

Introdução a Métodos Computacionais em EDOs

Notas de aula - Processos Iterativos Matriciais Lineares e
Resolução Iterativa de Sistemas Lineares

Prof. Yuri Dumaesq Sobral

Departamento de Matemática
Universidade de Brasília

2025

- Já aprendemos a analisar **processos iterativos escalares**, ou seja, aqueles processos iterativos definidos em termos de funções $g : \mathbb{R} \rightarrow \mathbb{R}$.
- Usamos estes processos para construir **métodos iterativos** para resolver **equações algébricas**. E se quiséssemos resolver um **sistema de equações** algébricas simultâneas?
- Como construir **métodos iterativos** para estes casos? Precisamos de **processos iterativos** associados simultaneamente a **várias variáveis**!
- Será que estes processos têm as mesmas propriedades que os processos iterativos escalares, e será que geram métodos **eficientes** quando aplicados a **sistemas muito grandes**?
- Precisamos estudar com um pouco mais de detalhe os **processos iterativos** construídos com **matrizes**! Vamos nos concentrar inicialmente apenas nos **processos lineares**.
(Lembrete: eles governam o **erro** de processos iterativos nas vizinhanças de seus pontos fixos!)

Relembrando:

- Considere uma função $g : \mathbb{R} \rightarrow \mathbb{R}$ tal que possamos construir, com base nela, o seguinte **processo iterativo escalar**:

$$x_{n+1} = g(x_n), \quad n = 0, 1, 2, 3, \dots$$

- O processo iterativo **escalar linear** mais geral que podemos construir é tomando $g(x) = ax + b$, isto é:

$$x_{n+1} = ax_n + b.$$

- Estudamos detalhadamente a **estabilidade** do seu **ponto fixo**

$$x^* = \frac{b}{1-a}.$$

- Se $|a| < 1$, então temos que x^* é **ponto fixo assintoticamente estável** do processo iterativo linear.
- Se $|a| > 1$, então temos que x^* é **ponto fixo instável** do processo iterativo linear.
- Se $a = 0$, então temos que $x^* = b$ é **ponto fixo estável (trivial)** do processo iterativo linear.

- Considere, agora, uma **matriz T quadrada de ordem m** , isto é, de tamanho $m \times m$. Considere os vetor $\mathbf{x}, \mathbf{b} \in \mathbb{R}^m$.
- O **processo iterativo matricial linear** mais geral que podemos construir é

$$\mathbf{x}_{n+1} = T\mathbf{x}_n + \mathbf{b}$$

- Vamos considerar o caso mais simples com $\mathbf{b} = \mathbf{0}$, isto é, $\mathbf{x}_{n+1} = T\mathbf{x}_n$, e vamos considerar \mathbf{x}_0 como valor inicial do processo. Neste caso, o processo iterativo se dá da seguinte maneira:

$$\mathbf{x}_1 = T\mathbf{x}_0, \quad \mathbf{x}_2 = T\mathbf{x}_1 = T(T\mathbf{x}_0) = T^2\mathbf{x}_0$$

$$\mathbf{x}_3 = T\mathbf{x}_2 = T^3\mathbf{x}_0, \quad \dots \quad \mathbf{x}_n = T\mathbf{x}_{n-1} = T^n\mathbf{x}_0.$$

- Portanto, o processo iterativo é determinado por **potências** da matriz T ! Muito parecido ao processo iterativo escalar linear $x_{n+1} = ax_n$, cuja solução é $x_n = a^n x_0$, com $a \in \mathbb{R}$.

- Apesar da semelhança com o caso **escalar**, não é possível encontrar um padrão óbvio em potências de matrizes. Vamos ver isto num exemplo.
- **Exemplo:** Considere $\mathbf{x}_{n+1} = \mathbf{T}\mathbf{x}_n$ com $m = 2$ e

$$\mathbf{T} = \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix}, \quad \mathbf{x}_n = \begin{pmatrix} x_n \\ y_n \end{pmatrix}, \quad \mathbf{x}_0 = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}.$$

Então, o processo iterativo é dado por:

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix} \Leftrightarrow \begin{cases} x_{n+1} = 0.6x_n + 0.2y_n \\ y_{n+1} = 0.2x_n + 0.6y_n \end{cases}$$

