

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO

**Modelagem, simulação e otimização da síntese de
álcoois superiores a partir do etanol**

Rogério Pazetto Saldanha da Gama

PROJETO FINAL DE GRADUAÇÃO

CENTRO TÉCNICO CIENTÍFICO - CTC

DEPARTAMENTO DE INFORMÁTICA

Curso de Graduação em Engenharia da Computação

Rio de Janeiro, Junho de 2022



Rogério Pazetto Saldanha da Gama

**Modelagem, simulação e otimização da síntese de
álcoois superiores a partir do etanol**

Projeto final apresentado ao Curso de
Engenharia da Computação como requisito
parcial para a obtenção do título de
Engenheiro de Computação

Orientador: Professor Marcelo Gattass
Co-orientadora: Professora Amanda
Lemette Teixeira Brandão

Rio de Janeiro, Junho de 2022

Resumo

Pazetto. Rogério. Gattass. Marcelo. Modelagem, simulação e otimização da síntese de álcoois superiores a partir do etanol. Rio de Janeiro. 2022. Número de páginas: 39. Relatório Final de Projeto de Graduação II - Departamento de Informática. Pontifícia Universidade Católica do Rio de Janeiro.

Os álcoois têm diversas aplicações no nosso cotidiano e na indústria química. Exemplos que podemos citar são o uso do etanol como antisséptico - que ganhou muita importância durante a pandemia da COVID-19 - uso do etanol como combustível, uso do butanol como combustível substituindo o diesel, com menos emissão de poluentes, entre outros. O objetivo deste trabalho é simular com a linguagem Python o processo de síntese de n-butanol, n-hexanol e n-octanol a partir do etanol em um reator do tipo PFR (plug flow reactor) com reciclo e otimizar o processo de modo a aumentar o rendimento dos produtos de interesse (n-butanol, n-hexanol e n-octanol). Adicionalmente, foi desenvolvida uma interface gráfica de modo a possibilitar o uso do modelo validado por usuários que não sabem programar.

Palavras-chave

Simulação, PFR, álcool, Python, Otimização

Abstract

Pazetto. Rogério. Gattass. Marcelo. Modeling, Simulation and Optimization of the Synthesis of Higher alcohols from an Ethanol Source. Rio de Janeiro. 2022. 39p. Project Report - Department of Informatics, Pontifical Catholic University of Rio de Janeiro.

Alcohols play important roles on our daily lives and in the chemical industry. We can mention examples such as the use of ethanol as antiseptic - which became more important during COVID-19 pandemic - use of ethanol as fuel, use of butanol as fuel replacing diesel with less harmful emissions, and so on. The objective of this work is simulate the synthesis of n-butanol, n-hexanol and n-octanol from an ethanol source in a PFR (plug flow reactor) with recycle and optimize the process in order to increase the yield of the interest products (n-butanol, n-hexanol and n-octanol). Additionally, it was developed a guided user interface to allow users who are not familiar with programming to use the model.

Keywords

Simulation, PFR, alcohol, Python, Optimization

Sumário

• 1 Introdução	1
• 2 Situação Atual	3
• 3 Objetivos	3
• 4 Atividades Realizadas	4
○ Solver de EDOs	4
○ Reciclo	8
○ Otimização do Processo	9
○ Interface Gráfica	9
• 5 Projeto e Especificação do Sistema	12
○ Solver de EDOs	12
○ Reciclo	13
○ Otimização do Processo	13
○ Interface Gráfica	14
• 6 Implementação e Avaliação	15
• 7 Considerações Finais	19
• 8 Referências	19
• Anexo 1. Solver de EDOs	22
• Anexo 2. Reciclo	28
• Anexo 3. Otimização do Processo	30
• Anexo 4. Interface Gráfica	32

Lista de Figuras

Figura	Página
Figura 1. Diagrama de blocos representando o processo envolvendo o reciclo.	9
Figura 2. Interface gráfica para usuários que não têm familiaridade com programação.	10
Figura 3. Conversão de etanol e rendimento dos produtos de interesse (n-butanol, n-hexanol e n-octanol). As curvas, marcadas com (n), são os resultados do solver e os pontos, marcados com (r), são os resultados experimentais [1].	15
Figura 4. Conversão de etanol com reciclo e sem reciclo.	16
Figura 5. Rendimento do n-butanol com reciclo e sem reciclo.	16
Figura 6. Rendimentos do n-hexanol e do n-octanol com reciclo e sem reciclo.	17

Lista de Tabelas

Tabela	Página
Tabela 1. Fator pré-exponencial (k_0) e energia de ativação (Ea) de cada reação j . Fonte: Tsuchida et al. (2006).	6
Tabela 2. Taxa de reação para cada composto, onde $\frac{dC_i}{dt}$ é a taxa de reação do composto i e k_j é a constante da reação j . Fonte: Tsuchida et al. (2006).	7
Tabela 3. Limites para a otimização.	18

1 Introdução

Os álcoois têm diversas aplicações no nosso cotidiano e na indústria química. Exemplos que podemos citar são o uso do etanol como antisséptico - que ganhou muita importância durante a pandemia da COVID-19 - uso do etanol como combustível, uso do hexanol e butanol na produção de solventes, entre outros. Alguns autores propuseram o uso do n-butanol como combustível, como Scheid et al. [4], que ressaltou em seu trabalho o alto valor energético do n-butanol (29,2 MJ/L) em comparação com o etanol (19,2 MJ/L).

As rotas de produção convencionais do n-butanol, n-hexanol e n-octanol - e de vários outros álcoois - envolvem fontes não renováveis, o que gera impactos ambientais, porém alguns autores sugeriram rotas alternativas para a produção, como por exemplo a fermentação com microorganismos e a síntese a partir de uma fonte de etanol. O etanol já é produzido de forma sustentável por meio da fermentação de açúcar com microorganismos [11].

Jin et al. [5] propôs o uso do n-butanol como combustível, estudou sua produção por uma rota biológica e comparou seu desempenho com o desempenho de outros combustíveis em motores. Han et al. (2017) estudou o uso do n-butanol em motores a compressão e comparou as emissões com as emissões do diesel. Em ambos os trabalhos a conclusão foi que o n-butanol pode ser utilizado nos motores, resultando em menos emissões e desempenho comparável ao desempenho obtido com combustíveis convencionais.

Ndaba et al. [9] comparou as rotas biológica e química para produzir n-butanol,

concluindo que a rota química é mais eficiente. O estudo também sugeriu o uso do n-butanol como combustível. Carvalho et al. [8] estudou a síntese de n-butanol a partir de etanol com catalisadores à base de magnésio e alumínio. Rittonen et al. [7] estudou a síntese de n-butanol a partir de uma fonte de etanol em um reator de leito fixo com um modelo de 5 reações. Wang et al. [10] produziu n-butanol a partir de etanol em um reator batelada com 25 % de conversão de etanol e 11 % de rendimento do n-butanol.

