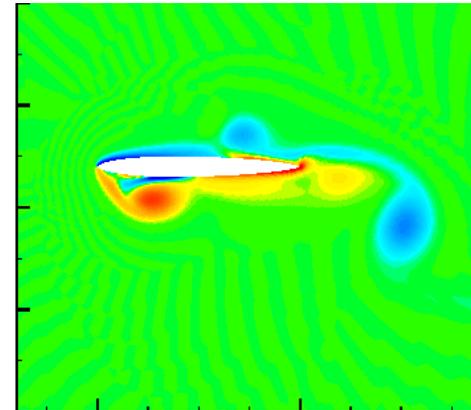
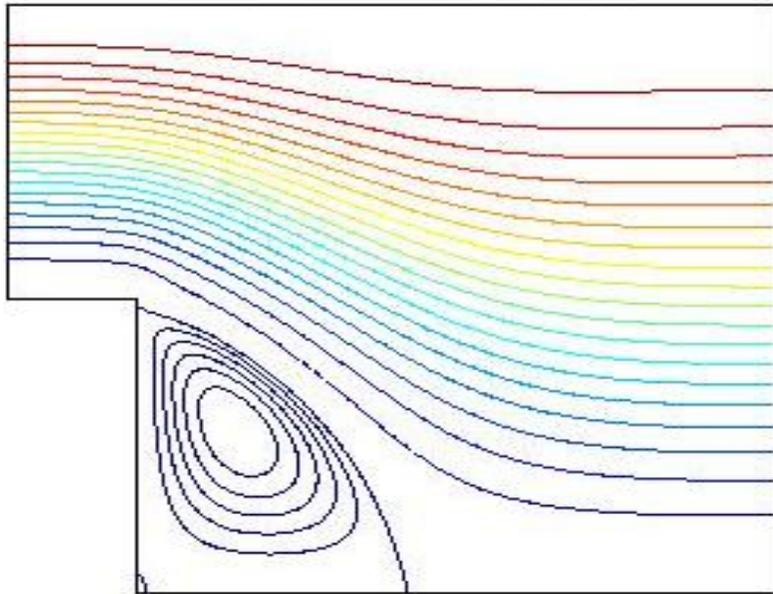


INTRODUÇÃO AO MÉTODO MULTIGRID



Prof. Diomar Cesar Lobão
UFF - Volta Redonda, RJ
Nov. 2006

O que é multigrid?

- Técnica para *acelerar a convergência* dos métodos iterativos de solução de sistemas de equações algébricas lineares e ou não-lineares.
- *Reduz em ordens de magnitude o tempo de solução de sistemas algébricos.*

Resolução numérica de sistemas algébricos

- Métodos de solução dos sistemas algébricos:
 - Diretos – ***Eliminação de Gauss, Fatoração de Crout.***
 - Iterativos - ***Jacobi, Gauss-Seidel e Sobre-relaxação.***
- Métodos iterativos são mais gerais

Resolução numérica de sistemas algébricos

- Sistema linear: $\mathbf{A}\mathbf{u} = \mathbf{b}$
- Aproximação para \mathbf{u} dada pelo método iterativo: \mathbf{v}
- Erro: $\mathbf{e} = \mathbf{u} - \mathbf{v}$
- Infelizmente, não sabemos nada do comportamento do erro! Usa-se...

Resolução numérica de sistemas algébricos

Comportamento do erro é monitorado por:

- ***Norma***: função real que fornece valor do “***comprimento***” de um vetor ou matriz
- ***Tipos de Normas*** mais comuns:

Infinita: L_∞

$$\|\mathbf{e}\|_\infty = \max_{1 \leq j \leq N} |e_j|$$

Euclidiana: L_2

$$\|\mathbf{e}\|_2 = \left\{ \sum_{j=1}^N e_j^2 \right\}^{1/2}$$

Resolução numérica de sistemas algébricos

- **Resíduo** $\mathbf{r} = \mathbf{b} - \mathbf{A}\mathbf{v}$
- **Resíduo**: mede o quanto a aproximação \mathbf{v} não *satisfaz* o sistema algébrico.

Resolução numérica de sistemas algébricos

- Substituindo-se a expressão $\mathbf{e} = \mathbf{u} - \mathbf{v}$ no sistema algébrico, obtemos:

$$\mathbf{A}\mathbf{e} = \mathbf{r}$$

- Chamada de equação do *resíduo*. Mostra que o erro “ \mathbf{e} ” satisfaz a mesma equação que o sistema algébrico, quando \mathbf{b} é substituído por \mathbf{r} .

Resolução numérica de sistemas algébricos

- A equação do **resíduo** é muito importante para *multigrid*!

$$\mathbf{Ae} = \mathbf{r}$$

- **Uma dica!**: se resolvermos a equação do **resíduo**, podemos encontrar a solução **u** a partir de

$$\mathbf{u} = \mathbf{v} + \mathbf{e}$$

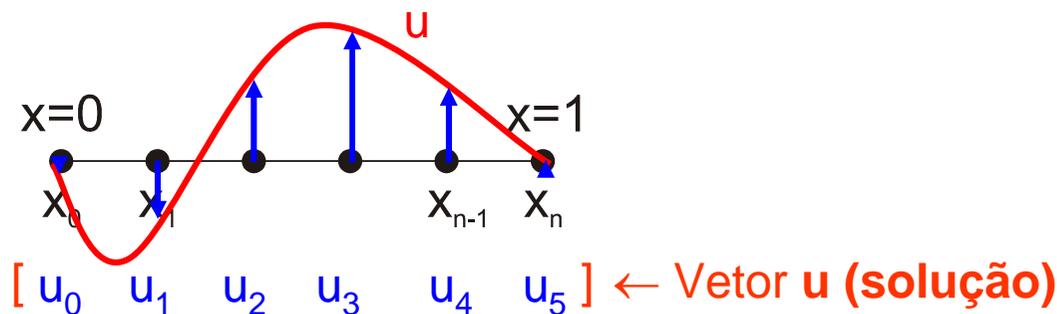
- **Problemas Modelo**
- Método de Relaxação
- Erro de convergência
- Malhas(Grids) Múltiplas-Multigrid
- Desempenho
- Considerações Teóricas

- Equação Diferencial in 1D

- $$-\frac{\partial^2 u(x)}{\partial x^2} + au(x) = f(x)$$

para $0 \leq x \leq 1$, $a > 0$ Fronteira: $u(0) = u(1) = 0$

- Discretizar* o problema contínuo em “n” *subintervalos* “avaliando” nos pontos $x_j = j\Delta h$, com $\Delta h = 1/(n)$
- Malha(Grid) M^h :



- Aproximação de diferença finita de segunda ordem

$$\frac{-v_{j-1} + 2v_j - v_{j+1}}{\Delta h^2} + av_j = f(x_j) \text{ com } 1 \leq j \leq n-1$$

$v_0 = v_n = 0$; com \mathbf{v} sendo a solução aproximada de \mathbf{u}

- Escrevendo na forma Matricial

$$\frac{1}{\Delta h^2} \begin{bmatrix} 2+ah^2 & -1 & & & \\ -1 & 2+ah^2 & -1 & & \\ & & \ddots & & \\ & & & -1 & 2+ah^2 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ \vdots \\ \vdots \\ v_{n-1} \end{bmatrix} = \begin{bmatrix} f(x_1) \\ \vdots \\ \vdots \\ f(x_{n-1}) \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ \vdots \\ f_{n-1} \end{bmatrix}$$

- Escrevendo na forma compacta: $A\mathbf{v} = \mathbf{f}$
- **Obs:** os termos para $j=0$, $j=n$ foram agregados ao lado direito

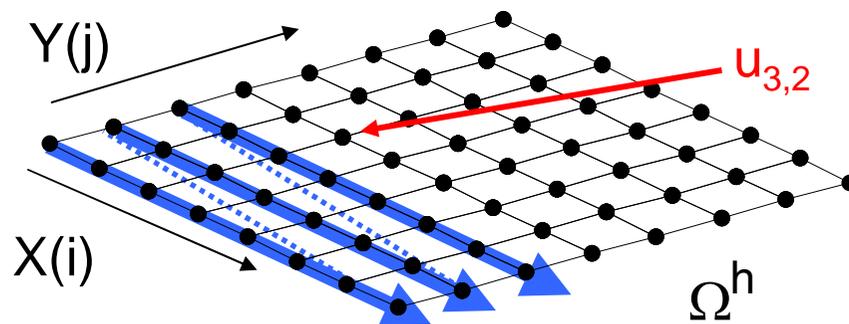
- Equação Diferencial Parcial Elíptica 2D

$$-\frac{\partial^2 u}{\partial x^2} - \frac{\partial^2 u}{\partial y^2} + au = f(x, y)$$

para $0 \leq x, y \leq 1$, $a > 0$

Fronteira: "Plano = 0"

