



UNIVERSIDADE ESTADUAL PAULISTA
“JÚLIO DE MESQUITA FILHO”
FACULDADE DE CIÊNCIAS FARMACÊUTICAS
Curso de Graduação Engenharia de Bioprocessos e Biotecnologia

MAYRA MANDOLESI VALVANO

**DESENVOLVIMENTO DE ALGORITMOS DE SIMULAÇÃO E
MODELAGEM COMPUTACIONAL DE REAÇÕES ENZIMÁTICAS EM
BATELADA EM LINGUAGEM PYTHON**

Araraquara, SP

2023

MAYRA MANDOLESI VALVANO

**DESENVOLVIMENTO DE ALGORITMOS DE SIMULAÇÃO E
MODELAGEM COMPUTACIONAL DE REAÇÕES ENZIMÁTICAS EM
BATELADA EM LINGUAGEM PYTHON**

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Bioprocessos e Biotecnologia da Faculdade de Ciências Farmacêuticas de Araraquara, da Universidade Estadual Paulista “Júlio de Mesquita Filho”, para obtenção do grau de Engenheira de Bioprocessos e Biotecnologia.

Orientador: Prof. Dr. Marcel Otávio Cerri

Araraquara, SP

2023

V215d Valvano, Mayra Mandolesi.
Desenvolvimento de algoritmos de simulação e modelagem computacional de reações enzimáticas em batelada em linguagem *Python* / Mayra Mandolesi Valvano. – Araraquara: [s.n.], 2023. 85 f. : il.

Orientador: Marcel Otávio Cerri.

Trabalho de Conclusão de Curso (Graduação – Engenharia de Bioprocessos e Biotecnologia) – Universidade Estadual Paulista “Júlio de Mesquita Filho”. Faculdade de Ciências Farmacêuticas.

1. Modelagem matemática. 2. Simulação. 3. Enzimas. 4. Cinética enzimática. 5. Computação científica. 6. Python. I. Cerri, Marcel Otávio, orient. II. Título.

DEDICATÓRIA

*Dedico este trabalho a minha mãe (in memoriam) e
a meu pai por todo apoio sempre.*

AGRADECIMENTOS

Primeiramente gostaria de agradecer aos meus pais e minha irmã, Isadora, por todo incentivo e apoio incondicional durante todos esses anos.

Agradeço aos meus amigos e amigas por todo carinho, companheirismo, paciência e pelos momentos divertidos ao longo de todos esses anos.

Obrigada também aos professores por todos os ensinamentos durante a graduação. Agradecimento especial ao meu orientador Prof. Dr. Marcel Cerri por todo suporte, paciência e dedicação, sempre disponível a compartilhar todo seu conhecimento.

RESUMO

Com o surgimento de novas tecnologias computacionais, a modelagem e simulação de sistemas tem se apresentado como alternativa economicamente vantajosa para a otimização, automação e controle de diferentes processos industriais. Capazes de compreender o comportamento de bioprocessos e determinar as melhores condições de operação, as simulações são definidas como representações de sistemas reais por meio de modelos matemáticos. Entre os modelos matemáticos, obtidos através da análise de balanço de massa dos componentes reacionais, destacam-se os modelos cinéticos, dentre os quais sobressai-se o modelo de Michaelis-Menten, que representa as taxas de catálises enzimáticas, relacionando proporcionalmente a velocidade reacional catalisada com a concentração, em mol/L, de substrato no meio, em duas etapas de reação. As enzimas, catalisadores biológicos, configuram-se como alternativas vantajosas aos catalisadores químicos em diversas áreas industriais, visto sua alta eficiência, velocidade e sustentabilidade. Desse modo, o presente trabalho teve como objetivo o desenvolvimento de algoritmos de computação científica para simulação e modelagem de reações enzimáticas realizadas em batelada. O *software Enzinetics* foi desenvolvido em linguagem Python, com cinéticas enzimáticas já estabelecidas na literatura – Cinética de Michaelis-Menten e os modelos adaptados com Inibição Competitiva, Inibição Não Competitiva e Inibição Incompetitiva, além da linearização de Lineweaver-Burk. A Interface Gráfica de Usuário do *software* foi desenvolvida através da biblioteca *Tkinter*, de modo *user-friendly*, permitindo sua fácil utilização sem prévios conhecimentos. As documentações dos modelos utilizados, assim como referências bibliográficas, também compõem a aplicação, facilitando ainda mais o estudo das cinéticas enzimáticas.

Palavras-chave: Modelagem matemática. Simulação. Enzimas. Cinética enzimática. Computação Científica. Python.

ABSTRACT

With the development of new computational technologies, the modeling and simulation of systems has been presented as an advantageous economic alternative for the optimization, automation and control of industrial processes. Capable of understanding the behavior of bioprocesses and determining the best operating conditions, the simulations are defined as representations of real systems through mathematical models, obtained from the mass and energy balances of the reaction components. Among the mathematical models, kinetic models stand out, among which the Michaelis-Menten model stands out, which represent the enzymatic catalysis rates, proportionally relating the reaction speed with the substrate concentration, in two reaction stages. Enzymes, biological catalysts, are advantageous alternatives to chemical catalysts in several industrial areas, given their high efficiency, speed and sustainability. Thus, the present work aimed to develop scientific computing algorithms for simulation and modeling of enzymatic reactions in batch operation. The Enzinetics software was developed in Python language, with enzymatic kinetics models already established in the literature – Michaelis-Menten Kinetics and adapted models with Competitive Inhibition, Non-Competitive Inhibition and Uncompetitive Inhibition, in addition to Lineweaver-Burk linearization. The software's Graphic User Interface was developed using the Tkinter library, in a user-friendly way, allowing its easy use without prior knowledge. The documentation of the models used, as well as bibliographical references, also make up the application, further facilitating the study of enzymatic kinetics.

Keywords: Mathematical modeling. Simulation. Enzymes. Enzymatic kinetics. Scientific computing. Python.

LISTA DE FIGURAS

Figura 1. Modelo de complementariedade geométrica entre enzima e substrato, chave-fechadura, proposto por Emil Fisher.....	16
Figura 2. Representação gráfica da velocidade de uma reação enzimática em função da concentração de substrato, segundo modelo de Michaelis-Menten.....	25
Figura 3. Representação esquemática dos mecanismos de inibição reversíveis de reações enzimáticas.....	26
Figura 4: Representação gráfica da Linearização de Lineweaver-Burk do Modelo cinético de Michaelis-Menten.....	32
Figura 5. Representação esquemática do modo de operação em batelada, empregada em biorreatores do tipo STR.....	35
Figura 6. Componentes essenciais da Computação Científica.....	37
Figura 7. Estrutura base de um programa escrito em linguagem Python e a ordem de execução dos comandos.....	38
Figura 8. Estrutura de interpretação de códigos desenvolvidos em linguagem Python.....	39
Figura 9. <i>Layout</i> inicial do <i>software</i> comercial <i>Anaconda Navigator</i>	41
Figura 10. Componentes base do IDE <i>Jupyter Notebook</i> e suas interações.....	42
Figura 11. <i>Layout</i> da interface gráfica <i>open source</i> <i>Jupyter Notebook</i>	42
Figura 12. <i>Layout</i> do Ambiente de Desenvolvimento Integrado (IDE) <i>PyCharm</i>	43
Figura 13. Bibliotecas Python fundamentais para Computação Científica.....	45
Figura 14. Logo da biblioteca Python <i>open source</i> <i>NumPy</i>	46
Figura 15. Logo da biblioteca científica Python <i>open source</i> <i>SciPy</i>	47
Figura 16. Logo da biblioteca científica Python <i>open source</i> <i>Pandas</i>	48
Figura 17. Estruturas básicas da biblioteca <i>Pandas</i>	49

Figura 18. Logo da biblioteca científica Python <i>open source Matplotlib</i>	50
Figura 19. Representação da estrutura base de Interface Gráfica de Usuário (GUI).....	51
Figura 20. Tela inicial do <i>software Enzinetics</i> desenvolvido em Python.....	55
Figura 21. Opções da barra de menu da tela inicial do <i>software Enzinetics</i>	56
Figura 22. Interface Gráfica Sobre, que exibe informações gerais do <i>software Enzinetics</i> , iniciada a partir de sua seleção no menu principal.....	56
Figura 23. Interface gráfica da Documentação do modelo cinético enzimático de Michaelis-Menten.....	57
Figura 24. Interface gráfica da Documentação da Definição de Inibição Enzimática e introdução aos Modelos de Inibição Enzimática: Competitiva, Não Competitiva e Incompetitiva.....	58
Figura 25. Interface gráfica da Documentação do modelo de Inibição Enzimática Competitiva.....	59
Figura 26. Interface gráfica da Documentação do modelo de Inibição Enzimática Não Competitiva.....	59
Figura 27. Interface gráfica da Documentação do modelo de Inibição Enzimática Incompetitiva.....	60
Figura 28. Interface gráfica da Documentação do modelo linearizado de Lineweaver-Burk.....	61
Figura 29. Interface gráfica da Documentação do modo de operação em batelada..	62
Figura 30. Interface gráfica de referências bibliográficas utilizadas para a elaboração da fundamentação teórica de toda a seção de Documentação.....	63
Figura 31. Interface gráfica da sessão do Modo: Simulação.....	64
Figura 32. Interface gráfica da sessão do Modo: Simulação, com a seleção do modelo de Simulação com Inibição Competitiva.....	65

Figura 33. Interface gráfica da sessão do Modo: Simulação, com a seleção do modelo de Simulação com Inibição Não-Competitiva.....	66
Figura 34. Interface gráfica da sessão do Modo: Simulação, com a seleção do modelo de Simulação com Inibição Incompetitiva.....	67
Figura 35. Interface gráfica da sessão do Modo: Simulação, com a exemplificação da exibição gráfica, a partir da inserção de valores aleatórios para os parâmetros cinéticos.....	68
Figura 36. Interface gráfica da sessão modelagem, na qual são inseridos dados experimentais e estimados os parâmetros cinéticos.....	69
Figura 37. Arquivo <i>excel</i> aberto ao selecionar o botão Baixar Template. O template deve ser utilizado como padrão pelo usuário para a importação dos dados experimentais para a modelagem da reação enzimática.....	70
Figura 38. Arquivo <i>excel</i> gerado pelo <i>software Enzinetics</i> com os valores aleatórios não-reais inseridos no sistema: $K_m = 2$, $E_0 = 5 \text{ mol/L}$, $K_{cat} = 3 \text{ 1/s}$, $S_0 = 100 \text{ mol/L}$ e $P_0 = 0 \text{ mol/L}$	72
Figura 39. Modelagem realizada pelo <i>software Enzinetics</i> , com retorno dos valores cinéticos estimados, que conferem com os valores inicialmente simulados.....	73

LISTA DE ABREVIATURAS E SIGLAS

API	Interface de programação de aplicações
GUI	Interfaces Gráficas de Usuário
IDE	Ambiente de Desenvolvimento Integrado
<i>Ndarray</i>	N-dimensional array
<i>NumPy</i>	Numerical Python
<i>Pandas</i>	Python Data Analysis
<i>PyPI</i>	Python Package Index
STR	Stirred Tank Reactor – Biorreator de tanque agitado
AG	Algoritmo Genético
[E]	Concentração de Enzimas
[S]	Concentração de Substrato
[P]	Concentração de Produto
[ES]	Concentração de Complexo Enzima-Substrato
[E ₀]	Concentração Inicial de Enzimas
[I]	Concentração de Inibidores
[EI]	Concentração de Complexo Enzima-Inibidor
[EIS]	Concentração de Complexo Enzima-Inibidor-Substrato
K _s	Constante de Equilíbrio
K _M	Constante de Michaelis-Menten
V _{max}	Velocidade Máxima de Reação
K _{M_ap}	Constante de Michaelis-Menten aparente
V _{max_ap}	Velocidade Máxima de Reação aparente

K_1	Constante de formação do complexo enzima-substrato
K_{-1}	Constante de dissociação do complexo enzima-substrato
K_2	Constante de formação de produto (catálise ou <i>turnover</i>)
K_{cat}	Constante de formação de produto (catálise ou <i>turnover</i>).
K_i	Constante de formação do complexo enzima-inibidor
V_P	Velocidade de formação de produto - catálise
V_{forES}	Velocidade de formação do complexo enzima-substrato.
V_{disES}	Velocidade de dissociação do complexo enzima-substrato
V	Volume
$[S]_0$	Concentração Inicial de Substrato
$[P]_0$	Concentração Inicial de Produto
$[S]_f$	Concentração Final de Substrato
$[P]_f$	Concentração Final de Produto
r	Velocidade de reação
T_0	Tempo Inicial de reação
T_f	Tempo Final de reação
$\frac{d[S]}{dt}$	Taxa de Consumo de Substrato
$\frac{d[P]}{dt}$	Taxa de Formação de Produto

SUMÁRIO

1. INTRODUÇÃO	14
2. FUNDAMENTAÇÃO TEÓRICA.....	16
2.1 ENZIMAS	16
2.2 ENZIMAS: APLICAÇÕES INDUSTRIAIS.....	17
2.3 CINÉTICA ENZIMÁTICA.....	19
2.4 SIMULAÇÃO E MODELAGEM	20
2.4.1 CINÉTICA DE MICHAELIS-MENTEN.....	21
2.4.1.1 INIBIÇÃO ENZIMÁTICA	25
2.4.1.1.1 INIBIÇÃO COMPETITIVA.....	26
2.4.1.1.2 INIBIÇÃO NÃO COMPETITIVA	27
2.4.1.1.3 INIBIÇÃO INCOMPETITIVA	29
2.4.1.1.4 LINEWEAVER-BURK	31
2.4.2 ALGORITMO GENÉTICO.....	32
2.4.3 ALGORITMO DE LEVENBERG-MARQUARDT	33
2.4.3 BIORREATORES	34
2.4.3.1 OPERAÇÃO EM BATELADA	35
2.5 COMPUTAÇÃO CIENTÍFICA.....	37
2.6 LINGUAGEM COMPUTACIONAL PYTHON	38
2.7 AMBIENTES DE DESENVOLVIMENTO INTEGRADOS (IDE).....	40
2.7.1 ANACONDA	40
2.7.2 JUPYTER NOTEBOOK	41
2.7.3 PYCHARM.....	43
2.8 BIBLIOTECAS PYTHON.....	44
2.8.1 IMPORTAÇÃO E ATUALIZAÇÃO DAS BIBLIOTECAS.....	44
2.9 BIBLIOTECAS PYTHON PARA COMPUTAÇÃO CIENTÍFICA.....	45
2.9.1 NUMPY	46
2.9.2 <i>SCIPY</i>	47
2.9.3 <i>PANDAS</i>	48
2.9.4 MATPLOTLIB	50
2.9.5 TKINTER	51
3 OBJETIVOS	52
3.1 OBJETIVOS GERAIS	52

3.2 OBJETIVOS ESPECÍFICOS.....	52
4 MATERIAIS E MÉTODOS.....	53
5 RESULTADOS E DISCUSSÃO.....	54
5.1 SOFTWARE ENZINETICS	54
5.2 SOBRE	56
5.3 DOCUMENTAÇÃO	57
5.3.1 DOCUMENTAÇÃO MICHAELIS-MENTEN	57
5.3.2 DOCUMENTAÇÃO INIBIÇÃO ENZIMÁTICA.....	58
5.3.3 DOCUMENTAÇÃO LINEWEAVER-BURK	60
5.3.4 DOCUMENTAÇÃO MODO DE OPERAÇÃO EM BATELADA.....	61
5.3.5 DOCUMENTAÇÃO REFERÊNCIAS BIBLIOGRÁFICAS	62
5.4 MODO: SIMULAÇÃO	63
5.4.1 SIMULAÇÃO SEM INIBIÇÃO.....	65
5.4.2 SIMULAÇÃO COM INIBIÇÃO COMPETITIVA.....	65
5.4.3 SIMULAÇÃO COM INIBIÇÃO NÃO-COMPETITIVA.....	66
5.4.4 SIMULAÇÃO COM INIBIÇÃO INCOMPETITIVA.....	66
5.4.5 EXIBIÇÃO GRÁFICA	67
5.5 MODO: MODELAGEM.....	69
5.5.1 MODELAGEM: BACK-END	71
5.5.2 MODELAGEM: TESTE FINAL	72
6. CONCLUSÃO	73
7. REFERÊNCIAS BIBLIOGRÁFICAS	75

1. INTRODUÇÃO

Os avanços tecnológicos e computacionais abriram espaço para que a modelagem e simulação de processos se destacassem no âmbito industrial, apresentando-se como alternativa extremamente viável do ponto de vista econômico e de gestão de qualidade. Assim, possibilitam não apenas a determinação das melhores condições operacionais (SMITH, 2022), como a automação, monitoramento e o controle de processos, garantindo maior eficiência e confiabilidade industrial (JONES *et al*, 2021).

As simulações de processos, definidas como representações de sistemas reais por meio de modelagem matemática (SAWAKI *et al*, 2015), são capazes de compreender o comportamento dos mesmos, promovendo diversos cenários experimentais (MENDESA; VALDMANA; DE SOUZA JÚNIOR, 2011). Utilizadas na otimização, escalonamento, automação e controle de processos (CRISTOVÃO *et al*, 2021), permitem o planejamento de experimentos e a predição empírica do comportamento dinâmico e estacionário de sistemas, possibilitando a determinação de melhores estratégias de operação (MENDOZA *et al*, 2022; OLIVEIRA *et al*, 2020).

De acordo com Schwaab e Pinto (2007), os modelos matemáticos, por sua vez, são representações físicas de sistemas através de conjuntos de equações algébrico-lineares. Obtidos através do balanço de massa e balanço de energia dos diferentes elementos presentes na reação, os modelos consideram as transformações bioquímicas e suas velocidades, assim como as taxas de troca de calor e transferência de massa, além de outras propriedades afetadas durante o processo (SCHMIDELL *et al*, 2001). Assim, são estabelecidas relações entre os parâmetros do sistema, podendo-se prever empiricamente as variáveis de saída, parâmetros cinéticos de reação e o estado do sistema apenas com os possíveis dados de entrada, através de técnicas de ajuste de equação a pontos experimentais (MENDESA; VALDMANA; DE SOUZA JÚNIOR, 2011).

O advento da biotecnologia tem destacado as técnicas de simulação e modelagem computacional como ferramentas para escalonamento de diversos bioprocessos, como a otimização de reações enzimáticas (VASIC-RACKI; KRAGL;

LIESE, 2003). As enzimas, moléculas catalisadoras de reações químicas, são responsáveis por tornar os processos não apenas mais rápidos, como mais eficientes e sustentáveis. Configuram-se, portanto, como alternativas vantajosas aos catalisadores químicos em diferentes áreas industriais (MONTEIRO; SILVA, 2009). Com o crescente uso das enzimas em larga escala, seu estudo e compreensão tornam-se fundamentais para o desenvolvimento de novos processos biotecnológicos (DORAN, 2013).

Assim, as características das enzimas mais importantes do ponto de vista industrial relacionam-se ao estudo cinético de seu processo catalítico. Ou seja, foca-se na sequência de eventos mecanísticos que ocorrem durante a transformação dos substratos em produtos, e aos fatores que o influenciam (VASIC-RACKI; KRAGL; LIESE, 2003). A determinação das velocidades de reação e dos parâmetros operacionais ótimos, como concentração de substrato e enzimas, temperatura e pressão, permitem maior conhecimento acerca do mecanismo de ação das mesmas, sendo essenciais para a maior eficiência das biocatálises e, conseqüentemente, dos processos em que são empregadas (PINTO; DE MENEZES, 2009).

Entre os modelos de cinética enzimática já estabelecidos na literatura, destaca-se o modelo mecanístico de Michaelis-Menten, o qual considera substrato e produto únicos na reação (PINTO; DE MENEZES, 2009). Neste modelo, proposto no ano de 1913, por Leonor Michaelis e Maud Menten, são descritas equações que representam as taxas de reações enzimáticas, relacionando proporcionalmente a velocidade reacional catalisada pela enzima com a concentração, em mol/L, de substrato no meio, em duas etapas de reação. Na primeira, reversível, o equilíbrio entre substrato, o reagente e a enzima é alcançado com a formação de um complexo enzima-substrato. Esse complexo é convertido em produto na segunda etapa reacional, irreversível, juntamente com a regeneração da enzima (DORAN, 2013).

Nesse sentido, o presente trabalho visa desenvolver algoritmos de simulação e modelagem de reações enzimáticas em um ambiente computacional em linguagem Python. A ampla utilização da linguagem, sua disponibilidade gratuita e a possibilidade de realização de tratamento de dados numéricos e determinação de soluções algébricas com facilidade, são fatores determinantes para sua predileção (SAYAMA, 2015).

2. FUNDAMENTAÇÃO TEÓRICA

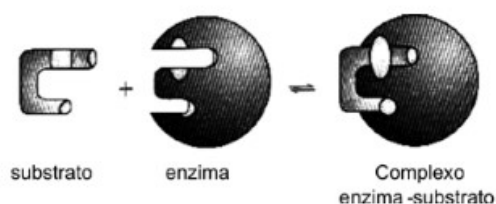
2.1 ENZIMAS

Enzimas são proteínas complexas, formadas por uma porção proteica que pode estar associada a outras moléculas, como lipídeos e carboidratos. Elas atuam como catalisadoras biológicas em reações químicas, acelerando a velocidade dessas reações sem serem consumidas ou alteradas no processo (JOHNSON, 2021). As enzimas apresentam estruturas moleculares primária, secundária, terciária e quaternária, e são altamente específicas para seus substratos reagentes (SMITH *et al.*, 2020).

Em relação à sua ação, as enzimas reduzem a chamada energia de ativação das reações catalisadas – diferença energética entre o estado fundamental e de transição dos reagentes, permitindo que o estado energético de transição seja atingido mais rapidamente, assim como a formação do produto, em uma reação que recebe o nome de catálise (MONTEIRO; SILVA, 2009). O estado intermediário deve ser atingido para que a reação ocorra, pois é o estado de maior energia livre, como descrito na Teoria do Estado de Transição, proposta por Eyring e Daniels (1930).

