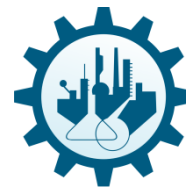




UNIVERSIDADE FEDERAL DE CAMPINA GRANDE  
CENTRO DE CIÊNCIAS E TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA



Damiris Valeska Farias Vilar

**OTIMIZAÇÃO EM TEMPO REAL DE UMA PLANTA DE  
TRATAMENTO DE EFLUENTES UTILIZANDO TÉCNICAS DE  
APRENDIZADO DE MÁQUINA**

CAMPINA GRANDE, PARAÍBA, BRASIL

Setembro de 2023

DAMIRIS VALESKA FARIAS VILAR

**OTIMIZAÇÃO EM TEMPO REAL DE UMA PLANTA DE TRATAMENTO DE  
EFLUENTES UTILIZANDO TÉCNICAS DE APRENDIZADO DE MÁQUINA**

Dissertação de Mestrado submetida à Coordenação do Programa de Pós-Graduação em Engenharia Química, da Universidade Federal de Campina Grande, como parte dos requisitos necessários para obtenção do título de Mestre em Engenharia Química.

Área de concentração: Modelagem e Simulação.

Orientador: Prof. Dr. Antonio Carlos Brandão de Araújo

CAMPINA GRANDE, PARAÍBA, BRASIL

Setembro de 2023

V697o

Vilar, Damiris Valeska Farias.

Otimização em tempo real de uma planta de tratamento de efluentes utilizando técnicas de aprendizado de máquina / Damiris Valeska Farias Vilar – Campina Grande, 2023.

101 f. : il. color.

Dissertação (Mestrado em Engenharia Química) - Universidade Federal de Campina Grande, Centro de Ciências e Tecnologia, 2023.

"Orientação: Prof. Dr. Antonio Carlos Brandão de Araújo."

Referências.

1. Tratamento de Águas Residuais. 2. Tratamento de Efluentes. 3. Reutilização de Águas Tratadas. 4. BSMI. 5. Metamodelos. 6. Kriging. 7. RTO. I. Araújo, Antonio Carlos Brandão de. II. Título.

CDU 628.2/3(043)

DAMIRIS VALESKA FARIAS VILAR

**OTIMIZAÇÃO EM TEMPO REAL DE UMA PLANTA DE TRATAMENTO DE  
EFLUENTES UTILIZANDO TÉCNICAS DE APRENDIZADO DE MÁQUINA**

Dissertação de Mestrado apresentada e aprovada em 28 de setembro de 2023

Banca Examinadora:

---

Dr. Antonio Carlos Brandão de Araújo (UAEQ/CCT/UFCG)  
Orientador

---

Dr. Heleno Bispo da Silva Junior (UAEQ/CCT/UFCG)  
Examinador Interno

---

Dr. Sidinei Kleber da Silva (UAEQ/CCT/UFCG)  
Examinador Externo

CAMPINA GRANDE, PARAÍBA, BRASIL

Setembro de 2023



MINISTÉRIO DA EDUCAÇÃO  
**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE**  
UNIDADE ACADÊMICA DE ENGENHARIA QUÍMICA  
Rua Aprígio Veloso, 882, - Bairro Universitário, Campina Grande/PB, CEP 58429-900  
Telefone: (83) 2101-1100  
Site: <http://cct.ufcg.edu.br>

### REGISTRO DE PRESENÇA E ASSINATURAS

ATA DA DEFESA PARA CONCESSÃO DO **GRAU DE MESTRE** EM ENGENHARIA QUÍMICA,  
REALIZADA EM 28 DE SETEMBRO DE 2023

(Nº 518)

CANDIDATO(A): **Damiris Valeska Farias Vilar**. COMISSÃO EXAMINADORA: Antonio Carlos Brandão de Araújo, Prof. Dr. UFCG, **Orientador**, Heleno Bispo da Silva Junior, Prof. Dr. UFCG/ Examinador Interno, Sidinei Kleber da Silva, Prof. Dr. Examinador Externo/ UFCG, Wagner Brandão Ramos, Prof. Dr. Examinador Interno /UFCG. TÍTULO DA DISSERTAÇÃO: “Otimização em Tempo Real de uma Planta de Tratamento de Efluentes Utilizando Técnicas de Aprendizado de Máquina”. ÁREA DE CONCENTRAÇÃO Desenvolvimento de Processos Químicos. HORA DE INÍCIO: **16:00 Horas** – LOCAL: **por videoconferência** Em sessão pública, após exposição de cerca de 50 minutos, o(a) candidato(a) foi arguido(a) oralmente pelos membros da Comissão Examinadora, tendo demonstrado suficiência de conhecimento e capacidade de sistematização, no tema de sua dissertação, obtendo conceito **APROVADO**. Face à aprovação, declara o(a) orientador, achar-se a examinada, legalmente habilitado(a) a receber o Grau de Mestre Engenharia Química, cabendo a Universidade Federal de Campina Grande, como de direito, providenciar a expedição do Diploma, a que o(a) mesmo(a) faz jus. Na forma regulamentar, foi lavrada a presente ata, que é assinada por mim, Maricé Pereira de Araújo – Secretária do PPGEQ, os membros da Comissão Examinadora e o(a) candidato(a). Campina Grande, 28 de setembro de 2023.

Maricé Pereira de Araújo

Secretária

Antonio Carlos Brandão de Araújo, Prof., Dr., UFCG

Orientador

Heleno Bispo da Silva Junior, Prof., Dr., UFCG

Examinador Interno

Wagner Brandão Ramos, Prof., Dr., UFCG

Examinador Interno

Sidinei Kleber da Silva, Prof., Dr., UFCG

Examinador Externo

Damiris Valeska Farias Vilar

Candidata



Documento assinado eletronicamente por **HELENO BISPO DA SILVA JUNIOR, PROFESSOR 3 GRAU**, em 28/09/2023, às 17:19, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



Documento assinado eletronicamente por **ANTONIO CARLOS BRANDAO DE ARAUJO, PROFESSOR 3 GRAU**, em 28/09/2023, às 17:20, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



Documento assinado eletronicamente por **SIDINEI KLEBER DA SILVA, PROFESSOR 3 GRAU**, em 28/09/2023, às 17:23, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



Documento assinado eletronicamente por **MARICE PEREIRA DA SILVA, SECRETÁRIA**, em 03/10/2023, às 15:27, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



Documento assinado eletronicamente por **Damiris Valeska Farias Vilar, Usuário Externo**, em 04/10/2023, às 22:19, conforme horário oficial de Brasília, com fundamento no art. 8º, caput, da [Portaria SEI nº 002, de 25 de outubro de 2018](#).



A autenticidade deste documento pode ser conferida no site <https://sei.ufcg.edu.br/autenticidade>, informando o código verificador **3807334** e o código CRC **OFA13E67**.

## **AGRADECIMENTOS**

À minha mãe, pela confiança depositada em mim, por todo amor e cuidado, paciência, compreensão, incentivo e apoio constante, e principalmente por não medir esforços para me ajudar nessa trajetória, me dando todo suporte necessário para realização dos meus objetivos.

Às minhas irmãs, Daiene, Daíse e Débora, pelos momentos de descontração e apoio, que mesmo distante se fizeram presentes.

Ao Professor Dr. Laércio Gomes, cuja presença foi constante ao longo de minha jornada, desde a graduação até a pós-graduação, por todas as conversas, orientações, apoio e conselhos valiosos que carregarei comigo ao longo de toda a minha vida.

Ao meu orientador, Antonio Carlos Brandão, por todo o apoio, paciência, compreensão e orientação proporcionada. Por em um momento tão desafiador dessa jornada ter sido a luz que eu precisava para extrair o lado positivo dessa experiência.

A todos que direto ou indiretamente me ajudaram para realização dessa pesquisa.

Ao CNPq pelo suporte financeiro.

## RESUMO

O tratamento de águas residuais utilizando o processo de lodos ativados é uma técnica de alta eficiência na remoção de compostos orgânicos e produtos nitrogenados, tornando-se amplamente utilizada e viabilizando a reutilização das águas tratadas. A modelagem computacional tem sido uma ferramenta essencial para melhorar o desempenho dos sistemas de tratamento, permitindo o planejamento e análise das estações de tratamento de efluentes. Diversos modelos preditivos, como o ASM para sistemas de lodos ativados e o BSM para o tratamento biológico em reatores de lodo ativado, têm sido desenvolvidos para avaliar estratégias de controle em estações de tratamento. Com o intuito de aprimorar esses processos, o uso de metamodelos tem sido explorado, oferecendo uma representação simplificada e otimizada do modelo original e resultando em economia de tempo e recursos computacionais em várias áreas, incluindo engenharia e ciência. A pesquisa propõe a otimização em tempo real de uma planta de tratamento de efluentes por meio de técnicas de aprendizado de máquina, utilizando o modelo BSM1 e metamodelos kriging. O estudo visou compreender o desempenho dos otimizadores nas simulações de processos de tratamento de águas residuais, avaliando diferentes métricas para identificar tendências e eficiência. Os resultados evidenciaram a confiabilidade do kriging na geração de metamodelos, com todas as combinações apresentando resultados satisfatórios. Os otimizadores "matlab", "filtersd" e "ipop" mostraram-se eficazes na função objetivo e no atendimento de restrições, enquanto o "nomad" e "nlopt" apresentaram desempenho inferior. A abordagem de otimização RTO demonstrou resultados satisfatórios, possibilitando uma melhor compreensão dos processos envolvidos. A combinação dessas técnicas mostra-se promissora para aprimorar a eficiência operacional das estações de tratamento de águas residuais, com o potencial de contribuir significativamente para uma gestão sustentável dos recursos hídricos e uma redução do impacto ambiental.

**Palavras-chave:** BSM1, Tratamento de efluentes, Metamodelos, Kriging, RTO.



## **ABSTRACT**

Wastewater treatment using the activated sludge process is a highly efficient technique for the removal of organic compounds and nitrogenous products, becoming widely used and enabling the reuse of treated waters. Computational modeling has been an essential tool to enhance the performance of treatment systems, allowing for the planning and analysis of effluent treatment plants. Several predictive models, such as ASM for activated sludge systems and BSM for biological treatment in activated sludge reactors, have been developed to evaluate control strategies in treatment plants. To improve these processes, the use of metamodels has been explored, offering a simplified and optimized representation of the original model and resulting in time and computational resources savings across various fields, including engineering and science. This research proposes real-time optimization of an effluent treatment plant through machine learning techniques, utilizing the BSM1 model and kriging metamodels. The study aimed to understand the performance of optimizers in simulations of wastewater treatment processes, evaluating different metrics to identify trends and efficiency. The results demonstrated the reliability of kriging in generating metamodels, with all combinations yielding satisfactory outcomes. The optimizers "matlab," "filtersd," and "ipop" proved effective in the objective function and compliance with constraints, while "nomad" and "nlopt" exhibited lower performance. The RTO optimization approach yielded satisfactory results, enabling a better understanding of the involved processes. The combination of these techniques holds promise to enhance the operational efficiency of wastewater treatment plants, with the potential to significantly contribute to sustainable water resource management and environmental impact reduction.

**Keywords:** BSM1, Wastewater treatment, Metamodels, Kriging, RTO.

## LISTA DE FIGURAS

<b>Figura 1</b> - Unidade reacional da etapa biológica de um processo de lodo ativado. ....	21
<b>Figura 2</b> - Fluxograma geral de uma unidade de tratamento de esgoto utilizado no BSM1. ...	24
<b>Figura 3</b> - Ciclo reacional no sistema de lodo ativado. ....	26
<b>Figura 4</b> - Modelo unidimensional do sedimentador por camadas. ....	39
<b>Figura 5</b> - LHS bidimensional. ....	44
<b>Figura 6</b> - Diagrama do desenvolvimento da pesquisa. ....	52
<b>Figura 7</b> - Processo do lodo ativado BSM1 em ambiente <i>Simulink</i> a) Visão geral e b) Representação por subsistemas. ....	60
<b>Figura 8</b> - Incorporação do modelo matemático em Matlab no ambiente do <i>Simulink</i> utilizando S-function a) Janela de parâmetros do bloco S-function e b) Início da função chamada pelo bloco. ....	61
<b>Figura 9</b> - Início do Script em MATLAB "rto_app.m". ....	63
<b>Figura 10</b> - Início do Script em MATLAB "optimizer_gpr_app.m". ....	65
<b>Figura 11</b> - Resultado da metamodelagem a partir de dados teste. ....	69
<b>Figura 12</b> - Tempo de Execução em relação a cada otimizador por distúrbio. ....	72
<b>Figura 13</b> - OCI em relação a cada otimizador por distúrbio. ....	73
<b>Figura 14</b> - Variáveis de qualidade do efluente em relação a cada otimizador por distúrbio a) Nitrogênio total; b) Demanda química de oxigênio, c) Concentração de amônia, d) Sólidos suspensos totais e e) Demanda bioquímica de oxigênio. ....	73
<b>Figura 15</b> - Evolução das principais variáveis do RTO. ....	76
<b>Figura 16</b> - Comportamento das variáveis manipuladas do RTO. ....	77

## LISTA DE TABELA

<b>Tabela 1</b> - Processos biológicos básicos do BSM1-ASM1. ....	27
<b>Tabela 2</b> - Variáveis de Estado do Modelo ASM1. ....	33
<b>Tabela 3</b> - Parâmetros estequiométricos do modelo ASM1. ....	33
<b>Tabela 4</b> - Parâmetros cinéticos do modelo ASM1. ....	34
<b>Tabela 5</b> - Parâmetros do modelo do sedimentador.....	40
<b>Tabela 6</b> - Variáveis no influente da planta BSM1. ....	58
<b>Tabela 7</b> - Variáveis manipuladas.....	59
<b>Tabela 8</b> - Variáveis de distúrbio.....	59
<b>Tabela 9</b> – Técnicas estatísticas utilizadas para cada parâmetro na geração do metamodelo. ....	67
<b>Tabela 10</b> – Variáveis de distúrbios aplicadas ao processo.....	68
<b>Tabela 11</b> - Comparação da performance entre os valores da função OCI dos respectivos otimizadores. ....	70

## Sumário

<b>1. INTRODUÇÃO</b> .....	13
<b>2. OBJETIVOS</b> .....	17
2.1 Objetivos específicos .....	17
<b>3. REVISÃO BIBLIOGRÁFICA</b> .....	18
3.1 Tratamento de águas residuais .....	18
3.2 Processo de lodos ativados .....	20
3.2.1 Vantagens do Tratamento Biológico com Lodo Ativado .....	21
3.2.2 Desvantagens do Tratamento Biológico com Lodo Ativado .....	22
3.3 Processo de Digestão Anaeróbica .....	22
3.4 Modelo de BSM1 ( <i>Benchmark Simulation Model N° 1</i> ) .....	23
3.4.1 ASM1 ( <i>Activated Sludge Model N° 1</i> ) .....	25
3.4.2 Processos Biológicos .....	27
3.4.3 Variáveis de Estado do Modelo ASM1 .....	32
3.4.4 Parâmetros Estequiométricos .....	33
3.4.5 Parâmetros Cinéticos .....	34
3.4.6 Restrições e limitações do modelo ASM1 .....	35
3.4.7 Balanço de massa do biorreator .....	36
3.4.8 Modelagem do sedimentador .....	37
3.5 Modelos Substitutos e Metamodelos .....	40
3.5.1 Regressão Linear Múltipla .....	42
3.5.2 Latin Hypercube Sampling .....	43
3.5.3 Modelo de Kriging .....	45
3.5.4 Toolbox Dace .....	47
3.6 Otimização de Processos .....	48
3.6.1 Real Time Optimization .....	49
3.6.2 Otimizadores .....	50
<b>4. MATERIAIS E MÉTODOS</b> .....	52
4.1 Modelagem Matemática .....	53
4.2 Funções de avaliação de desempenho .....	55
4.3 Variáveis de Entrada .....	58
4.4 Simulação do Processo .....	59
4.4.1 Script “rto_app.m” .....	62
4.4.2 Script “optimizer_gpr_app.m” .....	64
<b>5. RESULTADOS E DISCUSSÃO</b> .....	67

<b>5.1</b>	<b>Geração do metamodelo.....</b>	<b>67</b>
<b>5.2</b>	<b>Otimização do processo.....</b>	<b>70</b>
<b>6.</b>	<b>CONCLUSÃO .....</b>	<b>80</b>
	<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>81</b>
	<b>ANEXOS .....</b>	<b>87</b>
	<b>Anexo 1 – Script “rto_app.m” .....</b>	<b>87</b>
	<b>Anexo 2 – Script “optimizer_grp_app.m”.....</b>	<b>93</b>
	<b>Anexo 3 – Resultados obtidos da otimização do processo para diferentes distúrbios.....</b>	<b>94</b>

## 1. INTRODUÇÃO

O monitoramento da qualidade da água é de extrema importância e tem como fim verificar parâmetros para acompanhamento das condições da água ao longo do tempo. Com intuito de se estabelecer um padrão de potabilidade, legislações de qualidade foram criadas. Nelas são apresentados parâmetros e seus valores de referência, estes representam características de avaliação que podem ser divididos em três tópicos: físicos, químicos e biológicos e determinam se a água se adequa para uso e qual tratamento deverá ser submetido (Vilar, 2019).

O tratamento biológico por lodos ativados é amplamente utilizado para tratar águas residuais devido à sua alta eficiência na remoção de compostos orgânicos e produtos nitrogenados presentes nos efluentes. Esse processo de tratamento pode alcançar uma eficiência de remoção de matéria orgânica e produtos nitrogenados, entre 95% e 98%, tornando-o uma das opções mais comuns para o tratamento de efluentes e possibilitando sua reutilização (Hammer & Hammer, 2006; Wiesmann et al., 2007). Modelos de simulação computacional têm sido fundamentais para prever e compreender melhor o comportamento desses processos de tratamento, permitindo uma otimização do desempenho dos sistemas.

A modelagem computacional tem desempenhado um papel importante no planejamento e análise da operação de estações de tratamento de efluentes (ETEs), tendo início na década de 1970. Com o desenvolvimento de diferentes modelos preditivos voltados para a simulação do comportamento dos processos que ocorrem nessas estações e nos meios receptores, tornou-se possível prever e compreender melhor os resultados do tratamento de efluentes (Henze et al., 2000; Di Pierro, 2005; Da Silva, 2008).

Estes modelos de simulação são de grande importância para a adequada operação e gestão dos sistemas de tratamento de esgotos sanitários, visto que, permitem uma melhor compreensão dos processos envolvidos e uma otimização do desempenho do sistema. Eles são desenvolvidos por diferentes grupos de pesquisa, tendo como objetivo principal solucionar os problemas, em escala real, de componentes específicos do sistema (Henze et al., 2000; Di Pierro, 2005; Da Silva, 2008).

Alguns exemplos de modelos de simulação de sistemas de tratamento de efluentes incluem o modelo ASM (*Activated Sludge Model*), utilizado para a simulação de sistemas de lodos ativados, e o modelo WATS (*Water Treatment Simulation*), utilizado para a simulação de sistemas de tratamento de água. Estes modelos são constantemente atualizados e

aprimorados com o objetivo de melhorar a eficiência e a eficácia dos sistemas de tratamento de efluentes (Henze et al., 1987; Gujer et al., 1999).

Temos também o modelo BSM (*Benchmark Simulation Model*) visando estabelecer as condições ideais de operação e gestão das estações de tratamento de águas residuais. O BSM1 é um modelo matemático que simula o processo de tratamento biológico de águas residuais em um reator de lodo ativado, sendo um ambiente dinâmico desenvolvido pelo programa europeu COST/IWA 682 e 624, tendo como objetivo avaliar o desempenho de diferentes estratégias de controle em estações de tratamento de águas residuais (Alex et al., 2008a).

A utilização de metamodelos, que são funções matemáticas simplificadas baseadas em dados de simulações computacionais, possibilita substituir o modelo original em situações que exigem alto esforço computacional ou apresentam limitações de confiabilidade. Seu desenvolvimento é um processo iterativo e colaborativo, que requer a participação ativa dos usuários para garantir que o metamodelo seja eficaz e útil na prática, proporcionando economia de tempo, recursos computacionais e possibilitando a viabilidade de análises mais eficientes (Palmer e Realff, 2002a; Gomes, 2007).

A abordagem da Otimização em Tempo Real (RTO), combinada com o uso de metamodelos, tem se mostrado altamente promissora para otimizar a operação de estações de tratamento de águas residuais. Ao utilizar dados em tempo real e ajustar automaticamente os parâmetros de controle, a RTO possibilita uma tomada de decisão precisa e contínua para maximizar a eficiência dos sistemas, reduzindo custos operacionais e melhorando os resultados do tratamento (Calvo, 2015). Essa aplicação tem potencial para aprimorar significativamente a eficiência dos processos de tratamento de águas residuais, contribuindo para a preservação dos recursos hídricos e redução do impacto ambiental.

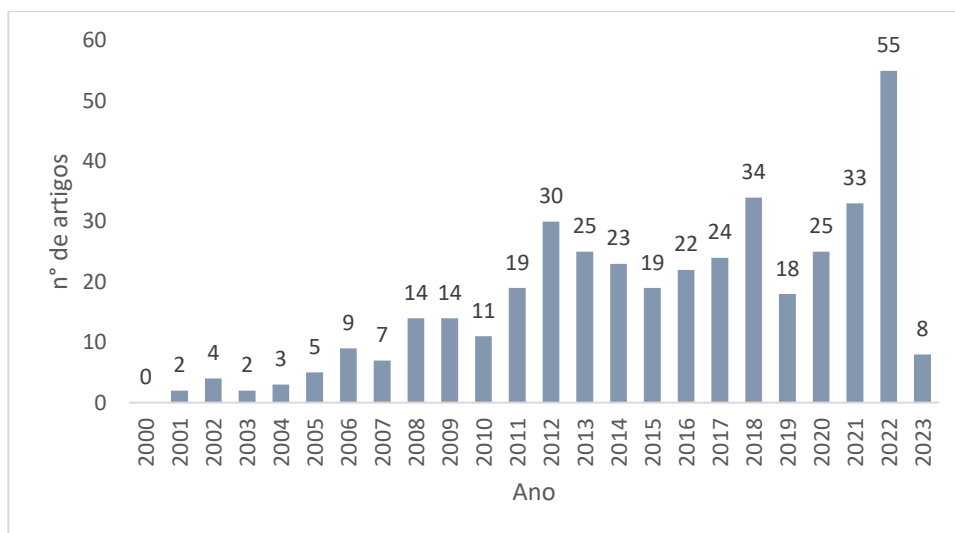
A aplicação da otimização em tempo real em sistemas de tratamento de águas residuais tem apresentado resultados promissores, melhorando significativamente a eficiência dos processos e atingindo as metas de tratamento, ao mesmo tempo em que reduz o consumo de energia e produtos químicos (Haq, 2022). Com o contínuo desenvolvimento de modelos preditivos, otimizadores e técnicas de aprendizado de máquina e o uso associado dessas técnicas permite aprimorar a eficiência do processo e conseqüentemente contribuir para a preservação dos recursos hídricos e redução do impacto ambiental.

Para uma maior confiabilidade e fundamentação da pesquisa, tanto para definição de metodologia, como também para confirmação de inovação, foi realizado um levantamento de dados na base de dados eletrônica SCOPUS, que tem como característica base de dados

multidisciplinar de resumos e de fontes de informação de nível acadêmico. Visto isso, utilizou-se de palavras-chaves para levantamento e criação dos infográficos, 1 e 2, como recurso de visualização em relação ao crescimento de artigos publicados no tema desta pesquisa.

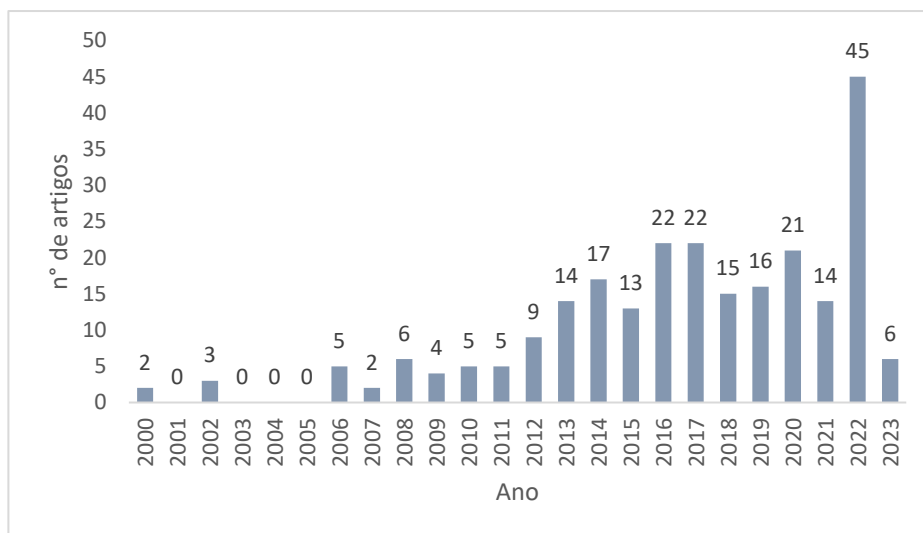
As palavra-chave definidas para esta pesquisa foram: BSM1 e BSM2, com uma faixa de pesquisa dos anos de 2000 a 2023.

**Gráfico 1 – Artigos publicados com a palavra-chave “BSM1”.**



Fonte: Elaborado pela autora.

**Gráfico 2 – Artigos publicados com a palavras-chave: BSM2.**



Fonte: Elaborado pela autora.

Baseado neste contexto a diretriz adotada é compreender as características desse sistema, foi utilizada a plataforma de simulação do *Benchmark Simulation Model N1* (BSM1) visando estabelecer as condições ideais de operação e gestão das estações de tratamento de águas residuais, mais especificamente sua otimização de funcionamento em tempo real através de modelos estatísticos. Essa perspectiva visa propiciar o desenvolvimento de métodos



eficientes para avaliar e validar o funcionamento de unidades reacionais de tratamento de efluentes.

## 2. OBJETIVOS

Este trabalho visa otimizar o processo de tratamento de águas residuais por meio da aplicação da técnica de otimização em tempo real utilizando o modelo BSM1 (*Benchmark Simulation Model n°1*) e metamodelos baseados em técnicas de aprendizado de máquina.

### 2.1 Objetivos específicos

- ❖ Identificar as variáveis do processo de tratamento de águas residuais, manipuladas e controladas, e possíveis distúrbios na alimentação do sistema, para otimizar a escolha das variáveis a serem utilizadas;
- ❖ Implementar o planejamento experimental com base nas variáveis selecionadas para o processo, a fim de coletar dados de entrada não enviesados para a construção dos metamodelos.
- ❖ Propor uma metodologia que utilize a técnica de kriging para otimizar um processo complexo de tratamento de águas residuais;
- ❖ Realizar a simulação do modelo rigoroso BSM1 utilizando a plataforma *Simulink* para obter simulações detalhadas do processo;
- ❖ Construir a *Real Time Optimization* (RTO) para a planta de tratamento de águas residuais, possibilitando a otimização contínua e precisa dos parâmetros de controle do sistema.
- ❖ Utilizar metamodelos na otimização em tempo real da unidade de tratamento de efluentes, permitindo uma representação simplificada e otimizada do modelo original, economizando tempo e recursos computacionais.
- ❖ Validar o metamodelo e otimização do processo com a comparação de dados obtidos da simulação e da otimização, a fim de avaliar a eficiência e a adequação das técnicas de aprendizado de máquina aplicadas.

### 3. REVISÃO BIBLIOGRÁFICA

#### 3.1 Tratamento de águas residuais

O tratamento de águas residuais é um processo essencial para remoção de contaminantes e poluentes da água antes do seu descarte ou reuso. Realizado de acordo com o tipo de necessidade, existem diferentes métodos de tratamento que são selecionados de acordo com o volume da água e o tipo e a quantidade de contaminantes presentes, os mesmos podem ser divididos em três categorias: físicos, químicos e biológicos (Costa, 2016).

Segundo Templeton e Butler (2011) e Costa (2016), seu objetivo é garantir sua disposição segura, sem poluir nossos corpos hídricos e sem afetar a saúde pública, o meio ambiente e a biodiversidade, possibilitando seu reuso e acelerando o sistema natural de depuração de esgotos devido à alta demanda de efluentes gerados diariamente, não sendo possível sua autodepuração por sistemas naturais.

O lançamento de efluentes em corpos d'água devem obedecer aos padrões determinados pelas legislações ambientais vigentes, as quais estabelecem limites de concentração para os diversos poluentes e seus parâmetros. Essas restrições legislativas têm como intuito garantir a preservação e qualidade dos corpos receptores.

Os limites estabelecidos pelas legislações ambientais variam de acordo com o tipo de corpo receptor e com o tipo de uso da água, como abastecimento público, irrigação de culturas, atividades de lazer, entre outros. Os parâmetros de qualidade da água monitorados incluem pH, turbidez, cor, sólidos suspensos, demanda bioquímica de oxigênio (DBO), demanda química de oxigênio (DQO), nutrientes, metais pesados, compostos orgânicos, entre outros.

No Brasil, estes padrões de qualidade são definidos pela Resolução CONAMA nº 430 de 13 de maio de 2011, estabelecendo os limites máximos permitidos de diversos parâmetros de qualidade da água para o lançamento de efluentes em corpos hídricos, e para os padrões de potabilidade de reuso a Resolução do CONAMA nº 357 de 2005 (Brasil, 2005; Brasil, 2011). No Art. n. 16, Resolução CONAMA 430, estão definidos os parâmetros de lançamento, porém, como os efluentes tem diferentes características, de acordo com sua origem, existem variações destes parâmetros, previstos em diferentes artigos da lei, onde essas variações levam em consideração o tipo de atividade geradora do efluente e suas características específicas, que de acordo com Koszma (2016) são:

- Efluentes oriundos de sistemas de tratamento de esgotos sanitários;

- Efluentes oriundos de sistemas de disposição final de resíduos sólidos;
- Efluentes lançados em corpos receptores com registro histórico de floração e cianobactérias;
- Efluentes lançados por meio de emissários submarino;
- Efluentes de serviço de saúde.

