



UNIVERSIDADE ESTADUAL DE SANTA CRUZ - UESC
DEPARTAMENTO DE ENGENHARIAS E COMPUTAÇÃO - DEC
ENGENHARIA QUÍMICA

Disciplina: CET 1012 Modelagem, Simulação e Otimização de Processos Químicos

Professor: Prof.Dr.E.R.Edwards

Projeto

**Projeto de Modelagem e Resolução Algébrica de Problemas Industriais Utilizando
Métodos Matriciais**

Introdução.

Ao longo da disciplina **CET1012** os discentes foram apresentados a conceitos e ferramentas fundamentais de **métodos matriciais** e **computação científica** aplicados a problemas de Engenharia Química. Entre os tópicos trabalhados destacam-se: operações matriciais (soma, subtração, multiplicação), matriz transposta, inversa, determinante, resolução de sistemas lineares para matrizes 3×3 , 4×4 , 5×5 até matrizes $m \times n$, bem como tratamentos especiais (matrizes tridiagonais, triangular superior/inferior). Esses conhecimentos foram aplicados em exemplos industriais e algoritmos numéricos.

O objetivo deste conjunto final de projetos é consolidar o aprendizado por meio da resolução de problemas de engenharia que obrigam o uso intensivo de métodos matriciais. Cada projeto

exige implementação computacional (Google Colab recomendado), aplicação de pelo menos uma matriz 8×8 (ou maior) e a comparação entre implementações algébricas manuais (ex.: eliminação de Gauss, algoritmo de Thomas, decomposição LU, Gauss–Seidel) e funções prontas (ex.: `numpy.linalg.solve`, `scipy.linalg`). O foco é forçar o discente a estudar, justificar escolhas numéricas, controlar erros e interpretar resultados fisicamente.

Objetivos gerais.

- Aplicar técnicas de álgebra linear e algoritmos numéricos na solução de problemas de Engenharia Química.
- Implementar e comparar métodos manuais e bibliotecas para resolução de sistemas lineares.
- Desenvolver habilidade em documentação científica e apresentação técnica (documento + vídeo).
- Validar e justificar os resultados obtidos, avaliando erro numérico e consistência física.

Requisitos comuns a todos os 5 projetos.

1. A resolução deve ser implementada preferencialmente em **Google Colab** (recomendado) ou Jupyter Notebook/VSCode. Entregar o link/arquivo Colab (.ipynb) exportado.
2. Cada projeto **deve usar explicitamente pelo menos uma matriz 8×8** (ou maior) na formação do sistema linear a ser resolvido.
3. O notebook deve conter um **cabeçalho** com: nome da instituição, disciplina (CET1012), título do projeto, nome do discente, data e breve instrução de execução do notebook.
4. O notebook deve apresentar o **enunciado**, a **modelagem matemática**, a **montagem matricial**, a **implementação passo-a-passo** (códigos com comentários), a **solução numérica** (pelo menos dois métodos: um manual/implementado pelo discente e outro

usando biblioteca), verificação de consistência (resíduo, comparação de soluções), e **interpretação física dos resultados**.

5. Implementar **pelo menos dois algoritmos** entre: eliminação de Gauss (com pivotamento parcial), decomposição LU, método de Thomas (para matrizes tridiagonais), Gauss–Seidel (iterativo), Jacobi, cálculo de inversa por eliminação e cálculo de determinante. Usar biblioteca para comparar (numpy/scipy).
6. Mostrar **análise de erro** (norma do resíduo, comparação relativa entre métodos) e, quando aplicável, estudar sensibilidade por variação de parâmetros (pequena perturbação nos coeficientes).
7. Gerar visualizações (gráficos) quando fizer sentido (por exemplo: perfil de temperatura, concentração ao longo de posições, convergência de método iterativo).
8. Entregar também um PDF com o relatório final (capa com os itens solicitados, enunciado e a solução formatada).
9. Gravar um **vídeo** (aprox. 15–25 minutos) demonstrando a resolução: o aluno deve ir executando o notebook, explicando passo-a-passo, justificando decisões e avaliando resultados. O vídeo deve ser atemporal (sem saudações temporais), conter capa inicial com nome/curso/disciplina/título, e ao final aguardar 5 segundos antes de encerrar. Conferir áudio antes do envio; vídeos sem áudio não serão aceitos.
10. Entregar na Pasta de Provas / subpasta **Prova P4** até a data **02/12/2025** (incluir arquivo .ipynb, PDF e vídeo).