A alta especificidade, responsável pela ligação e ação das enzimas, ocorre, segundo Emil Fischer (1894), devido a complementariedade geométrica entre a enzima e seu substrato. Ambos se ligam, criando o complexo enzima-substrato, representado graficamente pela Figura 1 abaixo. Este modelo ficou conhecido como modelo “chave-fechadura” (RINGE; PETSKO, 2008).

Figura 1. Modelo de complementariedade geométrica entre enzima e substrato, chave-fechadura, proposto por Emil Fisher.



Fonte: MONTEIRO; SILVA, 2009.

Assim, fica evidente que a atividade catalítica das enzimas está diretamente ligada à sua estrutura proteica original e, portanto, se a enzima sofrer o processo de desnaturação ou for desestruturada em subunidades, ela perde a sua função (VIEIRA, 2003). O processo de desnaturação refere-se à perda ou alteração da estrutura tridimensional das enzimas devido a fatores como temperaturas elevadas, pH extrema ou presença de agentes desnaturantes.

Cada enzima possui valores ótimos para pH e temperatura, sendo sensíveis às suas variações extremas. Em altas temperaturas, ocorre o rompimento de ligações e interações fracas, alterando a forma estrutural da enzima com a perda da estrutura terciária, processo irreversível denominado de desnaturação (SANT'ANNA JUNIOR, 2001). A alteração de pH, por sua vez, pode ocasionar mudanças de ionizações e cargas dos grupos químicos que compõem as enzimas, também afetando sua forma (MONTEIRO; SILVA, 2009).

O crescente uso de enzimas frente aos catalisadores químicos pode ser justificado por se caracterizarem como produtos naturais biológicos, biodegradáveis e ambientalmente sustentáveis de alta especificidade em reações. Devido ao seu baixo custo de investimento, facilidade de controle e eficiência energética, as enzimas vêm ganhando espaço no mercado e são aplicadas em diferentes processos industriais (PATEL, 2002; PIZARRO; PARK, 2003).

2.2 ENZIMAS: APLICAÇÕES INDUSTRIAIS

A constante busca pela redução de custos e melhoria na qualidade final do produto impulsiona diferentes segmentos industriais a investir em alternativas promissoras e menos onerosas. Deste modo, o mercado de enzimas foi marcado por elevado crescimento nos últimos anos, com potencial para expansão pelo desenvolvimento de novos campos de pesquisa e tecnologias (SILVA, 2012).

O emprego de enzimas no mercado industrial está relacionado às diversas áreas da ciência e tecnologia e objetiva a melhoria de processos, a utilização de novas matérias-primas e melhoria de suas qualidades físico-químicas. Assim sendo, enzimas comercialmente utilizáveis são aquelas que garantem um produto final

superior ao antes produzido e a redução de custos, além da possibilidade de obtenção de produtos anteriormente indisponíveis no mercado (ABRAHÃO NETO, 2001).

Característica positiva da utilização de catalisadores enzimáticos em processos industriais é sua atuação em pH e temperaturas brandas, o que facilita o processo de fabricação de produtos e diminui o custo para a indústria, tornando as enzimas biocatalisadores de alto potencial biotecnológico (MONTEIRO; SILVA, 2009).

Embora as enzimas de origem vegetal e animal sejam foco da maior gama de pesquisas enzimáticas, as enzimas microbianas apresentam um grande potencial industrial, devido à sua capacidade de produção em larga escala por meio de fermentação (SENA *et al.*, 2006; COLEN, 2006).

A indústria de alimentos é a maior consumidora de enzimas para uso industrial em todo o mundo (GARG; SEHRAWAT; YADAV, 2016). As enzimas são empregadas em vários processos para a fabricação de diversos produtos comerciais, como pães, bebidas, laticínios, suplementos e xaropes (FERNANDES, 2010).

Como exemplo, as enzimas são fundamentais na indústria de laticínios, proporcionando benefícios diversos, incluindo a degradação de produtos lácteos. Elas são utilizadas para produzir subprodutos do leite, como queijos, requeijões, além de desenvolver sabores específicos e reduzir o tempo de processo de cura de queijos (TREMONTÉ *et al.*, 2014; VITOLLO, 2001).

Na panificação, o uso de enzimas é essencial pois farinhas possuem baixa atividade enzimática. Adicionando enzimas como proteases e amilases, a manipulação da massa dos pães é facilitada, assim como ocorre a melhoria de suas características, como qualidade de tostagem e paladar, devido ao aumento da disponibilidade de açúcar para a fermentação (VAN DAM; HILLE, 1992).

Na produção de papel e celulose, as enzimas substituem o cloro no branqueamento das pastas, evitando impactos ambientais negativos (PAIVA; SÁ-PEREIRA, 2008). Na indústria de cosméticos, podem ser empregadas para facilitar ou dificultar reações bioquímicas na pele, além de remover ou destruir estruturas dos anexos cutâneos (FOX, 2005). Já na indústria têxtil, por sua vez, os catalisadores enzimáticos são utilizados para processar as fibras e melhorar a qualidade dos tecidos sem prejudicar o meio ambiente (ANDREAUS; CAVACO-PAULO, 2008).

2.3 CINÉTICA ENZIMÁTICA

O estudo cinético é uma importante forma de análise de uma reação química e se caracteriza pela determinação das velocidades das reações e da influência de fatores termodinâmicos, como pressão e temperatura, e do meio reacional sobre essa velocidade (PINTO; DE MENEZES, 2009).

A cinética de uma reação relaciona-se com a sequência de eventos para a transformação dos reagentes em produtos, possibilitando conclusões sobre seu mecanismo de ação. Assim sendo, a cinética enzimática refere-se à área de estudo dos mecanismos de ligação entre as enzimas e os substratos, os quais são transformados em produtos, ou seja, o comportamento da catálise enzimática em relação ao tempo (PINTO; DE MENEZES, 2009).

O conhecimento sobre a cinética enzimática fornece informações sobre o comportamento “in vivo” da enzima, permitindo, conseqüentemente, a previsão do seu mecanismo de ação (PEREIRA, 2015).

A cinética de uma reação enzimática pode ser estudada por um modelo matemático. Determinar uma equação modelo capaz de estimar valores de reação similares aos valores experimentalmente observados é essencial para a otimização de processos. O modelo de um sistema de reações deve considerar todos os parâmetros importantes e, assim que estabelecido, ser capaz de resolver e prever comportamentos comparados com dados experimentais (CUNHA, 2007).

Algumas destas variáveis envolvidas no modelo não são de fácil determinação, como as constantes cinéticas. É necessário, portanto, estimar os parâmetros do modelo, ajustando-os de modo que as previsões se aproximem aos valores experimentais medidos. Para garantir os melhores resultados, são utilizadas funções objetivas que comparam as diferenças entre os valores experimentais reais e os valores propostos a partir dos parâmetros estimados (CARVALHO *et al*, 2010; CUNHA, 2007).

Casos de diferenças significativas entre o previsto pelo modelo e a performance do sistema implicam a possibilidade de existência de outros fatores importantes que

não foram considerados, como por exemplo moléculas inibidoras (BAS; DUDAK; BOYACI, 2007).

2.4 SIMULAÇÃO E MODELAGEM

A modelagem e simulação matemática são frequentemente empregadas para compreensão da cinética das reações. Essa abordagem permite a melhoria de processos e produtos, além de redução nos custos de desenvolvimento, a medida em que avalia o consumo de substratos reagentes e a velocidade de formação de produtos (GONÇALVES, 2019)

No âmbito da engenharia, a palavra *modelo* pode ser definida como uma réplica (matemática ou física) de um sistema, com todas suas funções e propriedades. Dessa forma, modelagem matemática pode ser entendida como um conjunto de equações consistentes para a representação física de um problema (SINGH, 2009), assim dizendo, uma estrutura que procura descrever a realidade por equações (SCHWAAB; PINTO, 2007).

De acordo com o estudo de Mendesa, Valdman e De Souza (2011), a modelagem matemática tem como objetivo encontrar as possíveis relações entre os componentes de um sistema para prever os parâmetros reacionais (*outputs*) com base nos dados de entrada (*inputs*). O conjunto de relações, que consiste em equações algébrico-diferenciais, é chamado de modelo matemático.

Referindo-se a processos biotecnológicos, os modelos matemáticos são uma forma de representação, por meio de equações e balanços de massa, da variação dos componentes reacionais, associados às conversões bioquímicas do processo e suas velocidades (NIELSEN; VILLADSEN; LIDÉN, 2003).

De acordo com Boareto (2012), é fundamental desenvolver modelos matemáticos adequados, uma vez que alguns processos podem apresentar comportamentos oscilatórios. Por meio desses modelos, é possível prever os efeitos e o comportamento de um sistema, mesmo em condições ainda não foram experimentadas. Portanto, esses modelos são úteis para a otimização e controle de processos e podem ajudar a estabelecer condições operacionais e favorecer o desenvolvimento e escalonamento da produção. Além disso, esses modelos auxiliam

na compreensão de processos e na formulação de novas hipóteses (MASHREGI; YAZDI, 2014; SINGH; WOOLHISER, 2002).

A simulação de processos, por sua vez, envolve a representação de um sistema real através de uma modelagem matemática, que permite a criação de um modelo computacional capaz de entender e prever o comportamento do sistema, bem como analisar estratégias de operação em diferentes cenários experimentais. Essa técnica tem sido cada vez mais utilizada devido à sua capacidade de solucionar problemas complexos em diversas áreas de pesquisa, impulsionada pelo aumento do poder de processamento de máquinas (SAWAKI, 2015).

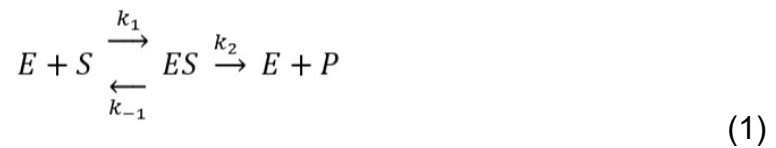
Para o desenvolvimento de um modelo matemático apto a projetar e otimizar processos enzimáticos em condições determinadas devem ser considerados os fenômenos associados aos processos enzimáticos, como a cinética enzimática entre as variáveis do processo e a enzima, estipulada através de experimentações (ALVES, 2010).

O modelo cinético de enzimas mais comumente utilizado foi proposto inicialmente por Leonor Michaelis e Maud Menten em 1913, no artigo intitulado "a cinética da invertase" (LERIN *et al.*, 2012; ROMERO *et al.*, 2011). Com o fortalecimento do mercado de enzimas industriais, esse modelo estabelecido tornou-se fundamental para a enzimologia industrial, pois possibilita a determinação da velocidade e da afinidade entre as enzimas e um substrato específico (MONTEIRO; SILVA, 2009).

2.4.1 CINÉTICA DE MICHAELIS-MENTEN

O modelo cinético enzimático elaborado por Michaelis-Menten (1913) é um dos mais conhecidos na enzimologia e propõe duas etapas na reação catalisada. Na primeira etapa, uma enzima (E) e um substrato (S) se ligam, compondo o complexo enzima-substrato (ES), em uma reação reversível. Na segunda etapa, este complexo (ES) é transformado em produto (P) com a reestruturação da enzima integralmente, como pode ser observado na Equação 1 abaixo.

Em um primeiro momento de reação, estabelece-se um período pré-estacionário, com aumento da concentração dos complexos enzima-substrato. Após este rápido período, a reação atinge seu estado estacionário, como descrito por Briggs e Haldane (1925). No estado estacionário, a concentração dos complexos intermediários torna-se constante. Desse modo, a segunda etapa de reação, a catálise, não pode ser considerada reversível (JOHNSON; GOODY, 2011).



Toda a modelagem matemática a seguir é apresentada seguindo a dedução inicialmente estruturada por Michaelis-Menten (1913), descrita na literatura. Na Equação 1, k_1 e k_{-1} denominam as constantes de formação e de dissociação, respectivamente, da reação do substrato com a enzima ($E+S \rightarrow ES$), enquanto k_2 denota a constante de catálise do complexo intermediário, na segunda etapa da reação ($ES \rightarrow E+P$).

A relação entre as constantes de velocidade de formação e dissociação da primeira fase de reação determinam a constante de equilíbrio, K_S , como mostra a Equação 2 a seguir:

$$K_S = \frac{k_1}{k_{-1}} \quad (2)$$

A equação de Michaelis-Menten assume situação em que a concentração de substrato é maior do que a concentração das proteínas catalisadoras, como demonstra a Equação 2.1 abaixo. Nessa configuração reacional, a velocidade da reação é limitada pela velocidade de catálise do complexo enzima-substrato, como explicado por Martins (2015).

$$[S] \gg [E] \quad (2.1)$$

A concentração total de enzimas, por sua vez, refere-se a soma de enzimas livres e enzimas complexadas. Representada por E_0 , é constante a medida em que as enzimas são recuperadas integralmente após a catálise:

$$[E_0] = [E] + [ES] \quad (3)$$

As velocidades de reação de cada uma das etapas podem ser descritas utilizando-se as concentrações das moléculas e as constantes de cada uma das etapas, como demonstrado pelas Equações 4, 5 e 6 abaixo, que representam, respectivamente, a velocidade de formação de produto, velocidade de formação do complexo enzima-substrato e de dissociação desse complexo intermediário.

$$V_o = V_p = \frac{d[P]}{dt} = k_2 * [ES] \quad (4)$$

$$V_{forES} = \frac{d[ES]}{dt} = k_1 * [E] * [S] \quad (5)$$

$$V_{disES} = \frac{d[ES]}{dt} = k_{-1} * [ES] \quad (6)$$

O balanço de massa do sistema, no estado estacionário, permite estabelecer a relação entre as velocidades das reações descritas acima, como demonstram as Equações 7, 7.1, 7.2 e 7.3 a seguir (MICHAELIS; MENTEN, 1913).

$$V_{forES} - V_p - V_{disES} = 0 \quad (7)$$

$$k_1 * [E] * [S] - k_2 * [ES] - k_{-1} * [ES] = 0 \quad (7.1)$$

$$\frac{[E]*[S]}{[ES]} = \frac{k_2+k_{-1}}{k_1} = K_M \quad (7.2)$$

$$[E] * [S] = K_M * [ES] \quad (7.3)$$

Como demonstrado pela Equação 3 anteriormente, pode-se fazer as devidas substituições na Equação 7.3.

$$([E_0] - [ES]) * [S] = K_M * [ES] \quad (7.4)$$

$$[E_0] * [S] = [ES] * ([S] + K_M) \quad (7.5)$$

$$[ES] = \frac{[E_0]*[S]}{K_M+[S]} \quad (7.6)$$

Como determinado anteriormente, a velocidade de catálise (V_p) é limitante. A partir da Equação 4, pode-se fazer a substituição dos valores na Equação 7.6 descrita acima. Segundo Michaelis-Menten ainda, a constante de catálise (K_2 ou K_{cat}), também pode ser chamada de constante de *turnover*, que se refere a quantidade máxima de

substrato que pode ser convertido em moléculas de produto durante um tempo determinado.

$$\frac{V_o}{k_2} = \frac{[E_0]*[S]}{K_M+[S]} \quad (7.7)$$

Se a concentração de substrato é maior do que a concentração de enzimas (Equação 2.1), é possível atingir uma condição na qual todas as enzimas estão saturadas com substrato, conforme Equação 8. Nessa situação, a velocidade de catálise atinge seu valor máximo e um aumento na concentração de substrato não altera a velocidade da reação, como evidenciado pela Equação 9. Portanto, conclui-se que a velocidade máxima da conversão enzimática depende somente da quantidade de enzima e da constante de *turnover* (MARTINS, 2015).

$$[ES] = [E_0] \quad (8)$$

$$V_{max} = k_2 * [E_0] \quad (9)$$

Desse modo, Michaelis-Menten (1913) descreveram o modelo cinético enzimático, que pode ser apresentado resumidamente pela Equação 10 abaixo, obtida pela substituição da Equação 9 descrita acima, na Equação 7.7.

$$V = \frac{V_{max}*[S]}{K_M+[S]} \quad (10)$$

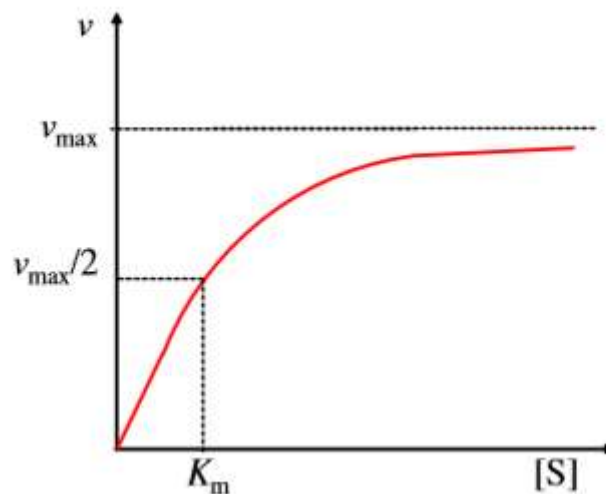
O modelo proposto descreve, portanto, as taxas de reações enzimáticas, relacionando a taxa de reação V , que se refere a taxa de formação de produto em relação a concentração de um substrato ($[S]$) (MARTINS, 2015).

Analisando os componentes do modelo, pode-se concluir que a constante de Michaelis-Menten (K_M) se equivale à concentração de substrato cuja taxa de reação seja metade de V_{max} , sendo característico de cada enzima (MICHAELIS; MENTEN, 1913). Quanto maior o valor da constante, menor a afinidade entre a enzima e o substrato, visto que é necessária uma concentração maior do mesmo para atingir a metade da velocidade máxima.

A Figura 2 apresentada abaixo ilustra a relação gráfica da velocidade reacional e a concentração de substrato catalisado pela enzima, como descrito pelo modelo de

Michaelis-Menten (1913). Quando a quantidade de substrato é alta, a velocidade se aproxima da velocidade máxima (V_{max}) e torna-se independente da concentração do substrato. Em contraste, para baixas concentrações de substrato, a velocidade de início é proporcional à concentração do substrato (LEHNINGER; NELSON; COX, 2012).

Figura 2. Representação gráfica da velocidade de uma reação enzimática em função da concentração de substrato, segundo modelo de Michaelis-Menten.



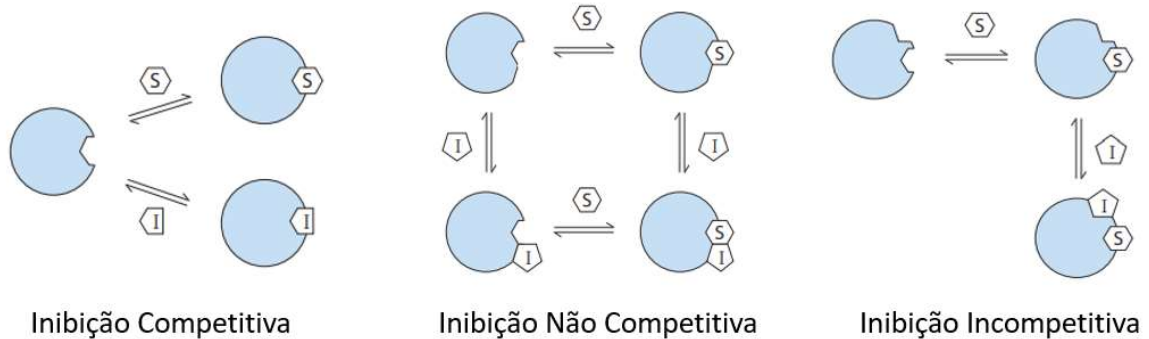
Fonte: PEREIRA *et al*, 2015.

2.4.1.1 INIBIÇÃO ENZIMÁTICA

Como forma de controle e regulação das reações enzimáticas podem ser introduzidas outras moléculas ao sistema, denominadas inibidores, que alteram a cinética enzimática, reduzindo a velocidade da reação, através de diferentes mecanismos (DORAN, 2013; LEHNINGER; NELSON; COX, 2012).

A Figura 3 abaixo mostra esquematicamente os principais modelos de inibição enzimática estabelecidos na literatura, dentre os quais encontram-se os modelos de inibição reversível, entre os quais destacam-se a Inibição Competitiva, Inibição Não-Competitiva e a Inibição Incompetitiva (LEHNINGER; NELSON; COX, 2012).

Figura 3. Representação esquemática dos mecanismos de inibição reversíveis de reações enzimáticas.



Fonte: Adaptado de LEHNINGER, NELSON; COX, 2012.

2.4.1.1.1 INIBIÇÃO COMPETITIVA

Os inibidores competitivos são aqueles que possuem semelhanças estruturais com o substrato da reação, e, dessa forma, ambos competem pelo sítio ativo da enzima. Assim, são formados, também, complexos enzima-inibidor (EI). Essa competição com os inibidores, de concentração I e constante K_i , dificulta a ligação da enzima pelo substrato e, conseqüentemente, a formação de produto (BORZANI *et al*, 2001), exemplificado pelas Equações 11 e 12 descritas abaixo.



