

unesp



UNIVERSIDADE ESTADUAL PAULISTA
Instituto de Biociências, Letras e Ciências Exatas
DEPARTAMENTO DE CIÊNCIAS DE COMPUTAÇÃO E ESTATÍSTICA

Métodos Numéricos para o Retoque Digital

Claudia Augusta dos Santos

Dissertação de Mestrado
Pós-Graduação em Matemática Aplicada
MAP - 090

Métodos Numéricos para o Retoque Digital

Claudia Augusta dos Santos

Dissertação de Mestrado apresentada ao Instituto de Biociências, Letras e Ciências Exatas da Universidade Estadual Paulista “Júlio de Mesquita Filho”, Câmpus de São José do Rio Preto, São Paulo, para a obtenção do título de Mestre em Matemática Aplicada.

Orientador: Prof. Dr. Maurílio Boaventura

São José do Rio Preto

Fevereiro de 2005

A Deus.

Aos meus pais, Rosi e Augusto.

Aos meus irmãos, Catarina e Thomás.

À Lethícia.

A Gilberto.

Dedico

Agradecimentos

Agradeço a todos que direta ou indiretamente participaram da realização desta dissertação e, em especial, agradeço:

Aos meus queridos pais, Augusto e Rosi, que me deram a vida e batalharam para que eu chegasse até aqui.

Ao Professor Doutor Maurílio Boaventura, pela orientação, confiança e dedicação na elaboração deste trabalho.

Ao meu amor Gibinha, que sempre me incentivou e esteve presente em todos os momentos da minha vida.

Aos meus familiares, em geral, que sempre me apoiaram.

À minha amiga Karina, uma pessoa que ficará pra sempre marcada na minha vida, pela amizade, garra e autenticidade.

Ao André, um amigo para qualquer hora e que também sempre me ajudou na parte computacional.

Aos amigos da pós-graduação em Matemática Aplicada.

À UNESP - São José do Rio Preto - SP, DCCE e seus funcionários pelo apoio.

Ao CNPq - Conselho Nacional de Desenvolvimento Científico e Tecnológico - pelo auxílio financeiro.

Resumo

O objetivo deste trabalho é aplicar Métodos Numéricos de ordem de precisão mais alta ao problema de Retoque Digital, visando melhorar a qualidade da aproximação quando comparada com o Método de Euler, que é geralmente utilizado para esse tipo de problema. Para testar a eficiência de tais métodos, utilizamos três modelos de Retoque Digital: o modelo proposto por Bertalmío, Sapiro, Ballester e Caselles (BSBC), o modelo de Rudin, Osher e Fatemi conhecido como Variacional Total (TV) e o modelo de Chan e Shen, chamado de Difusão Guiada pela Curvatura (CDD).

Palavras-chave: Retoque Digital, Método Previsor-Corretor, Métodos Lineares de Passo Múltiplo, Método de Euler, Diferenças Finitas, Desoclusão, Processamento Digital de Imagens, Métodos Numéricos.

Abstract

The purpose of this work is to apply Numerical Methods of higher order to the problem of Digital Inpainting, aiming to improve the quality of the approach when compared with the Euler's Method which is generally used for this kind of problem. To test the efficiency of these methods we use three models of Digital Inpainting: the model considered by Bertalmío, Sapiro, Ballester and Caselles (BSBC), the model of Rudin, Osher and Fatemi known as Total Variation (TV) and the model of Chan and Shen, named Curvature Driven Diffusion (CDD).

Key-words: Digital Inpainting, Predictor-Corrector Method, Linear Multistep Methods, Euler's Method, Finite Differences, Disocclusion, Processing of Digital Images, Numerical Methods.

Sumário

Introdução	1
1 Conceitos Preliminares	4
2 Métodos Numéricos para Resolução de Equações Diferenciais Ordinárias (EDO)	8
2.1 O Método de Diferenças Finitas	8
2.2 O Método de Euler	11
2.2.1 Estabilidade do Método de Euler	16
2.3 Métodos lineares de passo múltiplo	17
2.3.1 A forma geral dos métodos lineares de passo múltiplo	18
2.3.2 Consistência, Estabilidade e Convergência	21
2.3.3 Condições de Ordem	22
2.4 Os Métodos de Runge-Kutta	24
2.4.1 Ordem e Convergência do método geral de passo um explícito	25
2.4.2 Dedução dos métodos de Runge-Kutta clássicos	28
2.5 Método Previsor-Corretor	31
2.5.1 Algoritmo Previsor-Corretor ($m \geq 1$)	33
2.5.2 Erro de Truncamento Local do Método Previsor-Corretor	34
2.5.3 Conclusão Geral:	37
2.5.4 Estabilidade do Par Previsor-Corretor no Modo PECE	38
2.6 Considerações Finais	42

3	Modelos Matemáticos para Retoque Digital	44
3.1	Introdução	44
3.2	Modelo de Retoque Digital BSBC	44
3.3	Difusão no Processo de Retoque Digital	48
3.3.1	Discretização dos termos do modelo BSBC	49
3.3.2	Uma forma alternativa para a equação do modelo BSBC	50
3.3.3	Discretização do processo de difusão	52
3.4	Modelo Variacional Total (TV)	54
3.5	Modelo da Difusão Guiada pela Curvatura (CDD)	57
4	Resultados Numéricos	60
4.1	Resultados Experimentais	65
5	Conclusão	79
	Referências Bibliográficas	81

Introdução

Atualmente, tem-se observado um interesse crescente no desenvolvimento e aplicações em processamento de imagens. Uma das primeiras aplicações envolvendo técnicas de processamento de imagens consistiu na melhoria da qualidade de fotos transmitidas através do cabeamento submarino entre Londres e Nova York. As fotos eram codificadas antes de serem transmitidas e, posteriormente, reconstituídas no receptor.

Desde a metade da década de 60 do século passado, a área de processamento de imagens vem experimentando um vigoroso crescimento em aplicações diversas. Além disso, aplicações no programa aeroespacial, e também outras soluções vem sendo geradas para uma grande variedade de problemas, que, embora nem sempre diretamente relacionados, compartilham de métodos capazes de realçar informações relevantes. Em medicina, por exemplo, é possível beneficiar-se das imagens de raio-X pós-processadas de alto contraste, ou codificadas por intermédio de cores. Algumas técnicas similares também são empregadas por geógrafos no estudo de padrões de poluição a partir de imagens aéreas, ou obtidas através de satélites. Outras áreas, como a astronomia, biologia, medicina nuclear, indústria e defesa, utilizam-se das técnicas de processamento de imagens como ferramentas indispensáveis para a melhoria de interpretação ou diagnóstico.

Devido ao desenvolvimento notável da área de processamento de imagens, o tópico restauração tem sido amplamente estudado, no contexto de equações diferenciais parciais (EDP). Neste trabalho, abordamos o processo de retoque digital, ou seja, a técnica de modificar uma imagem danificada ou defeituosa, recompondo suas partes

degradadas. Aplicações desta técnica, incluem a reconstrução de fotografias antigas e filmes danificados; remoção de textos sobrepostos e a remoção de objetos inteiros de uma dada imagem.

A palavra “retoque” foi introduzida por restauradores profissionais, os quais referem-se a prática de preenchimento manual de domínios perdidos ou causados por danos acidentais.

O termo “retoque digital” foi inicializado por Bertalmío, Sapiro, Ballester e Caselles [1], quando propuseram um modelo de retoque digital baseado em uma EDP de terceira ordem e no transporte ao longo das linhas de nível.

Um outro modelo de retoque digital é conhecido como Variacional Total (TV) de Rudin, Osher e Fatemi. Este modelo surgiu a partir de um estudo do método Variacional Total e, por sua vez realiza com sucesso o processo de desocclusão. Um exemplo disto são as ilusões embaraçadas e analisadas por Kanizsa [13]. Apesar de ser muito útil, apresenta uma deficiência quando se trata de domínios grandes, devido ao custo direto da comunicação à longa distância. Inspirados nesta inconveniência, Chan e Shen, propuseram um modelo de retoque chamado de Difusão Guiada pela Curvatura (CDD) (“Curvature Driven Diffusion”), tendo como objetivo obter conexão à longa distância e empregar informações através da curvatura média na difusão.

O objetivo aqui é aplicar métodos numéricos aos modelos BSBC, CDD e TV. Além disso, analisamos também os erros, médio e máximo, os quais são definidos no Capítulo 4, para essas classes de métodos.

Este trabalho está organizado do seguinte modo:

No Capítulo 1, introduzimos alguns conceitos indispensáveis para o Processamento Digital de Imagens.

No Capítulo 2, fazemos um estudo dos métodos numéricos para resolução de equações diferenciais ordinárias, no qual apresentamos os métodos lineares de passo múltiplo e, também, o Método Previsor-Corretor.

No Capítulo 3, relatamos os métodos para o problema de Retoque Digital, bem como um detalhamento especial para o modelo BSBC.

No Capítulo 4, mostramos a aplicação dos métodos numéricos e os resultados experimentais. Para isto, usamos várias imagens testes e comparações com o Método

de Euler. Tais comparações foram realizadas visualmente e através do cálculo dos erros médio e máximo.

Finalmente, no Capítulo 5 relacionamos algumas conclusões a respeito dos métodos numéricos aplicados aos modelos, assim como algumas observações sobre o trabalho realizado.

Capítulo 1

Conceitos Preliminares

Neste capítulo, apresentamos algumas definições e alguns conceitos relacionados ao Processamento Digital de Imagens, que se fazem necessários para o desenvolvimento e entendimento deste trabalho. Tais definições e conceitos foram retirados de [11].

Definição 1.1. (*Imagem Digital*)

O termo imagem digital ou imagem monocromática refere-se à função bidimensional de intensidade da luz monocromática $f(x, y)$, em que x e y referem-se às coordenadas espaciais de brilho (ou gradação de cinza) de qualquer par (x, y) .

Definição 1.2. (*Nível de cinza*)

É a intensidade da luz monocromática (brilho) nos pontos (x, y) da imagem definida pela função $f(x, y)$.

Definição 1.3. (*Borda*)

No contexto de imagem digital, borda é o limite entre duas regiões com propriedades distintas de nível de cinza. Está diretamente relacionada à mudança brusca de nível de cinza em uma região.

Definição 1.4. (*Representação da Imagem Digital*)

Uma imagem digital pode ser representada por uma matriz, cujos índices de linhas e colunas identificam um ponto das linhas de nível. O correspondente valor

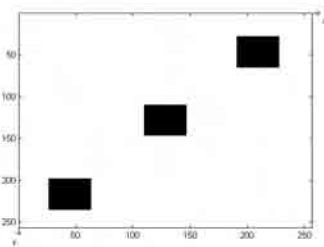


Figura 1.1: Representação de uma imagem.

deste ponto (x, y) na matriz caracteriza o nível de cinza do mesmo. A representação da imagem digital pode ser visualizada acima pela Figura 1.1.

Definição 1.5. (Contorno)

É a linha que “fecha” ou “limita” exteriormente uma região.

Definição 1.6. (Desoclusão)

É o processo de recuperar partes “escondidas” ou estragadas/faltantes de um objeto ou de uma área qualquer em uma imagem digital.

Definição 1.7. (Espaço Escala)

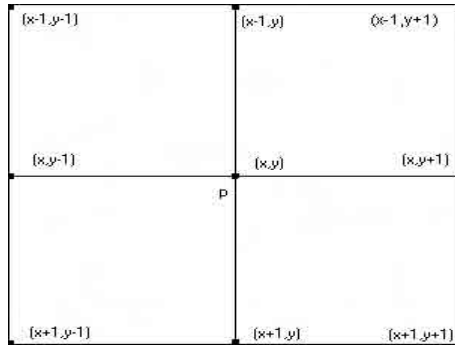
Quando nos referimos ao “tempo” t , estamos expressando um índice t da seqüência gerada pelas consecutivas reconstruções das imagens realizadas pelos modelos de restauração. Logo, a noção de tempo está diretamente ligada à evolução da imagem no processo de reconstrução, isto é, espaço escala.

Definição 1.8. (“Pixel”)

É o elemento básico da imagem (“picture elements”), que no contexto da representação da imagem digital é cada elemento da matriz.

Definição 1.9. (Vizinhança de um “Pixel”)

Um “pixel” qualquer, p , de uma imagem possui “pixels” vizinhos. Suponhamos que este “pixel” esteja na posição (x, y) e também não se encontra na borda da imagem. Sendo assim, este “pixel” possui “pixels” vizinhos, os quais serão: 2 vizinhos horizontais, 2 vizinhos verticais e 4 vizinhos diagonais, como mostramos a seguir:



Definição 1.10. (Gradiente)

Para uma função $f(x, y)$ o gradiente de f nas coordenadas (x, y) é definido como um vetor

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

e a magnitude deste vetor é dada por:

$$|\nabla f| = \left[\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2 \right]^{\frac{1}{2}},$$

sendo que este valor equivale à maior taxa de aumento de $f(x, y)$ por unidade de distância na direção de ∇f e baseia-se na obtenção das derivadas parciais $\frac{\partial f}{\partial x}$ e $\frac{\partial f}{\partial y}$ na posição de cada pixel (x, y) .

O operador gradiente é bastante utilizado na área de processamento de imagens digitais e seu objetivo é basicamente detectar bordas.

Definição 1.11. (Derivada Direcional)

Seja $f: A \subset \mathbb{R}^n \rightarrow \mathbb{R}$ uma função diferenciável, $a \in A$ e $v \in \mathbb{R}^n$ um vetor unitário. Se existe o limite

$$\lim_{t \rightarrow 0} \frac{f(a + tv) - f(a)}{t},$$

ele é chamado derivada direcional de f em a na direção de v . A derivada é denotada por $\frac{\partial f}{\partial v}(a)$.

Na prática, a derivada direcional de f na direção do vetor v pode ser calculada pela fórmula

$$\frac{\partial f}{\partial v} = \text{grad } v \quad \text{ou} \quad \frac{\partial f}{\partial v} = \nabla v.$$

Um outro conceito bastante utilizado em processamento de imagens é o de operador morfologicamente invariante.

Definição 1.12. (Morfologicamente Invariante)

Seja u_0 um dado observado (para retoque digital é a “máscara”, ou seja, região de retoque). Um operador $T : u = T(u_0)$ é dito ser morfologicamente invariante, se para alguma função estritamente crescente $g : [0, 1] \rightarrow [0, 1]$, tem-se $T(g(u_0)) = g(T(u_0))$.

Definição 1.13. (Função Gaussiana)

Seja $g : \mathbb{R}^n \rightarrow \mathbb{R}$. O espaço escala Gaussiano de g é uma função $T_g : \mathbb{R}^n \times \mathbb{R}_+ \rightarrow \mathbb{R}$, dada por

$$T_g(x, t) = g * G_t(x),$$

em que

$$G_t(x) = \frac{1}{(2\pi at)^{\frac{n}{2}}} e^{\frac{-(x_1^2 + \dots + x_n^2) - \mu}{2at}}$$

é a distribuição Gaussiana de dimensão n com média μ e variância at , onde a é uma constante positiva. O desvio-padrão é $\sigma = \sqrt{at}$ e $x = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$. O parâmetro t é chamado de escala.

Terminamos este capítulo enfatizando a importância dos conceitos introduzidos anteriormente para descrição dos modelos matemáticos que serão utilizados neste trabalho.

No próximo capítulo, faremos um estudo dos métodos numéricos para EDO, visando aproximar as EDP resultantes dos modelos de Retoque Digital.

Capítulo 2

Métodos Numéricos para Resolução de Equações Diferenciais Ordinárias (EDO)

Grande parte dos modelos matemáticos envolvem a resolução de equações diferenciais, tendo como incógnitas funções de uma ou mais variáveis relacionadas às suas derivadas. Em muitas situações não existem soluções analíticas exatas para essas equações, somente para casos especiais. Por essa falta de soluções analíticas para equações diferenciais, os métodos numéricos computacionais têm atuado como uma importante ferramenta. Sendo assim, são buscadas aproximações numéricas, que resultam em um conjunto solução discreto, isto é, aproximamos o valor da solução em pontos discretos do domínio.

Neste capítulo fazemos uma abordagem dos métodos numéricos para resolução de EDO.

2.1 O Método de Diferenças Finitas

A idéia básica do método das diferenças finitas é transformar o problema de encontrar a solução de uma equação diferencial em um problema de resolver um sistema de equações algébricas, sendo que para isto as derivadas presentes nas equações

são aproximadas por diferenças entre valores da solução discretizada. A ferramenta matemática principal para essa realização é a série de Taylor. Para o caso unidimensional, temos as seguintes definições:

Definição 2.1. (Cunha, M.C.C.) *Seja $y(x)$ uma função que admite derivada até ordem 2. Então a fórmula avançada para o cálculo de $y'(x)$ é dada por:*

$$y'(x) = \frac{y(x+h) - y(x)}{h} - \frac{h}{2}y''(\xi),$$

onde o erro cometido é dado por $-\frac{h}{2}y''(\xi)$, $x < \xi < x+h$.

Definição 2.2. (Cunha, M.C.C.) *Seja $y(x)$ uma função que admite derivada até ordem 2. Então a fórmula atrasada para o cálculo de $y'(x)$ é dada por:*

$$y'(x) = \frac{y(x) - y(x-h)}{h} - \frac{h}{2}y''(\xi),$$

onde o erro cometido é dado por $-\frac{h}{2}y''(\xi)$, $x-h < \xi < x$.

Definição 2.3. (Cunha, M.C.C.) *Seja $y(x)$ uma função que admite derivada até ordem 3. Então a fórmula centrada para o cálculo de $y'(x)$ é dada por:*

$$y'(x) = \frac{y(x+h) - y(x-h)}{2h} - \frac{h^2}{3!}y'''(\xi),$$

onde o erro cometido é dado por $-\frac{h^2}{3!}y'''(\xi)$, $x-h < \xi < x+h$.

Além disso, necessitamos definir uma malha, ou seja, um conjunto finito de pontos do domínio. Assim, para o caso unidimensional temos:

Definição 2.4. (Cunha, M.C.C.) *Seja x_0 um ponto de referência e h um número positivo. A malha de passo h associada a x_0 é constituída pelo seguinte conjunto de pontos:*

$$x_i = x_0 \pm ih, \quad i = 1, 2, \dots$$

Neste estudo, trabalhamos com imagens bidimensionais, ou seja, $n = 2$. Procuramos então, a solução $u(x)$, $x \in \mathbb{R}^2$. Dessa forma, discretizamos o domínio $u(x, y)$, isto é, a região Ω , em uma malha bidimensional de pontos:

$$(x_i, y_i) = (x_0 \pm ih, y_0 \pm ik), \quad i, j = 1, 2, \dots$$

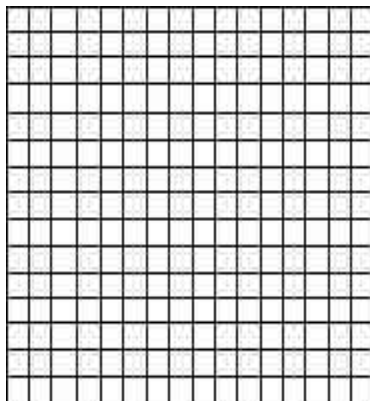


Figura 2.1: Malha Regular.

Particularmente, tomando $k = h$, a malha é considerada regular em (x, y) , [ver Figura 2.1].

Assim, generalizando as definições anteriores para o caso bidimensional, podemos obter as seguintes equações de diferenças relativas às derivadas parciais da função $u(x, y)$:

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial x} = u_x \simeq \frac{u(x_i+h, y_j) - u(x_i, y_j)}{h} = \frac{u_{i+h, j} - u_{i, j}}{h} \quad \text{fórmula avançada} \\ \frac{\partial u}{\partial x} = u_x \simeq \frac{u(x_i, y_j) - u(x_i-h, y_j)}{h} = \frac{u_{i, j} - u_{i-h, j}}{h} \quad \text{fórmula atrasada} \\ \frac{\partial u}{\partial x} = u_x \simeq \frac{u(x_i+h, y_j) - u(x_i-h, y_j)}{2h} = \frac{u_{i+h, j} - u_{i-h, j}}{2h} \quad \text{fórmula centrada} \end{array} \right.$$

$$\left\{ \begin{array}{l} \frac{\partial u}{\partial y} = u_y \simeq \frac{u(x_i, y_j+h) - u(x_i, y_j)}{h} = \frac{u_{i, j+h} - u_{i, j}}{h} \quad \text{fórmula avançada} \\ \frac{\partial u}{\partial y} = u_y \simeq \frac{u(x_i, y_j) - u(x_i, y_j-h)}{h} = \frac{u_{i, j} - u_{i, j-h}}{h} \quad \text{fórmula atrasada} \\ \frac{\partial u}{\partial y} = u_y \simeq \frac{u(x_i, y_j+h) - u(x_i, y_j-h)}{2h} = \frac{u_{i, j+h} - u_{i, j-h}}{2h} \quad \text{fórmula centrada} \end{array} \right.$$

$$\begin{aligned} \frac{\partial u}{\partial x \partial y} &= u_{xy} = (u_x)_y \simeq \frac{(u_x)_{j+h} - (u_x)_{j-h}}{2} \\ &\simeq \frac{\frac{u_{i+h, j+h} - u_{i-h, j+h}}{2h} - \frac{u_{i+h, j-h} - u_{i-h, j-h}}{2h}}{2h} \\ &= \frac{u_{i+h, j+h} - u_{i-h, j+h} - u_{i+h, j-h} + u_{i-h, j-h}}{4h} \\ &= \frac{u_{i+h, j+h} + u_{i-h, j-h} - u_{i-h, j+h} - u_{i+h, j-h}}{4h} \end{aligned}$$

$$\frac{\partial^2 u}{\partial x^2} = u_{xx} \simeq \frac{u_{i+h,j} - 2u_{i,j} + u_{i-h,j}}{h^2}$$

$$\frac{\partial^2 u}{\partial y^2} = u_{yy} \simeq \frac{u_{i,j+h} - 2u_{i,j} + u_{i,j-h}}{h^2}$$

Nas próximas seções, apresentamos alguns dos métodos numéricos mais conhecidos, na literatura, e que serão utilizados no decorrer deste trabalho.