Assim, o processo será dado por:

$$\begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix},$$

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} = \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} =$$

$$\begin{aligned}
 \dots &= \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix}^2 \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} 0.4 & 0.24 \\ 0.24 & 0.4 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \\
 \begin{pmatrix} x_3 \\ y_3 \end{pmatrix} &= \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix}^3 \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \\
 &\dots = \begin{pmatrix} 0.288 & 0.224 \\ 0.224 & 0.288 \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \\
 &\vdots \\
 \begin{pmatrix} x_{10} \\ y_{10} \end{pmatrix} &= \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix}^{10} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} = \begin{pmatrix} 0.05373\dots & 0.05363\dots \\ 0.05363\dots & 0.05373\dots \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \\
 &\vdots \\
 \begin{pmatrix} x_{100} \\ y_{100} \end{pmatrix} &= \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix}^{100} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} \approx \begin{pmatrix} 1.02 \cdot 10^{-10} & 1.02 \cdot 10^{-10} \\ 1.02 \cdot 10^{-10} & 1.02 \cdot 10^{-10} \end{pmatrix} \begin{pmatrix} x_0 \\ y_0 \end{pmatrix}, \\
 &\vdots
 \end{aligned}$$

Conclusão: Parece que $\mathbf{x}_n \rightarrow \mathbf{0}$ com $n \rightarrow \infty$. E parece que $T^n \rightarrow \mathbf{0}$ (matriz nula, isto é, $\{T^n\}_{ij} \rightarrow 0$), com $n \rightarrow \infty$.

- Vemos, portanto, que **é muito difícil** determinar o comportamento do sistema a partir das **potências das matrizes**.
- Será que não seria possível encontrar uma **estrutura** similar àquela dos processos iterativos **escalares** lineares?
- A idéia, portanto, seria **tentar escrevermos** o resultado do processo iterativo **matricial** linear como

$$\mathbf{x}_n = \lambda^n \mathbf{v}, \quad \mathbf{v} \neq \mathbf{0},$$

com algum $\lambda \in \mathbb{R}$ e algum vetor $\mathbf{v} \in \mathbb{R}^m$. Mas será que isto funciona?

- Vamos testar:

$$\mathbf{x}_{n+1} = T\mathbf{x}_n \Leftrightarrow \lambda^{n+1}\mathbf{v} = T\lambda^n\mathbf{v} \Leftrightarrow \lambda\mathbf{v} = T\mathbf{v}$$

- Portanto, se encontrarmos λ , \mathbf{v} que satisfaçam $T\mathbf{v} = \lambda\mathbf{v}$, a solução do processo iterativo será $\mathbf{x}_n = \lambda^n\mathbf{v}$ e conseguiremos determinar o que acontece com $n \rightarrow \infty$ de maneira mais trivial.

- Os números λ que satisfazem $T\mathbf{v} = \lambda\mathbf{v}$ são chamados de **autovalores** da matriz T .
- os vetores \mathbf{v} que que satisfazem $T\mathbf{v} = \lambda\mathbf{v}$ são chamados de **autovetores** da matriz T .
- Os **autovalores** são encontrados a partir do **polinômio característico** de T :

$$T\mathbf{v} = \lambda\mathbf{v} \Leftrightarrow T\mathbf{v} - \lambda\mathbf{v} = \mathbf{0} \Leftrightarrow (T - \lambda I)\mathbf{v} = \mathbf{0} \Leftrightarrow \det(T - \lambda I) = 0.$$

- Como o **polinômio característico** tem grau m e todos seus coeficientes são \mathbb{R} , uma matriz T tem m autovalores complexos, com alguns podendo ser valores repetidos (**multiplicidade!**).
- Então, para cada par λ, \mathbf{v} , o processo iterativo terá seu comportamento determinado por $\mathbf{x}_n = \lambda^n \mathbf{v}$. Mas e o caso geral? Vamos apenas considerar um caso geral mais simples:

- Se a matriz T for tal que seus m autovetores $\mathbf{v}_1 \dots \mathbf{v}_m$ forem base de \mathbb{R}^m , isto é, se eles forem linearmente independentes e se qualquer vetor de \mathbb{R}^m puder ser escrito como uma combinação linear deles, então

$$\mathbf{x}_n = c_1 \lambda_1^n \mathbf{v}_1 + c_2 \lambda_2^n \mathbf{v}_2 + \dots + c_m \lambda_m^n \mathbf{v}_m.$$

e as constantes $c_1 \dots c_m \in \mathbb{R}$ serão unicamente determinadas pelo valor de \mathbf{x}_0 .