Segundo Lehninger, Nelson e Cox (2012) ainda, a velocidade de formação de ambos os complexos, é dependente não apenas da concentração de substrato, mas da concentração de inibidores também e de suas constantes, descritas pelas Equações 13 e 14 abaixo. Nesse sentido, é importante destacar que em condições em que a concentração de substrato é de grandeza muito superior à concentração molecular de inibidores, o efeito dos mesmos torna-se desprezível.

$$K_M = \frac{[E]*[S]}{[ES]} \quad (13)$$

$$K_I = \frac{[E]*[I]}{[EI]} \quad (14)$$

A concentração de enzimas totais, por sua vez, deve considerar, neste caso, não apenas a concentração de enzimas livres e enzimas ligadas ao substrato, como também as enzimas ligadas aos inibidores (DORAN, 2013), como demonstrado abaixo por Equação 15.

$$[E_0] = [E] + [ES] + [EI] \quad (15)$$

Considerando-se a velocidade de catálise do complexo enzima-substrato como a limitante, pode-se dividir a Equação 4, descrita anteriormente, pela Equação 15, obtendo-se:

$$\frac{V}{[E_0]} = \frac{k_2 * [ES]}{[E] + [ES] + [EI]} = \frac{k_2 \frac{[E] * [S]}{K_M}}{[E] + \frac{[E] * [S]}{K_M} + \frac{[E] * [I]}{K_I}} = k_2 \frac{[S]}{K_M + [S] + \frac{[I] * K_M}{K_I}}$$

$$V = V_{max} \frac{[S]}{K_M \left(1 + \frac{I}{K_i}\right) + [S]} \quad (16)$$

Comparando-se a Equação 16 e Equação 10, do modelo original sem inibição de cinética enzimática de Michaelis-Menten, pode-se concluir, portanto, que a inibição competitiva afeta apenas o valor de K_M . Para simplificações, utiliza-se um valor de K_M aparente, como demonstra a Equação 17 abaixo.

$$K_{M_{ap}} = K_M \left(1 + \frac{[I]}{K_i}\right) \quad (17)$$

Assim sendo, o modelo cinético enzimático de inibição competitiva é definido pela Equação 18 a seguir.

$$V = V_{max} \frac{[S]}{K_{M_{ap}} + [S]} \quad (18)$$

2.4.1.1.2 INIBIÇÃO NÃO COMPETITIVA

A inibição do tipo não-competitiva é uma inibição mista que ocorre quando as moléculas inibidoras possuem afinidade com o catalisador enzimático, em um sítio

ativo diferente ao do substrato, e com o complexo enzima-substrato (BORZANI *et al*, 2001). Ocorre, então, a estruturação de um outro tipo de complexo, o enzima-substrato-inibidor (*EIS*), que impede a continuação da reação, diminuindo a velocidade da catálise. Em suma, pode-se afirmar que, concomitantemente, na reação, ocorrem a formação do complexo enzima-substrato, com consequente conversão em produto, do complexo enzima-inibidor e do complexo enzima-substrato-inibidor, que, por sua vez, pode ser seguir dois caminhos distintos de ligação: do inibidor ao complexo enzima-substrato ou do substrato ao complexo enzima-inibidor, como evidenciam as Equações 19, 20, 21 e 22 seguintes (LEHNINGER; NELSON; COX 2012).



A constante de formação de ambos os complexos é dependente da concentração de substrato, de inibidores e dos diferentes complexos formados, como mostram as Equações 23 e 24 abaixo. Na inibição não competitiva as constantes de ligação entre o inibidor e a enzima livre e o inibidor e o complexo enzima-substrato são equivalentes (LEHNINGER; NELSON; COX, 2012).

$$K_M = \frac{[E]*[S]}{[ES]} = \frac{[EI]*[S]}{[EIS]} \quad (23)$$

$$K_I = \frac{[E]*[I]}{[EI]} = \frac{[ES]*[I]}{[EIS]} = \frac{[E]*[S]*[I]}{K_M*[EIS]} \quad (24)$$

Neste caso, a concentração de enzimas totais considera a concentração de enzimas livres, enzimas ligadas ao substrato, enzimas ligadas aos inibidores e enzimas ligadas às duas moléculas, como demonstrado abaixo por Equação 25.

$$[E_0] = [E] + [ES] + [EI] + [EIS] \quad (25)$$

Considerando-se a velocidade de catálise do complexo enzima-substrato como a limitante da reação, pode-se dividir a Equação 4 pela Equação 25, obtendo-se:

$$\frac{V}{[E_0]} = \frac{k_2 * [ES]}{[E] + [ES] + [EI] + [EIS]} = \frac{k_2 \frac{[E] * [S]}{K_M}}{[E] + \frac{[E] * [S]}{K_M} + \frac{[E] * [I]}{K_I} + \frac{[E] * [S] * [I]}{K_M * K_I}}$$

$$V = k_2 \frac{[S]}{K_M + [S] + \frac{[I] * K_M}{K_I} + \frac{[S] * [I]}{K_I}}$$

$$V = \frac{V_{max}}{\left(1 + \frac{[I]}{K_i}\right)} \frac{[S]}{K_M + [S]} \quad (26)$$

Comparando o modelo cinético obtido (Equação 26) com o modelo de Michaelis-Menten original sem inibição (Equação 10), percebe-se que inibição não competitiva, por sua vez, não altera o valor de K_M , mas afeta a velocidade máxima (V_{max}), gerando a chamada velocidade máxima aparente, apresentada pela Equação 27 abaixo. A mesma pode ser substituída no modelo, gerando o modelo final de velocidade de reação com inibição não-competitiva, representada pela Equação 28.

$$V_{max_ap} = \frac{V_{max}}{\left(1 + \frac{[I]}{K_i}\right)} \quad (27)$$

$$V = V_{max_ap} \frac{[S]}{K_M + [S]} \quad (28)$$

2.4.1.1.3 INIBIÇÃO INCOMPETITIVA

A inibição incompetitiva é a inibição reversível que ocorre quando a molécula inibidora não possui afinidade diretamente à enzima que está disponível livremente, apenas ao complexo enzima-substrato (ES). Essa ligação do inibidor ao complexo se dá em uma região diferente do centro ativo do substrato, formando o complexo enzima-substrato-inibidor, inativo. Essa ligação acontece, tornando o complexo inativo e reduz, portanto, a velocidade da catálise enzimática (DORAN, 2013), como mostra

o modelo reacional abaixo e toda dedução dos balanços apresentada a seguir, de acordo com estudos de Lehninger, Nelson e Cox (2012).



A constante de formação do complexo enzima-substrato e a constante de formação do produto, com regeneração da enzima, foram demonstradas anteriormente pela Equação 23 e Equação 24. No caso da constante de inibição incompetitiva, esta é dependente da concentração de todas as moléculas envolvidas na reação, como mostra a Equação 29 abaixo.

$$K_I = \frac{[ES]*[I]}{[EIS]} = \frac{[E]*[S]*[I]}{K_M*[EIS]} \quad (29)$$

A concentração total de enzimas neste caso, é composta pela soma das concentrações das enzimas livres e das enzimas componentes dos complexos enzima-substrato e enzima-substrato-inibidor, como demonstrado pela Equação 30 a seguir.

$$[E_0] = [E] + [ES] + [EIS] \quad (30)$$

Considerando-se a velocidade limitante de catálise, pode-se dividir a Equação 4 da mesma pela Equação 30 descrita acima, fazendo-se as devidas simplificações, demonstradas abaixo.

$$\frac{V}{[E_0]} = \frac{k_2*[ES]}{[E]+[ES]+[EIS]} = \frac{k_2 \frac{[E]*[S]}{K_M}}{[E] + \frac{[E]*[S]}{K_M} + \frac{[E]*[S]*[I]}{K_M*K_I}} \quad (31)$$

$$V = k_2 \frac{[S]}{K_M + [S] + \frac{[S]*[I]}{K_I}} \quad (31.1)$$

$$V = \frac{V_{max}}{\left(1 + \frac{[I]}{K_i}\right) \left(\frac{K_M}{1 + \frac{[I]}{K_i}} + [S]\right)} \quad (31.2)$$

Comparando-se a Equação 31.2 acima com o modelo de Michaelis-Menten sem inibição, pode-se concluir que a inibição incompetitiva reduz os valores da

constante K_M e da velocidade máxima (V_{max}), gerando os valores aparentes indicados pelas Equações 32 e 33 abaixo. Com as devidas substituições no modelo original sem inibição, permitem a definição do modelo cinético de inibição incompetitiva, indicado pela Equação 34.

$$K_{M_ap} = \frac{V_{max}}{\left(1 + \frac{[I]}{K_i}\right)} \quad (32)$$

$$V_{max_ap} = \frac{V_{max}}{\left(1 + \frac{[I]}{K_i}\right)} \quad (33)$$

$$V = V_{max_ap} \frac{[S]}{K_{M_ap} + [S]} \quad (34)$$

2.4.1.1.4 LINEWEAVER-BURK

Afim de facilitar a compreensão gráfica do Modelo de Michaelis-Menten, sua equação pode ser submetida ao método de linearização algébrica, proposto por Lineweaver-Burk (1934). A Equação de Lineweaver-Burk é obtida, portanto, através da inversão da Equação de Michaelis-Menten, como demonstra a Equação 35 abaixo.

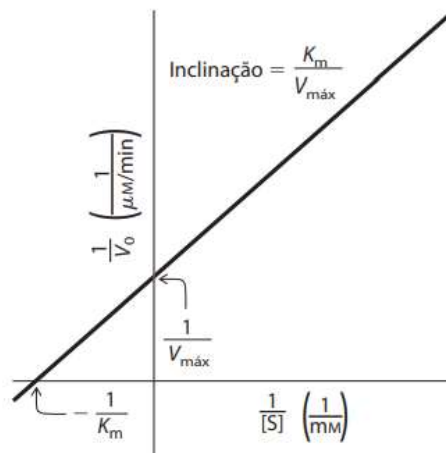
$$\frac{1}{V} = \frac{K_M + [S]}{V_{max} * [S]} \quad (35)$$

$$\frac{1}{V} = \frac{K_M}{V_{max} * [S]} + \frac{1}{V_{max}} \quad (35.1)$$

O modelo, conhecido como duplo recíproco, é representado por uma equação algébrica de primeiro grau, com coeficiente angular equivalente à relação entre K_M e V_{max} e com coeficiente linear do valor inverso de V_{max} (LEHNINGER; NELSON; COX, 2012).

Segundo Doran (2013), no gráfico, de eixos $1/[S]$ e $1/V$, pode-se determinar os valores de K_M e V_{max} , visto que a intersecção do eixo das abcissas é equivalente a $-1/K_M$, enquanto a intersecção do eixo das ordenadas é equivalente a $1/V_{max}$, como demonstra a Figura 4 abaixo.

Figura 4: Representação gráfica da Linearização de Lineweaver-Burk do Modelo cinético de Michaelis-Menten.



Fonte: LEHNINGER; NELSON; COX, 2012.

Apesar de muito utilizado na diferenciação qualitativa dos mecanismos de catálise enzimática e na análise dos modelos de inibição, o modelo linearizado de Lineweaver-Burk não pode ser considerado apropriado para a determinação de parâmetros cinéticos, visto distorcer o erro experimental, especialmente em baixas concentrações de substrato (PEREIRA *et al*, 2015). Neste caso, alguns outros modelos de linearização, como Eadie-Hofstee, por exemplo, poderia ser aplicado para maior assertividade.

2.4.2 ALGORITMO GENÉTICO

Os algoritmos genéticos (AGs) são algoritmos de otimização que se inspiram na seleção natural e na teoria evolucionista de Charles Darwin. Esses algoritmos são amplamente utilizados em problemas de otimização complexos e em modelagem de equações, nos quais são aplicados para estimar os parâmetros ideais de um modelo matemático, determinando soluções que melhor se ajustam aos dados experimentais (SINGH; DEB, 2021).

A aplicação dos AGs na modelagem de equações tem como objetivo encontrar uma função matemática que melhor represente os dados experimentais disponíveis. Para isso, é necessário encontrar os valores dos parâmetros do modelo que produzem o melhor ajuste aos dados experimentais. Os AGs são especialmente úteis nesse contexto, pois são capazes de lidar com problemas complexos de otimização e, ao mesmo tempo, evitar o risco de convergir para mínimos locais, o que pode ser um problema em outras técnicas de otimização (KHEREDDINE et al., 2018).

Um exemplo de aplicação dos AGs na modelagem de equações é a modelagem de sistemas dinâmicos, nos quais se faz necessário encontrar uma equação que descreva a evolução temporal de uma variável. Desse modo, podem ser aplicados para estimar parâmetros cinéticos das reações enzimáticas: constante de Michaelis-Menten e Velocidade Máxima (EIBEN; SMITH, 2003).

2.4.3 ALGORITMO DE LEVENBERG-MARQUARDT

O algoritmo de Levenberg-Marquardt é uma técnica de otimização amplamente utilizada em problemas de ajuste de curvas e modelagem de equações não lineares (MARQUARDT, 1963). Esse algoritmo combina a eficiência do método do gradiente descendente com a robustez do método de Gauss-Newton, permitindo encontrar soluções ótimas em regiões planas ou curvas suaves.

O método é especialmente útil em problemas de otimização não lineares que apresentam regiões planas ou curvas suaves, onde o método do gradiente pode ficar preso em mínimos locais (NOCEDAL; WRIGHT, 2006). Esse algoritmo é capaz de ajustar o parâmetro de penalização de acordo com a estrutura do problema, permitindo que o algoritmo encontre soluções ótimas, que minimizam o erro entre os dados experimentais e o modelo (DORN *et al.*, 2019).

Assim como o algoritmo genético, o algoritmo de Levenberg-Marquardt pode ser utilizado na modelagem de sistemas dinâmicos. Nesse contexto, é necessário ajustar os parâmetros do modelo para que ele descreva adequadamente a evolução temporal dos dados observados. Combinados, os algoritmos apresentam-se como

ferramentas ainda mais poderosas na determinação assertiva dos parâmetros cinéticos (DORN *et al.*, 2019).

2.4.3 BIORREADORES

As reações de catálise enzimática ocorrem nos chamados biorreatores, nos quais as enzimas utilizadas podem encontrar-se livres no meio reacional ou imobilizadas. Os biorreatores são sistemas desenvolvidos para realização de reações biológicas com um ambiente adequado para crescimento celular e síntese de bioprodutos (SCHMIDELL *et al.*, 2001).

Em relação à sua escala, os biorreatores são classificados em três grupos distintos. Os biorreatores industriais, utilizados na síntese de bioprodutos em larga escala, os biorreatores em escala piloto, que variam de dezenas a centenas de litros, importantes para transferência de processos em escala industrial, e os biorreatores de bancada, de menor litragem, utilizados em pesquisa de laboratório (SCHMIDELL *et al.*, 2001).

Entre a grande variedade de biorreatores disponíveis, constata-se que o mais difundido e empregado é o biorreator de tanque agitado (STR) em modo de operação em batelada, em especial em processos fermentativos e farmacêuticos laboratoriais e industriais (FOGLER, 1999), cuja configuração foi base para a elaboração do *software Enzimatics* desenvolvido.

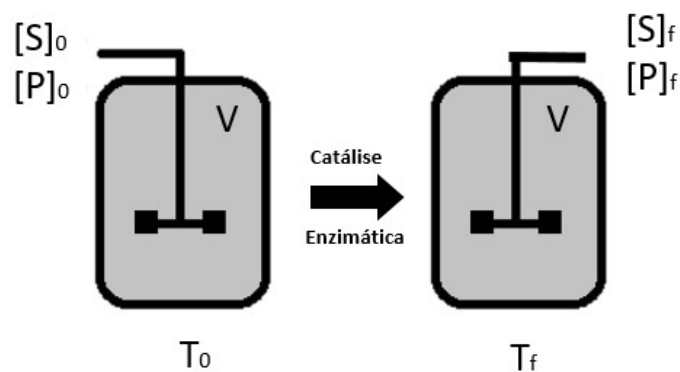
Caracteriza-se como um tanque cilíndrico, constituído geralmente de vidro, carbono ou aço inox, de proporções 2:1 e 3:1 da altura em relação ao diâmetro do tanque. O sistema é agitado pelo agitador mecânico no eixo central, composto por uma série de turbinas, sendo a mais utilizada as turbinas de pás planas, do inglês *flat blade*. Sua rotação é realizada por um motor conectada a esse eixo, acima ou abaixo do tanque (DORAN, 2013).

2.4.3.1 OPERAÇÃO EM BATELADA

No modo de operação em batelada, os reatores operam em sistema fechado, sem variação volumétrica ($V_0 = V_f$). O substrato e o produto são alimentados no início da operação (T_0) em suas concentrações iniciais ($[S]_0$ e $[P]_0$), juntamente com a enzima (E_0), que por ser apenas um biocatalisador de concentração constante, não é considerada no balanço de massa do sistema. Depois do tempo de reação, a enzima, o produto e o substrato (caso a conversão substrato-produto não seja integral) são retirados apenas ao fim da operação (T_f), não ocorrendo a entrada ou saída de matéria durante toda a reação de catálise (FOGLER, 1999), como demonstra a Figura 5 abaixo.

As principais vantagens dos reatores em regime de batelada são a possibilidade de higienização e assepsia após cada batelada produzida, permitindo, dessa forma, a utilização do mesmo equipamento para a produção de uma variedade de produtos (FOGLER, 1999).

Figura 5. Representação esquemática do modo de operação em batelada, empregada em biorreatores do tipo STR.



Fonte: Autora, 2023.

Tratando-se de um sistema fechado, então, pode-se realizar o balanço de massa do mesmo, como demonstram as deduções apresentadas a seguir, descritas por Borzani e colaboradores (2001).

$$\text{Acúmulo} = \text{Entrada} - \text{Saída} - \text{Reação} \quad (36)$$

$$\text{Acúmulo} = - \text{Reação} \quad (36.1)$$

Desse modo, pode-se deduzir que o acúmulo é a taxa de conversão de substrato em um volume V fixo, enquanto a taxa de reação depende do volume e da velocidade de reação (r).

$$\frac{d(V*[S])}{dt} = - (r * V) \quad (36.2)$$

$$\frac{d[S]}{dt} = - r \quad (36.3)$$

Em reações de catálise enzimática, a velocidade é determinada seguindo o modelo de Michaelis-Menten, como demonstrado anteriormente (Equação 10), obtendo-se a taxa final de conversão de substrato no sistema, descrita pela Equação 36.4 abaixo.

$$\frac{d[S]}{dt} = - \frac{V_{max}*[S]}{K_M+[S]} \quad (36.4)$$

Considerando-se tratar de um sistema de conversão, pode-se determinar a taxa de formação de produto, como demonstra a Equação 37 abaixo:

$$\frac{d[P]}{dt} = \frac{V_{max}*[S]}{K_M+[S]} \quad (37)$$

A conversão em produto, entretanto, é dependente da conversão máxima de reação, que pode ser determinada através da Equação 38 abaixo. No caso de uma conversão nula, não há alterações na concentração de substrato ou de produto iniciais, que se mantém constante. No caso de uma conversão de 100%, a concentração final de substrato é zerada, tendo sido integralmente convertido em produto.

$$\text{Conversão} = \frac{S_0 - S_f}{S_0} \quad (38)$$

2.5 COMPUTAÇÃO CIENTÍFICA

Nas últimas décadas, com os rápidos avanços tecnológicos, a computação tornou-se fundamental para o meio científico acadêmico, a medida em que passou a fornecer ferramentas para a solução dos mais variados problemas de forma rápida e eficiente (KULISCH; MIRANKER, 2014), unindo a ciência da computação, engenharia e matemática, como mostra a Figura 6 abaixo.

Entretanto, em um meio dominado por *softwares* comerciais e programadores certificados, o desenvolvimento de aplicações científicas e acadêmicas por parte dos mais diversos profissionais, como físicos, matemáticos e estatísticos, mostrou-se um difícil desafio. Isto se deve, especialmente, à prévia dominância na área, de linguagens de programação tradicionais, mais complexas (HEYS, 2017).

Desse modo, visando uma oportunidade do mercado crescente, foi desenvolvida uma nova classe de programas computacionais voltada ao público científico. Escritos em linguagem de alto nível e com fácil interação usuário-máquina, *softwares* como *MATLAB*, *Mathematica*, *Maple*, entre outros, popularizaram-se com rapidez no meio acadêmico (COELHO, 2007).

No entanto, esses ambientes apresentam várias desvantagens, como o alto custo de licença e a não portabilidade entre máquinas e sistemas operacionais. Essas características indesejáveis facilitaram a introdução de novos ambientes de computação científica à área, compostos por linguagens *open source* de alto nível, porém de fácil aprendizagem e execução, como a linguagem Python (COELHO, 2007).

Figura 6. Componentes essenciais da Computação Científica.



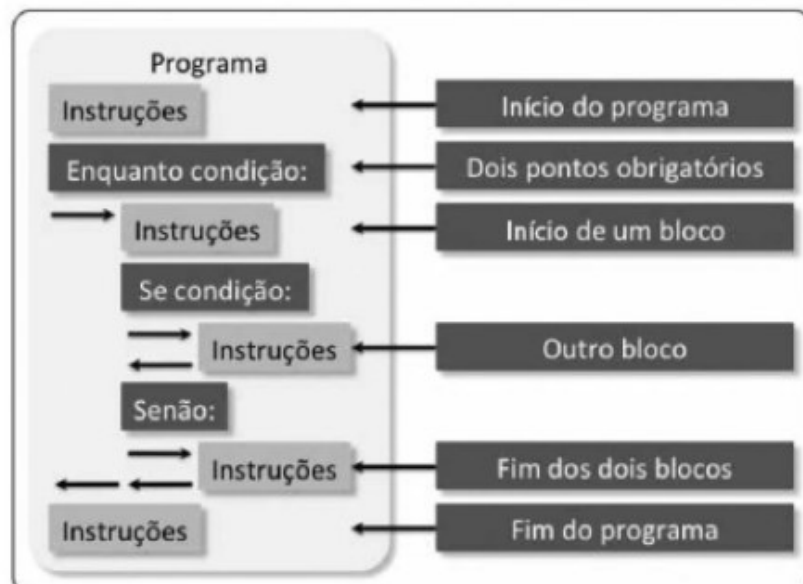
Fonte: Autora, 2023.

2.6 LINGUAGEM COMPUTACIONAL PYTHON

Python é uma linguagem computacional interpretada de código aberto e comunitário (*open source*), gerenciado pela *Python Software Foundation* (DE LIMA; COELHO; TAKIGAWA, 2021).

