais variáveis complexas *raramente* são fatoráveis [38],[77]. De fato, em [38], Hayes mostra que o conjunto de polinômios em duas ou mais variáveis que são redutíveis sobre o corpo dos complexos tem medida zero, o que, segundo o teorema 0.2, implica que

> *quase todas as imagens são unicamente recuperadas a partir das magnitudes de sua transformada de Fourier.*

0.5 Existência, consistência de dados, ruídos

O problema da fase é não linear e a existência de solução não é garantida. Seja H *imagem no dominio de frequências.* Então para que $|H|$ seja a amplitude da DFT de uma imagen real h, ela deve satisfazer a simetria $|H(n_1 - u, n_2 - v)| = |H(u,v)|$ (veja Proposições 1.1 e 1.2). Portanto o conjunto de amplitudes $|H|$ de imagens de tamanho $n_1 \times n_2$, que satisfazem tal simetria, é isomorfo a $\mathbb{R}_+^{2m_1 m_2 + 2}$. Dentro desse espaço, o subconjunto $\mathbf{C}$ de amplitudes para as quais o objeto recuperado h tem suporte reduzido nunca foi caracterizado analiticamente. Chamamos as $|H|'s$ pertencentes a $\mathbf{C}$ de *consistentes.*

Algebricamente é claro que $\mathbf{C}$ é um subconjunto *semi-algébrico* em $\mathbb{R}_+^{2m_1 m_2 + 2}$, ou seja, é uma união finita de conjuntos definidos por igualdades e desigualdades polinomiais ([7], pp. 23). Não encontramos nenhuma discussão sobre esse fato essencial na literatura do problema da fase.

Se na verdade $\mathbf{C}$ estiver contido em uma variedade $V \subset \mathbb{R}_+^{2m_1 m_2 + 2}$ poderíamos a principio obter uma condição *necessária* à consistência, em termos de um conjunto de *identidades polinomiais.* A referência [74] analisa a existência de identidades polinomiais nas variáveis $|F_{uv}|^2$ para o problema da fase. Sanz não encontrou, nem sequer enumerou, tais identidades e conjeturou que tal empreendimento pode ser difícil.

[4] Resultados adicionais foram obtidos por Fiddy *et al.* [23] e Nieto-Vesperinas e Dainty [68]. Barakat e Newsam [3] discutem condições *necessárias* para unicidade. Veja também caps. 6 e 13 de [82] para uma abordagem completa sobre a questão da unicidade.

Na prática, é comum o uso de problemas sintéticos, que usam dados a priori consistentes de amplitudes para a reconstrução . Na realidade, entretanto, dados de amplitudes são sempre corrompidos por *ruídos*, provindo de três fontes: ruído do meio ambiente, ruído de medidas, e ruído de discretização e processamento. Em sistemas óticos, ruídos do primeiro tipo representam, por exemplo, não -linearidades no modelo ótico, impuridades do ar, turbulências atmosféricas, etc. Em modelos de difração , são impuridades do cristal, etc. Na tecnologia atual, somente ruídos de discretização e processamento podem ser desprezados sem afetar a qualidade da reconstrução .

Infelizmente, o problema da fase não é bem-posto pois a existência de solução não é garantida. De acordo com Sanz em [74], pp. 662, ao perturbarmos dados consistentes obtemos, em geral, dados não consistentes. Mesmo dentro de **C**, não sabemos se perturbação pequena nos dados implica perturbação pequena na solução .

0.6 Algoritmos de reconstrução

Uma classe principal de algoritmos iterativos para o problema da fase oscila em cada passo entre o dominio de frequência e o dominio de objeto; no primeiro, eles corrigem erros de amplitude, no segundo, a falta de suporte reduzido (no dominio de objeto). O algoritmo mais natural e simples dessa classe é o algoritmo ER (Error Reduction) [31],[30], mas ele converge a "mínimos locais" de uma função custo associada e frequentemente não encontra a solução. **J. Fienup** sugeriu o algoritmo HIO (Hybrid Input-Output) [25] - [81], [82], que é considerado um dos melhores algoritmos iterativos existentes. Na verdade, acredita-se que o melhor dos iterativos é o algoritmo, também proposto por Fienup, que consiste numa combinação de ER e HIO.

O HIO é o único algoritmo iterativo conhecido que, acredita-se, converge à solução para quaisquer dados consistentes (ou seja, na ausência de ruídos) e em ambos os casos 1-D e 2-D. Entretanto, **nossos experimentos mostraram que no caso 1-D, HIO nem sempre convergiu a mínimo global** (veja tabela 1 do Cap. 4 com convergência em apenas 60% dos exemplos testados). Além disso um limitante para a taxa de convergência do HIO nunca foi encontrado. Para uma imagem de grande porte são necessários, tipicamente, centenas ou milhares de iterações [73]. De fato, **até o presente não foi encontrada uma prova de convergência para o HIO.**

Entre as desvantagens dos algoritmos desse tipo mencionamos a necessidade de calcular uma DFT e uma IDFT (DFT inversa) em cada iteração. Quanto ao algoritmo HIO, ele mostra pouca robustez a ruídos: ele raramente converge para dados com ruído, e a qualidade de reconstrução piora rapidamente ao diminuir-se o SNR (signal to noise ratio) [25], [26], [81], [77]. Fienup e outros autores sugeriram como

alternativas várias combinações dos algoritmos HIO e ER, sem resolver satisfatoriamente o problema de robustez [25], [82], pp. 250.

Entre as outras alternativas mencionamos algoritmos iterativos tipo Newton. Devido à natureza topológica do problema, a função custo é sempre não convexa [57]. No capítulo 3 propomos um novo algoritmo deste tipo.

0.7 Panorama da tese

A tese está dividida em 5 capítulos.

No capítulo 1, discutimos em detalhe o problema da fase para sinais 1-D e 2-D (digitais).

Na parte inédita deste capítulo, discutimos condições necessárias e suficientes à *consistência* dos dados. Damos indicações que tais condições serão *uma mistura de identidades e desigualdades*. Portanto, qualquer conjunto de *identidades* do tipo sugerido por Sanz não pode ser uma condição suficiente para a connsistência dos dados.

Para imagens de suporte reduzido de tamanho $(= m_1 \times m_2)$ 2×2 e 2×3 calculamos penosamente as condições necessárias e suficientes. Também calculamos para o caso geral o número esperado de *identidades*, usando o posto de um jacobiano associado. Esse número é $m_1 m_2 + 2$. Dessas identidades, encontramos $m_1 + m_2 + 1$ identidades, todas lineares nas variáveis $b_{uv} := |F_{uv}|^2$. No caso 2×3 são necessárias duas identidades adicionais, que são polinomiais nos $\sqrt{b_{uv}}$ mas não nos b_{uv}.

Mesmo que não foram encontradas *todas* as condições de consistência para o problema da fase, nossa discussão da natureza dessas condições é um primeiro passo necessário para sua análise no futuro.

No capítulo 2 descrevemos de maneira sucinta os principais métodos existentes de reconstrução para o problema da fase. Comentamos sobre a convergência destes métodos, na ausência ou presença de ruidos, e caracterizamos a convergência do algoritmo ER a minimizadores locais de uma função custo associada.

Na tentativa de superar o problema principal de reconstrução , ou seja, a instabilidade numérica, apresentamos no capítulo 3 um novo algoritmo que se baseia num método de otimização quasi-Newton, chamado L.BFGS.B [13], [95], de uma certa função custo não convexa, $L \geq 0$. Os mínimos globais de L atingem $L = 0$ e são as soluções do problema da fase.

Em vários experimentos foi constatado que o novo algoritmo convergiu a um nínimo local no sentido que, numericamente, o gradiente é zero e o Hessiano positivo semidefinido. No caso 1-D, ele de fato convergiu sempre a um nínimo global, ou seja, à solução do problema, com custo na ordem de 10^{-4} ou menos. Além disto, ele é mais robusto a ruidos que o algoritmo HIO. Entre suas desvantagens contamos a taxa de convergência que o torna, por enquanto, impróprio para imagens grandes

(de tamanho maior a 32×32). A originalidade desse novo método está no tipo de parâmetro adotado para a função custo, i.e., adota-se as fases da DFT, (e não os pixels da imagem, como a maioria dos métodos de otimização o fazem [67]) como parâmteros de minimização. Não foi encontrado na literatura nenhum trabalho que trata da minimização de tal função com respeito a esses tipos de parâmetros.

No capítulo 4 apresentaremos os principais resultados numéricos obtidos ao executarmos os algoritmos iterativos descritos nos capítulos 2 e 3. Apresentaremos várias tabelas comparando os resultados obtidos, apontando as vantagens e desvantagens de cada método, sob diversas condições específicas, em situações onde as amplitudes são consideradas ora com ruídos, ora sem ruídos.

No capítulo 5 examinamos a possibilidade de usar pré-filtragem para aproximar dados inconsistentes por dados "mais consistentes", usando as novas identidades encontradas no capítulo 1 como restrições para um problema de mínimos quadrados. Em seguida executamos os algoritmos sem e com esta pré-filtragem. O estudo completo da filtragem será feito apenas para o caso 1-D, tendo em vista que o trabalho para o caso 2-D ainda está incompleto e, portanto, indicado para pesquisas futuras.

Capítulo 1

Caracterização das amplitudes para o problema da fase

Nesse capítulo faremos um estudo do problema da fase (veja formulação na seção 1.2) para funções reais discretas que satisfazem a restrição de suporte. Discutiremos, entre outros assuntos, algumas condições necessárias que as amplitudes de Fourier devem satisfazer para que o problema da fase apresente solução (única).

1.1 Definições e notações

Repetiremos nesta seção algumas definições e conceitos já estabelecidos na introdução. Suponhamos que

$$\Omega = [0, n-1] \subset \mathbb{Z} \quad \text{(caso 1-D)},$$

ou

$$\Omega = [0, n_1 - 1] \times [0, n_2 - 1] \subset \mathbb{Z}^2 \quad \text{(caso 2-D)},$$

onde n, n_1 e n_2 são inteiros pares positivos dados por

$$n = 2m, \quad n_1 = 2m_1 \quad \text{e} \quad n_2 = 2m_2. \tag{1.1}$$

Seja $f = (f_j)_{j \in \Omega}$ um objeto real [1], que poderá representar um sinal (caso 1-D) ou uma imagem (caso 2-D), obtido pela discretização de uma determinada função $f(x)$. Para o caso 1-D, f é o vetor $n-$dimensional

$$f = (f_0, f_1, \ldots, f_{n-1})^T \in \mathbb{R}^n, \tag{1.2}$$

[1] Salvo menção em contrário, f será sempre real

e, para o caso 2-D, f é a matriz

$$f = \begin{bmatrix} f_{00} & f_{01} & \cdots & f_{0,n_2-1} \\ f_{10} & f_{11} & \cdots & f_{1,n_2-1} \\ \cdot & \cdot & \cdots & \cdot \\ f_{n_1-1,0} & f_{n_1-1,1} & \cdots & f_{n_1-1,n_2-1} \end{bmatrix} \in \mathbb{M}_{n_1 \times n_2}(\mathbb{R}). \tag{1.3}$$

Será conveniente, às vezes, que adotemos f na sua forma particionada por blocos. No caso 1-D, particionamos f em dois subvetores de tamanho $m \times 1$:

$$f = \begin{bmatrix} f_{(1)} \\ f_{(2)} \end{bmatrix}; \quad f_{(1)} = (f_0, f_1, \ldots, f_{m-1})^T; \quad f_{(2)} = (f_m, f_{m+1}, \ldots, f_{n-1})^T. \tag{1.4}$$

No caso 2-D particionamos f em blocos de tamanho $m_1 \times m_2$, i.e.,

$$f = \begin{bmatrix} f_{(11)} & f_{(12)} \\ f_{(21)} & f_{(22)} \end{bmatrix}. \tag{1.5}$$

Também denotaremos a $j-$ésima linha ($j = 0, 1, \ldots, n_1 - 1$) e a $k-$ésima coluna ($k = 0, 1, \ldots, n_2 - 1$) de uma matriz f respectivamente por

$$[f]_{(j)} \quad \text{e} \quad [f]^{(k)}.$$

Definição 1.1 *A transformada de Fourier discreta,* DFT, [11] *de* $f = (f_j)_{j \in \Omega}$ *é dada por* $F = (F_u)_{u \in \Omega}$, *onde*

$$F_u = \mathcal{F}(f_j) = \sum_{j \in \Omega} f_j \exp(-2\pi i \frac{u \cdot j}{n}) \tag{1.6}$$

Dizemos também que f *é a transformada de Fourier discreta inversa* (IDFT) *de* F.

No caso 2-D interpretamos (1.6) assim: u representa o par (u, v), j o par (j, k) e a expressão $(u \cdot j)/n$ torna-se igual a $uj/n_1 + vk/n_2$. É fácil concluir que a expressão para a IDFT é

$$f_j = \mathcal{F}^{-1}(F_u) = \frac{1}{N} \sum_{u \in \Omega} F_u \exp(2\pi i \frac{u \cdot j}{n}), \tag{1.7}$$

onde

$$\begin{aligned} N &= n, \quad \text{(caso 1-D), ou} \\ N &= n_1 n_2, \quad \text{(caso 2-D).} \end{aligned}$$

Diremos que (f, F) representa um *par de Fourier*[2] se f e F se relacionam segundo as fórmulas (1.6) e (1.7).

[2]Convém ressaltar que, na representação de um par de Fourier, adotaremos letras minúsculas para representar as funções objetos e as respectivas maiúsculas para representar suas transformadas de Fourier

Usualmente j é denominada coordenada no *dominio do tempo* (ou *domínio do objeto*), e u, coordenada no *domínio de frequências* (ou *domínio de Fourier*).

Adotaremos ao longo da tese a norma de Frobenius

$$||F|| = \left(\sum_{u \in \Omega} |F_u|^2 \right)^{1/2}.$$

O *teorema de Parseval* (veja p.ex. [11], pp. 130) garante que

$$||f||^2 = \frac{1}{N} ||F||^2. \tag{1.8}$$

Na forma polar

$$F_u = R_u + iI_u = |F_u| \exp(i\phi_u), \tag{1.9}$$

ϕ_u e $|F_u|$ representam, como antes, a fase e a amplitude da transformada de Fourier em u respectivamente. R_u e I_u são respectivamente as partes real e imaginária de F_u. ϕ_u e $|F_u|$ podem ser expressos em termos de I_u e R_u. É imediato que

$$|F_u| = \sqrt{R_u^2 + I_u^2}.$$

Quanto aos valores das fases, restringiremo-nos ao intervalo

$$-\pi < \phi_u \leq \pi, \quad \forall u.$$

Pelo fato de a função arcotangente tomar valores no intervalo aberto $(-\pi/2, \pi/2)$, obtemos as fases obedecendo-se à seguinte lei:

$$\phi_u = \begin{cases} \tan^{-1}[I_u/R_u] & \text{se } R_u \geq 0, \\ \tan^{-1}[I_u/R_u] - \pi & \text{se } R_u < 0. \end{cases}$$

Às vezes é mais conveniente trabalharmos com as expressões (1.6) e (1.7) dadas na forma matricial.

1.1.1 Forma matricial: caso 1-D

Para o caso unidimensional, as formas matriciais de (1.6) e (1.7) são respectivamente

$$F = \mathcal{W} f \tag{1.10}$$

e

$$f = \frac{1}{n} \mathcal{W}^* F, \tag{1.11}$$

onde f é o vetor dado em (1.2), F é o vetor

$$F = \mathcal{F}(f) = (F_0, F_1, \ldots, F_{n-1})^T \in \mathbb{C}^{n \times 1}, \tag{1.12}$$

e $\mathcal{W}$ é a *Matriz de Fourier* cujo termo geral é dado por

$$\mathcal{W}_{jk} = \omega^{-jk}, \quad j,k \in \{0,1,2,\ldots,n-1\}, \tag{1.13}$$

para $\omega = \exp(\pi i/m)$. O símbolo $*$ representa o *transposto conjugado complexo*. Verifica-se imediatamente que

$$\mathcal{W}\mathcal{W}^* = \mathcal{W}^*\mathcal{W} = nI,$$

onde I é a matriz identidade de ordem n. Particionada em blocos $m \times m$, $\mathcal{W}$ se escreve na forma

$$\mathcal{W} = \begin{bmatrix} \mathbf{W} & \mathbf{DW} \\ \mathbf{WD} & (-1)^m\mathbf{DWD} \end{bmatrix}, \tag{1.14}$$

onde

$$\mathbf{W} = \begin{bmatrix} I & 0 \end{bmatrix} \mathcal{W} \begin{bmatrix} I \\ 0 \end{bmatrix} \in \mathbb{M}_{m\times m}(\mathbb{C}),$$

e $\mathbf{D}$ é uma matriz diagonal cujos elementos $\mathbf{D}_{kk}$ são dados por

$$\mathbf{D}_{kk} = (-1)^k, \quad k \in \{0,1,2,\ldots,m-1\}. \tag{1.15}$$

$\mathcal{W}$ também pode ser expressa na seguinte forma particionada

$$\mathcal{W} = \begin{bmatrix} \mathcal{W}_{(\mathbf{1})} & \mathcal{W}_{(\mathbf{2})} \end{bmatrix},$$

onde

$$\mathcal{W}_{(\mathbf{1})} = \begin{bmatrix} \mathbf{W} \\ \mathbf{WD} \end{bmatrix}, \quad \mathcal{W}_{(\mathbf{2})} = \begin{bmatrix} \mathbf{DW} \\ (-1)^m\mathbf{DWD} \end{bmatrix}. \tag{1.16}$$

A proposição seguinte [11] nos diz que um certo tipo de *simetria* é típica das transformadas de Fourier de objetos reais 1-D.

Proposição 1.1 *Seja (f,F) um par de Fourier com $f \in \mathbb{C}^n$. Então f é real se, e somente se,*

$$F_{m+u} = \overline{F}_{m-u}, \quad u = 0,1,2,\ldots,m. \tag{1.17}$$

Como nosso objeto f é supostamente real, segue da proposição 1.1 que F_0 e F_m serão sempre números reais, e o vetor F poderá ser reescrito como

$$F = \begin{bmatrix} F_{(\mathbf{1})} \\ F_{(\mathbf{2})} \end{bmatrix}; \quad F_{(\mathbf{1})} = (F_0, F_1, \ldots, F_{m-1})^T; \quad F_{(\mathbf{2})} = (F_m, \overline{F}_{m-1}, \ldots, \overline{F}_1)^T. \tag{1.18}$$

Definição 1.2 *Seja F um vetor qualquer do espaço $\mathbb{C}^n$. Dizemos que F apresenta a simetria 1-D de Fourier, se as componentes de F satisfizerem as equações* (1.17).

Note que o vetor das amplitudes de Fourier

$$|F| := (|F_0|, |F_1|, \ldots, |F_{n-1}|)^T,$$

apresenta a *simetria* $|F_{m+u}| = |F_{m-u}|$.

Usando a representação polar em (1.9) e considerando a simetria 1-D de Fourier, o vetor F pode ser reescrito como

$$F = (\alpha_0 e^{i\phi_0}, \alpha_1 e^{i\phi_1}, \ldots, \alpha_{m-1} e^{i\phi_{m-1}}, \alpha_m e^{i\phi_m}, \alpha_{m-1} e^{-i\phi_{m-1}}, \ldots, \alpha_1 e^{-i\phi_1})^T \quad (1.19)$$

para

$$\alpha_u = |F_u|, \quad u = 0, 1, 2, \ldots, m. \quad (1.20)$$

Como F_0 e F_m são números reais, e $-\pi < \phi_u \leq \pi \ \ \forall\, u$, então

$$\phi_0, \phi_m \in \{0, \pi\}.$$

Consequentemente

$$\epsilon_u = e^{i\phi_u} = \pm 1, \text{ para } u = 0, m \quad (1.21)$$

e

$$F_0 = \epsilon_0 \alpha_0 = \pm \alpha_0, \quad F_m = \epsilon_m \alpha_m = \pm \alpha_m. \quad (1.22)$$

Observa-se que ϵ_u em (1.21) é o sinal de F_u ($\epsilon_u = \text{sgn}(F_u)$) para $u = 0, m$.

Desde que $F_0 = \sum_{j=0}^{n-1} f_j$, segue que

$$\text{se } f_j > 0 \ \ \forall\, j, \text{ então } \epsilon_0 = 1 \text{ e } \phi_0 = 0. \quad (1.23)$$

Por causa da simetria 1-D de Fourier, as amplitudes e as fases de F são completamente descritas pelos vetores

$$\alpha_F := (\alpha_0, \alpha_1, \ldots, \alpha_m)^T. \quad (1.24)$$

e

$$\phi_F := (\phi_0, \phi_1, \ldots, \phi_m)^T. \quad (1.25)$$

Assim, para o caso em que α_u e ϕ_u representam a amplitude e a fase de F_u, os vetores em (1.24) e (1.25) serão denominados *vetor amplitude* e *vetor fase* de F.

As condições satisfeitas em (1.22) nos permitem introduzir a noção dos toróides [73].

Definição 1.3 *Dados $\phi_0, \phi_m \in \{0, \pi\}$, o toróide $T_{(\phi_0,\phi_m)}$ é o subconjunto do $\mathbb{R}^{m+1}$ definido por*

$$T_{(\phi_0,\phi_m)} = \{(\phi_0; \theta; \phi_m)^T : \theta = (\theta_1, \theta_2, \ldots, \theta_{m-1})^T \in (-\pi, \pi]^{m-1}\}.$$

Dizemos que f está dentro do toróide $T_{(\phi_0,\phi_m)}$ se $\phi_F \in T_{(\phi_0,\phi_m)}$.

Cada um dos 4 toróides é topologicamente igual ao hipertoro S_1^{m-1}, onde S_1 é a esfera unitária. Se f for um sinal positivo, então resulta de (1.23) que f estará dentro de $T_{(0,0)}$ ou $T_{(0,\pi)}$ conforme se tenha $\epsilon_m = 1$ ou $\epsilon_m = -1$ respectivamente.

1.1.2 Forma matricial: caso 2-D

Para o caso bidimensional, as expressões (1.6) e (1.7) tornam-se equivalentes a

$$F = \mathcal{W}_1 f \mathcal{W}_2 \tag{1.26}$$

e

$$f = \frac{1}{n_1 n_2} \mathcal{W}_1^* F \mathcal{W}_2^*, \tag{1.27}$$

onde f é, agora, a matriz dada em (1.3), e F é a matriz complexa, $n_1 \times n_2$, cujos pixels são os valores, F_{uv}, $(u,v) \in \Omega = [0, n_1 - 1] \times [0, n_2 - 1]$. A matriz das amplitudes de Fourier será denotada por $|F|$. Assim

$$|F| = \left[\, |F_{uv}| \,\right]_{(u,v)\in\Omega}.$$

Para cada $j = 1, 2$, $\mathcal{W}_j$ $(\in \mathbb{M}_{n_j \times n_j}(\mathbb{C}))$[3] é a matriz de Fourier dada em (1.14) quando $m = m_j$ e $\omega = \omega_j = \exp(\pi i / m_j)$.

Similarmente ao caso 1-D, temos a seguinte proposição [11] que descreve a *simetria 2-D de Fourier* da transformada de Fourier de uma imagem real.

Proposição 1.2 *Seja f uma imagem complexa, $n_1 \times n_2$. Então f é uma imagem real se e somente se*

$$\overline{F}_{uv} = \begin{cases} F_{uv} & (u,v) \in \{0, m_1\} \times \{0, m_2\}, \\ F_{u,n_2-v} & u \in \{0, m_1\}, \;\; v \in \{1, \ldots, m_2 - 1\} \\ F_{n_1-u,v} & u \in \{1, \ldots, m_1 - 1\}, \;\; v \in \{0, m_2\} \\ F_{n_1-u,n_2-v} & u \in \{1, \ldots, m_1 - 1\}, \;\; v \notin \{0, m_2\}. \end{cases} \tag{1.28}$$

A simetria em (1.28) implica que F pode ser reescrita como

$$F = \begin{bmatrix} \boxed{F_{00}} & F_{01} & \cdots & F_{0k} & \longrightarrow & \boxed{F_{0m_2}} & \longleftarrow & \overline{F}_{0k} & \cdots & \overline{F}_{01} \\ F_{10} & F_{11} & \cdots & F_{1k} & \cdots & F_{1m_2} & \cdots & F_{1,n_2-k} & \cdots & F_{1,n_2-1} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ F_{j0} & F_{j1} & \cdots & F_{jk} & \cdots & F_{jm_2} & \cdots & F_{j,n_2-k} & \cdots & F_{j,n_2-1} \\ \downarrow & \vdots & & \vdots & \searrow & \downarrow & \swarrow & \vdots & & \vdots \\ \boxed{F_{m_1 0}} & F_{m_1 1} & \cdots & F_{m_1 k} & \longrightarrow & \boxed{F_{m_1 m_2}} & \longleftarrow & \overline{F}_{m_1 k} & \cdots & \overline{F}_{m_1 1} \\ \uparrow & \vdots & & \vdots & \nearrow & \uparrow & \nwarrow & \vdots & & \vdots \\ \overline{F}_{j0} & \overline{F}_{j,n_2-1} & \cdots & \overline{F}_{j,n_2-k} & \cdots & \overline{F}_{jm_2} & \cdots & \overline{F}_{jk} & \cdots & \overline{F}_{j1} \\ \vdots & \vdots & & \vdots & & \vdots & & \vdots & & \vdots \\ \overline{F}_{10} & \overline{F}_{1,n_2-1} & \cdots & \overline{F}_{1,n_2-k} & \cdots & \overline{F}_{1m_2} & \cdots & \overline{F}_{1k} & \cdots & \overline{F}_{11} \end{bmatrix} \tag{1.29}$$

[3] O leitor não deverá confundir, por exemplo, a notação $\mathcal{W}_1$ em (1.26) com a $\mathcal{W}_{(1)}$ dada em (1.16). Em (1.26), $\mathcal{W}_1$ representa a matriz de Fourier correspondente ao inteiro m_1, enquanto em (1.16), $\mathcal{W}_{(1)}$ representa um bloco da matriz de Fourier correspondente ao inteiro m.

Façamos alguns comentários para entendermos melhor a simetria 2-D de Fourier da matriz F em (1.29). Nos quadrados destacam-se os elementos da matriz que necessariamente são reais. Um par de setas orientadas em sentidos opostos indica que os elementos posicionados em suas origens, além de serem complexos conjugados, estão a uma mesma distância do elemento para o qual o par aponta. Assim, desde que $|z| = |\overline{z}|$ e $\arg(\overline{z}) = -\arg(z)$, segue que as amplitudes, α_{uv}, e as fases, ϕ_{uv}, de F são completamente descritas pelos conjuntos

$$\{\alpha_{uv} : (u,v) \in \Lambda\} \text{ e } \{\phi_{uv} : (u,v) \in \Lambda\}, \tag{1.30}$$

onde

$$\Lambda := \Lambda_1 \cup \Lambda_2 \cup \Lambda_3, \tag{1.31}$$

e

$$\begin{array}{rcl} \Lambda_1 & := & \{(0,v) : v \in \{0,1,\ldots,m_2\}\}, \\ \Lambda_2 & := & \{(u,v) : u \in \{1,2,\ldots,m_1-1\},\ v \in \{0,1,\ldots,n_2-1\}\}, \\ \Lambda_3 & := & \{(m_1,v) : v \in \{0,1,\ldots,m_2\}\}. \end{array} \tag{1.32}$$

Tais fases e amplitudes referem-se aos elementos de F interiores aos quadrados e aos elementos interiores à linha poligonal fechada.

Podemos ordenar as amplitudes em (1.30) e obter o vetor amplitude de F,

$$\alpha_F = (\alpha_{00}, \alpha_{01}, \ldots, \alpha_{m_1 m_2})^T \in \mathbb{R}_+^{2m_1 m_2+2}, \tag{1.33}$$

de modo que as m_2+1 primeiras coordenadas de α_F sejam os m_2+1 primeiros elementos da primeira linha da matriz $|F|$, ordenados da esquerda para a direita; os $(m_1-1)\times n_2$ elementos seguintes de α_F sejam os $(m_1-1)\times n_2$ elementos de $|F|$ compreendidos entre a segunda e a m_1−ésima linha, ordenados de acordo com a ordem lexográfica; e, finalmente, os m_2+1 elementos restantes de α_F sejam os m_2+1 primeiros elementos da (m_1+1)−ésima linha de $|F|$, também ordenados da esquerda para a direita.

De maneira análoga ordena-se as fases de F em (1.30) e obtém-se o vetor fase de F,

$$\phi_F = (\phi_{00}, \phi_{01}, \ldots, \phi_{m_1 m_2})^T \in (-\pi, \pi]^{2m_1 m_2+2}. \tag{1.34}$$

Similarmente ao caso 1-D,

$$\phi_{00}, \phi_{0m_2}, \phi_{m_1 0}, \phi_{m_1 m_2} \in \{0, \pi\}.$$

Consequentemente,

$$\epsilon_{uv} := e^{i\phi_{uv}} = \pm 1, \quad F_{uv} = \epsilon_{uv}\alpha_{uv} = \pm\alpha_{uv}, \quad (u,v) \in \{0,m_1\} \times \{0,m_2\}. \tag{1.35}$$

Note que $\epsilon_{uv} = \operatorname{sgn}(F_{uv}), \quad (u,v) \in \{0,m_1\} \times \{0,m_2\}$.

Desde que $F_{00} = \sum_j \sum_k f_{jk}$, segue que

$$\text{se } f_{jk} > 0 \ \ \forall\, (j,k) \in \Omega \text{ então } \epsilon_{00} = 1 \text{ e } \phi_{00} = 0. \tag{1.36}$$

Consideremos a ordem dos índices $(u,v) \in \Lambda$ estabelecida na formação do vetor ϕ_F. Denotemos por $\phi_{(1)} \in (-\pi,\pi]^{m_2-1}$, $\phi_{(2)} \in (-\pi,\pi]^{(m_1-1)n_2}$ e $\phi_{(3)} \in (-\pi,\pi]^{m_2-1}$ os subvetores de ϕ_F determinados pelas componentes de ϕ_F compreendidas entre ϕ_{00} e ϕ_{0m_2}; ϕ_{0m_2} e ϕ_{m_10}; e ϕ_{m_10} e $\phi_{m_1m_2}$ respectivamente. Então

$$\phi_F = (\phi_{00}; \phi_{(1)}; \phi_{0m}; \phi_{(2)}; \phi_{m0}; \phi_{(3)}; \phi_{mm})^T. \tag{1.37}$$

Similarmente à definição 1.3 [73] temos a

Definição 1.4 *Dados $\phi_{00}, \phi_{0m_2}, \phi_{m_10}, \phi_{m_1m_2} \in \{0,\pi\}$, o toróide $T_{(\phi_{00},\phi_{0m_2},\phi_{m_10},\phi_{m_1m_2})}$ é o subconjunto do $\mathbb{R}^{2m_1m_2+2}$ definido por*

$$\{(\phi_{00};\theta_{(1)};\phi_{0m_2};\theta_{(2)};\phi_{m_10};\theta_{(3)};\phi_{m_1m_2})^T : \theta_{(1)},\theta_{(3)} \in (-\pi,\pi]^{m_2-1}; \theta_{(2)} \in (-\pi,\pi]^{(m_1-1)n_2}\}.$$

Dizemos que f está dentro do toróide $T_{(\phi_{00},\phi_{0m_2},\phi_{m_10},\phi_{m_1m_2})}$ se $\phi_F \in T_{(\phi_{00},\phi_{0m_2},\phi_{m_10},\phi_{m_1m_2})}$.

Cada um dos 16 toróides é topologicamente igual ao hipertoro $S_1^{2(m_1m_2-1)}$.

1.2 Os problemas da fase e da autocorrelação

Nesta seção formularemos os problemas da fase e da autocorrelação, e discutiremos assuntos relevantes tais como a caracterização das amplitudes.

A seguir, redefinimos o suporte de um objeto, da mesma maneira como o fizemos na seção 0.1, desta vez incluindo o caso 1-D.

Definição 1.5 *Seja $f = (f_j)_{j\in\Omega}$ um objeto real, onde*

$$\begin{array}{rcll} \Omega & = & [0, n-1] & \text{(caso 1-D)}, \\ \Omega & = & [0, n_1-1] \times [0, n_2-1] & \text{(caso 2-D)}. \end{array}$$

Dizemos que f satisfaz a restrição de suporte se

$$f_j = 0, \quad \forall j \in \Omega \setminus \mathcal{S}, \tag{1.38}$$

onde $\mathcal{S}$ (o suporte de f) é dado por

$$\begin{array}{rcll} \mathcal{S} & = & [0, m-1] & \text{(caso 1-D), ou} \\ \mathcal{S} & = & [0, m_1-1] \times [0, m_2-1] & \text{(caso 2-D)}, \end{array}$$

Queremos estudar a seguinte versão do problema:

O Problema da Fase: *Dado* $\alpha_u \geq 0$, $u \in \Omega$, *encontrar* f *que satisfaz:*

$$\begin{array}{ll} \text{(i)} & |F_u| = \alpha_u, \quad \textit{onde} \quad F = \mathcal{F}(f), \\ \text{(ii)} & \textit{a condição de suporte } (1.38) \end{array} \tag{1.39}$$

O problema da fase pode ser estudado inteiramente no domínio do objeto, usando autocorrelação [82] (pp. 235-242), [67]. Dados dois objetos reais $f = (f_j)_{j\in\Omega}$ e $g = (g_j)_{j\in\Omega}$, definimos a *correlação discreta* entre eles por

$$f_j \star g_j := \sum_{i\in\Omega} f_i g_{(j+i)(\text{mod } n)}{}^4, \quad j \in \Omega. \tag{1.40}$$

A autocorrelação discreta nada mais é que

$$f_j \star f_j = f_j * f_{-j}, \tag{1.41}$$

onde $f_j * g_j$ (a *convolução discreta* entre f e g) é definida por

$$f_j * g_j = \sum_{i\in\Omega} f_i g_{(j-i)(\text{mod } n)}, \tag{1.42}$$

Similarmente ao teorema da convolução contínua dado na seção 0.2, temos o teorema da convolução discreta [33]

$$\mathcal{F}[f_j * g_j] = F_u G_u. \tag{1.43}$$

Também continua válida no caso discreto a Fórmula de Inversão Alternada

$$\mathcal{F}(f_{-j}) = [\mathcal{F}(f_j)]^* \tag{1.44}$$

que, em outras palavras, significa que toda imagem possui o mesmo módulo de Fourier da sua gêmea.

O Problema da Autocorrelação : *Dado* $\beta_u \geq 0$, $u \in \Omega$, *encontrar* f *que satisfaz:*

$$\begin{array}{ll} \text{(i)} & f_j \star f_j = \beta_u, \\ \text{(ii)} & \textit{a condição de suporte } (1.38) \end{array} \tag{1.45}$$

Os problemas (1.39) e (1.45) são equivalentes quando $\mathcal{F}(\beta_u) = \alpha_u^2$ pois, de acordo com (1.41), (1.43) e (1.44),

$$\mathcal{F}(f_j \star f_j) = |F_u|^2. \tag{1.46}$$

[4] Em geral, o termo $g_{k(\text{mod } n)}$ significa que $g_k = g_r$, onde r é o resto da divisão de k por n. Note que no caso 2-D, $g_{(j,k)(\text{mod } n_1,\text{mod } n_2)}$ nos diz que $g_{jk} = g_{rs}$, com r e s os restos respectivos das divisões de j por n_1 e k por n_2. Se $n_1 = n_2 := n$, adotaremos, para o caso 2-D, a notação mais simples $g_{jk(\text{mod } n)}$

O suporte de $f_j \star f_j$ é tipicamente o dobro do suporte de f_j. Então, para evitarmos o fenômeno chamado *aliasing* [11], [82] (pp. 235) nos cálculos de $|F_u|^2$ e, assim, obtermos unicidade na solução do problema (1.45), é necessário que consideremos objetos f_j que satisfaçam a restrição de suporte (1.38).

Algumas questões surgem de imediato, como a unicidade da solução para (1.39) e a caracterização das amplitudes [5], i.e., as condições necessárias e suficientes para a existência de uma solução para o problema (1.39). Também pergunta-se quais as condições adicionais para que a solução também satisfaça a *restrição de positividade*

$$f_x \geq 0 \quad \forall x \in \mathcal{S}. \tag{1.47}$$

Vimos na seção 0.4 que, no caso 2-D, a solução do problema (1.39) é tipicamente única a menos das ambiguidades triviais. Nas próximas seções discutiremos, então, a não -unicidade para o caso 1-D e a caracterização das amplitudes no problema da fase.

1.3 A falta de unicidade para o caso 1-D

Como já dissemos na introdução , é bem conhecido que o problema inverso (1.39) não apresenta solução única para o caso unidimensional. O número de soluções possíveis é finito, igual a no máximo 2^m, e podemos analisar essas soluções usando a teoria de variáveis complexas. Nossa discussão será baseada em [12] e [82], pp. 221.

Dado um sinal real f como em (3.1), sua transformada de Fourier F pode ser reescrita em termos de um polinômio, $P_F(z)$, avaliado sobre determinados pontos da esfera unitária: as raízes da unidade de ordem n, i.e.,

$$F_u = \mathcal{F}[f_x] = \sum_{x=0}^{m-1} f_x \exp(-2\pi i u \cdot x/n) = P_F(\omega^{-u}), \quad \omega = \exp(\pi i/m), \tag{1.48}$$

para

$$P_F(z) = \sum_{x=0}^{m-1} f_x \, z^x. \tag{1.49}$$

A expressão em (1.48) mostra que cada pixel F_u do vetor F é dado pelo polinômio P_F avaliado na raíz da unidade correspondente, ω^{-u}, $u = 0, 1, 2, \ldots, n-1$. O polinômio P_F é também chamado a *transformada-z* de f_x. Assim, desde que os coeficientes de P_F em (1.49) são dados pelos pixels de f, fica evidente a relação biunívoca entre os polinômios de grau $m-1$ e os sinais reais de suporte de tamanho m.

[5] Estas metas parecem ser atingíveis, porém não existe na literatura nenhum resultado que as comprove para o caso geral.

Agora, pelo teorema fundamental da álgebra, o polinômio em (1.49) pode ser fatorado completamente sobre o corpo dos números complexos, i.e.,

$$P_F(z) = c_0 \prod_{a_k \in \Gamma} (z - a_k) \cdot \prod_{z_k \in \Omega} (z - z_k)(z - z_k^*). \tag{1.50}$$

onde c_0 é uma certa constante real,

$$\Gamma \equiv \Gamma_F = \{a_1, a_2, \ldots, a_p\}$$

é o conjunto de todas as raízes reais e

$$\Omega \equiv \Omega_F = \{z_1, z_2, \ldots, z_q\}$$

o de todas as raízes complexas de P_F que têm a parte imaginária positiva. Então o conjunto de todas as raízes de P_F é $Z(P_F) = \Gamma \cup \Omega \cup \Omega^*$. Segue imediatamente que $p + 2q = m - 1$.

Dado um número real ou complexo, w, definimos o *flipping* de w (em torno do círculo unitário) como sendo a transformação que permuta w com o seu inverso multiplicativo $1/w$, i.e.,

$$\text{flip(w)} = 1/\text{w}.$$

Se fizermos um flipping em um número finito de raízes não unitárias de P_F em torno do círculo unitário, obteremos um novo polinômio, $P_{\tilde{F}}$ cuja transformada de Fourier associada, $\tilde{F}$, ainda possui a mesma magnitude, i.e., $|\tilde{F}_u| = |F_u|$. Isto equivale à multiplicação

$$P_{\tilde{F}}(z) = P_F(z)G(z), \tag{1.51}$$

onde

$$G(z) = \prod_{a_k \in \Gamma_1} \frac{1 - za_k}{z - a_k} \cdot \prod_{z_k \in \Omega_1} \frac{1 - zz_k^*}{z - z_k} \cdot \frac{1 - zz_k}{z - z_k^*}, \tag{1.52}$$

para alguns subconjuntos $\Gamma_1 \subseteq \Gamma$, $\Omega_1 \subseteq \Omega$. É imediata a verificação de que

$$|G(z)| = 1; \quad \forall \ |z| = 1. \tag{1.53}$$

Com efeito, dado $w \in \mathbb{C}$, se $|z| = 1$ então

$$\left|\frac{1 - zw^*}{z - w}\right| = \left|\frac{z(z^* - w^*)}{z - w}\right| = |z| = 1.$$

Assim, (1.51) torna-se equivalente a

$$\begin{aligned} P_{\tilde{F}}(z) \ = \ & c_0 \prod_{a_k \in \Gamma \setminus \Gamma_1} (z - a_k) \cdot \prod_{a_k \in \Gamma_1} (1 - za_k) \cdot \\ & \prod_{z_k \in \Omega \setminus \Omega_1} (z - z_k)(z - z_k^*) \cdot \prod_{z_k \in \Omega_1} (1 - zz_k)(1 - zz_k^*). \end{aligned} \tag{1.54}$$

Note que (1.54) nos diz que todas as raízes reais contidas no subconjunto Γ_1, assim como todas as raízes complexas de Ω_1 e seus conjugados correspondentes de Ω_1^*, foram flipadas em torno do círculo unitário. O Polinômio $P_{\tilde{F}}$, assim obtido, é um polinômio de coeficientes reais de grau $m-1$, i.e.,

$$P_{\tilde{F}}(z) = \sum_{x=0}^{m-1} \tilde{f}_x \, z^x, \tag{1.55}$$

cujas raízes são dadas por todas as raízes flipadas de P_F e também por todas aquelas que não foram flipadas. Em outras palavras, (1.54) nos diz que

$$\forall \; z_k \in Z(P_{\tilde{F}}), \;\; \exists \; w_l \in Z(P_F) \;\; \text{tal que} \;\; \text{ou } z_k = w_l, \; \text{ou } z_k = 1/w_l. \tag{1.56}$$

Diremos então que dois sinais f e $\tilde{f}$ são *flipping relacionados* se suas transformadas-z associadas P_F e $P_{\tilde{F}}$ satisfazem (1.56).

Observe que para garantirmos que os coeficientes de $P_{\tilde{F}}$ sejam reais é necessário que, ao fliparmos algumas das raizes complexas de P_F, também o façamos aos seus conjugados respectivos.

Observe que, em decorrência de (1.53),

$$|\tilde{F}_u| = |P_{\tilde{F}}(\omega^{-u})| = |P_F(\omega^{-u})||G(\omega^{-u})| = |P_F(\omega^{-u})| = |F_u|,$$

o que mostra que f_x e $\tilde{f}_x$ são duas[6] soluções reais distintas, com a restrição de suporte, para o problema inverso (1.39).

Reciprocamente, observamos segundo (1.41) e (1.46) que a transformada-z da autocorrelação $f_j \star f_j$ é $P_F(z) \cdot P_{\tilde{F}}(z)$, onde $\tilde{F}_u = \mathcal{F}(f_{-j})$, e seus zeros são os zeros de $P_F(z)$ e seus *flipados*. Usando essa observação podemos verificar [12] que os únicos sinais reais recuperáveis de $|F|$ são o próprio f, os sinais obtidos de f por flipping e seus negativos, no total não mais do que 2^m soluções.

1.4 As identidades das amplitudes

J.L.C. Sanz afirma em seu paper [74] que certas condições necessárias para que o problema da fase (1.39) admita solução são as de que as amplitudes quadradas, α_u^2, satisfaçam um certo conjunto, $\mathcal{P}$, de equações polinomiais. Além disso ele conjetura que **um problema desafiador é o de encontrar tais equações cuja tarefa pode ser extremamente complicada.** Na verdade elas nunca foram exibidas e até o momento não há na literatura qualquer registro de tais equações.

[6] Em outras palavras, o que acabamos de verificar é que as equações (1.51) e (1.53) mais o teorema da convolução, implicam que é sempre possível obter um novo sinal através da convolução do sinal original com um outro sinal arbitrário cuja transformada de Fourier tenha módulo unitário.

Nesta seção, **nós resolvemos parte desse problema em aberto**, exibindo para ambos os casos, 1-D e 2-D, um subconjunto $\mathcal{P}_1 \subset \mathcal{P}$, formado por identidades polinomiais lineares nos α_u^2. Nós acreditamos que a caracterização completa das amplitudes α_u (ou equivalentemente, as condições necessárias e suficientes que o vetor das amplitudes, α_F, deve satisfazer para que (1.39) admita solução (única)), é de natureza ainda mais complexa e é composta por 3 tipos de restrições : (1) as identidades do conjunto $\mathcal{P}$; (2) outras identidades polinomiais nas variáveis α_u, mas que não são polinomiais em α_u^2 (denominaremos o conjunto destas identidades por $\mathcal{A}$); (3) inequações nas variáveis α_u (cujo conjunto será denotado por $\mathcal{D}$). Infelizmente a caracterização completa das amplitudes para o problema generalizado da fase é possível somente para sinais e imagens de tamanho pequeno, pois encontrar as identidades do conjunto $(\mathcal{P} \setminus \mathcal{P}_1) \cup \mathcal{A}$ mais as desigualdades de $\mathcal{D}$ é uma tarefa extremamente difícil. Para o caso 1-D, nós damos a caracterização completa para $m = 2$ e $m = 3$. Já para $m = 4$ o processo para encontrar todas as desigualdades de $\mathcal{D}$ torna-se extremamente complicado. O mesmo acontece para o caso 2-D, onde conseguimos dar a caracterização completa apenas para $m_1 = m_2 = 2$ e $m_1 = 2$, $m_2 = 3$. É impressionante como as identidades do conjunto $\mathcal{A}$ são bem mais complicadas para $m_1 = 2$, $m_2 = 3$ do que para $m_1 = m_2 = 2$.

Em todas as aplicações os dados das magnitudes α_u em (1.39) são corrompidos por ruídos pois na prática eles são medidos através de equipamentos cuja aferição está condicionada a interferências externas. Assim, torna-se necessário saber quais os efeitos produzidos quando ruídos são acrescentados no sistema (1.39).

É fácil ver que o problema de recuperação da fase a partir das amplitudes é localmente *mal condicionado* (ou *mal posto*) [42] (pp. 9) no sentido de que pequenas perturbações nos dados das amplitudes de um problema factível produzem um problema não factível. Em outras palavras, como Sanz afirma em seu paper [74], a probabilidade de construir fases consistentes para amplitudes com ruídos é nula.

Diversos métodos para resolver o problema da fase com ruídos foram propostos, dentre eles se destacam os métodos que utilizam técnicas de regularização, [34], [42], [6], [80], [22], [66], [79], e os métodos iterativos de Fienup [25], [26], [81]. No capítulo 3 nós apresentamos um novo método para o problema da fase.

1.4.1 Identidade das amplitudes para o caso 1-D:

Teorema 1.1 *Seja* $f = (f_{(\mathbf{1})}^T, f_{(\mathbf{2})}^T)^T$ *um objeto real unidimensional e* F *sua* DFT. *Suponha que as amplitudes de* F *são dadas por* $\alpha_u = |F_u|$ $u = 0, 1, \ldots, m$. *Então*

$$\alpha_0^2 + 2\sum_{u=1}^{m-1}(-1)^k\alpha_u^2 + (-1)^m\alpha_m^2 = 2nf_{(\mathbf{2})}^T f_{(\mathbf{1})}. \tag{1.57}$$

Demonstração : Usando a expressão dada em (1.7) para calcular a transformada inversa da quantidade $|F_u|^2$, obtém-se

$$\mathcal{F}^{-1}[|F_u|^2] = \frac{1}{n}\sum_{u=0}^{n-1} |F_u|^2\, \omega^{uj}, \tag{1.58}$$

onde $\omega = \exp(\pi i/m)$. Por outro lado segue de (1.46) e (1.40) que

$$\mathcal{F}^{-1}[|F_u|^2] = \sum_{i=0}^{n-1} f_i f_{(j+i)(\mathrm{mod}\, n)}, \quad 0 \leq j \leq n-1. \tag{1.59}$$

Comparando as expressões de segundo membro de (1.58) e (1.59) obtemos

$$\frac{1}{n}\sum_{u=0}^{n-1} |F_u|^2\, \omega^{uj} = \sum_{i=0}^{n-1} f_i f_{(j+i)(\mathrm{mod}\, n)}, \quad 0 \leq j \leq n-1. \tag{1.60}$$

Atribuindo m à variável j em (1.60) temos

$$\frac{1}{n}\sum_{u=0}^{n-1} (-1)^u |F_u|^2 = \sum_{i=0}^{n-1} f_i f_{(m+i)(\mathrm{mod}\, n)}. \tag{1.61}$$

Verifica-se imediatamente que o segundo membro de (1.61) é igual a $2(f_0 f_m + f_1 f_{m+1} + \cdots + f_{m-1} f_{n-1}) \equiv 2f_{(\mathbf{1})}^T f_{(\mathbf{2})}$. Por outro lado, devido à simetria 1-D do vetor $|F|$ (veja definição 1.2), o primeiro membro de (1.61) torna-se idêntico a

$$\frac{1}{n}[\alpha_0^2 + 2\sum_{u=1}^{m-1}(-1)^u \alpha_u^2 + (-1)^m \alpha_m^2] \qquad \blacksquare$$

Corolário 1.1 (Identidade 1-D das amplitudes) *Seja f um sinal real que satisfaz a restrição de suporte* (1.38). *Então as amplitudes de F satisfazem a identidade*

$$\alpha_0^2 + 2\sum_{u=1}^{m-1}(-1)^u \alpha_u^2 + (-1)^m \alpha_m^2 = 0. \tag{1.62}$$

Demonstração : Basta tomarmos $f_{(\mathbf{2})} = 0$ no segundo membro de (1.57). $\blacksquare$

Chamaremos a equação (1.62) de *identidade 1-D das amplitudes*. Nota-se que a identidade 1-D das amplitudes é uma equação polinomial de grau 1 nas variáveis $\alpha_0^2, \alpha_1^2, \ldots, \alpha_m^2$. Diremos que (1.62) é uma *identidade linear* nas variáveis $\alpha_0^2, \alpha_1^2, \ldots, \alpha_m^2$.

Será conveniente expressarmos a fórmula de autocorrelação (1.60) quando f satisfizer a restrição de suporte. Assim, se $f_{(\mathbf{2})} = 0$ e levando-se em conta a simetria 1-D de Fourier do vetor $|F|$, (1.60) torna-se equivalente ao sistema

$$\frac{1}{n}[\alpha_0^2 + 2\sum_{u=1}^{m-1} \alpha_u^2 \mathrm{Re}(\omega^{ju}) + (-1)^j \alpha_m^2] = \sum_{i=0}^{m-1-j} f_i f_{j+i}, \quad 0 \leq j \leq m-1, \tag{1.63}$$

mais a identidade 1-D das amplitudes

$$\alpha_0^2 + 2\sum_{u=1}^{m-1} (-1)^u \alpha_u^2 + (-1)^m \alpha_m^2 = 0.$$

Note que (1.63) representa um sistema polinomial nas variáveis $f_0, f_1, \ldots, f_{m-1}$. Nieto-Vesperinas [67] faz uso do método de Levenberg-Marquardt [51], [59] para resolver esse sistema não linear sobredeterminado.

Será conveniente darmos uma demonstração alternativa do teorema 1.1.

Demonstração alternativa do teorema 1.1*:*

A prova de (1.57) é uma consequência imediata das igualdades

$$|F_0|^2 + 2\sum_{k=1}^{m-1} (-1)^k |F_k|^2 + (-1)^m |F_m|^2 = F^* \mathbf{\Delta} F = 2n f_{(\mathbf{2})}^T f_{(\mathbf{1})}, \tag{1.64}$$

onde $\mathbf{\Delta}$ é a matriz diagonal $n \times n$ dada por $\mathbf{\Delta}_{jj} = (-1)^j$, $j = 0, 1, \ldots, n-1$.

Para provarmos a primeira igualdade em (1.64), substituímos F em $F^* \mathbf{\Delta} F$ pelo vetor das suas componentes em (1.12) e em seguida aplicamos a relação de simetria (1.17). De fato

$$\begin{aligned} F^* \mathbf{\Delta} F &= \sum_{u=0}^{n-1} \overline{F}_u [(-1)^u F_u] = \sum_{u=0}^{n-1} (-1)^u |F_u|^2 \\ &= |F_0|^2 + 2\sum_{u=1}^{m-1} (-1)^u |F_u|^2 + (-1)^m |F_m|^2. \end{aligned}$$

Por outro lado, substituindo $\mathbf{\Delta}$ em $F^* \mathbf{\Delta} F$ pela sua forma particionada por blocos

$$\mathbf{\Delta} = \begin{bmatrix} \mathbf{D} & 0 \\ 0 & (-1)^m \mathbf{D} \end{bmatrix},$$

onde $\mathbf{D}$ é a matriz diagonal dada em (1.15) e, também, trocando F pelo segundo membro de (1.10), resulta que

$$\begin{aligned} F^* \mathbf{\Delta} F &= f^T (\mathcal{W}^* \mathbf{\Delta} \mathcal{W}) f = \begin{bmatrix} f_{(\mathbf{1})}^T & f_{(\mathbf{2})}^T \end{bmatrix} \begin{bmatrix} 0 & n\mathbf{I}_m \\ n\mathbf{I}_m & 0 \end{bmatrix} \begin{bmatrix} f_{(\mathbf{1})} \\ f_{(\mathbf{2})} \end{bmatrix} \\ &= 2n f_{(\mathbf{1})}^T f_{(\mathbf{2})}, \end{aligned}$$

o que prova a segunda igualdade em (1.64). ■

Como veremos mais adiante, resultados análogos ao teorema 1.1 e ao corolário 1.1 serão obtidos para o caso 2-D. Entretanto, o equivalente 2-D ao corolário 1.1 não nos dará todas as identidades lineares nas amplitudes quadradas que conhecemos para o caso 2-D, mas somente uma identidade. As outras identidades serão obtidas através da própria expressão de autocorrelação para o caso 2-D. Vale ressaltar ainda que estas identidades, obtidas da autocorrelação, não são ainda a versão 2-D das identidades das amplitudes. Na verdade, as *identidades 2-D das amplitudes* serão obtidas a partir da generalização 2-D do argumento utilizado na demonstração alternativa do teorema 1.1.

