Capítulo 12

Operações matriciais no MEF

Uma das maneiras de ver o MEF é considera-lo como uma transformação que converte um problema contínuo, dado em forma de um conjunto de equações diferenciais e devidas condições de contorno/iniciais, num problema discreto representado por um conjunto de equações algébricas em termos de um conjunto de coeficientes incógnitos (geralmente nodais). Num problema de mecânica dos sólidos estático, esse problema tem a forma padrão $\mathbf{KU} = \mathbf{F}$, onde $\mathbf{U}$ é o vetor que contém as incógnitas nodais do problema. Uma vez o sistema tenha sido resolvido para $\mathbf{U}$, todo o resto da solução é obtido de forma direta (deformações, reações, tensões, etc.). Em muitas situações em que o sistema algébrico bastante pequeno (envolvendo, por exemplo, 2 equações, 50 ou mesmo 1000 equações simultâneas), pode-se praticamente ignorar qualquer dificuldade em resolver o sistema. Por exemplo, utilizando uma rotina padrão de algum software para inverter $\mathbf{K}$ e em seguida calcular $\mathbf{U} = \mathbf{K}^{-1}\mathbf{F}$. Entretanto, quando se trata do MEF, simplesmente não é admissível esse procedimento devido a diversos fatores que tornariam, não raro, impossível a tarefa. Os fatores são, principalmente, os seguintes:

1. Os métodos de inversão de matrizes, alguns dos quais são sumarizados nesse capítulo, requerem uma quantidade de operações da ordem de N^3. Normalmente isso é representado por $Nop = O(N^3)$. Isso significa que uma matriz de $N = 1.000$, deve necessitar *cerca* de 10^9 operações de ponto flutuante (soma, subtração, divisão e multiplicação) para ser invertida. Entretanto, se $N = 10^6$, o número de operações para a inversão passa a ser $Nop = O(10^{18})$. Note-se que essa potência, 3, no número de operações é bastante perversa. Não importa o quanto a velocidade dos computadores cresça, a cada instante é possível identificar o tamanho de uma certa matriz cuja inversão demandaria um tempo de processamento proibitivamente longo. Um método que permita obter uma solução com um número de operações $O(N^2)$, por exemplo, será sempre mais rápido que um método com $Nop = O(N^3)$. Esse tipo de estimativa é o que torna proibido o uso de inversão como forma de solução no MEF.

2. Consequentemente, o único procedimento admissível consiste na solução do problema sem a inversão, utilizando algum dos métodos adequados, como: fatoração Gaussiana, fatoração de Cholesky, métodos de relaxação, métodos iterativos básicos e métodos dos gradientes conjugados, que são revisados nesse capítulo. Entretanto, o número de operações para a solução, utilizando a fatoração é cN^3, onde c é uma constante que não depende de N. É sabido que $Nop = O(N^3/3)$. Essa quantidade de operações, embora uma ordem abaixo daquela necessária para a inversão completa, ainda é inviável nas análises de EF. Nesse ponto deve-se enfatizar que a dificuldade central na aplicabilidade do MEF é exatamente o custo computacional na etapa de solução do problema algébrico. Tanto programas montados para testes de métodos e algoritmos, em atividades de pesquisa, quanto grandes programas comerciais, tem nesse ponto, a solução do sistema, seu ponto nevrálgico, sua etapa mais demorada. Todas as demais etapas do processamento (leitura de dados, geração de dados, cálculo das matrizes elementares, sobreposição, aplicação das condições de contorno, pós-processamento), ocupam menos tempo que a solução do sistema, exceto em problema muito pequenos em que N esteja abaixo de certo limite. Existe ainda mais um aspecto. O número de operações nessas

outras etapas são proporcionais à quantidade de nós ou de elementos, isto é, $O(Nnos)$ ou $O(nelem)$, isto é, se $Nnos$ ou $nelem$ for duplicado, a tendência é uma duplicação do trabalho computacional nessas etapas. Mas se $Nnos$ for duplicado, o esforço para resolver o sistema num algoritmo de $Nop = O(N^3)$ é aumentado em 8 vezes.

3. Um segundo aspecto a ser considerado é a **forma de armazenamento** dos termos da matriz. É sabido que as matrizes geradas no MEF são esparsas. Consequentemente, existem diversas formas tradicionais de fazer o armazenamento desses termos. Entretanto, a única forma "proibida" (exceto em pequenos problemas educacionais e de teste) é a forma de armazenamento como **matriz quadrada**, isto é, o armazenamento de todos os $N \times N$ termos numa área de memória dimensionada como $K(N, N)$. Essa forma de armazenamento apresenta dois inconvenientes : (1) ocupa uma área excessiva na memória do computador. (2) o aspecto mais grave é que, como enunciado no item 2 acima, o número de operações na fatoração é $O(N^3)$. Ocorre que o número de operações é grandemente reduzido quando se buscam formas de armazenamento que não armazene todos os termos nulos, e quando o método de solução também toma partido disso e não realiza operações sobre os termos nulos. Dependendo da forma de armazenamento e de solução utilizada, a quantidade de operações pode cair para $O(N^2)$, como na combinação tradicional (que de fato é bastante simples), de armazenamento em meia banda e método de Gauss adaptado a essa forma de armazenamento.

4. Deve-se notar que a relação entre o número de operações e o tempo de processamento só é direta em processamento sequencial. Em **processamento paralelo** a relação é distinta, embora nem sempre previsível, devido ao maior volume de operações de gerenciamento de informações.

O presente capítulo identifica as principais formas de armazenamento de matrizes, e em seguida faz um levantamento ligeiro dos principais métodos de solução existentes e utilizados com comparações entre os números de operações necessárias.

12.1 Tipos de armazenamento de matrizes

12.1.1 Matriz triangular

O termo matriz triangular, à primeira vista, aparenta estranho, mas se refere apenas a uma formatação dos termos que se deseja armazenar. Essa forma é a mais obvia quando se considera a simetria da matriz de rigidez ou de inércia do MEF na maioria das aplicações. Assim, é bastante natural que se armazene apenas os termos de um dos lados da diagonal principal. Por exemplo, consideremos uma matriz simétrica $\mathbf{A}$ que desejemos armazenar apenas seus termos acima da diagonal. Esses termos são visualizados da seguinte forma

$$\mathbf{A} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & \cdots \\ & A_{22} & A_{23} & \cdots \\ & & A_{33} & \cdots \\ \text{sim.} & & & \ddots \end{bmatrix}_{N \times N} . \tag{12.1}$$

A forma de armazenamento denominada **matriz triangular superior** consiste, de fato, em armazenar os termos acima da diagonal principal em uma área de memória dimensionada como arranjo unidimensional, que denominaremos no resto do texto como *vetor*. O vetor armazena coluna após coluna dos termos de $\mathbf{A}$ que estejam acima da diagonal. No exemplo o vetor tem a seguinte

forma:

$$\mathbf{V} = \left\{ \begin{array}{c} A_{11} \\ A_{12} \\ A_{22} \\ A_{13} \\ A_{23} \\ A_{33} \\ \vdots \\ A_{NN} \end{array} \right\}_{M\times 1} . \tag{12.2}$$

A quantidade total de termos no vetor é obtido pela fórmula (facilmente deduzível):

$$M = N(N+1)/2. \tag{12.3}$$

Numa rotina de cálculo, um termo arbitrário A_{ij} de $\mathbf{A}$ pode ser localizado na posição m de $\mathbf{V}$, isto é:

$$\boxed{\mathbf{V}(m), \qquad \text{onde} \quad m = (j-1)j/2 + i} \tag{12.4}$$

Note que a as operações $(j-1)j/2$ devem ser entendidas em forma computacional, onde as variáveis i, j e m são inteiros e devem ser operados com os truncamento, isto é, $11/2 = 5$, e não 5,5. Adicionalmente, as operações devem ser feitas na ordem mostrada. Por exemplo, o termo A_{23} está na posição $m = (j-1)j/2+i = (3-1)3/2+2 = 5$, isto é, é o termo $\mathbf{V}(5)$, como pode ser verificado em (12.2).

Nota-se que, com rearranjos adequados na definição acima, pode-se construir também uma matriz triangular inferior.

A programação para realizar operações referentes à matriz $\mathbf{A}$, operando em seus termos armazenados em $\mathbf{V}$, é feita utilizando a fórmula de mapeamento indicial (12.4). Por exemplo, o **produto** com um vetor arbitrário $\mathbf{U}$, isto é, $\mathbf{W} = \mathbf{AU}$, é obtido pelo seguinte fragmento em Fortran:

```
DO i = 1,N                      ! corre termos de W
   W(i) = 0.0d0
   DO j = 1,i                   ! corre colunas de A na linha i
      IF(j.le.i) mij = mm(j,i)
      IF(j.gt.i) mij = mm(ij)
      W(i) = W(i) + V(mij) * U(j)
    ENDDO
ENDDO
```
(12.5)

onde `mm(i,j)` é um subprograma função que calcula (12.4).