- Vamos voltar ao exemplo anterior e reescrevê-lo de acordo com o que aprendemos agora:

$$\begin{pmatrix} x_{n+1} \\ y_{n+1} \end{pmatrix} = \begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} \begin{pmatrix} x_n \\ y_n \end{pmatrix}.$$

Então, para determinarmos os autovalores da matriz T :

$$\det \left(\begin{pmatrix} 0.6 & 0.2 \\ 0.2 & 0.6 \end{pmatrix} - \begin{pmatrix} \lambda & 0 \\ 0 & \lambda \end{pmatrix} \right) = 0 \Leftrightarrow \det \begin{pmatrix} 0.6 - \lambda & 0.2 \\ 0.2 & 0.6 - \lambda \end{pmatrix} = 0$$

$$\Leftrightarrow (0.6-\lambda)^2-0.04=0 \Leftrightarrow \lambda^2-1.2\lambda-0.32=0 \Leftrightarrow \lambda_1=0.8, \lambda_2=0.4$$

- O autovetor associado ao autovalor $\lambda_1=0.8$ será dado por:

$$(T - \lambda_1 I) \mathbf{v}_1 = \mathbf{0} \Leftrightarrow (T - 0.8I) \mathbf{v}_1 = \mathbf{0} \Leftrightarrow$$

$$\begin{pmatrix} 0.6 - 0.8 & 0.2 \\ 0.2 & 0.6 - 0.8 \end{pmatrix} \begin{pmatrix} v_{1x} \\ v_{1y} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \Leftrightarrow \begin{pmatrix} -0.2 & 0.2 \\ 0.2 & -0.2 \end{pmatrix} \begin{pmatrix} v_{1x} \\ v_{1y} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

$$\Leftrightarrow v_{1x} = v_{1y} \Leftrightarrow \mathbf{v}_1 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

- O autovetor associado ao autovalor $\lambda_2=0.4$ será dado por:

$$\mathbf{v}_2 = \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

- Desta forma, podemos finalmente escrever a solução geral do processo iterativo:

$$\mathbf{x}_n = c_1 \cdot 0.8^n \begin{pmatrix} 1 \\ 1 \end{pmatrix} + c_2 \cdot 0.4^n \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

- Conhecendo o valor de \mathbf{x}_0 , podemos calcular c_1 e c_2 :

$$\mathbf{x}_n = 0.8^n \frac{x_0 + y_0}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} + 0.4^n \frac{y_0 - x_0}{2} \begin{pmatrix} -1 \\ 1 \end{pmatrix}.$$

- E, agora, podemos ver claramente que tanto 0.8^n quanto 0.4^n tendem a 0 quando $n \rightarrow \infty$, isto é, $\mathbf{x}_n \rightarrow \mathbf{0}$ $n \rightarrow \infty$.
- Vemos, também, que a convergência para o ponto fixo assintoticamente estável $\mathbf{x}^* = \mathbf{0}$ é dominada pelo maior autovalor $\lambda_1 = 0.8$, pois, com $n \rightarrow \infty$:

$$\mathbf{x}_n = 0.8^n \left(\underbrace{\frac{x_0 + y_0}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix}}_{\text{constante}} + \underbrace{0.5^n \frac{y_0 - x_0}{2} \begin{pmatrix} -1 \\ 1 \end{pmatrix}}_{\rightarrow 0} \right) \approx 0.8^n \frac{x_0 + y_0}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \rightarrow \mathbf{0}.$$

- Finalmente, vemos que quando $\mathbf{x}_n \rightarrow \mathbf{0}$, seu módulo $|\mathbf{x}_n| \rightarrow 0$. Estas condições são **equivalentes**. Então, vamos usar com certa frequência esta condição ao invés da condição vetorial.
- Com a intuição que ganhamos com os resultados deste exemplo, podemos agora sedimentar os seguintes conceitos fundamentais:
- O **ponto fixo** $\mathbf{x}^* = \mathbf{0}$ será **assintoticamente estável** se e somente se **todos os autovalores** λ_i da matrix T tiverem módulo menor que um, isto é, $|\lambda_i| < 1$. Se algum tiver módulo **maior** (ou **igual**) que 1, o ponto fixo do processo iterativo será **instável** (**estável**).
- Se uma matrix T é tal que todos seus autovalores, $|\lambda_i| < 1$, então $T^n \rightarrow \mathbf{0}$ com $n \rightarrow \infty$, isto é, todos os seus elementos $\{T^n\}_{ij} \rightarrow 0$ com $n \rightarrow \infty$. Dizemos que T é uma **matriz convergente**.

- O autovalor λ_i (complexo ou real) **de maior módulo** da matriz T , é chamado de **raio espectral** de T , denotado por $\rho(T)$.

$$\rho(T) = \max\{|\lambda_1|, |\lambda_2|, \dots, |\lambda_m|\}$$

- Portanto, uma matriz T será convergente se e somente se seu raio espectral $\rho(T) < 1$.
- O **raio espectral** de T determina quão rápido o processo vai convergir (ou divergir) para seu ponto fixo, pois:

$$\begin{aligned} |\mathbf{x}_n| &= |c_1 \lambda_1^n \mathbf{v}_1 + c_2 \lambda_2^n \mathbf{v}_2 + \dots + c_m \lambda_m^n \mathbf{v}_m| \\ &\leq |c_1 \lambda_1^n \mathbf{v}_1| + |c_2 \lambda_2^n \mathbf{v}_2| + \dots + |c_m \lambda_m^n \mathbf{v}_m| \quad (\text{desigualdade triangular}) \\ &\leq |\lambda_1^n| |c_1 \mathbf{v}_1| + |\lambda_2^n| |c_2 \mathbf{v}_2| + \dots + |\lambda_m^n| |c_m \mathbf{v}_m| \\ &\leq |\lambda_1^n| |c_1| |\mathbf{v}_1| + |\lambda_2^n| |c_2| |\mathbf{v}_2| + \dots + |\lambda_m^n| |c_m| |\mathbf{v}_m| \end{aligned}$$

