

**MÉTODOS ITERATIVOS PARA RESOLUÇÃO DE
SISTEMAS LINEARES**

Emanuele Furlan Pretto

FOZ DO IGUAÇU - PR

MARÇO - 2011

RESUMO

Este trabalho consiste num estudo dos métodos iterativos estacionários e o método dos gradientes conjugados para a resolução de sistemas de equações lineares. Elementos importantes da Álgebra Linear fundamentam o trabalho. Uma ênfase numérica com a utilização do MATLAB é apresentada.

Sumário

Introdução	1
1 Conceitos Básicos	4
1.1 Resolução de sistemas lineares	4
1.2 Autovalores e autovetores	7
1.3 Equações de diferenças	11
2 Métodos Iterativos Estacionários	13
2.1 Os métodos de Jacobi e Gauss-Seidel	14
2.1.1 O método de Jacobi	14
2.1.2 O método de Gauss-Seidel	15
2.2 O método SOR	17
2.3 Comparação dos métodos estacionários	19
2.4 Alguns resultados teóricos	20
3 O Método dos Gradientes Conjugados	23
3.1 O método do Gradiente	24
3.2 O método dos Gradientes Conjugados	24
4 Experiências Numéricas	29

4.1	Comparação entre os métodos	29
4.2	Desempenho do método dos Gradientes Conjugados	31
	Conclusões	37
	Referências Bibliográficas	39

Introdução

Os sistemas de equações lineares foram abordados inicialmente no oriente, data-se sua primeira aparição por volta do ano 300 a.C. No entanto, o conceito de *determinante* surge somente em 1683 com Seki Kowa e após dez anos, começa a desenvolver-se também no ocidente com Leibniz. Assim seguiu-se com alguns trabalhos isolados até que Jacobi (1804-1851) chamado de “o grande algorista” transforma todo esse estudo em algo parecido com a forma simples que conhecemos hoje. Isso porque, as descobertas de Gauss, por inúmeras vezes não eram publicadas, mesmo que auxiliariam ou, estivessem mais avançadas, para a matemática da época.

O determinante, foi concebido como uma ferramenta desenvolvida para resolução dos sistemas, conceito este que precedeu o conceito de *matriz* como o conhecemos hoje. Há muito tempo que se sabe que a eliminação Gaussiana é um método muito eficiente para resolução de sistemas lineares e na prática o determinante como método não apresenta utilidade, embora seja um conceito teórico fundamental da Matemática.

Nas aplicações relevantes de várias áreas das ciências, os modelos matemáticos são representados por sistemas de equações lineares. Até meados de 1970 os métodos diretos eram muito utilizados para resolução de modelos lineares. Entenda-se por eles, fatoração LU (variação da eliminação Gaussiana), fatoração QR, entre outros [4].

Na medida do incremento massivo do desenvolvimento tecnológico e computacional, os modelos lineares tornaram-se grandes e começaram a envolver matrizes esparsas [3]. Nestes casos, como por exemplo a discretização de Equações Diferenciais Parciais [2], os *métodos iterativos* são bastantes indicados. Neste contexto, os métodos diretos em geral são impraticáveis por exigirem muitos dados adicionais que devem ser processados gerando a necessidade de grande memória adicional. Além disso, o processamento massivo ocasiona erros cruciais em aritmética de ponto flutuante [5].

Os métodos iterativos apresentam duas vantagens importantes sobre os métodos diretos. Praticamente não exigem memória adicional e, mais importante, permitem obter aproximação de baixo custo com pouco processamento. De fato, a filosofia deles é *gerar solução parciais* a todo momento permitindo-nos avaliá-las e melhorá-las ou, aceitá-las, conforme a necessidade.

Os métodos iterativos são formulados de maneira relativamente simples, mas exigem um bom conhecimento de álgebra matricial numérica e alguma prática computacional para de fato serem explorados. Do ponto de vista teórico, o conceito básico para abordá-los são os *autovalores*. Por esses motivos, raramente são tratados nos cursos de graduação, principalmente de licenciatura.

O objetivo deste trabalho é apresentar alguns dos métodos iterativos mais utilizados na literatura clássica no estudo de resolução de sistemas lineares. Abordaremos métodos que levam os nomes dos matemáticos citados anteriormente, como o método de *Jacobi*, o de *Gauss-Seidel* e outros, como os *métodos SOR* e *Gradientes Conjugados*. Com isso, é possível desenvolver uma álgebra acessível a qualquer um que goste de matemática e busque solucionar problemas do mundo real. A junção da matemática pura com diversas áreas como economia, engenharia e até medicina, fornece situações importantes para toda a humanidade. Buscaremos, no entanto, apresentar de forma sucinta o que é mais importante para a resolução dos sistemas de equações lineares.

No capítulo 1 abordamos os elementos da Álgebra Linear que fundamentam o trabalho, dando ênfase sobretudo nos autovalores e autovetores.

No segundo capítulo iniciamos a apresentação dos *métodos iterativos estacionários*, buscamos dar exemplos idênticos para diferentes métodos, para que seja possível iniciar uma comparação. Ao final do capítulo são apresentadas as formas genéricas dos métodos de Jacobi, Gauss-Seidel e SOR.

O terceiro capítulo é dedicado somente aos *métodos dos Gradientes e dos Gradientes Conjugados*. Ideal para serem utilizados em sistemas lineares de grande porte com diversas entradas nulas. O algoritmo do método dos Gradientes Conjugados é apresentado ao final do capítulo.

No quarto e último capítulo fazemos as experiências numéricas com os métodos que obtivemos conhecimento. Por meio do MATLAB (software de programação matemá-

tica) é possível desenvolver exemplos, comparar os métodos e gerar gráficos para verificar se os resultados foram satisfatórios.

Busca-se então, mostrar conteúdos não abordados na maioria dos cursos de licenciatura em Matemática e novamente não apresentado na Especialização em Matemática. O gosto pessoal pelo assunto desenvolvido também foi um auxílio para que este trabalho se desenvolvesse.

Capítulo 1

Conceitos Básicos

Neste capítulo vamos escrever alguns conceitos básicos para a posterior apresentação dos métodos iterativos. Uma visão geral dos métodos de resolução usuais, baseados em fatorações e o problema de esparsidade são apresentados. Também enfatizamos o tema principal para a compreensão da convergência dos processos que são os autovalores. Além disso consideraremos duas importantes classes de matrizes: as diagonalizáveis e as simétricas positivas definidas. No caso do leitor se sentir confiante, pode suprimir esse capítulo, no entanto, lhe é recomendado, para que se acostume com a notação utilizada.

1.1 Resolução de sistemas lineares

Durante todo este texto vamos considerar o seguinte problema central: Dada uma matriz quadrada A de ordem n e $b \in \mathbb{R}^n$, queremos obter a solução do sistema de equações lineares

$$Ax = b$$

A condição da existência da solução é que $b \in \mathcal{R}(A)$ em que:

$$\mathcal{R}(A) = \{b \in \mathbb{R}^m \mid Ax = b; x \in \mathbb{R}^n\}$$

Isso pode ser visto interpretando o sistema acima como combinação linear de colunas:

$$b = x_1 a_1 + x_2 a_2 + \dots + x_n a_n$$

em que x_i é a componente i do vetor x e a_i é a coluna i da matriz A . Esta condição independe da singularidade de A , assim como, o seu posto.

Quando n não é grande e ainda a matriz A apresenta um padrão de esparsidade regular (veja em [3]), os métodos diretos são indicados. Um representante por excelência deles é a fatoração LU , uma variação da eliminação Gaussiana.