Desenvolvida para fins acadêmicos em 1991, por Guido van Rossum, a linguagem prioriza a legibilidade do código em detrimento a expressividade (BURIOL; ARGENTA, 2009). Desse modo, ao focar na produtividade e fácil entendimento, possui sintaxe clara e concisa, com poucas palavras-chaves e caracteres especiais, como demonstra a Figura 7 abaixo, que exemplifica a estrutura básica de um programa padrão em linguagem Python. Sua simplicidade foi característica fundamental para que se tornasse rapidamente a linguagem computacional de maior crescimento entre a comunidade de desenvolvedores, especialmente no campo da ciência de dados (MCKINNEY, 2019; BORGES, 2014).

Figura 7. Estrutura base de um programa escrito em linguagem Python e a ordem de execução dos comandos.



Fonte: BORGES, 2014.

Python é uma linguagem computacional orientada a objetos, modelo de programação que aproxima o raciocínio computacional à realidade, ao tratar de

objetos, suas características, funções e como se relacionam uns com os outros. Essa estrutura facilita ainda mais o entendimento de códigos em detrimento à programação estruturada tradicional (BURIOL; ARGENTA, 2009).

Por ser uma linguagem interpretada, assinala McKinney (2019) ainda, que a linguagem Python necessita de um interpretador que transforma seu código em *bytecodes*, permitindo sua execução em diferentes plataformas. O interpretador, então, roda o programa linha a linha, executando as instruções sequenciais separadamente, como mostra a Figura 8 abaixo.

Figura 8. Estrutura de interpretação de códigos desenvolvidos em linguagem Python.



Fonte: Adaptado de MCKINNEY, 2019.

A alta adesão ao Python deve-se também ao fato de a linguagem possuir variadas estruturas chamadas de alto nível, como listas, dicionários, tuplas e uma abrangente gama de módulos disponíveis, que lhe garantem alta flexibilidade e boa integração com outras linguagens de alto nível, como C e FORTRAN (DE SOUSA *et al*, 2020).

Outra vantagem da linguagem é sua tipagem dinâmica e forte, na qual os tipos de variáveis são identificados em tempo de execução, verificando se as operações designadas podem ser realizadas, evitando erros na execução do código fonte (DE LIMA; COELHO; TAKIGAWA, 2021).

Portanto, para o desenvolvimento de códigos em Python, são necessários ao usuário apenas três estruturas básicas: o Core – interpretador da linguagem, o Ambiente de Desenvolvimento Integrado – interface de desenvolvimento, e as Bibliotecas – coleção de módulos (MCKINNEY, 2012).

2.7 AMBIENTES DE DESENVOLVIMENTO INTEGRADOS (IDE)

O Ambiente de Desenvolvimento Integrado (IDE) é um *software* para desenvolvimento e edição de códigos, que reúne diversas ferramentas que permitem ao usuário executar *scripts*, acessar terminais, *debugar* e compilar códigos em um único ambiente de interface gráfica, substituindo o uso dos ultrapassados editores de texto (PIMENTEL *et al*, 2021).

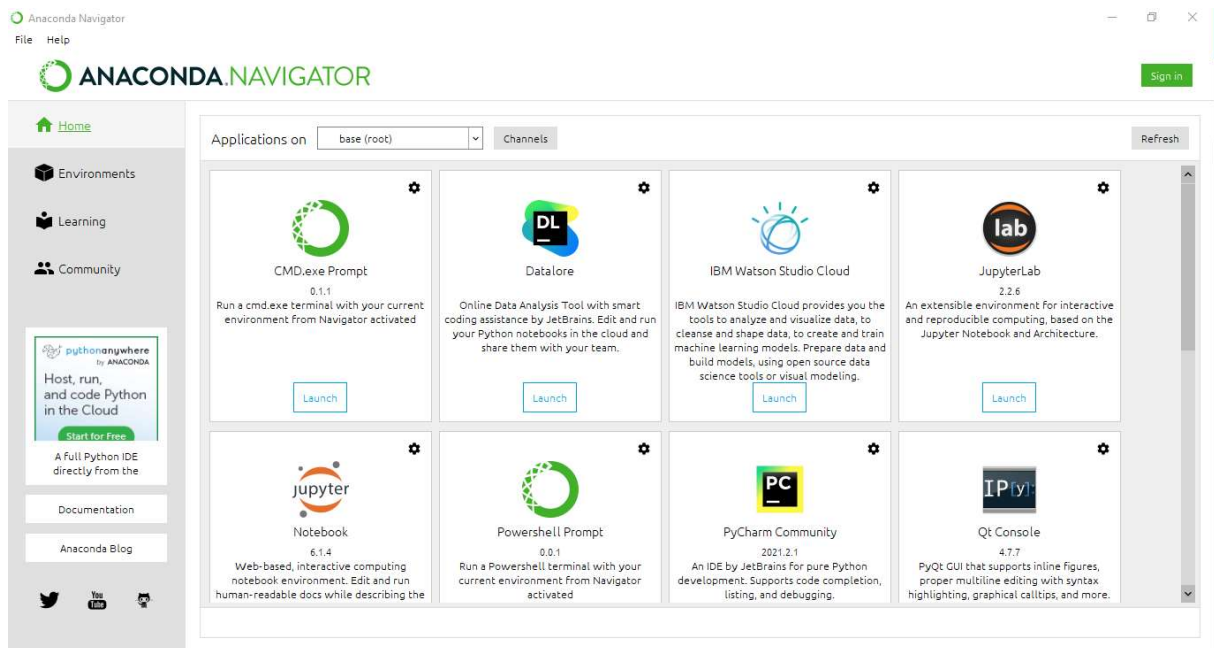
Compostos por um editor de código e debugger, os IDEs possuem também interpretadores, responsáveis na transformação do código em linguagem de máquina. No caso da linguagem Python, existem diversas IDEs gratuitas disponíveis para uso, entre as quais destacam-se o *Jupyter Notebook* e o *PyCharm*, ambos gerenciados pelo *software Anaconda* (ANACONDA, 2022).

2.7.1 ANACONDA

A Figura 9 abaixo exhibe o layout do gerenciador *Anaconda*, utilizado para gerenciar pacotes *open source*. Criado com o intuito de facilitar a preparação dos ambientes de programação em linguagem Python e R, é especialmente utilizado para fim científicos acadêmicos. Composto por mais de 8000 pacotes de ciência de dados e *machine learning*, possui compatibilidade com os maiores sistemas e arquiteturas computacionais atuais, vantagens que o tornam a plataforma de distribuição Python mais utilizada no mundo (ANACONDA, 2022).

Em sua instalação padrão, disponibiliza diversos Ambientes de Desenvolvimento Integrado (IDE) para desenvolvimento e testagem de código, como *RStudio*, *Spyder*, *Jupyter Notebook*, *PyCharm*, *Glueviz* entre outros.

Figura 9. Layout inicial do software comercial Anaconda Navigator.



Fonte: Autora, 2023.

2.7.2 JUPYTER NOTEBOOK

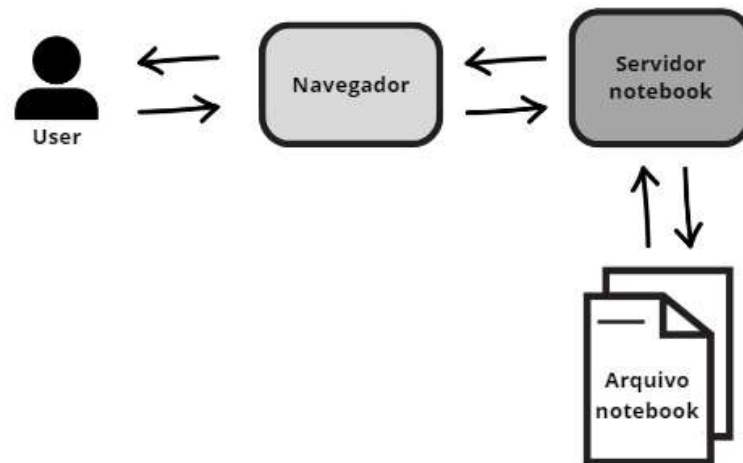
O *Jupyter Notebook* é uma interface gráfica *open source* que tem como base um navegador web (*browser*), utilizado para o desenvolvimento de códigos e análise de dados, com visualização rápida em um ambiente interativo e exploratório (RANDLES *et al*, 2017; PERKEL, 2018).

Criado com a finalidade de agilizar a escrita, testagem e depuração de códigos, o *Jupyter* surgiu como uma evolução do IPython, e rapidamente ganhou espaço no campo científico (MCKINNEY, 2019). Muito utilizado no desenvolvimento de algoritmos, permite a visualização de textos, imagens em diferentes formatos, exibindo ainda os erros de código (PIMENTEL *et al*, 2021).

Segundo McKinney (2019), a estrutura do *Jupyter* é composta por dois elementos principais: a aplicação web e os *Notebooks*. A aplicação web consiste no *browser* de internet que é utilizado como base para os *Notebooks*. Os *Notebooks*, por sua vez, configuram-se como documentos interativos que podem ser executados várias vezes separadamente (JUPYTER NOTEBOOK, 2022). Apresentando células

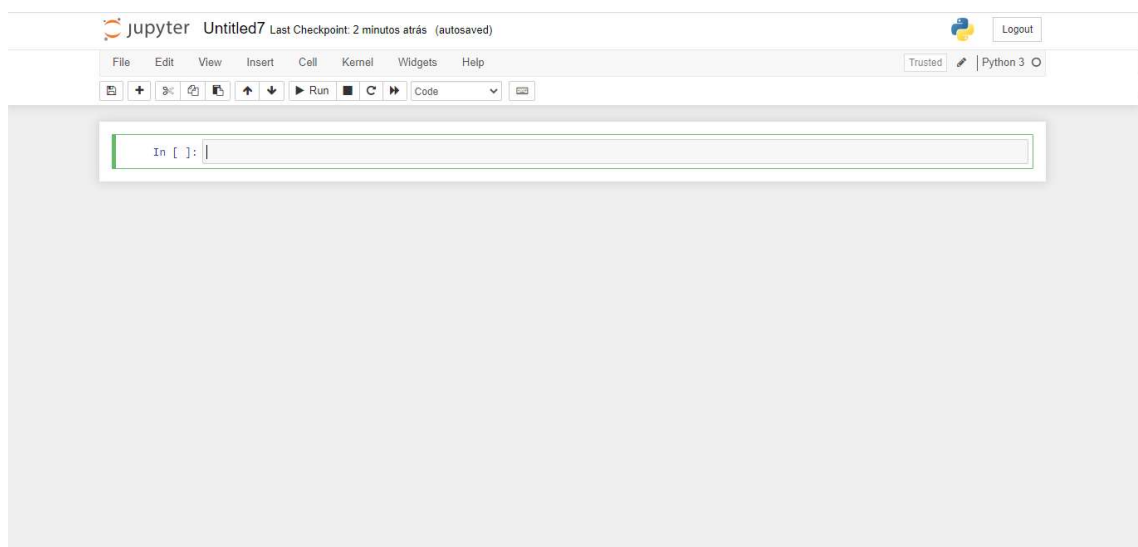
de código executável, células de *Markdown* e células brutas, os *Notebooks* são intuitivos e de fácil acesso, sendo muito utilizados nas primeiras etapas de desenvolvimento e teste de programas (PIMENTEL *et al*, 2019). Em relação ao armazenamento, os *Notebooks*, são enviados do browser para o servidor (*Notebook Server*), onde são armazenados como arquivos JSON e com extensão *.ipynb*, de onde podem ser acessados rapidamente pelo próprio *Jupyter*, como demonstra a Figura 10 abaixo. De *layout* fácil e interativo, destaca-se pela facilidade de uso por usuários leigos, como demonstra a Figura 11 em seguida.

Figura 10. Componentes base do IDE *Jupyter Notebook* e suas interações.



Fonte: Adaptado de JUPYTER NOTEBOOK, 2023.

Figura 11. *Layout* da interface gráfica *open source Jupyter Notebook*.



Fonte: Autora, 2023.

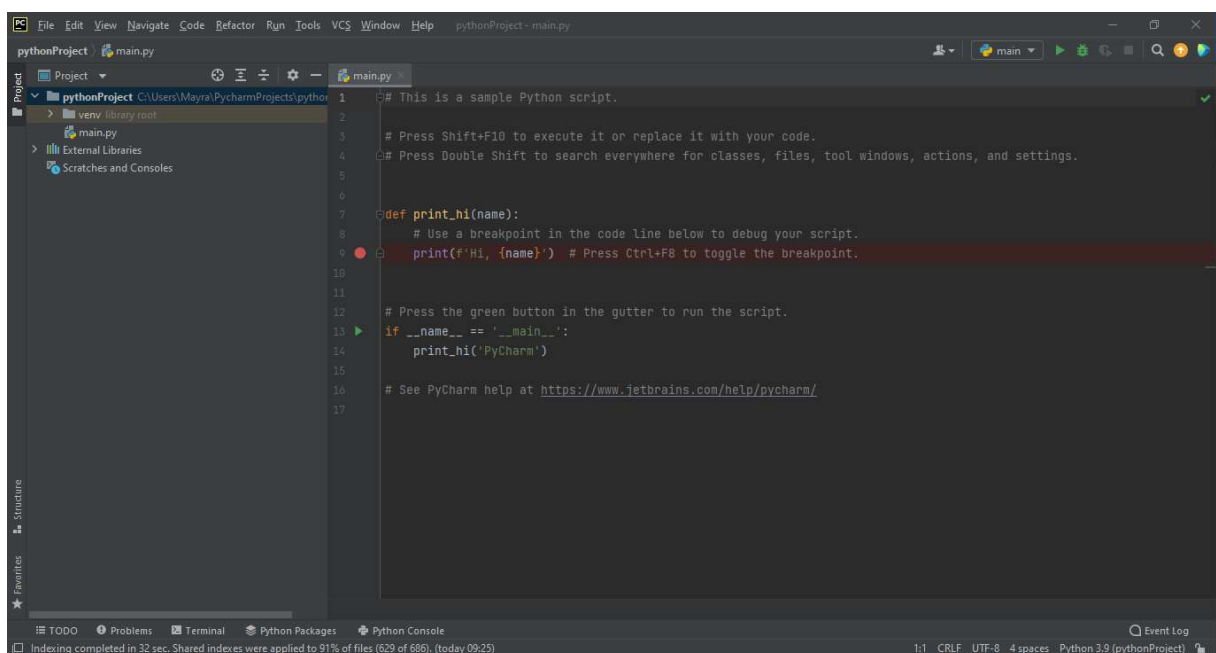
2.7.3 PYCHARM

O *PyCharm* é um Ambiente de Desenvolvimento Integrado (IDE) voltado a escrita, testagem e depuração de códigos em linguagem Python, distribuído pela empresa tecnológica *JetBrains* (Figura 12). Compatível com as plataformas *Windows*, *macOS* e *Linux*, o programa é disponibilizado em três edições diferentes, sendo duas delas *open source*, voltadas para a comunidade Python e para fins educacionais, e uma versão profissional paga, que apresenta alguns recursos de desenvolvimento web adicionais (JETBRAINS, 2021).

Com um layout minimalista intuitivo, o *PyCharm* apresenta uma série de ferramentas que facilitam a configuração e edição de códigos de modo a agilizar o desenvolvimento das mais diversas aplicações, especialmente nas áreas de desenvolvimento web, análise de dados e computação científica (ISLAM, 2015).

Entre suas melhores características, o IDE oferece um editor de código inteligente com suporte para Python, *JavaScript*, *HTML/CSS*, *AngularJS* e *TypeScript*, no qual há a complementação de código inteligente, com inspeção instantânea e indicação e/ou correção de erros, além de refatorações automatizadas e navegação avançada, agilizando ainda mais a escrita (ISLAM, 2015).

Figura 12. Layout do Ambiente de Desenvolvimento Integrado (IDE) *PyCharm*.



Fonte: Autora, 2023.

2.8 BIBLIOTECAS PYTHON

A linguagem Python oferece uma extensa biblioteca padrão integrada em sua instalação. Escrita em sua maioria em código de linguagem C, a biblioteca fornece uma diversidade de ferramentas e recursos básicos, com módulos matemáticos, como o *math*, *cmath*, *decimal* e *random*, módulos de tempo, como o *time* e *datetime*, módulos operacionais como o módulo *sys*, entre diversos outros (PYTHON SOFTWARE FOUNDATION, 2020), os quais não necessitam de instalação externa.

Segundo a documentação oficial fornecida pela *Python Software Foundation* (2020), ademais à biblioteca padrão, a linguagem Python oferece uma vasta gama de pacotes externos desenvolvidos e atualizados regularmente por diferentes colaboradores, visto tratar-se de um projeto popular de código aberto. A licença aberta da linguagem de alto nível não apenas cria uma comunidade colaborativa extensa, como disponibiliza ainda mais vantagens ao programador, uma vez que problemas comuns podem ser solucionados com a simples importação de pacotes com funções e módulos com códigos já elaborados que contribuem para a resolução.

2.8.1 IMPORTAÇÃO E ATUALIZAÇÃO DAS BIBLIOTECAS

A instalação das bibliotecas que não encontram-se integradas à biblioteca-padrão Python nem ao *Anaconda* pode ser realizada através da ferramenta de gerenciamento de pacotes *pip*, a qual apresenta-se incluída aos instaladores binários a partir da versão Python 3.4 (PYTHON SOFTWARE FOUNDATION, 2021).

Conectado à internet, o programa *pip* instala a versão atualizada do pacote desejado e suas dependências, depois de procurá-la no repositório de pacotes *PyPI* (*Python Package Index*), como afirmam McKinney (2019) e Borges (2014).

Desse modo, a instalação e a atualização dos pacotes desejados são realizadas através de comandos no *shell* (*prompt* de comando no caso do *Windows*), como descrito abaixo.

```
pip install nome_do_pacote
```

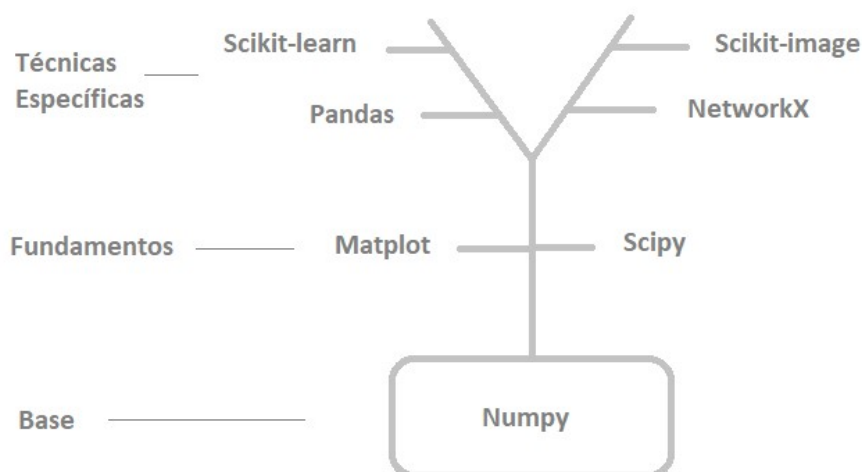
```
pip install --upgrade nome_do_pacote
```

2.9 BIBLIOTECAS PYTHON PARA COMPUTAÇÃO CIENTÍFICA

O ecossistema científico Python tem a capacidade de transformar seu ambiente interativo, com poucos comandos, em um ambiente de processamento de dados extremamente eficiente, rivalizando com os principais sistemas utilizados para a finalidade, como *MATLAB*, *Octave*, *R-Lab* e *SciLab* (SCIPY.ORG, 2019).

Os pacotes de computação científica, que possuem como base as três principais bibliotecas da área: *NumPy*, *SciPy* e *Matplotlib*, fornecem todas as ferramentas necessárias para a extração, tratamento, processamento e visualização rápida de grandes quantidades de dados (HARRIS *et al*, 2020), como mostra a Figura 13 abaixo.

Figura 13. Bibliotecas Python fundamentais para Computação Científica.



Fonte: Adaptado de HARRIS *et al*, 2020.

2.9.1 NUMPY

Na esfera da programação científica faz-se necessária a utilização de funções relacionadas não apenas a operações matemáticas como também a manipulação de vetores, matrizes, equações diferenciais e integrais. Por natureza, a linguagem Python não apresenta estas funcionalidades integradas, fato que evidencia a importância de bibliotecas específicas para a manipulação numérica, entre as quais destaca-se o *NumPy*, abreviação para *Numerical Python* (BRESSERT, 2012).

O *NumPy* é uma biblioteca Python *open source* (Figura 14) especializada em processamento numérico em linguagem de alto nível, base para o ecossistema científico Python (VAN DER WALT; COLBERT; VAROQUAUX, 2011). Desse modo, alicerça outras bibliotecas, como *SciPy*, *Matplotlib*, *Pandas*, *Scikit-learn* e *Sckit-image*, as quais possuem o objeto array *NumPy* como sua estrutura de dados primária e uma boa interoperabilidade com o pacote numérico (MCKINNEY, 2019; HARRIS, 2020).

Figura 14. Logo da biblioteca Python *open source* *NumPy*.



Fonte: NUMPY.ORG, 2021.

Desenvolvido colaborativamente no ano de 2005, o *NumPy* destacou-se desde sua criação por unir as principais características dos dois pacotes numéricos predecessores ao utilizar os recursos do *Numarray*, e o ótimo desempenho e a API escrita em linguagem C do *Numeric* (HARRIS, 2020). Desse modo, atualmente apresenta-se como essencial para a construção de modelos de *Machine Learning*, processamento de imagem e na computação gráfica, sendo utilizado principalmente na academia, na indústria e, recentemente com a ascensão da ciência de dados e *business intelligence*, no mundo corporativo, nas mais diversas aplicações (VAN DER WALT; COLBERT; VAROQUAUX, 2011).

O pacote de extensão do Python provê um objeto *array* multidimensional, chamado de *ndarray*, abreviação para *N-dimensional array*, ou tensor, e as funções para sua manipulação e processamento. Além disso provê ferramentas para

operações de álgebra linear com o submódulo *linalg*, transformadas de Fourier, manipulação polinomial, geração de números aleatórios e funções matemáticas básicas, como logarítmica, exponencial e até mesmo as funções trigonométricas seno, cosseno e tangente (MCKINNEY, 2019; OLIPHANT, 2006).