Efluentes industriais consistem em vários poluentes, como materiais particulados, óleos e graxas, substâncias tóxicas e microrganismos. A presença de matéria orgânica é um dos principais parâmetros que impactam a qualidade da água em corpos hídricos, uma vez que seu excesso pode provocar um decréscimo da concentração de oxigênio dissolvido (DBO), além de favorecer a proliferação de agentes patogênicos e a eutrofização, que pode produzir toxinas e causar problemas com odor e impedimento da passagem de luz (Bassin e Dezotti, 2008; Leu et al., 1998). Pela Resolução CONAMA 430, a concentração de DBO lançada em corpos hídricos deve ser, no mínimo, 60% menor que a DBO do efluente bruto, a fim de garantir a preservação da qualidade dos corpos hídricos.

Ao projetar um sistema de tratamento de efluentes devemos considerar diversos fatores, como o objetivo do tratamento, o nível de tratamento necessário, o impacto ambiental do lançamento nos corpos hídricos e a eficiência de remoção requerida para cada poluente (Sperling, 2005).

Os tipos de tratamentos de efluentes são feitos de acordo com as exigências do tipo de água e o objetivo de tratamento, sendo comumente classificados em níveis. De acordo com diversos autores, como Mota (2000), Von Sperling (2005), Templeton e Butler (2011) e Metcalf e Eddy (2015) esses níveis incluem:

- **Tratamento preliminar:** realizado através de mecanismo físico de separação, tem como objetivo a remoção de sólidos que podem gerar problemas operacionais no sistema, prevenindo as demais etapas da unidade de tratamento, como entupimentos e desgastes de equipamentos. Esta etapa é realizada com a utilização de grades e caixas de areais para retenção dos sólidos;
- **Tratamento primário:** realizado através de mecanismo físico de separação, tem como objetivo a remoção de sólidos sedimentáveis e parte da matéria orgânica. Nesta etapa é utilizado o decantador para sedimentação dos sólidos, formação do lodo primário bruto e eliminação de até 30% da DBO do efluente;

- **Tratamento secundário:** realizado através de mecanismo biológico e físico-químico, tem como objetivo a remoção da carga orgânica e sólidos não sedimentáveis. Nesta etapa é realizado a autodepuração, que consiste na reprodução de fenômenos naturais para estabilizar a matéria orgânica, e utiliza de equipamentos como biorreatores, lagoas anaeróbias e aeróbicas, tanques de aeração e decantador secundário. Com este tratamento é possível realizar a remoção biológica do efluente e deixá-lo de acordo com os padrões das normas legislativas, alcançando uma remoção de DBO de até 90%;

- **Tratamento terciário:** realizado para remoção de nutrientes e desinfecção da água, tem como objetivo a remoção de poluentes tóxicos ou não biodegradáveis. Utiliza de processos físico-químicos avançados como adsorção, ozonização, radiação UV e desinfecção por cloro ou outros agentes químicos. Este tratamento é especialmente importante em locais onde há uma necessidade de reutilização da água tratada para fins industriais, agrícolas ou mesmo potáveis.

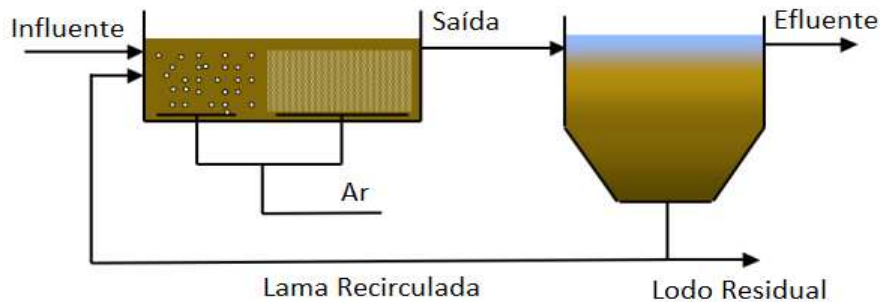
A escolha do tipo e nível de tratamento deve ser baseada em uma análise detalhada das características do efluente a ser tratado, levando em consideração as exigências legais e ambientais aplicáveis.

### 3.2 Processo de lodos ativados

De acordo com Metcalf e Eddy (2015), o tratamento de efluentes por lodo ativado foi desenvolvido na década de 1910 a partir do processo de aeração de esgotos em tanques na Inglaterra, onde foi observado que durante a oxidação da matéria orgânica a mesma ocorria de forma acelerada quando o esgoto era agitado nos tanques de aeração. Este processo foi posteriormente desenvolvido e aprimorado, dando origem ao processo de tratamento de efluentes por lodo ativado altamente eficiente na remoção de matéria orgânica e produtos nitrogenados, entre 95 e 98% de eficiência, tornando-o assim uma das opções mais utilizadas para o tratamento de efluentes e possibilitando sua reutilização (Hammer & Hammer, 2006; Wiesmann et al., 2007; Zoby et al., 2014).

O processo de tratamento por lodo ativado é constituído em um tanque de aeração (reator biológico), um tanque de sedimentação e um sistema de recirculação de lodo, como demonstrado na Figura 1.

**Figura 1** - Unidade reacional da etapa biológica de um processo de lodo ativado.



Fonte: Adaptado de Mulas, 2006.

No tanque de aeração, o efluente é misturado com microrganismos e é agitado para fornecer oxigênio aos microrganismos e estimular a oxidação da matéria orgânica presente no efluente, esse processo de oxidação resulta na formação de lodo ativado. O lodo ativado é então transferido para o tanque de sedimentação, onde a velocidade do fluxo é reduzida para que ocorra a separação entre o lodo e a água tratada (Von Sperling, 1996; Hammer & Hammer, 2006).

O lodo sedimentado é então recirculado para o tanque de aeração para reinoculação e manutenção da população de microrganismos, enquanto a água tratada será submetida para a etapa de tratamento primário. Esse processo de recirculação de lodo permite que o sistema mantenha uma população de microrganismos ativa e saudável, o que resulta em uma alta eficiência de remoção de carga orgânica e substratos nitrogenados do efluente (Von Sperling, 1996; Metcalf e Eddy, 2015; Moura, 2012).

A caracterização do efluente é a etapa mais crítica na implementação do processo de tratamento por lodo ativado, pois é a partir dessa caracterização que é possível determinar as condições ideais de operação do sistema, tais como o volume do tanque de aeração, a taxa de produção de lodo, a taxa de fornecimento de oxigênio, entre outros parâmetros importantes. Além disso, a caracterização do perfil de variação da vazão afluente e a concentração de parâmetros relevantes no efluente tratado são informações essenciais para a operação e monitoramento do sistema (Metcalf e Eddy, 2015).

### **3.2.1 Vantagens do Tratamento Biológico com Lodo Ativado**

Existem diversas vantagens apresentadas pelo tratamento aeróbio por lodos ativados, tais como (Von Sperling, 1997; Hammer e Hammer, 2006; Zoby, 2011):

- a) Alta eficiência na remoção de matéria orgânica biodegradável, podendo alcançar taxas de remoção de até 98%;
- b) Adaptação e variação de carga: o sistema de lodo ativado pode se adaptar facilmente a variações de carga orgânica e hidráulica, permitindo que o processo seja mantido estável em condições adversas;
- c) Diminuição do odor e da cor do efluente tratado;
- d) Possibilidade de remoção biológica de nitrogênio e fósforo;
- e) Reutilização de água reduzindo significativamente a carga poluidora do efluente tratado e possibilitando sua reutilização em atividades como irrigação;
- f) Fácil operação, tendo uma manutenção e gestão fácil e eficiente.

### **3.2.2 Desvantagens do Tratamento Biológico com Lodo Ativado**

As desvantagens apresentadas pelo tratamento aeróbio por lodos ativados são (Von Sperling, 1997; Hammer e Hammer, 2006; Zoby, 2011):

- a) O processo requer fornecimento contínuo de energia para manter o sistema de aeração e a circulação do lodo em funcionamento;
- b) Elevado custo de implantação com equipamentos e infraestrutura, além de custos contínuos de operação e manutenção;
- c) Sensibilidade e variações no efluente, onde o desempenho do processo pode ser afetado negativamente por variações no efluente, como flutuações na carga orgânica, pH, temperatura e nutrientes;
- d) Possíveis problemas ambientais com ruídos, aerossóis e geração de odores desagradáveis caso a ventilação do sistema não seja adequada;
- e) Volume de lodo resultante mais elevado, acrescendo o custo de tratamento e de disposição adequada.

### **3.3 Processo de Digestão Anaeróbica**

A digestão anaeróbica é um processo biológico que ocorre na ausência de oxigênio e é utilizado para decompor matéria orgânica em compostos mais simples, como ácidos orgânicos

e gases. Esse processo é uma alternativa eficiente para tratar efluentes com alta carga orgânica e resistência à digestão, com a vantagem de produzir baixas quantidades de lodo (Britton et al., 2005; Jordaan, 2010; Lyra e Pestana, 2007).

Comparado ao tratamento aeróbico, apresenta benefícios como a produção de biogás, a possibilidade de operar por longos períodos sem alimentação e a remoção de fósforo da biomassa, minimizando problemas com algas. Esse processo é usado em estações de tratamento de esgoto, usinas de biogás e em sistemas de tratamento de resíduos orgânicos (Britton et al., 2005; Jordaan, 2010; Verma, 2012).

O processo de digestão anaeróbica envolve diferentes etapas: hidrólise, acidogênese, acetogênese e metanogênese. Durante a hidrólise, as moléculas complexas de materiais orgânicos são quebradas em moléculas menores, como açúcares, proteínas e gorduras. Na etapa de acidogênese, os açúcares e aminoácidos são fermentados por bactérias anaeróbicas, produzindo ácidos orgânicos como o ácido acético.

Na etapa seguinte, acetogênese, as bactérias acetogênicas consomem os ácidos orgânicos produzidos na etapa anterior e produzem acetato, hidrogênio e dióxido de carbono. Por fim, a metanogênese é a etapa em que as bactérias metanogênicas consomem o hidrogênio e o dióxido de carbono para produzir metano, o qual pode ser coletado e utilizado como biocombustível.

O processo de digestão anaeróbica também oferece a possibilidade de aproveitamento energético do lodo, gerando biogás com alto teor de metano, além de permitir a redução significativa de sólidos voláteis e patógenos (Metcalf e Eddy, 2015). Este processo pode ser controlado de várias maneiras, incluindo a temperatura, o pH e a concentração de nutrientes no meio ambiente, sendo de grande importância manter as condições adequadas para permitir a atividade das bactérias anaeróbicas e garantir a eficiência do processo (Rajeshwari et al., 2000).

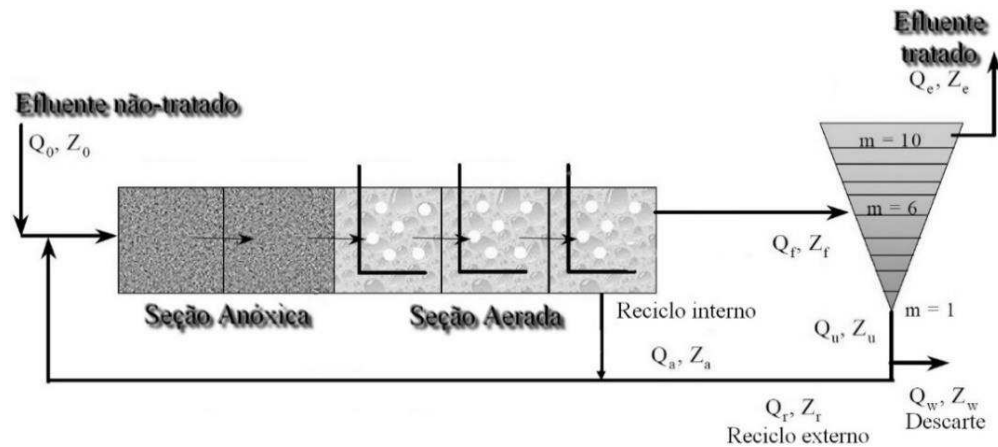
### **3.4 Modelo de BSM1 (*Benchmark Simulation Model N° 1*)**

O *Benchmark Simulation Model* (BSM) é um ambiente dinâmico de simulação desenvolvido pelo programa europeu COST/IWA 682 e 624 com o objetivo de avaliar diferentes estratégias de controle em estações de tratamento de águas residuais. O modelo inclui uma instalação de tratamento de águas residuais, um modelo de simulação, parâmetros do modelo e critérios de avaliação (Copp, 2001; Alex et al., 2008b).



O reator biológico do BSM1, fluxo demonstrado na Figura 2, é composto por cinco compartimentos perfeitamente agitados, onde os dois primeiros são mantidos em condições de anoxia e os três últimos em condições de aerobiose. Esse tipo de configuração é frequentemente utilizado para a remoção biológica de nitrogênio em estações de tratamento de grande escala. Os reatores de lama ativada são seguidos por um clarificador secundário, que tem a função de separar sólidos do líquido e produzir um efluente clarificado (Alex et al., 2008a).

**Figura 2** - Fluxograma geral de uma unidade de tratamento de esgoto utilizado no BSM1.



Fonte: Adaptado de Jeppsson et al., 2011.

O *Benchmark* utiliza o modelo ASM1 (*Active Sludge Model N° 1*) (Henze et al., 1987) para prever os fenômenos biológicos que ocorrem nos tanques de aeração. O decantador secundário é modelado como um sistema não-reativo e unidimensional, dividido em 10 camadas horizontais, de acordo com o modelo de decantação proposto por Takács et al. (1991). Em cada camada, são aplicados balanços de massa para garantir a precisão do modelo.

O modelo ASM1 é amplamente utilizado em estações de tratamento de águas residuais e foi escolhido para o BSM por ser relativamente simples e apresentar um bom desempenho na previsão dos processos biológicos envolvidos no tratamento de águas residuais. Já o modelo de decantação proposto por Takács et al. (1991) é utilizado para simular o processo de separação de sólidos e líquidos no decantador secundário.

Em conjunto, esses modelos permitem simular e avaliar o desempenho de diferentes estratégias de controle em estações de tratamento de águas residuais, identificando as melhores condições de operação e gestão do processo. A aplicação desses modelos pode levar a melhorias significativas no desempenho do processo de tratamento de águas residuais, resultando em uma redução nos custos operacionais e uma melhoria na qualidade do efluente tratado.

O BSM1 é frequentemente utilizado como modelo de referência para avaliar e comparar diferentes estratégias de controle em estações de tratamento de águas residuais. No entanto, como é um modelo relativamente complexo, requer dados precisos sobre as características das águas residuais e as condições de operação da estação de tratamento. A simulação do BSM1 pode ajudar a estabelecer as condições ideais de operação e gestão das estações de tratamento de águas residuais, bem como identificar as melhores estratégias de controle para otimizar o desempenho do processo (Rodrigues, 2013).

### **3.4.1 ASM1 (*Activated Sludge Model N° 1*)**

ASM1 (Modelo de Lodo Ativado No. 1) é um modelo matemático usado para simular e projetar estações de tratamento de águas residuais usando o processo de lodo ativado. Foi desenvolvido em 1987 por G. Gujer e M. Henze e é amplamente utilizado em todo o mundo para projetar e otimizar estações de tratamento de águas residuais.

Este modelo é baseado na ecologia microbiana das águas residuais e simula a interação de diferentes grupos de microrganismos, incluindo bactérias, protozoários e microrganismos nitrificantes e desnitrificantes. O modelo é composto por uma série de equações diferenciais que descrevem a dinâmica de crescimento dos microrganismos e a transformação dos contaminantes presentes nas águas residuais.

O modelo ASM1 é uma ferramenta útil para prever o desempenho de uma estação de tratamento de águas residuais em diferentes situações e pode ajudar os engenheiros a otimizar a operação da estação e reduzir os custos. Também é usado para avaliar o impacto ambiental das descargas de águas residuais e projetar sistemas de controle de poluição (Gernaey et al., 2004).

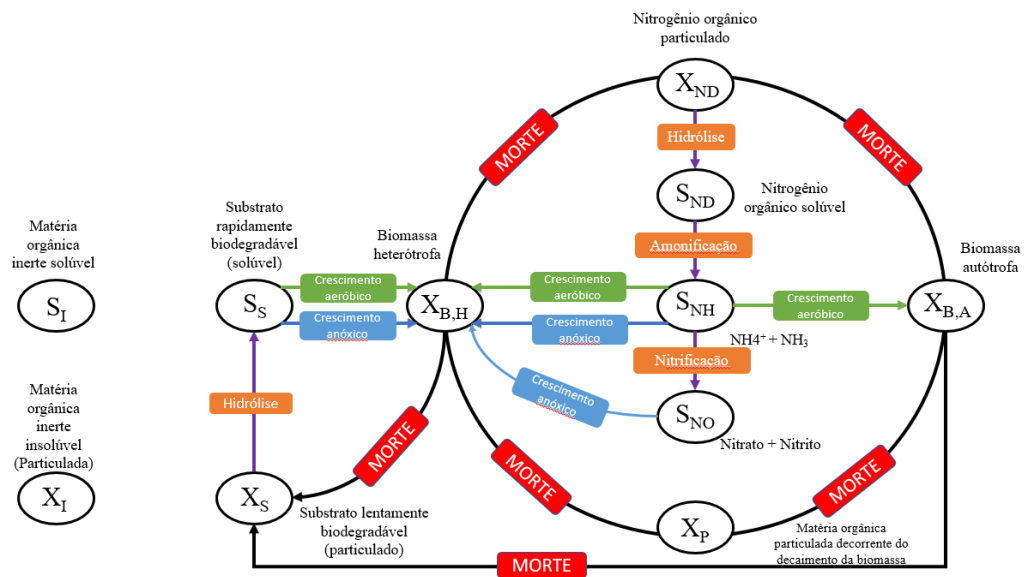
O ASM1 segmenta os compostos orgânicos presentes no efluente em frações com base na sua solubilidade em água e capacidade de biodegradação. Existem dois tipos de substratos: substrato lentamente biodegradável e substrato rapidamente biodegradável, onde o substrato lentamente biodegradável, que representa substâncias solúveis mais complexas que sofrem hidrólise extracelularmente, é definido como sólido e se transforma em substrato rapidamente biodegradável. O ASM1 também considera a existência de substâncias sólidas e solúveis não biodegradáveis, que são removidas do sistema com a massa de lodo (Jeppsson, 1997; Rodrigues, 2013).

Em relação as substâncias nitrogenadas, o ASM1 realiza seu fracionamento de acordo com sua biodegradabilidade e solubilidade em água. O modelo considera que a nitrificação, que é realizada por bactérias autotróficas, ocorre em uma única etapa, onde o  $NO_3$  é a única forma oxidada de nitrogênio. Porém, o nitrogênio na forma de gás ( $N_2$ ) liberado durante a desnitrificação não é representado no modelo, resultando em um balanço de massa para nitrogênio que não é completamente satisfeito (Jeppsson, 1997; Rodrigues, 2013).

A biomassa heterotrófica cresce à medida que consome o substrato de biodegradabilidade rápida, seja em condições aeróbicas ou anóxicas. Já o decaimento da biomassa produz matéria orgânica que pode ser de natureza não biodegradável ou de biodegradabilidade lenta (Levy, 2007).

O modelo ASM1 é composto por quatro conceitos fundamentais que descrevem os oito processos estabelecidos nele: crescimento celular, decomposição ou lise celular, amonificação do nitrogênio orgânico e hidrólise dos produtos particulados, conforme demonstrado na Figura 3. Esses processos são provenientes da dinâmica de duas populações bacterianas, as heterotróficas e as autotróficas, onde a cinética de crescimento celular é descrita pelo modelo cinético proposto por Monod, que relaciona a taxa de crescimento com a concentração do substrato disponível e um coeficiente de afinidade pelo substrato, enquanto a atividade de amonificação e a cinética de morte microbiana são representadas por modelos cinéticos de primeira ordem (Zoby, 2014; Jeppsson et al., 2011).

**Figura 3 -** Ciclo reacional no sistema de lodo ativado.



Fonte: Adaptado de Jeppsson et al., 2011.

### 3.4.2 Processos Biológicos

A seguir estão listados os 8 processos, Tabela 1, presentes no modelo ASM1, os quais, segundo Henze et al. (2002), são fundamentados no crescimento e morte das populações bacterianas heterotróficas e autotróficas, na hidrólise dos substratos complexos e na amonificação do nitrogênio orgânico solúvel.

**Tabela 1** - Processos biológicos básicos do BSM1-ASM1.

PROCESSO	REAÇÃO BÁSICA
Crescimento aeróbico de heterótrofos	$S_S + S_O + S_{NH} \rightarrow X_{BH}$
Crescimento anóxico de heterótrofos	$S_S + S_{NO} + S_{NH} \rightarrow X_{BH}$
Crescimento aeróbico de autótrofos	$S_O + S_{NH} \rightarrow X_{BA} + S_{NO}$
Decaimento de heterótrofos	$X_{BH} \rightarrow X_P + X_S + X_{ND}$
Decaimento de autótrofos	$X_{BA} \rightarrow X_P + X_S + X_{ND}$
Amonificação de nitrogênio solúvel	$S_{ND} \rightarrow S_{NH}$
Hidrólise de orgânicos encapsulados	$X_S \rightarrow S_S$
Hidrólise de nitrogênio encapsulado	$X_{ND} \rightarrow S_{ND}$

Fonte: Adaptado de Jeppsson et al., 2011.

- Crescimento aeróbico de heterótrofos:** Este processo é responsável pelo aumento da biomassa heterotrófica sob condições de aeração adequada. Durante o processo de crescimento aeróbico de heterótrofos, os micro-organismos utilizam o substrato orgânico rapidamente biodegradável como fonte de energia e nutrientes para aumentar sua biomassa, desde que haja aeração adequada. Ao longo desse processo, a amônia presente na solução é retirada e incorporada na massa celular. A taxa de crescimento celular é limitada pela concentração de substrato e oxigênio dissolvido, que de acordo com Henze et al. (1987), é descrita pelo modelo cinético de Monod, que relaciona a taxa de crescimento com a concentração do substrato disponível e um coeficiente de afinidade pelo substrato. Esse efeito é representado por uma função de saturação, conforme a Equação (1):

$$\rho_1 = \mu_{HT} \left( \frac{S_S}{K_S + S_S} \right) \left( \frac{S_O}{K_{O,H} + S_O} \right) X_{B,H} \quad (1)$$

Onde:  $\mu_{HT}$  é a taxa de crescimento dos heterotróficos,  $S_s$  é o substrato biodegradável,  $S_O$  é o oxigênio dissolvido,  $K_S$  é a constante de semi-saturação dos heterotróficos para o substrato,  $K_{O,H}$  é a constante de semi-saturação dos heterotróficos para o oxigênio dissolvido e  $X_{B,H}$  é a biomassa heterotrófica de ativação e desligamento.

- **Crescimento anóxico de heterótrofos:** De acordo com Henze et al. (1987), o crescimento anóxico refere-se à capacidade dos heterótrofos de se multiplicarem em ambientes com baixos níveis de oxigênio dissolvido. Durante este processo de crescimento anóxico, os heterótrofos utilizam diferentes fontes de carbono orgânico disponíveis, como ácidos graxos voláteis, polissacarídeos e outros compostos orgânicos solúveis, onde esses compostos são assimilados e metabolizados pelos heterótrofos para a produção de energia e biomassa.

Nesse contexto, a cinética de Monod é aplicada, mas com um fator de efetividade reduzido, abaixo de um. Isso ocorre devido a uma parcela menor dos heterótrofos ser capaz de se multiplicar em ambientes anóxicos, o que afeta a taxa máxima de crescimento. Esta menor efetividade no crescimento anóxico é atribuída à falta de oxigênio molecular disponível, onde em ambientes anóxicos, os heterótrofos recorrem à respiração anaeróbica para obter energia a partir dos substratos biodegradáveis.

Assim, a cinética de Monod, que descreve a taxa de crescimento dos heterótrofos em função da concentração de substrato, é ajustada para levar em consideração essa menor parcela de heterótrofos capazes de se multiplicar anoxicamente. O fator de efetividade menor que um reflete essa limitação e reduz a taxa máxima de crescimento observada nos heterótrofos anóxicos. Esse efeito pode ser descrito de acordo com a Equação (2):

$$\rho_2 = \mu_{HT} \left( \frac{S_s}{K_S + S_s} \right) \left( \frac{K_{O,H}}{K_{O,H} + S_O} \right) \left( \frac{S_{NO}}{K_{NO} + S_{NO}} \right) \eta_H X_{B,H} \quad (2)$$

Onde:  $S_{NO}$  é o nitrogênio na forma de nitrato e nitrito,  $K_{NO}$  é a constante de semi-saturação dos heterotróficos para os nitratos e  $\eta_H$  é o fator de correção para crescimentos dos heterótrofos em anoxia.

- **Crescimento aeróbico de autótrofos:** O processo de crescimento aeróbico de autótrofos refere-se à capacidade desses microrganismos de se multiplicarem em ambientes

ricos em oxigênio dissolvido. Segundo Henze et al. (1987), durante o crescimento aeróbico, os autótrofos utilizam a energia obtida através da fotossíntese para fixar o CO<sub>2</sub> e produzir compostos orgânicos, como carboidratos. Esses compostos orgânicos servem como fonte de carbono e energia para o seu crescimento e metabolismo.

Jeppsson et al. (1996) destacam a importância dos nutrientes para o crescimento aeróbico dos autótrofos, que necessitam de nutrientes como nitrogênio, fósforo e micronutrientes para o seu crescimento e metabolismo. Esses nutrientes são essenciais para a síntese de proteínas, ácidos nucleicos e outras moléculas importantes para as funções celulares.

O crescimento aeróbico dos autótrofos desempenha um papel fundamental na produção de biomassa, na produção de oxigênio para o meio ambiente e no equilíbrio dos ecossistemas aquáticos. Este processo resulta na formação de massa celular autotrófica e na conversão do nitrogênio em nitrato, onde a reação de oxidação da amônia em nitrato é acompanhada da presença de oxigênio, cuja proporção é considerada na equação da reação, também descrita pelo modelo de Monod, como demonstrado na Equação 3:

$$\rho_3 = \mu_{AT} \left( \frac{S_{NH}}{K_{NH} + S_{NH}} \right) \left( \frac{S_O}{K_{O,A} + S_O} \right) X_{B,A} \quad (3)$$

Onde:  $\mu_{AT}$  é a taxa de crescimento dos autótrofos,  $S_{NH}$  é o nitrogênio na forma de nitrato e nitrito,  $K_{NH}$  é a constante de semi-saturação dos autótrofos para o nitrogênio amoniacal,  $K_{O,A}$  é a constante de semi-saturação dos autótrofos para o oxigênio e  $X_{B,A}$  é a biomassa autotrófica ativa.

- **Decaimento de heterótrofos:** Conforme abordado por Henze et al. (1987), o decaimento de heterótrofos é modelado pela teoria de morte-regeneração e, ocorre devido a morte natural das células, a falta de substrato adequado para o seu crescimento e a ação de predadores ou parasitas. Outros fatores como as condições ambientais, a disponibilidade de nutrientes e a temperatura, também influenciam o processo de decaimento.

Nessa teoria, os organismos morrem a uma taxa específica, e parte do material resultante é considerada não biodegradável, sendo adicionada à concentração de particulados que se aglomeram devido à morte microbiana. A outra parte do material proveniente do decaimento se junta à concentração de substratos degradáveis mais lentamente.

De acordo com Jeppsson et al. (1996), é considerado constante a taxa de decaimento para seres aeróbicos, anóxicos e anaeróbicos, caracterizado pela degradação da biomassa e a

consequente liberação de nutrientes no meio, sugerindo assim, que a taxa de decaimento não varia com as condições de oxigênio disponíveis no meio. Dessa forma, a expressão matemática que descreve o processo de decaimento de heterótrofos pode ser formulada da seguinte maneira:

$$\rho_4 = b_{HT}X_{B,H} \quad (4)$$

Onde,

$b_{HT}$ : Constante de decaimento que caracteriza a taxa de mortalidade dos organismos heterótrofos;

$X_{B,H}$ : Concentração de biomassa heterotrófica.

O decaimento de heterótrofos é uma parte natural do ciclo de vida dos microrganismos em sistemas biológicos, como reatores de tratamento de águas residuais. É um processo contínuo e necessário para manter o equilíbrio entre o crescimento e a morte celular.

- **Decaimento de autótrofos:** O processo de decaimento de autótrofos, assim como o de heterótrofos, envolve a perda gradual de biomassa e atividade metabólica dos organismos ao longo do tempo. Embora existam diferenças metabólicas entre ambos, os grupos de microrganismos estão sujeitos a mecanismos de morte e decaimento semelhantes.

Dessa forma, a expressão matemática que descreve o processo de decaimento de autótrofos pode ser formulada da seguinte maneira:

$$\rho_5 = b_{AT}X_{B,A} \quad (5)$$

Onde,

$b_{AT}$ : Constante de decaimento, que caracteriza a taxa de mortalidade dos organismos autótrofos;

$X_{B,A}$ : Concentração de biomassa autotrófica.

- **Amonificação de nitrogênio solúvel:** De acordo com Henze et al. (1987), a amonificação ocorre como resultado da atividade de microrganismos heterotróficos no sistema de tratamento de águas residuais. Nesse processo os microrganismos decompõem a matéria orgânica nitrogenada solúvel, liberando compostos nitrogenados na forma de amônia, também conhecido como mineralização ou hidrólise nitrogenada.

Jeppsson et al. (1996) destaca que a taxa de amonificação é influenciada por fatores como a concentração de nitrogênio orgânico solúvel, a atividade dos microrganismos

responsáveis pela hidrólise e as condições ambientais, como pH, temperatura e disponibilidade de oxigênio.