2.2 O Método de Euler

O objetivo do método é aproximar um problema de valor inicial bem posto

$$y' = f(x, y), \quad a \leq x \leq b, \quad y(a) = \alpha. \quad (2.1)$$

Na verdade, uma aproximação contínua para a solução $y(x)$ não é realizada, mas encontramos aproximações para os pontos de y . Uma vez que a solução aproximada é obtida nos pontos, a solução aproximada nos outros pontos do intervalo podem ser feitas por interpolação.

Neste trabalho, assumimos que os pontos da malha são distribuídos igualmente no intervalo $[a, b]$. Esta condição é garantida pela escolha de um inteiro positivo N e selecionando os pontos da malha x_0, x_1, \dots, x_N como:

$$x_i = a + ih, \quad \text{para } i = 0, 1, \dots, N. \quad (2.2)$$

A distância comum entre os pontos, $h = (b - a)/N$, é chamada de tamanho do passo. Sendo assim, para obter o método de Euler, usamos o Teorema de Taylor.

Suponha que $y(x)$, a solução única de (2.1), admite derivada até segunda ordem em $[a, b]$, então para cada $i = 0, 1, \dots, N - 1$,

$$y(x_{i+1}) = y(x_i) + (x_{i+1} - x_i) y'(x_i) + \frac{(x_{i+1} - x_i)^2}{2} y''(\xi_i)$$

com ξ_i em (x_i, x_{i+1}) . Se $h = x_{i+1} - x_i$, então

$$y(x_{i+1}) = y(x_i) + h y'(x_i) + \frac{h^2}{2} y''(\xi_i)$$

e, desde que $y(x)$ satisfaz a equação diferencial (2.1),

$$y(x_{i+1}) = y(x_i) + h f(x_i, y(x_i)) + \frac{h^2}{2} y''(\xi_i). \quad (2.3)$$

O método de Euler tem como objetivo aproximar $y(x_i)$ por w_i , para cada $i = 1, 2, \dots, N$, anulando o termo restante. Então:

$$\begin{cases} w_0 = \alpha \\ w_{i+1} = w_i + h f(x_i, w_i), \quad \text{para cada } i = 1, 2, \dots, N-1. \end{cases} \quad (2.4)$$

A equação (3.6) é chamada de equação diferença associada ao método de Euler.

Note que quando w_i é uma aproximação de $y(x_i)$, a suposição que o problema é bem posto implica que

$$f(x_i, w_i) \approx y'(x_i) = f(x_i, y(x_i)).$$

Apresentaremos agora dois lemas, os quais serão utilizados para a prova do próximo teorema referente às aproximações pelo método de Euler. Tais resultados podem ser encontrados em [7].

Lema 2.1. *Para todo $x \geq -1$ e m positivo,*

$$0 \leq (1+x)^m \leq e^{mx}.$$

Demonstração: Aplicando o Teorema de Taylor com $f(x) = e^x$, $x_0 = 0$ e $n = 1$, obtemos

$$e^x = 1 + x + \frac{1}{2}x^2 e^\xi,$$

onde ξ está entre x e zero. Então,

$$0 \leq 1 + x + \frac{1}{2}x^2 e^\xi = e^x$$

e, desde que $1+x \geq 0$,

$$0 \leq (1+x)^m \leq (e^x)^m = e^{mx}.$$

■

Lema 2.2. Se s e t são números reais e positivos, $\{a_i\}_{i=0}^k$ é uma seqüência satisfazendo $a_0 \geq -t/s$, e

$$a_{i+1} \leq (1+s)a_i + t, \quad \text{para cada } i = 0, 1, \dots, k, \quad (2.5)$$

então

$$a_{i+1} \leq e^{(i+1)s} \left(\frac{t}{s} + a_0 \right) - \frac{t}{s}.$$

Demonstração: Para um inteiro i fixado, a desigualdade (2.5) implica que

$$\begin{aligned} a_{i+1} &\leq (1+s)a_i + t \\ &\leq (1+s)[(1+s)a_{i-1} + t] + t \\ &\leq (1+s)\{(1+s)[(1+s)a_{i-2} + t] + t\} + t \\ &\vdots \\ &\leq (1+s)^{i+1} + [1 + (1+s) + (1+s)^2 + \dots + (1+s)^i] + t. \end{aligned}$$

Mas, $1 + (1+s) + (1+s)^2 + \dots + (1+s)^i = \sum_{j=0}^i (1+s)^j$ é uma série geométrica com raio $(1+s)$ e, sendo assim, podemos dizer que

$$\frac{1 - (1+s)^{i+1}}{1 - (1+s)} = \frac{1}{s} [(1+s)^{i+1} - 1].$$

Então,

$$a_{i+1} \leq (1+s)^{i+1} a_0 + \frac{(1+s)^{i+1} - 1}{s} t = (1+s)^{i+1} \left(\frac{t}{s} + a_0 \right) - \frac{t}{s},$$

e pelo lema (2.1)

$$a_{i+1} \leq e^{(i+1)s} \left(\frac{t}{s} + a_0 \right) - \frac{t}{s}.$$

■

Passamos então ao resultado principal.

Teorema 2.1. *Suponhamos que f é contínua e satisfaz a condição de Lipschitz com a constante arbitrária L em*

$$D = \{(x, y) \mid a \leq x \leq b, -\infty < y < \infty\},$$

e existe $M > 0$ tal que

$$|y''(x)| \leq M, \quad \text{para todo } x \in [a, b].$$

Se $y(x)$ denota a solução única para o problema de valor inicial

$$y' = f(x, y), \quad a \leq x \leq b, \quad y(a) = \alpha,$$

e w_0, w_1, \dots, w_N são as aproximações geradas pelo método de Euler para algum inteiro N , então

$$|y(x_i) - w_i| \leq \frac{hM}{2L} [e^{L(x_i-a)} - 1], \quad \text{para cada } i = 0, 1, \dots, N.$$

Demonstração: Quando $i = 0$, o resultado é claramente verdadeiro, desde que $y(x_0) = w_0 = \alpha$. Da equação (2.3), temos para $i = 0, 1, \dots, N - 1$,

$$y(x_{i+1}) = y(x_i) + h f(x_i, y(x_i)) + \frac{h^2}{2} y''(\xi_i);$$

e da equação (2.4)

$$w_{i+1} = w_i + h f(x_i, w_i).$$

Conseqüentemente, usando a notação $y_i = y(x_i)$ e $y_{i+1} = y(x_{i+1})$,

$$y_{i+1} - w_{i+1} = y_i - w_i + h [f(x_i, y_i) - f(x_i, w_i)] + \frac{h^2}{2} y''(\xi_i)$$

e

$$|y_{i+1} - w_{i+1}| \leq |y_i - w_i| + h |f(x_i, y_i) - f(x_i, w_i)| + \frac{h^2}{2} |y''(\xi_i)|.$$

Como f satisfaz a condição de Lipschitz com a constante L e $|y''(x)| \leq M$, temos

$$|y_{i+1} - w_{i+1}| \leq |y_i - w_i| + (1 + hL) + \frac{h^2 M}{2}.$$

Aplicando o Lema 2.2 e fazendo $a_j = |y_j - w_j|$ para $j = 0, 1, \dots, N$, enquanto $s = hL$ e $t = h^2M/2$, obtemos

$$|y_{i+1} - w_{i+1}| \leq e^{(i+1)hL} \left(|y_0 - w_0| + \frac{h^2M}{2hL} \right) - \frac{h^2M}{2hL}.$$

Desde que $|y_0 - w_0| = 0$ e $(i+1)h = x_{i+1} - x_0 = x_{i+1} - a$, temos

$$|y_{i+1} - w_{i+1}| \leq \frac{hM}{2L} (e^{(x_{i+1}-a)L} - 1)$$

para cada $i = 0, 1, \dots, N-1$. ■

O Teorema 2.1, nos dá um limitante para o erro de truncamento do método de Euler. A principal importância dessa fórmula é que o limitante depende linearmente do tamanho do passo h . Conseqüentemente, diminuindo o tamanho do passo obtemos uma melhor precisão nas aproximações.

No resultado do Teorema 2.1, foi omitido o efeito que o erro de arredondamento representa na escolha do tamanho do passo, ou seja, à medida que diminuimos h , mais cálculos são necessários. E quando efetuamos mais cálculos, o erro vai acumulando gradativamente, podendo assim influenciar no erro de arredondamento final. Na realidade, então, a forma da equação diferença:

$$\begin{cases} w_0 = \alpha \\ w_{i+1} = w_i + h f(x_i, w_i), \quad \text{para cada } i = 1, 2, \dots, N-1, \end{cases} \quad (2.6)$$

não é usada para calcular a aproximação da solução y_i em cada ponto da malha x_i . Na verdade, a equação correta é da forma

$$\begin{cases} u_0 = \alpha + \delta_0, \\ u_{i+1} = u_i + h f(x_i, u_i) + \delta_{i+1}, \quad \text{para cada } i = 0, 1, \dots, N-1, \end{cases} \quad (2.7)$$

onde δ_i denota o erro de arredondamento associado a u_i . Utilizando passos similares aos da demonstração do Teorema 2.1, podemos obter o próximo resultado, que fornece uma fórmula mais adequada para um limitante para o erro das aproximações de y_i obtidas através do método de Euler.

Teorema 2.2. *Seja $y(x)$ a notação da solução única para o problema de valor inicial*

$$y' = f(x, y), \quad a \leq x \leq b, \quad y(a) = \alpha \quad (2.8)$$

e u_0, u_1, \dots, u_N as aproximações obtidas usando (2.7).

Se $|\delta_i| < \delta$, δ arbitrário, para cada $i = 0, 1, \dots, N$, e as hipóteses do Teorema 2.1 são válidas para (2.8), então

$$|y(x_i) - u_i| \leq \frac{1}{L} \left(\frac{hM}{2} + \frac{\delta}{h} \right) [e^{L(x_i-a)} - 1] + |\delta_0| e^{L(x_i-a)}, \quad (2.9)$$

para cada $i = 0, 1, \dots, N$.

O limitante do erro (2.9) não é mais linear em h , desde que

$$\lim_{h \rightarrow 0} \left(\frac{hM}{2} + \frac{\delta}{h} \right) = \infty.$$

Podemos usar o Cálculo para determinar um limitante inferior para o tamanho do passo h . Sendo $E(h) = \frac{hM}{2} + \frac{\delta}{h}$ implica que

$$E'(h) = \frac{M}{2} - \frac{\delta}{h^2}.$$

Se $h > \sqrt{\frac{2\delta}{M}}$, então $E'(h) < 0$ e $E(h)$ é decrescente.

Se $h < \sqrt{\frac{2\delta}{M}}$, então $E'(h) > 0$ e $E(h)$ é crescente.

O valor mínimo de $E(h)$ ocorre quando

$$h = \sqrt{\frac{2\delta}{M}}.$$

Normalmente, o valor de δ é suficientemente pequeno, para que este limitante de h não afete a operação do método de Euler.

2.2.1 Estabilidade do Método de Euler

Com o avanço tecnológico, aumentou-se a procura de processos computacionais para se resolver equações diferenciais. Logo, com o auxílio dos computadores, podemos observar que alguns algoritmos apresentam soluções com erros grandes e, em alguns casos, afastam-se cada vez mais da solução analítica. Isso contradiz, o pressuposto de que a discretização se aproxima do contínuo quando o passo tende a zero e caracteriza a instabilidade do esquema numérico.

Hoje temos em mãos várias maneiras de analisar e definir a estabilidade dos métodos numéricos, assim mostramos o procedimento mais comum.

Essa discretização é testada a partir do problema padrão:

$$\begin{cases} y'(x) = q y(x) \\ y(0) = 1, \end{cases}$$

onde q é uma constante negativa. O problema padrão é simples, e sua solução analítica é: $y(x) = \exp(qx)$. Observamos que $\lim_{x \rightarrow \infty} y(x) = 0$, pois $q < 0$. Portanto as aproximações numéricas devem satisfazer esta condição no limite.

Aplicando o Método de Euler, temos:

$$\frac{y_{n+1} - y_n}{h} = q y_n,$$

ou ainda,

$$y_{n+1} = y_n (1 + qh).$$

Como $y_0 = 1$, então:

$$y_1 = (1 + qh), \quad y_2 = (1 + qh)^2, \quad \dots, \quad y_n = (1 + qh)^n.$$

Notamos que, se $|1 + qh| > 1$, os valores de y_n , que seriam as aproximações para $y(x_n)$, crescem exponencialmente, isto é, $y_n \rightarrow \infty$ quando $n \rightarrow \infty$.

Temos então, uma condição de estabilidade $|1 + qh| < 1$ que deve ser respeitada se quisermos que as aproximações acompanhem, pelo menos qualitativamente, a solução analítica.

2.3 Métodos lineares de passo múltiplo

Antes da idéia de estender o método de Euler, o qual permite encontrar a solução aproximada em um ponto, Bashforth e Adams (1883) propuseram um método relacionando a informação disponível de pontos anteriores. Baseados nesta idéia, começaram a surgir os métodos lineares de passo múltiplo, sendo que a teoria moderna dos métodos lineares de passo múltiplo foi desenvolvida por Dahlquist (1956) e tornou-se amplamente conhecida desde a descoberta por Henrici (1962, 1963).

2.3.1 A forma geral dos métodos lineares de passo múltiplo

Os métodos lineares de passo múltiplo calculam y_n , como sendo uma aproximação para $y(x_n)$, ou seja, como uma combinação linear de y_{n-1} , y_{n-2} , ..., y_{n-k} e do cálculo obtido pela função f nos pontos y_n , y_{n-1} , ..., y_{n-k} . Então:

$$y_n = \alpha_1 y_{n-1} + \alpha_2 y_{n-2} + \dots + \alpha_k y_{n-k} + h [\beta_0 f(y_n) + \beta_1 f(y_{n-1}) + \dots + \beta_k f(y_{n-k})] \quad (2.10)$$

para números fixados $\alpha_1, \alpha_2, \dots, \alpha_k, \beta_0, \beta_1, \dots, \beta_k$.

Supomos que α_k e β_k sejam diferentes de zero, desde que o valor de k não possa ser reduzido e assim, o método obtido por (2.10) é conhecido como um método linear de passo k . Se o valor de β_0 é zero, o método é dito ser explícito; caso contrário é implícito. Para um método explícito, y_n pode ser calculado diretamente de (2.10), considerando que no caso implícito, podemos resolver uma equação da forma:

$$y_n - h \beta_0 f(y_n) = v, \quad (2.11)$$

onde o vetor v é independente de y_n e é dado por:

$$v = \sum_{i=1}^k [\alpha_i y_{n-i} + h \beta_i f(y_{n-i})].$$

Se f tem uma constante de Lipschitz pequena, então (2.11) pode ser resolvida pela iteração do ponto fixo, quando o limite da seqüência $z^{(0)}, z^{(1)}, \dots$, onde, para $j \geq 1$, $z^{(j)}$ é dado por:

$$z^{(j)} = v + h \beta_0 f(z^{(j-1)}).$$

No caso de problemas mais complicados, onde a constante de Lipschitz é necessariamente grande, este método iterativo não é convergente e é preciso usar alguma variação do método de Newton. Uma outra aproximação possível para o uso dos métodos implícitos é no contexto de pares previsor-corretor, os quais serão estudados nas próximas seções.

Agora, consideramos o problema

$$z'(x) = g(x, z(x)), \quad (2.12)$$

que pode ser escrito como

$$y'(x) = f(y(x)), \quad (2.13)$$

onde

$$f \begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} 1 \\ g(u, v) \end{pmatrix},$$

e

$$y(x) = \begin{pmatrix} x \\ z(x) \end{pmatrix}.$$

Se os valores exatos da primeira componente são dados em $x_{n-1}, x_{n-2}, \dots, x_{n-k}$, então aproximando (2.13) por (2.10) e escrevendo o resultado em termo destas componentes, temos:

$$x_n = \alpha_1 x_{n-1} + \alpha_2 x_{n-2} + \dots + \alpha_k x_{n-k} + h (\beta_0 + \beta_1 + \dots + \beta_k), \quad (2.14)$$

$$z_n = \alpha_1 z_{n-1} + \alpha_2 z_{n-2} + \dots + \alpha_k z_{n-k} + h [\beta_0 g(x_n, z_n) + \beta_1 g(x_{n-1}, z_{n-1}) + \dots + \beta_k g(x_{n-k}, z_{n-k})], \quad (2.15)$$

onde assumimos que a primeira componente é calculada exatamente como $x_n = x_{n-1} + h$.

Substituindo $x_{n-1} = x_n - h, x_{n-2} = x_n - 2h, \dots, x_{n-k} = x_n - kh$, em (2.14), encontramos:

$$x_n = (\alpha_1 + \alpha_2 + \dots + \alpha_k) x_n + h (\beta_0 + \beta_1 + \dots + \beta_k - \alpha_1 - 2\alpha_2 - \dots - k\alpha_k).$$

Para isto ser satisfeito, independentemente dos valores de x_n e h , devemos ter:

$$\alpha_1 + \alpha_2 + \dots + \alpha_k = 1, \quad (2.16)$$

$$\alpha_1 + 2\alpha_2 + \dots + k\alpha_k = \beta_0 + \beta_1 + \dots + \beta_k, \quad (2.17)$$

sendo que (2.16) e (2.17) são conhecidas como as condições de consistência.

Os métodos lineares implícitos de passo múltiplo mais conhecidos são:

i) Método de passo 2 de Adams-Moulton:

$$\begin{aligned} z_{i+1} &= z_i + \frac{h}{12} \left[5f(x_{i+1}, z_{i+1}) + 8f(x_i, z_i) - f(x_{i-1}, z_{i-1}) \right], \\ z_0 &= \alpha, \quad z_1 = \alpha_1, \end{aligned}$$

onde $i = 1, 2, \dots, N-1$. O erro de truncamento local é dado por $\tau_{i+1}(h) = -\frac{1}{24}z^{(4)}(\mu_i)h^3$, para $\mu_i \in (x_{i-1}, x_{i+1})$.

ii) Método de passo 3 de Adams-Moulton:

$$\begin{aligned} z_{i+1} &= z_i + \frac{h}{24} \left[9f(x_{i+1}, z_{i+1}) + 19f(x_i, z_i) - 5f(x_{i-1}, z_{i-1}) + f(x_{i-2}, z_{i-2}) \right], \\ z_0 &= \alpha, \quad z_1 = \alpha_1, \quad z_2 = \alpha_2, \end{aligned}$$

onde $i = 2, \dots, N-1$. O erro de truncamento local é dado por $\tau_{i+1}(h) = -\frac{19}{720}z^{(5)}(\mu_i)h^4$, para $\mu_i \in (x_{i-2}, x_{i+1})$.

iii) Método de passo 4 de Adams-Moulton:

$$\begin{aligned} z_{i+1} &= z_i + \frac{h}{720} \left[251f(x_{i+1}, z_{i+1}) + 646f(x_i, z_i) - 264f(x_{i-1}, z_{i-1}) + 106f(x_{i-2}, z_{i-2}) \right. \\ &\quad \left. - 19f(x_{i-3}, z_{i-3}) \right], \\ z_0 &= \alpha, \quad z_1 = \alpha_1, \quad z_2 = \alpha_2, \quad z_3 = \alpha_3, \end{aligned}$$

onde $i = 3, \dots, N-1$. O erro de truncamento local é dado por $\tau_{i+1}(h) = -\frac{3}{160}z^{(6)}(\mu_i)h^5$, para $\mu_i \in (x_{i-3}, x_{i+1})$.

Os métodos lineares explícitos de passo múltiplo mais conhecidos são:

i) Método de passo 2 de Adams-Bashforth:

$$\begin{aligned} z_{i+1} &= z_i + \frac{h}{2} \left[3f(x_i, z_i) - f(x_{i-1}, z_{i-1}) \right], \\ z_0 &= \alpha, \quad z_1 = \alpha_1, \end{aligned}$$

onde $i = 1, 2, \dots, N-1$. O erro de truncamento local é dado por $\tau_{i+1}(h) = \frac{5}{12}z^{(3)}(\mu_i)h^2$, para $\mu_i \in (x_{i-1}, x_{i+1})$.

ii) Método de passo 3 de Adams-Bashforth:

$$z_{i+1} = z_i + \frac{h}{12} \left[23f(x_i, z_i) - 16f(x_{i-1}, z_{i-1}) + 5f(x_{i-2}, z_{i-2}) \right],$$

$$z_0 = \alpha, \quad z_1 = \alpha_1, \quad z_2 = \alpha_2,$$

onde $i = 2, \dots, N - 1$. O erro de truncamento local é dado por $\tau_{i+1}(h) = -\frac{3}{8}z^{(4)}(\mu_i)h^3$, para $\mu_i \in (x_{i-2}, x_{i+1})$.

iii) Método de passo 4 de Adams-Bashforth:

$$z_{i+1} = z_i + \frac{h}{24} \left[55f(x_i, z_i) - 59f(x_{i-1}, z_{i-1}) + 37f(x_{i-2}, z_{i-2}) - 9f(x_{i-3}, z_{i-3}) \right],$$

$$z_0 = \alpha, \quad z_1 = \alpha_1, \quad z_2 = \alpha_2, \quad z_3 = \alpha_3,$$

onde $i = 3, \dots, N - 1$. O erro de truncamento local é dado por $\tau_{i+1}(h) = -\frac{251}{720}z^{(5)}(\mu_i)h^4$, para $\mu_i \in (x_{i-3}, x_{i+1})$.

2.3.2 Consistência, Estabilidade e Convergência

Nesta seção vamos introduzir o conceito de estabilidade para os métodos lineares de passo múltiplo, sendo que a estabilidade é uma condição necessária e suficiente para que os métodos consistentes convirjam. Passamos então a definição de estabilidade.