Se pudermos fatorar A na forma $A = LU$ com L e U , em princípio, não singulares temos:

$$Ax = b \quad \Rightarrow \quad LUx = b,$$

e podemos fazer

$$Ly = b \quad \text{e} \quad Ux = y.$$

que são sistemas fáceis de resolver.

Na fatoração LU a matriz U é triangular superior, aquela mesma obtida na eliminação Gaussiana e a matriz L é triangular inferior contendo os multiplicadores da eliminação e 1's na diagonal.

Exemplo 1.1 Considere o sistema linear:

$$A = \begin{pmatrix} 4 & 5 & 6 \\ 4 & 12 & 14 \\ 8 & 31 & 45 \end{pmatrix} x = \begin{pmatrix} 7 \\ 6 \\ 2 \end{pmatrix} = b$$

então,

$$A = \begin{pmatrix} 4 & 5 & 6 \\ 4 & 12 & 14 \\ 8 & 31 & 45 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 3 & 1 \end{pmatrix} \begin{pmatrix} 4 & 5 & 6 \\ 0 & 7 & 8 \\ 0 & 0 & 9 \end{pmatrix} = LU.$$

Agora, podemos resolver os dois sistemas triangulares. Primeiro o sistema triangular inferior:

$$Ly = b \Leftrightarrow \begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 2 & 3 & 1 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \\ y_3 \end{pmatrix} = \begin{pmatrix} 7 \\ 6 \\ 2 \end{pmatrix}$$

cuja solução do sistema acima é $y = (7, -1, -9)$. Agora o sistema triangular superior:

$$Ux = y \Leftrightarrow \begin{pmatrix} 4 & 5 & 6 \\ 0 & 7 & 8 \\ 0 & 0 & 9 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} 7 \\ -1 \\ -9 \end{pmatrix}$$

cuja solução do sistema acima é $x = (2, 1, -1)$, que é também a solução do sistema original.

O problema central que afeta a fatoração LU em problemas esparsos é a chamada criação de *fill in* ou preenchimento nas matriz L e U no processo da fatoração.

Suponha que A tenha o seguinte padrão de esparsidade:

$$A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & & & \\ \times & & \times & & \\ \times & & & \times & \\ \times & & & & \times \end{pmatrix},$$

em que \times representam elementos não nulos, onde não vemos elementos, entendamos, *zeros*.

Supondo que, após a fatoração L e U sejam armazenadas em uma mesma matriz, como de fato é feito nos códigos computacionais, teremos em geral um preenchimento completo:

$$A = \begin{pmatrix} \times & \times & \times & \times & \times \\ \times & \times & \bullet & \bullet & \bullet \\ \times & \bullet & \times & \bullet & \bullet \\ \times & \bullet & \bullet & \times & \bullet \\ \times & \bullet & \bullet & \bullet & \times \end{pmatrix},$$

gerando caracteres do tipo “ \bullet ” . Outro problema importante neste contexto é os erros propagados em aritmética de ponto flutuante [5]. Com os preenchimentos mais cálculos devem ser feitos e, portanto, mais erros são propagados, podendo resultar em imprecisão do resultado.

Outro fator negativo desses métodos é que a solução é obtida somente após um número fixo total de cálculos. Na eliminação, ou na fatoração LU só teremos a solução no final do processo de eliminação de todas as variáveis e depois da resolução dos sistemas triangulares, não sendo possível obter uma aproximação nas fases intermediárias do processo.

Os métodos iterativos tem por objetivo resolver essas deficiências:

- não criar preenchimento,
- produzir soluções intermediárias.

É o que veremos nos próximos capítulos em que os descreveremos, assim como, suas propriedades e aplicações.

Para análise do processo de convergência dos métodos iterativos é importante lembrar de um conceito central da Álgebra Linear: os *autovalores*. É o que faremos a seguir.

1.2 Autovalores e autovetores

Dada uma matriz quadrada A , queremos obter vetores x não nulos e escalares λ tais que:

$$Ax = \lambda x$$

em que os escalares λ são os *autovalores* e os vetores não nulos x são os *autovetores* associados da matriz A .

Como os autovetores devem ser não nulos, podemos encontrar primeiro os autovalores como as raízes do chamado polinômio característico

$$P(\lambda) = \det(A - \lambda I).$$

Obtido os autovalores, podemos obter facilmente os autovetores resolvendo os sistemas singulares associados a cada λ_i :

$$Ax = \lambda_i x \Leftrightarrow Ax - \lambda_i x = 0 \Leftrightarrow (A - \lambda_i I)x = 0.$$

As raízes do polinômio característico são os autovalores procurados que possibilitam o cálculo dos autovetores à eles associados.

Exemplo 1.2

$$A = \begin{pmatrix} 4 & -3 \\ 2 & -1 \end{pmatrix}.$$

Da equação característica $\det(A - \lambda I) = \lambda^2 - 3\lambda + 2 = 0$, temos os autovalores: $\lambda = 1$ e $\lambda = 2$. Os autovetores para cada λ são:

$\lambda_1 = 1$:

$$(A - \lambda_1 I)x = \begin{bmatrix} 3 & -3 \\ 2 & -2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow x_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$\lambda_2 = 2$:

$$(A - \lambda_2 I)x = \begin{bmatrix} 2 & -3 \\ 2 & -3 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow x_2 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

Calcular os autovalores de uma matriz é um problema da Álgebra Linear considerado *difícil*. De fato, a maneira pela qual calculamos os autovalores, utilizando a equação característica, não pode ser feito quando n é grande. Neste caso, métodos numéricos devem ser empregados [4].

Um resultado interessante relaciona os elementos da matriz com seus autovalores em termos do determinante e do traço da matriz conforme dito (em [12] e) a seguir:

Proposição 1 *Se A tem autovalores $\lambda_1, \lambda_2, \dots, \lambda_n$ então*

- $\det(A) = \lambda_1 \lambda_2 \cdots \lambda_n$

- $tr(A) = a_{11} + a_{22} + \dots + a_{nn} = \lambda_1 + \lambda_2 + \dots + \lambda_n$

Uma classe de matrizes importantes são as matrizes *diagonalizáveis*. É o caso da matriz do exemplo acima, que também ilustra os resultados que se seguem.

Definição 1 *Uma matriz A de ordem n é diagonalizável quando possui n autovetores linearmente independente (LI).*

Teorema 1 *Se A tem n autovalores distintos então, ela é diagonalizável.*

Teorema 2 *Se A é diagonalizável então, existe uma matriz X , não singular, tal que $A = X\Delta X^{-1}$ em que*

$$\Delta = \begin{pmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_n \end{pmatrix} \quad e \quad X = \begin{pmatrix} | & | & \dots & | \\ x_1 & x_2 & \dots & x_n \\ | & | & \dots & | \end{pmatrix}.$$

Prova: De fato,

Como $Ax_1 = \lambda_1 x_1$ e $Ax_2 = \lambda_2 x_2$, temos que:

$$\begin{pmatrix} | & | \\ Ax_1 & Ax_2 \\ | & | \end{pmatrix} = \begin{pmatrix} | & | \\ \lambda_1 x_1 & \lambda_2 x_2 \\ | & | \end{pmatrix}$$

o que implica em

$$A \begin{pmatrix} | & | \\ x_1 & x_2 \\ | & | \end{pmatrix} = \begin{pmatrix} | & | \\ \lambda_1 x_1 & \lambda_2 x_2 \\ | & | \end{pmatrix} \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}, \text{ ou seja,}$$

$$AX = X\Lambda, \text{ logo}$$

$$A = AXX^{-1} = X\Lambda X^{-1}. \quad \blacksquare$$

Proposição 2 *Se $Ax = \lambda x$ então, $A^k x = \lambda^k x$.*

Proposição 3 *Se A é diagonalizável, ou seja, $A = X\Lambda X^{-1}$ então, $A^k = X\Lambda^k X^{-1}$.*

Teorema 3 (Espectral) Se $A = A^t$ então, $A = Q\Lambda Q^t (= \lambda_1 q_1 q_1^t + \dots + \lambda_n q_n q_n^t)$ em que Q é uma matriz ortogonal ¹.