Lema 1.1 *Seja* $f = [\ f_0 \quad f_1 \quad \ldots \quad f_{m-1} \quad 0 \quad 0 \quad \ldots \quad 0\ ]^T \in \mathbb{R}^n$. *Se* $\alpha_u = |F_u|$, $u = 0, 1, \ldots, m$, *então as componentes do suporte de* f *satisfazem o sistema*

$$\alpha_u^2 - \frac{1}{n}||F||^2 = 2\sum_{p=0}^{m-2}\sum_{q=1}^{m-1-p} f_p f_{p+q}\mathrm{Re}(\omega^{uq}), \quad 0 \leq u \leq m. \tag{1.65}$$

Além disso,

$$\sum_{j=0}^{m-1} f_j = F_0, \quad \sum_{j=0}^{m-1}(-1)^j f_j = F_m. \tag{1.66}$$

Demonstração : Segue de (1.10) que

$$F_u = [\mathcal{W}]_{(u)} f = \sum_{s=0}^{m-1} \omega^{-us} f_s = \sum_{s=0}^{m-1} f_s \cos\frac{\pi us}{m} - i\sum_{s=0}^{m-1} f_s \,\mathrm{sen}\frac{\pi us}{m}. \tag{1.67}$$

Elevando ambos os lados de (1.67) ao quadrado, desenvolvendo os termos quadráticos e, em seguida, aplicando a identidade trigonométrica $\cos a\ \cos b + \mathrm{sen}a\ \mathrm{sen}b = \cos(a-b)$, obtemos

$$|F_u|^2 = 2\sum_{p=0}^{m-2}\sum_{q=1}^{m-1-p} f_p f_{p+q}\mathrm{Re}(\omega^{uq}) + ||f||^2, \quad 0 \leq u \leq m. \tag{1.68}$$

Aplicando a fórmula de Parseval, $||f||^2 = (1/n)||F||^2$, em (1.68) obtém-se imediatamente (1.65). Em particular, fazendo $u = 0$ e $u = m$ em (1.67), obtém-se (1.66) ■

O próximo lema trás resultados para $m = 2, 3$.

Lema 1.2 *Sejam* $f = [\ f_0 \quad f_1 \quad \ldots \quad f_{m-1} \quad 0 \quad 0 \quad \ldots \quad 0\ ]^T \in \mathbb{R}^n$; $\alpha_u = |F_u|\ \forall u$ *e* $\epsilon_u = \text{sgn}(F_u)$, *para* $u = 0, m$.

(a) *Se* $m = 2$, *então*

$$f_0 = (\epsilon_0\alpha_0 + \epsilon_2\alpha_2)/2, \quad f_1 = (\epsilon_0\alpha_0 - \epsilon_2\alpha_2)/2. \tag{1.69}$$

(b) *Se* $m = 3$, *então*

$$(f_0 = r^+,\ f_1 = r,\ f_2 = r^-) \tag{1.70}$$

ou

$$(f_0 = r^-,\ f_1 = r,\ f_2 = r^+), \tag{1.71}$$

onde

$$r^{\pm} = [6(\epsilon_0\alpha_0 + \epsilon_3\alpha_3) \pm \sqrt{\Delta}]/24,$$

$$r = (\epsilon_0\alpha_0 - \epsilon_3\alpha_3)/2,$$

$$\Delta = 12[16\alpha_2^2 - (\epsilon_0\alpha_0 - 3\epsilon_3\alpha_3)^2] \geq 0.$$

Demonstração : Veja Apêndice A.

Os próximos resultados exibem as condições necessárias e suficientes para a existência de solução *positiva* do problema da fase (1.39), para os casos particulares $m = 2$ e $m = 3$.

Teorema 1.2 *Sejam* α_u, $u = 0, 1, 2$, *números reais positivos.*

(a) *Existe um sinal real* $f = \begin{bmatrix} f_0 & f_1 & 0 & 0 \end{bmatrix}^T$, *tal que* $|F_u| = \alpha_u$ *se, e somente se,* $\{\alpha_u\}$ *satisfaz*

(i) *a identidade 1-D das amplitudes* $\alpha_0^2 - 2\alpha_1^2 + \alpha_2^2 = 0$.

(b) *Existe um sinal real* $f = \begin{bmatrix} f_0 & f_1 & 0 & 0 \end{bmatrix}^T$ *com* $f_0, f_1 > 0$ *e tal que* $|F_u| = \alpha_u$, *se, e somente se,* $\{\alpha_u\}$ *satisfaz* (i) *e*

(ii) $\alpha_0 > \alpha_2$.

Demonstração : A prova de que as amplitudes satisfazem (i) segue da aplicação do corolário 1.1 quando $m = 2$. Reciprocamente sejam α_u, $u = 0, 1, 2$, números reais positivos satisfazendo (i). Tomemos $f_0 = (\epsilon_0\alpha_0 + \epsilon_2\alpha_2)/2$, $f_1 = (\epsilon_0\alpha_0 - \epsilon_2\alpha_2)/2$, para $\epsilon_0, \epsilon_2 \in \{\pm 1\}$. Substituindo os valores de f_0 e f_1 na expressão (1.68), para $u = 0, 1, 2$, e usando a identidade 1-D das amplitudes (i), conclui-se facilmente que $|F_u| = \alpha_u \; \forall u$.

(b) Suponhamos agora que $f = \begin{bmatrix} f_0 & f_1 & 0 & 0 \end{bmatrix}^T$, com $f_0, f_1 > 0$ e $F_0 = \epsilon_0\alpha_0$. Pela parte (a), segue que α_u satisfaz (i). Segue de (1.23) que $\epsilon_0 = 1$. Pela parte (a) do lema 1.2, f_0 e f_1 são dados como em (1.69), para $\epsilon_0 = 1$. Então $f_0, f_1 > 0 \Rightarrow \alpha_0 \pm \alpha_2 > 0 \Rightarrow \alpha_0 > \pm\alpha_2 \Rightarrow \alpha_0 > \alpha_2$. Reciprocamente suponhamos (i) e (ii). Então basta escolhermos f_0 e f_1 como em (1.69) para $\epsilon_0 = 1$. Segue-se imediatamente que $f_0, f_1 > 0$. ■

No caso $m = 2$, o teorema 1.2 mostra que a identidade 1-D das amplitudes é condição necessária e suficiente para a solvabilidade do problema da fase. Já para $m = 3$ ela não é mais suficiente.

Teorema 1.3 *Sejam* α_u, $u = 0, 1, 2, 3$, *números reais positivos. Sejam* $\epsilon_0, \epsilon_3 \in \{\pm 1\}$. *Existe um sinal real* $f = \begin{bmatrix} f_0 & f_1 & f_2 & 0 & 0 & 0 \end{bmatrix}^T$ *tal que* $F_0 = \epsilon_0\alpha_0$, $|F_1| = \alpha_1$, $|F_2| = \alpha_2$, $F_3 = \epsilon_3\alpha_3$, *se, e somente se,* $\{\alpha_u\}$, ϵ_3 *e* α_3 *satisfazem*

(i) *a identidade 1-D das amplitudes* $\alpha_0^2 - 2\alpha_1^2 + 2\alpha_2^2 - \alpha_3^2 = 0$,

(ii) $4\alpha_2 \geq |\epsilon_0\alpha_0 - 3\epsilon_3\alpha_3|$

Demonstração : Suponhamos que exista $f = \begin{bmatrix} f_0 & f_1 & f_2 & 0 & 0 & 0 \end{bmatrix}^T$ tal que $|F_u| = \alpha_u \;\; \forall \; u \;\; \text{e} \;\; \epsilon_u = \text{sgn}(F_u)$, $u = 0, 3$. A condição (i) segue do corolário 1.1 e a condição (ii) segue da parte (b) do lema 1.2.

Reciprocamente, suponhamos que α_u sejam números reais positivos que satisfaçam (i) e (ii) para determinados inteiros $\epsilon_0, \epsilon_3 \in \{\pm 1\}$; e consideremos

$$f = \begin{bmatrix} f_0 & f_1 & f_2 & 0 & 0 & 0 \end{bmatrix}^T,$$

para f_0, f_1, f_2 definidos como em (1.70) [alternativamente (1.71)]. A condição (ii) garante que o discriminante que aparece na expressão de f_0, f_1, f_2 em (1.70) é não negativo, o que mostra que f é real. Falta mostrarmos que $|F_u| = \alpha_u$, $u = 0, 1, 2, 3$, onde $F = \mathcal{F}(f)$. Para isto, basta substituirmos f_0, f_1, f_2 em (1.70) [alternativamente (1.71)] no sistema (1.68) e aplicarmos (i) convenientemente.

Teorema 1.4 *Sejam* α_u, $u = 0, 1, 2, 3$, *números reais positivos.*

(a) *Existe um sinal real* $f = \begin{bmatrix} f_0 & f_1 & f_2 & 0 & 0 & 0 \end{bmatrix}^T$, *tal que* $|F_u| = \alpha_u \; \forall \; u$, *se, e somente se,* $\{\alpha_u\}$ *satisfaz*

(i) *a identidade 1-D das amplitudes* $\alpha_0^2 - 2\alpha_1^2 + 2\alpha_2^2 - \alpha_3^2 = 0$,

(ii) $4\alpha_2 \geq |\alpha_0 - 3\alpha_3|$

(b) *Existe um sinal real* $f = \begin{bmatrix} f_0 & f_1 & f_2 & 0 & 0 & 0 \end{bmatrix}^T$ *com* $f_0, f_1, f_2 > 0$ *e tal que* $|F_u| = \alpha_u$ *se, e somente se,* $\{\alpha_u\}$ *satisfaz* (i), (ii) *e as condições de positividade*

(iii) $4\alpha_2^2 < \alpha_0^2 + 3\alpha_3^2$,

(iv) $\alpha_0 > \alpha_3$.

Demonstração : (a) É um corolário imediato do Teorema 1.3.

(b) Suponhamos agora que as componentes de f satisfaçam a restrição de positividade $f_0, f_1, f_2 > 0$. Pela parte (a), as amplitudes α_u satisfazem (i) e (ii). Falta mostrarmos que α_u satisfazem (iii) e (iv). Primeiramente, de (1.23) segue que $\epsilon_0 = \text{sgn}(F_0) = 1$. Agora, pelo lema 1.2, f_0, f_1 e f_2 devem satisfazer (1.70) ou (1.71), para $\epsilon_0 = 1$. Então temos as seguintes implicações

$$f_0 > 0, f_1 > 0, f_2 > 0 \Longrightarrow \begin{cases} f_0 + f_2 > 0 \\ f_1 > 0 \end{cases} \Longrightarrow \begin{cases} \alpha_0 + \epsilon_3\alpha_3 > 0 \\ \alpha_0 - \epsilon_3\alpha_3 > 0 \end{cases} \Longrightarrow \alpha_0 > \alpha_3;$$

o que prova (iv). Além disso,

$$f_0 > 0,\ f_2 > 0 \Longrightarrow 6(\alpha_0 + \epsilon_3\alpha_3) > \sqrt{\Delta}, \tag{1.72}$$

onde $\Delta = 12[16\alpha_2^2 - (\alpha_0 - 3\epsilon_3\alpha_3)^2]$. Desde que $\alpha_0 > \alpha_3$, então $\alpha_0 + \epsilon_3\alpha_3 > 0$. Então, elevando-se ambos os membros da desigualdade em (1.72) ao quadrado, obtém-se a desigualdade (iii).

Reciprocamente, suponhamos que $\alpha_u,\ u = 0, 1, 2, 3$, satisfaçam (i), (ii), (iii) e (iv). Definimos f_0, f_1 e f_2 por (1.70), para $\epsilon_0 = \epsilon_3 = 1$. Então temos

$$\begin{aligned} &\text{(iv)} \Longrightarrow f_1 > 0, \\ &\alpha_0 > 0, \alpha_3 > 0 \Longrightarrow f_0 > 0, \\ &\text{(iii)} \Longrightarrow f_2 > 0. \end{aligned}$$ ■

1.4.2 Identidades das amplitudes para o caso 2-D:

Nesta subseção abordaremos os resultados equivalentes aos da subseção anterior para o caso bidimensional, ou seja, determinaremos as *identidades 2-D das amplitudes* e apresentaremos, para ambos os casos $(m_1 = m_2 = 2)$ e $(m_1 = 2, m_2 = 3)$, as condições necessárias e suficientes para que o problema da fase admita uma solução (positiva) que satisfaça à restrição de suporte.

A seguir, o equivalente 2-D do teorema 1.1.

Teorema 1.5 *Seja*

$$f = \begin{bmatrix} f_{(11)} & f_{(12)} \\ f_{(21)} & f_{(22)} \end{bmatrix}$$

uma imagem real $n_1 \times n_2$ *e* F *sua* DFT. *Suponha que as amplitudes de* F *são dadas por* $\alpha_F = (\alpha_{uv})$; $(u,v) \in \Lambda$. *Então*

$$S_{m_1+m_2}(\alpha_F) = 2n_1n_2 \text{ tr } [f_{(\mathbf{22})}^T f_{(\mathbf{11})} + f_{(\mathbf{21})}^T f_{(\mathbf{12})}], \tag{1.73}$$

onde

$$\begin{aligned} S_{m_1+m_2}(\alpha_F) \;&:=\; \alpha_{00}^2 + 2\sum_{v=1}^{m_2-1}(-1)^v\alpha_{0v}^2 + (-1)^{m_2}\alpha_{0m_2}^2 + 2\sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1}(-1)^{u+v}\alpha_{uv}^2 \\ &+\; (-1)^{m_1}\left[\alpha_{m_10}^2 + 2\sum_{v=1}^{m_2-1}(-1)^v\alpha_{m_1v}^2 + (-1)^{m_2}\alpha_{m_1m_2}^2\right]. \end{aligned} \tag{1.74}$$

Demonstração : Com efeito, a expressão de autocorrelação (1.46) aplicada ao caso 2-D torna-se equivalente a:

$$\begin{aligned} \frac{1}{n_1n_2}\sum_{u=0}^{n_1-1}\sum_{v=0}^{n_2-1}|F_{uv}|^2\omega_1^{uj}\omega_2^{vk} \;&=\; \mathcal{F}^{-1}(|F_{uv}|^2) \\ &=\; \sum_{p=0}^{n_1-1}\sum_{q=0}^{n_2-1} f_{pq}f_{(j+p,k+q)(\text{mod } n_1,\text{mod } n_2)}, \end{aligned} \tag{1.75}$$

para $j \in \{0,1,\ldots,n_1-1\}$, $k \in \{0,1,\ldots,n_2-1\}$. Aqui, como antes, $\omega_l = \exp(\pi i/m_l)$; $l = 1,2$. Fazendo $j = m_1$ e $k = m_2$ na expressão de autocorrelação (1.75), a expressão da esquerda torna-se uma combinação dos $f_j's$ que nada mais é do que 2 tr $[f_{(\mathbf{22})}^T f_{(\mathbf{11})} + f_{(\mathbf{21})}^T f_{(\mathbf{12})}]$. Já a somatória dupla que aparece no terceiro membro, após aplicarmos a simetria 2-D de Fourier sobre as componentes $|F_{uv}|$, torna-se igual ao polinômio $S_{m_1+m_2}(\alpha_F)$ definido em (1.74)

Segue agora a versão 2-D do corolário 1.1.

Corolário 1.2 *Seja* f *uma imagem real, como no teorema* 1.5, *que satisfaz a restrição de suporte. Então o vetor* α_F, *das amplitudes de* F, *satisfaz a identidade polinomial*

$$S_{m_1+m_2}(\alpha_F) = 0, \tag{1.76}$$

onde $S_{m_1+m_2}(\alpha_F)$ *é o polinômio definido em* (1.74).

Demonstração : Basta substituir $f_{(\mathbf{12})} = f_{(\mathbf{21})} = f_{(\mathbf{22})} = 0$ em (1.73) ∎

(1.76) não é a única identidade linear nas amplitudes quadradas, que deriva-se da autocorrelação . Como veremos no próximo teorema, entretanto, para obtermos mais identidades, precisamos primeiramente da expressão equivalente a (1.63) para

o caso 2-D. Tal expressão é obtida ao levarmos em conta a simetria 2-D de Fourier da matriz $|F|$ e a substituição $f_{(\mathbf{12})} = f_{(\mathbf{21})} = f_{(\mathbf{22})} = 0$ em (1.75). Após estas considerações, temos (1.75) equivalente a

$$\sum_{p=0}^{m_1-1-j}\sum_{q=0}^{m_2-1-k} f_{pq}f_{j+p,k+q} = \frac{1}{n_1 n_2}\left\{\alpha_{00}^2 + 2\sum_{v=1}^{m_2-1}\mathrm{Re}(\omega_2^{vk})\alpha_{0v}^2 + (-1)^k\alpha_{0m_2}^2 \right.$$
$$+2\sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1}\mathrm{Re}(\omega_1^{uj}\omega_2^{vk})\alpha_{uv}^2$$
$$\left. + (-1)^j\left[\alpha_{m_1 0}^2 + 2\sum_{v=1}^{m_2-1}\mathrm{Re}(\omega_2^{vk})\alpha_{m_1 v}^2 + (-1)^k\alpha_{m_1 m_2}^2\right]\right\}, \tag{1.77}$$

para todo $j \in \{0,1,\ldots,m_1\}$; $k \in \{0,1,\ldots,m_2\}$.

Note que o lado esquerdo de (1.77) é identicamente nulo quando $j = m_1$ ou $k = m_2$. Assim, se substituirmos $j = m_1$ em (1.77), obteremos as m_2 primeiras identidades envolvendo as coordenadas do vetor α_F :

$$S_k(\alpha_F) = 0; \quad k = 0,1,\ldots,m_2-1,$$

para

$$\begin{aligned} S_k(\alpha_F) \;:=\; & \alpha_{00}^2 + 2\sum_{v=1}^{m_2-1}\mathrm{Re}(\omega_2^{vk})\alpha_{0v}^2 + (-1)^k\alpha_{0m_2}^2 + 2\sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1}(-1)^u\mathrm{Re}(\omega_2^{vk})\alpha_{uv}^2 \\ & + (-1)^{m_1}\left[\alpha_{m_1 0}^2 + 2\sum_{v=1}^{m_2-1}\mathrm{Re}(\omega_2^{vk})\alpha_{m_1 v}^2 + (-1)^k\alpha_{m_1 m_2}^2\right]. \end{aligned} \tag{1.78}$$

De maneira análoga, a substituição de $k = m_2$ em (1.77) gera as próximas m_1 identidades:

$$S_{m_2+j}(\alpha_F) = 0; \quad j = 0,1,\ldots,m_1-1$$

para

$$\begin{aligned} S_{m_2+j}(\alpha_F) \;:=\; & \alpha_{00}^2 + 2\sum_{v=1}^{m_2-1}(-1)^v\alpha_{0v}^2 + (-1)^{m_2}\alpha_{0m_2}^2 + 2\sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1}(-1)^v\mathrm{Re}(\omega_1^{uj})\alpha_{uv}^2 \\ & + (-1)^j\left[\alpha_{m_1 0}^2 + 2\sum_{v=1}^{m_2-1}(-1)^v\alpha_{m_1 v}^2 + (-1)^{m_2}\alpha_{m_1 m_2}^2\right]. \end{aligned} \tag{1.79}$$

Finalmente, substituindo $j = m_1$ e $k = m_2$ em (1.77), obtemos a última das identidades (1.76). Notemos que (1.76) também pode ser obtida substituindo $j = m_1$ em (1.79). Logo, (1.79) vale para todo $j = 0,1,\ldots,m_1$. Temos, portanto, um total de

$m_1 + m_2 + 1$ identidades vindas da autocorrelação. Com isto demonstramos parte do próximo resultado.

Teorema 1.6 *Seja f uma imagem real de tamanho $n_1 \times n_2$, e $F \in \mathbb{M}_{n_1 \times n_2}(\mathbb{C})$ sua* DFT. *Suponha que as amplitudes de F são dadas por $\alpha_F = (\alpha_{uv})$; $(u, v) \in \Lambda$. Se $f_{(12)} = f_{(21)} = f_{(22)} = 0$, as amplitudes α_{uv} satisfazem as seguintes $m_1 + m_2 + 1$ identidades linearmente independentes (LI):*

$$S_k(\alpha_F) = 0; \quad k = 0, 1, \ldots, m_1 + m_2, \tag{1.80}$$

onde S_k são os polinômios dados em (1.78), (1.79) *e* (1.74).

Demonstração : Já mostramos que o vetor α_F satisfaz as identidades (1.80). Falta provarmos que elas são LI. Com efeito, seja

$$b = (\alpha_{00}^2, \alpha_{01}^2, \ldots, \alpha_{m_1 m_2}^2)^T \in \mathbb{R}_+^{2m_1 m_2 + 2} \tag{1.81}$$

o vetor das amplitudes quadradas de Fourier. Então o sistema (1.80), que em termos do vetor b se reescreve como

$$S_k(b) = 0; \quad k = 0, 1, \ldots, m_1 + m_2, \tag{1.82}$$

é linear na variável b, com representação matricial dada por

$$\mathbf{B}\, b = 0, \tag{1.83}$$

onde $\mathbf{B}$ é a matriz dos coeficientes de (1.82). É fácil ver que $\mathbf{B}$ é uma matriz de ordem $(m_1 + m_2 + 1) \times (2m_1 m_2 + 2)$ que satisfaz

$$\mathbf{B} = 2\mathrm{Re}[\tilde{\mathbf{B}}]\, \tilde{D}_2 = \mathrm{Re}[2\tilde{\mathbf{B}}\tilde{D}_2], \tag{1.84}$$

onde $\tilde{\mathbf{B}}$ é a matriz cujo termo geral é, para cada $0 \leq k \leq m_2 - 1$, dado por

$$\begin{cases} \tilde{\mathbf{B}}_{kv} = \omega_2^{vk}, & 0 \leq v \leq m_2, \\ \tilde{\mathbf{B}}_{kq} = (-1)^u \omega_2^{vk}, & 1 \leq u \leq m_1 - 1,\ 0 \leq v \leq n_2 - 1, \\ & q = m_2 + (u-1)n_2 + v + 1, \\ \tilde{\mathbf{B}}_{kq} = (-1)^{m_1} \omega_2^{vk}, & 0 \leq v \leq m_2,\ q = m_2 + (m_1 - 1)n_2 + v + 1, \end{cases} \tag{1.85}$$

e, para cada $0 \leq j \leq m_1$, por

$$\begin{cases} \tilde{\mathbf{B}}_{m_2+j,\,v} = (-1)^v, & 0 \leq v \leq m_2, \\ \tilde{\mathbf{B}}_{m_2+j,\,q} = (-1)^v \omega_1^{uj}, & 1 \leq u \leq m_1 - 1,\ 0 \leq v \leq n_2 - 1, \\ & q = m_2 + (u-1)n_2 + v + 1, \\ \tilde{\mathbf{B}}_{m_2+j,\,q} = (-1)^{j+v}, & 0 \leq v \leq m_2,\ q = m_2 + (m_1 - 1)n_2 + v + 1, \end{cases} \tag{1.86}$$

A matriz $\tilde{D}_2$ é dada por

$$\tilde{D}_2 = \tilde{D}_1 \oplus I_{n_2(m_1-1)} \oplus \tilde{D}_1$$

quando

$$\tilde{D}_1 = \frac{1}{2} \oplus I_{m_2-1} \oplus \frac{1}{2}.$$

A verificação de (1.84) é simples. De fato, seja

$$\tilde{b} := \tilde{D}_2 b.$$

Então os polinômios S_k se reescrevem na variável $\tilde{b}$ como

$$S_k(\tilde{b}) = 2\mathrm{Re}\left[\sum_{v=0}^{m_2} \omega_2^{vk}\ \tilde{b}_{0v} + \sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1}(-1)^u\omega_2^{vk}\ \tilde{b}_{uv} + \sum_{v=0}^{m_2}(-1)^{m_1}\omega_2^{vk}\ \tilde{b}_{m_1 v}\right] \tag{1.87}$$

para cada $k = 0, 1, \ldots, m_2 - 1$, e

$$S_{m_2+j}(\tilde{b}) = 2\mathrm{Re}\left[\sum_{v=0}^{m_2}(-1)^v\ \tilde{b}_{0v} + \sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1}(-1)^v\omega_1^{uj}\ \tilde{b}_{uv} + \sum_{v=0}^{m_2}(-1)^j(-1)^v\ \tilde{b}_{m_1 v}\right] \tag{1.88}$$

para cada $j = 0, 1, \ldots, m_1$. Se definimos

$$S(b) := (S_0(b), S_1(b), \ldots, S_{m_1+m_2}(b))^T,$$

verifica-se facilmente que as equações (1.87) e (1.88) se reescrevem na seguinte forma matricial

$$S(\tilde{b}) = (2\mathrm{Re}[\tilde{\mathbf{B}}])\,\tilde{b}.$$

Desde que

$$\mathbf{B}b = S(b) = S(\tilde{b}) = (2\mathrm{Re}[\tilde{\mathbf{B}}])\,\tilde{b} = (2\mathrm{Re}[\tilde{\mathbf{B}}]\,\tilde{D}_2)\,b,$$

conclui-se que

$$\mathbf{B} = 2\mathrm{Re}[\tilde{\mathbf{B}}]\,\tilde{D}_2 = 2\mathrm{Re}[\tilde{\mathbf{B}}\,\tilde{D}_2].$$

Falta provarmos que as linhas de $\mathbf{B}$ são LI, ou seja, que $\mathbf{B}$ tem posto completo. Para isto é suficiente mostrarmos que $\mathbf{B}\mathbf{B}^T$ é diagonalmente dominante. A demonstração será dada no Apêndice B.

Exemplo: Tomemos $m_1 = 2$ e $m_2 = 3$. O vetor b, neste caso, é

$$b = (\alpha_{00}^2, \alpha_{01}^2, \alpha_{02}^2, \alpha_{03}^2, \alpha_{10}^2, \alpha_{11}^2, \alpha_{12}^2, \alpha_{13}^2, \alpha_{14}^2, \alpha_{15}^2, \alpha_{20}^2, \alpha_{21}^2, \alpha_{22}^2, \alpha_{23}^2)^T.$$

Os polinômios $S_0(b), S_1(b), \ldots, S_5(b)$ são as componentes do vetor $\mathbf{B}b$, para

$$\mathbf{B} = \begin{bmatrix} 1 & 2 & 2 & 1 & -2 & -2 & -2 & -2 & -2 & -2 & 1 & 2 & 2 & 1 \\ 1 & 1 & -1 & -1 & -2 & -1 & 1 & 2 & 1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & -2 & 1 & 1 & -2 & 1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 & 2 & -2 & 2 & -2 & 2 & -2 & 1 & -2 & 2 & -1 \\ 1 & -2 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 2 & -2 & 1 \\ 1 & -2 & 2 & -1 & -2 & 2 & -2 & 2 & -2 & 2 & 1 & -2 & 2 & -1 \end{bmatrix}.$$

Vimos que a identidade 1-D das amplitudes, (1.62), pôde ser obtida alternativamente através do desenvolvimento do termo $F^*\mathbf{\Delta}F$ (veja equação (1.64)). Como a maioria dos resultados para o caso 1-D tem seu similar no caso 2-D, é de se esperar, portanto, que o conjunto das identidades do teorema 1.6 também possa ser obtido alternativamente através de procedimento similar. Na verdade, tal procedimento nos leva a um outro conjunto LI de identidades nas variáveis α_{uv} cuja semelhança com a identidade (1.62) foi o motivo que nos levou a escolhê-las para serem as *identidades 2-D das amplitudes*. A esta altura, estamos nos perguntando então se esses dois conjuntos de identidades LI são equivalentes, no sentido de que os aneis gerados pelos polinômios das identidades de cada conjunto são iguais. Na verdade nós verificamos para vários valores atribuídos a m_1 e m_2, sem excessões, que esses conjuntos são equivalentes. Entretanto não conseguimos dar uma prova analítica para esse resultado.

A seguir temos o teorema cuja demonstração é o análogo 2-D da demonstração alternativa do teorema 1.1.

Teorema 1.7 *Sejam f uma imagem real de tamanho $n_1 \times n_2$ particionada conforme* (1.5) *e $F \in \mathbb{M}_{n_1 \times n_2}(\mathbb{C})$ sua* DFT. *Suponhamos que as amplitudes de F sejam dadas por $\alpha_{uv} = |F_{uv}|$, $\forall (u, v)$. Se $f_{(\mathbf{12})} = f_{(\mathbf{21})} = f_{(\mathbf{22})} = 0$, as amplitudes α_{uv} satisfazem as seguintes identidades:*

$$\sum_{v=0}^{n_2-1} (-1)^v \alpha_{uv}^2 = 0, \quad u = 0, 1, \ldots, n_1 - 1 \tag{1.89}$$

$$\sum_{u=0}^{n_1-1} (-1)^u \alpha_{uv}^2 = 0, \quad v = 0, 1, \ldots, n_2 - 1 \tag{1.90}$$

Demonstração : Para cada $j = 1, 2$, considere a matriz diagonal $\mathbf{\Delta}_j$, de tamanho $n_j \times n_j$, dada por

$$\mathbf{\Delta}_j = \begin{bmatrix} \mathbf{D}_j & 0 \\ 0 & (-1)^{m_j} \mathbf{D}_j \end{bmatrix},$$

onde $\mathbf{D}_j$ é a matriz diagonal $m_j \times m_j$ dada por $(\mathbf{D}_j)_{kk} = (-1)^k$, $k = 0, 1, \ldots, m_j - 1$. Usando o valor de F dado pelo segundo membro de (1.26) temos por um lado

$$\begin{aligned} F^* \boldsymbol{\Delta}_1 F &= (\mathcal{W}_1 f \mathcal{W}_2)^* \boldsymbol{\Delta}_1 (\mathcal{W}_1 f \mathcal{W}_2) \\ &= n_1 \mathcal{W}_2^* \begin{bmatrix} f_{(11)}^T f_{(21)} + f_{(21)}^T f_{(11)} & f_{(11)}^T f_{(22)} + f_{(21)}^T f_{(12)} \\ f_{(12)}^T f_{(21)} + f_{(22)}^T f_{(11)} & f_{(12)}^T f_{(22)} + f_{(22)}^T f_{(12)} \end{bmatrix} \mathcal{W}_2. \quad (1.91) \end{aligned}$$

Similarmente

$$F \boldsymbol{\Delta}_2 F^* = n_2 \mathcal{W}_1 \begin{bmatrix} f_{(11)} f_{(12)}^T + f_{(12)} f_{(11)}^T & f_{(11)} f_{(22)}^T + f_{(12)} f_{(21)}^T \\ f_{(21)} f_{(12)}^T + f_{(22)} f_{(11)}^T & f_{(21)} f_{(22)}^T + f_{(22)} f_{(21)}^T \end{bmatrix} \mathcal{W}_1^*. \quad (1.92)$$

Por outro lado, se substituirmos F pela matriz de seus pixels, F_{uv}, na expressão do membro direito de, por exemplo, (1.91), obteremos uma matriz cujos elementos, a_{vv}, da diagonal principal são

$$\begin{aligned} a_{vv} &= [F^*]_{(v)} \boldsymbol{\Delta}_1 [F]^{(v)} = \sum_{u=0}^{n_1-1} [\overline{F}_{uv}][(-1)^u F_{uv}] \\ &= \sum_{u=0}^{n_1-1} (-1)^u |F_{uv}|^2, \quad v = 0, 1, \ldots n_2 - 1. \end{aligned}$$

Então se $f_{(12)} = f_{(21)} = f_{(22)} = 0$, o lado direito de (1.91) se anula, implicando $a_{vv} = 0 \; \forall \; v$, o que nos dá o sistema (1.90). Similarmente mostra-se que os elementos da diagonal principal de $F \boldsymbol{\Delta}_2 F^*$ são nulos, o que implica, portanto, que $\{\alpha_{uv}\}$ satisfaz (1.89). ■

Cada linha do lado esquerdo de (1.89) e (1.90) é uma equação polinomial de grau 2 nas variáveis α_{uv}, $0 \leq u \leq n_1 - 1$, $0 \leq v \leq n_2 - 1$. Se levarmos em conta a simetria 2-D de Fourier da matriz $|F|$ (veja (1.28)), então poderemos descartar as $m_1 - 1$ últimas equações do sistema (1.89) pois, para cada $p \in \{1, 2, \ldots, m_1 - 1\}$, cada equação de (1.89), correspondente a $u = m_1 + p$, é a mesma equação correspondente a $u = m_1 - p$. Similarmente conclui-se que as $m_2 - 1$ últimas equações do sitema (1.90) também poderão ser desprezadas. Além disso, a simetria de F permite reduzir o número de variáveis α_{uv} que aparecem nas equações de modo que restem apenas as componentes do vetor das amplitudes α_F (veja a expressão (1.33)). De fato, segue da simetria de F que

$$\begin{aligned} \sum_{v=0}^{n_2-1} (-1)^v \alpha_{uv}^2 &= \alpha_{u0}^2 + 2 \sum_{v=1}^{m_2-1} (-1)^v \alpha_{uv}^2 + (-1)^{m_2} \alpha_{u\, m_2}^2, \quad u = 0, \; m_1, \\ \sum_{u=0}^{n_1-1} (-1)^u \alpha_{uv}^2 &= \alpha_{0v}^2 + 2 \sum_{u=1}^{m_1-1} (-1)^u \alpha_{uv}^2 + (-1)^{m_1} \alpha_{m_1 v}^2, \quad v = 0, \; m_2, \\ \sum_{u=0}^{n_1-1} (-1)^u \alpha_{uv}^2 &= \sum_{u=0}^{m_1} (-1)^u \alpha_{uv}^2 + \sum_{u=1}^{m_1-1} (-1)^u \alpha_{u,n_2-v}^2, \quad 1 \leq v \leq m_2 - 1. \end{aligned}$$

Assim (1.89) e (1.90) tornam-se equivalentes ao sistema

$$\begin{aligned} P_u(\alpha_F) &= 0; \quad u = 0, 1, 2, \ldots, m_1, \\ R_v(\alpha_F) &= 0; \quad v = 0, 1, 2, \ldots, m_2, \end{aligned} \tag{1.93}$$

onde $P_u(\alpha_F)$ e $R_u(\alpha_F)$ são os seguintes polinômios:

$$\begin{aligned} P_u(\alpha_F) &= \alpha_{u0}^2 + 2\sum_{v=1}^{m_2-1}(-1)^v\alpha_{uv}^2 + (-1)^{m_2}\alpha_{u\,m_2}^2, && u = 0, m_1, \\ P_u(\alpha_F) &= \sum_{v=0}^{n_2-1}(-1)^v\alpha_{uv}^2, && 1 \le u \le m_1 - 1, \\ R_v(\alpha_F) &= \alpha_{0v}^2 + 2\sum_{u=1}^{m_1-1}(-1)^u\alpha_{uv}^2 + (-1)^{m_1}\alpha_{m_1 v}^2, && v = 0, m_2, \\ R_v(\alpha_F) &= \sum_{u=0}^{m_1}(-1)^u\alpha_{uv}^2 + \sum_{u=1}^{m_1-1}(-1)^u\alpha_{u,n_2-v}^2, && 1 \le v \le m_2 - 1. \end{aligned} \tag{1.94}$$

Seja b o vetor das amplitudes quadradas de Fourier definido em (1.81). Então os polinômios do sistema (1.94) são polinômios lineares na variável b.

Proposição 1.3 *O polinômio $P_0(b)$ se escreve como uma combinação linear dos demais polinômios do sistema* (1.94).

Demonstração : Mostra-se que os polinômios do sistema (1.94) satisfazem

$$\left[P_0(b) + 2\sum_{u=1}^{m_1-1}(-1)^u P_u(b) + (-1)^{m_1}P_{m_1}(b)\right] - \left[R_0(b) + 2\sum_{v=1}^{m_2-1}(-1)^v R_u(b) + (-1)^{m_2}R_{m_2}(b)\right] = 0. \quad \blacksquare$$

Uma consequência óbvia da proposição anterior é a de que podemos desprezar a primeira equação do sistema (1.93). As $m_1 + m_2 + 1$ equações restantes são LI e formam as *identidades 2-D das amplitudes*. Esse é basicamente o resultado do próximo teorema.

Teorema 1.8 (Identidades 2-D das Amplitudes) *Seja f uma imagem real de tamanho $n_1 \times n_2$, e $F \in \mathbb{M}_{n_1\times n_2}(\mathbb{C})$ sua* DFT. *Suponhamos que as amplitudes de F são dadas por $\alpha_F = (\alpha_{uv})$; $(u, v) \in \Lambda$. Se $f_{(\mathbf{12})} = f_{(\mathbf{21})} = f_{(\mathbf{22})} = 0$, as amplitudes α_{uv} satisfazem as seguintes $m_1 + m_2 + 1$ identidades L.I.:*

$$\begin{aligned} R_v(\alpha_F) &= 0; \quad v = 0, 1, 2, \ldots, m_2, \\ P_u(\alpha_F) &= 0; \quad u = 1, 2, \ldots, m_1, \end{aligned} \tag{1.95}$$

onde $P_u(\alpha_F)$ e $R_v(\alpha_F)$ são os polinômios dados em (1.94).

Demonstração : Para completarmos a demonstração deste teorema precisamos apenas mostrar que as identidades (1.95) são LI. Com efeito, o sistema (1.95) é equivalente à equação matricial

$$Q = \mathbf{A}b = 0,$$

onde b é o vetor em (1.81),

$$Q := (R_0(b), R_1(b), \ldots, R_{m_2}(b), P_1(b), P_2(b), \ldots, P_{m_1}(b))^T \tag{1.96}$$

e $\mathbf{A}$ é a matriz, de tamanho $(m_1 + m_2 + 1) \times (2m_1m_2 + 2)$, dada por

$$\mathbf{A} = \begin{bmatrix}
\mathbf{1} & 0 & \ldots & 0 & -2 & \ldots & 0 & 2 & \ldots & 0 & \ldots & (-1)^{m_1} & 0 & \ldots & 0 \\
0 & \mathbf{1} & \ldots & 0 & 0 & \ldots & 0 & 0 & \ldots & 1 & \ldots & 0 & (1)^{m_1} & \ldots & 0 \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
0 & 0 & \ldots & \mathbf{1} & 0 & \ldots & 0 & 0 & \ldots & 0 & \ldots & 0 & 0 & \ldots & (-1)^{m_1} \\
0 & 0 & \ldots & 0 & \mathbf{1} & \ldots & (-1)^{n_2-1} & 0 & \ldots & 0 & \ldots & 0 & 0 & \ldots & 0 \\
0 & 0 & \ldots & 0 & 0 & \ldots & 0 & \mathbf{1} & \ldots & (-1)^{n_2-1} & \ldots & 0 & 0 & \ldots & 0 \\
\ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\
0 & 0 & \ldots & 0 & 0 & \ldots & 0 & 0 & \ldots & 0 & \ldots & \mathbf{1} & -2 & \ldots & (-1)^{m_2}
\end{bmatrix} \tag{1.97}$$

Mostrar que as identidades (1.95) são LI é equivalente a mostrar que a matriz $\mathbf{A}$ tem posto completo.

Com efeito, seja $[\mathbf{A}]_{(j)}$, $0 \leq j \leq M := m_1 + m_2$, o $j-ésimo$ vetor linha da matriz $\mathbf{A}$. Denotemos por k_j a posição coluna do primeiro elemento não nulo de $[\mathbf{A}]_{(j)}$. Obedecendo-se à ordem dos polinômios do sistema (1.95) e à das componentes α^2_{uv} estabelecida na definição do vetor b em (1.81), vê-se que $\mathbf{A}_{0,k_0}, \mathbf{A}_{1,k_1}, \ldots, \mathbf{A}_{M,k_M}$ são respectivamente os coeficientes das componentes $\alpha^2_{00}, \alpha^2_{01} \ldots, \alpha^2_{0m_2}, \alpha^2_{10}, \alpha^2_{20}, \ldots, \alpha^2_{m_1 0}$, nesta ordem, que aparecem nas equações do sistema (1.95). Então segue que

$$\mathbf{A}_{0,k_0} = \mathbf{A}_{1,k_1} = \ldots = \mathbf{A}_{M,k_M} = 1 \text{ e } k_0 > k_1 > \ldots > k_M,$$

o que significa que a matriz $\mathbf{A}$ já se apresenta na forma escalonada por linhas, ou seja, $\mathbf{A}$ tem posto completo ■

Exemplo: Tomemos novamente $m_1 = 2$ e $m_2 = 3$. Os polinômios $P_u(b)$ e $R_v(b)$ são :

$$\begin{aligned}
P_0(b) &= \alpha^2_{00} - 2\alpha^2_{01} + 2\alpha^2_{02} - \alpha^2_{03} \\
P_1(b) &= \alpha^2_{10} - \alpha^2_{11} + \alpha^2_{12} - \alpha^2_{13} + \alpha^2_{14} - \alpha^2_{15} \\
P_2(b) &= \alpha^2_{20} - 2\alpha^2_{21} + 2\alpha^2_{22} - \alpha^2_{23} \\
R_0(b) &= \alpha^2_{00} - 2\alpha^2_{10} + \alpha^2_{20} \\
R_1(b) &= \alpha^2_{01} - \alpha^2_{11} + \alpha^2_{21} - \alpha^2_{15} \\
R_2(b) &= \alpha^2_{02} - \alpha^2_{12} + \alpha^2_{22} - \alpha^2_{14} \\
R_3(b) &= \alpha^2_{03} - 2\alpha^2_{13} + \alpha^2_{23}
\end{aligned}$$

O vetor Q e a matriz $\mathbf{A}$ são dados por

$$Q = (R_0(b), R_1(b), R_2(b), R_3(b), P_1(b), P_2(b))^T$$

e

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 2 & 1 \end{bmatrix}.$$

Observação : Sejam $\mathbf{A}$ e $\mathbf{B}$ as matrizes (1.97) e (1.84). Para vários valores atribuídos a m_1, m_2 foi verificado, sem excessões, que

$$\mathbf{B} = \mathbf{G}\mathbf{A},$$

para uma matriz inversível $\mathbf{G}$, especificamente,

$$\mathbf{G} = (\mathbf{B}\,\mathbf{A}^T)(\mathbf{A}\,\mathbf{A}^T)^{-1}.$$

Portanto, acredita-se que o sistemas (1.80) e (1.95) são equivalentes, ou seja, que o anel de polinômios gerado por $\{S_0(b), S_1(b), \ldots, S_{m_1+m_2}(b)\}$ é igual ao anel gerado por $\{R_0(b), \ldots, R_{m_2}(b), P_1(b), \ldots, P_{m_1}(b)\}$. Porém, não conseguimos mostrar analiticamente esse resultado para todo m_1, m_2.

A seguir daremos a versão do lema 1.1 para o caso 2-D. Para tanto será conveniente que trabalhemos com os elementos do suporte de um objeto f na forma de vetor coluna. Em outras palavras, se

$$f = \begin{bmatrix} f_{(\mathbf{11})} & 0 \\ 0 & 0 \end{bmatrix}, \tag{1.98}$$

ordenamos os elementos do bloco $f_{(\mathbf{11})} \in \mathbb{M}_{m_1 \times m_2}(\mathbb{R})$ de modo a formarmos um vetor coluna, $\mathbf{f}_{\mathcal{S}}$, de dimensão $m_1 m_2$, tal que os primeiros m_2 elementos de $\mathbf{f}_{\mathcal{S}}$, por exemplo, sejam os elementos da primeira linha de $f_{(\mathbf{11})}$, os próximos m_2 elementos sejam os elementos da segunda linha e assim por diante, para todas as m_1 linhas de $f_{(\mathbf{11})}$. Se representarmos

$$\mathbf{f}_{\mathcal{S}} = [f_0, f_1, \ldots, f_{m_1 m_2 - 1}]^T \quad \text{e} \quad f_{(\mathbf{11})} = (f_{jk}), \tag{1.99}$$

então

$$f_q = f_{jk} \tag{1.100}$$

para cada índice $q \in \{0, 1, \ldots, m_1 m_2 - 1\}$ que satisfaz

$$q = m_2 j + k,$$

onde $j \in \{0, 1, \ldots, m_1 - 1\}$ e $k \in \{0, 1, \ldots, m_2 - 1\}$ são respectivamente o quociente e o resto da divisão de q por m_2. Por causa da relação biunívoca entre (j, k) e q, será conveniente que indexemos j e k por q :

$$j_q := j \quad \text{e} \quad k_q := k.$$

A seguir o similar 2-D do lema 1.1.

Lema 1.3 *Sejam $f \in \mathbb{M}_{n_1 \times n_2}(\mathbb{R})$ dado como em (1.98) e $\mathbf{f}_S$ o vetor coluna formado pelas componentes do suporte de f, dadas como em (1.100). Se $F = \mathcal{F}(f)$ e $\alpha_{uv} = |F_{uv}|$, para todo $(u, v) \in \Lambda$, então*

$$\sum_{p=0}^{m_1 m_2 - 2} \sum_{q=1}^{m_1 m_2 - 1 - p} f_p f_{p+q} \mathrm{Re}(\omega_1^{u(j_{p+q} - j_p)} \omega_2^{v(k_{p+q} - k_p)}) = \frac{1}{2}[\alpha_{uv}^2 - \frac{1}{n_1 n_2}||F||^2], \quad (1.101)$$

$$\begin{aligned} &\sum_{q=0}^{m_1 m_2 - 1} f_q = F_{00}, && \sum_{q=0}^{m_1 m_2 - 1} (-1)^{k_q} f_q = F_{0 m_2}, \\ &\sum_{q=0}^{m_1 m_2 - 1} (-1)^{j_q} f_q = F_{m_1 0}, && \sum_{q=0}^{m_1 m_2 - 1} (-1)^{j_q + k_q} f_q = F_{m_1 m_2}. \end{aligned} \quad (1.102)$$

Demonstração : Como no caso 1-D, parte-se da expressão (1.26), $F = \mathcal{W}_1 f \mathcal{W}_2$, para se chegar em

$$|F_{uv}|^2 = 2 \sum_{p=0}^{m_1 m_2 - 2} \sum_{q=1}^{m_1 m_2 - 1 - p} f_p f_{p+q} \mathrm{Re}(\omega_1^{u(j_{p+q} - j_p)} \omega_2^{v(k_{p+q} - k_p)}) + ||\mathbf{f}_S||^2. \quad (1.103)$$

Em seguida aplica-se a fórmula de Parseval, $||f||^2 = (1/n_1 n_2)||F||^2$, em (1.103) para se obter (1.101). As fórmulas (1.102) são obtidas diretamente de (1.26) ■

O próximo lema relaciona os pixels de uma imagem 4×4, que satisfaz a restrição de suporte, com as amplitudes de sua transformada de Fourier.

Lema 1.4 *Seja*

$$f = \begin{bmatrix} f_0 & f_1 & 0 & 0 \\ f_2 & f_3 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (1.104)$$

um objeto real. Considere $\alpha_{uv} = |F_{uv}|$, *para todo* $(u,v) \in \Lambda$, *e* $\epsilon_{uv} = \mathrm{sgn}(F_{uv})$, *para* $(u,v) \in \{0, m_1\} \times \{0, m_2\}$. *Então*

$$\begin{aligned} f_0 &= [(\epsilon_{00}\alpha_{00} + \epsilon_{20}\alpha_{20}) + (\epsilon_{02}\alpha_{02} + \epsilon_{22}\alpha_{22})]/4, \\ f_1 &= [(\epsilon_{00}\alpha_{00} + \epsilon_{20}\alpha_{20}) - (\epsilon_{02}\alpha_{02} + \epsilon_{22}\alpha_{22})]/4, \\ f_2 &= [(\epsilon_{00}\alpha_{00} - \epsilon_{20}\alpha_{20}) + (\epsilon_{02}\alpha_{02} - \epsilon_{22}\alpha_{22})]/4, \\ f_3 &= [(\epsilon_{00}\alpha_{00} - \epsilon_{20}\alpha_{20}) - (\epsilon_{02}\alpha_{02} - \epsilon_{22}\alpha_{22})]/4. \end{aligned} \tag{1.105}$$

Demonstração : Obtém-se (1.105) resolvendo-se o sistema (1.102) para as incógnitas f_q, quando $m_1 = m_2 = 2$ ■

O próximo teorema aborda as condições necessárias e suficientes para a existência de solução *positiva* do problema da fase, para o caso $m_1 = m_2 = 2$.

Teorema 1.9 *Sejam* α_{uv}, $(u,v) \in \Lambda$, *números reais positivos.*

(a) *Existe uma imagem real* f *como em* (1.104) *tal que* $|F_{uv}| = \alpha_{uv}$ *se, e somente se,* $\{\alpha_{uv}\}$ *satisfazem*

(i) *as identidades 2-D das amplitudes*

$$\left\{ \begin{array}{rcl} \alpha_{00}^2 - 2\alpha_{10}^2 + \alpha_{20}^2 &=& 0, \\ \alpha_{01}^2 - \alpha_{11}^2 + \alpha_{21}^2 - \alpha_{13}^2 &=& 0, \\ \alpha_{02}^2 - 2\alpha_{12}^2 + \alpha_{22}^2 &=& 0, \\ \alpha_{10}^2 - \alpha_{11}^2 + \alpha_{12}^2 - \alpha_{13}^2 &=& 0, \\ \alpha_{20}^2 - 2\alpha_{21}^2 + \alpha_{22}^2 &=& 0, \end{array} \right. \tag{1.106}$$

(ii) *a identidade*

$$\alpha_{01}^2 - 2\alpha_{11}^2 + \alpha_{21}^2 = \epsilon_{00}\alpha_{00}\epsilon_{22}\alpha_{22} - \epsilon_{02}\alpha_{02}\epsilon_{20}\alpha_{20}, \tag{1.107}$$

onde $\epsilon_{00}, \epsilon_{22}, \epsilon_{02}, \epsilon_{20} \in \{\pm 1\}$.

(b) *Existe uma imagem real* f *como em* (1.104) *com* $f_0, f_1, f_2, f_3 > 0$ *e tal que* $|F_{uv}| = \alpha_{uv}$ *se, e somente se,* $\{\alpha_{uv}\}$ *satisfazem* (i), (ii) *com* $\epsilon_{00} = 1$, *e as condições*

$$\begin{array}{rcl} \alpha_{00} + \alpha_{20} &>& \alpha_{02} + \alpha_{22}, \\ \alpha_{00} - \alpha_{20} &>& |\alpha_{02} - \alpha_{22}|, \end{array} \tag{1.108}$$

Demonstração : (a) Suponhamos que exista um objeto f como em (1.104) tal que $|F_{uv}| = \alpha_{uv}$, $\forall\ (u,v)$. Seja $\epsilon_{uv} = \mathrm{sgn}(F_{uv})$, para $(u,v) \in \{0, m_1\} \times \{0, m_2\}$. As identidades em (1.106) é uma aplicação do teorema 1.8, i.e., elas são as identidades

(1.95) quando $m_1 = m_2 = 2$. Mostremos agora (ii). Fazendo $m_1 = m_2 = 2$ e $u = v = 1$ em (1.103), vem que

$$\alpha_{11}^2 = |F_{11}|^2 = 2(-f_0 f_3 + f_1 f_2) + ||\mathbf{f}_S||^2, \tag{1.109}$$

para $\mathbf{f}_S = (f_0, f_1, f_2, f_3)^T$. Pelo lema 1.4, f_0, f_1, f_2 e f_3 devem satisfazer (1.105) cujas expressões do lado direito devem ser levadas em (1.109) para se obter (1.107).

Reciprocramente suponhamos que $\{\alpha_{uv}\}$ satisfaçam (1.106) e (1.107) para alguns inteiros $\epsilon_{00}, \epsilon_{22}, \epsilon_{02}, \epsilon_{20} \in \{\pm 1\}$; e consideremos f como em (1.104) para f_0, f_1, f_2 e f_3 definidos conforme (1.105). Se $F = \mathcal{F}(f)$ então mostra-se que $|F_{uv}| = \alpha_{uv}\ \forall (u, v)$ substituindo-se os valores de f_0, f_1, f_2 e f_3 em (1.103) e aplicando-se as identidades (1.106) e (1.107) convenientemente.