Os processos químicos podem ocorrer em diversos reatores, sendo eles o CSTR (Continuous Stirred-Tank Reactor), que é um tanque com agitador dentro do qual ocorrem as reações, PFR (Plug Flow Reactor), que é o reator tubular e PBR (Packed Bed Reactor), que é o reator tubular com catalisadores para reações heterogêneas. O sistema de reações utilizado neste trabalho foi o proposto por Tsuchida et al. [1], que é a produção de n-butanol, n-hexanol e n-octanol a partir de etanol em um reator PFR. O modelo proposto envolve 13 reações.

Tsuchida et al. [1] executou o processo em laboratório e simulou computacionalmente o problema comparando os resultados encontrados. Porém a otimização do processo não foi o foco deste trabalho (apenas alguns parâmetros foram alterados em uma faixa bastante limitada).

A otimização é extremamente importante tendo em vista os elevados valores de mercado destes compostos. Um processo não otimizado pode resultar em desperdício de reagente, produção de uma quantidade maior e desnecessária de subprodutos nos quais não temos interesse e uma produção menor de produtos de interesse. Por estarmos interessados nos álcoois n-butanol, n-hexanol e n-octanol, os parâmetros devem ser ajustados visando aumentar suas concentrações na saída do processo.

2 Situação Atual

A rota estudada neste trabalho é a síntese diretamente a partir do etanol. Para isso, é necessário entender a cinética das reações envolvidas, que é descrita por um sistema de equações diferenciais relacionando as taxas de reação com as concentrações.

Para simular a rota química proposta por Tsuchida et al. [1], primeiro foi desenvolvido um modelo de balanço de massa, e este foi implementado em linguagem de programação Python dentro de uma classe. Outro objetivo do trabalho é a otimização do processo, focando na maximização da produção de n-butanol, n-hexanol e n-octanol. Para alcançar esse objetivo, foi desenvolvido um algoritmo para simular o reciclo de etanol não consumido na saída do reator e um algoritmo para encontrar as condições de entrada do processo que resultam no maior lucro possível, aumentando a produção de n-butanol, n-hexanol e n-octanol como consequência.

Por fim foi desenvolvida uma interface gráfica com a biblioteca Tkinter para que o modelo possa ser utilizado por usuários sem familiaridade com programação.

3 Objetivos

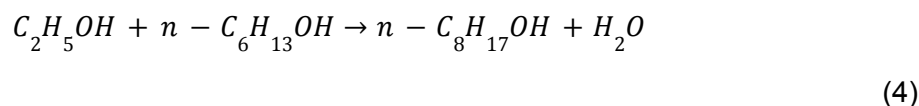
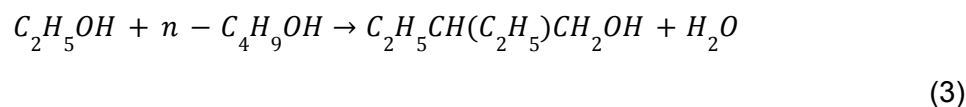
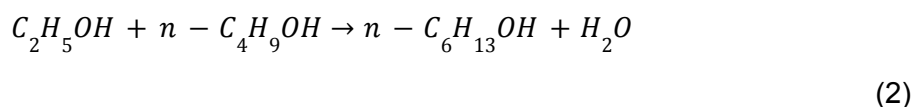
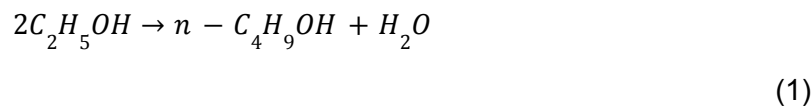
O objetivo deste trabalho é simular com a linguagem Python o processo de síntese de n-butanol, n-hexanol e n-octanol a partir do etanol em um reator do tipo PFR (plug flow reactor) com reciclo, otimizar o processo de modo a aumentar o rendimento destes produtos e desenvolver uma interface gráfica de modo a possibilitar o uso do modelo por usuários que não têm familiaridade com programação.

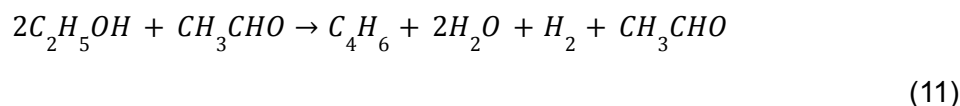
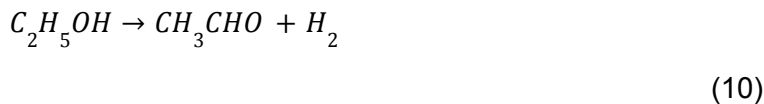
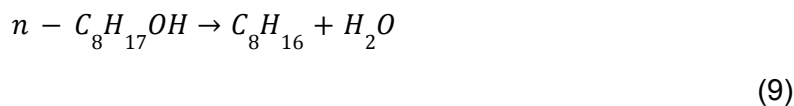
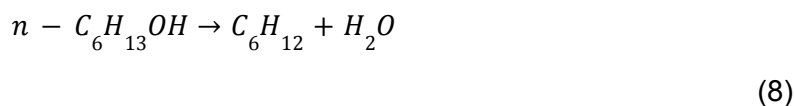
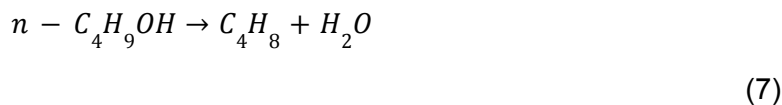
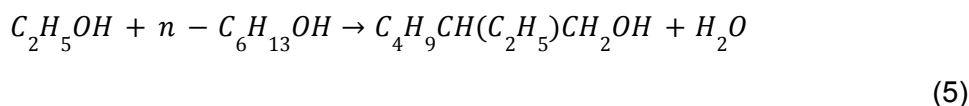
4 Atividades Realizadas

O código envolve um solver para o sistema de EDOs, um algoritmo para simular o reciclo de etanol, um otimizador para o processo e uma interface gráfica para que os usuários que não têm familiaridade com programação possam também simular o processo.

- **Solver de EDOs**

O modelo utilizado tem 13 reações:





Cada reação tem uma taxa, que é função da constante de reação (k) e das concentrações dos compostos envolvidos. O k é calculado a partir da Equação de Arrhenius:

$$k_j = k_{0,j} e^{\left(\frac{-Ea_j}{RT}\right)}$$

(14)

Onde:

k_j é a constante da reação j ;

$k_{0,j}$ é o fator pré exponencial da reação j ;

Ea_j é a energia de ativação da reação j ;

R é a constante universal dos gases;

T é a temperatura em K;

Para cada reação j , os parâmetros $k_{0,j}$ e Ea_j estão dados na Tabela 1:

Tabela 1. Fator pré-exponencial (k_0) e energia de ativação (Ea) de cada reação j . Fonte:

Tsuchida et al. [1].