A oxidação do nitrogênio orgânico solúvel ( $S_{ND}$ ) para nitrogênio amoniacal é descrita por um modelo cinético de primeira ordem em relação à biomassa heterotrófica ( $X_{B,H}$ ), independentemente das condições de aeração do meio. Logo, a taxa de oxidação permanece constante, independentemente da quantidade de oxigênio disponível no sistema, demonstrado na Equação (6):

$$\rho_6 = k_{aT} S_{ND} X_{B,H} \quad (6)$$

Onde,

$k_{aT}$ : Constante de velocidade de hidrólise de amonificação pelos heterótrofos;

$S_{ND}$ : Nitrogênio orgânico solúvel biodegradável.

- **Hidrólise de orgânicos encapsulados:** Segundo Henze et al. (1987) e Jeppsson et al. (1996), os compostos orgânicos encapsulados podem ser encontrados em diferentes formas, como partículas orgânicas, coloides ou matéria orgânica dissolvida. Esses compostos podem ser degradados através de reações de hidrólise, onde as moléculas orgânicas são quebradas por enzimas ou por ação de microrganismos extracelulares. Essa hidrólise permite a liberação de nutrientes e substratos biodegradáveis, tornando-os disponíveis para sua utilização como fonte de energia e nutrientes para os microrganismos heterotróficos presentes no sistema de tratamento de águas residuais.

Na conversão do substrato lentamente biodegradável ( $X_S$ ) em substrato rapidamente biodegradável ( $S_S$ ), ocorre a transformação de uma forma menos disponível de carbono em uma forma mais facilmente utilizável pelos microrganismos. De acordo com Henze et al. (1987) e Jeppsson et al. (1996), a conversão do  $X_S$  em  $S_S$  ocorre através de uma cinética de reação de adsorção. Nessa cinética, o  $X_S$  é adsorvido por microrganismos presentes no sistema, sendo preferencialmente utilizado em condições aeróbicas e anóxicas.

Em condições aeróbicas, o oxigênio está disponível para a respiração dos microrganismos, permitindo um aumento da taxa de adsorção e consequente conversão do  $X_S$  em  $S_S$ . Já em condições anóxicas, falta o oxigênio para a respiração, mas outros acceptor de elétrons, como o nitrato, pode estar presente, o que também permite a adsorção e conversão do  $X_S$  em  $S_S$ .



Dessa forma, a expressão matemática que descreve o processo de hidrólise de orgânicos encapsulados pode ser formulada da seguinte maneira:

$$\rho_7 = \frac{k_{hT} \left( \frac{X_S}{X_{B,H}} \right)}{K_X + \left( \frac{X_S}{X_{B,H}} \right)} \left[ \left( \frac{K_{O,H}}{K_{O,H} + S_O} \right) \left( \frac{S_{NO}}{K_{NO} + S_{NO}} \right) \eta_H + \left( \frac{S_O}{K_{O,H} + S_O} \right) \right] X_{B,H} \quad (7)$$

Onde,

$k_{hT}$ : Constante de velocidade de hidrólise de anoxia;

$X_S$ : Substrato lentamente biodegradável;

$K_X$ : Constante de semi-saturação dos heterotróficos para hidrólise;

$\eta_K$ : Fator de correção para a taxa de hidrólise em anoxia.

- **Hidrólise de nitrogênio encapsulado:** O processo de hidrólise de nitrogênio encapsulado envolve a quebra de compostos nitrogenados complexos em formas mais simples, como aminoácidos, por meio de reações de hidrólise. Essa etapa é fundamental para a disponibilização de nitrogênio amoniacal para os microrganismos e para a continuidade dos processos biológicos no tratamento de águas residuais, sua definição é análoga a hidrólise de orgânicos encapsulados. Logo, a expressão matemática que descreve o processo de hidrólise de nitrogênios encapsulados pode ser formulada da seguinte maneira:

$$\rho_8 = \frac{k_{hT} \left( \frac{X_S}{X_{B,H}} \right)}{K_X + \left( \frac{X_S}{X_{B,H}} \right)} \left[ \left( \frac{K_{O,H}}{K_{O,H} + S_O} \right) \left( \frac{S_{NO}}{K_{NO} + S_{NO}} \right) \eta_H + \left( \frac{S_O}{K_{O,H} + S_O} \right) \right] X_{B,H} \left( \frac{X_{ND}}{X_S} \right) \quad (7)$$

Onde,

$X_{ND}$ : são as partículas de nitrogênio orgânico biodegradável.

### 3.4.3 Variáveis de Estado do Modelo ASM1

A seguir estão listadas as 13 variáveis de estados, Tabela 2, presentes no modelo ASM1, as quais, sete delas são solúveis e seis delas particuladas.

**Tabela 2 - Variáveis de Estado do Modelo ASM1.**

Variável de Estado	Notação
Matéria orgânica solúvel inerte	$S_I$
Substrato facilmente biodegradável	$S_S$
Matéria orgânica particulada inerte	$X_I$
Substrato lentamente biodegradável	$X_S$
Biomassa heterotrófica ativa	$X_{BH}$
Biomassa autotrófica ativa	$X_{BA}$
Produtos particulados resultantes do decaimento da biomassa	$X_P$
Concentração de oxigênio dissolvido no reator	$S_O$
Nitrogênio na forma de nitrato	$S_{NO}$
Nitrogênio na forma de $NH_4^+ + NH_3$	$S_{NH}$
Nitrogênio orgânico solúvel biodegradável	$S_{ND}$
Nitrogênio orgânico particulado biodegradável	$X_{ND}$
Alcalinidade	$S_{ALK}$

Fonte: Adaptado de Henze et al., 1987.

### 3.4.4 Parâmetros Estequiométricos

A seguir estão listados os valores para os coeficientes estequiométricos empregados no modelo ASM1, como demonstrados na Tabela 3.

**Tabela 3 - Parâmetros estequiométricos do modelo ASM1.**

Parâmetros	Unidade	Valor
$Y_A$	g celular formada DQO. (g N oxidado) <sup>-1</sup>	0,24
$Y_H$	g celular formada DQO. (g DQO oxidado) <sup>-1</sup>	0,67
$f_p$	Adimensional	0,08
$i_{XB}$	g N. (g DQO) <sup>-1</sup> em biomassa	0,08
$i_{XP}$	g N. (g DQO) <sup>-1</sup> em produtos particulados	0,06

Fonte: Adaptado de Henze et al., 2000.

Onde:  $Y_H$  é o rendimento heterotrófico,  $Y_A$  é o rendimento autotrófico,  $f_p$  é a fração de biomassa que resulta em produtos particulados,  $i_{XB}$  é a massa N/massa DQO na biomassa e  $i_{XP}$  é a massa N/massa em DQO de produto de biomassa.

### 3.4.5 Parâmetros Cinéticos

A seguir estão listados os parâmetros cinéticos empregados no modelo ASM1, como demonstrados na Tabela 4.

**Tabela 4** - Parâmetros cinéticos do modelo ASM1.

Parâmetros	Unidade	Valor
$\mu_H$	$d^{-1}$	4,0
$K_S$	$gDQO.m^{-3}$	10,0
$K_{O,H}$	$g(-DQO).m^{-3}$	0,2
$K_{NO}$	$g NO_3-N.m^{-3}$	0,5
$b_H$	$d^{-1}$	0,3
$\eta_g$	Adimensional	0,8
$\eta_h$	Adimensional	0,8
$k_H$	$g \text{ de DQO lentamente biodegradável. (g celular de DQO. d)}^{-1}$	3,0
$K_x$	$g \text{ de DQO lentamente biodegradável. (g celular de DQO)}^{-1}$	0,1
$\mu_A$	$d^{-1}$	0,5
$K_{NH}$	$g NH_3-N.m^{-3}$	1,0
$b_A$	$d^{-1}$	0,05
$K_{O,A}$	$g (-DQO).m^{-3}$	0,4
$k_A$	$m^3. (g DQO.d)^{-1}$	0,05
$\mu_H$	$d^{-1}$	4,0

Fonte: Adaptado de Henze et al., 2000.

Onde:  $\mu_H$  é a taxa máxima de crescimento específico da biomassa heterotrófica,  $b_H$  é a taxa de decaimento da biomassa heterotrófica,  $K_S$  é o coeficiente de saturação para organismos heterotróficos,  $K_{O,H}$  é o coeficiente de saturação de oxigênio para organismos heterotróficos,

$K_{NO}$  é o coeficiente de saturação de nitrato para heterotróficos,  $\mu_A$  é a taxa máxima de crescimento específico da biomassa autotrófica,  $b_A$  é a taxa de decaimento da biomassa autotrófica,  $K_{O,A}$  é o coeficiente de saturação de oxigênio para organismos autotróficos,  $K_{NH}$  é o coeficiente de saturação de amônia para organismos autotróficos,  $k_a$  é a taxa específica de amonificação,  $k_h$  é a máxima taxa específica de hidrólise,  $K_X$  é o coeficiente de saturação para substrato lentamente biodegradável,  $\eta_g$  é o fator de correção para  $\mu_H$  sob condições anóxicas e  $\eta_h$  é o fator de correção para hidrólise sob condições anóxicas.

### 3.4.6 Restrições e limitações do modelo ASM1

O modelo ASM1 apresenta algumas restrições e limitações que devem ser consideradas ao utilizá-lo para a modelagem de processos de tratamento de águas residuais. De acordo com Henze et al. (1987) e Jeppsson et al. (2011), algumas das restrições do modelo ASM1 são:

- O modelo pressupõe que o sistema opera à temperatura constante, sem considerar variações significativas de temperatura ao longo do tempo. Caso ocorram flutuações de temperatura, é necessário implementar uma correção utilizando um modelo de Arrhenius ou similar;
- O pH é considerado constante e próximo da neutralidade. Embora o pH possua uma influência significativa em vários parâmetros do sistema, o modelo não leva em conta as variações de pH durante o processo;
- Assume-se que os coeficientes nas taxas de reação são constantes e não considera alterações relacionadas à natureza da matéria orgânica de cada fração. Isso pode limitar a precisão das previsões em situações em que a composição da matéria orgânica varia significativamente;
- As limitações de nutrientes inorgânicos, como nitrogênio, fósforo e outros, não são consideradas nas taxas de remoção de substrato orgânico e no crescimento das células. A presença adequada de nutrientes é fundamental para permitir um crescimento balanceado da biomassa e a falta de consideração dessas limitações pode afetar a sedimentação do lodo;
- O modelo ASM1 não inclui expressões cinéticas que lidem com as limitações dos organismos heterotróficos em relação ao nitrogênio e à alcalinidade. Isso pode resultar em previsões menos precisas em sistemas onde essas limitações são relevantes;

- O ASM1 considera nitrogênio solúvel biodegradável e nitrogênio orgânico particulado como componentes, o que pode dificultar a medição dessas variáveis e tornar o uso do modelo mais complexo do que o necessário;
- O modelo não leva em conta as cinéticas de amonificação, que não são facilmente quantificadas e podem ter um impacto significativo nas previsões da modelagem;
- O ASM1 assume que as hidrólises de material orgânico e nitrogênio orgânico ocorrem simultaneamente e em taxas iguais;
- O modelo não diferencia a origem do material particulado orgânico inerte, seja ele proveniente do influente ou do decaimento da biomassa. Na prática, é difícil diferenciar essas frações.

Essas restrições e limitações do modelo ASM1 são importantes e devem ser levadas em consideração ao aplicar o modelo na modelagem de processos.

### 3.4.7 Balanço de massa do biorreator

O reator biológico do BSM1 é composto por cinco compartimentos perfeitamente agitados, onde os dois primeiros são mantidos em condições de anoxia, não-aerados, e os três últimos em condições de aerobiose. Os compartimentos anóxicos têm um volume de 1000 m<sup>3</sup> cada, enquanto os compartimentos aerados possuem um volume de 1333 m<sup>3</sup> cada.

Os balanços de massa por componente em termos de concentração são descritos individualmente para cada compartimento, onde:

Para o primeiro compartimento (k = 1), temos que:

$$\frac{dZ_1}{dt} = \frac{1}{V_1} (Q_a Z_a + Q_r Z_r + Q_0 Z_0 + r_1 V_1 - Q_1 Z_1) \quad (8)$$

$$Q_1 = Q_a + Q_r + Q_0 \quad (9)$$

Para os compartimentos de 2 a 5 (k = 2 a 5), temos que:

$$\frac{dZ_k}{dt} = \frac{1}{V_k} (Q_{k-1} Z_{k-1} + r_k V_k - Q_k Z_k) \quad (10)$$

$$Q_k = Q_{k-1} \quad (11)$$

Para o oxigênio (S<sub>O,k</sub>):

$$\frac{dS_{O,k}}{dt} = \frac{1}{V_k} (Q_{k-1} S_{O,k-1} + r_k V_k + (K_{la})_k V_k (S_O^* - S_{O,k}) - Q_k S_{O,k}) \quad (12)$$

Onde,

$Z_k$ : Concentração de biomassa;

$Q_k$ : Vazão de entrada e saída do reator;

$r_k$ : Taxa de reação;

$V_k$ : Volume do reator;

$S^*$ : Concentração de saturação do oxigênio =  $8\text{gm}^3$ .

Podemos também considerar as seguintes igualdades:

$$Z_a = Z_5 \quad (13)$$

$$Z_f = Z_5 \quad (14)$$

$$Z_w = Z_r \quad (15)$$

$$Q_f = Q_5 - Q_a = Q_e + Q_r + Q_w = Q_e + Q_u \quad (16)$$

Essas igualdades relacionam as concentrações e vazões dos compartimentos específicos do biorreator, fornecendo informações importantes para o balanço global do sistema.

### 3.4.8 Modelagem do sedimentador

Os sedimentadores desempenham um papel fundamental nos sistemas de lodo ativado do modelo BSM1, pois através desta operação de sedimentação, ocorre a remoção da biomassa gerada no setor reacional do sistema, possibilitando o descarte da água. Para garantir a eficiência desse processo, é necessário dimensionar o sedimentador de forma adequada (Mulas, 2006).

No modelo BSM1 proposto por Jeppsson et al. (2011), o sedimentador é projetado de forma unidimensional, considerando perfis de velocidade horizontais uniformes e gradientes de concentração desprezíveis (Mulas, 2006). O sedimentador é dividido em 10 camadas com espessuras homogêneas, nas quais não ocorrem reações químicas. Com uma área de  $1500\text{ m}^2$ , altura de  $0,4\text{ m}$  em cada camada, resultando em uma altura total de  $4\text{ m}$  para o sedimentador e um volume de  $6000\text{ m}^3$ .

A correta operação do sedimentador é essencial para garantir a eficiência do sistema de lodo ativado, onde, o dimensionamento adequado levando em consideração fatores como a

vazão de recirculação de lodo e a vazão de água do sistema, permite a separação eficiente da biomassa e do líquido clarificado. Além disso, o sedimentador executa simultaneamente funções de clarificação, no topo, e de espessamento, na base, contribuindo para a concentração dos sólidos no fundo do sedimentador.

Ao remover parte da biomassa gerada no sistema de lodo ativado, o sedimentador desempenha um papel fundamental na manutenção do desempenho do sistema como um todo. Em que a biomassa separada no sedimentador pode ser direcionada de volta aos biorreatores de lodo ativado, garantindo a continuidade do processo de tratamento.

Considerando-se o fluxo de sólidos total no sedimentador, este pode ser considerado como resultado do fluxo devido à gravidade ( $J_S$ ) e devido a recirculação da massa ( $J_B$ ). Como explicado por Mulas (2006), o fluxo de recirculação da massa de água ( $J_B$ ) se dá devido as bombas de recirculação de lama que ficam embaixo do sedimentador, e o fluxo devido à gravidade faz com que os sólidos se depositem na base do deste. Logo:

No balanço de massa do sedimentador, a velocidade de sedimentação no decantador secundário é influenciada pela concentração da biomassa, onde o fluxo total de sólidos é composto pelo fluxo gravitacional ( $J_S$ ), e pelo fluxo de recirculação da massa ( $J_B$ ), resultante da saída do efluente clarificado ou da extração de lodo do fundo do sedimentador (Mulas, 2006). Dessa forma, o fluxo total de sólidos ( $J_T$ ), que representa a produção de sólidos por unidade de área e tempo, é dado pela Equação (17):

$$J_T = J_B + J_S = v_s X + UX \quad (17)$$

Onde,

$v_s$ : Velocidade de sedimentação (m/s);

U: Velocidade de recirculação da biomassa devido ao fluxo de líquido (m/s);

X: Concentração das lamas (mg/L).

O modelo de sedimentador proposto por Jeppsson et al. (2011) e por Mulas (2006), Figura 4, é fundamentado na modelagem anteriormente proposta por Vitasovic (1986), que estabelece a realização do balanço de massa para o sedimentador em cinco camadas diferentes: topo, base, alimentação e abaixo da alimentação.

Para simplificar o modelo, algumas suposições são feitas, como não considerar as reações biológicas no tanque, resultando em uma concentração constante de matéria dissolvida

em todas as camadas, apenas o fluxo vertical ascendente e descendente é considerado, e os sólidos são distribuídos uniformemente ao longo de toda a área transversal da camada de alimentação. O modelo também utiliza apenas uma variável de estado para os componentes particulados.

As velocidades ascendente e descendente do fluxo de sólidos são definidas como:

$$v_{descendente} = \frac{Q_u}{A} = \frac{Q_r + Q_w}{A} \quad (18)$$

$$v_{ascendente} = \frac{Q_e}{A} \quad (19)$$

Onde,

A: Área de seção transversal do decantador;

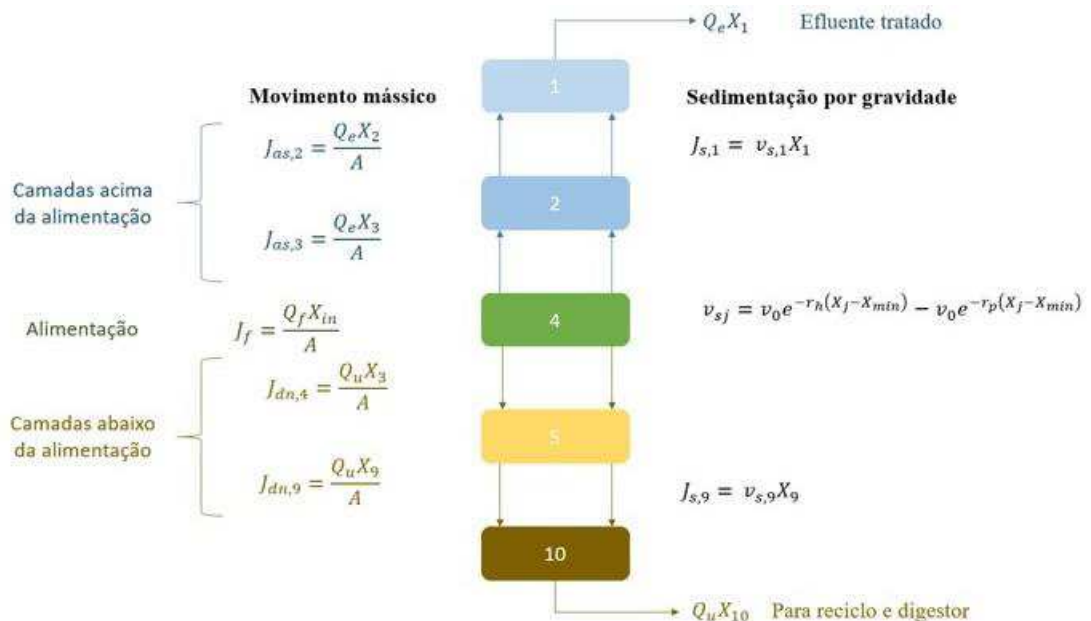
$Q_u$ : Vazão inferior;

$Q_r$ : Vazão reciclada;

$Q_w$ : Vazão desperdiçada;

$Q_e$ : Vazão do efluente.

**Figura 4** - Modelo unidimensional do sedimentador por camadas.



Fonte: Adaptado de Mulas, 2006.

No modelo BSM1, o fluxo de sólidos devido à sedimentação gravitacional é determinado pelo produto da concentração da camada correspondente no sedimentador pela velocidade de sedimentação dos sólidos. O modelo utilizado no BSM1 é o modelo de velocidade de sedimentação exponencial dupla, proposto por Takács et al. (1991).



A expressão que representa a velocidade de sedimentação ( $v_{sj}$ ) no modelo BSM1 é dada por:

$$v_{sj} = v_0 e^{-r_h(X_j - X_{min})} - v_0 e^{-r_p(X_j - X_{min})} \quad (20)$$

$$0 \leq v_{sj} \leq v_0' \quad (21)$$

Onde,

$v_0$ : Velocidade máxima de sedimentação;

$r_h$  e  $r_p$ : Constantes relacionadas a taxa de sedimentação;

$X_j$ : Concentração de sólidos na camada  $j$ ;

$X_{min}$ : Concentração mínima de sólidos para sedimentação.

A seguir estão listados os parâmetros empregados no modelo do sedimentador, como demonstrados na Tabela 5.

**Tabela 5** - Parâmetros do modelo do sedimentador.

Descrição	Parâmetro	Unidade	Valor
<b>Velocidade máxima de sedimentação (prática)</b>	$v_0'$	$\frac{m}{d}$	250
<b>Velocidade máxima de sedimentação (teórica/Vesilind)</b>	$v_0$	$\frac{m}{d}$	474
<b>Parâmetro de sedimentação retardada</b>	$r_h$	$\frac{m^3}{g\ SS}$	0,000576
<b>Parâmetro de sedimentação flocular</b>	$r_p$	$\frac{m^3}{g\ SS}$	0,00286
<b>Fração de não-sedimentáveis</b>	$f_{ns}$	Adimensional	0,00228

Fonte: Adaptado de Jeppsson et al., 2011.

### 3.5 Modelos Substitutos e Metamodelos

A utilização de metamodelos e modelos substitutos tem se tornado cada vez mais relevante em diversas áreas, oferecendo soluções eficientes para problemas que envolvem a utilização de modelos complexos ou de alto custo computacional. Essas abordagens foram desenvolvidas ao longo dos anos com o objetivo de simplificar processos e fenômenos, cujas descrições matemáticas são de alta complexidade, tornando o uso de modelos reduzidos uma alternativa viável e vantajosa.

De acordo com Gomes (2007), os modelos reduzidos são obtidos por meio de aproximações ou simplificações, permitindo que aplicações que demandam esforço computacional, tempo e confiabilidade possam ser desenvolvidas de maneira mais eficiente. Em muitos casos, o modelo rigoroso apresenta limitações, como problemas de convergência em procedimentos iterativos, tornando o uso de metamodelos de maior interesse (Palmer e Realff, 2002a).

A obtenção dos metamodelos envolve a coleta de dados provenientes do modelo rigoroso, por meio de um processo conhecido como planejamento experimental. Esse planejamento experimental desempenha um papel fundamental na otimização do modelo substituto e na simulação do processo, permitindo a identificação dos pontos relevantes no espaço de projeto (Viana et al., 2010).

Uma metodologia amplamente empregada na geração de metamodelos é a DACE (*Design and Analysis of Computer Experiments*), descrita por Sacks et al. (1989). Essa metodologia compreende a escolha de uma estratégia de planejamento experimental para a geração de dados, a seleção de um modelo adequado para representar os dados e o ajuste desse modelo aos dados obtidos.

Dentre as técnicas comumente utilizadas na construção de metamodelos, destacam-se a Regressão Linear Múltipla e o Modelo de Kriging. A Regressão Linear Múltipla busca estabelecer uma relação matemática simples entre os dados de entrada e saída, enquanto o modelo de Kriging utiliza interpolação espacial para aproximar as relações entre os dados (Gomes, 2007).

A validação dos metamodelos é uma etapa crucial, na qual eles são avaliados com o objetivo de verificar a sua precisão e confiabilidade. Esse processo é realizado por meio de comparações com o modelo rigoroso, considerando diferentes variáveis de resposta (Palmer e Realff, 2002a). Com isso, essa validação permite obter respostas satisfatórias e aproximadas do modelo rigoroso, viabilizando a otimização de processos e a redução dos esforços computacionais e temporais necessários.

A geração de um metamodelo segue uma metodologia básica, composta por etapas distintas, essenciais para obtenção de um metamodelo confiável e consistem em:

1. Planejamento experimental: A primeira etapa envolve a seleção de um planejamento experimental adequado para gerar os dados de entrada, definindo quais combinações de valores das variáveis de entrada devem ser usadas para alimentar o modelo.

Existem diferentes tipos de planejamentos experimentais, como o planejamento fatorial, o planejamento de Taguchi, o planejamento de pontos centrais, entre outros;

2. Alimentação dos dados de entrada no modelo rigoroso: Nesta etapa, os dados de entrada gerados pelo planejamento experimental são utilizados para alimentar o modelo rigoroso, que representa computacionalmente o sistema real e é capaz de gerar as saídas desejadas. Ao executar o modelo rigoroso com os dados de entrada, obtemos as saídas correspondentes, que servirão como dados de saída para a construção do metamodelo;

3. Geração do metamodelo: Com os dados de entrada gerados pelo planejamento experimental e os dados de saída obtidos a partir do modelo rigoroso, é possível aplicar um algoritmo específico para gerar o metamodelo. Existem diferentes técnicas para construção desse metamodelos, como redes neurais, polinômios interpoladores e kriging. Cada técnica possui suas características e suposições subjacentes, sendo importante selecionar aquela mais adequada para o processo em estudo;

4. Validação do metamodelo: Após a geração do metamodelo, é fundamental realizar sua validação para avaliar sua eficácia e confiabilidade. Isso é feito alimentando o metamodelo com novos conjuntos de dados de entrada, que não foram utilizados em sua construção, e comparando as respostas obtidas pelo metamodelo com as respostas fornecidas pelo modelo rigoroso. Essa comparação é conhecida como validação cruzada e permite avaliar a capacidade do metamodelo de generalizar o comportamento do sistema em diferentes situações.

A metodologia descrita acima estabelece as etapas principais para a geração de um metamodelo confiável. Esses metamodelos têm aplicações em diversas áreas, como engenharia de processos, otimização de projetos, análise de riscos e previsão de comportamento de sistemas complexos. Através dessa abordagem, é possível obter uma representação simplificada e eficiente do sistema, reduzindo a necessidade de execução repetitiva do modelo rigoroso e acelerando a análise e otimização dos processos.

### **3.5.1 Regressão Linear Múltipla**

A regressão linear múltipla é uma técnica estatística utilizada para modelar a relação entre uma variável resposta e várias variáveis explicativas, amplamente aplicada em diversas áreas, como economia, ciências sociais, engenharia e ciências da saúde. Segundo Walpole

(2011), a regressão linear múltipla é uma extensão da regressão linear simples, permitindo a inclusão de múltiplas variáveis independentes no modelo.

Na regressão linear múltipla, o objetivo é encontrar uma função linear que relacione as variáveis independentes  $(x_1, x_2, \dots, x_n)$  com a variável dependente  $(y)$ . Essa função é representada por uma equação da forma:

$$y = \beta_0 + \beta_1 x_1 + \dots + \beta_n x_n + \varepsilon \quad (22)$$

Onde,

$\beta_n$ : Coeficiente de regressão que representa as inclinações da relação linear entre as variáveis;

$\varepsilon$ : Erro aleatório que captura as variações que não representadas pelo modelo.

A estimação dos coeficientes de regressão é realizada por meio do método dos mínimos quadrados, que busca minimizar a soma dos quadrados dos resíduos fazendo a diferença entre os valores observados e os valores previstos pelo modelo. Esse método permite encontrar os valores ótimos dos coeficientes que melhor ajustam os dados observados.

Walpole (2011) ressalta a importância da verificação das premissas do modelo de regressão linear múltipla, como a linearidade da relação entre as variáveis, a independência dos erros e a normalidade dos resíduos. Ademais, é importante considerar a presença de multicolinearidade, que ocorre quando há alta correlação entre as variáveis independentes, o que pode afetar a interpretação dos coeficientes de regressão.

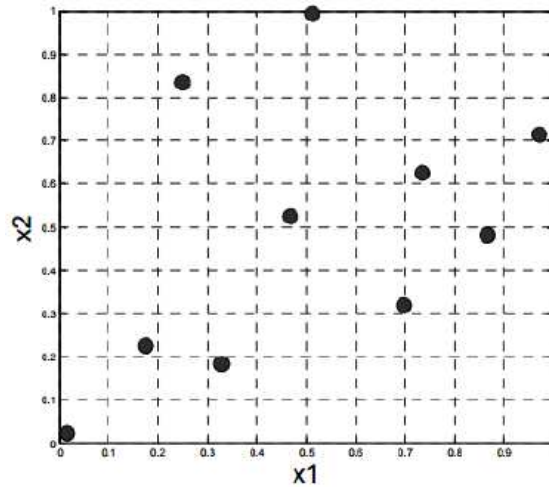
A regressão linear múltipla é uma poderosa técnica estatística, em que sua correta aplicação e interpretação dessa técnica requer atenção às premissas e considerações específicas do modelo, garantindo resultados confiáveis e relevantes, podendo ser aplicado em diferentes áreas. Na engenharia, uma das suas aplicações tem como intuito modelar a relação entre variáveis de processo e variáveis de saída em um sistema industrial.

### 3.5.2 Latin Hypercube Sampling

O *Latin Hypercube Sampling* (LHS), também conhecido como Amostragem de Hipercubo Latino, é uma técnica de amostragem estatística proposta por McKay et al. (1979), que visa maximizar a eficiência e a representatividade da amostra em relação a um espaço multidimensional de entrada.

O LHS é uma extensão da amostragem aleatória simples, que tem como objetivo garantir que a amostra abranja de forma adequada todo o espaço de entrada, evitando a concentração de pontos em regiões específicas. Isso é alcançado por meio da divisão do espaço de entrada em subintervalos igualmente espaçados em cada dimensão e selecionando um único ponto em cada subintervalo (Santos, 2014; Villar, 2016). Dessa forma, o LHS garante uma cobertura mais uniforme do espaço multidimensional, aumentando a representatividade da amostra, conforme demonstrado na Figura 5.