- Discreta por uma malha bi-dimensional (com: $x(i), i=1, \dots, m-1, y(i), j=1, \dots, n-1$ *nodos interiores*)



- Aproximando em diferenças finitas de segunda ordem

$$\frac{-v_{i-1,j} + 2v_{i,j} - v_{i+1,j}}{\Delta h^2} + \frac{-v_{i,j-1} + 2v_{i,j} - v_{i,j+1}}{\Delta h^2} + av_{i,j} = f_{i,j} \text{ com } 1 \leq j \leq n-1$$

$$v_{i,0} = v_{i,n} = v_{0,j} = v_{m,j} = 0, \text{ sendo } 1 \leq i \leq m-1$$

- Escrevendo na forma Matricial

$$\begin{bmatrix} B & -\frac{1}{\Delta h^2} I & & & \\ -\frac{1}{\Delta h^2} I & B & -\frac{1}{\Delta h^2} I & & \\ & & \ddots & & \\ & & & -\frac{1}{\Delta h^2} I & B \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ \vdots \\ \vdots \\ v_{m-1} \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ \vdots \\ f_{m-1} \end{bmatrix}$$

$v_i = (v_{i1} \quad \dots \quad v_{i,n-1})^T$
 $f_i = (f_{i1} \quad \dots \quad f_{i,n-1})^T$

B parece com a matriz do problema 1D - as dimensões são $(n-1) \times (n-1)$

I é uma $(n-1) \times (n-1)$ matriz identidade

\Rightarrow Dimensão da matriz é $(m-1) \cdot (n-1) \times (m-1) \cdot (n-1)$

(Matrix tri - diagonal de Bloco)

- Exemplo: Sistema para $a=0$, $n=4$, $h=1$

$$\begin{array}{c}
 \downarrow \\
 B
 \end{array}
 \left[\begin{array}{cccccccc}
 4 & -1 & 0 & -I & & & & \\
 -1 & 4 & -1 & & -I & & 0 & \\
 0 & -1 & 4 & & & -I & & \\
 -I & & & 4 & -1 & 0 & -I & \\
 & -I & & -1 & 4 & -1 & & -I \\
 & & -I & 0 & -1 & 4 & & \\
 & & & -I & & & 4 & -1 & 0 \\
 & & 0 & & -I & & -1 & 4 & -1 \\
 & & & & & -I & 0 & -1 & 4
 \end{array} \right] \cdot \begin{bmatrix} v_{1,1} \\ v_{1,2} \\ v_{1,3} \\ v_{2,1} \\ v_{2,2} \\ v_{2,3} \\ v_{3,1} \\ v_{3,2} \\ v_{3,3} \end{bmatrix} = \begin{bmatrix} f_{1,1} \\ f_{1,2} \\ f_{1,3} \\ f_{2,1} \\ f_{2,2} \\ f_{2,3} \\ f_{3,1} \\ f_{3,2} \\ f_{3,3} \end{bmatrix}$$

- Escrevendo na forma compacta: $A\mathbf{v} = \mathbf{f}$ (montagem por coluna)

- Problemas Modelo
- **Método de Relaxação**
- Erro de convergência
- Malhas(Grids) Múltiplas-Multigrid
- Desempenho
- Considerações Teóricas

Solução numérica de EDPs

- E.D.P. modelo:

- *1 dimensão*
$$\begin{cases} \nabla^2 u = 0 & \text{em } R \\ u = 0 & \text{em } \partial R \end{cases}$$

- Discretização pelo método de diferenças finitas, fórmula dos 3 pontos, segunda ordem.

Malha computacional

- Malha computacional com $N=64$ nodos.



- Equações:

$$-u_{j-1} + 2u_j - u_{j+1} = 0, \quad 1 \leq j \leq N - 1$$

Condição de Contorno : $u_0 = u_N = 0$

Sistema linear

- Pode ser escrito como: $\mathbf{A}\mathbf{u} = \mathbf{b}$, onde:

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

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-2} \\ u_{N-1} \end{bmatrix}$$

$$\mathbf{b} = \begin{bmatrix} u_0 \\ 0 \\ \vdots \\ 0 \\ u_N \end{bmatrix}$$

Aproximação inicial da solução

- Funções seno (ou modos de Fourier)

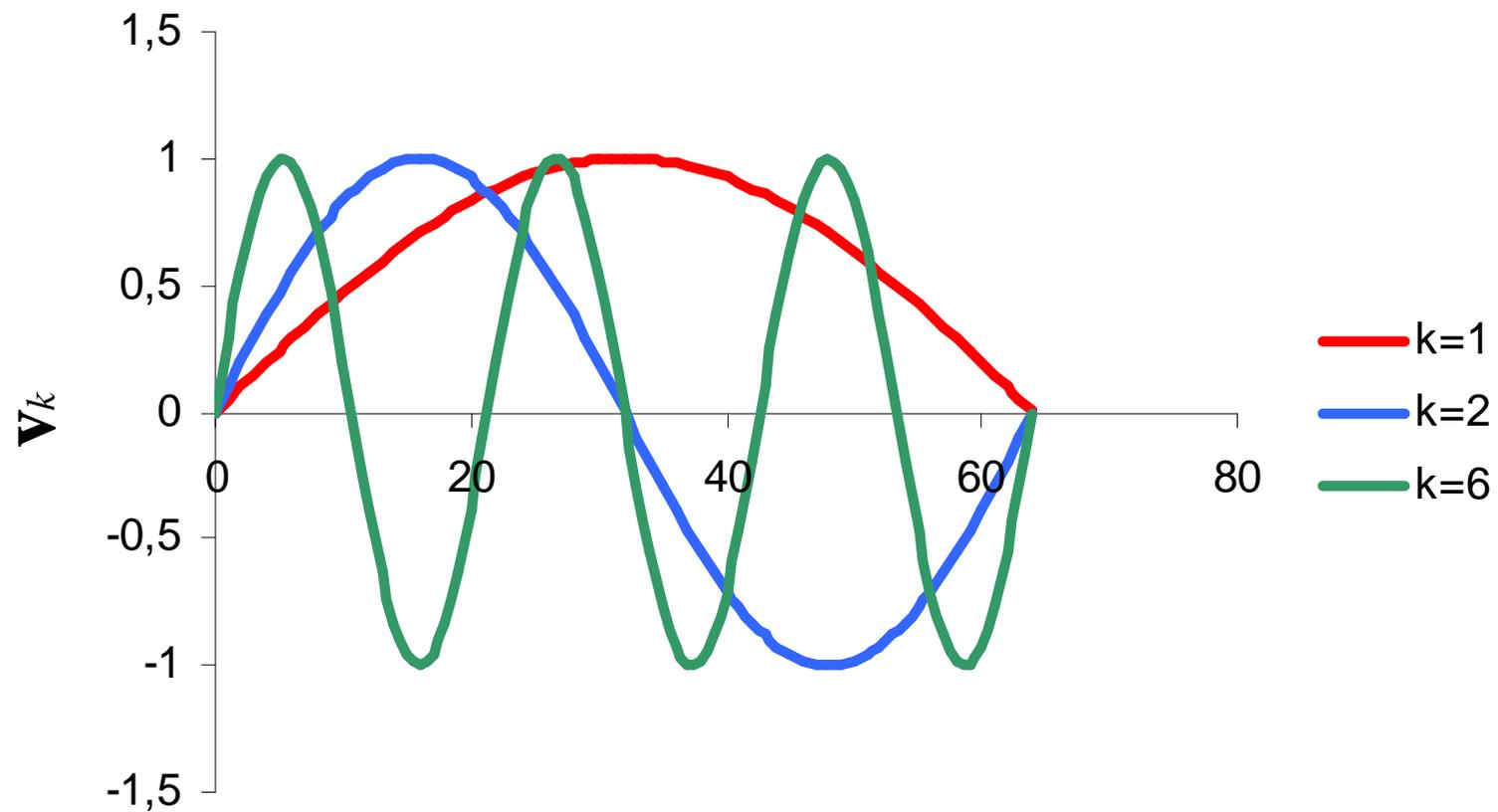
$$v_j^{(0)} = \text{sen}\left(\frac{jk\pi}{N}\right), \quad \begin{array}{l} 0 \leq j \leq N \\ 1 \leq k \leq N-1 \end{array}$$

- $v_j^{(0)}$ indica a j -ésima componente do vetor $\mathbf{v}^{(0)}$.
- k é o número de onda.
- \mathbf{v}_k é o vetor com número de onda k .