Em relação à espaço e memória o *NumPy*, importada usualmente com a abreviatura *np* (*import numpy as np*) apresenta outro benefício, visto que o *ndarray* trata-se de um objeto contíguo da memória da máquina, comumente denominado localidade de referência, que pode ser acessado de maneira rápida e eficiente (OLIPHANT, 2006).

2.9.2 SCIPY

O *SciPy* é uma biblioteca científica Python *open source* (Figura 15) desenvolvida para fins matemáticos, científicos e de engenharia, que, juntamente com a biblioteca *NumPy* constitui a base do processamento de dados numéricos na linguagem (MCKINNEY, 2019). Composto por uma variada gama de algoritmos matemáticos e rotinas numéricas, o pacote provê ferramentas fundamentais para a modelagem e simulação de sistemas complexos com sintaxe simples e de fácil compreensão (VIRTANEN *et al*, 2020).

Figura 15. Logo da biblioteca científica Python *open source* *SciPy*.



Fonte: SCIPY.ORG, 2021.

Constituído por pacotes secundários, os quais devem ser importados separadamente ao ambiente de programação, o *SciPy* é amplamente utilizado para problemas de álgebra linear, integração, interpolação, otimização, estatística, resolução de matrizes e sistemas lineares esparsos, e processamento de imagens e sinais (SCIPY.ORG, 2019; MCKINNEY, 2019).

Lançado inicialmente no ano de 2001 por um grupo de estudantes universitários, é atualizado e otimizado com regularidade por sua crescente e bem estabelecida comunidade *online* através de códigos e regras disponibilizados no *GitHub* (VIRTANEN *et al*, 2020; MILLMAN; AIVAZIS, 2011).

Inicialmente construído sobre a estrutura de *array* da biblioteca *Numeric*, teve a transição de sua base para o *NumPy*, com o surgimento do pacote no ano de 2005. Desse modo, possui grande interoperabilidade com o pacote numérico, que lhe fornece uma matriz multidimensional como estrutura de dados primária. O *SciPy*, por sua vez, tem seu código como base de outras bibliotecas científicas avançadas, como *Scikit-learn* e *Scikit-image* (VIRTANEN *et al*, 2020).

2.9.3 PANDAS

A análise de dados configura-se como um dos pilares da computação científica, visto que a manipulação, processamento, limpeza e extração de dados é fundamental para a modelagem de sistemas (NELLI, 2015).

Nesse contexto, a linguagem Python apresenta em sua instalação padrão a biblioteca *open source* de análise de dados *Pandas*, abreviação para *Python Data Analysis* (Figura 16). Desenvolvido no ano de 2010, o pacote apresenta ferramentas de alto nível para a manipulação de dados estruturados, os quais incluem dados tabulares, *arrays* multidimensionais e séries temporais, de modo intuitivo, rápido e flexível (MCKINNEY, 2019).

Figura 16. Logo da biblioteca científica Python *open source Pandas*.



Fonte: PANDAS.PYDATA.ORG, 2021.

Unindo o eficiente processamento de *arrays NumPy* com funções flexíveis de carregamento, manipulação e combinação de dados disponíveis em tabelas e em bancos de dados do tipo relacional, a biblioteca possibilita a automação e reprodutibilidade de análises em diferentes conjuntos de dados (CHEN, 2017; NELLI, 2015).

Dispondo de uma sofisticada indexação, o *Pandas* permite ainda a importação de base de dados em diferentes formatos, como arquivos em *csv*, *JSON*, *xlsx* e *xls*, ampliando sua aplicabilidade. Concorrente poderoso do *Microsoft Excel*, o pacote não apenas permite a importação dos documentos do pacote office como o supera ao operar em diferentes sistemas operacionais (MCKINNEY, 2012).

Usualmente importada no ambiente Python sob a abreviação *pd* (*import pandas as pd*), a biblioteca é constituída primariamente por duas estruturas, como mostra a Figura 17 abaixo: a *Serie*, estrutura unidimensional rotulada, composta por elementos homogêneos, e o *DataFrame*, estrutura multidimensional tabular orientada a colunas, composto pelas *Series* (CHEN, 2017).

Figura 17. Estruturas básicas da biblioteca *Pandas*.

Series		DataFrame																																
<table><thead><tr><th>Serie 1</th><th>Serie 2</th></tr></thead><tbody><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>10</td></tr><tr><td>2</td><td>20</td></tr><tr><td>3</td><td>30</td></tr><tr><td>4</td><td>40</td></tr></tbody></table>	Serie 1	Serie 2	0	0	1	10	2	20	3	30	4	40		<table><thead><tr><th></th><th>Serie 1</th><th>Serie 2</th></tr></thead><tbody><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>10</td></tr><tr><td>2</td><td>2</td><td>20</td></tr><tr><td>3</td><td>3</td><td>30</td></tr><tr><td>4</td><td>4</td><td>40</td></tr></tbody></table>		Serie 1	Serie 2	0	0	0	1	1	10	2	2	20	3	3	30	4	4	40		
Serie 1	Serie 2																																	
0	0																																	
1	10																																	
2	20																																	
3	30																																	
4	40																																	
	Serie 1	Serie 2																																
0	0	0																																
1	1	10																																
2	2	20																																
3	3	30																																
4	4	40																																

Fonte: Autora, 2023.

2.9.4 MATPLOTLIB

A visualização de dados apresenta-se como uma importante ferramenta não apenas para engenharia, como em diversas áreas do conhecimento. Parte integrante do processamento ou apenas para apresentação visual de dados, os gráficos permitem uma compreensão rápida e fácil de sistemas e dados, fundamental para a experimentação, cálculos e tomadas de decisão (CHEN, 2018).

Na linguagem computacional científica Python, existem diferentes módulos *open source* utilizados para visualização de dados, dentre os quais destaca-se o *Matplotlib* (Figura 18), devido sua versatilidade para a elaboração de gráficos em diferentes dimensões (2D e 3D) com qualidade adequada à publicação científica (ARI; USTAZHANOV, 2014).

Figura 18. Logo da biblioteca científica Python *open source* *Matplotlib*.



Fonte: MATPLOTLIB.ORG, 2022.

Entre suas principais vantagens, o *Matplotlib* apresenta recursos para a plotagem de diversos tipos de gráficos com poucos comandos simples, permitindo sua configuração e padronização com liberdade e facilidade (SILVA, 2020). Outros benefícios do pacote, que intensificam sua aplicabilidade, são a possibilidade da produção das imagens gráficas em diferentes extensões, como PNG, PDF, SVG, EPS, entre outros, e seu ambiente interativo entre plataformas (DE CAMPOS; DE OLIVEIRA, 2016; HUNTER *et al*, 2021).

Atendendo aos requisitos operacionais e de interface, o *Matplotlib* oferece ao usuário ainda o módulo *Pyplot*, com um *framework* para visualização de dados com funções semelhantes ao *MATLAB* (MASIERO; FERREIRA; AQUINO, 2012). Usualmente importado sob a abreviação *plt* (*import matplotlib as plt*), cada uma de suas funções é caracterizada por modificar um objeto no gráfico, porém com os

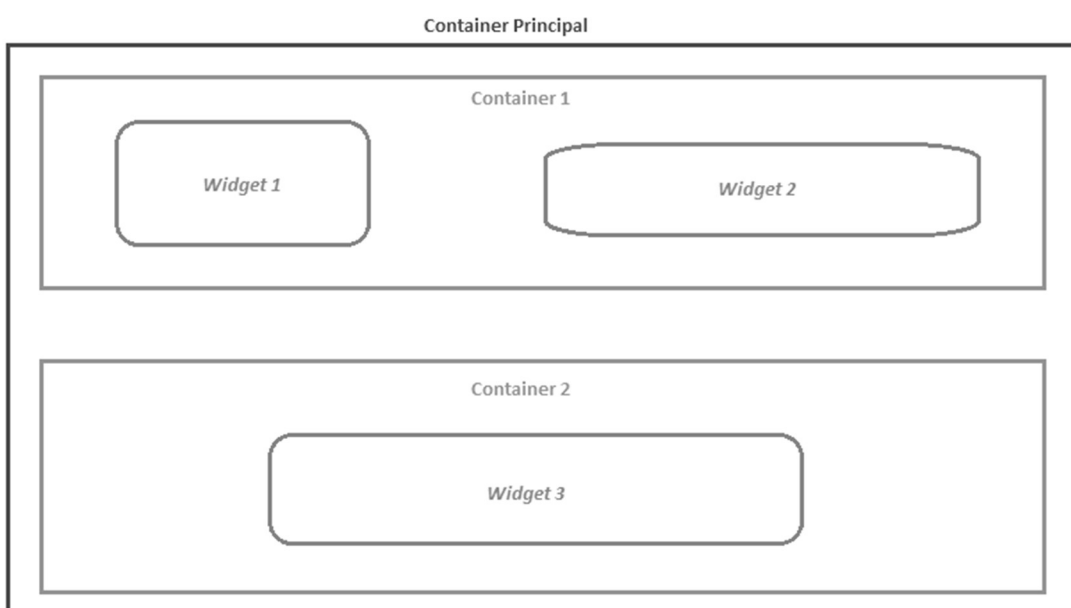
estados preservados com a chamada de novos comandos, mantendo as informações anteriores, característica que lhe garante eficiência em plotagens de comparação de dados e computação científica (HUNTER *et al*, 2021).

2.9.5 TKINTER

As Interfaces Gráficas de Usuário (*GUI – Graphical User Interface*), interfaces que permitem a interação do indivíduo com *softwares* através de imagens e botões, substituindo comando de texto, se popularizaram desde sua criação no ano de 1984 (KOLHE; KHETRI; DESHMUKH, 2012). Composta por dois elementos primários, entrada e saída, revolucionaram a computação, a medida em que facilitam e simplificam o uso de programas computacionais por parte de usuários leigos (GALITZ, 2007).

A estrutura das interfaces gráficas é composta fundamentalmente por *containers* (ou janelas), estruturas onde residem os *widgets*, objetos interativos básicos do framework como botões, ícones e imagens (PYTHON SOFTWARE FOUNDATION, 2020), como demonstra a Figura 19 abaixo.

Figura 19. Representação da estrutura base de Interface Gráfica de Usuário (*GUI*).



Fonte: Autora, 2023.

Nesse sentido, a linguagem Python apresenta vários módulos para a construção de interfaces gráficas nos programas desenvolvidos como *WxWidgets*, *Tkinter*, *Kivy*, *PySide*, *QT* e *PyGTK*. Entretanto, a maior parte da comunidade Python utiliza o *Tkinter*, visto a integração do mesmo à biblioteca padrão da linguagem científica (PYTHON SOFTWARE FOUNDATION, 2020).

O *Tkinter* é, portanto, uma biblioteca gratuita da linguagem Python que fornece as ferramentas necessárias para a criação de interfaces gráficas de modo simples, fácil e leve (GRAYSON, 2000).

Entre suas principais vantagens, destacam-se a sua implementação rápida e fácil e sua alta portabilidade, visto sua executabilidade entre os mais diversos sistemas operacionais, como *Linux*, *Windows*, *Unix* e *Mac*. Desse modo, o mesmo código tem funcionalidade em diferentes plataformas, com renderização com elementos nativos das mesmas (GRAYSON, 2000).

3 OBJETIVOS

3.1 OBJETIVOS GERAIS

O presente trabalho tem como objetivo desenvolver algoritmos de computação científica para simulação e modelagem de cinéticas enzimáticas em operação em batelada, na linguagem Python, à nível *Front-End* e *Back-End*.

3.2 OBJETIVOS ESPECÍFICOS

- Desenvolvimento de Algoritmos de Simulação de reações enzimáticas em batelada, com cálculo de concentrações de substrato e produto, conversão e velocidade de reação automáticos;
- Desenvolvimento de Algoritmos de Modelagem, com *output* de parâmetros cinéticos de reação estimados através da inserção de dados experimentais;

- Desenvolvimento de Interface Gráfica de Usuário (GUI) *user-friendly* para uso do *software*;
- Disponibilização de Documentação dos Modelos Cinéticos Enzimáticos utilizados para Simulação e Modelagem;
- Disponibilização do *software*, bem como do código-fonte na internet, com fins acadêmicos e de ensino.

4 MATERIAIS E MÉTODOS

A metodologia do trabalho consistiu no desenvolvimento de algoritmos de simulação e modelagem de reações enzimáticas em modo de operação em batelada, utilizando-se a linguagem Python, em sua versão 3.8.5.

O desenvolvimento inicial foi realizado no ambiente de programação científica *Jupyter Notebook*, interface gráfica de usuário, que permite a gestão de pacotes e ambientes computacionais sem o uso do terminal. Com arquivos com extensão *.ipynb*, é disponibilizado pelo *software open source Anaconda Navigator 4.9.2*. O ambiente de execução e testes do algoritmo utilizado, por sua vez, foi o ambiente de desenvolvimento integrado *PyCharm 2021.2*.

Para o tratamento de dados e desenvolvimento dos algoritmos foram importadas as bibliotecas *Pandas*, *Numpy* e *Scipy*. Para o desenvolvimento da interface gráfica do *software*, nomeado de *Enzinetics*, por sua vez, foi utilizado o pacote *Python Tkinter*, que fornece algoritmos de programação *Front-End*.

Na estrutura do *software*, a simulação é realizada através do *input* de parâmetros cinéticos, assim como escolha do tipo de inibição por parte do usuário. Através do balanço de massa e da Equação do Modelo cinético de Michaelis-Menten, da linearização de Lineweaver-Burk, e dos modelos de inibição, são calculados os dados de concentração de substrato e produto ao longo de um determinado tempo, assim como a taxa de conversã. Esses dados são obtidos a partir da resolução das Equações Diferenciais, etapa que é realizada através da função *odeint* do pacote

SciPy, com a entrada das concentrações iniciais, do tempo de reação e dos parâmetros cinéticos da reação.

Os dados numéricos experimentais inseridos na modelagem são extraídos de planilhas no formato de *Microsoft Excel*, de extensão *.xlsx*, mesmo formato em que são exportados os dados e os resultados obtidos pela simulação. O algoritmo de modelagem utiliza do Algoritmo Genético e de Levenberg-Marquardt para determinar a melhor curva ajustada aos dados experimentais de entrada (*input*). Desse modo, a programação *Back-end* retorna os melhores valores dos parâmetros cinéticos para o melhor ajuste.

No âmbito de testes finais, foram inseridos parâmetros cinéticos aleatórios não reais para simulação. Os dados gerados pelo algoritmo foram extraídos em formato de planilha *excel*. Estes dados numéricos, por sua vez, foram submetidos à modelagem, para comparação dos parâmetros cinéticos determinados pelo algoritmo.

Por fim, os algoritmos desenvolvidos no presente trabalho foram armazenados e disponibilizados no repositório universal *open source online* GitHub, para livre acesso.

5 RESULTADOS E DISCUSSÃO

5.1 SOFTWARE ENZINETICS

O *software Enzinetics* foi desenvolvido totalmente em linguagem Python, à nível *Back-End* e *Front-End*. Sua principal funcionalidade é a simulação de reações enzimáticas, seguindo a cinética de Michaelis-Menten, com ou sem inibição, de acordo com os modelos já estabelecidos na literatura, assim como a linearização do modelo, de acordo com Lineweaver-Burk. Ademais, é possível realizar a modelagem das reações a partir do *input* de dados experimentais de concentração de substrato e produto pelo tempo, com o retorno dos parâmetros cinéticos estimados.

Com a importação do pacote de desenvolvimento de interfaces gráficas *Tkinter* e suas ferramentas, o *software* teve sua interface estruturada de modo *user-friendly*,

facilitando sua utilização por todos os tipos de usuários, sem necessidade de conhecimentos prévios de programação.

Disponibilizado de maneira *online* como executável, pode ser utilizado em todos os tipos de computadores, não necessitando a instalação de nenhum programa e/ou sistema.

Ao iniciar o *software*, uma tela inicial é exibida, com função de menu principal do aplicativo, como mostra a Figura 20 abaixo. O nome do *software* é exibido, assim como sua versão atual e o nome da autora.

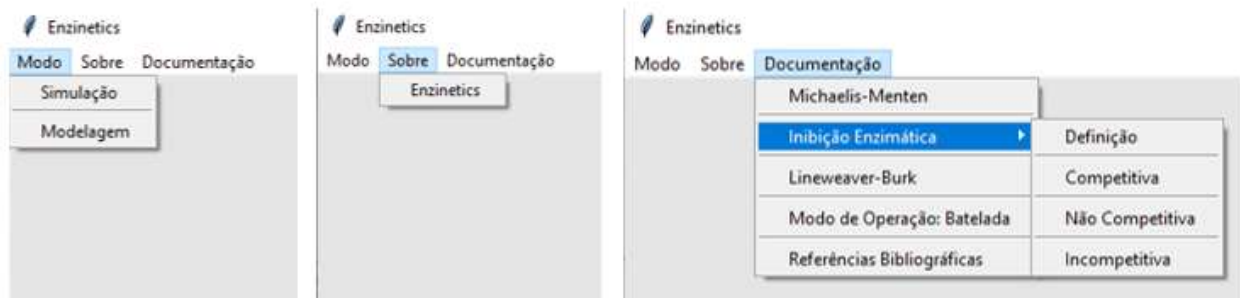
Figura 20. Tela inicial do *software Enzinetics* desenvolvido em Python.



Fonte: Autora, 2023.

Na Figura 21 a seguir, pode ser observado que uma barra de menu é exibida na parte superior da tela, com subdivisões: Modo, Sobre e Documentação (Figura 21). Cada uma das opções gerais, quando selecionadas, exibem outras opções específicas em cascata. Cada um dos botões específicos dá acesso a um novo *container* principal (tela), que é exibido por cima do menu, seguindo o estilo *TopLevel* de programação.

Figura 21. Opções da barra de menu da tela inicial do *software Enzinetics*.



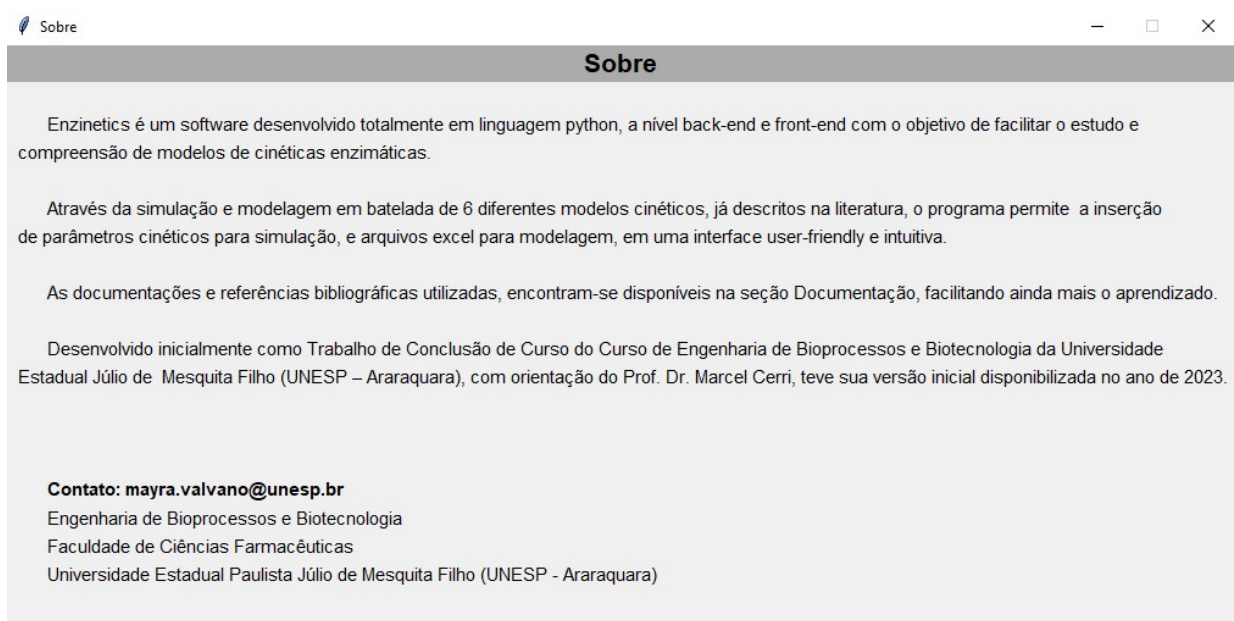
Fonte: Autora, 2023.

O código utilizado para todo desenvolvimento do *software* está disponível a todos no repositório *online* *GitHub*, pelo link: <https://github.com/MayraValvano/Enzinetics/blob/main/Enzinetics.py>, onde pode ser acessado e replicado livremente.

5.2 SOBRE

Com a seleção da opção *Sobre* e *Enzinetics*, uma nova tela é exibida com informações gerais do *software* e contato da autora, como mostra a Figura 22 abaixo.

Figura 22. Interface Gráfica *Sobre*, que exibe informações gerais do *software Enzinetics*, iniciada a partir de sua seleção no menu principal.



Fonte: Autora, 2023.

