

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO

Otimização Baseada em Metamodelos: Uma Abordagem para
Metamodelagem em Simulação a Eventos Discretos

João Victor Soares do Amaral

Itajubá, fevereiro de 2021

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO

João Victor Soares do Amaral

**Otimização Baseada em Metamodelos: Uma Abordagem para
Metamodelagem em Simulação a Eventos Discretos**

Dissertação submetida ao programa de Pós-Graduação em Engenharia de Produção como parte dos requisitos para obtenção do Título de Mestre em Ciências em Engenharia de Produção.

Área: Engenharia de Produção

Orientador: Prof. Dr. José Arnaldo Barra Montevechi

Coorientador: Prof. Dr. Rafael de Carvalho Miranda

Fevereiro de 2021

Itajubá

DEDICATÓRIA

A Deus, que nos dá o dom da vida, aos meus pais, Maria das Mercês e Juscelino, por todo apoio e ensinamentos, à minha namorada, Rafaela, e em especial ao meu querido avô, João Ribeiro, que sempre esteve ao meu lado.

AGRADECIMENTOS

A Deus, nunca terei como agradecer por todas as maravilhosas oportunidades que sempre colocou no meu caminho. O senhor é o meu pastor e nada me faltará.

Aos meus pais, Maria das Mercês Mota Soares e Juscelino da Silva Amaral, por todo apoio, amor e ensinamentos dados durante minha vida.

A minha namorada, Rafaela, por todo amor e compreensão nos momentos de ausência e por sempre me apoiar e incentivar a ir mais longe. Ao lado de um grande homem sempre há uma grande mulher.

A minha querida avó Efigênia, por todo carinho e amor durante toda minha vida, e ao meu avô Francisco, que aí do céu sempre olha por nós.

A minha avó Maria, por sempre cuidar, amar e orar por nós, mesmo hoje sendo uma estrelinha eu sei que sempre está ao meu lado.

Em especial, gostaria de agradecer ao meu avô João Ribeiro, ao qual convivi por poucos anos, que foram fundamentais para me tornar o homem que sou. Obrigado por sempre estar comigo, festejando nas vitórias e me carregando no seu colo nas dificuldades.

Ao meu padraсто Eugênio, por toda amizade e estímulo ao longo de meus estudos.

Ao meu orientador, José Arnaldo Barra Montevechi, pela orientação deste trabalho e me ensinar não apenas sobre engenharia, mas também que ser professor vai muito além de ter um título. É ter ética, ser justo e priorizar o conhecimento acima de qualquer coisa.

Ao meu co-orientador, Rafael de Carvalho Miranda, pelas contribuições valiosas a este trabalho e por sempre estar disposto a me ajudar e ensinar. Contigo aprendi que a engenharia não se resume apenas em otimizar processos, mas também serve como uma ferramenta valiosa para tentar melhorar o bem estar das pessoas.

A todos professores e servidores da UNIFEI, em especial aos professores Renato Lima, Carlos Melo, José Henrique, Edson Pamplona, Pedro Paulo Balestrassi, Carlos Sanches, André Medeiros, Alexandre Pinho, Anderson Paulo, Fabiano Leal, José Antônio de Queiroz e todos outros que pacientemente transmitiram seus conhecimentos.

Aos professores Tábata Fernandes Pereira e Aneirson Francisco da Silva pelas contribuições a esta pesquisa.

Aos amigos da UNIFEI, em especial Gustavo Teodoro, Carlos Henrique, Afonso, Alyson, Wesley, Mona Liza, Fernanda, Tanita, Milena, Flávio, Anderson Lino, Gustavo Leal, Laryssa, Breno e Bruno.

Aos amigos da Belge Engenharia e Sistemas, em especial Marcelo, Alain, Tiago, Lucas, Luiz, Ruano, Henrique, Ana, Fábio e Emilson.

A CAPES, FAPEMIG e CNPq pelo apoio e incentivo à pesquisa, em especial a essa.

A Universidade Federal de Itajubá, a quem sempre terei uma dívida eterna de gratidão.

A todos vocês, meu muito obrigado!

EPÍGRAFE

“A verdadeira medida de um homem não é como ele se comporta em momentos de conforto e conveniência, mas como ele se mantém em tempos de controvérsia e desafio.”

Martin Luther King Jr.

RESUMO

No contexto da indústria 4.0, a otimização via simulação (OvS) surge como uma das mais potentes ferramentas da indústria moderna, permitindo aos decisores alocarem seus recursos de forma mais assertiva. Todavia, em sistemas muito complexos, o uso de técnicas convencionais de OvS demandam um tempo computacional que, muitas vezes, inviabiliza sua aplicação. Nos últimos anos, o desenvolvimento na área de *machine learning* surgiram algoritmos com alta capacidade de aprendizado, tornando o uso das técnicas de otimização via simulação por metamodelagem (OvSM) para solucionar problemas complexos um campo de estudo promissor. Neste sentido, o presente estudo propõe um *framework* para OvSM embasado nos *insights* e análises provindos da revisão sistemática de literatura realizada. O *framework* proposto incorpora o uso de técnicas de simulação a eventos discretos, *design of experiments*, algoritmos de *machine learning*, e otimização de hiper-parâmetros via algoritmo genético para problemas de OvS. A fim de validar o *framework* proposto, esta dissertação testou e comparou seis algoritmos de *machine learning* (*Support Vector Machine*, Redes Neurais Artificiais, *Gradient-Boosted Trees*, *Randon Forest*, Regressão Polinomial e *Gaussian Process*) com e sem a etapa de otimização de hiper-parâmetros em dois arranjos experimentais (*Latin Hipercube Design* e Aleatório) aplicados ao problema de alocação de recursos em três casos reais da indústria. Com a aplicação do método nos objetos de estudo apresentados, os metamodelos de melhor performance obtiveram soluções que atingiram, respectivamente, 100%, 96,17%, e 100% do ótimo local *benchmark*, demandando, em média, 35,22% menos tempo computacional. Além disto, a incorporação da etapa de otimização de hiper-parâmetros no método de metamodelagem proposto permitiu uma redução de 31,28% no *root mean square error* dos metamodelos se comparado ao método tradicional, que não contempla esta etapa.

Palavras-chave: Simulação a eventos discretos; Otimização via simulação; Metamodelagem; *Machine learning*; *Design of experiments*, *Framework*.

ABSTRACT

In the context of industry 4.0, optimization via simulation (OvS) emerges as one of the most powerful tools in the modern industry, allowing decision-makers to allocate their resources more assertively. However, in very complex systems, the use of conventional OvS techniques requires computational time, which frequently, makes its application unfeasible. In recent years, the development in the machine learning area has emerged algorithms with high learning capacity, making the use of optimization via simulation by metamodeling (OvSM) techniques to solve complex problems a promising field of study. In this sense, the present study proposes a framework for OvSM based on the insights and analyses derived from the systematic literature review carried out. The proposed framework incorporates the use of discrete event simulation techniques, design of experiments, machine learning algorithms, and hyper-parameter optimization via genetic algorithm for OvS problems. To validate the proposed method, this dissertation tested and compared six machine learning algorithms (Support Vector Machine, Artificial Neural Networks, Gradient-Boosted Trees, Random Forest, Polynomial Regression, and Gaussian Process) with and without the hyper optimization step -parameters in two experimental arrangements (Latin Hypercube Design and Random) applied to the problem of resource allocation in three real cases in the industry. With the application of the method in the study objects presented, the best performing metamodels obtained solutions that reached, respectively, 100%, 96.17%, and 100% of the optimal benchmark location, demanding, on average, 35.22% less time computational. Also, the incorporation of the hyper-parameter optimization step in the proposed metamodeling method allowed a 31.28% reduction in the root mean square error of the metamodels compared to the traditional method, which does not include this step.

Keywords: *Discrete event simulation; Optimization via simulation; Metamodeling; Machine learning; Design of experiments, Framework.*

LISTA DE FIGURAS

Figura 1.1 - Número de publicações por ano	18
Figura 1.2 - Classificação da pesquisa	21
Figura 1.3 - Sistema de resolução de problemas	22
Figura 1.4 – Etapas da pesquisa.....	24
Figura 2.1 - Etapas de um projeto de simulação a eventos discretos	28
Figura 2.2 - Estrutura de uma rede neural artificial.....	34
Figura 2.3 - Estrutura geral de um neurônio.....	34
Figura 2.4 – Exemplos de arranjos LHD com 2 variáveis de entrada e 7 pontos amostrais	42
Figura 3.1 – Pilares da pesquisa e dados extraídos dos artigos	47
Figura 3.2 - Processo de busca/triagem	48
Figura 3.3 - Produção ao longo do tempo dos 10 autores mais produtivos.....	64
Figura 3.4 - Top 10 centros de pesquisa em OvSM	65
Figura 3.5 - Densidade de publicação e colaboração internacional	65
Figura 3.6 - Produção científica anual.....	66
Figura 3.7 - Principais Keywords Plus	67
Figura 3.8 - Estratégia de otimização baseada em metamodelo global vs. local, semelhanças e diferenças.....	69
Figura 4.1 - <i>framework</i> proposto para otimização via simulação por metamodelagem.....	76
Figura 5.1 - Comparativo do RMSE médio entre arranjos para o OE1	84
Figura 5.2 - IC95% para o RMSE	85
Figura 5.3 - (a) Comparativo de FO e tempo para otimização. (b) Comparativo de FO e tempo para treinar e otimizar os metamodelos para o OE1.....	87
Figura 5.4 - Comparativo do RMSE médio entre arranjos para o OE2	90
Figura 5.5 - (a) Comparativo de FO e tempo para otimização. (b) Comparativo de FO e tempo para treinar e otimizar os metamodelos para o OE2.....	92
Figura 5.6 - Comparativo do RMSE médio entre arranjos para o OE3	95
Figura 5.7 - (a) Comparativo de FO e tempo para otimização. (b) Comparativo de FO e tempo para treinar e otimizar os metamodelos para o OE3.....	97
Figura 5.8 - Comparativo de FO e tempo considerando o <i>SimRunner</i> no modo Cauteloso (<i>bench 02</i>).....	98
Figura 5.9 – Comparativo do FO médio entre os arranjos	101
Figura 5.10 – Comparativo do RMSE médio entre os arranjos.....	101

Figura 5.11 – Redução no RMSE com o uso da otimização de hiper-parâmetros	103
Figura 5.12 – Comparativo do RMSE e FO entre os metamodelos com hiper-parâmetros <i>default</i> e otimizados.....	103