Exemplo 1.3 Seja,

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

Então,

$$\lambda_1 = 1 \quad \Rightarrow \quad x_1 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{e} \quad \lambda_2 = 3 \quad \Rightarrow \quad x_2 = \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

Assim,

$$q_1 = \frac{x_1}{\|x_1\|} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \quad \text{e} \quad q_2 = \frac{x_2}{\|x_2\|} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

e portanto, $A = Q\Lambda Q^t$, isto é:

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & 3 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{bmatrix}.$$

Uma importante classe de matrizes simétricas que abordaremos com os métodos iterativos neste trabalho são as positivas definidas. Segue a definição delas.

Definição 2 Uma matriz simétrica $A \in \mathbb{R}^{n \times n}$ é positiva definida (SDP) se $v^t A v > 0$ para todo $v \in \mathbb{R}^n$ não nulo.

A matriz do exemplo anterior é um exemplo de matriz SDP. A condição estabelecida na definição não é efetivamente prática de ser verificada. Para tal consideramos um resultado importante que pode ser conferido em [12].

Definição 3 As seguintes afirmações são equivalentes:

- A é SDP;
- Todos os pivôs de A são positivos;

¹ q_k é o vetor dividido pela sua norma ($\sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$); $Q_{n \times n}$ é ortogonal quando $Q^t Q = Q Q^t = I$, ($Q^{-1} = Q^t$).

- Todos os autovalores de A são positivos.

Vimos que os autovalores de A , no exemplo anterior são positivos: 1 e 3. Lembrando, os pivôs são os elementos da diagonal da fatoração LU :

$$\begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{2} & 1 \end{bmatrix} \begin{bmatrix} 2 & -1 \\ 0 & \frac{3}{2} \end{bmatrix} = LU,$$

assim os pivôs são 2 e $\frac{3}{2}$.

A forma da definição é geralmente útil para verificar se uma matriz não é SDP. Por exemplo, seja

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

Se $v = (1, 0)$, então $v^t A v = -2 < 0$, logo a matriz dada não é SDP.

1.3 Equações de diferenças

Vamos considerar um conceito básico para análise da convergência dos métodos iterativos lineares.

Definição 4 Dado um vetor $v_0 \in \mathbb{R}^n$ e uma matriz $G \in \mathbb{R}^{n \times n}$. Uma equação de diferenças envolvendo uma sequência de vetores $\{v_k\}$ e G tem a expressão:

$$v_k = G v_{k-1}.$$

A solução da equação de diferenças é o limite da sequência $\{v_k\}$.

Note que:

$$v_1 = G v_0 \quad \Rightarrow \quad v_2 = G v_1 = G(G v_0) = G^2 v_0$$

e portanto,

$$v_k = G^k v_0.$$

Uma ferramenta básica para analisar o comportamento das equações de diferenças é o *raio espectral* de uma matriz.

Definição 5 *O maior autovalor em módulo de uma matriz quadrada G é chamado raio espectral que denotaremos por $\rho(G)$, isto é:*

$$\rho(G) = \max_i |\lambda_i|.$$

Para analisar o comportamento de uma equação de diferenças, observamos que se G é diagonalizável por $G = X\Lambda X^{-1}$, então podemos escrever:

$$v_k = Gv_{k-1} = G^k v_0 = X\Lambda^k X^{-1}v_0$$

considerando $c = X^{-1}v_0$, e como Λ é diagonal, podemos expandir a expressão anterior em termos de combinação linear:

$$v_k = c_1\lambda_1^k x_1 + c_2\lambda_2^k x_2 + \cdots + c_n\lambda_n^k x_n = \sum_{j=1}^n c_j \lambda_j^k x_j,$$

em que, c_j é a componente j do vetor c , λ_j é o autovalor j de G e x_j seu autovetor associado.

Esta expressão indica que o raio espectral de G *governa* o comportamento da equação de diferenças.

Por exemplo, nas famosas matrizes de *Markov* (veja em [12]) temos um autovalor, digamos, $\lambda_1 = 1$ e todos os outros $|\lambda_j| < 1$, $j = 2, \dots, n$. Então, neste caso a sequência $\{v_k\}$ converge para o vetor $c_1 x_1$, que é um autovetor associado ao autovalor $\lambda_1 = 1$. No caso em que o raio espectral é menor que a unidade, a sequência convergirá para o vetor nulo. Este é o caso que estaremos interessados, para avaliar o comportamento dos métodos iterativos, no próximo capítulo.

Capítulo 2

Métodos Iterativos Estacionários

Esclarecemos que, em quase todo este capítulo, o *super-índice* k representa a aproximação x para o k -ésimo iterado. Já o *sub-índice* i representa a i -ésimo componente do vetor. Portanto x_i^k representa a componente i do vetor x^k .

Os métodos iterativos estacionários baseam-se num processo do tipo

$$Sx^{k+1} = Tx^k + b$$

em que $A = S - T$. Isso acontece a partir da equação $Ax = b$ que fica, $(S - T)x = b$ e então, $Sx = Tx + b$. Esses métodos partem de uma escolha inicial x^0 e produzem uma aproximação x^{k+1} , utilizando x^k , mais próxima da solução exata x .

Escolher as matrizes S e T requer certos cuidados:

- i) S deve ser uma matriz simples e inverssível pois, o processo iterativo tem a forma

$$x^{k+1} = S^{-1}Tx^k + b.$$

A escolha de S implica na simplicidade do método pois, embora como veremos, não haver necessidade de *inverter* a matriz explicitamente, a expressão do processo iterativo deve ser bem simples no sentido da varredura das componentes de x^k .

- ii) A sequência de valores x^k deve convergir para a solução do sistema linear $x = A^{-1}b$. Para isso ocorrer, a sequência dos erros: $e^k = x^k - x$ deve convergir para o vetor nulo.

2.1 Os métodos de Jacobi e Gauss-Seidel

Costuma-se particionar a matriz A em matrizes triangular inferior, diagonal e triangular superior, cujos nomes são, respectivamente, L , D e U onde, $A = L + D + U$.

Em seguida, descreveremos os métodos de Jacobi, Gauss Seidel e SOR. Para facilitar o entendimento, vamos começar com o exemplo que se segue. Sejam

$$A = \begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix} \quad \text{e} \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix},$$

e tomamos a aproximação inicial $x^0 = (0, 0)^t$.

2.1.1 O método de Jacobi

No método de Jacobi

$$S = D \quad \text{e} \quad T = -(L + U).$$

Então teremos que

$$S = D = \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \quad \text{e} \quad T = -(L + U) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

Assim a partir da fórmula $Sx^{k+1} = Tx^k + b$ temos:

$$\begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} \begin{pmatrix} x_1^{k+1} \\ x_2^{k+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1^k \\ x_2^k \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Disso seguem as fórmulas

$$x_1^{k+1} = (x_2^k + 1)/2,$$

$$x_2^{k+1} = (x_1^k + 1)/2.$$

É importante observar, como salientamos anteriormente, que a implementação dos métodos estacionários, consiste apenas na varredura nas componentes como mostra as fórmulas acima. Assim, não é necessário armazenar a matriz para processar os dados.