**Figura 5 - LHS bidimensional.**



Fonte: Forrester, 2008.

Uma das principais vantagens do LHS é que ele permite a geração de amostras maiores com eficiência computacional comparável a outras técnicas de amostragem, como amostragem aleatória simples. Isso é particularmente útil em experimentos computacionais ou simulações de modelos complexos, onde a avaliação de cada ponto amostral pode ser demorada ou custosa (Dehlendorff, 2010).

A geração da amostra no LHS segue uma metodologia específica. Primeiramente, uma matriz  $P$  de dimensões  $n \times nv$  é criada, onde cada coluna representa uma permutação aleatória dos números de 1 a  $n$ . Em seguida, uma matriz  $R$  do mesmo tamanho é gerada, com seus elementos sendo números aleatoriamente distribuídos entre 0 e 1. A matriz  $S$  é então obtida através da seguinte equação (Olsson et al., 2003):

$$S = \frac{1}{n}(P - R) \quad (23)$$

A partir da matriz  $S$ , as amostras são geradas de acordo com a seguinte equação:

$$x_{ij} = F_{x_j}^{-1}(S_{ij}) \quad (24)$$

Onde,

$F_{x_j}^{-1}$ : é a inversa da função de distribuição acumulada de probabilidade da variável  $X_j$ .

O LHS é amplamente utilizado em diversas áreas, como engenharia, ciências ambientais, finanças e ciências da saúde, comumente empregada em análises de sensibilidade, calibração de modelos, avaliação de risco e otimização de processos. Essa técnica proporciona uma amostragem mais eficiente e representativa em espaços multidimensionais, auxiliando na compreensão das relações entre as variáveis de entrada e as saídas do modelo, bem como na tomada de decisões embasadas.

### 3.5.3 Modelo de Kriging

O modelo de Kriging é uma técnica de interpolação espacial criada por Krige (1951) e desenvolvida por Matheron em 1963. O método de Kriging permite estimar valores desconhecidos em locais não amostrados a partir de observações em pontos conhecidos, levando em consideração a correlação espacial dos dados.

Existem diferentes variantes do Kriging, como o Kriging Ordinário, o Kriging Universal e o Kriging com Derivadas. O Kriging Ordinário é o mais simples e assume que as médias das variáveis são constantes em todo o domínio de estudo, enquanto o Kriging Universal considera que as médias podem variar espacialmente. Já o Kriging com Derivadas leva em conta informações adicionais sobre a taxa de variação das variáveis no espaço (Cressie, 1993).

A abordagem do Kriging baseia-se no pressuposto de que as variáveis em uma determinada região possuem uma dependência espacial que pode ser modelada por meio de uma função de correlação (Chiles e Delfiner, 2012). A ideia principal do modelo é encontrar uma combinação linear ponderada das observações próximas ao local não amostrado, atribuindo maior peso às observações mais próximas e menor peso às mais distantes, de acordo com a estrutura de correlação espacial dos dados.

A modelagem do modelo de Kriging envolve diferentes etapas definidas a seguir:

1. Coleta de dados: Esses dados podem ser obtidos por meio de experimentos, observações ou simulações, sendo importante ter um conjunto representativo de dados para garantir resultados precisos;

2. Análise exploratória dos dados: Nesta etapa, os dados coletados são analisados para identificar padrões, tendências e possíveis relações espaciais entre as variáveis. Isso pode ser feito por meio de gráficos, estatísticas descritivas e técnicas de análise espacial.

3. Definição do modelo de variograma: O variograma é uma medida da variabilidade entre os pontos amostrados em relação à distância entre eles, que descreve a estrutura de correlação espacial dos dados. Nessa etapa é necessário modelar o variograma experimental para entender a dependência espacial dos dados e selecionar o modelo de variograma mais adequado.

4. Ajuste do modelo de variograma: Com base nos dados e no modelo de variograma selecionado, os parâmetros do variograma são ajustados por meio de métodos estatísticos, como a mínima verossimilhança ou o método dos momentos. O objetivo é encontrar os melhores valores para os parâmetros que representam a estrutura de correlação espacial dos dados.

5. Interpolação espacial: Com o modelo de variograma ajustado, é possível realizar a interpolação espacial para estimar os valores em locais não amostrados. O método de Kriging utiliza uma combinação linear ponderada dos valores observados próximos aos pontos de interesse, levando em consideração a estrutura de correlação espacial. Essa combinação ponderada é calculada com base nos pesos de Kriging.

6. Validação do modelo: Após a interpolação, é necessário a validação de dados que pode ser feita por meio de técnicas de validação cruzada ou utilizando conjuntos de dados independentes, quando disponíveis.

7. Utilização do modelo: Uma vez validado, o modelo de Kriging pode ser usado para estimar valores em locais não amostrados, fazer previsões espaciais, realizar otimizações ou qualquer outra análise que requeira uma representação espacial dos dados.

A modelagem do modelo de Kriging requer conhecimento em estatística espacial e geostatística, além de habilidades em manipulação e análise de dados espaciais. É uma técnica poderosa que permite a modelagem de fenômenos espaciais e a tomada de decisões baseadas em informações espacialmente correlacionadas.

A modelagem matemática do Kriging envolve a estimativa de valores em locais não amostrados com base em dados observados e na estrutura de correlação espacial. Neste modelo é dado um conjunto de dados  $X (x_1, x_2, \dots, x_n)^T$  e a matriz resposta  $Y (y_1, y_2, \dots, y_n)^T$  que representará as variáveis dependentes processo. Segundo Sacks et al. (1989), o conjunto de

metamodelos é construído a partir do modelo  $y(x)$  para os  $n$  dados de entrada, através de um modelo de regressão  $F$ , representado pela Equação 25:

$$y_i(x) = F(\beta_i, x) + z_i(x), \quad i = 1, \dots, q \quad (25)$$

Onde,

$y$ : Notação que simplifica as respostas prevista do modelo  $y(x)$ ;

$F$ : Funções randômica estocástica do modelo de regressão;

$z$ : Processo randômico gaussiano com média nula e variância  $\sigma^2$ ;

$\beta$ : Parâmetros de regressão obtidos para cada variável independente  $i$ .

Para cada função  $Z$ , existe uma covariância associada em que:

$$E[z(w), z(x)] = \sigma^2 \mathcal{R}(\theta, w, x) = cov(w, x) \quad (26)$$

Onde,

$R$ : Matriz de correlação entre os pontos e o vetor parâmetro  $\theta$ ;

$\theta$ : Parâmetro responsável pelo controle do modelo, influência dos pontos adjacentes e pela diferença na superfície de resposta.

O valor da correlação  $R$  é escolhido de acordo com o método de correlação utilizado. Este modelo de resposta pode ser interpretado como os desvios do modelo de regressão, onde mesmo que a resposta seja prevista, pode se assemelhar a um caminho de uma amostra de um processo aleatório  $Z$ , selecionado de forma apropriada (Pinto, 2014).

A aplicação do modelo de Kriging é ampla e diversificada, sendo frequentemente utilizado para a interpolação de dados espacialmente distribuídos, a previsão de valores em locais não amostrados, a análise de incerteza em modelos de simulação e a otimização de amostragem em estudos geoespaciais (Chiles e Delfiner, 2012). O Kriging também é aplicado em estudos de contaminação de solos e águas subterrâneas, análises climáticas, estudos de qualidade do ar e em muitas outras áreas onde a compreensão e o mapeamento espacial de fenômenos são essenciais.

### 3.5.4 Toolbox Dace

O toolbox DACE (*Design and Analysis of Computer Experiments*) do MATLAB é uma ferramenta poderosa para a construção de metamodelos e a realização de otimizações. O DACE é baseado no modelo de Kriging, que é uma técnica de interpolação que realiza a



estimativa e predição de respostas em pontos não amostrados, com base em um conjunto de dados experimentais.

Uma das principais vantagens do DACE é a sua capacidade de otimização, pois o mesmo fornece algoritmos avançados que exploram a função aproximadora construída pelo metamodelo para encontrar os melhores valores dos parâmetros de entrada. Essa abordagem permite uma otimização mais eficiente, especialmente quando a função objetivo é computacionalmente custosa de avaliar.

Esta ferramenta conta com dois principais programas: o "dacefit" e o "predictor". O programa "dacefit" é responsável por computar os elementos do modelo de Kriging, em particular os parâmetros  $\theta$  que são determinados para a solução do problema de otimização não-linear. Por meio do "dacefit", é possível ajustar o modelo de Kriging aos dados experimentais, encontrando os melhores parâmetros que representam a correlação entre as variáveis. Já o programa "predictor" é utilizado para prever as respostas em pontos não experimentados e estimar os erros associados a essas previsões (Lophaven et al., 2002).

A escolha adequada das funções de correlação é essencial para o desempenho do modelo de Kriging. A literatura apresenta diferentes opções, e a seleção deve ser feita levando em consideração o fenômeno subjacente ao processo a ser otimizado (Lophaven et al., 2002).

O toolbox DACE do *MATLAB* é uma ferramenta valiosa para a construção de metamodelos e a realização de otimizações. A combinação do poder do DACE com a versatilidade e recursos do *MATLAB* oferece aos usuários uma abordagem eficiente e eficaz para resolver problemas complexos de otimização, auxiliando na busca pela configuração ótima das variáveis e contribuindo para a redução do tempo e dos recursos necessários para a otimização de processos.

### **3.6 Otimização de Processos**

A modelagem e simulação de processos por métodos computacionais desempenham um papel fundamental no desenvolvimento e otimização de sistemas complexos em diversas áreas, como engenharia, ciências da computação, economia e biologia. Essas técnicas permitem criar representações matemáticas de sistemas físicos ou abstratos, a fim de estudar seu comportamento e realizar análises preditivas.

A modelagem consiste na formulação de equações matemáticas que descrevem as relações entre as variáveis do sistema. Essas equações podem ser baseadas em leis físicas,

princípios estatísticos ou modelos empíricos. Através da modelagem, é possível capturar as características essenciais do processo em estudo, permitindo uma compreensão mais profunda de seu funcionamento.

A simulação, por sua vez, utiliza esses modelos matemáticos para reproduzir o comportamento do sistema ao longo do tempo. Por meio de técnicas computacionais avançadas, é possível realizar experimentos virtuais e obter resultados que seriam inviáveis ou custosos de se obter na prática. A simulação permite explorar diferentes cenários, testar estratégias e tomar decisões embasadas em dados.

As técnicas de modelagem e simulação têm se beneficiado do avanço da capacidade computacional, permitindo a construção de modelos cada vez mais complexos e precisos. Além disso, o desenvolvimento de softwares especializados, como o *MATLAB*, o *Simulink* e o *COMSOL*, tem facilitado a implementação e análise desses modelos.

Diversas referências são fundamentais para a compreensão desses métodos computacionais. Alguns exemplos incluem o livro "*Process Modeling, Simulation, and Control for Chemical Engineers*" de W. L. Luyben, que aborda a modelagem de processos químicos e as técnicas de simulação aplicadas a esses sistemas e o livro "*Simulation Modeling and Analysis*" de A. M. Law e W. D. Kelton, que apresenta os princípios e as práticas da simulação de sistemas em geral.

A modelagem e simulação de processos por métodos computacionais são ferramentas poderosas para entender e otimizar sistemas complexos. Com a capacidade de criar modelos matemáticos e realizar simulações virtuais, é possível estudar o comportamento dos sistemas, realizar experimentos virtuais e tomar decisões embasadas em dados. Essas técnicas têm contribuído significativamente para o avanço científico e tecnológico em diversas áreas do conhecimento.

### **3.6.1 Real Time Optimization**

A otimização em tempo real (RTO - *Real Time Optimization*) é uma abordagem essencial e em constante expansão na indústria, especialmente nos setores de processos químicos e bioquímicos, com o objetivo de maximizar o desempenho das unidades de processo enquanto atende às restrições operacionais e ao perfil de produção definido pelas camadas superiores. Esta prática tem sido aplicada há décadas, buscando explorar ao máximo o potencial das unidades industriais.

A essência da RTO reside na determinação contínua e em tempo real do valor apropriado para as variáveis do processo, a fim de alcançar o máximo desempenho possível, sem violar as restrições do sistema. Isso é realizado através da integração de um modelo matemático do processo com algoritmos de otimização avançados. Este modelo matemático representa o comportamento do sistema e suas interações, enquanto o algoritmo de otimização calcula periodicamente as variáveis de controle ideais que otimizam uma função objetivo específica.

Essa abordagem tem aplicações em diversas áreas industriais, incluindo manufatura, petróleo e gás, tratamento de água e efluentes, entre outras. Um exemplo notável é a sua aplicação no setor de tratamento de águas residuais, onde a RTO pode melhorar significativamente a remoção de poluentes e reduzir o consumo de energia (Haq et al., 2022). Nesse contexto, a RTO adapta automaticamente os processos de tratamento, respondendo às variações nas características da água de entrada. Isso resulta em eficiência aprimorada na remoção de poluentes, minimizando o desperdício e melhorando a qualidade da água tratada.

No setor de petróleo e gás, a RTO é usada para otimizar processos de refino, maximizando a produção de produtos de alto valor e minimizando os custos operacionais (Gomes, 2007).

De acordo com Gomes (2007), as ferramentas de otimização em tempo real consistem em combinar um modelo matemático do processo com um algoritmo de otimização que periodicamente determina o ponto ótimo operacional a ser seguido. Isso permite que a planta industrial seja ajustada continuamente para maximizar a eficiência e, ao mesmo tempo, garantir o cumprimento das restrições operacionais e regulatórias.

Além disso, a complexidade crescente dos processos industriais, juntamente com a necessidade de maximizar o lucro econômico, torna a aplicação das estratégias de otimização em tempo real (RTO) uma prática fundamental. Conforme afirmado por Calvo (2015), a RTO é responsável por aprimorar, controlar e estabilizar a operação da planta, enquanto procura ativamente o ponto de operação mais lucrativo.

### **3.6.2 Otimizadores**

As ferramentas de otimização desempenham um papel fundamental em uma ampla variedade de domínios, desde a modelagem de sistemas complexos até a resolução de

problemas de programação não linear. Dentre as principais ferramentas, destacam-se o GPML, IPOPT, FILTERSD, NLOPT, NOMAD e o MATLAB fmincon.

O GPML, ou *Gaussian Processes Machine Learning*, oferece uma abordagem versátil para modelagem, permitindo uma modelagem flexível por meio de algoritmos de inferência aproximada para processos Gaussianos. Sua gratuidade e facilidade de uso o tornam uma escolha popular (Fernandes, 2022).

O IPOPT (*Interior Point Optimizer*) destaca-se na resolução de problemas não lineares suaves, especialmente em problemas convexos. Embora encontre apenas soluções locais, é uma referência em otimização de código aberto (Wächter & Biegler, 2006).

O FILTERSD é conhecido por sua eficiência na resolução de problemas não lineares suaves e sua rapidez, mesmo como um solucionador local (Gilbert, 2016).

A biblioteca NLOpt oferece uma interface comum para várias rotinas de otimização não linear gratuitas, tornando a otimização local e global mais acessível e rápida (Steven, 2016).

O NOMAD utiliza o algoritmo MADS para resolver problemas globais não lineares e não diferenciáveis, sendo especialmente adequado para problemas de média escala (Audet et al., 2021).

O MATLAB fmincon, parte do *Optimization Toolbox*, é amplamente utilizado em uma variedade de campos, resolvendo problemas de otimização com funções de restrição, limites e restrições de igualdade (MathWorks, 2022).

Essas ferramentas oferecem um conjunto diversificado de capacidades, tornando-as essenciais para a resolução de problemas complexos em áreas que vão desde a engenharia até a pesquisa científica, demonstrando a importância crítica das ferramentas de otimização.

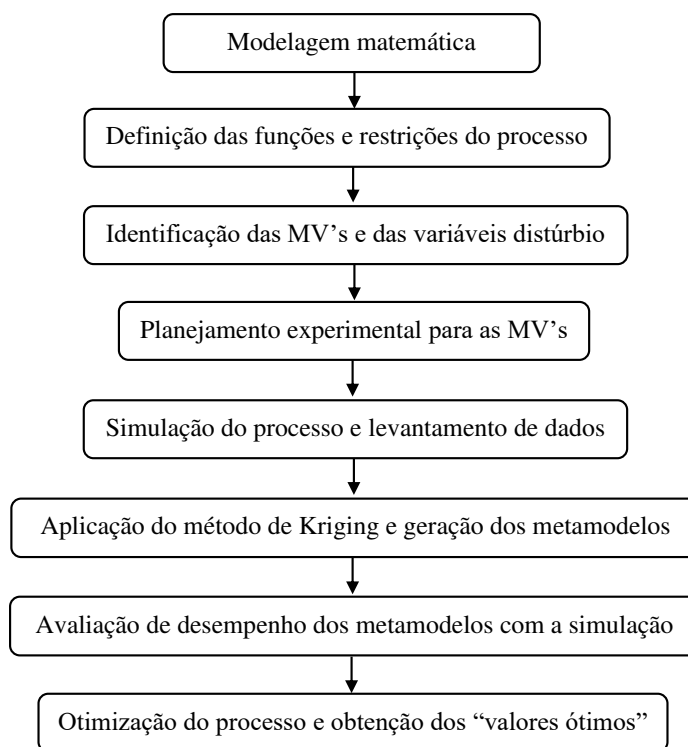
#### 4. MATERIAIS E MÉTODOS

As atividades desenvolvidas relacionada a pesquisa foram realizadas no Laboratório de Experimentação Numérica de Processos – LENP, da Unidade Acadêmica de Engenharia Química (UAEQ), no Centro de Ciências e Tecnologia (CCT) da Universidade Federal de Campina Grande (UFCG), com apoio do CNPq, Conselho Nacional de Desenvolvimento Científico e Tecnológico – Brasil.

Para desenvolvimento da pesquisa foi utilizada metodologia baseada nos trabalhos de Alex et al. (2008), Jeppsson et al. (2011) e Fernandes (2022). As seguintes etapas foram realizadas: modelagem matemática, definição das variáveis de controle e manipulação, restrições, planejamento experimental para as variáveis manipuladas (MV's), simulação do processo e levantamento de dados no *Simulink* e *Matlab*, aplicação e avaliação de metamodelos e a otimização do processo através da otimização em tempo real. Esta pesquisa foi desenvolvida a partir do *software Matlab®/Simulink*, da empresa *MathWorks, Inc.*, para a simulação do processo BSM1 e construção dos metamodelos.

Na Figura 6 está representado o diagrama de forma esquemática da metodologia utilizada na pesquisa.

**Figura 6** - Diagrama do desenvolvimento da pesquisa.



**Fonte:** Elaborado pela autora.

## 4.1 Modelagem Matemática

A modelagem e simulação do BSM1 foram realizadas por meio de equações e modelos que descrevem os processos ocorrentes em cada unidade da ETAR. Nas seções 3.6.7 e 3.6.8 foi apresentada a modelagem matemática do processo, incluindo os balanços de massa dos biorreatores e a modelagem do sedimentador. Essas equações são resolvidas pelo simulador, que utiliza os balanços em cada bloco adicionado para obter as soluções correspondentes.

A seguir, são apresentadas as taxas de conversão das concentrações, quais são definidas pela combinação dos processos biológicos, conforme apresentado pela Equação 27 (Jeppsson et al., 1996):

$$r_k = \sum v_{kj} \rho_j \quad (27)$$

Em relação à matéria orgânica inerte solúvel ( $S_1$ ), não há reação:

$$r_1 = 0 \quad (28)$$

Para o substrato rapidamente biodegradável (SI), ocorre consumo devido ao crescimento de micro-organismos heterotróficos em condições aeróbicas e anóxicas, bem como produção por meio da hidrólise do substrato lentamente biodegradável. Logo, temos que:

$$r_2 = -\frac{1}{Y_H} \rho_1 - \frac{1}{Y_H} \rho_2 + \rho_7 \quad (29)$$

Em relação a matéria orgânica inerte particulada ( $X_1$ ), não há reação. Logo:

$$r_3 = 0 \quad (30)$$

A concentração de substrato lentamente biodegradável (particulado), aumenta com o reciclo da matéria orgânica morta e é reduzida pela hidrólise, como expresso na Equação 31:

$$r_4 = (1 - f_p) \rho_4 + (1 - f_p) \rho_5 - \rho_7 \quad (31)$$

A biomassa heterótrofa ( $X_{B,H}$ ) aumenta devido ao crescimento microbiano aeróbico e anóxico heterótrofos, porém é reduzida devido ao decaimento da biomassa heterótrofa, conforme descrito na Eq. 32:

$$r_5 = \rho_1 + \rho_2 - \rho_4 \quad (32)$$

A biomassa autótrofa ( $X_{B,A}$ ) cresce com o crescimento autótrofo e decresce com à morte das células autotróficas, de acordo com a Eq. 33:

$$r_6 = \rho_3 - \rho_5 \quad (33)$$

A formação de produtos inertes particulados, provenientes do decaimento da biomassa autótrofa e heterótrofa, é dado por:

$$r_7 = f_p(\rho_4 + \rho_5) \quad (34)$$

A concentração de oxigênio dissolvido ( $S_O$ ) é reduzida pelo crescimento aeróbico da biomassa autotrófica e heterotrófica, quando há presença de oxigênio, conforme indicado na Equação 35:

$$r_8 = -\frac{1 - Y_H}{Y_H} \rho_1 - \frac{4,57 - Y_A}{Y_A} \rho_3 \quad (35)$$

A concentração de nitrato ( $S_{NO}$ ) aumenta devido nitrificação e diminui devido a desnitrificação, logo:

$$r_9 = -\frac{(1 - Y_H)}{2.86Y_H} \rho_2 + \frac{1}{Y_H} \rho_3 \quad (36)$$

A concentração de amônia ( $S_{NH}$ ) aumenta devido à reação de amonificação do nitrogênio orgânico solúvel e diminui devido ao crescimento de todos os tipos de micro-organismos, pois a amônia é uma fonte de nitrogênio para as células (Jeppson,1996). Sua taxa de reação é descrita conforme a Eq. 37:

$$r_{10} = -i_{XB}\rho_1 - i_{XB}\rho_2 - \left(i_{XB} + \frac{1}{Y_A}\right)\rho_3 + \rho_6 \quad (37)$$

A concentração de nitrogênio orgânico solúvel ( $S_{ND}$ ) é afetado pela amonificação e pela hidrólise:

$$r_{11} = -\rho_6 + \rho_8 \quad (38)$$

A concentração de nitrogênio orgânico particulado ( $X_{ND}$ ) aumenta com o decaimento da biomassa e diminui com a hidrólise do nitrogênio orgânico em nitrogênio solúvel. Logo, temos que:

$$r_{12} = (i_{XB} - f_P i_{XP})(\rho_4 + \rho_5) - \rho_8 \quad (39)$$

O ciclo de reações no sistema de lodo ativado pode ser observado na Figura 4, que demonstra as principais reações e os compostos envolvidos (Jeppsson, 1996).

## 4.2 Funções de avaliação de desempenho

No modelo BSM1, são considerados dois níveis de análise de desempenho para avaliar o sistema de tratamento de águas residuais. O primeiro nível aborda os critérios de qualidade do efluente, buscando atender aos índices estabelecidos para garantir a eficiência do tratamento. O segundo nível de análise está relacionado à violação dos limites de qualidade no efluente, ou seja, verifica se os parâmetros do efluente ultrapassam os valores permitidos, o que indica um mau desempenho do sistema.

Além disso, o modelo BSM1 também leva em consideração o índice de custo operacional do sistema de tratamento. Essa análise avalia os custos associados à energia durante seu tempo de operação, visando encontrar soluções eficientes do ponto de vista econômico.

- **Critérios de qualidade do efluente:**

Para avaliar a eficácia da modelagem proposta para a unidade de tratamento de efluentes, são comparados os principais componentes presentes no efluente. Esses componentes incluem o nitrogênio total, a demanda química de oxigênio (DQO), a concentração de amônia, os sólidos suspensos e a demanda biológica de oxigênio (DBO). Essas variáveis são obtidas a partir das concentrações dos produtos gerados no sistema reativo, que consistem em diferentes componentes químicos e biológicos. O seguinte conjunto de equações, baseado nos trabalhos de Alex et al. (2008) e Jeppsson et al. (2011), descreve essas relações, onde:

A demanda química de oxigênio será dada por:

$$COD = S_{S,e} + S_{I,e} + X_{S,e} + X_{I,e} + X_{B,H,e} + X_{B,A,e} + X_{P,e} \quad (40)$$



Já a demanda biológica de oxigênio é estimada como uma fração da DQO, levando em consideração a fração biodegradável dos componentes biológicos:

$$BOD_5 = 0,25 (S_{S,e} + X_{S,e} + (1 - f_p)(X_{B,H,e} + X_{B,A,e})) \quad (41)$$

A concentração de sólidos suspensos corresponde a 75% da biomassa particulada presente no efluente, incluindo os inertes e os produtos particulados resultantes do decaimento da biomassa:

$$TSS_e = 0,75(X_{S,e} + X_{I,e} + X_{B,H,e} + X_{B,A,e} + X_{P,e}) \quad (42)$$

A quantidade de nitrogênio total é dada pela expressão:

$$T_N = S_{NO,e} + S_{(NKj),e} \quad (43)$$

Onde  $S_{NK,e}$  corresponde a concentração de nitrogênio de Kjeldahl:

$$S_{NK,e} = S_{NH,e} + S_{ND,e} + X_{ND,e} + i_{XB}(X_{B,H,e} + X_{B,A,e}) + i_{XP}(X_{P,e} + X_{I,e}) \quad (44)$$

- **Limites de qualidade do efluente**

Tem-se como restrições de qualidade do efluente:

$$N_{tot,effluent} < 18 \frac{gN}{m^3}$$

$$COD_{effluent} < 100 \frac{gCOD}{m^3}$$

$$S_{NH,effluent} < 4 \frac{gN}{m^3}$$

$$TSS_{effluent} < 30 \frac{gSS}{m^3}$$

$$BOD5_{effluent} < 10 \frac{gBOD}{m^3}$$

$$14756.8 \frac{m^3}{day} \leq Q_r \leq 22135.2 \frac{m^3}{dia}$$

$$308 \frac{m^3}{day} \leq Q_w \leq 462 \frac{m^3}{dia}$$

$$44270.4 \frac{m^3}{day} \leq Q_a \leq 66405.6 \frac{m^3}{dia}$$

$$192 \leq k_L a(3) \leq 288$$

$$192 \leq k_L a(4) \leq 288$$

$$67.2 \leq k_L a(5) \leq 100.8$$

- **Custo total da planta (Overall Cost Index – OCI)**

O *Overall Cost Index* (OCI), que representa o custo total da planta, quantifica energeticamente (em kWh/dia) o custo de operação ao longo do tempo e, é dado pela seguinte expressão:

$$OCI = AE + PE + 5SP + ME \quad (45)$$

Onde,

Para a energia de aeração usa-se a relação:

$$AE = \frac{S_0^{Sat}}{T * 1,8 * 1000} \int_{t_0=0}^{t=t_{final}} \sum V_i K_L a_i(t) dt \quad (46)$$

PE corresponde à energia consumida pelo bombeamento e depende dos fluxos da unidade de tratamento de efluentes, bem como da disposição dos tanques na estação. Sua expressão é dada por:

$$PE = \frac{1}{T} \int_{t_0=0}^{t=t_{final}} (0,004Q_a(t) + 0,008Q_r(t) + 0,05Q_w(t)) dt \quad (47)$$

SP é a produção de lodo e é calculada a partir do fluxo de sólidos descartado do sistema e dos sólidos acumulados dentro do sistema durante o período de operação, onde:

$$SP = 0,75 \int_{t_0=0}^{t=t_{final}} (X_{S,w} + X_{I,u} + X_{B,H,w} + X_{B,A,w}) Q_w(t) dt \quad (48)$$

ME representa a energia utilizada para a mistura nos compartimentos do biorreator, a fim de evitar a sedimentação do substrato e da matéria orgânica, dada por:

$$ME = \begin{cases} \frac{24}{T} \int_{t_0=0}^{t=t_{final}} (0,005 \cdot V_i) dt, & \text{se } K_L a(t) < 20d^{-1} \\ 0, & \text{caso contrário} \end{cases} \quad (49)$$

Essas equações permitem avaliar o consumo energético da planta de tratamento de efluentes, levando em consideração os diferentes componentes e processos envolvidos.

### 4.3 Variáveis de Entrada

As variáveis de entrada no modelo BSM1 referem-se às condições do influente da planta. A Tabela 6 apresenta essas variáveis e suas unidades correspondentes.

**Tabela 6** - Variáveis no influente da planta BSM1.

Variável	Descrição	Unidade
Q	Fluxo volumétrico	$m^3 \cdot d^{-1}$
S <sub>I</sub>	Matéria Orgânica inerte solúvel	$gDQO \cdot m^{-3}$
S <sub>S</sub>	Matéria Orgânica rapidamente biodegradável (solúvel)	$gDQO \cdot m^{-3}$
X <sub>I</sub>	Matéria Orgânica particulada	$gDQO \cdot m^{-3}$
X <sub>S</sub>	Matéria Orgânica lentamente biodegradável (particulada)	$gDQO \cdot m^{-3}$
X <sub>B,H</sub>	Biomassa Heterótrofa	$gDQO \cdot m^{-3}$
X <sub>B,A</sub>	Biomassa Autótrofa	$gDQO \cdot m^{-3}$
S <sub>NH</sub>	Amônia e amônio	$gN \cdot m^{-3}$
S <sub>ND</sub>	Nitrogênio orgânico solúvel	$gN \cdot m^{-3}$
X <sub>ND</sub>	Oxigênio orgânico particulado (insolúvel)	$gN \cdot m^{-3}$
X <sub>P</sub>	Matéria Orgânica particulada resultante da morte microbiana	$gDQO \cdot m^{-3}$
S <sub>O</sub>	Oxigênio dissolvido	$g(-DQO) \cdot m^{-3}$
S <sub>NO</sub>	Nitrato e Nitrito	$gN \cdot m^{-3}$
TSS	Sólidos suspensos totais	$gSS \cdot m^{-3}$
COD	Demanda Química de Oxigênio	$gCOD \cdot m^{-3}$
BOD	Demanda Bioquímica de Oxigênio	$gBOD \cdot m^{-3}$

Fonte: Elaborado pela autora.