LISTA DE TABELAS

Tabela 1.1. - Análise bibliométrica	19
Tabela 2.1 -Comparativo de métodos comuns para otimização de hiper-parâmetros.....	39
Tabela 3.1 - Natureza da pesquisa	50
Tabela 3.2 - Variáveis e objetivos dos problemas	52
Tabela 3.3 - Método adotado.....	54
Tabela 3.4 - Principais hiper-parâmetros dos algoritmos de metamodelagem.....	59
Tabela 3.5 - Softwares e métricas.....	61
Tabela 3.6 - Número de publicações de acordo com o nome dos autores, nacionalidade e fonte	63
Tabela 5.1 - Variáveis de decisão do problema para OE1.....	80
Tabela 5.2 - Solução encontrada pelo otimizador <i>SimRunner</i> para o OE1	81
Tabela 5.3 - Seleção dos hiper-parâmetros para o OE1	82
Tabela 5.4 - Metamodelos aplicados ao OE1	83
Tabela 5.5 - Metamodelos aplicados ao OE1	86
Tabela 5.6 - Comparativo dos metamodelos para o OE1	86
Tabela 5.7 - Variáveis de decisão do problema para o OE2.....	88
Tabela 5.8 - Solução encontrada pelo otimizador <i>SimRunner</i> para o OE2	89
Tabela 5.9 - Metamodelos aplicados ao OE2	89
Tabela 5.10 - Metamodelos aplicados ao OE2	91
Tabela 5.11 - Comparativo dos metamodelos para o OE2	91
Tabela 5.12 - Variáveis de decisão para o OE3.....	93
Tabela 5.13 - Solução encontrada pelo otimizador <i>SimRunner</i> para o OE3	94
Tabela 5.14 - Metamodelos aplicados ao OE3	94
Tabela 5.15 - Metamodelos aplicados ao OE3	96
Tabela 5.16 - Comparativo dos metamodelos para o OE3	97
Tabela 5.17 - Comparativo dos metamodelos para o OE3 considerando o <i>SimRunner</i> no modo Cauteloso	98
Tabela 5.18 - Resultado dos testes estatísticos para os objetos de estudo.....	99
Tabela 5.19 - FO média: metamodelo vs. arranjo com hiper-parâmetros <i>default</i>	99
Tabela 5.20 - FO média: metamodelo vs. arranjo com hiper-parâmetros otimizados.....	100
Tabela 5.21 - RMSE média: metamodelo vs. arranjo com hiper-parâmetros <i>default</i>	100
Tabela 5.22 - RMSE média: metamodelo vs. arranjo com hiper-parâmetros otimizados.....	100

LISTA DE ABREVIACOES

AG	Algoritmo Genético
CCD	<i>Central Composite Design</i> (Design Composto Central)
CIMO	<i>Context, Intervation, Mechanism, Outcomes</i> (Contexto, Interveno, Mecanismo, Resultados)
R ²	Correlao Quadrtica
DOE	<i>Design of Experiments</i> (Delineamento de Experimentos)
EGO	<i>Efficient Global Optimization</i> (Otimizao Global Eficiente)
EI	<i>Expected Improvement</i> (Melhoria Esperada)
FO	Funo Objetivo
GP	<i>Gaussian Process</i> (Processo Gaussiano)
GBT	<i>Gradient Boosted Trees</i> (rvores com gradiente de aumento)
LHD	<i>Latin Hypercube Design</i> (Design Hiper cubo Latino)
ML	<i>Machine Learning</i> (Aprendizado de Mquina)
MAE	<i>Mean Absolut Error</i> (Erro Absoluto Mdio)
MAPE	<i>Mean Absolute Percentage Error</i> (Erro Percentual Absoluto Mdio)
MSE	<i>Mean Square Error</i> (Erro Quadrtico Mdio)
RSM	<i>Response Surface Methodology</i> (Metodologia de superfcie de resposta)
NSGA-II	Non-dominated Sorting Genetic Algorithm II (Algoritmo Gentico de Classificao No Dominada II)
OE	Objeto de Estudo
OvSM	Otimizao via simulao por metamodelagem
OvS	Otimizao via simulao
QP	Questes de Pesquisa
RBF	<i>Radial Basis Function</i> (Funo de Base Radial)
RF	Random Forest (Floresta Randmica)
RNA	Redes Neurais Artificiais
RP	Regresso Polinomial
RSL	Reviso Sistemtica de Literatura
RL	Revises de Literatura
RMSE	<i>Root Mean Square Error</i> (Raiz do Erro Quadrtico Mdio)
SED	Simulao a Eventos Discretos
SVM	<i>Support Vector Machine</i> (Mquina de vetores de suporte)

WoS

Web of Science (Teia da Ciência)

SUMÁRIO

1. INTRODUÇÃO.....	16
1.1. Objetivos.....	20
1.2. Método de Pesquisa.....	20
1.2.1. Classificação da Pesquisa.....	20
1.2.2. Modelagem e Simulação.....	22
1.2.3. Etapas do Método a Serem Desenvolvidas.....	23
1.3. Delimitação da Pesquisa.....	25
1.4. Estrutura do Trabalho.....	25
2. FUNDAMENTAÇÃO TEÓRICA.....	26
2.1. Simulação a Eventos Discretos.....	26
2.2. Otimização via Simulação.....	29
2.3. Definições Gerais de Metamodelagem.....	30
2.3.1. Algoritmos de <i>Machine Learning</i>	31
2.3.1.1. <i>Gaussian Process</i>	31
2.3.1.2. Regressão Polinomial.....	32
2.3.1.3. Redes Neurais Artificiais.....	33
2.3.1.4. <i>Random Forest</i>	35
2.3.1.5. <i>Gradient Boosted Trees</i>	35
2.3.1.6. <i>Support Vector Machine</i>	37
2.3.2. Métricas de Performance.....	38
2.3.3. Otimização de Hiper-parâmetros via Algoritmo Genético.....	39
2.3.4. <i>Latin Hypercube Design</i>	41
3. REVISÃO SISTEMÁTICA DE LITERATURA.....	43
3.1. Revisões de Literatura Sobre Otimização via Simulação por Metamodelagem.....	43
3.2. O Método de Revisão Sistemática de Literatura.....	44
3.3. Aplicação do Método de Revisão Sistemática de Literatura.....	45
3.3.1. Planejamento.....	45
3.3.2. Pesquisa e Triagem dos Artigos.....	47
3.3.3. Análise e Síntese.....	48

3.3.4. Apresentação dos Resultados	49
3.4. Resultados e discussões da Revisão Sistemática de Literatura.....	49
3.4.1. Natureza da pesquisa.....	49
3.4.2. Métodos adotados.....	54
3.4.3. Estado da arte em OvSM.....	63
3.4.4. <i>Frameworks</i> para OvSM	67
3.4.5. Futuras direções na pesquisa em OvSM	69
3.5. Considerações Finais da Revisão Sistemática de Literatura.....	71
4. FRAMEWORK DE METAMODELAGEM PROPOSTO	74
4.1. Técnicas Utilizadas para Proposição do <i>Framework</i>	74
4.2. <i>Framework</i> para Metamodelagem em problemas de Otimização via Simulação.....	75
4.3. Procedimentos Metodológicos do Trabalho	77
5. APLICAÇÃO DO FRAMEWORK PROPOSTO	80
5.1. Primeiro Objeto de Estudo.....	80
5.1.1. Apresentação do Problema.....	80
5.1.2. Aplicação e Comparação dos Metamodelos	81
5.2. Segundo Objeto de Estudo.....	87
5.2.1. Apresentação do Problema.....	87
5.2.2. Aplicação e Comparação dos Metamodelos	89
5.3. Terceiro Objeto de Estudo	92
5.3.1. Apresentação do Problema.....	92
5.3.2. Aplicação e Comparação dos Metamodelos	94
5.4. Análise dos Resultados	98
6. CONCLUSÕES.....	105
6.1. Verificação dos Objetivos Específicos	106
6.2. Sugestões para Futuras Pesquisas	107
REFERÊNCIAS BIBLIOGRÁFICAS	109
ANEXO A – Símbolos do IDEF-SIM.....	129
ANEXO B – Modelos Conceituais em IDEF-SIM	130
ANEXO C – Hiper-parâmetros.....	133
ANEXO D – Código para Treinamento, Otimização e Aplicação dos Metamodelos	134
APÊNDICE - Artigos Publicados em Congressos e Periódicos.....	148

1. INTRODUÇÃO

Assegurar a competitividade das organizações, satisfazer seus clientes e reduzir custos requer que os sistemas produtivos lancem mão de ferramentas cada vez mais efetivas para auxílio à tomada de decisões (SALAM; KHAN, 2016). De acordo com Law (2013), soluções analíticas podem ser eficazes quando o sistema estudado é relativamente simples, todavia o autor ressalta que a grande maioria dos sistemas reais apresentam grande complexidade para serem analisados analiticamente. Neste sentido, a simulação é uma técnica desenvolvida para prever e avaliar a performance de sistemas complexos e estocásticos, os quais são, muitas vezes, analiticamente intratáveis (XU *et al.*, 2016).

De acordo com Siebers *et al.* (2010), a técnica de simulação a eventos discretos (SED) busca representar computacionalmente sistemas reais, estudando o comportamento de variáveis que mudam de status em horizontes discretos de tempo (GABRIEL *et al.*, 2020; SOUSA JUNIOR *et al.*, 2019; LAW, 2013). Uma das vantagens do uso da SED é a redução do risco na tomada de decisão, uma vez que ela permite avaliar diversos cenários sem a interferência física no sistema produtivo (HELLENO *et al.*, 2015). No entanto, quando existe uma grande quantidade de cenários possíveis, torna-se praticamente impraticável testar todos estes cenários, neste sentido, a otimização via simulação (OvS) é a técnica mais indicada para se atingir estes objetivos (BARTON, 2009).

O campo da OvS progrediu significativamente nas últimas década, com o desenvolvimento de novos algoritmos, implementações e aplicativos. Significativas contribuições para este campo de pesquisa tem surgido de pesquisadores e profissionais, principalmente nas áreas de engenharia, operações industriais, programação matemática, estatística e aprendizado de máquina, bem como nas comunidades de ciência da computação (AMARAN *et al.*, 2016).

A OvS tem como objetivo encontrar os valores de entrada do modelo de simulação que resultem nos melhores *outputs* (BELGIN, 2019). De acordo com Fu (2002), o processo de OvS pode ser dividido em duas fases: 1) geração de possíveis candidatas à solução do problema, e 2) avaliação destas soluções por meio de estimativas em um modelo de simulação. De acordo com Miranda *et al.* (2017), nos últimos anos diversas técnicas de OvS foram desenvolvidas, como as heurísticas e metaheurísticas, porém ainda é um campo de pesquisa latente na literatura a ser explorado.

Os métodos de OvS podem ser classificadas em duas categorias: *model-based* e *metamodel-based*. Moghaddam e Mahlooji (2017) explicam que na categoria *model-based*, o

módulo de otimização e simulação possuem uma integração direta entre si, na qual a otimização escolhe quais soluções precisam ser avaliadas e a simulação as avalia. Já na abordagem *metamodel-based* surge um terceiro componente chamado metamodelo, que objetiva identificar e estimar a relação entre as entradas e saídas do modelo de simulação, na forma de uma função matemática determinística, a qual é utilizada para avaliar as possíveis soluções no processo de otimização (BARTON; MECKESHEIMER, 2006; MOGHADDAM; MAHLOOJI, 2017; PARNIANIFARD *et al.*, 2020a).

A abordagem por *model-based* é considerada a mais popular e difundida na literatura, todavia quando o modelo de simulação é muito complexo e o espaço de soluções do problema é grande, o processo de OvS por estas técnicas torna-se demasiadamente lento, podendo demandar de horas a meses para convergirem a um valor de ótimo local aceitável, inviabilizando seu uso em diversas aplicações (CAI *et al.*, 2019; MIRANDA *et al.*, 2014). Em contrapartida, visando a redução do tempo computacional gasto na otimização, o uso das técnicas de metamodelagem cresceu consideravelmente na última década (PARNIANIFARD *et al.*, 2019).

Atualmente existem diversos algoritmos para construção dos metamodelos, alguns exemplos incluem a metodologia de superfície de resposta (RSM) (HASSRNAAYEBI *et al.*, 2019; SHIRAZI; MAHDAVI; MAHDAVI-AMIRI, 2011), *Kriging* (BAQUELA; OLIVERA, 2019; PEDRIELLI *et al.*, 2020), *radial basis function* (RBF) (CHRISTELIS; MANTOGLOU, 2016; YIN *et al.*, 2016), e redes neurais artificiais (RNA) (RAEI *et al.*, 2019; STORTI *et al.*, 2019). Segundo Sousa Junior *et al.* (2019), a metamodelagem é o segundo método mais utilizado em problemas de otimização via simulação a eventos discretos, perdendo apenas para o uso de metaheurísticas.