5.3 DOCUMENTAÇÃO

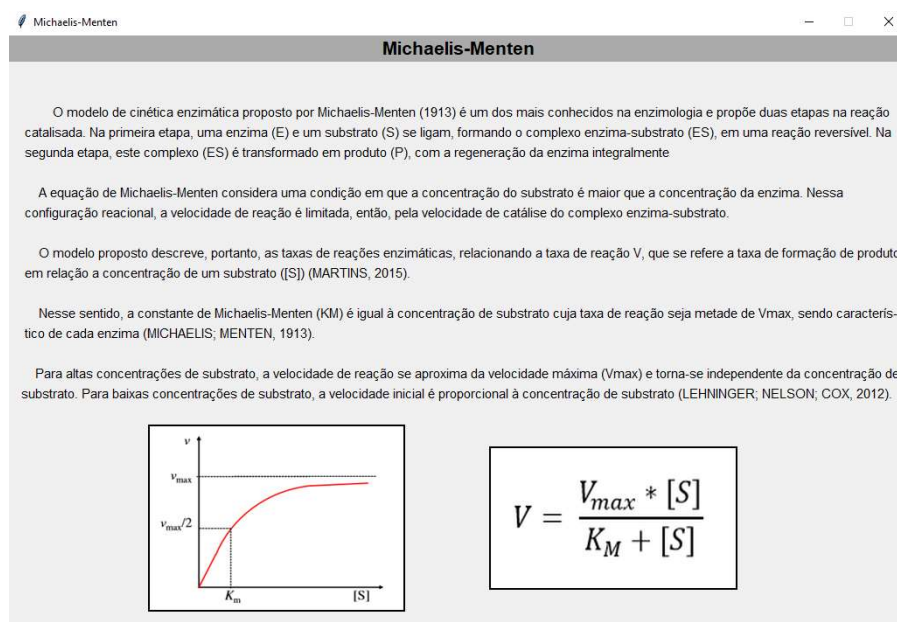
Com a seleção da opção Documentação, são exibidas algumas categorias: Michaelis-Menten, Inibição Enzimática, com subdivisão em Definição, Inibição Competitiva, Não Competitiva e Incompetitiva, Lineweaver-Burk, Modo de Operação: Batelada e Referências Bibliográficas. Cada uma delas, inicia um novo *container*, com informações gerais de cada um dos temas que as nomeiam.

Além de textos de fundamentação teórica, são apresentados também os principais balanços de massa, equações dos modelos, gráficos e outros dados relevantes. Os mesmos são fundamentais para melhor compreensão e estudo do comportamento das reações de catálise enzimática, podendo ser facilmente acessados pelo usuário.

5.3.1 DOCUMENTAÇÃO MICHAELIS-MENTEN

A partir da seleção da Documentação de Michaelis-Menten, o *container* com explicações do modelo é exibido, no qual constam um breve texto com os principais dados da cinética enzimática, uma representação gráfica geral do modelo e a Equação de velocidade estabelecida na literatura, como mostra a Figura 23 abaixo.

Figura 23. Interface gráfica da Documentação do modelo cinético enzimático de Michaelis-Menten.

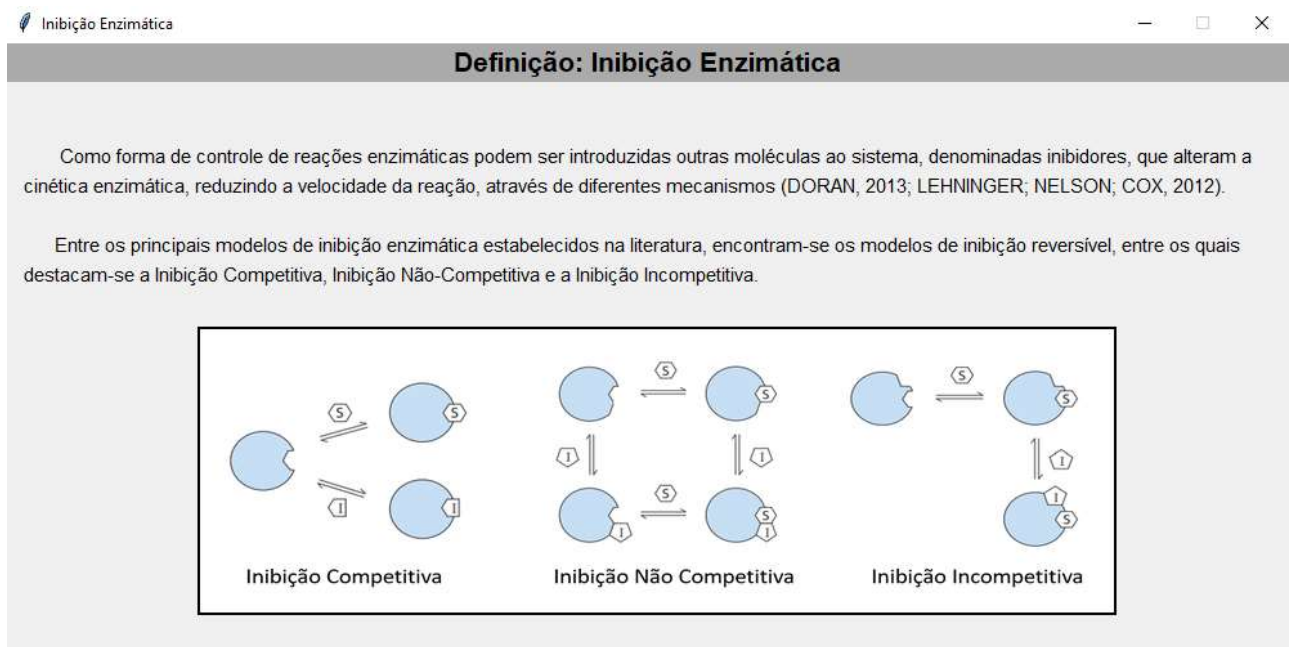


Fonte: Autora, 2023.

5.3.2 DOCUMENTAÇÃO INIBIÇÃO ENZIMÁTICA

Na Documentação de Inibição Enzimática, apresenta-se uma primeira interface gráfica com definição do conceito de inibição da atividade enzimática, introduzindo os modelos de inibição que são disponibilizados pelo *software* para simulação de diferentes cenários, como mostra a Figura 24 a seguir.

Figura 24. Interface gráfica de Documentação da Definição de Inibição Enzimática e introdução aos Modelos de Inibição Enzimática: Competitiva, Não Competitiva e Incompetitiva.



Fonte: Autora, 2023.

Cada um dos modelos de Inibição, por sua vez, possui seu próprio *container* (tela) de Documentação, na qual são exibidas breves fundamentações teóricas explicando o mecanismo de inibição, o balanço de reação, valores de K_M ou V_{max} aparentes, e o modelo de velocidade com os parâmetros de inibição, como demonstram as Figuras 25, 26 e 27.

Figura 25. Interface gráfica da Documentação do modelo de Inibição Enzimática Competitiva.

Inibição Competitiva

Os inibidores competitivos são aqueles que possuem semelhanças estruturais com o substrato da reação, e, dessa forma, ambos competem pelo sítio ativo da enzima. Assim, são formados, também, complexos enzima-inibidor (EI). Essa competição com os inibidores, de concentração I e constante K_i , dificulta a ligação do substrato a enzima e, conseqüentemente, a formação de produto (BORZANI et al, 2001).

A velocidade de formação de ambos os complexos, é dependente da concentração de substrato e de inibidores. A Inibição Competitiva afeta o valor da constante de Michaelis-Menten, que deve ser substituído por um K_m aparente (LEHNINGER; NELSON; COX, 2012).

$$E + S \xrightleftharpoons{K_M} ES \xrightarrow{K_2} E + P$$

$$E + I \xrightleftharpoons{K_i} EI$$

$$K_{M_ap} = K_M \left(1 + \frac{I}{K_i} \right)$$

$$V = V_{max} \frac{S}{K_{M_ap} + S}$$

Fonte: Autora, 2023.

Figura 26. Interface gráfica da Documentação do modelo de Inibição Enzimática Não Competitiva.

Inibição Não Competitiva

A inibição não competitiva mista ocorre quando as moléculas inibidoras possuem afinidade com a enzima, em um sítio ativo diferente do substrato, e com o complexo enzima-substrato (BORZANI et al, 2001). Ocorre, então, a formação de um outro tipo de complexo, o enzima-substrato-inibidor (EIS), que impede a continuação da reação, diminuindo a velocidade da catálise. Em suma, pode-se afirmar que concomitantemente, na reação, ocorrem a formação do complexo enzima-substrato, com formação de produto, do complexo enzima-inibidor e do complexo enzima-substrato-inibidor, que, por sua vez, pode ser formado pela ligação do inibidor no complexo enzima-substrato ou do substrato ao complexo enzima-inibidor (LEHNINGER; NELSON; COX, 2012)

A Inibição Não Competitiva, por sua vez, não altera o valor de K_m , mas afeta a velocidade máxima (V_{max}), gerando uma velocidade máxima aparente.

$$E + S \xrightleftharpoons{K_M} ES \xrightarrow{K_2} E + P$$

$$E + I \xrightleftharpoons{K_i} EI$$

$$EI + S \xrightleftharpoons{K_M} EIS$$

$$ES + I \xrightleftharpoons{K_i} EIS$$

$$V_{max_ap} = \frac{V_{max}}{\left(1 + \frac{[I]}{K_i} \right)}$$

$$V = V_{max_ap} \frac{[S]}{K_M + [S]}$$

Fonte: Autora, 2023.

Figura 27. Interface gráfica da Documentação do modelo de Inibição Enzimática Incompetitiva.

Inibição Incompetitiva

Inibição Incompetitiva

A inibição incompetitiva é a inibição reversível que ocorre quando a molécula inibidora não possui afinidade diretamente à enzima livre, apenas ao complexo enzima-substrato (ES). Essa ligação do inibidor ao complexo se dá em um sítio diferente do centro ativo ao qual se liga o substrato formando o complexo enzima-substrato-inibidor, que é inativo (DORAN, 2013).

A inibição incompetitiva reduz os valores da constante Km e da Velocidade Máxima (Vmax), gerando valores aparentes dos dois parâmetros cinéticos.

$$E + S \xrightleftharpoons{K_M} ES \xrightarrow{K_2} E + P$$

$$ES + I \xrightleftharpoons{K_I} EIS$$

$$K_{M_ap} = \frac{V_{max}}{\left(1 + \frac{[I]}{K_I}\right)}$$

$$V_{max_ap} = \frac{V_{max}}{\left(1 + \frac{[I]}{K_I}\right)}$$

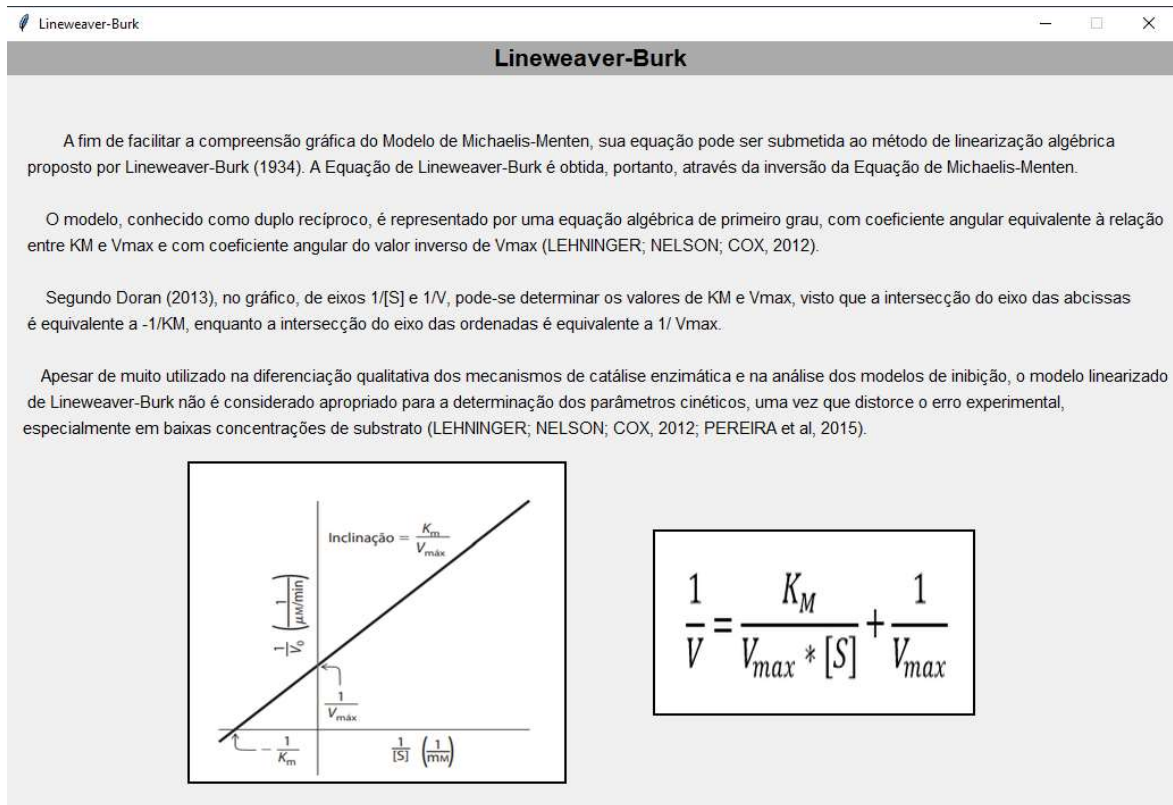
$$V = V_{max_ap} \frac{[S]}{K_{M_ap} + [S]}$$

Fonte: Autora, 2023.

5.3.3 DOCUMENTAÇÃO LINEWEAVER-BURK

Na Documentação de Lineweaver-Burk, são apresentados o conceito utilizado para linearização do modelo de Michaelis-Menten pelos autores, assim como a interpretação gráfica do modelo, plotado através dos eixos $1/[S]$ e $1/V$. A figura representativa gráfica da linearização e a equação algébrica linear também se encontram disponíveis para estudo, como demonstra a Figura 28 abaixo.

Figura 28. Interface gráfica da Documentação do modelo linearizado de Lineweaver-Burk.



Fonte: Autora, 2023.

5.3.4 DOCUMENTAÇÃO MODO DE OPERAÇÃO EM BATELADA

Na seção de Documentação do modo de operação em batelada, são descritos as definições de biorreatores e o comportamento das reações de catálise enzimática em batelada em reatores STR, ou seja, em sistemas fechados, sem saída ou entrada de reagentes ou produtos. É apresentada também uma figura esquemática do funcionamento da reação enzimática em biorreator de tanque agitado em tempos T_0 e T_f .

Por fim, são destacadas as taxas de consumo de substrato pelo tempo e de formação de produto pelo tempo, que são utilizadas pelo software Enzinetics para simulação e modelagem dos modelos, assim como o cálculo da taxa de conversão, como demonstra a Figura 29 a seguir.

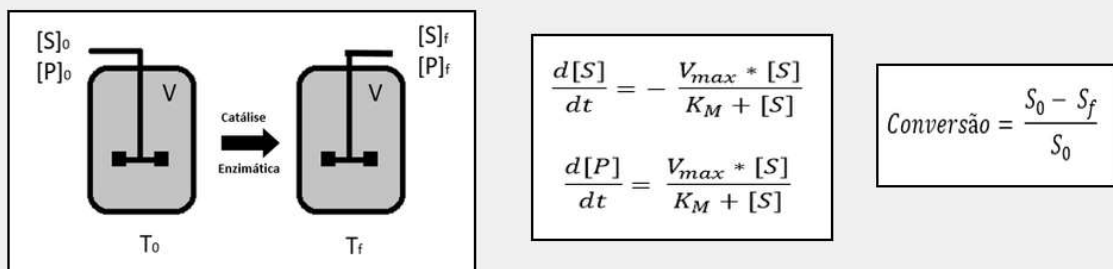
Figura 29. Interface gráfica da Documentação do modo de operação em batelada.

Operação em Batelada

As reações de catálise enzimática ocorrem nos chamados biorreatores, nos quais as enzimas utilizadas podem encontrar-se livres no meio reacional ou imobilizadas. Os biorreatores são sistemas onde ocorrem reações biológicas, ambiente adequado para síntese de bioprodutos e crescimento celular (SCHMIDELL et al., 2001).

Entre a grande variedade de biorreatores, constata-se que o mais difundido e empregado é o biorreator de mistura (STR) em modo de operação e batelada, em especial em processos fermentativos e farmacêuticos laboratoriais e industriais (FOGLER, 1999).

Neste modo de operação, os reatores operam em sistema fechado, sem variação volumétrica ($V_0 = V_f$). O substrato e o produto são alimentados no início da operação (T_0) em suas concentrações iniciais ($[S]_0$ e $[P]_0$), juntamente com a enzima, que por ser apenas um biocatalisador de concentração constante, não é considerada no balanço de massa do sistema. Depois do tempo de reação, a enzima, o produto e o substrato (caso a conversão substrato-produto não seja integral) são retirados apenas ao fim da operação (T_f), não ocorrendo a entrada ou saída de matéria durante toda a reação de catálise (FOGLER, 1999).

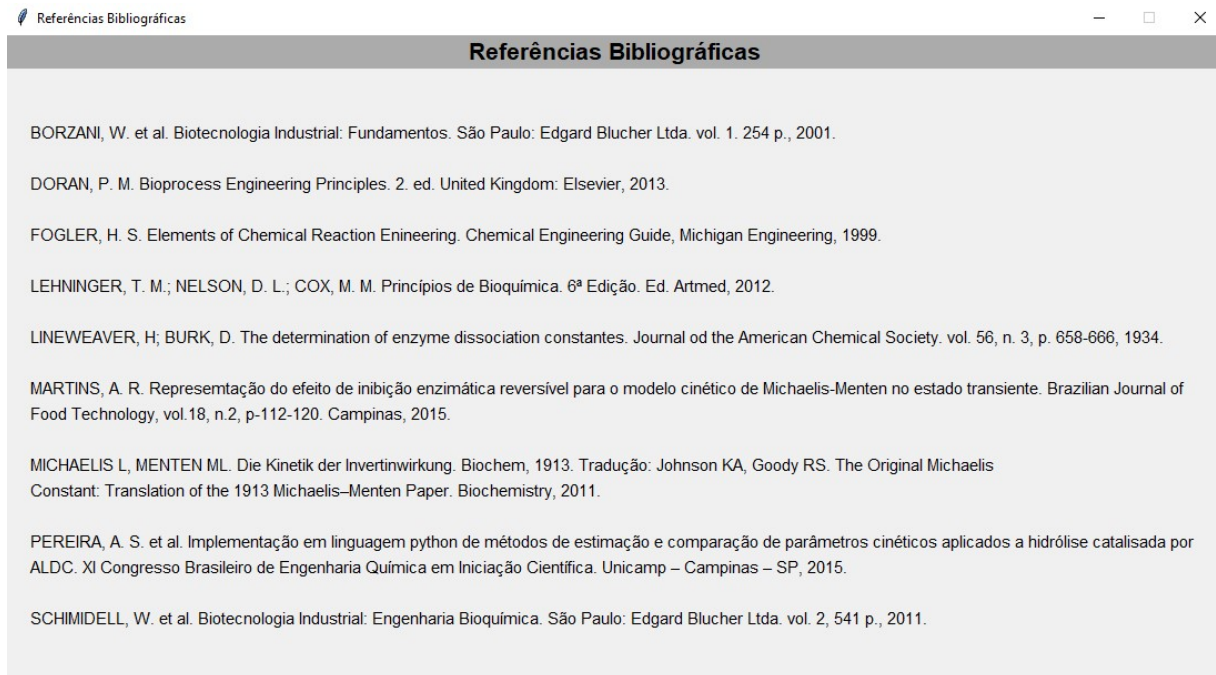


Fonte: Autora, 2023.

5.3.5 DOCUMENTAÇÃO REFERÊNCIAS BIBLIOGRÁFICAS

Na última das opções de Documentação, são apresentadas todas as referências bibliográficas de artigos e livros utilizados para a elaboração das documentações anteriores, como demonstra a Figura 30 abaixo. Desse modo, o usuário pode acessar com facilidade o material-fonte dos dados apresentados caso possua dúvidas ou queira aprofundar-se nos temas descritos.

Figura 30. Interface gráfica de referências bibliográficas utilizadas para a elaboração da fundamentação teórica de toda a seção de Documentação.



Fonte: Autora, 2023.

5.4 MODO: SIMULAÇÃO

Com a seleção do Modo Simulação do menu inicial, uma nova janela é exibida, na qual são realizadas as simulações dos biorreatores enzimáticos operados em modo batelada. Os principais componentes da sessão, como podem ser observados na Figura 31, são:

1. Imagem representativa de um biorreator em modo batelada e balanço de massa da reação enzimática em batelada, com aplicação do modelo cinético de Michaelis-Menten;
2. Entrada dos parâmetros cinéticos da reação por parte do usuário (*inputs*): constante de Michaelis-Menten (K_m), Concentração Inicial da Enzima (E_0) e constante de catálise – *turnover* (K_{cat}).
3. Seleção do tipo de inibição da reação enzimática: Sem Inibição, Inibição Competitiva, Inibição Não-Competitiva e Inibição Incompetitiva.

4. Estequiometria da reação, no qual a quantidade de mols de substrato é fixada em uma unidade para atender os critérios da cinética de Michaelis-Menten, porém com possibilidade da formação de mais de um mol de produto (*input* por parte do usuário).