As Tabelas 7 e 8 apresentam as variáveis manipuladas (MV) e as variáveis de distúrbio (d), respectivamente. Os valores nelas contidos estão definidos como  $\pm 20\%$  do valor nominal da tabela, uma escolha fundamentada em experimentos conduzidos por diversos autores, nos quais se pôde observar a ocorrência de picos.

**Tabela 7 - Variáveis manipuladas.**

Notação	Descrição	Valor nominal	Unidade
$Q_r$	Reciclo externo	18446	$m^3d^{-1}$
$Q_w$	Descarte	385	$m^3d^{-1}$
$Q_a$	Reciclo interno	55338	$m^3d^{-1}$
$K_1a_3$	Coefficiente de transferência de massa – compartimento 3	240	$d^{-1}$
$K_1a_4$	Coefficiente de transferência de massa – compartimento 4	240	$d^{-1}$
$K_1a_5$	Coefficiente de transferência de massa – compartimento 5	84	$d^{-1}$

Fonte: Elaborado pela autora.

**Tabela 8 - Variáveis de distúrbio.**

Variável	Valor nominal	Unidade
Q	18446	$m^3 \cdot d^{-1}$
$S_I$	30	$gDQO \cdot m^{-3}$
$S_S$	69.5	$gDQO \cdot m^{-3}$
$X_I$	51.2	$gDQO \cdot m^{-3}$
$X_S$	202.32	$gDQO \cdot m^{-3}$
$X_{B,H}$	28.17	$gDQO \cdot m^{-3}$
$S_{NH}$	31.56	$gN \cdot m^{-3}$
$S_{ND}$	6.95	$gN \cdot m^{-3}$
$X_{ND}$	10.59	$gN \cdot m^{-3}$

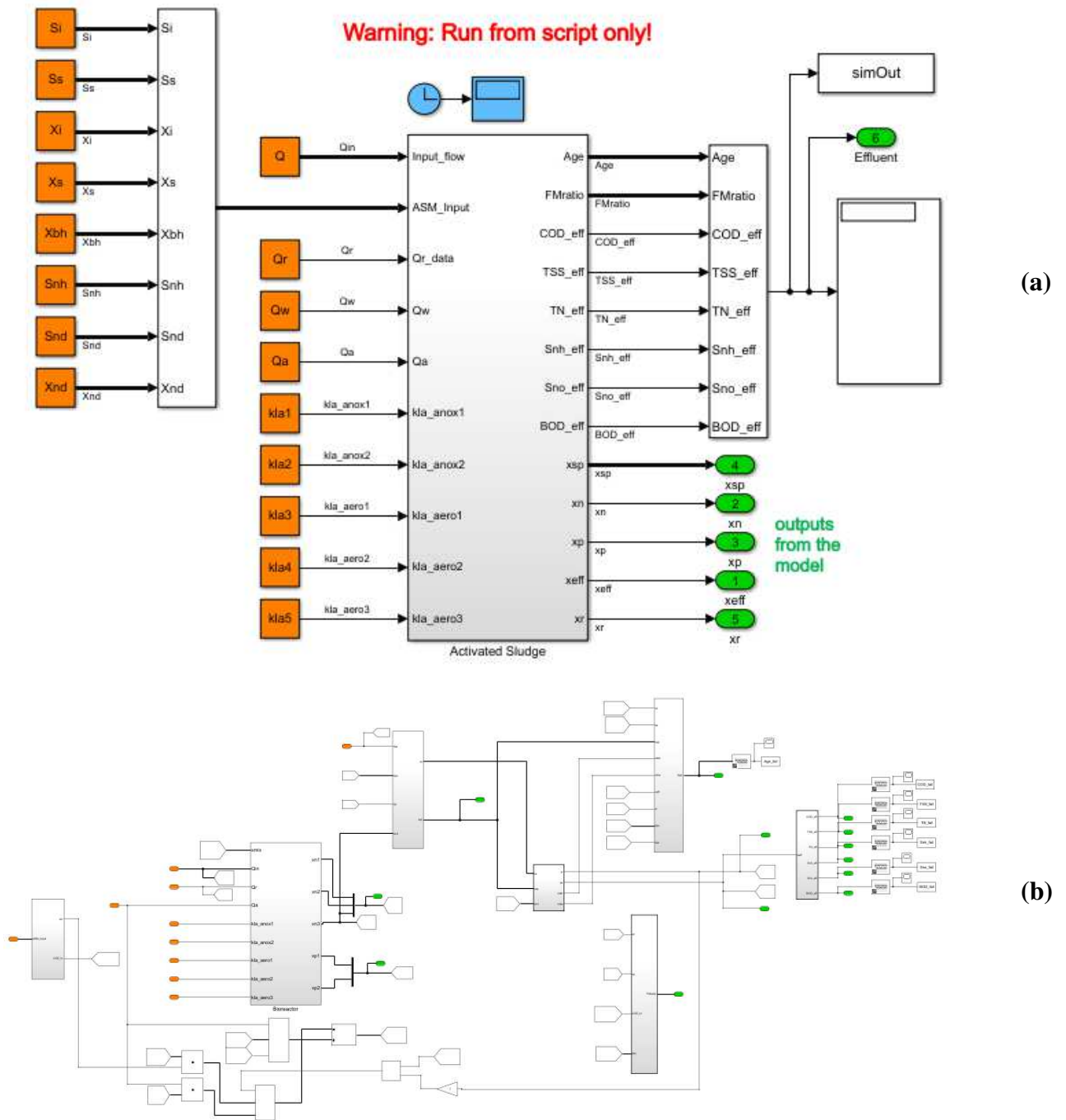
Fonte: Elaborado pela autora.

#### 4.4 Simulação do Processo

A simulação do processo, geração do metamodelo kriging e a otimização em tempo real foram implementadas no software *Simulink* e *MATLAB*<sup>®</sup>, respectivamente, utilizando a linguagem C. Essas ferramentas foram utilizadas por Jeppsson et al. (2011) para avaliar o modelo tanto em regime estacionário quanto dinâmico, pois a linguagem C oferece maior rapidez na obtenção dos resultados devido à compilação em vez de interpretação. No *Simulink*, as operações unitárias são representadas por subsistemas, em que cada bloco contém a modelagem interna da respectiva unidade.

A Figura 7, apresenta a implementação do modelo BSM1 desenvolvido por Jeppsson et al. (2011) no software *Simulink*, qual foi utilizado nesta pesquisa.

**Figura 7 -** Processo do lodo ativado BSM1 em ambiente *Simulink* a) Visão geral e b) Representação por subsistemas.



Fonte: Adaptado de Jeppsson et al., 2011.

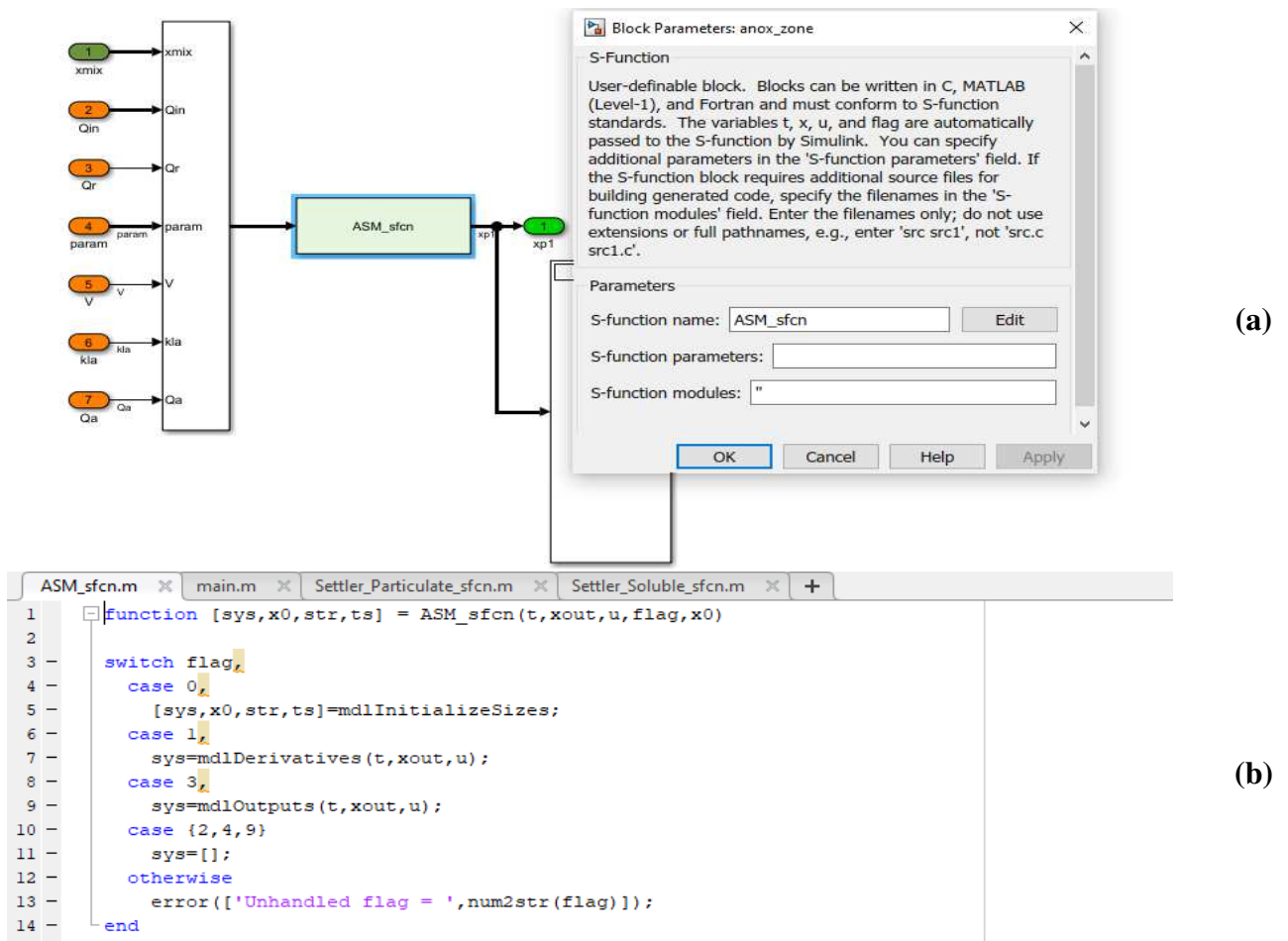
A modelagem e simulação de sistemas complexos desempenham um papel fundamental na compreensão e análise de processos em diversas áreas, incluindo engenharia química, ambiental e de controle. No contexto do tratamento de águas residuais, o sistema BSM1 tem sido amplamente utilizado como uma ferramenta de referência para avaliar o desempenho de estações de tratamento de efluentes e sua montagem no *Simulink* permite uma

representação detalhada dos componentes e processos envolvidos na estação de tratamento de águas residuais.

Para a construção do modelo, Figura 8, foi realizado a representação dos biorreatores, sedimentador, processos de conversão de compostos e interações entre as variáveis de interesse. Cada componente do sistema é modelado como um subsistema dentro do *Simulink*, o que permite uma abordagem modular e a fácil implementação de equações e algoritmos específicos para cada unidade do sistema.

A implementação dos modelos necessários para descrever o processo no reator biológico, decantador e unidades afluentes foi realizada a partir da interface gráfica do usuário no *Simulink* onde, foi criado os blocos e estabeleceu as conexões entre eles. Em seguida, como não havia um modelo pré-existente para representar o processo de lodo ativado, a linguagem *MATLAB* foi empregada para construir os modelos utilizando uma notação matemática. Esses modelos foram posteriormente incorporados ao ambiente do *Simulink* por meio da função S, como demonstrado na Figura 8b.

**Figura 8** - Incorporação do modelo matemático em Matlab no ambiente do Simulink utilizando S-function a) Janela de parâmetros do bloco S-function e b) Início da função chamada pelo bloco.



Fonte: Elaborado pela autora.

Essa abordagem combina a facilidade de uso e a flexibilidade gráfica do *Simulink* com o poder de programação e cálculos matemáticos do *MATLAB*. A interface gráfica intuitiva do *Simulink* permite uma montagem visual dos blocos do sistema, enquanto a linguagem *MATLAB* oferece recursos avançados para o desenvolvimento dos modelos matemáticos subjacentes.

A S-function é dividida em um segmento de código que executa uma tarefa com base na etapa atual da simulação, especificada pelo valor de flag:

```
[sys, x0, str, ts] = sfunc(t, x, u, flag)
    if flag = 0
        (...)
    else if flag = 4
        (...)
```

No *Simulink*, o operador *s* é usado para definir funções de transferência e equações diferenciais que descrevem o comportamento dos sistemas. Ele permite a representação matemática de sistemas lineares através de modelos de blocos interconectados.

A função *s* é usada em combinação com outras funções e operadores do *MATLAB* para construir modelos dinâmicos no *Simulink*. Por exemplo, ao definir uma função de transferência de um sistema linear contínuo no domínio da frequência, o operador *s* é usado para representar a variável complexa "s" nas expressões matemáticas. Sua utilização em conjunto com outras funções e operadores do *MATLAB* permite a criação de modelos complexos e a simulação de sistemas reais em diversos campos.

#### 4.4.1 Script “rto\_app.m”

O propósito desse código em *MATLAB* é realizar a otimização em tempo real do modelo BSM1 (*Benchmark Simulation Model No. 1*), tendo como objetivo encontrar as configurações ótimas para as variáveis manipuladas do modelo, levando em consideração as restrições impostas pelas variáveis de perturbação e as metas desejadas de desempenho do sistema de tratamento de efluentes, utilizando metamodelos baseados em Kriging para acelerar o processo de otimização e fornece relatórios e gráficos para análise dos resultados obtidos. A Figura 9 representa o início do código em *MATLAB* e o código completo pode ser encontrado no Anexo 1.

**Figura 9** - Início do Script em MATLAB "rto\_app.m".

```
1 - clearvars; clc; close all; format compact
2 - load resultsSimulink % the trained 'mdl' model (and all its properties as a struct) is here (check the .mat file f
3 - load states % x = nominal state variables values in dataset format
4 - load dv % dv = nominal manipulated variables values, dv = [(1) Q; (2) Si; (3) Ss; (4) Xi; (5) Xs; (6) Xbh; (7) Xba; (8) X
5 - load mv % mv = nominal disturbance variables values, mv = [(1) Qr; (2) Qw; (3) Qa; (4) kla1; (5) kla2; (6) kla3; (7) kla
6
7 % training Kriging metamodels
8 - load regressionGP % regressionGP. Used when metamodel is obtained by brute-force Kriging and not Thalita's algorit
9
10 % setting up GPML toolbox
11 - addpath('gpml-matlab-v4.2-2018-06-11') % add GPML toolbox to Matlab path
12 - run('startup.m') % start up GPML toolbox
13
14 % disturbances: Xba = 0, Xp = 0, So = 0, Sno = 0, and Salk = 7 are not considered, so d = [Q; Si; Ss; Xi; Xs; Xbh;
15 - d = dv([1:6 11:13]);
16 - dmin = 0.80*d;
```

Fonte: Elaborado pela autora.

A sequência de ações realizadas no código é a seguinte:

1. Carregamento dos dados: O código carrega os dados necessários para a otimização e simulação do modelo BSM1. Isso inclui o modelo treinado com o resultado do simulink, os valores nominais das variáveis de estado “states”, os valores nominais das variáveis manipuladas “dv”, e os valores nominais das variáveis de perturbação “mv”.

2. Treinamento dos metamodelos: O código utiliza os dados carregados para treinar os metamodelos baseados em Kriging, o arquivo “regressionGP” contém os metamodelos Kriging obtidos a partir do treinamento. Esses metamodelos são usados para representar de forma aproximada o comportamento do modelo BSM1, reduzindo assim o tempo computacional necessário para a otimização.

3. Configuração do GPML Toolbox: O código adiciona o GPML Toolbox ao caminho de busca do MATLAB e executa o arquivo “startup.m” para inicializar a toolbox.

4. Definição dos limites das variáveis de perturbação e manipulação: O código define os limites mínimos e máximos para as variáveis de perturbação (“dmin” e “dmax”) e para as variáveis de manipulação (“umin” e “umax”). Esses limites são utilizados na otimização para garantir que as soluções estejam dentro de faixas razoáveis.

5. Geração de dados de perturbação: O código gera dados de perturbação para a otimização. Os dados podem ser gerados utilizando o método Latin Hypercube Sampling (LHS) para garantir uma distribuição uniforme dentro dos limites definidos ou carregado a partir do arquivo “disturbances”.

6. Otimização: O código realiza a otimização para cada conjunto de dados de perturbação gerado. Ele utiliza diferentes otimizadores locais e globais disponíveis, como 'ipopt', 'matlab', 'filtersd', 'nlopt' e 'nomad'. A otimização é realizada para encontrar as configurações ótimas das variáveis de manipulação que minimizam a função objetivo e



satisfazem as restrições impostas. É realizado um loop para cada conjunto de perturbações, onde dentro do loop é chamada a função "optimizer\_gpr\_app" para otimizar as variáveis de manipulação com base nas perturbações e nos modelos de regressão.

**7. Melhores soluções:** Nesta etapa, ocorre o processamento e seleção das melhores soluções de controle encontradas durante o processo de otimização em tempo real, assim identificando as soluções que atendem aos requisitos do sistema e armazenando as informações relevantes para análise adicional, onde a melhor solução é utilizada na simulação em Simulink do modelo BSM1.

**8. Simulações:** O código realiza simulações do modelo BSM1 usando as configurações ótimas encontradas, onde é realizado um loop para simular o modelo com as novas variáveis. Também é registrado as trajetórias das variáveis de estado e manipulação ao longo do tempo.

**9. Armazenamento dos resultados:** Os resultados das simulações otimizadas são salvos em um arquivo chamado "results\_rto". Isso permite que os resultados da otimização e da simulação otimizada sejam analisados e comparados posteriormente para avaliar o desempenho do sistema sob as condições de controle ótimas.

**10. Relatórios e resultados:** O código gera tabelas e gráficos para apresentar os resultados da otimização. Ele relata os valores da função objetivo obtidos pela otimização em comparação com os valores da função objetivo da simulação e mostra os tempos de execução dos otimizadores, as soluções ótimas encontradas e os valores das restrições correspondentes. São geradas tabelas com os resultados das otimizações e as soluções ótimas encontradas para cada conjunto de perturbações e são criados gráficos para visualizar os resultados da simulação, mostrando as variáveis de saída e as variáveis de manipulação ao longo do tempo.

**11. Fechamento do modelo e finalização do código.**

#### **4.4.2 Script "optimizer\_gpr\_app.m"**

O propósito do código é a otimização em tempo real dos metamodelos para encontrar os valores ótimos das variáveis de controle (manipulações) que melhor atendem às restrições definidas e minimizam a função objetivo. A Figura 10 representa o início do código em *MATLAB* e o código completo pode ser encontrado no Anexo 2.

Figura 10 - Início do Script em MATLAB “optimizer\_gpr\_app.m”.

```
ASM_sfcn.m x main.m x Settler_Partuculate_sfcn.m x Settler_Soluble_sfcn.m x rto_app.m x optimizer_gpr_app.m x +
1 % computes optimal values of manipulations
2 % mdl = {(1)Age; (2)FMratio; (3)COD_eff; (4)TSS_eff; (5)TN_eff; (6)Snh_eff; (7)Sno_eff; (8)BOD_eff; (9)oci}
3 % solver = {'ipopt', 'matlab', 'filtersd', 'nlopt', 'nomad'}
4
5 % bound constraints (lb <= x <= ub)
6 lb = [d; umin];
7 ub = [d; umax];
8 % nonlinear constraints (cl <= nlcon(x) <= cu). Constraints from J. Alex, L. Benedetti, J. Copp, K.V. Gerbaey, U. J
9 nlcon = @(z) [myPredict(z(:)', [], mdl{5}, parX, parY, par); % 0 <= TN_eff <= 18
10 myPredict(z(:)', [], mdl{3}, parX, parY, par); % 0 <= COD_eff <= 100
11 myPredict(z(:)', [], mdl{6}, parX, parY, par); % 0 <= Snh_eff <= 4
12 myPredict(z(:)', [], mdl{4}, parX, parY, par); % 0 <= TSS_eff <= 30
13 myPredict(z(:)', [], mdl{8}, parX, parY, par); % 0 <= BOD_eff <= 10
14 myPredict(z(:)', [], mdl{9}, parX, parY, par); % 0 <= oci, ensuring the objective function is always pos
15
16 cl = [0; 0; 0; 0; 0; 0];
```

Fonte: Elaborado pela autora.

A sequência de ações realizadas no código é a seguinte:

1. Entradas da função:

- "d": Vetor contendo os valores das variáveis de perturbação (distúrbios) no modelo BSM1.
- "mdl": Arquivo contendo informações sobre as saídas do modelo BSM1, representadas por: Age, FMratio, COD\_eff, TSS\_eff, TN\_eff, Snh\_eff, Sno\_eff, BOD\_eff e oci.
- "u0": Vetor inicial contendo os valores iniciais das variáveis de controle (manipulações) do modelo BSM1.
- "umin" e "umax": Vetores contendo os limites inferiores e superiores, respectivamente, das variáveis manipuladas do modelo BSM1.
- "parX", "parY" e "par": Parâmetros necessários para fazer previsões usando o modelo BSM1.
- "solver": solucionadores de otimização a serem utilizados (por exemplo: 'ipopt', 'matlab', 'filtersd', 'nlopt', 'nomad').

2. Limites e restrições: O código estabelece limites superiores e inferiores para as variáveis de controle e impõe restrições ao modelo BSM1 usando a função "nlcon". Essas restrições garantem que os valores das variáveis de saída do modelo BSM1 estejam dentro de faixas específicas. As restrições são relacionadas a várias variáveis, como TN\_eff, COD\_eff, Snh\_eff, TSS\_eff, BOD\_eff e a função objetivo "oci" que deve ser sempre positiva.

Essas restrições garantem que os valores das variáveis de saída do modelo BSM1 estejam dentro de faixas específicas. As restrições são relacionadas a várias variáveis, como TN\_eff, COD\_eff, Snh\_eff, TSS\_eff, BOD\_eff e a função objetivo "oci" que deve ser sempre positiva.

3. Otimização: O código utiliza diferentes solucionadores de otimização fornecidos pelo usuário ('ipopt', 'matlab', 'filtersd', 'nlopt', 'nomad') para encontrar os valores ideais das variáveis de controle que minimizam a função objetivo "oci", que representa o índice de qualidade de operação, enquanto atendem às restrições especificadas. A função objetivo é

aplicada no otimizador como uma função anônima "fun" e as restrições não lineares são incorporadas usando a função "nlcon".

4. Saídas: A função retorna as seguintes saídas:

- O otimizador é executado para cada opção especificada em **solver**.
- O tempo decorrido durante a otimização é registrado no vetor “elapse”.
- Os valores ótimos das variáveis de manipulação são armazenados na variável “u”.
- Os valores das restrições e da função objetivo após otimização são armazenados nas variáveis “nonlcon” e “fOpt”, respectivamente.
- O código também registra o “exitflag” para indicar o estado de saída do otimizador.

Essa rotina é parte integrante do processo de otimização em tempo real do modelo, que é realizado para encontrar as melhores configurações das variáveis de manipulação com base nas restrições e nos objetivos desejados.

## 5. RESULTADOS E DISCUSSÃO

Todos os experimentos foram aplicados em uma planta de tratamento de efluentes simulada nos softwares *Simulink* e *MATLAB*. A pesquisa teve como intuito a otimização e simulação do modelo com os novos dados de controle, visando a melhora do cenário base de forma automática utilizando de técnicas de aprendizado de máquina. Comparou-se a performance do modelo com os dados da simulação e com os dados provenientes da otimização, como também a comparação entre os metamodelos gerados.

### 5.1 Geração do metamodelo

Para a construção do metamodelo, utilizaram-se os dados gerados pelo planejamento experimental e as respostas obtidas por meio do experimento computacional como entrada para o método de Kriging, onde todos os distúrbios foram utilizados para construção do metamodelo. A Tabela 9 apresenta os tipos de técnicas estatísticas empregadas na geração dos metamodelos.

**Tabela 9** – Técnicas estatísticas utilizadas para cada parâmetro na geração do metamodelo.

Variável	Metamodelo
Age	OLS com feature selection
FMratio	OLS com feature selection
COD_eff	OLS com feature selection
TSS_eff	OLS com feature selection
TN_eff	GPR com ard squared exponential covariance function
Snh_eff	GPR com ard matern52 covariance function
Sno_eff	GPR com ard matern32 covariance function
BOD5_eff	OLS com feature selection
oci	OLS com feature selection

Fonte: Elaborado pela autora.

O planejamento experimental com GRP, OLS e *Feature Selection* envolve a coleta de dados do modelo rigoroso no *Simulink* e o uso desses dados como entrada para criar metamodelos preditivos. No caso do GRP e do OLS, essas técnicas utilizam algoritmos estatísticos para ajustar os dados e criar um modelo que possa ser usado para fazer previsões com base nas entradas fornecidas. Já o *Feature Selection* é uma técnica utilizada para selecionar as melhores variáveis de entrada para a construção do metamodelo, eliminando aquelas que possam não ser relevantes para a predição.

O planejamento realizado com GRP, OLS e *Feature Selection* também pode acarretar erros de predição, principalmente quando a distribuição dos pontos amostrais não é uniforme. Esses erros podem ocorrer devido à aleatoriedade inerente ao processo e à falta de representatividade dos dados de treinamento em relação às situações reais que o modelo deve prever. Portanto, é importante avaliar cuidadosamente os resultados obtidos por meio dessas técnicas de planejamento experimental e considerar a possibilidade de ajustes e melhorias no modelo e na estratégia de coleta de dados para minimizar os erros de predição e aumentar a confiabilidade dos metamodelos gerados.

Para validar o metamodelo, utilizou-se a técnica de validação cruzada (*cross validation*), comparando os valores de saída da planta obtidos na simulação em *Simulink*, com os valores de saída preditos pelo metamodelo Kriging. A partir dos dados gerados, foram obtidos diferentes resultados do modelo rigoroso, representando as variáveis de qualidade do efluente tratado. A Tabela 10 apresenta os valores obtidos para cada variável da corrente de alimentação do modelo BSM1.

**Tabela 10** – Variáveis de distúrbios aplicadas ao processo.

Distúrbios	Q	S <sub>I</sub>	S <sub>S</sub>	X <sub>I</sub>	X <sub>S</sub>	X <sub>B,H</sub>	S <sub>NH</sub>	S <sub>ND</sub>	X <sub>ND</sub>
Dist. 1	17301.5	33.8	65.2	48.9	219.3	26.9	35.6	7.0	11.8
Dist. 2	19198.9	26.0	59.7	45.4	230.2	32.3	26.7	6.3	9.0
Dist. 3	22078.8	34.7	78.1	42.6	177.4	25.5	32.6	5.8	9.8
Dist. 4	19865.4	28.7	67.0	53.1	213.0	27.1	31.6	6.8	10.3
Dist. 5	16912.9	29.2	76.2	51.8	186.8	26.0	36.0	7.6	10.5
Dist. 6	20626.5	26.6	73.8	55.0	183.8	28.3	33.0	8.1	12.3
Dist. 7	20860.8	33.4	61.6	49.9	216.1	29.4	30.1	7.9	11.3
Dist. 8	18937.1	24.1	60.2	56.7	235.0	33.4	29.2	5.6	12.0
Dist. 9	14819	35.4	68.1	50.7	163.6	24.1	27.7	6.5	10.8
Dist. 10	21105.8	25.6	83.4	60.0	210.4	24.8	37.6	6.2	12.6
Dist. 11	20027.7	28.2	73.1	59.2	166.7	31.5	28.0	6.5	8.6
Dist. 12	18271.6	30.9	77.2	47.2	179.6	28.9	26.1	8.2	10.9
Dist. 13	15602.1	24.8	70.1	55.4	172.1	24.7	30.7	6.9	9.4
Dist. 14	17354.7	30.4	66.6	46.8	225.5	32.0	34.9	7.4	10.0
Dist. 15	16414.8	31.5	71.4	57.7	241.7	33.0	34.2	7.6	11.6
Dist. 16	16119.9	35.3	63.9	44.3	196.3	23.0	34.0	6.0	9.1
Dist. 17	21474.9	31.8	79.8	60.5	205.4	27.9	25.8	8.0	11.0
Dist. 18	18727.7	32.7	56.0	43.1	234.5	23.1	31.0	5.8	9.7
Dist. 19	15408.2	29.7	81.3	41.6	201.6	30.8	37.0	7.4	12.3
Dist. 20	17905.6	27.5	57.5	54.1	193.2	30.1	28.8	7.1	8.8

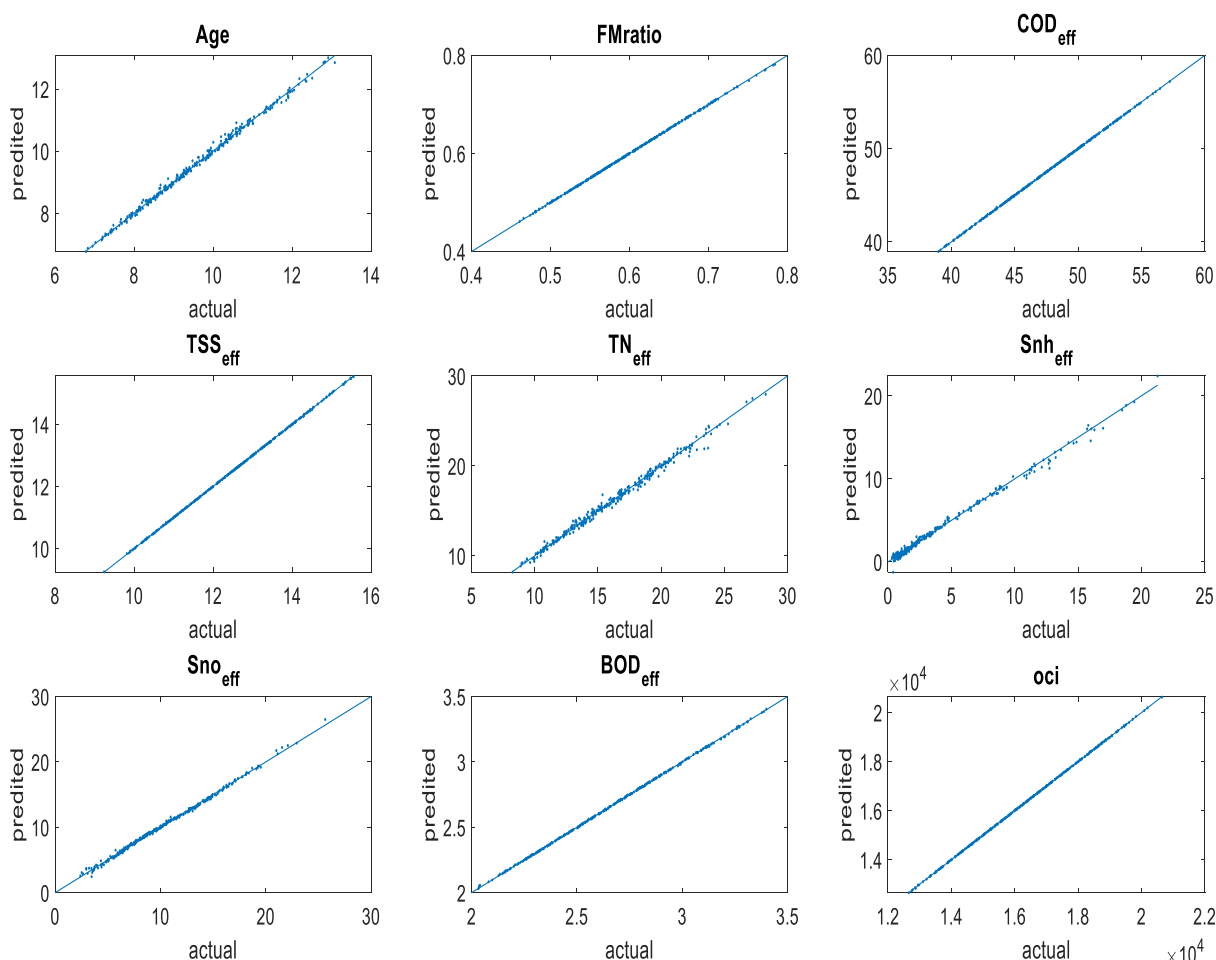
Fonte: Elaborado pela autora.