O desenvolvimento na área de *machine learning* (ML) tem contribuído para o desenvolvimento de técnicas de otimização capazes de solucionar problemas de maior complexidade, como por exemplo, a criação de metamodelos mais precisos e flexíveis (BATUR; BEKKI; CHEN, 2018). Os algoritmos de ML possuem diversos hiper-parâmetros, os quais impactam drasticamente em sua precisão, sendo sua correta seleção uma importante etapa do processo de metamodelagem (XIA *et al.*, 2017). Neste sentido, o presente trabalho tem como tema de pesquisa, o estudo de técnicas de metamodelagem aplicadas em problemas de OvS complexos.

Jalali e Nieuwenhuys (2015) ressaltam, que embora a metamodelagem seja menos popular que a utilização de metaheurísticas ou métodos híbridos, ela oferece meios para solucionar problemas de OvS robustos. A fim de identificar a relevância do tema, em maio de

2020 foi realizada uma análise bibliométrica na base de dados *Scopus*, buscando por trabalhos em inglês que contenham os termos correlatos para metamodelagem (*metamodel* OU *metamodeling* OU “*meta-model*” OU “*machine learning*”) e OvS (“*simulation optimization*” OU “*simulation-based optimization*” OU “*optimization via simulation*” OU OvS). Na Figura 1.1 observa-se uma forte tendência de crescimento, principalmente nos últimos anos, no número de publicações envolvendo o uso de metamodelagem e OvS, evidenciando a contemporaneidade desta pesquisa.

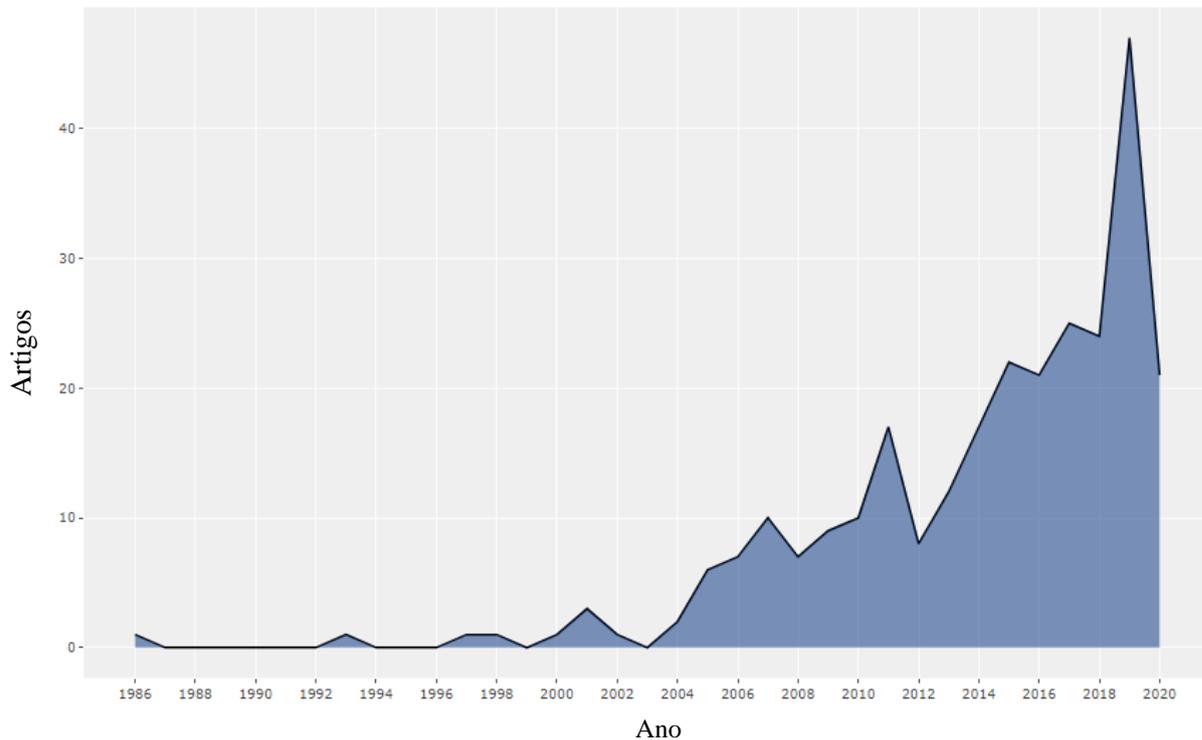


Figura 1.1 - Número de publicações por ano

Fonte: Base de dados da *Scopus*

Também foi desenvolvida uma outra análise bibliométrica nas bases de dados *Scopus* e *Web of Science*, referente ao mesmo período, pesquisando-se pelos termos correlatos em inglês referentes a: otimização via simulação, metamodelagem, e *machine learning*, bem como a combinação destes termos. Mongeon e Paul-Hus (2016) ressaltam que a *Scopus* e *Web of Science* como principais fontes de dados para citações. Os resultados encontrados são apresentados na Tabela 1.1.

Tabela 1.1. - Análise bibliométrica

Base de dados	“Simulation optimization” OR “simulation-based optimization” OR “optimization via simulation”	“Metamodel”	“Machine Learning”	“Simulation Optimization” OR “Simulation-Based Optimization” OR “Optimization via Simulation” AND “Metamodel” AND “Machine Learning”
<i>Web of Science</i>	3.962	4.304	117.874	4
<i>Scopus</i>	5.967	7.094	192.300	5

Nota-se, pela Tabela 1.1, um número expressivo de publicações relacionados aos termos pesquisados separadamente. Todavia observa-se que, quando se procura pela intersecção entre estas áreas de estudo, ainda há um número reduzido de trabalhos publicados, revelando uma lacuna a ser explorada na literatura. Como observado na Tabela 1.1 foram encontrados um total de quatro artigos na *Web of Science* e cinco na *Scopus* (sendo que um artigo ocorreu em ambas as bases). De estes destaca-se os trabalhos de Hannah, Powell e Dunson (2014), que propuseram um método de otimização via simulação por metamodelagem (OvSM) baseada em regressão semi-convexa, e De La Fuente e Smith (2017), que compararam diversos algoritmos de metamodelagem para OvS em um hospital.

Também no setor de *healthcare*, Yousefi *et al.* (2018) utilizaram métodos de ensemble em um modelo de simulação baseada em agentes para redução do tempo médio de permanência, e Yousefi e Yousefi (2019) aplicaram a mesma metodologia para alocação ótima de recursos, com a finalidade de reduzir o tempo médio de *Door to Doctor* em uma emergência hospitalar.

No campo das engenharias, Souza Junior *et al.* (2019) estudaram a integração entre metamodelagem por ML, processamento paralelo e metaheurísticas aplicados em problemas de engenharia de produção e Woldemariam e Lemu (2019) utilizaram metamodelagem por RNAs e algoritmo genético (AG) para otimização de modelos para análise dinâmica de fluidos. Hüllen *et al.* (2020) propõe estratégias para lidar com a incerteza na OvS, combinando ML com programação estocástica, otimização robusta e modelagem de discrepâncias. Sousa Junior *et al.* (2020) estudaram 12 configurações para otimização, integrando 33 algoritmos de ML, 2 metaheurísticas, e 2 hardwares para processamento paralelo em um problema de alocação de recursos de chão de fábrica.

Embora os trabalhos descritos tratem de metamodelagem utilizando técnicas de ML, não se encontrou na literatura um trabalho que trate em um mesmo contexto problemas de OvS utilizando algoritmos de ML e uma estratégia para otimização de seus hiper-parâmetros, o que potencializa os resultados deste tipo de aplicação. Neste sentido, a presente pesquisa visa investigar se o uso da metamodelagem em problemas de otimização via simulação a eventos

discretos pode ser competitivo se comparado às metaheurísticas adotadas nos *softwares* comerciais no que diz respeito ao tempo computacional e qualidade da resposta.

1.1. Objetivos

O presente trabalho objetiva o desenvolvimento de um *framework* para condução de projetos de metamodelagem que integra, de forma inovadora, o uso de algoritmos de ML e otimização de hiper-parâmetros para solução de problemas de otimização em simulação a eventos discretos. Para tanto, este trabalho abrange os objetivos específicos elencados a seguir:

- Propor um *framework* para projetos de OvSM que integre SED, ML, e otimização de hiper-parâmetros;
- Conduzir o desenvolvimento, treinamento, otimização e validação dos metamodelos para cada um dos objetos de estudo;
- Aplicar o *framework* proposto na otimização de três modelos de SED;
- Avaliar os resultados obtidos e comparar os resultados da otimização via simulação por metaheurística com os resultados da otimização via metamodelagem.

Para tanto, este trabalho contribui com a literatura a partir do desenvolvimento de um *framework* para condução de projetos de metamodelagem que integra, de forma inovadora, o uso de algoritmos de ML e otimização de hiper-parâmetros para solução de problemas de otimização em simulação a eventos discretos. A fim de comprovar sua efetividade, o *framework* proposto é aplicado a problemas reais da indústria.

Além disto, este trabalho reflete sua contribuição tecnológica pelo desenvolvimento de um método que poderá ser utilizado na solução de problemas complexos no contexto da indústria e serviços 4.0. Desta forma, espera-se que esta pesquisa sirva de base para que pesquisadores e práticos possam explorar a potencialidade da integração entre estes dois campos de pesquisa (Simulação e Inteligência Artificial).

1.2. Método de Pesquisa

1.2.1. Classificação da Pesquisa

A classificação do método de pesquisa empregados neste trabalho concorda com o proposto por Miguel *et al.* (2018), assim como apresentado na Figura 1.2.

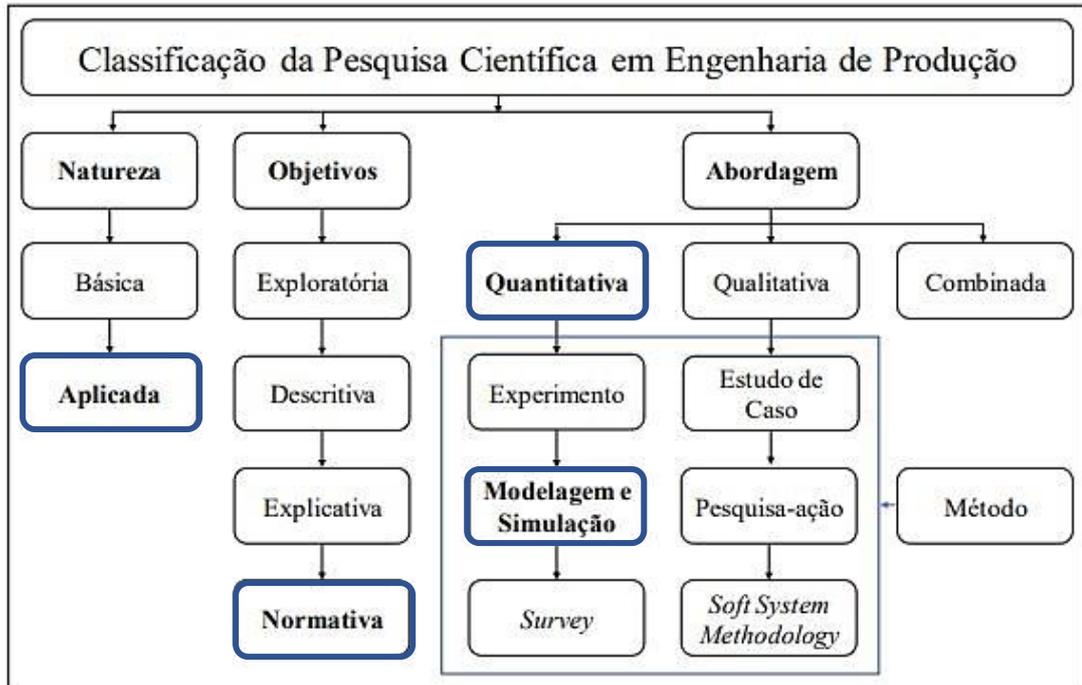


Figura 1.2 - Classificação da pesquisa

Fonte: Adaptado de Miguel *et al.* (2018)

Turrioni e Mello (2012) e Miguel *et al.* (2018) classificam as pesquisas científicas na área de engenharia de produção em quatro dimensões básicas, a saber: quanto à natureza, objetivos, abordagem e método. Esta pesquisa pode ser classificada como “aplicada”, pois segundo Turrioni e Mello (2012, p.80) uma pesquisa aplicada “caracteriza-se por seu interesse prático, isto é, que os resultados sejam aplicados ou utilizados imediatamente na solução de problemas que ocorrem na realidade”.