Como $\rho(T) = \max\{|\lambda_1|, |\lambda_2|, \dots, |\lambda_m|\}$, então:

$$\begin{aligned}
 |\mathbf{x}_n| &\leq |\lambda_1|^n |c_1| |\mathbf{v}_1| + |\lambda_2|^n |c_2| |\mathbf{v}_2| + \cdots + |\lambda_m|^n |c_m| |\mathbf{v}_m| \\
 &\leq \rho(T)^n \underbrace{\left(|c_1| |\mathbf{v}_1| + |c_2| |\mathbf{v}_2| + \cdots + |c_m| |\mathbf{v}_m| \right)}_{K \in \mathbb{R}} = K \rho(T)^n
 \end{aligned}$$

- Portanto, se $\rho(T) < 1$, temos que $|\mathbf{x}_n| \leq K \rho(T)^n \rightarrow 0$, quando $n \rightarrow \infty$, $\mathbf{x}^* = \mathbf{0}$ é um ponto fixo assintoticamente estável e T é uma matriz convergente. Caso contrário, se $\rho(T) \geq 1$, $\mathbf{x}^* = \mathbf{0}$ é um ponto fixo instável ou apenas estável.
- Quanto menor for $\rho(T)$, mais rapidamente o processo iterativo converge.
- Portanto, vamos querer construir processos iterativos com matrizes com o menor $\rho(T)$ possível! (lembra a escolha de $|g'(x^*)|$ no caso de processos iterativos escalares!)
- PROBLEMA:** calcular autovalores é caro, principalmente para sistemas grandes. Teremos que ser espertos!

- Agora que conhecemos um pouco mais sobre os **processos iterativos matriciais** e sua **convergência**, podemos construir métodos iterativos para resolver sistemas de equações lineares.
- Para tal, vamos nos concentrar em problemas do tipo

$$A\mathbf{x} = \mathbf{b},$$

em que a matriz dos coeficientes A é **invertível**, ou seja, existe A^{-1} (**matriz inversa**), e o sistema possui **uma única solução**.

- Vamos tentar, portanto, transformar este sistema em um processo iterativo do tipo

$$\mathbf{x}_{n+1} = T\mathbf{x}_n + \mathbf{c},$$

em que T é construída a partir da matriz dos coeficientes A e o vetor \mathbf{c} a partir de A e de \mathbf{b} . Queremos que a solução do sistema seja o ponto fixo \mathbf{x}^* deste processo iterativo, isto é

$$A\mathbf{x}^* = \mathbf{b} \quad \Leftrightarrow \quad \mathbf{x}^* = T\mathbf{x}^* + \mathbf{c}.$$

- A idéia, portanto, é construir os processos iterativos de tal forma que \mathbf{x}^* seja um ponto fixo **assintoticamente estável**.
- Para isto, precisamos analisar como o **erro** do processo iterativo evolui ao longo das iterações:

$$\mathbf{e}_{n+1} = \mathbf{x}_{n+1} - \mathbf{x}^* = (T\mathbf{x}_n + \mathbf{c}) - (T\mathbf{x}^* + \mathbf{c}) = T(\mathbf{x}_n - \mathbf{x}^*) = T\mathbf{e}_n \Leftrightarrow$$

$$\mathbf{e}_{n+1} = T\mathbf{e}_n.$$

- Portanto, o **erro** do processo iterativo satisfaz um **processo iterativo matricial linear**, e já sabemos que sua solução geral é

$$\mathbf{e}_n = T^n \mathbf{e}_0.$$

- Então, teremos $\mathbf{e}_n \rightarrow \mathbf{0}$ com $n \rightarrow \infty$ **se e somente se** a matriz T for **convergente**, isto é, $T^n \rightarrow \mathbf{0}$ com $n \rightarrow \infty$. Para isto, precisamos que seu **raio espectral** $\rho(T) < 1$.