(b) Mostremos que se f_0, f_1, f_2 e f_3 são dados como em (1.105), para $\epsilon_{00} = 1$, então $f_0, f_1, f_2, f_3 > 0$ se, e somente se, as desigualdades (1.108) ocorrem. Com efeito,

$$\begin{array}{lllll} f_0, f_1 > 0 & \Longleftrightarrow & \alpha_{00} + \epsilon_{20}\alpha_{20} > |\epsilon_{02}\alpha_{02} + \epsilon_{22}\alpha_{22}| & \Longleftrightarrow & \alpha_{00} + \epsilon\ \alpha_{20} > |\alpha_{02} + \epsilon^{'}\alpha_{22}| \\ f_2, f_3 > 0 & \Longleftrightarrow & \alpha_{00} - \epsilon_{20}\alpha_{20} > |\epsilon_{02}\alpha_{02} - \epsilon_{22}\alpha_{22}| & \Longleftrightarrow & \alpha_{00} - \epsilon\ \alpha_{20} > |\alpha_{02} - \epsilon^{'}\alpha_{22}| \end{array}$$

para $\epsilon := \epsilon_{20}$, $\epsilon^{'} := \epsilon_{22}\epsilon_{02}$. Analisando as 4 possibilidades de valores, $(+1, +1)$, $(+1, -1)$, $(-1, +1)$ e $(-1, -1)$, para o par $(\epsilon, \epsilon^{'})$, verifica-se que par de desigualdades

$$\begin{array}{l} \alpha_{00} + \epsilon\ \alpha_{20} > |\alpha_{02} + \epsilon^{'}\alpha_{22}| \\ \alpha_{00} - \epsilon\ \alpha_{20} > |\alpha_{02} - \epsilon^{'}\alpha_{22}| \end{array}$$

é equivalente a (1.108). Então a demonstração da parte (b) deste teorema é uma consequência do resultado provado acima, mais o lema 1.4 e a implicação dada em (1.36). ■

1.5 Comentários sobre o paper de Sanz

Nesta seção exibiremos algumas das equações polinomiais, nas amplitudes quadradas, que Sanz, em [74], afirma existirem mas que não foram até hoje registradas. Na verdade tais equações nada mais são do que as identidades das amplitudes. Para o caso 1-D a identidade encontrada é a única equação. Quanto ao caso 2-D, damos o número máximo destas equações e acreditamos que as identidades encontradas em (1.95) são as *únicas* equações polinomiais de grau 1 nos α_u^2.

A seguir daremos a versão do problema inverso da fase abordado no paper [74] de Sanz.

Suponhamos que $H : \mathbb{C}^M \longrightarrow \mathbb{C}^N$ seja uma transformação linear, com $N \geq M$. Uma versão generalizada do problema da fase com restrição de suporte, (1.39), formulada por Sanz em [74], consiste na resolução da equação

$$|(Hx)(u)|^2 = b_u, \quad u = 0, 1, \ldots, N-1 \tag{1.110}$$

para $x \in \mathbb{R}^M$, onde os $b_u's$ são supostamente conhecidos. A restrição de suporte aparece na formulação deste problema quando a solução x que buscamos em (1.110) constituir-se apenas dos valores do objeto dentro do suporte. Assim, na comparação da expressão (1.110) com (1.39), para o caso 1-D por exemplo, temos

$$\begin{array}{l} M = m, \quad N = m+1, \quad b_u = \alpha_u^2, \\ Hx = \mathcal{F}(f) = \mathcal{W}_{(1)} f_{(1)}, \quad H = \mathcal{W}_{(1)}, \quad x = f_{(1)}, \end{array} \tag{1.111}$$

onde $\mathcal{W}_{(1)}$ e $f_{(1)}$ são a matriz em (1.16) e o vetor em (1.4) respectivamente.

As equações em (1.110) podem ser vistas como equações polinomiais, a coeficientes complexos, na variável x, i.e.,

$$P_u(x) \;=\; b_u, \quad u = 0, 1, \ldots, N-1,$$

onde $P_u : \mathbb{R}^M \longrightarrow \mathbb{R}$, para cada $u = 0, 1, 2, \ldots, N-1$, é a função polinomial

$$P_u(x) = |(Hx)(u)|^2 = \sum_{j=0}^{M-1} \sum_{k=0}^{M-1} h_{uj} h_{uk}^* x_j x_k. \tag{1.112}$$

Definição 1.6 *Dizemos que uma sequência* $b = (b_0, b_1, \ldots, b_{N-1})^T \in \mathbb{R}_+^N$ *é* admissível *se a equação* $P(x) = b$ *tem uma solução real* x, *onde* $P : \mathbb{R}^M \longrightarrow \mathbb{R}^N$ *é o polinômio dado por*

$$P(x) = (P_0(x), P_1(x), \ldots, P_{N-1}(x))^T, \tag{1.113}$$

com $P_u(x)$, $u = 0, 1, \ldots, N-1$, *dados por (1.112).*

Teorema 1.10 (Teorema de Sanz - [74]) *Consideremos o problema sobredeterminado* $(N \geq M)$ *de recuperação da fase* (1.110) *e assumamos que a matriz envolvida* $H = [h_{jk}]$ *satisfaz a propriedade:*

$$P_u(x) = 0, \; u = 0, 1, \ldots, N-1, \text{ tem somente a solução trivial.} \tag{1.114}$$

Então existem polinômios $Q_0, \ldots, Q_{r-1}$ *tais que qualquer sequência positiva admissível* $b = (b_0, \ldots, b_{N-1})^T$ *satisfaz*

$$Q_j(b) \;=\; 0, \quad j = 0, 1, \ldots, r-1. \tag{1.115}$$

O teorema de Sanz implica que o conjunto imagem de P,

$$\text{ImP} := P(\mathbb{R}^M) = \{b \in \mathbb{R}_+^N \;:\; \exists\, x \in \mathbb{R}^M \text{ tal que } P(x) = b\},$$

está contido em uma variedade algébrica. Mais precisamente

$$\text{ImP} \subset \{b \in \mathbb{R}_+^N \ : \ Q_0(b) = 0, \ldots, Q_{r-1}(b) = 0\}.$$

Além disso o teorema aborda apenas condições *necessárias* para que as amplitudes sejam dados admissíveis ao problema da fase. Para que as amplitudes fiquem completamente caracterizadas (em outras palavras para que determinemos o conjunto $P(\mathbb{R}^M)$) é necessário que conheçamos também as condições *suficientes*. Na subseção 1.4 foram encontradas 1 identidade desse tipo no caso 1-D, e $m_1 + m_2 + 1$ identidades LI no caso 2-D (veja teorema 1.8). Essas identidades tomam parte das condições necessárias de Sanz (1.115). Porém, acreditamos que para obtermos condições necessárias e *suficientes*, esse conjunto de equações deve ser complementado por outras identidades e *inequações* , nos $\alpha_u's$, que ainda permanecem desconhecidas, exceto para os casos abordados nos teoremas 1.2, 1.4 e 1.9. Para problemas maiores do que os apresentados nestes teoremas, encontrar tais condições diretamente torna-se uma tarefa praticamente impossível.

Definição 1.7 *O posto de uma aplicação diferenciável* $P : \mathbb{R}^M \longrightarrow \mathbb{R}^N$ *num ponto* x *é o posto da matriz do seu jacobiano*

$$J_P(x) : \mathbb{R}^M \longrightarrow \mathbb{R}^N, \quad J_P(x) = \left[\frac{\partial P_u}{\partial x_k}\right]_{N \times M}.$$

Denotaremos o posto de uma matriz J por $\rho(J)$. Evidentemente

$$\rho(J_P(x)) \leq \min\{N, M\}.$$

Será conveniente listarmos algumas propriedades elementares da Geometria Algébrica que relaciona o número, r, de equações polinomiais do teorema de Sanz com a dimensão N e o posto do jacobiano $J_P(x)$, onde P é o polinômio em (1.113). Suponhamos, então , $N \geq M$ e $\rho(J_P(x)) = M$ em quase todo ponto (abreviadamente q.t.p.) $x \in \mathbb{R}^M$. Denotemos por $\mathbb{R}[y]$ o anel dos polinômios em $y \in \mathbb{R}^N$ com coeficientes reais. Definimos os conjuntos

$$\begin{aligned} Z(\text{ImP}) &= \{Q \in \mathbb{R}[y] \ : \ Q(b) = 0, \ \forall\, b \in \text{ImP}\}, \\ \mathcal{C} &= \{b \in \mathbb{R}_+^N \ : \ Q(b) = 0, \ \forall\, Q \in Z(\text{ImP})\}. \end{aligned}$$

Segue da Geometria Algébrica [7], [50] os seguintes resultados elementares

1º) $Z(\text{ImP})$ é um ideal de $\mathbb{R}[y]$,

2º) $\mathcal{C}$ é a menor variedade algébrica que contém ImP, portanto,

$$\dim \mathcal{C} \geq \dim \text{ImP} = \rho(J_P(x)) = M,$$

3º) $Z(\text{ImP})$ é gerado por no máximo $N - \rho(J_P(x)) = N - M$ polinômios do anel $\mathbb{R}[y]$.

Com base nesses resultados, concluímos que se $\rho(J_P(x)) = M$, q.t.p. $x \in \mathbb{R}^M$, e se $Q_j(y) \in \mathbb{R}[y]$, $j = 0, 1, \ldots r-1$, são os polinômios de Sanz do teorema 1.10, então

$$\mathcal{C} = \{b \in \mathbb{R}_+^N \ : \ Q_j(b) = 0, \ j = 0, 1, \ldots, r-1\}, \tag{1.116}$$

com

$$r \leq N - \rho(J_P(x)) = N - M. \tag{1.117}$$

1.5.1 Polinômios de Sanz para o caso 1-D:

Usaremos a notação $\mathbf{f}_S$ para representar os valores de f dentro do suporte. Assim na representação de f em (1.4), $\mathbf{f}_S = f_{(\mathbf{1})}$. Segue da comparação feita em (1.111) que o polinômio $P_u(x)$ em (1.112) se reescreve como

$$\begin{aligned} P_u(\mathbf{f}_S) &= |(\mathcal{W}_{(\mathbf{1})}\mathbf{f}_S)_u|^2 = \mathbf{f}_S^T[\mathcal{W}_{(\mathbf{1})}^*]^{(u)}[\mathcal{W}_{(\mathbf{1})}]_{(u)}\mathbf{f}_S, \\ &= \sum_{j=0}^{m-1}\sum_{k=0}^{m-1} \omega^{u(k-j)} f_j f_k, \quad u = 0, 1, \ldots, m. \end{aligned} \tag{1.118}$$

onde $[\mathcal{W}_{(\mathbf{1})}]_{(u)}$, como já dissemos, representa a $u-$ésima coluna da matriz $\mathcal{W}_{(\mathbf{1})}$. Então resolver o problema da fase com restrição de suporte, (1.39), equivale, de acordo com (1.110), a achar $\mathbf{f}_S \in \mathbb{R}^m$ para a equação polinomial

$$P(\mathbf{f}_S) = b, \tag{1.119}$$

onde $b = (b_0, \ldots, b_m)^T$ é tal que $b_u = \alpha_u^2$, $u = 0, \ldots, m$, são dados do problema; e $P : \mathbb{R}^m \longrightarrow \mathbb{R}^{m+1}$ é o polinômio

$$P(\mathbf{f}_S) = (P_0(\mathbf{f}_S), P_1(\mathbf{f}_S), \ldots, P_m(\mathbf{f}_S))^T. \tag{1.120}$$

Sanz mostra em seu paper [74] que $\mathcal{W}_{(\mathbf{1})}$ satisfaz a propriedade (1.114), logo, de acordo com o teorema 1.10, se $\mathbf{f}_S$ é uma solução para (1.119) então existem r equações polinomiais nas variáveis $(b_0, \ldots, b_m)$. Segue imediatamente do corolário 1.1 (veja teorema 1.11) que uma destas equações é a identidade 1-D das amplitudes (1.62).

Proposição 1.4 *O jacobiano $J_P(\mathbf{f}_S)$ da aplicação $\mathbf{f}_S \longmapsto P(\mathbf{f}_S)$ definida por* (1.120) *e* (1.118) *é uma matriz real, $(m+1) \times m$, e satisfaz*

$$U(\mathbf{f}_S) = \frac{1}{n}\, \mathbf{M}_1\, \tilde{D}_1\, J_P(\mathbf{f}_S),$$

onde $U(\mathbf{f}_S)$ é a matriz Hankel+Toeplitz

$$U(\mathbf{f}_S) \;=\; H(\mathbf{f}_S) + T(\mathbf{f}_S),$$

para

$$H(\mathbf{f}_S) := \begin{bmatrix} f_0 & f_1 & f_2 & \cdots & f_{m-2} & f_{m-1} \\ f_1 & f_2 & f_3 & \cdots & f_{m-1} & 0 \\ f_2 & f_3 & f_4 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ f_{m-1} & 0 & 0 & \cdots & 0 & 0 \end{bmatrix}, \quad T(\mathbf{f}_S) := \begin{bmatrix} f_0 & f_1 & f_2 & \cdots & f_{m-2} & f_{m-1} \\ 0 & f_0 & f_1 & \cdots & f_{m-3} & f_{m-2} \\ 0 & 0 & f_0 & \cdots & f_{m-4} & f_{m-3} \\ \vdots & \vdots & \vdots & \cdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & f_0 \end{bmatrix},$$

$\mathbf{M}_1$ *é a matriz $m \times (m+1)$ cujo termo geral é*

$$(\mathbf{M}_1)_{ju} = 2\mathrm{Re}(\omega^{ju}), \quad 0 \le j \le m-1,\ 0 \le u \le m$$

e $\tilde{D}_1$ é matriz diagonal de ordem $m+1$ dada por

$$\tilde{D}_1 = \frac{1}{2} \oplus I_{m-1} \oplus \frac{1}{2}.$$

Consequentemente, $\rho(J_P(\mathbf{f}_S)) = m$ em quase todo ponto $\mathbf{f}_S \in \mathbb{R}^m$.

Demonstração : Com efeito, se derivarmos os dois membros da expressão de autocorrelação (1.63), quando $b_u = \alpha_u^2$, com relação a f_k, $k = 0, 1, \ldots, m-1$, obtemos para $j = 0, 1, \ldots, m-1$,

$$\sum_{i=0}^{m-1-j} \left[\frac{\partial f_i}{\partial f_k} f_{j+i} + f_i \frac{\partial f_{j+i}}{\partial f_k} \right] = \frac{1}{n} \left[\frac{\partial b_0}{\partial f_k} + 2 \sum_{u=1}^{m-1} \mathrm{Re}(\omega^{ju}) \frac{\partial b_u}{\partial f_k} + (-1)^j \frac{\partial b_m}{\partial f_k} \right]. \quad (1.121)$$

Verifica-se imediatamente que a forma matricial do segundo membro de (1.121) é o produto $(1/n)\ \mathbf{M}_1\ \tilde{D}_1\ J_P(\mathbf{f}_S)$. Falta mostrarmos que o primeiro membro é igual à matriz $U(\mathbf{f}_S)$. De fato, como

$$\frac{\partial f_i}{\partial f_k} = \delta_{ik} := \begin{cases} 1; & \text{se } i = k, \\ 0; & \text{se } i \neq k, \end{cases}$$

então o primeiro membro de (1.121) torna-se igual a $f_{k+j} + f_{k-j}$, para $0 \le j \le m-1$, $0 \le k \le m-1$. Levando em conta que

$$\begin{cases} f_{k+j} \;=\; 0, & \forall\, m-k \le j \le m, \\ f_{k-j} \;=\; 0, & \forall\, k+1 \le j \le m, \end{cases}$$

conclui-se que

$$[f_{k+j}]_{0\,\le\, j,k\,\le\, m-1} = H(\mathbf{f}_S), \quad [f_{k-j}]_{0\,\le\, j,k\,\le\, m-1} = T(\mathbf{f}_S).$$

Provamos assim que $U(\mathbf{f}_S) = (1/n)\ \mathbf{M}_1\ \tilde{D}_1\ J_P(\mathbf{f}_S)$. Finalmente mostremos que $J_P(\mathbf{f}_S)$ tem posto completo em quase todo ponto $\mathbf{f}_S \in \mathbb{R}^m$. De fato, o determinante de $U(\mathbf{f}_S)$ é um polinômio nas componentes $f_j's$ e, sobretudo, não identicamente nulo, pois $\det(U(\mathbf{f}_S)) = 2 \neq 0$ para $\mathbf{f}_S = (1, 0, \ldots, 0)^T \in \mathbb{R}^m$. Segue que o conjunto dos zeros da equação $\det(U(\mathbf{f}_S)) = 0$ forma um conjunto de medida nula em $\mathbb{R}^m$. Consequentemente $\rho(U(\mathbf{f}_S)) = m$ em quase todo ponto $\mathbf{f}_S$. Logo,

$$m = \rho(U(\mathbf{f}_S)) = \rho(\mathbf{M}_1\ \tilde{D}_1\ J_P(\mathbf{f}_S)) \le \rho(J_P(\mathbf{f}_S)) \le m, \quad \text{q.t.p. } \mathbf{f}_S$$

ou seja,

$$\rho(J_P(\mathbf{f}_S)) = m, \quad \text{q.t.p. } \mathbf{f}_S \qquad \blacksquare$$

Na seguinte proposição mostramos uma fórmula alternativa para o jacobiano que é associada com a controlabilidade de sistemas lineares [44].

Proposição 1.5 *Sejam* $f = (\mathbf{f}_S^T, 0)^T \in \mathbb{R}^n$, *e* $F = \mathcal{F}(f) \in \mathbb{R}^n$. *Denotemos por* $\mathbf{v}$ *o vetor das* $(m+1)$ *primeiras componentes de* F, *i.e.,*

$$\mathbf{v} := (F_0, \ldots, F_m)^T.$$

Então

$$J_P(\mathbf{f}_S) = 2\ \mathrm{Re}\left[\mathbf{v} \quad D\mathbf{v} \quad D^2\mathbf{v} \quad \ldots \quad D^{m-1}\mathbf{v}\right],$$

onde D *é a matriz diagonal* $D = \mathrm{diag}(\omega^0, \omega^1, \ldots, \omega^m)$, $\omega = \exp(\pi i/m)$. *Consequentemente a matriz* $\mathrm{Re}\left[\mathbf{v} \quad D\mathbf{v} \quad D^2\mathbf{v} \quad \ldots \quad D^{m-1}\mathbf{v}\right]$ *tem posto completo em quase todo ponto* $\mathbf{f}_S$.

Demonstração : Para obtermos esta expressão alternativa do jacobiano, devemos diferenciar o terceiro membro de (1.118) com respeito à componente f_k. Feito isso, temos

$$\begin{aligned}\frac{\partial P_u}{\partial f_k} &= 2\mathrm{Re}\left\{\frac{\partial \mathbf{f}_S^T}{\partial f_k}[\mathcal{W}_{(\mathbf{1})}^*]^{(u)}[\mathcal{W}_{(\mathbf{1})}]_{(u)}\mathbf{f}_S\right\} \\ &= 2\mathrm{Re}\left\{\mathcal{W}_{ku}^*[\mathcal{W}_{(\mathbf{1})}]_{(u)}\mathbf{f}_S\right\} = 2\mathrm{Re}\left\{\omega^{ku}F_u\right\},\end{aligned}$$

para todo $u = 0, 1, \ldots, m$, $k = 0, 1, \ldots, m-1$. Seja

$$\frac{\partial P}{\partial f_k} := (\frac{\partial P_0}{\partial f_k}, \ldots, \frac{\partial P_m}{\partial f_k})^T$$

a $k-$ésima coluna da matriz $J_P(\mathbf{f}_S)$. Segue então que

$$\frac{\partial P}{\partial f_k} = 2\mathrm{Re}\left\{D^k\mathbf{v}\right\}, \quad k = 0, \ldots, m-1.$$

Segue como consequência da proposição 1.4 que a matriz $2\ \mathrm{Re}\left[\mathbf{v} \quad D\mathbf{v} \quad D^2\mathbf{v} \quad \ldots \quad D^{m-1}\mathbf{v}\right]$ tem posto completo em quase todo ponto $\mathbf{f}_S$ $\blacksquare$

Teorema 1.11 *Consideremos o problema da fase 1-D* (1.119),

$$P(\mathbf{f}_S) = b, \quad \mathbf{f}_S \in \mathbb{R}^m,$$

para $P(\mathbf{f}_S) = (P_0(\mathbf{f}_S), P_1(\mathbf{f}_S), \ldots, P_m(\mathbf{f}_S))^T$, *onde cada* $P_u(\mathbf{f}_S)$ *é dado como em* (1.118). *Então para qualquer sequência admissível* $b = (b_0, \ldots, b_m) \in \mathbb{R}_+^{m+1}$,

$$Q_0(b_0, \ldots, b_m) = 0,$$

onde $Q_0(y) \in \mathbb{R}[y]$ *é o polinômio linear*

$$Q_0(y_0, \ldots, y_m) = y_0 + 2\sum_{k=1}^{m-1}(-1)^k y_k + (-1)^m y_m.$$

Além disso, se

$$Z(\mathrm{ImP}) = \{Q \in \mathbb{R}[y] \ : \ Q(b) = 0, \ \forall\, b \in \mathrm{ImP}\},$$

então $Z(\mathrm{ImP})$ *é gerado por* $Q_0(y)$, *i.e.*,

$$Z(\mathrm{ImP}) \ = < Q_0(y) > .$$

Demonstração: A existência do polinômio $Q_0(y)$ é consequência imediata do corolário 1.1. Agora, desde que $\rho(J_P(\mathbf{f}_S)) = m$, q.t.p. $\mathbf{f}_S \in \mathbb{R}^m$, segue do 3º) resultado elementar da Geometria Algébrica (abaixo da definição 1.7) e de (1.117) que o número máximo de polinômios que geram o ideal $Z(\mathrm{ImP})$ é $r = (m+1) - m = 1$ ∎

Como dissemos anteriormente, acreditamos que o conjunto $P(\mathbb{R}^m)$ seja descrito por equações e inequações nas amplitudes de Fourier. Nós encontramos $P(\mathbb{R}^m)$ para o casos $m = 2$ e $m = 3$. Com efeito, o conjunto $P(\mathbb{R}^2)$ é, de acordo com o teorema 1.2, dado por

$$P(\mathbb{R}^2) = \{b = (b_0, b_1, b_2) \in \mathbb{R}_+^3 : \ b_0 - 2b_1 + b_2 = 0\},$$

enquanto $P(\mathbb{R}^3)$ é, de acordo com a parte (a) do teorema 1.4, dado por

$$P(\mathbb{R}^3) = \{b = (b_0, b_1, b_2, b_3) \in \mathbb{R}_+^4 : \ b_0 - 2b_1 + 2b_2 - b_3 = 0, \ 4\sqrt{b_2} \ \geq \ |\sqrt{b_0} - 3\sqrt{b_3}| \ \},$$

1.5.2 Polinômios de Sanz para o caso 2-D:

Agora extenderemos os mesmos resultados da subseção anterior para o caso bidimensional. Por questões de simplicidade nos restringiremos ao caso de imagens quadradas ($m = m_1 = m_2$). Resultados similares aos que serão exibidos nesta subseção são facilmente extensíveis ao caso mais geral $m_1 \neq m_2$.

Suponhamos

$$f = \begin{bmatrix} f_{(11)} & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^n,$$

e $F = \mathcal{F}(f)$. Segue de (1.26), (1.14) e (1.16) que

$$\begin{aligned} F = \mathcal{W} f \mathcal{W} &= \begin{bmatrix} \mathbf{W} f_{(11)} \mathbf{W} & \mathbf{W} f_{(11)} \mathbf{D} \mathbf{W} \\ \mathbf{W} \mathbf{D} f_{(11)} \mathbf{W} & \mathbf{W} \mathbf{D} f_{(11)} \mathbf{D} \mathbf{W} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{W} \\ \mathbf{W} \mathbf{D} \end{bmatrix} f_{(11)} \begin{bmatrix} \mathbf{W} & \mathbf{D} \mathbf{W} \end{bmatrix} \\ &= \mathcal{W}_{(1)} f_{(11)} \mathcal{W}_{(1)}^T. \end{aligned}$$

A igualdade para o último membro da expressão acima decorre de $\mathbf{W}^T = \mathbf{W}$ e $\mathbf{D}^T = \mathbf{D}$. Também resulta desta mesma expressão que

$$F_{uv} = (\mathcal{W}_{(1)} f_{(11)} \mathcal{W}_{(1)}^T)_{uv} = [\mathcal{W}_{(1)}]_{(u)} f_{(11)} [\mathcal{W}_{(1)}^T]^{(v)}. \tag{1.122}$$

Para darmos a formulação 2-D do problema da fase equivalente a (1.119), será novamente conveniente que definamos o vetor $\mathbf{f}_\mathcal{S} \in \mathbb{R}^{m^2}$ em termos da matriz $f_{(11)} \in \mathbb{M}_m(\mathbb{R})$, de acordo com (1.99) e (1.100). Em outras palavras, se

$$f_q, \quad q \in \{0, 1, \ldots, m^2 - 1\},$$

são as componentes de $\mathbf{f}_\mathcal{S}$ e

$$f_{jk}, \quad (j, k) \in \{0, 1, \ldots, m-1\} \times \{0, 1, \ldots, m-1\},$$

são as entradas de $f_{(11)}$, então

$$f_q = f_{jk}, \tag{1.123}$$

quando j e k forem respectivamente o quociente e o resto da divisão de q por m, ou seja,

$$q = mj + k. \tag{1.124}$$

Assim, para o caso 2-D, resolver o problema da fase com restrição de suporte, (1.39), equivale, de acordo com (1.110), a achar $\mathbf{f}_\mathcal{S} \in \mathbb{R}^{m^2}$ para a equação polinomial

$$P(\mathbf{f}_\mathcal{S}) = b,$$

para

$$b = (b_{uv})_{(u,v)\in\Lambda} \in \mathbb{R}^{2m^2+2},$$

onde $b_{uv} = |F_{uv}|^2$ são dados do problema. O polinômio P em (1.119) é, neste caso, dado por

$$P(\mathbf{f}_S) = (P_{uv}(\mathbf{f}_S))_{(u,v)\in\Lambda}, \tag{1.125}$$

onde $P_{uv}(\mathbf{f}_S)$ é, de acordo com (1.122),

$$P_{uv}(\mathbf{f}_S) = F_{uv}^* F_{uv} = [\ \overline{\mathcal{W}}_{(1)}\ ]_{(v)}\ f_{(\mathbf{11})}^T\ [\ \overline{\mathcal{W}}_{(1)}^T\ ]^{(u)}\ [\mathcal{W}_{(\mathbf{1})}]_{(u)}\ f_{(\mathbf{11})}\ [\mathcal{W}_{(\mathbf{1})}^T]^{(v)}, \tag{1.126}$$

para todo $(u,v) \in \Lambda$, Λ dado por (1.31). Aqui $\overline{\mathcal{W}}_{(1)}$ simboliza o complexo conjugado de $\mathcal{W}_{(\mathbf{1})}$.

Segue agora a Proposição do caso 2-D, análoga à Proposição 1.4.

Proposição 1.6 *O jacobiano $J_P(\mathbf{f}_S)$ da aplicação $\mathbf{f}_S \longmapsto P(\mathbf{f}_S)$ definida por* (1.125), (1.126) *é uma matriz real, $(2m^2+2)\times m^2$, e satisfaz*

$$U(\mathbf{f}_S) = \frac{1}{n^2}\,\mathbf{M}_2\,\tilde{D}_2\,J_P(\mathbf{f}_S),$$

onde $U(\mathbf{f}_S)$ é a matriz Hankel por blocos + Toeplitz por blocos

$$U(\mathbf{f}_S) \;=\; H(\mathbf{f}_S) + T(\mathbf{f}_S),$$

para

$$H(\mathbf{f}_S) := \begin{bmatrix} H_0 & H_1 & \dots & H_{m-2} & H_{m-1} \\ H_1 & H_2 & \dots & H_{m-1} & 0 \\ \dots & \dots & \dots & \dots & \dots \\ H_{m-2} & H_{m-1} & \dots & 0 & 0 \\ H_{m-1} & 0 & \dots & 0 & 0 \end{bmatrix},\quad T(\mathbf{f}_S) := \begin{bmatrix} T_0 & T_1 & \dots & T_{m-2} & T_{m-1} \\ 0 & T_0 & \dots & T_{m-3} & T_{m-2} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & T_0 & T_1 \\ 0 & 0 & \dots & 0 & T_0 \end{bmatrix},$$

onde cada bloco H_j é matriz de hankel e cada bloco T_j é matriz de Toeplitz, dadas respectivamente por

$$H_j = \begin{bmatrix} f_{j0} & f_{j1} & \dots & f_{j,m-2} & f_{j,m-1} \\ f_{j1} & f_{j2} & \dots & f_{j,m-1} & 0 \\ \dots & \dots & \dots & \dots & \dots \\ f_{j,m-2} & f_{j,m-1} & \dots & 0 & 0 \\ f_{j,m-1} & 0 & \dots & 0 & 0 \end{bmatrix},\quad T_j := \begin{bmatrix} f_{j0} & f_{j1} & \dots & f_{j,m-2} & f_{j,m-1} \\ 0 & f_{j0} & f_{j1} & \dots & f_{j,m-2} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & f_{j0} & f_{j1} \\ 0 & 0 & \dots & 0 & f_{j0} \end{bmatrix}.$$

A matriz $\mathbf{M}_2$ é de ordem $m^2 \times (2m^2+2)$, cuja representação por blocos é

$$\mathbf{M}_2 = \begin{bmatrix} M_{20} & M_{21} & \dots & M_{2m} \end{bmatrix},$$

onde M_{2t}, $t = 0, m$, *são de ordem* $m^2 \times (m+1)$, *com termo geral*

$$(M_{2t})_{mj+k,v} = 2\mathrm{Re}(\omega^{jt+kv}), \quad 0 \leq v \leq m, \;\; 0 \leq j,k \leq m-1,$$

e M_{2t}, $0 \leq t \leq m-1$, *são de ordem* $m^2 \times n$, *com termo geral*

$$(M_{2t})_{mj+k,v} = 2\mathrm{Re}(\omega^{jt+kv}), \quad 0 \leq v \leq n-1, \;\; 0 \leq j,k \leq m-1.$$

A matriz $\tilde{D}_2$ *é diagonal, de ordem* $2m^2+2$, *expressa em soma direta como*

$$\tilde{D}_2 = \tilde{D}_1 \oplus I_{n(m-1)} \oplus \tilde{D}_1,$$

quando

$$\tilde{D}_1 = \frac{1}{2} \oplus I_{m-1} \oplus \frac{1}{2}.$$

Consequentemente, $\rho(J_P(\mathbf{f}_S)) = m^2$ *em quase todo ponto* $\mathbf{f}_S \in \mathbb{R}^{m^2}$.

Demonstração: A demonstração é totalmente análoga à da Proposição 1.4, bastando-nos apenas acompanhá-la em cada passo, fazendo-se as devidas adaptações ao caso 2-D das expressões que lá aparecem. De fato, observa-se que

$$\frac{\partial f_{pq}}{\partial f_{rs}} = \begin{cases} 1; & \text{se } (p,q) = (r,s), \\ 0; & \text{caso contrário,} \end{cases} \tag{1.127}$$

e

$$f_{pq} \;=\; 0, \quad \begin{array}{l}\text{sempre que } p \text{ e } q \text{ não pertencerem ao} \\ \text{suporte } [0, m-1] \times [0, m-1].\end{array} \tag{1.128}$$

Diferenciando o primeiro membro da expressão de autocorrelação (1.77) para $m_1 = m_2 = m$, e valores de $j,k = 0, 1, \ldots, m-1$, temos

$$\sum_{p=0}^{m-1-j} \sum_{q=0}^{m-1-k} \left[\frac{\partial f_{pq}}{\partial f_{rs}} f_{j+p,k+q} + f_{pq} \frac{\partial f_{j+p,k+q}}{\partial f_{rs}} \right].$$

Em seguida, aplicando (1.127), obtemos

$$f_{j+r,k+s} + f_{r-j,s-k}, \quad 0 \;\leq\; j,k,r,s \;\leq m-1,$$

cuja estrutura de blocos, após aplicação de (1.128), nada mais é do que a soma das matrizes Hankel por blocos, $H(\mathbf{f}_S)$, e Toeplitz por blocos, $T(\mathbf{f}_S)$, que aparecem definidas no enunciado da presente proposição. Depois, fazendo $b_{uv} = \alpha_{uv}^2$ e $m_1 = m_2 = m$ no segundo membro de (1.77) e, então, derivando-o com relação a f_{rs}, para $j,k = 0, 1, \ldots, m-1$ e $r,s \in \{0, 1, \ldots, m-1\}$, o que se obtem é exatamente o

produto $(1/n^2)\ \mathbf{M}_2\ \tilde{D}_2\ J_P(\mathbf{f}_S)$, com as matrizes $\mathbf{M}_2$ e $\tilde{D}_2$ definidas de acordo com o enunciado desta proposição e

$$J_P(\mathbf{f}_S) = \left[\frac{\partial b_{uv}}{\partial f_q}\right], \quad (u,v) \in \Lambda, \ \ q \in \{0,1,\ldots,m^2-1\}, \tag{1.129}$$

para $q = mr + s, \ \ r, s, \in \{0,1,\ldots,m^2-1\}$, ou seja, o jacobiano da aplicação $b = P(\mathbf{f}_S)$. Finalmente mostremos que $J_P(\mathbf{f}_S)$ tem posto completo em quase todo ponto $\mathbf{f}_S \in \mathbb{R}^{m^2}$. De fato, como o determinante de $U(\mathbf{f}_S)$ é um polinômio nas componentes $f'_{pq}s$ e, sobretudo, não identicamente nulo (pois, $\det(U(\mathbf{f}_S)) = [\det(H_0 + T_0)]\ [\det T_0]^{m-1} = 2 \neq 0$, para $\mathbf{f}_S = (1,0,\ldots,0)^T \in \mathbb{R}^{m^2}$), segue que $\{\mathbf{f}_S \in \mathbb{R}^{m^2} : \det(U(\mathbf{f}_S)) = 0\}$ tem medida nula. Consequentemente

$$m^2 = \rho(U(\mathbf{f}_S)) = \rho(\mathbf{M}_2\ \tilde{D}_2\ J_P(\mathbf{f}_S)) \leq \rho(J_P(\mathbf{f}_S)) \leq m^2, \quad \text{q.t.p. } \mathbf{f}_S \in \mathbb{R}^{m^2},$$

ou seja,

$$\rho(J_P(\mathbf{f}_S)) = m^2, \quad \text{q.t.p. } \mathbf{f}_S \in \mathbb{R}^{m^2} \qquad \blacksquare$$

Segue agora o análogo 2-D da Proposição 1.5

Proposição 1.7 *Sejam $f \in \mathbb{M}_n(\mathbb{R})$ imagem real com restrição de suporte, conforme* (1.98)*, e $F = \mathcal{F}(f) \in \mathbb{M}_n(\mathbb{C})$. Seja $\mathbf{f}_S \in \mathbb{R}^{m^2}$, definida por* (1.123) *e* (1.124)*. Definimos o vetor*

$$\mathbf{v} := (F_{uv})_{(u,v)\in\Lambda} \in \mathbb{C}^{2m^2+2}$$

Então a matriz jacobiana $J_P(\mathbf{f}_S)$ satisfaz

$$J_P(\mathbf{f}_S) = 2\ \mathrm{Re}\left[\mathbf{v} \quad D_1\mathbf{v} \quad D_2\mathbf{v} \quad \ldots \quad D_{m^2-1}\mathbf{v}\right],$$

onde cada D_q, $q = 0,1,\ldots,m^2-1$, é matriz diagonal, de ordem $(2m^2+2)$, tal que

$$D_q = D^j \tilde{D}^k,$$

para D e $\tilde{D}$, também matrizes diagonais de ordem $(2m^2+2)$, definidas por

$$D = I_{m+1} \oplus\ \omega I_n\ \oplus\ \ldots\ \oplus\ \omega^{m-1} I_n\ \oplus\ (-1) I_{m+1},$$

e

$$\tilde{D} = \tilde{\tilde{D}}_1 \oplus \underbrace{\tilde{\tilde{D}}_2 \oplus \ldots \oplus \tilde{\tilde{D}}_2}_{m-1 \text{ termos}} \oplus \tilde{\tilde{D}}_1,$$

com

$$\tilde{\tilde{D}}_1 = \mathrm{diag}(1,\omega,\ldots,\omega^{m-1},-1); \quad \tilde{\tilde{D}}_2 = \mathrm{diag}(1,\omega,\ldots,\omega^{n-1}).$$

Os inteiros j e k são respectivamente o quociente e o resto da divisão de q por m. Consequentemente a matriz $\mathrm{Re}\left[\mathbf{v} \quad D_1\mathbf{v} \quad D_2\mathbf{v} \quad \ldots \quad D_{m^2-1}\mathbf{v}\right]$ tem posto completo em quase todo ponto $\mathbf{f}_S$.

Demonstração : Vimos que o jacobiano $J_P(\mathbf{f}_S)$ da aplicação $\mathbf{f}_S \longmapsto P(\mathbf{f}_S)$, no caso 2-D, é a matriz real $(2m^2+2) \times m^2$ dada por (1.129), com $f_q = f_{jk}$, $q = mj + k$. Derivando parcialmente o terceiro membro de (1.126) com respeito à coordenada f_q de $\mathbf{f}_S$, obtemos:

$$\begin{aligned}
\frac{\partial P_{uv}}{\partial f_q} \equiv \frac{\partial P_{uv}}{\partial f_{jk}} &= 2\mathrm{Re}\left\{ [\,\overline{\mathcal{W}}_{(1)}\,]_{(v)} \frac{\partial \mathbf{f}_S^T}{\partial f_{jk}} [\,\overline{\mathcal{W}}_{(1)}^T\,]^{(u)} [\mathcal{W}_{(\mathbf{1})}]_{(u)}\, \mathbf{f}_S\, [\mathcal{W}_{(\mathbf{1})}^T]^{(v)} \right\} \\
&= 2\mathrm{Re}\left\{ \mathcal{W}_{ju}^*\, \mathcal{W}_{kv}^*\, [\mathcal{W}_{(\mathbf{1})}]_{(u)}\, \mathbf{f}_S\, [\mathcal{W}_{(\mathbf{1})}^T]^{(v)} \right\} \\
&= 2\mathrm{Re}\left\{ \omega^{ju}\, \omega^{kv}\, F_{uv} \right\}.
\end{aligned}$$

Consequentemente

$$\frac{\partial P}{\partial f_q} = 2\mathrm{Re}\left\{ D_q \mathbf{v} \right\}, \quad q = 0, 1, \ldots, m^2 - 1,$$

para $\mathbf{v}$ e D_q definidos como no enunciado desta proposição, onde

$$\frac{\partial P}{\partial f_q} := [\frac{\partial P_{uv}}{\partial f_q}]_{(u,v)\in\Lambda},$$

denota a $q-$ésima coluna da matriz $J_P(\mathbf{f}_S)$. O fato de $\mathrm{Re}\left[\mathbf{v} \;\; D_1\mathbf{v} \;\; D_2\mathbf{v} \;\; \ldots \;\; D_{m^2-1}\mathbf{v}\right]$ ter posto completo em quase todo ponto $\mathbf{f}_S$ é uma consequência da Proposição 1.6 ■

O teorema a seguir é uma consequência imediata dos teoremas 1.10 e 1.8.

Teorema 1.12 *Consideremos o problema da fase* (1.119),

$$P(\mathbf{f}_S) = b, \quad \mathbf{f}_S \in \mathbb{R}^{m^2},$$

para P definido por (1.125) *e* (1.126). *Então para qualquer sequência admissível* $b = (b_{uv})_{(u,v)\in\Lambda} \in \mathbb{R}_+^{2m^2+2}$,

$$Q_0(b) = 0,\; Q_1(b) = 0,\; \ldots,\; Q_{2m}(b) = 0,$$

onde $Q_0, Q_1, \ldots, Q_{2m} \in \mathbb{R}[y]$; $y = (y_{00}, y_{0,1}, \ldots, y_{mm})^T \in \mathbb{R}^{2m^2+2}$, *são os polinômios lineares*

$$\begin{aligned}
Q_v(y) &= y_{0v} + 2\sum_{u=1}^{m-1} (-1)^u y_{uv} + (-1)^m y_{mv}, && v = 0, m, \\
Q_v(y) &= \sum_{u=0}^{m} (-1)^u y_{uv} + \sum_{u=1}^{m-1} (-1)^u y_{u,n-v}, && 1 \le v \le m-1, \\
Q_{m+u}(y) &= \sum_{v=0}^{n-1} (-1)^v y_{uv}, && 1 \le u \le m-1, \\
Q_{2m}(y) &= y_{m0} + 2\sum_{v=1}^{m-1} (-1)^v y_{mv} + (-1)^m y_{mm}.
\end{aligned}$$

Além disso, se

$$Z(\mathrm{ImP}) = \{Q \in \mathbb{R}[y] \ : \ Q(b) = 0, \ \ \forall\, b \in \mathrm{ImP}\},$$

então $Z(\mathrm{ImP})$ *é gerado por no máximo*

$$2m^2 + 2 - \rho(J_P(\mathbf{f}_S)) = m^2 + 2 \tag{1.130}$$

polinômios do anel $\mathbb{R}[y]$, *dentre eles* $Q_j(y)$, $j = 0, 1, \ldots, 2m$.

Demonstração : Inteiramente análoga ao caso 1-D (veja Teorema 1.11), logo, será omitida ■

Observemos que (1.130) é válida só para $\mathbf{f}_S$ genérico.

De acordo com o teorema 1.12, o número de identidades polinomiais que faltam para obtermos um conjunto de geradores do anel $Z(\mathrm{ImP})$ é de no máximo

$$(m-1)^2 \quad \text{identidades.}$$

Para os casos particulares $m_1 = m_2 = 2$ e $m_1 = 2$, $m_2 = 3$, as identidades restantes (além daquelas que já conhecemos pelo Teorema 1.8) que nós encontramos não são identidades em termos dos $b'_{uv}s$, mas sim em termos dos $\alpha'_{uv}s$. Para o caso $m_1 = m_2 = 2$, a identidade encontrada está dada em (1.107). Para $m_1 = 2$, $m_2 = 3$, as outras 2 identidades serão exibidas nos próximos parágrafos. Os cálculos para obtê-las foram muito mais difíceis e trabalhosos do que os do caso 2×2.

Podemos, através do teorema 1.9, dar a caracterização completa das amplitudes para o problema inverso da fase correspondente ao caso 2×2. Com efeito, o conjunto admissível no caso 2×2 é

$$P(\mathbb{R}^4) = \bigcup_{\varepsilon} \mathcal{C}_\varepsilon$$

quando

$$\mathcal{C}_\varepsilon := \left\{ \begin{array}{ll} b \in \mathbb{R}^{10}_+ \ : & Q_0(b) = 0, \ \ Q_1(b) = 0, \ \ \ldots\ldots \ , Q_4(b) = 0, \\ & b_{01} - 2b_{11} + b_{20} = (\epsilon_{00}\sqrt{b_{00}})(\epsilon_{22}\sqrt{b_{22}}) - (\epsilon_{02}\sqrt{b_{02}})(\epsilon_{20}\sqrt{b_{20}}). \end{array} \right\}.$$

e $\varepsilon := (\epsilon_{00}, \epsilon_{02}, \epsilon_{20}, \epsilon_{22}) = (\pm 1, \pm 1, \pm 1, \pm 1)$.

Levando em conta a restrição de positividade (1.47); temos $P(\mathbb{R}^4_+) = \bigcup_\varepsilon \mathcal{C}_\varepsilon$ para

$$\mathcal{C}_\varepsilon := \left\{ \begin{array}{ll} b \in \mathbb{R}^{10}_+ \ : & Q_0(b) = 0, \ \ Q_1(b) = 0, \ \ \ldots\ldots \ , Q_4(b) = 0, \\ & b_{01} - 2b_{11} + b_{20} = (\sqrt{b_{00}})(\epsilon_{22}\sqrt{b_{22}}) - (\epsilon_{02}\sqrt{b_{02}})(\epsilon_{20}\sqrt{b_{20}}) \ , \\ & \sqrt{b_{00}} + \sqrt{b_{20}} \ > \ \sqrt{b_{02}} + \sqrt{b_{22}} \ , \ \ \sqrt{b_{00}} - \sqrt{b_{20}} \ > \ |\sqrt{b_{02}} - \sqrt{b_{22}}\,|. \end{array} \right\},$$

para algum $\varepsilon := (1, \epsilon_{02}, \epsilon_{20}, \epsilon_{22})$. Os polinômios $Q_{j's}$ são dados pelas expressões do lado esquerdo de (1.106) quando $b = (b_{uv}) \equiv (\alpha^2_{uv})$, com

$$(u, v) \in \Lambda = \{(0,0), (0,1), (0,2), (1,0), (1,1), (1,2), (1,3), (2,0), (2,1), (2,2)\}.$$

1.5.3 Identidades L.I. para imagens 2×3

Como dissemos no início da subseção anterior, todos os resultados correspondentes a imagens quadradas podem ser extendidos a imagens retangulares. Neste caso, nossos Teoremas adaptados ao caso retangular nos diz que

- $\rho(J_P(\mathbf{f}_S)) = m_1 m_2$ em quase todo ponto $\mathbf{f}_S \in \mathbb{R}^{2m_1 m_2+2}$,
- para dados genéricos existem no máximo $(2m_1m_2 + 2) - \rho(J_P(\mathbf{f}_S)) = m_1m_2+2$ identidades polinomiais L.I. nas variáveis b_{uv}. Nós encontramos $m_1 + m_2 + 1$ delas que estão exibidas em (1.95). Portanto o número de identidades L.I. ainda não conhecidas é de no máximo $(m_1 - 1)(m_2 - 1)$, podendo não serem todas elas identidades polinomiais nas variáveis b_{uv}.

Nesta subseção vamos estabelecer todas as $m_1m_2 + 2 = 8$ identidades L.I. nas variáveis b_{uv} quando $m_1 = 2$, $m_2 = 3$. Deste total nós já conhecemos as 6 identidades polinomiais lineares dadas por (1.95). Falta portanto determinarmos as 2 identidades restantes. Estas identidades, como veremos, não são identidades polinomiais nas variáveis $b'_{uv}s$, mas sim nos $\alpha'_{uv}s$.

Para obtermos todas as identidades que faltam, tivemos que resolver o sistema (1.101) para as incógnitas f_q, fazendo uso inclusive das equações de (1.102). Esse processo torna-se extremamente exaustivo à medida que os valores de m_1 e m_2 crescem. É impressionante como o nível de complexidade das equações aumenta quando apenas passamos do caso 2×2 para 2×3. Assim, por causa de tamanha complexidade, todos os cálculos referente à resolução do sistema serão omitidos.

Suponhamos

$$f = \begin{bmatrix} f_0 & f_1 & f_2 & 0 & 0 & 0 \\ f_3 & f_4 & f_5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \tag{1.131}$$

logo $\mathbf{f}_S = (f_0, f_1, f_2, f_3, f_4, f_5)^T$. Sejam $F = \mathcal{F}(f)$ e $b = (b_{uv})_{(u,v)\in\Lambda}$ para $b_{uv} = \alpha_{uv}^2 = |F_{uv}|^2$. Consideremos o sistema (1.101) - (1.102), adaptado ao caso 2×3. Troquemos o termo $||F||^2$ que aparece em (1.101) pelo seu valor equivalente $\sum_{u=0}^{3}\sum_{v=0}^{5} b_{uv}$. Mostra-se, neste caso, que o sistema (1.101) - (1.102) é equivalente ao seguinte sistema

$$\begin{array}{lll} \text{(i)} & f_1 = K, & f_4 = L, \\ \text{(ii)} & f_0 + f_2 = M, & f_3 + f_5 = N, \\ \text{(iii)} & Lf_0 + Kf_5 = R, & Nf_0 - Mf_3 = S, \\ \text{(iv)} & f_0f_5 + f_2f_3 - f_0f_3 - f_2f_5 = T, & \\ \text{(v)} & f_0f_2 + f_3f_5 = U & \\ \text{(vi)} & Q_j(b) = 0,\ j = 0, 1, \ldots, 5. & \end{array} \tag{1.132}$$

para

$$
\begin{aligned}
K &= [(F_{00}+F_{20})-(F_{03}+F_{23})]/4,\\
L &= [(F_{00}-F_{20})-(F_{03}-F_{23})]/4,\\
M &= [(F_{00}+F_{20})+(F_{03}+F_{23})]/4,\\
N &= [(F_{00}-F_{20})+(F_{03}-F_{23})]/4,\\
R &= \sqrt{3}(-b_{11}-b_{12}+b_{14}+b_{15})/24+(b_{01}-b_{02}-b_{21}+b_{22})/8,\\
S &= \sqrt{3}(-b_{11}+b_{12}-b_{14}+b_{15})/12,\\
T &= [(b_{01}-9b_{02}+5b_{03})-(b_{21}-9b_{22}+5b_{23})-3(F_{00}F_{03}-F_{20}F_{23})]/24,\\
U &= [b_{00}+b_{03}+b_{20}+b_{23}-4(b_{12}-b_{13}+b_{14})]/24.
\end{aligned}
$$

Os polinômios Q_j são os polinômios do Teorema 1.12.

As equações em (i) e (ii) do sistema (1.132) nada mais são do que uma combinação das equações do sistema (1.102). As equações em (vi) são as identidades 2-D das amplitudes para imagens 2×3 (veja teorema 1.12 adaptado a imagens 2×3 ou expressões (1.95)). Notemos que o sistema original (1.101) - (1.102) é composto de 18 equações, enquanto que o seu equivalente (1.132) contém 14. Isto ocorreu porque após a realização de operações elementares sobre as linhas do sistema original, quatro equações tornaram-se redundantes e, portanto, foram descartadas.

O nosso problema inverso então consiste na determinação do vetor $\mathbf{f}_S$ cujas componentes satisfazem todas as equações de (1.132), onde supõem -se conhecidos apenas os números reais positivos b_{uv}, $(u,v) \in \Lambda$. Notemos que os números $F_{uv} \equiv \epsilon_{uv}\sqrt{b_{uv}}$, $(u,v) \in \{0,m_1\}\times\{0,m_2\}$, não são totalmente conhecidos, pois eles também dependem dos valores de $\epsilon_{uv} \in \{-1,1\}$. Por isso resolvemos primeiramente o sistema (1.132) para as componentes de $\mathbf{f}_S$ sem nos preocuparmos *a priori* com os valores exatos dos ϵ_{uv}. Com isso obtemos as componentes do vetor $\mathbf{f}_S$ em termos dos números positivos b_{uv} e dos números reais F_{00}, F_{03}, F_{20} e F_{23}. Como existem 16 possibilidades de atribuição de valores ao vetor $\varepsilon = (\epsilon_{00}, \epsilon_{03}, \epsilon_{20}, \epsilon_{23}) \in \{(\pm 1, \pm 1, \pm 1, \pm 1)\}$, teremos portanto 16 possibilidades de solução para $\mathbf{f}_S$. Devemos escolher aquela que corresponde à solução verdadeira (1.131). O procedimento para tal escolha é simples. Tomemos um dos vetores $\mathbf{f}_S$ encontrados. A seguir formemos a matriz 2×3 como em (1.131), tendo como elementos do suporte as componentes do vetor $\mathbf{f}_S$ escolhido. Em seguida, calculemos $F = \mathcal{F}(f)$ e suas amplitudes $\alpha_{uv} = |F_{uv}|$. Se α_{uv}^2 for igual ao dado do problema b_{uv}, para cada $(u,v) \in \Lambda$, o procedimento termina; senão escolhe-se outro candidato para $\mathbf{f}_S$ e repete-se o procedimento acima. A escolha correta do vetor ε implica a determinação do toróide (veja definição 1.4) sobre o qual está a solução verdadeira f.

Notemos que as equações (i), (ii), (iii) representam um conjunto de 6 equações lineares que, sob determinadas condições de existência e unicidade, podem ser resolvidas para as 6 incógnitas que definem $\mathbf{f}_S$. Uma vez tendo encontradas estas incógnitas, devemos substituí-las nas equações (iv) e (v) para obtermos, portanto, as 2 novas identidades.

