

UNIVERSIDADE FEDERAL DO AMAPÁ
PRÓ-REITORIA DE ENSINO E GRADUAÇÃO
CURSO DE LICENCIATURA PLENA EM MATEMÁTICA

*MÉTODOS DIRETOS E ITERATIVOS NA SOLUÇÃO DE SISTEMAS
DE EQUAÇÕES LINEARES*

MACAPÁ-AP

2011

ROGÉRIO BRASIL DA SILVA

***MÉTODOS DIRETOS E ITERATIVOS NA SOLUÇÃO DE SISTEMAS
DE EQUAÇÕES LINEARES***

Trabalho de conclusão de curso apresentado ao colegiado de Matemática da Universidade Federal do Amapá, como parte das exigências para a obtenção do título de Licenciatura Plena em Matemática, sob orientação do Prof. Dr. José Walter Cárdenas Sotil.

MACAPÁ-AP

2011

ROGÉRIO BRASIL DA SILVA

Métodos Diretos e Iterativos na Solução de Sistemas de Equações Lineares

Trabalho de Conclusão de Curso apresentado como pré-requisito para obtenção do título de Licenciatura Plena em Matemática da Universidade Federal do Amapá, submetida à aprovação da banca examinadora composta pelos seguintes membros:

Prof. Dr. José Walter Cárdenas Sotil

Prof. Dr. Guzmán Isla Chamilco

Prof. Dr. Erasmo Senger

Avaliado em: ____ / ____ / ____

MACAPÁ-AP

2011

*Espera no SENHOR e guarda seu
caminho! ...*

(Salmos 36, 34.)

Agradecimentos

Inicialmente agradeço a Deus por ter me proporcionado mais esta realização, dando-me força e coragem necessária para enfrentar obstáculos que ao longo da vida nos deparamos. Sou muito grato a Deus pois colocou verdadeiros anjos no meu caminho, anjos que me ofereceram um lar, dando-me oportunidade recomeçar.

Ao professor José Walter Cárdenas Sotil pela orientação, dedicação, paciência, amizade e incentivo durante todo o curso.

Agradeço a todos os professores do Colegiado de Matemática que contribuíram para minha formação acadêmica, em especial aos professores Guzmán Isla Chamilco e José Walter Cárdenas Sotil, pelos incentivos, pela dedicação, pelas oportunidades que me proporcionaram de chegar mais longe como no que diz respeito a um crescimento matemático.

Agradeço a minha mãe, minha avó e a meus irmãos que são a minha fonte de energia, dando-me força para que eu consiga ir sempre em busca de alcançar meus objetivos.

Vale a pena ressaltar minha gratidão aos colegas e amigos feitos ao longo de minha vida acadêmica amizades essas que pretendo levar por toda a vida. Agradeço a todos da Turma de Matemática 2007 da UNIFAP, pelas conversas, grupos de estudo, companherismo, brincadeiras(ping-pong, bola, counter-strike, etc.), festas.

Sou grato a UNIFAP, pois além de me proporcionar a formação acadêmica deu-me oportunidade de trabalho e alguns auxílios financeiros que serviram de subsídios durante esses quatro anos.

Resumo

Neste Trabalho de Conclusão de Curso, foram analisados métodos numéricos diretos e iterativos para a solução de sistemas lineares algébricos, os quais são de uso frequente para resolver sistemas de equações diferenciais. É definido o conceito de norma no espaço das matrizes, com os quais analisamos a convergência dos métodos diretos e iterativos. Apresentamos os métodos de Eliminação de Gauss, da Decomposição LU e Cholesky como métodos diretos, e os métodos de Jacobi, Gauss-Seidel e SOR como métodos iterativos. Para cada um destes métodos são descritos algoritmos numéricos para gerar a solução, e implementados no MATLAB. Os resultados mostram a eficiência dos algoritmos numéricos implementados neste Trabalho. Uma comparação entre os métodos iterativos mostram a alta performance do método de Gauss-Seidel e SOR. Métodos diretos são preferidos em matrizes de pequeno porte por ter menos restrições ao seu uso que os métodos iterativos. Para matrizes de grandes dimensões os métodos iterativos são preferidos por ter menos erros de arredondamento, precisar de menor memória RAM e armazenar menos dados no computador.

Palavras-Chave: Sistemas Lineares, Métodos Diretos, Métodos Iterativos, Normas de Matrizes.

Resumen

En este Trabajo de Fin de curso, se analizan métodos numéricos directos y iterativos para la solución de sistemas lineales algebraicos, los cuales son de uso frecuente en métodos numéricos para resolver sistemas de ecuaciones diferenciales. Es definido el concepto de norma en el espacio de las matrices, lo que permite realizar el análisis de la convergencia de los métodos directos y iterativos. Son estudiados los métodos de Eliminación de Gauss, Descomposición LU y Cholesky como métodos directos, y los métodos de Jacobi, Gauss-Seidel e SOR como métodos iterativos. Para cada uno de estos métodos se describen algoritmos numéricos para generar la solución, y implementados en el software numérico MATLAB. Los resultados muestran que los algoritmos numéricos implementados en este trabajo son eficientes. Una comparación entre los métodos iterativos muestran mejor desempeño de los métodos de Gauss-Seidel e SOR. Métodos directos son usados en matrices de pequeño porte debido a tener menos restricciones que los métodos iterativos. Para matrices de grandes dimensiones los métodos iterativos tienen preferencia por presentar un menor error de arredondamiento, precisar de menos memoria RAM y almacenar menos informaciones en el computador.

Palabras clave: Sistemas Lineales, Métodos Directos, Métodos Iterativos, Normas de Matrices.

Sumário

1	Introdução	5
2	Teoria de Matrizes	7
2.1	Matrizes	7
2.1.1	Operações com Matrizes	9
2.2	Sistemas Lineares	17
2.2.1	Solução de um Sistema Linear	21
2.2.2	Sistema Linear Homogêneo	22
2.2.3	Sistemas Equivalentes	23
3	Normas	25
3.1	Normas em \mathbb{R}^n	26
3.2	Normas Matriciais	28
3.3	Convergência em Norma	33
4	Métodos Numéricos	35
4.1	Métodos Diretos	35
4.1.1	Sistema Triangular Superior	35
4.1.2	Método de Eliminação de Gauss	36
4.1.3	Fatoração LU	42
4.1.4	Fatoração de Cholesky	44
4.2	Métodos Iterativos	48
4.2.1	Obtenção dos Métodos Iterativos	49
4.2.2	Critério de Convergência dos Métodos Iterativos	51
4.2.3	Estimativa de Erro na Iteração $x^{(k+1)}$	52
4.2.4	Método de Jacobi	53

4.2.5	Método de Gauss-Seidel	57
4.2.6	Métodos SOR	62
4.3	Comparação entre Jacobi, Gauss-Seidel e SOR	67

Lista de Tabelas

4.1	Jacobi	56
4.2	Gauss-Seidel	61
4.3	Métodos SOR com $\omega=0,9$	65
4.4	Jacobi X Gauss-Seidel X SOR	68
4.5	Métodos SOR com $\omega = 0,5$ e $\omega = 1,9$	68
4.6	Exemplo 4.9 comparação Jacobi X Gauss-seidel	69
4.7	Exemplo 4.9 SOR com $\omega = 1,2$	70

Capítulo 1

Introdução

A necessidade de resolver sistemas de equações lineares aparece numa grande quantidade de problemas científicos. Estima-se que em 75% dos problemas científicos a solução de um sistema linear de equações aparece em algum estágio da solução[5].

A título de exemplo, uma fonte substancial de sistema de equações, lineares ou não-lineares, conforme o caso, é a solução de equações diferenciais por via dos métodos de discretização, como diferenças finitas ou elementos finitos. Em geral, os sistemas que resultam da discretização das equações diferenciais são muito grandes e têm características de esparsidade (muitos elementos nulos nas matrizes) que auxiliam em sua solução numérica. Outras fontes de sistemas são problemas de ajuste de dados, de minimização de funções e problemas diversos.

Um sistema de equações $Ax = b$ pode-se resolver como $x = A^{-1}b$ desde que a inversa A^{-1} exista, entretanto o cálculo direto da inversa é dispendioso e sujeito a erros de arredondamento. Por isto, métodos mais eficientes são necessários para resolver um sistema de equações lineares.

Respeitando a distinção dos métodos, é comum catalogá-los em dois grupos. Os *Métodos Diretos* são aqueles que conduzem à solução exata a menos de erros de arredondamento produzidos pela máquina, após um número finito de passos. Os *Métodos Iterativos* são aqueles que se baseiam na construção de sequências de aproximações. Em um método iterativo, cada passo, os valores calculados anteriormente são usados para melhorar a aproximação. É claro que o método iterativo será útil se a sequência das aproximações construídas pelo o método convergir para a solução do sistema[4].

No capítulo 2 apresentamos o conceito de matrizes, operações com matrizes, inversas de

matrizes e o conceito de solução de sistema de equações lineares. No capítulo 3 definimos as normas de vetores e matrizes, assim como o conceito de convergência em norma de uma sequência de vetores e matrizes. No capítulo 4 apresentamos métodos diretos (Eliminação de Gauss, LU e Cholesky) e iterativos (Jacobi, Gauss-Seidel e SOR) na solução de sistemas de equações lineares, assim como algoritmos e simulações numéricas no MATLAB.

Capítulo 2

Teoria de Matrizes

Neste capítulo se apresentam os conceitos básicos da teoria de matrizes necessários para desenvolver métodos numéricos para resolver sistemas de equações lineares.

2.1 Matrizes

Definição 2.1. Dados dois números m e n naturais e não nulos, chama-se matriz m por n (indica-se $m \times n$) toda tabela A formada por números reais distribuídos em m linhas e n colunas. Em uma matriz qualquer A , cada elemento é indicado por a_{ij} . O índice i indica a linha e o índice j indica a coluna a quais os elementos pertence. Com a convenção de que as linhas sejam numeradas de cima para baixo (de 1 até m) e as colunas, da esquerda para a direita (de 1 até n) uma matriz $m \times n$ é representada por [6]:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix}$$

ou,

$$A = (a_{ij})_{m \times n} = \{a_{ij} \mid i = 1, \dots, m; \quad j = 1, \dots, n\}$$

Indicaremos por $M_{m \times n}(\mathbb{R})$ o conjunto das matrizes reais do tipo $m \times n$. Se $m = n$, ao invés de $M_{n \times n}(\mathbb{R})$, usaremos a notação $M_n(\mathbb{R})$. Cada matriz de $M_n(\mathbb{R})$ chama-se *matriz quadrada de ordem n* . Em contraposição, quando $m \neq n$, uma matriz do tipo $m \times n$ se diz *retangular*. Uma matriz 1×1 se identifica com o número real a_{11} [2].

Definição 2.2. Chama-se *diagonal principal* de uma matriz quadrada de ordem n o conjunto dos elementos que têm os dois índices iguais, isto é:

$$\{a_{ij} \mid i = j\} = \{a_{11}, a_{22}, a_{33}, \dots, a_{nn}\}.$$

Chama-se *diagonal secundária* de uma matriz quadrada de ordem n o conjunto dos elementos que têm soma dos índices igual a $n + 1$, isto é:

$$\{a_{ij} \mid i + j = n + 1\} = \{a_{1n}, a_{2,n-1}, a_{3,n-2}, \dots, a_{n1}\}.$$

Exemplo 2.1. A matriz

$$M = \begin{bmatrix} 1 & 3 & -2 \\ 4 & 6 & 1 \\ 9 & 5 & -8 \end{bmatrix}$$

é quadrada de ordem 3. Sua diagonal principal é $\{1, 6, -8\}$ e sua diagonal secundária é $\{9, 6, -2\}$.

Definimos a seguir matrizes que são comuns na teoria:

1. **Matriz Linha.** É toda matriz do tipo $1 \times n$, isto é, é uma matriz que tem uma única linha:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \end{bmatrix}$$

2. **Matriz Coluna.** É toda matriz do tipo $m \times 1$, isto é, é uma matriz que tem uma única coluna.

$$\begin{bmatrix} a_{m1} \\ a_{m2} \\ \vdots \\ a_{m1} \end{bmatrix}$$

3. **Matriz Nula.** É toda matriz que tem todos os elementos iguais a zero.
4. **Matriz Diagonal.** É toda matriz quadrada em que os elementos que não pertencem a diagonal principal são iguais a zero.
5. **Matriz Identidade.** Denotada por I_n é uma matriz quadrada de ordem n em que os elementos da diagonal principal são iguais a 1, e os demais elementos são iguais

a zero.

$$I_n = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

2.1.1 Operações com Matrizes

A seguir definimos operações com matrizes, as quais permitem formular equações algébricas em termos de matrizes.

Definição 2.3. [6] (**Igualdade de Matrizes**). Duas matrizes $A = (a_{ij})_{m \times n}$ e $B = (b_{ij})_{m \times n}$ são iguais quando

$$a_{ij} = b_{ij}, \quad \forall i = 1, \dots, m; \quad \forall j = 1, \dots, n$$

Isto significa que para serem iguais duas matrizes devem ser do mesmo tipo e apresentar todos os elementos correspondentes (elementos com índices iguais) iguais.

Exemplo 2.2. Se

$$A = \begin{bmatrix} 1 & 3 \\ 5 & -6 \end{bmatrix} \quad B = \begin{bmatrix} 1 & 3 \\ 5 & -6 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 3 \\ 5 & -8 \end{bmatrix},$$

então $A = B$ pois $a_{11} = b_{11}, a_{12} = b_{12}, a_{21} = b_{21}$ e $a_{22} = b_{22}$ e $A \neq C$ pois $a_{22} \neq c_{22}$

Definição 2.4. [6] (**Adição de Matrizes**) Dadas duas matrizes $A = (a_{ij})_{m \times n}$ e $B = (b_{ij})_{m \times n}$, chama-se *soma* $A + B$ a matriz $C = (c_{ij})_{m \times n}$ tal que $c_{ij} = a_{ij} + b_{ij}$, para todo i e todo j . Isto significa que a soma de duas matrizes A e B do tipo $m \times n$ é uma matriz C do mesmo tipo em que cada elemento é a soma dos elementos correspondentes em A e B .

Exemplo 2.3.
$$\begin{bmatrix} 2 & 3 \\ 6 & -6 \end{bmatrix} + \begin{bmatrix} 0 & 4 \\ 1 & 7 \end{bmatrix} = \begin{bmatrix} 2+0 & 3+4 \\ 6+1 & -6+7 \end{bmatrix} = \begin{bmatrix} 2 & 7 \\ 7 & 1 \end{bmatrix}$$

Propriedade 2.1. [6] Se $A, B, C \in M_{m \times n}(\mathbb{R})$, então:

1. Associatividade: $(A + B) + C = A + (B + C)$
2. Comutatividade: $A + B = B + A$
3. Elemento Neutro: Existe M tal que, $A + M = M + A = A$
4. Elemento Simétrico: Para toda matriz $A \exists A'$ tal que $A + A' = A' + A = M$.

Demonstração.

1. Fazendo $(A + B) + C = X$ e $A + (B + C) = Y$, temos:

$$x_{ij} = (a_{ij} + b_{ij}) + c_{ij} = a_{ij} + (b_{ij} + c_{ij}) = y_{ij}, \quad \forall i \text{ e } \forall j.$$

$$\text{Portanto, } (A + B) + C = A + (B + C)$$

2. Fazendo $A + B = X$ e $B + A = Y$, temos:

$$x_{ij} = a_{ij} + b_{ij} = b_{ij} + a_{ij} = y_{ij}, \quad \forall i \text{ e } \forall j.$$

$$\text{Portanto, } A + B = B + A$$

3. Impondo $A + M = A$, resulta:

$$a_{ij} + m_{ij} = a_{ij} \Rightarrow m_{ij} = 0 \Rightarrow M = 0$$

isto é, o elemento neutro é a matriz nula do tipo $m \times n$.

4. Impondo $A + A' = M = 0$, resulta:

$$a_{ij} + a'_{ij} = 0 \Rightarrow a'_{ij} = -a_{ij} \quad \forall i, \forall j$$

isto é, a simétrica da matriz A para a adição é a matriz A' de mesmo tipo que A , na qual cada elemento é simétrico do correspondente em A .

□

Definição 2.5. [6] (**Multiplicação por escalar.**) Dado um número α e uma matriz $A = (a_{ij})_{m \times n}$ chama-se **produto** de $\alpha.A$ a matriz $B = (b_{ij})_{m \times n}$ tal que $b_{ij} = \alpha.a_{ij}$, para todo i e todo j . Isto significa que multiplicar uma matriz A por um número α é construir uma matriz B formados pelos os elementos de A todos multiplicados por α .

Exemplo 2.4. $3 \cdot \begin{bmatrix} 3 & -5 & 7 \\ 4 & 6 & -1 \end{bmatrix} = \begin{bmatrix} 9 & -15 & 21 \\ 12 & 18 & -3 \end{bmatrix}$

Propriedade 2.2. Se $A, B \in M_{m \times n}(\mathbb{R})$ e $\alpha, \beta \in \mathbb{R}$, verifica-se que:

1. $\alpha \cdot (\beta \cdot A) = (\alpha \cdot \beta) \cdot A$
2. $\alpha \cdot (A + B) = \alpha \cdot A + \alpha \cdot B$
3. $(\alpha + \beta) \cdot A = \alpha \cdot A + \beta \cdot A$
4. $1 \cdot A = A$

Definição 2.6. [6] (**Produto de Matrizes**). Dadas duas matrizes $A = (a_{ij})_{m \times n}$ e $B = (b_{jk})_{n \times p}$, chama-se **produto** de AB a matriz $C = (c_{ik})_{m \times p}$ tal que

$$c_{ik} = a_{i1}b_{1k} + a_{i2}b_{2k} + \cdots + a_{in}b_{nk} = \sum_{j=1}^n a_{ij}b_{jk}, \quad \forall i = 1, \dots, m, \quad k = 1, \dots, p.$$

Exemplo 2.5. Dadas as matrizes

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \quad \text{e} \quad B = \begin{bmatrix} 7 \\ 8 \\ 9 \end{bmatrix} \quad \text{calcular } AB.$$

Sendo A do tipo 2×3 e B do tipo 3×1 , decorre que existe AB e é do tipo 2×1 .