Quanto aos objetivos da pesquisa, pode-se classificá-los como objetivo exploratório, descritivo, explicativo ou normativo. De acordo com Bertrand e Fransoo (2002), uma pesquisa normativa é aquela que busca encontrar soluções ótimas para novas definições ou está interessada no desenvolvimento de políticas e estratégias para aperfeiçoar resultados encontrados na literatura. Logo, esta pesquisa possui objetivos normativos, uma vez que objetiva propor um *framework* para OvSM visando a obtenção de resultados ótimos em problemas complexos de SED.

Além disto, este trabalho pode ser entendido como uma abordagem quantitativa, a qual parte da teoria para se formular hipóteses, coletar dados através de observações, analisá-los e gerar resultados (MIGUEL *et al.*, 2018). Dentro desta abordagem existem diversos métodos e para esta pesquisa o método de modelagem e simulação mostrou-se o mais adequado, uma vez que, segundo Miguel *et al.* (2018), representa a manipulação de variáveis em um modelo representativo da realidade, todavia sem afetar o ambiente real durante esta manipulação.

Acrescentando-se as classificações anteriores, Gil (2002) propões a classificação da pesquisa quanto aos procedimentos técnicos utilizados, em especial referente ao meio de coleta de dados. Neste critério, o autor sugere que uma pesquisa pode ser classificada em bibliográfica, documental, experimental, *ex-post facto*, coorte, levantamento, de campo, estudo de caso, pesquisa-ação e participante. De acordo com suas peculiaridades esta dissertação pode ser entendida como experimental, pois, segundo Gil (2002, p. 47), é aquela que “consiste em determinar um objeto de estudo, selecionar as variáveis que seriam capazes de influenciá-lo, definir as formas de controle e de observação dos efeitos que a variável produz no objeto”

1.2.2. Modelagem e Simulação

Mitroff *et al.* (1974) propuseram uma metodologia científica de resolução de problemas voltadas para área de pesquisa operacional, e em especial para modelagem e simulação. Em seu modelo os autores propuseram a subdivisão do projeto de simulação em quatro etapas, a saber: conceitualização, modelagem, solução do modelo, e implementação. Esta dissertação adotará o este método pois Mitroff *et al.* (1974) é um método clássico, robusto e amplamente discutido pela literatura de pesquisa operacional, porém ainda considerado um método válido e aplicável nos tempos atuais (WOLLMANN; TORTATO, 2019). A Figura 1.3 apresenta estas fases e as ações necessárias para evoluir entre as fases.

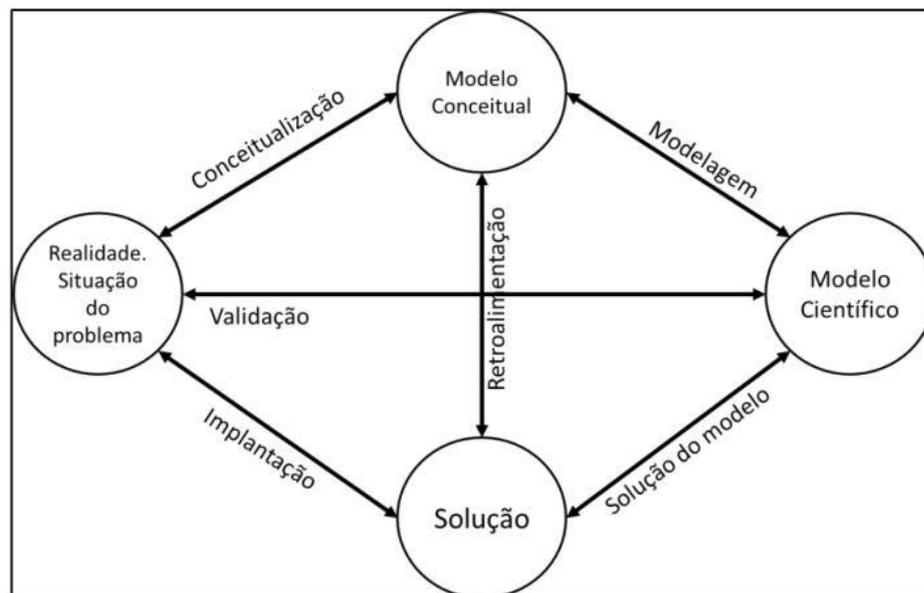


Figura 1.3 - Sistema de resolução de problemas

Fonte: Adaptado de Mitroff *et al.* (1974)

Bertrand e Fransoo (2002) propõem a classificação das pesquisas quantitativas com base nas etapas da metodologia de Mitroff *et al.* (1974), sendo divididas em quatro tipos a depender do ciclo do projeto, a saber:

- Empírica descritiva: projeto em que o pesquisador envolve o ciclo conceitualização, modelagem e validação;
- Empírica normativa: engloba as ações de conceitualização, modelagem, solução do modelo e implantação;
- Axiomática descritiva: envolve a ação de modelagem, passando do modelo conceitual para o modelo científico;
- Axiomática normativa: envolve as ações de modelagem e solução do modelo.

Dada a classificação proposta por Bertrand e Fransoo (2002), esta pesquisa pode ser entendida como axiomática normativa, ao passo que não pretende basear-se na observação de um sistema real, mas a partir de um modelo conceitual pré-existente desenvolver o modelo científico e prover uma solução para o problema a partir otimização de modelos de SED com o uso de metamodelagem.

1.2.3. Etapas do Método a Serem Desenvolvidas

A Figura 1.4 apresenta as etapas a serem seguidas na elaboração desta pesquisa.

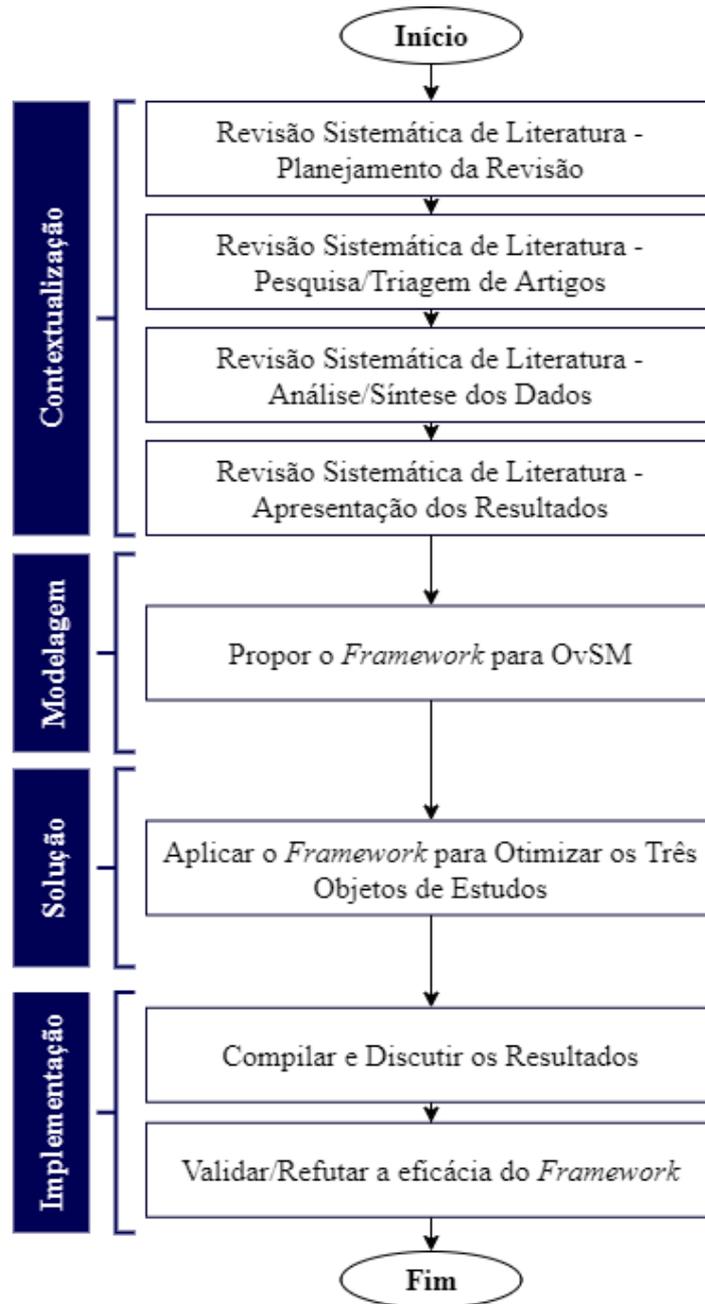


Figura 1.4 – Etapas da pesquisa

A primeira etapa do método de pesquisa adotado, a contextualização, se dará pela análise da literatura acerca do tema. Nesta etapa será conduzida uma RSL (*Capítulo 3*) com o intuito de identificar as principais técnicas adotadas em cada etapa de um projeto de metamodelagem (amostragem, simulação, otimização, validação, entre outros) e a definição dos principais algoritmos de ML e seus hiper-parâmetros a serem utilizados nesta pesquisa. O resultado desta etapa promoverá o embasamento teórico necessário para construção do método de metamodelagem proposto neste trabalho.

Na etapa de modelagem (*Capítulo 4*), será proposto o framework para otimização via simulação por metamodelagem, visando a melhoria da qualidade de resposta e tempo

computacional. Ao final desta etapa o método estará apto a ser testado e validado para otimização de modelos de SED. Na etapa de solução do modelo (**Capítulo 5**), o *framework* proposto será utilizado para otimização de modelos de SED (três objetos de estudo), confrontando seus resultados perante um *software* comercial baseado em metaheurísticas. Nesta etapa, também será avaliado o impacto sofrido pelos resultados quanto ao uso de diversos níveis de arranjos amostrais e ao uso da otimização de hiper-parâmetros.

A última etapa, implementação (**Capítulo 6**), discute os resultados da implementação do método, estando o mesmo pronto e validado para ser utilizado em futuros projetos. Para validar comparou-se o resultado com uma ferramenta comercial amplamente difundida na literatura para OvS por metaheurísticas.

1.3. Delimitação da Pesquisa

Nesta pesquisa será proposto um *framework* para OvSM em problemas de SED, no qual será incorporado conceitos de ML, otimização de hiper-parâmetros e *design of experiments* (DOE). Ressalta-se que não é escopo nem objetivo deste trabalho discutir sobre demais métodos de otimização empregados em OvS, tais como metaheurísticas e heurísticas, detalhar e incrementar a formulação matemática dos algoritmos de ML, analisar os diversos algoritmos para otimização de hiper-parâmetros, comparar sistematicamente arranjos de DOE ou propor formulações para o tamanho ideal de amostras em projetos de OvSM.