Em (i) já temos os valores de f_1 e f_4. Para encontrarmos as outras incógnitas, usamos eliminação gaussiana [32] no sistema linear formado por (ii) e (iii):

$$\begin{array}{rcl} f_0 &=& 4[M(R-NK)-SK]/(F_{00}F_{23}-F_{03}F_{20}); \\ f_2 &=& -4[M(R-ML)-SK]/(F_{00}F_{23}-F_{03}F_{20}); \\ f_3 &=& 4[N(R-NK)-SL]/(F_{00}F_{23}-F_{03}F_{20}); \\ f_5 &=& -4[N(R-ML)-SL]/(F_{00}F_{23}-F_{03}F_{20}). \end{array} \tag{1.133}$$

Durante o processo da eliminação foi necessário supor $F_{00}+F_{03} \neq F_{20}+F_{23}$ e $F_{00}F_{23} \neq F_{03}F_{20}$ para se chegar em (1.133). Como estas condições raramente ocorrem, conclui-se que as componentes definidas em (1.133) são soluções genéricas do sistema (ii) e (iii). Finalmente substituindo (1.133) em (iv) e (v) encontramos as 2 novas identidades

$$\begin{array}{rcl} c_0 + c_1F_{00}F_{03} + c_2F_{20}F_{23} + c_3F_{00}F_{03}F_{20}F_{23} &=& 0, \\ d_0 + d_1F_{00}F_{03} + d_2F_{20}F_{23} + d_3F_{00}F_{03}F_{20}F_{23} &=& 0. \end{array} \tag{1.134}$$

onde

$$\begin{array}{rcl} c_0 &=& 2(b_{20}s_1^2 + b_{23}s_2^2) + 2(b_{00}b_{23}+b_{03}b_{20})(b_{20}-16b_{22}+9b_{23}), \\ c_1 &=& 24b_{20}b_{23}, \\ c_2 &=& -4[3(b_{00}b_{23}+b_{03}b_{20}) + s_1s_2], \\ c_3 &=& -4(b_{20}-16b_{22}+9b_{23}), \\ d_0 &=& 2(b_{00}s_1^2 + b_{03}s_2^2) + 2(b_{00}b_{23}+b_{03}b_{20})(b_{00}-16b_{02}+9b_{03}), \\ d_1 &=& -4[3(b_{00}b_{23}+b_{03}b_{20}) + s_1s_2], \\ d_2 &=& 24b_{00}b_{03}, \\ d_3 &=& -4(b_{00}-16b_{02}+9b_{03}), \\ s_1 &=& b_{11}-3b_{12}+3b_{14}-b_{15}, \\ s_2 &=& 3b_{11}-b_{12}+b_{14}-3b_{15}. \end{array}$$

Observemos a diferença do nível de complexidade entre as novas identidades dos casos 2×2 e 2×3. De fato, se reescrevemos (1.107) em termos de b_{uv} e F_{uv} teremos

$$(b_{01}-2b_{11}+b_{21}) - F_{00}F_{22} + F_{02}F_{20} = 0.$$

Como se vê, esta equação é bem mais simples do que as equações (1.134).

Capítulo 2

Os algoritmos iterativos e outros métodos de reconstrução da fase

Neste capítulo faremos uma breve exposição dos principais métodos existentes para o problema de recuperaração da fase de Fourier. Daremos uma ênfase maior nos *algoritmos iterativos* por serem considerados o *estado da arte* para a solução deste problema. Além da definição, demonstraremos alguns resultados teóricos e faremos alguns comentários numéricos a respeito da convergência desses métodos para os casos 1-D e 2-D. Uma parte destes comentários será feita no capítulo 4 onde resultados serão exibidos e analisados. Os outros métodos serão brevemente discutidos na última seção deste capítulo.

Na duas próximas seções discutiremos, então, dois algoritmos iterativos básicos: o *Error-Reduction* (ER) e o *Hybrid Input-Output* (HIO); e na seção seguinte enunciaremos aquele que parece ser o mais eficiente dos algoritmos iterativos, o qual é obtido por uma combinação do ER e do HIO.

Existem outros algoritmos iterativos, dos quais não faremos nenhum comentário, que foram abordados por Fienup tais como o *algoritmo básico Input-Output* e o *algoritmo Output-Output*. Maiores detalhes sobre estes métodos podem ser encontrados em [25], [5], [82] e [73].

2.1 Os algoritmos Error-Reduction (ER) e Hybrid Input-Output (HIO)

Os algoritmos iterativos foram extensamente estudados por Gerchberg, Saxton e Fienup, e têm sido considerados métodos de muito sucesso para o problema da fase. Apesar de serem métodos amplamente difundidos e utilizados neste tipo de problema, teorias matemáticas, até hoje, não conseguem explicar o bom funcionamento deles. Eles funcionam para os tipos mais gerais de objetos, usam todos os dados e

restrições disponíveis e não são computacionalmente caros como a maioria dos outros métodos tais como o nosso novo método, que será descrito no próximo capítulo. Entretanto são métodos que também apresentam vários problemas como por exemplo estagnação, mal-condicionamento e ambiguidades. Não é nosso propósito fazer uma descrição rigorosa e detalhada destes métodos, tendo em vista que eles já foram extensamente abordados e divulgados em diversos artigos. Maiores detalhes relacionados, por exemplo, a regiões de estagnação, taxa de convergência, mal-condicionamento, regiões planares (ou *plateaus*), análise de erro e soluções ambíguas, poderão ser encontradas em [25], [81] e [82]. Também no capítulo 8 de [82] e em [5] encontra-se uma abordagem destes algoritmos como um caso especial do método de projeções não lineares. Especificamente, a referência [5] contém resultados mais recentes que estabelecem novas conexões entre os algoritmos iterativos e os métodos clássicos de otimização convexa.

A maioria dos algoritmos iterativos pode ser considerada uma variação do algoritmo de Gerchberg-Saxton ([31], [30] e [78]). Trata-se de procedimentos iterativos de idas e vindas sucessivas entre os espaços do domínio do objeto (onde conhecimentos a priori sobre o objeto, tais como não -negatividade e suporte, são aplicados) e o domínio de Fourier (onde os dados dos módulos de Fourier são aplicados). O mais simples deles, e o primeiro a ser considerado por Fienup, é o *Error-Reduction* [25], [82]. Ele é usado na recuperação da fase ϕ_u, da transformada de Fourier, F_u, de um objeto real não -negativo, f_j, a partir do módulo de Fourier, $|F_u|$. Segue, então, os 4 passos básicos (correspondentes à k-ésima iteração) que compõem o algoritmo ER: (1) calcula-se a transformada de Fourier, $G_u^{(k)}$, do valor estimado, $g_j^{(k)}$, de f_j; (2) efetua-se mudança na amplitude de $G_u^{(k)}$, de modo que a restrição no domínio de Fourier seja satisfeita, para formar $H_u^{(k)}$, uma estimativa de F_u; (3) calcula-se a transformada inversa de Fourier de $H_u^{(k)}$, obtendo-se $h_j^{(k)}$; e (4) efetua-se mudanças em $h_j^{(k)}$, de modo que a restrição no domínio do objeto seja satisfeita, para formar uma nova estimativa, $g_j^{(k+1)}$, do objeto. Em termos matemáticos, os 4 passos são :

$$\begin{aligned}
&\text{Passo 1:} \quad G_u^{(k)} = |G_u^{(k)}| \exp[i\phi_u^{(k)}] = \mathcal{F}(g_j^{(k)}),\\
&\text{Passo 2:} \quad H_u^{(k)} = |F_u| \exp[i\phi_u^{(k)}] = |F_u| \frac{G_u^{(k)}}{|G_u^{(k)}|},\\
&\text{Passo 3:} \quad h_j^{(k)} = \mathcal{F}^{-1}(H_u^{(k)}),\\
&\text{Passo 4:} \quad g_j^{(k+1)} = \begin{cases} h_j^{(k)}, & j \notin \gamma, \\ 0, & j \in \gamma, \end{cases}
\end{aligned} \tag{2.1}$$

onde γ é o conjunto de pontos para os quais $h_j^{(k)}$ viola as restrições no domínio do objeto, i.e., as restrições de não -negatividade e de suporte. Aqui, $g^{(k)}, H^{(k)}$ e $\phi^{(k)}$ são as estimativas de f, F e a fase ϕ de F respectivamente.

O algoritmo *Hybrid Input-Output* ([25], [82]), uma das versões mais bem sucedidas dos algoritmos iterativos, consiste na troca do passo 4 do algoritmo ER por

$$g_j^{(k+1)} = \begin{cases} h_j^{(k)}, & j \notin \gamma, \\ g_j^{(k)} - \beta h_j^{(k)}, & j \in \gamma, \end{cases}$$

onde β é uma constante pequena, ou *parâmetro de feedback*. Valores de $\beta = 0.1$ funcionam bem. Quando usamos o algoritmo HIO , $h^{(k)}$ é nada menos do que uma aproximação da estimativa de f.

Os algoritmos são iniciados tipicamente com uma condição inicial aleatória $g_j^{(0)}$ ou uma fase inicial aleatória $\phi_u^{(0)}$, exceto quando alguma outra informação adicional sobre o objeto original ou a fase original esteja disponível.

O número estipulado de iterações em nossos experimentos tem sido em média 2000 iterações para o algoritmo ER e 3000 para o HIO . Às vezes um número maior de iterações se faz necessário para se obter convergência, podendo tal número chegar até a 80 mil iterações . É claro que, em nossos experimentos, pudemos abusar do número de iterações, sem nos preocuparmos com o tempo de duração gasto para a execução dos programas, pois na maior parte dos exemplos numéricos trabalhamos com sinais e imagens pequenas.

A seguir daremos um tratamento vetorial do algoritmo ER, visto como uma projeção sobre conjuntos em um espaço de Hilbert. Inicialmente nos restringiremos ao caso unidimensional.

2.1.1 Caracterização de pares fixos para o ER: o caso 1-D

Seja $\mathcal{H} = (\mathbb{C}^n; < \cdot, \cdot >)$ o espaço de Hilbert das n-uplas de números complexos munido do produto interno canônico

$$< \mathbf{x}, \mathbf{y} >= \sum_k \mathbf{x}_k \overline{\mathbf{y}}_k = \mathbf{y}^* \mathbf{x}.$$

Assim, dado um objeto f e sua transformada de Fourier F como em (1.2) e (1.12), segue de (1.6) e (1.7) que F_u e f_j podem ser vistos em termos do produto interno $< \cdot, \cdot >$ da seguinte forma

$$F_u = \frac{1}{n} < f, \mathbf{e}_u > = \frac{1}{n} \mathbf{e}_u^* f, \quad u = 0, 1, \ldots, n-1 \tag{2.2}$$

$$f_j = < F, (\mathbf{e}_j^*)^T > = (\mathbf{e}_j)^T F, \quad j = 0, 1, \ldots, n-1 \tag{2.3}$$

onde $\mathbf{e}_u$, para cada $u = 0, 1, \ldots, n-1$, é o vetor de $\mathbb{C}^n$ cujas coordenadas são dadas por

$$\mathbf{e}_u(j) = \exp(2\pi i u j/n) \equiv \omega^{uj}, \ j = 0, 1, \ldots, n-1.$$

Note que

$$\{\mathbf{e}_u : u = 0, 1, \ldots, n-1\}$$

é uma base para $\mathcal{H}$.

Como era de se esperar, a matriz de Fourier $\mathcal{W}$ definida por (1.13) satisfaz

$$\mathcal{W}^* = [\, \mathbf{e}_0 \;\; \mathbf{e}_1 \;\; \cdots \;\; \mathbf{e}_{n-1} \,],$$

donde se conclui imediatamente que as expressões (1.10) e (1.11) são equivalentes a (2.2) e (2.3) respectivamente.

2.1.1.1 ER visto como um algoritmo de projeções

A seguir daremos uma interpretação do ER como um algoritmo de projeções [82], [5] sobre determinados subconjuntos de $\mathcal{H}$. Dado qualquer elemento

$$h = (h_0, h_1, \ldots, h_{n-1})^T \in \mathcal{H},$$

denotaremos sua transformada de Fourier pela letra maiúscula H. Sabemos que h é real se e somente se os pixels, H_u, de H satisfazem as condições de simetria dadas por (1.17). Assim, dado o vetor das magnitudes de Fourier de F, α_F, como em (1.24), definimos os seguintes subconjuntos de $\mathcal{H}$:

$$\begin{aligned}
\tau &= \{h \in \mathbb{R}^n : \quad |H_u| = \alpha_u\}, \\
\hat{\tau} &= \mathcal{F}(\tau), \\
\vartheta &= \{g = (g_0, g_1, \ldots, g_{n-1})^T \in \mathbb{R}^n : g_m = g_{m+1} = \ldots = g_{n-1} = 0\}.
\end{aligned}$$

Denotaremos o espaço tangente a τ no ponto h por $T \equiv T_h(\tau)$. Similarmente $\hat{T} \equiv T_H(\hat{\tau})$ denotará o espaço tangente a $\hat{\tau}$ no ponto H. Segue que $\hat{T} = \mathcal{F}(T)$. Como ϑ é um espaço linear, $T_g(\vartheta) \equiv \vartheta$.

Sejam as sequências de vetores $\{h^{(k)}; k = 0, 1, 2, \ldots\}$ e $\{g^{(k)}; k = 0, 1, 2, \ldots\}$ geradas pelo ER, onde $g^{(0)}$ é a condição inicial gerada aleatoriamente. Definimos as projeções

$$P_\tau : g \in \mathbb{R}^n \longmapsto h = P_\tau g \in \tau,$$

e

$$P_\vartheta : h \in \mathbb{R}^n \longmapsto g = P_\vartheta h \in \vartheta.$$

respectivamente por

$$h = P_\tau g \Longleftrightarrow H_u = |F_u| \frac{G_u}{|G_u|} \;\; \forall u \tag{2.4}$$

e

$$g = P_\vartheta h = Mh = [h_{(1)} \;\; 0]^T = (h_0, h_1, \ldots, h_{m-1}, 0, 0, \ldots, 0)^T, \tag{2.5}$$

onde

$$M = \begin{bmatrix} I & O \\ O & O \end{bmatrix}. \tag{2.6}$$

É imediata a verificação (veja [82], cap. 8) de que P_τ e P_ϑ são projeções ortogonais. Por definição os pontos $h^{(k)}$ e $g^{(k)}$, gerados pelo ER satisfazem

$$g^{(k+1)} = P_\vartheta h^{(k)}, \quad h^{(k)} = P_\tau g^{(k)} \quad \forall\, k. \tag{2.7}$$

Definição 2.1 *Dizemos que (g, h) é um par de pontos fixos para o algoritmo* ER *se $g = P_\vartheta h$ e $h = P_\tau g$.*

Note que se $g = h$ na definição acima, então g torna-se uma estimativa de f que satisfaz as restrições em ambos os domínios, o do objeto e o das frequências, e, portanto, é uma solução para o problema inverso associado.

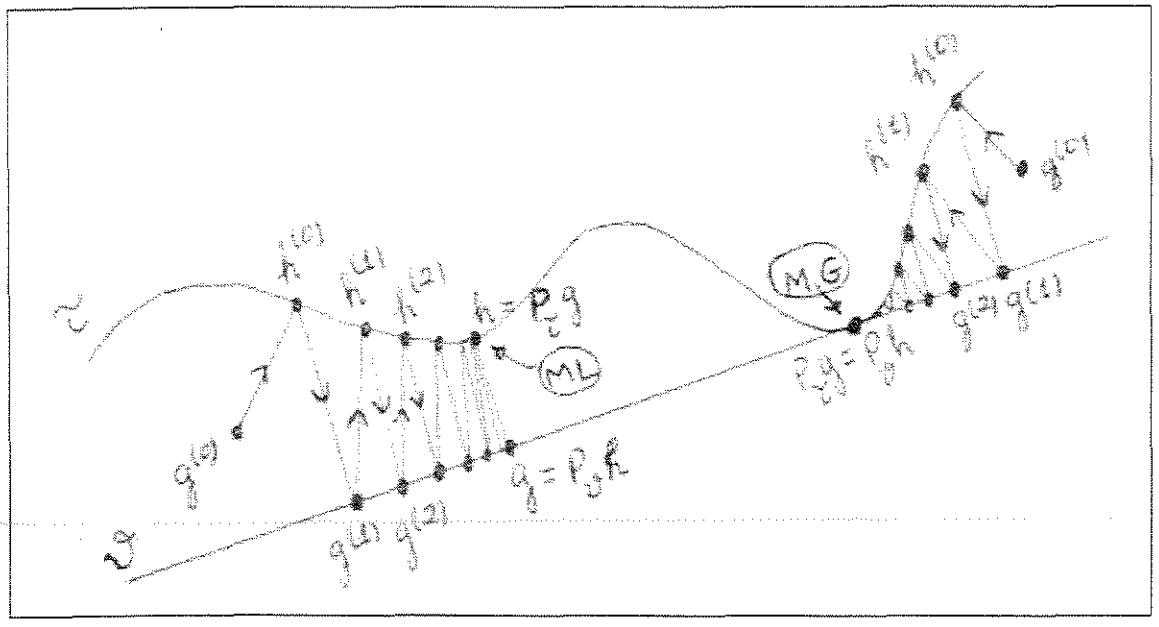

Figura 2.1: ER visto como algoritmo de projeções : (g, h) é um par de pontos fixos. À esquerda h representa um ponto de mínimo local e à direita, um mínimo global da função $d(h)$, definida para os casos 1-D e 2-D, conforme as expressões (2.21) e (2.22) respect.

A seguir daremos uma condição equivalente que caracteriza os pares de pontos fixos do algoritmo ER. Antes precisaremos do seguinte

Lema 2.1 *$\hat{T}$ é gerado pelos vetores*

$$\Delta_u = iH_u e_{u+1} - i\overline{H}_u e_{n-u+1}, \quad u = 1, 2, \ldots, m-1,$$

onde $e_u = (0, \ldots, 0, \underbrace{1}_{u^o}, 0, \ldots, 0)^T \in \mathbb{R}^n$.

Demonstração : Pela definição de τ, o espaço tangente $\hat{T}$ tem a seguinte representação

$$\begin{aligned}\hat{T} &= \mathcal{F}(T) \\ &= \{V \in \mathbb{C}^n : \ V_{m+u} = \overline{V}_{m-u}, \ |H_u + \epsilon V_u|^2 = |H_u|^2 + o(\epsilon^2) \ \ \forall u = 0, 1, 2, \ldots, m\}\end{aligned}$$

Desprezando os termos de ordem ϵ^2 obtemos imediatamente que:

$$\begin{aligned} |H_u + \epsilon V_u|^2 &= |H_u|^2 + o(\epsilon^2) \quad \Longleftrightarrow \\ H_u\overline{H}_u + \epsilon(H_u\overline{V}_u + \overline{H}_u V_u) &= H_u\overline{H}_u \quad \Longleftrightarrow \\ \mathrm{Re}(H_u\overline{V}_u) &= 0 \quad \Longleftrightarrow \\ H_u\overline{V}_u &\in i\mathbb{R} \quad \Longleftrightarrow \\ V_u &= iq_u H_u; \ \ q_u \in \mathbb{R}, \end{aligned}$$

para cada $u = 0, 1, \ldots, n-1$. Desde que V_0, V_m, H_0, H_m são reais, segue que $q_0 = q_m = 0$. Além disso mostra-se também que $q_{m+u} = -q_{m-u} \ \forall u = 1, 2, \ldots, m-1$. Consequentemente temos

$$\begin{aligned} V &= (V_0, V_1, \ldots, V_{m-1}, V_m, V_{m+1}, \ldots, V_{n-1})^T \\ &= (0, iq_1 H_1, \ldots, iq_{m-1}H_{m-1}, 0, -iq_{m-1}\overline{H}_{m-1}, \ldots, -iq_1\overline{H}_1)^T \\ &= \sum_{u=1}^{m-1} q_u \boldsymbol{\Delta}_u, \end{aligned}$$

o que prova que $B = \{\boldsymbol{\Delta}_1, \boldsymbol{\Delta}_2, \ldots \boldsymbol{\Delta}_{m-1}\}$ gera $\hat{T}$. Desde que B é um conjunto LI, segue que B é uma base de $\hat{T}$ ■

Proposição 2.1 *Sejam $h \in \tau$ e $g = P_\vartheta h = Mh$. Então (g, h) é um par de pontos fixos para o algoritmo* ER *se e somente se*

$$H^*\mathcal{W}_{(\mathbf{1})}\mathcal{W}^*_{(\mathbf{1})}\boldsymbol{\Delta}_u = 0; \ \ \forall \, u = 1, 2, \ldots, m-1, \tag{2.8}$$

onde $H = \mathcal{F}(h)$, $\boldsymbol{\Delta}_u$ é o vetor do lema (2.1) correspondente a H; e $\mathcal{W}_{(\mathbf{1})}$ é a submatriz da matriz de Fourier $\mathcal{W}$ definida como em (1.16).

Demonstração : (g, h) é um par de pontos fixos para ER se e somente se h é a projeção ortogonal de g sobre $\tau \iff G - H \perp \hat{T} \iff < \boldsymbol{\Delta}_u, G - H > = 0 \ \forall \ u \iff (G - H)^*\boldsymbol{\Delta}_u = 0 \ \forall \ u$. Por outro lado como $g = P_\vartheta h = Mh$, então segue que

$$G = \mathcal{W}g = \mathcal{W}Mh = \mathcal{W}M\frac{1}{n}\mathcal{W}^*H = \frac{1}{n}(\mathcal{W}M\mathcal{W}^*)H = \frac{1}{n}\mathcal{W}_{(\mathbf{1})}\mathcal{W}^*_{(\mathbf{1})}H.$$

Além disso, por $\boldsymbol{\Delta}_u$ pertencer ao espaço tangente $\hat{T}$, segue que $H^*\boldsymbol{\Delta}_u = 0$. Consequentemente

$$\begin{aligned}(G - H)^*\boldsymbol{\Delta}_u &= \frac{1}{n}(\mathcal{W}_{(\mathbf{1})}\mathcal{W}^*_{(\mathbf{1})}H - H)^*\boldsymbol{\Delta}_u = \frac{1}{n}H^*\mathcal{W}_{(\mathbf{1})}\mathcal{W}^*_{(\mathbf{1})}\boldsymbol{\Delta}_u - H^*\boldsymbol{\Delta}_u \\ &= \frac{1}{n}H^*\mathcal{W}_{(\mathbf{1})}\mathcal{W}^*_{(\mathbf{1})}\boldsymbol{\Delta}_u \qquad \blacksquare\end{aligned}$$

2.1.2 Caracterização de pares fixos para o ER: o caso 2-D

Resultados similares são obtidos para o caso 2-D. Sem perda de generalidades, suporemos, para efeito de simplificação na notação , que $m = m_1 = m_2$. Aqui o espaço de Hilbert a ser considerado é o espaço das matrizes complexas de ordem $n \times n$ munido do produto interno

$$< \mathbf{X}, \mathbf{Y} >= \sum_j \sum_k \mathbf{X}_{jk} \overline{\mathbf{Y}}_{jk} = \mathrm{tr}(\mathbf{Y}^* \mathbf{X}).$$

As variedades τ e ϑ são definidas similarmente ao caso 1-D levando-se em conta agora que a simetria das transformadas de Fourier de objetos reais bidimensionais devam satisfazer equações como as dadas em (1.28), i.e.,

$$\begin{aligned} \tau &= \{h \in \mathbb{M}_n(\mathbb{R}): \quad |H_{uv}| = \alpha_{uv},\ H \text{ satisfaz } (1.28)\}, \\ \hat{\tau} &= \mathcal{F}(\tau), \\ \vartheta &= \{g \in \mathbb{M}_n(\mathbb{R}): \quad g_{(12)} = g_{(21)} = g_{(22)} = 0\}. \end{aligned}$$

Definimos as projeções

$$P_\tau : g \in \mathbb{M}_n(\mathbb{R}) \longmapsto h = P_\tau g \in \tau,$$

e

$$P_\vartheta : h \in \mathbb{M}_n(\mathbb{R}) \longmapsto g = P_\vartheta h \in \vartheta.$$

respectivamente por

$$h = P_\tau g \Longleftrightarrow H_{uv} = |F_{uv}| \frac{G_{uv}}{|G_{uv}|} \tag{2.9}$$

e

$$g = P_\vartheta h = MhM = \begin{bmatrix} h_{(\mathbf{11})} & 0 \\ 0 & 0 \end{bmatrix}. \tag{2.10}$$

Os pontos $h^{(k)}$ e $g^{(k)}$, gerados pelo ER, continuam satisfazendo (2.7) e a definição de par de pontos fixos é a mesma dada pela definição 2.1, levando-se em conta agora as novas dimensões de g e h. Similarmente definimos os vetores que geram o espaço tangente $\hat{T} \equiv T_H(\hat{\tau})$:

$$\mathbf{\Delta}_{jk} = \begin{cases} iH_{jk}E_{j+1,k+1} - i\overline{H}_{jk}E_{j+1,n-k+1}; & j = 0, m;\ k = 1, 2, \ldots, m-1 \\ iH_{jk}E_{j+1,k+1} - i\overline{H}_{jk}E_{n-j+1,k+1}; & j = 1, 2, \ldots, m-1;\ k = 0, m \\ iH_{jk}E_{j+1,k+1} - i\overline{H}_{jk}E_{n-j+1,n-k+1}; & j, k = 1, 2, \ldots, m-1 \\ iH_{j,m+k}E_{j+1,m+k+1} - i\overline{H}_{j,m+k}E_{n-j+1,m-k+1}; & j, k = 1, 2, \ldots, m-1 \end{cases} \tag{2.11}$$

onde

$$\mathbf{E}_{jk} = \begin{bmatrix} 0 & \cdots & 0 & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \cdots & 1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & \cdots & 0 & \cdots & 0 \end{bmatrix}_{n \times n}, \tag{2.12}$$

com o elemento não nulo ocupando a posição (j,k). A prova de que o espaço $\hat{T}$ é gerado pelos elementos $\mathbf{\Delta}_{jk}$ é inteiramente análoga à prova do lema 2.1 e, portanto, será omitida. A seguir daremos a versão da proposição 2.1 para o caso 2-D

Proposição 2.2 *Sejam $h \in \tau$ e $g = P_\vartheta h = MhM$. Então (g,h) é um par de pontos fixos para o algoritmo* ER *se e somente se*

$$\mathrm{tr}(H^* \mathcal{W}_{(1)} \mathcal{W}_{(1)}^* \mathbf{\Delta}_{jk} \mathcal{W}_{(1)} \mathcal{W}_{(1)}^*) = 0; \quad \forall\, (j,k) \tag{2.13}$$

onde $H = \mathcal{F}(h)$, $\mathbf{\Delta}_{jk}$ é a matriz definida como em (2.11) *associada a H; e $\mathcal{W}_{(1)}$ é a submatriz da matriz de Fourier $\mathcal{W}$ definida como em* (1.16).

A demonstração da proposição 2.2 é similar à do caso 1-D e, portanto, será omitida.

2.2 ER e HIO: convergência e pontos de mínimos locais (globais)

Nesta seção faremos um breve comentário sobre o comportamento de convergência de ambos os métodos ER e HIO , ficando os detalhes e comentários dos principais resultados numéricos a serem discutidos no capítulo 4. Para o caso 2-D suporemos, sem perda de generalidades, que $m = m_1 = m_2$.

É claro que estaremos preocupados em analisar o comportamento destes métodos tanto na ausência quanto na presença de ruídos nos dados das amplitudes, pois sabemos que, na prática, os dados das amplitudes são medidos e portanto estão sempre sujeitos a interferências produzidas por, por exemplo, turbulências na atmosféra, problemas de aferição do aparelho de medição e etc. Entretanto deixaremos a análise dos resultados, relativos aos dados das amplitudes com ruídos, para os capítulos 4 e 5.

Para medir quão próximo o ponto de convergência de um dado algoritmo se encontra do objeto original, usaremos duas métricas de erro (veja [81] para maiores detalhes) que medem as distâncias em ambos os domínios, o do objeto e o das frequências. Dados dois objetos reais, f_j e g_j, com a restrição de suporte, nós definimos o erro (ou distância) no domínio de Fourier, entre $|F_u|$ e $|G_u|$, por

$$\epsilon(f,g) = \sqrt{\frac{\sum_u [\, c_{f,g}|G_u| - |F_u|\,]^2}{\|F\|^2}}, \tag{2.14}$$

onde

$$c_{f,g} = \left[\frac{\sum_u |F_u|^2}{\sum_u |G_u|^2}\right]^{1/2} = \frac{\|F\|}{\|G\|}$$

é um fator de normalização.

Métrica similar define o erro entre f_j e g_j no domínio do objeto:

$$\delta(f,g) = \sqrt{\frac{\sum_j [c_0 g_j - f_j]^2}{\|f\|^2}}, \quad \text{para } c_0 = c_{f,g}\text{sign}\left[\sum_j f_j g_j\right]. \tag{2.15}$$

As somatórias em u são tomadas sobre toda a grade onde está definida a transformada de Fourier, enquanto que as somatórias em j levam em conta apenas os valores dos objetos tomados dentro do suporte. Assim, por exemplo, para imagens reais de tamanho $n \times n$, cujo suporte seja de tamanho $m \times m$, as somatórias que definem $\epsilon(f,g)$ e $c_{f,g}$ devem ser tomadas sobre a grade $u = 0,1,\ldots,n-1$, $v = 0,1,\ldots,n-1$; enquanto as que definem $\delta(f,g)$ e c_0, sobre o suporte $j = 0,1,\ldots,m-1$, $k = 0,1,\ldots,m-1$.

Desde que g e sua gêmea $\hat{g}$ (veja definição de gêmea na seção 0.4 da Introdução) possuem o mesmo módulo de Fourier, atribuímos para $\delta(f,g)$ o menor valor obtido ao computarmos a fórmula (2.15) para ambas g e $\hat{g}$.

Como dissemos anteriormente o algoritmo ER pode ser visto como um método de projeções ortogonais [82] representadas pelo par das aplicações (P_τ, P_ϑ), definidas em (2.4) e (2.5) [caso 1-D] ou em (2.9) e (2.10) [caso 2-D]. As sequências de pontos $g^{(k)} = [g_j^{(k)}]$ e $h^{(k)} = [h_j^{(k)}]$ geradas pelo ER são obtidas por estas projeções de acordo com (2.7). Pode ser mostrado que o algoritmo ER é monótono no sentido de que o erro quadrado não normalizado

$$\begin{aligned} e_{Fk}^2 &= N^{-1}\sum_u \left[\,|G_u^{(k)}| - |F_u|\,\right]^2 \\ &= N^{-1}\sum_u \left[\,|G_u^{(k)}| - |H_u^{(k)}|\,\right]^2 \\ &= \sum_j \left|g_j^{(k)} - h_j^{(k)}\right|^2 \end{aligned} \tag{2.16}$$

não pode crescer à medida que se aumenta o número de iterações. Aqui, $N = n$ (caso 1-D) ou $N = n^2$ (caso 2-D). A prova desse fato encontra-se em [82] ou [25], e resume-se em mostrar que

$$e_{Fk}^2 \leq e_{ok}^2 \leq e_{F,k+1} \tag{2.17}$$

onde

$$e_{ok}^2 := \sum_{j\in\gamma}\left[h_j^{(k)}\right]^2 \equiv \sum_j \left|g_j^{(k+1)} - h_j^{(k)}\right|^2 \equiv \|g^{(k+1)} - h^{(k)}\|^2 \tag{2.18}$$

Note que o erro e_{ok}^2 nada mais é do que a soma dos quadrados dos valores dos pixels

de $h^{(k)}$ fora do suporte[1], i.e.,

$$\begin{aligned} e_{ok}^2 &= \|g^{(k+1)} - h^{(k)}\|^2 = \|P_\vartheta h^{(k)} - h^{(k)}\|^2 \\ &= \begin{cases} \|h_{(2)}^{(k)}\|^2 & ; \text{ se caso } 1-\text{D}, \\ \|h_{(12)}^{(k)}\|^2 + \|h_{(21)}^{(k)}\|^2 + \|h_{(22)}^{(k)}\|^2 & ; \text{ se caso } 2-\text{D}. \end{cases} \end{aligned}$$

2.2.1 Os pontos de mínimo local (global)

Nesta subseção falaremos dos mínimos locais (globais) atingidos eventualmente pelos algoritmos iterativos ER e HIO .

2.2.1.1 Caso 1-D

Considere o par de sequências $(g^{(k)}, h^{(k)})$ geradas pelo algoritmo ER . Para o caso 1-D, o erro e_{ok}^2 torna-se

$$e_{ok}^2 = \|(I - M)h^{(k)}\|^2,$$

onde M é a matriz dada por (2.6). Assim, fazendo k tender ao infinito na desigualdade (2.17), e supondo

$$\tilde{g} = \lim g^{(k)}, \quad \tilde{h} = \lim h^{(k)}, \tag{2.19}$$

teremos

$$e_{o\infty}^2 = \lim e_{Fk}^2 = \lim e_{F,k+1}^2 = \|\tilde{g} - \tilde{h}\|^2 = \|(I - M)\tilde{h}\|^2 = \|\tilde{h}_{(2)}\|^2. \tag{2.20}$$

Mostra-se que $\tilde{h}$ é um *ponto crítico* (consequência das proposições 2.1 e 3.2) para a seguinte função custo:

$$d(h) = \frac{1}{2}\|(I - M)h\|^2 = \frac{1}{2}\|h_{(2)}\|^2, \quad h \in \tau. \tag{2.21}$$

Em outros termos, $d'(\tilde{h}) = 0$, onde d' representa a derivada da função d sobre a variedade diferenciável τ [62]. Esta derivada pode ser expressa em termos da derivada de uma função definida em $\mathbb{R}^{m-1}$. De fato, conforme mostraremos na seção 3.6, h é uma função das fases de Fourier $\theta_1, \theta_2, \ldots, \theta_{m-1}$ e, portanto, se representamos $\theta = (\theta_1, \ldots, \theta_{m-1})$, mostramos que $d'(\tilde{h}) = \nabla_\theta(d \circ h)(\tilde{\theta})$ (veja seção 3.6 para mais detalhes).

De modo análogo, representamos a *matriz Hessiana* [62], $d''(h)$, de d em $h \in \tau$, em termos de θ, por

$$[d''(h)]_{jk} = \frac{\partial^2(d \circ h)(\theta)}{\partial\theta_j \partial\theta_k}, \quad j, k \in \{1, 2, \ldots, m-1\}.$$

[1] desde que $g_j^{(k)} \geq 0 \ \forall j \in \mathcal{S}, \ \forall k$, o que foi verificado numericamente em todas os experimentos com imagens iniciais $g^{(0)}$ satisfazendo a restrição de positividade (1.47).

Escreveremos

$$d''(h) > 0 \ \text{[respect. } d''(h) \geq 0\text{]}$$

para indicar que $d''(h)$ é definida positiva [respect. positiva semi-definida].

2.2.1.2 Caso 2-D:

Comentários similares valem para o caso 2-D. Por exemplo, a expressão equivalente para a função $d(h)$ é, neste caso, dada por

$$d(h) = \frac{1}{2}\|MhM - h\|^2 = \frac{1}{2}(\|h_{(12)}\|^2 + \|h_{(21)}\|^2 + \|h_{(22)}\|^2), \quad h \in \tau. \tag{2.22}$$

Como consequência das proposições 2.2 e 3.3 segue que se $\tilde{h}$ é dado como em (2.19), então $\tilde{h}$ é um ponto crítico de $d(h)$.

Segue o importante e conhecido resultado de otimização [57]:

$$\begin{array}{rcl} d'(\tilde{h}) = 0;\ d''(\tilde{h}) > 0 & \Longrightarrow & \tilde{h} \text{ é ponto de mínimo local de } d \\ \tilde{h} \text{ é ponto de mínimo local de } d & \Longrightarrow & d'(\tilde{h}) = 0;\ d''(\tilde{h}) \geq 0. \end{array}$$

Na seção 3.6 exibiremos uma expressão para a derivada $d'(h)$ e o hessiano $d''(h)$ para os dois casos uni e bidimensionais.

Definição 2.2 *Consideremos o par de sequências* $(g^{(k)}, h^{(k)})$ *geradas pelo algoritmo* ER (respect. HIO) *e suponhamos que existam os limites*

$$\tilde{g} = \lim g^{(k)}, \quad \tilde{h} = \lim h^{(k)}. \tag{2.23}$$

Diremos que o algoritmo ER (respect. HIO) *convergiu a mínimo local se* $\tilde{h}$ *for um ponto de mínimo local da função* $d(h)$ *definida por* (2.21) (caso 1-D) *ou* (2.22) (caso 2-D), *ou equivalentemente, se*

$$d'(\tilde{h}) = 0;\ d''(\tilde{h}) > 0. \tag{2.24}$$

Se além disso

$$d(\tilde{h}) = 0, \tag{2.25}$$

diremos que ER (respect. HIO) *convergiu a mínimo global*[2].

[2]Numericamente consideraremos (2.23), (2.24) e (2.25) estabelecidas quando $\|g^{(k)} - g^{(k+1)}\| < \epsilon_1$, $\|h^{(k)} - h^{(k+1)}\| < \epsilon_2$, $\|d'(\tilde{h})\| < \epsilon_3$, $d''(\tilde{h}) + \epsilon_4 I > 0$ e $d(\tilde{h}) < \epsilon_5$, para ϵ_1, ϵ_2, ϵ_3, ϵ_4 e ϵ_5 suficientemente pequenos.

2.2.2 A convergência

A convergência do ER pode ser acompanhada pelo decrescimento do erro e_{Fk} dado em (2.16). Embora o erro e_{Fk} seja não crescente com as iterações, tal fenômeno não é suficiente para garantir a aproximação para uma solução. e_{Fk} pode decrescer rapidamente nas primeiras iterações e depois muito lentamente ou até *estagnar* sem que o algoritmo tenha encontrado a solução. Constatamos numericamente, no caso 1-D, que esses pontos de estagnação são na verdade pontos de mínimo local da função custo d. Apesar de não termos verificado para o caso 2-D que os pontos de estagnação também são mínimos locais, acreditamos que tal é o caso.

Muito raramente o algoritmo ER converge a mínimo global, salvo os casos de imagens pequenas de suporte 2×2 e 4×4, onde observamos a sua convergência, na maioria dos exemplos testados, à solução verdadeira.

Para o caso 2-D e na ausência de ruídos, o HIO mostrou-se muito mais eficiente do que o ER em localizar os mínimos globais (ou as suas ambiguidades triviais - veja definição na seção 0.4 da Introdução). Embora ele também apresente algumas formas de estagnação (Fienup descreveu algumas destas estagnações e propôs métodos para sair delas: [26] e [73]) que nos impedem de sabermos se atingimos ou não a solução, foi verificado em todos os exemplos testados a sua convergência à solução verdadeira ou à gêmea correspondente.

Assim, fazendo um resumo da análise de nossos experimentos numéricos, constatamos que:

- Na ausência de ruídos nos dados das amplitudes e iniciando-se de uma condição aleatória qualquer, ER e HIO convergiram, em todos os exemplos testados, a um ponto crítico da função d. No caso 1-D, constatamos numericamente, em 100% dos exemplos testados, que esses pontos críticos são na verdade pontos de mínimos locais de d. Já no caso 2-D não fizemos nenhuma verificação numérica, porém acreditamos que a maioria dos pontos críticos atingidos por ER são também pontos de mínimos locais. Por isso acreditamos que o número de mínimos locais para a função d, no caso 2-D, seja extremamente elevado. Em particular, no caso 1-D, a convergência do ER a algum mínimo global foi verificada em 100% dos casos testados, enquanto que para o HIO (com parâmetro de feedback, β, sempre igual a 0.1) este índice caiu para 60% (Veja a tabela da seção 4.3.1). Mesmo nos casos de convergência do HIO a mínimo global, a precisão do erro foi pior do que a do ER . Também verificou-se para ambos, ER e HIO , que quanto maior o valor de m, mais iterações foram necessárias para garantir a convergência destes algoritmos a mínimo global.

- Para o caso 1-D, com ruídos, HIO tornou-se completamente instável,

nunca convergindo para ponto algum. Nesta mesma situação, ER mostrou-se mais estável, convergindo a mínimos locais.

- Para o caso 2-D sem ruídos, HIO sempre convergiu, a menos de estagnação , à solução verdadeira ou à gêmea correspondente. Já o ER não conseguiu atingir nenhum mínimo global nos casos testados, exceto em exemplos de imagens de suporte 2×2 e 4×4.

- Para o caso 2-D com ruídos, HIO e ER mantiveram as tendências do caso 1-D, ou seja, HIO instável, nunca convergindo, e ER estável, convergindo sempre a um ponto crítico.

Na prática o algoritmo HIO é um dos melhores na classe dos métodos iterativos propostos por Fienup. **Não existe nenhuma prova de sua convergência,** embora se saiba que **ele sempre converge, a menos de estagnação , a mínimo global, na ausência de ruídos** [26], [83], [84], [85], [86]. A instabilidade do HIO na presença de ruídos também não é fato novo, tendo já sido verificada por vários pesquisadores [25], [82], [83], [84], [85], [86].

2.3 O algoritmo EH

Talvez o mais eficaz dos métodos iterativos seja o método proposto por Fienup, obtido pela combinação do ER com HIO , i.e., o método que consiste de vários ciclos de iterações , onde um ciclo consiste de K_1 iteracoes do HIO seguido de K_2 iterações do ER . Designaremos esse método simplesmente pelas iniciais EH[3]. De acordo com os experimentos realizados por Fienup ([82], seção 7.4 B), para problemas bidimensionais, com imagens de tamanho razoável (por exemplo, imagens com suporte de tamanho 64×64), valores de K_1 variando de 20 a 100, de K_2 entre 5 e 10, e parâmetros de feedback, β, de 0.5 a 1.0 (por exemplo, $\beta = 0.7$) funcionam bem. Em nossos experimentos tentamos outros valores para K_1, K_2 e β (em todos os casos experimentamos $\beta = 0.1$). Também não estipulamos um número exato de ciclos em nossos métodos pois a convergência de qualquer método que envolva o HIO depende de uma série de fatores como por exemplo o tipo de condição inicial utilizada, o número de pontos de estagnação que houver durante a busca, etc. Quanto ao número de iterações para o HIO e/ou ER, escolhemos aquele que for o mais conveniente de acordo com a dimensão do problema e o nível de ruído envolvido nos dados das amplitudes. Em geral, o número de iterações que trabalhamos varia desde 50 até 80 mil.

Em muitos casos, o método EH costuma convergir a uma solução após um número pequeno de ciclos de iterações. Se existirem múltiplas soluções, o método será capaz de encontrar cada uma delas, dependendo, é claro, da condição inicial dada. Pontos

[3] Nomenclatura do autor, não usada na literatura

iniciais perto da solução verdadeira certamente reduz o número de iterações exigidas e pode evitar alguns problemas de estagnação.

Desde que o HIO, na ausência de ruídos, sempre converge, a menos de estagnação, a mínimo global, e desde que ele é altamente instável no caso contrário, o EH torna-se, portanto, o método mais indicado, nestas situações, para localizar mínimos globais, ou pontos próximos a eles, do que o HIO e o ER, quando executados isoladamente um do outro. Em experimentos onde há convergência do HIO (ou do ER) a mínimo global, o uso do EH torna-se completamente dispensável. Por esses motivos, só usaremos EH em problemas com ruídos ou em problemas sem ruídos onde houver estagnação do HIO em algum ponto.

Muitos testes com o EH, ER e HIO já foram realizados por Fienup e outros pesquisadores, por isso omitiremos aqui comentários, detalhes e resultados numéricos relativos à maior parte deles. Em nossos experimentos nós testamos tais métodos em condições específicas que serão descritas em detalhes no Cap. 4.

2.4 Outros métodos

Nesta seção faremos breves comentários sobre outros métodos que foram propostos para o problema de recuperação da fase.

2.4.1 Métodos de minimização - Regularização

Considere a forma generalizada de um problema inverso não linear mal-posto dada pela equação

$$G(x) = y, \tag{2.26}$$

onde $G : \mathcal{D}(G) \subset X \longrightarrow Y$, é um operador não linear, e X e Y são espaços de Hilbert munidos do produto interno $||\cdot||$. Em (2.26), $y = (y_1, y_2, \ldots, y_N)^T \in Y$ é supostamente conhecido e $x = (x_1, x_2, \ldots, x_M)^T \in X$ é o vetor das incógnitas, com $N > M$. Se denotarmos $G(x) = (g_1(x), \ldots, g_N(x))^T$, (2.26) poderá ser reescrita como um conjunto sobredeterminado de equações não lineares:

$$g_j(x_1, \ldots, x_M) = y_j, \quad j = 1, 2, \ldots, N. \tag{2.27}$$

2.4.1.1 Algoritmo de Levenberg-Marquardt

O *algoritmo de Levenberg-Marquardt* (LM) [51], [59] é um método de otimização baseado em quadrados mínimos não lineares para resolver (2.27), i.e.,para buscar um mínimo global da função custo

$$L(x) = \frac{1}{2}||G(x) - y||^2 = \frac{1}{2}(G(x) - y)^T(G(x) - y), \tag{2.28}$$

e é descrito pelo método iterativo

$$x^{(k+1)} = x^{(k)} - [G'(x^{(k)})^* G'(x^{(k)}) + \mu_k I]^{-1} G'(x^{(k)})^* (G(x^{(k)}) - y), \qquad (2.29)$$

onde μ_k é uma sequência de números positivos tendendo a zero. Aqui $G'(\cdot)$ denota o jacobiano.

Como se observa, o algoritmo LM é uma modificação do *método de Gauss-Newton* (GN) [27], [17]

$$x^{(k+1)} = x^{(k)} - [G'(x^{(k)})^* G'(x^{(k)})]^{-1} G'(x^{(k)})^* (G(x^{(k)}) - y).$$

O acréscimo de $\mu_k I$ ao termo $G'(x^{(k)})^* G'(x^{(k)})$ em (2.29) é para garantir que a matriz, assim obtida, além de ser uma boa aproximação da matriz Hessiana da função $L(x)$ em $x^{(k)}$, seja também definida positiva. Existem vários procedimentos numéricos para encontrar um valor ótimo para a constante μ_k. Um deles, baseado no método de busca direta, é bastante eficiente e foi desenvolvido por Moré em seu paper [64].

O método LM pode então ser aplicado ao problema da fase se quisermos resolver o sistema (1.75) posto na forma (2.27). As componentes do vetor x serão os pixels f_{jk} da imagem f, e os termos do segundo membro de (2.27) serão as transformadas inversas $\mathcal{F}^{-1}(|F_{uv}|^2)$. Este é o procedimento adotado por Nieto-Vesperinas em [67]. É claro que na representação de (1.75) na forma (2.27), é levado em conta a restrição de suporte reduzido. Como a maioria dos outros métodos de reconstrução, o método LM usado por Nieto-Vesperinas para resolver o problema da fase também mostra-se ineficiente para imagens grandes. O número de pontos de mínimo local cresce dramaticamente com o tamanho das imagens, fazendo com que o método seja impraticável para imagens de suporte maior que 6×6.

2.4.1.2 Método de Gauss-Newton regularizado iterativamente

Desde que na prática nós temos disponível apenas uma aproximação, y_δ, para o vetor y, com

$$||y_\delta - y|| \leq \delta,$$

é necessário, então, que tratemos o problema (2.26) na forma regularizada. *Regularização de Tikhonov* é certamente o método de regularização mais conhecido na literatura, indicado para resolver problemas inversos mal-postos. Groetsch [34] fez uma análise deste método aplicado a problemas lineares. Já Seidman e Vogel [80], Engl *et al.* [22], Neubauer [66], e Scherzer *et al* [79] analisaram o método para problemas não lineares.

Em [36], Hanke *et al.* discutiu a convergência do método

$$x^{(k+1)} = x^{(k)} - G'(x^{(k)})^* (G(x^{(k)}) - y_\delta),$$

conhecido como *método de iteração de Landweber* (IL). Note que o método IL é obtido do método GN, quando trocamos o termo $[G'(x^{(k)})^*G'(x^{(k)})]^{-1}$ simplesmente pelo operador identidade.

O *método de Gauss-Newton regularizado iterativamente* (GNRI), originalmente proposto por Bakushinskii [2], é

$$\begin{aligned} x^{(k+1)} &= x^{(k)} - (G'(x^{(k)})^*G'(x^{(k)}) + \mu_k I)^{-1}(G'(x^{(k)})^*(G(x^{(k)}) - y_\delta) \\ &\quad + \mu_k(x^{(k)} - \zeta), \end{aligned} \tag{2.30}$$

onde, novamente, μ_k é uma sequência de números positivos tendendo a zero. Usualmente considera-se $\zeta = x^{(0)}$, mas isto não é necessário. Note que o método GNRI é obtido a partir do método LM. De fato obtém-se (2.30) acrescentando-se em (2.29) o termo $\mu_k(x^{(k)} - \zeta)$.

Blaschke *et al* em [6] demonstram que o método (2.30), para a situação especial $\zeta = x^{(0)}$, é localmente convergente desde que o operador G satisfaça uma determinada condição de suavidade. Para dados perturbados, eles propõem *critérios de parada* que garantam a convergência das iterações, desde que o nível de ruído tenda a zero. Também em [18], Deuflhard discute outras condições que garantem a convergência do método (2.30), para o caso $\zeta \neq x^{(0)}$.

2.4.1.3 ER visto como método de descida rápida

Fienup mostra em [25] que o algoritmo Error-Reduction é similar ao método de *descida rápida* [57]. Primeiramente ele considera a função erro quadrado definida, como em (2.16), a menos do fator 1/2, por[4] :

$$B \quad =^5 \quad \frac{1}{2N}\sum_u [\,|G_u| - |F_u|\,]^2, \tag{2.31}$$

como sendo a função custo a ser minimizada. Aqui, os N valores de g (a estimativa de f) são tratados como N parâmetros independentes. O que se faz então é minimizar o erro, B, como uma função dos N parâmetros, g_j, sujeita às restrições no domínio do objeto. Em resumo, o método consiste nos seguintes três passos:

Passo 1. Em um dado ponto do $k-$ésimo passo, $g^{(k)}$, computa-se as derivadas parciais da função B com relação a cada uma das componentes, g_j, de g de modo a formar o gradiente de B, $\nabla_g B$. Fienup mostra que as componentes de $\nabla_g B$ satisfazem

$$(\nabla_g B)_j := \frac{\partial B}{\partial g_j} = \frac{1}{2}[g_j - h_j], \tag{2.32}$$

[4]Originalmente, Fienup não leva em conta o fator 1/2 na definição de B, entretanto, com o uso de tal constante, podemos concluir a expressão (2.33) sem fazer uso do artifício do *passo de duplo comprimento* para obter o minimizador de B na direção do gradiente.

[5]Lembremos que se estivermos lidando com o caso 1-D, $N = n$ e a coordenada u no domínio de frequências é unidimensional. No caso 2-D, u é coordenada bidimensional, e nossa restrição no espaço do objeto é o das imagens quadradas $n \times n$ de suporte $m \times m$, portanto $N = n^2$.

onde h_j é tal que

$$H_u = |F_u| \frac{G_u}{|G_u|}.$$

Observa-se que, aqui, o cálculo de h é idêntico aos três primeiros passos do algoritmo ER (veja (2.1)).

Passo 2. Move-se, então, a partir de $g^{(k)}$ na direção oposta à do gradiente a um novo ponto $\tilde{g}^{(k)}$ que reduz o erro B. Fienup mostra que o ponto $\tilde{g}^{(k)}$ satisfaz

$$\tilde{g}^{(k)} - g^{(k)} = h^{(k)} - g^{(k)} \tag{2.33}$$

ou seja,

$$\tilde{g}^{(k)} = h^{(k)}.$$

De fato, desde que $|H_u^{(k)}| = |F_u|$, mover em direção a $h^{(k)}$ reduz o erro (Eq.(2.31)) para exatamente zero.

Passo 3. Como passo final do método, obtém-se uma nova estimativa $g^{(k+1)}$ para f, a partir de $\tilde{g}^{(k)}$, exigindo-se que as retrições no domínio do objeto sejam cumpridas, o qual é, precisamente, o quarto passo do algoritmo ER em (2.1). Isto será feito iterativamente até que um mínimo (espera-se que seja global) seja atingido.

Fienup mostra ainda que o ER pode ser visto como um caso especial de uma classe mais geral de métodos de gradiente. De fato, combinando as equações (2.32) e (2.33), conclui-se imediatamente que

$$\tilde{g}^{(k)} = g^{(k)} - 2\nabla_{g^{(k)}} B. \tag{2.34}$$

Agora, se trocamos o fator 2 em (2.34) por um parâmetro genérico (que mede, em outras palavras, o comprimento do passo na busca do minimizador $\tilde{g}^{(k)}$), μ_k, o método assim obtido é o *método de gradiente* na sua forma generalizada, i.e.,

$$\tilde{g}^{(k)} = g^{(k)} - \mu_k \nabla_{g^{(k)}} B. \tag{2.35}$$

Teríamos, então, associado ao método (2.35), um algoritmo iterativo, o qual poderia ser denominado *Error-Reduction Generalizado* e que consistiria dos três primeiros passos do ER (veja (2.1)) mais o quarto passo dado pela expressão

$$g_j^{(k+1)} = \begin{cases} \tilde{g}_j^{(k)}, & j \notin \gamma, \\ 0, & j \in \gamma. \end{cases}$$

Um método de gradiente, superior ao de rápida descida, é o *método gradiente conjugado*, o qual é dado por

$$\tilde{g}^{(k)} = g^{(k)} + \mu_k D_k, \tag{2.36}$$

onde

$$D_k := h^{(k)} - g^{(k)} + (B_k / B_{k-1}) D_{k-1},$$

onde a primeira iteração começa com $D_1 = h^{(1)} - g^{(1)}$. Aqui, $B_k = B(g^{(k)})$.