- Desta forma, devemos ter cuidado com a maneira de construir os processos iterativos!
- Mas como construir estes métodos a partir do sistema original? Vamos ver alguns exemplos.
- Vamos começar com um exemplo de um sistema típico com três equações e três incógnitas:

$$\begin{cases} a_{11}x + a_{12}y + a_{13}z = b_1 \\ a_{21}x + a_{22}y + a_{23}z = b_2 \\ a_{31}x + a_{32}y + a_{33}z = b_3 \end{cases}$$

- Uma maneira **ingênua** de gerar um processo iterativo é **isolando** uma variável em cada uma das equações. Por exemplo, se $a_{ii} \neq 0$:

$$\begin{cases} a_{11}x = b_1 - a_{12}y - a_{13}z \\ a_{22}y = b_2 - a_{21}x - a_{23}z \\ a_{33}z = b_3 - a_{31}x - a_{32}y \end{cases} \Leftrightarrow \begin{cases} x = \frac{b_1}{a_{11}} - \frac{a_{12}}{a_{11}}y - \frac{a_{13}}{a_{11}}z \\ y = \frac{b_2}{a_{22}} - \frac{a_{21}}{a_{22}}x - \frac{a_{23}}{a_{22}}z \\ z = \frac{b_3}{a_{33}} - \frac{a_{31}}{a_{33}}x - \frac{a_{32}}{a_{33}}y \end{cases}$$

$$\begin{cases} x_{n+1} = \frac{b_1}{a_{11}} - \frac{a_{12}}{a_{11}} y_n - \frac{a_{13}}{a_{11}} z_n \\ y_{n+1} = \frac{b_2}{a_{22}} - \frac{a_{21}}{a_{22}} x_n - \frac{a_{23}}{a_{22}} z_n \\ z_{n+1} = \frac{b_3}{a_{33}} - \frac{a_{31}}{a_{33}} x_n - \frac{a_{32}}{a_{33}} y_n \end{cases}$$

- Este sistema pode ser escrito em forma matricial da seguinte forma:

$$\underbrace{\begin{pmatrix} x_{n+1} \\ y_{n+1} \\ z_{n+1} \end{pmatrix}}_{\mathbf{x}_{n+1}} = \underbrace{\begin{pmatrix} 0 & -\frac{a_{12}}{a_{11}} & -\frac{a_{13}}{a_{11}} \\ -\frac{a_{21}}{a_{22}} & 0 & -\frac{a_{23}}{a_{22}} \\ -\frac{a_{31}}{a_{33}} & -\frac{a_{32}}{a_{33}} & 0 \end{pmatrix}}_T \underbrace{\begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix}}_{\mathbf{x}_n} + \underbrace{\begin{pmatrix} \frac{b_1}{a_{11}} \\ \frac{b_2}{a_{22}} \\ \frac{b_3}{a_{33}} \end{pmatrix}}_c.$$

- Notem que esta é exatamente a forma que temos em mente para o processo iterativo. **PORÉM...** Já fica clara uma grande limitação destes métodos: é muito difícil (ou impossível) determinar **a priori** se T será ou não uma **matriz convergente**.

- Vamos generalizar o método **ingênuo** que discutimos acima. Consideremos um sistema de M equações e M incógnitas dado por $A\mathbf{x} = \mathbf{b}$.
- Vamos decompor a matriz dos coeficientes A da seguinte maneira:

$$A = L + D + U,$$

isto é, decomparamos a matriz A em suas partes **estritamente triangular inferior** L , **diagonal** D e **estritamente triangular superior** U , e reescrevemos o sistema como:

$$A\mathbf{x} = \mathbf{b} \Leftrightarrow (L + D + U)\mathbf{x} = \mathbf{b} \Leftrightarrow D\mathbf{x} + (L + U)\mathbf{x} = \mathbf{b} \Leftrightarrow$$

$$D\mathbf{x} = -(L + U)\mathbf{x} + \mathbf{b} \Leftrightarrow \mathbf{x}_{n+1} = -D^{-1}(L + U)\mathbf{x}_n + D^{-1}\mathbf{b}.$$

- Portanto, $T_J = -D^{-1}(L + U)$ e $\mathbf{c} = D^{-1}\mathbf{b}$. Este processo iterativo é chamado de **Método de Gauss-Jacobi**.