Critérios de avaliação (aplicáveis a cada projeto).

- Correção matemática e montagem da matriz: 30%
- Código e implementação (clareza + comentários): 20%
- Uso de métodos manuais + comparação com biblioteca: 15%

- Análise de erro e verificação (resíduo, sensibilidade): 15%
- Relatório escrito (PDF) claro e bem formatado: 10%
- Vídeo demonstrativo (didática, clareza, cumprimento do roteiro): 10%

Projetos (enunciados)

Instruções: escolha um dos projetos abaixo (ou seja atribuído pelo professor). Cada projeto deve originar um notebook Colab com a solução completa e o vídeo explicativo. Cada enunciado explicita qual a estrutura matricial exigida e quais métodos mínimos implementar.

Projeto 1 — Distribuição de Temperatura em uma Barra 1D (Difusão estacionária) — Matriz tridiagonal 8×8

Enunciado: Considere uma barra unidimensional homogênea e estacionária com comprimento L . Discretize a equação do equilíbrio térmico (equação de difusão unidimensional no regime estacionário):

$$\frac{d^2T}{dx^2} = -\frac{q(x)}{k}$$

usando o método das diferenças finitas com **8 nós internos** (ou seja, gerar um sistema linear para as temperaturas nos 8 nós internos). Use condições de contorno de Dirichlet em ambas as extremidades $T(0) = T_0$ e $T(L) = T_L$. Considere um termo fonte $q(x)$ conhecido (por exemplo constante ou função simples).

Requisitos específicos:

- Montar explicitamente a **matriz tridiagonal** 8×8 e o vetor de termos independentes.
- Implementar o **algoritmo de Thomas** (resolução tridiagonal) manualmente e comparar com `numpy.linalg.solve`.
- Calcular resíduo e norma do erro entre soluções.
- Plotar o perfil de temperatura ao longo da barra e comentar a solução.
- Estudar sensibilidade: variar a condutividade k em $\pm 10\%$ e comentar efeitos.

Projeto 2 — Rede de 8 CSTRs em Série com Recycle (Balanços de Massa) — Matriz densa 8×8

Enunciado: Modele uma linha de 8 reatores contínuos perfeitamente misturados (CSTRs) em série, com alimentação no primeiro reator e uma taxa de recirculação que retorna parte do

efluente do último reator ao primeiro. Cada reator tem mesma vazão volumétrica F e volume V . Suponha uma reação de primeira ordem $r = kC$ em cada reator. Determine as concentrações em regime estacionário C_1, \dots, C_8 .

Requisitos específicos:

- Derivar os balanços de massa em regime estacionário e montar a **matriz** 8×8 do sistema linear.
- Implementar eliminação de Gauss com pivotamento parcial (manualmente) e comparar com `numpy.linalg.solve`.
- Calcular a inversa da matriz por eliminação (ou decomposição LU) e verificar que $A^{-1}b$ gera a mesma solução.
- Estudar o efeito da variação do coeficiente cinético k e do coeficiente de recirculação sobre as concentrações.
- Interpretar fisicamente os resultados (quais reatores apresentam maior conversão e por quê).

Projeto 3 — Malha de Trocadores de Calor Discretizada (Acoplamento Condução/Conveção)
— Matriz banda-banda ou blocada $\geq 8 \times 8$

Enunciado: Considere um trocador simplificado ou um arranjo de tubos discretizado em 8 volumes (controles) acoplados por condução e termos convectivos com vazões conhecidas que impõem condições de troca entre nós. Modele as equações de balanço de energia em regime estacionário para os 8 volumes, resultando num sistema linear acoplado.

Requisitos específicos:

- Montar a matriz do sistema (ela pode ser banda ou blocada) com dimensão mínima 8×8 .
- Implementar decomposição LU manualmente (ou via algoritmo já explicado em aula) e compare com `scipy.linalg.lu` e `numpy.linalg.solve`.

- Se o sistema for rígido à convergência, implementar Gauss–Seidel e analisar taxa de convergência (plot da norma do resíduo por iteração).
- Verificar conservação energética total (quando aplicável) e interpretar fisicamente as temperaturas obtidas.