Para maiores detalhes sobre a performance desses métodos, consulte [25]

Finalizamos esta subseção com o método proposto por Lane [49], também baseado em minimização por gradientes conjugados. Lane propõe a combinação das restrições nos domínios do objeto e das frequências em uma única função custo. Mais precisamente, Lane propõe a seguinte métrica de erro.

$$E_c := E_i + E_f,$$

onde E_i e E_f são versões contínuas de métricas discretas similares respectivamente às métricas δ e ϵ, Eqs. (2.15) e (2.14). As restrições no domínio do objeto são incorporadas em E_i, enquanto as no domínio de frequências são incorporadas em E_f.

2.4.2 Projeções sobre conjuntos convexos (POCS)

O método de projeções sobre conjuntos convexos (veja capítulos 2 e 8 de [82] para maiores detalhes) consiste de uma família de algoritmos recursivos para achar um ponto na interseção de m conjuntos convexos dados. Este método foi aperfeiçoado quatro décadas atrás pelos russos Bregman [9] e Gubin *et al.* [35] e foi extendido e adaptado a processamento de sinais por Youla [93], [94]. Já na década de oitenta deu-se um importante passo nesta área quando o algoritmo ER foi identificado com um algoritmo de projeção não convexa, conforme descrevemos na seção 2.1.1.

Apesar de já ser um método clássico, o POCS ainda é discutido em artigos atuais. Em recente publicação, Bauschke *et al.* [5] estabelece novas conexões entre este método e os algoritmos iterativos de Fienup. Ele mostra que o algoritmo básico input-output ([25], como dissemos no início deste capítulo, não comentaremos este método neste trabalho) corresponde ao algoritmo de Dykstra [20], [8] e que o HIO pode ser visto como o algoritmo de Douglas-Rachford [21].

2.4.3 Zero Crossing 2-D: o algoritmo de Izraelevitz-Lim

Izraelevitz e Lim [43] sugerem um algoritmo que pretende resolver o problema da fase 2-D usando fatoração 1-D. Partindo da relação

$$R(w,z) = F(w,z)\hat{F}(w,z), \quad |w|,|z| \ = \ 1,$$

entre as transformadas-z de $r(j,k) := \mathcal{F}^{-1}\left[|F(u,v)|^2\right]$ (lado esquerdo) e as de $f(j,k)$ e sua gêmea $\hat{f}(j,k) := f(-j,-k)$ (lado direito), e cancelando eventuais potências de w e z se necessário, obtém-se uma relação envolvendo 3 polinômios, do tipo

$$p(w,z) = q(w,z)\hat{q}(w,z), \tag{2.37}$$

quando $\hat{q}(w,z) = q(w^{-1},z^{-1})w^{\alpha}z^{\beta}$, para alguns inteiros α e β. Para z fixo, podemos calcular os zeros $w_i(z)$ do lado direito desta equação, já que p é conhecido.

Mudando o parâmetro $z = z(t)$, podemos traçar esses zeros, $w(t) = w_i(z(t))$, usando o PVI

$$\frac{dw(t)}{dt} = -\frac{\frac{\partial p}{\partial z}(w(t), z(t))}{\frac{\partial p}{\partial w}(w(t), z(t))}\frac{dz(t)}{dt}, \quad w(0) = w_0.$$

Assim encontramos zeros do lado direito de (2.37) do tipo

$$z_i, w_{i_k}, \quad i = 1, \ldots, M,\ k = 1, \ldots, N,$$

que nos permita calcular $q(w, z)$ unicamente, a menos das ambiguidades triviais.

Uma séria limitação desse método está relacionada ao cálculo de zeros de polinômios de grau muito grande (para imagens com suporte de tamanho 25 por 25, por exemplo, o polinômio $p(w, z)$ é de grau 48 em cada variável), o que torna o método computacionalmente caro e numericamente instável.

2.4.4 Métodos que utilizam informações adicionais do objeto

O método proposto por Yagle e Bell em [92] resolve o problema da fase usando informação adicional sobre o objeto. Especificamente, presume-se que o objeto é do tipo *minimum-phase* ou *subminimum-phase* (sua transformada-z tem todos os polos e zeros interiores a um círculo de raio limitado - no caso de minimum-phase exige-se que o raio seja unitário). Alternativamente, presume-se que as fases de algumas componentes de F são conhecidas. O problema é, então , reduzido a um sistema linear do tipo $Ax = b$ com A, matriz simétrica e de Toeplitz. O artigo discute o caso 1-D com algumas extensões para o caso 2-D. Por se tratar de um método direto de resolver o problema, sua aplicação é, de novo, limitada a problemas de tamanho pequeno.

Métodos que utilizam informações adicionais do objeto foram propostos por outros autores. Em [90], por exemplo, Hayes discute a reconstrução de objetos a partir das *magnitudes de Fourier com sinal*, as quais serão oportunamente definidas no capítulo 4. Neste mesmo capítulo apresentaremos resultados numéricos de nosso método (veja cap. 3) e dos algoritmos iterativos de Fienup, em casos específicos onde tais informações são utilizadas.

Também em [87], Taratorin e Sideman abordam o problema de reconstrução a partir de informações das fases e amplitudes de Fourier, ambas corrompidas por ruídos. No capítulo 4 nós também fazemos algo semelhante ao usarmos as fases de Fourier com ruídos como condição inicial para os métodos que escolhemos em nossas simulações .

Capítulo 3

Um novo algoritmo para a recuperacão da fase a partir das magnitudes de Fourier

O problema da fase pode ser abordado como problema de otimização, usando uma função de custo que penaliza os valores de f fora do suporte. Como variáveis podemos considerar simplesmente as pixels do objeto. No caso 2-D com imagem de tamanho $2m_1 \times 2m_2$ teremos então $4m_1m_2$ variáveis, e para valores típicos de $m_1, m_2 > 100$ teremos mais que 10^4 variáveis, mostrando a alta complexidade do problema.

Alternativamente, poderíamos considerar os valores da DFT como variáveis; já que as amplitudes são dadas, só teríamos que considerar as fases. Por outro lado, o termo de penalização seria mais complicado.

Como foi observado, o algoritmo ER pode ser visto como algoritmo de projeção não convexa; ademais, uma função de custo $L \geq 0$, quadrada nos pixels do objeto, que penaliza os valores fora do suporte, decresce em cada passo do algoritmo. Porém, o ER usa um conjunto redundante de variáveis, ou seja, em cada iteração ele atualiza tanto os pixels do objeto, como as fases e amplitudes da DFT. O algoritmo HIO também usa o mesmo conjunto redundante de variáveis, e sua interpretação como método de descida é apenas hipotética. No entanto, para um problema consistente ele se distingue por resolver a minimização global.

Nesse capítulo queremos considerar apenas as fases como variáveis, fixando as amplitudes nos valores desejados e usando a mesma função de custo L. Em termos das fases, esta função não é mais quadrática, nem convexa. De fato, em ambos 1-D e 2-D ela tipicamente apresenta um grande número de mínimos locais.

Para testar numericamente essa abordagem, em princípio qualquer método de gradientes pode ser aplicado; porém, pelo tamanho do problema será necessário evitar o cálculo do Hessiano, e até o cálculo do gradiente pode ser demorado. Nesse

trabalho adotamos o método de BFGS, que é um método tipo Newton que em cada passo atualiza o Hessiano acrescentando a ele uma matriz simétrica de posto 2. Mais especificamente, usamos o pacote L.BFGS.B [95], escrito em Fortran, desenvolvido por pesquisadores do ANLOTC (*Argonne National Laboratory Optimization Technology Center*). Para calcular a FFT (transformada rápida de Fourier discreta), faremos uso de uma subrotina escrita em linguagem C, cuja performance tem-se mostrado tipicamente superior à de outros softwares de domínio público disponíveis. Esta subrotina faz parte do pacote FFTW (*Fastest Fourier Transform in the West*) e sua versão 2.1.3 pode ser baixada da página www.fftw.org.

Do ponto de vista da teoria de otimização , nossa abordagem não passa de aplicação de métodos bem conhecidos; porém, em termos do problema da fase, penalização do objeto fora do suporte em termos das fases, fixando as amplitudes, é uma nova abordagem. Nosso algoritmo é definitivamente superior ao ER em termos de convergência, apesar de ambos compartilharem a mesma função custo; e em algumas situações nosso método se mostra mais robusto a ruídos que o HIO. No entanto, ele é bem mais lento, sugerindo a necessidade de encontrar alternativas implementações numéricas, por exemplo, simplificando o cálculo do gradiente.

Na próxima seção daremos a formulação do problema para ambos os casos 1-D e 2-D, separadamente; na seção 3.3 faremos a descrição do método numérico utilizado. Os resultados numéricos serão apresentados no capítulo 4.

3.1 Formulação do problema

Nós estamos interessados em reconstruir objetos dados na forma

$$f = [f_0, f_1, \ldots, f_{m-1}, 0, 0, \ldots, 0]^T \in \mathbb{R}^n \tag{3.1}$$

para o caso unidimensional, e

$$f = \begin{bmatrix} f_{00} & f_{01} & \cdots & f_{0,m-1} & 0 & 0 & \cdots & 0 \\ f_{10} & f_{11} & \cdots & f_{1,m-1} & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ f_{m-1,0} & f_{m-1,1} & \cdots & f_{m-1,m-1} & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & 0 & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{M}_n(\mathbb{R}), \tag{3.2}$$

para o caso bidimensional, onde como antes $n = 2m$. Note que no caso 2-D estamos nos restringindo a imagens quadradas para simplificação da notação , porém todos os resultados permanecem válidos para matrizes retangulares.

Recuperar o objeto f equivale, como já observamos antes, a reconstruir as fases de F a partir de suas amplitudes. Nosso método inicia-se ao considerarmos F como uma função de suas fases.

3.1.1 O caso 1-D

Considere os vetores das fases e das amplitudes de F, Eqs. (1.25) e (1.24),

$$\phi_F = (\phi_0, \phi_1, \ldots, \phi_m)^T, \quad \alpha_F = (\alpha_0, \alpha_1, \ldots, \alpha_m)^T,$$

onde F é a DFT do sinal f em (3.1) que queremos recuperar. Denotaremos as fases e as amplitudes nos *cantos,* definidos pelas posições 1 e $m+1$, pelos vetores

$$\phi_c := (\phi_0, \phi_m)^T, \quad \alpha_c := (\alpha_0, \alpha_m)^T.$$

As demais fases e amplitudes, que se encontram fora desses cantos, serão representadas pelos vetores

$$\phi := (\phi_1, \phi_2, \ldots, \phi_{m-1})^T \text{ e } \alpha := (\alpha_1, \alpha_2, \ldots, \alpha_{m-1})^T.$$

Assim

$$\phi_F = (\phi_0; \phi; \phi_m)^T, \quad \alpha_F = (\alpha_0; \alpha; \alpha_m). \tag{3.3}$$

Chamaremos as componentes do vetor ϕ de *fases intermediárias de* F.

A expressão (1.19) nos diz que o vector F pode ser visto como uma função complexa nos parâmetros α_c, α, ϕ_c, e ϕ. Entre estes parâmetros, nós só temos informação a respeito de α_c e α. Entretanto como $\phi_0, \phi_m \in \{0, \pi\}$, podemos admitir que ϕ_c é também conhecido se pré fixarmos um valor 0 ou π para cada uma de suas componentes. Assim podemos olhar para F como uma função que depende apenas do parâmetro ϕ, i.e., $F = F(\phi)$, para a função $F : \mathbb{R}^{m-1} \longrightarrow \mathbb{C}^n$,

$$F(\theta) = (\alpha_0 e^{i\phi_0}, \alpha_1 e^{i\theta_1}, \ldots, \alpha_{m-1} e^{i\theta_{m-1}}, \alpha_m e^{i\phi_m}, \alpha_{m-1} e^{-i\theta_{m-1}}, \ldots, \alpha_1 e^{-i\theta_1})^T. \tag{3.4}$$

Quando quizermos enfatizar as amplitudes e as fases nos cantos, escreveremos

$$F(\phi) = F(\alpha_c, \phi_c, \alpha, \phi). \tag{3.5}$$

Por causa da simetria de F, segue da proposição (1.17) que a transformada inversa de $F(\theta)$,

$$f(\theta) \equiv \left[\begin{array}{cc} f_{(1)}(\theta) & f_{(2)}(\theta) \end{array} \right]^T = \mathcal{F}^{-1}[F(\theta)], \tag{3.6}$$

é uma função real de θ.

Assim, seja f o objeto original como em (3.1), e assumindo que as amplitudes α_c, α, e as fases, ϕ_c, de sua DFT sejam conhecidas, estaremos buscando um vetor

$$\tilde{\phi} = (\tilde{\phi}_1, \tilde{\phi}_2, \ldots, \tilde{\phi}_{m-1})^T \in (-\pi, \pi]^{m-1} \tag{3.7}$$

tal que

$$f_{(\mathbf{2})}(\tilde{\phi}) = 0. \tag{3.8}$$

Apesar de as fases pertencerem ao intervalo $(-\pi, \pi]$, não há problema em trabalharmos com f definida sobre $\mathbb{R}^{m-1}$, pois $e^{i\tilde{\phi}_j} = e^{i(\tilde{\phi}_j + 2k\pi)}$.

Seja $\tilde{f} = f(\tilde{\phi})$. Nosso próximo passo é verificar se $\tilde{f} = f$. Por causa da não unicidade para o caso 1-D (discutida na seção 1.3), pode ocorrer que a $\tilde{f}$ encontrada, satisfazendo (3.8), seja uma ambígua não trivial do sinal original f. Na execução do nosso método e dos algoritmos iterativos de Fienup, verificou-se que quanto maior o valor de m, menor a probabilidade de $\tilde{f}$ ser uma ambígua trivial de f. Note também que (3.3) nos diz, em outras palavras, que o vetor ϕ_F pertence, de acordo com a definição 1.3 do capítulo 1, ao toróide $T_{(\phi_0,\phi_m)}$. Assim, mesmo que não tenhamos qualquer informação a respeito de ϕ_c, devemos resolver o problema (3.8) separadamente sobre cada um dos 4 toróides $T_{(\phi_0,\phi_m)}$; $\phi_0, \phi_m \in \{0, \pi\}$. Sobre pelo menos um desses 4 toróides espera-se então encontrar um ponto $(\phi_0; \tilde{\phi}; \phi_m)$ tal que $\tilde{\phi}$ seja solucão de (3.8). Se o objeto original f em (3.1) é assumidamente não negativo, o problema (3.8) deve então ser resolvido apenas sobre os toróides $T_{(0,0)}$ e $T_{(0,\pi)}$, já que $\epsilon_0 = +1$.

Assim, dados $(\alpha_0, \alpha_1, \ldots, \alpha_m) \in \mathbb{R}_+^{m+1}$, $\phi_0, \phi_m \in \{0, \pi\}$, formemos $F(\theta)$ como em (3.4) e consideremos $f_{(\mathbf{2})}(\theta)$ como em (3.6). A fim de encontrarmos uma solução (3.7) que satisfaça (3.8), nós consideramos o seguinte problema de otimização

$$\tilde{\theta} = \text{argmin}_\theta \frac{1}{2}\|f_{(\mathbf{2})}(\theta)\|^2. \tag{3.9}$$

De acordo com o teorema de Parseval [11], a norma de $f(\theta)$ é independente de θ, pois

$$\begin{aligned} \|f(\theta)\|^2 &= \frac{1}{n}\|F(\theta)\|^2 \\ &= \frac{1}{n}[\alpha_0^2 + \alpha_m^2 + 2\sum_{k=1}^{m-1}\alpha_k^2] \\ &= K_\alpha = \text{const.} \end{aligned}$$

Assim o problema (3.9) torna-se equivalente a:

$$\tilde{\theta} = \text{argmin}_{\theta\in[-\pi,\pi)^{m-1}} L(\theta); \quad L(\theta) := \frac{1}{2}K_\alpha - \frac{1}{2}\|f_{(\mathbf{1})}(\theta)\|^2. \tag{3.10}$$

Observemos que a função custo em (3.10) pode ser reescrita como

$$L = \frac{1}{2}K_\alpha - \frac{1}{2}(Pf)^T(Pf), \tag{3.11}$$

onde P é a projeção do vetor f sobre o primeiro subvetor $f_{(\mathbf{1})}$, i.e.,

$$P = \begin{bmatrix} I & O \end{bmatrix} \in \mathbb{M}_{m\times n}(\mathbb{R}). \tag{3.12}$$

I é a matriz identidade $m \times m$.

3.1.2 O caso 2-D

Seja F a DFT da imagem f, em (3.2), que queremos recuperar. Segue que F é dada pela expressão (1.29), adaptada ao caso de imagens quadradas. Assim, similarmente ao caso 1-D, após fazermos as substituições $m_1 = m_2 = m$ e $n_1 = n_2 = n$ em (1.29), a matriz F, assim obtida, poderá ser vista como uma função de $\mathbb{R}^M$, $M = 2m^2 - 2$, em $\mathbb{M}_n(\mathbb{C})$, nos parâmetros α_c, α, ϕ_c, e ϕ, ou seja,

$$F \equiv F(\alpha_c, \phi_c, \alpha, \phi) \equiv F(\phi),$$

onde, desta vez,

$$\alpha_c = (\alpha_{00}, \alpha_{0m}, \alpha_{m0}, \alpha_{mm})^T, \quad \phi_c = (\phi_{00}, \phi_{0m}, \phi_{m0}, \phi_{mm})^T,$$

$$\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_M)^T, \quad \phi = (\phi_1, \phi_2, \ldots, \phi_M)^T.$$

As componentes de α_c e α formam o conjunto de todas as amplitudes de F que aparecem na matriz (1.29), bem como os vetores ϕ_c e ϕ representam todas as fases. Se conhecemos os valores de α_c, α e ϕ_c então F dependerá somente de ϕ.

Notemos que $\phi_1, \phi_2, \ldots, \phi_M$ são as fases ϕ_{jk} de F associadas aos pixels de posição

$$(j,k) \in \tilde{\Lambda} := \Lambda \setminus \{(0,0), (0,m), (m,0), (m,m)\}. \tag{3.13}$$

onde o conjunto Λ é definido como em (1.31) e (1.32), com as dimensões devidamente adaptadas à matrizes quadradas. Em outras palavras as componentes do vetor ϕ são as fases dos elementos de F interiores à linha poligonal fechada que aparece em (1.29). Podemos ainda, equivalentemente, tratar ϕ como sendo o vetor obtido de ϕ_F (veja (1.34)) após removermos as *fases dos cantos* $(0,0)$, $(0,m)$, $(m,0)$ e (m,m), de suas respectivas posições ; ou seja, adotando as notações introduzidas em (1.37), teremos

$$\phi = (\phi_{(1)}, \phi_{(2)}, \phi_{(3)})^T \in \mathbb{R}^M. \tag{3.14}$$

Como no caso 1-D, nós também chamaremos as componentes de ϕ de *fases intermediárias* de F.

Comentários análogos podem ser feitos para o vetor α.

Seja f um objeto como em (3.2) e suponha conhecidas as amplitudes e as fases dos cantos de sua DFT, F, digamos, $\alpha \in \mathbb{R}_+^M$, $\alpha_c \in \mathbb{R}_+^4$ e $\phi_c \in \{0, \pi\}^4$. Formemos a matriz $F(\theta)$, como em (1.29), que tem suas amplitudes dadas por α e α_c; as fases dos cantos dadas por ϕ_c e as fases intermediárias dadas por um vetor genérico qualquer $\theta \in \mathbb{R}^M$. Tomemos

$$f(\theta) = \mathcal{F}^{-1}[F(\theta)].$$

Desejamos encontrar um vetor

$$\tilde{\phi} = (\tilde{\phi}_1, \tilde{\phi}_2, \ldots, \tilde{\phi}_M)^T \in (-\pi, \pi]^M \tag{3.15}$$

de modo que

$$f_{(12)}(\tilde{\phi}) = f_{(21)}(\tilde{\phi}) = f_{(22)}(\tilde{\phi}) = 0, \tag{3.16}$$

onde $f_{(12)}(\tilde{\phi})$, $f_{(21)}(\tilde{\phi})$ e $f_{(22)}(\tilde{\phi})$ são subblocos da matriz $f(\tilde{\phi})$. No caso de encontrarmos tal vetor, o objeto

$$\tilde{f} := f(\tilde{\phi})$$

é, a menos de um conjunto de medida zero, ou o próprio objeto original f, ou uma ambiguidade trivial dele (veja seção 0.4).

Como no caso unidimensional esperamos encontrar $\tilde{\phi}$ através do problema de minimização

$$\tilde{\theta} = \text{argmin}_\theta \frac{1}{2}(\|f_{(12)}(\theta)\|^2 + \|f_{(21)}(\theta)\|^2 + \|f_{(22)}(\theta)\|^2)$$

ou, equivalentemente, após aplicar o teorema de Parseval,

$$\tilde{\theta} = \text{argmin}_\theta L(\theta)$$

quando

$$\begin{aligned} L(\theta) &:= \frac{1}{2n^2}\|F\|^2 - \frac{1}{2}\|f_{(11)}(\theta)\|^2 \\ &= \frac{1}{2}K_\alpha - \frac{1}{2}\text{tr}(f_{(11)}^T f_{(11)}) \\ &= \frac{1}{2}K_\alpha - \frac{1}{2}\text{tr}\{(PfP^T)^T(PfP^T)\}, \end{aligned} \tag{3.17}$$

onde

$$K_\alpha := \frac{1}{n^2}\|F\|^2 \equiv \frac{1}{n^2}(\|\alpha_c\|^2 + 2\|\alpha\|^2) = \text{const},$$

e P é a matriz de projeção dada em (3.12).

3.2 O cálculo do gradiente

Nesta seção calculamos o gradiente da função custo $L(\theta)$.

3.2.1 O caso 1-D

Se tomarmos a derivada com respeito a θ_j, $j = 1, 2, \ldots, m-1$, em ambos os lados de (3.11), o gradiente

$$\mathbf{g}(\theta) = (g_1(\theta), g_2(\theta), \ldots, g_{m-1}(\theta))$$

da função custo tem então a seguinte representação :

$$g_j \equiv \frac{\partial L}{\partial \theta_j} = -(Pf)^T (P\frac{\partial f}{\partial \theta_j}), \tag{3.18}$$

onde

$$\frac{\partial f}{\partial \theta_j} = \left[\frac{1}{n}\mathcal{W}^* \frac{\partial F}{\partial \theta_j}\right] \equiv \mathcal{F}^{-1}[\frac{\partial F}{\partial \theta_j}], \tag{3.19}$$

e

$$\frac{\partial F}{\partial \theta_j} = [0, \ldots, 0, i\alpha_j e^{i\theta_j}, 0, \ldots, 0, -i\alpha_j e^{-i\theta_j}, 0, \ldots, 0]^T. \tag{3.20}$$

Os elementos não nulos de (3.20) ocupam as posições j e $n-j$ respectivamente. Notemos que o vetor em (3.20) é o próprio vetor $\mathbf{\Delta}_j$ do lema 2.1.

O vetor $\mathbf{g}$ também pode ser dado na forma compacta

$$\mathbf{g} \equiv \nabla_\theta L = -(Pf)^T (P\nabla_\theta f) \tag{3.21}$$

onde $\nabla_\theta f$ é uma matriz $n \times (m-1)$ dada por

$$\nabla_\theta f = \left[\frac{\partial f}{\partial \theta_1}, \ldots, \frac{\partial f}{\partial \theta_{m-1}}\right].$$

3.2.2 O caso 2-D

O gradiente, $\mathbf{g} = (g_1, g_2, \ldots, g_M)^T$, da função custo $L(\theta)$ em (3.17) é:

$$g_J \equiv \frac{\partial L}{\partial \theta_J} = -\mathrm{tr}\left\{(PfP^T)^T (P\frac{\partial f}{\partial \theta_J}P^T)\right\}, \quad J = 1, 2, \ldots, M, \tag{3.22}$$

onde

$$\frac{\partial f}{\partial \theta_J} = \mathcal{F}^{-1}\left[\frac{\partial F}{\partial \theta_J}\right] = \left[\frac{1}{n^2}\mathcal{W}^* \frac{\partial F}{\partial \theta_J}\mathcal{W}^*\right]. \tag{3.23}$$

e $\partial F/\partial \theta_J$ é, para cada $J \in \{1, 2, \ldots, M\}$, matriz $n \times n$ dada por

$$\frac{\partial F}{\partial \theta_J} = i\alpha_{jk} e^{i\theta_{jk}} \mathbf{E}_{jk} - i\alpha_{jk} e^{-i\theta_{jk}} \mathbf{E}_{n-j,n-k}, \quad (j,k) \in \tilde{\Lambda}, \tag{3.24}$$

onde $\mathbf{E}_{jk}$ é a matriz de ordem $n \times n$ que tem 1 na posição (j,k) e 0 nas demais, para $j, k = 0, 1, \ldots, n-1$ (veja (2.12)). Novamente observa-se aqui que a matriz em (3.24) é a matriz $\mathbf{\Delta}_{jk}$ de (2.11).

3.2.2.1 Otimização do cálculo do gradiente no caso 2-D

Como veremos nesta subseção , o cálculo do gradiente realizado no domínio do objeto (Eq. (3.22)) é muito mais caro computacionalmente do que no domínio de Fourier. Para que possamos verificar esta conclusão , precisamos primeiramente obter uma expressão equivalente a (3.22) no domínio de Fourier.

Para obtermos uma nova fórmula do gradiente, exploraremos a estrutura de esparsidade da matriz $\partial F/\partial\theta_J$ em (3.24). Mas antes introduziremos novas notações e um resultado elementar que nos serão úteis em nossos cálculos.

Consideremos a matriz de Fourier, $\mathcal{W}$, como em (1.13), dada pelo termo geral

$$\mathcal{W}_{jk} = \omega^{-jk}, \quad j,k \in \{0,1,2,\ldots,n-1\}, \quad \omega = \exp(\pi i/m)$$

e seja $P = \begin{bmatrix} I & O \end{bmatrix}$ o operador projeção sobre $\mathbb{R}^m$. Definimos a seguinte matriz de ordem $(n+1)\times(n+1)$

$$\mathcal{B} := (P\mathcal{W}^*)^*(P\mathcal{W}^*). \tag{3.25}$$

Lema 3.1 *A matriz $\mathcal{B}$ em (3.25) é tal que*

$$\mathcal{B} = 2\begin{bmatrix} v_0 & v_1 & 0 & v_2 & 0 & v_3 & \ldots & 0 & v_m \\ \overline{v}_1 & v_0 & v_1 & 0 & v_2 & 0 & \ldots & v_{m-1} & 0 \\ 0 & \overline{v}_1 & v_0 & v_1 & 0 & v_2 & \ldots & 0 & v_{m-1} \\ \overline{v}_2 & 0 & \overline{v}_1 & v_0 & v_1 & 0 & \ldots & v_{m-2} & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \overline{v}_m & 0 & \overline{v}_{m-1} & 0 & \overline{v}_{m-2} & \ldots & 0 & \overline{v}_1 & v_0 \end{bmatrix}, \tag{3.26}$$

onde

$$v_0 = m/2, \quad v_k = (1-\omega^{2k-1})^{-1}, \quad k = 1,2,\ldots,m.$$

A prova do lema acima é bastante simples e, portanto, será omitida. Note que $\mathcal{B}$ é uma matriz de *Toeplitz* e *hermitiana*. Como $v_{m-k+1} = (1-\omega^{2(m-k+1)-1})^{-1} = (1-\omega^{-2k+1})^{-1} = \overline{v}_k$, ela é também *circulante*.

Se substituirmos f pela expressão IDFT de F, $(1/n^2)\mathcal{W}^*F\mathcal{W}^*$ (veja (1.27)), e $\partial f/\partial\theta_J$ pelo lado direito de (3.23), na equação (3.22), obtemos

$$g_J = -\frac{1}{n^4}\mathrm{tr}\left\{(P\mathcal{W})\,(F^*\,\mathcal{B}\,\frac{\partial F}{\partial\theta_J})\,(P\mathcal{W})^*\right\}. \tag{3.27}$$

Lema 3.2 *Seja $\mathbf{Q}_{jk}$ a matriz $m\times m$ definida por*

$$\mathbf{Q}_{jk} = (P\mathcal{W})\,(F^*\,\mathcal{B}\,\mathbf{E}_{jk})\,(P\mathcal{W})^*, \quad (j,k)\in\tilde{\Lambda}. \tag{3.28}$$

Então ,

$$\mathbf{Q}_{n-j,n-k} = \overline{\mathbf{Q}}_{jk}.$$

Demonstração : Seja J a matriz $n \times n$ definida por

$$J = \begin{bmatrix} 1 & 0 \\ 0 & J' \end{bmatrix},$$

onde J' é a matriz reversa da identidade de ordem $(n-1) \times (n-1)$, i.e.,

$$J' = \begin{bmatrix} 0 & \dots & 0 & 1 \\ 0 & \dots & 1 & 0 \\ \dots & \dots & \dots & \dots \\ 1 & \dots & 0 & 0 \end{bmatrix}.$$

Desde que $JJ = I$, podemos então inserir o termo JJ entre os fatores da matriz $\mathbf{Q}_{n-j,n-k}$, cuja expressão é dada de maneira análoga a (3.28), de tal modo que

$$\mathbf{Q}_{n-j,n-k} = (P\mathcal{W}J)\ (J\,F^*\,J)\ (J\,\mathcal{B}\,J)\ (J\,\mathbf{E}_{n-j,n-k}\,J)\ (J(P\mathcal{W})^*). \tag{3.29}$$

Mostra-se que

$$\begin{aligned} P\mathcal{W}J &= \overline{P\mathcal{W}}, \\ J\,F^*\,J &= F^T, \\ J\,\mathcal{B}\,J &= \overline{\mathcal{B}}, \\ J\,\mathbf{E}_{n-j,n-k}\,J &= \mathbf{E}_{jk}. \end{aligned} \tag{3.30}$$

A verificação das equações (3.30) é simples, por isso nem todas serão feitas aqui. Verifiquemos, por exemplo, a segunda e a quarta. Com efeito, seja A uma matriz qualquer. Multiplicar A à direita por J equivale a manter a primeira coluna de A e permutar as demais, trocando a segunda com a última, a terceira com a penúltima e assim sucessivamente. O mesmo procedimento com as linhas de A vale quando multiplicamos A, à esquerda, por J. Assim, conclui-se que os termos gerais das matrizes JAJ e A se relacionam por

$$(JAJ)_{jk} = A_{n-j,n-k}. \tag{3.31}$$

Note portanto que a quarta equação de (3.30) é uma consequência imediata de (3.31). Já a prova da segunda segue de

$$(J\,F^*\,J)_{jk} = (F^*)_{n-j,n-k} \;=\; (\overline{F}^T)_{n-j,n-k} = \overline{F}_{n-k,n-j} \;=\; F_{kj} \;=\; (F^T)_{jk}.$$

Note que na primeira igualdade aplicamos (3.31), enquanto que na penúltima, usamos as relações de simetria da matriz F dadas em (1.28).

Finalmente a substituição de (3.30) em (3.29) nos dá

$$\mathbf{Q}_{n-j,n-k} = (\overline{P\mathcal{W}})\ (F^T\,\overline{\mathcal{B}}\,\mathbf{E}_{jk})\ (P\mathcal{W})^T = \overline{(P\mathcal{W})\ (F^*\,\mathcal{B}\,\mathbf{E}_{jk})\ (P\mathcal{W})^*} = \overline{\mathbf{Q}}_{jk} \quad \blacksquare$$

Proposição 3.1 *A componente g_J do gradiente em (3.27) é dada por*

$$g_J = -\frac{2}{n^4}\mathrm{Re}[\, i\,\alpha_{jk}\, e^{i\theta_{jk}}\, \mathrm{tr}(\mathbf{Q}_{jk})\,] = \frac{2\alpha_{jk}}{n^4}\,\mathrm{Im}[e^{i\theta_{jk}}\, \mathrm{tr}(\mathbf{Q}_{jk})\,], \tag{3.32}$$

para $\mathbf{Q}_{jk}$ dada em (3.28).

Demonstração: Obtém-se a prova desta proposição substituindo-se $\partial F/\partial\theta_J$ em (3.27) pelo membro direito da equação (3.24) e, em seguida, aplicando-se o lema 3.2 ■

O custo computacional para o cálculo de $\mathrm{tr}(\mathbf{Q}_{jk})$ ainda está elevado. De fato, se usamos um método de otimização que necessite de K passos para convergência, então em cada passo $k \in \{1, 2, \ldots, K\}$ o vetor gradiente $\mathbf{g}$ deverá ser calculado. Para cada componente g_J será necessário o cálculo do traço de uma matriz de tamanho $m \times m$, cuja expressão (3.28) envolve o produto de cinco matrizes. Se m for muito grande, o número de matrizes $\mathbf{Q}_{jk}$ que teríamos que calcular em cada passo, k, seria ainda maior, desde que $M = 2m^2 - 2$. Finalmente levando em conta o número de passos K do algoritmo, teríamos então que realizar $5\times(2m^2-2)\times K$ multiplicações de matrizes $m \times m$, o que não é nada animador. Assim, para que tornemos o ćalculo de $\mathrm{tr}(\mathbf{Q}_{jk})$ menos caro, adotamos os seguintes procedimentos: (1) determinamos apenas os elementos da diagonal principal de $\mathbf{Q}_{jk}$, já que o cálculo de $\mathrm{tr}(\mathbf{Q}_{jk})$ só depende destes elementos, (2) fazemos uso da estrutura de esparsidade da matriz $\mathbf{E}_{jk}$, (3) da simetria da matriz F^*, (4) da propriedade de $\mathcal{B}$ ser hermitiana e circulante, e (5) da estrutura de blocos da matriz de Fourier $\mathcal{W}$ dada na expressão (1.14).

Com efeito, levando em conta os procedimentos (1), (2) e (5), é possível mostrar o seguinte

Corolário 3.1 *A expressão do gradiente em (3.32) é equivalente a*

$$\begin{aligned} g_J &= -\frac{2}{n^4}\mathrm{Re}\left\{ -i\,\alpha_{jk}\, e^{-i\theta_{jk}} \sum_{r=0}^{m-1}\sum_{s=0}^{m-1} \omega^{r(s-k)} M_r(j,s) \right\} \\ &= -\frac{2}{n^4}\mathrm{Re}\left\{ -i\,\alpha_{jk}\, e^{-i\theta_{jk}} \left[\sum_{r\ \mathrm{par}}\sum_{s=0}^{m-1} \omega^{r(s-k)} M_0(j,s) + \sum_{r\ \mathrm{impar}}\sum_{s=0}^{m-1} \omega^{r(s-k)} M_1(j,s) \right] \right\}, \end{aligned}$$

onde

$$M_r(j,s) = (\mathcal{B}F)_{js} + (-1)^r (\mathcal{B}F)_{j,m+s}. \tag{3.33}$$

Note que o que diferencia $M_0(j,s)$ de $M_1(j,s)$ é o sinal de $(-1)^r$ que assume valores 1 ou -1 conforme r seja par ou ímpar. Por isso é suficiente calcularmos M_r

apenas para os valores $r = 0$ e $r = 1$. Note também que apesar de termos uma expressão que determina $M_r(j,s)$ para todo $0 \leq j \leq n-1$, $0 \leq s \leq m-1$, iremos considerar apenas os valores correspondentes a $(j,s) \in \tilde{\Lambda}$ (veja (3.13)).

Os procedimentos (3) e (4) serão usados no desenvolvimento do lado direito de (3.33), para obtermos as seguintes expressões de $M_r(j,s)$, $r = 0, 1$,

$$\begin{aligned}
M_r(0,s) &= 2\left\{v_0\left[F_{0,s} + (-1)^r F_{0,m+s}\right]\right. \\
&\quad \left. + \sum_{p=1}^{m} v_p\left[F_{2p-1,s} + (-1)^r F_{2p-1,m+s}\right]\right\}, \\
M_r(2q-1,s) &= 2\left\{\sum_{p=1}^{q} \overline{v}_{q-p+1}\left[F_{2p-2,s} + (-1)^r F_{2p-2,m+s}\right]\right. \\
&\quad + v_0\left[F_{2q-1,s} + (-1)^r F_{2q-1,m+s}\right] \\
&\quad \left. + \sum_{p=1}^{m-q} v_p\left[F_{2q+2p-2,s} + (-1)^r F_{2q+2p-2,m+s}\right]\right\}, \\
&\quad q = 1, 2, \ldots, m, \\
M_r(2q,s) &= 2\left\{\sum_{p=1}^{q} \overline{v}_{q-p+1}\left[F_{2p-1,s} + (-1)^r F_{2p-1,m+s}\right]\right. \\
&\quad + v_0\left[F_{2q,s} + (-1)^r F_{2q,m+s}\right] \\
&\quad \left. + \sum_{p=1}^{m-q} v_p\left[F_{2q+2p-1,s} + (-1)^r F_{2q+2p-1,m+s}\right]\right\}, \\
&\quad q = 1, 2, \ldots, m-1.
\end{aligned} \tag{3.34}$$

3.2.2.2 Comparação do custo computacional do cálculo do gradiente

O número de multiplicações e adições ([11], pp. 193,194) de números reais gastos no cálculo da FFT (*Fast Fourier Transform*) [ou IFFT - *Inverse Fast Fourier Transform*] de uma matriz de tamanho $n \times n$, para $n^2 = 2^\gamma$, $\gamma \in \mathbb{Z}_+$, depende do sistema numérico adotado pelo algoritmo que calcula a transformação. Em algoritmos que utilizam sistema de base 2, por exemplo, esse número é respectivamente

$$((2\gamma - 4)n^2 + 4) \text{ multiplicações} \quad \text{e} \quad ((3\gamma - 2)n^2 + 2) \text{ adições}.$$

Para sistemas de base 4, 8 e 16, o número de multiplicações e adições gastas são respectivamente

$$\begin{gathered}
(1.5\gamma - 4)\, n^2 + 4 \quad \text{e} \quad (2.75\gamma - 2)\, n^2 + 2, \\
(1.333\gamma - 4)\, n^2 + 4 \quad \text{e} \quad (2.75\gamma - 2)\, n^2 + 2, \\
(1.3125\gamma - 4)\, n^2 + 4 \quad \text{e} \quad (2.71875\gamma - 2)\, n^2 + 2.
\end{gathered}$$

De acordo com Brigham em [11], pp. 195, algoritmos que adotam base 4 e base 8 parecem ser os mais eficientes e, ao mesmo tempo, mais fáceis para a computação dos cálculos. Escolhemos, então, sistema de base 4 para comparar o custo computacional

do gradiente. Desde que $\gamma = \log_2 n^2$, o número total de operações, neste sistema, para o cálculo da IFFT de uma matriz real $n \times n$ é

$$4.25\, n^2 \log_2 n^2 - 6\,(n^2 - 1).$$

Para calcularmos cada componente g_J através da fórmula (3.22), são necessárias $2m^2 - 2$ IFFT's para a determinação das matrizes $\partial f/\partial \theta_J$, e 1 IFFT para a determinação de f. Além das IFFT's, precisamos de mais m^2 multiplicações e $m(m-1)$ adições de números reais para calcularmos o traço do produto das matrizes que aparecem em (3.22). Assim o número total de operações, N_1, gastas para o cálculo de $\mathbf{g}$ através de (3.22), num sistema de base 4, em cada passo k do algoritmo de otimização, é de

$$N_1 = (2m^2 - 1)\,[4.25\, n^2 \log_2 n^2 - 6(n^2 - 1)] + 2m^2 - m,$$

ou, equivalentemente, após substituirmos $n = 2m$,

$$N_1 = (20 + 68 \log_2 m)\, m^4 + (4 - 34 \log_2 m)\, m^2 - m - 6.$$

Temos portanto

$$\boxed{N_1 \simeq (20 + 68 \log_2 m)\, m^4.}$$

A seguir computamos o número de operações gastas para o cálculo do gradiente, feito através da fórmula otimizada, dada no corolário 3.1. As matrizes M_0 e M_1 são calculadas uma única vez, em cada passo k do algoritmo. Uma rápida inspeção nas fórmulas (3.34) nos permite concluir que para o cálculo de M_0 e M_1 são gastas $4m(m+1)^2$ operações (multiplicações e adições) de números complexos. De acordo com a expressão dada pelo corolário 3.1, além do custo para o cálculo de M_0 e M_1, temos também as multiplicações pelas potências de ω, as duas somatórias em r e s e a multiplicação final pelo termo $-i\ \alpha_{jk}\ e^{-i\theta_{jk}}$. Finalmente, desde que 1 soma de dois números complexos equivale a 2 somas de dois números reais; e 1 multiplicação de dois complexos equivale a 4 multiplicações e 2 adições reais, segue que o número total de operações reais, N_2, para o cálculo do gradiente $\mathbf{g}$ através da fórmula do corolário 3.1, num sistema de base 4, em cada passo k do algoritmo de otimização, é de $N_2 = 12m^4 + 16m^3 + 20m^2 + 16m + 24$. Temos portanto

$$\boxed{N_2 \simeq 12\, m^4.}$$

Comparando os valores aproximados de N_1 e N_2, vemos que

$$N_1 = qN_2; \quad q = \frac{20 + 68 \log_2 m}{12}.$$

Assim, para uma imagem de suporte 32×32, por exemplo, $q = 30$, i.e., o método que utiliza IFFT's é 30 vezes mais caro que o otimizado, portanto leva um tempo aproximadamente 30 vezes maior, para calcular o gradiente em cada passo.

3.3 Descrição do método L.BFGS.B

Para minimizar a função custo $L(\theta)$ em (3.11), bem como em (3.17), nós aplicamos o método L.BFGS.B [13], sigla em inglês que significa *limited memory BFGS with bound constraints*, desenvolvido pela equipe do ANLOTC, baseado no *método de otimização quasi-Newton de memória limitada.* O método tem se mostrado eficaz para resolver problemas de otimização não linear de grande porte, com ou sem restrições, e é tão eficiente para esses tipos de problemas quanto à sua versão anterior usada apenas para problemas sem restrições, o L.BFGS (*limited memory BFGS*, ou *BFGS de memória limitada*), que se encontra descrito em [55]. Os artigos em formato Postscript que abordam os métodos L.BFGS e L.BFGS.B estão disponíveis, via *ftp*, no endereço eecs.nwu.edu, no diretório pub/lbfgs/lbfgs - bcm.

O procedimento básico do método ([13], [95]) [1] consiste na minimização de uma função não linear, L, de M variáveis, $\theta = (\theta_1, \theta_2, \ldots, \theta_M)^T$,

$$\min L(\theta), \tag{3.35}$$

restrita a uma caixa limitada do $\mathbb{R}^M$. Apesar de ser recomendado para problemas com restrições, o L.BFGS.B tem se mostrado eficiente também para problemas sem restrições. Por isso, aproveitamos a periodicidade-2π de L respeito a cada fase, em nosso problema, para usar a versão do método sem restrições. No nosso caso, a função custo L a ser minimizada é dada pela expressão em (3.11) [ou (3.17), para o caso 2-D]. Tudo o que o usuário necessita fazer é determinar o valor da função custo L e de seu gradiente $\mathbf{g}$ avaliado em cada passo do algoritmo. Por esta razão o algoritmo pode ser útil para resolver grandes problemas dos quais o cálculo do Hessiano e/ou sua inversa torna-se difícil.

Como dissemos, o L.BFGS.B é uma extensão do L.BFGS que, por sua vez, se baseia no BFGS. O procedimento básico do BFGS se resume, como sabemos, na atualização de matrizes, que são aproximações do Hessiano, por uma matriz simétrica de posto 2 em cada passo. Em outras palavras:

Passo 0. Comece com uma matriz simétrica positiva definida qualquer, H_0, com qualquer ponto inicial $\theta^{(0)}$, e com $k = 0$.

Passo 1. Resolva $H_k \mathbf{d} = -\mathbf{g}_k$ para obter $\mathbf{d}_k$.

Passo 2. Encontre $\mathbf{p}_k = \gamma_k \mathbf{d}_k$ através de busca linear e, então, encontre $\theta^{(k+1)} = \theta^{(k)} + \mathbf{p}_k$ e $\mathbf{g}_{k+1}$.

Passo 3. Atribua $\mathbf{q}_k = \mathbf{g}_{k+1} - \mathbf{g}_k$ e

$$H_{k+1} = H_k + \frac{\mathbf{q}_k \mathbf{q}_k^T}{\mathbf{q}_k^T \mathbf{p}_k} - \frac{(H_k \mathbf{p}_k)(H_k \mathbf{p}_k)^T}{\mathbf{p}_k^T H_k \mathbf{p}_k}.$$

[1] Esta seção é uma tradução resumida das seções 1 e 3 da referência [95]. Ela explica resumidamente o método L.BFGS.B . Para uma descrição mais detalhada e completa desse método veja [13].

Passo 4. Atualize k e retorne ao passo 1.

Aqui $\mathbf{g}_k$ denota o gradiente avaliado no ponto em curso $\theta^{(k)}$.

A fórmula recursiva para as inversas das matrizes H_{k+1}, no Passo 3, é

$$H_{k+1}^{-1} = (I - \frac{\mathbf{p}_k \mathbf{q}_k^T}{\mathbf{q}_k^T \mathbf{p}_k}) H_k^{-1} (I - \frac{\mathbf{q}_k \mathbf{p}_k^T}{\mathbf{q}_k^T \mathbf{p}_k}) + \frac{\mathbf{p}_k \mathbf{p}_k^T}{\mathbf{q}_k^T \mathbf{p}_k}.$$

Usando esta recursão , resolve-se facilmente a equação do Passo 2 que determina $\mathbf{d}_k$. Entretanto, este procedimento requer a armazenagem de todos os k vetores dos passos prévios, $\mathbf{p}_k's$ e $\mathbf{q}_k's$; dos respectivos cálculos de produtos internos envolvendo estes vetores e os gradientes $\mathbf{g}_k's$; e o cálculo de $H_0^{-1}\mathbf{g}_0$. Com o método L.BFGS, apenas um número fixo de vetores $\mathbf{p}_k$ e $\mathbf{q}_k$, digamos 5 pares[2], são retidos. Quando novos vetores são adicionados na armazenagem, os antigos vetores são descartados.

Assim, em cada iteração a matriz de aproximação do Hessiano, H_{k+1}, obtida pelo método L.BFGS é atualizada e, então , usada para definir um modelo quadrático da função custo L. Uma direção de busca é então computada usando-se uma aproximação feita em dois passos: primeiro, usa-se o método de projeção dos gradientes para identificar um conjunto de variáveis ativas (não relevante em nosso caso); então o modelo quadrático é aproximadamente minimizado com respeito às variáveis livres. A direção de busca é definida como sendo o vetor direcionado a partir do ponto da iteração em curso ao ponto minimizador da aproximação quadrática. A busca linear ao longo desta direção segue as subrotinas descritas em [14]. Esta é portanto a descrição resumida do L.BFGS.B .

As vantagens do L.BFGS.B são : (i) a rotina é fácil de usar; o usuário não precisa fornecer informações sobre a matriz Hessiana; (ii) a armazenagem dos dados é modesta e pode ser controlada pelo usuário; (iii) o custo da iteração é baixo e independe das propriedades da função custo. Assim, L.BFGS.B é fortemente recomendado para problemas muito grandes dos quais a matriz Hessiana é não esparsa ou é difícil de ser computada.

Entretanto L.BFGS.B apresenta as seguintes desvantagens: (i) ele não garante a convergência local quadrática típica ao algoritmo de Newton; (ii) em problemas altamente mal-condicionados, ele pode falhar ao tentar obter uma solução com alta precisão (*oversolving*); (iii) ele não permite fazer uso do conhecimento da estrutura do problema para se obter uma convergência mais rápida. (Respeito à primeira desvantagem, notamos que o Método de Newton que calcula, e ademais inverte, o Hessiano, não é viável aqui).

[2]de fato adotamos esse número fixo de pares em nosso algoritmo para todas as simulações numéricas nesta tese.

São várias as formas de se interromper a execução do algoritmo. Algumas delas podem ser determinadas pelo próprio usuário através da inclusão de instruções específicas ao programa diretor. As outras baseiam-se nos critérios de parada que são condições impostas aos valores da função custo e da projeção do gradiente a serem atingidos em uma situação limite. Em outras palavras, o algoritmo é interrompido quando a função custo L satisfizer ao primeiro critério de parada:

$$\frac{|L_k - L_{k+1}|}{\max(|L_{k+1}|, |L_k|, 1)} \leq \textbf{factr} * \textbf{epsmch}, \tag{3.36}$$

onde **epsmch** é a precisão da máquina, a qual é gerada automaticamente pela rotina, e **factr** é um parâmetro controlado pelo usuário. Esse critério força a parada do algoritmo quando a mudança para a próxima iterada da função custo torna-se suficientemente pequena. Valores típicos para **factr** em um computador com 15 dígitos para variáveis em dupla precisão são: **factr** = 1.0e+10 para baixa precisão; **factr** = 1.0e+7 para precisão moderada e **factr** = 1.0e+1 para precisão extremamente alta. Se **factr** = 0, o algoritmo é interrompido somente se a função custo permanece inalterada após uma iteração.

O segundo critério de parada, indicado para problemas com restrições (e também para problemas sem restrições, conforme discussão no parágrafo seguinte) é baseado na projeção do vetor gradiente sobre o espaço tangente às restrições ativas, e pode ser igual a zero num ponto de mínimo local. O algoritmo pára de rodar quando a norma-infinito do gradiente projetado torna-se suficientemente pequena,

$$\|\text{proj } \mathbf{g}\|_\infty \leq \textbf{pgtol}. \tag{3.37}$$

O parâmetro **pgtol** também é controlado pelo usuário.

Existe uma variável, intrínseca ao programa diretor do L.BFGS.B , que serve para identificar se o problema a ser solucionado é com ou sem restrições. Em nosso caso, por se tratar de problema sem restrições , o valor atribuído a esta variável deverá ser *zero*. Após tal atribuição, o algoritmo, sabendo que se trata de problema sem restrições , passa a armazenar em "proj **g**" simplesmente o valor do gradiente **g**. Por esse motivo, mesmo em problemas sem retrições , consideraremos o critério (3.37), controlado por valores atribuídos ao parâmetro **pgtol**, tendo-se em mente que os valores obtidos para $\|\text{proj } \mathbf{g}\|_\infty$ referem-se à norma infinita do gradiente (não projetado) da função custo. Maiores detalhes sobre o funcionamento do método L.BFGS.B bem como instruções de operacionalização e interface com o ususário podem ser encontrados em [13], [95] e [14].

Vamos agora fazer um resumo do nosso "novo" método para resolver o problema da fase. Ele nada mais é do que a aplicação do L.BFGS.B para a minimização da nossa função custo específica, definida em (3.11) - caso 1-D - e em (3.17) - caso 2-D. Por isso designá-lo-emos pelo nome *Minimização dos pixels do Objeto Fora do Suporte via* L.BFGS.B , ou simplesmente MOFS/L.BFGS.B .

3.4 Descrição do método MOFS/L.BFGS.B

O método L.BFGS.B é composto de um programa diretor e várias subrotinas, escritos em linguagem *Fortran*, que o auxiliam na busca do minimizador. O programa diretor, além de conter instruções sobre o funcionamento do método, permite que o usuário defina a função custo a ser otimizada, juntamente com o seu respectivo gradiente, e a precisão do erro de convergência, através do controle dos parâmetros **factr** e **pgtol**, conforme comentamos na seção anterior. Além disto, ele permite a incorporação de novas subrotinas ao programa principal, modificações em algumas subrotinas já existentes, de modo a atender às necessidades específicas do problema. Além dos métodos matemáticos adotados, este tipo de iteração dinâmica com o usuário faz do pacote desenvolvido pela equipe do ANLOTC um instrumento bastante eficiente na busca de soluções para problemas de otimização não lineares e que são bastante complexos. Assim, o método que iremos descrever foi obtido a partir de uma adaptação do Pacote ao nosso problema específico. Criamos muitas novas subrotinas, escritas em *Fortran* e *Octave* [3], e alteramos outras já existentes. Algumas das novas subrotinas foram criadas para gerar os dados das amplitudes e os vários tipos de condições iniciais. Outras foram criadas para fazer a conexão entre nosso método e os algoritmos de Fienup. Também fizemos modificações no programa diretor do pacote e criamos uma nova subrotina, a qual chamaremos de BFGS.f, onde podemos controlar critérios de paradas, precisão dos resultados e, o mais importante, definir a função custo e seu respectivo gradiente. Esta é a principal subrotina do programa, pois é nela que são calculados os pontos da sequência que espera-se convergir à solução do problema. Os parâmetros da nossa subrotina BFGS.f são muitos, por isso exibiremos apenas alguns deles. Os parâmetros principais de entrada são aqueles que definem os dados do problema, que em nosso caso são representados pelas amplitudes de Fourier e as fases de Fourier nos cantos, e a condição inicial para as fases intermediárias de Fourier. O parâmetro de saída é a estimativa da solução.