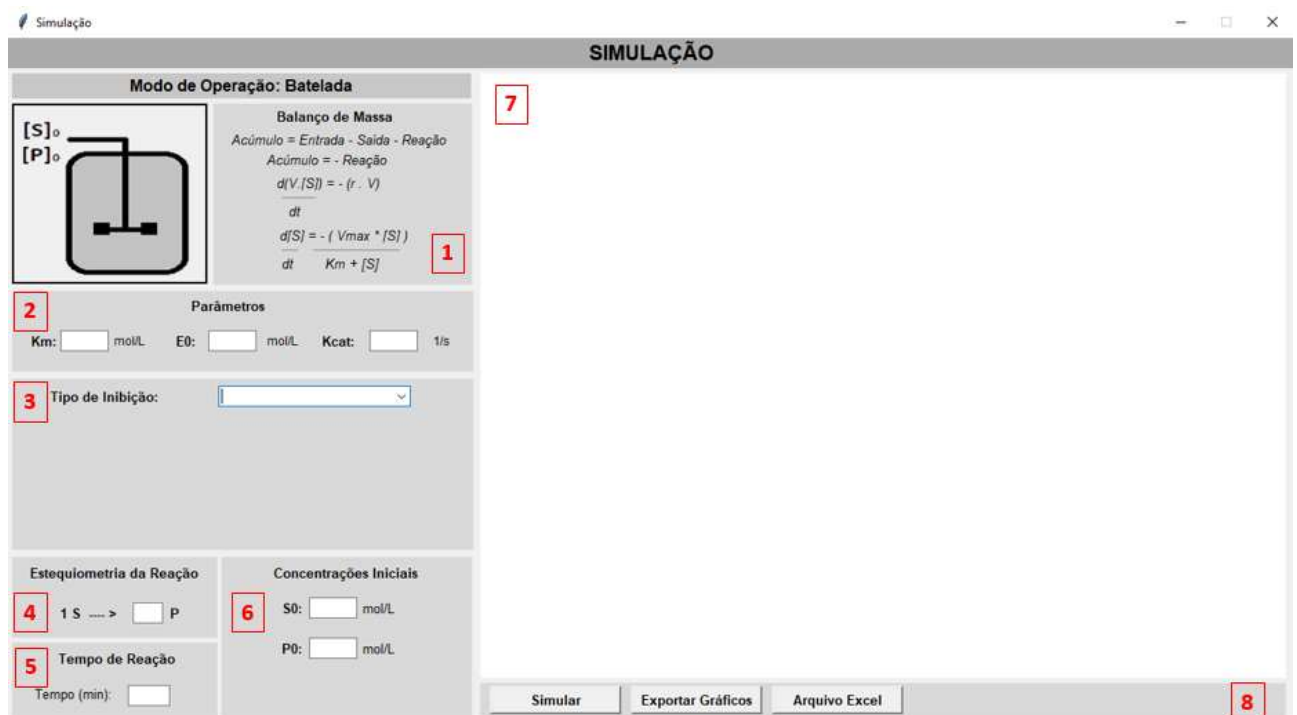
5. Tempo de reação em minutos que serão exibidos pela simulação.

6. Concentrações Iniciais de Substrato (S0) e de Produto (P0).

7. Área de exibição gráfica, onde, quando realizada a simulação da reação enzimática, são exibidos quatro gráficos: Velocidade x Substrato, 1/Velocidade x 1/Substrato, Concentração x Tempo e Conversão x Tempo.

8. Barra com botões de funcionalidade de simulação: Simular, Exportar Gráficos e Arquivo Excel.

Figura 31. Interface gráfica da sessão do Modo: Simulação.



Fonte: Autora, 2023.

5.4.1 SIMULAÇÃO SEM INIBIÇÃO

Com a seleção da opção Sem Inibição na caixa de rolagem, não há mudanças no modelo de Michaelis-Menten, visto que não há agentes externos influenciando o comportamento reacional. A simulação é realizada através de equações deduzidas do balanço de massa de substrato, de acordo com o modelo de Michaelis-Menten. Desse modo, o programa, em sua estrutura *Back-End*, acessa a função da equação diferencial obtida através do balanço de massa, realizando a integração da mesma pelo método *odeint* da biblioteca *Scipy*, com as condições iniciais da reação (S_0 e P_0), utilizando o método numérico de Runge-Kutta.

5.4.2 SIMULAÇÃO COM INIBIÇÃO COMPETITIVA

Com a seleção da opção Inibição Competitiva, um novo *widget* é exibido com a equação da constante de Michaelis-Menten ajustada para o modelo desejado ($Km_{ap} - Km$ aparente), como demonstra a Figura 32 abaixo. Considerando-se a presença de inibidores e sua influência na reação, novos campos para *inputs* de dados são apresentados: concentração de inibidor competitivo (I) e constante de inibição (Ki), que são considerados para a simulação.

Figura 32. Interface gráfica da sessão do Modo: Simulação, com a seleção do modelo de Simulação com Inibição Competitiva.

The screenshot shows the 'SIMULAÇÃO' window with the following sections:

- Modo de Operação: Batelada**: Includes a schematic of a stirred tank reactor with substrate $[S]_0$ and product $[P]_0$ inputs.
- Balanco de Massa**:

$$\begin{aligned} \text{Acúmulo} &= \text{Entrada} - \text{Saída} - \text{Reação} \\ \text{Acúmulo} &= - \text{Reação} \\ \frac{d(V \cdot [S])}{dt} &= - (r \cdot V) \\ \frac{d[S]}{dt} &= - \left(\frac{Vmax \cdot [S]}{Km + [S]} \right) \end{aligned}$$
- Parâmetros**:
 - Km : mol/L
 - E_0 : mol/L
 - $Kcat$: 1/s
- Tipo de Inibição**: Dropdown menu set to 'Competitiva'.
- Inibidor**:
 - I : mol/L
 - Ki : mol/L
 - Equation: $Km_{ap} = Km \cdot \left(1 + \frac{I}{Ki} \right)$
- Estequiometria da Reação**: $1 S \rightarrow \dots \rightarrow$ P
- Concentrações Iniciais**:
 - S_0 : mol/L
 - P_0 : mol/L
- Tempo de Reação**: Tempo (min):
- Buttons**: Simular, Exportar Gráficos, Arquivo Excel

Fonte: Autora, 2023.

5.4.3 SIMULAÇÃO COM INIBIÇÃO NÃO-COMPETITIVA

Quando selecionada a opção Inibição Não-Competitiva, por sua vez, o sistema possui comportamento semelhante: um novo *widget* é apresentado ao usuário com a equação da velocidade máxima ajustada para o modelo reacional com inibição não-competitiva (V_{max_ap} : *Vmax aparente*), assim como campos para preenchimento dos novos dados necessários. Pode observar-se na a Figura 33 os novos campos de preenchimento: concentração de inibidor competitiva (I) e constante de inibição (K_i), que são considerados para a simulação.

Figura 33. Interface gráfica da sessão do Modo: Simulação, com a seleção do modelo de Simulação com Inibição Não-Competitiva.

The screenshot shows the 'SIMULAÇÃO' window with the following components:

- Modo de Operação: Batelada**: A schematic of a stirred tank reactor with input streams for substrate $[S]_0$ and product $[P]_0$.
- Balanco de Massa**:

$$\text{Acúmulo} = \text{Entrada} - \text{Saída} - \text{Reação}$$

$$\text{Acúmulo} = - \text{Reação}$$

$$\frac{d(V \cdot [S])}{dt} = - (r \cdot V)$$

$$\frac{d[S]}{dt} = - \left(\frac{V_{max} \cdot [S]}{K_m + [S]} \right)$$
- Parâmetros**:
 - Km: mol/L
 - E0: mol/L
 - Kcat: 1/s
- Tipo de Inibição**:
- Inibição Parameters**:
 - I: mol/L
 - Ki: mol/L
$$V_{max_ap} = \frac{V_{max}}{1 + (I / K_i)}$$
- Estequiometria da Reação**: 1 S \rightarrow P
- Concentrações Iniciais**:
 - S0: mol/L
 - P0: mol/L
- Tempo de Reação**: Tempo (min):
- Buttons**: Simular, Exportar Gráficos, Arquivo Excel

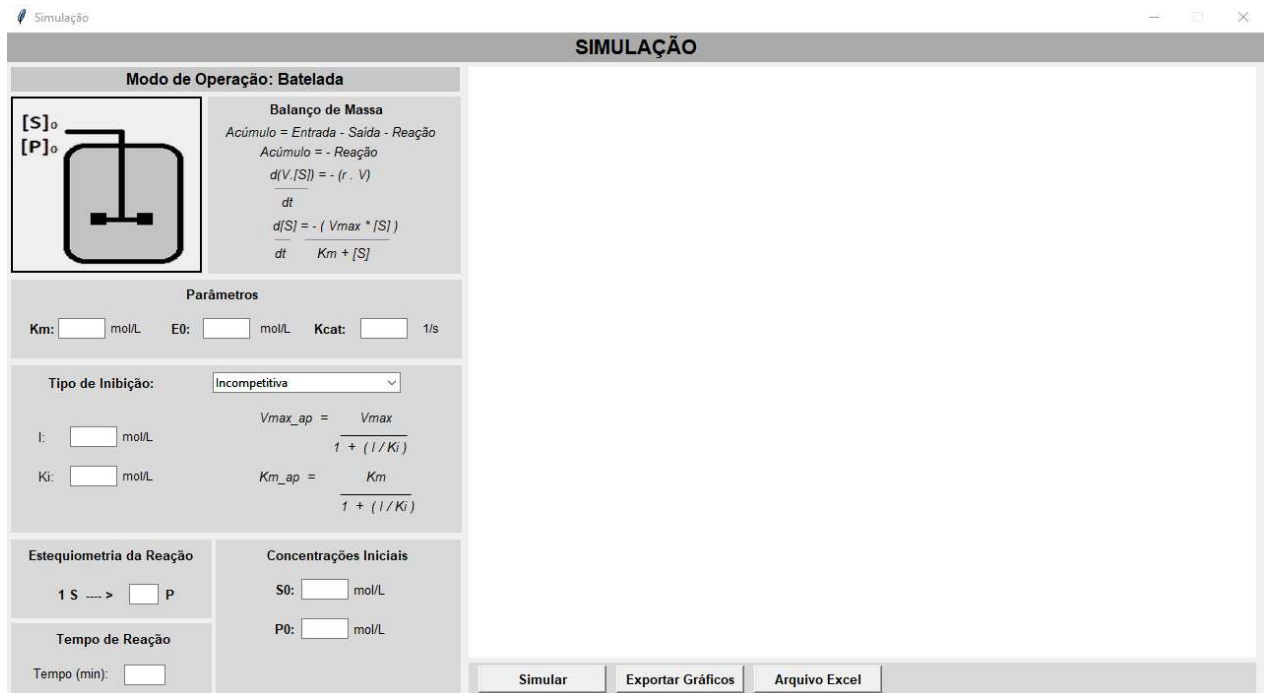
Fonte: Autora, 2023.

5.4.4 SIMULAÇÃO COM INIBIÇÃO INCOMPETITIVA

Por fim, com a seleção do modelo com Inibição Incompetitiva, os campos de entrada de dados de concentração de moléculas inibidoras (I) e de constante de inibição (K_i) também são exibidos. Entretanto, como neste modelo de reação não apenas a velocidade máxima ($V_{m\acute{a}x}$) é afetada, como também a constante de

Michaelis-Menten (K_m), as equações de ajuste de ambos os parâmetros cinéticos são exibidas (V_{max_ap} e K_{m_ap}), como mostra a Figura 34 a seguir.

Figura 34. Interface gráfica da sessão do Modo: Simulação, com a seleção do modelo de Simulação com Inibição Incompetitiva.



Fonte: Autora, 2023.

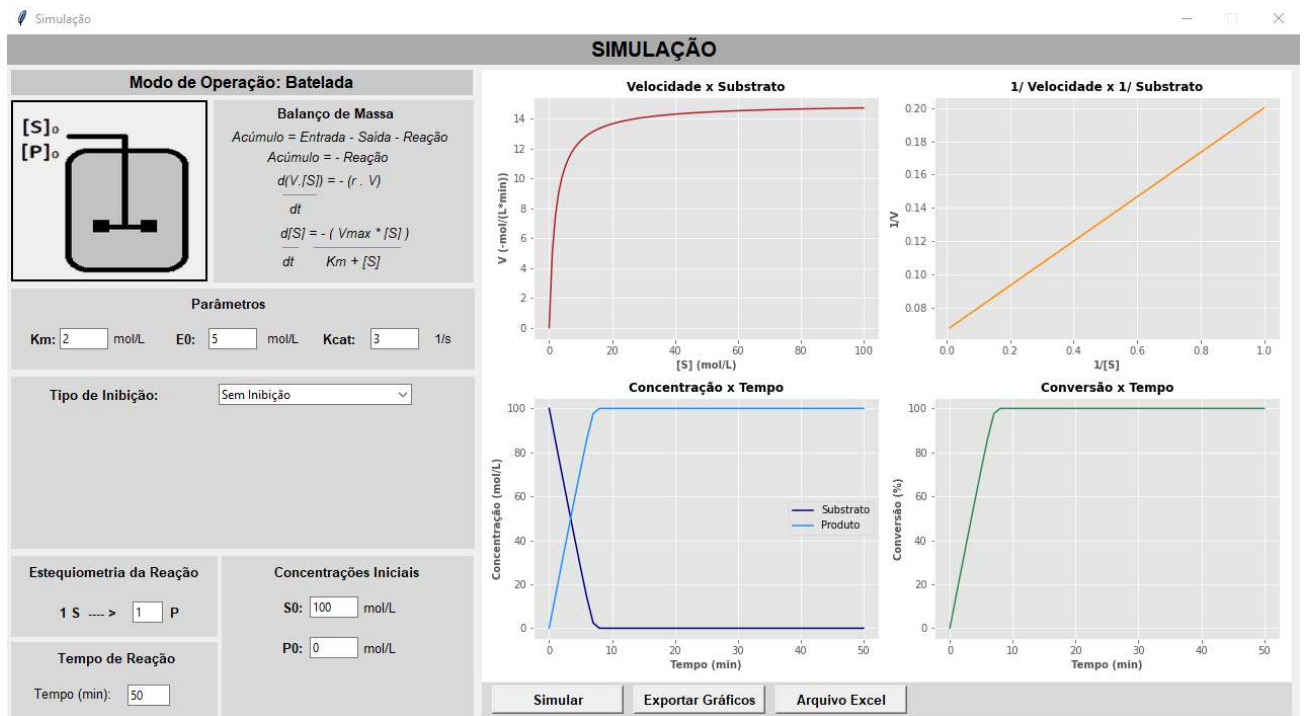
5.4.5 EXIBIÇÃO GRÁFICA

Quando inseridos todos os parâmetros cinéticos necessários para a simulação e acionado o botão Simular, o *software* realiza todos os cálculos em *background* e os dados são exibidos de maneira gráfica, em quatro plotagens diferentes.

Os gráficos apresentam-se como parte fundamental do *software*, visto que são uma importante ferramenta de aprendizado, à medida em que permitem a observação da influência dos diferentes parâmetros de maneira visual, mais compreensível. Desse modo, o usuário pode variar cada um dos valores dos parâmetros, observando as mudanças no comportamento reacional, exibidas claramente pelos gráficos.

Para fim de exemplificação do funcionamento da exibição gráfica, foram inseridos valores aleatórios não-reais para cada um dos parâmetros, como observado na Figura 35 abaixo.

Figura 35. Interface gráfica da sessão do Modo: Simulação, com a exemplificação da exibição gráfica, a partir da inserção de valores aleatórios para os parâmetros cinéticos.



Fonte: Autora, 2023.

Com a simulação realizada e os gráficos exibidos, é possível exportar os gráficos como imagem de extensão *.png*, acessando o botão Exportar Gráficos. Ao mesmo tempo, quando é gerada a simulação, os pontos experimentais simulados são gravados em um arquivo *excel* que pode ser exportado através do botão Arquivo Excel. Neste arquivo *excel*, em uma aba são gravados os dados temporais dinâmicos de concentração de substrato e de produto, e em uma segunda aba são escritos os dados utilizados para a elaboração gráfica do modelo de Michelis-Menten: concentração de substrato e velocidade calculada de acordo com o modelo para os diversos pontos considerados.

5.5 MODO: MODELAGEM

A interface gráfica do modo de Modelagem, na qual são inseridos os dados experimentais por parte do usuário e estimados valores ótimos para os parâmetros cinéticos enzimáticos do modelo de Michaelis-Menten, por sua vez, pode ser dividida entre quatro partes distintas, como pode observar-se na Figura 36 a seguir.

Figura 36. Interface gráfica da sessão modelagem, na qual são inseridos dados experimentais e estimados os parâmetros cinéticos



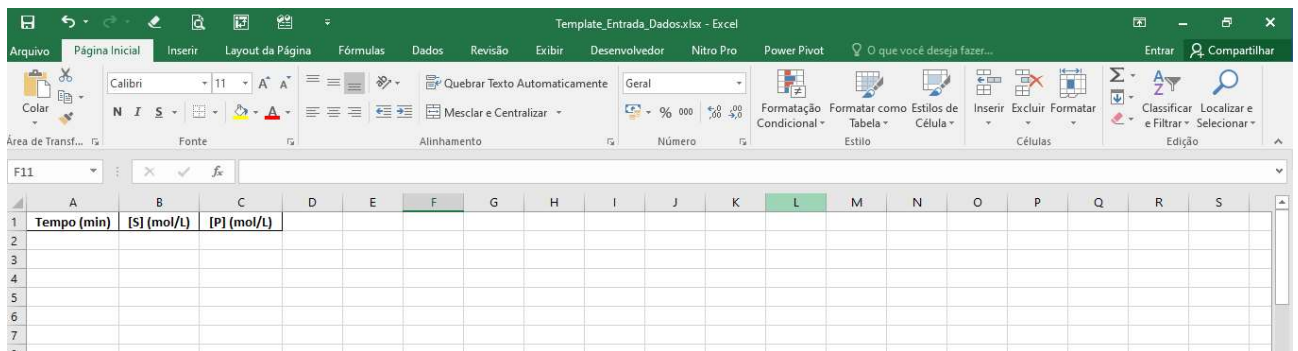
Fonte: Autora, 2023.

1. Imagem representativa de um biorreator STR em batelada e balanço de massa da reação enzimática em batelada, com aplicação do modelo cinético de Michaelis-Menten, da mesma forma que é exibida na sessão de Simulação;

2. Área de inserção de dados. Ao selecionar o primeiro botão, de Baixar Template, abre-se um arquivo *excel* com o template aceito pelo *software*. O arquivo, de nome “Template_Entrada_Dados”, como mostra a Figura 37 abaixo, é utilizado para inserção de dados experimentais por parte do usuário. As colunas a serem preenchidas são de Tempo (min), concentração do substrato - [S] (mol/L) e concentração de produto - [P] (mol/L).

O botão Importar Arquivo, por sua vez, abre a caixa de seleção de arquivos do sistema do computador do usuário, permitindo a seleção de arquivos no formato *excel*, com os dados experimentais a serem modelados. Esses arquivos devem sempre seguir o modelo do template.

Figura 37. Arquivo *excel* aberto ao selecionar o botão Baixar Template. O template deve ser utilizado como padrão pelo usuário para a importação dos dados experimentais para a modelagem da reação enzimática.



Fonte: Autora, 2023.

3. Parâmetros da enzima modelada, onde o usuário deve inserir o dado de concentração inicial da enzima em mol/L (E_0). É apresentada também a equação da Velocidade Máxima a partir do produto da concentração de enzima ($[E_0]$) com a constante enzimática (k_{cat}). O botão Modelar deve ser acionado pelo usuário após o preenchimento do campo, assim como a importação do arquivo em *excel* com os dados experimentais.

4. Nesta área, quando concluída a modelagem da reação enzimática a partir dos dados experimentais inseridos, são apresentados os parâmetros estimados de Velocidade Máxima ($V_{m\acute{a}x}$), Constante de Michaelis-Menten (K_m) e Constante enzimática (k_{cat}).

5.5.1 MODELAGEM: BACK-END

De maneira geral, a modelagem é realizada a partir da importação de um arquivo *excel* com dados de concentração de substrato e de produto por um período de tempo, como definido pelo template. Após a importação dos dados experimentais, deve ser inserido o dado de concentração inicial de enzima (E0) e acionado o botão de Modelar.

O *software* processa todos os dados e através da integração dos Algoritmos Genético e de Levenberg-Marquardt e determina os parâmetros estimados da curva cinética otimizada.

Inicialmente, a convergência global é realizada pelo Algoritmo Genético, por meio da evolução de grupos de soluções ao longo de várias gerações. Uma população inicial é criada aleatoriamente, e cada uma das soluções é avaliada em *scores* em relação ao problema de otimização, que indicam o quão ideal é o conjunto de parâmetros.

Os *scores* de aptidão são calculados com base na função de aptidão, que é a função matemática que avalia cada solução gerada em relação ao problema de otimização proposto (SIVASHANMUGAM; SANKARANARAYANAN, 2018). O *score* de aptidão é geralmente definido como o inverso da função de aptidão, ou seja, melhores soluções possuem um *score* mais elevado.

As melhores soluções são, então, selecionadas e combinadas a fim de produzir novos conjuntos descendentes, que são, por sua vez, avaliados. Dessa maneira, o algoritmo vai aproximando-se da resposta ótima para o problema até convergir em uma solução ótima ou a mais próxima possível. Neste ponto, a população se estabiliza em torno da solução que tem o *score* de aptidão considerado alto o suficiente (KHAN; KHAN; AHMED, 2015).

O Algoritmo elaborado por Levenberg-Marquardt, por sua vez, foi aplicado como parte do processo de convergência do Algoritmo Genético, utilizando a resposta do primeiro como conjunto inicial para iniciar o processo de convergência local, dentro do intervalo de busca delimitado pelo Algoritmo Genético, permitindo soluções ainda mais precisas e otimizadas.

Seu mecanismo de funcionamento é centrado no cálculo do erro quadrático atribuído à avaliação da distância numérica entre os dados experimentais inseridos, gerados através da simulação, e os valores tratados pelo algoritmo como modelo. Esse cálculo é realizado através da equação linear descritiva do processo, com intuito de potencializar e promover uma convergência adicional localmente à solução já determinada anteriormente pelo outro algoritmo (DEB *et al.*, 2002).

Desse modo, o *software* retorna os valores estimados otimizados dos parâmetros cinéticos da reação enzimática: velocidade máxima, constante de catálise e constante de Michaelis-Mente, a partir dos dados experimentais que foram inseridos no sistema.

5.5.2 MODELAGEM: TESTE FINAL

Os dados gerados pelo exemplo da simulação (Item 5.4.5 Exibição Gráfica) no *excel* foram exportados pelo botão Arquivo Excel disponível no *software* e foram inseridos na Modelagem, como mostra a Figura 38 abaixo.

Figura 38. Arquivo *excel* gerado pelo *software Enzinetics* com os valores aleatórios não-reais inseridos no sistema: $K_m = 2$, $E_0 = 5$ mol/L, $K_{cat} = 3$ 1/s, $S_0 = 100$ mol/L e $P_0 = 0$ mol/L.

Tempo (min)	S (mol/L)	P (mol/L)
0	100	0
1	85,3175794	14,68242064
2	70,6936296	29,30637036
3	56,1541403	43,84585972
4	41,7470839	58,25291611
5	27,5764226	72,42357742
6	13,9407191	86,05928092
7	2,43250494	97,56749506
8	0,00452974	99,99547026
9	2,5111E-06	99,99999749
10	1,4871E-09	100

Fonte: Autora, 2023.