Projeto 4 — Rede de Tubulações: Equilíbrio de Pressões/Nós (Análise de Fluxo Linearizado) — Matriz 8×8 ou maior

Enunciado: Considere uma rede de tubulações com 8 nós e ramificações simples (pontos de junção), onde as quedas de pressão são linearizadas em termos de condutância hidráulica (resistência linearizada). Monte o sistema linear para determinar pressões nodais P_i a partir do balanço de fluxos em cada nó (lembre-se da conservação de massa: somatório de fluxos = 0 para nós internos).

Requisitos específicos:

- Construir a **matriz de condutância** (incidência/coeficientes) com dimensão pelo menos 8×8 .
- Resolver o sistema usando eliminação de Gauss (implementada) e comparar com `numpy.linalg.solve`.
- Calcular fluxos nas tubulações a partir das diferenças de pressão e interpretar sinal (direção do fluxo).
- Realizar análise de sensibilidade: aumentar a resistência (reduzir condutância) de uma das linhas e descrever o efeito global.
- Discutir limitações do modelo linearizado e possíveis extensões não-lineares.

Projeto 5 — Coluna de Destilação Simplificada com 8 Pratos (Balances Lineares de Composição) — Matriz 8×8

Enunciado: Modele uma coluna de destilação com 8 pratos ideais (ou estágios), com alimentação em um prato intermediário. Utilizando um modelo linearizado (por exemplo, abordagem de equações de balanço de massa em regime estacionário com coeficientes de transferência

linearizados ou suposições de equilíbrio linear), monte o sistema linear que determina as composições líquidas x_1, \dots, x_8 ou uma variável análoga (pode-se linearizar as relações de equilíbrio para obter equações lineares aproximadas).

Requisitos específicos:

- Montar a **matriz** 8×8 correspondente aos balanços; explique claramente as hipóteses de linearização adotadas.
- Implementar método iterativo (ex.: Gauss–Seidel) e comparar com solução direta via `numpy.linalg.solve`.
- Estudar convergência e, se necessário, usar relaxação (SOR) e analisar melhoria.
- Verificar e discutir coerência física (ex.: monotonicidade das composições ao longo da coluna) e sensibilidade à posição da alimentação.

Orientações de implementação e boas práticas (detalhadas).

- **Código bem documentado:** cada célula do notebook deve conter comentário explicativo; funções devem ter docstrings; variáveis com nomes significativos.
- **Reprodutibilidade:** definir semeamento quando necessário e fornecer instruções de execução no topo do notebook.
- **Comparaçāo entre métodos:** por exemplo, mostre a solução por eliminação de Gauss, LU e `numpy.linalg.solve`, calcule o resíduo $r = b - Ax$ e as normas $\|r\|_2$, $\|r\|_\infty$.
- **Pivotamento:** estudar caso o sistema exija pivotamento parcial; justificar numericamente.
- **Performance:** discuta custo computacional (tempo e complexidade) dos métodos empregados quando pertinente; medir tempo no notebook (ex.: `time.time()`).
- **Validação:** sempre que possível, comparar com um caso analítico simples (ex.: para o Projeto 1 com $q(x) = 0$, solução linear conhecida).

- **Apresentação final:** o PDF deve conter: capa, enunciado, modelagem, equações, montagem da matriz, resultados e conclusões.

Orientações para gravação do vídeo (15–25 minutos).

- Inicie com capa: nome, curso, disciplina, título do projeto.
- Estruture a fala: breve introdução, apresentação do enunciado, demonstração ao vivo do notebook (executando células selecionadas), interpretação dos resultados, análise de erro e conclusões.
- Seja didático: explique cada passo matemático e computacional; não apenas execute código sem comentar.
- Verifique áudio e vídeo antes do envio; não deixe silêncio no início/fim do vídeo (aguarde 5 segundos antes de finalizar).
- O vídeo deve ser **atemporal** (não mencionar data/hora/saudações temporais) e não deve conter material protegido sem citação.

Observações finais.

- Estes projetos foram formulados para que o discente aplique e interiorize técnicas algébricas e de programação científica. O professor pode adaptar parâmetros (valores numéricos, condições de contorno e propriedades físicas) conforme necessidade da turma.
- Caso deseje, cada aluno pode propor uma pequena variação no enunciado (por exemplo outra função fonte em Projeto 1, outro arranjo de recycle no Projeto 2), desde que o professor aprove previamente.
- Os notebooks bem elaborados poderão ser utilizados como material de referência para turmas futuras.

Fim do documento — CET1012, Projeto Final (5 trabalhos).