12.1.2 Matriz banda

A forma de armazenamento em matriz triangular representa uma melhoria substancial em relação à matriz quadrada, entretanto não leva em conta a característica de que as matrizes do MEF são, dentro de certas condições, "bandeadas", isto é, dependendo da forma de numeração nodal, a matriz tem todos os seus termos não nulos agrupados em torno da diagonal principal, na seguinte forma

o número de dimensões):

$$\boxed{N = O\left(\frac{1}{h^d}\right) \quad \text{e} \quad b = O\left(\frac{1}{h^{d-1}}\right)} \tag{12.15}$$

12.1.3 Matriz skyline

Uma forma mais sofisticada de armazenar termos não nulos é a chamada matriz skyline. Em vez de armazenar todos os termos abaixo da linha da banda, o que é feito é armazenar, de cada coluna, apenas os termos abaixo do termo não nulo situado à maior distância vertical da diagonal. Por exemplo, consideremos a matriz com a seguinte forma

$$\mathbf{A} = \begin{bmatrix} A_{11} & & & & \\ & A_{22} & A_{23} & & A_{25} \\ & & A_{33} & & 0 \\ & & & A_{44} & A_{45} \\ \text{sim.} & & & & A_{55} \end{bmatrix}_{5\times 5}, \tag{12.16}$$

onde os termos acima da diagonal não indicados são nulos. A meia-banda dessa matriz é $b = 4$, de forma que não se conseguiria uma economia suficiente de memória nem de tempo de processamento fazendo o armazenamento em banda. Já o armazenamento skyline consiste em trabalhar com os dados, num vetor único, armazenando coluna após coluna, a partir do primeiro termo da coluna até a diagonal. Para o exemplo se tem o seguinte vetor coluna de armazenamento:

$$\mathbf{V} = \left\{ \begin{array}{ccccccccc} A_{11} & A_{22} & A_{23} & A_{33} & A_{44} & A_{25} & 0 & A_{45} & A_{55} \end{array} \right\}^T_{N_A\times 1}. \tag{12.17}$$

Nota-se que na coluna 5 armazenamos todos os termos a partir do primeiro não nulo da coluna, A_{25}, inclusive o 0 da posição A_{35}. No exemplo, temos o armazenamento de apenas $N_A = 9$ termos em $\mathbf{V}$. Para armazenamento em banda teríamos $5 \times 4 = 20$ termos, para matriz triangular $M = N(N+1)/2 = 15$ e em matriz quadrada $N \times N = 5 \times 5 = 25$. Conforme a ordem da matriz cresce, a diferença cresce de forma potencial. Nota-se que a matriz banda só apresenta vantagem se a numeração nodal for adequada para gerar uma banda estreita, isto é, $b << N$. Do contrário ela se comporta de forma pior que o armazenamento triangular.

O armazenamento em matriz skyline trabalha com dois vetores. O vetor com os dados da matriz, como o $\mathbf{V}$ em (12.17), e um **vetor de controle** Max$A(N)$ que indica, para cada coluna j, a posição em $\mathbf{V}$ do termo A_{jj} da matriz:

$$\boxed{V(\text{Max}A(j)) = A_{jj}} \tag{12.18}$$

No exemplo, o vetor de controle é

$$\text{Max}A = \{1, 2, 4, 5, 9\}^T_{N\times 1}. \tag{12.19}$$

Então o termo A_{33} encontra-se em $V(\text{Max}A(3)) = V(4)$. O termo A_{22} encontra-se em $V(\text{Max}A(2)) = V(2)$. Com isso determina-se o número de termos na coluna 3 sob o perfil skyline: $nk = \text{Max}A(3) - \text{Max}A(2) = 4 - 2 = 2$. Em geral, pode-se calcular, unicamente a partir de Max$A$, a primeira linha não nula li na coluna k:

$$\boxed{\begin{array}{l} nk = \text{Max}A(k) - \text{Max}A(k-1), \\ li = k - nk + 1. \end{array}} \tag{12.20}$$

No exemplo, $li = k - nk + 1 = 3 - 2 + 1 = 2$. Então a coluna 3 inicia-se na linha 2 e termina na linha $k = 3$.

12.1.4 Matriz esparsa

Essa é a forma mais sofisticada de armazenamento, em que apenas os termos não nulos da matriz são armazenados num vetor de dados, e dois vetores de controle indicam a posição de cada termo da matriz no vetor de dados. Em suma, definem-se três vetores de dimensão Nz:

$$\begin{array}{l}\text{Vetor } \mathbf{V} \text{ em que } V(l) = \text{valor de algum termo } A_{ij} \text{ da matriz.}\\ \text{Vetor } Lin, \text{ em que } Lin(l) = \text{número da linha em } \mathbf{A}, \text{ isto é, } Lin(l) = i.\\ \text{Vetor } Kol, \text{ em que } Kol(l) = \text{número da linha em } \mathbf{A}, \text{ isto é, } Kol(l) = j.\end{array} \tag{12.21}$$

Para a matriz em (12.16), armazenam-se apenas os $Nz = 8$ termos não nulos, e os três vetores são:

$$\mathbf{V} = \left\{\begin{array}{c} A_{11}\\ A_{22}\\ A_{23}\\ A_{33}\\ A_{44}\\ A_{25}\\ A_{45}\\ A_{55}\end{array}\right\}_{Nz\times 1} , \quad Lin = \left\{\begin{array}{c} 1\\ 2\\ 2\\ 3\\ 4\\ 2\\ 4\\ 5\end{array}\right\}_{Nz\times 1} , \quad Kol = \left\{\begin{array}{c} 1\\ 2\\ 3\\ 3\\ 4\\ 5\\ 5\\ 5\end{array}\right\}_{Nz\times 1} . \tag{12.22}$$

Essa é a forma em que os termos da matriz são armazenados coluna após coluna. Existe a forma similar de armazenamento linha por linha. Na literatura essas formas de armazenamento são conhecidas como lista de coordenadas (COO - Coordinate List). Outras formas existem de armazenamento. Ver por exemplo, Golub [40].

12.2 Métodos de solução de sistemas algébricos estáticos

Existem duas grandes famílias de métodos para a solução de um sistema algébrico: os métodos diretos e os métodos iterativos. Os **métodos diretos** determinam a solução exata, a menos de erros de truncamento, **após um número fixo de operações**. Praticamente todos os programas de elementos finitos utilizam um método baseado na fatoração de Gauss, que será revisado nessa seção. Entretanto, pelo menos dois outros métodos diretos eficientes existem, o **método de Givens** e o de **Householder**, ambos baseados em matrizes ortogonais. Não faremos revisão desses métodos no presente texto. Adicionalmente, existem métodos baseados na fatoração de Gauss em conjunção com o procedimento de **condensação estática**. Os **métodos iterativos**, por sua vez, geram apenas soluções aproximadas, e o número de operações é dependente do número de iterações. Por sua vez, o número de iterações depende do número de *condição da matriz*, como será visto no texto. Assim, a solução de cada matriz por um método iterativo demandará mais ou menos tempo de processamento que outra matriz do mesmo tamanho. Além dos métodos baseados em Fatoração de Gauss, o método mais utilizado em grandes programas comerciais de MEF é o de **gradientes conjugados** com pré-condicionamento de Cholesky (ambos revisados nesse capítulo). Entretanto, historicamente, o método de Gauss-Seidel foi intensamente utilizado em MEF e apresenta diversas características interessantes.

12.2.1 Eliminação de Gauss

Os métodos baseados na fatoração de Gauss formam o fundamento para todos os **métodos diretos** mais eficientes. Dada uma matriz real $\mathbf{A}$, de dimensões $N \times N$, isto é, de ordem N, não singular, não necessariamente simétrica. Dado também um vetor $\mathbf{F}$ de ordem N, deseja-se determinar o vetor $\mathbf{U}$ de ordem N tal que

$$\mathbf{AU} = \mathbf{F}. \tag{12.23}$$

O processo padrão da eliminação de Gauss é feito em duas etapas.

Etapa 1 - Triangularização

$$\boxed{\begin{aligned} U_n &= \frac{F_N}{S_{NN}}, \\ U_i &= \frac{1}{S_{ii}}\left[F_i - \sum_{j=i+1}^{j_f} S_{ij}U_j\right], \quad \text{para } i = N-1, \cdots, 1 \end{aligned}} \tag{12.38}$$

O valor final no somatório, j_f, depende do tipo de esparsidade da matriz, se matriz cheia (quadrada) ou se tem banda b:

$$\begin{aligned} j_f &= N \text{ se matriz completa ou se tiver banda e } i+b > N, \\ j_f &= i+b \text{ se tem banda e } i+b \leq N. \end{aligned} \tag{12.39}$$

A etapa de triangularização pode ser resumida na forma mostrada no seguinte fluxograma, para uma matriz quadrada, não singular, não necessariamente simétrica. A matriz **A** é armazenada em forma quadrada, completa. Ao final do processamento, as fatorações **L** e **S** (tal que **A** = **LS**), aparecem na mesma área de memória que **A**. **L** contém diagonal unitária.