Portanto, se

$$x^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \text{ então, } x^1 = \begin{pmatrix} \frac{1}{2}(0+1) \\ \frac{1}{2}(0+1) \end{pmatrix} = \begin{pmatrix} 1/2 \\ 1/2 \end{pmatrix}.$$

A sequência de iterados fica:

$$x^2 = \begin{pmatrix} 3/4 \\ 3/4 \end{pmatrix}, \quad x^3 = \begin{pmatrix} 7/8 \\ 7/8 \end{pmatrix}, \quad x^4 = \begin{pmatrix} 15/16 \\ 15/16 \end{pmatrix},$$
$$x^5 = \begin{pmatrix} 31/32 \\ 31/32 \end{pmatrix}, \quad x^6 = \begin{pmatrix} 63/64 \\ 63/64 \end{pmatrix}.$$

Assim podemos observar a sequência *convergindo* para a solução exata $x = (1, 1)$.

Para justificar a convergência, vejamos o valor do raio espectral:

$$S^{-1}T = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & 0 \end{pmatrix}.$$

Logo,

$$\rho(S^{-1}T) = \frac{1}{2} < 1.$$

2.1.2 O método de Gauss-Seidel

No método de Gauss-Seidel

$$S = L + D \quad e \quad T = -U.$$

Usaremos o mesmo exemplo anterior para que possamos comparar a convergência dos métodos.

Neste caso temos

$$S = \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ -1 & 2 \end{pmatrix}$$

e

$$T = - \begin{pmatrix} 0 & -1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}.$$

Portanto, temos
$$\begin{pmatrix} 2 & 0 \\ -1 & 2 \end{pmatrix} \begin{pmatrix} x_1^{k+1} \\ x_2^{k+1} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_1^k \\ x_2^k \end{pmatrix} + \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

O que resulta nas equações:

$$x_1^{k+1} = (x_2^k + 1)/2,$$

$$x_2^{k+1} = (x_1^{k+1} + 1)/2.$$

Como $x^0 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$ temos, $x^1 = \begin{pmatrix} 1/2 \\ 3/4 \end{pmatrix}$, $x^2 = \begin{pmatrix} 7/8 \\ 15/16 \end{pmatrix}$, $x^3 = \begin{pmatrix} 31/32 \\ 63/64 \end{pmatrix}$,
 $x^4 = \begin{pmatrix} 127/128 \\ 255/256 \end{pmatrix}$, \dots , $x = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$ mais rapidamente.

Vejam os valores do raio espectral:

$$S^{-1}T = \begin{pmatrix} \frac{1}{2} & 0 \\ \frac{1}{4} & \frac{1}{2} \end{pmatrix} \cdot \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & \frac{1}{2} \\ 0 & \frac{1}{4} \end{pmatrix}.$$

Logo,

$$\rho(S^{-1}T) = \frac{1}{4} < 1.$$

Observando este exemplo, poderíamos ser levados a concluir que o método de Gauss-Seidel sempre é melhor que o método de Jacobi. De fato, isso acontece na maioria das aplicações práticas. Mas, este fato nem sempre é verdadeiro.

Considere a matriz:

$$A = \begin{pmatrix} 1 & 2 & -2 \\ 1 & 1 & 1 \\ 2 & 2 & 1 \end{pmatrix}.$$

Fazendo os cálculos do raio espectral veremos que a matriz de iteração dos erros resulta em:

$$\rho_J = 0 \quad \text{e} \quad \rho_{GS} = 2,$$

portanto, Jacobi converge e Gauss-Seidel não converge.

2.2 O método SOR

O método SOR é considerado uma aceleração do Gauss-Seidel através de um fator ω . A matriz A , continuará sendo dividida em triangular inferior L , diagonal D e triangular superior U , mas, o sistema linear será multiplicado por um fator de relaxação ω . Ou seja, $Ax = b$, vai ser escrito $\omega Ax = \omega b, \omega \in R$. Como $A = L + D + U$ então,

$$\begin{aligned}\omega(L + D + U) &= \omega L + \omega D + \omega U \\ \omega(L + D + U) &= \omega L + D - D + \omega D + \omega U \\ \omega(L + D + U) &= \underbrace{\omega L + D}_S - \underbrace{[(1 - \omega)D - \omega U]}_T\end{aligned}$$

Assim a equação $Sx^{k+1} = Tx^k + b$ fica

$$(\omega L + D)x^{k+1} = [(1 - \omega)D - \omega U]x^k + \omega b.$$

Note que quando $\omega = 1$ temos o método de Gauss-Seidel novamente.

Aplicando o método SOR para o mesmo problema dos exemplos anteriores podemos verificar uma evolução comparando com o método de Gauss-Seidel.

Para aplicar o método SOR calcula-se a matriz

$$(\omega L + D) = \omega \begin{pmatrix} 0 & 0 \\ -1 & 0 \end{pmatrix} + \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 2 & 0 \\ -\omega & 2 \end{pmatrix}$$

e a matriz

$$[(1 - \omega)D - \omega U] = (1 - \omega) \begin{pmatrix} 2 & 0 \\ 0 & 2 \end{pmatrix} - \omega \begin{pmatrix} 0 & -1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 2(1 - \omega) & \omega \\ 0 & 2(1 - \omega) \end{pmatrix},$$

portanto, podemos escrever

$$\begin{pmatrix} 2 & 0 \\ -\omega & 2 \end{pmatrix} x^{k+1} = \begin{pmatrix} 2(1 - \omega) & \omega \\ 0 & 2(1 - \omega) \end{pmatrix} x^k + \omega b.$$

Também devemos nos preocupar em encontrar o ω que faça o sistema convergir mais rapidamente. Isso consiste em tornar o maior autovalor de $S^{-1}T$ o menor possível. Inicialmente calculemos

$$S^{-1}T = \begin{pmatrix} 2 & 0 \\ -\omega & 2 \end{pmatrix}^{-1} \begin{pmatrix} 2(1 - \omega) & \omega \\ 0 & 2(1 - \omega) \end{pmatrix} = \begin{pmatrix} 1 - \omega & \omega \\ \frac{1}{2}(1 - \omega) & 1 - \omega + \frac{1}{4}\omega^2 \end{pmatrix}.$$

O produto dos autovalores de $S^{-1}T$ é igual ao determinante dessa matriz, ou seja, $\lambda_1\lambda_2 = \det(S^{-1}T) = (1 - \omega)^2$. O produto dos autovalores no caso de matrizes $n \times n$ sempre serão $(1 - \omega)^n$. Esse é o motivo porque nunca teremos $\omega \geq 2$, se isso acontecesse teríamos o produto dos autovalores da matriz $S^{-1}T$ muito grande, e nesse caso $|\lambda_i| < 1$ seria impossível, logo o método de iteração não seria convergente. Observamos que ω tem seu valor ótimo quando os autovalores são iguais (com valor $\omega - 1$), pois estes são inversamente proporcionais. Observando a última matriz acima e usando a propriedade do *traço* temos:

$$\lambda_1 + \lambda_2 = (\omega_{opt} - 1) + (\omega_{opt} - 1) = 2 - 2\omega_{opt} + \frac{1}{4}\omega_{opt}^2,$$

ou seja, ω_{opt} é solução da equação:

$$\frac{1}{4}\omega_{opt}^2 - 4\omega_{opt} + 4 = 0,$$

resultando em: $\omega_{opt} = 8 - 4\sqrt{3} \approx 1,07$ com $\lambda_1 = \lambda_2 \approx 1,07$.