Fazendo $AB = C$, devemos calcular c_{11} e c_{21} :

$$\begin{aligned} C &= \begin{bmatrix} c_{11} \\ c_{21} \end{bmatrix} = \begin{bmatrix} (1^\circ \text{ linha de } A \times 1^\circ \text{ coluna de } B) \\ (2^\circ \text{ linha de } A \times 1^\circ \text{ coluna de } B) \end{bmatrix} \\ &= \begin{bmatrix} \begin{pmatrix} 1 \times 7 \\ 2 \times 8 \\ 3 \times 9 \end{pmatrix} \\ \begin{pmatrix} 4 \times 7 \\ 5 \times 8 \\ 6 \times 9 \end{pmatrix} \end{bmatrix} = \begin{bmatrix} (7 + 16 + 27) \\ (28 + 40 + 54) \end{bmatrix} = \begin{bmatrix} 50 \\ 122 \end{bmatrix} \end{aligned}$$

Teorema 2.1. [6] Se $A = (a_{ij})_{m \times n}$, então $A.I_n = A$ e $I_m.A = A$.

Demonstração.

I) Sendo $I_n = (\delta_{ij})_{n \times n}$ e $B = AI_n = (b_{ij})_{n \times n}$, temos:

$$\begin{aligned} b_{ij} &= a_{i1}\delta_{1j} + a_{i2}\delta_{2j} + a_{i3}\delta_{3j} + \cdots + a_{ii}\delta_{ii} + \cdots + a_{in}\delta_{nj} \\ &= a_{i1} \cdot 0 + a_{i2} \cdot 0 + a_{i3} \cdot 0 + \cdots + a_{ii} \cdot 1 + \cdots + a_{in} \cdot 0 \\ &= a_{ii} \quad \forall i, j \end{aligned}$$

Logo, $A \cdot I_n = A$.

II) Analogamente.

□

Propriedade 2.3. [6]

A multiplicação de matrizes goza das seguintes propriedades:

1. Associatividade

$$(AB)C = A(BC) ,$$

para quaisquer que sejam as matrizes $A = (a_{ij})_{m \times n}$, $B = (b_{jk})_{n \times p}$ e $C = (c_{kl})_{p \times r}$

2. Distributividade à direita em relação à adição:

$$(A + B)C = AC + BC$$

para quaisquer que sejam as matrizes $A = (a_{ij})_{m \times n}$, $B = (b_{ij})_{m \times n}$ e $C = (c_{jk})_{n \times p}$

3. Distributividade à esquerda em relação à adição:

$$C(A + B) = CA + CB ,$$

para quaisquer que sejam as matrizes $A = (a_{ij})_{m \times n}$, $B = (b_{ij})_{m \times n}$ e $C = (c_{ki})_{p \times m}$

4. $(\alpha A)B = A(\alpha B) = \alpha(AB)$

para quaisquer que sejam as matrizes $A = (a_{ij})_{m \times n}$, $B = (b_{jk})_{n \times p}$ e $\alpha \in \mathbb{R}$

Demonstração. 1. Fazendo $D = AB = (d_{ik})_{m \times p}$, $E = (AB)C = (e_{il})_{m \times r}$ e $F = BC =$

$(f_{jl})_{n \times r}$, temos:

$$e_{il} = \sum_{k=1}^p d_{ik} \cdot c_{kl} = \sum_{k=1}^p \left(\sum_{j=1}^n a_{ij} \cdot b_{jk} \right) \cdot c_{kl}$$

$$\begin{aligned}
&= \sum_{k=1}^p \left(\sum_{j=1}^n a_{ij} \cdot b_{jk} \cdot c_{kl} \right) = \sum_{j=1}^n a_{ij} \cdot \left(\sum_{k=1}^p b_{jk} \cdot c_{kl} \right) \\
&= \sum_{j=1}^n a_{ij} \cdot f_{jl}
\end{aligned}$$

então, $(AB)C = A(BC)$.

2. Fazendo $D = (A + B)C = (d_{ik})_{m \times p}$, temos:

$$\begin{aligned}
d_{ik} &= \sum_{j=1}^n (a_{ij} + b_{ij}) \cdot c_{jk} = \sum_{j=1}^n (a_{ij} \cdot c_{jk} + b_{ij} \cdot c_{jk}) \\
&= \sum_{j=1}^n a_{ij} \cdot c_{jk} + \sum_{j=1}^n b_{ij} \cdot c_{jk}
\end{aligned}$$

então, $(A + B)C = AC + BC$.

3. Análoga a 2.

4. Fazendo $C = \alpha \cdot A = (c_{ij})_{m \times n}$, $D = \alpha \cdot B = (d_{jk})_{n \times p}$ e $E = AB = (e_{ik})_{m \times p}$, temos:

$$\begin{aligned}
(\alpha \cdot A)B &= CB = \sum_{j=1}^n c_{ij} \cdot b_{jk} = \sum_{j=1}^n (\alpha \cdot a_{ij}) \cdot b_{jk} = \alpha \cdot \sum_{j=1}^n a_{ij} \cdot b_{jk} = \alpha \cdot E = \alpha \cdot (AB) \\
A(\alpha \cdot B) &= AD = \sum_{j=1}^n a_{ij} \cdot d_{jk} = \sum_{j=1}^n a_{ij} \cdot (\alpha \cdot b_{jk}) = \alpha \cdot \sum_{j=1}^n a_{ij} \cdot b_{jk} = \alpha \cdot E = \alpha \cdot (AB)
\end{aligned}$$

então, $(\alpha \cdot A)B = A(\alpha \cdot B) = \alpha \cdot (AB)$

□

Observações

1. É importante notar que a multiplicação de matrizes não é comutativa, isto é, para duas matrizes quaisquer A e B é falso que $AB = BA$ necessariamente.
2. Quando A e B são tais que $AB = BA$, dizemos que A e B comutam. Notemos que uma condição necessária para A e B comutarem é que sejam quadradas e de mesma ordem.
3. É importante observar também que a implicação:

$$AB = 0 \Rightarrow A = 0 \text{ ou } B = 0$$

não é válida para as matrizes, isto é, é possível encontrar duas matrizes não nulas cujo produto é igual a matriz nula [6].

Exemplo 2.6.

$$\begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Definição 2.7. [6] (**Matriz Transposta**). Dada uma matriz $A = (a_{ij})_{m \times n}$, chama-se **transposta** de A a matriz $A^T = (a'_{ij})_{n \times m}$ tal que $a'_{ji} = a_{ij}$, para todo i e todo j . Isto significa que, por exemplo, $a'_{11}, a'_{21}, a'_{31}, \dots, a'_{n1}$ são respectivamente iguais a $a_{11}, a_{21}, a_{31}, \dots, a_{n1}$; Vale dizer que a 1º coluna de A^T é igual à 1º linha de A . Repetindo o raciocínio, chegaríamos à conclusão de que as colunas de A^T são ordenadamente iguais às linhas de A .

Exemplo 2.7.

$$A = \begin{bmatrix} 1 & 3 & 8 \\ -2 & 9 & -5 \end{bmatrix} \Rightarrow A^T = \begin{bmatrix} 1 & -2 \\ 3 & 9 \\ 8 & -5 \end{bmatrix}$$

Propriedade 2.4. [6] A matriz transposta apresenta as seguintes propriedades:

1. $(A^T)^T = A$ para toda matriz $A = (a_{ij})_{m \times n}$;
2. Se $A = (a_{ij})_{m \times n}$ e $B = (b_{ij})_{m \times n}$, então $(A + B)^T = A^T + B^T$
3. Se $A = (a_{ij})_{m \times n}$ e $\alpha \in \mathbb{R}$, então $(\alpha.A)^T = \alpha.A^T$
4. Se $A = (a_{ij})_{m \times n}$ e $B = (b_{jk})_{n \times p}$, então $(AB)^T = B^T A^T$

Demonstração.

1. Fazendo $(A^T)^T = (a''_{ij})_{m \times n}$, resulta:

$$a''_{ij} = a'_{j1} = a_{ij} \quad \forall i, j.$$

2. Fazendo $A + B = C = (c_{ij})_{m \times n}$ e $(A + B)^T = C^T = (c'_{ji})_{n \times m}$, temos:

$$(A + B)^T = C^T = c'_{ji} = c_{ij} = a_{ij} + b_{ij} = a'_{ji} + b'_{ji} = A^T + B^T \quad \forall i, j.$$

Portanto, $(A + B)^T = A^T + B^T$

3. Fazendo $(\alpha.A)^T = (a''_{ji})_{n \times m}$, resulta:

$$a''_{ji} = \alpha.a_{ij} = \alpha.a'_{ji} \quad \forall i, j.$$

4. Fazendo $AB = C = (c_{ik})_{m \times p}$ e $(AB)^T = C^T = (c'_{ki})_{p \times m}$, resulta:

$$(AB)^T = C^T = c'_{ki} = c_{ik} = \sum_{j=1}^n a_{ij} \cdot b_{jk} = \sum_{j=1}^n b_{jk} \cdot a_{ij} = \sum_{j=1}^n b'_{kj} \cdot a'_{ji} = B^T A^T.$$

Portanto, $(AB)^T = B^T A^T$.

□

Definição 2.8. [6] (**Matriz Simétrica**) Chama-se **matriz simétrica** toda matriz quadrada A , de ordem n , tal que $A^T = A$.

Decorre da definição que, se $A = (a_{ij})$ é uma matriz simétrica, temos:

$$a_{ij} = a_{ji}; \text{ para todo } i, j \in \{1, 2, 3, \dots, n\}$$

isto é, os elementos simetricamente dispostos em relação a diagonal principal são iguais.

Exemplo 2.8. São simétricas as matrizes:

$$\begin{bmatrix} a & b \\ b & d \end{bmatrix}, \quad \begin{bmatrix} a & b & c \\ b & d & e \\ c & e & f \end{bmatrix}, \quad \begin{bmatrix} a & b & c & d \\ b & e & f & g \\ c & f & h & i \\ d & g & i & j \end{bmatrix}$$

Definição 2.9. [6] (**Matriz Anti-Simétrica**). Chama-se **matriz anti-simétrica** a toda matriz quadrada A de ordem n , tal que $A^T = -A$.

Decorre da definição que, se $A = (a_{ij})$ é uma matriz anti-simétrica, temos:

$$a_{ij} = -a_{ji}; \text{ para todo } i, j \in \{1, 2, 3, \dots, n\},$$

isto é, os elementos simetricamente dispostos em relação a diagonal principal são opostos.

Exemplo 2.9. São anti-simétrica as matrizes:

$$\begin{bmatrix} 0 & a \\ -a & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & a & b \\ -a & 0 & c \\ -b & -c & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & a & b & c \\ -a & 0 & d & e \\ -b & -d & 0 & f \\ -c & -e & -f & 0 \end{bmatrix}.$$

Definição 2.10. [6] (**Matriz inversível**). Uma matriz quadrada A é chamada de **invertível** ou **não singular** se existe uma matriz B tal que $AB = BA = I$, onde I é a matriz identidade. Se A não é inversível, dizemos que A é uma matriz singular. Chamaremos a tal matriz B de **inversa** de A e a denotamos por A^{-1} .

Teorema 2.2. [6] *Se A é inversível, então é única a matriz A^{-1} tal que $AA^{-1} = A^{-1}A = I$*

Demonstração. Suponha que exista as matrizes A_1^{-1} e A_2^{-1} tais que:

$$AA_1^{-1} = A_1^{-1}A = I$$

$$AA_2^{-1} = A_2^{-1}A = I$$

então,

$$A_1^{-1} = A_1^{-1}I = A_1^{-1}(AA_2^{-1}) = (A_1^{-1}A)A_2^{-1} = IA_2^{-1} = A_2^{-1}$$

Portanto, a inversa de uma matriz é única.

□

Exemplo 2.10. A matriz $A = \begin{bmatrix} 1 & 2 & 7 \\ 0 & 3 & 1 \\ 0 & 5 & 2 \end{bmatrix}$ é inversível e $A^{-1} = \begin{bmatrix} 1 & 31 & -19 \\ 0 & 2 & -1 \\ 0 & -5 & 3 \end{bmatrix}$ pois:

$$AA^{-1} = \begin{bmatrix} 1 & 2 & 7 \\ 0 & 3 & 1 \\ 0 & 5 & 2 \end{bmatrix} \begin{bmatrix} 1 & 31 & -19 \\ 0 & 2 & -1 \\ 0 & -5 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I_3$$

$$A^{-1}A = \begin{bmatrix} 1 & 31 & -19 \\ 0 & 2 & -1 \\ 0 & -5 & 3 \end{bmatrix} \begin{bmatrix} 1 & 2 & 7 \\ 0 & 3 & 1 \\ 0 & 5 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = I_3$$

Uma classe importante de matrizes em sistemas de equações são as matrizes diagonalmente dominantes, as quais permitem por exemplo realizar a eliminação gaussiana sem necessidade de pivotamento. Estas matrizes se definem a seguir.

Definição 2.11. [7] Uma matriz A é **diagonalmente dominante** se seus elementos

verificam:

$$|a_{ii}| \geq \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \quad , \quad i = 1, 2, \dots, n \quad (2.1)$$

e *estritamente diagonalmente dominante* se

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| \quad , \quad i = 1, 2, \dots, n, \quad (2.2)$$

isto é, em cada linha, o elemento da diagonal é maior em módulo que a soma dos módulos dos outros elementos.

Um outro tipo importante de matrizes que aparecem frequentemente nas aplicações são as simétrica e positiva definida.

Definição 2.12. [7] Uma matriz é simétrica e positiva definida se

$$A = A^T \quad \text{e} \quad x^T A x > 0, \text{ para todo vetor } x \neq 0$$

Exemplo 2.11. Consideremos a matriz $A = \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$ é o vetor $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \neq 0$. Logo,

$$\begin{aligned} x^T A x &= \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= \begin{bmatrix} 2x_1 + x_2 & x_1 + 2x_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \\ &= 2x_1^2 + x_1x_2 + x_2x_1 + 2x_2^2 > 0 \\ &= 2x_1^2 + 2x_1x_2 + 2x_2^2 = x_1^2 + 2x_1x_2 + x_2^2 + x_1^2 + x_2^2 > 0 \\ &= (x_1 + x_2)^2 + x_1^2 + x_2^2 > 0. \end{aligned}$$

Portanto, A é positiva definida

2.2 Sistemas Lineares

A Resolução de sistemas lineares surge em diversas áreas do conhecimento. O caso geral em que o sistema linear envolve m equações com n incógnitas, o sistema pode apresentar

uma única solução, infinitas soluções ou não admitir solução. Este tipo de problema é tratado na Álgebra Linear usando o processo de escalonamento[3]. Vamos analisar esquemas numéricos para soluções de sistemas lineares de n equações com n incógnitas, isto é

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n = b_3 \\ \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \\ a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \cdots + a_{mn}x_n = b_n \end{cases}$$

onde a_{ij} são os coeficientes, x_j são as incógnitas e os b_j são as constantes. Este sistema pode ser escrito na forma matricial $Ax = b$ com $A \in M_n(\mathbb{R})$ e $x, b \in M_{n \times 1}\mathbb{R}$. Analisaremos duas classes de esquemas numéricos: Métodos Diretos e Métodos Iterativos.

Definição 2.13. [6] (**Equação Linear**) Chamamos de equação linear, nas desconhecidas x_1, x_2, \dots, x_n , toda equação do tipo

$$a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b.$$

Os números $a_{11}, a_{12}, a_{13}, \dots, a_{1n}$, todos reais, são chamados coeficientes e b , também real, é o termo independente da equação.

Exemplo 2.12.

$$1^\circ) 3x_1 + 4x_2 - 5x_3 - x_4 = 5$$

$$2^\circ) 2x_1 - x_2 - x_3 = 0$$

$$3^\circ) 0x_1 + 0x_2 - 0x_3 + 0x_4 = 0$$

Observamos que não são lineares as equações:

$$1^\circ) 2x_1^2 + 4x_2 + x_3 = 0$$

$$2^\circ) 2x_1x_2 + x_3 + x_4 = 3$$

$$3^\circ) x_1 + \sqrt{x_2} - x_3 = 4$$

Definição 2.14. [6] (**Solução de uma Equação Linear**) Dizemos que a sequência ou *n-upla* ordenada de números reais $(\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n)$ é uma solução da equação linear $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n = b$, se

$$a_{11}\alpha_1 + a_{12}\alpha_2 + a_{13}\alpha_3 + \dots + a_{1n}\alpha_n = b \text{ for uma sentença verdadeira.}$$

Exemplo 2.13.

1. Seja a equação linear

$$2x_1 + 3x_2 - x_3 + x_4 = 3$$

a sequência $(1, 2, 3, -2)$ é solução, pois $2.(1) + 3.(2) - (3) + (-2) = 3$ é sentença verdadeira, porém a sequência $(1, 1, 2, 1)$ não é solução, pois $2.(1) + 3.(1) - (2) + (1) = 3$ é sentença falsa.

2. Seja a equação linear

$$0x + 0y + 0z = 0$$

é fácil observar que qualquer tripla ordenada $(\alpha_1, \alpha_2, \alpha_3)$ é solução da equação.

3. Seja a equação linear

$$0x + 0y + 0z + 0t = 2$$

é fácil observar que qualquer quádrupla ordenada $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ não satisfaz a equação, pois

$$0\alpha_1 + 0\alpha_2 + 0\alpha_3 + 0\alpha_4 = 2$$

é sentença falsa $\forall \alpha_1, \alpha_2, \alpha_3, \alpha_4$.

Definição 2.15. [6] (**Sistema Linear.**) Sistema linear é um conjunto de $m(m \geq 1)$ equações lineares, nas incógnitas $x_1, x_2, x_3, \dots, x_n$. Assim, o sistema

$$S \left\{ \begin{array}{l} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n = b_3 \\ \vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \\ a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \cdots + a_{mn}x_n = b_n \end{array} \right.$$

é linear.

Lembrando a definição de produto de matrizes, notemos que o sistema linear S pode ser escrito na forma matricial .

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}$$

Exemplo 2.14.