Número de onda k

- Indica quantas *meia-senóides* existem no domínio.
- Se $1 \leq k < N / 2$, os modos de Fourier são denominados *suaves*.
- Se $N / 2 \leq k \leq N - 1$, os modos de Fourier são denominados de *oscilatórios*.

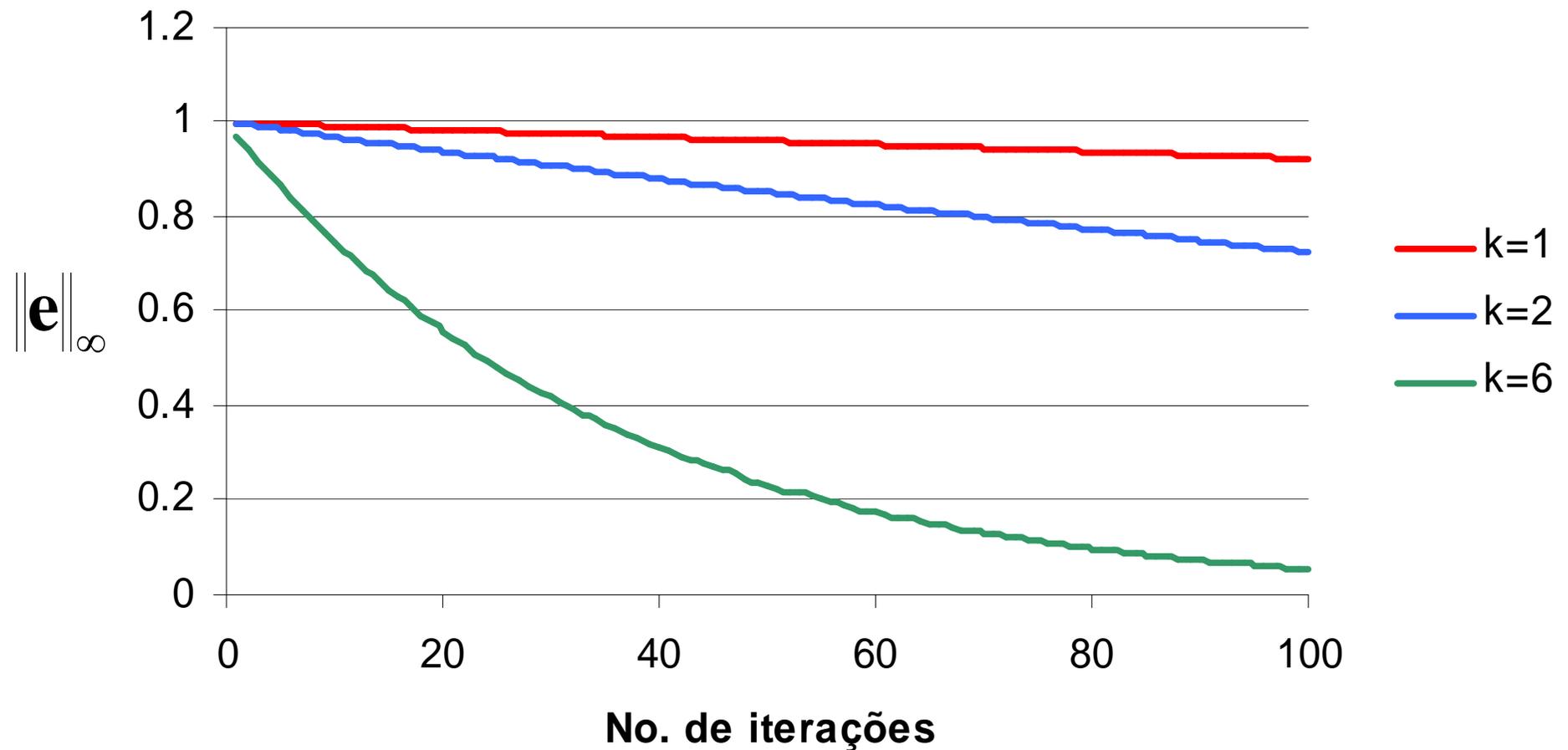
Gráfico da aproximação inicial da solução



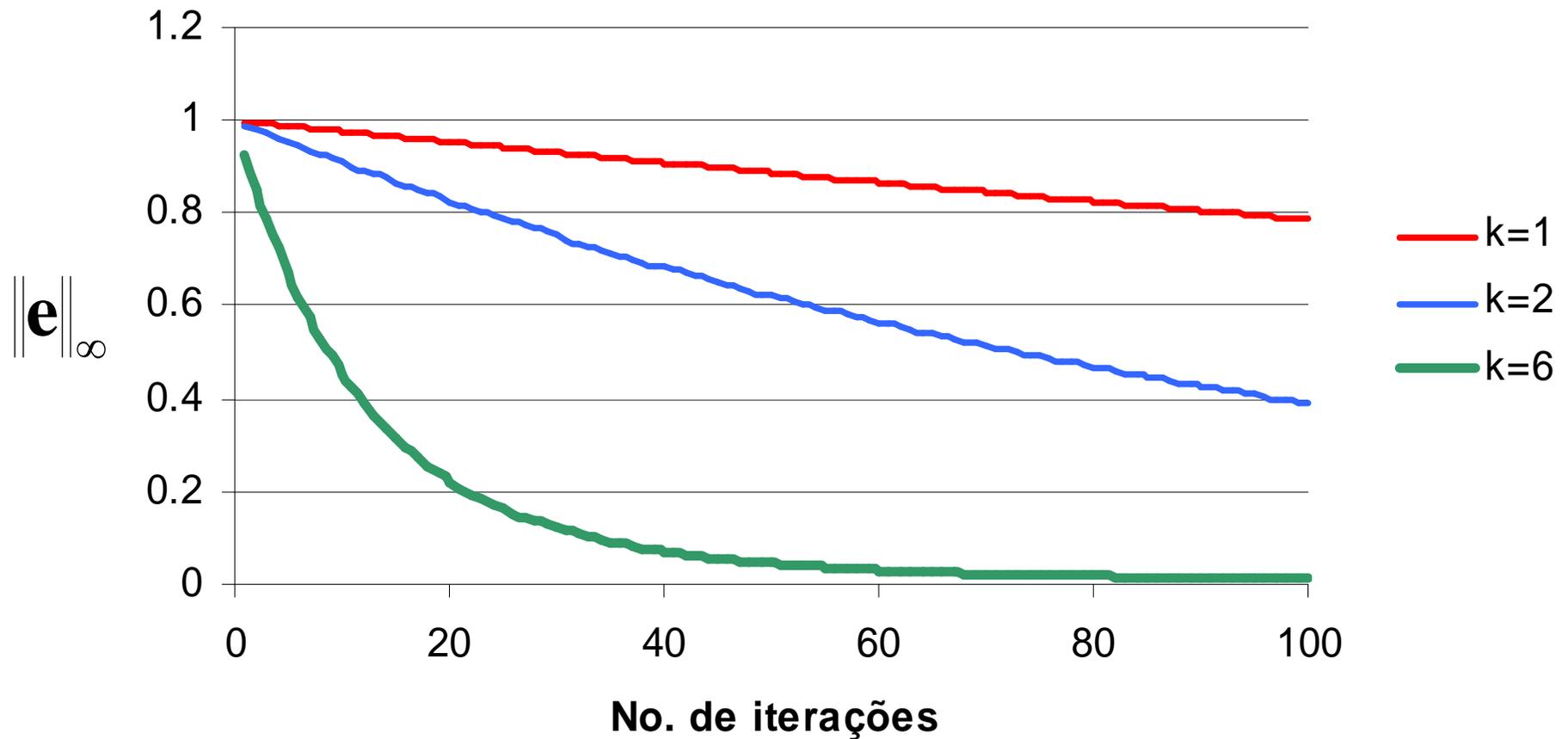
Solução do sistema algébrico

- **Gauss-Seidel**
$$u_j^{(k+1)} = \frac{u_{j-1}^{(k+1)} + u_{j+1}^{(k)}}{2}$$
- **Jacobi com Relaxação**
$$u_j^{(k+1)} = (1-\omega)u_j^{(k)} + \omega \left(\frac{u_{j-1}^{(k)} + u_{j+1}^{(k)}}{2} \right)$$
- Fator de relaxação: $0 < \omega \leq 1$
- Aproximação inicial: \mathbf{v}_1 , \mathbf{v}_2 e \mathbf{v}_6 .
- *Norma infinita* após 100 iterações:
$$\|\mathbf{e}\|_\infty = \max_{1 \leq j \leq N} |e_j|$$