- Um processo iterativo um pouco mais sofisticado pode ser intuitivamente determinado a partir do **Método de Gauss-Jacobi**.
- No processo iterativo do **Método de Gauss-Jacobi**, cada nova aproximação $n + 1$ de uma variável é determinada apenas a partir das antigas aproximações n . Por exemplo:

$$\begin{cases} x_{n+1} = \frac{b_1}{a_{11}} - \frac{a_{12}}{a_{11}} y_n - \frac{a_{13}}{a_{11}} z_n \\ y_{n+1} = \frac{b_2}{a_{22}} - \frac{a_{21}}{a_{22}} x_n - \frac{a_{23}}{a_{22}} z_n \\ z_{n+1} = \frac{b_3}{a_{33}} - \frac{a_{31}}{a_{33}} x_n - \frac{a_{32}}{a_{33}} y_n \end{cases}$$

- A priori, já teríamos uma **melhor estimativa** para a variável x , calculada na primeira equação, que poderia ter sido utilizada nas equações subsequentes.
- De fato, o processo iterativo para o sistema acima poderia ser dado por:

$$\begin{cases} x_{n+1} = \frac{b_1}{a_{11}} - \frac{a_{12}}{a_{11}} y_n - \frac{a_{13}}{a_{11}} z_n \\ y_{n+1} = \frac{b_2}{a_{22}} - \frac{a_{21}}{a_{22}} x_{n+1} - \frac{a_{23}}{a_{22}} z_n \\ z_{n+1} = \frac{b_3}{a_{33}} - \frac{a_{31}}{a_{33}} x_{n+1} - \frac{a_{32}}{a_{33}} y_{n+1} \end{cases}$$

- Este processo também pode ser escrito em uma forma matricial. Note que os termos que **já são conhecidos** na nova iteração $n + 1$ são aqueles que correspondem aos coeficientes abaixo da diagonal principal de A (associados à matriz L).
- Considerando a mesma decomposição da matriz A feita anteriormente, temos:

$$A\mathbf{x} = \mathbf{b} \Leftrightarrow (L + D + U)\mathbf{x} = \mathbf{b} \Leftrightarrow (L + D)\mathbf{x} + U\mathbf{x} = \mathbf{b} \Leftrightarrow$$

$$(L + D)\mathbf{x} = -U\mathbf{x} + \mathbf{b} \Leftrightarrow \mathbf{x}_{n+1} = -(L + D)^{-1}U\mathbf{x}_n + (L + D)^{-1}\mathbf{b}.$$

- Portanto, $T_S = -(L + D)^{-1}U$ e $\mathbf{c} = (L + D)^{-1}\mathbf{b}$. Este processo iterativo é chamado de **Método de Gauss-Seidel**.
- Tal como no **Método de Gauss-Jacobi**, no **Método de Gauss-Seidel** não temos como garantir a priori que a matriz T_S será convergente.

- Além disto, apesar de parecer ser **melhor**, não há garantias de que o **Método de Gauss-Seidel** será de fato **melhor**, isto é, que ele convergirá quando o **Método de Gauss-Jacobi** divergir, e que será mais rápido que o **Método de Gauss-Jacobi**.
- Como vimos antes, não podemos afirmar, a priori, se as matrizes T_J e T_S serão convergentes. Para isto, precisamos determinar seu **raio espectral**, o que é muito caro.
- Um resultado que pode facilitar um pouco a análise é o seguinte: é possível mostrar, para qualquer matriz A , que

$$\rho(A) \leq \max_{i=1,\dots,M} \sum_{j=1}^M |a_{ij}| = S_{\ell}^{Max}.$$

- Isto é, o **raio espectral** é limitado pelo **maior valor** da **soma dos valores absolutos** de **todos os elementos de uma linha**, S_{ℓ}^{Max} .

- Um caso particular bem conhecido, e bastante frequente em problemas reais, é o caso de sistemas lineares cuja matriz de coeficientes A é **estritamente diagonalmente dominante**.
- Nestas matrizes, o **valor absoluto** do **elemento da diagonal**, $|a_{ii}|$, é **estritamente maior** que a **soma dos valores absolutos** de **todos os outros elementos da linha**, S_i^p .
- Exemplo:

$$A = \begin{pmatrix} 10 & 0 & -1 & 2 & 4 \\ -2 & -18 & 2 & 1 & -5 \\ 0 & 2 & 6 & 1 & -1 \\ -1 & -2 & -1 & -7 & 1 \\ 0 & 2 & 1 & 0 & 14 \end{pmatrix} \quad \begin{array}{l} S_1^p = |0| + |-1| + |2| + |4| = 7 < |10| = |a_{11}| \\ S_2^p = |-2| + |2| + |1| + |-5| = 10 < |-18| = |a_{22}| \\ S_3^p = |0| + |2| + |1| + |-1| = 4 < |6| = |a_{33}| \\ S_4^p = |-1| + |-2| + |-1| + |1| = 5 < |-7| = |a_{44}| \\ S_5^p = |0| + |2| + |1| + |0| = 3 < |14| = |a_{55}| \end{array}$$

- Neste caso, por causa da divisão pelos elementos da diagonal nos Métodos de Gauss-Jacobi e de Gauss-Seidel, é fácil perceber que $S_\ell^{Max} < 1$ (**estritamente menor que 1**) e, portanto, tanto $\rho(T_J) < 1$, como $\rho(T_S) < 1$.