Devido à complexidade dos programas e adaptações envolvidas, faremos apenas um breve resumo do nosso novo método para que o leitor compreenda o seu funcionamento. Para simplificação, denotaremos

$$L_k := L(\theta^{(k)}); \quad \mathbf{g}_k := \mathbf{g}(\theta^{(k)}).$$

Vale ressaltar novamente que para construirmos L_k, devemos primeiramente determinar $F(\theta^{(k)})$, depois $f(\theta^{(k)}) = \mathcal{F}^{-1}[F(\theta^{(k)})]$, e então aplicar a fórmula (3.11), se caso 1-D, ou (3.17), se caso 2-D. Para o cálculo do gradiente no caso 1-D, calculamos primeiramente $\partial f(\theta^{(k)})/\partial\theta_j$ através de (3.19) e (3.20) para, então, substituí-lo, juntamente com $f(\theta^{(k)})$, em (3.18). Para o caso 2-D, nós substituímos as entradas da

[3] Gnu Octave, v. 2.1.44, é um pacote, de domínio público, similar ao Matlab, que pode ser baixado da página www.octave.org.

matriz $F(\theta^{(k)})$ em (3.34) para obtermos as matrizes M_0 e M_1 que deverão ser, em seguida, substituídas na expressão do gradiente do corolário 3.1.

Como já observamos, a construção de $F(\theta^{(k)})$ depende não só de θ, mas também dos outros parâmetros, aqui ocultos, α, α_c e ϕ_c. Consequentemente nossa função custo $L(\theta)$ também depende destes parâmetros que, como já sabemos, são mantidos fixos durante o processo de busca do minimizador. Portanto será conveniente, às vezes, que explicitemos estes parâmetros na notação da função custo, tais como

$$L(\theta) = L(\alpha_c, \phi_c, \alpha, \theta).$$

O Algoritmo MOFS/L.BFGS.B

- Calcule α, α_c e ϕ_c, relativos ao objeto f.
- Determine uma condição inicial $\theta^{(0)}$ e as tolerâncias **factr** e **pgtol**.
- Calcule L_0 e $\mathbf{g}_0$.
- Aplique L.BFGS.B em L_0 e $\mathbf{g}_0$ para obter $\theta^{(1)}$.
- Calcule L_1 e $\mathbf{g}_1$ e faça $k = 1$.
- Enquanto $\dfrac{|L_k - L_{k-1}|}{\max(|L_{k-1}|, |L_k|, 1)} > \mathbf{factr} \star \mathbf{epsmch}$ ou $\|\text{proj } \mathbf{g}_k\|_\infty > \mathbf{pgtol}$,
  - Aplique L.BFGS.B em L_k e $\mathbf{g}_k$ para obter $\theta^{(k+1)}$.
  - Calcule L_{k+1} e $\mathbf{g}_{k+1}$ e atualize k para $k + 1$.

  Fim
- Calcule a estimativa, $\tilde{f} = \mathcal{F}^{-1}[F(\tilde{\phi})]$, do objeto, onde $\tilde{\phi} = \lim \theta^{(k)}$.

No capítulo 4 descreveremos em detalhes alguns tipos de condição inicial $\theta^{(0)}$ que iremos adotar em nossos experimentos.

De acordo com o primeiro passo do algoritmo, nós estamos pré fixando para a variável ϕ_c o mesmo valor das fases dos cantos da DFT do objeto original f. Em outras palavras, a busca do minimizador da função custo $L(\theta)$ está sendo feita sobre o mesmo toróide, T_{ϕ_c} (veja Definições 1.3 e 1.4), que *contém* o objeto original. Em geral esta não é uma regra rígida do algoritmo. Dependendo da quantidade de ruídos introduzidos nas amplitudes de Fourier, ou em métodos que utilizam combinações do MOFS/BFGS com o HIO, será conveniente fazermos a busca sobre outros toróides. Nestes casos, devemos atribuir às componentes ϕ_c valores (iguais a 0's ou π's) correspondentes aos valores que definem o toróide sobre o qual se deseja fazer a busca.

3.5 MOFS/L.BFGS.B : convergência e mínimos locais (globais)

Nesta seção faremos uma breve descrição do comportamento de convergência do método MOFS/L.BFGS.B em ambos os casos 1-D e 2-D.

Os pontos críticos da função custo $L(\theta)$ são os pontos $\tilde{\phi}$ que anulam o gradiente de L. Assim, para sabermos se um dado ponto crítico $\tilde{\phi}$ é um ponto de mínimo local para a função L, devemos calcular a matriz hessiana de L neste ponto. Se a matriz hessiana for positiva definida, então $\tilde{\phi}$ será um mínimo local. Se além disso tivermos $L(\tilde{\phi}) = 0$, então $\tilde{\phi}$ será um mínimo global de L. A expressão que determina a matriz hessiana de L está dada em (3.46) e (3.48).

Definição 3.1 *Seja $\tilde{\phi}$ o ponto de convergência obtido pelo método* MOFS/L.BFGS.B *ao minimizarmos a função custo $L(\theta)$. Diremos que o algoritmo* MOFS/L.BFGS.B *convergiu a mínimo local se $\tilde{\phi}$ for um ponto de mínimo local da função $L(\theta)$. Se além disso $L(\tilde{\phi}) = 0$, diremos que* MOFS/L.BFGS.B *convergiu a mínimo global*[4]

Às vezes será mais conveniente tratarmos o próprio objeto $f(\tilde{\phi})$ como o mínimo local (global) atingido pelo MOFS/L.BFGS.B , e não a fase $\tilde{\phi}$. Cabe ao leitor, portanto, descobrir qual o sentido utilizado em cada contexto.

Em nossos experimentos numéricos foi verificado que:

- Em problemas sem ruídos nos dados das amplitudes, se iniciamos de um ponto aleatório qualquer, $\theta^{(0)}$, o método MOFS/L.BFGS.B sempre converge a um ponto crítico de L. No caso 1-D constatamos numericamente em todos os exemplos testados que tais pontos críticos são na verdade pontos de mínimo local. Já no caso 2-D não fizemos nenhuma verificação numérica, porém acreditamos que a maioria dos pontos críticos encontrados são também pontos de mínimo local. O algoritmo converge eventualmente a mínimo global. Isto foi verificado mesmo no caso em que os dados das amplitudes são acrescidos de ruídos.

- Para o caso 1-D, sem ruídos, a convergência se deu, em 100% dos testes realizados, a mínimo globais. Assim podemos conjecturar que

  no caso unidimensional, para dados das amplitudes sem ruídos, MOFS/L.BFGS.B *sempre converge numericamente a mínimos globais.*

[4] É claro que a nossa definição sobre convergência é feita sempre sob o ponto de vista numérico, onde se admite erros suficientemente pequenos nos cálculos da função custo, do gradiente, do hessiano, etc.

• Para o caso 2-D, o método mostrou-se incapaz de localizar os mínimos globais quando usamos condições iniciais aleatórias (veja seção 4.1), exceto para casos de imagens de tamanho 2×2 e 4×4 dentro do suporte, onde constatamos convergência a mínimo global em todos os exemplos testados. Entretanto, utilizando certas informações (seção 4.1) sobre a fase original na condição inicial, verificamos que MOFS/L.BFGS.B convergiu a mínimo global em todos os exemplos testados. Mesmo na presença de ruídos nos dados das amplitudes, constatamos convergência do método a pontos bem próximos da solução verdadeira. Este resultado já mostra a superioridade do método MOFS/L.BFGS.B em relação ao HIO em localizar a imagem procurada (pelo menos a vizinhança na qual ela pertence) na presença de ruídos e quando se tem disponível certas informações a respeito da fase original. Nestas mesmas condições , o método também mostrou-se superior ao ER com relação à precisão do erro entre a imagem original e a imagem encontrada, medido no domínio do objeto.

3.6 Relação entre os pares de pontos fixos do ER e os pontos críticos de $L(\phi)$ encontrados pelo MOFS/L.BFGS.B

Nesta seção daremos a relação entre os pares de pontos fixos do algoritmo ER e os pontos críticos da função $L(\phi)$. Como sempre, faremos as análises para os casos 1-D e 2-D separadamente.

3.6.1 A relação no caso 1-D:

Consideremos a função d dada em (2.21). De acordo com o teorema de Parseval [11], $d(h)$ poderá ser reescrita como

$$d(h) = \frac{1}{2n}\|F\|^2 - \frac{1}{2}\|h_{(1)}\|^2. \tag{3.38}$$

Desde que estamos sobre uma variedade, devemos levar em conta a restrição $h \in \tau$ ao calcularmos a derivada da função $d(h)$. Assim, dado $h \in \mathbb{R}^n$, sabemos que $h \in \tau$ se e somente se a transformada de Fourier H tem as mesmas amplitudes de F e tem a simetria típica das transformadas de Fourier de objetos reais, i.e., $h \in \tau \Longleftrightarrow$

$$H = (\alpha_0 e^{i\theta_0}, \alpha_1 e^{i\theta_1}, \ldots, \alpha_{m-1} e^{i\theta_{m-1}}, \alpha_m e^{i\theta_m}, \alpha_{m-1} e^{-i\theta_{m-1}}, \ldots, \alpha_1 e^{-i\theta_1})^T \tag{3.39}$$

onde $\alpha_u \equiv F_u$ e θ_u representa a fase de H_u. Se denotarmos $\phi_{cH} = (\theta_0, \theta_m)^T$, é fácil verificar que a transformada H dada em (3.39) satisfaz

$$H = F(\alpha_c, \phi_{cH}, \alpha, \theta) \tag{3.40}$$

para F definida como em (3.4). Então

$$h = f(\theta) = \mathcal{F}^{-1}[F(\theta)]. \tag{3.41}$$

Omitiremos os parâmetros das expressões (3.40) e (3.41) para obtermos as expressões equivalentes

$$H = F, \quad h = f. \tag{3.42}$$

Assim a função d dada em (3.38) se reescreve em termos da composta

$$(d \circ f)(\theta) = \frac{1}{2n}\|F\|^2 - \frac{1}{2}\|f_{(1)}(\theta)\|^2. \tag{3.43}$$

Desde que α_u são dados conhecidos e θ_0, θ_m podem também ser vistos como constantes (pois seus valores são sempre iguais a 0 ou π), concluímos que diferenciar d sobre τ com respeito a h equivale a diferenciar a composta $d \circ f$ sobre o aberto $\mathbb{R}^{m-1}$ com respeito a $\theta \equiv (\theta_1, \theta_2, \ldots, \theta_{m-1})^T$.

É importante observarmos que **a função $d \circ f$ dada por (3.43) é a própria função custo L do problema de minimização dado em (3.10).** Assim, podemos fazer uso da expressão de sua derivada para obtermos uma expressão para $d'(h)$. Em outros termos,

$$d'(h) = \nabla_\theta (d \circ f)(\theta) = \left[\frac{\partial (d \circ f)(\theta)}{\partial \theta_j}\right]_{j=1}^{m-1},$$

onde $\partial(d \circ f)/\partial\theta_j$ é, de acordo com (3.18), dado por

$$\frac{\partial (d \circ f)}{\partial \theta_j} = -(Pf)^T (P\frac{\partial f}{\partial \theta_j}), \tag{3.44}$$

e $\partial f/\partial\theta_j$ dado em (3.19).

Derivando os dois membros de (3.44) parcialmente em relação a θ_k, obtemos uma expressão para o hessiano

$$[d''(h)]_{jk} = \frac{\partial^2 (d \circ f)(\theta)}{\partial\theta_k \partial\theta_j}, \quad j,k \in \{1, 2, \ldots, m-1\}, \tag{3.45}$$

para

$$\frac{\partial^2 (d \circ f)}{\partial\theta_k \partial\theta_j} = \begin{cases} -\left(P\dfrac{\partial f}{\partial\theta_k}\right)^T \left(P\dfrac{\partial f}{\theta_j}\right); & k \neq j \\ -\left[\left(P\dfrac{\partial f}{\partial\theta_j}\right)^T \left(P\dfrac{\partial f}{\partial\theta_j}\right) + (Pf)^T \left(P\dfrac{\partial^2 f}{\partial\theta_j^2}\right)\right]; & j = k. \end{cases} \tag{3.46}$$

Aqui, o vetor $\partial^2 f/\partial\theta_j^2$ tem a mesma forma do vetor $\partial F/\partial\theta_j$ dado em (3.20), com os elementos não nulos dados respectivamente por $-\alpha_j e^{i\theta_j}$ e $-\alpha_j e^{-i\theta_j}$.

Assim os pares de pontos fixos do algoritmo ER estão relacionados com os pontos críticos da função $d \circ f$ como mostra a seguinte

Proposição 3.2 *Seja* $h \in \tau$. *Então*

$$\frac{\partial(d \circ f)}{\partial\theta_j} = -\frac{1}{n^2} H^* \mathcal{W}_{(1)} \mathcal{W}_{(1)}^* \boldsymbol{\Delta}_j \tag{3.47}$$

Demonstração : Comparando o vetor $\boldsymbol{\Delta}_j$ do lema 2.1 com a expressão dada em (3.20), verifica-se imediatamente que

$$\boldsymbol{\Delta}_j = \frac{\partial F}{\partial\theta_j}.$$

Então , desde que h é real, segue de (3.44) e das relações estabelecidas pela expressão (3.42) que

$$\begin{aligned}
\frac{\partial(d \circ f)}{\partial\theta_j} &= -(Pf)^*(P\frac{\partial f}{\partial\theta_j}) = -\left[P\frac{1}{n}\mathcal{W}^* F\right]^* \left[P\frac{1}{n}\mathcal{W}^*\frac{\partial F}{\partial\theta_j}\right] \\
&= -\frac{1}{n^2} H^* \mathcal{W} P^T P \mathcal{W}^* \boldsymbol{\Delta}_j = -\frac{1}{n^2} H^* \mathcal{W} M \mathcal{W}^* \boldsymbol{\Delta}_j \\
&= -\frac{1}{n^2} H^* \mathcal{W}_{(1)} \mathcal{W}_{(1)}^* \boldsymbol{\Delta}_j \quad \blacksquare
\end{aligned}$$

3.6.2 A relação no caso 2-D:

Resultados similares podem ser inteiramente reproduzidos para o caso 2-D. Por exemplo, as expressões equivalentes à (3.43), (3.44) para o caso 2-D são :

$$\begin{aligned}
(d \circ f)(\theta) &= \frac{1}{2n^2}\|F\|^2 - \frac{1}{2}\|f_{(11)}(\theta)\|^2, \\
\frac{\partial(d \circ f)}{\partial\theta_J} &= -\mathrm{tr}\left\{(PfP^T)^T(P\frac{\partial f}{\partial\theta_J}P^T)\right\},
\end{aligned}$$

onde $\partial f/\partial\theta_J$ agora representa a transformada inversa da expressão em (3.24).

A expressão equivalente a (3.46) é:

$$\frac{\partial^2(d \circ f)}{\partial\theta_K\partial\theta_J} = \begin{cases} -\mathrm{tr}\left[\left(P\dfrac{\partial f}{\partial\theta_K}P^T\right)^T\left(P\dfrac{\partial f}{\theta_J}P^T\right)\right]; & K \neq J \\ -\mathrm{tr}\left[\left(P\dfrac{\partial f}{\partial\theta_J}P^T\right)^T\left(P\dfrac{\partial f}{\partial\theta_J}P^T\right) + (PfP^T)^T\left(P\dfrac{\partial^2 f}{\partial\theta_J^2}P^T\right)\right]; & J = K, \end{cases} \tag{3.48}$$

para todo $J, K = 1, 2, \ldots, 2(m^2 - 1)$.

Aqui, a matriz $\partial^2 f/\partial\theta_J^2$ tem a mesma representação da expressão de $\partial f/\partial\theta_J$ em (3.24) com os elementos não nulos dados respectivamente por $-\alpha_{jk}e^{i\theta_{jk}}$ e $-\alpha_{jk}e^{-i\theta_{jk}}$.

Proposição 3.3 *Seja $h \in \tau$. Então*

$$\frac{\partial(d \circ f)}{\partial\theta_J} = -\frac{1}{n^4}\mathrm{tr}[H^*(\mathcal{W}_{(\mathbf{1})}\mathcal{W}_{(\mathbf{1})}^*\boldsymbol{\Delta}_J\mathcal{W}_{(\mathbf{1})}\mathcal{W}_{(\mathbf{1})}^*)] \tag{3.49}$$

onde $\boldsymbol{\Delta}_J$ é a matriz $\boldsymbol{\Delta}_{jk}$ dada por (2.11).

Demonstração : Comparando a matriz $\boldsymbol{\Delta}_{jk}$ em (2.11) com a expressão dada em (3.24), verifica-se imediatamente que

$$\boldsymbol{\Delta}_{jk} = \frac{\partial F}{\partial\theta_J}.$$

Então

$$\begin{aligned}
\frac{\partial(d \circ f)}{\partial\theta_J} &= -\mathrm{tr}\left\{(PfP^*)^*(P\frac{\partial f}{\partial\theta_J}P^*)\right\} \\
&= -\mathrm{tr}\left\{[P(\frac{1}{n^2}\mathcal{W}^*F\mathcal{W}^*)P^*]^*[P(\frac{1}{n^2}\mathcal{W}^*\frac{\partial F}{\partial\theta_J}\mathcal{W}^*)P^*]\right\} \\
&= -\frac{1}{n^4}\mathrm{tr}[P\mathcal{W}H^*(\mathcal{W}P^TP\mathcal{W}^*)\boldsymbol{\Delta}_J\mathcal{W}^*P^T] \\
&= -\frac{1}{n^4}\mathrm{tr}[H^*(\mathcal{W}P^TP\mathcal{W}^*)\boldsymbol{\Delta}_J(\mathcal{W}^*P^TP\mathcal{W})] \\
&= -\frac{1}{n^4}\mathrm{tr}[H^*(\mathcal{W}_{(\mathbf{1})}\mathcal{W}_{(\mathbf{1})}^*\boldsymbol{\Delta}_J\mathcal{W}_{(\mathbf{1})}\mathcal{W}_{(\mathbf{1})}^*)] \quad \blacksquare
\end{aligned}$$

3.7 Comentários:

Os comentários a seguir referem-se ao caso 2-D, para dados das amplitudes sem ruídos. Comentários similares valem para o caso 1-D.

Como dissemos anteriormente a função $d \circ f$ é a própria função custo L do problema de minimização apresentado na seção 3.1. Assim, segue que os pares de pontos fixos do algoritmo ER estão relacionados com os pontos críticos de L da seguinte forma: suponha que, dada uma condição inicial $g^{(0)}$, seja $(\tilde{g}, \tilde{h})$ o par de pontos fixos obtido pelo ER. Sejam $\phi_{\tilde{G}}$ e $\phi_{\tilde{H}}$ os vetores fases (ver (1.34)) de $\tilde{G}$ e $\tilde{H}$ respectivamente. Segue do passo 2 do algoritmo ER que $\phi_{\tilde{G}} = \phi_{\tilde{H}}$. Escrevamos $\phi_{\tilde{G}}$, de acordo com (1.37), como

$$\phi_{\tilde{G}} = (\tilde{\phi}_{00}; \tilde{\phi}_{(1)}; \tilde{\phi}_{0m}; \tilde{\phi}_{(2)}; \tilde{\phi}_{m0}; \tilde{\phi}_{(3)}; \tilde{\phi}_{mm})^T,$$

e seja

$$\tilde{\phi} = (\tilde{\phi}_{(1)}; \tilde{\phi}_{(2)}; \tilde{\phi}_{(3)})^T$$

o vetor das fases intermediárias de $\tilde{G}$. As proposições 2.2 e 3.3 nos dizem que $(\tilde{g}, \tilde{h})$ é um par de pontos fixos para o algoritmo ER se e somente se $\tilde{\phi}$ é um ponto crítico de L. É de se esperar, portanto, que os pontos de mínimos locais localizados por MOFS/L.BFGS.B e ER sejam preservados por estes métodos. Em outras palavras, se iniciarmos MOFS/ L.BFGS.B a partir do ponto $\tilde{\phi}$, espera-se obter uma sequência de pontos, $\phi^{(k)}$, que se mantém fixa em $\tilde{\phi}$, i.e., $\phi^{(k)} \equiv \tilde{\phi}\ \forall\ k$. Reciprocamente, dada uma condição inicial $\phi^{(0)}$, suponha que MOFS/L.BFGS.B convirja a $\tilde{\phi}$. Seja $\tilde{f} = f(\tilde{\phi})$. Agora, usando $\tilde{f}$, como condição inicial, espera-se que ER mantenha-se em $\tilde{f}$.

Numericamente, no entanto, este fenômeno não foi verificado, em 100% dos casos (mas foi verificado na maioria deles). De fato, dado o vetor α_F, se nosso método MOFS/L.BFGS.B for excutado sobre o mesmo Toróide, T_{ϕ_c}, da imagem original, então ele permanece e converge, digamos para $\tilde{f} = f(\tilde{\phi})$, sobre esse mesmo Toróide. Em seguida, ao aplicarmos ER sobre $\tilde{f}$, a instabilidade numérica relativa a, por exemplo erros de arredondamento, etc, poderá forçar o ER a mudar de Toróide, pois do modo como ele foi definido (veja 2.1), suas fases, inclusive as dos cantos (que definem os Toróides), são sempre alteradas, em cada passo do algoritmo. Reciprocamente, se ER convergir para $\tilde{f}$ e em seguida aplicamos MOFS/L.BFGS.B , devemos ter o cuidado de executar nosso método sobre o próprio Toróide de $\tilde{f}$, para que sua convergência seja o próprio $\tilde{f}$.

Capítulo 4

Resultados numéricos

Neste capítulo apresentaremos os resultados numéricos obtidos ao rodarmos os algoritmos MOFS/L.BFGS.B , HIO, ER e combinações tais como ER+HIO (= EH), MOFS/L.BFGS.B + HIO, etc. Para simplificação de notação, daqui em diante nós nos referiremos ao método MOFS/L.BFGS.B simplesmente pelas iniciais MOFS .

Todos os experimentos foram realizados em um Pentium IV, 1.8MHz, Toshiba Sattelite 2410 S 203, com precisão (veja (3.36)) **epsmch** $= 1.084 \times 10^{-19}$.

Não faremos aqui nenhuma descrição detalhada do comportamento dos métodos ER , HIO e EH pois eles já foram extensivamente abordados por diversos autores [25], [26], [81], [83], [84], [85], [86], [5]. Discutiremos tais métodos apenas em alguns casos específicos, onde faremos comparações com nosso método.

Na seção 4.3.1 discutimos os principais resultados no caso 1-D na ausência de ruídos nos dados das amplitudes (os resultados relativos aos dados com ruídos serão discutidos no capítulo 5). Os resultados relativos ao caso 2-D, com e sem ruídos, serão discutidos no resto deste capítulo.

Para aplicarmos MOFS, bem como os algoritmos de Fienup, nós precisamos de dados das amplitudes que sejam consistentes. Por isso precisamos inicialmente gerar o objeto original e em seguida calcular sua DFT, F, para então obtermos as amplitudes de Fourier, que deverão ser usadas na formação dos vetores α e α_c. Para a construção de F e α, α_c, nós usamos respectivamente as funções intrínsecas do Octave[1], **fft2** e **abs**. Para a determinação de ϕ_c nós usamos **angle**.

Para o problema da recuperação de objetos reais com restrição de suporte, nós consideramos dois tipos básicos de objetos: os *objetos aleatórios* e os *não aleatórios.*

Para objetos do primeiro tipo nós geramos f com valores nulos fora do suporte e valores aleatórios entre 0 e 1 dentro do suporte. Fazemos isso usando a função **rand** do Octave que gera variáveis aleatórias através do comando:

$$\begin{array}{c} \mathbf{rand}(\text{'seed'}, s) \\ X = \mathbf{rand}(\text{p,q}). \end{array} \tag{4.1}$$

[1]Usamos a versão 2.1.44 que vem incluída no software Linux-Mandrake, v. 9.0

A função desses comandos é a de atribuir à variável X (que pode ser um número real, vetor ou matriz, conforme valores especificados aos inteiros positivos p e q) números reais aleatórios entre 0 e 1. O parâmetro s tem a função de *indexar* a variável X aos números inteiros positivos. Assim, se em um determinado experimento, atribuirmos, p.ex., o valor 1 à variável s, então após a execução do comando (4.1), obtemos as componentes de X cujos valores passam a ser identificadas pelo número inteiro 1. Então , sempre que quisermos fazer um novo experimento, usando para X as mesmas componentes do experimento anterior, basta que atribuamos, neste novo experimento, o mesmo valor 1 para a variável s. Por convenção, escreveremos

$$s = \text{seed}(X),$$

quando quisermos identificar o parâmetro *seed* que *gerou* X.

Para objetos não aleatórios (considerados apenas para o caso 2-D), nós atribuímos valores específicos às suas componentes, de modo a formar dentro do suporte, imagens que não apresentem nenhum tipo de simetria.

Para problemas com ruídos, será necessário adotarmos critérios específicos sobre como adicionar ou multiplicar ruídos a uma dada variável. No caso de *ruídos aditivos*, obtemos uma perturbação , $\tilde{\mathbf{x}}$, de um determinado vetor $\mathbf{x} \in \mathbb{R}^p$ ao adicionarmos *pequenas* quantidades às componentes de $\mathbf{x}$ (neste capítulo assumiremos $\mathbf{x} = \alpha_F$ ou $\mathbf{x} = \phi$). Faremos isso obedecendo ao seguinte critério

$$\tilde{\mathbf{x}} = \mathbf{x} + \eta_{\mathbf{x}}, \tag{4.2}$$

onde $\eta_{\mathbf{x}} \in \mathbb{R}^p$ é o ruído a ser acrescentado ao vetor $\mathbf{x}$, calculado através da seguinte fórmula

$$\eta_{\mathbf{x}} = \varepsilon_{\mathbf{x}} \frac{\|\mathbf{x}\|}{p}(\mathbf{r}_{\mathbf{x}} - 0.5). \tag{4.3}$$

A soma algébrica entre vetor (ou matriz) e número real, como em $(\mathbf{r}_{\mathbf{x}} - 0.5)$, por exemplo, significa a soma entre cada componente deste vetor (ou matriz) com este número.

O número $\varepsilon_{\mathbf{x}} > 0$ é um parâmetro calibrador e será denominado o *índice de ruído* de $\eta_{\mathbf{x}}$.

O vetor $\mathbf{r}_{\mathbf{x}}$ é um vetor aleatório, gerado no Octave, de mesma dimensão do vetor $\mathbf{x}$. Para gerar $\mathbf{r}_{\mathbf{x}}$, usamos a função **rand** do Octave. Por convenção, adotaremos

$$\text{seed}(\eta_{\mathbf{x}}) = \text{seed}(\mathbf{r}_{\mathbf{x}}).$$

Note que como os valores gerados por **rand** estão compreendidos entre 0 e 1, adicionamos a quantidade -0.5 a cada componente de $\mathbf{r}_{\mathbf{x}}$ para que possamos garantir a inclusão de ruídos tanto positivos quanto negativos, num intervalo entre -0.5 e 0.5, às coordenadas do vetor $\mathbf{x}$. Entretanto devemos ser cautelosos quanto à escolha de $\varepsilon_{\mathbf{x}}$

quando $\mathbf{x} = \alpha_F$, pois sendo α_F um vetor de coordenadas sempre não negativas, $\tilde{\alpha}_F$ deverá ainda ser um vetor de coordenadas não negativas. Assim, ao criarmos o vetor $\tilde{\mathbf{x}}$, devemos escolher $\varepsilon_{\mathbf{x}}$ satisfazendo $\varepsilon_{\mathbf{x}} \, \|\mathbf{x}\| \, |r_i - 0.5| \, / \, p \, \leq \, |x_i| \;\; \forall \, i = 0, 1, \ldots, p-1$, para garantir que ele tenha coordenadas positivas. Desde que $0 \leq r_i \leq 1$, segue que $\varepsilon_{\mathbf{x}}$ deve satisfazer

$$\varepsilon_{\mathbf{x}} \leq \frac{2p}{\|\mathbf{x}\|} \min |x_i|. \tag{4.4}$$

Aqui x_i e r_i são as coordenadas dos vetores $\mathbf{x}$ e $\mathbf{r}_{\mathbf{x}}$ respectivamente.

Para o caso de *ruídos multiplicativos*, nós consideramos

$$\tilde{\mathbf{x}} = \mathbf{x}. * \eta_{\mathbf{x}}, \quad \eta_{\mathbf{x}} = 2. \wedge (\varepsilon_{\mathbf{x}}(\mathbf{r}_{\mathbf{x}} - 0.5)). \tag{4.5}$$

Aqui,

$$\begin{array}{rcl} \mathbf{x}. * \mathbf{y} & := & (x_0 y_0, \ldots, x_{p-1} y_{p-1}), \\ a. \wedge \mathbf{x} & := & (a^{x_0}, \ldots, a^{x_{p-1}}). \end{array} \tag{4.6}$$

Para compararmos a quantidade de ruídos introduzidos no vetor $\mathbf{x}$, usaremos a métrica

$$d(\mathbf{x}, \tilde{\mathbf{x}}) = \frac{\|\mathbf{x} - \tilde{\mathbf{x}}\|}{\|\mathbf{x}\|}.$$

Em particular, quando $\mathbf{x} = \alpha_F$, denominaremos $1/d(\mathbf{x}, \tilde{\mathbf{x}})$ de SNR (sigla em inglês para designar *Signal to Noise Ratio*), i.e.,

$$\text{SNR} = \frac{\|\alpha_F\|}{\|\alpha_F - \tilde{\alpha}_F\|}. \tag{4.7}$$

Observemos que SNR é inversamente proporcional ao índice de ruído ε_{α_F}.

Na seção 4.3, nós apresentamos várias tabelas contendo resultados que nos permite saber se nossos métodos convergiram a mínimo global ou não. Mais especificamente, suponhamos que f seja o objeto que desejamos recuperar e $\tilde{f}$ a solução obtida por um dos nossos métodos. Como sabemos que, no caso 2-D, a solução é *quase sempre* única, então nós calculamos a distância, $\delta(f, \tilde{f})$, entre f e $\tilde{f}$, dentro do suporte, para avaliarmos quão próximo $\tilde{f}$ encontra-se do objeto original f. Pode ocorrer de $\tilde{f}$ estar mais próxima da gêmea, $\hat{f}$, de f do que da própria f. Neste caso faz-se necessário também calcularmos $\delta(\hat{f}, \tilde{f})$. Já no caso 1-D, tais cálculos não fazem sentido pois, como já esperávamos, os métodos convergiram na maior parte dos experimentos, a uma ambígua não trivial de f. Assim, para sabermos, no caso 1-D, se $\tilde{f}$ é ou não um mínimo global, nós calculamos $\|\tilde{f}_{(\mathbf{2})}\|$. Se tal norma for suficientemente pequena, então podemos dizer que $\tilde{f}$ é um mínimo global. Em todos os exemplos desta seção, os objetos originais são todos aleatórios. Resultados relativos a objetos não aleatórios, serão vistos na seção 4.6.

Experimentos numéricos para dados das amplitudes com ruídos aditivos serão abordados na subseção 4.3.3 e os para ruídos multiplicativos, na seção 4.6.

Na próxima seção descrevemos os tipos de condições iniciais que iremos utilizar em nossos experimentos numéricos.

4.1 Determinação das condições iniciais

As **condições iniciais aleatórias,** $\theta^{(0)}$, para o método MOFS serão construídas obedecendo-se ao seguinte critério:

- determina-se um objeto inicial aleatório qualquer $g^{(0)}$ que satisfaça a restrição de suporte;
- calcula-se o vetor fase, $\phi_{G^{(0)}}$, de sua transformada de Fourier, $G^{(0)}$;
- determina-se o vetor $\theta^{(0)}$ cujas componentes são dadas pelas fases intermediárias (veja definição nas seções 3.1.1 e 3.1.2) de $G^{(0)}$.

Também iremos considerar **condições iniciais obtidas por pequenas perturbações sobre a fase original**, i.e., pontos do tipo

$$\theta^{(0)} = \phi + \eta_\phi \tag{4.8}$$

onde ϕ é o vetor das fases intermediárias de F e η_ϕ é um ruído definido como em (4.3).

Outros tipos de **condições iniciais que usam informação sobre o as amplitudes de F com sinal** (veja [90] para maiores detalhes) também serão consideradas. Mais especificamente, sejam

$$F_u = \alpha_u \exp(i\phi_u)$$

a forma polar da DFT de f e $\mathrm{sgn}(F_u)$ o sinal de F_u. Note que como α_u é sempre positivo, então

$$\mathrm{sgn}(F_u) = \begin{cases} +1, & \text{se } -\pi/2 \leq \phi_u \leq \pi/2, \\ -1, & \text{caso contrário.} \end{cases} \tag{4.9}$$

Para as condições iniciais, $\theta^{(0)} = (\theta_u^{(0)})$, que fazem uso da informação de $\mathrm{sgn}(F_u)$, definimos

$$\theta_u^{(0)} = \begin{cases} 0 & \text{se } \mathrm{sgn}(F_u) = +1, \\ \pi & \text{se } \mathrm{sgn}(F_u) = -1. \end{cases} \tag{4.10}$$

Para determinarmos as condições iniciais dos métodos de Fienup, procedemos da seguinte forma: Primeiro consideramos a condição inicial, $\theta^{(0)}$, gerada para o método MOFS . Depois usamos esta mesma fase inicial para gerarmos a condição inicial, $f^{(0)}$, dos métodos de Fienup, fazendo

$$f^{(0)} = f(\alpha_c, \phi_c, \alpha, \theta^{(0)})^2. \tag{4.11}$$

[2]Lembremos que os vetores α_c, ϕ_c e α referem-se ao objeto que desejamos recuperar e são supostamente conhecidos.

Assim, podemos considerar que MOFS e os métodos de Fienup foram iniciados a partir de um mesmo ponto $f^{(0)}$.

4.2 Critérios para convergência

A convergência dos métodos ER, HIO e MOFS a mínimo global é, como já discutimos antes, determinada através da métrica $L(\theta) = (d \circ f)(\theta)$, ou seja, a norma 2 do objeto fora do suporte. De fato, se $\tilde{f} = f(\tilde{\phi})$ é um ponto de convergência obtido por algum desses métodos, então $\tilde{f}$ é, numericamente, um mínimo global se $L(\tilde{\phi})$ for suficientemente pequeno.

Entretanto, a *qualidade da convergência* desses métodos será avaliada em 1-D pelo próprio valor de $L(\theta)$ e no caso 2-D pela métrica δ definida em (2.15). No caso 2-D utilizamos a métrica δ para sabermos se numericamente o mínimo global encontrado é o próprio objeto original ou a sua gêmea. A razão para não adotarmos δ no caso 1-D é a falta de unicidade da solução.

4.3 Resultados numéricos para objetos aleatórios

4.3.1 Caso 1-D sem ruídos

Agora faremos alguns comentários sobre os resultados numéricos obtidos por ER, HIO e MOFS, no caso 1-D, para dados das amplitudes sem ruídos. Os resultados relativos a dados com ruídos serão discutidos no capítulo 5.

Diremos que o valor numérico de uma determinada variável real é da ordem de e$-n$, n inteiro, se esta variável assumir um valor dado na notação ponto flutuante por $0.a_1a_2a_3\ldots \times 10^{-n}$, com $a_1 \neq 0$. Por exemplo, dizemos que -0.12390×10^{-4} é da ordem de e-4. Daqui em diante omitiremos o algarismo 0 na representação ponto flutuante.

Seja $\tilde{f} = f(\tilde{\phi})$ um mínimo local[3] obtido por qualquer um dos métodos citados acima. Dada a representação em blocos, $\tilde{f} = (\tilde{f}_{(1)}, \tilde{f}_{(2)})^T$, consideremos o *erro*

$$e(\tilde{f}) := \|\tilde{f}_{(2)}\| \equiv \sqrt{2\, L(\tilde{\phi})}. \tag{4.12}$$

Em termos numéricos, podemos concluir que o método convergiu a um mínimo global, $\tilde{f}$, de $L(\theta)$ se $e(\tilde{f})$ for *suficientemente* pequeno. Mais precisamente,

> *Se $e(\tilde{f})$ for da ordem de* e-4 *ou menos, então $\tilde{f}$ é numericamente um ponto de mínimo global da função custo $L(\theta)$.*

[3]Conforme já registramos, no caso 1-D os algoritmos sempre convergem satisfatoriamente a mínimos locais

A Tabela 1 seguinte mostra os resultados obtidos após a execução dos três algoritmos, iniciados de um ponto aleatório, conforme descrição feita na seção 4.1. Os algoritmos iterativos foram executados, em média, com 20 mil iterações . Para o MOFS adotamos critérios de alta precisão de convergência ao atribuirmos **factr** = 1.0e+1 e **pgtol** = 1.0e-10 (veja (3.36) e (3.37)) como critérios de paradas. Em cada exemplo atribuímos diferentes valores para seed(f) e seed($\theta^{(0)}$). Como já esperávamos, nestes experimentos foi verificado em 100% dos exemplos testados que MOFS convergiu a algum mínimo global, com uma precisão do erro $e(\tilde{f})$ dado em média por valores da ordem de e-7. O ER também convergiu a mínimos globais, com precisão um pouco inferior à do MOFS em alguns exemplos. **Quanto ao HIO, sua convergência a mínimos globais foi verificada apenas em 60% dos exemplos.** Não encontramos na literatura nenhum registro sobre esse comportamento do HIO para o caso 1-D sem ruídos. Acredita-se que o HIO sempre converge na ausência de ruídos, porém este não é o caso para alguns exemplos que encontramos (veja Tabela 1). Também foi confirmado o que já sabíamos: todos os mínimos globais obtidos são flipping relacionadas (veja seção 1.3) com o sinal original f.

Tab.1 Determinação do erro obtido pela norma da estimativa $\tilde{f}$ avaliada fora do suporte. **Amplitudes sem ruídos.**
$\theta^{(0)}$ = Condição inicial aleatória

m	seed(f)	seed($\theta^{(0)}$)	$e(\tilde{f})\|_{MOFS}$	$e(\tilde{f})\|_{\text{ER}}$	$e(\tilde{f})\|_{\text{HIO}}$
8	123002	2	.18748e-6	.18748e-6	.18746e-6
8	3141592	2951413	.85968e-7*	.85964e-7*	.85964e-7*
8	1012	2101	.74095e-7*	.74074e-7*	.74074e-7*
8	7	908	.42373e-7*	.42371e-7*	.42371e-7*
8	1	2	.15527e-6	.15525e-6*	.15525e-6*
16	435	32	.62640e-7*	.62421e-7*	.69586e-1
16	9215143	295113	.86429e-7	.86027e-7	.72045e-2
16	2100	2101	.71904e-7•	.79504e-7•	.12066e-4•
32	709	908	.53796e-7	.23946e-5	.10921e-1
32	21709	12908	.57170e-7	.69481e-7	.54988e-7
64	71027	76908	.24268e-7	.56388e-4	.40088e-1
64	82038	87109	.24485e-7•	.49799e-7•	.70395e-4
64	93149	982110	.21555e-7•	.21356e-6•	.21508e-1
64	1042510	1093221	.15342e-7•	.90675e-7•	.15293e-1
128	2153	4327	.22501e-6	.59403e-4	.22739e-5

Para valores de $m = 8$, ER e HIO não necessitaram mais do que 3000 iterações para convergir a mínimo global. Entretanto à medida que o valor de m aumenta, torna-se necessário aumentar o número de iterações de ER e HIO para que tal convergência

ocorra. Para seed(f) = 71027, por exemplo, foram necessárias 60 mil iterações para cada um e, mesmo assim, HIO não conseguiu convergir a mínimo global. Também para seed(f) = 2153, ER precisou de 80 mil iterações para atingir um mínimo global com precisão de erro (e-4) abaixo da média obtida pelo MOFS (e-7).

A tabela acima mostra que MOFS, bem como o ER, convergiu a um mínimo global em todos os experimentos realizados. Também percebe-se que a precisão do erro do MOFS manteve-se sempre maior ou igual do que a do ER. Isto foi verificado não apenas nos exemplos da Tabela 1 como também em muitos outros exemplos que não foram aqui tabelados. O mesmo não aconteceu com o HIO que convergiu a mínimo local nos casos seed(f) = 435, 9215143, 709, 71027, 93149 e 1042510.

Finalmente comentemos os resultados assinalados na Tab. 1. Os números assinalados com $*$ indicam que os métodos correspondentes às colunas as quais eles pertencem convergiram a um mesmo mínimo global. Por exemplo, para seed(f) = 3141592, verificou-se que os três métodos convergiram a um mesmo mínimo global. Já para seed(f) = 1, ER e HIO convergiram a um mesmo mínimo global, diferente daquele atingido por MOFS. O símbolo $\bullet$ indica que os mínimos globais atingidos são diferentes, porém muito próximos um do outro. Por exemplo, para seed(f) = 82038, constatamos que $\delta(\tilde{f}|_{\text{MOFS}}, \tilde{f}|_{\text{ER}}) = .64341e-1$, onde δ é o erro definido por (2.15). Para seed(f) = 2100, os três métodos convergiram a mínimos globais, distantes um do outro de acordo com os erros

$$\begin{aligned} \delta(\tilde{f}|_{\text{MOFS}}, \tilde{f}|_{\text{ER}}) &= .42531e-1, \\ \delta(\tilde{f}|_{\text{MOFS}}, \tilde{f}|_{\text{HIO}}) &= .22997e-1, \\ \delta(\tilde{f}|_{\text{MOFS}}, \tilde{f}|_{\text{HIO}}) &= .36442e-1. \end{aligned}$$

Essas constatações nos faz acreditar na existência de muitos mínimos globais, que são relativamente próximos um do outro.

4.3.1.1 Comentários

Ao contrário do caso 2-D, onde acreditamos existirem muitos pontos de mínimos locais afastados dos mínimos globais, (que, como já sabemos, são únicos, quase sempre, a menos das ambiguidades triviais), no caso 1-D os pontos de mínimos locais parecem estar sempre muito próximos de algum mínimo global, ou caso exista algum mínimo local distante, os métodos não foram capazes de localizá-los. Isto pôde ser confirmado em vários experimentos. Para todos os exemplos da Tabela 1 em que HIO convergiu a mínimo local, $\tilde{f}$, verificou-se que tal mínimo é muito próximo à alguma flipping relacionada (veja definição na seção 1.3) do sinal original f. Assim, se não formos tão exigentes quanto à precisão do erro, podemos considerar estes mínimos locais como sendo, eles próprios, pontos de mínimos globais. É claro que neste caso estaríamos trabalhando com uma aproximação bastante grosseira.

Em nossos experimentos constatamos que MOFS e ER não apenas convergem sempre a mínimos globais como também não estagnam nos pontos localizados por

HIO . Além disso como já sabemos que o problema inverso da fase não admite solução única para o caso 1-D, é de se esperar que os mínimos globais encontrados pelos métodos nem sempre coincidam com o *mínimo global verdadeiro* (entenda-se por mínimo global verdadeiro o próprio objeto original f). De fato, foi verificado apenas em 17% dos casos rodados que os métodos convergiram à solução verdadeira do problema. Nos outros 83%, a convergência se deu a um mínimo global (ou a um ponto muito próximo a ele), correspondente a uma das 2^m possíveis soluções do problema inverso. Também constatou-se que nos casos onde houve convergência ao mínimo global verdadeiro, ela se deu apenas para valores de $m = 4$ e $m = 8$ (e, em alguns casos, para $m = 16$ porém, com uma frequência muito menor). Isto significa que à medida que crescemos o valor de m, maior se torna o número de mínimos globais (pois 2^m torna-se muito grande) e portanto menor a probabilidade de algum algoritmo convergir ao mínimo global verdadeiro.

Como dissemos anteriormente, em 100% dos casos rodados, independente do tipo de condição inicial utilizada, foi verificado que MOFS converge a mínimo global, podendo esse mínimo ser eventualmente o mínimo global verdadeiro, i.e., o sinal original f. Em outras palavras, dada uma condição inicial aleatória qualquer, verificou-se em todos os experimentos realizados, para diversos valores de m, que MOFS converge a um ponto, digamos $\tilde{\phi}$, de modo que $\tilde{f} \equiv f(\tilde{\phi})$ não apenas satisfaz a restrição de suporte mas também é flipping relacionada com f. Assim, para recuperarmos o sinal original, f, a partir de $\tilde{f}$, devemos flipar as raízes de $\tilde{f}$, sob todas as combinações possíveis, até que encontremos um polinômio $P_{F^{(*)}}(z)$ (veja (1.50)) tal que suas raízes sejam exatamente as mesmas raízes de $P_F(z)$. Então teremos $f^{(*)} = cf$, para alguma constante c, onde $f^{(*)} = \mathcal{F}^{-1}[F^{(*)}]$. Tal procedimento no entanto torna-se inviável para problemas maiores. Sua única utilidade é a de responder à nossa pergunta inicial sobre a possibilidade de se recuperar a fase da transformada de Fourier de um sinal original usando as amplitudes como os únicos dados conhecidos do problema.

4.3.2 Caso 2-D sem ruídos

Nesta subseção apresentamos algumas tabelas contendo resultados relativos aos dados das amplitudes sem ruídos. Testamos nossos métodos para diferentes tipos de condições iniciais (seção 4.1).

Seja $\tilde{f}$ o ponto obtido por um dos métodos, MOFS, ER ou HIO. Como dissemos na seção 4.2, analisaremos a convergência destes métodos, para o caso 2-D, através do erro $\delta(f, \tilde{f})$ ou $\delta(\hat{f}, \tilde{f})$, onde δ é a métrica em (2.15), para sabermos quão próximo $\tilde{f}$ encontra-se da imagem original f ou de sua gêmea

$$\hat{f} = [f_{m-1-x,m-1-y}].$$

Se $\delta(f, \tilde{f})$ (ou $\delta(\hat{f}, \tilde{f})$) for suficientemente próximo de zero, teremos então con-

vergência do método correspondente ao mínimo global. Mais precisamente, seja

$$\delta_{\tilde{f}} := \min\left\{\delta(f,\tilde{f}),\ \delta(\hat{f},\tilde{f})\right\}, \tag{4.13}$$

Se $\delta_{\tilde{f}}$ for da ordem de e-4 *ou menos*, então $\tilde{f}$ é numericamente um ponto de mínimo global da função $L(\theta)$.

Para condições iniciais aleatórias, $\theta^{(0)}$, verificou-se na maioria dos casos que MOFS e ER , como já dissemos, convergiram a mínimos locais; e HIO convergiu sempre a uma das soluções ambíguas triviais, f ou $\hat{f}$..

Entretanto se utilizamos alguma informação a respeito da fase original, ϕ, de F na formação da condição inicial, **a capacidade do MOFS em atingir o mínimo global aumenta consideravelmente, com um desempenho muito melhor do que o apresentado pelo ER, nas mesmas condições .** Usando por exemplo a informação das amplitudes de Fourier com sinal [90] na condição inicial (i.e., condições iniciais do tipo em (4.10)), verificou-se em 100% dos experimentos que MOFS convergiu à imagem original f enquanto que o algoritmo ER convergiu, na maioria das vezes, a pontos que, acreditamos, são mínimos locais. A mesma conclusão foi obtida para condições iniciais dadas por perturbações sobre a fase original.

4.3.2.1 Condições iniciais com a informação das amplitudes de Fourier com sinal

Na Tabela 2 apresentamos resultados numéricos relativos ao caso 2-D, para amplitudes sem ruídos. Diversos experimentos foram feitos considerando-se diferentes valores de m. Na maioria dos casos consideramos valores de $m = 8$ e $m = 16$, pelo fato de serem valores de grandeza razoável para a execução do MOFS e ao mesmo tempo por serem potências de 2, o que torna os cálculos para a transformada rápida de Fourier mais velozes.

A condição inicial, $\theta^{(0)}$, aqui utilizada é do tipo em (4.10). Como dissemos na seção 4.1, usamos o próprio $\theta^{(0)}$ como condição inicial para o método MOFS e $f^{(0)} = f(\alpha_c, \phi_c, \alpha, \theta^{(0)})$, onde α_c, ϕ_c e α são relativas à DFT da imagem original f, para ER e HIO. Executamos, então , cada um dos 3 algoritmos a partir da sua respectiva condição inicial para obtermos uma estimativa, $\tilde{f}$, da imagem original f ou de sua gêmea $\hat{f}$. Em seguida calculamos os erros, $\delta_{\tilde{f}}$, relativos aos 3 métodos MOFS, ER e HIO. Se $\tilde{f}$ estiver mais próximo da gêmea $\hat{f}$ do que da própria f, assinalamos à direita do valor do erro correspondente o símbolo asterisco $(*)$ conforme se vê na coluna do erro $\delta_{\tilde{f}}$(HIO) das Tabelas 2, 4 e 5. Nestes experimentos, HIO foi executado com 1000 iterações e valores de $\beta = 0.1$. ER foi executado com 10 mil iterações , o suficiente para garantir convergência. Para o MOFS adotamos critérios de média precisão de convergência: **factr**=1.0e+7 e **pgtol**=1.0e-7.

Tab.2: Erro no domínio do objeto. **Amplitudes sem ruídos.** Condição inicial com a informação das amplitudes de Fourier com sinal				
m	seed(f)	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)	$\delta_{\tilde{f}}$(HIO)
8	222	.41394e-5	.22827	.75070e-7
8	24	.92552e-5	.65245e-2	.60842e-5*
8	5432	.45133e-1	.10094e-6	.10082e-6
8	1504	.91240e-5	.23262	.10471e-6
16	40981	.69449e-5	.13596	.46895e-7
16	38160	.20855e-4	.16512	.84674e-5
16	8264	.62916e-5	.17775	.35376e-5
16	78345	.10422e-4	.18780	.12987e-5
16	438664	.10586e-4	.78117e-1	.14093e-4

Como se pode verificar na tabela acima, MOFS apresentou, exceto no caso seed(f) = 5432, melhor precisão do que ER . Isto mostra claramente a inferioridade do método ER em relação ao MOFS em atingir a solução verdadeira, nas condições apresentadas nesta subseção . De fato vê-se que ER , em 88% dos exemplos testados, convergiu a pontos de mínimo local, com uma precisão do erro na ordem de e+00. Como era de se esperar HIO convergiu a mínimo global em todos os casos testados, com erro dado por valores variando em torno de 1.0e-6.

4.3.2.2 Condições iniciais obtidas a partir de pequenas perturbações da fase original

Nesta subseção consideraremos condições iniciais dadas por

$$\theta^{(0)} = \phi + \eta_\phi, \quad \eta_\phi = \varepsilon_\phi \frac{\|\phi\|}{2(m^2-1)}(\mathbf{r}_\phi - 0.5), \tag{4.14}$$

quando

$$\varepsilon_\phi = 20.0.$$

Como antes, ϕ representa a fase original de Fourier.

Na Tabela 3, nós apresentamos os erros para diferentes valores atribuídos ao par (seed(f), seed(η_ϕ)). Novamente, consideramos os valores das amplitudes de Fourier livres de ruídos.

Aqui ambos, HIO e ER, foram tomados com 10 mil iterações . Novamente o parâmetro do HIO, β, foi tomado igual a 0.1 e os critérios de parada para o MOFS foram **factr**=1.0e+7 e **pgtol**=1.0e-7.

Tab.3: Erro no domínio do objeto. **Amplitudes sem ruídos**. Condição inicial obtida por pequenas perturbações na fase original, com $\varepsilon_\phi = 20.0$

m	seed(f)	seed(η_ϕ)	δ_f(MOFS)	δ_f(ER)	δ_f(HIO)
8	222	333	.60572e-5	.39591	.75070e-7
8	24	556	.13166e-4	.17265	.11300e-6
8	5432	10923	.66524e-5	.10094e-6	.10094e-6
8	1504	4051	.55012e-5	.10236	.10471e-6
16	40981	900432	.65597e-5	.14193	.47184e-7
16	38160	332109	.37221e-4	.11206	.51765e-7
16	8264	31090	.45873e-5	.26786e-1	.42022e-7
16	78345	521045	.75564e-5	.17406e-1	.45314e-7
16	438664	63290	.83946e-5	.10356e-1	.47368e-7

Consultando a Tabela 3, vemos novamente a **superioridade do MOFS em relação ao ER em atingir o mínimo global.**

4.3.3 Caso 2-D com ruídos aditivos

Nesta subseção as tabelas trazem resultados de experimentos que usam dados das amplitudes com ruídos relativas à transformada de Fourier de objetos aleatórios, cujos valores de *seed* serão os mesmos da Tabela 3.

Para introduzir ruídos nas amplitudes, usamos as fórmulas (4.2) e (4.3) quando $\mathbf{x} = \alpha_F$, i.e.,

$$\tilde{\alpha}_F = \alpha_F + \eta_{\alpha_F}, \quad \eta_{\alpha_F} = \varepsilon_{\alpha_F} \frac{\|\alpha_F\|}{2(m^2+1)} (\mathbf{r}_{\alpha_F} - 0.5).$$

Em todos os experimentos desta subseção e nos da Tabela 7 da seção 4.4, trabalharemos com um valor fixo para ε_{α_F}, dado por

$$\varepsilon_{\alpha_F} = 0.9.$$

Observa-se nas Tabelas 4 e 5 que, mesmo na presença de ruídos, MOFS continua apresentando bons resultados, convergindo sempre a pontos bem próximos da imagem original. ER , nestas mesmas condições , continua, no entanto, convergindo a pontos afastados da solução verdadeira. HIO , como era de se esperar, apresentou um comportamento completamente instável, atingindo, a cada ciclo de iterações , um ponto diferente, não apresentando, portanto, nenhum comportamento previsível que possa caracterizar convergência.