Reação	Fator pré exponencial	Energia de ativação (J/mol)
1	$2.1 \cdot 10^3$	$61 \cdot 10^3$
2	$1.7 \cdot 10^4$	$70 \cdot 10^3$
3	$2.0 \cdot 10^2$	$50 \cdot 10^3$
4	$1.2 \cdot 10^6$	$92 \cdot 10^3$
5	4.2	$21 \cdot 10^3$
6	$3.4 \cdot 10^7$	$119 \cdot 10^3$

7	$1.0 \cdot 10^6$	$91 \cdot 10^3$
8	$5.5 \cdot 10^2$	$40 \cdot 10^3$
9	$2.7 \cdot 10^2$	$30 \cdot 10^3$
10	$3.8 \cdot 10^6$	$102 \cdot 10^3$
11	$1.6 \cdot 10^3$	$49 \cdot 10^3$
12	$1.1 \cdot 10^6$	$105 \cdot 10^3$
13	$2.1 \cdot 10^5$	$91 \cdot 10^3$

Com isso, podemos encontrar a taxa de reação para cada composto:

Tabela 2. Taxa de reação para cada composto, onde $\frac{dC_i}{dt}$ é a taxa de reação do composto i e k_j é a constante da reação j . Fonte: Tsuchida et al. [1].

Composto	Identificação	Taxa de reação
C_2H_5OH	0	$\frac{dC_0}{dt} = -2k_1C_0^2 - (k_2 + k_3)C_0C_1 - (k_4 + k_5)C_0C_2 - (k_6 + k_{11} + k_{13})C_0 - 2k_{10}C_0C_{10}$
$n - C_4H_9OH$	1	$\frac{dC_1}{dt} = k_1C_0^2 - (k_2 + k_3)C_0C_1 - k_7C_1$
$n - C_6H_{13}OH$	2	$\frac{dC_2}{dt} = k_2C_0C_1 - (k_4 + k_5)C_0C_2 - k_8C_2$
$C_2H_5CH(C_2H_5)CH_2OH$	3	$\frac{dC_3}{dt} = k_3C_0C_1$

$n - C_8H_{17}OH$	4	$\frac{dC_4}{dt} = k_4 C_0 C_2 - k_9 C_4$
$C_4H_9CH(C_2H_5)CH_2OH$	5	$\frac{dC_5}{dt} = k_3 C_0 C_2$
C_2H_4	6	$\frac{dC_6}{dt} = k_6 C_0$
C_4H_8	7	$\frac{dC_7}{dt} = k_7 C_1$
C_6H_{12}	8	$\frac{dC_8}{dt} = k_8 C_2$
C_8H_{16}	9	$\frac{dC_9}{dt} = k_9 C_4$
CH_3CHO	10	$\frac{dC_{10}}{dt} = k_{10} C_0$
C_4H_6	11	$\frac{dC_{11}}{dt} = k_{11} C_0 C_{10}$
<i>aromatics</i>	12	$\frac{dC_{12}}{dt} = k_{12} C_0$
<i>others</i>	13	$\frac{dC_{13}}{dt} = k_{13} C_0$

O sistema de equações da Tabela 2 nos permite encontrar o perfil de concentrações de cada composto.

Apesar de o sistema de equações descrever o comportamento do processo, não podemos assumir que o resultado é fisicamente possível uma vez que não existe qualquer restrição que evite geração ou consumo de matéria no interior do reator. Para evitar que

isso aconteça, uma das concentrações foi calculada pelo balanço molar envolvendo a entrada e a quantidade dos demais compostos.

- **Reciclo**

Na saída do reator, foi implementado um separador ideal, que tem como função separar totalmente o etanol dos produtos. A saída de etanol é então direcionada à entrada do PFR. Para evitar um excesso de etanol dentro do reator, que resultaria em um menor tempo de residência e, conseqüentemente, uma conversão menor, foi implementado também um sistema de controle para manter a vazão de etanol na entrada do reator constante.

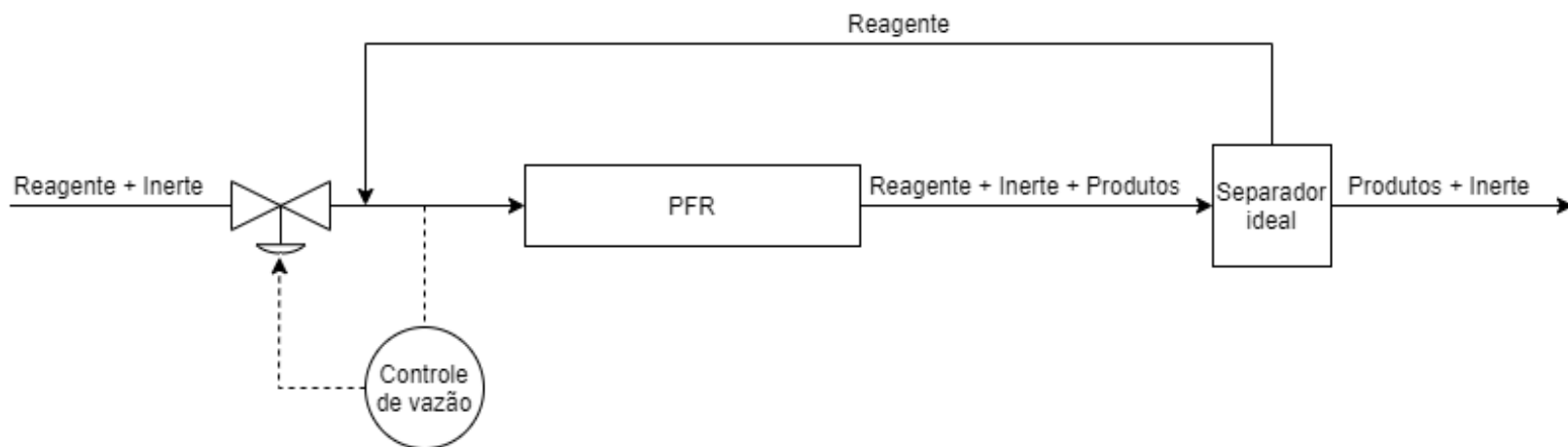


Figura 1. Diagrama de blocos representando o processo envolvendo o reciclo.

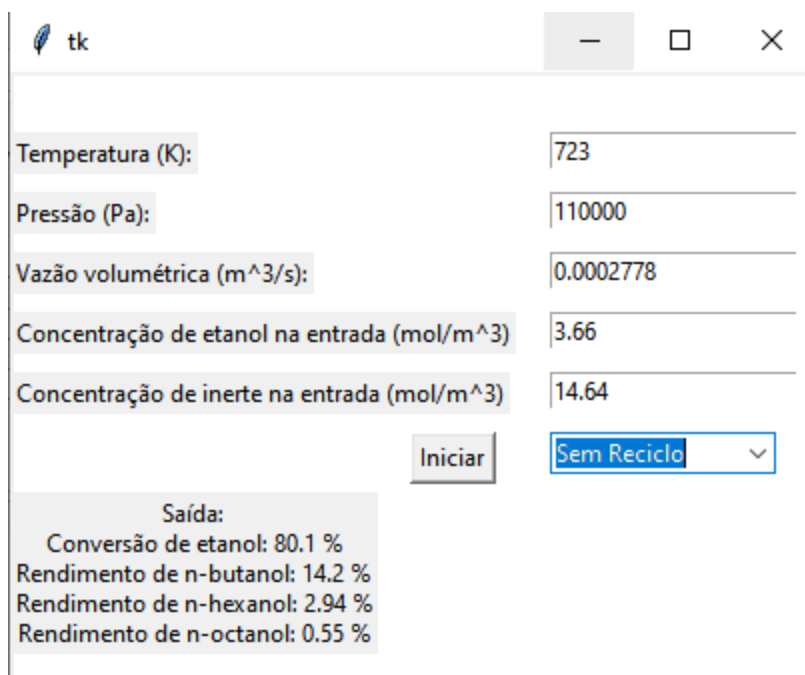
O sistema descrito foi implementado como uma função dentro da qual o solver de EDOs foi chamado.