1. O sistema linear

$$S_1 \left\{ \begin{array}{l} 2x + 3y = 4 \\ x - y = 2 \end{array} \right.$$

pode ser escrito na forma matricial

$$\begin{bmatrix} 2 & 3 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

2. O sistema linear

$$S_2 \left\{ \begin{array}{l} 3x + y - z = 4 \\ 2x + 5y + 7z = 0 \end{array} \right.$$

pode ser escrito na forma matricial

$$\begin{bmatrix} 3 & 1 & -1 \\ 2 & 5 & 7 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 4 \\ 0 \end{bmatrix}$$

2.2.1 Solução de um Sistema Linear

Dizemos que a sequência ou n -upla ordenada de números reais $(\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n)$ é uma solução de um sistema linear S , se for solução de todas as equações de S , isto é,

$$\begin{array}{cccccccc} a_{11}\alpha_1 + & a_{12}\alpha_2 + & a_{13}\alpha_3 + & \cdots + & a_{1n}\alpha_n = & b_1 & (\text{sentença verdadeira}) \\ a_{21}\alpha_1 + & a_{22}\alpha_2 + & a_{23}\alpha_3 + & \cdots + & a_{2n}\alpha_n = & b_2 & (\text{sentença verdadeira}) \\ a_{31}\alpha_1 + & a_{32}\alpha_2 + & a_{33}\alpha_3 + & \cdots + & a_{3n}\alpha_n = & b_3 & (\text{sentença verdadeira}) \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \\ a_{m1}\alpha_1 + & a_{m2}\alpha_2 + & a_{m3}\alpha_3 + & \cdots + & a_{mn}\alpha_n = & b_n & (\text{sentença verdadeira}) \end{array}$$

Exemplo 2.15.

$$1. \text{ O sistema } S \begin{cases} x + y + z = 6 \\ 2x + y - z = 1 \\ 3x - y + z = 4 \end{cases}$$

admite como solução a tripla ordenada $(1, 2, 3)$ pois

$$\begin{array}{l} 1 + 2 + 3 = 6 \quad (\text{sentença verdadeira}) \\ 2 \cdot (1) + (2) - (3) = 1 \quad (\text{sentença verdadeira}) \\ 3 \cdot (1) - (2) + (3) = 4 \quad (\text{sentença verdadeira}) \end{array}$$

S não admite, porém, como solução a tripla ordenada $(-5, 11, 0)$ pois

$$\begin{array}{l} (-5) + (11) + (0) = 6 \quad (\text{sentença verdadeira}) \\ 2 \cdot (-5) + (11) - (0) = 1 \quad (\text{sentença verdadeira}) \\ 3 \cdot (-5) - (11) + (0) = 4 \quad (\text{sentença falsa}) \end{array}$$

2. O sistema linear

$$S \begin{cases} x + 2y + 3z = 5 \\ x - y + 4z = 1 \\ 0x - 0y + 0z = 6 \end{cases}$$

não admite solução, pois a última equação não é satisfeita por nenhuma tripla ordenada $(\alpha_1, \alpha_2, 0\alpha_3)$.

Definição 2.16. [6](**Sistema possível**) Se um sistema linear S tiver pelo menos uma solução, diremos que ele é possível; caso não tenha nenhuma solução, diremos que S é impossível.

Definição 2.17. (Sistema impossível) Os sistemas que têm solução única são chamados possíveis e determinados. Já Os sistemas que têm infinitas soluções são chamados possíveis e indeterminados.

2.2.2 Sistema Linear Homogêneo

Definição 2.18. [6] Chamamos de sistema linear homogêneo todo aquele em que o termo independente de todas as equações é igual a zero, isto é:

$$S \begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = 0 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = 0 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n = 0 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + a_{m3}x_3 + \cdots + a_{mn}x_n = 0 \end{cases}$$

Exemplo 2.16.

$$S_1 \begin{cases} x + y + z = 0 \\ 2x - y + z = 0 \end{cases}, \quad S_2 \begin{cases} 3x + 4y + 3z = 0 \\ x - y + 4z = 0 \\ 5x - 7y + 9z = 0 \end{cases}$$

Observe que um sistema linear homogêneo admite sempre como solução a sequência $(\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n)$ em que $\alpha_i = 0$, $\forall i = 1, 2, \dots, n$, chamada de *solução nula*, *trivial*, ou *imprópria*. Portanto um sistema linear homogêneo é sempre possível. Se o sistema linear homogêneo for *determinado* apresentará apenas uma solução (a nula), e se for *indeterminado* apresentará além da solução nula, outras soluções não nulas também chamadas *soluções próprias*.

Definição 2.19. [6](**Matrizes de um Sistema**) Dado um sistema linear S de m equações e n incógnitas, consideremos as matrizes:

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} \end{bmatrix}$$

e

$$B = \begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} & b_1 \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} & b_2 \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} & b_3 \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \cdots & a_{mn} & b_m \end{bmatrix}$$

A é chamada *matriz incompleta do sistema* e B , *matriz completa*

Notemos que B foi obtida a partir de A , acrescentando-se a esta a coluna formada pelos termos independentes das equações do sistema.

Exemplo 2.17.

$$\begin{aligned} S_1 \begin{cases} 2x + y &= 3 \\ x - y &= 4 \end{cases} & A = \begin{bmatrix} 2 & 1 \\ 1 & -1 \end{bmatrix} \text{ e } B = \begin{bmatrix} 2 & 1 & 3 \\ 1 & -1 & 4 \end{bmatrix} \\ S_2 \begin{cases} 3x - y + z &= 1 \\ 4x + y &= 7 \end{cases} & A = \begin{bmatrix} 3 & -1 & 1 \\ 4 & 1 & 0 \end{bmatrix} \text{ e } B = \begin{bmatrix} 3 & -1 & 1 & 1 \\ 4 & 1 & 0 & 7 \end{bmatrix} \end{aligned}$$

2.2.3 Sistemas Equivalentes

Definição 2.20. [6] Dizemos que dois sistemas lineares $Ax = b$ e $\overline{A}x = \overline{b}$ são equivalentes, se toda solução de $Ax = b$ for solução de $\overline{A}x = \overline{b}$ e toda solução de $\overline{A}x = \overline{b}$ for solução de $Ax = b$

Exemplo 2.18.

$$S_1 \begin{cases} x + 2y &= 3 \\ 2x + y &= 1 \end{cases}$$

$$S_2 \begin{cases} x + 2y &= 3 \\ -3y &= -5 \end{cases}$$

S_1 e S_2 são equivalentes, pois ambos são determinados ($D \neq 0$, nos dois) e admitem como solução $(-\frac{1}{3}; \frac{5}{3})$

Capítulo 3

Normas

Na teoria de vetores é definida a norma do vetor $\vec{v} = (v_1, v_2)$ como $\sqrt{v_1^2 + v_2^2}$, como uma medida do módulo do vetor. O conceito de norma se estende para vetores em \mathbb{R}^n como $\|\vec{v}\| = \sqrt{v_1^2 + \dots + v_n^2}$. O conceito de norma pode ser aplicado para análise de erros na teoria da aproximação, no sentido de analisar o erro entre o valor aproximado v_{ap} e o valor exato v_{ex} : $\|v_{ap} - v_{ex}\|$.

A norma apresentada é a norma euclidiana, no sentido de ser calculada com base no teorema de Pitágoras. Entretanto este conceito pode ser generalizado para qualquer espaço vetorial desde que verifique um conjunto de propriedades. Como as matrizes da mesma ordem com as operações de adição e multiplicação por um escalar (definidas no capítulo 2) formam um espaço vetorial, pode-se aplicar a norma de matrizes para avaliar o erro na teoria da aproximação.

Definição 3.1. [7] Se V é um espaço vetorial, uma norma é uma função $\|\cdot\| : V \rightarrow R$ que verifica as seguintes propriedades:

- i) $\|x\| > 0$, se $x \neq 0$, $x \in V$
- ii) $\|\lambda x\| = |\lambda| \cdot \|x\|$, se $\lambda \in R$, $x \in V$
- iii) $\|x + y\| \leq \|x\| + \|y\|$, se $x, y \in V$ (desigualdade triangular)

3.1 Normas em \mathbb{R}^n

De especial interesse, quando $V = \mathbb{R}^n$, são as normas l_p definidas por:

$$\|x\|_p = \left(\sum_{i=1}^n |x_i| \right)^{1/p}, \quad p \geq 1$$

$$\|x\|_\infty = \max_{1 \leq i \leq n} \{|x_i|\}$$

Propriedade 3.1. A função $\|\cdot\|_1 : \mathbb{R}^n \rightarrow \mathbb{R}$ definida por:

$$\|x\|_1 = |x_1| + |x_2| + \cdots + |x_n| = \sum_{i=1}^n |x_i|,$$

é uma norma em \mathbb{R}^n , denominada Norma l_1 ou Norma da Soma.

Demonstração. Considere $x, y \in \mathbb{R}^n$ e $\lambda \in \mathbb{R}$.

i) Se $x = (x_1, x_2, \dots, x_n) \neq 0$, então $\exists x_k \neq 0 \implies |x_k| > 0$. Logo,

$$\|x\|_1 = \sum_{i=1}^n |x_i| \geq |x_k| > 0$$

ii) $\|\lambda x\|_1 = \|\lambda(x_1, x_2, \dots, x_n)\|_1 = \|(\lambda x_1, \lambda x_2, \dots, \lambda x_n)\|_1$

$$= \sum_{i=1}^n |\lambda x_i| = \sum_{i=1}^n |\lambda| \cdot |x_i| = |\lambda| \sum_{i=1}^n |x_i| = |\lambda| \|x\|_1$$

iii) $\|x + y\|_1 = \|(x_1, x_2, \dots, x_n) + (y_1, y_2, \dots, y_n)\|_1$

$$= \|(x_1 + y_1, x_2 + y_2, \dots, x_n + y_n)\|_1$$

$$= \sum_{i=1}^n |x_i + y_i| \leq \sum_{i=1}^n (|x_i| + |y_i|) \leq \sum_{i=1}^n |x_i| + \sum_{i=1}^n |y_i|$$

$$\text{Daí,} \quad \|x + y\|_1 \leq \|x\|_1 + \|y\|_1.$$

De i), ii) e iii) e da Definição 3.1 temos que $\|\cdot\|_1$ é uma norma em \mathbb{R}^n . □

Propriedade 3.2. A função $\|\cdot\|_2 : \mathbb{R}^n \rightarrow \mathbb{R}$ definida por:

$$\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2} = \sqrt{\sum_{i=1}^n x_i^2},$$

é uma norma em \mathbb{R}^n , denominada Norma l_2 ou Norma Euclideana.

Demonstração. Considere $x, y \in \mathbb{R}^n$ e $\lambda \in \mathbb{R}$.

i) Se $x = (x_1, x_2, \dots, x_n) \neq 0$, então, $\exists i \in \{1, \dots, n\}$ tal que $x_i \neq 0$. Logo,

$$\|x\|_2 = \sqrt{x_1^2 + x_2^2 + \dots + \underbrace{x_i^2}_{\neq 0} + \dots + x_n^2} \geq \sqrt{x_i^2} = |x_i| > 0$$

ii) $\|\lambda x\|_2 = \|\lambda(x_1, x_2, \dots, x_n)\|_2 = \|(\lambda x_1, \lambda x_2, \dots, \lambda x_n)\|_2$

$$= \sqrt{(\lambda x_1)^2 + (\lambda x_2)^2 + \dots + (\lambda x_n)^2}$$

$$= \sqrt{\lambda^2(x_1^2 + x_2^2 + \dots + x_n^2)}$$

$$= \sqrt{\lambda^2} \sqrt{x_1^2 + x_2^2 + \dots + x_n^2} = |\lambda| \|x\|_2$$

Portanto, $\|\lambda x\|_2 = |\lambda| \cdot \|x\|_2$

iii) $\|x + y\|_2^2 = \langle x + y, x + y \rangle = \langle x, x \rangle + \langle x, y \rangle + \langle y, x \rangle + \langle y, y \rangle$

$$= \|x\|_2^2 + 2\langle x, y \rangle + \|y\|_2^2 \leq \|x\|_2^2 + 2\|x\|\|y\| + \|y\|_2^2$$

(Desigualdade de Cauchy-Schwarz)

$$\|x + y\|_2^2 \leq (\|x\|_2 + \|y\|_2)^2$$

$$\therefore \|x + y\|_2 \leq \|x\|_2 + \|y\|_2$$

De i), ii) e iii) e da Definição 3.1 temos que $\|\cdot\|_2$ é uma norma em \mathbb{R}^n

□

Propriedade 3.3. A função $\|\cdot\|_\infty : \mathbb{R}^n \rightarrow \mathbb{R}$ definida por:

$$\|x\|_\infty = \max_{1 \leq i \leq n} \{|x_i|\},$$

é uma norma em \mathbb{R}^n , denominada Norma l_∞ ou Norma do Máximo.

Demonstração. Considere $x, y \in \mathbb{R}^n$ e $\lambda \in \mathbb{R}$.

i) Se $x = (x_1, x_2, \dots, x_n) \neq 0$, então $\exists x_k \neq 0 \implies |x_k| > 0$. Logo,

$$\|x\|_\infty = \max_{0 \leq i \leq n} \{|x_1| + |x_2| + \dots + \underbrace{|x_k|}_{\neq 0} + \dots + |x_n|\} \geq |x_k| > 0.$$

$$\text{ii)} \quad \|\lambda x\|_\infty = \|\lambda(x_1, x_2, \dots, x_n)\|_\infty = \|(\lambda x_1, \lambda x_2, \dots, \lambda x_n)\|_\infty$$

$$= \max_{0 \leq i \leq n} \{|\lambda x_i|\} = \max_{0 \leq i \leq n} \{|\lambda| |x_i|\}$$

$$= |\lambda| \max_{0 \leq i \leq n} \{|x_i|\} = |\lambda| \|x\|_\infty$$

$$\therefore \|\lambda x\|_\infty = |\lambda| \|x\|_\infty$$

$$\text{iii)} \quad \|x + y\|_\infty = \|(x_1, x_2, \dots, x_n) + (y_1, y_2, \dots, y_n)\|_\infty$$

$$= \|(x_1 + y_1, x_2 + y_2, \dots, x_n + y_n)\|_\infty$$

$$= \max_{0 \leq i \leq n} \{|x_i + y_i|\} \leq \max_{0 \leq i \leq n} \{|x_i| + |y_i|\}$$

(Pela desigualdade triangular em \mathbb{R})

$$\leq \max_{0 \leq i \leq n} \{|x_i|\} + \max_{0 \leq i \leq n} \{|y_i|\}$$

$$\leq \|x\|_\infty + \|y\|_\infty$$

$$\text{Logo,} \quad \|x + y\|_\infty \leq \|x\|_\infty + \|y\|_\infty$$

De i), ii) e iii) e da Definição 3.1 temos que $\|\cdot\|_\infty$ é uma norma em \mathbb{R}^n . □

Exemplo 3.1. Para o vetor $x = (4, 4, -4, 4)$ temos

$$\text{i)} \quad \|x\|_2 = \sqrt{4^2 + 4^2 + (-4)^2 + 4^2} = \sqrt{64} = 8$$

$$\text{ii)} \quad \|x\|_\infty = \max_{1 \leq i \leq 4} \{|x_i|\} = \max_{1 \leq i \leq 4} \{|4|, |4|, |-4|, |4|\} = 4$$

$$\text{iii)} \quad \|x\|_1 = \sum_{i=1}^4 |x_i| = |4| + |4| + |-4| + |4| = 16.$$

3.2 Normas Matriciais

Similar às normas em \mathbb{R}^n , poderia-se procurar funções que verifiquem as condições i) ii) e iii) da Definição 3.1 de modo a gerar normas para matrizes. Entretanto, se prefere definir a norma de uma matriz quadrada de ordem n associada à norma de um vetor em \mathbb{R}^n , como segue na seguinte definição.

Definição 3.2. [7] Seja $\|\cdot\|$ uma norma em \mathbb{R}^n , e A uma matriz de ordem n . Definimos a norma de uma matriz A , denotada por $\|A\|$ como

$$\|A\| = \sup\{\|Au\| : u \in \mathbb{R}^n, \|u\| = 1\} \quad (3.1)$$

A seguinte propriedade verifica que $\|A\|$ definida por (3.1) é uma norma no espaço das matrizes.

Propriedade 3.4. Seja $\|\cdot\|$ uma norma em \mathbb{R}^n , então:

$$\|A\| = \sup\{\|Au\| : u \in \mathbb{R}^n, \|u\| = 1\}$$

define uma norma no espaço vetorial da matrizes quadradas de ordem n , isto é:

- i) $\|A\| > 0$, se $A \neq 0$, $A \in M_n(\mathbb{R})$
- ii) $\|\alpha A\| = |\alpha| \cdot \|A\|$, se $\alpha \in \mathbb{R}$, $A \in M_n(\mathbb{R})$
- iii) $\|A + B\| \leq \|A\| + \|B\|$, se $A, B \in M_n(\mathbb{R})$

Demonstração. Sejam $A, B \in M_n(\mathbb{R})$ e $\alpha \in \mathbb{R}$.

- i) Se $A \neq 0$, então A tem pelo menos uma coluna distinta de zero. Digamos que $A^{(j)} \neq 0$. Considere um vetor em que 1 esteja na j -ésima componente, isto é, $x = (0, \dots, 0, 1, 0, \dots, 0)^T$. É obvio que $x \neq 0$ e que o vetor $v = x/\|x\|$ tem norma igual a 1. Além mais $Ax = A^{(j)}$ e $\|Ax\| = \|A^{(j)}\| \neq 0$. Segue, pela definição de $\|A\|$,

$$\|A\| = \sup\{\|Au\| : u \in \mathbb{R}^n, \|u\| = 1\} \geq \|Av\| = \frac{\|Ax\|}{\|x\|} = \frac{\|A^{(j)}\|}{\|x\|} > 0$$

$$\begin{aligned} \text{ii)} \quad \|\alpha A\| &= \sup\{\|(\alpha A)u\| : \|u\| = 1\} = \sup\{\|\alpha(Au)\| : \|u\| = 1\} \\ &= \sup\{|\alpha| \cdot \|Au\| : \|u\| = 1\} = |\alpha| \cdot \sup\{\|Au\| : \|u\| = 1\} = |\alpha| \cdot \|A\| \end{aligned}$$

$$\begin{aligned} \text{iii)} \quad \|A + B\| &= \sup\{\|(A + B)u\| : \|u\| = 1\} \\ &= \sup\{\|Au + Bu\| : \|u\| = 1\} \leq \sup\{\|Au\| + \|Bu\| : \|u\| = 1\} \\ &\leq \sup\{\|Au\| : \|u\| = 1\} + \sup\{\|Bu\| : \|u\| = 1\} \leq \|A\| + \|B\| \end{aligned}$$

De i), ii) e iii) e da Definição 3.1 temos que $\|A\|$ é uma norma em $M_n(\mathbb{R})$. □

Como consequência da definição de norma matricial 3.1, as seguintes propriedades se verificam:

Propriedade 3.5. [7]

- i) $\|Ax\| \leq \|A\| \cdot \|x\|$, $\forall A \in M_n(\mathbb{R}), \forall x \in \mathbb{R}^n$
- ii) $\|I\| = 1$, onde I é a matriz identidade em $M_n(\mathbb{R})$
- iii) $\|AB\| \leq \|A\| \cdot \|B\|$, $\forall A, B \in M_n(\mathbb{R})$

Demonstração.