Definição 2.5. *O método dado por (2.10) é estável se todas as soluções da equação diferença*

$$y_n = \alpha_1 y_{n-1} + \alpha_2 y_{n-2} + \dots + \alpha_k y_{n-k}$$

são limitadas quando $n \rightarrow \infty$.

Seja o polinômio p definido por

$$p(z) = (z - z_1)(z - z_2)\dots(z - z_k) = z^k - \alpha_1 z^{k-1} - \dots - \alpha_k.$$

Agora definimos uma condição em $\alpha_1, \alpha_2, \dots, \alpha_k$ expressa em termos de z_1, z_2, \dots, z_k .

Definição 2.6. *O método dado por (2.10) é dito satisfazer a condição essencial, se $|z_1| \leq 1, |z_2| \leq 1, \dots, |z_k| \leq 1$ e se, para $i \neq j, z_i = z_j$, então $|z_i| < 1$.*

Usando a teoria de equações diferenças, temos que um método é estável se satisfaz a condição essencial.

O conceito de convergência é obviamente mais complicado do que nos método de passo um. Além de especificar os k valores iniciais para começar a integração numérica, especificamos através deles a dependência do valor de h . Se ϕ_n denota a relação funcional de y_0, y_1, \dots, y_{k-1} , estes valores iniciais e o valor de h , então

$$y_n = \phi_n(y_0, y_1, \dots, y_{k-1}, h).$$

Desde que a solução exata seja contínua e $y(x_i) \rightarrow y(x_0)$ quando $h \rightarrow 0$ para $i = 0, 1, 2, \dots, k-1$, é apropriado considerar regras para gerar valores iniciais y_0, y_1, \dots, y_{k-1} , tais que em cada um deles o limite é $y(x_0)$ quando $h \rightarrow 0$.

Definição 2.7. *O método (2.10) é dito ser convergente se para algum problema padrão no intervalo $[x_0, \bar{x}]$,*

$$\phi_n(y_0, y_1, \dots, y_{k-1}, (\bar{x} - x_0)/n) \rightarrow y(\bar{x})$$

quando $y_0, y_1, \dots, y_{k-1} \rightarrow y(x_0)$ e $n \rightarrow \infty$.

2.3.3 Condições de Ordem

Vamos definir, nesta seção, a ordem do erro de truncamento local para os métodos lineares de passo múltiplo.

Definição 2.8. *O método dado por (2.10) é de ordem m se (2.10) tende a zero quando y é um polinômio de grau m .*

Teorema 2.3. *O método dado por (2.10) é da ordem m se*

$$\alpha_1 + \alpha_2 + \dots + \alpha_k = 1, \tag{2.18}$$

$$\alpha_1 + 2\alpha_2 + \dots + k\alpha_k = \beta_0 + \beta_1 + \dots + \beta_k, \tag{2.19}$$

$$\alpha_1 + 2^j\alpha_2 + \dots + k^j\alpha_k = j(\beta_1 + 2^{j-1}\beta_2 + \dots + k^{j-1}\beta_k), \quad j = 2, 3, \dots, m. \tag{2.20}$$

Demonstração: Desde que os polinômios p_0, p_1, \dots, p_m definidos por $p_0(x) = 1, p_1(x) = (x - x_n), p_j(x) = (x - x_n)^j, (j = 2, 3, \dots, m)$ formam uma base para o espaço linear dos polinômios de grau m , é necessário e suficiente, para o método ser de ordem m , que (2.10) tende a zero em $y = p_0, y = p_1, y = p_j, j = 2, 3, \dots, m$. Estimando o erro de truncamento local para estes casos obtemos (2.18), (2.19) e (2.20).

Note que (2.18) e (2.19) são idênticos às condições de consistência, (2.16) e (2.17). Assim temos os seguintes resultados.

Corolário 2.1. *Um método linear de passo múltiplo é consistente se e somente se é de ordem 1.*

Teorema 2.4. *Se $y \in C^{m+1}[x_{n-k}, x_n]$ e o método dado por (2.10) é de ordem m , então o erro de truncamento local é $O(h^{m+1})$ quando $h \rightarrow 0$.*

Demonstração: Expandindo $y(x_{n-1}), y'(x_{n-1}), \dots, y(x_{n-k}), y'(x_{n-k})$, em série de Taylor em torno de x_n , temos

$$\begin{aligned} y(x) &= y(x_n)p_0(x) + \frac{y'(x_n)p_1(x)}{1!} + \dots + \frac{y^{(m)}(x_n)p_m(x)}{m!} \\ &+ \frac{y^{(m+1)}(\xi(x))p_{m+1}(x)}{(m+1)!}, \end{aligned} \quad (2.21)$$

$$\begin{aligned} y'(x) &= \frac{y'(x_n)p'_1(x)}{1!} + \frac{y''(x_n)p'_2(x)}{2!} + \dots + \frac{y^{(m)}(x_n)p'_m(x)}{m!} \\ &+ \frac{y^{(m+1)}(\eta(x))p'_{m+1}(x)}{(m+1)!}, \end{aligned} \quad (2.22)$$

onde p_0, p_1, \dots foram introduzidos na prova do Teorema (2.3) e $\xi(x)$ e $\eta(x)$ denotam números de valores significativos entre x_n e x . Se L_y denota a expressão (2.10) então, pela condição de ordem, $Lp_0 = Lp_1 = \dots = Lp_m = 0$. Entretanto, usando (2.21) e (2.22), encontramos:

$$\begin{aligned} L_y &= - \sum_{i=1}^k \alpha_i \frac{y^{(m+1)}(\xi(x_i))p_{m+1}(x_{n-i})}{(m+1)!} \\ &- h \sum_{i=1}^k \beta_i \frac{y^{(m+1)}(\eta(x_i))p'_{m+1}(x_{n-i})}{(m+1)!} \end{aligned}$$

sendo assim,

$$\|L_y\| \leq \sup_{t \in [x_{n-k}, x_n]} \|y^{(m+1)}(t)\| M h^{m+1},$$

onde

$$M = \frac{1}{(m+1)!} \sum_{i=1}^k |\alpha_i| i^{m+1} + \frac{1}{m!} \sum_{i=1}^k |\beta_i| i^m.$$

Portanto,

$$\|L_y\| = O(h^{m+1}), \quad \text{quando } h \rightarrow 0.$$

Na verdade, para obter métodos de uma ordem particular m , que não seja superior a $k+1$, podemos primeiramente escolher $\alpha_1, \alpha_2, \dots, \alpha_k$ satisfazendo (2.18) e então resolver para $\beta_0, \beta_1, \dots, \beta_k$, satisfazendo o sistema linear (2.19) e (2.20), com $j = 2, 3, \dots, m$. Por outro lado, dependendo da escolha de $\alpha_1, \alpha_2, \dots, \alpha_k$ podemos obter uma ordem maior que $k+1$. Isto pode ser verificado, se tomarmos $m = 2k$, pois o sistema linear com as incógnitas $\alpha_1, \alpha_2, \dots, \alpha_k, \beta_0, \beta_1, \dots, \beta_k$ consiste de (2.18), (2.19) e (2.20), para $j = 2, 3, \dots, 2k$ ser não singular e assim, podemos encontrar métodos com ordens melhores. Entretanto, para $k > 2$ tais métodos são instáveis. Isto foi mostrado por Dahlquist (1956), que $k+1$ é sempre a melhor ordem que pode ser atingida pelos métodos estáveis a menos que k seja par, e neste caso $k+2$ pode ser atingido.

2.4 Os Métodos de Runge-Kutta

A idéia de estender o método de Euler para possibilitar uma multiplicidade de avaliações da função f dentro de cada passo foi proposta originalmente por Runge (1895). Contribuições futuras foram feitas por Hein (1900) e por Kutta (1901). Kutta caracterizou completamente o conjunto dos métodos de Runge-Kutta de quarta ordem e propôs o primeiro método de quinta ordem. Os métodos especiais para equação diferencial de segunda ordem foram feitos por Nyström (1925) que também contribuiu para o desenvolvimento dos métodos de primeira ordem.

Com o avanço dos computadores, cresceu então o interesse nos métodos de Runge-Kutta, e um grande número de pesquisadores tem contribuído para a expansão da teoria e desenvolvimento particular dos métodos. Embora os estudos foram dedicados inteiramente para os métodos de Runge-Kutta explícitos, interessa agora estendê-los para os métodos implícitos, os quais são recomendados para equações diferenciais difíceis.

Consideramos o problema de valor inicial

$$y' = f(x, y), \quad y(a) = \eta.$$

De todos os métodos computacionais para a solução numérica deste problema, o mais fácil de implementar é o método de Euler, citado anteriormente,

$$y_{n+1} - y_n = h f(x_n, y_n) \equiv h f_n.$$

Este método é explícito e de passo um, não requer condições iniciais adicionais, permite uma mudança do comprimento do passo durante o cálculo e entretanto, sua ordem de convergência é baixa. Os métodos lineares de passo múltiplo têm ordem mais elevada e mantêm a linearidade com relação a y_{n+j} , f_{n+j} , $j = 0, 1, \dots, k$. Quando a ordem é elevada pode-se perder a linearidade, mas para os primeiros métodos propostos por Runge e subsequentemente desenvolvidos por Kutta e Heun, isso não acontece. Portanto os métodos de Runge-Kutta mantêm as vantagens dos métodos de passo um, mas devido à essa perda de linearidade a análise dos erros é consideravelmente mais difícil que no caso dos métodos lineares de passo múltiplo. Usamos, neste estudo, a frase “método de Runge-Kutta” que significa “método de Runge-Kutta explícito”. Sendo assim, um método de Runge-Kutta tem que ser considerado como um caso particular do método geral de passo um explícito

$$y_{n+1} - y_n = h \phi(x_n, y_n, h). \quad (2.23)$$

2.4.1 Ordem e Convergência do método geral de passo um explícito

Definição 2.9. O método (2.23) é dito ter ordem p se p é o maior inteiro tal que

$$y(x+h) - y(x) - h \phi(x, y(x), h) = O(h^{p+1}), \quad (2.24)$$

onde $y(x)$ é a solução teórica do problema de valor inicial (2.1).

Definição 2.10. O método (2.23) é dito ser consistente com o problema de valor inicial se

$$\phi(x, y, 0) \equiv f(x, y). \quad (2.25)$$

Se o método (2.23) é consistente com o problema de valor inicial (ou simplesmente “consistente”), então

$$y(x+h) - y(x) - h \phi(x, y(x), h) = h y'(x) - h \phi(x, y(x), 0) + O(h^2) = O(h^2) \quad (2.26)$$

desde que $y'(x) = f(x, y(x)) = \phi(x, y(x), 0)$, por (2.25). Então um método consistente tem ordem maior ou igual a um.

O único método linear de passo múltiplo que ocorre dentro da classe (2.23) é o método de Euler:

$$\phi(x, y, h) = \phi_E(x, y, h) \equiv f(x, y).$$

(O subscrito E denota Euler.) A condição de consistência (2.25) é então satisfeita e um simples cálculo mostra que a ordem está de acordo com (2.24).

O algoritmo de Taylor de ordem p também ocorre dentro da classe (2.23) e é obtido por

$$\phi(x, y, h) = \phi_T(x, y, h) \equiv f(x, y) + \frac{h}{2!} f'(x, y) + \dots + \frac{h^{p-1}}{p!} f^{(p-1)}(x, y). \quad (2.27)$$

onde

$$f^{(q)}(x, y) = \frac{d^q}{dx^q} f(x, y), \quad q = 1, 2, \dots, (p-1).$$

(O subscrito T denota Taylor.)

O próximo teorema afirma quais são as condições suficientes e necessárias para o método (2.23) ser convergente.

Teorema 2.5. *Se a função $\phi(x, y; h)$ é contínua na região definida por $x \in [a, b]$, $y \in (-\infty, \infty)$, $h \in [0, h_0]$, onde $h_0 > 0$, e existe uma constante L tal que*

$$|\phi(x, y^*; h) - \phi(x, y; h)| \leq L|y^* - y| \quad (2.28)$$

para todos os pontos $(x, y^*; h)$, $(x, y; h)$ na região definida, então a relação

$$\phi(x, y; 0) = f(x, y) \quad (2.29)$$

é uma condição necessária e suficiente para a convergência do método definido pela função incremento ϕ .

A condição (2.29) é chamada de condição de consistência. Resumindo, o teorema estabelece que a consistência é necessária e suficiente para a convergência.

Demonstração: Seja $\phi(x, y; 0) = g(x, y)$, onde a função $g(x, y)$ satisfaz as condições do teorema. Segue que para algum η o problema de valor inicial

$$z' = g(x, z), \quad z(a) = \eta \tag{2.30}$$

tem uma única solução $z(x)$. Assim, o ponto $(x, z(x)) \in R$ para $x \in [a, b]$, onde R denota a região compacta.

Devemos mostrar que os números z_n definidos por $z_0 = \eta$,

$$z_{n+1} = z_n + h \phi(x, z_n; h), \quad n = 0, 1, 2, \dots; \quad x_n \in [a, b] \tag{2.31}$$

convergem para $z(x)$. Subtraindo de (2.31) a relação $z(x_{n+1}) = z(x_n) + h \Delta(x_n, z(x_n); h)$ encontramos o erro $e_n = z_n - z(x_n)$. Logo,

$$e_{n+1} = e_n + h [\phi(x_n, z_n; h) - \Delta(x_n, z(x_n); h)].$$

Pelo Teorema do Valor Médio,

$$\Delta(x_n, z(x_n); h) = \frac{z(x_{n+1}) - z(x_n)}{h} = g(x_n + \theta h, z(x_n + \theta h)),$$

onde $0 < \theta < 1$. A expressão pode ser escrita como

$$\begin{aligned} \phi(x_n, z_n; h) &- \phi(x_n, z(x_n); h) + \phi(x_n, z(x_n); h) \\ &- \phi(x_n, z(x_n); 0) + g(x_n, z(x_n)) - g(x_n + \theta h, z(x_n + \theta h)). \end{aligned}$$

A função $\phi(x, y; h)$ é uniformemente contínua no conjunto compacto $x \in [a, b]$, $y = z(x)$, $0 \leq h \leq h_0$. Portanto, podemos concluir que a igualdade,

$$\zeta(h) = \max_{x \in [a, b]} |\phi(x, z(x); h) - \phi(x, z(x); 0)|$$

tende a zero quando $h \rightarrow 0$. Similarmente,

$$\mathcal{X}(h) = \max_{x \in [a, b]} |g(x, z(x)) - g(x + k, z(x + k))|$$

tende a zero quando $h \rightarrow 0$. Usando (2.28), obtemos a estimativa

$$|e_{n+1}| \leq |e_n| + h[L|e_n| + \zeta(h) + \mathcal{X}(h)].$$

Podemos estimar $|e_{n+1}| \leq A|e_n| + B$, (ver detalhes em [12]), e tomando $A = 1 + hL$, $B = h(\zeta(h) + \mathcal{X}(h))$, temos:

$$|e_n| \leq [\zeta(h) + \mathcal{X}(h)]E_L(x_n - a).$$

Desde que a expressão do lado direito tende a zero quando $h \rightarrow 0$, segue que

$$\lim_{h \rightarrow 0} z_n = z(x), \quad x \in [a, b].$$

Isto determina a condição de consistência (2.29) para a convergência. Se $g(x, y) = f(x, y)$ então $z(x) = y(x)$. Primeiramente, temos a necessidade de assumir que o método definido por $\phi(x, y; h)$ é convergente, mas $g(x, y) \neq f(x, y)$ em algum ponto (x, y) . Segue que existe um η tal que a solução $y(t)$ do problema de valor inicial passe pelo ponto (x, y) . Os valores z_n definidos por $z_0 = \eta$ e (2.31) tendem a $y(t)$ dos dois lados, o que foi provado anteriormente, para a solução $z(t)$ do problema de valor inicial (2.30). Se $z(x) \neq y(x)$, temos uma contradição imediata; se $z(x) = y(x)$, então

$$z'(x) = g(x, y) \neq f(x, y) = y'(x).$$

Logo, novamente, $z(t) \neq y(t)$. Portanto, a hipótese que $g(x, y) \neq f(x, y)$ leva à uma contradição. ■

2.4.2 Dedução dos métodos de Runge-Kutta clássicos

O método de Runge-Kutta tem como forma geral, a seguinte expressão:

$$y_{n+1} - y_n = h \phi(x_n, y_n, h) \tag{2.32}$$

com

$$\left. \begin{aligned} \phi(x, y, h) &= \sum_{r=1}^R c_r K_r \\ K_1 &= f(x, y), \\ K_r &= f(x + ha_r, y + h \sum_{s=1}^{r-1} b_{rs} K_s), \quad r = 2, 3, \dots, R \end{aligned} \right\} \tag{2.33}$$

e

$$a_r = \sum_{s=1}^{r-1} b_{rs}, \quad r = 2, 3, \dots, R. \tag{2.34}$$

Observe que um método de Runge-Kutta, no estágio R , envolve R avaliações da função por passo. Cada uma das funções $K_r(x, y, h)$, $r = 1, 2, \dots, R$ deve ser interpretada como uma aproximação para a derivada $y'(x)$, e a função $\phi(x, y, h)$ como parte importante desta aproximação. Do mesmo modo que nos métodos de passo múltiplo, a consistência demanda que $\sum_{r=1}^R c_r = 1$. Se podemos escolher valores para as constantes c_r, a_r, b_{rs} , tais que a expressão da função $\phi(x, y, h)$ definida por (2.33), na potência de h diverge da expansão $\phi_T(x, y, h)$ dada por (2.27) somente na p -ésima potências de h , então o método claramente tem ordem p . (Note que assumimos em (2.27) que $y(x) \in C^p[a, b]$.)

Há muita manipulação algébrica envolvida na dedução dos métodos de Runge-Kutta de ordem elevada, por essa razão, deduzimos apenas métodos até a ordem três e citamos alguns métodos bem conhecidos de quarta ordem.

Introduzindo uma notação simplificada

$$f = f(x, y), \quad f_x = \frac{\partial f(x, y)}{\partial x}, \quad f_{xx} = \frac{\partial^2 f(x, y)}{\partial x^2}, \quad f_{xy} = \frac{\partial^2 f(x, y)}{\partial x \partial y}, \quad \text{etc.}$$

e usando as expressões das derivadas totais de f , podemos escrever a expressão (2.27) na forma

$$\phi_T(x, y, h) = f + \frac{1}{2}hF + \frac{1}{6}h^2(Ff_y + G) + O(h^3) \quad (2.35)$$

onde

$$F = f_x + ff_y, \quad G = f_{xx} + 2ff_{xy} + f^2f_{yy}. \quad (2.36)$$

Isto implica que é possível obter ordem três com $R = 3$, e então necessitamos de expressões somente para as funções K_1 , K_2 e K_3 . Usando (2.33) e (2.34) temos:

$$\begin{aligned} K_1 &= f(x, y) = f \\ K_2 &= f(x + ha_2, y + ha_2K_1), \\ K_3 &= f(x + ha_3, y + h(a_3 - b_{32})K_1 + hb_{32}K_2). \end{aligned} \quad (2.37)$$

Expandindo K_2 como uma série de Taylor sobre o ponto (x, y) obtemos:

$$K_2 = f + ha_2(f_x + K_1f_y) + \frac{1}{2}h^2a_2^2(f_{xx} + 2K_1f_{xy} + K_1^2f_{yy}) + O(h^3).$$

Substituindo em K_1 e usando (2.36) encontramos:

$$K_2 = f + ha_2F + \frac{1}{2}h^2a_2^2G + O(h^3). \quad (2.38)$$

Tratando K_3 , similarmente obtemos:

$$\begin{aligned} K_3 &= f + h\{a_3f_x + [(a_3 - b_{32})K_1 + b_{32}K_2]f_y\} \\ &+ \frac{1}{2}h^2\{a_3^2f_{xx} + 2a_3[(a_3 - b_{32})K_1 + b_{32}K_2]f_{xy} + [(a_3 - b_{32})K_1 + b_{32}K_2]^2f_{yy}\} + O(h^3). \end{aligned}$$

Após algumas manipulações, utilizando K_1 e K_2 , chegamos em:

$$K_3 = f + ha_3F + h^2(a_2b_{32}Ff_y + \frac{1}{2}a_3^2G) + O(h^3). \quad (2.39)$$

Agora, substituindo (2.37), (2.38) e (2.39) em (2.33), as expressões seguem de $\phi(x, y, h)$, como definida no método (2.32), e obtemos:

$$\begin{aligned} \phi(x, y, h) &= (c_1 + c_2 + c_3)f + h(c_2a_2 + c_3a_3)F \\ &+ \frac{1}{2}h^2[2c_3a_2b_{32}Ff_y + (c_2a_2^2 + c_3a_3^2)G] + O(h^3). \end{aligned} \quad (2.40)$$

Agora temos que partir para as expressões (2.35) e (2.40). Como vimos, pode ser realizado com $R = 1, 2, 3$.

Para $R = 1$, $c_2 = c_3 = 0$, (2.40) se reduz a

$$\phi(x, y, h) = c_1f + O(h^3). \quad (2.41)$$

Fazendo $c_1 = 1$, (2.41) diverge da expansão (2.35) para ϕ_T pelo termo de ordem h . Então o método resultante, é o método de Euler, que tem ordem um.

Para $R = 2$, $c_3 = 0$, (2.40) se reduz a

$$\phi(x, y, h) = (c_1 + c_2)f + hc_2a_2F + \frac{1}{2}h^2c_2a_2^2G + O(h^3).$$

Se partirmos da expansão (2.35) temos, então, que satisfazer as equações

$$c_1 + c_2 = 1, \quad c_2a_2 = \frac{1}{2}. \quad (2.42)$$

Este é um conjunto de duas equações com três incógnitas e então há uma família de soluções com parâmetro livre. Entretanto, é claro que há solução nesta família que torna as expansões (2.35) e (2.27) divergentes para o termo de ordem mais

elevada que h^2 . Então existe um número infinito de métodos de Runge-Kutta, estágio 2, de ordem 2 e não de ordem maior que 2. Esta falta de unicidade é típica de todas as origens de Runge-Kutta. As duas soluções particulares de (2.42) produzem os métodos bem conhecidos:

(i) Para $c_1 = 0$, $c_2 = 1$, $a_2 = \frac{1}{2}$, o método resultante é:

$$y_{n+1} - y_n = hf(x_n + 1/2h, y_n + 1/2hf(x_n, y_n)),$$

que nos referimos como sendo o Método de Euler Modificado.