- **Otimização do Processo**

O processo foi então otimizado, de forma a encontrar os parâmetros de entrada que resultam na saída mais lucrativa possível. Inicialmente foi definida uma função objetivo que, para um conjunto de parâmetros de entrada, retorna o lucro obtido.

- **Interface Gráfica**

Por fim foi implementada uma interface gráfica com a biblioteca Tkinter para facilitar o uso do programa por pessoas que não têm familiaridade com programação.



tk

Temperatura (K): 723

Pressão (Pa): 110000

Vazão volumétrica (m³/s): 0.0002778

Concentração de etanol na entrada (mol/m³): 3.66

Concentração de inerte na entrada (mol/m³): 14.64

Iniciar Sem Reciclo

Saída:

Conversão de etanol: 80.1 %

Rendimento de n-butanol: 14.2 %

Rendimento de n-hexanol: 2.94 %

Rendimento de n-octanol: 0.55 %

Figura 2. Interface gráfica para usuários que não têm familiaridade com programação.

- Cronogramas

Cronograma real:

Mês/Ano	Atividade
---------	-----------

Agosto/2021	Revisão bibliográfica
Setembro/2021	Revisão bibliográfica
Outubro/2021	Implementação do solver de EDO
Novembro/2021	Implementação do solver de EDO Entrega da proposta
Dezembro/2021	Implementação do reciclo Entrega do relatório
Janeiro/2022	Implementação do reciclo
Fevereiro/2022	Implementação do otimizador
Março/2022	Implementação do otimizador
Abril/2022	Implementação da interface gráfica
Maio/2022	Implementação da interface gráfica
Junho/2022	Implementação da interface gráfica Entrega do relatório final

Cronograma proposto:

Mês/Ano	Atividade
Agosto/2021	Revisão bibliográfica
Setembro/2021	Revisão bibliográfica
Outubro/2021	Implementação do solver de EDO
Novembro/2021	Implementação do solver de EDO

	Entrega da proposta
Dezembro/2021	Implementação do reciclo
Janeiro/2022	Implementação do reciclo Entrega do relatório
Fevereiro/2022	Implementação do otimizador
Março/2022	Implementação do otimizador
Abril/2022	Implementação da interface gráfica
Maior/2022	Implementação da interface gráfica
Junho/2022	Implementação da interface gráfica

5 Projeto e Especificação do Sistema

O reator foi simulado em uma classe na qual são declaradas as variáveis referentes ao sistema (diâmetro do reator, parâmetros da Tabela 1, etc.) e são recebidas as variáveis referentes às condições do processo (temperatura, pressão, vazão, etc.) no método construtor. A classe contém também o método “simula”, que, com as variáveis armazenadas, simula o processo.

- **Solver de EDOs**

O método utilizado para resolver o sistema foi o método de Euler explícito [12] seguindo a regra de recorrência

$$C_{i,k} = C_{i,k-1} + \frac{dC_{i,k-1}}{dt} \frac{1}{F} \Delta z \quad (15)$$

Onde:

$C_{i,k}$ é a concentração do composto i no instante de tempo k ;

$C_{i,k-1}$ é a concentração do composto i no instante de tempo $k - 1$;

$\frac{dC_{i,k-1}}{dt}$ é a taxa de reação do composto i no instante de tempo $k - 1$ (dada pela equação do composto i na Tabela 2);

F é a velocidade de escoamento;

Δz é o passo na posição.

Tendo em vista que o método explícito de Euler discretiza as variáveis, a posição avança Δz de forma discreta em cada passo.

A regra de recorrência (15) foi implementada no método “simula”, que retornou uma matriz com a concentração de cada componente para cada posição no reator.

- **Reciclo**

O reciclo foi implementado como uma função fora da classe do reator que recebe os mesmos parâmetros da classe do reator. Dentro da função, o solver de EDOs é chamado em várias iterações até a saída parar de oscilar. Em cada iteração o solver é chamado com os parâmetros recebidos e a vazão de etanol novo é ajustada em função da saída da iteração anterior de forma a manter a vazão na entrada do reator constante.

A função que simula o reciclo retorna uma matriz no mesmo formato do método “simula”, que resolve o sistema de EDOs.

- **Otimização do Processo**

A função objetivo retorna o lucro obtido ao simular o processo multiplicado por -1 (16). Os parâmetros que minimizam essa função são os mesmos parâmetros que maximizam o lucro.

$$f_{obj}(param) = - \left[F \cdot (C_{1,s}(param) \cdot V_1 + C_{2,s}(param) \cdot V_2 + C_{4,s}(param) \cdot V_4) - F_{in}(C_{0,e}(param) \cdot V_0) \right]$$

(16)

Onde:

$param$ representa o conjunto de parâmetros de entrada para a simulação do processo (temperatura, pressão, etc.);

f_{obj} é a função objetivo, que retorna o lucro do processo multiplicado por -1;

$C_{i,s}(param)$ é a concentração do composto i na saída do processo para o conjunto de parâmetros dado;

$C_{0,e}(param)$ é a concentração de etanol na entrada do processo;

V_i é o valor de mercado do composto i .

A função foi então minimizada utilizando o método Powell da biblioteca scipy [13].

- **Interface Gráfica**

Uma interface gráfica foi desenvolvida com a biblioteca Tkinter para que usuários que não têm familiaridade com programação possam utilizar o modelo. Ao executar o código, a interface é aberta com caixas de texto para que o usuário defina as condições de entrada do processo. Após definir as condições de entrada, o usuário seleciona se o processo tem ou não um reciclo e, por fim, inicia a simulação no botão “Iniciar”. O botão “Iniciar” chama o solver ou a função que resolve o problema com reciclo (dependendo da opção do usuário) e retorna as saídas mais relevantes do processo: conversão de etanol e rendimentos dos produtos de interesse: n-butanol, n-hexanol e n-octanol.

6 Implementação e Avaliação

Inicialmente o modelo foi validado comparando a saída do solver com os

resultados obtidos por Tsuchida et al. [1].