1.4. Estrutura do Trabalho

Este trabalho encontra-se estruturado em 6 Capítulos. No Capítulo 2 encontra-se a fundamentação teórica a qual este trabalho se baseia, apresentando os conceitos gerais acerca de SED, OvS e metamodelagem, e o Capítulo 3 apresenta a revisão sistemática de literatura (RSL) sobre metamodelagem em problemas de OvS. No Capítulo 4 será apresentado o método de metamodelagem proposto neste trabalho e o Capítulo 5 discutirá a aplicação do método em três estudos de caso e os resultados obtidos. Por fim, o Capítulo 6 conclui o trabalho e apresenta oportunidades para futuras pesquisas sobre o tema.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo serão apresentados os conceitos principais sobre SED, OvS e metamodelagem. Ressalta-se que o embasamento teórico presente neste Capítulo constitui a base na qual foi construída esta dissertação e a qual suporta os resultados e conclusões obtidos com este trabalho. A revisão de literatura abrangeu, principalmente, periódicos nacionais e internacionais, artigos de congressos, livros, teses e dissertações com temas correlatos a este.

2.1. Simulação a Eventos Discretos

Para Montevechi *et al.* (2015) e Law (2013), a simulação a eventos discretos é uma ferramenta de modelagem que visa replicar computacionalmente sistemas complexos, permitindo testar e analisar mudanças sem intervenções no sistema real. A SED pode ser aplicada nos mais diversos campos, tais como manufatura, logística, *healthcare*, cadeia de suprimentos entre outras, com um mesmo objetivo: prover aos tomadores de decisões análises mais realísticas dos processos (HELLENO *et al.*, 2015). Greasley (2004) reforça que o uso da SED reduz o risco das decisões, uma vez que é possível quantificar as melhorias no sistema antes mesmo de qualquer investimento.

De acordo com Law (2013), a SED pode ser definida como a modelagem de um sistema que evolui ao longo de uma linha temporal, o qual mudam de estado em pontos discretos no tempo. Desde sua criação nos anos de 1950s a simulação vem sendo empregada com sucesso na melhoria de processos, fornecendo uma ferramenta de modelagem e análise de sistemas complexos que são analiticamente intratáveis (MIRANDA, RAFAEL DE CARVALHO *et al.*, 2017).

Banks *et al.* (2014) afirmam que um modelo computacional, uma vez validado, permite ao decisor avaliar o comportamento do sistema no tempo, permitindo responder perguntas como “O que ocorre se”. Os mesmos autores citam como vantagens do uso da SED a possibilidade de se obter análises específicas de determinada parte do sistema, podendo assim reduzir sua complexidade.

Law (2013) ressalta que a SED é uma alternativa de baixo custo para a experimentação real, evitando paradas no fluxo produtivo bem como custos de implementação no chão de fábrica. Além disto, a SED permite ao analista experimentar diversos cenários que o levaria à melhor configuração do sistema estudado. Porém, de acordo com Xu *et al.* (2016), no estudo de sistemas complexos e com inúmeras possibilidades de configuração, faz-se necessário o uso

de técnicas de otimização em conjunto com a SED, com o objetivo de encontrar o melhor *design* para o sistema.

Montevecchi *et al.* (2007) descrevem as etapas de um projeto de simulação a eventos discretos, como visto na Figura 3.1. A primeira etapa constitui na concepção do modelo conceitual, definindo e descrevendo o objeto de estudo (OE). A segunda etapa refere-se a etapa de implementação e validação do modelo computacional, e na última etapa, chamada de Análise, é realizada a experimentação e análise dos resultados do modelo, recorrendo-se muitas vezes ao uso das técnicas de OvS.

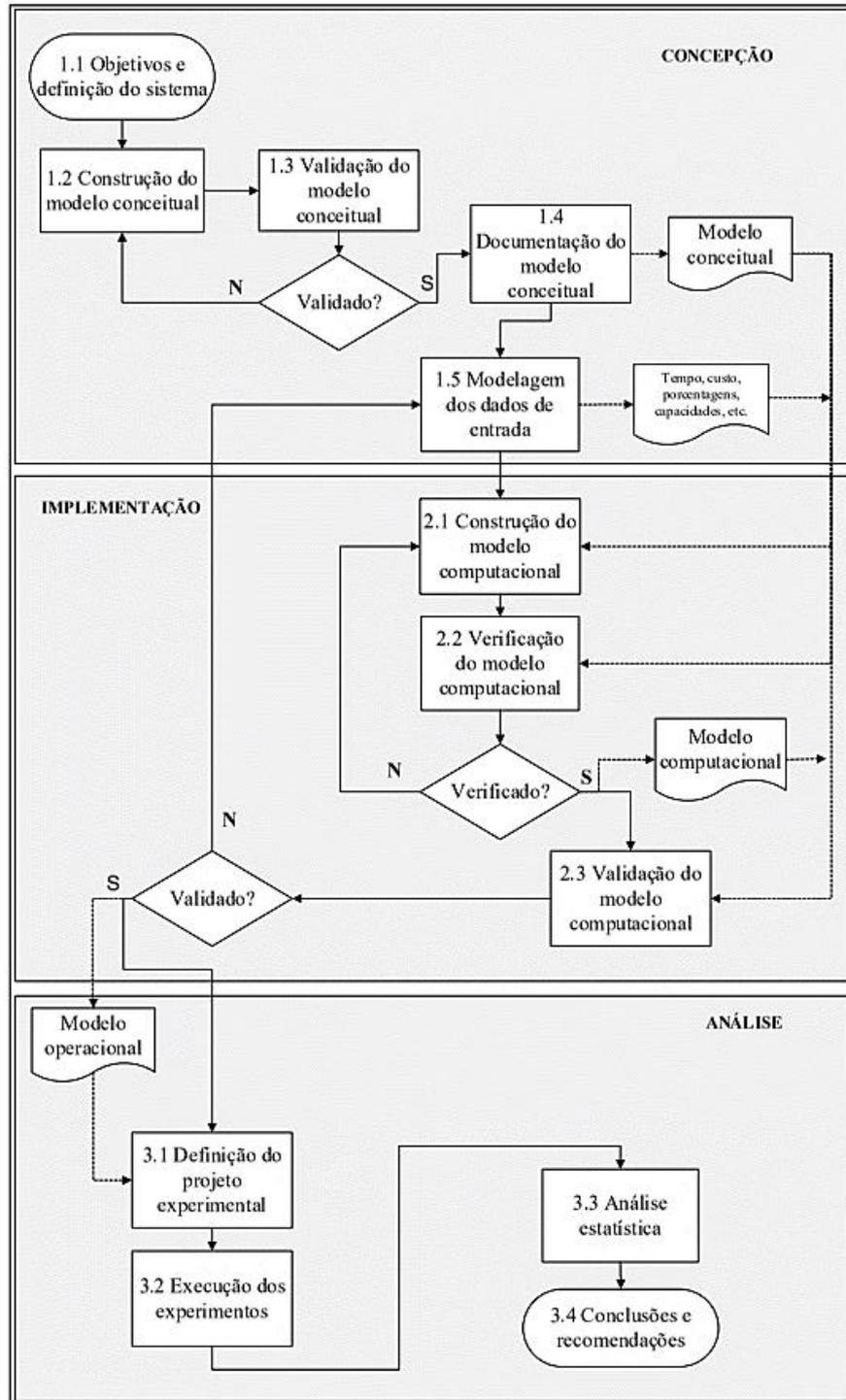


Figura 2.1 - Etapas de um projeto de simulação a eventos discretos

Fonte: Adaptado de Montevechi *et al.* (2007)

Montevechi *et al.* (2015) salientam que uma das partes mais fundamentais em um projeto de simulação é o desenvolvimento do modelo conceitual. Segundo Montevechi *et al.* (2010), o IDEF-SIM é uma técnica de modelagem conceitual que utiliza elementos e símbolos do IDEF0 e do IDEF3, porém com fluxos e significados adequados para simulações. Segundo os autores, o IDEF-SIM possui como vantagens sobre as demais técnicas de modelagem conceitual o seu direcionamento para o processo de simulação, o qual busca destacar

informações que são comumente necessárias durante a etapa de modelagem computacional. Dada estas características, este estudo utilizará desta técnica para construção dos modelos conceituais para os objetos de estudo. A simbologia utilizada no IDEF-SIM estão apresentadas no Anexo A deste trabalho.

2.2. Otimização via Simulação

De acordo com Lin e Chen (2015), otimização via simulação é uma poderosa ferramenta de tomada de decisões, pois esta técnica é capaz de capturar o relacionamento entre várias entidades de um sistema complexo do mundo real, identificando a melhor solução para o problema. Souza Junior *et al.* (2019) ensinam que OvS é utilizada para auxiliar tomadores de decisão a investir ou alocar recursos em sistemas de produção novos ou existentes.

Amaran *et al.* (2016) definem OvS como sendo a otimização de uma função objetivo (FO) sujeita a restrições, a qual é avaliada por meio de uma simulação estocástica. Carson e Maria (1997) referem-se a OvS como o processo de identificação dos melhores valores de entrada para variáveis de um sistema simulado, avaliando as soluções por meio de um *loop* criado entre o algoritmo de otimização e o modelo de simulação.

O campo do estudo da OvS evoluiu significativamente na última década, com o desenvolvimento de diversos algoritmos, *softwares* e aplicações. Amaran *et al.* (2016) ensinam que a formulação geral para um problema de OvS pode ser representado pela Eq. (1).

$$\begin{aligned}
 & \text{Min } \mathbb{E}_\omega[f(x, y, \omega)] \\
 & \text{s. a. } \mathbb{E}_\omega[g(x, y, \omega)] \leq 0 \\
 & \quad h(x, y) \leq 0 \\
 & \quad x_l \leq x \leq x_u \\
 & \quad x \in \mathbb{R}^n, y \in \mathbb{D}^m
 \end{aligned} \tag{1}$$

No qual \mathbb{E}_ω representa o valor esperado para a função f , que é avaliada através da simulação a partir de entradas contínuas x , ou discretas y , submetidas a um vetor de números aleatórios ω . Da mesma forma as restrições são definidas pelos valores da função g que são avaliados a cada simulação. O problema ainda pode conter outras restrições (representadas por h) que não envolvem variáveis aleatórias, bem como restrições vinculadas as variáveis de decisão.

Em contrapartida, Miranda *et al.* (2017) ressaltam que em problemas de OvS em sistemas complexos e com espaço de soluções muito grandes, demanda-se um longo tempo computacional para que o algoritmo de otimização convirja em um bom resultado. Barton (2009) cita que estas características levaram pesquisadores a desenvolverem métodos

especializados para OvS, dentre eles destacam-se: *Ranking* e seleção; heurísticas e metaheurísticas; busca randômica; e metamodelagem.

2.3. Definições Gerais de Metamodelagem

De acordo com De La Fuente e Smith (2017), quanto mais complexo é o sistema estudado, mais tempo é demandado na execução da otimização. Segundo os autores, para se obter bons resultados em tempo hábil, faz-se necessário elevar o problema a um outro nível de abstração, chamado de metamodelo.

A metamodelagem é uma abordagem comumente usada para desenvolver um modelo representativo de um modelo de simulação (DE LA FUENTE; ERAZO; SMITH, 2019). O metamodelo é capaz de capturar a relação entre os valores das variáveis de decisão e os *outputs* da simulação, provendo uma aproximação da FO em um tempo de execução muito menor que a simulação (JALALI; NIEUWENHUYSE, 2015).

A fim de criar metamodelos mais fidedignos, nos últimos anos surgiram diversos métodos de metamodelagem na literatura. Barton (2009) comparou os principais métodos utilizados para este tipo de aplicação, a saber: RSM; RBF; *Kriging*; RNA; e *spline*. Entretanto, com o advento da inteligência artificial, muitos pesquisadores buscam incorporar técnicas de ML na simulação a fim de melhorar a velocidade de execução, precisão e confiabilidade, sendo que muitas destas aplicações estão relacionadas com: metamodelo e análise; predição de parâmetros; e tomada de decisão inteligente (DE LA FUENTE; ERAZO; SMITH, 2019).