- i) Se $x = 0$ então $\|x\| = 0$ e $\|Ax\| = 0$, e portanto $\|Ax\| = \|A\| \cdot \|x\|$.

Se $x \neq 0$ então $\|x\| \neq 0$ e $\|Ax\| \neq 0$. Fazendo $v = \frac{\|Ax\|}{\|x\|}$ temos que $\|v\| = 1$. Logo,

$$\|A\| = \sup\{\|Au\| : \|u\| = 1\} \geq \|Av\| \geq \left\| A \cdot \frac{x}{\|x\|} \right\| \geq \frac{1}{\|x\|} \cdot \|Ax\|.$$

Multiplicando por $\|x\|$ resulta que $\|Ax\| \leq \|A\| \cdot \|x\|$.

- ii) $\|I\| = \sup\{\|Iu\| : \|u\| = 1\} = \sup\{\|u\| : \|u\| = 1\} = \sup\{1\} = 1$

- iii) $\|AB\| = \sup\{\|(AB)u\| : \|u\| = 1\} = \sup\{\|A(Bu)\| : \|u\| = 1\}$

$$\leq \sup\{\|A\| \cdot \|Bu\| : \|u\| = 1\} \leq \|A\| \cdot \sup\{\|Bu\| : \|u\| = 1\} \leq \|A\| \cdot \|B\|$$

o que completa a prova. □

Os seguintes teoremas determinam uma forma prática de calcular as normas matriciais associadas às normas vetoriais l_∞ e l_1 .

Teorema 3.1. *Se a norma vetorial $\|\cdot\|_\infty$ é definida por $\|x\|_\infty = \max_{1 \leq i \leq n} \{|x_i|\}$, então a norma matricial associada é calculada por:*

$$\|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$$

Demonstração. Sabemos que a norma de A é definida como:

$$\|A\| = \sup\{\|Au\| : u \in \mathbb{R}^n, \|u\| = 1\}$$

Vamos supor que N representa o valor máximo. Então temos que $\|Au\| \leq N$, para todo vetor unitário u .

Vamos mostrar que $\|Au\|_\infty \leq N$ e determinar um vetor unitário específico para o qual a igualdade seja válida.

Para as A_i , onde $i = 1 : n$, linhas da matriz, temos $N = \max_{1 \leq i \leq n} \{|A_i|\}$, o valor máximo dentre as somas dos valores absolutos nas linhas e $\|u\|_\infty = 1$.

Pela definição de norma vetorial do máximo, temos:

$$\|u\|_\infty = \max_{1 \leq i \leq n} \{|u_1|, |u_2|, |u_3|, \dots, |u_n|\} = 1$$

Seja a matriz $A_{n \times n}$ e o vetor u :

$$Au = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \cdot \begin{pmatrix} u_1 & u_2 & \cdots & u_n \end{pmatrix}^T = \begin{pmatrix} a_{11}u_1 + a_{12}u_2 + \cdots + a_{1n}u_n \\ a_{21}u_1 + a_{22}u_2 + \cdots + a_{2n}u_n \\ \vdots \\ a_{n1}u_1 + a_{n2}u_2 + \cdots + a_{nn}u_n \end{pmatrix}$$

Tomando a norma do máximo, temos:

$$\|A\|_\infty = \max_{1 \leq i \leq n} \{|a_{11}u_1 + a_{12}u_2 + \cdots + a_{1n}u_n|, |a_{21}u_1 + a_{22}u_2 + \cdots + a_{2n}u_n|, \dots, |a_{n1}u_1 + a_{n2}u_2 + \cdots + a_{nn}u_n|\}$$
 Usando a Desigualdade Triangular:

$$\begin{aligned} &\leq \max_{1 \leq i \leq n} \{|a_{11}u_1| + |a_{12}u_2| + \cdots + |a_{1n}u_n|, |a_{21}u_1| + |a_{22}u_2| + \cdots + |a_{2n}u_n|, \dots, |a_{n1}u_1| + |a_{n2}u_2| + \cdots + |a_{nn}u_n|\} \\ &\leq \max_{1 \leq i \leq n} \{|a_{11}||u_1| + |a_{12}||u_2| + \cdots + |a_{1n}||u_n|, |a_{21}||u_1| + |a_{22}||u_2| + \cdots + |a_{2n}||u_n|, \dots, |a_{n1}||u_1| + |a_{n2}||u_2| + \cdots + |a_{nn}||u_n|\} \\ &= \max_{1 \leq i \leq n} \{|A_1||u| + |A_2||u| + \cdots + |A_n||u|\} = \max_{1 \leq i \leq n} \{|A_1| + |A_2| + \cdots + |A_n|\} = N \end{aligned}$$

Se o valor máximo ocorre na linha k , então $u = e_k$, onde e_k é um vetor unitário do \mathbb{R}^n . Logo,

$$\|Ae_k\|_\infty = \|A_k\|_\infty = N$$

Portanto,

$$\|A\| = \sup\{\|Au\| : u \in \mathbb{R}^n, \|u\| = 1\} = \|A\|_\infty = \max_{1 \leq i \leq n} \sum_{j=1}^n |a_{ij}|$$

o que completa a prova [8]. □

Teorema 3.2. Se a norma vetorial $\|\cdot\|_1$ se define por $\|x\|_1 = \sum_{i=1}^n |x_i|$, então a norma matricial associada é definida por:

$$\|A\|_1 = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|$$

Demonstração. Sabendo que a norma de A é definida como:

$$\|A\| = \sup\{ \|Au\| : u \in \mathbb{R}^n, \|u\| = 1 \}$$

Vamos supor que M representa o valor máximo. Então temos que $\|Ax\|_1 \leq M$, para qualquer vetor unitário $\|x\|$. Achar x , usando a norma da soma para validar a igualdade. $M = \max\{\|A_j\|_1\}$ o valor máximo da soma absoluta das colunas e $\|x\| = 1$.

$$Ax = \begin{pmatrix} a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \\ a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \\ \vdots \\ a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n \end{pmatrix}$$

$$\|Ax\|_1 = |a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n| + |a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n| + \cdots + |a_{n1}x_1 + a_{n2}x_2 + \cdots + a_{nn}x_n|$$

Utilizando a propriedade de módulo, temos:

$$\|Ax\|_1 \leq (|a_{11}| + |a_{12}| + \cdots + |a_{1n}|)|x_1| + \cdots + (|a_{n1}| + |a_{n2}| + \cdots + |a_{nn}|)|x_n|$$

$$\|Ax\|_1 \leq |A_1||x_1| + |A_2||x_2| + \cdots + |A_n||x_n|$$

$$\|Ax\|_1 \leq M|x_1| + M|x_2| + \cdots + M|x_n|$$

$$\|Ax\|_1 \leq M(|x_1| + |x_2| + \cdots + |x_n|) = M\|x\|_1 = M$$

Se na coluna, ocorrer que seja o valor máximo dentre a soma dos valores absolutos das colunas, conclui-se que, para $x = e_k$, obtemos:

$$\|Ax\|_1 = \|Ae_k\|_1 = \|A_k\|_1 = M$$

Portanto,

$$\|A\|_1 = M = \max_{1 \leq j \leq n} \{ \|A_j\|_1 \} = \max_{1 \leq j \leq n} \sum_{i=1}^n |a_{ij}|,$$

o que completa a prova [9]. □

Exemplo 3.2. Para a matriz $A = \begin{bmatrix} 4 & -3 & 2 \\ -1 & 0 & 6 \\ 2 & 6 & 2 \end{bmatrix}$ temos que,

i) Soma dos módulos dos elementos da matriz em cada linha:

$$\text{Linha 1: } |4| + |-3| + |2| = 9$$

$$\text{Linha 2: } |-1| + |0| + |6| = 7$$

$$\text{Linha 3: } |2| + |6| + |2| = 10.$$

$$\text{Logo, } \|A\|_\infty = \max\{9, 7, 10\} = 10.$$

ii) Soma dos módulos dos elementos da matriz em cada coluna:

$$\text{Coluna 1: } |4| + |-1| + |2| = 7$$

$$\text{Coluna 2: } |-3| + |0| + |6| = 9$$

$$\text{Coluna 3: } |2| + |6| + |-2| = 10.$$

$$\text{Logo, } \|A\|_1 = \max\{7, 9, 10\} = 10.$$

3.3 Convergência em Norma

Um espaço vetorial normado $(V, \|\cdot\|)$ é um espaço vetorial V com norma $\|\cdot\|$. O conceito de norma nos permite definir o conceito de convergência sobre espaços vetoriais [7].

Definição 3.3. A sequência $x^{(1)}, x^{(2)}, \dots$ em $(V, \|\cdot\|)$ converge ao vetor $x \in (V, \|\cdot\|)$, escrevemos $x^{(k)} \rightarrow x$, se

$$\lim_{k \rightarrow \infty} \|x^{(k)} - x\| = 0. \quad (3.2)$$

A convergência 3.2 coincide com a idéia intuitiva que a distância entre os vetores $x^{(k)}$ e o vetor limite x se aproxima de zero quando k aumenta.

Em espaços de dimensão finita, como é o caso de \mathbb{R}^n e $M_n(\mathbb{R})$, se uma sequência converge em uma norma converge em qualquer outra norma. Entretanto, esta afirmação não se verifica em espaços de dimensão infinita.

Exemplo 3.3. Se $x^{(k)} = \begin{bmatrix} 2 - k^{-1} \\ -5 + k^{-2} \\ 3 + e^{-k} \end{bmatrix}$ e $x = \begin{bmatrix} 2 \\ -5 \\ 3 \end{bmatrix}$, então $x^{(k)} - x = \begin{bmatrix} -k^{-1} \\ k^{-2} \\ e^{-k} \end{bmatrix}$.

Em \mathbb{R}^3 ,

$$\lim_{k \rightarrow \infty} \|x^{(k)} - x\|_\infty = \lim_{k \rightarrow \infty} k^{-1} = 0.$$

Como \mathbb{R}^3 é um espaço de dimensão finita, verifica-se imediatamente que

$$\lim_{k \rightarrow \infty} \|x^{(k)} - x\|_1 = 0 \text{ e } \lim_{k \rightarrow \infty} \|x^{(k)} - x\|_2 = 0.$$

Em espaços vetoriais normados de dimensão finita toda sequência $\{x^{(k)}, k = 1, 2, \dots\}$ que verifica o Critério de Cauchy:

$$\lim_{k \rightarrow \infty} \sup_{i, j \geq 1} \|x^{(i)} - x^{(j)}\| = 0, \quad (3.3)$$

é convergente, isto é, existe $x \in V$ tal que $x^{(k)} \rightarrow x$ em norma[7].

Se $x^{(k)}$ é a k-ésima aproximação num método iterativo que resolve a equação $Ax = b$, então pode-se definir um critério de parada de acordo com uma tolerância $\varepsilon > 0$ pre-definida. Os critérios de parada mais usais são:

i) Erro Absoluto: $\|x^{k+1} - x^k\| \leq \varepsilon$

ii) Erro Relativo: $\frac{\|x^{k+1} - x^k\|}{\|x^k\|} \leq \varepsilon$

iii) Residual: $\|b - Ax^k\| \leq \varepsilon$.

Capítulo 4

Métodos Numéricos

Os métodos numéricos para a resolução de um sistema linear da forma $Ax = b$ podem ser divididos em dois grupos: *Métodos Diretos* e *Métodos Iterativos*.

Métodos Diretos são aqueles que, a menos de erros de arredondamento, fornecem a solução exata do sistema linear, caso ela exista, após um número finito de operações.

Os *Métodos Iterativos* geram uma sequência de vetores $\{x^{(k)}\}$, a partir de uma aproximação inicial $\{x^{(0)}\}$. Sob certas condições esta sequência converge para a solução x , caso ela exista.

4.1 Métodos Diretos

Os Métodos Diretos são aqueles que após um número finito de operações fornecem a solução exata do sistema, a menos dos erros de arredondamentos. Estes métodos são baseados no processo de escalonamento estudado em Álgebra Linear. São eficientes para sistemas de pequeno porte (não mais que 50 equações) e para sistemas de bandas, como por exemplo sistemas tridiagonais. Primeiramente vamos considerar os sistemas lineares triangulares.

4.1.1 Sistema Triangular Superior

Um sistema triangular superior é um sistema de equações lineares da forma $Ux = b$, com $U = (u_{ij})$, $b = (b_i)^T$ e $x = (x_j)^T$, onde U é uma matriz triangular superior quadrada de ordem n , isto é, com seus elementos $u_{ij} = 0$ para todo $i > j$, e com $u_{ii} \neq 0$.

Podemos então escrever o sistema triangular inferior como:

$$(*) \left\{ \begin{array}{l} u_{11}x_1 + u_{12}x_2 + u_{13}x_3 + u_{4n}x_n + \cdots + u_{1n}x_n = b_1 \\ u_{22}x_2 + u_{23}x_3 + u_{14}x_4 + \cdots + u_{2n}x_n = b_2 \\ u_{33}x_3 + u_{34}x_4 + \cdots + u_{3n}x_n = b_3 \\ u_{44}x_4 + \cdots + u_{4n}x_n = b_4 \\ \vdots \\ u_{nn}x_n = b_n \end{array} \right.$$

O sistema $(*)$ obtém sua solução por Retro-Substituição, isto é, iniciando da última equação obtemos x_n , da penúltima x_{n-1} e assim consecutivamente.

Exemplo 4.1. Dado o sistema triangular

$$\left\{ \begin{array}{ll} x + 2y - z + 3t = 6 & (I) \\ y + 3z - t = -5 & (II) \\ 5z + 7t = 21 & (III) \\ 2t = 6 & (IV) \end{array} \right.$$

Temos:

$$\text{em } (IV) \quad 2t = 6 \Rightarrow t = 3$$

$$\text{em } (III) \quad 5z + 7 \cdot 3 = 21 \Rightarrow 5z = 0 \Rightarrow z = 0$$

$$\text{em } (II) \quad y + 3 \cdot 0 - 3 = -5 \Rightarrow y = -2$$

$$\text{em } (I) \quad x + 2 \cdot (-2) - 0 + 3 \cdot 3 = 6 \Rightarrow x = 1$$

Portanto a solução do sistema é $(1, -2, 0, 3)$

4.1.2 Método de Eliminação de Gauss

Propriedade 4.1. A solução do sistema $Ax = b$ não se altera se são realizadas operações elementares do tipo:

i) Troca de ordem duas equações

ii) Multiplicação de uma equação por uma constante não nula

iii) Adicionar um múltiplo de uma equação a uma outra equação.

Isto é, ao realizar operações elementares no sistema $Ax = b$ obtemos um novo sistema equivalente $\bar{A}x = \bar{b}$.

O Método de Eliminação de Gauss consiste em eliminar variáveis do sistema sucessivamente, até obter um sistema modificado em que a matriz dos coeficientes seja triangular superior. Em termos gerais esquema de eliminação consiste em:

- 1) Transformar o sistema $Ax = b$ no sistema $Ux = C$, onde U é uma matriz triangular superior.
- 2) Resolver o sistema triangular $Ux = C$ por substituição retroativa.

Para descrever o Método de Eliminação de Gauss vamos considerar o sistema linear

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \cdots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 + \cdots + a_{2n}x_n = b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 + \cdots + a_{3n}x_n = b_3 \\ \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \quad \quad \quad \vdots \\ a_{n1}x_1 + a_{n2}x_2 + a_{n3}x_3 + \cdots + a_{nn}x_n = b_n \end{cases}$$

onde $\det(A) \neq 0$, isto é, o sistema admite uma única solução. Um sistema linear pode ser representado na forma de matriz estendida $(A^0 | b^0) = A | b$, ou seja

$$\left[\begin{array}{ccccc|c} a_{11}^{(0)} & a_{12}^{(0)} & a_{13}^{(0)} & \cdots & a_{1n}^{(0)} & b_1^{(0)} \\ a_{21}^{(0)} & a_{22}^{(0)} & a_{23}^{(0)} & \cdots & a_{2n}^{(0)} & b_2^{(0)} \\ a_{31}^{(0)} & a_{32}^{(0)} & a_{33}^{(0)} & \cdots & a_{3n}^{(0)} & b_3^{(0)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ a_{n1}^{(0)} & a_{n2}^{(0)} & a_{n3}^{(0)} & \cdots & a_{nn}^{(0)} & b_n^{(0)} \end{array} \right]$$

onde o índice superior indica a etapa do processo.

A eliminação é feita por colunas, chamaremos de etapa k do processo a fase em que se elimina a variável x_k das equações $k+1, k+2, \dots, n$. Usaremos a notação $a_{ij}^{(k)}$ para

denotar o coeficiente da linha i e coluna j no final da k -ésima etapa, bem como $b_i^{(k)}$ será o i -ésimo elemento do vetor constante no final da etapa k .

Considerando que $\det(A) \neq 0$, é sempre possível reescrever o sistema linear de forma que o elemento da posição a_{11} seja diferente de zero, usando a operação elementar $i)$ na Propriedade 4.1. O procedimento da Eliminação de Gauss segue os seguintes passos:

1. Eliminar a incógnita x_1 das equações $k = 2, 3, \dots, n$. Sendo $a_{11}^{(0)} \neq 0$, usaremos a operação elementar $iii)$ e subtraímos da linha k a 1ª linha multiplicada por

$$m_{k1} = \frac{a_{k1}^{(0)}}{a_{11}^{(0)}}$$

Os elementos m_{k1} são chamados de multiplicadores e o elemento $a_{11}^{(0)}$ é chamado de pivô da Etapa 1. Indicando a linha k da matriz entendida por $L_k^{(0)}$ esta etapa se resume em

$$\begin{aligned} L_1^{(1)} &= L_1^{(0)} \\ L_k^{(1)} &= L_k^{(0)} - m_{k1}L_1^{(0)}, \quad k = 2, 3, \dots, n \end{aligned}$$

Ao final desta etapa teremos a matriz

$$\left[\begin{array}{ccccc|c} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} & b_1^{(1)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & \cdots & a_{2n}^{(1)} & b_2^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & \cdots & a_{3n}^{(1)} & b_3^{(1)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & a_{n2}^{(1)} & a_{n3}^{(1)} & \cdots & a_{nn}^{(1)} & b_n^{(1)} \end{array} \right]$$

que representa um sistema linear equivalente ao sistema original, onde a incógnita x_1 foi eliminada das equações $k = 2, 3, \dots, n$.