Observamos que como $\omega \approx 1,07$, a convergência do SOR dobra em relação ao Gauss-Seidel.

Na prática, a escolha do parâmetro (ω_{opt}) é feita por aproximação, já que é impraticável calcular o raio espectral quando estamos apenas interessados em uma aproximação da solução de um sistema linear.

Vamos ver um exemplo de escolha prática do parâmetro para o método SOR.

Considere o sistema com os seguintes dados:

$$A = \begin{pmatrix} 1 & 2 & 1 & 0 & 6 \\ 0 & 4 & 3 & 1 & -5 \\ -3 & 8 & 1 & 0 & 0 \\ 0 & 0 & 5 & 2 & -1 \\ 1 & -1 & 0 & 0 & 7 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{pmatrix},$$

começando com o ponto inicial $x^0 = (0, 0, 0, 0, 0)$.

Assim, executamos o método para vários valores de ω , na busca de um valor que resulte em um menor número de iterações

No caso de $\omega = 1$, retornando ao método de Gauss-Seidel, tivemos 38 iterações. As outras iterações, de acordo com um ω definido aleatoriamente, dentro dos padrões obviamente, seguem conforme a tabela.

ω	0,1	0,2	0,4	0,6	0,8	0,9	0,95	0,98	1,1	1,2	1,5
iter	50	50	50	50	50	41	25	28	50	50	50

Podemos observar que uma aproximação boa para (ω_{opt}) é aproximadamente 0,95 com 25 iterações. Voltaremos a considerar novos casos no Capítulo 4.

2.3 Comparação dos métodos estacionários

Tendo visto os três métodos para o exemplo básico, podemos estabelecer a forma genérica dos mesmos:

- JACOBI

$$x_i^{k+1} = (b_i - \sum_{j=1}^n a_{ij}x_j^k)/a_{ii}$$

- GAUSS-SEIDEL

$$x_i^{k+1} = (b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^n a_{ij}x_j^k)/a_{ii}$$

- SOR

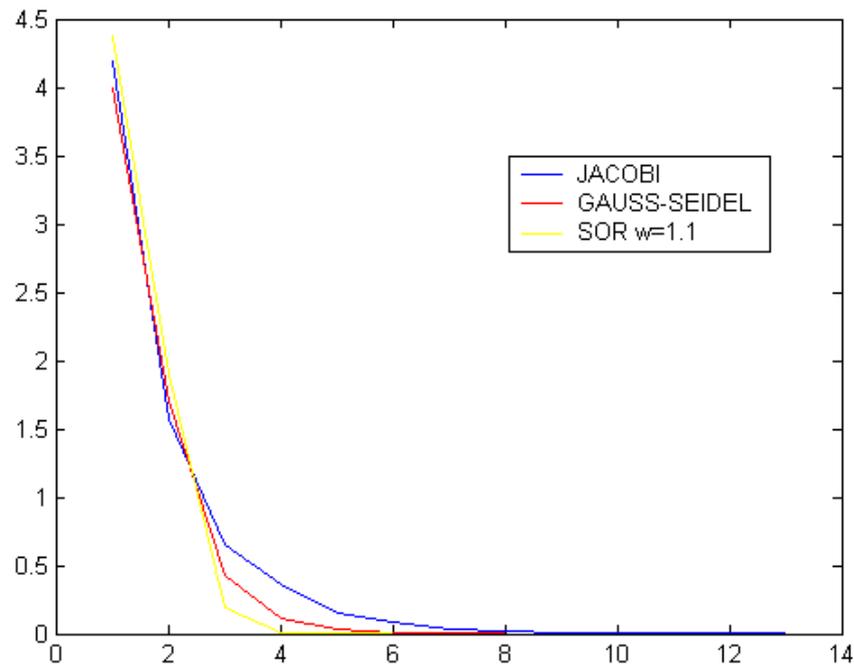
$$x_i^{k+1} = (1 - w)x_i^k + \frac{w}{a_{ii}}(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{k+1} - \sum_{j=i+1}^n a_{ij}x_j^k)/a_{ii}$$

Note que nesses métodos, não há necessidade do armazenamento da matriz. Assim somente há uma varreduras nas componentes do vetor de aproximações x , bastando apenas acessar a forma das equações do sistema. Isso favorece o uso em sistemas esparsos.

Vamos considerar um outro exemplo para comparação dos métodos. Considere o sistema linear e a aproximação inicial:

$$A = \begin{pmatrix} 7 & 1 & -1 & 2 \\ 1 & 8 & 0 & -2 \\ -1 & 0 & 4 & -1 \\ 2 & -2 & -1 & 6 \end{pmatrix}, \quad b = \begin{pmatrix} 3 \\ -5 \\ 4 \\ -3 \end{pmatrix}, \quad x_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}.$$

Os resultados, com precisão de 0,0001 estão apresentados no gráfico



Observe o desempenho melhor do método SOR seguido do Gauss-Seidel e Jacobi respectivamente. Este comportamento é padrão e em geral é assim que os métodos se comportam.

2.4 Alguns resultados teóricos

Vamos estabelecer alguns resultados suficientes para a convergência dos métodos de Jacobi e Gauss-Seidel.

Teorema 4 *Se A é estritamente diagonalmente dominante, então os métodos de Jacobi e Gauss-Seidel convergem.*

Prova: Sejam λ e x autovalor e autovetor da matriz de iteração de Jacobi. Assim,

$$-D^{-1}(L + U)x = \lambda x$$

que é o mesmo que

$$D(\lambda x) = -(L + U)x.$$

Suponha x_m a maior componente em módulo de x e que $|x_m| = 1$. Tomando a equação m deste sistema acima, temos:

$$\lambda A_{mm}x_m = -\sum_{j \neq m} A_{mj}x_j$$

tomando o módulo de ambos e lados e usando a desigualdade triangular temos:

$$|\lambda| |A_{mm}| |x_m| \leq \sum_{j \neq m} |A_{mj}| |x_j| < \sum_{j \neq m} |A_{mj}|,$$

finalmente

$$|\lambda| \leq \sum_{j \neq m} \frac{|A_{mj}|}{|A_{mm}|} < 1.$$

A demonstração para o método de Gauss-Seidel é ligeiramente diferente, mas segue a mesma idéia acima (veja em [10]).

Como uma abordagem mais geral vamos considerar uma condição central para a convergência dos métodos iterativos estacionários. Ela é baseada na equação de diferenças do erro do sistema linear, considerando matrizes convergentes e raio espectral.

Vamos agora voltar ao sub-índice k para designar um iteração arbitrária.

Condição sobre a sequência dos erros

Fazendo a diferença das equações:

$$Sx = Tx + b$$

$$Sx_{k+1} = Tx_k + b$$

temos,

$$S(x_{k+1} - x) = T(x_k - x)$$

ou,

$$e_{k+1} = (S^{-1}T)e_k.$$

Esta é uma equação de diferenças em e_k . Portanto, como vimos no Capítulo 1, a sequência dos erros convergirá para o vetor nulo quando todos os autovalores em módulo de $G = S^{-1}T$ for menor que a unidade. Ou, em termos do raio espectral, consideramos o seguinte resultado:

Teorema 5 *Seja $c = S^{-1}b$. Para qualquer x_0 , a sequência $\{x_k\}$ definida por*

$$x_{k+1} = Gx_k + c$$

converge para a solução $x = A^{-1}b$, se e somente se, $\rho(G) < 1$.