Arthur (1959) define ML como sendo o campo de estudo que permite a aprendizagem sem uma programação explícita. Géron (2017) entende ML como a ciência (ou arte) da programação de computadores que aprende a partir de dados.

Carneiro (2019) discute que a partir do tipo de aprendizagem, as técnicas de ML podem ser divididas em aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço.

- Aprendizado supervisionado: neste método, o algoritmo aprende a partir de dados devidamente rotulados. Estes algoritmos podem ser de classificação, o qual visa atribuir classes aos dados, e regressivos, no qual os rotulos são valores numéricos (HASTIE; TIBSHIRANI; FRIEDMAN, 2008);
- Aprendizado não supervisionado: neste método, o algoritmo aprende com dados que não são rotulados, neste caso o objetivo é identificar padrões e agrupá-los, segundo determinadas características. Este tipo de ML é utilizado em problemas de clusterização (GÉRON, 2017);

- Aprendizado por reforço: neste tipo de aprendizagem o chamado “agente” observa o ambiente, seleciona e executa ações e recebe recompensas em troca, aprendendo por si só qual é a melhor estratégia para obter a maior recompensa ao longo do tempo (KARA; DOGAN, 2018).

Considerando seu treinamento por meio de dados amostrais rotulados, os metamodelos podem ser classificados como uma técnica de aprendizado supervisionado, que busca estabelecer um relacionamento de causa e efeito entre as variáveis de entrada e saída do modelo (AMARAN *et al.*, 2016; DUNKE; NICKEL, 2020; FU, 2002; XU *et al.*, 2016). Hassayebi *et al.* (2019) destacam que as aplicações de metamodelagem em problemas da indústria e serviços cresceram devido a sua capacidade de reduzir o número de simulações necessário no processo de otimização.

Metamodelos são aproximações matemáticas utilizadas para substituir modelos de simulação, cuja complexidade os torna caros de se executar (BARTON; MECKESHEIMER, 2006; LI *et al.*, 2010; GONZALEZ; JALALI; NIEUWENHUYSE, 2020). A metamodelagem pode ser definida como o processo de construção de modelos matemáticos capazes de capturar o relacionamento *input/output* da simulação, sendo capaz de gerar previsões para o valor da FO de determinada solução, em um tempo muito menor que o modelo de simulação original (DELLINO; KLEIJNEN; MELONI, 2015; KLEIJNEN, 2014; PARNIANIFARD *et al.*, 2018; WANG; YUAN; NG, 2020).

2.3.1. Algoritmos de *Machine Learning*

Assim como será apresentado na Seção 4.4.2, os algoritmos de metamodelagem mais frequentes em projetos de OvS identificados pela RSL e que serão utilizados neste trabalho são: *Gaussian Process* (GP), Regressão Polinomial (RP), RNA, *Ensembles* (*Randon Forest* e *Gradient Boosted Regression Tree*) e *Support Vector Machine* (SVM).

2.3.1.1. *Gaussian Process*

Também conhecido como *Kriging*, o GP é uma abordagem Bayesiana não paramétrica, com variáveis seguindo uma distribuição gaussiana (KLEIJNEN, 2014). A ideia principal é definir uma probabilidade para infinitas funções, adicionando pontos finitos que resultam em uma função que passa exatamente sobre esses pontos (ØSTERGÅRD; JENSEN; MAAGAARD, 2018). Nguyen-Tuong, Seeger e Peters (2009) afirmam que dado um conjunto de n dados de treino $\{x_i, y_i\}_{i=1}^n$, busca-se fazer com que a função $f(x_i)$ aprenda a partir da

transformação dos dados de entrada (x_i) em variáveis de resposta (y_i), convergindo na Eq. (2), sendo ϵ_i o ruído da função gaussiana, com média zero e variância σ_n^2 .

$$y_i = f(x_i) + \epsilon_i \quad (2)$$

O GP é um processo estocástico, no qual qualquer grupo finito de pontos de entrada do modelo são incorporados a uma distribuição Gaussiana, sendo escrita como $f(x) \sim GP[\mu(x), K(X, X)]$, onde $\mu(x)$ e $K(X, X)$ são a média e a função de covariância, respectivamente (WAN; REN, 2015). Neste caso, as funções de covariância são normalmente dadas por um kernel gaussiano, assim como descritos na Eq. (3) (NGUYEN-TUONG; SEEGER; PETERS, 2009).

$$k(x_p, x_q) = \sigma_s^2 \exp\left(-\frac{1}{2}(x_p - x_q)^T W(x_p - x_q)\right) \quad (3)$$

No qual σ_s^2 e W representam a variância e amplitude do kernel gaussiano. Além disto, observa-se na Eq. (4) a distribuição conjunta dos valores previstos ($f(x_*)$) e observados (y), dado um determinado *input* x_* . A partir do condicionamento da distribuição conjunta é possível extrair o valor médio previsto ($f(x_*)$) e sua variância associada ($V(x_*)$), demonstrado na Eq. (5) e Eq. (6) (NGUYEN-TUONG; SEEGER; PETERS, 2009).

$$\begin{bmatrix} y \\ f(x_*) \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} K(X, X) + \sigma_*^2 & k(X, x_*) \\ k(x_*, X) & k(x_*, x_*) \end{bmatrix}\right) \quad (4)$$

$$f(x_*) = k_*^T (K + \sigma_*^2 I)^{-1} y = k_*^T \alpha \quad (5)$$

$$V(x_*) = k(x_*, x_*) - k_*^T (K + \sigma_*^2 I)^{-1} k_* \quad (6)$$

No qual $k_* = k(X, x_*)$, $K = K(X, X)$ e α representa os vetores de previsão.

2.3.1.2. Regressão Polinomial

A RP é uma coleção de técnicas estatísticas e matemáticas aplicáveis ao desenvolvimento, melhoria e otimização de processos, sendo utilizada com êxito nas últimas décadas para representar sistemas estocásticos (PARNIANIFARD *et al.*, 2020a). A RP foi desenvolvida a mais de 50 anos para exploração e extrapolação (BARTON; MECKESHEIMER, 2006), e segundo Parnianifard *et al.* (2019) as aplicabilidade de RP encontradas na literatura são 1) aproximar a relação entre variáveis dependentes e independentes; 2) investigar e determinar as melhores condições de operação para processos; e 3) implementar maior robustez nas respostas.

Segundo Myers, Montgomery e Anderson-Cook (2016), o conceito da aproximação polinomial de uma determinada resposta é baseado na teoria de expansão da série de Taylor em

torno de um conjunto de dados. A formulação geral de uma regressão linear (RP de primeira ordem) pode ser representada pela Eq.(7) (PARNIANIFARD *et al.*, 2019).

$$y = f(X) = \hat{\beta}_0 + \sum_{i=1}^k \hat{\beta}_i x_i + \varepsilon \quad (7)$$

Sendo k o número de variáveis independentes. Caso a superfície de resposta seja mais complexa que não possa ser representada por uma RP de primeira ordem, pode-se empregar uma RP quadrática (segunda ordem). Sua formulação geral pode ser observada na Eq.(8) (DENGIZ; IÇ; BELGIN, 2015; OSORIO; BIERLAIRE, 2013; PARNIANIFARD *et al.*, 2019; PARNIANIFARD *et al.*, 2020a).

$$y = f(X) = \hat{\beta}_0 + \sum_{i=1}^k \hat{\beta}_i x_i + \sum_{i=1}^k \hat{\beta}_{ii} x_i^2 + \sum_{i=1}^k \sum_{i < i'=2}^k \hat{\beta}_{ii'} x_i x_{i'} + \varepsilon \quad (8)$$

Sendo $\hat{\beta}_0, \hat{\beta}_i, \hat{\beta}_{ii}$ e $\hat{\beta}_{ii'}$ são os coeficientes (desconhecidos) e ε é o componente de ruído do modelo. O número de expressões, p , de uma regressão linear é dado por $p = k + 1$, enquanto para regressão quadrática $p = \frac{1}{2}(k + 1)(k + 2)$, e para regressão cúbica $p = \frac{1}{6}(k + 1)(k + 2)(k + 3)$ (PARNIANIFARD *et al.*, 2019). Uma das grandes vantagens da RP é a interpretabilidade e adaptabilidade dos modelos perante diferentes tipos de dados, todavia Østergard, Jesen e Maagaard (2018) ressaltam que podem ser computacionalmente ineficientes quando comparado a expansões mais sofisticadas com alta dimensão, como o GP e SVM.

2.3.1.3. Redes Neurais Artificiais

As RNA's tem sido vastamente aplicadas em problemas de classificação, regressão e *time series*, e consiste em uma estrutura com um camada de entrada (*inputs layer*), uma ou mais camadas ocultas (*hidden layers*), e uma camada de saída (*output layer*) (HASSOUN, 1995). A estrutura geral de uma RNA e de um neurônio estão representados nas Figura 3.2 e Figura 3.3, respectivamente.

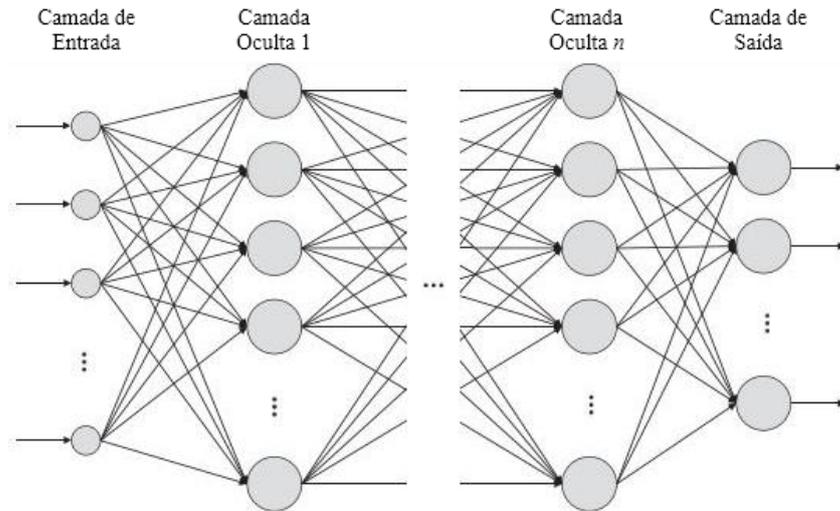


Figura 2.2 - Estrutura de uma rede neural artificial

Fonte: Adaptado de Storti *et al.* (2019)

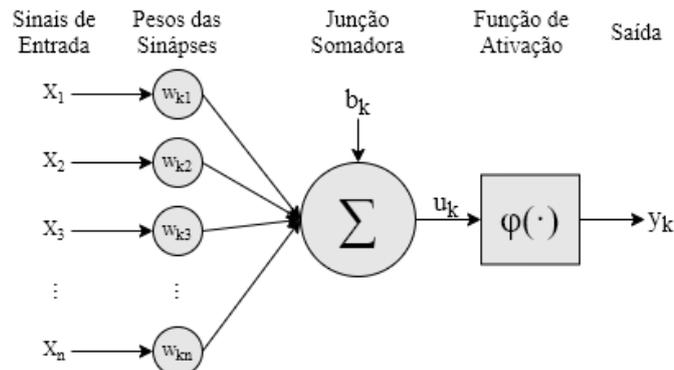


Figura 2.3 - Estrutura geral de um neurônio

Fonte: Adaptado de Storti *et al.* (2019)

De acordo com Li *et al.* (2010), a transmissão de informação é devida a conexões entre os neurônios, por meio das camadas, e, segundo os autores, uma RNA típica é representada por três camadas e apenas um neurônio na camada de saída, podendo ser representada pela seguinte formulação matemática:

$$\hat{y} = \hat{f}(X) = \sum_{j=1}^j w_j f \left(\sum_{i=1}^k v_{ij} f(x_i) + a_j \right) + \beta + \varepsilon \quad (9)$$

Sendo X um vetor de dimensão k , $f(\cdot)$ representa a função de transferência, ε é o erro aleatório com média 0, v_{ij} refere-se ao peso das conexões entre os i neurônios da camada de entrada e os j neurônios da camada oculta, a_j é a bias do j th neurônio da camada oculta, w_j é o peso da conexão entre o j th neurônio oculto e o neurônio de saída, j é o número total de neurônios na camada oculta e β é a bias do neurônio de saída (LI *et al.*, 2010).