2. Eliminar a incógnita x_2 das equações $k = 3, 4, \dots, n$. Sendo $a_{22}^{(1)} \neq 0$, vamos tomar este elemento como pivô desta etapa e desta forma os multiplicadores são dados por

$$m_{k2} = \frac{a_{k2}^{(1)}}{a_{22}^{(1)}}$$

A eliminação se procede da seguinte forma:

$$\begin{aligned} L_1^{(2)} &= L_1^{(1)} \\ L_2^{(2)} &= L_2^{(1)} \\ L_k^{(2)} &= L_k^{(1)} - m_{k2}L_2^{(1)}, \quad k = 3, 4, \dots, n \end{aligned}$$

obtendo ao final da etapa a matriz

$$\left[\begin{array}{ccccc|c} a_{11}^{(2)} & a_{12}^{(2)} & a_{13}^{(2)} & \cdots & a_{1n}^{(2)} & b_1^{(2)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2n}^{(2)} & b_2^{(2)} \\ 0 & 0 & a_{33}^{(2)} & \cdots & a_{3n}^{(2)} & b_3^{(2)} \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} & b_n^{(2)} \end{array} \right]$$

3. Com procedimentos análogos ao das etapas 1 e 2 podemos eliminar as incógnitas x_k das equações $k + 1, k + 2, \dots, n$ e ao final de $n - 1$ etapas teremos a matriz

$$\left[\begin{array}{ccccc|c} a_{11}^{(n-1)} & a_{12}^{(n-1)} & a_{13}^{(n-1)} & \cdots & a_{1n}^{(n-1)} & b_1^{(n-1)} \\ 0 & a_{22}^{(n-1)} & a_{23}^{(n-1)} & \cdots & a_{2n}^{(n-1)} & b_2^{(n-1)} \\ 0 & 0 & a_{33}^{(n-1)} & \cdots & a_{3n}^{(n-1)} & b_3^{(n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn}^{(n-1)} & b_n^{(n-1)} \end{array} \right]$$

Esta matriz representa um sistema triangular superior equivalente ao sistema original. Logo a solução deste sistema, obtido pela Retro-Solução, é solução do sistema original[3].

O algoritmo computacional para o Método de Eliminação de Gauss é descrito a seguir.

Algoritmo 4.1. [3] Método de Eliminação de Gauss

Input: Matriz A e vetor $b \in \mathbb{R}^n$

Eliminação:

Para $k = 1, 2, \dots, n - 1$, faça:

Para $i = k + 1, 2, \dots, n$, faça:

$$m \leftarrow \frac{a_{ij}}{a_{k,k}}$$

Para $j = k + 1, 2, \dots, n$, faça:

$$a_{ij} \leftarrow a_{ij} - m * a_{kj}$$

fim para

$$b_i \leftarrow b_i - m * b_k$$

fim para

fim para

Retro-Solução:

$$x_n \leftarrow b_n / a_{n,n}$$

Para $k = n - 1, n - 2, \dots, 1$, faça:

$$x_k \leftarrow \frac{1}{a_{k,k}} \left(b_k - \sum_{j=k+1}^n a_{k,j} x_j \right)$$

fim para

Output: $x \in \mathbb{R}^n$: solução do sistema

Exemplo 4.2. Consideremos o sistema linear

$$\begin{cases} 3x_1 + 2x_2 - x_3 = 1 \\ 7x_1 - x_2 - x_3 = -2 \\ x_1 + x_3 = 1 \end{cases}$$

Escrevendo na forma de matriz estendida teremos

$$\left(\begin{array}{ccc|c} 3 & 2 & -1 & 1 \\ 7 & -1 & -1 & -2 \\ 1 & 0 & 1 & 1 \end{array} \right)$$

Seguindo os passos do Método de Eliminação de Gauss temos,

1. Eliminar x_1 das linhas 2 e 3.

$$\begin{aligned} L_1^{(1)} &= L_1^{(0)} \\ L_2^{(1)} &= L_2^{(0)} - m_{21}L_1^{(0)}, \text{ onde } m_{21} = \frac{a_{21}^{(0)}}{a_{11}^{(0)}} = \frac{7}{3} \\ L_3^{(1)} &= L_3^{(0)} - m_{31}L_1^{(0)}, \text{ onde } m_{31} = \frac{a_{31}^{(0)}}{a_{11}^{(0)}} = \frac{1}{3} \end{aligned}$$

e com isto obtemos a matriz

$$\left(\begin{array}{ccc|c} 3 & 2 & -1 & 1 \\ 0 & -17/3 & 4/3 & -13/3 \\ 0 & -2/3 & 4/3 & 12/3 \end{array} \right)$$

2. Eliminar x_2 da linha 3.

$$\begin{aligned} L_1^{(2)} &= L_1^{(1)} \\ L_2^{(2)} &= L_2^{(1)} \\ L_3^{(2)} &= L_3^{(1)} - m_{32}L_2^{(1)}, \text{ onde } m_{32} = \frac{a_{32}^{(1)}}{a_{22}^{(1)}} = \frac{2}{17} \end{aligned}$$

obtendo assim a matriz

$$\left(\begin{array}{ccc|c} 3 & 2 & -1 & 1 \\ 0 & -17/3 & 4/3 & -13/3 \\ 0 & 0 & 20/17 & 20/17 \end{array} \right)$$

3. **Retro-Solução.** Encontrar a solução do sistema triangular superior.

$$\begin{aligned} x_3 &= \frac{b_3}{a_{33}} = 1 \\ x_2 &= \frac{1}{a_{22}}(b_2 - a_{23}x_3) = 1 \\ x_1 &= \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3) = 0 \end{aligned}$$

Logo a solução do sistema é dada por $x = (0, 1, 1)^T$

A solução encontrada é a solução exata, pois mantivemos os números resultantes na forma de fração. Porém máquinas digitais representam estes números na forma de ponto flutuante finita e erros de arredondamento podem ocorrer. Em sistemas lineares de grande porte estes erros vão se acumulando e prejudicando a solução do sistema[3].

4.1.3 Fatoração LU

Seja o sistema linear $Ax = b$. O processo de *Fatoração LU* para a resolução deste sistema consiste em decompor a matriz A dos coeficientes em um produto de dois ou mais fatores e, em seguida, resolver uma sequência de sistemas lineares que nos conduzirá à solução do sistema linear original.

Por exemplo, se pudermos realizar a fatoração: $A = LU$, o sistema linear $Ax = b$ pode ser escrito como, $(LU)x = b$. Seja $y = Ux$, então resolver o sistema linear $Ax = b$ é equivalente a resolver o sistema linear $Ly = b$ e, em seguida, o sistema linear $Ux = y$.

A vantagem dos processos de fatoração é que podemos resolver qualquer sistema linear que tenha A como matriz dos coeficientes. Se o vetor b for alterado, a resolução do novo sistema linear será quase que imediata.

A *fatoração LU de Doolite* é um dos processos de fatoração mais empregados. Nesta fatoração a matriz L é triangular inferior com elementos da diagonal igual a 1, e a matriz U é triangular superior.

Teorema 4.1. [10] *Dada uma matriz A quadrada de ordem n , seja A_k a matriz constituída das primeiras k linhas e colunas de A . Suponha que $\det(A_k) \neq 0$ para $k = 1, 2, \dots, (n-1)$. Então, existe uma única matriz triangular inferior $L = (l_{ij})$, com $l_{ii} = 1$, $1 \leq i \leq n$ e uma única matriz triangular superior $U = (u_{ij})$ tais que $A = LU$. Ainda mais, $\det(A) = u_{11}u_{22}u_{33} \dots u_{nn}$.*

Exemplo 4.3. Seja a matriz

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

vamos obter a fatoração LU de Doolite:

$$\begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix} \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} \\ l_{21}u_{11} & l_{21}u_{12} + u_{22} & l_{21}u_{13} + u_{23} \\ l_{31}u_{11} & l_{31}u_{12} + l_{32}u_{22} & l_{31}u_{13} + l_{32}u_{23} + u_{33} \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{bmatrix}$$

Cálculo da 1ª linha de U : u_{11}, u_{12}, u_{13}

$$u_{11} = 1, u_{12} = -1, u_{13} = 0$$

Cálculo da 1ª coluna de L : l_{21}, l_{31}

$$l_{21}u_{11} = -1 \implies l_{21} = \frac{-1}{u_{11}} = -1$$

$$l_{31}u_{11} = 0 \implies l_{31} = \frac{0}{u_{11}} = 0$$

Cálculo da 2ª linha de U : u_{22}, u_{23}

$$l_{21}u_{12} + u_{22} = 2 \implies u_{22} = 2 - l_{21}u_{12} = 2 - (-1)(-1) = 1$$

$$l_{21}u_{13} + u_{23} = -1 \implies u_{23} = -1 - l_{21}u_{13} = -1 - (-1)(0) = -1$$

Cálculo da 2ª coluna de L : l_{32}

$$l_{31}u_{12} + l_{32}u_{22} = -1 \implies l_{32} = \frac{-1 - l_{31}u_{12}}{u_{22}} = \frac{-1 - (0)(-1)}{1} = -1$$

Cálculo da 3ª linha de U : u_{33}

$$l_{31}u_{13} + l_{32}u_{23} + u_{33} = 2 \implies u_{33} = 2 - l_{31}u_{13} - l_{32}u_{23} = 2 - (0)(0) - (-1)(-1) = 1$$

$$\text{Portanto, } L = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & -1 & 1 \end{bmatrix} \quad U = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 1 \end{bmatrix}$$

Algoritmo 4.2. [4] Fatoração LU

Dados: Matriz $A = [a_{ij}]$

Para $i = 1 : n$, faça:

Para $j = i : n$, faça:

$$u_{ij} = a_{ij} - \sum_{k=1}^{i-1} l_{ik}u_{kj}$$

Para $j = i + 1 : n$, faça:

$$l_{ji} = \left(a_{ji} - \sum_{k=1}^{i-1} l_{jk}u_{ki} \right) / u_{ii}$$

Algoritmo 4.3. Fatoração LU no MATLAB

```
function [l,u]=doolite(A);
da=size(A)
n=da(1)
%Esta parte verifica se o sistema é quadrado
if n~=da(2);
    disp('??? A matriz deve ser quadrada');
    return;
end
for m=1:n
    for j=m:n
        soma=A(m,j);
        for k=1:m-1
            soma=soma-l(m,k)*u(k,j);
        end
        u(m,j)=soma;
    end
    for i=m+1:n
        soma=A(i,m);
        for k=1:m-1
            soma=soma-l(i,k)*u(k,m);
        end;
        l(i,m)=soma/u(m,m);
        l(m,m)=1;
    end
end
for m=n:n
    soma=A(n,n);
    for k=1:n-1
        soma=soma-l(n,k)*u(k,n);
    end
    u(n,n)=soma;

    l(n,n)=1;
end
```

4.1.4 Fatoração de Cholesky

Há uma fatoração de matrizes que é muito útil em algumas situações. A esta fatoração se é dado o nome do matemático André Louis Cholesky, que demonstrou o seguinte resultado:

Teorema 4.2. [7] *Se A é uma matriz real simétrica positiva definida, então admite uma fatoração única $A = \tilde{L}\tilde{L}^T$, onde \tilde{L} é uma matriz triangular inferior com diagonal positiva.*

Demonstração.

i) A é uma matriz não singular. Pois, se A for singular existe $x \neq 0$ tal que $Ax = 0$.

Logo, $x^T Ax = x \cdot (A^T x) = x \cdot 0 = 0$, o qual contradiz a hipótese que A é definida positiva.

ii) Seja $x = (x_1 \ x_2 \ \cdots \ x_k \ 0 \ \dots \ 0)$, definimos $\bar{x} = (x_1 \ x_2 \ \cdots \ x_k)$. Logo,

$$x^T Ax = \bar{x}^T A_k \bar{x} > 0.$$

Portanto, todos os menores principais A_k da matriz A são definidas positivas. Pelo Teorema 4.1 da decomposição LU, a matriz A admite uma decomposição LU , isto é, $A = LU$.

iii) Como A é simétrica, temos

$$LU = A = A^T = (LU)^T$$

$$LU = U^T L^T$$

$$U = L^{-1} U^T L^T$$

$$\underbrace{U(L^T)^{-1}}_{\text{matriz triangular superior}} = \underbrace{L^{-1} U^T}_{\text{matriz triangular inferior}}$$

logo existe uma matriz diagonal tal que:

$$U(L^T)^{-1} = L^{-1} U^T = D$$

$$U = DL^T.$$

Logo, $A = LU = L(DL^T) = LDL^T$, conhecida como decomposição LDL^T .

Afirmção: D é positiva definida

Prova: Como A é positiva definida, então

$$x^T Ax > 0, \forall x \neq 0$$

$$x^T (LDL^T)x > 0, \forall x \neq 0$$

$$(x^T L)D(x^T L)^T > 0, \forall x \neq 0$$

Fazendo, $z = (x^T L)^T \implies z^T = x^T L$. Segue que:

$$z^T A z > 0, \forall z \neq 0$$

Portanto, D é positiva definida

iv) Como $x^T A x > 0, \forall x \neq 0$, temos

$$\begin{aligned} & \begin{pmatrix} x_1 & x_2 & \dots & x_n \end{pmatrix} \begin{pmatrix} D_{11} & 0 & 0 & \dots & 0 \\ 0 & D_{22} & 0 & \dots & 0 \\ 0 & 0 & D_{33} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & D_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \\ &= \begin{pmatrix} D_{11}x_1 & D_{22}x_2 & \dots & D_{nn}x_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \\ &= \begin{pmatrix} D_{11}x_1^2 + D_{22}x_2^2 + \dots + D_{nn}x_n^2 \end{pmatrix} > 0, \forall x \neq 0 \end{aligned}$$

logo $D_{ii} > 0, \forall i = 1, 2, \dots, n$

v) Definimos $D_{ij}^{1/2} = \delta_{ij} \sqrt{D_{ij}}$

$$D^{1/2} = \begin{pmatrix} \sqrt{D_{11}} & 0 & 0 & \dots & 0 \\ 0 & \sqrt{D_{22}} & 0 & \dots & 0 \\ 0 & 0 & \sqrt{D_{33}} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \sqrt{D_{nn}} \end{pmatrix}$$

Como $D = D^{1/2} D^{1/2}$, temos que

$$A = L D L^T = L D^{1/2} D^{1/2} L^T = L D^{1/2} (D^{1/2} L^T) = \bar{L} \bar{L}^T, \text{ onde } \bar{L} = L D^{1/2}.$$

Por tanto, $A = \bar{L} \bar{L}^T$. □

Algoritmo 4.4. [4] Fatoração de Cholesky

Dados: Matriz $A = [a_{ij}]$, simétrica e positiva definida

Para $i = 1 : n$, faça:

$$d_j = a_{jj} - \sum d_k l_{jk}$$

Para $i = j + 1 : n$, faça:

$$l_{ij} = \left(a_{ij} - \sum_{k=1}^{j-1} d_k l_{ik} l_{jk} \right) / d_j$$

Algoritmo 4.5. Fatoração de Cholesky no MATLAB

```
function [l,d]=cholesky(A)
clc
d = 0;
n=length(A);
l = eye(n);
for m=1:n
l(m,1) = A(m,1);
end
for j=1:n
soma=A(j,j);
for k=1:(j-1)
soma=soma - d(k)*l(j,k)*l(j,k);
end
d(j)=soma;
for i=(j+1):n
soma=A(i,j);
for k=1:j-1
soma=soma-d(k)*l(i,k)*l(j,k);
end
l(i,j)=soma/d(j);
end
end
d=diag(d);
```

Exemplo 4.4. Utilizando o MATLAB vamos encontrar uma fatoração de Cholesky da matriz A abaixo

```
>> A=[1 2 2;2 7 7;2 7 9]
```

A =

1	2	2
2	7	7
2	7	9

```
>> [l,d]=cholesky(A)
```

```
l =
```

```
    1    0    0
    2    1    0
    2    1    1
```

```
d =
```

```
    1    0    0
    0    3    0
    0    0    2
```

4.2 Métodos Iterativos

Os métodos iterativos são aqueles que se baseiam na construção de sequências de aproximações. Em um método iterativo, a cada passo, os valores calculados anteriormente são usados para melhorar a aproximação. O método iterativo será útil se a sequência das aproximações construídas pelo método convergir para a solução do sistema.

Métodos iterativos requerem um chute inicial $x^{(0)}$, um vetor inicial que aproxima a solução exata x (se não há nenhuma informação disponível sobre a solução exata, de modo que não temos como construir o chute inicial de forma inteligente, $x^{(0)}$ pode ser uma aproximação muito ruim de x). Uma vez que $x^{(0)}$ é dado, o método iterativo gera a partir de $x^{(0)}$ uma nova aproximação $x^{(1)}$, que esperamos deve aproximar melhor a solução exata. Em seguida, $x^{(1)}$ é usada para gerar uma nova melhor aproximação $x^{(2)}$ e assim por diante. Desta forma, gera-se uma sequência de vetores $x^{(k)}$ que espera-se convergir para x . Como na prática não podemos iterar para sempre, algum critério de parada deve ser estabelecido a priori. Uma vez que $x^{(k)}$ esteja suficientemente próximo da solução exata quanto se precise, de acordo com uma margem de tolerância aceita, pára-se o processo de iteração e aceita-se $x^{(k)}$ como a solução aproximada adequada para o problema. Por exemplo, o critério de parada pode ser estabelecido através de uma *cota de tolerância* ε : quando

$$\|b - Ax^{(k)}\| < \varepsilon$$

ou quando

$$\|x^{(k+1)} - x^{(k)}\| < \varepsilon$$

as iterações são interrompidas e o último valor aproximado obtido é aceito como a melhor aproximação da solução dentro das circunstâncias.

Os métodos discutidos aqui não necessitam de um bom chute inicial (embora, é claro, quanto melhor o chute inicial, menor o número de iterações necessárias para se chegar à solução aproximada com a precisão especificada).