(ii) Para $c_1 = 1/2$, $c_2 = 1/2$, $a_2 = 1$, o método resultante é:

$$y_{n+1} - y_n = \frac{1}{2}h[f(x_n, y_n) + f(x_n + h, y_n + hf(x_n, y_n))],$$

que é conhecido como o Método de Euler Melhorado.

2.5 Método Previsor-Corretor

Para resolver um problema de valor inicial utilizando um método linear de passo k implícito, ou seja, usando:

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \sum_{j=0}^k \beta_j f_{n+j}, \quad \beta_k \neq 0,$$

em cada um dos estágios devemos resolver para y_{n+k} a equação:

$$y_{n+k} = h \beta_k f(x_{n+k}, y_{n+k}) + h \sum_{j=0}^{k-1} \beta_j f_{n+j} - \sum_{j=0}^{k-1} \alpha_j y_{n+j}$$

onde y_{n+j} , f_{n+j} , $j = 0, 1, \dots, k-1$ são conhecidos.

Para que este problema exista f tem que ser uma função não-linear em y e assim aplicamos o método iterativo:

$$y_{n+k}^{[i+1]} = h \beta_k f(x_{n+k}, y_{n+k}^{[i]}) + K, \quad i = 0, 1, \dots,$$

onde $y_{n+k}^{[0]}$ é arbitrário.

O valor de $y_{n+k}^{[0]}$ é calculado pelo método explícito:

$$y_{n+k}^{[0]} = h \sum_{j=0}^{k-1} \beta_j^* f_{n+j} - \sum_{j=0}^{k-1} \alpha_j^* y_{n+j}$$

que chamamos de previsor.

Após o cálculo deste valor, podemos através do método implícito obter $y_{n+k}^{[1]}$, $y_{n+k}^{[2]}$, ... e a esse método denominamos corretor. Portanto, temos então o método previsor-corretor.

Seja m um valor pré-fixado, onde m é o número de iterações para cada passo. Sendo assim, teríamos: $y_{n+k}^{[1]}$, $y_{n+k}^{[2]}$, ..., $y_{n+k}^{[m]}$, onde $y_{n+k}^{[m]} \equiv y_{n+k}$ que é a solução obtida pelo método implícito.

Depois de escolher o par previsor-corretor, fazemos o processo de parada, isto é:

a) $|y_{n+k}^{[i+1]} - y_{n+k}^{[i]}| < \epsilon$, onde ϵ é um número pré-fixado.

b) Caso contrário de (a), fixamos um número m de iterações do corretor.

Geralmente, utilizamos $m = 2$ e assim utilizamos o critério de parada (b).

Por simplicidade, denotamos por:

P: aplicação do previsor;

C: aplicação do corretor;

E: cálculo da $f(x_{n+k}, y_{n+k})$.

E assim, se $m = 1$, encontramos:

$$\left. \begin{array}{l} y_{n+k}^{[0]} \rightarrow \text{P} \\ f_{n+k}^{[0]} = f(x_{n+k}, y_{n+k}^{[0]}) \rightarrow \text{E} \\ y_{n+k}^{[1]} \rightarrow \text{C} \end{array} \right\} \therefore PEC \quad (2.43)$$

Como $m = 1$, paramos aqui. Agora, se ao invés disto, tomarmos $m = 2$, temos:

$$\left. \begin{array}{l} y_{n+k}^{[0]} \rightarrow \text{P} \\ f_{n+k}^{[0]} = f(x_{n+k}, y_{n+k}^{[0]}) \rightarrow \text{E} \\ y_{n+k}^{[1]} \rightarrow \text{C} \\ f_{n+k}^{[1]} = f(x_{n+k}, y_{n+k}^{[1]}) \rightarrow \text{E} \\ y_{n+k}^{[2]} \rightarrow \text{C} \end{array} \right\} \therefore PECEC = P(EC)^2 \quad (2.44)$$

Aplicando o mesmo processo m vezes para a função f e m vezes para o corretor temos $P(EC)^m$.

Para esse caso, $m = 2$, após calcular $y_{n+k}^{[2]} \equiv y_{n+k}$ temos que determinar

y_{n+k+1} , ou seja:

$$\begin{aligned} y_{n+k+1}^{[0]} &\rightarrow P \\ f_{n+k+1}^{[0]} &\rightarrow E \\ y_{n+k+1}^{[1]} &\rightarrow C \\ f_{n+k+1}^{[1]} &\rightarrow E \\ y_{n+k+1}^{[2]} &\rightarrow C \Rightarrow y_{n+k+1}^{[2]} = y_{n+k+1} \end{aligned}$$

e assim por diante.

Se calcularmos $f_{n+k}^{[2]}$ antes de $y_{n+k+1}^{[0]}$ então temos $P(EC)^2E$, ou seja, temos dois modos diferentes:

1° modo - $P(EC)^m$;

2° modo - $P(EC)^mE$.

2.5.1 Algoritmo Previsor-Corretor ($m \geq 1$)

Usamos:

$$\begin{aligned} \sum_{j=0}^k \alpha_j^* y_{n+j} &= h \sum_{j=0}^{k-1} \beta_j^* f_{n+j} \rightarrow \text{Previsor} \\ \sum_{j=0}^k \alpha_j y_{n+j} &= h \sum_{j=0}^k \beta_j f_{n+j} \rightarrow \text{Corretor} \end{aligned}$$

1° modo - $P(EC)^m$

$$\left. \begin{aligned} y_{n+k}^{[0]} + \sum_{j=0}^{k-1} \alpha_j^* y_{n+j}^{[m]} &= h \sum_{j=0}^{k-1} \beta_j^* f_{n+j}^{[m-1]} \\ f_{n+k}^{[i]} &= f(x_{n+k}, y_{n+k}^{[i]}) \\ y_{n+k}^{[i+1]} + \sum_{j=0}^{k-1} \alpha_j y_{n+j}^{[m]} &= h \beta_k f_{n+k}^{[i]} + h \sum_{j=0}^{k-1} \beta_j f_{n+j}^{[m-1]} \end{aligned} \right\} i = 0, 1, \dots, m-1$$

2° modo - $P(EC)^mE$

$$\left. \begin{aligned} y_{n+k}^{[0]} + \sum_{j=0}^{k-1} \alpha_j^* y_{n+j}^{[m]} &= h \sum_{j=0}^{k-1} \beta_j f_{n+j}^{[m]} \\ f_{n+k}^{[i]} &= f(x_{n+k}, y_{n+k}^{[i]}) \\ y_{n+k}^{[i+1]} + \sum_{j=0}^{k-1} \alpha_j y_{n+j}^{[m]} &= h \beta_k f_{n+k}^{[i]} + h \sum_{j=0}^{k-1} \beta_j f_{n+j}^{[m]} \end{aligned} \right\} i = 0, 1, \dots, m-1$$

$$f_{n+k}^{[m]} = f(x_{n+k}, y_{n+k}^{[m]})$$

2.5.2 Erro de Truncamento Local do Método Previsor-Corretor

Consideremos

$$\sum_{j=0}^k \alpha_j^* y_{n+j} = h \sum_{j=0}^{k-1} \beta_j^* f_{n+j},$$

como sendo o previsor com ordem q^* e uma constante $C_{q^*+1}^*$, e também

$$\sum_{j=0}^k \alpha_j y_{n+j} = h \sum_{j=0}^k \beta_j f_{n+j},$$

como corretor com ordem q e uma constante C_{q+1} .

Vamos encontrar expressões para o erro de truncamento local principal (ETLP) em x_{n+k} do método previsor-corretor definido por P e C nos modos $P(EC)^m$ e $P(EC)^m E$, assumindo que a solução numérica nos pontos x_{n+j} , $j = 0, 1, \dots, k-1$, são exatas. Além disso, admitimos também que a solução teórica $y(x)$ do problema de valor inicial é suficientemente diferenciável.

Seja \mathcal{L}^* o operador diferença associado ao previsor. Assim:

$$\mathcal{L}^*[y(x_n), h] = y(x_{n+k}) - y_{n+k}^{[0]} = C_{q^*+1}^* h^{q^*+1} y^{(q^*+1)}(x_n) + O(h^{q^*+2})$$

onde $C_{q^*+1}^* h^{q^*+1} y^{(q^*+1)}(x_n) = ETLP$.

Analogamente, seja \mathcal{L} o operador diferença associado ao corretor, isto é:

$$\mathcal{L}[y(x_n), h] = y(x_{n+k}) - y_{n+k}^{[m]} = C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2})$$

onde $C_{q+1} h^{q+1} y^{(q+1)}(x_n) = ETLP$.

Passamos agora a estudar o ETLP do par previsor-corretor.

O previsor é definido por:

$$y_{n+k}^{[i+1]} + \sum_{j=0}^{k-1} \alpha_j y_{n+j}^{[m]} = h \beta_k f_{n+k}^{[i]} + h \sum_{j=0}^{k-1} \beta_j f_{n+j}^{[m-s]} \quad (2.45)$$

com $s = 0$ no modo $P(EC)^m E$ e $s = 1$ no modo $P(EC)^m$. E ainda, temos para o

corretor

$$\begin{aligned}
 \mathcal{L}[y(x_n), h] &= \sum_{j=0}^k \alpha_j y(x_{n+j}) - h \sum_{j=0}^k \beta_j f(x_{n+j}, y(x_{n+j})) \\
 &= y(x_{n+k}) + \sum_{j=0}^{k-1} \alpha_j y(x_{n+j}) - h \beta_k f(x_{n+k}, y(x_{n+k})) \\
 &\quad - h \sum_{j=0}^{k-1} \beta_j f(x_{n+j}, y(x_{n+j})). \tag{2.46}
 \end{aligned}$$

Subtraindo (2.45) de (2.46), encontramos:

$$y(x_{n+k}) - y_{n+k}^{[i+1]} = h \beta_k f(x_{n+k}, y(x_{n+k})) - h \beta_k f(x_{n+k}, y_{n+k}^{[i]}) + \mathcal{L}[y(x_n), h] \tag{2.47}$$

pois $y_{n+j}^{[m]} = y(x_{n+j})$, $j = 0, 1, \dots, k-1$ e $f_{n+j}^{[m-s]} = f(x_{n+j}, y(x_{n+j}))$, $j = 0, 1, \dots, k-1$, isto é, os valores anteriores são considerados exatos, pois o erro de truncamento é local.

Pelo Teorema do Valor Médio, temos:

$$y(x_{n+k}) - y_{n+k}^{[i+1]} = h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi)[y(x_{n+k}) - y_{n+k}^{[i]}] + \mathcal{L}[y(x_n), h].$$

Assim,

$$y(x_{n+k}) - y_{n+k}^{[i+1]} = h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi)[y(x_{n+k}) - y_{n+k}^{[i]}] + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \tag{2.48}$$

e

$$y(x_{n+k}) - y_{n+k}^{[0]} = C_{q+1}^* h^{q^*+1} y^{(q^*+1)} + O(h^{q^*+2})$$

Observamos que:

- (i) Quando $m = 1 \Rightarrow i = 0$.
- (ii) Como as expressões não envolvem a f então valem para qualquer modo.

1° caso: $q^* \geq q$

a) Seja $m = 1 \Rightarrow i = 0$.

Logo, utilizando (2.47)

$$\begin{aligned}
 y(x_{n+k}) - y_{n+k}^{[1]} &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi)[y(x_{n+k}) - y_{n+k}^{[0]}] + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\
 &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi)[C_{q^*+1}^* h^{q^*+1} y^{(q^*+1)} + O(h^{q^*+2})] \\
 &\quad + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}).
 \end{aligned}$$

Portanto,

$$y(x_{n+k}) - y_{n+k}^{[1]} = C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}). \quad (2.49)$$

Portanto, se $m = 1$, o ETLP do par previsor-corretor é o mesmo do corretor.

b) Seja $m = 2$ e usando (2.49), obtemos:

$$\begin{aligned} y(x_{n+k}) - y_{n+k}^{[2]} &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) [y(x_{n+k}) - y_{n+k}^{[1]}] + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) [C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2})] \\ &\quad + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}). \end{aligned}$$

Novamente, o ETLP do par previsor-corretor é o do corretor, e assim sucessivamente.

Sendo assim, podemos concluir que para $q^* \geq q$, $m \geq 1$, o ETLP do método previsor-corretor é o mesmo do ETLP do corretor.

2° caso: $q^* = q - 1$

a) Seja $m = 1 \Rightarrow i = 0$.

Temos:

$$\begin{aligned} y(x_{n+k}) - y_{n+k}^{[1]} &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) [y(x_{n+k}) - y_{n+k}^{[0]}] + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) [C_q^* h^q y^{(q)}(x_n) + O(h^{q+1})] \\ &\quad + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= [\beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) C_q^* y^{(q)}(x_n) + C_{q+1} y^{(q+1)}(x_n)] h^{q+1} + O(h^{q+2}). \end{aligned}$$

Logo, se $m = 1$, o ETLP do par previsor-corretor é da ordem do ETLP do corretor, mas é diferente.

b) Seja $m = 2$.

$$\begin{aligned} y(x_{n+k}) - y_{n+k}^{[2]} &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) [y(x_{n+k}) - y_{n+k}^{[1]}] + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) \{ [\beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi) C_q^* y^{(q)}(x_n) + C_{q+1} y^{(q+1)}(x_n)] h^{q+1} \\ &\quad + O(h^{q+2}) \} + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= C_{q+1} h^{q+1} y^{(q+1)}(x_n) h^{q+1} + O(h^{q+2}). \end{aligned}$$

Com isso, o ETLP do par previsor-corretor é o do corretor. Consequentemente, se $q^* = q - 1$ e $m \geq 2$, o ETLP do método previsor-corretor é o mesmo do ETLP do corretor.

3° caso: $q^* = q - 2$

a) Seja $m = 1$.

$$\begin{aligned} y(x_{n+k}) - y_{n+k}^{[1]} &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi)[y(x_{n+k}) - y_{n+k}^{[0]}] + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= h \beta_k \frac{\partial f}{\partial y}[C_{q-1}^* h^{q-1} y^{(q-1)}(x_n) + O(h^q)] \\ &\quad + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= [\beta_k \frac{\partial f}{\partial y} C_{q-1}^* y^{(q-1)}(x_n) h^q + O(h^{q+1})]. \end{aligned}$$

Assim, se $m = 1$, o ETLP do par previsor-corretor é uma ordem a menos do corretor.

b) Seja $m = 2$.

$$\begin{aligned} y(x_{n+k}) - y_{n+k}^{[2]} &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi)[y(x_{n+k}) - y_{n+k}^{[1]}] + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi)[\beta_k \frac{\partial f}{\partial y} C_{q-1}^* y^{(q-1)}(x_n) h^q + O(h^{q+1})] \\ &\quad + C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}) \\ &= [\beta_k \frac{\partial f}{\partial y} \beta_k \frac{\partial f}{\partial y} C_{q-1}^* y^{(q-1)}(x_n) + C_{q+1} y^{(q+1)}(x_n)] h^{q+1} + O(h^{q+2}). \end{aligned}$$

Portanto, se $m = 2$, o ETLP do par previsor-corretor é da mesma ordem do corretor, mas é diferente.

c) Seja $m = 3$.

$$y(x_{n+k}) - y_{n+k}^{[3]} = C_{q+1} h^{q+1} y^{(q+1)}(x_n) + O(h^{q+2}).$$

Logo, se $m = 3$, o ETLP do par previsor-corretor é o do corretor. Dessa maneira, se $q^* = q - 2$, $m \geq 3$, o ETLP do método previsor-corretor é o mesmo do ETLP do corretor.

2.5.3 Conclusão Geral:

Se $q^* \geq q$, o ETLP do par previsor-corretor é o do corretor;

Se $q^* = q - j$, $0 \leq j \leq q$, então o ETLP do previsor-corretor é:

- a) do corretor se $m \geq j + 1$.
- b) da mesma ordem do corretor mas diferente se $m = j$.
- c) da forma $k h^{q-j+m+1}$ se $m \leq j - 1$.

Quando $q^* = q$ é possível estimar o ETLP do par previsor-corretor, sem calcular as derivadas de ordem mais elevadas de $y(x)$. Essa técnica é conhecida como Técnica de Milne.

2.5.4 Estabilidade do Par Previsor-Corretor no Modo PECE

Sejam $\tilde{y}_{n+k}^{[0]}$ e $\tilde{y}_{n+k}^{[1]}$ as aproximações para $y(x_{n+k})$ dadas pelo previsor e corretor respectivamente, quando erros de arredondamento ocorrem. Assim, para $m = 1$, temos:

$$\tilde{y}_{n+k}^{[0]} + \sum_{j=0}^{k-1} \alpha_j^* \tilde{y}_{n+j}^{[1]} = h \sum_{j=0}^{k-1} \beta_j^* f(x_{n+j}, \tilde{y}_{n+j}^{[1]}) + R_n^* \quad (2.50)$$

e

$$\tilde{y}_{n+k}^{[1]} + \sum_{j=0}^{k-1} \alpha_j \tilde{y}_{n+j}^{[1]} = h \beta_k \sum_{j=0}^{k-1} f(x_{n+j}, \tilde{y}_{n+j}^{[0]}) + h \sum_{j=0}^{k-1} \beta_j f(x_{n+j}, \tilde{y}_{n+j}^{[1]}) + R_n, \quad (2.51)$$

onde R_n^* e R_n são os erros de arredondamento cometidos em cada estágio do previsor e do corretor, respectivamente. A solução teórica $y(x)$ do problema de valor inicial, satisfaz:

$$\sum_{j=0}^k \alpha_j^* y(x_{n+j}) - h \sum_{j=0}^{k-1} \beta_j^* f(x_{n+j}, y(x_{n+j})) = \mathcal{L}^*[y(x_n), h] \quad (2.52)$$

e

$$\sum_{j=0}^k \alpha_j y(x_{n+j}) - h \sum_{j=0}^k \beta_j f(x_{n+j}, y(x_{n+j})) = \mathcal{L}[y(x_n), h] \quad (2.53)$$

com $\mathcal{L}^*[y(x_n), h]$ e $\mathcal{L}[y(x_n), h]$ sendo os erros de truncamento local do previsor e do corretor, respectivamente.

Vamos definir o erro global $\tilde{e}_n^{[0]}$ e $\tilde{e}_n^{[1]}$ por:

$$\tilde{e}_n^{[0]} = y(x_n) - \tilde{y}_n^{[0]}, \quad \tilde{e}_n^{[1]} = y(x_n) - \tilde{y}_n^{[1]}.$$

Subtraindo (2.50) de (2.52), encontramos:

$$\tilde{e}_{n+k}^{[0]} + \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+k}^{[1]} = h \sum_{j=0}^{k-1} \beta_j^* \frac{\partial f}{\partial y}(x_{n+j}, \xi_{n+j}) \tilde{e}_{n+j}^{[1]} + \mathcal{L}^* - R_n^*,$$

e subtraindo (2.51) de (2.53),

$$\sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[1]} = h \beta_k \frac{\partial f}{\partial y}(x_{n+k}, \xi_{n+k}) \tilde{e}_{n+k}^{[0]} + h \sum_{j=0}^{k-1} \beta_j \frac{\partial f}{\partial y}(x_{n+j}, \xi_{n+j}) + \tilde{e}_{n+j}^{[1]} + \mathcal{L} - R_n,$$

Assumindo que $\frac{\partial f}{\partial y} = \lambda$, $\mathcal{L}^* - R_n^*$ e $\mathcal{L} - R_n$ são constantes, obtemos:

$$\tilde{e}_{n+k}^{[0]} + \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[1]} = h \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[1]} + cte \quad (2.54)$$

e

$$\sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[1]} = \bar{h} \beta_k \tilde{e}_{n+k}^{[0]} + \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[1]} + cte, \quad (2.55)$$

considerando $\bar{h} = h\lambda$.

Novamente, substituindo (2.54) em (2.55), temos:

$$\sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[1]} = \bar{h} \beta_k \left[- \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[1]} + \bar{h} \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[1]} + cte \right] + \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[1]} + cte.$$

Portanto

$$\sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[1]} - \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[1]} = -\bar{h} \beta_k \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[1]} + \bar{h} \beta_k \bar{h} \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[1]}. \quad (2.56)$$

Mas

$$\sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[1]} = \sum_{j=0}^k \beta_j^* \tilde{e}_{n+j}^{[1]},$$

pois como o método é explícito $\Rightarrow \beta_k = 0$.

Além disso, somando em ambos os membros o termo: $-\bar{h} \beta_k \tilde{e}_{n+k}^{[1]}$, ficamos

com:

$$-\bar{h} \beta_k \tilde{e}_{n+k}^{[1]} - \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[1]} = -\bar{h} \sum_{j=0}^k \beta_j \tilde{e}_{n+j}^{[1]}$$

e

$$-\bar{h} \beta_k \tilde{e}_{n+k}^{[1]} - \bar{h} \beta_k \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[1]} = -\bar{h} \beta_k \sum_{j=0}^k \alpha_j^* \tilde{e}_{n+j}^{[1]}.$$

Logo, substituindo em (2.56), obtemos:

$$\sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[1]} - \bar{h} \sum_{j=0}^k \beta_j \tilde{e}_{n+j}^{[1]} = -\bar{h} \beta_k \sum_{j=0}^k \alpha_j^* \tilde{e}_{n+j}^{[1]} + \bar{h} \beta_k \bar{h} \sum_{j=0}^k \beta_j^* \tilde{e}_{n+j}^{[1]} + cte.$$

Assim,

$$\sum_{j=0}^k \{(\alpha_j - \bar{h} \beta_j) + \bar{h} \beta_k (\alpha_j^* - \bar{h} \beta_j^*)\} \tilde{e}_{n+j}^{[1]} = cte.$$

Dessa maneira, obtemos uma equação diferença, cuja solução é:

$$\tilde{e}_n^{[1]} = \sum_{s=1}^k c_s r_s^n + cte$$

onde r_s são raízes do polinômio,

$$\sum_{j=0}^k \{(\alpha_j - \bar{h} \beta_j) + \bar{h} \beta_k (\alpha_j^* - \bar{h} \beta_j^*)\} r^j = \prod_{PECE} (r, \bar{h}).$$

Entretanto,

$$\prod_{PECE} (r, \bar{h}) = \rho(r) - \bar{h} \sigma(r) + \bar{h} \beta_k [\rho^*(r) - \bar{h} \sigma^*(r)]$$

é o polinômio de estabilidade do par previsor-corretor no modo PECE.