Veja em [2] ou [4].

Capítulo 3

O Método dos Gradientes Conjugados

Uma maneira de resolver o sistema linear $Ax = b$ quando a matriz A é simétrica positiva definida (SDP), é minimizar a quadrática

$$q(x) = \frac{1}{2}x^tAx - b^tx .$$

Isso porque $\nabla f(x) = Ax - b$, e portanto $\nabla f(x) = 0 \Leftrightarrow Ax = b$.

Assim quando A é SDP, $x = A^{-1}b$ é o minimizador da quadrática [12].

Neste capítulo, utilizaremos a notação mais usual para uma sequência de vetores gerados por um esquema iterativo. Contrariamente dos métodos estacionários, os métodos não-estacionários geram uma sequência de aproximação em todas as componentes em bloco. Assim denotaremos a k -ésima aproximação por x_k e não precisaremos referenciar as componentes.

O método que vamos tratar aqui tem a estrutura geral de um método de minimização baseado em:

$$x_{k+1} = x_k + \alpha_k p_k ,$$

em que x_0 é dado, e p_k é a direção (ou passo) que busca a solução e α_k é o parâmetro que controla o passo.

Seja A uma matriz simétrica positiva definida. Considere então a quadrática

$$q(x) = \frac{1}{2}x^tAx - b^tx .$$

O cálculo do α_k é solução do problema

$$\min q(x_k + \alpha p_k)$$

cuja solução é

$$\alpha_k = -\frac{\nabla f(x^k)^t p_k}{p_k^t A p_k}$$

(veja em [8]).

O método dos Gradientes Conjugados (GC) consiste num processo de minimização de um funcional quadrático em cada iteração. Antes vamos considerar o método dos Gradientes.

3.1 O método do Gradiente

No método do Gradiente a direção é definida por $p_k = -\nabla q(x_k)$. Assim o processo iterativo é:

$$x_{k+1} = x_k - \alpha_k \nabla q(x_k).$$

Agora $-\nabla q(x_k) = b - Ax_k = r_k$ temos

$$\alpha_k = \frac{r_k^t r_k}{r_k^t A r_k},$$

ou seja,

$$x_{k+1} = x_k + \alpha_k r_k.$$

3.2 O método dos Gradientes Conjugados

Observemos que nos métodos iterativos que estamos considerando, é possível obter uma forma de atualização do resíduo r_{k+1} :

$$r_{k+1} = b - A_{k+1} = b - A(x_k + \alpha_k p_k) = b - Ax_k - \alpha_k A p_k,$$

ou seja,

$$r_{k+1} = r_k - \alpha_k A p_k.$$

Este fato é importante para o teste de convergência do método.

A idéia do método GC é escolher as direções não ortogonais, fato que limita a convergência do método do Gradiente. Assim as direções são escolhidas de modo a ser *A-conjugadas*:

$$p_i^t A p_j = 0, \quad i \neq j.$$

Se for assim, surge uma propriedade importante:

O resíduo na iteração corrente é ortogonal às direções anteriores:

$$r_k^t p_i = 0, \quad i = 0, \dots, k-1$$

(veja em [8]).

Outra idéia intuitiva que caracteriza o método GC é que a direção corrente é uma combinação linear do resíduo com a direção anterior:

$$p_k = -r_k + \beta_k p_{k-1},$$

em que $\beta_k \in \mathbb{R}$ é obtido pela propriedade anterior (A-conjugacidade):

$$p_{k-1}^t A p_k = -p_{k-1}^t A r_k + \beta_k p_{k-1}^t A p_{k-1} = 0,$$

ou seja,

$$\beta_k = \frac{p_{k-1}^t A r_k}{p_{k-1}^t A p_{k-1}}.$$

Um resultado muito importante é a ortogonalidade dos resíduos.

Teorema 6 *Os resíduos gerados pelo método GC são ortogonais, ou seja,*

$$r_k^t r_i = 0, \quad i = 0, \dots, k-1.$$

(Teorema 5.5 pág. 234 [8])

Este fato tem duas consequências importantes. Primeiro permite estabelecer a convergência em n passos. De fato, nesse sentido o método GC pode ser considerado um método direto. Além disso, podemos obter uma fórmula mais simples para os parâmetros α_k e β_k :

$$\alpha_k = \frac{r_k^t r_k}{p_k^t A p_k},$$

$$\beta_k = \frac{r_{k+1}^t r_{k+1}}{r_k^t r_k}.$$

Assim podemos finalmente obter o algoritmo completo do método dos Gradientes Conjugados:

Algoritmo do método dos Gradientes Conjugados

Dado x_0 faça: $r_0 = b - Ax_0$, $p_0 = r_0$, $k = 0$.

Enquanto $r_k \neq 0$

$$\begin{aligned} \alpha_k &\leftarrow \frac{r_k^t r_k}{p_k^t A p_k} \\ x_{k+1} &\leftarrow x_k + \alpha_k p_k \\ r_{k+1} &\leftarrow r_k - \alpha_k A p_k \\ \beta_k &\leftarrow \frac{r_{k+1}^t r_{k+1}}{r_k^t r_k} \\ p_k &\leftarrow r_k + \beta_k p_{k-1} \\ k &\leftarrow k + 1 \end{aligned}$$

Observe que o maior custo de uma iteração do GC é a multiplicação da matriz-vetor $A p_k$. Em problemas grandes esta operação deve ser feita de modo eficiente.

O seguinte resultado estabelece o comportamento do método dos Gradientes Conjugados:

Teorema 7 *Se A é SDP, x_k é uma solução aproximada gerada pelo método CG e x a solução exata de um sistema linear, então:*

$$\|x_k - x\|_A \leq 2 \left(\frac{\sqrt{c(A)} - 1}{\sqrt{c(A)} + 1} \right) \|x_{k-1} - x\|_A.$$

em que, $\|v\|_A = \|v^T A v\|_A$ e $c(A) = \frac{\lambda_{max}}{\lambda_{min}}$ que é conhecido como *número de condição* de A ¹.

¹A definição precisa de $c(A)$ é dada pelo coeficiente entre o maior e o menor valor singular (veja em [4]) que no caso de matrizes SDP coincide com a definição dada acima.

Note que a expressão anterior implica também em:

$$\|x_k - x\|_A \leq 2 \left(\frac{\sqrt{c(A)} - 1}{\sqrt{c(A)} + 1} \right)^k \|x_0 - x\|_A$$

que nos fornece uma boa estimativa para o processo de convergência a partir de um ponto inicial x_0 arbitrário.

O número de condição $c(A)$ é vital para o comportamento do GC. Quanto menor o quociente entre os extremos dos autovalores mais rápida se dará a convergência.

Vamos considerar agora dois exemplos práticos para o método do GC.

Exemplo 3.1

$$A = \begin{pmatrix} 4 & -1 & -1 & 0 \\ -1 & 4 & 0 & -1 \\ -1 & 0 & 4 & -1 \\ 0 & -1 & -1 & 4 \end{pmatrix}, \quad b = \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \end{pmatrix},$$

com $x_0 = (0, 0, 0, 0)^t$.

Para este caso o GC converge com 2 iterações com precisão de 10^{-5} .