- Note que, tanto no Método de Gauss-Jacobi, como no Método de Gauss-Seidel, a matriz T tem **elementos nulos** na **diagonal**. Isto quer dizer que valores das iterações anteriores da própria variável **não entram** no cálculo das próximas iterações.
- Podemos, então, pensar similarmente ao que feito no Método de Gauss-Seidel e tentar **melhorar a convergência** dos métodos.
- Para isto, vamos considerar, como anteriormente, a decomposição da matriz A da seguinte forma:

$$A\mathbf{x} = \mathbf{b} \Leftrightarrow (\mathbf{L} + \mathbf{D} + \mathbf{U})\mathbf{x} = \mathbf{b}.$$

- Vamos reescrever, agora, esta decomposição da seguinte maneira:

$$(\mathbf{L} + \mathbf{D} + \mathbf{U})\mathbf{x} = \mathbf{b} \Leftrightarrow (\mathbf{L} + (1 + \alpha - \alpha)\mathbf{D} + \mathbf{U})\mathbf{x} = \mathbf{b} \Leftrightarrow$$

$$[\mathbf{L} + \alpha\mathbf{D} + (1 - \alpha)\mathbf{D} + \mathbf{U}]\mathbf{x} = \mathbf{b}$$

- Vamos pensar em construir um método com base no Método de Gauss-Seidel. Portanto:

$$[L + \alpha D + (1 - \alpha)D + U]x = b \Leftrightarrow (L + \alpha D)x = -[(1 - \alpha)D + U]x + b,$$

e, normalmente, dividimos tudo por α e tomamos $\omega = 1/\alpha$, obtendo o seguinte processo iterativo:

$$x_{n+1} = (\omega L + D)^{-1}[(1 - \omega)D - \omega U]x_n + \omega(\omega L + D)^{-1}b.$$

- Portanto, a matriz que define este processo iterativo é

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

e vemos que ela depende de um parâmetro livre ω . Será que podemos escolher adequadamente ω para minimizar (ou pelo menos diminuir) o seu raio espectral $\rho(T_\omega)$?

- Precisamos associar, de alguma maneira, $\rho(T_\omega)$ com ω .
- Vamos calcular o determinante de T_ω :

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

- Como L e U são matrizes estritamente triangulares, os elementos de suas diagonais são todos zero. Portanto, os determinantes das matrizes $\omega L + D$ e $(1 - \omega)D - \omega U$ são determinados apenas por D .
- Usando as propriedades do determinante de uma matriz, temos, então:

$$\det(T_\omega) = \det\left((\omega L + D)^{-1}\right) \det((1 - \omega)D - \omega U) = \frac{1}{\det(D)} (1 - \omega)^M \det(D)$$

- E, portanto, $\det(T_\omega) = (1 - \omega)^M$.
- Por outro lado, assumindo que T_ω tem M autovalores e seus M autovetores forem base de \mathbb{R}^M (já vimos esta hipótese antes!), é possível mostrar que

$$\det(T_\omega) = \lambda_1 \cdot \lambda_2 \cdot \lambda_3 \cdot \dots \cdot \lambda_M,$$

isto é, o determinante de T_ω é dado pelo produto de seus M autovalores.

- Comparando os dois resultados:

$$|\det(T_\omega)| = |(1 - \omega)^M| = |\lambda_1 \cdot \lambda_2 \cdot \lambda_3 \cdot \dots \cdot \lambda_M| \leq \rho(T_\omega)^M.$$

- Como precisamos que $\rho(T_\omega) < 1$ para que T_ω seja convergente, então:

$$|(1 - \omega)^M| \leq \rho(T_\omega)^M < 1 \Leftrightarrow |1 - \omega| < 1 \Leftrightarrow$$

$$-1 < 1 - \omega < 1 \Leftrightarrow 0 < \omega < 2.$$

- Assim, **não podemos escolher qualquer valor** para ω !
- O parâmetro ω é chamado de **parâmetro de relaxação** do método e dizemos que o método é:
 - sub-relaxado**, se $0 < \omega < 1$
 - Gauss-Seidel**, se $\omega = 1$
 - sobre-relaxado**, se $1 < \omega < 2$
- Se $\omega < 0$ ou $\omega > 2$, claramente o método diverge.
- Normalmente, os métodos **sobre-relaxados** são os mais eficientes!
- Para algumas **matrizes especiais**, é possível calcular o **valor ótimo** de ω para que $\rho(T_\omega)$ seja **mínimo**!
- Exemplo:** matrizes simétricas, tridiagonais por blocos, positivas definidas ($\mathbf{x}^T \cdot \mathbf{A} \cdot \mathbf{x} > 0 \quad \forall \mathbf{x} \neq \mathbf{0}$) são tais que:

$$\rho(T_J)^2 = \rho(T_S), \quad \omega_{ot} = \frac{2}{1 + \sqrt{1 - \rho(T_S)}} \quad \text{e} \quad \rho(T_\omega) = \omega_{ot} - 1.$$