Definição 2.11. (α, β) é um intervalo de estabilidade absoluta para o previsor-corretor, se para qualquer $\bar{h} \in (\alpha, \beta)$ as raízes do $\prod_{PECE}(r, \bar{h})$ satisfazem $|r_s| < 1$, $s = 1, 2, \dots, k$.

Definição 2.12. (α, β) é um intervalo de estabilidade relativa para o previsor-corretor, se para qualquer $\bar{h} \in (\alpha, \beta)$ as raízes do $\prod_{PECE}(r, \bar{h})$ satisfazem $|r_s| < |r_1|$, $s = 2, \dots, k$.

Apresentamos agora, a estabilidade do par previsor-corretor para o modo $P(EC)^2E$, ou seja, temos que determinar $\tilde{e}_n^{[2]}$.

Fazendo os cálculos, encontramos:

$$\tilde{e}_{n+k}^{[0]} + \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[2]} = \bar{h} \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[2]} + cte, \quad (2.57)$$

$$\tilde{e}_{n+k}^{[1]} + \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[2]} = \bar{h} \beta_k \tilde{e}_{n+k}^{[0]} + \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} + cte, \quad (2.58)$$

$$\tilde{e}_{n+k}^{[2]} + \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[2]} = \bar{h} \beta_k \tilde{e}_{n+k}^{[1]} + \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} + cte. \quad (2.59)$$

Substituindo (2.57) em (2.58):

$$\tilde{e}_{n+k}^{[1]} + \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[2]} = \bar{h} \beta_k \left[- \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[2]} + \bar{h} \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[2]} \right] + \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} + cte,$$

Do mesmo modo, substituindo em (2.59), geramos:

$$\begin{aligned} \tilde{e}_{n+k}^{[2]} + \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[2]} &= \bar{h} \beta_k \left[- \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[2]} - \bar{h} \beta_k \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[2]} + \bar{h} \beta_k \bar{h} \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[2]} \right. \\ &\quad \left. + \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} \right] + \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} + cte. \end{aligned}$$

Logo,

$$\begin{aligned} \sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[2]} + \bar{h} \beta_k \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[2]} - \bar{h} \beta_k \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} - \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} \\ = -\bar{h} \beta_k \bar{h} \beta_k \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[2]} + \bar{h} \beta_k \bar{h} \beta_k \bar{h} \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[2]} + cte. \end{aligned} \quad (2.60)$$

Agora, somando e subtraindo, no primeiro membro, o termo $\bar{h} \beta_k \tilde{e}_{n+k}^{[2]}$, produzimos:

$$\begin{aligned} \bar{h} \beta_k \tilde{e}_{n+k}^{[2]} + \bar{h} \beta_k \sum_{j=0}^{k-1} \alpha_j \tilde{e}_{n+j}^{[2]} &= \bar{h} \beta_k \sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[2]} \\ -\bar{h} \beta_k \tilde{e}_{n+k}^{[2]} - \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} &= -\bar{h} \sum_{j=0}^k \beta_j \tilde{e}_{n+j}^{[2]}. \end{aligned}$$

Ainda, somando em ambos os membros de (2.60), o termo $\bar{h} \beta_k \bar{h} \beta_k \tilde{e}_{n+k}^{[2]}$, ficamos com:

$$\begin{aligned} -\bar{h} \beta_k \bar{h} \beta_k \tilde{e}_{n+k}^{[2]} - \bar{h} \beta_k \bar{h} \sum_{j=0}^{k-1} \beta_j \tilde{e}_{n+j}^{[2]} &= -\bar{h} \beta_k \bar{h} \sum_{j=0}^k \beta_j \tilde{e}_{n+j}^{[2]}; \\ -\bar{h} \beta_k \bar{h} \beta_k \tilde{e}_{n+k}^{[2]} - \bar{h} \beta_k \bar{h} \beta_k \sum_{j=0}^{k-1} \alpha_j^* \tilde{e}_{n+j}^{[2]} &= -\bar{h} \beta_k \bar{h} \beta_k \sum_{j=0}^k \alpha_j^* \tilde{e}_{n+j}^{[2]}, \end{aligned}$$

e também: $\bar{h} \beta_k \bar{h} \beta_k \bar{h} \sum_{j=0}^{k-1} \beta_j^* \tilde{e}_{n+j}^{[2]} = \bar{h} \beta_k \bar{h} \beta_k \bar{h} \sum_{j=0}^k \beta_j^* \tilde{e}_{n+j}^{[2]}$, pois como o método é explícito, $\beta_k = 0$.

Assim, substituindo em (2.60),

$$\begin{aligned} \sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[2]} + \bar{h} \beta_k \sum_{j=0}^k \alpha_j \tilde{e}_{n+j}^{[2]} - \bar{h} \beta_k \bar{h} \sum_{j=0}^k \beta_j \tilde{e}_{n+j}^{[2]} - \bar{h} \sum_{j=0}^k \beta_j \tilde{e}_{n+j}^{[2]} \\ = -\bar{h} \beta_k \bar{h} \beta_k \sum_{j=0}^k \alpha_j^* \tilde{e}_{n+j}^{[2]} + \bar{h} \beta_k \bar{h} \beta_k \bar{h} \sum_{j=0}^k \beta_j^* \tilde{e}_{n+j}^{[2]} + cte. \end{aligned}$$

Com isso,

$$(1 + \bar{h} \beta_k) \sum_{j=0}^k (\alpha_j + \bar{h} \beta_j) \tilde{e}_{n+j}^{[2]} = -(\bar{h} \beta_k)^2 \sum_{j=0}^k (\alpha_j^* + \bar{h} \beta_j^*) \tilde{e}_{n+j}^{[2]} + cte,$$

ou

$$\sum_{j=0}^k (\alpha_j + \bar{h} \beta_j) + \frac{(\bar{h} \beta_k)^2}{1 + \bar{h} \beta_k} (\alpha_j^* + \bar{h} \beta_j^*) \tilde{e}_{n+j}^{[2]} = cte.$$

Todavia, encontramos uma equação diferença, cuja solução é:

$$\tilde{e}_n^{[2]} = \sum_{s=1}^k c_s r_s^n + cte,$$

onde r_s são as raízes do polinômio:

$$\prod_{P(EC)^2 E} (r, \bar{h}) = \rho(r) - \bar{h} \sigma(r) + \frac{(\bar{h} \beta_k)^2}{1 + \bar{h} \beta_k} (\rho^*(r) - \bar{h} \sigma^*(r)),$$

que é o polinômio de estabilidade do par previsor-corretor no modo $P(EC)^2 E$.

Essa análise pode ser estendida, dado o seguinte polinômio de estabilidade do par previsor-corretor no modo $P(EC)^m E$:

$$\prod_{P(EC)^m} (r, \bar{h}) = [\rho(r) - \bar{h} \sigma(r)] + \frac{(\bar{h} \beta_k)^m (1 - \bar{h} \beta_k)}{1 - (\bar{h} \beta_k)^m} [\rho^*(r) \sigma(r) - \rho(r) \sigma^*(r)], \quad m = 0, 1, \dots$$

2.6 Considerações Finais

Neste capítulo, fizemos um estudo de métodos numéricos para resolução de EDO. Tais métodos são muito utilizados como auxílio na resolução das EDP que modelam o problema de Retoque Digital. Dentre os métodos mencionados anteriormente, o mais simples de ser implementado é o Método de Euler e essa é a razão pela qual ele é geralmente utilizado.

O objetivo de nosso trabalho é o de aprimorar a qualidade do Retoque Digital, melhorando a qualidade da aproximação na equação $u_t = L(u)$, que descreve o problema.

O método de Euler, pelo seu caráter explícito e por ser de passo um é frequentemente aplicado, porém, como foi dito anteriormente, é apenas de ordem h . Outros métodos de ordens superiores foram estudados, ou seja, os métodos lineares de passo múltiplo, os métodos de Runge-Kutta e o método Previsor-Corretor. Os métodos de Runge-Kutta apresentam uma complexidade na adaptação da EDP do problema devido aos cálculos das funções envolvidas pelo método, dificultando assim na implementação. Sendo assim, escolhemos os métodos lineares explícitos de passo 2, 3 e 4, e o método previsor-corretor, para testar a eficiência. Para o método previsor-corretor, tomamos como previsor um método explícito de passo um e como corretor um método implícito também de passo um, com ordem de convergência h^2 , e para os métodos lineares explícitos de passo múltiplo utilizamos os métodos de Adams-Bashford de passos 2, 3 e 4 com respectivas ordens h^2 , h^3 e h^4 .

Essas escolhas fazem com que a complexidade computacional seja diminuída em comparação com os métodos de Runge-Kutta, e a ordem de convergência aumentada com relação ao método de Euler.

A seguir, apresentamos os modelos para o problema de Retoque Digital.

Capítulo 3

Modelos Matemáticos para Retoque Digital

3.1 Introdução

O objetivo deste capítulo é relatar alguns modelos matemáticos que foram desenvolvidos nas últimas décadas para retoque digital de imagens. O primeiro modelo apresentado foi o modelo de Bertalmío, Sapiro, Ballester e Caselles (BSBC), que propuseram uma EDP de terceira ordem baseada no transporte ao longo das linhas de nível, para solucionar o problema de retoque digital. Os demais modelos foram originados a partir deste.

Apresentamos primeiramente, o modelo BSBC, em seguida o modelo Variacional Total (TV) proposto por Rudin, Osher e Fatemi, logo após o modelo da Difusão Guiada pela Curvatura (CDD) de Chan e Shen.

3.2 Modelo de Retoque Digital BSBC

A modificação de imagens indetectáveis para um observador que não conhece a imagem original é uma prática tão antiga quanto a criação artística. A arte medieval começou a ser restaurada já na Renascença, motivada pelo desejo de atualizar as pinturas que apresentavam partes perdidas ou danificadas. Baseados neste fato histórico, Bertalmío, Sapiro, Ballester e Caselles propuseram um modelo de retoque digital reapli-

cando a técnica básica usada por estes restauradores profissionais. Para isso, o usuário deve selecionar as regiões a serem realçadas, sendo que estas, serão “completadas” com a estrutura de suas vizinhanças, ao longo das linhas de nível.

Desse modo, necessitamos traduzir esses conceitos manuais de retoque para uma linguagem matemática.

Seja $u_0(i, j) : [0, M] \times [0, N] \rightarrow \mathbb{R}$, com $[0, M] \times [0, N] \subset \mathbb{N} \times \mathbb{N}$ uma imagem de níveis de cinza em duas dimensões (2D) discreta. O objetivo do retoque digital é construir uma família de imagens $u(i, j, n) : [0, M] \times [0, N] \times \mathbb{N} \rightarrow \mathbb{R}$, tal que $u(i, j, 0) = u_0(i, j)$ e $\lim_{n \rightarrow \infty} u(i, j, n) = u_R(i, j)$, onde $u_R(i, j)$ é a imagem obtida após aplicar o algoritmo.

Sejam Ω a região de retoque, $u^n(i, j)$ cada “pixel” da imagem dentro de Ω no tempo n e $u_t^n(i, j)$ a atualização da imagem $u^n(i, j)$.

De acordo com Bertalmío, Sapiro, Ballester e Caselles [6], o modelo propaga informações de Ω em Ω . Sendo assim, chamando de $L^n(i, j)$ a informação que será propagada e $\vec{N}^n(i, j)$ a direção de propagação, os autores definiram a atualização da imagem $u^n(i, j)$ como sendo:

$$u_t^n(i, j) = \overrightarrow{\delta L}^n(i, j) \cdot \vec{N}^n(i, j), \quad (3.1)$$

onde $\overrightarrow{\delta L}^n(i, j)$ é a medida de alteração da informação $L^n(i, j)$. Esta equação estima a informação $L^n(i, j)$ e calcula sua alteração ao longo da direção \vec{N}^n .

Agora, usando a aproximação pelo método de Euler para $u_t^n(i, j)$, obtemos:

$$u_t^n(i, j) = \frac{u^{n+1}(i, j) - u^n(i, j)}{\Delta t},$$

ou ainda,

$$u^{n+1}(i, j) = u^n(i, j) + \Delta t u_t^n(i, j), \quad \forall (i, j) \in \Omega. \quad (3.2)$$

onde Δt é a taxa de aperfeiçoamento.

Com esta equação, a imagem $u^{n+1}(i, j)$ é uma versão “melhorada” de $u^n(i, j)$, sendo que essa “melhora” é dada por $u_t^n(i, j)$. Por outro lado, note que o estado estável, ou seja, quando o algoritmo converge, $u^{n+1} = u^n$, e de (3.1)-(3.2), temos que

$$\overrightarrow{\delta L}^n \cdot \vec{N}^n = 0,$$

o que significa que a informação L foi propagada na direção \vec{N} .

Relatamos ainda, que uma medida de propagação de segunda ordem pode ser expressa na forma geral:

$$L(u) = f(\nabla u, \nabla \otimes \nabla u),$$

com ∇u sendo o vetor gradiente e $\nabla \otimes \nabla u$ a matriz Hessiana (2x2). Em nosso algoritmo, queremos que a propagação seja suave, logo $L^n(i, j)$ será então um estimador de suavização da imagem. Sendo assim, uma simples escolha do Laplaciano, discretizado da forma $L^n(i, j) = u_{xx}^n(i, j) + u_{yy}^n(i, j)$ é suficiente. (Segundo Chan e Shen, outro estimador, menos suave, pode também ser usado, embora resultados mais satisfatórios sempre foram obtidos com a escolha do Laplaciano.)

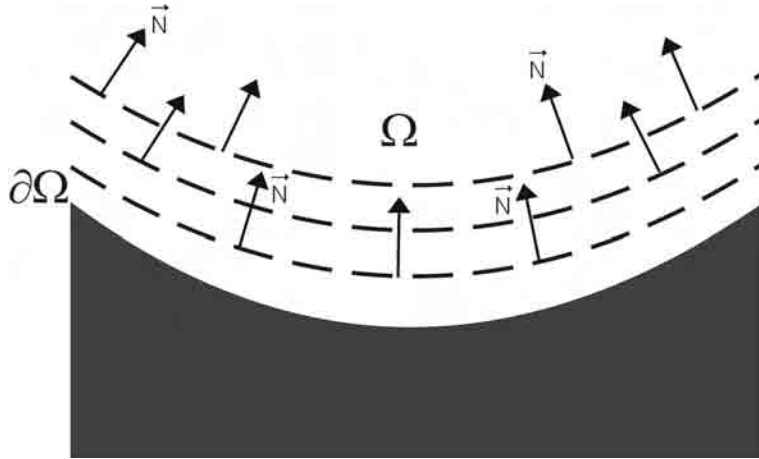


Figura 3.1: Direção de propagação normal à região de contorno a ser realçada.

Agora, passamos a calcular a variação $\overline{\delta L^n}$ ao longo da direção \vec{N} . Para isto temos que definir qual será a direção \vec{N} da propagação da informação em duas dimensões. Uma escolha é definir como sendo a direção normal, ou seja, as informações irão se propagar ponto a ponto até atingir o contorno da região de retoque (isto pode ser visto na Figura 3.1). Esta escolha é motivada por garantir que uma propagação normal até o contorno leva à continuidade das linhas de nível no contorno. Isto acontece, pois as linhas de nível chegam em $\partial\Omega$ (região do contorno) para alinhar-se em \vec{N} (ver Figura 3.2).



Figura 3.2: Propagação das informações na direção normal ao contorno.

A direção de propagação da informação, depende da escolha que fazemos para \vec{N} . A escolha na direção normal é uma escolha mal sucedida como pode ser observado na Figura 3.2. À esquerda: detalhes da imagem original, onde a parte branca é a região a ser realçada e à direita a imagem restaurada. Como as linhas de nível tendem à alinhar-se com \vec{N} , a melhor escolha para \vec{N} é então as direções das linhas de nível.

Dessa forma, usaremos um estimador variável na direção das linhas de nível. Dado um ponto (i, j) , o vetor gradiente discretizado $\nabla u^n(i, j)$ dá a direção da maior mudança espacial, então $\nabla^\perp u^n(i, j)$ rotacionado 90° dá a direção da menor mudança espacial. Assim, o vetor $\nabla^\perp u^n(i, j)$ dá a direção das linhas de nível. Seu campo \vec{N} é então dado pela variação de tempo $\vec{N}(i, j) = \nabla^\perp u^n(i, j)$. Este estimulador, de variação de tempo, é primitivo no princípio, mas progressivamente ele encontra a continuidade desejada em $\partial\Omega$ (região de contorno), em vez de fixar o campo $\vec{N}(i, j)$ que necessitaria conhecer as direções das linhas de nível iniciais.

Note que a direção do campo não é normalizada, sua norma é a norma do gradiente de $\nabla u^n(i, j)$, pois:

$$|\nabla u(i, j)| = |(u_x, u_y)^t| = \sqrt{u_x^2(i, j) + u_y^2(i, j)},$$

$$|\nabla^\perp u(i, j)| = |(u_y, -u_x)^t| = \sqrt{u_x^2(i, j) + u_y^2(i, j)}.$$

Esta escolha ajuda na estabilidade numérica do algoritmo.

Desde que desempenhamos retoque ao longo das linhas de nível, é irrelevante se $\nabla^\perp u^n(i, j)$ é obtido como uma rotação de sentido horário ou anti-horário de $\nabla u^n(i, j)$. Em ambos os casos, a mudança de $\nabla u^n(i, j)$ ao longo daquelas direções será minimizada.

Resumindo, estimamos uma variação da suavização, dada pela discretização

do Laplaciano em duas dimensões, a qual calculamos como sendo:

$$\overrightarrow{\delta L^n}(i, j) := (L^n(i+1, j) - L^n(i-1, j), L^n(i, j+1) - L^n(i, j-1)), \quad (3.3)$$

e projetamos esta variação na direção das linhas de nível. Esta projeção é usada para atualizar os valores da imagem dentro da região de retoque.

Da matemática contínua, temos que $\overrightarrow{\delta L} = \nabla L$. Assim, reescrevendo a equação (3.1) na forma de EDP, temos a equação de retoque:

$$u_t = \nabla(\Delta u) \cdot \nabla^\perp u. \quad (3.4)$$

3.3 Difusão no Processo de Retoque Digital

Para garantir uma evolução correta do campo de direções, um processo de difusão é entrefolheado pelo processo de retoque da imagem descrito anteriormente. Isto é, a cada poucos passos, aplicamos poucas iterações de difusão. Esta difusão estabelece o limite que as linhas de nível se curvam, para evitá-las de cruzá-las, uma com a outra. Usamos difusão anisotrópica (ver mais detalhes em [1], [15]), primeiramente para alcançar este objetivo sem perda de intensidade na reconstrução. Em particular, aplicamos uma discretização direta do seguinte espaço contínuo da equação de difusão anisotrópica:

$$\frac{\partial u}{\partial t}(x, y, t) = g_\epsilon(x, y)\kappa(x, y, t)|\nabla u(x, y, t)|, \quad \forall (x, y) \in \Omega^\epsilon \quad (3.5)$$

onde Ω^ϵ é uma dilatação de Ω com uma bola de raio ϵ , κ é a curvatura Euclidiana das linhas de nível de u e $g_\epsilon(x, y)$ é a função suave em Ω^ϵ , tal que $g_\epsilon(x, y) = 0$ em $\partial\Omega^\epsilon$, e $g_\epsilon(x, y) = 1$ em Ω^ϵ .

De acordo com os autores de [6], a curvatura Euclidiana das linhas de nível é calculada como sendo $k = \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right)$. Assim, o processo de difusão é dado por:

$$\begin{cases} u_t = |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = \frac{u_{xx}u_y^2 - 2u_xu_yu_{xy} + u_{yy}u_x^2}{u_x^2 + u_y^2} & \text{para } (x, y) \in \Omega, \\ u_t = 0, & \text{para } (x, y) \notin \Omega \end{cases} \quad (3.6)$$

3.3.1 Discretização dos termos do modelo BSBC

Para a realização do algoritmo proposto em [6], necessitamos da imagem para ser restaurada e a “máscara” que delimita a parte a ser realçada. Depois disso, os valores dentro da região Ω são modificados, de acordo com a implementação numérica dada abaixo:

$$u_t^n(i, j) = \left(\overrightarrow{\delta L^n}(i, j) \cdot \frac{\overrightarrow{N}(i, j, n)}{|\overrightarrow{N}(i, j, n)|} \right) |\nabla u^n(i, j)|, \quad (3.7)$$

onde

$$\overrightarrow{\delta L^n}(i, j) := (L^n(i+1, j) - L^n(i-1, j), L^n(i, j+1) - L^n(i, j-1)), \quad (3.8)$$

$$L^n(i, j) = u_{xx}^n(i, j) + u_{yy}^n(i, j), \quad (3.9)$$

$$\frac{\overrightarrow{N}(i, j, n)}{|\overrightarrow{N}(i, j, n)|} := \frac{(-u_y^n(i, j), u_x^n(i, j))}{\sqrt{(u_x^n(i, j))^2 + (u_y^n(i, j))^2}}, \quad (3.10)$$

$$\beta^n(i, j) = \overrightarrow{\delta L^n}(i, j) \cdot \frac{\overrightarrow{N}(i, j, n)}{|\overrightarrow{N}(i, j, n)|}, \quad (3.11)$$

Calculamos o estimador $2D$ em (3.9) e a direção das linhas de nível em (3.10). Então em (3.11) calculamos β^n , a projeção de $\overrightarrow{\delta L}$ sobre o vetor normalizado \overrightarrow{N} . Finalmente, multiplicamos β^n por uma versão limitada da norma do gradiente da imagem, $|\nabla u|$, dada a seguir por (3.12). Essa versão limitada é utilizada devido à falta de comunicação entre as linhas de nível, deste modelo, pois podem ocorrer “saltos” grandes de um domínio para o outro, causando discontinuidades. Estas discontinuidades produzem vorticidade ao longo da propagação de brilho, que podem ser detectadas pela curva local. Além disso, essa vorticidade pode resultar em “choques” na propagação de brilho.