	Fatoração de Gauss:	
(1)	do $j = 1, N-1$	(corre as colunas de **A**)
(2)	do $l = j+1, l_f$	(corre as linhas sob a diagonal)
(3)	if $A_{jj} \geq \varepsilon$, $c = A_{lj}/A_{jj}$	
(4)	if $A_{jj} < \varepsilon$, msg de erro, stop	
(5)	do $m = j, m_f$	(corre as colunas da linha l)
(6)	$A_{lm} = A_{lm} - cA_{jm}$	(compõe **S** e parte de **L**)
	enddo	
(7)	$A_{lj} = c$	(compõe **L**)
	enddo	
	enddo	
(8)	Para matriz cheia,	$l_f = m_f = N$
(9)	Para matriz com banda b,	$l_f = \begin{cases} j+b, \text{ se } j+b \leq N, \\ N, \quad \text{se } j+b > N. \end{cases}$
		$m_f = \begin{cases} j+b, \text{ se } j+b \leq N, \\ N, \quad \text{se } j+b > N. \end{cases}$

(12.40)

12.2.2 Método de Cholesky

Caso a matriz **A** seja simétrica, sua decomposição de Gauss torna-se $\mathbf{A} = \mathbf{LDL}^T$. Se, adicionalmente, **A** for positiva-definida, essa decomposição pode ser colocada na forma

$$\mathbf{A} = \mathbf{CC}^T, \quad \text{onde} \quad \mathbf{C} \equiv \mathbf{LD}^{1/2}. \tag{12.41}$$

A condição de que **A** seja positiva-definida se revela, entre outras coisas, pelo fato de que todos os termos **D** são positivos. A fatoração $\mathbf{A} = \mathbf{CC}^T$ pode ser feita como pós processamento da fatoração de Gauss, porém o cálculo feito diretamente conforme o método desenvolvido por Cholesky revela-se aproximadamente **duas vezes mais rápida que a fatoração de Gauss**. O fluxograma a seguir é baseado em que a matriz **A** ocupa um arranjo quadrado de ordem N, e ao final essa mesma área de memória ocupa, em seu triângulo inferior, a matriz **C**. Todas as operações são feitas apenas sobre os termos do triângulo inferior de **A**. Ao final do fluxograma são apresentados ajustes nos limites

dos somatórios para limitar as operações aos termos dentro da banda da matriz, caso ela a possua.

	Fatoração de Cholesky:	
(1)	do $k = 1, N$	(corre as colunas de $\mathbf{A}$)
(2)	$A_{kk} = \sqrt{A_{kk}}$.	
(3)	do $i = k+1, i_f$	(linhas da coluna k sob a diagonal)
(4)	$A_{ik} = A_{ik}/A_{kk}$	
	enddo	
(5)	do $j = k+1, j_f$	(colunas a direita da k)
(6)	do $p = j, p_f$	(linhas da coluna j sob a diagonal j)
(7)	$A_{pj} = A_{pj} - A_{pk}A_{jk}$	
	enddo	
	enddo	
	enddo	
(8)	Para matriz cheia,	$i_f = p_f = j_f = N$
(9)	Para matriz com banda b,	$i_f = \begin{cases} N, & \text{se } q_i > N, \\ q_i, & \text{se } q_i \leq N. \end{cases}$
(10)		$p_f = \begin{cases} N, & \text{se } q_p > N, \\ q_p, & \text{se } q_p \leq N. \end{cases}$
(11)		$j_f = \begin{cases} N, & \text{se } q_j > N, \\ q_j, & \text{se } q_j \leq N. \end{cases}$
(12)		$q_p = q_i = q_j = k + b$

(12.42)

As operações de **substituição progressiva e regressiva** são as mesmas do método de Gauss, mostradas nas eqs. (12.35), (12.36) e (12.38), (12.39), apenas substituindo $\mathbf{L}$ e $\mathbf{S}$ por $\mathbf{C}$ e $\mathbf{C}^T$ respectivamente.

12.2.3 Contagem de operações no método de Gauss - matriz cheia

Do ponto de vista computacional, o aspecto mais importante de um método que gera solução exata (a menos de erros de truncamento), como os métodos diretos, é o tempo de processamento, que se traduz no número de operações necessárias para gerar a solução. Esse número de operações geralmente é contado usando o acrônimo inglês "flop" - *float point operation*, o número de operações de ponto flutuante. É a contagem do número de operações de soma, subtração, produto e divisão entre números reais. São as operações mais demoradas. A forma de contagem não é uniforme na literatura. Algumas vezes são consideradas apenas os produtos e divisões, sendo subentendido que os cálculos envolverão igual quantidade de soma-subtrações.

Fatoração

Consideremos uma matriz quadrada não simétrica $\mathbf{A}$, e as operações de fatoração de Gauss mostradas no fluxograma da eq. (12.40). Incialmente consideremos as operações para um dado valor de j, o laço mais externo. As operações são aquelas nos laços das linhas 2 e 5, de $l = j+1, N$ e $m = j, N$. Cada termo no laço interno envolve um produto e uma subtração. O número de termos é $(N-j)(N-j+1)$: é o número de termos no retângulo da matriz definido pelos limites dos laços definido por j. Esse retângulo é mostrado no seguinte exemplo, para $j = 3$:

12.2.4 Contagem de operações no método de Cholesky

O número de operações na fatoração de Cholesky é obtido de forma similar ao da de Gauss, a partir das linhas 5-7 do fluxograma da eq. (12.42):

$$n_{fC} = \sum_{k=1}^{N} \sum_{l=k+1}^{N} \left[1 \text{ produto} + \sum_{m=j}^{N} (1 \text{ soma} + 1 \text{ produto}) \right], \text{ isto é,}$$

$$n_{fC} = \frac{1}{3} N(N+1)(N-1) \qquad \rightarrow \qquad \boxed{n_{fC} \approx \frac{1}{3} N^3} \tag{12.56}$$

As **substituições progressivas e regressivas** demandam a mesma quantidade de operações que no método de Gauss, eqs. (12.50).

Inversão pelo método de Cholesky

Consideramos o procedimento para obter a inversa a partir da representação fatorada de Cholesky para $\mathbf{A}$, isto é, $\mathbf{A} = \mathbf{C}\mathbf{C}^T$. Então

$$\begin{aligned} \mathbf{A}\mathbf{A}^{-1} &= \mathbf{I} \\ \mathbf{C}\mathbf{C}^T\mathbf{A}^{-1} &= \mathbf{I} \end{aligned} \qquad \rightarrow \qquad \boxed{\mathbf{A}^{-1} = \mathbf{C}^{-T}\mathbf{C}^{-1}} \tag{12.57}$$

Ambas as matrizes $\mathbf{C}^{-T}$ e $\mathbf{C}^{-1}$ são triangulares (superior e inferior respectivamente). As operações para determinar o triângulo superior de $\mathbf{A}^{-1}$ são práticamente as mesmas mostradas em (12.53) e o número de operações, $n_{IC\,\mathrm{sup}}$, é o mesmo mostrado em (12.54).

Devido à simetria da inversa, não há necessidade de fazer operações para seu triângulo inferior. Então, o custo total de determinar a inversa é dado pelas operações de fatoração e do produto matricial:

$$\begin{aligned} n_{IG} &= n_{fC} + n_{IC\,\mathrm{sup}}, \\ &= \frac{1}{3} N^3 + \frac{1}{3} N(N+1)(N+2) \qquad \boxed{n_{IG} \approx \frac{2}{3} N^3} \end{aligned} \tag{12.58}$$

Essa é a metade no número de operações para fatoração de Gauss, eq. (12.55).

12.2.5 Matriz banda

Consideremos por exemplo a seguinte matriz quadrada

$$\mathbf{A} = \begin{bmatrix} A_{11} & \mathrm{x} & \mathrm{x} & \mathrm{x} & & & & & & & \\ \boxed{\begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix}} & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{d} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} & & & & & & \\ & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ A_{44} \end{matrix} & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{x} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{x} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} & & & \\ & & & & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{d} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} & & & \\ & & & & & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{d} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} & \\ & & & & & & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{d} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} \\ & & & & & & & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{d} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} \\ & & & & & & & & & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{d} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{x} \end{matrix} \\ & & & & & & & & & & & A_{88} & \mathrm{x} & \mathrm{x} & \mathrm{x} \\ & & & & & & & & & & & \boxed{\begin{matrix} \mathrm{y} \\ \mathrm{y} \\ \mathrm{y} \end{matrix}} & \begin{matrix} \mathrm{d} \\ \mathrm{y} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{d} \\ \mathrm{y} \end{matrix} & \begin{matrix} \mathrm{x} \\ \mathrm{x} \\ \mathrm{d} \end{matrix} \\ & & & & & & & & & & & & & & \mathrm{x} \\ & & & & & & & & & & & & & & \mathrm{x} \end{bmatrix} \tag{12.59}$$

com banda, $b = 3$. A contagem de operações pode ser feita usando os mesmos somatórios usados no caso de matriz cheia, apenas ajustando alguns limites dos somatórios para eliminar operações com os termos nulos fora da banda.