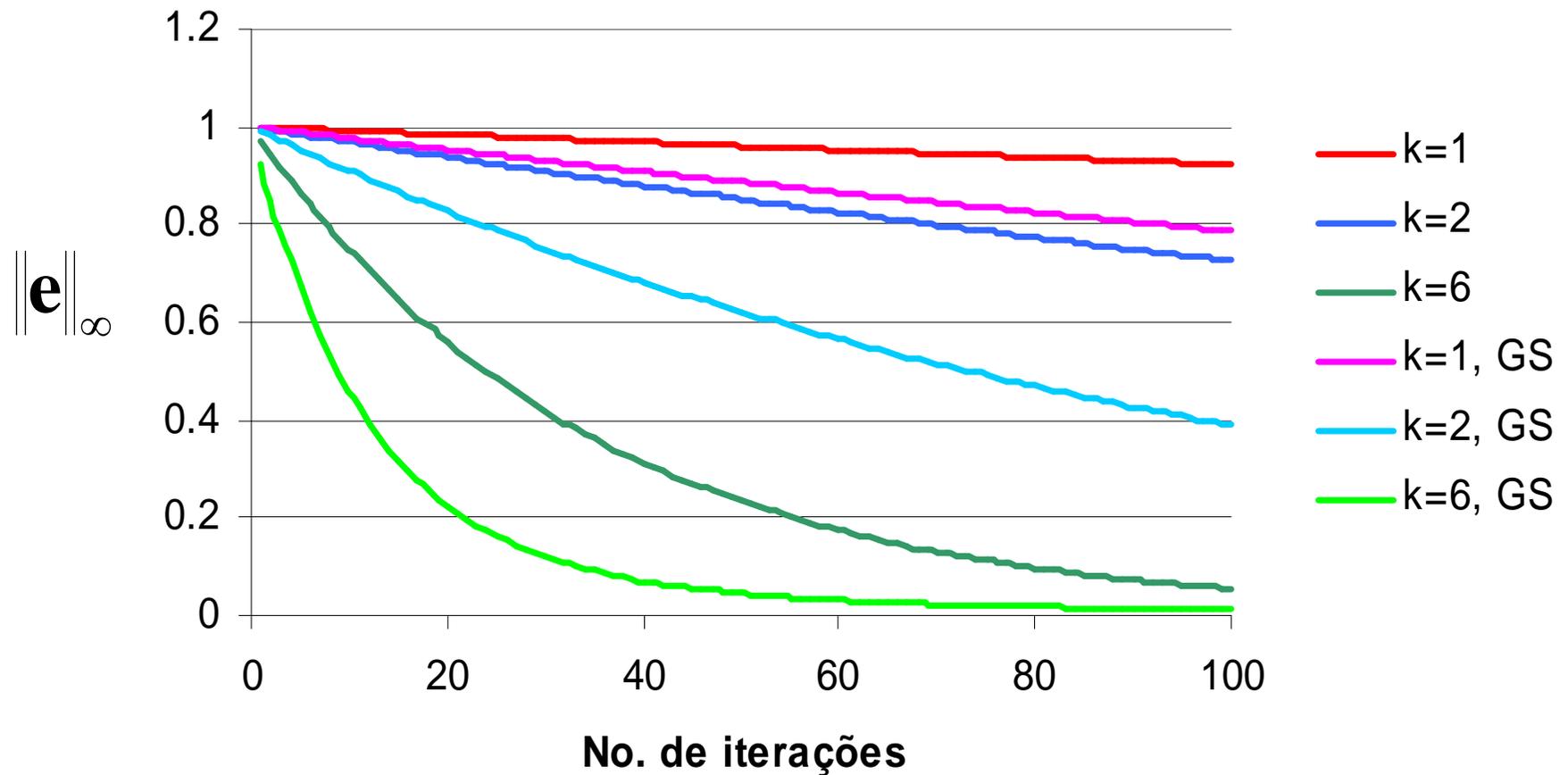
Jacobi com fator de relaxação: $\omega = 2/3$



Gauss-Seidel



Gauss-Seidel e Jacobi comparados



É importante notar...

- Que os métodos *Jacobi* e *Gauss-Seidel* (e métodos iterativos em geral):

eliminam com eficiência apenas as componentes de alta frequência do erro

- Que as componentes de *baixa frequência* são eliminadas *lentamente*.

Método Multigrid

- Problemas Modelo
- Métodos de Relaxação
- **Erro de convergência**
- Malhas (Grids) Múltiplas-Multigrid
- Desempenho
- Considerações Teóricas

- Problema Simplificado : $A\mathbf{u} = \mathbf{0}$
 $\Rightarrow \mathbf{v}$ deve convergir para $\mathbf{0}$, implica que $\rightarrow \mathbf{e} = \mathbf{u}$
- Em qual maneira **Jacobi Relaxado** decresce o error?
 \Rightarrow Análise dos *auto-vetores* da matriz de iteração
- Auto-vetores w_k das matrizes A e R_j (n - número de nodos)

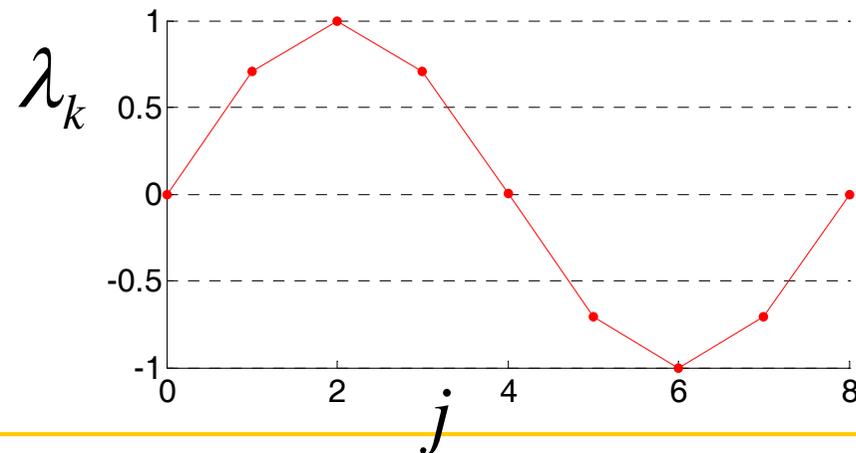
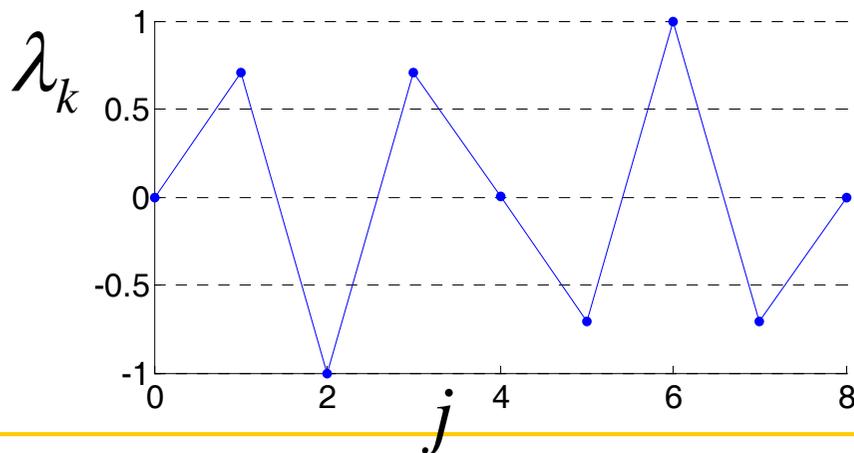
$$w_{k,j} = \sin\left(\frac{jk\pi}{n}\right), \quad \text{com } 1 \leq k \leq n-1, 0 \leq j \leq n$$

- Vetor w_k é também a $k^{\text{ésimo}}$ *modo de Fourier*
- Auto-valores λ_k da matriz R_j (geralmente: $R_j w_k = \lambda_k w_k$)