Para solucionar este problema, um caminho seria a parametrização do movimento da curva e a discretização desta parametrização no conjunto dos pontos selecionados (domínio de retoque). As posições dos pontos marcados são processadas no tempo de acordo com a aproximação das equações do movimento. Tal técnica pode ser aproximada na tentativa de seguir os movimentos de pequenas perturbações. Entretanto, para perturbações grandes, o movimento torna-se complexo e vários problemas

podem ocorrer. Um destes problemas acontece quando os pontos selecionados atingem regiões de propagação de curvas, causando instabilidade numérica. Esta instabilidade ocorre, no modelo estudado, quando aplicamos aproximações com diferenças centradas para o termo $|\nabla u|$. Uma maneira de tornar nosso problema estável é aplicar uma técnica de regressão, juntamente com passos intermediários de difusão. Neste caso, utilizamos a difusão anisotrópica dada por (3.6) e a técnica de regressão dada por:

$$|\nabla u^n(i, j)| = \begin{cases} \sqrt{(u_{xbm}^n)^2 + (u_{xfM}^n)^2 + (u_{ybm}^n)^2 + (u_{yfM}^n)^2}, & \text{se } \beta^n > 0 \\ \sqrt{(u_{xbM}^n)^2 + (u_{xfm}^n)^2 + (u_{ybm}^n)^2 + (u_{yfm}^n)^2}, & \text{se } \beta^n < 0, \end{cases} \quad (3.12)$$

onde

$$\begin{aligned} u_{xbm}^n(i, j) &= \min(u^n(i-1, j) - u^n(i, j), 0) \\ u_{xbM}^n(i, j) &= \max(u^n(i-1, j) - u^n(i, j), 0) \\ u_{xfm}^n(i, j) &= \max(u^n(i+1, j) - u^n(i, j), 0) \\ u_{xfM}^n(i, j) &= \min(u^n(i+1, j) - u^n(i, j), 0) \\ u_{ybm}^n(i, j) &= \min(u^n(i, j-1) - u^n(i, j), 0) \\ u_{ybM}^n(i, j) &= \max(u^n(i, j-1) - u^n(i, j), 0) \\ u_{yfm}^n(i, j) &= \max(u^n(i, j+1) - u^n(i, j), 0) \\ u_{yfM}^n(i, j) &= \min(u^n(i, j+1) - u^n(i, j), 0), \end{aligned}$$

com os sub-índices f e b denotando diferenças avançadas e atrasadas respectivamente, enquanto os sub-índices m e M , o mínimo e máximo respectivamente.

3.3.2 Uma forma alternativa para a equação do modelo BSBC

Batista, em sua dissertação de Mestrado [4], propôs uma maneira alternativa de implementar o modelo BSBC.

Vamos então apresentar essa forma alternativa proposta por Batista. Consideramos a EDP do modelo dada pela equação (3.4), com $\overrightarrow{\delta L^n} = \nabla L$, ou seja:

$$\frac{\partial u}{\partial t} = \nabla L(u) \cdot \nabla^\perp u,$$

e tomando $L(u)$ como sendo o Laplaciano temos:

$$\frac{\partial u}{\partial t} = \nabla(\Delta u) \cdot \nabla^\perp u.$$

Assim,

$$\begin{aligned}
 \frac{\partial u}{\partial t} &= \nabla(\Delta u) \cdot \nabla^\perp u \\
 &= \nabla(\nabla(\nabla u)) \cdot (\nabla u)^\perp \\
 &= \nabla\left(\nabla\left(\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)u\right)\right) \cdot \left(\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)u\right)^\perp \\
 &= \nabla\left(\nabla\left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)\right) \cdot \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)^\perp \\
 &= \nabla\left(\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right) \cdot \left(\frac{\partial u}{\partial x}, \frac{\partial u}{\partial y}\right)\right) \cdot \left(-\frac{\partial u}{\partial y}, \frac{\partial u}{\partial x}\right) \\
 &= \nabla\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) \cdot \left(-\frac{\partial u}{\partial y}, \frac{\partial u}{\partial x}\right) \\
 &= \left(\left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}\right)\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)\right) \cdot \left(-\frac{\partial u}{\partial y}, \frac{\partial u}{\partial x}\right) \\
 &= \left(\frac{\partial^3 u}{\partial x^3} + \frac{\partial^3 u}{\partial x y^2}, \frac{\partial^3 u}{\partial x^2 y} + \frac{\partial^3 u}{\partial y^3}\right) \cdot \left(-\frac{\partial u}{\partial y}, \frac{\partial u}{\partial x}\right) \\
 &= \left(\frac{\partial^3 u}{\partial x^3} + \frac{\partial^3 u}{\partial x y^2}\right)\left(-\frac{\partial u}{\partial y}\right) + \left(\frac{\partial^3 u}{\partial x^2 y} + \frac{\partial^3 u}{\partial y^3}\right)\frac{\partial u}{\partial x} \\
 &= \left(\frac{\partial^3 u}{\partial x^2 y} + \frac{\partial^3 u}{\partial y^3}\right)\frac{\partial u}{\partial x} - \left(\frac{\partial^3 u}{\partial x^3} + \frac{\partial^3 u}{\partial x y^2}\right)\frac{\partial u}{\partial y}
 \end{aligned}$$

Portanto, a forma alternativa para a equação do modelo para o domínio de retoque é dada por:

$$\frac{\partial u}{\partial t} = \left(\frac{\partial^3 u}{\partial x^2 y} + \frac{\partial^3 u}{\partial y^3}\right)\frac{\partial u}{\partial x} - \left(\frac{\partial^3 u}{\partial x^3} + \frac{\partial^3 u}{\partial x y^2}\right)\frac{\partial u}{\partial y}. \quad (3.13)$$

Caso contrário, isto é, fora do domínio de retoque, a equação será:

$$\frac{\partial u}{\partial t} = 0.$$

Sendo assim, utilizando a aproximação pelo método de Euler, o esquema numérico torna-se:

$$u^{n+1}(i, j) = u^n(i, j) + \Delta t u_t^n(i, j),$$

onde $u_t^n(i, j)$ é dada por (3.13), ou seja:

$$u_t^n(i, j) = \left(u_{xxy}(i, j) + u_{yyy}(i, j)\right)u_x(i, j) - \left(u_{xxx}(i, j) + u_{xyy}(i, j)\right)u_y(i, j).$$

De maneira análoga à discretização proposta por Bertalmío, Sapiro, Ballester e Caselles [6], o processo é intercalado com a mesma equação de difusão anisotrópica.

A implementação pode ser efetuada utilizando derivadas centradas, ou seja, aproximando-se as derivadas presentes na equação (3.13) por:

$$\begin{aligned}
 \frac{\partial u}{\partial x} &\cong \frac{u(x+h, y) - u(x-h, y)}{2h} \\
 \frac{\partial u}{\partial y} &\cong \frac{u(x, y+h) - u(x, y-h)}{2h} \\
 \frac{\partial^2 u}{\partial x^2} &\cong \frac{u(x+2h, y) - 2u(x, y) + u(x-2h, y)}{4h} \\
 \frac{\partial^2 u}{\partial y^2} &\cong \frac{u(x, y+2h) - 2u(x, y) + u(x, y-2h)}{4h} \\
 \frac{\partial^2 u}{\partial x \partial y} &\cong \frac{u(x+h, y+h) - u(x-h, y+h) - u(x+h, y-h) + u(x-h, y-h)}{4h} \\
 \frac{\partial^3 u}{\partial x^3} &\cong \frac{u(x+3h, y) - 3u(x+h, y) + 3u(x-h, y) - u(x-3h, y)}{8h} \\
 \frac{\partial^3 u}{\partial y^3} &\cong \frac{u(x, y+3h) - 3u(x, y+h) + 3u(x, y-h) - u(x, y-3h)}{8h}
 \end{aligned}$$

$$\begin{aligned}
 \frac{\partial^3 u}{\partial x^2 \partial y} &\cong \frac{u(x+2h, y+h) - 2u(x, y+h) + u(x-2h, y+h)}{8h} \\
 &\quad - \frac{u(x+2h, y-h) - 2u(x, y-h) + u(x-2h, y-h)}{8h} \\
 \frac{\partial^3 u}{\partial x \partial y^2} &\cong \frac{u(x+h, y+2h) - 2u(x+h, y) + u(x+h, y-2h)}{8h} \\
 &\quad - \frac{u(x-h, y+2h) - 2u(x-h, y) + u(x-h, y-2h)}{8h}.
 \end{aligned}$$

Apresentamos a seguir, a discretização do processo de difusão.

3.3.3 Discretização do processo de difusão

O processo de difusão é dado por:

$$\frac{\partial u}{\partial t} = |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right),$$

onde $u(x, y)$ é a função que representa a imagem.

Desenvolvendo esse termo obtemos:

$$\begin{aligned}
 |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) &= \sqrt{u_x^2 + u_y^2} \left[\operatorname{div} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}}, \frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right) \right] \\
 &= \sqrt{u_x^2 + u_y^2} \left[\frac{d}{dx} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}} \right) + \frac{d}{dy} \left(\frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right) \right] \\
 &= \sqrt{u_x^2 + u_y^2} \left[\frac{u_{xx} \sqrt{u_x^2 + u_y^2} - u_x \frac{d}{dx} \left(\sqrt{u_x^2 + u_y^2} \right)}{u_x^2 + u_y^2} \right. \\
 &\quad \left. + \frac{u_{yy} \sqrt{u_x^2 + u_y^2} - u_y \frac{d}{dy} \left(\sqrt{u_x^2 + u_y^2} \right)}{u_x^2 + u_y^2} \right].
 \end{aligned}$$

Além disso, temos:

$$\begin{aligned}
 (i) \quad \frac{\partial}{\partial x} \left(\sqrt{u_x^2 + u_y^2} \right) &= \frac{1}{2} (u_x^2 + u_y^2)^{-1/2} (2u_x u_{xx} + 2u_y u_{yx}) \\
 (ii) \quad \frac{\partial}{\partial y} \left(\sqrt{u_x^2 + u_y^2} \right) &= \frac{1}{2} (u_x^2 + u_y^2)^{-1/2} (2u_x u_{xy} + 2u_y u_{yy})
 \end{aligned}$$

Daí,

$$\begin{aligned}
 |\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) &= \sqrt{u_x^2 + u_y^2} \left[\frac{u_{xx} \sqrt{u_x^2 + u_y^2} - u_x \left(\frac{1}{2\sqrt{u_x^2 + u_y^2}} (2u_x u_{xx} + 2u_y u_{yx}) \right)}{u_x^2 + u_y^2} \right] \\
 &\quad + \sqrt{u_x^2 + u_y^2} \left[\frac{u_{yy} \sqrt{u_x^2 + u_y^2} - u_y \left(\frac{1}{2\sqrt{u_x^2 + u_y^2}} (2u_x u_{xy} + 2u_y u_{yy}) \right)}{u_x^2 + u_y^2} \right] \\
 &= \frac{u_{xx}(u_x^2 + u_y^2) - u_x(u_x u_{xx} + u_y u_{yx})}{u_x^2 + u_y^2} + \frac{u_{yy}(u_x^2 + u_y^2) - u_y(u_x u_{xy} + u_y u_{yy})}{u_x^2 + u_y^2} \\
 &= \frac{u_{xx}u_x^2 + u_{xx}u_y^2 - u_x^2 u_{xx} - u_x u_y u_{yx} + u_{yy}u_x^2 + u_{yy}u_y^2 - u_y u_x u_{xy} - u_y^2 u_{yy}}{u_x^2 + u_y^2} \\
 &= \frac{u_{xx}u_y^2 - 2u_x u_y u_{xy} + u_{yy}u_x^2}{u_x^2 + u_y^2}.
 \end{aligned}$$

Logo,

$$|\nabla u| \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = \frac{u_x^2 u_{xx} - 2u_x u_y u_{xy} + u_y^2 u_{yy}}{u_x^2 + u_y^2}. \quad (3.14)$$

Usando os operadores de diferenças finitas introduzidos na seção 2.1, obtemos uma expressão numérica para o cálculo do termo (3.14).

Na próxima seção apresentamos o modelo de Retoque Digital Variacional Total.

3.4 Modelo Variacional Total (TV)

O modelo TV ilustra com sucesso alguns aspectos do processo de desocclusão na visão psicológica, incluindo as ilusões embaraçadas e analisadas pelo psicólogo Kanizsa[13].

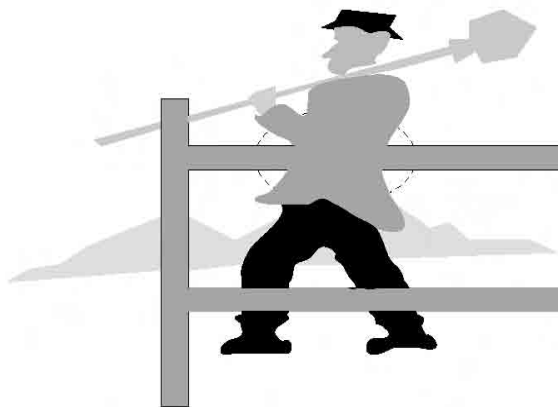


Figura 3.3: O homem embaraçado de Kanizsa

A Figura 3.3 mostra como a percepção visual pode subconscientemente contradizer a situação real. O fato é que a parte superior do corpo do homem está atrás da cerca, enquanto, nossa percepção prefere afirmar que a parte superior do corpo está em frente a cerca. Esta situação é aparentemente causada pela presença da mesma cor na cerca e na parte superior do corpo do homem. Este é um dos exemplos das ilusões embaraçadas e analisadas por Kanizsa.

O modelo Variacional Total proposto por Rudin, Osher e Fatemi [17] consiste em resolver o problema de minimização do seguinte funcional:

$$J[u] = \int_{\Omega} |\nabla u| \, dx \, dy$$

sujeito às restrições

$$\int_{\Omega} u \, dx \, dy = \int_{\Omega} I \, dx \, dy$$

e

$$\int_{\Omega} |u - I|^2 \, dx \, dy = \eta^2.$$

A primeira restrição nos informa que o ruído é Gaussiano e a segunda usa o fato do desvio padrão do ruído ser $\eta(x, y)$.

Este problema de minimização está diretamente relacionado ao problema de minimização sem restrições:

$$J[u] = \int_{\Omega} (|\nabla u| + \beta |u - I|^2) \, dx \, dy.$$

A solução desse problema é obtida encontrando-se a solução do estado estacionário da equação diferencial parcial, a qual é denominada a evolução da equação de Euler-Lagrange de $J[u]$.

Assim, consideremos

$$P = |\nabla u| + \frac{\beta}{2} |u - I|^2 = \sqrt{u_x^2 + u_y^2} + \frac{\beta}{2} |u - I|^2 = \sqrt{u_x^2 + u_y^2} + \frac{\beta}{2} (u - I)^2.$$

Então, para $P(x, y, u, u_x, u_y)$ temos a seguinte equação de Euler-Lagrange:

$$\frac{\partial P}{\partial u} - \frac{\partial}{\partial x} \frac{\partial P}{\partial u_x} - \frac{\partial}{\partial y} \frac{\partial P}{\partial u_y} = 0. \quad (3.15)$$

Daí,

$$\frac{\partial P}{\partial u} = 2 \frac{\beta}{2} (u - I) = \beta(u - I).$$

Pois,

$$\frac{\partial P}{\partial u_x} = \frac{2u_x}{2\sqrt{u_x^2 + u_y^2}}$$

$$\frac{\partial}{\partial x} \frac{\partial P}{\partial u_x} = \frac{u_{xx} \sqrt{u_x^2 + u_y^2} u_x \left(\frac{2u_x u_{xx} + 2u_y u_{xy}}{2\sqrt{u_x^2 + u_y^2}} \right)}{u_x^2 + u_y^2} = \frac{u_{xx} |\nabla u| - u_x \left(\frac{u_x u_{xx} + u_y u_{xy}}{|\nabla u|} \right)}{|\nabla u|^2}$$

$$\frac{\partial P}{\partial u_y} = \frac{2u_y}{2\sqrt{u_x^2 + u_y^2}}$$

$$\frac{\partial}{\partial y} \frac{\partial P}{\partial u_y} = \frac{u_{yy} \sqrt{u_x^2 + u_y^2} u_y \left(\frac{2u_x u_{xy} + 2u_y u_{yy}}{2\sqrt{u_x^2 + u_y^2}} \right)}{u_x^2 + u_y^2} = \frac{u_{yy} |\nabla u| - u_y \left(\frac{u_x u_{xy} + u_y u_{yy}}{|\nabla u|} \right)}{|\nabla u|^2}$$

Assim a equação (3.15) torna-se:

$$0 = \beta |u - I| - \left(\frac{u_{xx} |\nabla u| - u_x \left(\frac{u_x u_{xx} + u_y u_{xy}}{|\nabla u|} \right) + u_{yy} |\nabla u| - u_y \left(\frac{u_x u_{xy} + u_y u_{yy}}{|\nabla u|} \right)}{|\nabla u|^2} \right). \quad (3.16)$$

Por outro lado,

$$\frac{\nabla u}{|\nabla u|} = \frac{1}{|\nabla u|} (u_x, u_y) = \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}}, \frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right).$$

Assim,

$$\operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = \frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}} \right) + \frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right).$$

Como

$$\frac{\partial}{\partial x} \left(\frac{u_x}{\sqrt{u_x^2 + u_y^2}} \right) = \frac{u_{xx} \sqrt{u_x^2 + u_y^2} - u_x \left(\frac{2u_x u_{xx} + 2u_y u_{xy}}{2\sqrt{u_x^2 + u_y^2}} \right)}{u_x^2 + u_y^2} = \frac{u_{xx} |\nabla u| - u_x \left(\frac{u_x u_{xx} + u_y u_{xy}}{|\nabla u|} \right)}{|\nabla u|^2}$$

e

$$\frac{\partial}{\partial y} \left(\frac{u_y}{\sqrt{u_x^2 + u_y^2}} \right) = \frac{u_{yy} \sqrt{u_x^2 + u_y^2} - u_y \left(\frac{2u_x u_{xy} + 2u_y u_{yy}}{2\sqrt{u_x^2 + u_y^2}} \right)}{u_x^2 + u_y^2} = \frac{u_{yy} |\nabla u| - u_y \left(\frac{u_x u_{xy} + u_y u_{yy}}{|\nabla u|} \right)}{|\nabla u|^2}$$

temos:

$$\operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = \frac{u_{xx} |\nabla u| - u_x \left(\frac{u_x u_{xx} + u_y u_{xy}}{|\nabla u|} \right) + u_{yy} |\nabla u| - u_y \left(\frac{u_x u_{xy} + u_y u_{yy}}{|\nabla u|} \right)}{|\nabla u|^2}. \quad (3.17)$$

De (3.16) e (3.17), encontramos:

$$\beta(u - I) - \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) = 0.$$

Portanto, a minimização do funcional $E[u]$ é obtida resolvendo-se :

$$u_t = \operatorname{div} \left(\frac{\nabla u}{|\nabla u|} \right) - \beta(u - I), \quad (3.18)$$

com $u(x, 0) = I(x)$, $x \in \mathbb{R}^2$.

No domínio de retoque, o modelo aplica um simples processo de difusão anisotrópica dado por:

$$\frac{\partial u}{\partial t} = \operatorname{div} \cdot \left(\frac{\nabla u}{|\nabla u|} \right), \quad (3.19)$$

o qual, vem sendo estudado extensivamente nos últimos anos.

Da equação (3.18), na ausência de ruído, como é o caso do nosso trabalho, (isto é, $\beta = 0$ dentro do domínio de retoque), o modelo de retoque Variacional Total é morfologicamente invariante. O modelo exige apenas evoluir a equação (3.19) no tempo. Sendo assim, é adicionado o termo $|\nabla u|$ para balancear a equação, tornando-se então:

$$\frac{\partial u}{\partial t} = |\nabla u| \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right),$$

que é exatamente o movimento da curvatura média e é muito útil por apresentar convergência numérica rápida.

Podemos observar também que o modelo TV é exatamente igual ao termo de difusão, logo sua discretização já foi mostrada na seção 3.2.

Na seção seguinte, apresentamos o modelo CDD.

3.5 Modelo da Difusão Guiada pela Curvatura (CDD)

Apesar da qualidade do mecanismo de transporte ao longo das linhas de nível ser um esquema estável, análises matemáticas no modelo BSBC mostram-se muito difíceis. Entretanto, inspirados neste fato, Chan e Shen propuseram um modelo de retoque, o qual é fundamentado no princípio variacional, desde que o funcional energia seja baseado no modelo TV.

Com isso, o modelo CDD proposto por Chan e Shen é um aprimoramento da difusão anisotrópica do modelo TV, tendo como objetivo obter conexão à longa distância, isto é, atender ao Princípio da Conectividade. Este princípio estabelece que todas as regiões incompletas sejam totalmente conectadas, ao invés de aparecerem partes separadas (mais detalhes em [13]). Além disso, o modelo de retoque CDD emprega informações através da curvatura na difusão.