matrizes, $\mathbf{A}^{-1}$, $\mathbf{S}^{-1}$ e $\mathbf{L}^{-1}$, possuem banda b, sendo que as duas últimas são banda superior e inferior, respectivamente. Os termos no triângulo superior de $\mathbf{A}^{-1}$, até a linha $n-b$, são obtidos por:

$$\begin{array}{l} \text{Do } i = 1, N-b \qquad\qquad (\text{linha de } \mathbf{A}^{-1}) \\ \quad \text{Do } j = i, i+b \qquad (\text{coluna de } \mathbf{A}^{-1}) \\ \quad\quad \text{Do } m = j, i+b \\ \quad\quad\quad \left(\mathbf{A}^{-1}\right)_{ij} = \left(\mathbf{A}^{-1}\right)_{ij} + \left(\mathbf{S}^{-1}\right)_{im} \left(\mathbf{L}^{-1}\right)_{mj} \end{array} \tag{12.65}$$

e os termos após a linha a linha $n-b+1$ (o triângulo superior definido pelo quadrado iniciado pelo termo diagonal $n-b+1$) são

$$\begin{array}{l} \text{Do } i = N-b+1, N \qquad\qquad (\text{linha de } \mathbf{A}^{-1}) \\ \quad \text{Do } j = i, N \qquad (\text{coluna de } \mathbf{A}^{-1}) \\ \quad\quad \text{Do } m = j, N \\ \quad\quad\quad \left(\mathbf{A}^{-1}\right)_{ij} = \left(\mathbf{A}^{-1}\right)_{ij} + \left(\mathbf{S}^{-1}\right)_{im} \left(\mathbf{L}^{-1}\right)_{mj} \end{array} \tag{12.66}$$

A quantidade principal de operações é contada do fragmento de programa (12.65), para 1 soma e 1 produto. O resultado para o triângulo superior da inversa, usando banda, é

$$n_{IGb\,\text{sup}} = (N-b-1)(b+2)(b+3) \approx Nb^2. \tag{12.67}$$

Caso a matriz não seja simétrica, torna-se necessário realizar aproximadamente a mesma quantidade de operações para identificar o triângulo inferior da inversa. Então, o custo total de determinar a inversa é dado pelas operações de fatoração, (12.60), e do produto matricial $\mathbf{S}^{-1}\mathbf{L}^{-1}$:

$$\begin{aligned} n_{IGb} &= n_{fGb} + 2n_{IGb\,\text{sup}}, \\ &= 2Nb^2 + 2Nb^2 \qquad\qquad \boxed{n_{IG} \approx 4Nb^2} \end{aligned} \tag{12.68}$$

Logo é um custo fortemente menor que a inversa obtida pelo procedimento mostrado em (12.64).

Fatoração de Cholesky, banda

Consideramos as linhas 2-5 do fluxograma da eq. (12.42), com os limites j_f e p_f ajustados conforme as linhas 9-11. O efeito dessas limitações é que, para cada termo k, as operações realizadas são limitadas ao triângulo inferior definido pelo termo A_{jj}. No exemplo da matriz (12.59) estão marcados os retângulos associados aos termos A_{11} e A_{88}. Note que essa é a mesma argumentação usada no caso da fatoração de Gauss em banda, exceto que lá as operações eram feitas sobre todo o retângulo (o que levou à estimativa (12.59)), enquanto aqui é apenas sobre o triângulo. Por exemplo, para $k=1$, A_{11}, o triângulo começa em A_{22} e tem lados 3×3 se $b=3$. Cada triângulo possui $b(b+1)/2$ termos, sendo que em cada termo são feitas uma subtração e um produto. Somamos apenas os retângulos de A_{11} a $A_{(N-b)(N-b)}$, isto é, tomamos $(N-b)$ retângulos. Em resumo, o número de flop's é aproximadamente

$$n_{fCb} = 2\frac{1}{2}b(b+1)(N-b), \qquad \rightarrow \qquad \boxed{n_{fCb} \approx Nb^2} \tag{12.69}$$

Uma outra forma pode ser obtida tomando todos os laços do fluxograma (12.42):

$$\begin{aligned} n_{fCb} &= \sum_{k=1}^{N-b} \sum_{j=k+1}^{k+b} \left[1 \text{ produto} + \sum_{i=j}^{k+b} (1 \text{ soma} + 1 \text{ produto}) \right], \\ &= b(b+2)(N-b), \end{aligned} \tag{12.70}$$

cuja aproximação assintótica é a mesma de (12.59). Note que essas estimativas ignoram o número

Tabela 12.1: Sumário das estimativas de números assintóticos de operações para operações típicas usando fatoração de Gauss e de Cholesky. As colunas 2D e 3D são para o problema padrão.

Método	Operação	Esparsidade	O(flop's)	2D	3D
Gauss	Fatoração **L, S**	completa	$\frac{2}{3}N^3$		
	Subst. progr. + regr.		$2N^2$		
	Inversão		$\frac{4}{3}N^3$		
Cholesky	Fatoração **B**	completa	$\frac{1}{3}N^3$		
	Subst. progr. + regr.		$2N^2$		
	Inversão		$\frac{2}{3}N^3$		
Gauss	Fatoração **L, S**	Banda b	$2Nb^2$	$2N^2$	$2N^{7/3}$
	Subst. progr. + regr.		$4Nb$	$4N^{3/2}$	$4N^{5/3}$
	Inversão		$4Nb^2$	$4N^2$	$4N^{7/3}$
Cholesky	Fatoração **B**	Banda b	Nb^2	N^2	$N^{7/3}$
	Subst. progr. + regr.		$4Nb$	$4N^{3/2}$	$4N^{5/3}$
	Inversão		$2Nb^2$	$2N^2$	$2N^{7/3}$

$$\begin{aligned} \text{Em } 2D, \quad N &= O\left(\frac{1}{h^2}\right), \text{ logo } b = O\left(\frac{1}{h}\right) = O\left(N^{1/2}\right), \quad \text{e} \\ \text{Em } 3D, \quad N &= O\left(\frac{1}{h^3}\right), \text{ logo } b = O\left(\frac{1}{h^2}\right) = O\left(N^{2/3}\right). \end{aligned} \tag{12.74}$$

Essas estimativas para b podem ser aplicadas nas estimativas da coluna 4 da Tabela 12.1 para estimar o número de operações na malha padrão, em matriz banda, mostrados nas colunas 5 e 6. A Figura 12.2 mostra uma comparação entre as estimativas assintóticas do número de operações para fatoração usando os métodos de Gauss (matriz não simétrica) e de Cholesky, com matriz completa e banda. Para o caso de matriz banda utilizou-se as estimativas para a malha padrão em 2D e 3D mostradas na Tabela 12.1. Nota-se que a diferença de 2 entre o número de operações de Gauss e de Cholesky parecem pouco nítida no gráfico, devido às escalas logarítmicas. Já as diferenças de inclinação das retas são evidentes. O trabalho com matriz completa parece inadmissível quando comparado ao uso da banda no processo de fatoração. De fato, a tendência é que outros métodos de armazenamento sejam usados, além da banda, (como skyline, esparsa), de forma a obter ainda mais vantagens.

12.3 Método iterativos baseados em minimização de potencial

Os métodos iterativos se caracterizam por não serem capazes de fornecer a solução $\mathbf{U}$ do problema $\mathbf{AU} = \mathbf{F}$, mas apenas uma aproximação dela, exceto em situações muito particulares, como quando o sistema é muito pequeno ou quando a estimativa inicial é proporcional à solução exata.

Diversos métodos, como o do Gradiente, são baseados na minimização de um potencial. Caso a matriz $\mathbf{A}$ seja a de rigidez e $\mathbf{F}$ um vetor força nodal, obtidos pelo MEF em um problema elastoestático, o potencial é a aproximação da energia potencial total do sistema. Entretanto, qualquer que seja a origem e a interpretação física de $\mathbf{A}$, se ela for simétrica e positiva definida, **prova-se** que o vetor $\mathbf{U} \in R^N$ que minimiza o potencial

$$\begin{aligned} V(\mathbf{U}) &= \tfrac{1}{2}\mathbf{U}\cdot\mathbf{AU} - \mathbf{U}\cdot\mathbf{F} \quad \text{em notação vetorial, ou} \\ &= \tfrac{1}{2}\mathbf{U}^T\mathbf{AU} - \mathbf{U}^T\mathbf{F} \quad \text{em notação matricial.} \end{aligned} \tag{12.75}$$

também resolve o sistema

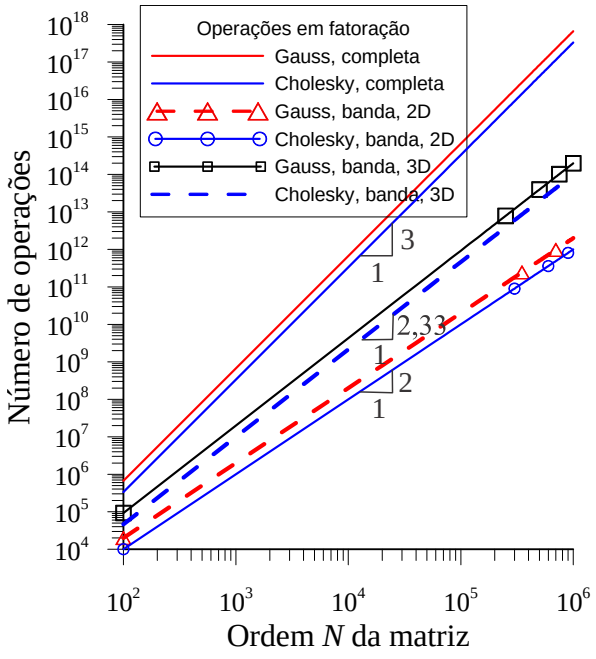

Figura 12.2: Comparação entre estimativas assintóticas de número de operações para fatoração de Gauss e de Cholesky, com matriz completa e banda.