4.3.3.1 Condições iniciais com a informação das amplitudes de Fourier com sinal

Nesta subseção , nossos algoritmos se iniciam de condições do tipo em (4.10). Aqui, ER foi tomado com 10 mil iterações . Novamente o parâmetro do HIO, β,

foi tomado igual a 0.1 e os critérios de parada para o MOFS foram os de média precisão, dados por **factr**=1.0e+7 e **pgtol**=1.0e-7.

Tab.4: Erro no domínio do objeto. **Amplitudes com ruídos aditivos.** $\varepsilon_{\alpha_F} = 0.9$
Condição inicial com a informação das amplitudes de Fourier com sinal

m	seed(f)	seed(η_{α_F})	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)	$\delta_{\tilde{f}}$(HIO)
8	222	55	.35016e-1	.35017e-1	.45349/.43514 .069153*/.12512* .11684*/.51489*
8	24	60	.46754e-1	.68488e-1	.19614/.21696* .30455/.37859 .21184/.20814
8	5432	98006	.57707e-1	.37836e-1	.47720*/.10915* .28758/.21969 .14828/.40653
8	1504	500	.34288e-1	.23368	.46331/.29630 .48019*/.34356* .44236/.48557
16	40981	3312	.18175e-1	.14076	.095648/.11124 .18145/.12743 .30828/.12535
16	38160	149988	.17999e-1	.16829	.26463/.18645 .18398/.14646 .21951/.11463
16	8264	2918	.16848e-1	.18226	.10319/.33318 .10765/.19491 .19082/.26840
16	78345	354387	.18514e-1	.18609	.063314/.26998 .26077*/.11115* .16898*/.066751*
16	438664	46683	.17119e-1	.79892e-1	.17242/.14012 .20473/.18148 .20000/.26979

Para mostrarmos o comportamento de instabilidade do HIO , nós o executamos em 6 ciclos de iterações , cada ciclo consistindo de um número de iterações dado respectivamente por 3000, 2000, 2000, 1000, 1000 e 2000. Estas 6 execuções foram feitas de modo que o ponto obtido após a execução do algoritmo para um determinado ciclo seja o ponto de partida para o próximo ciclo. Os valores de $\delta_{\tilde{f}}$(HIO) obtidos após cada ciclo estão na última coluna da Tabela 4. Como já dissemos, o símbolo asterisco (*) que aparece à direita do erro indica o valor da distância δ entre o ponto atingido pelo algoritmo e a gêmea $\hat{f}$ de f. Por exemplo, para seed(f) = 222, HIO convergiu, após o primeiro ciclo de iterações ($K = 3000$), a um ponto $\tilde{f}$, distante de f por $\delta(f, \tilde{f}) = .45349$. A partir deste ponto, $\tilde{f}$, após o segundo ciclo (2000

iterações), ele convergiu a outro ponto, $\tilde{\tilde{f}}$, com erro $\delta(f, \tilde{\tilde{f}}) = .43514$. No terceiro ciclo, usando $\tilde{\tilde{f}}$ como condição inicial e aplicando mais 2000 iterações , HIO convergiu próximo à gêmea, $\hat{f}$, da imagem original, com erro dado por .069153, e assim sucessivamente.

Note que MOFS apresentou melhor desempenho do que os métodos ER e HIO. Seu desempenho foi melhor do que ER por ter convergido bem mais próximo ao mínimo global (com erro da ordem de e-1, em 100% dos exemplos testados) do que ER (com erro da ordem de e+00 em 55% dos casos). Como veremos nos exemplos da seção 4.6, métodos que apresentam erros da ordem de e-1, produzem imagens mais nítidas do que os que apresentam erros da ordem de e+00.

Relativamente ao HIO , MOFS também sai ganhando. Enquanto MOFS apresentou um comportamento estável, convergindo sempre próximo à solução verdadeira, HIO , como já previamos, ficou oscilando em torno de vários pontos diferentes entre si, apresentando portanto um comportamento de não convergência. Mais precisamente, dado um número de iterações, não é possível prever em qual região do domínio do objeto o algoritmo irá parar. Nesse sentido, Tanto MOFS quanto ER são métodos *robustos*, enquanto HIO é altamente sensível a ruídos.

4.3.3.2 Condições iniciais obtidas a partir de pequenas perturbações da fase original

Nesta subseção consideramos condições iniciais dadas por (4.14), com $\varepsilon_\phi = 20.0$. Consideraremos os mesmos valores atribuídos ao par $(\text{seed}(f), \text{seed}(\eta_\phi))$ na Tab. 3. Os valores de $\text{seed}(\eta_{\alpha_F})$ serão os mesmos dos da Tabela 4.

Novamente executamos ER com 10 mil iterações, adotamos critérios de parada de média precisão para o MOFS e executamos HIO em 6 ciclos, cada ciclo com o mesmo número de iterações considerados para a Tabela 4. Lembremos que estamos sempre considerando $\beta = 0.1$.

Tab.5: Erro no domínio do objeto **Amplitudes com ruídos aditivos**
Condição inicial obtida por pequenas perturbações na fase original.
$\varepsilon_{\alpha_F} = 0.9$ e $\varepsilon_\phi = 20.0$

m	seed(f)	seed(η_{α_F})	seed(η_ϕ)	$\delta_{\bar{f}}$(MOFS)	$\delta_{\bar{f}}$(ER)	$\delta_{\bar{f}}$(HIO)
8	222	55	333	.35018e-1	.38299	.22093/.087711* .14262/.10955 .073312*/.41672
8	24	60	556	.46763e-1	.18320	.32970*/.22479* .19420*/.19070 .17660/.071733*
8	5432	98006	10923	.37833e-1	.37836e-1	.12941/.26180 .15170/.14473 .40735*/0.10274*
8	1504	500	4051	.34289e-1	.11342	.44425/.29666* .18964/.44659* .41957*/.28710*
16	40981	3312	900432	.18175e-1	.14283	.10949/.13891 .23469/.10018 .13592/.19146
16	38160	149988	332109	.18002e-1	.11260	.090347/.15733 .16875/.14230 .17446/.20990
16	8264	2918	31090	.16849e-1	.33924e-1	.16852/.24797 .19231/.17347 .34208/.36239
16	78345	354387	521045	.18512e-1	.23937e-1	.061509/.11270 .44446/.087246* .34783*/.12002*
16	438664	46683	63290	.17118e-1	.19408e-1	.37742/.23963 .14255/.21247 .27630/.23207

Novamente MOFS apresentou melhor desempenho do que os métodos ER e HIO.

4.4 Distância Máxima (D_M) para convergência do MOFS e ER

Vimos que para condições iniciais obtidas por perturbação da fase original,

$$\theta^{(0)} = \phi + \eta_\phi, \quad \eta_\phi = \varepsilon_\phi \frac{\|\phi\|}{2(m^2-1)}(\mathbf{r}_\phi - 0.5), \tag{4.15}$$

e escolha de

$$\varepsilon_\phi = 20.0,$$

MOFS apresentou boa convergência, ao mínimo global, mesmo na presença de ruídos nos dados das amplitudes, situação em que o erro foi da ordem de e-1. ER também teve o mesmo desempenho, porém em apenas 44% dos exemplos testados. Uma questão natural que surge, então, é a seguinte: qual a *distância máxima* que a fase perturbada deve se encontrar da fase original para garantirmos a convergência do MOFS (respectivamente do ER) a um ponto suficientemente próximo do mínimo global? Será que estas distâncias são as mesmas para MOFS e ER? Para respondermos a estas perguntas devemos analisar a distância entre a fase perturbada, $\theta^{(0)}$, e a fase original, ϕ, dada pela norma relativa

$$N_R(\varepsilon_\phi) := \frac{\|\theta^{(0)} - \phi\|}{\|\phi\|} = \frac{\|\eta_\phi\|}{\|\phi\|} = \frac{\varepsilon_\phi}{2(m^2-1)}\|\mathbf{r}_\phi - 0.5\|. \tag{4.16}$$

A expressão (4.15) mostra que $\theta^{(0)}$ depende da escolha de ε_ϕ, ou seja, $\theta^{(0)} = \theta^{(0)}(\varepsilon_\phi)$. O valor da distância máxima para a convergência de um determinado método, digamos MOFS, é obtido atribuindo-se diferentes valores para ε_ϕ em (4.16), até obtermos um valor limite, $D_M = N_R(\varepsilon^*)$, tal que se $N_R(\varepsilon_\phi) > D_M$, o método deixa de convergir com a precisão de erro desejada. Assim, para determinarmos D_M nós iniciamos ε_ϕ com um valor baixo, digamos $\varepsilon_\phi = \varepsilon_0$, que garanta a convergência do método e, então, aumentamos seu valor gradativamente, de modo a obtermos uma sequência de índices $\varepsilon_0, \varepsilon_1, \ldots, \varepsilon_k, \ldots$. Em cada passo k, calculamos, então, o valor de $N_R(\varepsilon_k)$ e o erro $\delta_{\tilde{f}}(\text{MOFS})(\theta_k^{(0)})$, obtido após aplicarmos MOFS sobre a condição inicial

$$\theta_k^{(0)} := \theta^{(0)}(\varepsilon_k).$$

Os valores de $\delta_{\tilde{f}}(\text{MOFS})(\theta_k^{(0)})$ vão aumentando à medida que ε_k também vai crescendo. O processo é interrompido quando encontrarmos ε_k e ε_{k+1} suficientemente próximos um do outro, tais que $\delta_{\tilde{f}}(\text{MOFS})(\theta_k^{(0)})$ é de ordem e-N, ou menos, e $\delta_{\tilde{f}}(\text{MOFS})(\theta_{k+1}^{(0)})$ de ordem maior ou igual a e-(N-1), onde N é um inteiro previamente estabelecido[4]. Então definimos

$$D_M = D_M|_{\text{MOFS}} = \delta_{\tilde{f}}(\text{MOFS})(\theta_k^{(0)}).$$

Adotaremos $N = 1$. A explicação de tal escolha para a determinação de D_M é a de que mínimos locais que apresentam erro de convergência da ordem de e-1, ainda apresenta boa nitidez de imagem, em relação ao objeto original.

Assim, para cada imagem original f dada aleatoriamente, obtemos uma estimativa, $D_M|_{\text{MET}}$ (MET = MOFS ou ER), do valor máximo da distância que a condição inicial $\theta^{(0)}$ deve estar da fase original ϕ, para garantir a convergência de MET a um ponto próximo da imagem original, com uma precisão da ordem de e-1.

[4]Em problemas sem ruídos, adotamos $N = 1$. Em problemas com ruídos, o valor de N depende da quantidade de ruídos introduzidos nas amplitudes, i.e., do valor do SNR. Para problemas com ruídos aditivos pequenos tais que SNR $\geq$ 33.333, adotamos $N = 1$.

Se $\theta^{(0)}$ estiver a uma distância de ϕ maior do que $D_M|_{\text{MET}}$, então MET convergirá a um mínimo local, distante da imagem original, por um valor de ordem igual ou superior a e+00.

É claro que os resultados que apresentaremos para esta distância máxima, nas tabelas a seguir, são apenas uma aproximação grosseira do valor exato. Entretanto será possível determinarmos o intervalo que contém o valor exato desta distância máxima.

Na Tabela 6 indicamos alguns valores de ε_ϕ para acharmos uma estimativa da distância máxima para os casos seed$(f) = 5432$, 40981, 78345 e 438664, e amplitudes sem ruídos.

Na Tabela 7 consideramos os mesmos casos da Tabela 5, para amplitudes com ruídos aditivos.

Nestas tabelas, o valores de $D_M|_{\text{MOFS}}$ e $D_M|_{\text{ER}}$ aparecem em negrito, na coluna correspondente ao seus respectivos métodos. Também, em negrito, aparecem os valores dos erros $\delta_{\tilde{f}}$(MOFS) e $\delta_{\tilde{f}}$(ER), quando iniciamos estes métodos a partir de pontos que estão à sua distância máxima de afastamento da fase original.

Para entendermos melhor a leitura destas Tabelas, tomemos o primeiro exemplo da Tabela 6, correspondente a $m = 8$, seed$(f) = 5432$. Seja a condição inicial dada, de acordo com (4.15), por

$$\theta^{(0)}(\varepsilon_\phi) = \phi + \eta_\phi = \phi + \varepsilon_\phi \frac{\|\phi\|}{126}(\mathbf{r}_\phi - 0.5). \tag{4.17}$$

para seed(η_ϕ) = seed$(\mathbf{r}_\phi) = 10923$. Fazendo $\varepsilon_\phi = 27$ em (4.17), obtemos a condição inicial $\theta^{(0)}(27)$, distante $N_R = 0.69526$ unidade da fase original ϕ. Iniciando-se de $\theta^{(0)}(27)$, MOFS converge para $\tilde{f} \simeq f$, com erro $\delta(f, \tilde{f}) = 0.53787$e-5. O algoritmo ER, iniciado do mesmo ponto, $\theta^{(0)}(27)$, converge a $\tilde{\tilde{f}} \simeq f$, com erro $\delta(f, \tilde{\tilde{f}})$ = 0.43970e-1. À medida que aumentamos os valores de ε_ϕ sucessivamente para 28, 32.08, 32.089 e 32.0899, o erro $\delta_{\tilde{f}}$(ER) aumenta para 0.13326 e depois permanece em 0.18788. Com isso obtemos a aproximação $D_M|_{\text{ER}}$ = 0.43970e-1. Note que MOFS converge com erro da ordem de e-5 para as condições iniciais

$$\theta^{(0)}(27),\ \theta^{(0)}(28),\ \theta^{(0)}(32.08),\ \theta^{(0)}(32.089)$$

e com erro da ordem de e+00 para $\theta^{(0)}(32.0899)$. Logo, de acordo com a Tabela 6, $D_M|_{\text{MOFS}}$ = 0.82630.

Em todos os exemplos das tabelas 6 e 7, executamos ER com 20 mil iterações e adotamos critérios de parada de média precisão , **factr** = 1.0e+7, **pgtol** = 1.0e-7 para MOFS.

Tab.6: Distância Máxima (D_M) para a convergência dos algoritmos MOFS e ER.
Amplitudes sem ruídos.

Caso $m = 8$; seed(f) = 5432, seed(η_ϕ) = 10923			
ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
27	**.69526**	.53787e-5	**.43970e-1**
28	.72101	.57722e-5	.13326
32.08	.82607	.68213e-5	.18788
32.089	**.82630**	**.67059e-5**	.18788
32.0899	.82632	.28866	.18788
Caso $m = 16$; seed(f) = 40981, seed(η_ϕ) = 900432			
ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
8.988	**.11889**	.18944e-4	**.47184e-7**
8.989	.13213	.93104e-5	.14124
40	.52911	.65480e-5	.16031
50	**.66138**	**.33714e-1**	.27491
55	0.72752	.17936	.32070
Caso $m = 16$; seed(f) = 78345, seed(η_ϕ) = 521045			
ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
26	**.32959**	.12773e-4	**.83473e-1**
26.5	.33592	.80695e-5	.12909
46	.58311	.60245e-5	.22805
47	**.59579**	**.72654e-1**	.22805
50	.63382	.13949	.22805
Caso $m = 16$; seed(f) = 438664, seed(η_ϕ) = 63290			
ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
45.35	**.59291**	.46175e-5	**.46054e-1**
45.4	.59357	.77378e-5	.14383
51	.66678	.19217e-4	.19590
52	**.67986**	**.27180e-1**	.19438
52.5	.68639	.27178	.36976

Os valores de $D_M|_{\text{MOFS}}$ e $D_M|_{\text{ER}}$ aparecem em negrito na coluna N_R.
Os valor em negrito que aparece nas colunas $\delta_{\tilde{f}}$(MOFS) e $\delta_{\tilde{f}}$(ER) representa o erro do respectivo método, cuja condição inicial dista da fase original pelo valor correspondente do D_M encontrado

Tab.7: Distância Máxima (D_M) para a convergência dos algoritmos MOFS e ER.
Amplitudes com ruídos aditivos. $\varepsilon_{\alpha_F} = 0.9$.

$m = 8$; seed$(f) = 5432$, seed$(\eta_{\alpha_F}) = 98006$
SNR= 43.375, seed$(\eta_\phi) = 10923$

ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
27.3	**.70298**	.37835e-1	**.57708e-1**
27.4	.70556	.37837e-1	.12664
32.4	.83431	.37835e-1	.18958
32.5	**.83688**	**.37835e-1**	.18958
32.6	.83946	.29315	.18958

$m = 16$; seed$(f) = 40981$, seed$(\eta_{\alpha_F}) = 3312$
SNR= 88.013, seed$(\eta_\phi) = 900432$

ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
8.987	**.11888**	.18176e-1	**.18915e-1**
8.988	.11889	.18176e-1	.14213
50	**.66138**	**.19146e-1**	.27511
51	0.67461	.17023	.27511

$m = 16$; seed$(f) = 78345$, seed$(\eta_{\alpha_F}) = 354387$
SNR= 91.191, seed$(\eta_\phi) = 521045$

ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
25	**.31691**	.18516e-1	**.85516e-1**
26	.32959	.18515e-1	.12971
48	**.60847**	**.74508e-1**	.22474
49	.62114	.11296	.22474

$m = 16$; seed$(f) = 438664$, seed$(\eta_{\alpha_F}) = 46683$
SNR= 88.222, seed$(\eta_\phi) = 63290$

ε_ϕ	N_R	$\delta_{\tilde{f}}$(MOFS)	$\delta_{\tilde{f}}$(ER)
46	**.60141**	.17119e-1	**.45464e-1**
47	.61449	.17112e-1	.13975
54	**.70600**	**.88354e-1**	.33932
55	.71908	.18155	.33932

Como se pode observar nas Tabelas 6 e 7, $D_M|_{\text{MOFS}}$ é maior do que $D_M|_{\text{ER}}$ em todos os casos. Isto aconteceu também em muitos outros exemplos que não foram aqui tabelados. Na seção 4.6, nós ilustramos esse fenômeno com imagens não aleatórias. **Esta é, pois, mais uma vantagem do MOFS sobre o ER.**

4.5 Caso 2-D - imagens aleatórias: conclusões

Foi observado de um modo geral que, na ausência de ruídos nos dados das amplitudes e utilizando condições iniciais que utilizam determinadas informações da fase original, o método MOFS mostrou-se mais eficaz do que ER em localizar os mínimos globais. Em todas os experimentos realizados, MOFS convergiu sempre à solução verdadeira f do problema inverso. ER , ao contrário do MOFS, convergiu, na maioria dos casos a pontos, muitas vezes distantes de f.

Outra observação a ser feita é que, ao contrário do HIO , que muitas vezes, na ausência de ruídos, convergiu à gêmea da imagem original f, o MOFS, nas vezes em que convergiu a mínimo global, nunca convergiu à nenhuma gêmea, bem como a nenhuma das ambíguas triviais, exceto à própria imagem original. Isto se justifica pelo fato de MOFS ter sido executado sempre sobre o mesmo Toróide, T_{ϕ_c}, da imagem original

Conforme constatamos em nossos experimentos, MOFS, na ausência de ruídos, sempre converge ao mínimo global verdadeiro quando iniciado de um ponto suficientemente próximo a ele. Quanto mais afastados estivermos do mínimo global, mais afastado dele estará o ponto de convergência obtido pelo método. A Tabela 6 exibe a distância máxima de segurança que devemos estar da solução para que ela seja atingida após a execução do método. Foi considerada nesta tabela a situação em que os dados das amplitudes não são alterados por ruídos. Note que não fizemos nesta tabela nenhuma comparação do MOFS com o HIO porque, nesta situação, como já dissemos, HIO não precisa necessariamente de condições iniciais próximas do mínimo global para convergir a tal mínimo. Ele sempre atinge o mínimo global, independente da condição inicial utilizada. É claro que temos que enfrentar o problema dos pontos de estagnação. Mesmo assim, tal problema pode ser solucionado, atribuindo-se diferentes valores aleatórios às condições iniciais, ou fazendo combinações de vários ciclos do algoritmo com o ER . Fazendo isso, podemos vencer os problemas de estagnação do HIO e garantir sempre a convergência do método ao mínimo global, o que não acontece com o MOFS, o qual depende de informações adicionais da fase original (uma situação que não se aplica para fins práticos, uma vez que na maioria dos problemas de aplicação, da recuperação da fase a partir das amplitudes, não se tem disponível informações da fase original) para garantir a convergência ao mínimo global. Isto faz do HIO , na ausência de ruídos, um método mais eficiente do que o MOFS para buscar as soluções.

Entretanto, ao introduzirmos ruídos nos dados das amplitudes, HIO perde essa posição por apresentar um comportamento completamente instável. Mesmo iniciando-se próximo da solução , vemos pela Tabela 5 que HIO não converge, oscilando sempre em torno de vários pontos, como se pode verificar pelos resultados do erro δ_f(MOFS) na última coluna desta Tabela. Já o método MOFS mostrou-se bastante estável na presença de ruídos, no sentido de continuar convergindo muito perto do

mínimo global, dependendo, é claro, do valor da distância máxima. A Tabela 7 exibe a distância máxima de segurança que devemos estar da solução, na situação em que temos ruídos nas amplitudes, para que ela seja atingida após a execução do método.

Ao contrário do MOFS, ER , na maioria dos exemplos rodados, não foi capaz de localizar a solução global, quando iniciado numa vizinhança dela.

Nieto-Vesperinas, em seu método de otimização [67] (veja seção 2.4.1), constatou que a não -linearidade e o número de mínimos locais da função custo, que ele definiu para se aplicar o método, crescem dramaticamente com o tamanho do objeto, tornando o seu método não prático para imagens maiores que 6×6. Além disso, o número de mínimos locais próximos do mínimo global, que é a solução do problema, também torna-se extremamente grande com o tamanho das imagens, tornando o seu método ineficaz para localizar a solução, mesmo quando iniciado próximo dela. Assim, podemos concluir que nas condições de ausência de ruídos nos dados das amplitudes e de condições iniciais próximas da solução global, MOFS mostrou-se mais eficaz do que o ER e o método de Nieto-Vesperinas, pois enquanto estes métodos, nestas condições , atingem, na maioria das vezes, os mínimos locais próximos à solução verdadeira, o método MOFS vai direto à própria solução. Mesmo na presença de ruído, MOFS converge satisfatoriamente próxima à solução verdadeira, como veremos em alguns exemplos na próxima seção.

4.6 Resultados numéricos para objetos não aleatórios - caso 2-D - ruídos multiplicativos

Nesta seção trabalharemos com imagens não aleatórias. Consideraremos apenas o caso de amplitudes com ruídos multiplicativos, i.e.,

$$\tilde{\alpha}_F = \alpha_F .* \eta_{\alpha_F}, \quad \eta_{\alpha_F} = 2. \wedge (\varepsilon_{\alpha_F}(\mathbf{r}_{\alpha_F} - 0.5)), \tag{4.18}$$

onde $.*$ e $.\wedge$ são as operações definidas em (4.6). Fixaremos para $\text{seed}(\eta_{\alpha_F})$ o seguinte valor

$$\text{seed}(\eta_{\alpha_F}) = \text{seed}(\mathbf{r}_{\alpha_F}) = 24.$$

Também avaliaremos o nível de ruído nas amplitudes através do

$$\text{SNR} = \frac{\|\alpha_F\|}{\|\alpha_F - \tilde{\alpha}_F\|}.$$

Para condições iniciais aleatórias , $\theta^{(0)}$, assumiremos sempre

$$\text{seed}(\theta^{(0)}) = 25.$$

Para condições obtidas por perturbações da fase original,

$$\theta^{(0)} = \phi + \eta_\phi, \quad \eta_\phi = \varepsilon_\phi \frac{\|\phi\|}{2(m^2+1)}(\mathbf{r}_\phi - 0.5), \tag{4.19}$$

o seed(η_ϕ) também será sempre o mesmo, dado por

$$\text{seed}(\eta_\phi) = \text{seed}(\mathbf{r}_\phi) = 25.$$

Nos experimentos desta seção , exibiremos resultados obtidos pelos seguintes métodos: MOFS, ER, EH e MHIO. Antes de comentar os métodos EH e MHIO, convém ressaltar que, como agora nossas amplitudes são supostamente corrompidas de ruídos, a imagem inicial, $f^{(0)}$, deverá ser construída a partir de $\tilde{\alpha}_F$ e não de α_F. Ou seja,

$$f^{(0)} = f(\tilde{\alpha}_c, \phi_c, \tilde{\alpha}, \theta^{(0)}),$$

onde $\tilde{\alpha}_c$ são as amplitudes dos cantos e $\tilde{\alpha}$ as amplitudes intermediárias que compõem o vetor $\tilde{\alpha}_F$.

1. O método EH, proposto por Fienup e já introduzido na seção 2.3, é, como sabemos, uma combinação de ER com HIO, consistindo de vários ciclos de iterações, onde um ciclo consiste de K_1 iteracoes do HIO seguido de K_2 iterações do ER. Em nossos experimentos fixaremos $K_2 = 1000$. Para o número de iterações , K_1, do HIO adotaremos critério similar ao utilizado para o método MHIO em (4.20). Também para o HIO usaremos o parâmetro $\beta = 0.1$. O número de ciclos vai depender de cada exemplo, até que se obtenha uma convergência mais próxima possível da solução verdadeira. O método é ilustrado pelo seguinte diagrama

$$f^{(0)} \xrightarrow{\boxed{\text{ER}}} \tilde{f}^{(1)} \xrightarrow{\boxed{\text{HIO}}} \tilde{f}^{(2)} \xrightarrow{\boxed{\text{ER}}} \tilde{f}^{(3)} \xrightarrow{\boxed{\text{HIO}}} \cdots\cdots \xrightarrow{\boxed{\text{ER}}} \tilde{f}.$$

Figura 4.1: Diagrama do método EH

2. O método MHIO será, similarmente ao EH, uma combinação do MOFS com HIO. Aqui, um ciclo de MHIO consiste de K iterações do HIO (novamente consideramos $\beta = 0.1$) seguido da execução do MOFS, munido com critérios de parada de baixíssima precisão :

$$\textbf{factr} = 1.\text{e+}13; \quad \textbf{pgtol} = 1.\text{e-}4.$$

Assim, O k-ésimo ciclo de MHIO, $k = 1, 2, \ldots$, se inicia com a execução do MOFS, tendo a imagem $\tilde{f}^{(2k-2)}$ como ponto de partida e a imagem $\tilde{f}^{(2k-1)}$ como ponto de chegada, e termina com a execução do HIO, com K iterações, tendo $\tilde{f}^{(2k-1)}$ como ponto de partida e $\tilde{f}^{(2k)}$ como ponto de chegada. Por definição, $\tilde{f}^{(0)} = f^{(0)}$.

O número, K, de iterações para o HIO dependerá do valor de

$$\delta^{(k)}(\text{MOFS }) := \min\left\{\delta(f, \tilde{f}^{(2k-1)}),\ \delta(\hat{f}, \tilde{f}^{(2k-1)})\right\},$$

onde $f = f(\alpha_c, \phi_c, \alpha, \phi)$, é a imagem original e $\hat{f}$, a sua gêmea, conforme a seguinte convenção :

$$\begin{array}{lll} 0.00 \leq \delta^{(k)}(\text{MOFS }) \leq 0.09 & \Longrightarrow & K = 50, \\ 0.09 < \delta^{(k)}(\text{MOFS }) \leq 0.24 & \Longrightarrow & K = 100, \\ 0.24 < \delta^{(k)}(\text{MOFS }) \leq 0.34 & \Longrightarrow & K = 200, \\ 0.34 < \delta^{(k)}(\text{MOFS }) \leq 0.44 & \Longrightarrow & K = 300, \quad \text{etc.} \end{array} \tag{4.20}$$

O critério adotado em (4.20) não deve ser rigoroso, por isso, salvo menção em contrário, ele será sempre adotado para os métodos MHIO e EH (neste último caso, desde que substituamos $\delta^{(k)}(\text{MOFS })$ por $\delta^{(k)}(\text{ER })$ em (4.20)).

$$\ldots\ldots \tilde{f}^{(2k-2)} \xrightarrow{\boxed{\text{MOFS}}} \tilde{f}^{(2k-1)} \xrightarrow{\boxed{\text{HIO}}} \tilde{f}^{(2k)} \ldots\ldots$$

Figura 4.2: k-ésimo ciclo do MHIO

Uma pergunta que surge naturalmente é sobre quais os Toróides onde o método MOFS deverá ser executado. Inicialmente, no primeiro ciclo, executamos MOFS sobre o mesmo Toróide que contém a imagem original, i.e, aplicamos o método para resolver o problema

$$\tilde{\theta} = \text{argmin}_\theta L(\tilde{\alpha}_c, \phi_c, \tilde{\alpha}, \theta), \quad \theta^{(0)} = \text{condição inicial.}$$

Em seguida aplicamos HIO usando $\tilde{f}^{(1)}$ como condição inicial. Como, em cada iteração , HIO altera todas as fases das DFT's do ponto em curso da sequência que ele gera, o ponto de chegada, $\tilde{f}^{(2)}$, do HIO não estará necessariamente sobre o mesmo Toróide da imagem inicial $\tilde{f}^{(1)}$. Esta propriedade é decorrente da maneira como HIO foi definido (veja Passos 1,2,3 de (2.1) e (2.2)). ER Também se comporta desta maneira. Assim, ao fim de cada execução do HIO, não sabemos sobre qual dos 16 possiveis Toróides ele estará. Consideremos então o k-ésimo ciclo do MHIO e suponhamos que ele tenha convergido ao ponto $\tilde{f}^{(2k)}$. Seja $\tilde{F}^{(2k)} = \mathcal{F}(\tilde{f}^{(2k)})$. Então $\tilde{F}^{(2k)} = F(\tilde{\alpha}_c, \tilde{\phi}_c^{(2k)}, \tilde{\alpha}, \theta^{(2k)})$, onde $\tilde{\phi}_c^{(2k)}$ e $\theta^{(2k)}$ são respectivamente as fases dos cantos e as fases intermediárias (veja definições na subseção 3.1.2) de $\tilde{F}^{(2k)}$. Então vemos que $\tilde{f}^{(2k)}$ está sobre o Toróide $T_{\tilde{\phi}_c^{(2k)}}$. Logo, no próximo ciclo, MOFS

deverá ser executado sobre este Toróide ou, equivalentemente, deveremos aplicar o método L.BFGS.B para resolver o problema

$$\tilde{\theta} = \text{argmin}_{\theta} L(\tilde{\alpha}_c, \tilde{\phi}_c^{(2k)}, \tilde{\alpha}, \theta), \quad \theta^{(2k)} = \text{condição inicial.}$$

Finalmente, estabeleceremos critérios para a obtenção do ponto de convergência do método MHIO. Nosso procedimento será o seguinte: aplicamos um número de ciclos, digamos M, que seja suficiente para atingirmos uma boa estimativa, $\tilde{f}$, para f. Esta estimativa será controlada pelo erro

$$\delta_{\tilde{f}}(\text{MHIO }) := \min\left\{\delta(f, \tilde{f}),\ \delta(\hat{f}, \tilde{f})\right\}^{5}.$$

Após a execução dos M ciclos, obtemos o ponto $\tilde{f}^{(2M)}$. Aplicamos MOFS mais uma vez, sobre este último ponto, para, então, obtermos um novo ponto de convergência que será, por definição, o ponto de convergência, $\tilde{f}$, do MHIO.

Critério análogo valerá para o EH.

4.6.1 Condições iniciais aleatórias

Nesta subseção exibiremos resultados dos métodos para índices de ruídos

$$\varepsilon_{\alpha_F} = 0.1 \ \text{ e } \ 0.2.$$

Para valores de índices $\varepsilon_{\alpha_F} \geq 0.2$, observamos que os métodos EH e MHIO não conseguiram recuperar as imagens originais com boa nitidez. Isto se deve ao fato de a condição inicial não ser um ponto suficientemente próximo do mínimo global. O mesmo não ocorre com condições iniciais obtidas por perturbação da fase original, pois neste caso, estando $f^{(0)}$ mais próximo de f, uma recuperação mais nítida torna-se possível, mesmo em situações onde o índice de ruído ε_{α_F} torna-se maior. Isto será confirmado nos exemplos da próxima subseção.

Já os métodos MOFS e ER, quando executados isoladamente, convergiram, na maioria dos casos, a pontos, muitas vezes distantes do mínimo global. Por causa da robustêz dos métodos MOFS e ER e por acreditarmos na existência de muitos mínimos locais, MOFS e ER tornam-se, portanto, métodos de pouca aplicabilidade, quando executados isoladamente e quando nenhuma informação da fase original é usada na condição inicial. Quanto ao HIO, na presença de ruídos, sua instabilidade faz com que ele não estacione nas vizinhanças dos pontos de estagnação, mas o permite visitar várias outras vizinhanças onde estão vários outros pontos de estagnação,

[5] Em nossos experimentos, para índices $\varepsilon_{\alpha_F} \geq 0.1$, o menor valor que $\delta_{\tilde{f}}(\text{MHIO })$ atingiu foi da ordem de e-1. Dependendo da quantidade de ruídos, esse número pode crescer muito. Em alguns experimentos com $\varepsilon_{\alpha_F} = 0.5$, por exemplo, a melhor estimativa que encontramos apresentou erro $\delta_{\tilde{f}}(\text{MHIO }) = 0.46$.

até que ele, eventualmente caia numa vizinhança do mínimo global. Nesse momento devemos aplicar MOFS ou ER sobre o ponto atingido por HIO, na tentativa de atingirmos o mínimo global. Assim, juntando a robustez do MOFS (respect. do ER) com a instabilidade do HIO, obtemos um método poderoso, como MHIO (respect. EH), para recuperação de imagens, na presença de ruídos. Os problemas graves que surgem são : (1) a dificuldade de se saber quantas iterações serão necessárias ao HIO para que MHIO (ou EH) atinja, em um menor número de ciclos possível, o mínimo global, e (2) os pontos de estagnação que, em conjunto, formam uma pequena região do domínio do objeto onde MHIO (respect. EH) permanece estacionado durante um grande número de ciclos. Em nossos experimentos constatamos várias *regiões de estagnação*. Elas ocorreram em quase todos os experimentos que realizamos. Para evitá-las, tivemos de *quebrar* as regras de (4.20) em alguns ciclos de iterações.

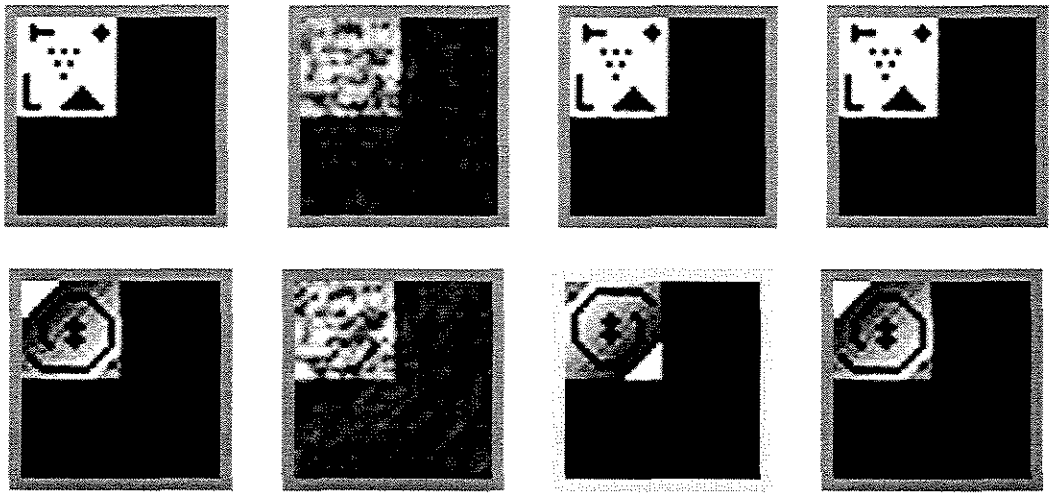

Figura 4.3: As imagens acima, na sequência da esquerda para a direita, representam respectivamente: o objeto original, f; a condição inicial aleatória, $f^{(0)}$; a imagem recuperada pelo método MHIO, $\tilde{f}|_{\text{MHIO}}$, e a recuperada pelo EH, $\tilde{f}|_{\text{EH}}$. Na primeira fileira, as imagens $\tilde{f}|_{\text{MHIO}}$ e $\tilde{f}|_{\text{EH}}$ são próximas a f, com erros dados respect. por $\delta_{\tilde{f}}$(MHIO) = .29769e-1 e $\delta_{\tilde{f}}$(EH) = .19845e-1. Na segunda fileira, MHIO convergiu próximo à imagem gêmea, com erro $\delta_{\tilde{f}}$(MHIO) = .19234e-1, e EH convergiu próximo à imagem verdadeira, com erro $\delta_{\tilde{f}}$(EH) = .18036e-1. Nestes experimentos, o índice de ruído é $\varepsilon_{\alpha_F} = 0.1$ e os valores de SNR são respectivamente 53.763 e 53.686. No primeiro experimento, ambos MHIO e EH foram executados com 1 ciclo. No segundo, MHIO foi executado com 3 ciclos e EH com 6.

Outra desvantagem do HIO que constatamos em nossos experimentos foi a de que quanto maior o ruído introduzido nas amplitudes, maior a dificuldade dele de se aproximar do mínimo global. Também maior é a probabilidade de ele estagnar em alguma região distante do mínimo global. Isto aconteceu em vários experimentos feitos para valores de $\varepsilon_{\alpha_F} = 0.3$ e $\varepsilon_{\alpha_F} = 0.5$.

Em resumo, podemos concluir que, para condições iniciais aleatórias, os métodos funcionam bem para $\varepsilon_{\alpha_F} = 0.1$. Também não foi verificado nenhuma melhora significativa do MHIO sobre EH, exceto que na maioria dos experimentos, o número de ciclos para o MHIO foi menor do que para o EH.

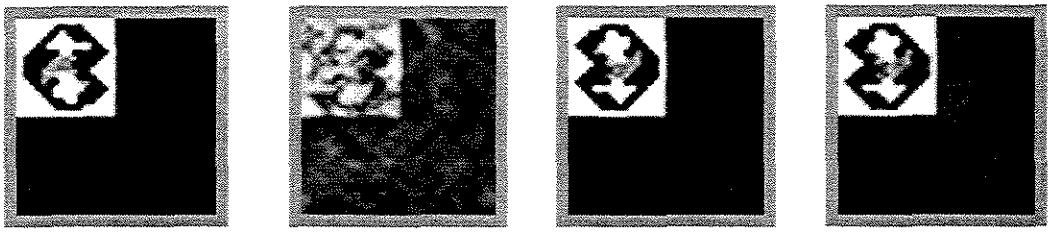

Figura 4.4: Experimentos com indíce de ruídos nas amplitudes e SNR dados respect. por $\varepsilon_{\alpha_F} = 0.1$ e SNR = 56.402. Condição inicial aleatória. Na sequência da esquerda para a direita temos f, $f^{(0)}$, $\tilde{f}|_{\text{MHIO}}$ e $\tilde{f}|_{\text{EH}}$. Ambos os métodos convergiram, em 2 ciclos, próximos à gêmea da imagem original, com erro $\delta_{\tilde{f}}$(MHIO) = .23139e-1 e $\delta_{\tilde{f}}$(EH) = .63065e-1.

A seguir apresentamos os resultados de MHIO e EH para índices de ruídos nas amplitudes, dado por $\varepsilon_{\alpha_F} = 0.2$. Para o exemplo da figura abaixo, ambos MHIO e EH estagnaram em uma região, apresentando em média erro dado por $\delta_{\tilde{f}}(\text{MHIO}) \simeq 0.3$. Por isso, quebramos a convenção (4.20), executando MHIO e EH com um número fixo de iterações igual a 50 para o HIO. Nesta nova situação , os dois algoritmos estagnaram primeiramente em uma região , cujo erro, para ambos, ficou próximo de 0.26. Depois eles saíram desta região de estagnação e foram para uma outra cujo erro, para ambos, ficou acima de 0.27. Em vista da imprevisibilidade desses métodos decidimos, então , interromper nossa busca, em ambos os casos, e coletar os resultados obtidos no terceiro ciclo, que foram os melhores dentre os outros. Assim, ao fim do terceiro ciclo, obtivemos, para ambos MHIO e EH, um mínimo

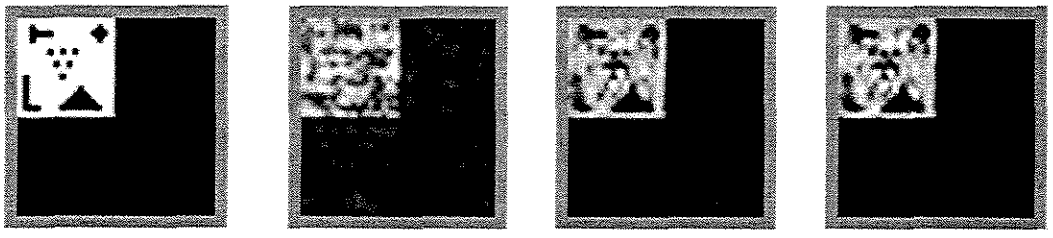

Figura 4.5: Experimentos com indíce de ruídos nas amplitudes e SNR dados respect. por $\varepsilon_{\alpha_F} = 0.2$ e SNR = 26.713. Condição inicial aleatória. Na sequência da esquerda para a direita temos f, $f^{(0)}$, $\tilde{f}|_{\text{MHIO}}$ e $\tilde{f}|_{\text{EH}}$. Os erros, após o terceiro ciclo de iterações, correspondentes às duas últimas imagens foram $\delta_{\tilde{f}}$(MHIO) = .26553 e $\delta_{\tilde{f}}$(EH) = .26145.

local, distante da imagem original pelos valores $\delta_{\tilde{f}}$(MHIO) = .26553 e $\delta_{\tilde{f}}$(MHIO) =

.26145 respectivamente.

Com as outras imagens, obtivemos igualmente, estagnação tanto do MHIO quanto do EH. Para $\varepsilon_{\alpha_F} \geq 0.3$ a situação fica ainda pior, ocorrendo sempre estagnação dos métodos, com erros de convergência ainda maiores e, portanto, com pontos de convergência associados a imagens ainda menos nítidas.

4.6.2 Condições iniciais obtidas por perturbação da fase original

Na Figura seguinte apresentamos as imagens produzidas por MOFS, ER e EH, a partir de perturbações da fase original, com índice de ruído $\varepsilon_\phi = 200$ e norma relativa $N_R = .63429$. O índice de ruído nas amplitudes e o SNR foram respectivamente $\varepsilon_{\alpha_F} = 0.4$, e SNR = 17.555.

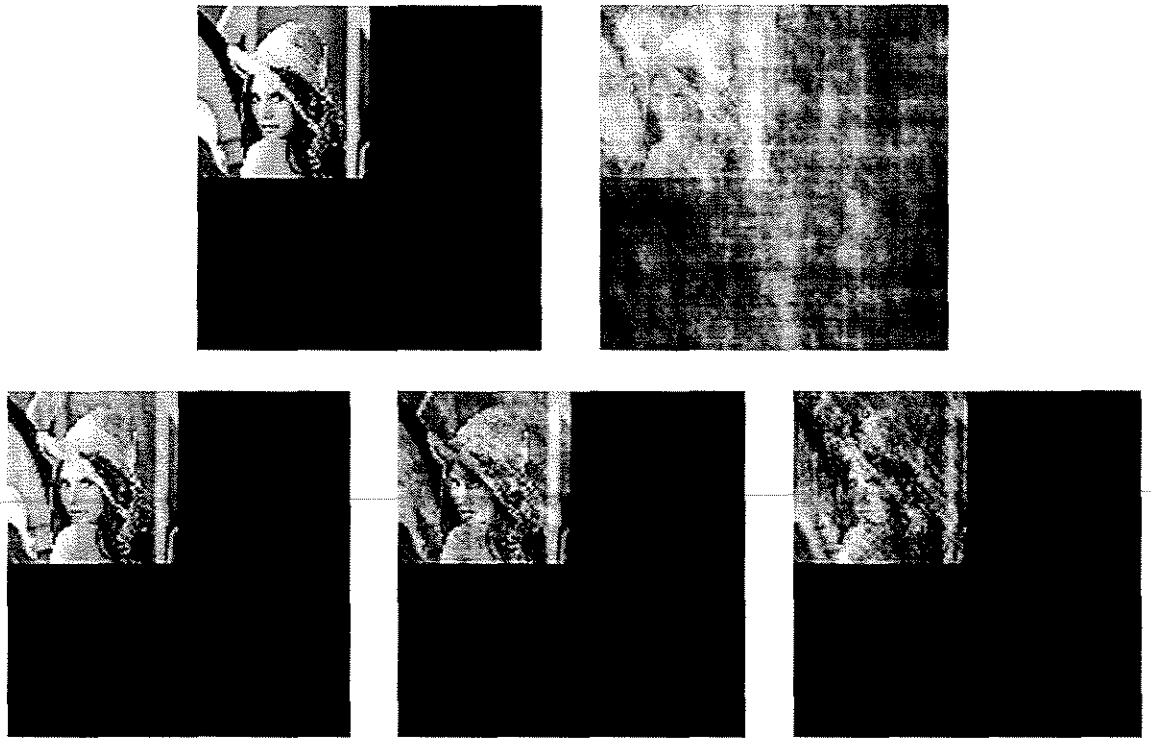

Figura 4.6: As duas imagens do topo representam o objeto original e a condição inicial, obtida por perturbação da fase original, com índice de ruído $\varepsilon_\phi = 200$ e $N_R = .63429$. O índice de ruído nas amplitudes e o SNR são $\varepsilon_{\alpha_F} = 0.4$, e SNR = 17.555.As imagens de baixo são , da esquerda para direita, os pontos (que acreditamos serem mínimos locais) obtidos pelos métodos MOFS, ER e EH, com erros dados por $\delta_{\tilde{f}}$(MOFS) = .25411, $\delta_{\tilde{f}}$(ER) = .28921 e $\delta_{\tilde{f}}$(EH) = .31983 respectivamente.

Como dissemos, níveis de ruídos com $\varepsilon_{\alpha_F} \geq 0.2$ já faz do HIO um método

não muito eficiente para visitar regiões de mínimos globais. Essa ineficiência aumenta à medida que ε_{α_F} cresce. $\varepsilon_{\alpha_F} = 0.4$, p.ex., já é considerado um valor alto para esses prósitos. Quanto ao ER, sabemos que a distância máxima para a sua convergência (veja seção 4.4), na maioria dos casos, é sempre menor que a exigida pelo MOFS (veja Tab. 7 da seção 4.4). Este fenômeno nos faz acreditar que, na maioria dos casos em que temos uma convergência *razoável*[6] do MOFS, para condições iniciais cuja distância máxima satisfaz $D_M|_{\text{MOFS}} \geq 0.6$, temos uma convergência ruim do ER, quando iniciado à mesma distância $D_M|_{\text{MOFS}}$ da fase original. Assim, para essess tipos de condições iniciais, MOFS tem apresentado melhores resultados do que ER e o EH. No exemplo da Figura, o erro relativo aos métodos MOFS e ER não foram tão distantes um do outro ($\delta_{\tilde{f}}$(MOFS) = .25411, $\delta_{\tilde{f}}$(ER) = .28921). Mesmo assim, o valor excedente de $\delta_{\tilde{f}}$(ER) sobre $\delta_{\tilde{f}}$(MOFS) (= .03510) já é o suficiente para produzir uma piora na nitidez da imagem recuperada pelo ER, quando comparada com a da imagem recuperada pelo MOFS. Dos 3 métodos, EH foi o que apresentou pior erro. Para atingir a imagem que a Figura mostra, foram necessários 7 ciclos, com quebra da convenção (4.20) nos sexto e sétimo ciclos, onde foram executados para o HIO, em cada um, 50 iterações ao invés de 400 e 300 respectivamente. O menor valor que conseguimos para o erro $\delta_{\tilde{f}}$(EH), entre todos aqueles calculados ao longo dos ciclos, exceto o da primeira execução do ER ($\delta_{\tilde{f}}$(ER) = .28921), foi .31983.

No próximo exemplo, tanto MOFS quanto EH apresentaram bons resultados. ER foi o que apresentou pior resultado. A situação considerada neste exemplo é a de que todos os três métodos foram iniciados do mesmo ponto, $\theta^{(0)}$, cuja distância da fase original considerada foi $D_M|_{\text{MOFS}} = .62273$. Neste experimento, consideramos o indíce de ruídos nas amplitudes dado por $\varepsilon_{\alpha_F} = 0.3$. O valor do SNR correspondente é SNR = 18.653. O índice da perturbação da fase que determinou a distância máxima para a convergência do MOFS foi $\varepsilon_\phi = 50$. Para esta condição inicial, os erros apresentadoos pelos três métodos foram $\delta_{\tilde{f}}$(MOFS) = .77266e-1, $\delta_{\tilde{f}}$(ER) = .36071e+00 e $\delta_{\tilde{f}}$(EH) = .62914e-1.

Apesar de termos obtido bons resultados para ambos, MOFS e EH, a vantagem do nosso método sobre o EH, nas condições citadas acima, é a de que não necessitamos do processo exaustivo de busca do EH, com tentativas aleatórias de mudança no número de iterações para o HIO, sem ao menos podermos pré estabelecer qualquer critérios (como fizemos em (4.20), devido à completa imprevisibilidade do HIO. Neste exemplo, que iremos exibir a seguir, foram necessários exaustivos 11 ciclos, com quebra da regra que determina o número de iterações para o HIO a partir do quinto ciclo, onde foram sempre tomadas 50 iterações, com excessão do quinto ciclo, onde consideramos 100 iterações para o HIO.

[6]no sentido de a imagem produzida apresentar alguma nitidez,

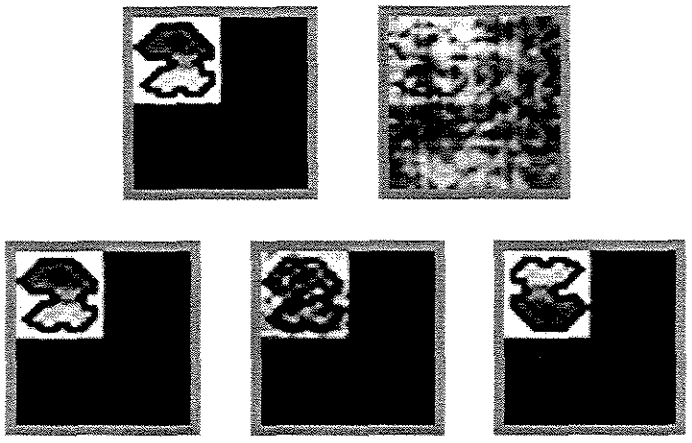

Figura 4.7: As duas imagens do topo representam o objeto original e a condição inicial, obtida por perturbação da fase original, que determina a distância máxima para a convergência do MOFS. O índice de ruído é $\varepsilon_\phi = 50$ e $D_M|_{\text{MOFS}} = .62273$. O índice de ruído nas amplitudes e o SNR são $\varepsilon_{\alpha_F} = 0.3$, e SNR = 18.653. As imagens de baixo são, da esquerda para direita, os pontos obtidos pelos métodos MOFS, ER e EH, com erros dados por $\delta_{\tilde{f}}$(MOFS) = .77266e-1, $\delta_{\tilde{f}}$(ER) = .36071e+00 e $\delta_{\tilde{f}}$(EH) = .62914e-1 respectivamente.

4.6.3 Condições iniciais com a informação das amplitudes com sinal

Encerramos nossos exemplos em problemas que usam a informação das amplitudes com sinal, na construção da condição inicial. O sinal de $F_u = \alpha_u \exp(i\phi_u)$ (veja (4.9)) traz bastante informação a respeito da imagem verdadeira. De fato, em todos os experimentos sem ruídos, (veja Tabela 2) foi verificado a convergência do MOFS ao mínimo global, quando usamos esse tipo de condição inicial. ER, embora tenha convergido, apresentou menor precisão no erro de convergência. A própria condição inicial, construída a partir das amplitudes com sinal, já traz informação da imagem original. De fato, para estes pontos, a imagem do suporte aparece novamente, e invertida, no quarto quadrante da imagem $f^{(0)}$. Esta aparência que a imagem $f^{(0)}$ apresenta, se mantém, mesmo em problemas com ruídos nas amplitudes. Mesmo que o indíce do ruído ainda seja grande. Por isso, quando iniciamos de condições do tipo em (4.10), MOFS sempre recupera a imagem original, quase que completamente, desde que o ruído das amplitudes não seja muito grande.

Nós não apresentamos os resultados nem do MHIO e nem do EH devido ao péssimo desempenho do HIO, que se manteve sempre muito distante da imagem original em todos os ciclos. O fracasso do HIO se justifica pela enorme quantidade de ruído introduzida nas amplitudes: $\varepsilon_{\alpha_F} = 1.0$ para o primeiro exemplo e $\varepsilon_{\alpha_F} = 1.5$ para o segundo.