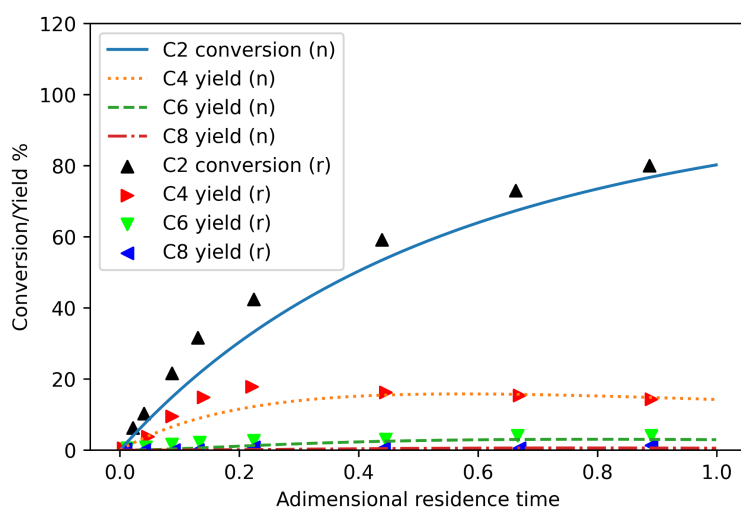


Figura 3. Conversão de etanol e rendimento dos produtos de interesse (n-butanol, n-hexanol e n-octanol). As curvas, marcadas com (n), são os resultados do solver e os pontos, marcados com (r), são os resultados experimentais [1].

Os resultados na Figura 3 - experimentais e simulados - foram obtidos com as mesmas condições de operação do processo. Como se pode observar, os resultados da simulação foram condizentes com os resultados experimentais, validando o modelo. Os resultados obtidos com a simulação foram 80,20 % de conversão de etanol, 14,25 % de rendimento para o n-butanol, 2,94 % de rendimento para o n-hexanol e 0,56 % de rendimento para o n-octanol.

Após validar o modelo, o simulador foi executado considerando o reciclo.

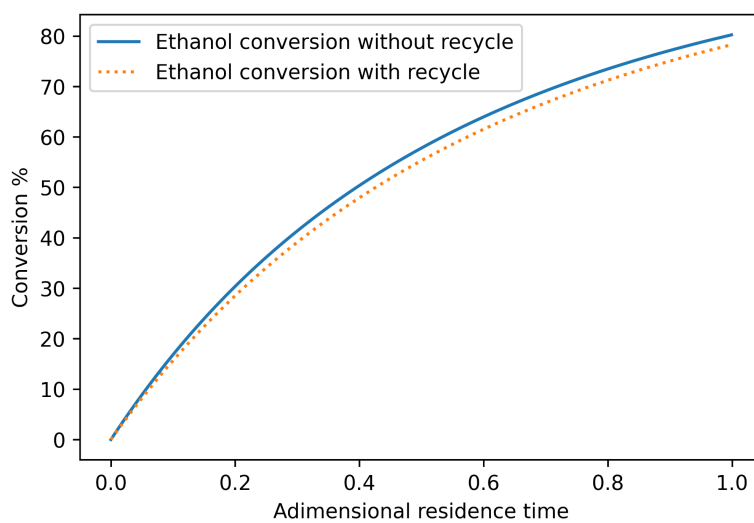


Figura 4. Conversão de etanol com reciclo e sem reciclo.

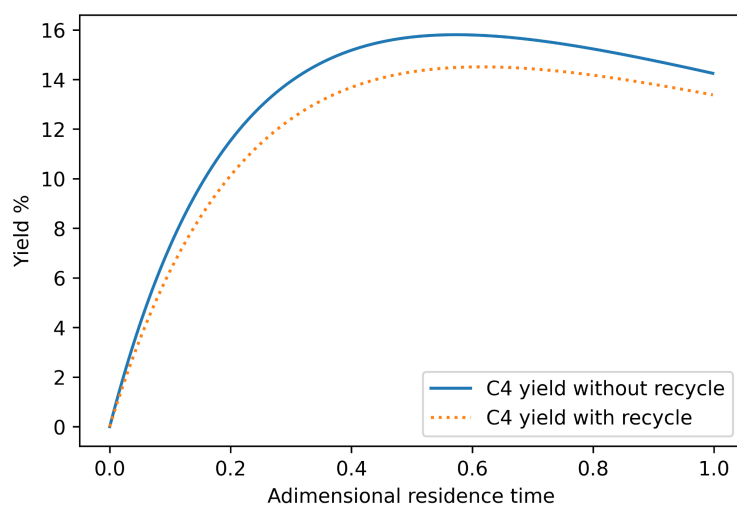


Figura 5. Rendimento do n-butanol com reciclo e sem reciclo.

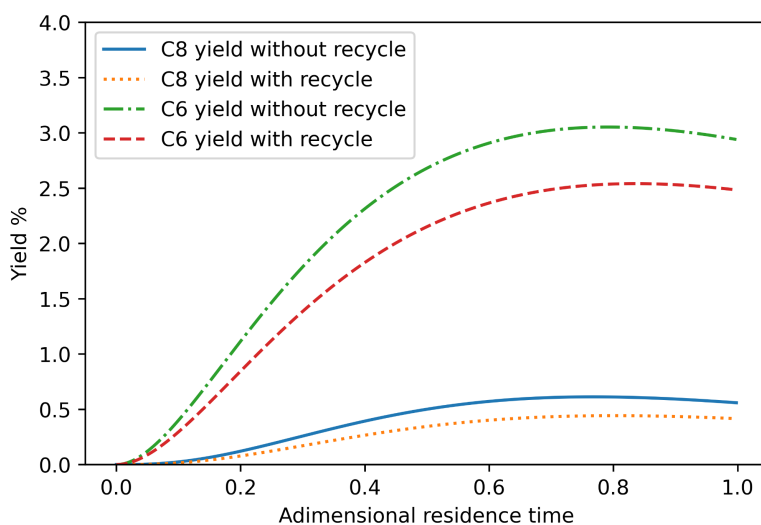


Figura 6. Rendimentos do n-hexanol e do n-octanol com reciclo e sem reciclo.

Os rendimentos foram plotados em figuras diferentes pois a diferença entre o rendimento do n-butanol e dos demais componentes resultaria em uma escala inadequada.

Os resultados com reciclo foram de 78,23 % de conversão de etanol, 13,38 % de rendimento para o n-butanol, 2,48 % de rendimento para o n-hexanol e 0,42 % de rendimento para o n-octanol. Apesar de os resultados sugerirem que o processo ficou menos eficiente, devemos considerar que os gráficos são referentes apenas ao reator, e não ao processo, que tem um reciclo que separa o etanol na saída e direciona para a entrada do reator, resultando em uma conversão global de 100 % de etanol.

Conforme apresentado nos gráficos, os rendimentos dos produtos de interesse não são máximos necessariamente na saída do reator, indicando claramente que o processo pode ser otimizado de forma a aumentar seu lucro e, ainda assim, sem gerar impactos ambientais.

Ao aplicar o método Powell para minimizar a Equação (16), os resultados obtidos foram uma temperatura de 695,4 K, pressão de 1000000 Pa, vazão de 0,001 m³/s e etanol puro na entrada (fração molar igual a 1,0). Utilizando estes parâmetros o lucro diário será de R\$ 122289,42, considerando que o sistema opera em estado estacionário 24 horas por dia.

Para evitar resultados impossíveis, foram definidos limites para o otimizador.

Tabela 3. Limites para a otimização.

Parâmetro	Mínimo	Máximo
Temperatura (K)	300	1000
Pressão (Pa)	100000	1000000
Vazão volumétrica (m ³ /s)	0	0.001
Fração molar de etanol na entrada	0	1

Ao executar a função objetivo com as condições utilizadas por Tsuchida et al. [1], o resultado é R\$ 779.97 de lucro por dia com reciclo e R\$ 3528,52 de prejuízo sem reciclo.