O modelo é dado pela EDP:

$$\frac{\partial u}{\partial t} = \nabla \cdot \left(\frac{g(k)}{|\nabla u|} \nabla u \right); \quad k = \nabla \cdot \frac{\nabla u}{|\nabla u|} \quad (3.20)$$

onde $g : \mathbb{R} \rightarrow [0, +\infty)$ é uma função contínua, satisfazendo: $g(0) = 0$ e $g(\pm\infty) = +\infty$. Estas características da g definem uma função dita “aniquiladora” de grandes curvaturas. A introdução de $g(k)$ é para suavizar as linhas de nível, desde que $D = \frac{g(k)}{|\nabla u|}$ denota o coeficiente de condutividade. Este coeficiente de condutividade é dado por $\frac{g(k)}{|\nabla u|}$, ao invés de $\frac{1}{|\nabla u|}$ como no modelo TV, o qual depende da curvatura das linhas de nível. Por essa razão o modelo é denominado CDD (difusão guiada pela curvatura).

Fundamentalmente, a grande vantagem do modelo CDD sobre o modelo TV, é o fato deste obedecer o Princípio da Conectividade e, ao mesmo tempo, ser sensível a ruídos, ou seja, o modelo CDD elimina eventuais ruídos em imagens digitais.

A escolha de $g(k) = 1$ para pontos fora do domínio de retoque indica que o modelo efetuará, de forma satisfatória, a eliminação do ruído fora do domínio Ω , exatamente como no modelo TV. Enquanto isso, g poderá ser qualquer função apropriada que “eliminará” as grandes curvaturas e estabilizará as pequenas curvaturas dentro do domínio de retoque. Esta é a razão pela qual definimos g como sendo uma função “aniquiladora” de grandes curvaturas e estabilizadora de pequenas curvas.

Tendo esta condição satisfeita, o modelo estende as linhas de nível para dentro do domínio de retoque, e como resultado, tem-se objetos totalmente conectados. Dessa forma, o modelo CDD obedece de forma restrita ao Princípio da Conectividade e a equação (3.20) reflete o processo de Retoque Digital com redução de ruídos.

Um exemplo simples, seria tomar $g(s) = |s|^p, \forall p > 0$.

A discretização do modelo CDD também é análoga ao termo de difusão, a menos do acréscimo da função g que tomamos como sendo $g(k) = k^2$.

3.5. MODELO DA DIFUSÃO GUIADA PELA CURVATURA (CDD)

Neste capítulo apresentamos as equações que descrevem matematicamente o problema de Retoque Digital, bem como suas discretizações.

No próximo capítulo, mostramos os resultados obtidos com a aplicação dos modelos BSBC, TV e CDD, e também alguns comentários e observações com relação às aproximações de tais modelos.

Capítulo 4

Resultados Numéricos

Vamos aproximar os modelos matemáticos BSBC, TV e CDD por métodos numéricos de ordem de precisão mais alta que o método de Euler, que é geralmente utilizado. Para isto, utilizamos o Método Previsor-Corretor e Métodos Lineares Explícitos de Passo 2, 3 e 4.

Para aproximar o modelo BSBC pelos métodos numéricos, mencionados acima, usamos uma discretização diferente da proposta em [6] para os termos da EDP.

Tomamos uma malha regular, $h = k$, e buscamos encontrar aproximações para as derivadas utilizando nove pontos desta malha (ver Figura 4.1).

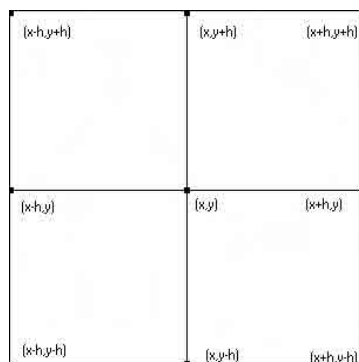


Figura 4.1

Para encontrar essas aproximações, precisamos expandir $u(x, y)$ em Série de Taylor numa vizinhança de x , isto é, nos pontos $(x + h, y)$ e $(x - h, y)$, ou seja:

$$u(x + h, y) = u(x, y) + h u_x(x, y) + \frac{h^2}{2!} u_{xx}(x, y) + \frac{h^3}{3!} u_{xxx}(x, y) + \frac{h^4}{4!} u_{xxxx}(\xi_1, y),$$

$$\xi_1 \in (x, x + h) \tag{4.1}$$

$$u(x - h, y) = u(x, y) - h u_x(x, y) + \frac{h^2}{2!} u_{xx}(x, y) - \frac{h^3}{3!} u_{xxx}(x, y) + \frac{h^4}{4!} u_{xxxx}(\xi_2, y),$$

$$\xi_2 \in (x - h, x). \tag{4.2}$$

Subtraindo (4.2) de (4.1) obtemos:

$$u(x + h, y) - u(x - h, y) = 2h u_x(x, y) + \frac{2h^3}{3!} u_{xxx}(x, y) + O(h^5)$$

$$\Rightarrow u_x(x, y) = \frac{u(x + h, y) - u(x - h, y)}{2h} + O(h^2).$$

$$\Rightarrow u_x(x, y) \approx \frac{u(x + h, y) - u(x - h, y)}{2h}. \tag{4.3}$$

Em (4.3) encontramos uma aproximação para $u_x(x, y)$.

Por outro lado, se expandirmos em Série de Taylor $u_x(x, y)$ em torno dos pontos $(x, y + h)$ e $(x, y - h)$, produzimos:

$$u_x(x, y + h) = u_x(x, y) + h u_{xy}(x, y) + \frac{h^2}{2!} u_{xyy}(x, y) + \frac{h^3}{3!} u_{xyyy}(x, y) + \frac{h^4}{4!} u_{xyyyy}(x, \xi_3),$$

$$\xi_3 \in (y, y + h) \tag{4.4}$$

e

$$u_x(x, y - h) = u_x(x, y) - h u_{xy}(x, y) + \frac{h^2}{2!} u_{xyy}(x, y) - \frac{h^3}{3!} u_{xyyy}(x, y) + \frac{h^4}{4!} u_{xyyyy}(x, \xi_4),$$

$$\xi_4 \in (y - h, y). \tag{4.5}$$

Somando (4.4) e (4.5) temos:

$$u_x(x, y) = \frac{u_x(x, y + h) + u_x(x, y - h)}{2} + O(h^2)$$

$$\Rightarrow u_x(x, y) \approx \frac{u_x(x, y + h) + u_x(x, y - h)}{2} \tag{4.6}$$

Sendo assim, encontramos uma outra aproximação para $u_x(x, y)$.

Logo, se substituirmos a aproximação encontrada em (4.3) na expressão (4.6), encontramos:

$$u_x(x, y) \approx \frac{u_x(x, y + h) + u_x(x, y - h)}{2} \approx \frac{u(x + h, y - h)}{4h} + \frac{u(x + h, y + h) - u(x - h, y - h) - u(x - h, y + h)}{4h}. \quad (4.7)$$

Como foi dito anteriormente, buscamos uma aproximação para a derivada e como encontramos duas, vamos fazer uma média entre elas. Para isto, somamos (4.3) com (4.7) e dividimos por 2. Após algumas manipulações algébricas, chegamos ao resultado da discretização desejada, isto é:

$$u_x(x, y) \approx \frac{1}{4h} \left[u(x + h, y) - u(x - h, y) + \frac{1}{2} \left(u(x + h, y - h) - u(x - h, y - h) + u(x + h, y + h) - u(x - h, y + h) \right) \right]. \quad (4.8)$$

Essa discretização é da ordem h^2 , pois se expandirmos em Série de Taylor os termos $u(x + h, y)$, $u(x - h, y)$, $u(x + h, y - h)$, $u(x - h, y - h)$, $u(x + h, y + h)$ e $u(x - h, y + h)$, encontramos:

$$\frac{1}{4h} \left[u(x + h, y) - u(x - h, y) + \frac{1}{2} \left(u(x + h, y - h) - u(x - h, y - h) + u(x + h, y + h) - u(x - h, y + h) \right) \right] = u_x(x, y) + O(h^2) \quad (4.9)$$

Analogamente, encontramos as outras aproximações como sendo:

$$u_y(x, y) \approx \frac{1}{4h} \left[u(x, y + h) - u(x, y - h) + \frac{1}{2} \left(u(x + h, y - h) - u(x - h, y - h) + u(x + h, y + h) - u(x - h, y + h) \right) \right] \quad (4.10)$$

$$u_{xx}(x, y) \approx \frac{1}{2h^2} \left[u(x + h, y) - 2u(x, y) + u(x - h, y) - u(x, y + h) - u(x, y - h) + \frac{1}{2} \left(u(x + h, y + h) + u(x - h, y - h) + u(x + h, y - h) + u(x - h, y + h) \right) \right]. \quad (4.11)$$

$$u_{yy}(x, y) \approx \frac{1}{2h^2} \left[u(x, y + h) - 2u(x, y) + u(x, y - h) - u(x + h, y) - u(x - h, y) + \frac{1}{2} \left(u(x + h, y + h) + u(x - h, y - h) + u(x + h, y - h) + u(x - h, y + h) \right) \right] \quad (4.12)$$

$$\begin{aligned}
 u_{xy}(x, y) \approx & \frac{1}{32h^2} \left[4u(x+h, y+h) + 4u(x-h, y-h) - 4u(x+h, y-h) \right. \\
 & - 4u(x-h, y+h) + u(x+h, y+2h) - u(x-h, y+2h) \\
 & + u(x-h, y-2h) + u(x+h, y-2h) + u(x-2h, y-h) \\
 & \left. - u(x-2h, y+h) + u(x+2h, y+h) - u(x+2h, y-h) \right]. \quad (4.13)
 \end{aligned}$$

Do mesmo modo que para u_x , as aproximações para as derivadas u_y , u_{xx} , u_{yy} e u_{xy} também tem ordem de h^2 .

Queremos ressaltar que a magnetude do vetor u foi calculada, da forma tradicional, como sendo:

$$|\nabla u^n(i, j)| = \sqrt{u_x^2(i, j) + u_y^2(i, j)}.$$

Apesar de ambas as discretizações apresentarem ordem h^2 , com a discretização das médias não foi necessário a técnica de regressão utilizada em [6], para o cálculo de $|\nabla u^n(i, j)|$. Isto foi comprovado apenas por testes computacionais.

Os demais termos do modelo BSBC foram calculados usando as discretizações para u_x e u_y dadas por (4.8) e (4.10), propostos pelos autores. Do mesmo modo, para os modelos TV e CDD, utilizamos as discretizações das médias dadas anteriormente.

Vamos então usar essas discretizações das médias tanto para a versão utilizando o Método Previsor-Corretor quanto para os Métodos Lineares Explícitos de Passo 2, 3 e 4.

Vale ressaltar que consideramos uma imagem como sendo uma matriz $m \times n$, onde o valor de cada elemento corresponde a uma tonalidade de cinza no “pixel” correspondente. A escala h , corresponde ao espaçamento espacial nas direções horizontal e vertical. Como esse espaçamento corresponde a distância entre os “pixels” horizontais e verticais, tomamos essa distância como sendo 1, correspondente a 1 unidade de “pixel”. Desta forma, tomamos, neste trabalho $h = 1$.

Para o Método Previsor-Corretor, tomamos o Método Euler como previsor e o Método Euler Melhorado como corretor, chegando ao seguinte esquema numérico:

$$u^{n+1}(i, j) = u^n(i, j) + \frac{\Delta t}{2} \left[u_t^n(i, j) + u_*^{n+1}(i, j) \right] \quad (Euler\ Melhorado)$$

com

$$u_*^{n+1}(i, j) = u^n(i, j) + \Delta t u_t^n(i, j), \quad (Euler)$$

onde u_t^n é a EDP do modelo de retoque que queremos aproximar. Note que esses métodos são casos particulares dos métodos de Runge-Kutta.

Agora, para os Métodos Lineares Explícitos temos os seguintes esquemas numéricos:

Passo 2:

$$u^{n+1}(i, j) = u^n(i, j) + \frac{\Delta t}{2} \left[u_t^n(i, j) - u_t^n(i-1, j-1) \right]$$

sendo que para a primeira iteração, $u_t^n(i-1, j-1)$ foi calculada usando o método de Euler.

Passo 3:

$$u^{n+1}(i, j) = u^n(i, j) + \frac{\Delta t}{12} \left[23u_t^n(i, j) - 16u_t^n(i-1, j-1) + 5u_t^n(i-2, j-2) \right]$$

sendo que para a primeira e segunda iterações, $u_t^n(i-1, j-1)$ e $u_t^n(i-2, j-2)$, foram calculadas usando o método de Euler.

Passo 4:

$$\begin{aligned} u^{n+1}(i, j) &= u^n(i, j) + \frac{\Delta t}{24} \left[55u_t^n(i, j) - 59u_t^n(i-1, j-1) + 37u_t^n(i-2, j-2) \right. \\ &\quad \left. - 9u_t^n(i-3, j-3) \right] \end{aligned}$$

sendo que para a primeira, segunda e terceira iterações, $u_t^n(i-1, j-1)$, $u_t^n(i-2, j-2)$ e $u_t^n(i-3, j-3)$, foram calculadas usando o método de Euler.

Na próxima seção, apresentamos os resultados experimentais e alguns comentários sobre eles.

4.1 Resultados Experimentais

Em nossos experimentos trabalhamos com matrizes de dimensões 256 x 256, onde cada elemento $u_{i,j}$ da matriz é um valor real correspondente a uma tonalidade de cinza da imagem $u(x,y)$. Além disso, temos as imagens originais de cada uma das imagens utilizadas. Sendo assim, calculamos o valor absoluto da diferença entre a imagem original e a imagem reconstruída. Após essa execução, somamos todos os “pixels” correspondentes à região de retoque, dividimos pelo número de “pixels” desta região e chamamos este processo de “erro médio”. De modo semelhante, verificamos o valor máximo, em módulo, que essa diferença pode atingir e denominamos de “erro máximo”.

Destacamos aqui que os melhores resultados foram obtidos com a aplicação do par previsor-corretor. O cálculo dos erros médio e máximo, foi uma maneira, dentre outras, de mostrar que o método previsor-corretor apresentou os melhores resultados.

Vale ressaltar também que os modelos foram implementados em linguagem C++, com o compilador *DEV_C++*. Para isto, utilizamos uma máquina *ADM Athlon* 1GHz, com 256 Mb de memória, no sistema operacional Windows.

Um fato que devemos levar em consideração é o problema de contraste.



Figura 4.1: Imagem original

Grandes alterações nos valores dos “pixels” que compõem a imagem podem afetar o contraste, prejudicando a qualidade visual dessa imagem. Mostramos a seguir, um exemplo (Figura 4.2), no qual chamamos de u a imagem original e alteramos o valor de um “pixel” que se encontra na posição $u(9,20)$, de 77 para -223 (imagem da



Figura 4.2: Imagens alteradas.

esquerda) e de 77 para 377 (imagem da direita). Com isso, percebemos que realmente um único “pixel” pode modificar completamente o aspecto visual da imagem.

Alterações deste tipo serão percebidas nos próximos resultados.

Os resultados experimentais foram encontrados através das aplicações dos seguintes Métodos Numéricos: Método de Euler (ME), Método Previsor-Corretor (MPC) e Métodos Lineares Explícitos de passo 2 (MLE2), passo 3 (MLE3) e passo 4 (MLE4). Para o Método PC utilizamos o modo $P(EC)^m$, onde m é o número de vezes que aplicamos o esquema numérico e, em nosso trabalho, tomamos somente o caso $m = 1$.

Para todas as imagens apresentadas neste trabalho, aplicamos passos de difusão anisotrópica e tomamos $\Delta t=0.1$.

O primeiro exemplo trata-se de uma paisagem, cuja região de retoque possui 1205 “pixels”. Foram necessárias 70 iterações, com passos intermediários de uma difusão anisotrópica para cada iteração.

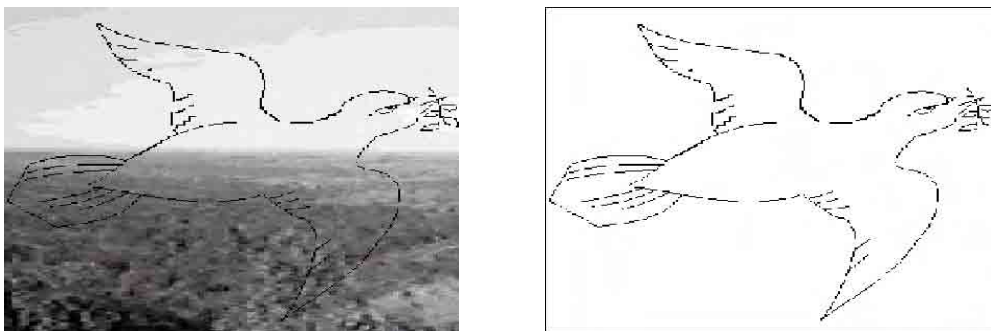


Figura 4.3: (1) e (2) - (1) Imagem a ser retocada. (2) Região a ser realçada.

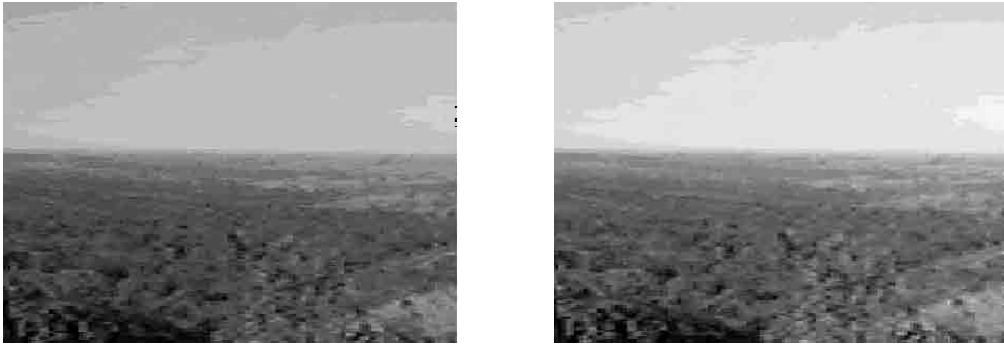


Figura 4.4: (3) e (4) - (3) Imagem reconstruída pelo método de Euler. (4) Imagem reconstruída pelo método PC.



Figura 4.5: (5) e (6) - (5) Imagem reconstruída pelo método explícito de passo 2. (6) Imagem reconstruída pelo método explícito de passo 3.



Figura 4.6: (7)- Imagem reconstruída pelo método explícito de passo 4.

Como pode ser observado, o método PC apresenta visualmente um resultado melhor. Isso também pode ser comprovado através da Tabela 4.1, que mostra o cálculo dos erros. Por outro lado, esta imagem apresenta o problema de contraste quando aproximamos pelo ME, MLE2, MLE3 e MLE4.

Método	Erro Médio	Erro Máximo
Euler	5.97	255.00
MLE2	5.22	124.70
MLE3	5.23	124.68
MLE4	5.25	124.66
PC	4.63	112.00

Tabela 4.1

Agora, utilizando a mesma imagem, mas com uma máscara contendo 2172 “pixels”, 100 iterações e com uma difusão anisotrópica por iteração.

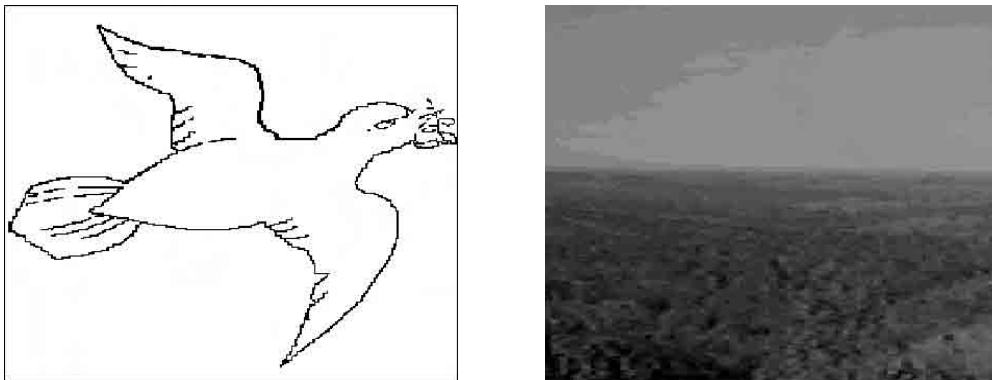


Figura 4.7: (8) e (9) - (8) Região a ser realçada. (9) Imagem reconstruída pelo método de Euler.

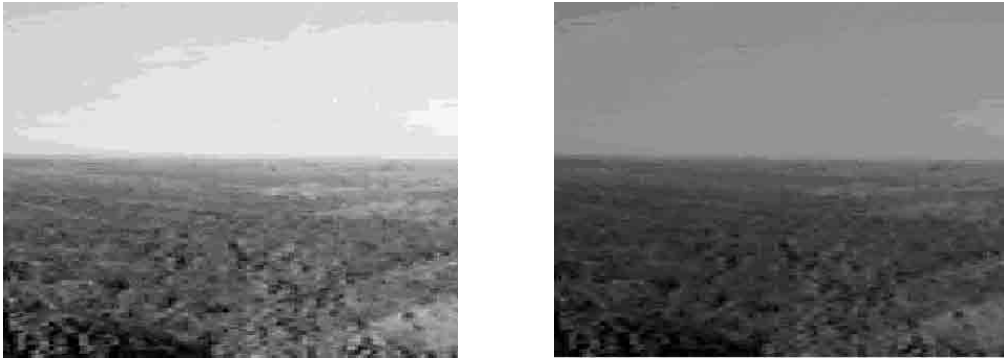


Figura 4.8: (10) e (11) - (10) Imagem reconstruída pelo método PC. (11) Imagem reconstruída pelo método explícito de passo 2.