- Então, se tivermos uma matriz A tal como acima com $\rho(T_J) = 0.99$ (**péssimo!**), então teremos que

$$\rho(T_S) = \rho(T_J)^2 = 0.99^2 = 0.9801$$

$$\omega_{ot} = \frac{2}{1 + \sqrt{1 - 0.9801}} = 1.7527 \quad \text{e} \quad \rho(T_\omega) = 0.7527.$$

- Fazendo as contas, vemos que $\rho(T_\omega) \approx \rho(T_S)^{14} \approx \rho(T_J)^{28}!!!!$
O método sobre-relaxado é 28 vezes mais rápido que o método de Gauss-Jacobi para estas matrizes!
- Os métodos **sobre-relaxados** costumam ser chamados de **Métodos SOR (successive over-relaxation)** e, de fato, podemos reescrevê-los de uma maneira bem fácil e intuitiva!

- Relembrando o método de Gauss-Seidel em sua forma matricial:

$$\mathbf{x}_{n+1} = -(\mathbf{L} + \mathbf{D})^{-1} \mathbf{U} \mathbf{x}_n + (\mathbf{L} + \mathbf{D})^{-1} \mathbf{b} \Leftrightarrow (\mathbf{L} + \mathbf{D}) \mathbf{x}_{n+1} = -\mathbf{U} \mathbf{x}_n + \mathbf{b}.$$

- E, portanto, podemos escrevê-lo como:

$$\mathbf{D} \mathbf{x}_{n+1} = -\mathbf{U} \mathbf{x}_n + \mathbf{b} - \mathbf{L} \mathbf{x}_{n+1} \Leftrightarrow \mathbf{x}_{n+1}^{\text{Sei}} = \mathbf{D}^{-1} \left(-\mathbf{U} \mathbf{x}_n + \mathbf{b} - \mathbf{L} \mathbf{x}_{n+1} \right).$$

- Por outro lado, voltando ao método SOR:

$$\mathbf{x}_{n+1} = (\omega \mathbf{L} + \mathbf{D})^{-1} [(1 - \omega) \mathbf{D} - \omega \mathbf{U}] \mathbf{x}_n + \omega (\omega \mathbf{L} + \mathbf{D})^{-1} \mathbf{b} \Leftrightarrow$$

$$(\omega \mathbf{L} + \mathbf{D}) \mathbf{x}_{n+1} = [(1 - \omega) \mathbf{D} - \omega \mathbf{U}] \mathbf{x}_n + \omega \mathbf{b} \Leftrightarrow$$

$$\mathbf{D} \mathbf{x}_{n+1} = (1 - \omega) \mathbf{D} \mathbf{x}_n - \omega \mathbf{U} \mathbf{x}_n + \omega \mathbf{b} - \omega \mathbf{L} \mathbf{x}_{n+1} \Leftrightarrow$$

$$\mathbf{x}_{n+1} = (1 - \omega) \mathbf{x}_n + \omega \mathbf{D}^{-1} \left(-\mathbf{U} \mathbf{x}_n + \mathbf{b} - \mathbf{L} \mathbf{x}_{n+1} \right) \Leftrightarrow$$

$$\mathbf{x}_{n+1} = (1 - \omega) \mathbf{x}_n + \omega \mathbf{x}_{n+1}^{\text{Sei}} \quad (\text{SOR})$$

- Finalmente, temos que decidir **quando parar** as iterações destes métodos! Existem vários **critérios de parada** diferentes.
- Todos se baseiam em um determinado valor de **tolerância**, **TOL**, que podemos escolher tão pequeno quanto quisermos. Quanto **menor** for **TOL**, mais iterações serão necessárias.
- Alguns exemplos de critérios frequentemente usados são:
 - máxima diferença absoluta das componentes do vetor \mathbf{x} em duas iterações consecutivas:

$$\delta_{n+1} = \max_{1 \leq i \leq M} |x_{i,n+1} - x_{i,n}| < \text{TOL}$$

- erro relativo associado a δ_{n+1} :

$$\varepsilon_{n+1} = \frac{\delta_{n+1}}{\max_{1 \leq i \leq M} |x_{i,n+1}|} < \text{TOL}$$

- norma euclidiana dos **resíduos** \mathbf{R}_{n+1} :

$$\mathbf{R}_{n+1} = \mathbf{b} - \mathbf{A} \cdot \mathbf{x}_{n+1}, \quad |\mathbf{R}_{n+1}| = \sqrt{\sum_{i=1}^M R_{i,n+1}^2} < \text{TOL}$$