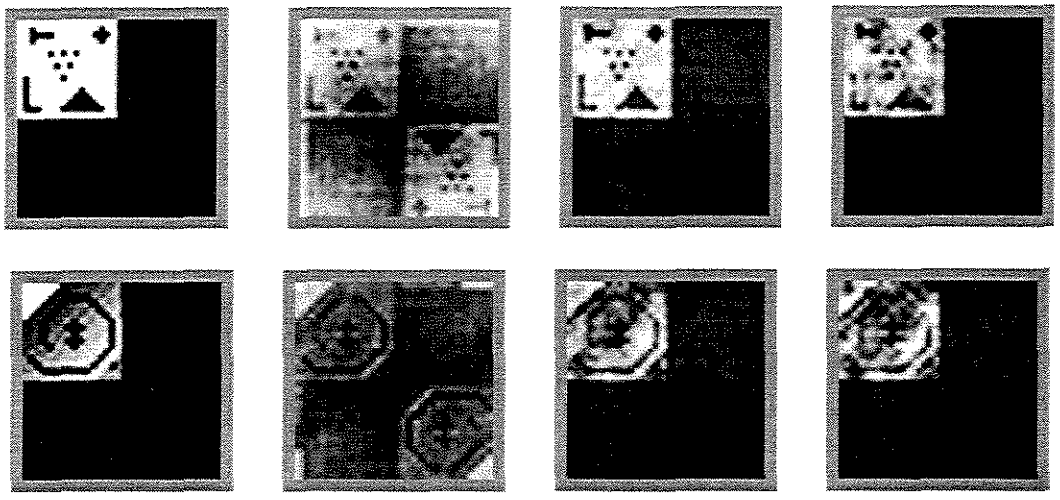

Figura 4.8: As imagens acima, na sequência da esquerda para a direita, representam respectivamente: o objeto original, f; a condição inicial, $f^{(0)}$, com a informação das amplitudes com sinal; a imagem recuperada pelo método MOFS, $\tilde{f}|_{\text{MOFS}}$, e a recuperada pelo ER, $\tilde{f}|_{\text{ER}}$. O índice de ruído nas amplitudes e o valor do SNR são dados por $\varepsilon_{\alpha_F} = 1.0$, SNR = 5.0078, relativos às imagens da primeira fileira e $\varepsilon_{\alpha_F} = 1.5$, SNR = 3.2136, relativos às da segunda. Os erros são $\delta_{\tilde{f}}$(MOFS) = .22354e+00, $\delta_{\tilde{f}}$(ER) = .29382e+00, para os mínimos locais da primeira fileira, e $\delta_{\tilde{f}}$(MOFS) = .46189e+00, $\delta_{\tilde{f}}$(ER) = .53301e+00 para os da segunda.

4.7 Conclusões

Depois de muitos experimentos, concluímos que, na situação de amplitudes com ruídos, nosso método apresenta vantagens sobre os métodos de Fienup quando usamos os seguintes tipos de condições iniciais: as obtidas por uma perturbação da fase original e as que usam informação da amplitude com sinal. Especificamente, para as condições do primeiro tipo, MOFS converge bem, mesmo quando o SNR é relativamente baixo. Isto se deve ao fato de o método de otimização L.BFGS.B , que usamos na composição do MOFS, ser um método *tipo Newton* e, portanto, convergir sempre a um mínimo global da função custo associada, para todas as condições iniciais que estiverem suficientemente próximas deste mínimo global. Esse comportamento, como vimos nas Tabelas 5 e 7, não ocorre com o ER, tendo em vista que, em todos os experimentos em que adotamos $f^{(0)} = f(\alpha_c, \phi_c, \alpha, \theta^{(0)})$ como condição inicial para o ER, onde $\theta^{(0)}$ é a condição inicial que define a distância máxima para a convergência do MOFS, ele (o ER) não conseguiu atingir o mínimo global. Vemos portanto que apesar de ER e MOFS serem métodos que reduzem a mesma função custo, seus comportamentos são diferentes, o que para nós não é novidade, pois como sabemos, ER é um algoritmo de projeção e o MOFS é um algoritmo do tipo Newton.

Outra situação que põe MOFS em vantagem sobre os outros métodos é aquela

em que o índice de ruído introduzido nas amplitudes seja relativamente alto, de modo que MOFS continue convergindo *próximo* da solução global, mas que HIO permaneça sempre afastado desse mínimo. Como consequência, MHIO será também melhor do que EH. Outra vantagem do MHIO sobre o EH que temos observado em muitos experimentos é que, em caso de boa convergência para ambos, o primeiro tem exigido menos ciclos para se chegar perto do mínimo global do que o segundo.

Uma das desvantagens do MOFS é o tempo de execução para a busca da solução. Para o caso de imagens de suporte 64×64, por exemplo, e critérios de parada de baixíssima precisão, MOFS tem gasto em média 01 (uma) hora, enquanto ER e HIO, com 10 mil iterações, não gastam mais do que 2 minutos. Entretanto, em situações especiais de ruídos nas amplitudes que põem MOFS em situação de vantagem sobre o EH (como comentamos no parágrafo anterior), pode ocorrer de o tempo gasto pelo MOFS, mesmo sendo grande, ainda ser menor do que o tempo total gasto pelo EH para atingir um ponto que esteja suficientemente próximo da solução, tendo em vista a imprevisibilidade do número de ciclos necessários para EH atingir esse ponto, devido ao alto número de pontos de estagnação.

Capítulo 5

Pós-processamento

Geralmente na prática os valores coletados das amplitudes de uma certa transformada de Fourier, F, vêm acompanhados de ruídos que precisam ser eliminados. Na tentativa de eliminarmos tais ruídos, nós sugerimos neste capítulo, para o caso 1-D, um método de filtragem dos dados das amplitudes com ruídos baseado na teoria dos multiplicadores de Lagrange [57]. Basicamente a idéia é encontrar o vetor mais próximo do vetor das amplitudes com ruídos que satisfaça à identidade das amplitudes, uma condição necessária para a existência de solução do problema inverso.

Seja

$$\alpha_F = (\alpha_0, \alpha_1, \ldots, \alpha_m)^T \in \mathbb{R}^{m+1} \tag{5.1}$$

o vetor das amplitudes de F. Suponhamos que, em um dado experimento, as amplitudes em (5.1) foram detectadas com a presença de ruídos aditivos, i.e.,

$$\tilde{\alpha}_F = \alpha_F + \eta, \tag{5.2}$$

onde η é o ruído definido, de acordo com (4.3), por

$$\eta = \varepsilon \frac{\|\alpha_F\|}{m+1}(\mathbf{r} - 0.5).$$

Queremos encontrar um vetor

$$\tilde{\tilde{\alpha}}_F \in \mathbb{R}^{m+1}, \tag{5.3}$$

o mais próximo possível de $\tilde{\alpha}_F$ e que satisfaça à identidade das amplitudes (1.62). Em outros termos, queremos resolver o seguinte problema de mínimos quadrados com restrição :

$$\begin{aligned} &\min_{\beta \in \mathbb{R}^{m+1}_+} && \|\beta - \tilde{\alpha}_F\|^2; \\ &\text{restrito a:} && \beta_0^2 + 2\sum_{k=1}^{m-1}(-1)^k \beta_k^2 + (-1)^m \beta_m^2 = 0 \end{aligned} \tag{5.4}$$

A condição de restrição (5.4) pode ser reescrita na seguinte forma de produto interno

$$< \beta, D\beta > = 0, \tag{5.5}$$

onde D é a matriz diagonal, de tamanho $(m+1) \times (m+1)$, dada por

$$D = \text{diag}(1, -2, 2, \ldots, -2(-1)^m, (-1)^m).$$

Resolveremos o problema de minimização acima usando o método dos multiplicadores de Lagrange. Para tanto, devemos considerar o seguinte funcional definido por

$$J(\beta, \lambda) = \|\beta - \tilde{\alpha}_F\|^2 - \lambda < \beta, D\beta >, \tag{5.6}$$

onde $\lambda \in \mathbb{R}$ é o multiplicador de Lagrange. Sabemos que o ponto $(\tilde{\beta}, \tilde{\lambda})$ que minimiza o funcional J deve satisfazer

$$\begin{array}{l} \nabla_\beta J(\tilde{\beta}, \tilde{\lambda}) = 0; \\ \nabla_\lambda J(\tilde{\beta}, \tilde{\lambda}) = 0. \end{array} \tag{5.7}$$

Mostra-se que

$$\nabla_\beta J = 2(\beta - \tilde{\alpha}_F - \lambda D\beta).$$

Desde que a segunda equação de (5.7) é exatamente a equação de restrição (5.5), segue que a solução $(\tilde{\beta}, \tilde{\lambda})$ de (5.7) satisfaz:

$$\tilde{\beta} = [I - \tilde{\lambda}D]^{-1}\tilde{\alpha}_F; \tag{5.8}$$

$$< \tilde{\beta}, D\tilde{\beta} >= 0. \tag{5.9}$$

Uma substituição de (5.8) em (5.9) implica na seguinte equação :

$$\frac{1}{(1-\tilde{\lambda})^2}\tilde{\alpha}_0^2 - \frac{2}{(1+2\tilde{\lambda})^2}\tilde{\alpha}_1^2 + \frac{2}{(1-2\tilde{\lambda})^2}\tilde{\alpha}_2^2 - \ldots$$

$$\ldots + \frac{2(-1)^{m+1}}{(1+2(-1)^m\tilde{\lambda})^2}\tilde{\alpha}_{m-1}^2 + \frac{(-1)^m}{(1-(-1)^m\tilde{\lambda})^2}\tilde{\alpha}_m^2 = 0. \tag{5.10}$$

Assim, o ponto $(\tilde{\beta}, \tilde{\lambda})$ que minimiza o funcional J dado em (5.6) deve satisfazer (5.8) quando $\tilde{\lambda}$ satisfaz a equação (5.10).

Desde que $\tilde{\lambda}$ deve ser real, mostraremos então que a equação (5.10) possui sempre uma raiz real. De fato para o caso em que m é par, (5.10) se reescreve como

$$p_1(\tilde{\lambda}) := \frac{A}{(1-\tilde{\lambda})^2} + \frac{B}{(1+2\tilde{\lambda})^2} + \frac{C}{(1-2\tilde{\lambda})^2} = 0, \tag{5.11}$$

onde $A = \tilde{\alpha}_0^2 + \tilde{\alpha}_m^2 > 0$, $B = -2(\tilde{\alpha}_1^2 + \tilde{\alpha}_3^2 + \ldots + \tilde{\alpha}_{m-1}^2) < 0$, $C = 2(\tilde{\alpha}_2^2 + \tilde{\alpha}_4^2 + \ldots \tilde{\alpha}_{m-2}^2) > 0$; e para o caso em que m é ímpar, (5.10) se reescreve como

$$p_2(\tilde{\lambda}) := \frac{D}{(1-\tilde{\lambda})^2} + \frac{E}{(1+2\tilde{\lambda})^2} + \frac{F}{(1-2\tilde{\lambda})^2} + \frac{G}{(1+\tilde{\lambda})^2} = 0, \tag{5.12}$$

onde $D = \tilde{\alpha}_0^2 > 0$, $E = -2(\tilde{\alpha}_1^2 + \tilde{\alpha}_3^2 + \ldots + \tilde{\alpha}_{m-2}^2) < 0$, $F = 2(\tilde{\alpha}_2^2 + \tilde{\alpha}_4^2 + \ldots + \tilde{\alpha}_{m-1}^2) > 0$, $G = -\tilde{\alpha}_m^2 < 0$.

Agora, desde que $p_1(1/2) = p_2(1/2) = +\infty$ e $p_1(-1/2) = p_2(-1/2) = -\infty$ segue que tanto (5.11) quanto (5.12) possuem uma raiz no intervalo aberto $(-1/2, 1/2)$, o que garante, portanto, (5.8).

O nosso candidato a $\tilde{\tilde{\alpha}}_F$ será, então, o vetor $\tilde{\beta}$ dado pela equação (5.8), onde $\tilde{\lambda}$ é solução real de (5.10).

No procedimento de determinação do vetor $\tilde{\alpha}_F$, nós escolhemos um valor para o índice de ruído, de modo a garantir sempre a existência de raiz real para p_1 e p_2.

A seguir exibiremos uma tabela com os resultados dos erros $e(\tilde{f})$ e $e(\tilde{\tilde{f}})$ obtidos após executarmos os métodos ER e MOFS, para cada um dos exemplos da Tabela 1, nas situações anterior e posterior à filtragem, ou seja, quando tomamos os vetores $\tilde{\alpha}_F$ e $\tilde{\tilde{\alpha}}_F$, respectivamente, como representantes para o vetor das amplitudes. Aqui, $\tilde{f}$ e $\tilde{\tilde{f}}$ representam as estimativas de f quando consideramos $\tilde{\alpha}_F$ e $\tilde{\tilde{\alpha}}_F$ como o vetor das amplitudes respectivamente.

Observemos que não incluímos desta vez os resultados da execução do HIO devido à sua instabilidade na presença de ruídos (também verificou-se nos experimentos que o processo de filtragem não garante a convergência do HIO).

Para cada exemplo da tabela consideraremos o índice, ε_η, do ruído η em (5.2), constante e igual a 0.9. Também assumiremos

$$seed(\eta) = 87$$

em todos os exemplos.

Em todos os experimentos, ER foi executado com 4000 iterações.

Tab.8 Determinação do erro obtido pela norma da estimativa f avaliada fora do suporte. **Amplitudes com ruídos.** $\theta^{(0)}$ = condição inicial aleatória. $\varepsilon_\eta = 0.9$

m	seed(f)	seed($\theta^{(0)}$)	$\epsilon(\tilde{f}, \tilde{\tilde{f}})$	$e(\tilde{f})\|_{MOFS}$ // $e(\tilde{\tilde{f}})\|_{MOFS}$	$e(\tilde{f})\|_{\text{ER}}$ // $e(\tilde{\tilde{f}})\|_{\text{ER}}$
8	123002	2	.36893e-1	.86377e-1//.57591e-1	.86377e-1//.57591e-1
8	3141592	2951413	.38335e-1	.70938e-1//.15498e-1	.70938e-1//.15476e-1
8	1012	2101	.27749e-1	.86229e-1//.71007e-1	.86229e-1//.71007e-1
8	7	908	.42810e-1	.56709e-1//.13317e-1	.56708e-1//.13316e-1
8	1	2	.19655e-1	.37342e-1//.88310e-4	.37341e-1//.97460e-7
16	435	32	.16200e-1	.22006e+0//.21652e+0	.68242e-1//.57728e-1
16	9215143	295113	.80676e-2	.36648e-1//.32942e-1	.50791e-1//.48436e-1
16	2100	2101	.14102e-1	.69364e-1//.59164e-1	.69329e-1//.59145e-1
32	709	908	.60690e-2	.59914e-1//.57231e-1	.62097e-1//.59423e-1
32	21709	12908	.98627e-2	.45772e-1//.31960e-1	.58040e-1//.41220e-1
64	71027	76908	.52598e-2	.33833e-1//.25798e-1	.71835e-1//.68043e-1
64	82038	87109	.47894e-2	.56755e-1//.51834e-1	.62097e-1//.57471e-1
64	1042510	1093221	.66947e-2	.44000e-1//.29354e-1	.48762e-1//.36685e-1
128	2153	4327	.27083e-2	.39592e-1//.32654e-1	.42101e-1//.38252e-1

Além dos erros, exibiremos na tabela o valor da distância $\epsilon(\tilde{f}, \tilde{\tilde{f}})$ entre as estimativas $\tilde{f}$ e $\tilde{\tilde{f}}$ da imagem original f, calculada no domínio de Fourier.

Esta distância é útil para avaliar quão próximo os vetores $\tilde{\alpha}_F$ e $\tilde{\tilde{\alpha}}_F$ se encontram um do outro. A fórmula para o cálculo de $\epsilon(\tilde{f}, \tilde{\tilde{f}})$ está dada em (2.14).

Comparando os resultados da Tabela 8, vemos que, ao contrário do caso sem ruídos (ver tabela da seção 4.3.1), MOFS apresenta um comportamento completamente similar ao ER. Em muitos experimentos observou-se inclusive que, partindo de uma mesma condição inicial, ER e MOFS convergiram a um mesmo mínimo local.

A quarta coluna da tabela acima mostra que os vetores $\tilde{\alpha}_F$ e $\tilde{\tilde{\alpha}}_F$ estão muito próximos um do outro, o que explica, talvez, o fato de a filtragem não ter sido um método tão eficiente assim, capaz de garantir a convergência dos algoritmos a um ponto mais próximo de um mínimo global. Isto pode ser constatado na tabela, ao notarmos que houve uma redução pouco significativa nos valores de $e(\tilde{f})$ para $e(\tilde{\tilde{f}})$.

Apêndice A

Prova do Lema 1.2

Neste apêndice resolvemos explicitamente o problema da fase 1-D nos casos $m = 2$ e $m = 3$.

Lema 1.2 *Sejam* $f = [\ f_0\ \ f_1\ \ \ldots\ \ f_{m-1}\ \ 0\ \ 0\ \ \ldots\ \ 0\ ]^T \in \mathbb{R}^n$; $\alpha_u = |F_u|\ \forall u$ *e* $\epsilon_u = \mathrm{sgn}(F_u)$, *para* $u = 0, m$.

(a) Se $m = 2$, *então*

$$f_0 = (\epsilon_0\alpha_0 + \epsilon_2\alpha_2)/2, \quad f_1 = (\epsilon_0\alpha_0 - \epsilon_2\alpha_2)/2. \tag{A.1}$$

(b) *Se* $m = 3$, *então*

$$(f_0 = r^+,\ f_1 = r,\ f_2 = r^-) \tag{A.2}$$

ou

$$(f_0 = r^-,\ f_1 = r,\ f_2 = r^+), \tag{A.3}$$

onde

$$r^\pm = [6(\epsilon_0\alpha_0 + \epsilon_3\alpha_3) \pm \sqrt{\Delta}]/24,$$

$$r = (\epsilon_0\alpha_0 - \epsilon_3\alpha_3)/2,$$

$$\Delta = 12[16\alpha_2^2 - (\epsilon_0\alpha_0 - 3\epsilon_3\alpha_3)^2] \geq 0.$$

Demonstração : Suponhamos que $\alpha_u = |F_u|\ \forall\ u$. (a) Se $m = 2$, obtém-se (A.1) resolvendo-se o sistema (1.66) para f_0 e f_1. (b) Suponhamos $m = 3$. Mostra-se que o sistema formado pelas equações (1.65) e (1.66) é equivalente ao sistema

$$\begin{aligned} f_0 f_2 &= \alpha_0^2/4 + \alpha_3^2/4 - ||F||^2/12, \\ f_0 + f_2 &= (F_0 + F_3)/2, \\ f_1 &= (F_1 - F_3)/2, \\ \alpha_0^2 - 2\alpha_1^2 + 2\alpha_2^2 - \alpha_3^2 &= 0. \end{aligned} \tag{A.4}$$

Consideremos então o sistema (A.4). A terceira equação nos dá imediatamente $f_1 = (\epsilon_0\alpha_0 - \epsilon_3\alpha_3)/2$. Notemos que a última equação é a identidade das amplitudes. Se substituímos a segunda equação na primeira obtemos, após simplificação dos termos [1], a seguinte equação do segundo grau, em f_0,

$$12f_0^2 - 6(F_0 + F_3)f_0 + (\alpha_0^2 - 4\alpha_2^2 + 3\alpha_3^2) = 0. \tag{A.5}$$

Como f_0 é solução real de uma equação do segundo grau, então o discriminante desta equação deve ser não negativo, i.e.,

$$\Delta = 192\alpha_2^2 - 12(\alpha_0^2 - 6F_0F_3 + 9\alpha_3^2) \geq 0,$$

ou equivalentemente

$$4\alpha_2 \geq |\epsilon_0\alpha_0 - 3\epsilon_3\alpha_3|.$$

Segue que $f_0 = r^+$ ou $f_0 = r^-$, onde

$$r^+ = \frac{\epsilon_0\alpha_0 + \epsilon_3\alpha_3}{4} + \frac{\sqrt{\Delta}}{24}$$

e

$$r^- = \frac{\epsilon_0\alpha_0 + \epsilon_3\alpha_3}{4} - \frac{\sqrt{\Delta}}{24}$$

são as raízes de (A.5). Substituindo f_0 por r^+ ou r^- na segunda equação de (A.4), otemos

$$f_2 = \frac{\epsilon_0\alpha_0 + \epsilon_3\alpha_3}{4} - \frac{\sqrt{\Delta}}{24} = r^-$$

ou

$$f_2 = \frac{\epsilon_0\alpha_0 + \epsilon_3\alpha_3}{4} + \frac{\sqrt{\Delta}}{24} = r^+. \quad \blacksquare$$

[1] no processo de simplificação dos termos, utilizamos a identidade das amplitudes dada no sistema (A.4)

Apêndice B

Matriz diagonalmente dominante

Neste apêndice provamos que $\mathbf{BB}^T$ é diagonalmente dominante por linhas, onde $\mathbf{B}$ é a matriz dos coeficientes do sistema (1.82) ou, equivalentemente, a matriz dada por (1.84).

Proposição B.1 *Sejam os inteiros $m_1 \geq 1$, $m_2 \geq 1$ e seja $\mathbf{B}$ a matriz dos coeficientes do sistema linear (1.83), i.e., a matriz dada por (1.84), (1.85) e (1.86). Então o termo geral, c_{jk}, da matriz simétrica $\mathbf{BB}^T$, para $0 \leq j \leq k \leq m_1 + m_2$, é*

$$c_{jk} = \begin{cases} 4(2m_1m_2 - 1); & j = k = 0, \\ -2(1 + (-1)^{j+k}); & 0 \leq j < k \leq m_2 - 1, \\ 4(m_1m_2 - 1); & 1 \leq j = k \leq m_2 - 1, \end{cases}$$

$$c_{j,m_2+k} = -[1 + (-1)^{m_2+j}]\,[1 + (-1)^{m_1+k}]; 0 \leq j \leq m_2 - 1,\ 0 \leq k \leq m_1,$$

$$c_{m_2+j,m_2+k} = \begin{cases} 4(2m_1m_2 - 1); & j = k = 0,\ ou\ j = k = m_1, \\ -2(1 + (-1)^{j+k}); & 0 \leq j < k \leq m_1, \\ 4(m_1m_2 - 1); & 1 \leq j = k \leq m_1 - 1, \end{cases}$$

Consequentemente, $\mathbf{BB}^T$ é diagonalmente dominante por linhas, para todos os valores inteiros positivos m_1 e m_2 tais que $m_1m_2 > 1$.

Demonstração : Para provarmos o resultado principal, precisaremos de outro resultado importante, cuja verificação é imediata:

$$\sum_{v=0}^{m-1} \omega^{pv} = \begin{cases} \dfrac{1 - (-1)^p}{1 - \omega^p}; & \text{se } \omega^p \neq 1, \\ m; & \text{se } \omega^p = 1, \end{cases} \tag{B.1}$$

quando $\omega = \exp(\pi i/m)$. Se tomarmos a parte real de ambos os lados de (B.1), e levando em conta que

$$\frac{1}{1 \pm \mathrm{Re}(\omega^p)} = \mathrm{Re}\left[\frac{1 \pm \omega^{-p}}{(1 \pm \omega^p)(1 \pm \omega^{-p})}\right] = \frac{1 \pm \mathrm{Re}(\omega^{-p})}{2(1 \pm \mathrm{Re}(\omega^{-p}))} = \frac{1}{2}, \tag{B.2}$$

obtemos

$$\sum_{v=0}^{m-1} \mathrm{Re}(\omega^{pv}) = \begin{cases} \frac{1}{2}[1-(-1)^p]; & \text{se } \omega^p \neq 1, \\ m; & \text{se } \omega^p = 1. \end{cases} \tag{B.3}$$

Apesar de termos uma expressão explícita para a matriz $\mathbf{B}$ em (1.85) e (1.86), será mais fácil provarmos a expressão que determina o termo geral de $\mathbf{B}\mathbf{B}^T$ se utilizarmos o sistema de equações polinomiais (1.82):

$$S_j(b) = 0; \quad j = 0, 1, \ldots, m_1 + m_2, \tag{B.4}$$

quando

$$\begin{aligned} S_j(b) \;&:=\; b_{00} + 2\sum_{v=1}^{m_2-1} \mathrm{Re}(\omega_2^{vj}) b_{0v} + (-1)^j b_{0m_2} + 2\sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1} (-1)^u \mathrm{Re}(\omega_2^{vj}) b_{uv} \\ &+\; (-1)^{m_1}\left[b_{m_1 0} + 2\sum_{v=1}^{m_2-1} \mathrm{Re}(\omega_2^{vj}) b_{m_1 v} + (-1)^j b_{m_1 m_2} \right], \end{aligned} \tag{B.5}$$

para $j = 0, 1, \ldots, m_2 - 1$, e

$$\begin{aligned} S_{m_2+k}(\alpha_F) \;&:=\; b_{00} + 2\sum_{v=1}^{m_2-1} (-1)^v b_{0v} + (-1)^{m_2} b_{0m_2} + 2\sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1} (-1)^v \mathrm{Re}(\omega_1^{uk}) b_{uv} \\ &+\; (-1)^k\left[b_{m_1 0} + 2\sum_{v=1}^{m_2-1} (-1)^v b_{m_1 v} + (-1)^{m_2} b_{m_1 m_2} \right], \end{aligned} \tag{B.6}$$

para $k = 0, 1, \ldots, m_1$.

Notemos que

$$c_{jk} = < [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(k)} >,$$

onde $[\mathbf{B}]_{(j)}$ representa a j-ésima linha da matriz $\mathbf{B}$ e, $< \cdot, \cdot >$, o produto interno euclidiano. Como a matriz $\mathbf{B}\mathbf{B}^T$ é simetrica, de ordem $m_1 + m_2 + 1$, obteremos c_{jk} apenas para valores de $0 \leq j \leq k \leq m_1 + m_2$. O cálculo de $< [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(k)} >$ é obtido multiplicando-se os coeficientes do polinômio $S_j(b)$ pelos seus correspondentes em $S_k(b)$ e, então, somando-se os produtos assim obtidos. Deveremos então considerar os três seguintes casos

(I) $< [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(k)} >; \quad 0 \leq j \leq k \leq m_2 - 1,$

(II) $< [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(m_2+k)} >; \quad 0 \leq j \leq m_2 - 1, \quad 0 \leq k \leq m_1$ e

(III) $< [\mathbf{B}]_{(m_2+j)}, [\mathbf{B}]_{(m_2+k)} >; \quad 0 \leq j \leq k \leq m_1.$

A demonstração dos três casos são análogas entre si, portanto a faremos somente para um deles, digamos o caso (II).

Devemos mostrar que

$$< [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(m_2+k)} > = -[1+(-1)^{m_1+k}]\,[1+(-1)^{m_2+j}],$$

para todo $0 \leq j \leq m_2 - 1, \quad 0 \leq k \leq m_1$. De fato multiplicando os coeficientes de $S_j(b)$, dados em (B.5), pelos seus correspondentes em $S_{m_2+k}(b)$, dados em (B.6), e depois somando os termos, assim obtidos, teremos

$$\begin{aligned} < [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(m_2+k)} > \;\; = \;\; & 1 + 4\sum_{v=1}^{m_2-1}(-1)^v \mathrm{Re}(\omega_2^{vj}) + (-1)^{m_2+j} \\ & + 4\sum_{u=1}^{m_1-1}\sum_{v=0}^{n_2-1}(-1)^v \mathrm{Re}(\omega_2^{vj})(-1)^u \mathrm{Re}(\omega_1^{uk}) \\ & + (-1)^{m_1+k}\left[1 + \; 4\sum_{v=1}^{m_2-1}(-1)^v \mathrm{Re}(\omega_2^{vj}) + (-1)^{m_2+j}\right]. \end{aligned} \tag{B.7}$$

Denotemos

$$\sum\nolimits_1 := \sum_{u=1}^{m_1-1}(-1)^u \mathrm{Re}(\omega_1^{uk}), \quad \sum\nolimits_2 := \sum_{v=1}^{m_2-1}(-1)^v \mathrm{Re}(\omega_2^{vj}),$$

$$a_j := 1 + (-1)^{m_2+j}, \qquad b_k := 1 + (-1)^{m_1+k}.$$

Verifica-se imediatamente que

$$\sum_{v=0}^{n_2-1}(-1)^v \mathrm{Re}(\omega_2^{vj}) = a_j\,(1 + \textstyle\sum_2)\,. \tag{B.8}$$

Então levando o segundo membro das expressões que definem $\sum_1$, $\sum_2$, a_j, b_k e, também, o segundo membro de (B.8), em (B.7), obtemos

$$< [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(m_2+k)} > \;\; = \;\; a_j b_k + 4b_k \textstyle\sum_2 + 4a_j \sum_1 (\sum_2 + 1)\,. \tag{B.9}$$

Desde que

$$\sum\nolimits_1 = \sum_{u=1}^{m_1-1} \mathrm{Re}(\omega_1^{u(m_1+k)}),$$

então, fazendo uso de (B.3) e observando que, para o caso (II),

$$\omega_1^{m_1+k} = 1 \quad \text{somente quando } k = m_1,$$

consegue-se mostrar que

$$\sum\nolimits_1 = \begin{cases} -\dfrac{1}{2}b_k; & 0 \leq k \leq m_1 - 1 \\ m_1 - 1; & k = m_1. \end{cases} \tag{B.10}$$

Se também levarmos em conta que, para o caso (II), $\omega_2^{m_2+j}$ é sempre $\neq 1$, então, novamente por (B.3), conclui-se que

$$\sum_2 = -\frac{1}{2}a_j; \quad 0 \leq j \leq m_2 - 1. \tag{B.11}$$

Agora, levando o lado direito de (B.10) e (B.11) em (B.9) e levando em conta que

$$a_j^2 = 2a_j,$$

temos:

caso (II.1): se $0 \leq j \leq m_2 - 1$, e $0 \leq k \leq m_1 - 1$, então

$$\begin{aligned} < [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(m_2+k)} > &= a_j b_k + 4b_k(-a_j/2) + 4a_j(-b_k/2)\{-(a_j/2)+1\} \\ &= a_j^2 b_k - 3a_j b_k = -a_j b_k. \end{aligned}$$

caso (II.2): se $0 \leq j \leq m_2 - 1$, e $k = m_1$, então

$$\begin{aligned} < [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(m_2+m_1)} > &= a_j b_{m_1} + 4b_{m_1}(-a_j/2) + 4a_j(m_1-1)\{-(a_j/2)+1\} \\ &= a_j b_{m_1} - 2b_{m_1}a_j - 2a_j^2(m_1-1) + 4a_j(m_1-1) \\ &= -a_j b_{m_1}. \end{aligned}$$

Assim, mostramos que

$$< [\mathbf{B}]_{(j)}, [\mathbf{B}]_{(m_2+k)} > = -a_j b_k = -[1+(-1)^{m_2+j}]\,[1+(-1)^{m_1+k}],$$

para todo $0 \leq j \leq m_2 - 1$, $0 \leq k \leq m_1$.

A demonstração dos outros casos, (I) e (III), são inteiramente análogas à do caso (II) e, portanto, serão omitidas.

Finalmente mostremos que $\mathbf{B}\mathbf{B}^T$ é diagonalmente dominante por linhas, i.e., que

$$|c_{jj}| > \sum_{\substack{k=0 \\ k \neq j}}^{m_1+m_2} |c_{jk}|, \tag{B.12}$$

para todo $0 \leq j \leq m_1 + m_2$, com $m_1 m_2 > 1$. Para simplificação de notação, denotaremos a soma do lado direito de (B.12) simplesmente por $\sum_{k \neq j} |c_{jk}|$.

Os elementos da diagonal principal da matriz $\mathbf{B}\mathbf{B}^T$ assume um dos dois valores indicados abaixo:

$$c_{jj} = \begin{cases} 4(2m_1m_2 - 1); & \text{se } j = 0,\ m_2,\ m_2 + m_1, \\ 4(m_1m_2 - 1); & \text{se } 1 \leq j \leq m_2 - 1 \text{ ou } m_2 + 1 \leq j \leq m_2 + m_1 - 1. \end{cases}$$

Notemos também que os elementos não pertencentes à diagonal principal assumem valores iguais a 0 ou a -4. Mostraremos agora que as linhas $j = 0,\ j = m_2$ e $j = m_1 + m_2$, da matriz $\mathbf{B}\mathbf{B}^T$, contêm no máximo $m_2 + m_1 - 1$ elementos $c_{jk},\ j \neq k$, com $|c_{jk}| = 4$, ou, equivalentemente, que cada uma elas contém pelo menos um elemento nulo. De fato, por inspeção rápida nas fórmulas que definem o termo geral c_{jk}, conclui-se que

$$\begin{aligned} c_{01} &= 0, \\ c_{m_2,m_2+1} &= 0, \\ c_{m_2+m_1,m_2+m_1-1} &= c_{m_2+m_1-1,m_2+m_1} = 0 \end{aligned}$$

Segue portanto que

$$\sum_{k\neq j} |c_{jk}| \ \leq\ 4(m_1 + m_2 - 1), \quad \text{para}\ \ j = 0,\ m_2,\ m_2 + m_1.$$

Então, como $m_1 m_2 > 1$, resulta

$$\begin{aligned} |c_{jj}| - \sum_{k\neq j} |c_{jk}| &\geq 4(2m_1m_2 - 1) - 4(m_2 + m_1 - 1) \\ &= 8m_1m_2 - 4m_1 - m_2 \\ &= 4(m_1 - 1)m_2 + 4(m_2 - 1)m_1 \\ &> 0, \end{aligned}$$

para $j = 0,\ m_2,\ m_1 + m_2$.

Em seguida mostraremos que cada linha j, com

$$1 \leq j \leq m_2 - 1 \quad \text{ou} \quad m_2 + 1 \leq j \leq m_2 + m_1 - 1,$$

possuem pelo menos três elementos nulos. De fato, se $1 \leq j \leq m_2 - 1$, então

$$\begin{aligned} c_{j,j-1} &= c_{j-1,j} = 0, \\ c_{j,j+1} &= 0, \\ c_{j,m_2+m_1-1} &= 0, \end{aligned}$$

e, para $m_2 + 1 \leq j \leq m_2 + m_1 - 1$,

$$\begin{aligned} c_{j,m_2-1} &= c_{m_2-1,j} = 0, \\ c_{j,j-1} &= c_{j-1,j} = 0, \\ c_{j,j+1} &= 0. \end{aligned}$$

Portanto, se $1 \leq j \leq m_2 - 1$ ou $m_2 + 1 \leq j \leq m_2 + m_1 - 1$, temos

$$\sum_{k\neq j} |c_{jk}| \ \leq\ 4(m_1 + m_2 - 3).$$

Consequentemente

$$\begin{aligned} |c_{jj}| - \sum_{k \neq j} |c_{jk}| &\geq 4(m_1 m_2 - 1) - 4(m_2 + m_1 - 3) \\ &= 4(m_1 m_2 - m_1 - m_2 + 2) \\ &= 4[(m_1 - 1)(m_2 - 1) + 1)] \\ &> 0, \end{aligned}$$

para $1 \leq j \leq m_2 - 1$ ou $m_2 + 1 \leq j \leq m_2 + m_1 - 1$. Conclui-se portanto a demonstração de que $\mathbf{B}\mathbf{B}^T$ é diagonalmente dominante por linhas, sempre que $m_1 m_2 > 1$. O caso trivial $m_1 = m_2 = 1$ não é considerado pois, neste caso,

$$\mathbf{B}\mathbf{B}^T = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}$$

não é diagonalmente dominante por linhas ■

Bibliografia

[1] ARSENAULT, H.A.; CHALASINSKA-MACUKOW, K. - The solution to the phase retrieval problem using the sampling theorem, *Optics communications*, **47** (6), (Oct. 1983), 380-6.

[2] BAKUSHINSKII, A.B. - The problem of the convergence of the iteratively regularized Gauss-Newton method, *Comput. Math. Math. Phys.*, **32**, (1992), 1353-9.

[3] BARAKAT, R.; NEWSAM, G. - Necessary conditions for a unique solution to two-dimensional phase recovery, *J. Math. Phys.*, **25** (11), (Nov. 1984), 3190-93.

[4] BATES, R.H.T. - Astronomical speckle imaging, *Physics Reports(Reviews Sect. of Phys. Lett.)*, **90**, (1982), 203-97.

[5] BAUSCHKE, H.H.; COMBETTES, P.L.; LUKE, D.R. - Phase retrieval, error reduction algorithm, and Fienup variants: a view from convex optimization, *JOSA A*, **19** (7), (Jul. 2002), 1334-1345.

[6] BLASCHKE, B., NEUBAUER, A.; SCHERZER, O. - On convergence rates for the iteratively regularized Gauss-Newton method, *IMA J. Numer. Anal.*, **17**, (1997), 421-36.

[7] BOCHNAK, J.; COSTE, M.; ROY, M-F. - *Géométrie Algébrique Réelle*, Springer-Verlag, (1987).

[8] BOYLE, J.P., DYKSTRA, R.L. - A method for finding projections onto the intersection of convex sets in Hilbert spaces, in *Advances in Order Restricted Statistical Inference* (Springer, Berlin), (1986), 28-47.

[9] BREGMAN, L.M. - The method of successive projections for finding a common point of convex sets, *Dokl. Akad. Nauk SSSR*, **162** (3), (1965), 487-90.

[10] BRICOGNE, G. - Geometric sources of redundancy in intensity data and their use for phase determination, *Acta Cryst.*, **A30**, (1974), 395-400.

[11] BRIGHAM, E.O. - *The Fast Fourier Transform and its Application*, Prentice Hall, New Jersey (1974)

[12] BRUCK, Y.M.; SODIN, L.G. - On the ambiguity of the image reconstruction problem, *Opt. Commun.*, **30,** (1979), 304-8.

[13] BYRD, R.H.; LU P.; NOCEDAL J.; ZHU, C. - A limited memory algorithm for bound constrained optimization, *SIAM J. Scientific Computing,* **16** (5), (1995), 1190-208

[14] BYRD, R.H.; NOCEDAL, J.; SCHNABEL, R.B. - Representation of quasi-Newton matrices and their use in limited memory methods, *Mathematical Programming* **63** (4), (1994), 129-56

[15] DAINTY, J.C. - *Laser Speckle and Related Phenomena*, Springer Verlag, Heidelberg, 2nd edition, (1984), 255-320.

[16] DAINTY, J.C. - Progress in image reconstruction in astronomy, *Proc. SPIE,* **492,** (1984).

[17] DENNIS, j.e., Jr, SCHNABEL, R.B. - *Numerical Methods for Unconstrained Optimization and Non-linear Equations*, New york: Prentice Hall, (1983).

[18] DEUFLHARD, P.; ENGL, H.W.; SCHERZER, O. - A convergence analysis of iterative methods for the solution of nonlinear ill-posed problems under affinely invariant conditions, *Inverse Problems,* **14**, (1998), 1081-106.

[19] DOBSON, D. - Phase reconstruction via nonlinear least squares, *Inverse Problems,* **8**, (1992), 541-48.

[20] DYKSTRA, R.L. - An algorithm for restricted least squares regression, *J. Am. Stat. Assoc.,* **78**, (1983), 837-42.

[21] ECKSTEIN, J., BERTSEKAS, D.P. - On the Douglas-Rachford splitting method and the proximal point algorithm for maximal monotone operators, *Math. Program. (Ser. A),* **55**, (1992), 293-318.

[22] ENGL, H.W.; KUNISCH, K.; NEUBAUER, A. - Convergence rates for Tikhonov regularization of nonlinear ill-posed problems, *Inverse Problems,* **5**, (1989), 523-40.

[23] FIDDY, M.A.; BRAMES, B.J.; DAINTY, J.C. - Enforcing irreducibility for phase retrieval in two dimensions, *Opt. Lett.*, 8, (1983), 96-8.

[24] FIDDY, M.A.; GREENAWAY, A.H. - Phase Retrieval using zero information, *Opt. Commun.,* **29** (3), (1979), 270-72.

[25] FIENUP, J.R. - Phase retrieval algorithms: a comparison, *Applied Optics,* **21**(15), (Aug. 1982), 2758-69

[26] FIENUP, J.R.; WACKERMAN, C.C. - Phase-retrieval stagnation: problems and solutions, *J. Opt. Soc. Am. A,* **3** (11), (1986), 1897-907.

[27] FLETCHER, R. - *Practical Methods of Optimization,* Vol. 1, New york: Wiley, (1980).

[28] FUKS, B. - Introduction to the theory of analytic functions of several complex variables, *Translations of Mathematical Monographs, Amer. Math. Soc.,* **8,** Providence, R.I., (1963).

[29] GALLAGHER, N.C.,LIU, B. - Method for computing kinoforms that reduces image reconstruction errors, *Appl. Opt.,* **12**, (1973), 2328-35.

[30] GERCHBERG, R.W. - Super-resolution through error energy reduction, *Optica Acta,* **21**, (1974), 709-20.

[31] GERCHBERG, R.W.; SAXTON, W.O. - A practical algorithm for the determination of phase from image and diffraction plane pictures, *Optik,* **35**, (1972), 237-46.

[32] GOLUB, G.H.; VAN LOAN, C.F. - *Matrix Computations,* The Johns Hopkins Univ. press, (1983).

[33] GONZALEZ, R.C.; WOODS, R.E. - *Digital Image Processing* Addison Wesley, (1993)

[34] GROETSCH, C.W. - *The theory of Tikhonov Regularization for Fredholm Equations of the First Kind,* Boston: Pitman, (1984).

[35] GUBIN, L.G., POLYAK, B.T., RAIK, E.V. - The method of projections for finding the common point of convex sets, *USSR Computational Mathematics and Mathematical Physics,* **7**, (6), (1967), 1-24.

[36] HANKE, M.; NEUBAUER, A.; SCHERZER, O. - A convergence analysis of the Landweber iteration for nonlinear ill-posed problems, *Numer. Math.,* **72**, (1995), 21-37.

[37] HAYES, M.H. - The Reconstruction of a Multidimensional Sequence from the Phase or Magnitude of Its Fourier Transform, *IEEE Trans. Acoust., Speech, Sig. Proc.,* **ASSP-30** (2), (1982), 140-54.

[38] HAYES, M.H.; McCLELLAN, J.H. - Reducible polynomials in more than one variable, *Proc. IEEE,* **70** (2), (Feb. 1982), 197-98.

[39] HAUPTMAN, H. - The role of crystallographic symmetry in the direct methods of X-ray crystallography. Crystal symmetries. *Comp. Math Appl.*, **16**, 5-8, (1988), 385-96.

[40] HORMANDER, L. - *Linear Partial Differential Operators*, Springer-Verlag, New York, (1963).

[41] HUISER, A.M.J., FERWERDA, H.A. - On the phase retrieval problem in electron microscopy from image and diffraction pattern, *Optik*, **46**, (1976), 407-20.

[42] HURT, N.E. -*Phase Retrieval and Zero Crossings*, Kluwer, Dordrecht, (1989).

[43] IZRAELEVITZ, D.; LIM, J.S. - A new direct algorithm for image reconstruction from fourier transform magnitude, *IEEE trans. on acoustics sppech and sig. proc.*, **ASSP-35** (4), (Apr. 1987), 511-19.

[44] KAILATH, T. - *Linear Systems*, Prentice Hall, Englewood Cliffs, (1980).

[45] KLIBANOV, M.V. - On uniqueness of the determination of a compactly supported function from the modulus of its Fourier transform, *Sov. Math.-Dokl.*, **32**, (1985), 668-70.

[46] KLIBANOV, M.V. - Determination of a function with compact support from the absolute value of its Fourier transform, and an inverse scattering problem, *Differential Equations*, **22**, (1987), 1232-40.

[47] KLIBANOV, M.V., SACKS, P.E., TIKHONRAVOV, A.V. - The phase retrieval problem, *Inverse Problems*, **11**, (1995), 1-28.

[48] LADD, M.F.C.; PALMER, R.A. - *Structure Determination by X-Ray Crystallography*, New York, Plenum.

[49] LANE, R.G. - Phase retrieval using conjugate gradient minimization, *Journal of Modern Optics*, **38** (9), (1991), 1797-813

[50] LANG, S. - *Algebra*, 2nd ed., Addison-Wesley, (1984).

[51] LEVENBERG, K. - A method for the solution of certain non-linear problems in least squares, *Quart. Appl. Math.*, **2**, (1944), 164-68.

[52] LEVI, A. - Image Restoration by the Method of Projections with Application to the Phase and Magnitude Retrieval Problems, Ph.D. Thesis, Renssealaer Polytechnic Institute, Dept of ECSE, (Dec. 1983).

[53] LEVI, A.; STARK, H. - Image restoration by the method of generalized projections with application to restoration from magnitude, *J. Opt. Soc. Am.*, **1** (2), (1984), 932-43.

[54] LEVI, A.; STARK, H. - Signal restoration from phase by projections onto convex sets, *J. Opt. Soc. Am.*, **73**, (1983), 810-22.

[55] LIU, D.C.; NOCEDAL, J. - On the limited memory BFGS method for large scale optimization methods, *Mathematical Programming*, **45**, (1989), 503-28.

[56] LOUIS, A.K. - Ghosts in tomography - the null space of the Radon transform, *Math. Meth. Appl. Sci.*, **3**, (1981), 1-10.

[57] LUENBERGER, D.G. - *Linear and Nonlinear Programming*, second edition, Addison-Wesley publishing co. (1984)

[58] LUDWIG, D. - The Radon transform on Euclidean space, *Comm. Pure. Appl. Math.*, **XIX**, (1966), 49-81.

[59] MARQUARDT, D.W. - An algorithm for least-squares estimation of nonlinear parameters *SIAM, J. Soc. Indust. Appl. Math.*, **11**, (1963), 431-41.

[60] MARTINEZ, A.G. - O problema da recuperação da fase da transformada de Fourier a partir de duas magnitudes. Tese de Mestrado, Imecc/Unicamp, (1999).

[61] METHERELL, A.F. - *The relative importance of phase and amplitude in acoustical holography,* in Acoustical Holography, Vol.2 (A.F. Metherell and L. Larimore, eds.), Plenum Press, New York, (1970), Chapt. 14.

[62] MILNOR, J. - *Morse Theory*, Princeton University Press, (1969).

[63] MIURA, N.; BABA, N. - Image reconstruction from spectral magnitude under a nonnegativity constraint, *Optics Letters*, **21** (13), (Jul. 1996), 979-81.

[64] MORÉ, J.J. - *Numerical Analysis*, edited by G.A. Watson (Lecture Notes in Mathematics, Vol.630), Berlin: Springer, (1977), 105-6.

[65] MOU-YAN, Z.; UNBEHAUEN, R. - Methods for Reconstruction of 2-D Sequences from Fourier Transform Magnitude, *IEEE Trans. on Image Proc.*, **6**(2), (Feb. 1997), 222-33

[66] NEUBAUER, A. - Tikhonov regularization for nonlinear ill-posed problems: optimal convergence and finite-dimensional approximation, *Inverse Problems*, **5**, (1989), 541-57.

[67] NIETO-VESPERINAS, M. - A study of the performance of nonlinear least-square optimization methods in problem of phase retrieval, *Optica Acta,* **33**(6), (1986), 713-22

[68] NIETO-VESPERINAS, M.;DAINTY, J.C. - A note on Eisenstein's irreducibility criterion for two-dimensional sampled objects, *Opt. Commun.,* **54,** (1985), 333-4.

[69] PLANCHEREL, M., PÓLYA, G. - Fonctions entiéres et intégrales de Fourier multiples, I, *Comment. Math. Helv.,* **9**, (1937), 224-48.

[70] PLANCHEREL, M., PÓLYA, G. - Fonctions entiéres et intégrales de Fourier multiples, II, *Comment. Math. Helv.,* **10**, (1938), 110-63.

[71] RAMACHANDRAN, G.N.; SRINIVASAN, R. - *Fourier Methods in Crystallography*, Wiley-Interscience, NewYork, 1970.

[72] RONKIN, L. - Introduction to the theory of entire functions of several variables, *Translations of Mathematical Monographs, Amer. Math. Soc.,* **44,** Providence, R.I., (1974).

[73] SALVADOR, C.F. - *O Problema da Recuperação da Fase da Transformada de Fourier: Novos Resultados*, Tese Doutoramento, Imecc/Unicamp (Dez. 1997).

[74] SANZ, J.L.C. - Mathematical considerations for the problem of Fourier transform phase retrieval from magnitude, *SIAM J. Appl. Math.,* **45** (4), (Aug. 1985), 651-64.

[75] SANZ, J.L.C.; HUANG T.S. - Phase reconstruction from magnitude of band-limited multidimensional signals, *J. of mathematical analysis and appl.,* **104** (1984), 302-8.

[76] SANZ, J.L.C.; HUANG T.S. - Unique reconstruction of a band-limited multidimensional signal from its phase or magnitude, *J. Opt. Soc. Am.,* **73**, (1983), 1446-50.

[77] SANZ, J.L.C.; HUANG T.S.; CUKIERMAN, F. - Stability of unique Fourier transform phase reconstruction, *J. Opt. Soc. Am.,* **73** (11) (Nov. 1983), 1442-45.

[78] SAXTON, W.O. - *Computer Techniques for Image Processing in Electron Microscopy*, Academic Press, New York, (1978).

[79] SCHERZER, O.; ENGL, H.W.; KUNISCH, K. - Optimal a-posteriori parameter choice for Tikhonov regularization for solving nonlinear ill-posed problems, *SIAM J. Numer. Anal.,* **30**, (1993), 1796-838.

[80] SEIDMAN, T.I.; VOGEL, C.R. - Well-posedness and convergence of some regularization methods for nonlinear ill-posed problems, *Inverse Problems,* **5**, (1989), 227-38.

[81] SELDIN, J.H.; FIENUP, J.R. - Numerical investigation of the uniqueness of phase retrieval, *JOSA A,* **7** (3), (1990), 412-27.

[82] STARK, H. - *Image Recovery: Theory and Application*, Academic Press, inc., San Diego (1986)

[83] TAKAJO, H.; TAKAHASHI, T.; KAWANAMI, H. ; UEDA, R. - Numerical investigation of the iterative phase-retrieval stagnation problem: territories of convergence objects and holes in their boundaries, *JOSA-A,* **14** (12), (Dec. 1997), 3175-87

[84] TAKAJO, H.; TAKAHASHI, T.; UEDA, R.; TANINAKA, M. - Study on the convergence property of the hybrid input-output algorithm used for phase retrieval, *JOSA-A,* **15** (11), (Nov. 1998), 2849-61.

[85] TAKAJO, H.; TAKAHASHI, T.; SHIZUMA, T. - Further study on the convergence property of the hybrid input-output algorithm used for phase retrieval, *JOSA-A,* **16** (9), (Sep. 1999), 2163-8.

[86] TAKAJO, H.; SHIZUMA, T.; TAKAHASHI, T.; TAKAHATA, S. - Reconstruction of an object from its noisy Fourier modulus: ideal estimate of the object to be reconstructed and a method that attempts to find that estimate, *Applied Optics,* **38** (26), (Sep. 1999), 5568-76.

[87] TARATORIN, A.M., SIDERMAN, S. - Signal reconstruction from noisy-phase and-magnitude data, *Applied Optics,* **33** (23), (Aug. 1994), 5415-25.

[88] URIELI, S. - Image analysis and representation by spectral phase, M.Sc. Thesis, EE, Technium, (Mar. 1996)

[89] URIELI, S.; COHEN, N.; PORAT, M. - Image representation by spectral phase, *IEEE-Image Processing,* **7** (6), (1998), 838-53.

[90] VAN HOVE, P.L.; HAYES, M.H.; LIM, J.S.; OPPENHEIM, A.V. - Signal reconstruction from signed Fourier transform magnitude, *IEEE trans. on acoustics, speech, and signal proc.,* **ASSP-31** (5), (1983), 1286-93.

[91] WOLF, E. - Is a complete determination of the energy spectrum of light possible from measurements of the degree of coherence?, *Proc. Phys. Soc.,* **80**, (1962), 1269-72.

[92] YAGLE, A.E.; BELL, A.E. - One-and Two-Dimensional Minimum and Non-minimum Phase Retrieval by Solving Linear Systems of Equations, *IEEE trans. signal proc.*, **47** (11), (Nov. 1999), 2978-89.

[93] YOULA, D.C. - Image restoration by the method of projections onto convex sets - Part1., *Polytechnic Institute of New York Report,* No. POLY-MRI, (1981) 1420-81.

[94] YOULA, D.C., WEBB, H. - Image restoration by the method of convex projections: Part1 - Theory., *IEEE Trans. Medical Imaging,* **MI-1,** (1982), 81-94.

[95] ZHU, C.; BYRD, R.H.; LU P.; NOCEDAL, J. - *L-BFGS-B: Fortran subroutines for large scale bound constrained optimization*, Tech. Report, NAM-11, EECS Department, Northwestern University (1994)