Após a importação do arquivo gerado no módulo de Simulação e a inserção do dado de concentração inicial de enzima (E_0) de 5 mol/L (como simulado inicialmente),

o programa fez as devidas estimativas através do algoritmo genético e algoritmo de Levenberg-Marquardt combinados, e retornou os valores estimados, os quais conferem com os parâmetros inicialmente inseridos na simulação, como demonstra a Figura 39 a seguir, demonstrando a assertividade do algoritmo desenvolvido.

Figura 39. Modelagem realizada pelo *software Enzinetics*, com retorno dos valores cinéticos estimados, que conferem com os valores inicialmente simulados.



Fonte: Autora, 2023.

6. CONCLUSÃO

Com o avanço da computação científica, a simulação e modelagem de reações cinéticas apresentam-se como vantajosas ferramentas para otimização e controle de bioprocessos, visto a crescente utilização de enzimas na indústria. Desse modo, a compreensão e estudo das cinéticas enzimáticas faz-se fundamental.

A linguagem Python emerge como uma alternativa poderosa aos softwares pagos já estabelecidos no mercado. No desenvolvimento do *software Enzinetics*, a linguagem mostrou-se altamente eficiente, atendendo a todos os requisitos necessários, tanto para a construção dos algoritmos *Back-End* de simulação e modelagem, quanto para a estruturação *Front-End*.

Ademais, as visualizações gráficas e a documentação de cada um dos modelos, assim como do modo de operação em batelada, mostraram-se ferramentas facilitadoras para o ensino e compreensão do comportamento dos bioprocessos enzimáticos.

Os algoritmos desenvolvidos, portanto, atingiram seu objetivo ao integrarem uma ferramenta de fácil uso, disponibilizada universalmente, que pode ser utilizada para compreensão das melhores condições de operação e otimização dos processos enzimáticos.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ABRAHÃO NETO, J. Algumas aplicações de enzimas. In: LIMA, U. A.; AQUARONE, E.; BORZANI, W.; SCHMIDELL, W. (coord.). **Biotecnologia industrial - processos fermentativos e enzimáticos**. p. 405-412. São Paulo: Edgard Blücher Ltda., 2001.

ALVES, F. **Modelagem e simulação de biorreator operando com fungos *Trametes versicolor* para produção de enzima lacase**. 2010. Dissertação (Mestrado em Engenharia de Processos Químicos e Bioquímicos) - Escola de Engenharia Mauá, São Caetano do Sul, 2010.

ANACONDA. **Anaconda Library**. 2022. Disponível em: <https://www.anaconda.com/library>. Acesso em: 13 de julho de 2022.

ANDREAUS, J.; CAVACO-PAULO, A. Enzimas no processamento de fibras têxteis. In: BON, E. P. S.; FERRARA, M. A. A.; CORVO, M. L. (ed.). **Enzimas em Biotecnologia: produção, aplicações e mercado**. Rio de Janeiro: Interciência, p. 179-204, 2008.

ARI, N.; USTAZHANOV, M. Matplotlib in python. In: 11TH INTERNATIONAL CONFERENCE ON ELECTRONICS, COMPUTER AND COMPUTATIONS (ICECCO). 2014.

BAS, D.; DUDAK, F.C.; BOYACI, I.H. Modeling and optimization IV: investigation of reaction kinetics and kinetic constants using a program in which artificial neural network (ANN) was integrated. **J. Food Eng**, n. 79, p.1152-1158, 2007.

BOARETO, A. J. M.; **Modelagem matemática híbrida cibernética / neural do processo contínuo de produção de etanol por *Zymomonas mobilis***. 2012. 249f. Tese (Doutorado em Ciência em Tecnologia de Processos Químicos e Bioquímicos) – Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2012.

BORGES, L. E. **Python para desenvolvedores**: aborda Python 3.3. Novatec Editora. São Paulo, 2014.

BORZANI, W. *et al.* **Biotecnologia Industrial: fundamentos**. São Paulo: Editora Edgard Blucher Ltda, 2001.

BRESSERT, E. **SciPy and NumPy: an overview for developers**. Editora O'Reilly Media Inc., 2012.

BRIGGS, G. E.; HALDANE, J. B. S. A Note on the Kinetics of Enzyme Action. **Biochem. J.**, vol. 19, p. 338-339, 1925.

BURIOL, T. M.; ARGENTA, M. A. Acelerando o desenvolvimento e o processamento de análises numéricas computacionais utilizando Python e CUDA. **Métodos Numéricos e Computacionais em Engenharia-CMNE CILAMCE**, 2009.

CAMPOS, V. H. G.; DE OLIVEIRA, R. **Computação Científica usando Python**. Rondonópolis: Edição do autor, 2016.

CARVALHO, N. M. F. *et al.* Uso de equações lineares na determinação dos parâmetros de MichaelisMenten. **Química Nova**, n. 33, p. 1607-1611, 2010.

CHEN, D. Y. **Análise de dados com Python e Pandas**. Novatec Editora, 2018.

CHEN, D. Y. **Pandas for everyone: Python data analysis**. Addison-Wesley Professional, 2017.

COELHO, F. C. **Computação Científica com Python**. Edição do autor. Petrópolis: 2007.

COLEN, G. **Isolamento e seleção de fungos filamentosos produtores de lipases**. 2006. 206 f. Tese (Doutorado em Ciência de Alimentos) – Universidade Federal de Minas Gerais, Belo Horizonte, 2006.

CRISTOVÃO, L. M., *et al.* Modelagem e simulação aplicadas à otimização de processos industriais. **Revista de Engenharia Industrial**, vol. 25, p. 32-47, 2021.

CUNHA, A. G. **Purificação e imobilização de lipases microbianas em suportes com diferentes graus de hidrofobicidade**. 2007. Dissertação de Mestrado - Instituto de Química, Universidade Federal do Rio de Janeiro. Rio de Janeiro, Brasil, 2007.

DEB, K. *et al.* A fast and elitist multiobjective genetic algorithm: NSGA-II. **IEEE transactions on evolutionary computation**, v. 6, n. 2, p. 182-197, 2002.

DORAN, P. M. **Bioprocess Engineering principles**. 2. ed. United Kingdom: Elsevier, 2013.

DORN, C. *et al.* A novel Levenberg-Marquardt based approach for parameter estimation in dynamic models. **Chemical Engineering Science**, v. 204, p. 120-131, 2019.

EIBEN, A. E. ; SMITH, J. E. **Introduction to evolutionary computing**. Springer, 2003.

EYRING, H.; DANIELS, F. The decomposition of nitrogen pentoxide in inert solvents. **Journal of the American Chemical Society**, v. 52, n. 4, p. 1472-1486, 1930.

FERNANDES, P. Enzymes in food processing: a condensed overview on strategies for better biocatalysts. **Enzyme Research**, v. 201, n. 1, 2010.

FISCHER, E. Einfluss der Configuration auf die Wirkung der Enzyme. **Berichte der deutschen chemischen Gesellschaft**, v. 27, n. 3, p. 2985-2993, 1894.

FOGLER, H. S. Elements of chemical reaction engineering. **Chemical Engineering Guide**, Michigan Engineering from University of Michigan, 1999.

FOX, C. A baby care skin protectant. **Cosmetics & Toiletries**, n. 8, p. 34-38, ago. 2005.

GALITZ, W. O. **The Essential Guide to User Interface Design: An Introduction to GUI Design Principles and Techniques**. 3rd. ed. Indianapolis: Wiley Publishing, Inc., 2007.

GARG, G.; SEHRAWAT, N.; YADAV, M. Role of Enzymes in Food IndustriesFrontiers in Food Biotechnology. In: SHARMA, C., SHARMA A. K., ANEJA, K. R. **Frontiers in Food Biotechnology**. New York: Nova Publishers, cap. 9, p. 219-252, 2016.

GONÇALVES, C. F. **Modelagem, simulação e otimização de cultivos autotróficos de Pseudoneochloris marina em fotobiorreatores airlift**. 113f. 2019. Dissertação (Mestrado em Ciências e Tecnologia dos Alimentos) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2019.

GRAYSON, J. E. **Python and Tkinter programming**. Greenwich: Manning Publications Co., 2000.

HARRIS, C. R. *et al.* Array programming with NumPy. **Nature**, v. 585, n. 7825, p. 357-362, 2020.

HEYS, J. J. **Chemical and biomedical engineering calculations using Python**. Montana: Editora John Wiley & Sons, 2017.

HUNTER, J. *et al.* **Matplotlib**. 2021. Disponível em: <https://matplotlib.org/Matplotlib.pdf>. Acesso em: 11 de novembro de 2022.

ISLAM, Q. N. **Mastering PyCharm**. Birmingham: Editora Packt Publishing Ltd., 2015.

JETBRAINS. **PyCharm documentarion** – version 2021.2. 2021. Disponível em: <https://www.jetbrains.com/help/pycharm/quick-start-guide.html>. Acesso em: 28 de novembro de 2021.

JOHNSON, A. W. **Enzymes: Biochemical Catalysts**. New York: Springer, 2021.

JOHNSON, K. A.; GOODY, R. S. The original Michaelis constant: translation of the 1913 Michaelis-Menten paper. **Biochemistry**, Washington, v. 50, n. 39, p. 8264-8269, 2011.

JONES, C. D. *et al.* Avanços na automação e controle de processos industriais: uma revisão abrangente. **Journal of Industrial Technology**, vol. 34, p. 87-105, 2021.

JUPYTER NOTEBOOK. **Jupyter Notebook: User Documentation**. 2022. Disponível em: <https://jupyter-notebook.readthedocs.io/en/latest/notebook.html>. Acesso em: 13 de julho de 2022.

KHAN, M. A.; KHAN, N. M.; AHMED, M. O. A. Genetic algorithms: a review of their applications in data mining. **Journal of Emerging Technologies in Web Intelligence**, v. 7, n. 3, p. 254-267, 2015.

KHEREDDINE, R. *et al.* A Comparative study between genetic algorithm and particle swarm optimization in fitting mathematical models. **International Journal of Advanced Computer Science and Applications**, v. 9, n. 4, 2018.

KOLHE, P. R.; KHETRI, G. P.; DESHMUKH, N. K. Study of standard assumptions of graphical user interface (GUI) based on usability, adaptability and security factors. **International Journal of Advanced Research in Computer Science and Software Engineering**, v. 2, p. 1-7, 2012.

KULISCH, U. W.; MIRANKER, W. L. **A new approach to scientific computation**. Nova Iorque: Editora Elsevier, 2014.

LEHNINGER, T. M.; NELSON, D. L.; COX, M. M. **Princípios de Bioquímica**. 6ª Edição. Madison: Ed. Artmed, 2012.

LERIN, L. A *et. al.* Enzymatic synthesis of ascorbyl palmitate in organic solvents: process optimization and kinetic evaluation. **Food Bioprocess Technol**, vol. 5, p. 1068-1076, 2012.

LIMA, M.; COELHO, B.; TAKIGAWA, F. Ferramentas e recursos disponíveis para reconhecimento de fala em Português Brasileiro, v. 12, p. 475-479, 2021.

LINEWEAVER, H.; BURK, D. The determination of enzyme dissociation constants. **Journal of the American chemical society**, vol. 56, n. 3, p. 658-666, 1934.

MARQUARDT, D. W. An algorithm for least-squares estimation of nonlinear parameters. **Journal of the Society for Industrial and Applied Mathematics**, v. 11, n. 2, p. 431-441, 1963.

MARTINS, A. R. Representação do efeito de inibição enzimática reversível para o modelo cinético de Michaelis-Menten no estado transiente. **Brazilian Journal of Food Technology**, vol. 18, n.2, p-112-120. Campinas, 2015.

MASHREGHI, A.; YAZDI, H. S. A recursive algorithm for optimizing differentiation. **J. Comput. Appl. Math**, vol. 263, p. 1-13, 2014.

MASIERO, A. A.; FERREIRA, L. A.; AQUINO, P. T. Algoritmos de clusterização e python científico apoiando modelagem de usuário. Livro dos Tutoriais do sobre Fatores Humanos, p. 42, 2012.

MATPLOTLIB.ORG.[Funções básicas biblioteca Matplotlib]. 2022. Disponível em: <https://matplotlib.org/>. Acesso em: 22 abr. 2022.

MCKINNEY, W. **Pandas**: powerful Python data analysis toolkit. Release 0.7.3. 2012. Disponível em: <https://pandas.pydata.org/pandas-docs/version/0.7.3/pandas.pdf>. Acesso em: 23 de novembro de 2021.

MCKINNEY, W. **Python para análise de dados**: Tratamento de dados com Pandas, NumPy e IPython. Novatec Editora, 2019.

MENDESA, Á. J. B.; VALDMANA, B.; SOUZA JÚNIOR, M. B. Uma revisão de modelagem matemática em bioprocessos. **Ciência e Tecnologia de Alimentos**, p. 40, 2011.

MENDOZA, A. B., *et al.* Avanços recentes em modelagem e simulação para otimização de processos industriais. **Journal of Industrial Engineering and Management**, vol. 15, p. 156-175, 2022.

MICHAELIS L.; MENTEN M. L. Die Kinetik der Invertinwirkung. **Biochemistry**, 1913.

MILLMAN, K. J.; AIVAZIS, M. Python for scientists and engineers. **Computing in Science & Engineering**, v. 13, n. 2, p. 9-12, 2011.

MONTEIRO, V. N.; SILVA, R. N. Aplicações industriais da biotecnologia enzimática. **Revista processos químicos**, v. 3, n. 5, p. 9-23, 2009.

NELLI, F. **Python data analytics**: Data analysis and science using Pandas, Matplotlib and the Python programming language. Apress, 2015.

NIELSEN, J. *et al.* **Princípios de engenharia de bioreação**. Ed. 2. Nova Iorque: Kluwer Academic/ Plenum Publishers, 2003.

NOCEDAL, J.; WRIGHT, S. J. **Numerical Optimization**. Springer, 2006.

NUMPY.ORG. [Funções básicas biblioteca Numpy]. 2021. Disponível em: <https://numpy.org/>. Acesso em: 23 abr. 2022.

OLIPHANT, T. E. **A guide to NumPy**. Nova Iorque: Trelgol Publishing, 2006.

OLIVEIRA, R. S., *et al.* Estratégias de operação otimizadas utilizando modelagem e simulação de processos: estudo de caso na indústria química. **Revista Brasileira de Engenharia Química**, vol. 37, p. 1123-1136, 2020.

PAIVA, C. L. A.; SÁ-PEREIRA, P. A aplicação da Biologia Molecular no aprimoramento da produção de enzimas. n: BON, E. P. S.; FERRARA, M. A. A.; CORVO, M. L. (ed.). **Enzimas em Biotecnologia: produção, aplicações e mercado**. Rio de Janeiro: Interciência, p. 29-53, 2008.

PANDAS.PYDATA.ORG. [Funções básicas biblioteca Pandas]. 2021. Disponível em: <https://pandas.pydata.org/>. Acesso em: 23 abr. 2022.

PATEL, R. N. Microbial/enzymatic synthesis of chiral intermediates for pharmaceuticals. **Enzyme and Microbial Technology**, Atlanta, v. 31, n. 6, p. 804-826, 2002.

PEREIRA, A. S. *et al.* Implementação em linguagem python de métodos de estimação e comparação de parâmetros cinéticos aplicados a hidrólise catalisada por ALDC. Campinas, 2015.

PEREIRA, J. A. Introdução à cinética enzimática. Cinética da catálise pela invertase. **Termodinâmica MIB**. Instituto de Ciências Biomédicas de Abel Salazar – ICBAS-UP, 2015.

PERKEL, J. M. Why Jupyter is data scientists' computational notebook of choice. **Nature**, vol. 563, p. 145–146, 2018.

PIMENTEL, J. F. *et al.* **A large-scale study about quality and reproducibility of jupyter notebooks**. p. 507–517, 2019.

PIMENTEL, J. F. *et al.* Ciência de dados com reprodutibilidade usando jupyter. **Sociedade Brasileira de Computação**, 2021.

PINTO, G. F; DE MENEZES, R. R. **Cinética enzimática**. Rio de Janeiro: Editora E-papers, p. 7-11, 2009.

PIZARRO, A.V.L.; PARK, E.Y. Lipase-catalysed production of biodiesel fuel from vegetable oils contained in waste activated bleaching earth. **Process Biochemistry**, London, v. 38, n. 7, p. 1077-1082, 2003.

PYTHON SOFTWARE FOUNDATION. **Python 3.10 documentation** – instalando módulos Python, 2021. Disponível em: <https://docs.python.org/pt-br/3.10/installing/index.html>. Acesso em: 25 de novembro de 2022.

PYTHON SOFTWARE FOUNDATION. **Python Language documentation**, 2020. Disponível em: <https://www.python.org/doc/PythonLanguage>. Acesso em: 06 de novembro de 2022.

RANDLES, B. M. *et al.* **Using the Jupyter Notebook as a Tool for Open Science: An Empirical Study**, 2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL), pg. 1-2, 2017.

RINGE, D.; PETSKO, G. A. How enzymes work. **Science**, v. 320, n. 5882, p. 1428-1429, 2008.

ROMERO, M. D. *et al.* Kinetic resolution of (R,S)-2-butanol using enzymatic synthesis of esters. **Appl. Biochem. Biotechnol**, n. 165, 2011.

SANT'ANNA JUNIOR, G. L. Produção de enzimas microbianas. In: LIMA, U. A.; AQUARONE, E.; BORZANI, W.; SCHMIDELL, W. (coord.). **Biotecnologia industrial - processos fermentativos e enzimáticos**. São Paulo: Edgard Blücher Ltda., p. 351-362, 2001.

SAWAKI, R. V. *et al.* Modelagem e simulação do desempenho de reatores de fluxo contínuo e em batelada no tratamento de efluente têxtil utilizando o software livre python. **Blucher Chemical Engineering Proceedings**, v. 1, n. 3, p. 2648-2654, 2015.

SAYAMA, H. **Introduction to the modeling and analysis of complex systems**. Open SUNY Textbooks, 2015.

SCHMIDELL, W. *et al.* **Biotecnologia Industrial: Engenharia Bioquímica**. São Paulo: Edgard Blucher Ltda. vol 2, 2001.

SCHWAAB, M.; PINTO, J. C.; **Análise de dados experimentais I: fundamentos de estatística e estimação de parâmetros**. Rio de Janeiro: E-papers Serviços Editoriais, Brasil, 2007.

SCIPY.ORG. **SciPy user guide**, 2019. Disponível em: <https://docs.scipy.org/doc/scipy/reference/tutorial/general.html>. Acesso em: 20 de novembro de 2022.

SENA, A. R *et al.* Seleção de fungos do semi-árido baiano secretores de hidrolases de interesse em alimentos. **Sitientibus**, Feira de Santana, n. 35, p. 91-98, 2006.

SILVA, G. C. Monitoramento remoto do espectro solar. n. 24, 2020.

SILVA, J. L. **Modelagem e simulação de reatores autoclave para produção de PEBD**. 2012. Dissertação (Mestrado em Engenharia, Programa De Pós-Graduação Em Engenharia Química) (Área de Concentração: Modelagem e Simulação de Processos) - Departamento de Engenharia Química, Universidade Federal do Rio Grande do Sul, Porto Alegre, 2012.

SINGH, S.; DEB, K. **Multi-objective optimization using evolutionary algorithms: an introduction**. Índia: John Wiley & Sons, 2021.

SINGH, V. P. **Modelação e simulação de sistemas**. Índia: New Age International Publishers, 2009.

SINGH, V. P.; WOOLHISER, D. A. Mathematical modeling of watershed hydrology. **J. Hydrol. Eng.** n. 7, p. 270-292, 2002.

SIVASHANMUGAM, P.; SANKARANARAYANAN, S. A review on genetic algorithms and its applications. **Alexandria Engineering Journal**, v. 57, n. 1, p. 363-371, 2018.

SMITH, A. B. Modelagem e simulação de processos industriais: tendências recentes e desafios futuros. **Revista Internacional de Engenharia Industrial**, vol. 18, n. 3, p. 45-62, 2022.

SMITH, L. R. *et al.* **Principles of enzymology: structure, function, mechanism**. 3rd ed. Oxford: Oxford University Press, 2020.

SOUSA, J. R. *et al.* Python e predição de dados usando redes neurais multicamadas. **Brazilian Journal of Development**, v. 6, n. 7, p. 54181-54185, 2020.

TREMONTE, P. *et al.* Raw milk from vending machines: effects of boiling , microwave treatment , and refrigeration on microbiological quality. **Journal of Dairy Science**, v. 97, n. 6, p. 3314-3320, 2014.

VAN DAM, H. W.; HILLE, J. D. R. Yeast and enzymes in breadmaking. **Cereal Foods World**, Saint Paul, v. 37, n. 3, p. 245-252, 1992.

VAN DER WALT, S.; COLBERT, S. C.; VAROQUAUX, G. The NumPy array: a structure for efficient numerical computation. **Computing in science & engineering**, v. 13, n. 2, p. 22-30, 2011.

VASIC-RACKI, D.; KRAGL, U.; LIESE, A. Benefits of enzyme kinetics modelling. **Chem. Biochem. Eng. Q.** vol.17, ed. 1, p. 3–14, 2003.

VIEIRA, S. L. Oportunidade para o uso de enzimas em dietas vegetarianas. Chapecó: 2003.

VIRTANEN, P. *et al.* SciPy 1.0: fundamental algorithms for scientific computing in Python. **Nature methods**, v. 17, n. 3, p. 261-272, 2020.

VITOLLO, M. 2001. Aplicações de enzimas na tecnologia de alimentos. In: AQUARONE, E.; BORZANI, W.; SCHMIDELL, W.; LIMA, U. A. (coord.). **Biotecnologia na produção de alimentos**. São Paulo: Edgard Blücher Ltda., p. 387-420, 2001.