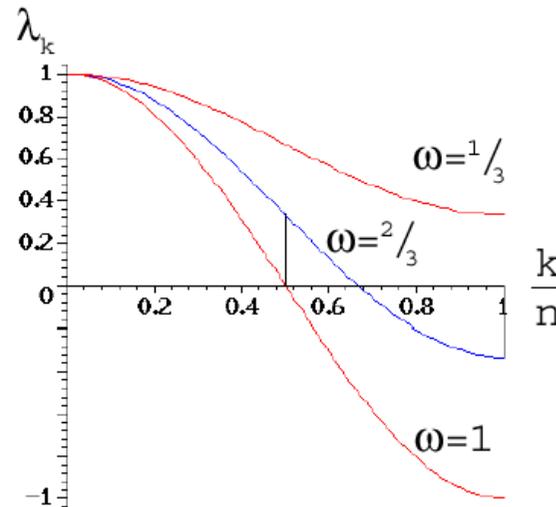
$$\lambda_k(R_\omega) = 1 - 2\omega \sin^2\left(\frac{k\pi}{2n}\right), \quad \text{com } 1 \leq k \leq n-1$$

- Para: $\lambda_k w_k < 1 \Rightarrow |\lambda_k| < 1$, iteração converge

- Auto-valores $\lambda_k(R_\omega) = 1 - 2\omega \sin^2\left(\frac{k\pi}{2n}\right)$, com $1 \leq k \leq n-1$
- Suave, baixa-freqüência modos de Fourier: $1 < k < \frac{1}{2}n$
 - $|\lambda_k|$ é próximo de 1 \Rightarrow amortecimento insatisfatório
- Oscilatório, alta-freqüência modos de Fourier.
 $\frac{1}{2}n < k < n-1$
 - Para λ , $|\lambda_k|$ próximo de 0 \Rightarrow bom amortecimento



- Diagrama de amortecimento do *método de Jacobi com Relaxação*

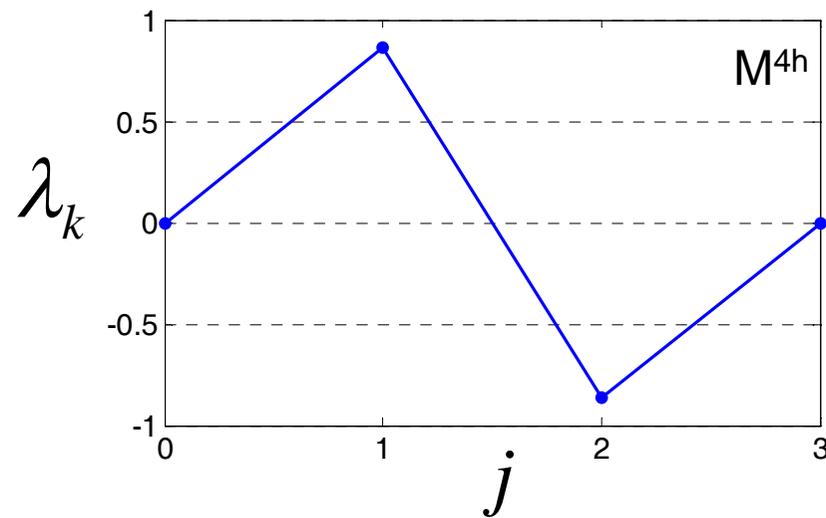
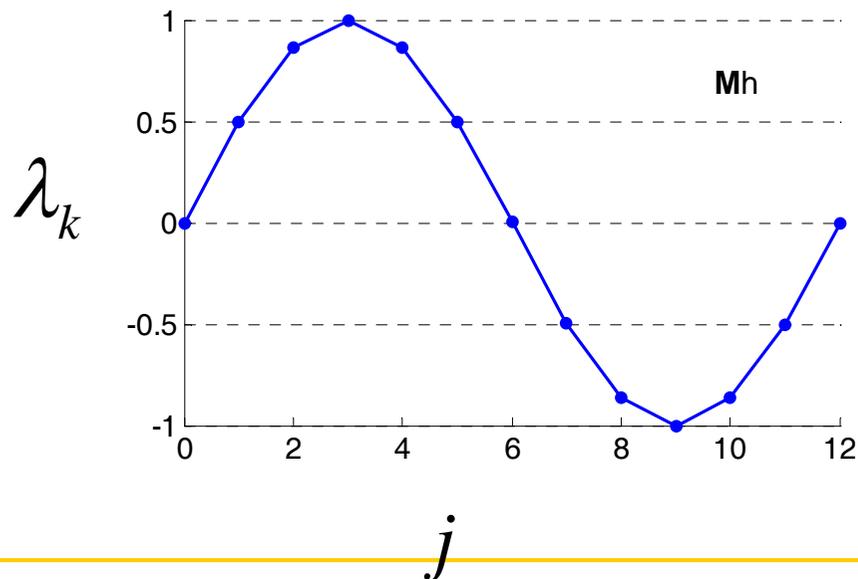


$$\lambda_k (R_\omega) = 1 - 2\omega \sin^2 \left(\frac{k\pi}{2n} \right), \quad \text{com } 1 \leq k \leq n - 1$$

- Modos Oscilatório do erro são muito bem amortecido (Associados às malhas finas).
- Modos Suave são difíceis de amortecer (Associados às malhas grossas).

- Problemas Modelo
- Método de Relaxação
- Erro de convergência
- **Malhas (Grids) Múltiplas - Multigrid**
- Desempenho
- Considerações Teóricas

- *Idéia fundamental do método multigrid*
 - Faz modos suave parecer oscilatório!
 - Modo suave em M^h parece oscilatório na malha M^{4h}



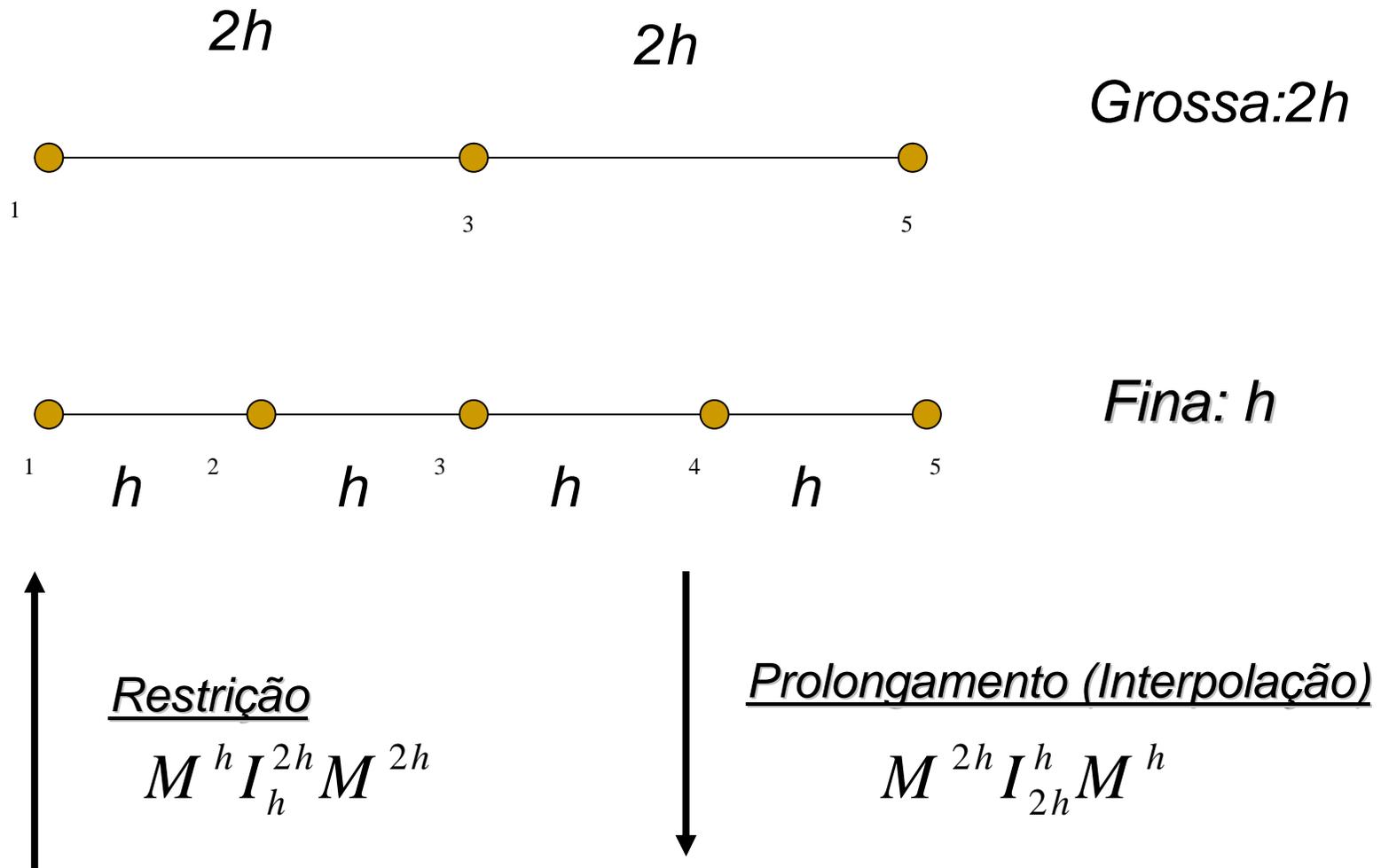
- Transferência da malha grossa p/ fina: I_{2h}^h **Prolongamento** ↓
 - $M^{2h} I_{2h}^h M^h$.
 - Método usado: *Interpolação Linear*, é efetiva