$$\mathbf{AU} = \mathbf{F}. \tag{12.76}$$

Assim, a solução do problema algébrico pode ser obtida buscando o vetor que minimiza o potencial. Uma prova sucinta da equivalencia de ambas as soluções pode ser feita como segue. Considera-se o sistema em **dois estados**:

Estado 1, caracterizado pelo vetor deslocamento[2] $\mathbf{U}$ e o potencial $V(\mathbf{U})$;

Estado 2, caracterizado por um vetor vizinho ao estado 1, dado por $\mathbf{U}+\delta\mathbf{U}$ e potencial $V(\mathbf{U}+\delta\mathbf{U})$.

$\delta\mathbf{U}$ é um deslocamento virtual (variação de $\mathbf{U}$) aplicado ao estado 1. A variação total do potencial é obtida usando (12.75)

$$\begin{aligned} \Delta V &\equiv V(\mathbf{U}+\delta\mathbf{U}) - V(\mathbf{U}), \\ &= \frac{1}{2}(\mathbf{U}+\delta\mathbf{U})^T\,\mathbf{A}\,(\mathbf{U}+\boldsymbol{\delta}\mathbf{U}) - (\mathbf{U}+\boldsymbol{\delta}\mathbf{U})^T\mathbf{F} - \frac{1}{2}\mathbf{U}^T\mathbf{A}\mathbf{U} + \mathbf{U}^T\mathbf{F}. \end{aligned} \tag{12.77}$$

Uma vez que os produtos triplos resultam em um escalar, $\delta\mathbf{U}^T\mathbf{A}\mathbf{U} = \mathbf{U}^T\mathbf{A}\delta\mathbf{U}$. Então a variação simplifica-se para

$$\Delta V = \underbrace{\delta\mathbf{U}^T\,[\mathbf{AU}-\mathbf{F}]}_{\delta V} + \frac{1}{2}\delta\mathbf{U}^T\mathbf{A}\delta\mathbf{U}. \tag{12.78}$$

Se $\mathbf{U}$ for a solução de (12.76), então a primeira variação fica $\delta V = \delta\mathbf{U}^T\,[\mathbf{AU}-\mathbf{F}] = 0$, e a variação total do potencial fica

$$\Delta V = \frac{1}{2}\delta\mathbf{U}^T\mathbf{A}\delta\mathbf{U}.$$

[2] Utilizaremos aqui a notação força/deslocamento/rigidez como num problema de mecânica dos sólidos, apenas para ajuntar algum significado físico às grandezas, mas a dedução é geral, dentro de suas premissas, para outros fenômenos físicos.

Uma vez que $\mathbf{A}$ é requerida ser positiva-definida, segue-se que ΔV é sempre não negativo, e é nulo apenas se $\delta\mathbf{U} = \mathbf{0}$. Segue-se que o vetor $\mathbf{U}$ que é a solução (única) de (12.76) é o mesmo vetor que minimiza V. Nessa e na próxima seção alguns dos procedimentos mais eficientes para obter o mínimo são tratados: o método do gradiente, o dos gradientes conjugados e um processo de condicionamento.

Aspectos gerais de variação e de minimização podem ser vistos nas seções 10.2 e 14.5.

12.3.1 Método do gradiente

Esse é o método mais intuitivo para determinação do mínimo da função potencial V. Também é chamado de método do máximo decréscimo ("steepest descent method").

Considere-se um processo iterativo para obter o mínimo de (12.75), em que se parte de uma estimativa inicial $\mathbf{U}^{(0)} \in R^N$ para a solução exata $\mathbf{U}$. Em seguida, considera-se uma sucessão de aproximações $\mathbf{U}^{(k)}$, $k = 1, 2, \cdots$, na forma

$$\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)}, \qquad k = 0, 1, 2, \cdots, \tag{12.79}$$

onde cada $\mathbf{d}^{(k)} \in R^N$ é um vetor que indica a **direção da busca** da nova aproximação. $a_k > 0$, $a_k \in R$ é escolhido para gerar a melhor estimativa possível ao longo da direção $\mathbf{d}^{(k)}$. A Figura 12.3 ilustra a função potencial para um caso simples de apenas duas dimensões, uma estimativa de solução $\mathbf{U}^{(k)}$.

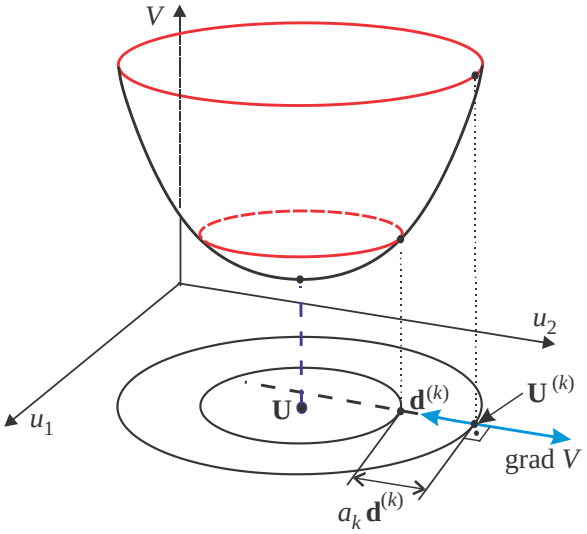

Figura 12.3: Ilustração de uma função potencial para $N = 2$ coordenadas, com curvas de nível, vetor gradiente grad V e vetor direção de busca d.

Diversas formas existem para definir a_k e $\mathbf{d}^{(k)}$, cada uma delas definindo um método com características próprias. Caso se aplique em (12.76) qualquer vetor distinto da solução exata se tem um resíduo, um erro no equilíbrio. Numa dada iteração k, o resíduo é

$$\mathbf{r}^{(k)} = \mathbf{A}\mathbf{U}^{(k)} - \mathbf{F}. \tag{12.80}$$

O potencial é uma função das componentes de deslocamento, isto é, $V = V(\mathbf{U}) = \mathbf{V}(u_1,u_1,...,u_N)$. As componentes cartesianas do vetor gradiente do potencial são

$$\nabla V = \left\{ \frac{\partial V}{\partial u_1}, \frac{\partial V}{\partial u_2}, \cdots, \frac{\partial V}{\partial u_N} \right\} \qquad \longrightarrow \qquad \nabla V_i = \frac{\partial V}{\partial u_i}. \tag{12.81}$$

O operador potencial (12.75), em notação indicial, é

$$V = \frac{1}{2} u_i A_{ij} u_j - F_i u_i. \tag{12.82}$$

Então seu gradiente é

$$\nabla V_i = A_{ij}u_j - F_i \qquad \longrightarrow \qquad \nabla V = \mathbf{AU} - \mathbf{F}. \tag{12.83}$$

Comparando com (12.80) tem-se que para uma estimativa $\mathbf{U}^{(k)}$, a força residual é igual ao gradiente, isto é, $\mathbf{r}^{(k)} = \nabla V^{(k)} = \mathbf{AU}^{(k)} - \mathbf{F}$. Uma forma de definir o vetor direção de busca $\mathbf{d}^{(k)}$ é toma-lo na direção oposta à do gradiente. Deve-se lembrar que o vetor gradiente aponta na direção de maior crescimento da função. Então, a direção oposta deve apontar para uma direção de redução de V. Assim, toma-se

$$\boxed{\mathbf{d}^{(k)} = -\mathbf{r}^{(k)} = -\nabla V^{(k)} = -\mathbf{AU}^{(k)} + \mathbf{F}} \tag{12.84}$$

Entretanto, $\mathbf{d}^{(k)}$ aponta para uma região de menor potencial, mas não necessariamente para o ponto de mínimo. Por isso é necessário um processo iterativo. O gradiente é ilustrado na Figura 12.3. A determinação do comprimento do passo de correção a_k, em (12.79), é feita buscando o valor de a_k que minimiza o potencial em $k+1$, isto é,

$$V(\mathbf{U}^{(k+1)}) \equiv V(\mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)}) \qquad \longrightarrow \qquad \frac{\partial}{\partial a_k}\left[V(\mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)})\right] = 0. \tag{12.85}$$