Para a resolução de um sistema da forma $Ax = b$, começando com uma aproximação inicial $x^{(0)} = \begin{bmatrix} x_1^{(0)} & x_2^{(0)} & x_3^{(0)} & \cdots & x_n^{(0)} \end{bmatrix}^T$ para $x = \begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \end{bmatrix}^T$, onde:

$x_1^{(0)}$ é a aproximação para x_1

$x_2^{(0)}$ é a aproximação para x_2

$x_3^{(0)}$ é a aproximação para x_3

\vdots $\quad \quad \quad \vdots$

$x_n^{(0)}$ é a aproximação para x_n

obtêm-se sucessivas aproximações $x^{(1)} = [x_1^{(1)} \ x_2^{(1)} \ x_3^{(1)} \ \cdots \ x_n^{(1)}]^T$,
 $x^{(2)} = [x_1^{(2)} \ x_2^{(2)} \ x_3^{(2)} \ \cdots \ x_n^{(2)}]^T$, $x^{(3)} = [x_1^{(3)} \ x_2^{(3)} \ x_3^{(3)} \ \cdots \ x_n^{(3)}]^T$, \dots
 Para $x = [x_1 \ x_2 \ x_3 \ \cdots \ x_n]^T$

4.2.1 Obtenção dos Métodos Iterativos

Dado o sistema linear $Ax = b$, decompõem-se a matriz A na forma: $A = M + N$, onde M é uma matriz não-singular. Logo temos:

$$\begin{aligned} Ax &= b \\ (M + N)x &= b \\ Mx + Nx &= b \\ Mx &= b - Nx \\ M^{-1}Mx &= M^{-1}b - M^{-1}Nx \\ Ix &= M^{-1}b - M^{-1}Nx \\ x &= M^{-1}b - M^{-1}Nx \end{aligned} \tag{4.1}$$

O método iterativo é então definido por:

$$\boxed{x^{(k+1)} = M^{-1}b - M^{-1}Nx^{(k)} \quad \text{com } k = 0, 1, 2, \dots} \tag{4.2}$$

onde, $x^{(0)}$ é um vetor dado.

O sistema linear pode ser escrito na forma

$$x^{(k+1)} = Cx^{(k)} + g, \quad (4.3)$$

onde, $g \in M_{n \times 1}(\mathbb{R})$ e $C \in M_n(\mathbb{R})$, isto é, os métodos iterativos seguem um esquema semelhante aos métodos para o cálculo de zeros de funções. $C = -M^{-1}N$ é chamada de **matriz de iteração**. É importante que a matriz M seja muito mais simples do que A , porque senão não estaríamos a simplificar o problema. Note-se que esta iteração pode ser feita de outra forma, não havendo necessidade de se proceder verdadeiramente ao cálculo da inversa de M .

Dentre os métodos iterativos, iremos abordar os seguintes métodos:

- Método de Jacobi
- Método de Gauss-Seidel.
- Métodos SOR

Os métodos iterativos são definidos aqui, por uma escolha particular da matriz M . A matriz M é geralmente diagonal, triangular ou tridiagonal.

Seja o sistema $Ax = b$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \cdots & a_{1n} \\ a_{21} & a_{22} & a_{23} & \cdots & a_{2n} \\ a_{31} & a_{32} & a_{33} & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}$$

Consideremos a matriz A do sistema, decomposta na soma de três matrizes

$$A = L + D + U$$

$$A = \underbrace{\begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ a_{21} & 0 & 0 & \cdots & 0 \\ a_{31} & a_{32} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & a_{n3} & \cdots & 0 \end{bmatrix}}_L + \underbrace{\begin{bmatrix} a_{11} & 0 & 0 & \cdots & 0 \\ 0 & a_{22} & 0 & \cdots & 0 \\ 0 & 0 & a_{33} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & a_{nn} \end{bmatrix}}_D + \underbrace{\begin{bmatrix} 0 & a_{12} & a_{13} & \cdots & a_{1n} \\ 0 & 0 & a_{23} & \cdots & a_{2n} \\ 0 & 0 & 0 & \cdots & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}}_U$$

Onde L é uma matriz triangular estritamente inferior, U uma matriz triangular estritamente superior e D a matriz diagonal.

Notamos que a matriz D não deverá ter zeros na diagonal principal, isto é, seus elementos $a_{ii} \neq 0$. Caso aconteça de $a_{ii} = 0$, deve-se efetuar uma troca de linhas ou colunas na matriz A , para obtermos uma matriz D com elementos $a_{ii} \neq 0$.

4.2.2 Critério de Convergência dos Métodos Iterativos

Vamos estabelecer critérios que nos permitam determinar quando existe convergência para estes métodos iterativos. Pretendem-se métodos que satisfaçam:

$$\lim_{k \rightarrow \infty} x^{(k)} = x \quad \text{ou} \quad \lim_{k \rightarrow \infty} \|x^{(k)} - x\| = 0$$

De (4.1) – (4.2) temos:

$$\begin{aligned} x - x^{(k+1)} &= -M^{-1}N x + M^{-1}N x^{(k)} \\ x - x^{(k+1)} &= -M^{-1}N(x - x^{(k)}) \end{aligned}$$

Fazendo $C = -M^{-1}N$, onde a matriz C é denominada *matriz de iteração*, e $x - x^{(k+1)}$ é o erro cometido na k -ésima iteração, tem-se:

$$x - x^{(k+1)} = C(x - x^{(k)}) \tag{4.4}$$

Atribuindo valores para $k \geq 0$ temos:

$$\begin{aligned} x - x^{(1)} &= C(x - x^{(0)}) \\ x - x^{(2)} &= C(x - x^{(1)}) = C.C(x - x^{(0)}) = C^2(x - x^{(0)}) \\ x - x^{(3)} &= C(x - x^{(2)}) = C.C^2(x - x^{(0)}) = C^3(x - x^{(0)}) \\ x - x^{(4)} &= C(x - x^{(3)}) = C.C^3(x - x^{(0)}) = C^4(x - x^{(0)}) \\ \vdots & \qquad \qquad \qquad \vdots \qquad \qquad \qquad \vdots \end{aligned}$$

Em geral tem-se

$$x - x^{(k+1)} = C^{(k+1)}(x - x^{(0)})$$

Aplicando $\|\cdot\|$ a norma da matriz associada temos

$$0 \leq \|x - x^{(k+1)}\| = \|C^{(k+1)}(x - x^{(0)})\| \leq \|C^{(k+1)}\| \|x - x^{(0)}\|$$

Por outro lado temos que

$$\|C^{(k+1)}\| \leq \|C\|^{(k+1)}$$

Dai segue que

$$0 \leq \|x - x^{(k+1)}\| \leq \|C\|^{(k+1)} \|x - x^{(0)}\|$$

Vemos que se $\|C\| < 1$ então

$$\lim_{k \rightarrow \infty} \|C\|^{(k+1)} = 0 \implies \lim_{k \rightarrow \infty} \|x^{(k)} - x\| = 0$$

Portanto o método iterativo converge se $\|C\| < 1$, para qualquer que seja a iteração inicial $x^{(0)}$. Logo temos provado o seguinte teorema:

Teorema 4.3. *Se $\|C\| < 1$ para alguma norma matricial, então o método iterativo definido por $x^{(k+1)} = Cx^{(k)} + g$ converge qualquer que seja o vetor $x^{(0)}$ dado.*

4.2.3 Estimativa de Erro na Iteração $x^{(k+1)}$

De (4.4) temos que

$$\begin{aligned} x - x^{(k+1)} &= C(x - x^{(k)}) \\ x - x^{(k+1)} &= C(x - x^{(k+1)} + x^{(k+1)} - x^{(k)}) \end{aligned}$$

Aplicando norma $\|\cdot\|$ resulta,

$$\begin{aligned} \|x - x^{(k+1)}\| &= \|C(x - x^{(k+1)} + x^{(k+1)} - x^{(k)})\| \\ &\leq \|C\| \|x - x^{(k+1)} + x^{(k+1)} - x^{(k)}\| \\ \|x - x^{(k+1)}\| &\leq \|C\| (\|x - x^{(k+1)}\| + \|x^{(k+1)} - x^{(k)}\|) \\ &= \|C\| \|x - x^{(k+1)}\| + \|C\| \|x^{(k+1)} - x^{(k)}\| \\ \|x - x^{(k+1)}\| &\leq \|C\| \|x - x^{(k+1)}\| + \|C\| \|x^{(k+1)} - x^{(k)}\| \\ \|x - x^{(k+1)}\| - \|C\| \|x - x^{(k+1)}\| &\leq \|C\| \|x^{(k+1)} - x^{(k)}\| \\ \|x - x^{(k+1)}\| (1 - \|C\|) &\leq \|C\| \|x^{(k+1)} - x^{(k)}\|. \end{aligned}$$

Logo, a estimativa de erro na iteração $k+1$ dos métodos iterativos é dada pela expressão:

$$\|x - x^{(k+1)}\| \leq \frac{\|C\|}{1 - \|C\|} \|x^{(k+1)} - x^{(k)}\|$$

4.2.4 Método de Jacobi

Seja o sistema linear $Ax = b$ e considerando $A = L + D + U$, podemos então escrever o sistema na forma

$$\begin{aligned} Ax &= b \\ (L + D + U)x &= b \\ (L + U)x + Dx &= b \\ Dx &= b - (L + U)x \\ D^{-1}Dx &= D^{-1}b - D^{-1}(L + U)x \\ Ix &= D^{-1}b - D^{-1}(L + U)x \\ x &= D^{-1}b - D^{-1}(L + U)x \end{aligned}$$

O método iterativo definido por:

$$\boxed{x^{(k+1)} = D^{-1}b - D^{-1}(L + U)x^{(k)} \quad \text{com } k = 0, 1, 2, \dots}$$

onde $x^{(0)}$ é um vetor dado, é chamado de **Método de Jacobi**.

Este método é do tipo iterativo definido por $x^{(k+1)} = M^{-1}b - M^{-1}Nx^{(k)}$ com:

$$M = D \text{ e } N = L + U$$

da equação $x^{(k+1)} = D^{-1}b - D^{-1}(L + U)x^{(k)}$ temos:

$$x^{(k+1)} = D^{-1}(b - (L + U)x^{(k)})$$

Desta forma podemos escrever o método iterativo de Jacobi como:

$$\begin{aligned} x_1^{(k+1)} &= \frac{1}{a_{11}}[b_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)} + a_{14}x_4^{(k)} + a_{15}x_5^{(k)} + \dots + a_{1n}x_n^{(k)})] \\ x_2^{(k+1)} &= \frac{1}{a_{22}}[b_2 - (a_{21}x_1^{(k)} + a_{23}x_3^{(k)} + a_{24}x_4^{(k)} + a_{25}x_5^{(k)} + \dots + a_{2n}x_n^{(k)})] \\ x_3^{(k+1)} &= \frac{1}{a_{33}}[b_3 - (a_{31}x_1^{(k)} + a_{32}x_2^{(k)} + a_{34}x_4^{(k)} + a_{35}x_5^{(k)} + \dots + a_{3n}x_n^{(k)})] \\ &\vdots \\ x_n^{(k+1)} &= \frac{1}{a_{nn}}[b_n - (a_{n1}x_1^{(k)} + a_{n2}x_2^{(k)} + a_{n3}x_3^{(k)} + a_{n4}x_4^{(k)} + \dots + a_{nn-1}x_{n-1}^{(k)})] \end{aligned}$$

Em geral $x_i^{(k+1)}$ pode ser obtido pela fórmula:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} x_j^{(k)} \right), \quad i = 1, 2, 3, \dots, n$$

O algoritmo de Jacobi implementado no Matlab é descrito no Algoritmo 4.6 abaixo.

Algoritmo 4.6. Método de Jacobi no MATLAB

```
function [X,delta,Z] = jacobi(A,b,X0,tol,max)
%-----Comentário-----
% Implementa o método iterativo de Jacobi para determinar
% uma solução aproximada de Ax=b
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Executar%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% [X,delta] = jacobi(A,B,P,delta,max1)
% [X,delta,Z] = jacobi(A,B,P,delta,max1)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Entrada%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% A a matriz A do sistema
% b o vetor dos termos independentes
% X0 a solução inicial
% eps se abs(X(k)-X(k-1))< eps FIM !!!
% max número máximo de iterações
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gera%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% X o vetor com a solução
% delta a norma do vetor abs(X(k)-X(k-1))
% Z Matrix com todas as iterações (uma por linha)
%-----
n = length(b);
Xant = X0; % inicializa Xant
X=X0; % inicializa X
Z = X0'; % inicializa Z
for k=1:max, % iterar até max vezes
for j = 1:n, % para cada equação
Sum = b(j) - A(j,[1:j-1,j+1:n])*Xant([1:j-1,j+1:n]);
X(j) = Sum/A(j,j);
end
Z = [Z;X']; % armazena a história
delta = norm(abs(X-Xant),1);
if (delta<tol) break, end
Xant = X;
end
```

Exemplo 4.5. Seja o sistema linear

$$\begin{bmatrix} 10 & 2 & 1 \\ 1 & 5 & 1 \\ 2 & 3 & 10 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 7 \\ -8 \\ 6 \end{bmatrix} \quad \text{cuja solução exata é: } x_1 = 1, x_2 = -2 \text{ e } x_3 = 1$$

Aplicando o método de Jacobi $x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ j \neq i}}^n a_{ij} x_j^{(k)} \right)$ ao sistema acima, temos:

$$x_1^{(k+1)} = \frac{1}{a_{11}} [b_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)})] = \frac{1}{10} \cdot [7 - (2x_2^{(k)} + x_3^{(k)})] = 0,7 - 0,2x_2^{(k)} - 0,1x_3^{(k)}$$

$$x_2^{(k+1)} = \frac{1}{a_{22}} [b_2 - (a_{21}x_1^{(k)} + a_{23}x_3^{(k)})] = \frac{1}{5} \cdot [-8 - (x_1^{(k)} + x_3^{(k)})] = -1,6 - 0,2x_1^{(k)} - 0,2x_3^{(k)}$$

$$x_3^{(k+1)} = \frac{1}{a_{33}} [b_3 - (a_{31}x_1^{(k)} + a_{32}x_2^{(k)})] = \frac{1}{10} \cdot [6 - (2x_1^{(k)} + 3x_2^{(k)})] = 0,6 - 0,2x_1^{(k)} - 0,3x_2^{(k)}$$

$$\text{Tomando } x^{(0)} = [0,7 \quad -1,6 \quad 0,6]^T$$

$$\text{Para } k = 0, \text{ vamos calcular } x^{(1)} = [x_1^{(1)} \quad x_2^{(1)} \quad x_3^{(1)}]^T$$

$$x_1^{(1)} = 0,7 - 0,2x_2^{(0)} - 0,1x_3^{(0)} = 0,7 - 0,2(-1,6) - 0,1(0,6) = 0,9600$$

$$x_2^{(1)} = -1,6 - 0,2x_1^{(0)} - 0,2x_3^{(0)} = -1,6 - 0,2(0,7) - 0,2(0,6) = -1,8600$$

$$x_3^{(1)} = 0,6 - 0,2x_1^{(0)} - 0,3x_2^{(0)} = 0,6 - 0,2(0,7) - 0,3(-1,6) = 0,9400$$

$$\text{Portanto, } x^{(1)} = [0,96 \quad -1,86 \quad 0,94]^T$$

$$\text{Para } k = 1, \text{ vamos calcular } x^{(2)} = [x_1^{(2)} \quad x_2^{(2)} \quad x_3^{(2)}]^T$$

$$x_1^{(2)} = 0,7 - 0,2x_2^{(1)} - 0,1x_3^{(1)} = 0,7 - 0,2(-1,86) - 0,1(0,94) = 0,9780$$

$$x_2^{(2)} = -1,6 - 0,2x_1^{(1)} - 0,2x_3^{(1)} = -1,6 - 0,2(0,96) - 0,2(-1,86) = -1,9800$$

$$x_3^{(2)} = 0,6 - 0,2x_1^{(1)} - 0,3x_2^{(1)} = 0,6 - 0,2(0,96) - 0,3(-1,86) = 0,9660$$

$$\text{Portanto, } x^{(2)} = [0,9780 \quad -1,9800 \quad 0,9660]^T$$

$$\text{Para } k = 2, \text{ vamos calcular } x^{(3)} = [x_1^{(3)} \quad x_2^{(3)} \quad x_3^{(3)}]^T$$

$$x_1^{(3)} = 0,7 - 0,2x_2^{(2)} - 0,1x_3^{(2)} = 0,7 - 0,2(-1,9800) - 0,1(0,9660) = 0,9994$$

$$x_2^{(3)} = -1,6 - 0,2x_1^{(2)} - 0,2x_3^{(2)} = -1,6 - 0,2.(0,9780) - 0,2.(0,9660) = -1,9888$$

$$x_3^{(3)} = 0,6 - 0,2x_1^{(2)} - 0,3x_2^{(2)} = 0,6 - 0,2.(0,9780) - 0,3.(-1,9800) = 0,9984$$

$$\text{Portanto, } x^{(3)} = [0,9994 \quad -1,9888 \quad 0,9984]^T$$

As iterações do algoritmo de Jacobi se descrevem na Tabela 4.1. Os resultados mostram que a solução numérica converge para $x_1 = 1$, $x_2 = -2$ e $x_3 = 1$, coincidindo com a solução exata do sistema.

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0,7000	-1,6000	0,6000
1	0,9600	-1,8600	0,9400
2	0,9780	-1,9800	0,9660
3	0,9994	-1,9888	0,9984
4	0,9979	-1,9996	0,9968
5	1,0002	-1,9989	1,0003
6	0,9998	-2,0001	0,9996
7	1,0001	-1,9999	1,0001
8	1,0000	-2,0000	1,0000
9	1,0000	-2,0000	1,0000
10	1,0000	-2,0000	1,0000

Tabela 4.1: Jacobi

Teorema 4.4. *Se A é uma matriz de diagonal estritamente dominante por linhas ou por colunas então o método de Jacobi converge para a solução do sistema linear $Ax = b$ qualquer que seja o vetor inicial $x^{(0)}$.*

Demonstração. :

$$\text{Como a matriz de iteração é } C = -M^{-1}N = D^{-1}(L + U)$$

$$c_{ij} = \begin{cases} 0 & \text{se } i = j \\ -\frac{a_{ij}}{a_{ii}} & \text{se } i \neq j \end{cases}$$

e lembrando que

$$\|C\|_{\infty} = \max_{1 \leq i \leq n} \sum_{j=1}^n |c_{ij}|$$

ao exigirmos que a norma do máximo seja inferior a 1, para convergir, isto significa

$$\max_{1 \leq i \leq n} \sum_{j=1}^n \left| -\frac{a_{ij}}{a_{ii}} \right| < 1$$

Logo, uma condição suficiente que nos garante isso, é

$$\sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}| < |a_{ii}| \quad , \quad i = 1, 2, \dots, n$$

isto é, que a matriz **A** tem **diagonal estritamente dominante por linhas**.