Ainda, neste caso,

$$\lambda_1 = 6, \quad \lambda_2 = 2 \quad \Rightarrow \quad c(A) = 3.$$

Exemplo 3.2

$$A = \begin{pmatrix} 1 & -1 & -1 & -1 & -1 \\ -1 & 2 & 0 & 0 & -0 \\ -1 & 0 & 3 & 1 & 1 \\ -1 & 0 & 1 & 4 & 2 \\ -1 & 0 & 1 & 2 & 5 \end{pmatrix}, \quad b = \begin{pmatrix} -3 \\ 1 \\ 4 \\ 6 \\ 7 \end{pmatrix},$$

com $x_0 = (0, 0, 0, 0)^t$.

Para este caso o GC converge com 5 iterações com precisão de 10^{-5} .

Ainda, neste caso

$$\lambda_1 \approx 0,0086, \quad \lambda_5 \approx 7,4875 \quad \Rightarrow \quad c(A) \approx 866.$$

Note que no segundo exemplo o número de condição é grande em relação ao primeiro exemplo. Por isso, o GC precisou de mais iterações para a convergência com a mesma precisão.

Os exemplos acima são conhecidos como matrizes de *Poisson* e *Moler* respectivamente. Voltaremos a elas em problemas esparsos de dimensão maior no próximo capítulo.

Capítulo 4

Experiências Numéricas

Neste capítulo vamos apresentar a implementação computacional dos métodos descritos no trabalho, utilizando o MATLAB.

Em primeiro lugar vamos considerar exemplos comparando o desempenho entre todos os métodos descritos.

4.1 Comparação entre os métodos

Exemplo 4.1 Observe o sistema linear:

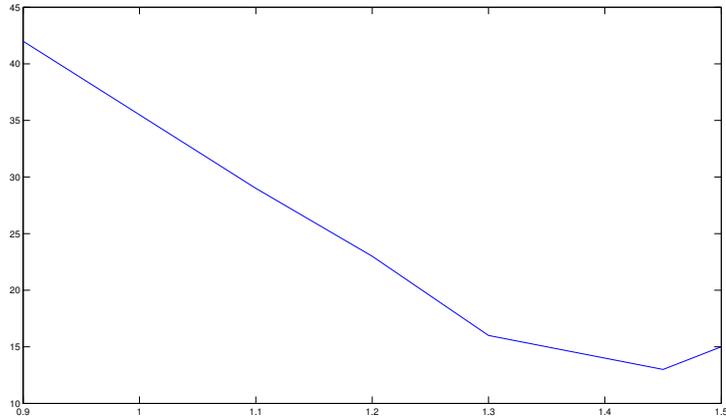
$$\begin{cases} x_1 + \frac{1}{2}x_2 = \frac{5}{21} \\ \frac{1}{2}x_1 + \frac{1}{3}x_2 = \frac{11}{84} \end{cases}$$

Ele possui solução $x = [\frac{1}{6}; \frac{1}{7}] \approx [0,1667; 0,1429]$. Agora analise a tabela obtida utilizando os métodos estudados e as respectivas respostas.

Método	Nº de Iterações	x_k
Jacobi	50	[0,1665;0,1427]
Gauss-Seidel	33	[0,1667;0,1429]
SOR ($\omega = 1,45$)	13	[0,1667;0,1429]
Gradiente Conjugado	2	[0,1667;0,1429]

Observe como os métodos trabalharam com esse sistema e como o método Gra-

diente Conjugado foi extremamente superior aos demais. A seguir verifique o gráfico do método SOR para encontrar o ω_{opt} , na abcissa o valor de ω e na ordenada o número de iterações.



Vamos apresentar agora uma matriz maior e esparça para a comparação dos métodos.

Exemplo 4.2 Observe o seguinte sistema linear:

$$\begin{bmatrix}
 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\
 -1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
 0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\
 -1 & 0 & 0 & 4 & -1 & 0 & 0 & 0 & 0 & 0 \\
 0 & -1 & 0 & -1 & 4 & -1 & -1 & 0 & 0 & 0 \\
 0 & 0 & -1 & 0 & -1 & 4 & 0 & -1 & 0 & 0 \\
 0 & 0 & 0 & 0 & -1 & 0 & 4 & -1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & -1 & -1 & 4 & -1 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 4 & -1 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 4
 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \end{bmatrix} = \begin{bmatrix} -110 \\ -30 \\ -40 \\ -110 \\ 0 \\ -15 \\ -90 \\ -25 \\ -55 \\ -65 \end{bmatrix} .$$

A solução obtida no MATLAB pelo método do Gradiente Conjugado é:

$$x = (-48,4945; -35,4155; -25,5648; -49,0669; -37,7035; -26,9867; -39,3946; -29,6866; -26,9184; -23,0702)^t .$$

Novamente observaremos na tabela a seguir o desempenho dos métodos iterativos estudados, utilizando $x_0 = (20, \dots, 20)$.

Método	Nº de Iterações
Jacobi	39
Gauss-Seidel	24
SOR ($\omega = 1,2$)	12
Gradiente Conjugado	5

Comparado com os demais, podemos perceber a robustez do método Gradiente Conjugado.

4.2 Desempenho do método dos Gradientes Conjugados

Exemplo 4.3 Matrizes de *Moler*.

Este conjunto de exemplos consiste de matrizes simétricas positivas definidas, porém com número de condição *grande*, também chamadas matrizes mal condicionadas. São casos em que o GC deve apresentar dificuldades de convergência.

caso 1: $n = 7$,

A =

1	-1	-1	-1	-1	-1	-1
-1	2	0	0	0	0	0
-1	0	3	1	1	1	1
-1	0	1	4	2	2	2
-1	0	1	2	5	3	3
-1	0	1	2	3	6	4
-1	0	1	2	3	4	7

Resultados:

iterações	resíduo	erro	no. cond.
7	$4,5 \times 10^{-9}$	$8,7 \times 10^{-9}$	$2,7 \times 10^4$

caso 2: $n = 10$,

A =

1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	2	0	0	0	0	0	0	0	0
-1	0	3	1	1	1	1	1	1	1
-1	0	1	4	2	2	2	2	2	2
-1	0	1	2	5	3	3	3	3	3
-1	0	1	2	3	6	4	4	4	4
-1	0	1	2	3	4	7	5	5	5
-1	0	1	2	3	4	5	8	6	6
-1	0	1	2	3	4	5	6	9	7
-1	0	1	2	3	4	5	6	7	10

Resultados:

iterações	resíduo	erro	no. cond.
11	$5,9 \times 10^{-12}$	$9,0 \times 10^{-11}$	$3,6 \times 10^6$

caso 3: $n = 15$

A=

1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1
-1	2	0	0	0	0	0	0	0	0	0	0	0	0	0
-1	0	3	1	1	1	1	1	1	1	1	1	1	1	1
-1	0	1	4	2	2	2	2	2	2	2	2	2	2	2
-1	0	1	2	5	3	3	3	3	3	3	3	3	3	3
-1	0	1	2	3	6	4	4	4	4	4	4	4	4	4
-1	0	1	2	3	4	7	5	5	5	5	5	5	5	5
-1	0	1	2	3	4	5	8	6	6	6	6	6	6	6
-1	0	1	2	3	4	5	6	9	7	7	7	7	7	7
-1	0	1	2	3	4	5	6	7	10	8	8	8	8	8
-1	0	1	2	3	4	5	6	7	8	11	9	9	9	9
-1	0	1	2	3	4	5	6	7	8	9	12	10	10	10
-1	0	1	2	3	4	5	6	7	8	9	10	13	11	11
-1	0	1	2	3	4	5	6	7	8	9	10	11	14	12
-1	0	1	2	3	4	5	6	7	8	9	10	11	12	15

Resultados:

iterações	resíduo	erro	no. cond.
19	$1,4 \times 10^{-14}$	$4,6 \times 10^{-8}$	$9,1 \times 10^9$

Exemplo 4.4 Matrizes de *Poisson*.