Os resultados foram analisados por meio de gráficos de dispersão comparando as variáveis de resposta preditas pelo metamodelo Kriging com as variáveis de resposta do modelo rigoroso. Através dessa análise, observou-se uma notável similaridade no comportamento das curvas após a obtenção dos resultados.

Na Figura 11, é apresentado o ajuste dos dados obtidos pelo metamodelo Kriging, utilizando as correlações e regressões mencionadas anteriormente, em conjunto com os dados de resposta do modelo rigoroso. Cada gráfico corresponde a uma das restrições (Age, FMratio, COD, TSS, TN, SNH, SNO, BOD) e a função objetivo (OCI) definida para o modelo BSM1.

Essa análise permitiu verificar a eficácia do metamodelo em representar o comportamento das variáveis de interesse em relação ao modelo rigoroso, validando assim sua aplicabilidade na resolução do problema de otimização proposto.

**Figura 11** - Resultado da metamodelagem a partir de dados teste.



Fonte: Elaborado pela autora.

Observa-se que os metamodelos apresentam um bom ajuste para a predição das variáveis. No entanto, é possível notar um erro de predição maior ao avaliar os parâmetros que foram tratados com a técnica de aprendizado de máquina GRP, este comportamento pode ser

explicado devido a sua complexidade, maior flexibilidade e tamanho reduzido do conjunto de dados de treinamento, visto que o OLS e a *Feature Selection* são abordagens mais paramétricas, reduzindo o risco de *overfitting*.

## 5.2 Otimização do processo

Os resultados da seção de otimização foram obtidos através da avaliação do desempenho de diferentes *solvers* utilizados para otimizar o modelo BSM1. O critério de avaliação utilizado foi a função objetivo OCI (*Objective Cost Index*), que relaciona os custos operacionais de todas as unidades presentes na planta de tratamento de águas residuais, sujeita às restrições de concentração dos componentes envolvidos. A Tabela 11 apresenta os valores da função objetivo para os diferentes distúrbios (dist) gerados, comparando-o a performance do modelo com os dados obtidos da simulação em *Simulink* com os dados obtidos através da otimização. Essa tabela permite a análise e comparação do desempenho do modelo em diferentes cenários.

**Tabela 11** - Comparação da performance entre os valores da função OCI dos respectivos otimizadores.

Distúrbios	Função objetivo da otimização	Função objetivo da simulação	Variação	Otimizador
<b>Dist. 1</b>	14222	14243	-0.15%	ipopt
<b>Dist. 2</b>	15395	15393	0.01%	filtersd
<b>Dist. 3</b>	16121	16135	-0.09%	matlab
<b>Dist. 4</b>	16170	16174	-0.02%	filtersd
<b>Dist. 5</b>	13475	13486	-0.08%	ipopt
<b>Dist. 6</b>	16455	16458	-0.02%	filtersd
<b>Dist. 7</b>	16821	16832	-0.07%	matlab
<b>Dist. 8</b>	16320	16337	-0.10%	matlab
<b>Dist. 9</b>	11373	11359	0.12%	filtersd
<b>Dist. 10</b>	18405	18363	0.23%	filtersd
<b>Dist. 11</b>	15387	15393	-0.04%	filtersd
<b>Dist. 12</b>	13838	13851	-0.09%	nomad
<b>Dist. 13</b>	12304	12288	0.13%	nomad
<b>Dist. 14</b>	14422	14439	-0.12%	filtersd
<b>Dist. 15</b>	14921	14959	-0.25%	ipopt
<b>Dist. 16</b>	12329	12320	0.07%	nomad
<b>Dist. 17</b>	18052	18011	0.23%	filtersd
<b>Dist. 18</b>	14655	14662	-0.05%	filtersd
<b>Dist. 19</b>	12505	12535	-0.24%	ipopt
<b>Dist. 20</b>	13958	13957	0.01%	nomad

Fonte: Elaborado pela autora.

A tabela apresenta uma comparação entre os valores da função custo dos respectivos otimizadores em diferentes distúrbios. A análise desses resultados revela as diferenças percentuais entre a função custo obtida a partir da otimização e a função custo obtida por meio da simulação.

Observa-se que a maioria dos otimizadores apresenta diferenças mínimas entre a função custo da otimização e a função custo da simulação. Essas diferenças são próximas de zero ou inferiores a 0,3%, indicando que os otimizadores estão obtendo resultados semelhantes aos esperados pela simulação.

Avaliando as diferenças obtidas para os distúrbios com o mesmo tipo de otimizador, nota-se que os distúrbios 10 e 17 mostram uma diferença de 0,23% para o otimizador “filtersd”. Essas diferenças indicam que o otimizador não conseguiu alcançar resultados tão próximos da simulação nesses casos específicos.

Analisando os otimizadores individualmente, observa-se que alguns apresentam um desempenho mais consistente em termos da função custo. Por exemplo, o otimizador “matlab” mostra diferenças mínimas em vários distúrbios, enquanto o otimizador “nomad” apresenta diferenças mais significativas em alguns distúrbios. Essas diferenças de desempenho podem estar relacionadas às abordagens algorítmicas utilizadas pelos otimizadores e à capacidade de convergência para resultados ótimos.

Em geral, a comparação dos valores da função custo dos otimizadores com as simulações fornece insights sobre a eficácia dessas técnicas de otimização em processos de tratamento de efluentes. Embora a maioria dos otimizadores tenha produzido resultados semelhantes aos da simulação, é importante considerar que o desempenho pode variar dependendo das características específicas do problema e dos algoritmos utilizados.

A otimização teve como objetivo a minimização da função custo OCI, avaliando o comportamento de cada solver para o distúrbio gerado e buscando encontrar os valores ótimos para as cinco restrições do processo: concentração de nitrogênio total (NT), demanda química de oxigênio (COD), concentração de amônia (SNH), sólidos suspensos totais (TSS) e demanda bioquímica de oxigênio (BOD). Utilizando os scripts definidos no ambiente do *Matlab*®, o processo de otimização foi executado, buscando obter os valores ótimos para as restrições mencionadas, garantindo uma concentração adequada dos componentes envolvidos no tratamento de águas residuais.

No Anexo 3 é possível observar todos os resultados obtidos para as variáveis manipuladas, função objetivo e tempo de processamento para otimização dos metamodelos. A

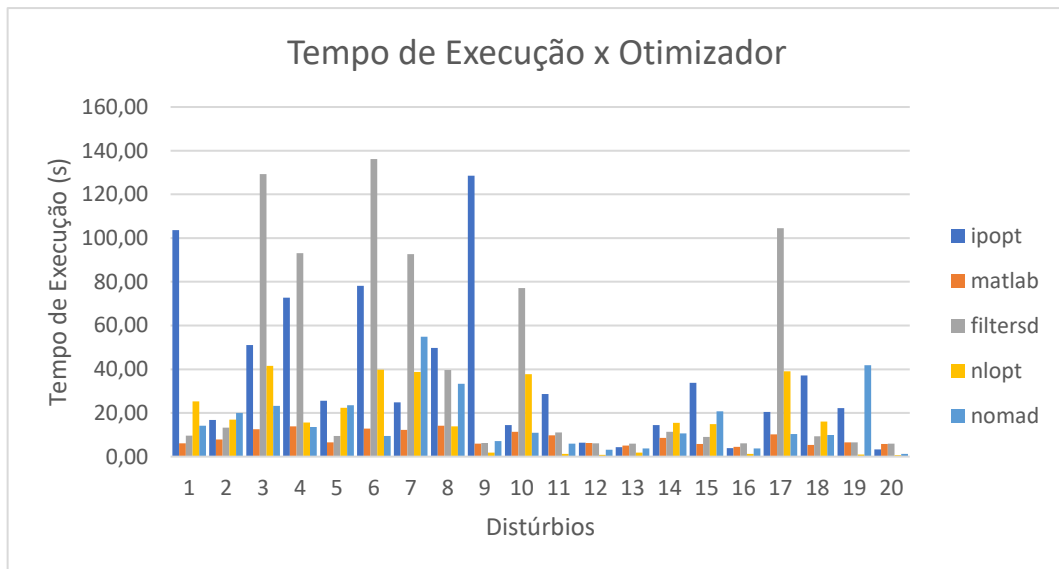


partir desses resultados, foi possível realizar uma análise comparativa da eficiência dos diferentes *solvers*, bem como avaliar a capacidade de generalização dos metamodelos gerados via kriging, sendo possível identificar possíveis melhorias no modelo e nas estratégias de otimização, visando o aprimoramento do modelo BSM1.

A coleta de dados para os resultados das otimizações é realizada para os diferentes distúrbios, usando os otimizadores: *ipopt*, *matlab*, *filtersd*, *nlopt* e *nomad*. Os resultados estão organizados em métricas, como tempo de execução e diferentes critérios de qualidade, como índice de custo operacional (OCI) e limites para as variáveis, já apresentados anteriormente.

Na Figura 12 encontra-se a relação do tempo de execução do otimizador para controle do distúrbio.

**Figura 12** - Tempo de Execução em relação a cada otimizador por distúrbio.

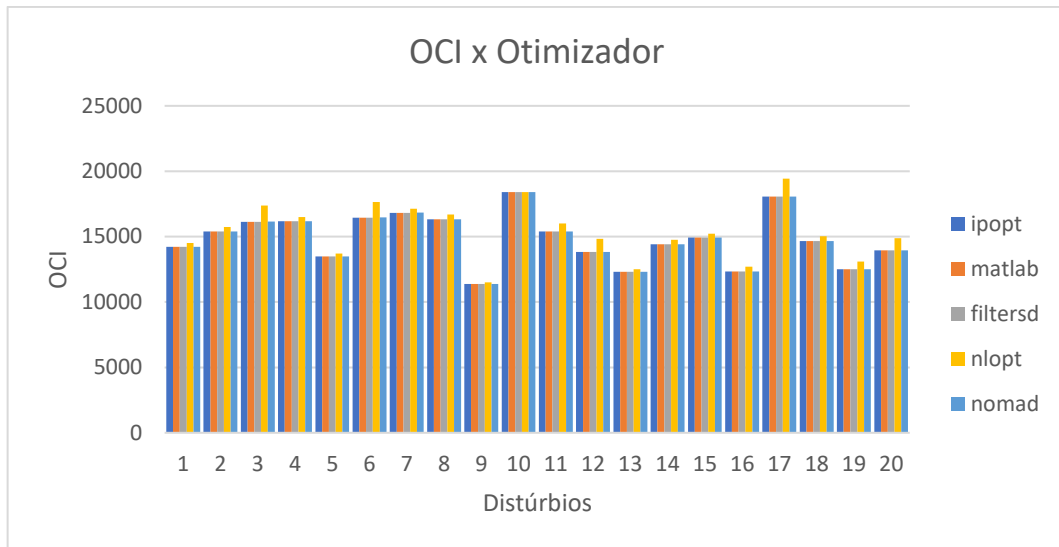


Fonte: Elaborado pela autora.

Nota-se que o otimizador "matlab" geralmente possui o menor tempo de execução em suas otimizações, seguido pelo "filtersd". O "nomad" e o "nlopt" apresentam tempos intermediários, enquanto o "ipopt" tem o tempo mais longo na maioria dos casos. Essa informação é importante para a escolha do otimizador, pois um tempo de execução mais curto pode ser preferível a depender da sua aplicação.

Na Figura 13 encontra-se a relação do OCI para cada distúrbio realizado.

**Figura 13** - OCI em relação a cada otimizador por distúrbio.

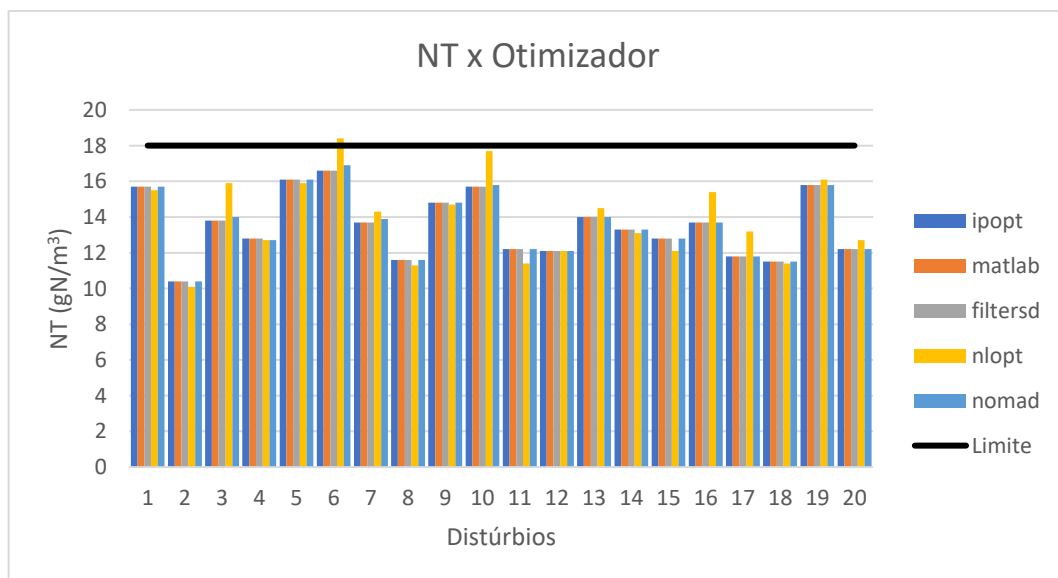


Fonte: Elaborado pela autora.

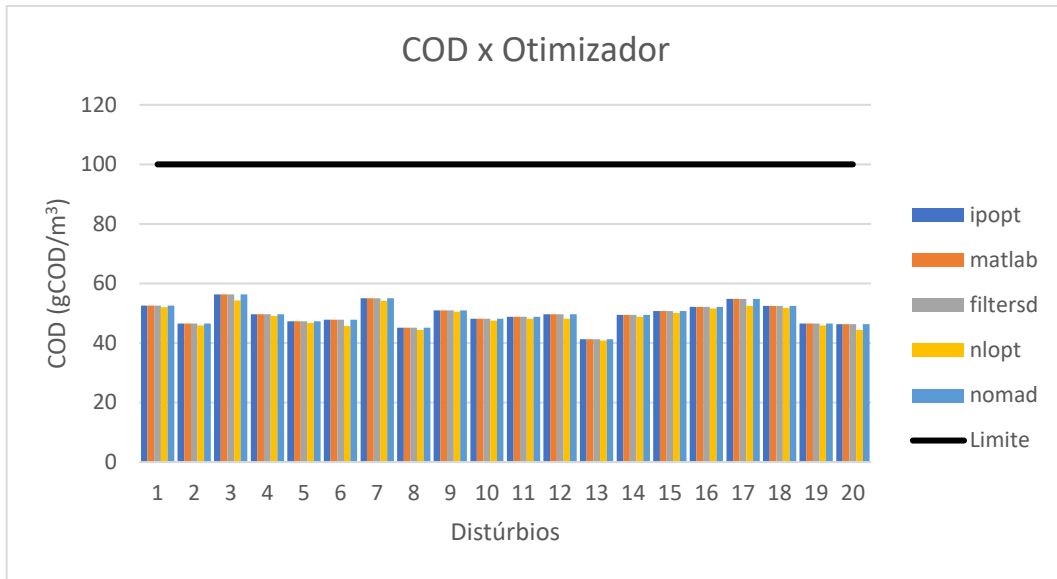
O índice de custo operacional (OCI) é usado como um critério de qualidade para a otimização. A maioria dos otimizadores atingiram valores de OCI semelhantes em todas as otimizações, sem ocorrer muita variação entre eles nesse aspecto. Isso sugere que os algoritmos utilizados pelos otimizadores não tiveram um impacto significativo na eficiência geral do tratamento.

Na Figura 14 encontra-se a relação das variáveis de qualidade do efluente para cada distúrbio realizado.

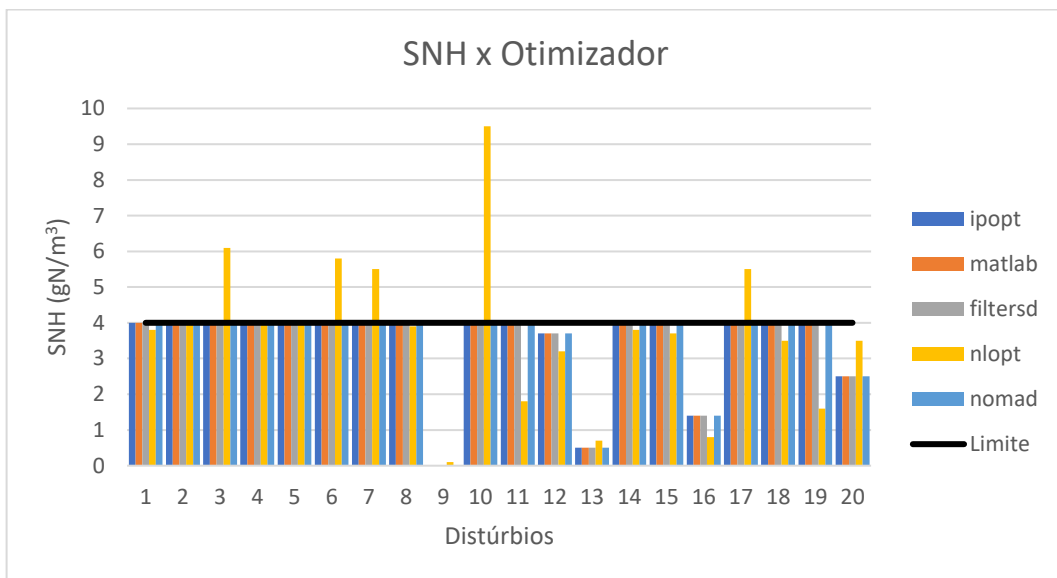
**Figura 14** - Variáveis de qualidade do efluente em relação a cada otimizador por distúrbio a) Nitrogênio total; b) Demanda química de oxigênio, c) Concentração de amônia, d) Sólidos suspensos totais e e) Demanda bioquímica de oxigênio.



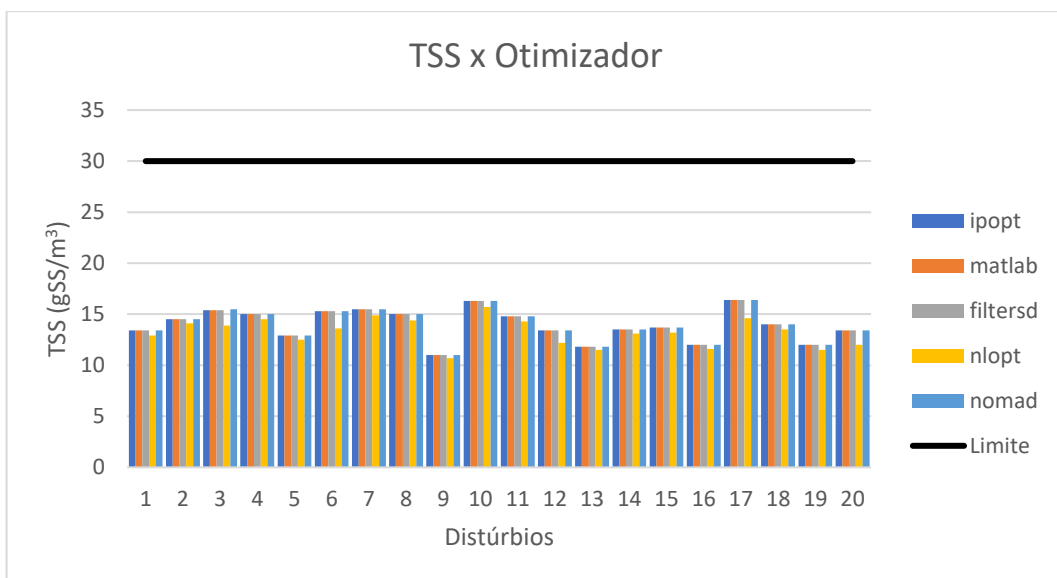
(a)



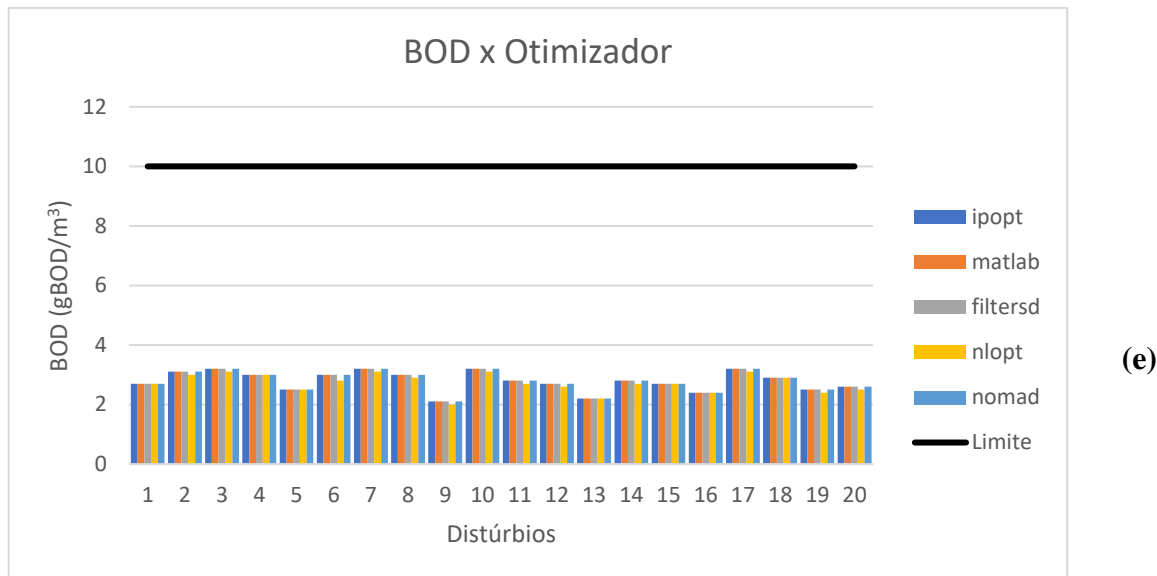
(b)



(c)



(d)



Fonte: Elaborado pela autora.

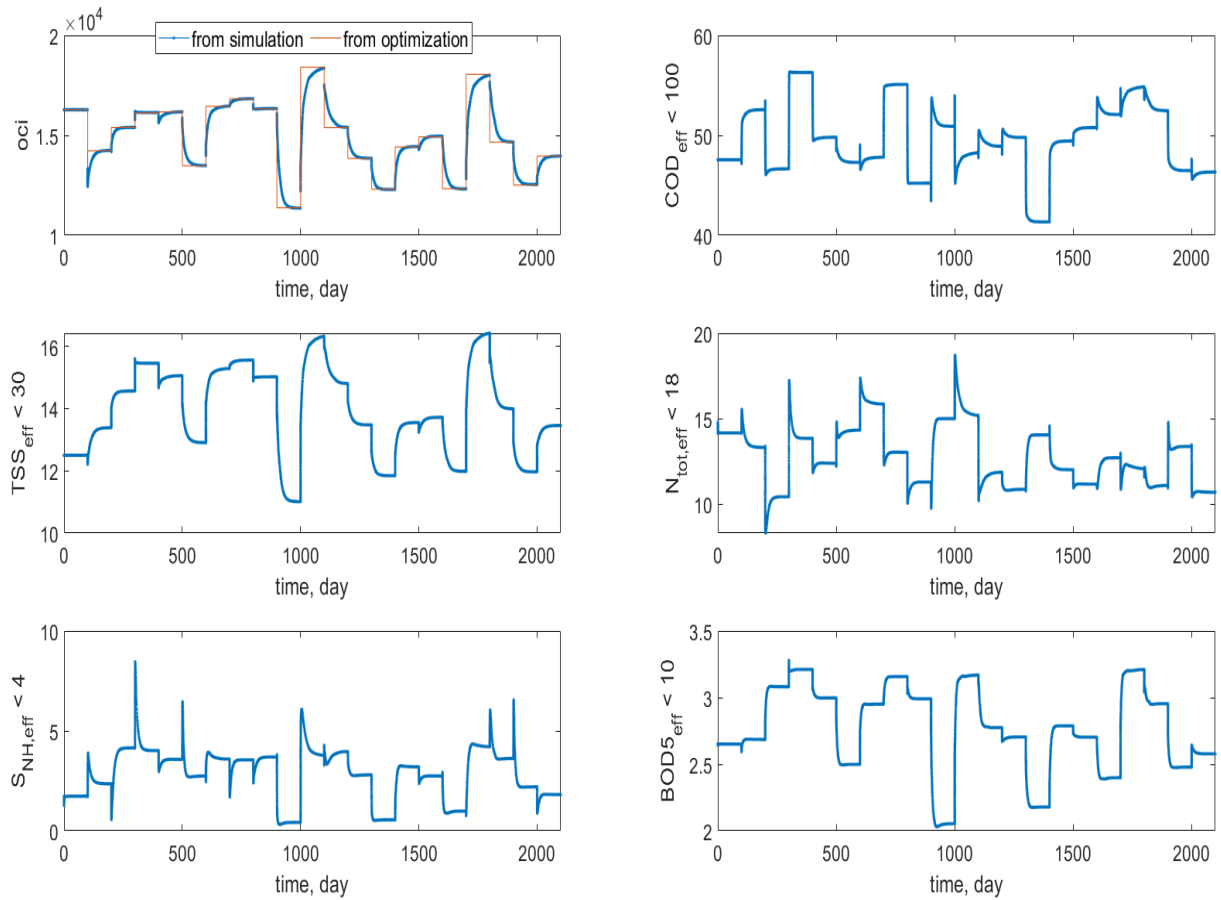
Os limites para as variáveis de manipulação variam dependendo do distúrbio e do otimizador usado, onde:

- NT: A maioria dos otimizadores conseguiram manter esta variável dentro do limite permitido, exceto o otimizador "nlopt" no distúrbio 6;
- COD: Todos os otimizadores apresentaram resultados semelhantes, mantendo o COD abaixo de 100 em todas as simulações;
- SNH: A maioria dos otimizadores conseguiram manter esta variável dentro do limite permitido, exceto o otimizador "nlopt" nos distúrbios 3, 7, 10 e 17;
- TSS: os otimizadores conseguiram manter a variável abaixo de 30 em todas as simulações;
- BOD: Todos os otimizadores conseguiram manter a variável abaixo de 10 em todas as simulações.

No geral, os otimizadores tendem a encontrar valores dentro dos limites especificados para a maioria das otimizações, exceto o otimizador "nlopt" que apresentou variações em relação a variável NT e SNH. Isso pode indicar problemas na otimização para esses casos específicos ou pode ser atribuído a restrições mais rigorosas para essas variáveis.

A Figura 15 apresenta a evolução das principais variáveis do RTO para uma maior visibilidade de seu comportamento.