De acordo com Oliveira *et al.* (2010), o ajuste dos pesos e bias é feito por algoritmos de treino supervisionado, sendo o *back-propagation* o mais utilizado. Nele, o modelo é alimentado

com entradas padrões e o sinal é propagado até a camada de saída a fim de gerar uma previsão, logo calcula-se o erro entre o valor predito e valor real conhecido, a principal métrica utilizada é o erro médio quadrático. Este erro é então utilizado para ajustar os pesos entre as conexões dos neurônios de saída com os neurônios ocultos, e destes com os neurônios de entrada.

2.3.1.4. *Random Forest*

Em seus estudos, Breiman (2001) introduziu o método conhecido como *Random Forest* (RF), que consiste em combinar um grande número de árvores de decisão de forma aleatória e não-correlacionadas, a fim de obter uma previsão com base no voto médio de todos indivíduos (LI *et al.*, 2020). O RF é um dos algoritmos de ML mais eficientes em termos de acurácia e velocidade de execução em bases de dados extensas, além de ser um método simples e de fácil interpretação das variáveis e resultados (CRISCI *et al.*, 2012).

Dai *et al.* (2018) explicam que a RF é uma combinação de árvores de decisão $\{h(x, \theta_k), k = 1, 2, \dots\}$, no qual cada árvore recebe um vetor aleatório $\{\theta_k\}$. Em problemas de regressão os *outputs* são numéricos e assume-se que os dados de treino são independentemente e identicamente distribuídos (BREIMAN, 2001). Para o vetor X e *output* Y o erro médio quadrático associado a cada árvore $h(x)$ é definido pela Eq. (10) (DAI *et al.*, 2018).

$$E_{X,Y}(Y - h(X))^2 \quad (10)$$

O procedimento para ajustar a RF é simples, o primeiro passo é ajustar todas as m árvores independentes, em seguida, a cada interação é obtida uma amostra de tamanho N e selecionado aleatoriamente k atributos a fim de gerar novas árvores de forma não correlacionadas (DE LA FUENTE; SMITH, 2017). As previsões individuais de cada m -ésima árvore são guiadas por um conjunto de regras *if-then-else*, caracterizadas por Θ (HASTIE *et al.*, 2008). Para Breiman (2001), a previsão da RF é calculada pela média das k árvores $h(x)$ pertencentes a floresta, sendo dada pela Eq.(11).

$$f(X) = \frac{1}{M} \sum_{m=1}^M T(X; \theta_m) \quad (11)$$

Sendo M o número de árvores e $T(X; \theta_m)$ o modelo que representa uma única árvore, com X o vetor de dados de treino e θ_m o conjunto de regras destas árvores.

2.3.1.5. *Gradient Boosted Trees*

Boosting é uma método proposto inicialmente por Schapire (1990) e visa construir um modelo de alta acurácia com base na convergência de algoritmos com baixa aprendizagem, que no caso de *Gradient Boosted Trees* (GBT) são árvores de decisão. Friedman (2002) propôs uma

variação deste método chamado *Stochastic Gradient Boosting*, sendo introduzida a estocasticidade no procedimento proposto por Schapire (1990). Nele, Friedman (2002) ensina que a cada interação do algoritmo, as árvores são treinadas utilizando uma amostra aleatória \tilde{N} dos dados de treino, sem reposição, de modo que $\tilde{N} < N$, no qual N é o tamanho dos dados de treino. O algoritmo GBT se diferencia de métodos de *bagging*, pois o primeiro visa construir modelos (árvores) sequencialmente, enquanto o segundo é ajustado com base em modelos construídos em paralelo (XIA *et al.*, 2017).

Este algoritmo tem como base conjuntos de árvores de decisão, sendo que a cada interação uma nova árvore é gerada com base nos resíduos das anteriores (DE LA FUENTE; SMITH, 2017). Dado um conjunto de dados $\{x_i; y_i\}$ de tamanho N , os atributos de entrada são representados por x_i , e y_i é a variável de resposta, sendo o objetivo do algoritmo determinar a função $F^*(x)$ que expressa o relacionamento entre x e y , de forma a minimizar a função de perda dada pela Eq. (12). Este relacionamento é aproximado de forma aditiva pela Eq. (13) (XIA *et al.*, 2017).

$$F^*(x) = \arg \min_f \Psi(y, F(x)) \quad (12)$$

$$F_m^*(x) = \sum_{m=0}^M \beta_m h(x; \theta_m) \quad (13)$$

Sendo os *base learners* ($h(x; \theta_m)$) ideais representados por $\beta_m h(x; \theta_m)$, no qual β_m e θ_m são os coeficientes e parâmetros ótimos do *base learner*, respectivamente. Friedman (2002) apresenta o pseudocódigo para o GBT assim como descrito no Algoritmo 1:

Algoritmo 1: *Stochastic Gradient-boosted trees*

```

1   $F_0(x) = \arg \min_{\gamma} \Psi(y_i, \gamma)$ 
2  For  $m = 1$  to  $M$  do:
3     $\{\pi(i)\}_1^N = \text{rand}_{\text{perm}} \{i\}_1^N$ 
4     $\tilde{y}_{\pi(i)m} = - \left[ \frac{\partial \Psi(y_{\pi(i)}, F(x_{\pi(i)}))}{\partial F(x_{\pi(i)})} \right]_{F(x)=F_{m-1}(x)}$ ,  $i = 1, \tilde{N}$ 
5     $\{R_{lm}\}_1^N = L - \text{terminal node tree}(\{\tilde{y}_{\pi(i)m}, x_{\pi(i)}\}_1^{\tilde{N}})$ 
6     $\gamma_{lm} = \arg \min_{\gamma} \sum_{x_{\pi(i)} \in R_{lm}} \Psi(y_{\pi(i)}, F_{m-1}(x_{\pi(i)}) + \gamma)$ 
7     $F_m(x) = F_{m-1}(x) + \nu \cdot \gamma_{lm} I(x \in R_{lm})$ 
8  End For.
```

Fonte: Friedman (2002).

No qual $\Psi(y_i, \gamma)$ representa a função de perda, R_{lm} representa cada região formada por l nós das m -ésima árvores, $\{\pi(i)\}_1^N$ representa a permutação dos inteiros $(1, \dots, N)$ dos dados de treino $\{y_i, x_i\}_1^N$. A linha (4) denota o gradiente da função de perda (ou pseudo-resíduos) da última interação, γ_{lm} é definido como o conjunto de coeficientes de expansão ótimos que correspondem a cada nó da árvore na m -ésima interação.

2.3.1.6. Support Vector Machine

Proposto por Vapnik (1999), o SVM é um dos métodos de ML mais populares dos últimos anos (CAO *et al.*, 2020). Foi inicialmente construído para problemas de classificação, mas também vem sendo muito utilizado em problemas de regressão (CAO *et al.*, 2020, HASTIE *et al.*, 2008).

Em problemas de classificação, o método busca o limite que separa os dados de treinamento em suas respectivas classes, com o mínimo de erro possível. Neste caso, é definido o hiperplano com base em vetores construídos pelas observações que estão localizadas nos limites, logo os dados que se encontram lados distintos do hiperplano são alocados em classes diferentes (ØSTERGÅRD; JENSEN; MAAGAARD, 2018).

Jeng (2006) explica, que neste tipo de problema, o SVM tenta encontrar a margem máxima entre as duas classes através da combinação linear dos vetores, sendo solucionado por uma programação quadrática com restrições lineares. Para os autores, com a introdução do parâmetro \mathcal{E} , que representa a função de perda, este algoritmo pode ser estendido à solução de problemas de regressão não lineares. Li *et al.* (2010) apresentam a formulação matemática para o SVM (Eq. 14):

$$\hat{y} = \hat{f}(X) = w\phi(X) + b \quad (14)$$

Sendo $X \in \mathfrak{R}^k$ é o input k-dimensional, ϕ é a transformação não-linear de \mathfrak{R}^k para \mathfrak{R}^h , com $h > k$, e $\hat{f}(X)$ descreve o hiperplano em \mathfrak{R}^h . Os autores ressaltam que os coeficientes w e b são obtidos pela minimização da função de risco, dada pela Eq. (15).

$$R_{svm}(C) = C \sum_{i=1}^n C(\hat{f}(X_i), y_i) + \frac{1}{2} \|w\|^2 \quad (15)$$

Sendo:

$$C(\hat{f}(X_i), y_i) = \begin{cases} |\hat{f}(X) - y| - \mathcal{E} & \text{se } |\hat{f}(X) - y| \geq \mathcal{E} \\ 0 & \text{Caso Contrário} \end{cases} \quad (16)$$

Solucionado o problema de minimização, a função de previsão final do método SVM é obtida pela Eq. (17) (VAPNIK, 1999).

$$f(X, a_i, a_i^*) = \sum_{i=1}^n (a_i - a_i^*)K(X, X_i) + b \quad (17)$$

No qual X é o novo input do modelo, $K(X, X_i)$ é a função Kerniana que representa a distância entre X e X_i e a_i, a_i^* e b são os coeficientes obtidos pela minimização da função de risco (LI *et al.*, 2010).

Oliveira *et al.* (2010) afirmam que a boa eficiência do SVM depende da escolha correta dos hiper-parâmetros C , ϵ (\mathcal{E}) e Kernel (K). De acordo com Cherkassky e Ma (2004) pode-se calcular o valor ideal de C segundo a Eq. (18):

$$C = \max(|\bar{y} + 3\sigma_y|, |\bar{y} - 3\sigma_y|) \quad (18)$$

No qual \bar{y} e σ_y são a média e o desvio padrão dos valores de y nos dados de treino. Para o cálculo de \mathcal{E} , os autores recomendam uma primeira execução do algoritmo com os parametros default, com isto pode-se calcular o erro de previsão δ associado a base de dados, onde $\delta = \hat{y} - y_i, (i = 1, \dots, n)$.

$$\mathcal{E} = 3\sigma \sqrt{\frac{\ln n}{n}} \quad (19)$$

Sendo σ o desvio padrão do erro δ , e n o número de exemplos da base de dados.

2.3.2. Métricas de Performance

Um importante passo na construção de metamodelos é a sua avaliação, a fim de se escolher o modelo que melhor se adeque aos dados, assim como guiar a seleção e/ou otimização de seus parâmetros. Bergmeir e Benítez (2012) destaca o *Mean Square Error* (MSE) ou *Root MSE* (RMSE), o *Mean Absolut Error* (MAE) e o *Mean Absolute Percentage Error* (MAPE) como as principais métricas de erro utilizadas na literatura. Segundo Hu, Palta e Shao (2006), outra importante métrica em problemas de regressão é a correlação quadrática (R^2).

Considerando y_i como sendo o valor observado (real) no ponto i e \hat{y}_i o valor previsto, e n o tamanho dos dados de teste, pode-se calcular o RMSE, MAE e MAPE, assim como demonstrado na Eq.(20), Eq.(21) e Eq.(22), respectivamente (BERGMEIR; BENÍTEZ, 2012; CHAI; DRAXLER, 2014).