$$\begin{array}{l} \text{Fronteira:} \rightarrow \\ \text{Interior:} \rightarrow \end{array} \left\{ \begin{array}{l} v_{2j}^h = v_j^{2h} \\ v_{2j+1}^h = \frac{1}{2}(v_j^{2h} + v_{j+1}^{2h}), \quad 1 < j \leq n \end{array} \right.$$

$$I_{2h}^h \mathbf{v}^{2h} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}_{2h} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}_h = \mathbf{v}^h$$

Método Multigrid

Malhas Múltiplas



- Transferência de malha *fina* grossa: **Restrição** ↑

- $M^h \ I_h^{2h} \ M^{2h}$.

- Método simples: Injeção(Injection)

- Melhor: “Full weighting”

$$v_j^{2h} = \frac{1}{4}(v_{2j-1}^h + 2v_{2j}^h + v_{2j+1}^h), \quad 1 \leq j \leq n$$

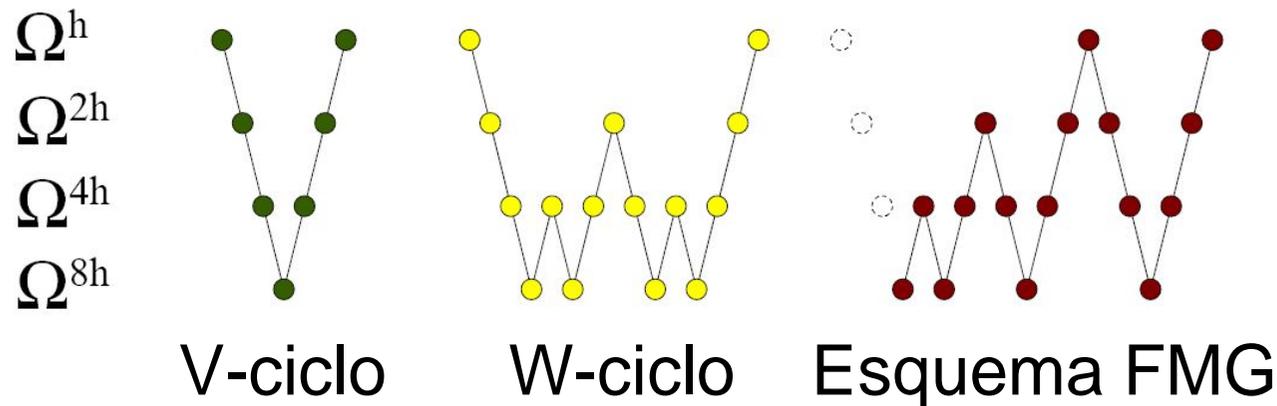
- Operador de Restrição: I_h^{2h}

$$I_h^{2h} = \frac{1}{4} \begin{bmatrix} 1 & 2 & 1 & 0 & 0 \\ 0 & 1 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 1 \end{bmatrix}; \begin{bmatrix} v_1^h \\ v_2^h \\ v_3^h \\ v_4^h \\ v_5^h \end{bmatrix}_h \quad \mathbf{v}^{2h} = I_h^{2h} \mathbf{v}^h$$

- Operação de Transferência: $M^h \ I_h^{2h} \ M^{2h}$, é suficiente

- **Erros:** *modos oscilatórios* em M^h serão representados como *modos suave* sobre M^{2h}
- *Simple* esquema de correção para “duas-malhas”
- Na malha M^h , relaxar w_1 vezes $A^h \mathbf{v}^h = \mathbf{0}$, usando $\mathbf{v}^{(0)h}$
 - **Restringir** o resíduo \mathbf{r}^h da malha fina para a malha grossa
 - Na malha M^{2h} , relaxar w_2 vezes $A^{2h} \mathbf{e}^{2h} = \mathbf{r}^{2h}$ usando $\mathbf{e}^{(0)h} = \mathbf{0}$
 - **Prolongar (Interpolar)** o erro da malha grossa M^{2h} para malha fina M^h
- Corrigir a malha-fina: $\mathbf{v}^h \leftarrow \mathbf{v}^h + \mathbf{e}^h$
- Na malha M^h , relaxar w_1 vezes $A^h \mathbf{v}^h = \mathbf{0}$ usando \mathbf{v}^h

- Estratégias *Multigrid*
 - *Iteração aninhada*: Use malha grossa p/ gear valor inicial melhorado
 - *Correção Malha Grossa*: Aproxima erro relaxando sobre a equação do resíduo na malha grossa



- Esquema V-Ciclo (Correção Malha Grossa)
 - V-Ciclo(\mathbf{v}^h , \mathbf{f}^h)
 - Relaxar w_1 vezes em $A^h\mathbf{v}^h = \mathbf{0}$ com condição inicial \mathbf{v}^h
 - If (malha atual = malha grossa) goto último ponto
 - Else: $\mathbf{f}^{2h} = \text{Restrict}(\mathbf{f}^h - A^h\mathbf{v}^h)$
 - $\mathbf{v}^{2h} = \mathbf{0}$
 - Call $\mathbf{v}^{2h} = \text{V-Ciclo}(\mathbf{v}^{2h}, \mathbf{f}^{2h})$
 - Corrigir: $\mathbf{v}^h += \text{Interpolate}(\mathbf{v}^{2h})$
 - Relaxar w_2 vezes em $A^h\mathbf{v}^h = \mathbf{0}$ com condição inicial \mathbf{v}^h

Gauss-Seidel para equação de Laplace: **Definição do operador "L"**

$$Lu_{i,j} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}$$

Definimos o resíduo (erro) como: $R_{i,j} = Lu_{i,j}$

Este resíduo tende a zero na Convergência:

$$\lim_{iter \rightarrow \infty} R_{i,j} = Lu_{i,j} = 0$$

Termo de correção escrito na forma do “Delta Form”

$$\Delta u_{i,j} = u_{i,j} - u_{i,j}^n, \text{ sendo } u_{i,j} = \text{Solução Exata}$$

$$u_{i,j}^n = \text{Solução numérica}$$

Aplicando o operador “L” no termo de correção, malha regular, obtemos:

$$L\Delta u_{i,j} = \cancel{Lu_{i,j}^0} - Lu_{i,j}^n$$

$$L\Delta u_{i,j} = -R_{i,j}$$

$$\Delta u_{i-1,j} - 4\Delta u_{i,j} + \Delta u_{i+1,j} + \Delta u_{i,j+1} + \Delta u_{i,j-1} = -R_{i,j}\Delta x^2$$

$R_{i,j}$

$$\Delta u_{i,j} = \frac{1}{4} \left[\Delta u_{i-1,j} + \Delta u_{i+1,j} + \Delta u_{i,j+1} + \Delta u_{i,j-1} + R_{i,j}\Delta x^2 \right]$$

Malha regular

- Se depois de calcular R mante-lo fixo, é possível calcular Δu que será adicionado as condições iniciais no processo de obter a solução convergida.
- Pode também atualizar R depois de cada iteração, e então a medida que o processo converge, Δu irá convergir para zero.