**Figura 15** - Evolução das principais variáveis do RTO.

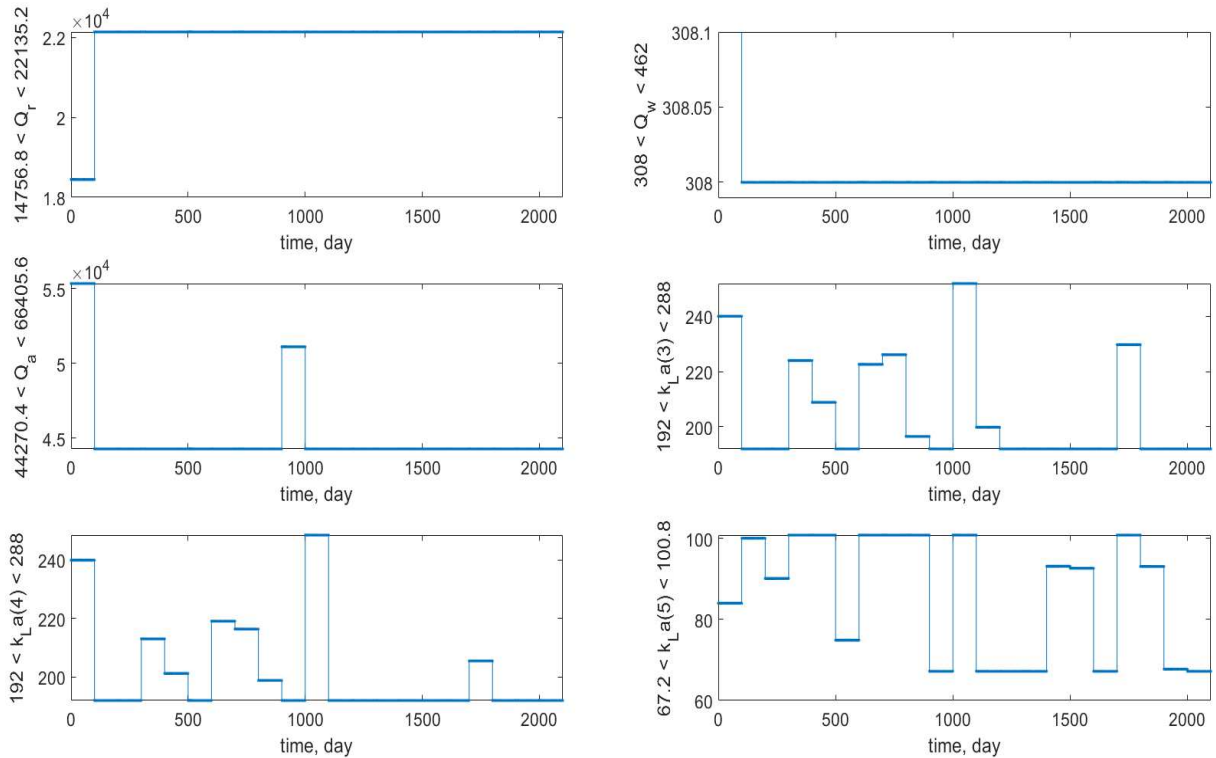


Fonte: Elaborado pela autora.

É possível visualizar como as variáveis de interesse (eficiências de remoção) variam ao longo do tempo e como os valores ótimos obtidos na otimização se comparam aos valores da simulação. Nota-se que a função objetivo se mantém próximo ao determinado pelo otimizador, já as restrições em NT e SNH não são violadas em estado estacionário, violações dinâmicas inevitáveis são de curta duração.

A Figura 16 apresenta o comportamento das variáveis manipuladas do RTO ( $Q_r$ ,  $Q_w$ ,  $Q_a$ ,  $k_{la3}$ ,  $k_{la4}$ ,  $k_{la5}$ ), sendo possível observar como as MV's variam ao longo do tempo, seguindo as decisões ótimas encontradas na otimização.

**Figura 16 - Comportamento das variáveis manipuladas do RTO.**



Fonte: Elaborado pela autora.

Cada gráfico mostra a evolução do valor da manipulação específica ao longo do tempo, notando-se padrões e tendências entre os valores das manipulações, observa-se também que as variáveis  $Q_r$ ,  $Q_a$  e  $Q_w$  são restrições ativas nos respectivos limites e podem ser mantidas constantes independentemente do distúrbio.

Analisando as otimizações realizadas, seu impacto nos valores da função objetivo e no atendimento das restrições, considerou-se as medidas de tempo de execução e os valores da função objetivo (OCI) para cada otimizador em diferentes distúrbios, a fim de definirmos o otimizador que teve o melhor desempenho durante a otimização do modelo BSM1.

#### 1. Ipopt:

- Tempo de execução (s): 103.7
- Resultados em geral estão próximos dos valores desejados para as restrições e da função objetivo.
- Desempenho inferior em relação ao tempo de execução.

#### 2. Matlab:

- Tempo de execução (s): 6.1

- Resultados em geral estão próximos dos valores desejados para as restrições e da função objetivo.
- Apresentou bom desempenho em relação ao tempo de execução.

### 3. Filtersd:

- Tempo de execução (s): 9.6
- Alguns resultados não atingiram os valores desejados para as restrições.
- Apresentou bom desempenho em relação ao tempo de execução.

### 4. Nlopt:

- Tempo de execução (s): 25.3
- Alguns resultados não atingiram os valores desejados para as restrições.
- Desempenho inferior em relação ao tempo de execução.

### 5. Nomad:

- Tempo de execução (s): 14.1
- Resultados em geral estão próximos dos valores desejados para as restrições e da função objetivo
- Desempenho superior em relação ao tempo de execução.

Para um maior entendimento dos critérios, dividiu-se as análises em três categorias: Tempo de execução, Valor da Função Objetivo (OCI) e Controle das Manipulações.

→ Tempo de Execução:

- O otimizador "matlab" foi o mais rápido em todas as iterações, seguido pelo "filtersd".
- O otimizador "nomad" e "nlopt" tiveram tempos de execução intermediários.
- O otimizador "ipopt" foi o mais lento em todas as iterações.

→ Valor da Função Objetivo (OCI):

- Em algumas iterações, diferentes otimizadores tiveram resultados semelhantes em termos de OCI.

- Na iteração 3, o otimizador "nomad" teve um valor muito maior do que os outros, indicando um desempenho inferior.
- Na iteração 4, o otimizador "ipopt" teve uma grande melhoria no OCI, tornando-se o melhor desempenho entre os otimizadores.
- Nas demais iterações, os otimizadores "matlab" e "filtersd" geralmente tiveram valores de OCI semelhantes, indicando um bom desempenho.

→ Controle das Manipulações:

- Analisando os valores das variáveis manipuladas em cada iteração, pode-se observar que os otimizadores geralmente atendem às restrições.
- Em algumas iterações, o otimizador "nomad" resultou em valores muito altos ou baixos para algumas variáveis manipuladas, indicando que ele pode não ter controlado bem as manipulações em determinados momentos.

No geral, os otimizadores "matlab", "filtersd" e "ipopt" apresentaram resultados satisfatórios em relação aos valores da função objetivo e ao atendimento das restrições. O otimizador "nomad" teve desempenho inferior em algumas restrições, enquanto o "nlopt" teve desempenho inferior.



## 6. CONCLUSÃO

A pesquisa realizada sobre a otimização em tempo real de uma planta de tratamento de efluentes utilizando técnicas de aprendizado de máquina, em conjunto com o modelo BSM1 e metamodelos kriging, proporcionou uma compreensão abrangente do desempenho dos otimizadores nas simulações de processos de tratamento de efluentes. Os resultados obtidos comprovaram a eficácia do kriging na geração de metamodelos, evidenciando sua confiabilidade em relação às variáveis-respostas do modelo.

A análise dos otimizadores "matlab", "filtersd" e "ipop" demonstrou resultados satisfatórios tanto na função objetivo quanto no atendimento das restrições, tornando-os opções viáveis para futuros projetos de otimização em tratamento de efluentes. O desempenho inferior dos otimizadores "nomad" e "nlopt" em algumas restrições não os descarta completamente, mas sinaliza a necessidade de considerar suas limitações e peculiaridades.

A utilização dos otimizadores permitiu avaliar o desempenho de cada um e compará-los com os dados da simulação realizada no Simulink, fornecendo informações valiosas para orientar a seleção adequada de otimizadores em projetos futuros e otimizar a eficácia dos processos de tratamento de efluentes.

Além disso, o trabalho também destacou a vantagem dos modelos reduzidos em substituição ao modelo rigoroso BSM1, pois os metamodelos kriging alcançaram alto desempenho ao minimizar o esforço computacional. A devida atenção às restrições de processo em todos os casos estudados reforça a aplicabilidade e a relevância da abordagem de otimização RTO utilizada, que apresentou resultados satisfatórios ao longo da pesquisa.

Em suma, os resultados fornecem *insights* importantes para o avanço das técnicas de otimização em tempo real de plantas de tratamento de efluentes, com o potencial de aprimorar a eficiência operacional e contribuir para a preservação ambiental. No entanto, é importante ressaltar que, como em qualquer pesquisa, existem limitações que podem ser abordadas em trabalhos futuros, como a inclusão de outros otimizadores e a consideração de cenários mais complexos para uma compreensão ainda mais abrangente desse processo.

## REFERÊNCIAS BIBLIOGRÁFICAS

ALEX, J., BENEDETTI, L., COPP, J., GERNAEY, K. V., JEPSSON, U., NOPENS, I., PONS, N., RIEGER, L., ROSEN, C., STEYER, J. P., VANROLLEGHEM, P., WINKLER, S. “Benchmark Simulation Model no. 1 (BSM1),” Department of Industrial Electrical Engineering and Automation, Lund University, Tech. Rep. CODEN: LUTEDX/(TEIE-7229)/1-62/(2008), 2008.

AUDET, C., DIGABEL, S. L., MONTPLAISIR, V. R., & TRIBES, C. NOMAD version 4: Nonlinear optimization with the MADS algorithm. **arXiv preprint arXiv:2104.11627**, 2021.

BASSIN, J.P., DEZOTTI, M., 2008, Tratamento primário, secundário e terciário de efluentes. In: DEZOTTI M., Processos e Técnicas para o Controle Ambiental de Efluentes Líquidos, Rio de Janeiro: E-Papers. (“Rep. USP - Detalhe do registro: Ozonização e processos oxidativos avançados”).

BRITTON, A. ET AL. Pilot-scale struvite recovery from anaerobic digester supernatant at an enhanced biological phosphorus removal wastewater treatment plant. **Journal of Environmental Engineering and Science**, London, v. 4, p. 265-277, 2005.

CALVO, A. **Integração de otimização em tempo real (RTO) e controle preditivo de modelo (MPC) de um divisor industrial de propeno/propano**. 2015. Tese de Doutorado, Universidade de São Paulo, Brasil, 2015.

CHILES, J.P.; Delfiner, P. (2012). **Geostatistics: Modeling Spatial Uncertainty**. Wiley.

COPP, J. B. The COST simulation benchmark: Description and simulator manual (a product of COST Action 624 & Action 682). (“Simulator Manual | PDF”) Technical report, Office for official publications of the European Communities, Luxembourg, 2001b.

CRESSIE, N. A. C. Statistics for Spatial Data, Revised Edition. **Wiley interscience**. 1993.

DEHLENDORFF, C. **Monte Carlo Analysis**. Technical University of Denmark. 2010.

DOMINGOS, S. **Água Instituto Carbono Brasil, Desenvolvimento Sustentável, Ecossistema**. Disponível em < <http://www.institutocarbonobrasil.org.br/ecossistemas/agua> >, acesso em 02 de janeiro de 2022.

FORRESTER, A., SOBESTER, A., KEANE, A. **Engineering Design via Surrogate Modelling: A Practical Guide.**, John Wiley & Sons, 2008.

GERNAEY, K. V., LOOSDRECHT, M. C. M. V., HENZE, M., LIND, M., JORGENSEN, S. B., 2004, “Activated Sludge Wastewater Treatment Plant Modelling and Simulation: State of the Art, Review”, **Environmental Modeling & Software**, v. 19, pp. 763-783.

GILBERT, David. **OPTI Toolbox: A free Matlab toolbox for optimization.** 2016.

GOMES, M. V. C. **Otimização Sequencial por Aproximações – Uma aplicação em tempo real para o refino do petróleo.** 2007. Tese de Doutorado. D. Sc. Thesis.

HAMMER M. J.; HAMMER, JR. M. J. Water and Wastewater Technology, 5th edition, Prentice Hall of India, New Delhi, p. 540, 2006.

HAQ, A.; SRINIVASAN, B.; BONVIN, D. Real-Time Optimization of Wastewater Treatment Plants via Constraint Adaptation. *Processes* 2022, 10, 990. <https://doi.org/10.3390/pr10050990>

HENZE, M., LESLIE GRADY, C. P., GUJER, W., MARIS, G. V. R., MATSUO, T. Activated Sludge Process Model No. 1. Scientific and Technical Report 1, IAWQ, London, UK, 1987

HENZE, M.; GUJER, W.; VAN LOOSDRECHT, M. C. M. Activated sludge models ASM1, ASM2 and ASM3. **IWA Scientific and Technical Report** No. 9, Londres, 2000.

HENZE, M.; GUJER, W.; VAN LOOSDRECHT, M. C. M. Activated sludge models ASM1, ASM2 and ASM3. **IWA Scientific and Technical Report** No. 9, Londres, 2002.

JEPPSSON, U. Modelling aspects of wastewater treatment processes. PhD thesis, IEA, Lund Institute of Technology, Lund, Sweden, 1996.

JEPPSSON, U., NOPENS, I., BENNEDETTI, L., PONS, M.N., ALEX, J., COOP, J., GERNAEY, K.V., ROSEN, C., STEYER, J.P., VANROLLEGHEM, P. Benchmark simulation model No 2. Department of Industrial Electrical Engineering and Automation, Lund University, Tech. Rep. 2011

JORDAAN, E. M.; ACKERMAN, J.; CICEK, N. Phosphorus removal from anaerobically digested swine wastewater through struvite precipitation. **Water Science and Technology**, Oxford, v. 66, n. 12, p. 3228-3234, 2010.

KOSZMA, Isadora Inez Alves. **Modelagem Matemática e Análise de Biorreatores de Lodo Ativo para o Tratamento de Efluentes Líquidos**. São Cristóvão, 2016. 106 p. : il. Trabalho Acadêmico Orientado. Centro de Ciências Exatas e Tecnologia, Universidade Federal de Sergipe, São Cristóvão.

LAW, A. M., & KELTON, W. D. Simulation Modeling and Analysis. **McGraw-Hill Education**, 2014.

LEU, H., LEE, C., OUYANG, C. F., 1998 “Effects of Organic Matter on the Conversion Rates of Nitrogenous Compounds in a Channel”, *Water Research*, v. 32, pp. 0–8.

LEVY, ANA LUCIA L. Modelagem e Análise de Sensibilidade do Processo de Tratamento de Lodo Ativado com Reciclo, Dissertação (Mestre em Engenharia Química) – Universidade Federal do Rio de Janeiro, COPPE, 2007.

LOPHAVEN, S.N., NIELSEN, H.B., SONDERGAARD, J. 2002, DACE A MATLAB Kriging Toolbox, **Technical Report IMM-TR-2002-12**, version 2.0, Technical University of Denmark, Denmark, 2002.

LUYBEN, W. L. Process Modeling, Simulation, and Control for Chemical Engineers. **McGraw-Hill Education**, 2017.

LYARA, D. G. P.; PESTANA, M. **Apostila de Tratamento de Efluentes**. 69 f. Centro Federal de Educação Tecnológica – CEFET/BA, 2007.

MATHERON, G. (1963). Principles of Geostatistics. **Economic Geology**, 58(8), 1246-1266.

**MATHWORKS**. 2022. "Global Optimization." Disponível em: <<https://web.archive.org/web/20171125150100/https://www.mathworks.com/products/global-optimization.html>>. Acesso em: 12 de Agosto de 2023.

MCKAY, M. D., BECKMAN, R. J. & CONOVER, W. J. A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code. **Technometrics**, Vol. 21, No. 2, pp. 239-245., 1979.

METCALF, L, EDDY, H. P. Tratamento de Efluentes e Recuperação de Recursos. McGraw Hill Brasil, 5ed. 2015

MULAS, M. Modelling and Control of Activated Sludge Processes. Tese de Dsc. Università degli Studi di Cagliari. Cagliari, Itália, 2006.

OISSON, A.; SANDBERG, G.; DAHLBLOM, O. On Latun hypercube sampling for structural reliability analysis. **Structural Safety**. Vol. 25, no. 1, pag. 47-68, 2003.

PALMER, K., REALFF, M., 2002, "Optimization and Validation of Steady-state Flowsheet Simulation Metamodels", **Trans IChemE**, v. 80, Part A, pp. 773-782.

PINTO, J. W. O., "Uso da otimização sequencial aproximada a problemas Uni e multiobjectivos de gerenciamento de reservatórios". **Dissertação** (Universidade Federal de Pernambuco) Recife, PE, Brasil, 2014.

RAJESHWARI, K. V.; BALAKRISHNAN, M.; KANSAL, A.; LATA, K.; KISHORE, V. V. N. State-of-the-Art of Anaerobic Digestion Technology for Industrial Wastewater Treatment. ("The effect of various decomposers on quality of organic fertilizer ...") **Renew. Sustain. Energy Rev.** 2000, 4, 135-156.

RODRIGUES, S. C. G., "Projeto de Estrutura de Controle Aplicado a Sistema de Tratamento de Efluentes" 148 f, Tese (Doutorado em Engenharia Química) - UFCG, Campina Grande, 2013.

SACKS, J., WELCH, J.W., MITCHELL, T.J. ET AL., 1989, "Design and Analysis of Computer Experiments", **Statistical Science**, v.4, pp. 409-435.

SANTOS, K. R. M. Técnicas de amostragem inteligente em simulação Monte Carlo. **Dissertação**. São Carlos. Universidade de São Paulo. 2014.

SIQUEIRA, L. A. **Água fonte de vida**. Monografia (Graduação), UTFPR, Medianeira – PR, 2011.

SPERLING, Marcos Von. "Introdução à qualidade das águas e ao tratamento de esgotos". 2 ed. Departamento de Engenharia Sanitária e Ambiental – DESA, Belo Horizonte, Universidade Federam de Minas Gerais, vol. 1, 2005 a. 243 p.

STEVEN, G. Johnson. **The NLOpt nonlinear-optimization package**, 2016.

TAKACS, I.; PATRY, G. G.; NOLASCO, D. A dynamic model of the clarification-thickening process. **Water Science Technology**, v-25 (10), p. 1263-1271, 1991.

TEMPLETON, M. R; BUTLER, D. Introduction to wastewater treatment. Londres: Bookboon, 2011.

VERMA, S. **Anaerobic digestion of biodegradable organics in municipal solid wastes**. ("Enhanced Anaerobic Digestion of Organic Waste") 2012. Master of Science Degree, Columbia University, 2012.

VIANA, F. A. C., VENTER, G. and BALABANOV, V. "An Algorithm for fast Optimal Latin Hypercube Design of Experiments" **International Journal for Numerical Methods in Engineering**, Int. J. Numer. Meth. Engng 2010; 82: 135-156, 2010.

VILAR, DAMIRIS VALESKA FARIAS. **Aplicação de filtros de areia agregado a biomassa e ao carvão ativado no tratamento de águas para reuso**. Monografia (Graduação em Engenharia Química) – Universidade Federal de Campina Grande, Centro de Ciências e Tecnologia, 2019.

VILLAR, S.B.B., DAS NEVES, T.G., DA COSTA, A.B., DA SILVA, S.K., MULAS, M., NETO., A.T.P., DOS ANJOS, D.A., ARAÚJO, A.C.B. Otimização Utilizando metamodelo kriging: Uma aplicação à separação de propeno por destilação. CILAMCE 2016, **Proceedings of the XXXVII Iberian Latin-American Congress on Computational Methods in Engineering**. Brasília, DF, Brasil, novembro 6-9, 2016.

VITASOVIC, Z. Z. **An integrated control strategy for the activated sludge process**. Tese de Dsc, Rice University, Houston, Texas. 1986.

VON SPERLING, M., 1996, "Introdução à Qualidade das Águas e ao Tratamento de Esgotos", **Princípios do Tratamento Biológico de Águas Residuárias**, 2. ed., Belo Horizonte: UFMG.

WÄCHTER, Andreas; BIEGLER, Lorenz T. "On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming," *Mathematical Programming*. v. 106, n. 1, pág. 25-57, 2006.

WIESMANN, U.; CHOI, I. S.; DOMBROWSKI, E. M. Fundamentals of biological wastewater treatment. **Weinheim: Willey-VCH Verlag GmbH & Co.**, 355p, 2007.

ZOBY JR, L. C, NETO, L. M. P, DALPANE, T.F, PAIVA, T. M. N, da SILVA, M. C. L, MOTTA SOBRINHO, M. A. Avaliação do Desempenho das Estações de Tratamento por Lodo ativo no Grande Recife Associada com Análise de Imagens dos Microrganismos Presentes no Lodo ativo, XX Congresso Brasileiro de Engenharia Química. Florianópolis, 2014.

## ANEXOS

### Anexo 1 – Script “rto\_app.m”

```
clearvars; clc; close all; format compact
load resultsSimulink % the trained 'mdl' model (and all its
properties as a struct) is here (check the .mat file for variable
names)
load states % x = nominal state variables values in dataset format
load dv % dv = nominal manipulated variables values, dv = [(1) Q;(2)
Si;(3) Ss;(4) Xi;(5) Xs;(6) Xbh;(7) Xba;(8) Xp;(9) So;(10) Sno;(11)
Snh;(12) Snd;(13) Xnd;(14) Salk]
load mv % mv = nominal disturbance variables values, mv = [(1)
Qr;(2) Qw;(3) Qa;(4) kla1;(5) kla2;(6) kla3;(7) kla4;(8) kla5]

% training Kriging metamodels
load regressionGP % regressionGP. Used when metamodel is obtained by
brute-force Kriging and not Thalita's algorithm

% setting up GPML toolbox
addpath('gpml-matlab-v4.2-2018-06-11') % add GPML toolbox to Matlab
path
run('startup.m') % start up GPML toolbox

% disturbances: Xba = 0, Xp = 0, So = 0, Sno = 0, and Salk = 7 are
not considered, so d = [Q; Si; Ss; Xi; Xs; Xbh; Snh; Snd; Xnd]
d      = dv([1:6 11:13])';
dmin   = 0.80*d;
dmax   = 1.20*d;

% manipulations: kla1 = 0 and kla2 = 0 are not considered, so u =
[Qr; Qw; Qa; kla3; kla4; kla5]
u      = mv([1:3 6:8])';
umin   = 0.80*u;
umax   = 1.20*u;

% disturbance data
% Nd = 20;
% d_ = lhsdesign(Nd,size(dmin(:),1)); % generating input data with
X \in [0,1]
% d_ = dmin' + (dmax' - dmin').*d_; % scaling X \in [xmin, xmax]
load disturbances % d_
Nd = size(d_,1);

% optimization
solver = {'ipopt', 'matlab', 'filtersd', 'nlopt', 'nomad'}; %
'ipopt', 'matlab' and 'filtersd' are local optimizers while 'nlopt'
and 'nomad' are global optimizers
u0 = u;
tic
for k = 1:Nd
    fprintf('optimizing for disturbance %d of %d\n',k,Nd)
    dist = d_(k,:); % applying disturbance
```



```

[u_(:, :, k), nonlcon(:, :, k), fOpt_metamodel(k, :), exitflag(k, :), elapse_(
k, :)] =
optimizer_gpr_app_(dist(:), u0(:), umin, umax, regressionGP, solver); %
Note that nonlcon(constraint, solver, disturbance), where nonlcon =
[TN_eff, COD_eff, Snh_eff, TSS_eff, BOD_eff, oci]
    elapse(k) = toc;
end

% reports !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[fOpt_metamodelSort, I] = sort(fOpt_metamodel, 2, 'descend');
solverMatrix = repmat(solver, Nd, 1);
for k = 1:size(I, 1)
    elapseSort(k, :) = elapse_(k, I(k, :));
    solverMatrixSort(k, :) = solverMatrix(k, I(k, :));
end

% best solutions
nonlcon_ = nonlcon;
nonlcon_(end, :, :) = -nonlcon_(end, :, :); % the last constraint is of
the form c1 - c(x) <= 0
nonlcon_ = [18; 100; 4; 30; 10; 0] - nonlcon_; % matrix of
constraints of the form c(x) - cu or c1 - c(x)
ind_violation = [];
for k = 1:Nd
    ind = all(nonlcon_(:, :, k) > 0); % detecting constraint violation
    if all(ind == 0)
        ind_violation = [ind_violation k];
        continue
    end
    u_Aux = u_(:, ind, k);
    nonlconAux = nonlcon(:, ind, k);
    fOpt_metamodelAux = fOpt_metamodel(k, ind);
    exitflagAux = exitflag(k, ind);
    elapse_Aux = elapse_(k, ind);
    solverAux = solver(ind);

    kOpt = find(fOpt_metamodelAux == min(fOpt_metamodelAux));
    % in case more than one maximum pops up, which has already
happened before, pick the "fastest optimizer"
    if size(kOpt, 1) > 1
        aux = elapse_Aux(kOpt);
        aux = aux == min(aux); % picking the "fastest optimizer"...
        kOpt = kOpt(aux);
        kOpt = kOpt(1); % in case more than one pops up...
    end
    uOpt(:, k) = u_Aux(:, kOpt); % uOpt = [Qr; Qw; Qa; kla3;
kla4; kla5]
    nonlconOpt(:, k) = nonlconAux(:, kOpt); % nonlconOpt = [TN_eff,
COD_eff, Snh_eff, TSS_eff, BOD_eff, oci]
    fOpt(k) = fOpt_metamodelAux(kOpt);
    exitflagOpt(k) = exitflagAux(kOpt);
    elapseOpt(k) = elapse_Aux(kOpt);
    solverOpt{k} = solverAux{kOpt};
end
end

```

```

% simulations
x_initial = x; % initial states to pass to Simulink model
sysSim = 'BSM_open';
load_system(sysSim)

% running nominal steady-state for some time t0
t0 = 100;
dt = 100;
set_param(sysSim, 'StartTime', '0', 'StopTime', num2str(t0))
% disturbance variables: Xba, Xp, So, Sno, and Salk are not
considered, so Xba = 0, Xp = 0, So = 0, Sno = 0, and Salk = 7 and d
= [Q; Si; Ss; Xi; Xs; Xbh; Snh; Snd; Xnd]
Q      = d(1);
Si     = d(2);
Ss     = d(3);
Xi     = d(4);
Xs     = d(5);
Xbh    = d(6);
Xba    = 0;
Xp     = 0;
So     = 0;
Sno    = 0;
Snh    = d(7);
Snd    = d(8);
Xnd    = d(9);
Salk   = 7;
% manipulated variables: kla1 and kla2 are not considered, so kla1 =
0 and kla2 = 0 and u = [Qr; Qw; Qa; kla3; kla4; kla5]
Qr     = u(1);
Qw     = u(2);
Qa     = u(3);
kla1   = 0; % anoxic reactor
kla2   = 0; % anoxic reactor
kla3   = u(4); % aerobic reactor
kla4   = u(5); % aerobic reactor
kla5   = u(6); % aerobic reactor

simOut = sim(sysSim, 'ReturnWorkspaceOutputs', 'on');
y = simOut.ASM_output.Data; % y = [Age; FMratio; COD_eff; TSS_eff;
TN_eff; Snh_eff; Sno_eff; BOD_eff; oci]
oci = y(end,end);
cod_eff = y(end,3);
tss_eff = y(end,4);
tn_eff = y(end,5);
snh_eff = y(end,6);
bod5_eff = y(end,8);
t = simOut.tout;
y_oci = repmat(oci, size(t(:),1),1); % value of objective function
from nominal for plotting purposes
y_cod = repmat(cod_eff, size(t(:),1),1); % value of constraint from
nominal for plotting purposes
y_tss = repmat(tss_eff, size(t(:),1),1); % value of constraint from
nominal for plotting purposes

```

```

y_tn = repmat(tn_eff,size(t(:),1),1); % value of constraint from
nominal for plotting purposes
y_snh = repmat(snh_eff,size(t(:),1),1); % value of constraint from
nominal for plotting purposes
y_bod5 = repmat(bod5_eff,size(t(:),1),1); % value of constraint from
nominal for plotting purposes

u_qr = repmat(Qr,size(t(:),1),1); % value of manipulation from
nominal for plotting purposes
u_qw = repmat(Qw,size(t(:),1),1); % value of manipulation from
nominal for plotting purposes
u_qa = repmat(Qa,size(t(:),1),1); % value of manipulation from
nominal for plotting purposes
u_kla3 = repmat(kla3,size(t(:),1),1); % value of manipulation from
nominal for plotting purposes
u_kla4 = repmat(kla4,size(t(:),1),1); % value of manipulation from
nominal for plotting purposes
u_kla5 = repmat(kla5,size(t(:),1),1); % value of manipulation from
nominal for plotting purposes

t1 = t0;
no_violation = setdiff(1:Nd,ind_violation);
for k = no_violation
    d = d_(k,:); % applying disturbance
    u = uOpt(:,k); % optimal inputs
    % disturbance variables: Xba, Xp, So, Sno, and Salk are not
considered, so Xba = 0, Xp = 0, So = 0, Sno = 0, and Salk = 7 and d
= [Q; Si; Ss; Xi; Xs; Xbh; Snh; Snd; Xnd]
    Q = d(1);
    Si = d(2);
    Ss = d(3);
    Xi = d(4);
    Xs = d(5);
    Xbh = d(6);
    Xba = 0;
    Xp = 0;
    So = 0;
    Sno = 0;
    Snh = d(7);
    Snd = d(8);
    Xnd = d(9);
    Salk = 7;
    % manipulated variables: kla1 and kla2 are not considered, so
kla1 = 0 and kla2 = 0 and u = [Qr; Qw; Qa; kla3; kla4; kla5]
    Qr = u(1);
    Qw = u(2);
    Qa = u(3);
    kla1 = 0; % anoxic reactor
    kla2 = 0; % anoxic reactor
    kla3 = u(4); % aerobic reactor
    kla4 = u(5); % aerobic reactor
    kla5 = u(6); % aerobic reactor

t2 = t1 + dt;
set_param(sysSim,'StartTime',num2str(t1),'StopTime',num2str(t2))