De forma análoga (usando uma norma semelhante à das colunas), podemos concluir que se

$$\sum_{\substack{i=1 \\ i \neq j}}^n |a_{ij}| < |a_{jj}| \quad , \quad j = 1, 2, \dots, n$$

isto é, se a matriz **A** tem **diagonal estritamente dominante por colunas**, então o método de Jacobi converge. □

Teorema 4.5. Se *A* é uma matriz simétrica e positiva definida então o método de Jacobi aplicado ao sistema linear $Ax = b$ converge para a solução do mesmo.

4.2.5 Método de Gauss-Seidel

Um método iterativo que converge cerca de duas vezes mais rápido que o método de Jacobi (pelo menos em várias aplicações) é o método de *Gauss-Seidel*, onde os valores de *x* são atualizados dentro de cada iteração, sem esperar pela próxima. Em outras palavras, obtido o valor de $x_j^{(k+1)}$ este é usado no lugar de $x_j^{(k)}$ no cálculo seguinte.

Seja o sistema de equações lineares $Ax = b$, com $A = L + D + U$, daí temos que:

$$Ax = b$$

$$(L + D + U)x = b$$

$$(L + D)x + Ux = b$$

$$\begin{aligned}
(L + D)x &= b - Ux \\
(L + D)^{-1}(L + D)x &= (L + D)^{-1}b - (L + D)^{-1}Ux \\
Ix &= (L + D)^{-1}b - (L + D)^{-1}Ux \\
x &= (L + D)^{-1}b - (L + D)^{-1}Ux
\end{aligned}$$

O método iterativo definido por:

$$x^{(k+1)} = (L + D)^{-1}b - (L + D)^{-1}Ux^{(k)} \quad \text{com } k = 0, 1, 2, \dots$$

onde $x^{(0)}$ é um vetor dado, é chamado de **Método de Gauss-Seidel**.

Este método é do tipo iterativo definido por $x^{(k+1)} = M^{-1}b - M^{-1}Nx^{(k)}$ com:

$$M = L + D \text{ e } N = U$$

Temos que:

$$\begin{aligned}
x^{(k+1)} &= (L + D)^{-1}b - (L + D)^{-1}Ux^{(k)} \\
(L + D)x^{(k+1)} &= (L + D)(L + D)^{-1}b - (L + D)(L + D)^{-1}Ux^{(k)} \\
Lx^{(k+1)} + Dx^{(k+1)} &= Ib - IUx^{(k)} \\
Dx^{(k+1)} &= b - Lx^{(k+1)} - Ux^{(k)} \\
D^{-1}Dx^{(k+1)} &= D^{-1}(b - Lx^{(k+1)} - Ux^{(k)}) \\
x^{(k+1)} &= D^{-1}(b - Lx^{(k+1)} - Ux^{(k)})
\end{aligned}$$

podemos escrever o método iterativo de Gauss-Seidel como:

$$\begin{aligned}
x_1^{(k+1)} &= \frac{1}{a_{11}} [b_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)} + a_{14}x_4^{(k)} + \dots + a_{1n}x_n^{(k)})] \\
x_2^{(k+1)} &= \frac{1}{a_{22}} [b_2 - (a_{21}x_1^{(k+1)} + a_{23}x_3^{(k)} + a_{24}x_4^{(k)} + \dots + a_{2n}x_n^{(k)})] \\
x_3^{(k+1)} &= \frac{1}{a_{33}} [b_3 - (a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)} + a_{34}x_4^{(k)} + \dots + a_{3n}x_n^{(k)})] \\
&\vdots \\
x_n^{(k+1)} &= \frac{1}{a_{nn}} [b_n - (a_{n1}x_1^{(k+1)} + a_{n2}x_2^{(k+1)} + a_{n3}x_3^{(k+1)} + \dots + a_{nn-1}x_{n-1}^{(k+1)})]
\end{aligned}$$

De uma forma geral $x_i^{(k+1)}$ pode ser obtido pela fórmula:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right), \quad i = 1, 2, 3, \dots, n$$

O algoritmo de Gauss-Seidel implementado no MATLAB é descrito no Algoritmo 4.7.

Algoritmo 4.7. Método de Gauss-Seidel no MATLAB

```
function [X,delta,Z] = gseidel(A,b,X0,tol,max)
%-----Comentário-----
% Implementa o método iterativo de Gauss-Seidel para determinar
% uma solução aproximada de Ax=b
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Executar %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% [X,delta] = gseidel(A,B,P,delta,max1)
% [X,delta,Z] = gseidel(A,B,P,delta,max1)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Entrada %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% A a matriz A do sistema
% b o vetor dos termos independentes
% X0 a solução inicial
% eps se abs(X(k)-X(k-1))< eps FIM !!!
% max número máximo de iterações
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gera %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% X o vetor com a solução
% delta a norma do vetor abs(X(k)-X(k-1))
% Z Matrix com todas as soluções (uma por linha)
%-----
n = length(b);
Xant = X0; % inicializa Xant
X=X0; % inicializa X
Z = X0'; % inicializa Z
for k=1:max, % iterar até max vezes
for j = 1:n, % para cada equação
if j==1
Sum = b(1) - A(1,2:n)*Xant(2:n);
elseif j==n
Sum = b(n) - A(n,1:n-1)*X(1:n-1);
else
Sum = b(j)-A(j,1:j-1)*X(1:j-1)-A(j,j+1:n)*Xant(j+1:n);
end
X(j) = Sum/A(j,j);
end
Z = [Z;X']; % armazena a história
delta = norm(abs(X-Xant),1);
if (delta<tol) break, end
Xant = X;
end
```

Exemplo 4.6. Seja o sistema linear

$$\begin{bmatrix} 5 & 1 & 1 \\ 3 & 4 & 1 \\ 3 & 3 & 6 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 5 \\ 6 \\ 0 \end{bmatrix} \quad \text{cuja solução exata é: } x_1 = 1, x_2 = 1 \text{ e } x_3 = -1$$

$$\text{Aplicando o método de Gauss-seidel } x_i^{(k+1)} = \frac{1}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right)$$

ao sistema acima, temos:

$$x_1^{(k+1)} = \frac{1}{a_{11}} [b_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)})] = \frac{1}{5} \cdot [5 - (x_2^{(k)} + x_3^{(k)})]$$

$$x_1^{(k+1)} = 1 - 0,2x_2^{(k)} - 0,2x_3^{(k)}$$

$$x_2^{(k+1)} = \frac{1}{a_{22}} [b_2 - (a_{21}x_1^{(k+1)} + a_{23}x_3^{(k)})] = \frac{1}{4} \cdot [6 - (3x_1^{(k)} + x_3^{(k)})]$$

$$x_2^{(k+1)} = 1,5 - 0,75x_1^{(k+1)} - 0,25x_3^{(k)}$$

$$x_3^{(k+1)} = \frac{1}{a_{33}} [b_3 - (a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)})] = \frac{1}{6} \cdot [0 - (3x_1^{(k+1)} + 3x_2^{(k+1)})]$$

$$x_3^{(k+1)} = -0,5x_1^{(k+1)} - 0,5x_2^{(k+1)}$$

$$\text{Tomando } x^{(0)} = [0 \ 0 \ 0]^T$$

$$\text{Para } k = 0, \text{ vamos calcular } x^{(1)} = [x_1^{(1)} \ x_2^{(1)} \ x_3^{(1)}]^T$$

$$x_1^{(1)} = 1 - 0,2x_2^{(0)} - 0,2x_3^{(0)} = 1 - 0,2 \cdot (0) - 0,2 \cdot (0) = 1$$

$$x_2^{(1)} = 1,5 - 0,75x_1^{(1)} - 0,25x_3^{(0)} = 1,5 - 0,75 \cdot (1) - 0,25 \cdot (0) = 0,75$$

$$x_3^{(1)} = -0,5x_1^{(1)} - 0,5x_2^{(1)} = -0,5 \cdot (1) - 0,5 \cdot (0,75) = -0,875$$

$$\text{Portanto, } x^{(1)} = [1 \ 0,75 \ -0,875]^T$$

$$\text{Para } k = 1, \text{ vamos calcular } x^{(2)} = [x_1^{(2)} \ x_2^{(2)} \ x_3^{(2)}]^T$$

$$x_1^{(2)} = 1 - 0,2x_2^{(1)} - 0,2x_3^{(1)} = 1 - 0,2 \cdot (0,75) - 0,2 \cdot (-0,875) = 1,025$$

$$x_2^{(2)} = 1,5 - 0,75x_1^{(2)} - 0,25x_3^{(1)} = 1,5 - 0,75 \cdot (1,025) - 0,25 \cdot (-0,875) = 0,95$$

$$x_3^{(2)} = -0,5x_1^{(2)} - 0,5x_2^{(2)} = -0,5 \cdot (1,025) - 0,5 \cdot (0,95) = -0,9875$$

$$\text{Portanto, } x^{(2)} = [1,025 \ 0,95 \ -0,9875]^T$$

Para $k = 2$, vamos calcular $x^{(3)} = [x_1^{(3)} \ x_2^{(3)} \ x_3^{(3)}]^T$

$$x_1^{(3)} = 1 - 0,2x_2^{(2)} - 0,2x_3^{(2)} = 1 - 0,2.(0,95) - 0,2.(-0,9875) = 1,0075$$

$$x_2^{(3)} = 1,5 - 0,75x_1^{(3)} - 0,25x_3^{(2)} = 1,5 - 0,75.(1,0075) - 0,25.(-0,9875) = 0,99125$$

$$x_3^{(3)} = -0,5x_1^{(3)} - 0,5x_2^{(3)} = -0,5.(1,0075) - 0,5.(0,99125) = -0,999375$$

$$\text{Portanto, } x^{(3)} = [1,0075 \ 0,99125 \ -0,999375]^T$$

As iterações do algoritmo de Gauss-Seidel se descrevem na Tabela 4.3. Os resultados mostram que a aproximação numérica converge para a solução exata $x_1 = 1$, $x_2 = 1$ e $x_3 = -1$.

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0	0	0
1	1.0000	0.7500	-0.8750
2	1.0250	0.9500	-0.9875
3	1.0075	0.9913	-0.9994
4	1.0016	0.9986	-1.0001
5	1.0003	0.9998	-1.0001
6	1.0000	1.0000	-1.0000
7	1.0000	1.0000	-1.0000
8	1.0000	1.0000	-1.0000
9	1.0000	1.0000	-1.0000
10	1.0000	1.0000	-1.0000

Tabela 4.2: Gauss-Seidel

Se a matriz é diagonal dominante ou simétrica positiva definida, então o método de Gauss-Seidel converge, como se enuncia nos seguintes teoremas. A prova é descrita em [7].

Teorema 4.6. *Se A é uma matriz de diagonal estritamente dominante por linhas ou por colunas então o método de Gauss-Seidel converge para a solução do sistema linear $Ax = b$ qualquer que seja o vetor inicial $x^{(0)}$*

Teorema 4.7. *Se A é uma matriz simétrica e positiva definida então o método de Gauss-Seidel aplicado ao sistema linear $Ax = b$ converge para a solução do mesmo.*

4.2.6 Métodos SOR

Apresentamos agora uma variante do método de Gauss-Seidel, denominado métodos SOR (ou também conhecida com o nome de relaxação sucessiva) que consiste em considerar um parâmetro real ω não nulo. No método SOR se escolhem as matrizes M e N como:

$$M = L + \frac{1}{\omega} \quad \text{e} \quad N = (1 - \frac{1}{\omega})D + U$$

A utilização deste parâmetro permite obter uma convergência mais rápida, havendo um valor ω que é o parâmetro optimal, no sentido em que minimiza o raio espectral da matriz $C = M^{-1}N$.

Portanto o método SOR é definido por:

$$x^{(k+1)} = (1 - \omega)x^{(k)} + \omega D^{-1}(b - Lx^{(k+1)} - Ux^{(k)}) \quad \text{com } k = 0, 1, 2, \dots$$

onde $x^{(0)}$ é um vetor dado.

O método SOR pode ser expresso pela formula,

$$x_i^{(k+1)} = (1 - \omega)x_i^{(k)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right) \quad , \quad i = 1, 2, 3, \dots, n$$

Notando que no caso em que $\omega = 1$ temos o método de Gauss-Seidel. A escolha $1 < \omega < 2$ caracteriza os métodos de sobre-relaxação, ao passo que os métodos de sub-relaxação são obtidos por valores $0 < \omega < 1$. A prática mostra que melhores resultados são obtidos na sobre-relaxação[4].

O algoritmo do Método SOR implementado no MATLAB é descrito no Algoritmo 4.8.