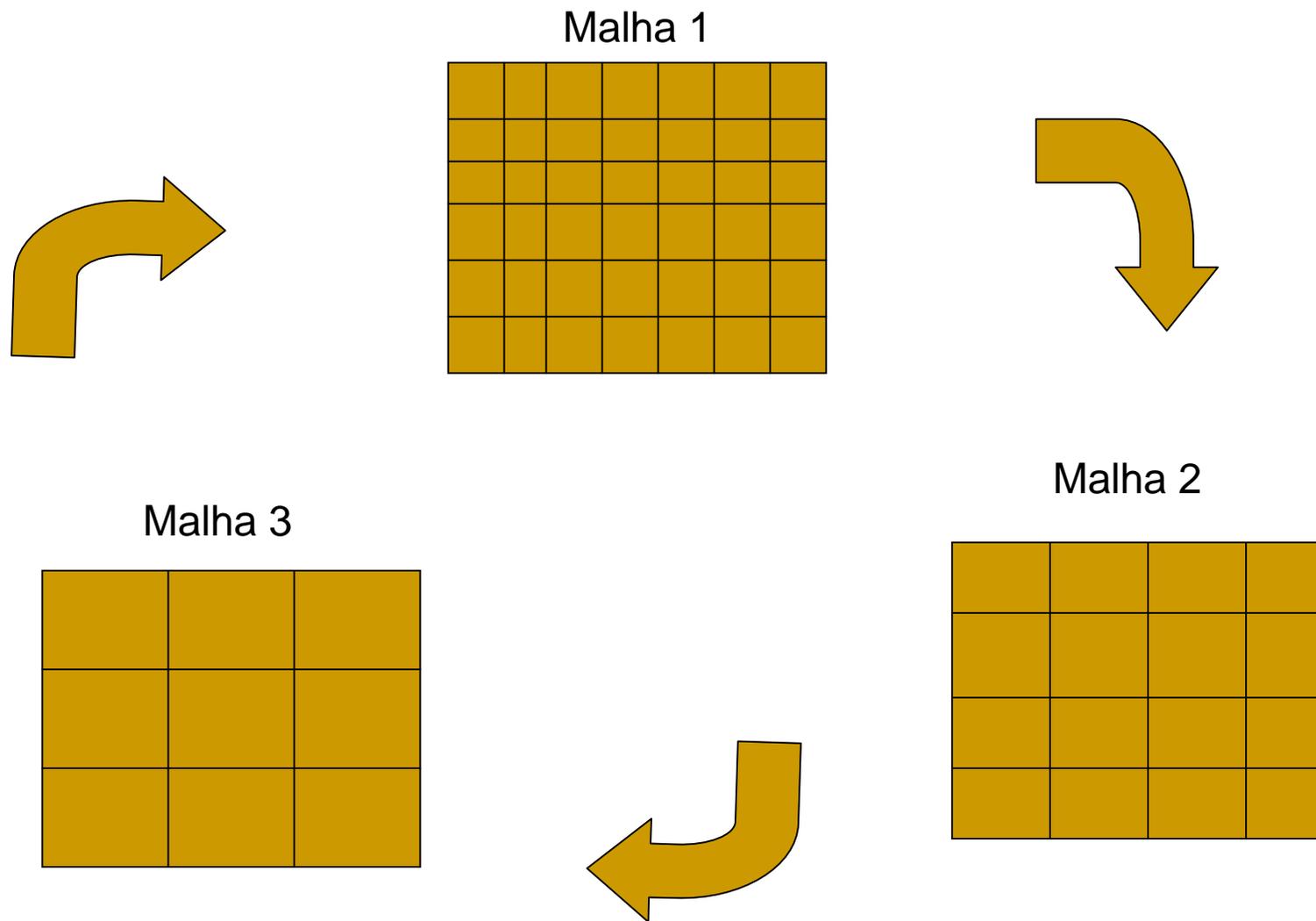
$$L \Delta u_{i,j} = -R_{i,j}$$

- Esta última equação é a chave mestra do método Multigrid, desde que ela é usada na malha grossa.
- Uma vez achado Δu , então calcula-se a solução, como:

$$u_{i,j} = u_{i,j}^n + \Delta u_{i,j}$$

Método Multigrid

Problema Modelo 2D

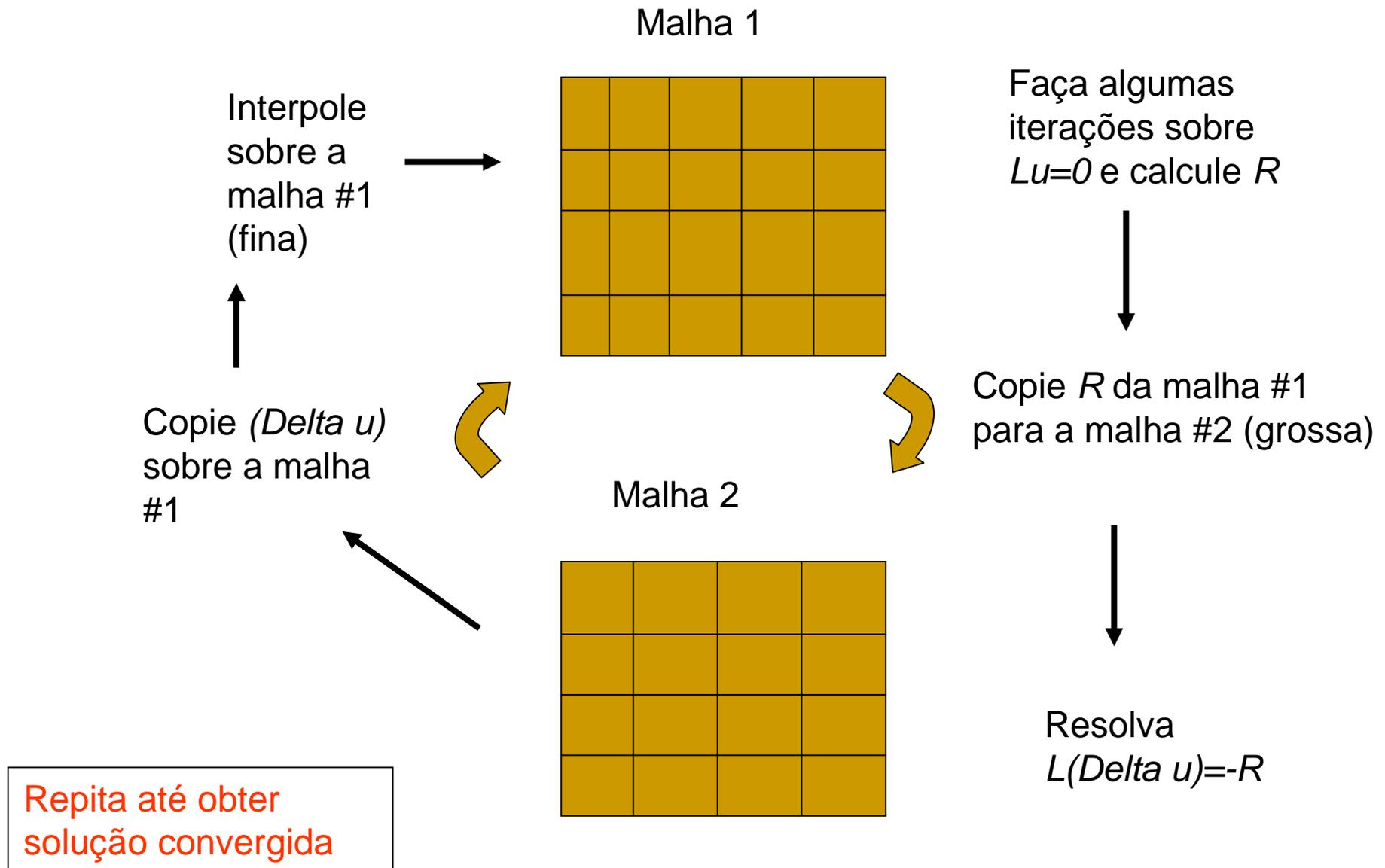


Passos de Implementação para 2 malhas:

- 
1. Faça algumas iterações na malha #1 usando *Gauss-Seidel*
 2. Calcule o resíduo R em cada ponto (i,j) sobre a malha #1 e então onde a malha #2 e malha #1 coincidem, copie o resíduo para a malha #2 (*Restrição*), caso contrário use FW
 3. Sobre a malha #2, resolva a equação de correção:
$$L\Delta u_{i,j} = -R_{i,j}$$
 4. Copie a correção (*Delta u*) na malha #1 (Malha fina), nos pontos que situam entre a malha #2 (grossa), utilize *Prolongamento (Interpolação)*. Simples média é suficiente
 5. Repete