Simulando o processo com reciclo com as condições retornadas pelo otimizador, a conversão de etanol no reator é de 97,01 % e os rendimentos para n-butanol, n-hexanol e n-octanol são, respectivamente, 19,35 %, 11,44 % e 15,25 %.

7 Considerações Finais

Os resultados ressaltam a importância de simular um processo químico antes de iniciar a operação. Mesmo que esta simulação não envolva reciclo ou otimização, seus resultados fornecem informações relevantes e essenciais sobre o processo.

Ao simular com otimização, as condições ideais para operar o processo serão conhecidas. No caso de maximizar o lucro, as condições ideais são uma temperatura de 695,4 K, pressão de 1000000 Pa, vazão de 0,001 m³/s e fração molar de etanol igual a 1,0, resultando em um lucro diário de R\$ 122289,42.

A função objetivo pode ser desenvolvida de forma diferente, por exemplo, considerando o lucro de outros compostos, ou buscando maximizar o rendimento de um determinado composto. A minimização da função objetivo também pode ser feita utilizando outros algoritmos, não se limitando apenas ao método Powell.

O processo também pode ser investigado modificando as características do sistema, como ao considerar um separador real ao invés de um separador ideal e ao modificar as dimensões do reator.

Tais implementações seriam estudos válidos e com grande contribuição científica, o que abre possibilidades para continuar a investigação deste processo.

8 Referências

- 1 - TSUCHIDA, T. et al. Direct **Synthesis of n-Butanol from Ethanol over**

- Nonstoichiometric Hydroxyapatite.** Industrial & Engineering Chemistry Research, v. 45, n. 25, p. 8634–8642, dez. 2006.
- 2 - KUMAR, R. et al. **Effect of a sustainable biofuel – *n*-octanol – on the combustion, performance and emissions of a DI diesel engine under naturally aspirated and exhaust gas recirculation (EGR) modes.** Energy Conversion and Management. Journal homepage: www.elsevier.com/locate/enconman
- 3 - XIAOYE, H. et al. **Clean combustion of *n*-butanol as a next generation biofuel for diesel engines.** Applied Energy. Journal homepage: www.elsevier.com/locate/apenergy
- 4 - SCHEID, J. et al. **Mechanism and Kinetic Modeling of Ethanol Conversion to 1-Butanol over Mg and Al Oxide Derived from Hydrotalcites.** Industrial & Engineering Chemistry Research 2019 58 (29), 12981-12995 DOI: 10.1021/acs.iecr.9b01491
- 5 - JIN, C. et al. **Progress in the production and application of *n*-butanol as a biofuel.** Renewable and Sustainable Energy Reviews. Journal homepage: www.elsevier.com/locate/rser
- 7 - RITTONEN, T. et al. **Direct Synthesis of 1-Butanol from Ethanol in a Plug Flow Reactor: Reactor and Reaction Kinetics Modeling**
- 8 - CARVALHO, R. et al. **Mg and Al mixed oxides and the synthesis of *n*-butanol from ethanol**

- 9 - NDABA, B. et al. ***n*-Butanol derived from biochemical and chemical routes: A review**
- 10 - WANG, D. et al. **Synthesis of 1-Butanol from Ethanol over Calcium Ethoxide: Experimental and Density Functional Theory Simulation**
- 11 - THAMMASITTIRONG S., et al. **Improvement of ethanol production by ethanol-tolerant *Saccharomyces cerevisiae* UVNR56**
- 12 - Boyce, W., E., DiPrima, R. C. - **Equações Diferenciais Elementares e Problemas de Valores de Contorno**
- 13 - Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, CJ Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E.A. Quintero, Charles R Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. (2020) SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. Nature Methods, 17(3), 261-272.