Algoritmo 4.8. Métodos SOR no MATLAB

```
function [X,delta,Z] = sor(A,b,X0,tol,max,w)
%-----Comentário-----
% Implementa o método iterativo de SOR para determinar
%uma solução aproximada de Ax=b
```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Executar %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% [X,delta] = sor(A,b,X0,tol,max,w)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Entrada %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% A a matriz A do sistema
% b o vetor coluna dos termos independentes
% X0 a solução inicial
% eps se abs(X(k)-X(k-1))< eps FIM !!!
% max número máximo de iterações
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Gera %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% X o vetor com a solução
% delta a norma do vetor abs(X(k)-X(k-1))
% Z Matrix com todas as soluções (uma por linha)
%-----
n = length(b);
Xant = X0; % inicializa Xant
X=X0; % inicializa X
Z = X0'; % inicializa Z
for k=1:max, % iterar até max vezes
for j = 1:n, % para cada equação
if j==1
Sum = b(1) - A(1,2:n)*Xant(2:n);
elseif j==n
Sum = b(n) - A(n,1:n-1)*X(1:n-1);
else
Sum = b(j)-A(j,1:j-1)*X(1:j-1)-A(j,j+1:n)*Xant(j+1:n);
end
X(j) =(1-w)*X(j)+(w/A(j,j))*Sum;
% X(j) = Sum/A(j,j);
end
Z = [Z;X']; % armazena a história
delta = norm(abs(X-Xant),1);
if (delta<eps) break, end
Xant = X; % iteração anterior
end

```

Exemplo 4.7. Trabalhando com o sistema linear do exemplo (4.6)

$$\begin{bmatrix} 5 & 1 & 1 \\ 3 & 4 & 1 \\ 3 & 3 & 6 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 5 \\ 6 \\ 0 \end{bmatrix} \quad \text{cuja solução exata é: } x_1 = 1, x_2 = 1 \text{ e } x_3 = -1$$

$$\text{Aplicando o métodos SOR } x_i^{(k+1)} = (1-\omega)x_i^{(k)} + \frac{\omega}{a_{ii}} \left(b_i - \sum_{j=1}^n a_{ij}x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij}x_j^{(k)} \right)$$

ao sistema acima com $\omega = 0.9$, temos:

$$x_1^{(k+1)} = (1-\omega)x_1^{(k)} + \frac{\omega}{a_{11}} [b_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)})] = 0,1x_1^{(k)} + \frac{0,9}{5} \cdot [5 - (x_2^{(k)} + x_3^{(k)})]$$

$$x_1^{(k+1)} = 0,1x_1^{(k)} + 0,9 - 0,18x_2^{(k)} - 0,18x_3^{(k)}$$

$$x_2^{(k+1)} = (1-\omega)x_2^{(k)} + \frac{\omega}{a_{22}} [b_2 - (a_{21}x_1^{(k+1)} + a_{23}x_3^{(k)})] = 0,1x_2^{(k)} + \frac{0,9}{4} \cdot [6 - (3x_1^{(k)} + x_3^{(k)})]$$

$$x_2^{(k+1)} = 0,1x_2^{(k)} + 1,35 - 0,675x_1^{(k+1)} - 0,225x_3^{(k)}$$

$$x_3^{(k+1)} = (1-\omega)x_3^{(k)} + \frac{\omega}{a_{33}} [b_3 - (a_{31}x_1^{(k+1)} + a_{32}x_2^{(k+1)})]$$

$$x_3^{(k+1)} = 0,1x_3^{(k)} + \frac{0,9}{6} \cdot [0 - (3x_1^{(k+1)} + 3x_2^{(k+1)})] = 0,1x_3^{(k)} - 0,45x_1^{(k+1)} - 0,45x_2^{(k+1)}$$

$$\text{Tomando } x^{(0)} = [0 \ 0 \ 0]^T$$

$$\text{Para } k = 0, \text{ vamos calcular } x^{(1)} = [x_1^{(1)} \ x_2^{(1)} \ x_3^{(1)}]^T$$

$$x_1^{(1)} = 0,1x_1^{(0)} + 0,9 - 0,18x_2^{(0)} - 0,18x_3^{(0)} = 0,1 \cdot (0) + 0,9 - 0,18 \cdot (0) - 0,18 \cdot (0)$$

$$x_1^{(1)} = 0,9$$

$$x_2^{(1)} = 0,1x_2^{(0)} + 1,35 - 0,675x_1^{(1)} - 0,225x_3^{(0)} = 0,1 \cdot (0) + 1,35 - 0,675 \cdot (0,9) - 0,225 \cdot (0)$$

$$x_2^{(1)} = 0,7425$$

$$x_3^{(1)} = 0,1x_3^{(0)} - 0,45x_1^{(1)} - 0,45x_2^{(1)} = 0,1 \cdot (0) - 0,45 \cdot (0,9) - 0,45 \cdot (0,7425)$$

$$x_3^{(1)} = -0,7391$$

$$\text{Portanto, } x^{(1)} = [0,9 \ 0,7425 \ -0,7391]^T$$

$$\text{Para } k = 1, \text{ vamos calcular } x^{(2)} = [x_1^{(2)} \ x_2^{(2)} \ x_3^{(2)}]^T$$

$$x_1^{(2)} = 0,1x_1^{(1)} + 0,9 - 0,18x_2^{(1)} - 0,18x_3^{(1)} =$$

$$= 0,1.(0,9) + 0,9 - 0,18.(0,7425) - 0,18.(-0,7391) = 0,9894$$

$$x_1^{(2)} = 0,9894$$

$$x_2^{(2)} = 0,1x_2^{(1)} + 1,35 - 0,675x_1^{(2)} - 0,225x_3^{(1)}$$

$$= 0,1.(0,7425) + 1,35 - 0,675.(0,9894) - 0,225.(-0,7391) = 0,9227$$

$$x_2^{(2)} = 0,9227$$

$$x_3^{(2)} = 0,1x_3^{(1)} - 0,45x_1^{(2)} - 0,45x_2^{(2)}$$

$$= 0,1.(-0,7391) - 0,45.(0,9894) - 0,45.(0,9227) = -0,9344$$

$$x_3^{(2)} = -0,9344$$

$$\text{Portanto, } x^{(2)} = [0,9894 \quad 0,9227 \quad -0,9344]^T$$

As iterações do métodos SOR são apresentados na Tabela 4.3. Os resultados mostram que a aproximação numérica se aproxima da solução exata $x_1 = 1$, $x_2 = 1$ e $x_3 = -1$.

SOR com $\omega=0,9$			
k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0	0	0
1	0.9000	0.7425	-0.7391
2	0.9894	0.9227	-0.9344
3	1.0010	0.9768	-0.9835
4	1.0013	0.9931	-0.9958
5	1.0006	0.9979	-0.9989
6	1.0002	0.9994	-0.9997
7	1.0001	0.9998	-0.9999
8	1.0000	0.9999	-1.0000
9	1.0000	1.0000	-1.0000
10	1.0000	1.0000	-1.0000

Tabela 4.3: Métodos SOR com $\omega=0,9$

Dentre os resultados de convergência dos métodos SOR tem-se:

Teorema 4.8. *(Condição Necessária) Para que haja convergência do método SOR, qualquer que seja a iterada inicial, é necessário que $0 < \omega < 2$*

Demonstração. [1] A matriz de iteração do método SOR é

$$C = -M^{-1}N$$

$$\begin{aligned} C &= \left(\frac{1}{\omega} D - L \right)^{-1} \left(\frac{1-\omega}{\omega} D + U \right) = \left[\frac{1}{\omega} D (I - \omega D^{-1} L) \right]^{-1} \left(\frac{1-\omega}{\omega} D + U \right) \\ &= (I - \omega D^{-1} L)^{-1} \omega D^{-1} \left(\frac{1-\omega}{\omega} D + U \right) \end{aligned}$$

ou,

$$C = (I - \omega D^{-1} L)^{-1} [(1 - \omega) I + \omega D^{-1} U]$$

se $\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_n$ são os autovalores de C, então

$$\det R = \lambda_1 \cdot \lambda_2 \cdot \lambda_3 \cdot \dots \cdot \lambda_n$$

Mas,

$$\begin{aligned} \det R &= \det \left\{ (I - \omega D^{-1} L)^{-1} [(1 - \omega) I + \omega D^{-1} U] \right\} \\ &= \det (I - \omega D^{-1} L)^{-1} \det [(1 - \omega) I + \omega D^{-1} U] \\ &= (1 - \omega)^n \end{aligned}$$

pois $I - \omega D^{-1} L$ é uma matriz triangular inferior com elementos iguais a 1 na diagonal e $(1 - \omega) I + \omega D^{-1} U$ é uma matriz triangular superior com elementos iguais a $1 - \omega$ na diagonal principal. Logo:

$$\lambda_1 \cdot \lambda_2 \cdot \lambda_3 \cdot \dots \cdot \lambda_n = (1 - \omega)^n$$

Em particular, pelo menos um dos autovalores λ_i de R deve satisfazer $|\lambda_i| \geq |1 - \omega|$

Mas, se o método SOR converge, devemos ter também $\lambda < 1$ para todo autovalor λ de R. Logo

$$|1 - \omega| < 1,$$

donde

$$0 < \omega < 2$$

□

Teorema 4.9. *(Condição Suficiente) Se A é simétrica e positiva definida, então o método converge para qualquer $0 < \omega < 2$; em particular como vimos anteriormente, neste caso o Método de Gauss-Seidel é convergente[4].*

4.3 Comparação entre Jacobi, Gauss-Seidel e SOR

Nesta seção apresentamos alguns exemplos de sistemas lineares onde utilizaremos o MATLAB para fazer simulação dos resultados calculados usando os métodos iterativos.

Exemplo 4.8. Trabalhando com o sistema do exemplo 4.5,

$$\begin{bmatrix} 10 & 2 & 1 \\ 1 & 5 & 1 \\ 2 & 3 & 10 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 7 \\ -8 \\ 6 \end{bmatrix}, \quad \text{cuja solução exata é: } x_1 = 1, x_2 = -2 \text{ e } x_3 = 1$$

Neste exemplo note que a matriz do sistema é diagonalmente dominante, o que nos assegura a convergência do sistema para qualquer que seja $x^{(0)}$. Consideremos $x^{(0)} = [0, 7 \quad -1, 6 \quad 0, 6]^T$.

As iterações dos algoritmos de Jacobi, Gauss-Seidel e SOR com $\omega = 1,5$ são apresentados na Tabela 4.4. Os resultados mostram o melhor desempenho do Método de Gauss-Seidel que na quarta iteração chega à solução exata, enquanto o método de Jacobi chega à solução exata na 8ª iteração. O método SOR com $\omega = 1,5$ teve o pior desempenho atingindo a solução exata após a 10ª iteração.

Na Tabela 4.5 se apresenta o método SOR para a sub-relaxação com $\omega = 0,5$ e sobre-relaxação com $\omega = 1,9$. Os resultados mostram diferentes graus de convergência, entretanto para este exemplo o valor ótimo ocorre para $\omega = 1,0$ coincidindo com o método de Gauss-Seidel. Para sistemas de grande porte verifica-se que o método SOR é preferível ao de Gauss-Seidel por acelerar a convergência. podemos verificar que o método de Gauss-Seidel converge mais rapidamente que o método de Jacobi e no caso do métodos SOR dependendo do valor de ω verificamos que podemos obter uma convergência mais rápida.

	Jacobi			Gauss-Seidel			SOR com $\omega = 1.5$		
k	$x_1^{(k)}_{jacobi}$	$x_2^{(k)}_{jacobi}$	$x_3^{(k)}_{jacobi}$	$x_1^{(k)}_{GS}$	$x_2^{(k)}_{GS}$	$x_3^{(k)}_{GS}$	$x_1^{(k)}_{SOR}$	$x_2^{(k)}_{SOR}$	$x_3^{(k)}_{SOR}$
0	0,7000	-1,6000	0,6000	0.7000	-1.6000	0.6000	0.7000	-1.6000	0.6000
1	0,9600	-1, 8600	0,9400	0.9600	-1.9120	0.9816	1.0900	-2.1070	1.2211
2	0,9780	-1,9800	0,9660	0.9842	-1.9932	1.0011	0.9539	-1.9990	0.9028
3	0,9994	-1,9888	0,9984	0.9985	-1.9999	1.0003	1.0373	-1.9825	1.0295
4	0.9979	-1.9996	0.9968	1.0000	-2.0000	1.0000	0.9717	-2.0091	0.9978
5	1.0002	-1.9989	1.0003	1.0000	-2.0000	1.0000	1.0172	-2.0000	0.9959
6	0.9998	-2.0001	0.9996	1.0000	-2.0000	1.0000	0.9920	-1.9964	1.0028
7	1.0001	-1.9999	1.0001	1.0000	-2.0000	1.0000	1.0025	-2.0034	0.9994
8	1.0000	-2.0000	1.0000	1.0000	-2.0000	1.0000	0.9999	-1.9981	0.9995
9	1.0000	-2.0000	1.0000	1.0000	-2.0000	1.0000	0.9996	-2.0007	1.0007
10	1.0000	-2.0000	1.0000	1.0000	-2.0000	1.0000	1.0003	-2.0000	0.9995

Tabela 4.4: Jacobi X Gauss-Seidel X SOR

SOR com $\omega = 0.5$				SOR com $\omega = 1.9$			
k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$
0	0.7000	-1.6000	0.6000	0	0.7000	-1.6000	0.6000
1	0.8300	-1.7430	0.7784	1	1.1940	-2.2817	1.4469
2	0.9004	-1.8394	0.8751	2	0.8476	-1.8583	0.5750
3	0.9404	-1.9012	0.9287	3	1.1641	-2.0284	1.3363
4	0.9639	-1.9399	0.9589	4	0.7992	-2.0259	0.7884
5	0.9780	-1.9636	0.9762	5	1.2308	-1.9840	1.0936
6	0.9865	-1.9781	0.9862	6	0.7684	-1.9620	0.9821
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
17	0.9999	-1.9999	1.0000	103	1.0000	-2.0001	1.0000
18	1.0000	-2.0000	1.0000	104	1.0000	-2.0000	1.0000

Tabela 4.5: Métodos SOR com $\omega = 0,5$ e $\omega = 1,9$

Exemplo 4.9. Seja o sistema linear

$$\begin{bmatrix} 10 & 1 & 2 & 3 & 4 \\ 1 & 9 & -1 & 2 & -3 \\ 2 & -1 & 7 & 3 & -5 \\ 3 & 2 & 3 & 12 & -1 \\ 4 & -3 & -5 & -1 & 15 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 12 \\ 27 \\ 14 \\ -17 \\ 12 \end{bmatrix},$$

cujas soluções exatas são: $x_1 = 1$, $x_2 = -2$, $x_3 = 3$, $x_4 = -2$ e $x_5 = 1$.

A matriz do sistema é simétrica definida positiva, o que assegura a convergência dos métodos iterativos. Na Tabela 4.6 se mostram as iterações de Jacobi e Gauss-Seidel e na Tabela 4.7 as iterações do método SOR com $\omega = 1, 3$. Em todos os casos, o chute inicial usado foi $x^{(0)} = [-1 \ 4 \ 0 \ 1 \ -2]^T$.

Neste exemplo a dimensão da matriz do sistema aumentou para $n = 5$. Os resultados mostram que o método de Gauss-Seidel foi superior que o método de Jacobi. O método de Gauss-Seidel chega à solução exata em 31 iterações, enquanto o método de Jacobi precisa de 52 iterações para chegar à solução exata. A convergência é acelerada usando o método SOR com $\omega = 1, 2$, na qual a solução exata é atingida em 19 iterações. Como comentado no exemplo anterior, esta aceleração da convergência do método SOR é ainda mais evidente para sistemas de grande porte, em este método supera amplamente ao método de Jacobi.

Jacobi						Gauss-Seidel					
k	$x_{1,jacobi}^{(k)}$	$x_{2,jacobi}^{(k)}$	$x_{3,jacobi}^{(k)}$	$x_{4,jacobi}^{(k)}$	$x_{5,jacobi}^{(k)}$	k	$x_{1,GS}^{(k)}$	$x_{2,GS}^{(k)}$	$x_{3,GS}^{(k)}$	$x_{4,GS}^{(k)}$	$x_{5,GS}^{(k)}$
0	-1.0000	4.0000	0.0000	1.0000	-2.0000	0	-1.0000	4.0000	0.0000	1.0000	-2.0000
1	1.3000	-3.7778	1.0000	-2.0000	1.9333	1	1.3000	-4.0333	-0.8048	-1.0349	-0.6906
2	1.2044	-1.9444	3.3270	-1.2009	-0.1022	2	2.3510	-3.3509	0.7999	-1.7035	-0.3440
3	1.1302	-2.5314	1.8198	-2.2340	1.1189	3	2.0238	-2.8721	1.4958	-1.8465	0.0614
4	1.3118	-2.0540	3.0721	-1.6390	0.4500	4	1.7175	-2.5938	1.9740	-1.9021	0.3544
5	1.1027	-2.2902	2.3556	-2.1328	0.9541	5	1.4934	-2.4058	2.2980	-1.9340	0.5576
6	1.2161	-2.0688	2.9534	-1.8200	0.6909	6	1.3381	-2.2777	2.5195	-1.9550	0.6971
7	1.0858	-2.1722	2.6305	-2.0567	0.9251	7	1.2315	-2.1901	2.6711	-1.9692	0.7927
8	1.1381	-2.0630	2.9216	-1.9066	0.8157	8	1.1585	-2.1301	2.7748	-1.9789	0.8581
9	1.0677	-2.1062	2.7799	-2.0198	0.9307	9	1.1085	-2.0891	2.8459	-1.9856	0.9028
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
50	1.0000	-2.0000	2.9999	-2.0000	1.0000	29	1.0001	-2.0000	2.9999	-2.0000	1.0000
51	1.0000	-2.0000	2.9999	-2.0000	1.0000	30	1.0000	-2.0000	2.9999	-2.0000	1.0000
52	1.0000	-2.0000	3.0000	-2.0000	1.0000	31	1.0000	-2.0000	3.0000	-2.0000	1.0000

Tabela 4.6: Exemplo 4.9 comparação Jacobi X Gauss-seidel

	SOR com $\omega = 1.2$				
k	$x_{1\text{ SOR}}^{(k)}$	$x_{2\text{ SOR}}^{(k)}$	$x_{3\text{ SOR}}^{(k)}$	$x_{4\text{ SOR}}^{(k)}$	$x_{5\text{ SOR}}^{(k)}$
0	-1.0000	4.0000	0.0000	1.0000	-2.0000
1	1.7600	-5.7013	-1.4094	-1.0649	-1.2205
2	3.0796	-3.2625	0.5683	-2.0509	-0.5011
3	2.0581	-2.7997	1.7260	-1.9152	0.2669
4	1.5115	-2.3940	2.3399	-1.9669	0.6270
5	1.2705	-2.2033	2.6677	-1.9847	0.8075
6	1.1369	-2.1030	2.8290	-1.9915	0.9023
7	1.0699	-2.0529	2.9130	-1.9958	0.9501
8	1.0357	-2.0269	2.9556	-1.9978	0.9745
9	1.0182	-2.0137	2.9773	-1.9989	0.9870
10	1.0093	-2.0070	2.9884	-1.9994	0.9934
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
17	1.0001	-2.0001	2.9999	-2.0000	0.9999
18	1.0000	-2.0000	2.9999	-2.0000	1.0000
19	1.0000	-2.0000	3.0000	-2.0000	1.0000

Tabela 4.7: Exemplo 4.9 SOR com $\omega = 1, 2$

Considerações Finais

Neste trabalho, foram apresentados Métodos Diretos e Iterativos para resolução de sistemas lineares. Estes métodos são amplamente usados em Modelagem Matemática como o estágio final na aproximação da solução do modelo em estudo. Muitos problemas são formulados em termos de equações diferenciais ordinárias ou parciais, ou ainda sistemas de equações diferenciais, os quais não tem solução analítica conhecida. Estes problemas são então discretizados por métodos como diferenças finitas, elementos finitos, volumes finitos, técnicas espectrais e outros. A resultante desta discretização são um sistema algébrico a ser resolvido, os quais podem ter milhares de equações. Um passo crucial então é resolver elas usando técnicas que otimizem o tempo de computo, minimizem o armazenamento das componentes da matriz e que controlem os erros de arredondamento. Verificamos que a escolha do método a ser utilizado depende das peculiaridades do problema.

Os métodos diretos são processos finitos e, portanto, teoricamente obtém a solução de qualquer sistema não-singular de equações. Já os métodos iterativos têm convergência assegurada apenas sob determinadas condições, podem ser mais rápidos e necessitar de menos memória do computador. Nos sistemas esparsos, isto é, naqueles em que é grande a porcentagem de coeficientes nulos, diminui o número de cálculos em métodos iterativos por iteração. Entretanto, nos métodos diretos, são realizados cálculos desnecessários, provocam o preenchimento da matriz e no processo de Eliminação (escalonamento) geram elementos não-nulos, onde originalmente tínhamos elementos nulos. Métodos iterativos não alteram a estrutura da matriz dos coeficientes, o método iterativo tem aí vantagem sobre os métodos diretos .

Quanto aos erros de arredondamento os métodos diretos apresentam sérios problemas, os quais para serem contornadas precisam de técnicas especiais ou de pivotamento nas linhas da matriz aumentada. Já os métodos iterativos apresentam menos erros de arredondamento, já que assegurada a convergência, ela independe da aproximação inicial.

Assim, somente o erros de arredondamento cometidos na última iteração é que irão afetar a solução encontrada, pois os erros de arredondamento não levarão à divergência do processo, ou à convergência para outro vetor que não seja a solução do sistema.

Os algoritmos e as simulações numéricas usadas neste trabalho foram elaborado no software MATLAB R2009b. Esse software apresenta uma grande versatilidade podendo ser utilizado inclusive por usuários que tenham pouco conhecimento em linguagens de programação. Isso se deve, porque as versões recentes do produto melhoram significativamente o ambiente interativo, incluindo facilidades gráficas de visualização e impressão, e principalmente pela facilidade de trabalhar com vetores e matrizes.

Para sistemas de ordem muito elevada deve-se preferir os métodos iterativos desde que sejam convergentes, principalmente o Método SOR que acelera a convergência. Entretanto, para sistemas de baixa ordem, pode ser preferível um método direto, pois tem menos restrições e não precisam de fornecer um chute inicial.

Referências Bibliográficas

- [1] BIEZUNER, Rodney Josué Biezuner. Notas de aula do curso *Tópicos em Análise: Autovalores do Laplaciano*, Instituto de Ciências Exatas (ICEx), UFMG, 2006.
- [2] CALLIOLI, Carlos Alberto; DOMINGUES. Hygino H.; COSTA. Roberto C. F. *Álgebra Linear e Aplicações*. 6ª edição. São Paulo: Atual Editora, 1990.
- [3] CASTILHO, J. Eduardo. *Cálculo Numérico*. Universidade Federal de Uberlândia Faculdade de Matemática, 2001.
- [4] CUNHA, Maria Cristina C. *Métodos Numéricos*. 2ª edição rev. e ampliada. Campinas-SP: Editora da UNICAMP, 2000.
- [5] FREITAS, Sergio Roberto. *Métodos Numéricos*. Departamento de Computação e Estatística Centro de Ciências Exatas e Tecnologia, UFMS, 2000.
- [6] IEZZI, Gelson; HAZZAN, Samuel. *Fundamentos de Matemática Elementar* Vol. 4. Editora Atual, 2ª edição, S. Paulo/SP, 1977.
- [7] KINCAID, David; CHENEY, Ward. *Análisis Numérico* Vol. Editora Adisson-Wesley Iberoamericana, USA, 1994.
- [8] LIMA, Rômulo. *Normas Matriciais associada à Norma vetorial l_∞* . Disponível em cardenas.webnode.com, 2010.
- [9] MENDES, Rafaela. *Normas Matriciais associada à Norma vetorial l_1* . Disponível em cardenas.webnode.com, 2010.
- [10] RUGGIERO, M. A. G. e Lopes, V. L. R. *Cálculo Numérico - Aspectos teóricos e computacionais*. Ed. McGraw-Hill, S. Paulo/SP. 1988.

- [11] SANTOS, Vitoriano Ruas de Barros. *Curso de Cálculo Numérico*. Editora LTC, 4ª edição, 2000.