Método Multigrid

Problema Modelo 2D



Equação Não-Linear:
$$\frac{\partial u^2}{\partial x} + \frac{\partial^2}{\partial x^2} u = 0$$

Aplicando diferença finita centrada de segunda ordem na derivas contínuas, obtém-se:

$$\frac{u_{i+1}^2 - u_{i-1}^2}{2\Delta x} + \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2} = 0$$

- Produz um sistema *não-linear* de equações algébricas...
- Para obter um sistema algébrico linear da E.D.P. acima fazemos uso da “*Forma Delta*” da seguinte maneira:

$\Delta u_i = u_i^{n+1} - u_i^n$, então podemos escrever :

$$u_i^{n+1} = u_i^n + \Delta u_i$$

Substituindo esta relação na equação de diferença finita anterior, obtém-se:

$$\frac{(u_{i+1}^n + \Delta u_{i+1})^2 - (u_{i-1}^n + \Delta u_{i-1})^2}{2\Delta x} +$$
$$\frac{(u_{i+1}^n + \Delta u_{i+1}) - 2(u_i^n + \Delta u_i) + (u_{i-1}^n + \Delta u_{i-1})}{\Delta x^2} = 0$$

Esta equação pode ainda ser re-escrita como:

$$\frac{2u_{i+1}^n \Delta u_{i+1} - 2u_{i-1}^n \Delta u_{i-1}}{2\Delta x} + \frac{\Delta u_{i+1} - 2\Delta u_i + \Delta u_{i-1}}{\Delta x^2} +$$

$$\frac{(\Delta u_{i+1})^2 - (\Delta u_{i-1})^2}{2\Delta x} =$$

Assumir negligenciável

$$- \left[\frac{(u_{i+1}^n)^2 - (u_{i-1}^n)^2}{2\Delta x} + \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} \right] = 0$$

Finalmente,

$$\frac{2u_{i+1}^n \Delta u_{i+1} - 2u_{i-1}^n \Delta u_{i-1}}{2\Delta x} + \frac{\Delta u_{i+1} - 2\Delta u_i + \Delta u_{i-1}}{\Delta x^2} =$$
$$- \left[\frac{(u_{i+1}^n)^2 - (u_{i-1}^n)^2}{2\Delta x} + \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2} \right]$$

Em forma linear matricial compacta: $A(u_i^n) \cdot \Delta u_i = RHS_i^n$

$$LHS_i^n = A(u_i^n) \quad \text{Então:} \quad LHS_i^n \cdot \Delta u_i^n = RHS_i^n$$

Esta forma é muito usada em CFD e é chamada de “Full Approximation Storage Multigrid Method”. É necessário armazenar, *Delta u*, *u*, resíduo *RHS* e *LHS* em cada nível de iteração para cada nível de malha.

- Problemas Modelo
- Método de Relaxação
- Erro de convergência
- Malhas (Grids) Múltiplas - Multigrid
- **Desempenho**
- Considerações Teóricas

- Requerimentos de armazenamento
 - Vetores \mathbf{v} e \mathbf{f} para $n = 16$ com valores de fronteira ($n=0$, $n=16$)

\mathbf{v}	<table border="1"><tr><td>M^h</td><td>M^{2h}</td><td>M^{4h}</td><td>M^{8h}</td></tr><tr><td>17</td><td>9</td><td>5</td><td>3</td></tr></table>	M^h	M^{2h}	M^{4h}	M^{8h}	17	9	5	3
M^h	M^{2h}	M^{4h}	M^{8h}						
17	9	5	3						
\mathbf{f}	<table border="1"><tr><td>M^h</td><td>M^{2h}</td><td>M^{4h}</td><td>M^{8h}</td></tr><tr><td>17</td><td>9</td><td>5</td><td>3</td></tr></table>	M^h	M^{2h}	M^{4h}	M^{8h}	17	9	5	3
M^h	M^{2h}	M^{4h}	M^{8h}						
17	9	5	3						

$$\text{EspaçoMemoria} < \frac{2n^d}{1-2^{-d}}, \quad \text{com } d : \text{Dimensão do problema}$$

- Para $d = 1$, requerimento de memória é menor que o dobro do problema de malha fina sozinho

Obs: $EM=64 < 2 \times (17+17)=68$

■ **Custo Computacional**

- 1 unidade de trabalho (UT): uma varredura de relaxação sobre M^h
- $O(UT) = O(N)$, com N: Número Total de pontos de malhas
- *Transferência entre malha é negligenciada*
- Uma varredura de relaxação por nível ($w_i = 1$)

$$Custo_{V-Ciclo} < \frac{2}{1 - 2^{-d}} UT$$

$$Custo_{FMG-Ciclo} < \frac{2}{(1 - 2^{-d})^2} UT$$

- Problema 1D : Simple V-Ciclo custa $\sim 4UT$,
Completo FMG ciclo $\sim 8UT$.

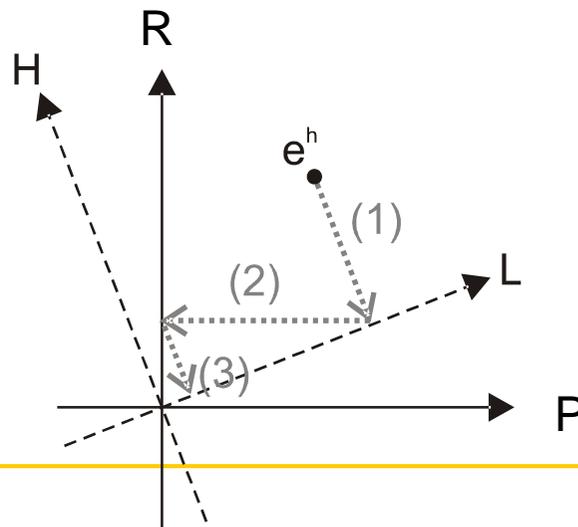
- ***Ferramentas de Diagnostico***
 - Ajuda a “debugar” seu código
 - *Plano Metódico para testar* módulos
 - *Iniciar Simples* com pequenos problemas!
 - *Expor Problema* – dificuldades podem estar escondidas
 - *Propriedade de Ponto Fixo* – relaxação pode não mudar a solução (*Problemas stiff*)
 - *Problema Homogêneo*: norma do erro e resíduo devem decrescer para zero

- Problemas Modelo
- Método de Relaxação
- Erro de convergência
- Malhas (Grid) Múltiplas- Multigrid
- Desempenho
- **Considerações Teóricas**

- O Impacto da transferência *Intermalha* e o método *iterativo* pode ser expressado e provado de maneira formal (*análise de complexidade de algoritmo*).
- *Quadro Espectral do método multigrid*
 - Relaxação amortece *modos oscilatórios*
 - Interpolação & Restrição *amortece modos suave*
- *Quadro Algébrico do método multigrid*
 - Decompõe o espaço do erro: $M^h = L \oplus H$
 - L é similar a P, e H é similar a R

$$P = P(I_{2h}^h A^{2h}) \quad R = R(I_h^{2h} A^h)$$

- Operações de método de ***multigrid***, visualizada
 - Plano representando: M^h
 - Erro e^h é sucessivamente projetado sobre um dos eixos
 - Relaxação sobre a malha fina (1)
 - Correção Duas-Malhas (2)
 - Novamente, relaxação sobre a malha fina (3)



1. A. Brandt. "Multi-level adaptive solutions to boundary-value problems." *Math. Comp.*, **31**, 333-390, (1977)
2. W. L. Briggs et al. *A Multigrid Tutorial*, Second Edition. SIAM. 2000
3. T. L. Beck *Rev. Mod. Phys.* 72 1041. 2000.
4. Anderson, Tannehill, & Pletcher, 1984.
5. W. Hackbusch and U. Trottenberg. *Lecture Notes in Mathematics*, **960**, (*Multigrid Methods*), (1982)
6. A. Brandt. "Guide to multigrid development.", *ibid.*, 220-312, (1982)
7. W. Hackbusch. "Multi-grid convergence theory.", *ibid.*, 177-219, (1982)
8. S. F. McCormick. *Frontiers in Applied Mathematics*, **3**, (*Multigrid Methods*), (1987)
9. P. Wesseling. *An Introduction to Multigrid Methods*, (1992)
10. U. Rude. "Fully adaptive multigrid methods." *SIAM J. Numer. Anal.*, **30**, 230-248, (1993)

- Pense nisso...?