Anexo 1. Solver de EDOs

class OCM2:

```
def __init__(self,T,P,vz_STP,C_et0,C_inerte):

    self.T = T # K

    self.P = P # Pa

    self.vz_STP = vz_STP # m3/s

    self.L = 22 # comprimento do reator [m]

    self.Dt = 5/1000 # diametro do reator [m] [Tsuchida]

    self.R = 8.314 # [J/(mol K) ou m3 Pa/(K mol)]

    self.Nreac = 13 # Numero de reacoes

    self.Ncomp = 15 # Numero de componentes

    self.k0 =
    np.array([2.1e3,1.7e4,2.0e2,1.2e6,4.2,3.4e7,1.0e6,5.5e2,2.7e2,3.8e6,1.6e3,1.1e6,2.
    1e5]) #[Tsuchida]

    self.Ea =
    [61e3,70e3,50e3,92e3,21e3,119e3,91e3,40e3,30e3,102e3,49e3,105e3,91e3]
    #[J/mol] [Tsuchida]

    self.m = [0.24,1.0,0.57,1.0,0.95,1.0,0.0,0.97,1.0,1.0]

    self.n = [0.76,0.40,0.85,0.55,0.37,0.96,0.0,0.0,1.0,1.0]

    self.A = math.pi*(self.Dt**2)/4 # transversal area [m2]

    self.C0 = C_et0 # concentracao de etanol em z = 0 [mol/m3]
```

```
self.inerte=C_inerte  
  
self.vz=vz_STP/self.A  
  
self.dz=self.L/1000  
  
self.CI = [self.CO,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,self.inerte]# Initial condition
```

```
def simula(self,metodo):
```

```
    k1 = self.k0[0]*np.exp(-self.Ea[0]/(self.R*self.T)) # r1  
    k2 = self.k0[1]*np.exp(-self.Ea[1]/(self.R*self.T)) # r2  
    k3 = self.k0[2]*np.exp(-self.Ea[2]/(self.R*self.T)) # r3  
    k4 = self.k0[3]*np.exp(-self.Ea[3]/(self.R*self.T)) # r4  
    k5 = self.k0[4]*np.exp(-self.Ea[4]/(self.R*self.T)) # r5  
    k6 = self.k0[5]*np.exp(-self.Ea[5]/(self.R*self.T)) # r6  
    k7 = self.k0[6]*np.exp(-self.Ea[6]/(self.R*self.T)) # r7  
    k8 = self.k0[7]*np.exp(-self.Ea[7]/(self.R*self.T)) # r8  
    k9 = self.k0[8]*np.exp(-self.Ea[8]/(self.R*self.T)) # r9  
    k10 = self.k0[9]*np.exp(-self.Ea[9]/(self.R*self.T)) # r10  
    k11 = self.k0[10]*np.exp(-self.Ea[10]/(self.R*self.T)) # r11  
    k12 = self.k0[11]*np.exp(-self.Ea[11]/(self.R*self.T)) # r12  
    k13 = self.k0[12]*np.exp(-self.Ea[12]/(self.R*self.T)) # r13
```

```
C=self.Cl
vesc=self.vz
res=[]
res.append(self.Cl);
dcdz = np.zeros(self.Ncomp)
z=0;
if metodo==0: #explícito
    while z<=self.L :
        Ctmp=[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
        dcdz[0] =
        (-2*k1*C[0]*C[0]-(k2+k3)*C[0]*C[1]-(k4+k5)*(C[0]*C[2])-(k6+k11+k13)*C[0]-2*k10*C[0]
        ]*C[10])/vesc
        dcdz[1] = (k1*C[0]*C[0]-(k2+k3)*C[0]*C[1]-k7*C[1])/vesc
        dcdz[2] = (k2*C[0]*C[1]-(k4+k5)*C[0]*C[2]-k8*C[2])/vesc
        dcdz[3] = (k3*C[0]*C[1])/vesc
        dcdz[4] = (k4*C[0]*C[2]-k9*C[4])/vesc
        dcdz[5] = (k5*C[0]*C[2])/vesc
        dcdz[6] = (k6*C[0])/vesc
        dcdz[7] = (k7*C[1])/vesc
        dcdz[8] = (k8*C[2])/vesc
```

```
dcdz[9] = (k9*C[4])/vesc
dcdz[10] = (k10*C[0])/vesc
dcdz[11] = (k11*C[0]*C[10])/vesc
dcdz[12] = 0 #***
dcdz[13] = (k13*C[0])/vesc
dcdz[14] = 0
cnt1=0
while cnt1<15:
    Ctmp[cnt1]=C[cnt1]+dcdz[cnt1]*self.dz
    cnt1=cnt1+1

#2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][3]+8*res[cnt1][4]+8*res[cnt1]
[5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]+2*res[cnt1][10]+4*res[c
nt1][11]+6*res[cnt1][12]+8*res[cnt1][13]

C[12]=(self.C0*2-(C[0]*2+C[1]*4+C[2]*6+C[3]*6+C[4]*8+C[5]*8+C[6]*2+C[7]*4+C[8]*6
+C[9]*8+C[10]*2+C[11]*4+C[13]*8))/6 #*****

C=Ctmp
res.append(C)
z=z+self.dz

if metodo==1: #implícito
```

```
while z<=self.L :  
  
    C=res[-1]  
  
    cnt1=0  
  
    while cnt1<100:  
  
        dcdz[0] =  
        (-2*k1*C[0]*C[0]-(k2+k3)*C[0]*C[1]-(k4+k5)*(C[0]*C[2])-(k6+k11+k13)*C[0]-2*k10*C[0]  
        ]*C[10])/vesc  
  
        dcdz[1] = (k1*C[0]*C[0]-(k2+k3)*C[0]*C[1]-k7*C[1])/vesc  
  
        dcdz[2] = (k2*C[0]*C[1]-(k4+k5)*C[0]*C[2]-k8*C[2])/vesc  
  
        dcdz[3] = (k3*C[0]*C[1])/vesc  
  
        dcdz[4] = (k4*C[0]*C[2]-k9*C[4])/vesc  
  
        dcdz[5] = (k5*C[0]*C[2])/vesc  
  
        dcdz[6] = (k6*C[0])/vesc  
  
        dcdz[7] = (k7*C[1])/vesc  
  
        dcdz[8] = (k8*C[2])/vesc  
  
        dcdz[9] = (k9*C[4])/vesc  
  
        dcdz[10] = (k10*C[0])/vesc  
  
        dcdz[11] = (k11*C[0]*C[10])/vesc  
  
        dcdz[12] = (k12*C[0])/vesc  
  
        dcdz[13] = (k13*C[0])/vesc
```

```
dcdz[14] = 0  
cnt2=0  
while cnt2<15:  
    C[cnt2]=C[cnt2]+dcdz[cnt2]*self.dz  
    cnt2=cnt2+1  
    cnt1=cnt1+1  
res.append(C)  
z=z+self.dz  
return res
```


Anexo 2. Reciclo

```
def simulaComReciclo(T,P,vz_STP,C_et0,inerte,purga=0,vet=np.zeros((2,20)),tv=20):
```

```
    vz_i=vz_STP #vazão volumétrica inicial de etanol "novo", sempre igual à vazão de
    etanol depois do reciclo
```

```
    vm_i=vz_STP*C_et0 #vazão molar inicial de etanol "novo"
```

```
    ocm = OCM2(T,P,vz_STP,C_et0,inerte)
```

```
    df = ocm.simula(0)
```

```
    dfa= ocm.simula(0)
```

```
    R=8.314
```

```
    dfa[-1][0]=0
```

```
    cnt1=0
```

```
    while cnt1<tv:
```

```
        vm_et0=df[-1][0]*vz_i*(1-purga) #Vazão molar de etanol reciclado [mol/s]
```

```
    vz_et0=vz_i*df[-1][0]*(1-purga)/(df[-1][0]+df[-1][1]+df[-1][2]+df[-1][3]+df[-1][4]+df[-1][5]+df[-1][6]+df[-1][7]+df[-1][8]+df[-1][9]+df[-1][10]+df[-1][11]+df[-1][12]+df[-1][13]) #Vazão
    volumétrica de etanol reciclado [m3/s]
```

```
    vz_STP=vz_i-vz_et0 #vazão volumétrica de etanol "novo"
```

```
    vet[0][cnt1]=vz_et0
```

```
    vet[1][cnt1]=vz_STP#vazão volumétrica de etanol "novo"
```

```
#print(f'vm_et0={vm_et0}, vz_et0={vz_et0}, vz_STP={vz_STP}, vm_i={vm_i},  
(vm_et0+vz_STP*C_et0)/vz_i={(vm_et0+vz_STP*C_et0)/vz_i}, vz_i={vz_i}')  
  
ocm = OCM2(T,P,vz_i,(vm_et0+vz_STP*C_et0)/vz_i,inerte)  
  
dfa=df  
  
df = ocm.simula(0)  
  
cnt1=cnt1+1  
  
return df
```

Anexo 3. Otimização do Processo

```
def fobj(X):  
    a,b,c,d=X; #temperatura,pressão,vazão,fração molar de etanol  
    a=lim(a,300,1000)  
    b=lim(b,100000,1000000)  
    c=lim(c,0,0.001)  
    d=lim(d,0,1)  
    Ctot=b/(8.314*a)  
    C_et0=d*Ctot  
    C_in=Ctot-C_et0  
    tvet=20  
    vet=np.zeros((2,tvet))  
    df = simulaComReciclo(a,b,c,C_et0,C_in,0)  
    vC2=df[-1][0]  
    vC4=df[-1][1]  
    vC6=df[-1][2]  
    vC8=df[-1][4]  
    #v=(vC4*vmC4+vC6*vmC6+vC8*vmC8)/(vmC2*df[0][0])  
    v2=((vC4*vmC4+vC6*vmC6+vC8*vmC8)*c-C_et0*vet[1][-1]*vmC2)*3600*24
```

```
#print(df)
```

```
return -v2
```