Porém,

$$V(\mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)}) = \frac{1}{2}\left(\mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)}\right) \cdot \mathbf{A}\left(\mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)}\right) - \left(\mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)}\right)^T \mathbf{F}.$$

Essa expressão pode ser expandida, diferenciada e simplificada (usando $\mathbf{d}^{(k)T}\mathbf{F} = \mathbf{F} \cdot \mathbf{d}^{(k)}$ e $\mathbf{U}^{(k)} \cdot \mathbf{Ad}^{(k)} = \mathbf{d}^{(k)} \cdot \mathbf{AU}^{(k)}$), de forma que (12.85) se torna

$$\frac{\partial}{\partial a_k}\left[V(\mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)})\right] = \mathbf{d}^{(k)} \cdot \left[\mathbf{AU}^{(k)} - \mathbf{F}\right] + a_k \mathbf{d}^{(k)} \cdot \mathbf{A}\ \mathbf{d}^{(k)} = 0. \tag{12.86}$$

Resolvendo para a_k obtém-se o comprimento ótimo de correção na direção $\mathbf{d}^{(k)}$. Assim,o **método do gradiente** pode ser sumarizado no seguinte:

1. Estimativa inicial:	Definir $\mathbf{U}^{(0)} \in R^N$ e $\mathbf{r}^{(0)} = \mathbf{AU}^{(0)} - \mathbf{F}$. $k = -1$.
2. Nova iteração:	$k = k + 1$
3. Direção de correção:	$\mathbf{d}^{(k)} = -\mathbf{r}^{(k)}$,
4. Comprimento da correção:	$a_k = -\dfrac{\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k)}}{\mathbf{d}^{(k)} \cdot \mathbf{Ad}^{(k)}}$
5. Atualização da estimativa:	$\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)} = \mathbf{r}^{(k)} + \mathbf{Ad}^{(k)}$
6. Residuo:	$\mathbf{r}^{(k+1)} = \mathbf{AU}^{(k+1)} - \mathbf{F}$
7. Teste de convergência:	$\dfrac{\left\|\mathbf{r}^{(k+1)}\right\|}{\|\mathbf{F}\|} \leqslant \text{tol}$ e/ou $a_k \dfrac{\left\|\mathbf{d}^{(k)}\right\|}{\left\|\mathbf{U}^{(k+1)}\right\|} \leqslant \text{TOL}$
8. Se não convergiu, ir ao passo 2.	

(12.87)

Quando nada se sabe sobre a solução, a estimativa inicial geralmente é feita fazendo $\mathbf{U}^{(0)} = \mathbf{0}$. Em problemas não lineares, essa estimativa pode ser tomada como a solução convergida do tempo ou nível de carga anterior.

Como este é um problema de minimização de um potencial, a modificação do potencial para satisfazer **condições de restrição** pode ser feita da maneira usual. Ver por exemplo o Capítulo 10.

Resolvendo (12.102) esse erro pode ser relacionado ao número de iterações por

$$\epsilon = \frac{\left\|\mathbf{e}^{(n+1)}\right\|}{\left\|\mathbf{e}^{(0)}\right\|} \geq \left(1-\frac{1}{c\,(\mathbf{A})}\right)^n \qquad \longrightarrow \qquad -\,n\;\ln\left(1-\frac{1}{c\,(\mathbf{A})}\right) \geq \ln\frac{1}{\epsilon}. \tag{12.104}$$

Nesse ponto utiliza-se uma relação que pode ser demonstrada de diversas formas, como com a ajuda gráfica: para $x \leq 1$ tem-se que $-\ln(1-x) \leq x$ Nesse caso, (12.104) pode ser modificada para

$$\frac{n}{c\,(\mathbf{A})} \geq -n\;\ln\left(1-\frac{1}{c\,(\mathbf{A})}\right) \geq \ln\frac{1}{\epsilon} \qquad \longrightarrow \qquad \boxed{n \geq c\,(\mathbf{A})\ln\frac{1}{\epsilon}} \tag{12.105}$$

Essa expressão é o objetivo da presente dedução. Ela mostra que a quantidade de iterações para reduzir o erro inicial a um valor ϵ é proporcional ao número de condição da matriz, desde que seja utilizado um valor adequado do comprimento da correção a.

Exemplo 12.1 - Número de operações em malha padrão pelo método do gradiente

Pode-se fazer algumas experiências com a expressão (12.105). Tomemos um valor típico de erro, $\epsilon = 10^{-6}$. Então, $n \geq 6\;c\,(\mathbf{A})$. Consideremos o domínio unitário 2D padrão da Figura 12.1. Prova-se que numa modelagem de elementos finitos de um problema como o de transferência de calor, plano, com uma única variável, a temperatura, definido pelo operador diferencial laplaciano, **o número de condição da matriz coeficiente é** $c\,(\mathbf{A}) = O(h^{-2})$, onde h é o tamanho do lado do elemento. Então, na malha padrão se tem que o número de graus de liberdade é $N = O(h^{-2})$, conforme (12.12). Isso significa que $c\,(\mathbf{A}) = O(N)$.

Assim, numa malha com $N = 1.000$ graus de liberdade se teria o número de interações estimado por (12.105) como

$$n \geq c\,(\mathbf{A})\ln\frac{1}{\epsilon} = N\ln\frac{1}{\epsilon} = 1000\;13,8 = 13.800 \text{ iterações}, \tag{12.106}$$

que é uma quantidade proibitivamente grande de iterações para atingir o erro requerido na solução. Estimativas similares são válidas em problemas estáticos de mecânica dos sólidos, e outros associados a operadores diferenciais de equações elípticas.

Esse tipo de comportamento do método do gradiente explica porque tornou-se necessário buscar métodos mais eficientes. O mais utilizado deles é uma variação do método do gradiente, sumarizado na próxima seção.

12.3.2 Método do gradiente conjugado - GC

O método do gradiente conjugado foi proposto inicialmente por Hestenes [47] em 1952, com a proposta de ser um método para solução de sistemas lineares de equação de grande porte, definido por matriz simétrica e positiva-definida. É um método mais eficiente que o do gradiente, em que as direções de procura, $\mathbf{d}^{(k)}$, são conjugadas, isto é,

$$\mathbf{d}^{(i)} \cdot \mathbf{A}\mathbf{d}^{(j)} = 0, \qquad \forall i \neq j. \tag{12.107}$$

Nesse método, em vez da **direção de correção** $\mathbf{d}^{(k+1)}$ ser feita na direção contrária ao do gradiente, ela é feita de forma **a ser perpendicular a todas as direções anteriores**. Isso é conseguido fazendo $\mathbf{d}^{(k+1)} = -\mathbf{r}^{(k)} + b_k\mathbf{d}^{(k)}$, onde b_k é determinado de forma que a ortogonalidade seja satisfeita. Detalhes sobre o método podem ser vistos em [54] e [91].

As etapas do método GC são as seguintes:

1. Inicializações:	Dado $\mathbf{U}^{(0)} \in R^N$ e $b_0 = 0$.
1.1 Calcular:	$\mathbf{r}^{(0)} = \mathbf{A}\mathbf{U}^{(0)} - \mathbf{F}$, $\mathbf{d}^{(0)} = -\mathbf{r}^{(0)}$.
	$k = -1$
2. Nova iteração:	$k = k + 1$
3. Comprimento da correção:	$a_k = -\dfrac{\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k)}}{\mathbf{d}^{(k)} \cdot \mathbf{A}\ \mathbf{d}^{(k)}}$
4. Atualização da estimativa:	$\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + a_k\ \mathbf{d}^{(k)}$
5. Residuo:	$\mathbf{r}^{(k+1)} = \mathbf{A}\mathbf{U}^{(k+1)} - \mathbf{F} = \mathbf{r}^{(k)} + a_k \mathbf{A}\ \mathbf{d}^{(k)}$
6. Direção de correção:	$\mathbf{d}^{(k+1)} = -\mathbf{r}^{(k+1)} + b_k \mathbf{d}^{(k)}$
7. Fator de ortogonalização:	$b_k = \dfrac{\mathbf{r}^{(k+1)} \cdot \mathbf{A}\ \mathbf{d}^{(k)}}{\mathbf{d}^{(k)} \cdot \mathbf{A}\ \mathbf{d}^{(k)}} = -\dfrac{\mathbf{r}^{(k+1)} \cdot \mathbf{r}^{(k+1)}}{\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k)}}$
8. Teste de convergência:	$\dfrac{\left\|\mathbf{r}^{(k+1)}\right\|}{\|\mathbf{F}\|} \leqslant \text{tol}$ e/ou $\dfrac{a_k \left\|\mathbf{d}^{(k)}\right\|}{\left\|\mathbf{U}^{(k+1)}\right\|} \leqslant \text{TOL}$
9. Se não convergiu, ir ao passo 2.	