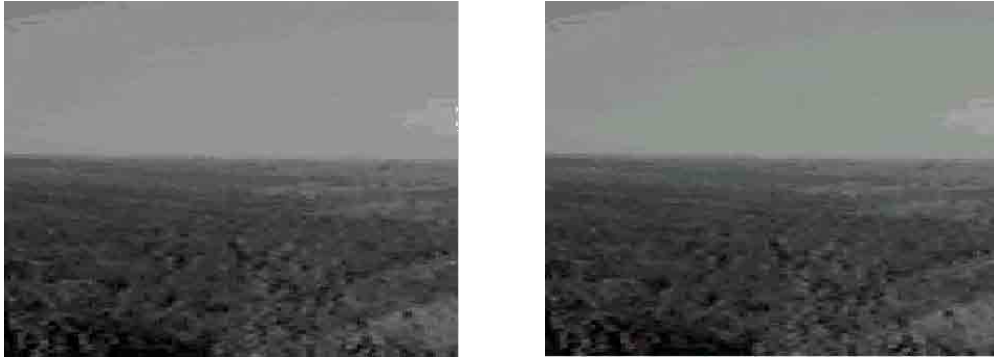


Figura 4.9: (12) e (13) - (12) Imagem reconstruída pelo método explícito de passo 3. (13) Imagem reconstruída pelo método explícito de passo 4.

Esta imagem apresentou uma grande alteração no erro máximo, retratando bem o problema de contraste citado anteriormente. Os valores dos erros, médio e máximo, estão na Tabela abaixo:

Método	Erro Médio	Erro Máximo
Euler	5.71	255.00
MLE2	5.19	152.61
MLE3	5.21	152.59
MLE4	5.22	152.57
PC	4.66	89.00

Tabela 4.2

Neste caso, como o “salto” no valor do “pixel” é maior, o efeito na qualidade visual da imagem também é maior. (ver Figuras 4.7(9), 4.8(11), 4.9(12) e 4.9(13))

No próximo exemplo, a região a ser realçada possui 15236 “pixels”. Foram utilizadas 100 iterações, com passos intermediários de cinco difusões anisotrópicas a cada iteração.

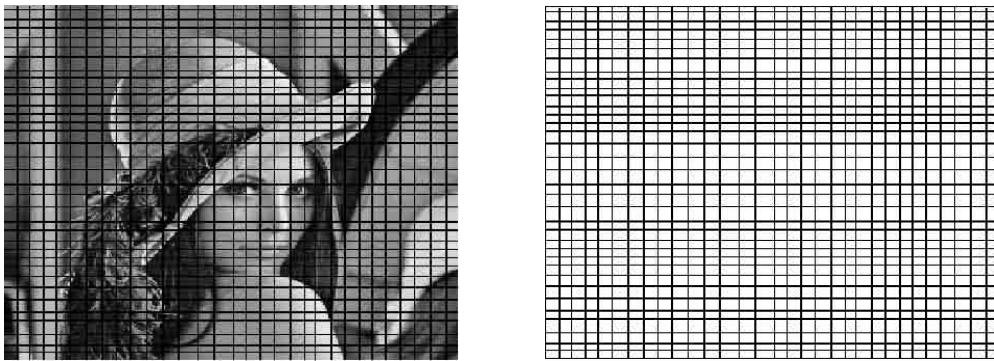


Figura 4.10: (14) e (15) - (14) Imagem a ser retocada. (15) Região a ser realçada.



Figura 4.11: (16) e (17) - (16) Imagem reconstruída pelo método de Euler. (17) Imagem reconstruída pelo método PC.



Figura 4.12: (18) e (19) - (18) Imagem reconstruída pelo método explícito de passo 2.
(19) Imagem reconstruída pelo método explícito de passo 3.



Figura 4.13: (20)- Imagem reconstruída pelo método explícito de passo 4.

Método	Erro Médio	Erro Máximo
Euler	11.09	223.00
MLE2	11.64	219.00
MLE3	12.43	219.00
MLE4	10.97	219.00
PC	9.92	191.20

Tabela 4.3

Neste exemplo, ilustramos a boa performance do método PC quando aplicado a uma região grande. Por outro lado, se considerarmos os valores dos erros máximos

da Tabela 4.3, concluímos que alterações pequenas nos valores dos “pixels” da imagem, causam pequenos problemas de contraste no resultado visual. (ver Figuras 4.11(16), 4.12(18), 4.12(19) e 4.13 (20))

Apresentamos agora uma imagem sintética, na qual realizamos 230 iterações, com quinze difusões anisotrópicas por iteração.

Para essa imagem obtemos os valores dos erros, médio e máximo, como sendo:

Método	Erro Médio	Erro Máximo
Euler	1.50	84.67
MLE2	1.47	84.50
MLE3	1.47	84.95
MLE4	1.49	84.94
PC	1.46	84.40

Tabela 4.4

Como pode ser observado, esta imagem não apresenta problemas de contraste. Observamos aqui que trata-se de uma imagem bastante simples, com poucas tonalidades de cinza em sua estrutura.

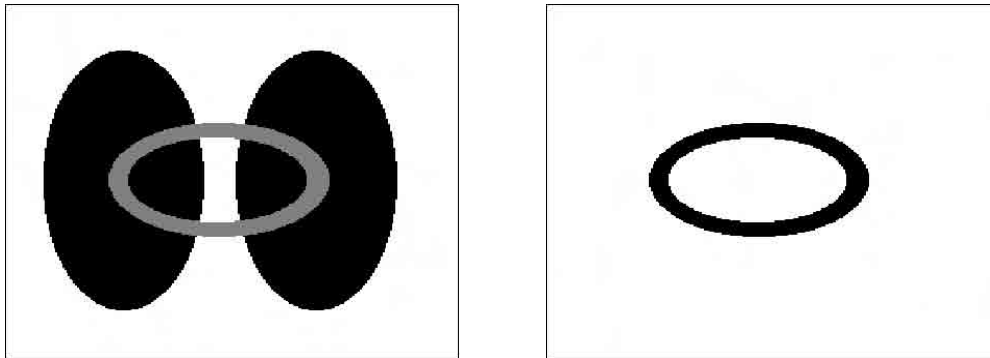


Figura 4.14: (21) e (22) - (21) Imagem a ser retocada. (22) Região a ser realçada.

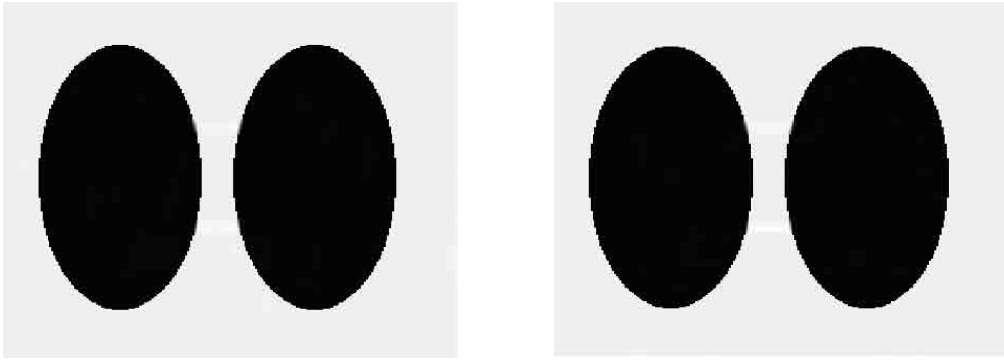


Figura 4.15: (23) e (24) - (23) Imagem reconstruída pelo método de Euler. (24) Imagem reconstruída pelo método PC.

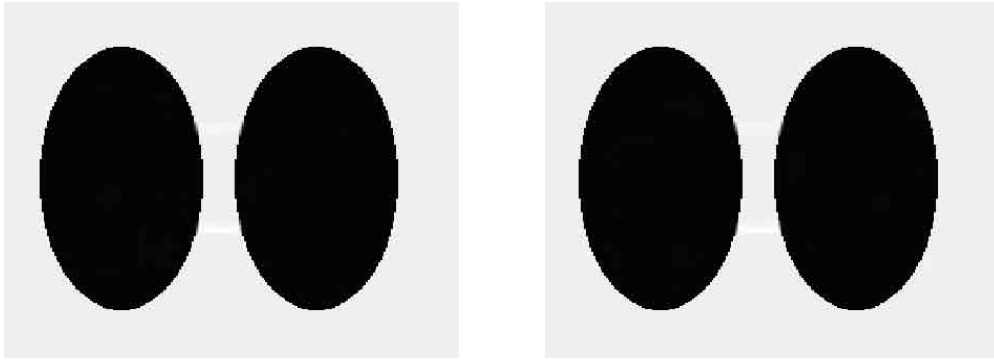


Figura 4.16: (25) e (26) - (25) Imagem reconstruída pelo método explícito de passo 2. (26) Imagem reconstruída pelo método explícito de passo 3.

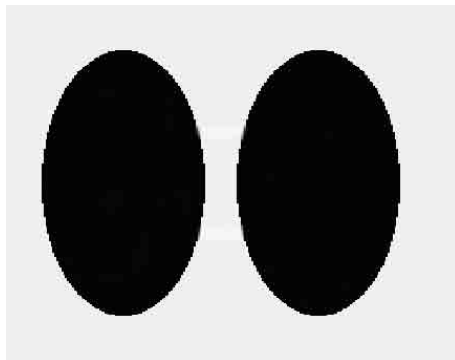


Figura 4.17: (27)- Imagem reconstruída pelo método explícito de passo 4.

A próxima imagem apresenta uma complexidade maior, pois é constituída de diferentes texturas, entretanto como no exemplo anterior, não mostra problemas no contraste. Essa imagem possui 6422 “pixels” na região de retoque e foram realizadas 100 iterações, com uma difusão anisotrópica por iteração. Para tal imagem, temos:

Método	Erro Médio	Erro Máximo
Euler	7.28	137.60
MLE2	7.16	137.58
MLE3	7.16	137.55
MLE4	7.17	137.52
PC	7.16	137.60

Tabela 4.5

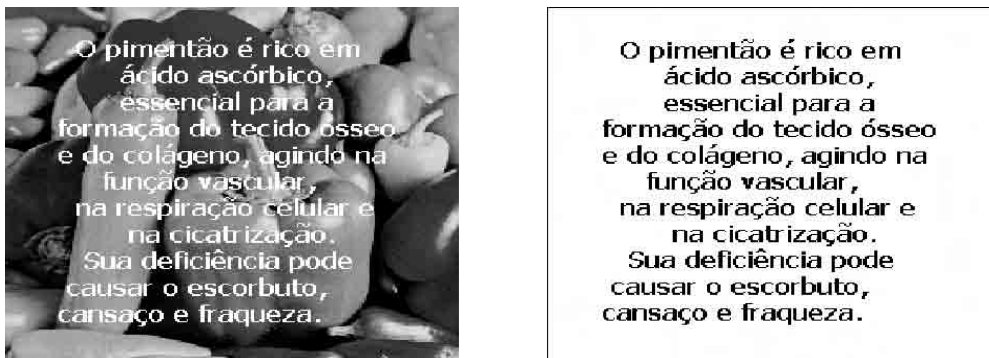


Figura 4.18: (28) e (29) - (28) Imagem a ser retocada. (29) Região a ser realçada.



Figura 4.19: (30) e (31) - (30) Imagem reconstruída pelo método de Euler. (31) Imagem reconstruída pelo método PC.



Figura 4.20: (32) e (33) - (32) Imagem reconstruída pelo método explícito de passo 2. (33) Imagem reconstruída pelo método explícito de passo 3.



Figura 4.21: (34)- Imagem reconstruída pelo método explícito de passo 4.

Observamos que outros testes computacionais foram realizados, sendo que um deles foi o aumento no número de iterações. O erro médio encontrado foi menor, mas para isso foram necessárias muito mais iterações, gastando-se um tempo computacional muito maior para encontrar um valor um pouco mais preciso. Por exemplo, para a primeira imagem (Figura 4.3(1)), utilizamos os métodos de Euler e PC, aumentamos de 70 para 800 iterações e obtemos um erro médio de 5.20 por Euler e 3.86 pelo PC. Isto mostra uma diferença de 0.77 no valor do erro, quando comparamos com os valores encontrados na Tabela 4.1, não compensando o esforço computacional necessário.

Destacamos ainda, que o Método Previsor-Corretor foi utilizado para aproximar os outros dois modelos, TV e CDD. Do mesmo modo, fizemos uma análise dos resultados comparando com o Método de Euler. Os resultados encontrados foram praticamente análogos ao do modelo BSBC e em virtude disto apresentamos apenas uma imagem para exemplificar cada modelo.

Para a segunda imagem apresentada anteriormente (Figura 4.7(8)), com a máscara de 2172 “pixels” e os mesmos parâmetros, aproximamos o modelo CDD pelos métodos de Euler e Previsor-Corretor. Com isso, encontramos um erro médio de 5.71 com o método de Euler e 4.26 pelo PC. O erro máximo foi de 90.22 para o PC e 255 por Euler.



Figura 4.22: (35) e (36) - (35) Imagem reconstruída pelo método de Euler. (36) Imagem reconstruída pelo método PC.

Comparando as Figuras 4.22(35) e 4.22(36), podemos dizer que esta imagem também apresenta sérios problemas de contraste.

A seguir, mostramos um dos resultados aplicados ao modelo TV.

Para este experimento, usamos o terceiro exemplo (Figura 4.10(14)) que aplicamos ao modelo BSBC, ou seja, uma imagem da vida real. Relembrando, essa imagem possui 15236 “pixels” na região de retoque e implementamos com os mesmos parâmetros. Baseado nestas condições, chegamos a um erro médio de 9.9 para o PC e 11.0 para Euler, enquanto o erro máximo foi de 223 por Euler e 189 pelo PC.



Figura 4.23: (37) e (38) - (37) Imagem reconstruída pelo método de Euler. (38) Imagem reconstruída pelo método PC.

Novamente, esta imagem relata uma alteração pequena nos valores dos “pixels”, resultando em uma modificação pequena na qualidade visual desta imagem.

No Capítulo 3, mencionamos a forma alternativa de implementação proposta por Batista [4], para o modelo BSBC. De maneira análoga, aproximamos esta discretização pelo método de Euler e pelo método previsor-corretor.

Para ilustrar tal forma, utilizamos a Figura 4.3(1) que contém uma região de retoque com 1205 “pixels”.

A Figura 4.24(39) foi obtida utilizando-se a forma proposta por Batista e utilizando Euler para aproximação. Foram necessárias 11000 iterações, sendo intercaladas 100 iterações de difusão a cada 10 iteração de transporte, para se chegar a esse resultado.

Por outro lado, quando aplicamos a aproximação pelo método de PC produzimos a Figura 4.24(40). Podemos observar que não foi possível eliminar completamente o desenho inserido quando aproximamos u_t pelo método de Euler, fato que não ocorre

quando aproximamos pelo método PC.

Observamos que de um modo geral, a implementação da equação dada por [6] produziu, para testes realizados, melhores resultados que a forma alternativa proposta por Batista [4], quando u_t é aproximada por Euler. Já para o método previsor-corretor ambas as formas apresentou bons resultados.

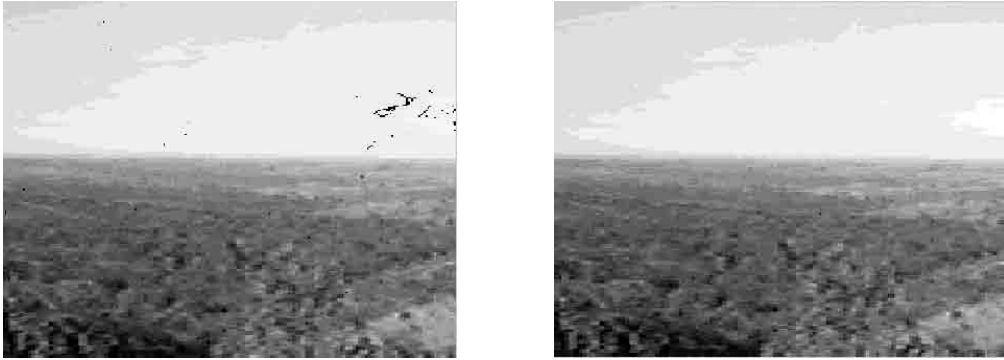


Figura 4.24: (39) e (40) - (39) Imagem reconstruída pelo método de Euler. (40) Imagem reconstruída pelo método PC.

Capítulo 5

Conclusão

No início deste trabalho fizemos um estudo sobre as equações diferenciais ordinárias e como resolvê-las através dos métodos numéricos. Tal estudo foi realizado, visando aproximar as EDP dos modelos de Retoque Digital, considerando t como uma variável no tempo.

As equações diferenciais parciais representam uma ferramenta alternativa e importante no contexto de processamento digital de imagens, pois visam explorar o caráter explícito das equações, proporcionando um baixo custo computacional.

Com relação ao modelo BSBC, podemos observar, através dos resultados mostrados no capítulo anterior, que aproximando-o pelo método previsor-corretor as imagens reconstruídas são visualmente melhores ou apresentam no mínimo resultados semelhantes quando comparado com os resultados das aproximações realizadas através dos métodos de Euler e dos lineares explícitos de passo 2, passo 3 e passo 4.

De maneira análoga, o método previsor-corretor mostrou-se mais preciso quando analisamos os erros, médio e máximo, definidos no Capítulo 4, com relação aos outros métodos.

Um outro fato diz respeito aos modelos CDD e TV, pois encontramos resultados semelhantes ao do modelo BSBC, quando aproximamos pelo método previsor-corretor. Para verificar tal fato, temos as figuras: Figura 4.22(36) e Figura 4.23(38).

Queremos ressaltar que outros testes computacionais foram realizados, sendo que um deles refere-se ao aumento no número de iterações. O exemplo citado no

Capítulo 4, caracteriza bem esta mudança, pois ao evoluirmos de 70 para 800 iterações, os resultados encontrados foram bem próximos, mas com um tempo computacional relativamente grande, não justificando, portanto, o aumento do número de iterações. Um outro teste, foi a forma alternativa de implementar o modelo BSBC, a qual foi proposta por Batista [4]. Esta forma alternativa apresentou resultados inferiores aos encontrados com a forma proposta em [6].

Uma outra vantagem da aplicação do método previsor-corretor é que ele não apresentou o problema de contraste. Ao contrário disto, o método de Euler e os métodos lineares de passo 2, passo 3 e passo 4 apresentaram grandes alterações nos “pixels”, prejudicando visualmente a qualidade das imagens reconstruídas. Este fato pode ser verificado nas figuras: Figura 4.4(3), Figuras 4.5(5)-(6), Figura 4.6(7), Figura 4.7(9), Figura 4.8(10), Figuras 4.9(12)-(13) e Figura 4.22(35).

Gostaríamos de concluir este trabalho, deixando claro que há muito o que se fazer para evoluir nesta área de processamento de imagens. Logo, como desafio futuro propomos a utilização de métodos numéricos com ordem de aproximação mais alta, que possa evitar, ou pelo menos reduzir, o problema de contraste e também tenha um baixo custo computacional, pois apesar do método previsor-corretor ter apresentado resultados melhores quando comparados com o método de Euler, seu custo computacional é maior.

Esperamos ter contribuído de alguma forma com nosso trabalho.

Referências Bibliográficas

- [1] ALVAREZ L., LIONS P. L. e MOREL J. M. Image selective smoothing and edge detection by nonlinear diffusion. *SIAM J. Numer. Anal.*, p. 845-866, 1992.
- [2] ATKINSON K. E., *An Introduction to Numerical Analysis*, John Wiley and Sons, 1988.
- [3] BALLESTER C., BERTALMÍO M., SAPIRO G. e CASELLES V. Filling-in by joint interpolation of vector fields and grey levels, *Computer Graphics, SIGGRAPH*, University of Minnesota IMA TR, April 2000.
- [4] BATISTA M. A. *Retoque Digital*. Dissertação de Mestrado, Universidade Federal de Uberlândia, 2003.
- [5] BERTALMÍO M., *Processing of flat and non-flat image information on arbitrary mani-folds using Partial Differential Equations*, Tese de Doutorado , University of Minnesota, 2001.
- [6] BERTALMÍO M., SAPIRO G., CASELLES V. e BALLESTER C. Image Inpainting. *Computer Graphics, SIGGRAPH*, New Orleans, p. 417-424, July 2000.
- [7] BURDEN R. L., e FAIRES J. D., *Numerical Analysis*, PWS-KENT Publishing Company, 1993.
- [8] BUTCHER J. C., *The Numerical Analysis of Ordinary Differential Equations*, John Wiley and Sons, 1987.
- [9] CHAN T. e SHEN J., Morphologically Invariant PDE Inpaintings. *J. W. Arrowsmith Ltd*, Bristol, 1973.

- [10] CUNHA, M.C.C., *Métodos Numéricos*, 2^a Edição, Editora da UNICAMP, São Paulo, 2000.
- [11] GONZALEZ R. e WOODS R. E., *Processamento de Imagens Digitais*, Edgard Blücher, 2000.
- [12] HENRICI P., *Discrete Variable Methods in Ordinary Differential Equations*, John Wiley and Sons, 1962.
- [13] KANIZSA, G. Organization in vision: Essays on gestalt perception, *Praeger*, New York, 1979.
- [14] LAMBERT J. D., *Computational Methods in Ordinary Differential Equations*, John Wiley and Sons, 1973.
- [15] OSHER S. e SETHIAN J. A. Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations, *Journal of Computation Physics*, v. 79, p. 12-49, 1988.
- [16] PERONA P. e MALIK J. Scale-space and edge detection using anisotropic diffusion, *IEEE-PAMI*, v.12, p. 629-639, 1990.
- [17] RUDIN L., OSHER S. e FATEMI E. Nonlinear total variation based noise removal algorithms, *Physica D*, v.60, p. 259-268, 1992.
- [18] SILVA Jr. E. C., *Um novo modelo de difusão não linear com preservação de bordas para segmentação e eliminação de ruídos em imagens e análise comparativa com modelos correlatos*, Dissertação de Mestrado, UNESP, São José do Rio Preto - SP, 2002.