Anexo 4. Interface Gráfica

```
def
simula_graf(temperatura,pressao,vazao,concentracao_etanol,concentracao_inerte,reciclo)
:

    fim=-1

    if reciclo:

res=simulaComReciclo(temperatura,pressao,vazao,concentracao_etanol,concentracao_in
erte)

        conv_etanolr=np.zeros((2,len(res)))

        yC4r=np.zeros((2,len(res)))

        yC6r=np.zeros((2,len(res)))

        yC8r=np.zeros((2,len(res)))

        cnt1=0

        while cnt1<len(res):

conv_etanolr[1][cnt1]=100*(1-2*res[cnt1][0]/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*
res[cnt1][3]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*re
s[cnt1][9]+2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13]))

yC4r[1][cnt1]=100*(4*res[cnt1][1])/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][
3]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]
+2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13])
```

```
yC6r[1][cnt1]=100*(6*res[cnt1][2])/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][3]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]+2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13])
```

```
yC8r[1][cnt1]=100*(8*res[cnt1][4])/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][3]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]+2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13])
```

```
conv_etanolr[0][cnt1]=cnt1/len(res)
```

```
yC4r[0][cnt1]=cnt1/len(res)
```

```
yC6r[0][cnt1]=cnt1/len(res)
```

```
yC8r[0][cnt1]=cnt1/len(res)
```

```
cnt1=cnt1+1
```

```
plt.plot(conv_etanolr[0][:fim],conv_etanolr[1][:fim],label='C2 conversion')
```

```
plt.plot(yC4r[0][:fim],yC4r[1][:fim],label='C4 yield')
```

```
plt.plot(yC6r[0][:fim],yC6r[1][:fim],label='C6 yield')
```

```
plt.plot(yC8r[0][:fim],yC8r[1][:fim],label='C8 yield')
```

```
plt.legend()
```

```
plt.ylim([0,100])
```

```
plt.show()
```

```
return (conv_etanolr[1][-2],yC4r[1][-2],yC6r[1][-2],yC8r[1][-2]) #Na saída: conversão de etanol, seletividades de C4, C6 e C8
```

```
simulador=OCM2(temperatura,pressao,vazao,concentracao_etanol,concentracao_inerte)

res=simulador.simula(0)

conv_etanol=np.zeros((2,len(res)))

yC4=np.zeros((2,len(res)))

yC6=np.zeros((2,len(res)))

yC8=np.zeros((2,len(res)))

#estrutura: res[posição][componente]

C_et=np.zeros(len(res))

cnt1=0

conv_etanol=np.zeros((2,len(res)))

yC4=np.zeros((2,len(res)))

yC6=np.zeros((2,len(res)))

yC8=np.zeros((2,len(res)))

while cnt1<len(res):

conv_etanol[1][cnt1]=100*(1-2*res[cnt1][0]/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][3]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]+2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13]))

yC4[1][cnt1]=100*(4*res[cnt1][1])/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][3]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]+
```

```
2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13])
```

```
yC6[1][cnt1]=100*(6*res[cnt1][2])/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][3]  
]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]+  
2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13])
```

```
yC8[1][cnt1]=100*(8*res[cnt1][4])/(2*res[cnt1][0]+4*res[cnt1][1]+6*res[cnt1][2]+6*res[cnt1][3]  
]+8*res[cnt1][4]+8*res[cnt1][5]+2*res[cnt1][6]+4*res[cnt1][7]+6*res[cnt1][8]+8*res[cnt1][9]+  
2*res[cnt1][10]+4*res[cnt1][11]+6*res[cnt1][12]+8*res[cnt1][13])
```

```
conv_etanol[0][cnt1]=cnt1/len(res)
```

```
yC4[0][cnt1]=cnt1/len(res)
```

```
yC6[0][cnt1]=cnt1/len(res)
```

```
yC8[0][cnt1]=cnt1/len(res)
```

```
cnt1=cnt1+1
```

```
'''
```

```
plt.plot(conv_etanol[0][:fim],conv_etanol[1][:fim],label='C2 conversion')
```

```
plt.plot(yC4[0][:fim],yC4[1][:fim],label='C4 yield')
```

```
plt.plot(yC6[0][:fim],yC6[1][:fim],label='C6 yield')
```

```
plt.plot(yC8[0][:fim],yC8[1][:fim],label='C8 yield')
```

```
plt.legend()
```

```
plt.ylim([0,100])
```



```
plt.show()

'''

    return (conv_etanol[1][-2],yC4[1][-2],yC6[1][-2],yC8[1][-2]) #Na saída: conversão de
    etanol, seletividades de C4, C6 e C8

global ret

def iniciar():

    saida=simula_graf(float(t1.get()),float(t2.get()),float(t3.get()),float(t4.get()),float(t5.get()),rec.
    get()=='Com Reciclo')

    fim=4

    if rec.get()=='Sem Reciclo':

        s6.set("Saída:\nConversão de etanol: "+str(saida[0][:fim])+" %\nRendimento de
        n-butanol: "+str(saida[1][:fim])+" %\nRendimento de n-hexanol: "+str(saida[2][:fim])+"
        %\nRendimento de n-octanol: "+str(saida[3][:fim])+" %")

    if rec.get()=='Com Reciclo':

        s6.set("Saída:\nConversão de etanol: "+str(saida[0][:fim])+" %\nRendimento de
        n-butanol: "+str(saida[1][:fim])+" %\nRendimento de n-hexanol: "+str(saida[2][:fim])+"
        %\nRendimento de n-octanol: "+str(saida[3][:fim])+" %\n*Valores referentes à saída do
        reator")

top=tkinter.Tk()

cnv = tkinter.Canvas(top, bg="white", height=300, width=400)
```

```
b1 = tkinter.Button(top, text ="Iniciar",command=iniciar)
```

```
s1=tkinter.StringVar()
```

```
s1.set("Temperatura (K):")
```

```
l1 = tkinter.Label(top,textvariable=s1)
```

```
s2=tkinter.StringVar()
```

```
s2.set("Pressão (Pa):")
```

```
l2 = tkinter.Label(top,textvariable=s2)
```

```
s3=tkinter.StringVar()
```

```
s3.set("Vazão volumétrica (m3/s):")
```

```
l3 = tkinter.Label(top,textvariable=s3)
```

```
s4=tkinter.StringVar()
```

```
s4.set("Concentração de etanol na entrada (mol/m3)")
```

```
l4 = tkinter.Label(top,textvariable=s4)
```

```
s5=tkinter.StringVar()
```

```
s5.set("Concentração de inerte na entrada (mol/m^3)")
```

```
l5 = tkinter.Label(top,textvariable=s5)
```

```
s6=tkinter.StringVar()
```

```
s6.set("Saída:")
```

```
l6 = tkinter.Label(top,textvariable=s6)
```

```
t1=tkinter.Entry(top)
```

```
t2=tkinter.Entry(top)
```

```
t3=tkinter.Entry(top)
```

```
t4=tkinter.Entry(top)
```

```
t5=tkinter.Entry(top)
```

```
rec = ttk.Combobox(top,values=["Sem Reciclo",'Com Reciclo'],width=15,height=30)
```

```
rec.place(x=270,y=180)
```

```
rec.current(0)
```

```
entradas=[t1,t2,t3,t4,t5];
```

b1.place(x=200,y=180)

l1.place(x=0,y=30)

l2.place(x=0,y=60)

l3.place(x=0,y=90)

l4.place(x=0,y=120)

l5.place(x=0,y=150)

l6.place(x=0,y=210)

t1.place(x=270,y=30)

t2.place(x=270,y=60)

t3.place(x=270,y=90)

t4.place(x=270,y=120)

t5.place(x=270,y=150)

cnv.pack()

top.mainloop()