(12.108)

Pós-multiplicando $(12.108)_6$ por $\mathbf{A}\mathbf{d}^{(k)}$ e impondo a ortogonalidade (12.107) do lado esquerdo obtém-se

$$\mathbf{d}^{(k+1)T} \mathbf{A}\ \mathbf{d}^{(k)} = -\mathbf{r}^{(k+1)T} \mathbf{A}\ \mathbf{d}^{(k)} + b_k \mathbf{d}^{(k)T} \mathbf{A}\ \mathbf{d}^{(k)} = 0$$

Em seguida se pode isolar a expressão para b_k, que é aquela vista em $(12.108)_7$.

Existe uma série de teoremas sobre as propriedades das entidades no método dos gradientes conjugados. Uma das principais propriedades é a **ortogonalidade dos resíduos**:

$$\boxed{\mathbf{r}^{(i)} \cdot \mathbf{r}^{(j)} = 0, \qquad \forall i \neq j} \tag{12.109}$$

Uma outra propriedade pode ser deduzida:

$$\boxed{\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k+1)} = 0} \tag{12.110}$$

Isso pode ser verificado pré-multiplicando $\mathbf{r}^{(k+1)}$ em $(12.108)_5$ por $\mathbf{d}^{(k)}$:

$$\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k+1)} = \mathbf{d}^{(k)} \cdot \mathbf{r}^{(k)} + a_k \mathbf{d}^{(k)} \cdot \mathbf{A}\ \mathbf{d}^{(k)}.$$

De $(12.108)_3$, $\mathbf{d}^{(k)} \cdot \mathbf{A}\ \mathbf{d}^{(k)} = -\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k)} / a_k$. Então se chega a (12.110). Esse resultado mostra que o novo resíduo é ortogonal à direção de busca $\mathbf{d}^{(k)}$.
Pré-multiplicando $\mathbf{r}^{(k+1)}$ em $(12.108)_5$ por $\mathbf{d}^{(k+1)}$:

$$\mathbf{d}^{(k+1)} \cdot \mathbf{r}^{(k+1)} = \mathbf{d}^{(k+1)} \cdot \mathbf{r}^{(k)} + a_k \mathbf{d}^{(k+1)} \cdot \mathbf{A}\ \mathbf{d}^{(k)},$$

e usando a ortogonalidade de $\mathbf{d}^{(k+1)}$, chega-se a uma outra relação:

$$\boxed{\mathbf{d}^{(k+1)} \cdot \mathbf{r}^{(k+1)} = \mathbf{d}^{(k+1)} \cdot \mathbf{r}^{(k)}} \tag{12.111}$$

Essa expressão pode ser colocada na forma $\mathbf{d}^{(k+1)} \cdot \left(\mathbf{r}^{(k+1)} - \mathbf{r}^{(k)}\right) = 0$. Porém, de $(12.108)_5$, o parêntesis é igual a

$$\left(\mathbf{r}^{(k+1)} - \mathbf{r}^{(k)}\right) = a_k \mathbf{A}\ \mathbf{d}^{(k)}, \tag{12.112}$$

o que resulta na condição de ortogonalidade $\mathbf{d}^{(k+1)} \cdot \mathbf{A}\ \mathbf{d}^{(k)} = 0$.

A segunda igualdade em $(12.108)_7$, é uma forma mais barata de obter b_k, sem envolver produtos matriz $\times$ vetor. Consideremos o numerador da expressão, usando (12.112)

$$n \geq \frac{1}{2}\sqrt{c(\mathbf{A})}\ln\frac{2}{\epsilon} = \frac{1}{2}N^{1/2}\ln\frac{2}{\epsilon} = \frac{1}{2}1000^{1/2}\ 13,8 = 229 \text{ iterações} \tag{12.117}$$

Deve-se comparar com a estimativa de 14 mil iterações necessárias com o uso do método do gradiente vista no Exemplo 1. A grande diferença é que no método do gradiente, o número de iterações cresce com o número de condição, enquanto no método dos gradientes conjugados ele cresce com sua raiz.

Observação: Uma característica de ambos os métodos baseados no gradiente é que suas etapas envolvem apenas produtos matriz x vetor, vetor x vetor e somas vetor + vetor. Essa característica favorece o uso de formas de **armazenamento esparso da matriz**, apenas dos termos não nulos, de forma a reduzirem ao máximo o número de operações por iteração.

12.3.3 Método do gradiente conjugado pré-condicionado

O número de operações no método dos gradientes conjugados é proporcional à raiz quadrada do número de condição da matriz. Assim, é de interesse identificar algum tipo de transformação que possa reduzir esse número pela melhoria do condicionamento da matriz. O processo mais utilizado é o **pré-condicionamento pela fatorização incompleta de Cholesky**, que será brevemente descrito a seguir (ver [6] por exemplo).

Considera-se o problema $\mathbf{AU} = \mathbf{F}$ em (12.76). Considera-se uma transformação vetorial dada por

$$\bar{\mathbf{U}} = \mathbf{TU} \tag{12.118}$$

onde $\mathbf{T}$ é uma matriz de transformação, não singular, de dimensões $N \times N$. Substituindo $\mathbf{U} = \mathbf{T}^{-1}\bar{\mathbf{U}}$ no problema original (12.76), e pré-multiplicando o resultado por $\mathbf{T}^{-T}$, obtem-se

$$\mathbf{AU} = \mathbf{F} \quad \longrightarrow \quad \mathbf{AT}^{-1}\bar{\mathbf{U}} = \mathbf{F} \quad \longrightarrow \quad \underbrace{\mathbf{T}^{-T}\mathbf{AT}^{-1}}_{\bar{\mathbf{A}}}\bar{\mathbf{U}} = \underbrace{\mathbf{T}^{-T}\mathbf{F}}_{\bar{\mathbf{F}}}, \tag{12.119}$$

isto é, tem-se um problema algébrico $\bar{\mathbf{A}}\bar{\mathbf{U}}=\bar{\mathbf{F}}$ em coordenadas transformadas ("deslocamentos" de difícil interpretação física). Se esse problema for resolvido pelo método do gradiente, se tem a $(k+1)$-ésima aproximação de $\bar{\mathbf{U}}$ dada por $(12.87)_5$:

$$\bar{\mathbf{U}}^{(k+1)} = \bar{\mathbf{U}}^{(k)} - a_k\left[\bar{\mathbf{A}}\bar{\mathbf{U}}^{(k)} - \bar{\mathbf{F}}\right]. \tag{12.120}$$

As iterações podem prosseguir até a convergência de $\bar{\mathbf{U}}^{(k)}$, mas nesse ponto será necessário voltar ao espaço físico fazendo a transformação inversa pela resolução do sistema algébrico

$$\mathbf{TU} = \bar{\mathbf{U}}. \tag{12.121}$$

Em vez da transformação (12.118), uma **outra forma usual** de apresentar o condicionamento de forma geral consiste em identificar uma matriz $\mathbf{M}$, simétrica, positiva definida, e pré-multiplicar $\mathbf{AU} = \mathbf{F}$ pela sua inversa:

$$\underbrace{\mathbf{M}^{-1}\mathbf{A}}_{\hat{\mathbf{A}}}\mathbf{U} = \underbrace{\mathbf{M}^{-1}\mathbf{F}}_{\hat{\mathbf{F}}}. \tag{12.122}$$

Os pré-condicionadores $\mathbf{T}$ e $\mathbf{M}$ devem ser escolhidos de forma que $\hat{\mathbf{A}}$ seja melhor condicionado que $\mathbf{A}$. Além disso, para permitir o uso do método de gradiente conjugado, $\hat{\mathbf{A}}$ deve ser simétrico e positivo definido. Uma forma de garantir essas características consiste em considerar que $\mathbf{M}$ possa ser gerado como simétrica e positiva definida, na forma

$$\mathbf{M} = \mathbf{P}^{-T}\mathbf{P}^{-1}, \text{ tal que } \mathbf{M}^{-1} = \mathbf{PP}^T. \tag{12.123}$$

- **Item 4**. Fazendo as transformações temos

$$\begin{aligned} \bar{\mathbf{U}}^{(k+1)} &= \bar{\mathbf{U}}^{(k)} + \bar{a}_k\ \bar{\mathbf{d}}^{(k)}, \\ \mathbf{TU}^{(k+1)} &= \mathbf{TU}^{(k)} + \bar{a}_k\ \mathbf{TD}^{(k)}, \quad \rightarrow \quad \boxed{\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + \bar{a}_k\ \mathbf{d}^{(k)}} \end{aligned} \tag{12.134}$$

- **Item 5**. O novo resíduo fica (usando (12.130)):

$$\begin{aligned} \bar{\mathbf{r}}^{(k+1)} &= \bar{\mathbf{r}}^{(k)} + \bar{a}_k \bar{\mathbf{A}}\ \bar{\mathbf{d}}^{(k)}, \\ \mathbf{T}^{-T}\mathbf{r}^{(k+1)} &= \mathbf{T}^{-T}\mathbf{r}^{(k)} + \bar{a}_k \left(\mathbf{T}^{-T}\mathbf{A}\mathbf{T}^{-1}\right)\ \mathbf{T}\mathbf{d}^{(k)}, \\ &\rightarrow \quad \boxed{\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} + \bar{a}_k \mathbf{A}\mathbf{d}^{(k)}} \end{aligned} \tag{12.135}$$