```

```

set_param(sysSim, 'LoadInitialState', 'on', 'InitialState', 'simOut.xFinal');
simOut =
sim(sysSim, 'SrcWorkspace', 'current', 'ReturnWorkspaceOutputs', 'on');
z = simOut.ASM_output.Data; % z = [Age; FMratio; COD_eff;
TSS_eff; TN_eff; Snh_eff; Sno_eff; BOD_eff; oci]
oci = z(:,end);
fSim(k) = oci(end);
t = [t; simOut.tout];
y = [y; z];
y_oci = [y_oci; repmat(fOpt(k), size(simOut.tout(:),1),1)]; %
value of objective function from optimization for plotting purposes
% nonlconOpt = [TN_eff, COD_eff, Snh_eff, TSS_eff, BOD_eff, oci]
y_tn = [y_tn;
repmat(nonlconOpt(1,k), size(simOut.tout(:),1),1)]; % value of
constraint from optimization for plotting purposes
y_cod = [y_cod;
repmat(nonlconOpt(2,k), size(simOut.tout(:),1),1)]; % value of
constraint from optimization for plotting purposes
y_snh = [y_snh;
repmat(nonlconOpt(3,k), size(simOut.tout(:),1),1)]; % value of
constraint from optimization for plotting purposes
y_tss = [y_tss;
repmat(nonlconOpt(4,k), size(simOut.tout(:),1),1)]; % value of
constraint from optimization for plotting purposes
y_bod5 = [y_bod5;
repmat(nonlconOpt(5,k), size(simOut.tout(:),1),1)]; % value of
constraint from optimization for plotting purposes
% u = [Qr; Qw; Qa; kla3; kla4; kla5]
u_qr = [u_qr; repmat(Qr, size(simOut.tout(:),1),1)]; % value of
manipulation from optimization for plotting purposes
u_qw = [u_qw; repmat(Qw, size(simOut.tout(:),1),1)]; % value of
manipulation from optimization for plotting purposes
u_qa = [u_qa; repmat(Qa, size(simOut.tout(:),1),1)]; % value of
manipulation from optimization for plotting purposes
u_kla3 = [u_kla3; repmat(kla3, size(simOut.tout(:),1),1)]; %
value of manipulation from optimization for plotting purposes
u_kla4 = [u_kla4; repmat(kla4, size(simOut.tout(:),1),1)]; %
value of manipulation from optimization for plotting purposes
u_kla5 = [u_kla5; repmat(kla5, size(simOut.tout(:),1),1)]; %
value of manipulation from optimization for plotting purposes
set_param(sysSim, 'LoadInitialState', 'off', 'InitialState', '')
t1 = t2;
end
save results_rto u_nonlcon exitflag fOpt_metamodel elapse_
no_violation fOpt_metamodelSort solver solverMatrix solverMatrixSort
elapseSort uOpt nonlconOpt fOpt elapseOpt solverOpt fSim t y d_

% results
for k = 1:Nd
aux = ['Dist. ', num2str(k)];
name1{k} = aux;
end
name1 = name1(no_violation);

```

```

name2 = {'objective function from optimization','objective function
from simulation'};

tbl1 = array2table([fOpt(no_violation); fSim(no_violation)]');
tbl1.Properties.RowNames = name1;
tbl1.Properties.VariableNames = name2

tbl2 = array2table(fOpt_metamodelSort(no_violation,:));
tbl2.Properties.RowNames = name1

tbl3 = array2table(solverMatrixSort(no_violation,:));
tbl3.Properties.RowNames = name1

tbl4 = array2table(elapseSort(no_violation,:));
tbl4.Properties.RowNames = name1

tab = [];
for k = 1:Nd
    % nonlcon = [TN_eff, COD_eff, Snh_eff, TSS_eff, BOD_eff] and u =
[Qr; Qw; Qa; kla3; kla4; kla5]
    tb = [elapse_(k,:); fOpt_metamodel(k,:);
squeeze(nonlcon(1:5, :, k)); squeeze(u_(:, :, k))];
    tab = [tab; tb];
end

fSize = 16;
figure
subplot(3,2,1), plot(t,y(:,end),'.-',t,y_oci), xlim([0 t1]),
xlabel('time, day', 'FontSize', fSize), ylabel('oci', 'FontSize',
fSize), legend('from simulation', 'from optimization', 'FontSize',
fSize), ax = gca; ax.FontSize = fSize;
subplot(3,2,2), plot(t,y(:,3),'.-'), xlim([0 t1]),
xlabel('time, day', 'FontSize', fSize), ylabel('COD_{eff} < 100',
'FontSize', fSize), ax = gca; ax.FontSize = fSize;
subplot(3,2,3), plot(t,y(:,4),'.-'), xlim([0 t1]),
xlabel('time, day', 'FontSize', fSize), ylabel('TSS_{eff} < 30',
'FontSize', fSize), ax = gca; ax.FontSize = fSize;
subplot(3,2,4), plot(t,y(:,5),'.-'), xlim([0 t1]),
xlabel('time, day', 'FontSize', fSize), ylabel('N_{tot,eff} < 18',
'FontSize', fSize), ax = gca; ax.FontSize = fSize;
subplot(3,2,5), plot(t,y(:,6),'.-'), xlim([0 t1]),
xlabel('time, day', 'FontSize', fSize), ylabel('S_{NH,eff} < 4',
'FontSize', fSize), ax = gca; ax.FontSize = fSize;
subplot(3,2,6), plot(t,y(:,8),'.-'), xlim([0 t1]),
xlabel('time, day', 'FontSize', fSize), ylabel('BOD5_{eff} < 10',
'FontSize', fSize), ax = gca; ax.FontSize = fSize;

figure
subplot(3,2,1), plot(t,u_qr, '.-'), xlim([0 t1]), xlabel('time,
day', 'FontSize', fSize), ylabel([num2str(umin(1)), ' < Q_r < ',
num2str(umax(1))], 'FontSize', fSize), ax = gca; ax.FontSize =
fSize;
subplot(3,2,2), plot(t,u_qw, '.-'), xlim([0 t1]), xlabel('time,
day', 'FontSize', fSize), ylabel([num2str(umin(2)), ' < Q_w < ',

```

```

num2str(umax(2))], 'FontSize', fSize), ax = gca; ax.FontSize =
fSize; ylim([307.99, 308.1])
subplot(3,2,3), plot(t,u_qa,'.-'), xlim([0 t1]), xlabel('time,
day', 'FontSize', fSize), ylabel([num2str(umin(3)), ' < Q_a < ',
num2str(umax(3))], 'FontSize', fSize), ax = gca; ax.FontSize =
fSize;
subplot(3,2,4), plot(t,u_kla3,'.-'), xlim([0 t1]), xlabel('time,
day', 'FontSize', fSize), ylabel([num2str(umin(4)), ' < k_La(3) < ',
num2str(umax(4))], 'FontSize', fSize), ax = gca; ax.FontSize =
fSize;
subplot(3,2,5), plot(t,u_kla4,'.-'), xlim([0 t1]), xlabel('time,
day', 'FontSize', fSize), ylabel([num2str(umin(5)), ' < k_La(4) < ',
num2str(umax(5))], 'FontSize', fSize), ax = gca; ax.FontSize =
fSize;
subplot(3,2,6), plot(t,u_kla5,'.-'), xlim([0 t1]), xlabel('time,
day', 'FontSize', fSize), ylabel([num2str(umin(6)), ' < k_La(5) < ',
num2str(umax(6))], 'FontSize', fSize), ax = gca; ax.FontSize =
fSize;

close_system(sysSim)

```

## Anexo 2 – Script “optimizer\_grp\_app.m”

```

function [u,nonlcon,fOpt,exitflag,elapse] =
optimizer_gpr_app(d,mdl,u0,umin,umax,parX,parY,par,solver)
% computes optimal values of manipulations
% mdl = {(1)Age; (2)FMratio; (3)COD_eff; (4)TSS_eff; (5)TN_eff;
(6)Snh_eff; (7)Sno_eff; (8)BOD_eff; (9)oci}
% solver = {'ipopt', 'matlab', 'filtersd', 'nlopt', 'nomad'}

% bound constraints (lb <= x <= ub)
lb = [d; umin];
ub = [d; umax];
% nonlinear constraints (cl <= nlcon(x) <= cu). Constraints from J.
Alex, L. Benedetti, J. Copp, K.V. Gernaey, U. Jeppsson, I. Nopens,
M.-N. Pons, L. Rieger, C. Rosen, J.P. Steyer, P. Vanrolleghem, S.
Winkler. Benchmark Simulation Model no. 1 (BSM1), p. 17
nlcon = @(z) [myPredict(z(:)', [],mdl{5},parX,parY,par); % 0 <=
TN_eff <= 18
myPredict(z(:)', [],mdl{3},parX,parY,par); % 0 <=
COD_eff <= 100
myPredict(z(:)', [],mdl{6},parX,parY,par); % 0 <=
Snh_eff <= 4
myPredict(z(:)', [],mdl{4},parX,parY,par); % 0 <=
TSS_eff <= 30
myPredict(z(:)', [],mdl{8},parX,parY,par); % 0 <=
BOD_eff <= 10
myPredict(z(:)', [],mdl{9},parX,parY,par)]; % 0 <=
oci, ensuring the objective function is always positive!!!

cl = [0; 0; 0; 0; 0; 0];

```

```

cu = [18; 100; 4; 30; 10; Inf];

% opti optimization
fun = @(z) myPredict(z(:)', [], mdl{end}, parX, parY, par); % the "z(:)"
argument is to avoid errors. This is the objective function to be
minimized: in this case the last element of mdl, i.e., oci
x0 = [d; u0]';
elapse_gmatlab = [];
for k = 1:length(solver)
    tic
    fprintf('      optimizing using %s...', solver{k})
    switch solver{k}
        case 'nlopt'
            options =
optiset('solver', solver{k}, 'solverOpts', nloptset('algorithm', 'AUGLAG
'), 'maxiter', 750);
            opt =
opti('fun', fun, 'sense', 1, 'nl', nlcon, cl, cu, 'bounds', lb, ub, 'options', o
ptions); % set 'sense' = 1 for minimization
            [x(:,k), fOpt(k), exitflag(k)] = solve(opt, x0);
            nonlcon(:,k) = nlcon(x(:,k));
        otherwise
            options = optiset('solver', solver{k}, 'maxiter', 750);
            opt =
opti('fun', fun, 'sense', 1, 'nl', nlcon, cl, cu, 'bounds', lb, ub, 'options', o
ptions); % set 'sense' = 1 for minimization
            [x(:,k), fOpt(k), exitflag(k)] = solve(opt, x0);
            nonlcon(:,k) = nlcon(x(:,k));
    end
    elapse(k) = toc;
    elapse_gmatlab = [elapse_gmatlab elapse(k)];
    fprintf(' optimum of %6.4f found in %6.2f
seconds...\n', fOpt(k), elapse(k))
end
u = x(size(d,:), 1)+1:end, :);

```

### Anexo 3 – Resultados obtidos da otimização do processo para diferentes distúrbios.

Dist.↓	Otimizador →	ipopt	matlab	filtersd	nlopt	nomad
1	Tempo de execução (s)	103.7	6.1	9.6	25.3	14.1
	<i>OCI</i>	14222.2	14222.2	14222.2	14515.5	14222.2
	$N_{tot,effluent} < 18$	15.7	15.7	15.7	15.5	15.7
	$COD_{effluent} < 100$	52.6	52.6	52.6	52	52.6
	$S_{NH,effluent} < 4$	4	4	4	3.8	4
	$TSS_{effluent} < 30$	13.4	13.4	13.4	12.9	13.4
	$BOD5_{effluent} < 10$	2.7	2.7	2.7	2.7	2.7
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18560.7	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308

	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55290.6	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	196.5	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	205.8	192
	$67.2 \leq k_L a(5) \leq 100.8$	100	100	100	86.2	100
2	Tempo de execução (s)	16.8	7.8	13.2	16.9	20
	<i>OCI</i>	15395.1	15395.1	15395.1	15735.3	15395.2
	$N_{tot,effluent} < 18$	10.4	10.4	10.4	10.1	10.4
	$COD_{effluent} < 100$	46.5	46.5	46.5	45.9	46.5
	$S_{NH,effluent} < 4$	4	4	4	4	4
	$TSS_{effluent} < 30$	14.5	14.5	14.5	14.1	14.5
	$BOD5_{effluent} < 10$	3.1	3.1	3.1	3	3.1
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18554.5	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55234.3	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	201.1	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	193.2	192
	$67.2 \leq k_L a(5) \leq 100.8$	90.1	90.1	90.1	83.4	90.1
3	Tempo de execução (s)	51.1	12.5	129.3	41.6	23.3
	<i>OCI</i>	16121.7	16120.8	16120.8	17384.8	16143
	$N_{tot,effluent} < 18$	13.8	13.8	13.8	15.9	14
	$COD_{effluent} < 100$	56.3	56.3	56.3	54.3	56.3
	$S_{NH,effluent} < 4$	4	4	4	6.1	4
	$TSS_{effluent} < 30$	15.4	15.4	15.4	13.9	15.5
	$BOD5_{effluent} < 10$	3.2	3.2	3.2	3.1	3.2
	$14756.8 \leq Q_r \leq 22135.2$	22125.5	22135.2	22135.2	18446	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	385	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.7	44270.4	44270.4	55338	44270.4
	$192 \leq k_L a(3) \leq 288$	223.9	224	224	240	247.5
	$192 \leq k_L a(4) \leq 288$	213.2	213.1	213.1	240	193.7
	$67.2 \leq k_L a(5) \leq 100.8$	100.8	100.8	100.8	84	100.8
4	Tempo de execução (s)	72.7	13.9	93.1	15.6	13.6
	<i>OCI</i>	16172.2	16170.4	16170.4	16504.4	16174.9
	$N_{tot,effluent} < 18$	12.8	12.8	12.8	12.7	12.7
	$COD_{effluent} < 100$	49.7	49.7	49.7	49.1	49.7
	$S_{NH,effluent} < 4$	4	4	4	4	4
	$TSS_{effluent} < 30$	15	15	15	14.5	15
	$BOD5_{effluent} < 10$	3	3	3	3	3



	$14756.8 \leq Q_r \leq 22135.2$	22130.7	22135.2	22135.2	18560.1	22135.2
	$308 \leq Q_w \leq 462$	308.1	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44276.7	44270.4	44270.4	55280.9	44270.5
	$192 \leq k_L a(3) \leq 288$	209.1	208.8	208.8	198.5	198.4
	$192 \leq k_L a(4) \leq 288$	201.2	201.2	201.2	213.1	212.2
	$67.2 \leq k_L a(5) \leq 100.8$	100.7	100.8	100.8	100.3	100.8
5	Tempo de execução (s)	25.6	6.6	9.4	22.3	23.5
	<i>OCI</i>	13475.1	13475.1	13475.1	13697.7	13475.1
	$N_{tot,effluent} < 18$	16.1	16.1	16.1	15.9	16.1
	$COD_{effluent} < 100$	47.3	47.3	47.3	46.7	47.3
	$S_{NH,effluent} < 4$	4	4	4	4	4
	$TSS_{effluent} < 30$	12.9	12.9	12.9	12.5	12.9
	$BOD5_{effluent} < 10$	2.5	2.5	2.5	2.5	2.5
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18480.8	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55319	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	194.2	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	192	192
	$67.2 \leq k_L a(5) \leq 100.8$	74.9	74.9	74.9	70.3	74.9
6	Tempo de execução (s)	78.1	12.9	136.2	39.8	9.4
	<i>OCI</i>	16456.7	16454.9	16454.9	17650.3	16466.1
	$N_{tot,effluent} < 18$	16.6	16.6	16.6	18.4	16.9
	$COD_{effluent} < 100$	47.8	47.8	47.8	45.7	47.8
	$S_{NH,effluent} < 4$	4	4	4	5.8	4
	$TSS_{effluent} < 30$	15.3	15.3	15.3	13.6	15.3
	$BOD5_{effluent} < 10$	3	3	3	2.8	3
	$14756.8 \leq Q_r \leq 22135.2$	22134.4	22135.2	22135.2	18446	22135.2
	$308 \leq Q_w \leq 462$	308.1	308	308	385	308
	$44270.4 \leq Q_a \leq 66405.6$	44271.6	44270.4	44270.4	55338	44270.4
	$192 \leq k_L a(3) \leq 288$	222.7	222.6	222.6	240	239.5
	$192 \leq k_L a(4) \leq 288$	219.2	219.1	219.1	240	204.5
	$67.2 \leq k_L a(5) \leq 100.8$	100.8	100.8	100.8	84	100.8
7	Tempo de execução (s)	24.9	12.2	92.6	38.7	54.9
	<i>OCI</i>	16822.3	16820.8	16820.8	17141.9	16849.1
	$N_{tot,effluent} < 18$	13.7	13.7	13.7	14.3	13.9
	$COD_{effluent} < 100$	55	55	55	54.2	55
	$S_{NH,effluent} < 4$	4	4	4	5.5	4

	$TSS_{effluent} < 30$	15.5	15.5	15.5	14.9	15.5
	$BOD5_{effluent} < 10$	3.2	3.2	3.2	3.1	3.2
	$14756.8 \leq Q_r \leq 22135.2$	22134.4	22135.2	22135.2	18523.7	22135.2
	$308 \leq Q_w \leq 462$	308.1	308	308	315.9	308
	$44270.4 \leq Q_a \leq 66405.6$	44271.2	44270.4	44270.4	55249.2	44270.4
	$192 \leq k_L a(3) \leq 288$	226.2	226.1	226.1	219.5	255.1
	$192 \leq k_L a(4) \leq 288$	216.5	216.4	216.4	205	192.9
	$67.2 \leq k_L a(5) \leq 100.8$	100.8	100.8	100.8	100.8	100.8
<b>8</b>	Tempo de execução (s)	49.8	14.2	39.7	13.9	33.3
	<i>OCI</i>	16321.4	16319.9	16319.9	16700.2	16319.9
	$N_{tot,effluent} < 18$	11.6	11.6	11.6	11.3	11.6
	$COD_{effluent} < 100$	45.1	45.1	45.1	44.4	45.1
	$S_{NH,effluent} < 4$	4	4	4	3.9	4
	$TSS_{effluent} < 30$	15	15	15	14.4	15
	$BOD5_{effluent} < 10$	3	3	3	2.9	3
	$14756.8 \leq Q_r \leq 22135.2$	22134.6	22135.2	22135.2	18530.9	22135.2
	$308 \leq Q_w \leq 462$	308.1	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44271.1	44270.4	44270.4	55324	44270.4
	$192 \leq k_L a(3) \leq 288$	196.6	196.5	196.5	200.1	195.9
	$192 \leq k_L a(4) \leq 288$	198.9	198.9	198.9	207	199.4
	$67.2 \leq k_L a(5) \leq 100.8$	100.8	100.8	100.8	96.1	100.8
<b>9</b>	Tempo de execução (s)	128.5	6	6.2	1.8	7.1
	<i>OCI</i>	11373.2	11373.2	11373.2	11506	11373.2
	$N_{tot,effluent} < 18$	14.8	14.8	14.8	14.7	14.8
	$COD_{effluent} < 100$	50.9	50.9	50.9	50.5	50.9
	$S_{NH,effluent} < 4$	0.00E+00	0.00E+00	0.00E+00	0.1	0.00E+00
	$TSS_{effluent} < 30$	11	11	11	10.7	11
	$BOD5_{effluent} < 10$	2.1	2.1	2.1	2	2.1
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18559.3	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	51104.8	51104.8	51104.8	55281.1	51104.8
	$192 \leq k_L a(3) \leq 288$	192	192	192	192	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	192.1	192
	$67.2 \leq k_L a(5) \leq 100.8$	67.2	67.2	67.2	67.4	67.2
<b>10</b>	Tempo de execução (s)	14.4	11.4	77.1	37.8	10.9
	<i>OCI</i>	18406.8	18405	18405	18397.2	18407.9
	$N_{tot,effluent} < 18$	15.7	15.7	15.7	17.7	15.8

	$COD_{effluent} < 100$	48.2	48.2	48.2	47.5	48.2
	$S_{NH,effluent} < 4$	4	4	4	9.5	4
	$TSS_{effluent} < 30$	16.3	16.3	16.3	15.7	16.3
	$BOD5_{effluent} < 10$	3.2	3.2	3.2	3.1	3.2
	$14756.8 \leq Q_r \leq 22135.2$	22134.5	22135.2	22135.2	18542.2	22135.2
	$308 \leq Q_w \leq 462$	308.1	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44271.1	44270.4	44270.4	55339.4	44270.4
	$192 \leq k_L a(3) \leq 288$	251.9	251.9	251.9	207.3	261.2
	$192 \leq k_L a(4) \leq 288$	248.6	248.6	248.6	236.9	239.9
	$67.2 \leq k_L a(5) \leq 100.8$	100.8	100.8	100.8	100.8	100.8
<b>11</b>	Tempo de execução (s)	28.7	9.7	11.1	1.2	5.9
	<i>OCI</i>	15389	15386.8	15386.8	16002.9	15387
	$N_{tot,effluent} < 18$	12.2	12.2	12.2	11.4	12.2
	$COD_{effluent} < 100$	48.8	48.8	48.8	48.1	48.8
	$S_{NH,effluent} < 4$	4	4	4	1.8	4
	$TSS_{effluent} < 30$	14.8	14.8	14.8	14.3	14.8
	$BOD5_{effluent} < 10$	2.8	2.8	2.8	2.7	2.8
	$14756.8 \leq Q_r \leq 22135.2$	22134.4	22135.2	22135.2	18489.4	22135.2
	$308 \leq Q_w \leq 462$	308.1	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44271	44270.4	44270.4	55299.3	44270.4
	$192 \leq k_L a(3) \leq 288$	199.9	199.8	199.8	192	198.6
	$192 \leq k_L a(4) \leq 288$	192	192	192	238.9	193.3
$67.2 \leq k_L a(5) \leq 100.8$	67.2	67.2	67.2	77.3	67.2	
<b>12</b>	Tempo de execução (s)	6.4	6.3	6.1	0.8	3.1
	<i>OCI</i>	13838.5	13838.5	13838.5	14828.3	13838.5
	$N_{tot,effluent} < 18$	12.1	12.1	12.1	12.1	12.1
	$COD_{effluent} < 100$	49.7	49.7	49.7	48.1	49.7
	$S_{NH,effluent} < 4$	3.7	3.7	3.7	3.2	3.7
	$TSS_{effluent} < 30$	13.4	13.4	13.4	12.2	13.4
	$BOD5_{effluent} < 10$	2.7	2.7	2.7	2.6	2.7
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18514.8	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	373.1	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55287.7	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	218	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	192	192
$67.2 \leq k_L a(5) \leq 100.8$	67.2	67.2	67.2	67.2	67.2	
<b>13</b>	Tempo de execução (s)	4.4	5	5.9	1.9	3.7

	<i>OCI</i>	12304.5	12304.5	12304.5	12517	12304.5
	$N_{tot,effluent} < 18$	14	14	14	14.5	14
	$COD_{effluent} < 100$	41.3	41.3	41.3	40.8	41.3
	$S_{NH,effluent} < 4$	0.5	0.5	0.5	0.7	0.5
	$TSS_{effluent} < 30$	11.8	11.8	11.8	11.5	11.8
	$BOD5_{effluent} < 10$	2.2	2.2	2.2	2.2	2.2
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18509.1	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55302.7	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	193.2	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	194.4	192
	$67.2 \leq k_L a(5) \leq 100.8$	67.2	67.2	67.2	68.2	67.2
<b>14</b>	Tempo de execução (s)	14.5	8.6	11.3	15.5	10.7
	<i>OCI</i>	14422.4	14422.4	14422.4	14748.7	14422.4
	$N_{tot,effluent} < 18$	13.3	13.3	13.3	13.1	13.3
	$COD_{effluent} < 100$	49.4	49.4	49.4	48.8	49.4
	$S_{NH,effluent} < 4$	4	4	4	3.8	4
	$TSS_{effluent} < 30$	13.5	13.5	13.5	13.1	13.5
	$BOD5_{effluent} < 10$	2.8	2.8	2.8	2.7	2.8
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18516.5	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55284.8	44270.9
	$192 \leq k_L a(3) \leq 288$	192	192	192	210.1	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	196.4	192
	$67.2 \leq k_L a(5) \leq 100.8$	93.1	93.1	93.1	78.1	93.1
<b>15</b>	Tempo de execução (s)	33.8	5.8	9	14.9	20.8
	<i>OCI</i>	14921.4	14921.4	14921.4	15231.1	14921.4
	$N_{tot,effluent} < 18$	12.8	12.8	12.8	12.1	12.8
	$COD_{effluent} < 100$	50.7	50.7	50.7	50.1	50.7
	$S_{NH,effluent} < 4$	4	4	4	3.7	4
	$TSS_{effluent} < 30$	13.7	13.7	13.7	13.2	13.7
	$BOD5_{effluent} < 10$	2.7	2.7	2.7	2.7	2.7
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18515	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55233.3	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	197.9	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	201.9	192

	$67.2 \leq k_L a(5) \leq 100.8$	92.6	92.6	92.6	80.9	92.6
<b>16</b>	Tempo de execução (s)	3.9	4.5	6.1	1.2	3.8
	<i>OCI</i>	12329.4	12329.4	12329.4	12701.1	12329.4
	$N_{tot,effluent} < 18$	13.7	13.7	13.7	15.4	13.7
	$COD_{effluent} < 100$	52.1	52.1	52.1	51.6	52.1
	$S_{NH,effluent} < 4$	1.4	1.4	1.4	0.8	1.4
	$TSS_{effluent} < 30$	12	12	12	11.6	12
	$BOD5_{effluent} < 10$	2.4	2.4	2.4	2.4	2.4
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18480.1	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55286.9	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	192.3	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	192	192
	$67.2 \leq k_L a(5) \leq 100.8$	67.2	67.2	67.2	94.5	67.2
<b>17</b>	Tempo de execução (s)	20.4	10.2	104.6	39	10.3
	<i>OCI</i>	18053.6	18052	18052	19441.5	18052.9
	$N_{tot,effluent} < 18$	11.8	11.8	11.8	13.2	11.8
	$COD_{effluent} < 100$	54.8	54.8	54.8	52.5	54.8
	$S_{NH,effluent} < 4$	4	4	4	5.5	4
	$TSS_{effluent} < 30$	16.4	16.4	16.4	14.6	16.4
	$BOD5_{effluent} < 10$	3.2	3.2	3.2	3.1	3.2
	$14756.8 \leq Q_r \leq 22135.2$	22134.7	22135.2	22135.2	18446	22135.2
	$308 \leq Q_w \leq 462$	308.1	308	308	385	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.8	44270.4	44270.4	55338	44270.4
	$192 \leq k_L a(3) \leq 288$	229.8	229.7	229.7	240	234.6
	$192 \leq k_L a(4) \leq 288$	205.6	205.5	205.5	240	201.1
	$67.2 \leq k_L a(5) \leq 100.8$	100.8	100.8	100.8	84	100.8
<b>18</b>	Tempo de execução (s)	37.1	5.3	9.3	16.1	9.9
	<i>OCI</i>	14655.2	14655.2	14655.2	15039	14655.2
	$N_{tot,effluent} < 18$	11.5	11.5	11.5	11.4	11.5
	$COD_{effluent} < 100$	52.4	52.4	52.4	51.8	52.4
	$S_{NH,effluent} < 4$	4	4	4	3.5	4
	$TSS_{effluent} < 30$	14	14	14	13.5	14
	$BOD5_{effluent} < 10$	2.9	2.9	2.9	2.9	2.9
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18582.6	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	308	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55254.9	44270.4

	$192 \leq k_L a(3) \leq 288$	192	192	192	202.4	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	203.5	192
	$67.2 \leq k_L a(5) \leq 100.8$	93.1	93.1	93.1	85.5	93.1
<b>19</b>	Tempo de execução (s)	22.2	6.5	6.6	1	41.9
	<i>OCI</i>	12504.8	12504.8	12504.8	13088.9	12504.8
	$N_{tot,effluent} < 18$	15.8	15.8	15.8	16.1	15.8
	$COD_{effluent} < 100$	46.5	46.5	46.5	45.9	46.5
	$S_{NH,effluent} < 4$	4	4	4	1.6	4
	$TSS_{effluent} < 30$	12	12	12	11.5	12
	$BOD5_{effluent} < 10$	2.5	2.5	2.5	2.4	2.5
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18543.9	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	314.7	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55278.1	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	238.7	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	192.8	192
	$67.2 \leq k_L a(5) \leq 100.8$	67.7	67.7	67.7	67.2	67.7
<b>20</b>	Tempo de execução (s)	3.3	5.8	5.9	0.7	1.2
	<i>OCI</i>	13958.2	13958.2	13958.2	14874.4	13958.2
	$N_{tot,effluent} < 18$	12.2	12.2	12.2	12.7	12.2
	$COD_{effluent} < 100$	46.3	46.3	46.3	44.4	46.3
	$S_{NH,effluent} < 4$	2.5	2.5	2.5	3.5	2.5
	$TSS_{effluent} < 30$	13.4	13.4	13.4	12	13.4
	$BOD5_{effluent} < 10$	2.6	2.6	2.6	2.5	2.6
	$14756.8 \leq Q_r \leq 22135.2$	22135.2	22135.2	22135.2	18582.6	22135.2
	$308 \leq Q_w \leq 462$	308	308	308	387.6	308
	$44270.4 \leq Q_a \leq 66405.6$	44270.4	44270.4	44270.4	55336.6	44270.4
	$192 \leq k_L a(3) \leq 288$	192	192	192	196.1	192
	$192 \leq k_L a(4) \leq 288$	192	192	192	192	192
	$67.2 \leq k_L a(5) \leq 100.8$	67.2	67.2	67.2	67.2	67.2