Este conjunto de exemplos consiste de matrizes simétricas grandes e esparsas, mas com bom condicionamento, o que favorece o uso do método GC.

As matrizes de Poisson são originadas da discretização de uma Equação Diferencial Parcial (EDP) [2]:

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

juntamente com as condições de contorno, $u(x, 0) = g(x)$ e $u(0, y) = h(x)$.

As funções f , g e h são dadas e dão origem ao lado direito do sistema.

O processo de discretização consiste em conduzir a equação diferencial em equações

algébricas. Este processo é essencial pois, em geral, estes problemas não possuem solução analítica.

Quanto mais precisão se exige para descrever o comportamento das EDP's, mais equações algébricas são necessárias, portanto, maior é a dimensão dos sistemas. Por outro lado, este fato aumenta o número de iterações e o esforço computacional.

Vamos considerar alguns casos:

caso 1: $n = 7$,

A =

4	-1	0	-1	0	0	0	0	0
-1	4	-1	0	-1	0	0	0	0
0	-1	4	0	0	-1	0	0	0
-1	0	0	4	-1	0	-1	0	0
0	-1	0	-1	4	-1	0	-1	0
0	0	-1	0	-1	4	0	0	-1
0	0	0	-1	0	0	4	-1	0
0	0	0	0	-1	0	-1	4	-1
0	0	0	0	0	-1	0	-1	4

Resultados:

iterações	resíduo	erro	no. cond.
3	$1,6 \times 10^{-16}$	$3,3 \times 10^{-16}$	2,2

caso 2: $n = 16$,

A =

4	-1	0	0	-1	0	0	0	0	0	0	0	0	0	0	0
-1	4	-1	0	0	-1	0	0	0	0	0	0	0	0	0	0
0	-1	4	-1	0	0	-1	0	0	0	0	0	0	0	0	0
0	0	-1	4	0	0	0	-1	0	0	0	0	0	0	0	0
-1	0	0	0	4	-1	0	0	-1	0	0	0	0	0	0	0
0	-1	0	0	-1	4	-1	0	0	-1	0	0	0	0	0	0
0	0	-1	0	0	-1	4	-1	0	0	-1	0	0	0	0	0
0	0	0	-1	0	0	-1	4	0	0	0	-1	0	0	0	0
0	0	0	0	-1	0	0	0	4	-1	0	0	-1	0	0	0
0	0	0	0	0	-1	0	0	-1	4	-1	0	0	-1	0	0
0	0	0	0	0	0	-1	0	0	-1	4	-1	0	0	-1	0
0	0	0	0	0	0	0	-1	0	0	-1	4	0	0	0	-1
0	0	0	0	0	0	0	0	-1	0	0	0	4	-1	0	0
0	0	0	0	0	0	0	0	0	-1	0	0	-1	4	-1	0
0	0	0	0	0	0	0	0	0	0	-1	0	0	-1	4	-1
0	0	0	0	0	0	0	0	0	0	0	-1	0	0	-1	4

Resultados:

iterações	resíduo	erro	no. cond.
3	$3,5 \times 10^{-16}$	$2,0 \times 10^{-15}$	3,6

Observamos nos dados resultantes que as matrizes de Moler resultam em sistemas que demandam um número crescente de iterações a medida que a dimensão aumenta. Este acréscimo tem origem no condicionamento do sistema que é considerado alto.

Nos sistemas de Poisson, o número de iterações não aumenta muito com a dimensão. Em geral, isso acontece com sistemas de discretização baseados nesta EDP pois, as matrizes tendem a possuir um número de condição controlado.

De qualquer maneira, o aumento da dimensão faz com que o produto matricial demande mais operações, o que em dimensão muito grande, como dezenas de milhares de variáveis [10], pode ser um problema.

Conclusões

Neste trabalho estudamos os métodos iterativos estacionários e o método dos Gradientes Conjugados para resolver sistemas lineares. Estes métodos são indicados quando os sistemas são grandes e esparsos, caso em que a fatoração da matriz pode ser impraticável.

Nos métodos estacionários é necessário apenas uma varredura, determinada pelas fórmulas das equações, nas componentes do vetor de aproximação. Para que sejam competitivos, esta vantagem deve compensar o grande número de iterações necessários para a convergência.

O método do Gradiente Conjugado é bem robusto. Converte em geral em poucas iterações. Esta vantagem deve compensar o custo por iteração, que basicamente consiste no produto matriz-vetor.

Como futuras abordagens para o tema, podemos citar o estudo dos métodos multimalhas [1] para acelerar os métodos não estacionários. Já para os métodos do tipo Gradientes Conjugados a idéia é investir em técnicas de condicionamento do sistema [4] e [10].

A ênfase do trabalho foi uma abordagem numérica, na qual utilizamos a programação no ambiente computacional do MATLAB. Esta experiência de programação foi muito importante já que não é usual em muitos cursos de graduação. Além disso, foi uma ótima oportunidade para apreender e desenvolver conceitos cujos quais ainda não havíamos trabalhado.

Reconhecemos neste trabalho uma quantidade importante de conhecimento adquirido que possibilita um boa oportunidade de continuidade de estudo e pesquisa em várias áreas de atuação pois, os modelos lineares são muito comuns em praticamente todas as

ciências.

A principal dificuldade encontrada para o desenvolvimento desse trabalho foi a distância física com o orientador, o que estendia por um pouco mais de tempo o sanar de dúvidas.

Notifica-se, no entanto, a importância deste trabalho para a formação, pelo tema abordado e pelas ferramentas escolhidas para se trabalhar.

Referências Bibliográficas

- [1] BRIGGS, W. L.; HENSON, VAN E.; MCCORMICK, S. F. *Multigrid tutorial*. 2nd Ed. SIAM, 2000.
- [2] BURDEN L. R. AND FAIRES J. D..*Análise Numérica*. Pioneira Thompson Learning, 2003.
- [3] DUFF I. S., ERISMAN A. M. AND REID J. K. *Direct Methods for Sparse Matrices*. Clarendon Press, Oxford, 1986.
- [4] GOLUB, G. H. AND C. F. VAN LOAN. *Matrix Computations*.-2nd ed.- Baltimore: Johns Hopkins University Press,1989.
- [5] HIGHAM, N. J.*Accuracy and Stability of Numerical Algorithms*. Philadelphia: Siam, 1996.
- [6] LAY, D. C. *Álgebra linear e suas aplicações*. Rio de Janeiro: LTC, 1999.
- [7] NOBLE, B. AND J. W. DANIEL. *Applied Linear Algebra*.-3rd ed.- Englewood Cliffs, N. J.: Prentice Hall, 1988.
- [8] NOCEDAL, J. AND WRIGHT, S. *Numerical Optimization*. 2nd ed. New York (N.Y.): Springer, 2006
- [9] PART-ENANDER, E. AND SJOBERG, A. *The Matlab handbook 7*, Addison Wesley Harlow, UK, 2005.
- [10] SAAD. Y.*Iterative methods for sparse linear systems*. , PWS Thompson Publishing Inc. 1996.
- [11] STEWART, G. W. *Introduction to Matrix Computations*. New Yor: Academic Press, Inc.,1973.

- [12] STRANG, G. *Linear Algebra e suas Aplicações* Cengage Learning, 2010.
- [13] WATKINS, D. *Fundamentals of Matrix Computations*. New York: John Wiley & Sons, 1991.