- **Item 6**. A nova direção pode ser transformada por (usando (12.130)):

$$\begin{aligned} \bar{\mathbf{d}}^{(k+1)} &= -\bar{\mathbf{r}}^{(k+1)} + \bar{b}_k \bar{\mathbf{d}}^{(k)}, \\ \mathbf{T}\mathbf{d}^{(k+1)} &= -\mathbf{T}^{-T}\mathbf{r}^{(k+1)} + \bar{b}_k \mathbf{T}\mathbf{d}^{(k)}, \\ &\rightarrow \quad \boxed{\mathbf{d}^{(k+1)} = -\mathbf{M}^{-1}\mathbf{r}^{(k+1)} + \bar{b}_k \mathbf{d}^{(k)}} \end{aligned} \tag{12.136}$$

- **Item 7**. O fator de ortogonalidade também pode ser transformado:

$$\begin{aligned} \bar{b}_k &= -\frac{\bar{\mathbf{r}}^{(k+1)} \cdot \bar{\mathbf{r}}^{(k+1)}}{\bar{\mathbf{d}}^{(k)} \cdot \bar{\mathbf{r}}^{(k)}}, \\ &= -\frac{\mathbf{r}^{(k+1)} \cdot \mathbf{T}^{-1}\mathbf{T}^{-T}\mathbf{r}^{(k+1)}}{\mathbf{d}^{(k)} \cdot \mathbf{T}^{T}\mathbf{T}^{-T}\mathbf{r}^{(k)}}. \\ &\rightarrow \quad \boxed{\bar{b}_k = -\frac{\mathbf{r}^{(k+1)} \cdot \check{\mathbf{r}}^{(k+1)}}{\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k)}}, \ \text{ onde } \check{\mathbf{r}}^{(k+1)} = \mathbf{M}^{-1}\mathbf{r}^{(k+1)}} \end{aligned} \tag{12.137}$$

O fluxograma do método do GC incorporando um pré-condicionamento de matriz dividida, fica na seguinte forma

1.1 Inicializações:	$k = -1, \bar{b}_0 = 0.$
	$\mathbf{r}^{(0)} = \mathbf{A}\mathbf{U}^{(0)} - \mathbf{F}, \ \mathbf{d}^{(0)} = -\mathbf{M}^{-1}\mathbf{r}^{(0)}.$
2. Nova iteração:	$k = k + 1$
3. Comprimento da correção:	$\bar{a}_k = \dfrac{\mathbf{r}^{(k)} \cdot \check{\mathbf{r}}^{(k)}}{\mathbf{d}^{(k)} \cdot \mathbf{A}\mathbf{d}^{(k)}}, \ \text{ onde } \check{\mathbf{r}}^{(k)} = \mathbf{M}^{-1}\mathbf{r}^{(k)}$
4. Atualização da estimativa:	$\mathbf{U}^{(k+1)} = \mathbf{U}^{(k)} + \bar{a}_k\ \mathbf{d}^{(k)}$
5. Residuo:	$\mathbf{r}^{(k+1)} = \mathbf{r}^{(k)} + \bar{a}_k \mathbf{A}\mathbf{d}^{(k)}$
6. Gradiente e direção de correção:	$\mathbf{d}^{(k+1)} = -\check{\mathbf{r}}^{(k+1)} + \bar{b}_k \mathbf{d}^{(k)}$
	onde $\check{\mathbf{r}}^{(k+1)} = \mathbf{M}^{-1}\mathbf{r}^{(k+1)}$
7. Fator de ortogonalização:	$\bar{b}_k = -\dfrac{\mathbf{r}^{(k+1)} \cdot \check{\mathbf{r}}^{(k+1)}}{\mathbf{d}^{(k)} \cdot \mathbf{r}^{(k)}}$

(12.138)

Nota-se que em cada iteração é necessário resolver o sistema linear

$$\mathbf{M}\check{\mathbf{r}}^{(k+1)} = \mathbf{r}^{(k+1)}, \tag{12.139}$$

para obter $\check{\mathbf{r}}^{(k+1)}$ para as etapas 6 e 7. É o custo do método, exceto pela identificação do condicionante $\mathbf{M}$. Outros custos do fluxograma são matriz × vetor, uma vez na inicialização, $\mathbf{A}\mathbf{U}^{(0)}$, e uma

Laplace é estimado por $c\,(\mathbf{A}) = O(h^{-2})$ e por $c(\hat{\mathbf{A}}) = O(h^{-1})$ na matriz pré-condicionada, onde h é o tamanho do lado do elemento. Então, na malha padrão se tem que o número de graus de liberdade é $N = O(h^{-2})$, conforme (12.12). Isso significa $c\left(\hat{\mathbf{A}}\right) = O(N^{1/2})$. Assim, numa malha com $N = 1.000$ graus de liberdade se teria o número de interações estimado por (12.116) como

$$n \geq \frac{1}{2}\sqrt{c(\hat{\mathbf{A}})}\ln\frac{2}{\epsilon} = \frac{1}{2}N^{1/4}\ln\frac{2}{\epsilon} = \frac{1}{2}1000^{1/4}\ 13,8 = 41 \text{ iterações}$$

Deve-se comparar com a estimativa de 14 mil iterações necessárias com o uso do método do gradiente vista no Exemplo 1, e 229 iterações para o GC não condicionado. A grande diferença é que no método do GC, o número de iterações cresce com $N^{1/2}$ enquanto no método dos GC condicionado ele cresce com $N^{1/4}$.

O número de operações por iteração pode ser estimado da seguinte forma:

1 produto matriz $\times$ vetor ($\mathbf{A}\mathbf{d}^{(k)}$):	- $\mathbf{A}$ cheia: $O(2N^2)$
	- $\mathbf{A}$ banda: $O(2bN)$
1 produto matriz $\times$ vetor ($\mathbf{M}^{-1}\mathbf{r}^{(k)}$) :	- $\mathbf{A}$ banda: $O(4bN)$
2 produtos escalares ($\mathbf{r}^{(k)} \cdot \check{\mathbf{r}}^{(k)}$ e $\left\|\mathbf{r}^{(k+1)}\right\|$):	- $O(4N)$

(12.143)

12.4 Comentários gerais

O número total de operações nas três diferentes versões dos método de gradiente pode ser estimado para o problema padrão 2D, para matriz com banda, com tolerância de erro nas iterações $\epsilon = 10^{-6}$. Nota-se que no problema padrão 2D, $N = O(h^{-2})$ e $b = O(h^{-1}) = O(N^{1/2})$. Nota-se que o método de Cholesky em matriz banda apresenta estimativa de $2N^2$ operações para o mesmo problema padrão (Tabela 12.1).

Tabela 12.2: Sumário das estimativas de números assintóticos de operações para os métodos de gradiente, para o problema padrão 2D.

	Gradiente	GC	GC condicionado
Oper./iter	$O(2bN) = O(2N^{3/2})$	$O(2bN) = 2N^{3/2}$	$O(6bN) = 6N^{3/2}$
Num. iter $n \geq$	$c(\mathbf{A})\log\frac{1}{\epsilon}$	$\frac{1}{2}\sqrt{c(\mathbf{A})}\log\frac{2}{\epsilon}$	$\frac{1}{2}\sqrt{c(\hat{\mathbf{A}})}\log\frac{2}{\epsilon}$
Num. condic. c	$O(N)$	$O(N)$	$O(N^{1/2})$
Oper. total	$2N^{5/2}\ln\frac{1}{\epsilon} = 28\ N^{2,5}$	$N^2\ln\frac{2}{\epsilon} = 15\ N^2$	$3N^{7/4}\ln\frac{2}{\epsilon} = 44\ N^{1,75}$

A estimativas assintóticas de número de operações para as três variantes iterativas são $28N^{2,5}$, $15N^2$ e $44N^{1,75}$, respectivamente. Esses valores podem ser comparados à estimativa para o método de Cholesky em matriz banda, que é $2N^2$. Observa-se na Tabela 12.2 que a parte mais importante na composição do expoente de N é o número de operações por iteração, que aparecem como $O(2bN) = 2N^{3/2}$ para os três métodos. É o custo do produto matriz banda $\times$ vetor. A estimativa feita aqui considerou a matriz densa de termos não nulos sob a banda. Matrizes de MEF são tipicamente esparsas, de forma que algoritmos podem ser construídos para fazer o produto matriz-vetor de forma mais eficiente, eliminando as operações com zero. Dessa forma, o número de operações por iteração pode cair para próximo de $O(N)$. Isso pode levar ao custo total no método GC condicionado para $O(aN^{1,5})$, para algum $a > 0$.

O método de gradiente conjugado pré-condicionado é o mais eficiente para a solução do sistema algébrico com matriz simétrica positiva-definida, comparado aos métodos diretos baseados em fatoração de Gauss, quando aplicado em problemas regulares, com geometria e malha padronizados. Os métodos iterativos sofrem bastante perda de eficiência com matrizes mal condicionadas (alto