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (20)$$

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (21)$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| 100 \frac{y_i - \hat{y}_i}{y_i} \right| \quad (22)$$

Quando se trata de métricas de avaliação de metamodelos, não se possui um consenso universal sobre valores para a acurácia de um modelo. Todavia, é plausível definir que um

metamodelo com $R^2 > 0.90$ é razoável, $R^2 > 0.95$ possui boa acurácia e $R^2 > 0.99$ é considerado quase perfeito (ØSTERGÅRD; JENSEN; MAAGAARD, 2018).

O R^2 é uma métrica de acurácia com uma amplitude que varia de 0 a 1. Como indicado na Eq.(23), seu cálculo é dado pela divisão da soma dos quadrados dos resíduos $(y_i - \hat{y}_i)^2$ pela soma total dos quadrados $(y_i - \bar{y}_i)^2$, sendo que altos valores de R^2 significam um bom ajuste entre o modelo e os dados de treino, porém não garante que o erro foi reduzido (HU; PALTA; SHAO, 2006).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (23)$$

Sendo:

$$\bar{y}_i = \frac{1}{n} \sum_{i=1}^n y_i \quad (24)$$

2.3.3. Otimização de Hiper-parâmetros via Algoritmo Genético

A escolha adequada dos hiper-parâmetros dos algoritmos de ML é uma etapa fundamental quando se deseja aumentar a capacidade de aprendizagem destes algoritmos. Perante a inviabilidade da definição manual destes parâmetros, torna-se imprescindível o uso de técnicas de otimização para encontrar a configuração que extraia a melhor performance do algoritmo (XIA *et al.*, 2017).

Atualmente existem diversos métodos de otimização para seleção de hiper-parâmetros, tais como AG, colônia de formigas, *grid search*, *bayesian optimization*, entre outros (HÜLLEN *et al.*, 2020; KIM; BOUKOUVALA, 2020; LI; TAN; NG, 2019). Para este trabalho optou-se pelo uso de AG devido a sua velocidade de execução, capacidade de adaptação perante diverentes tipos de variáveis e não requerer a definição de valores iniciais para performar a otimização. A Tabela 2.1 apresenta um breve comparativo entre os principais métodos de otimização de hiper-parâmetros.

Tabela 2.1 -Comparativo de métodos comuns para otimização de hiper-parâmetros

Método de otimização	Pontos fortes	Limitações	Complexidade de tempo
<i>Grid Search</i>	Simples	Consome muito tempo Eficiente apenas para variáveis categóricas	n^k
<i>Rando Search</i>	Mais eficiente que Grid Search Permite paralelismo	Não considera resultados prévios Não é eficiente com variáveis condicionais	n
<i>Gradient-based</i>	Convergencia rápida para variáveis contínuas	Suporta apenas variáveis contínuas	n^k

		Detecta apenas ótimos locais	
<i>Bayesian GP</i>	Convergencia rápida para variáveis contínuas	Pouca capacidade de paralelismo Não é eficiente para variáveis condicionais	n^3
<i>Bayesian TPE</i>	Eficiente para todos os tipos de variáveis Mantém dependências condicionais	Pouca capacidade de paralelismo	$n \log n$
<i>Sequential model-based algorithm configuration</i>	Eficiente para todos os tipos de variáveis	Pouca capacidade de paralelismo Não é eficiente para variáveis condicionais	$n \log n$
<i>Hyperband</i>	Permite paralelismo	Requere pequenos subconjuntos para ser representativo	$n \log n$
<i>Bayesian hyperBand</i>	Eficiente para todos os tipos de variáveis Permite paralelismo	Requere pequenos subconjuntos para ser representativo	$n \log n$
<i>Genetic algorithm</i>	Eficiente para todos os tipos de variáveis Não requer uma boa inicialização	Pouca capacidade de paralelismo	n^2
<i>Particle swarm</i>	Permite paralelismo Eficiente para todos os tipos de variáveis	Requer inicialização apropriada	$n \log n$

k é o número de hiper-parâmetros e n é número de valores para os hiper-parâmetros.

Fonte: Adaptado de Yang e Shami (2020).

O AG é uma técnica heurística inspirada na teoria biológica da seleção natural. Proposto inicialmente por De Jong (1975) e Holland (1975), é baseada no princípio da sobrevivência do mais forte de Darwin, sendo utilizado com sucesso em problemas de programação linear e não linear (BASHIRI; GERANMAYEH, 2011; LESSMANN; STAHLBOCK; CRONE, 2006). No AG, o espaço de busca é composto por soluções candidatas, sendo cada uma representada por um cromossomo (ALABAS; ALTIPARMAK; DENGIZ, 2002). O cromossomo é composto por genes, representando uma codificação dos parâmetros de uma solução candidata e um valor de aptidão, ou FO, para alcançar a melhor solução do problema. O objetivo do AG é promover a combinação dos genes de modo a obter novos cromossomos com melhores valores de FO (LI *et al.*, 2010).

O algoritmo genético é composto por três operadores básicos: reprodução, cruzamento e mutação. A reprodução é o processo em que os parâmetros, ou genes, de um indivíduo são copiados considerando seus valores de FO, na qual os indivíduos com maior de FO possuem maior probabilidade de gerar um ou mais descendentes para próxima geração. Quando um parâmetro é escolhido para reprodução, esta cria uma réplica exata de si mesmo, e então o

operador de cruzamento é responsável por combinar aleatoriamente os parâmetros recém-produzidos. Por último, a mutação é a ação que gera uma modificação nos valores dos parâmetros, assumindo um papel secundário devido a taxa de mutação ser, em geral, baixa (ALABAS; ALTIPARMAK; DENGIZ, 2002; STORTI *et al.*, 2019).

O uso do AG para otimização de hiper-parâmetros pode ser observada em diversos trabalhos (*e.g.*, HOU *et al.*, 2019; HÜLLEN *et al.*, 2020). Este método avalia diversas combinações possíveis para os parâmetros $\{a_1, \dots, a_i\}$ do algoritmo de ML, e para cada combinação é utilizada a técnica de *cross-validation* para calcular o erro associado a esta. Considerada uma das mais importantes técnicas de avaliação de métodos de regressão e classificação, a *k-fold cross-validation* divide igualmente e aleatoriamente os dados de treino em k partes, utilizando-se $k-1$ partes para treinamento do algoritmo e o restante para teste. Cada arranjo tem seu erro calculado k vezes, sendo que em cada interação é selecionado uma parte k'_i diferente para teste, com $i = \{1, \dots, k\}$. Então, a performance é medida com base na média dos erros, nas k interações de validação, performando uma métrica mais robusta do que se medida em apenas uma observação (BERGMEIR; BENÍTEZ, 2012). Roy e Datta (2019) sugerem o uso de *10-fold cross-validation*.

2.3.4. Latin Hypercube Design

Segundo Montevechi *et al.* (2012), um experimento pode ser entendido como a realização de uma série de testes em que mudanças propositais nas variáveis de entrada são feitas com o intuito de identificar e analisar os efeitos destas mudanças nas variáveis de resposta do sistema. Segundo os autores, DOE corresponde a um conjunto de métodos que visam delinear e conduzir experimentos. Diversas técnicas de DOE podem ser encontradas na literatura, tais como *orthogonal array* (MIRANDA *et al.*, 2017), *Central Composite Design* (CCD) (VILLARREAL-MARROQUÍN *et al.*, 2013), planejamento fatorial (DENGIZ; BEKTAS; ULTANIR, 2006), entre outros.

Baseado na RSL apresentada no Capítulo 3, o método de DOE mais comum em projetos de metamodelagem é o *Latin Hypercube Design* (LHD). Segundo Parnianifard *et al.* (2020), o LHD é uma técnica de DOE computacional baseado no conceito de *space-filling*, onde para n variáveis de entrada, m pontos amostrais são obtidos aleatoriamente dentro de m intervalos de igual probabilidade. A Figura 2.4 demonstra um exemplo de arranjo LHD com $n = 2$ (variáveis de decisão) e $m = 7$ (tamanho amostral).

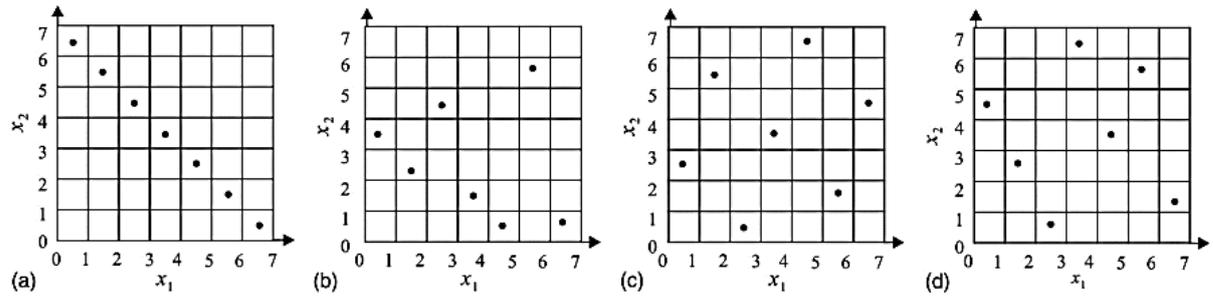


Figura 2.4 – Exemplos de arranjos LHD com 2 variáveis de entrada e 7 pontos amostrais

Fonte: Adaptado de Luo, Ji e Lu (2019)

De acordo com Parnianifard *et al.* (2020), os procedimentos do LHD são:

- As faixas de valores para cada variável são divididas em m subfaixas de magnitude equiprováveis. O número de pontos amostrais m é definido pelo pesquisador e, em geral, é maior que o total de pontos produzidos pelo CCD.
- O segundo passo é alocar os m pontos entre os limites inferiores e superiores de cada intervalo, sendo que cada ponto (1, 2, ..., m) aparece apenas em uma linha e coluna do espaço amostral.

3. REVISÃO SISTEMÁTICA DE LITERATURA

Este capítulo está inserido na primeira etapa do método de pesquisa empregado nesta dissertação, ou seja, a fase de contextualização do problema.

3.1. Revisões de Literatura Sobre Otimização via Simulação por Metamodelagem

Na busca por trabalhos envolvendo revisões de literatura (RL) que tratem de OvSM ou temas correlatos foi realizada em julho de 2020 uma busca nas bases de dados científica *Scopus* e *Web of Science*. Nesta etapa foram pesquisados, com as lógicas booleanas (AND/OR), os termos correlatos a metamodelagem (*metamodel*, *metamodeling*, *meta-model*, ou *machine learning*), otimização via simulação (*Simulation optimization*, *simulation-based optimization*, ou *optimization via simulation*), e revisão de literatura (*literature review*, *state of the art*, *overview* ou *survey*), sendo aplicado apenas o filtro para documentos em inglês.

Nesta pesquisa foram encontradas 12 RLs sobre temas correlatos a OvSM, no entanto, estas focaram em determinada etapa de um projeto de metamodelagem e/ou um ou poucos algoritmos de metamodelagem e tipo de simulação, sendo que em nenhuma delas encontrou-se uma discussão completa sobre os conhecimentos de OvSM que devem ser considerados por acadêmicos e práticos no desenvolvimento de seus projetos. Desta maneira o presente capítulo demonstra sua contribuição a partir de uma análise abrangente sobre OvSM, sob a ótica e rigor científico das metodologias de RSL (XIAO, YU; WATSON, 2019) e CIMO (*Context, Intervention, Mechanism, Outcomes*) (DENYER; TRANFIELD; VAN AKEN, 2008).

O primeiro trabalho encontrado foi publicado no *Winter Simulation Conference* em 1986, no qual foi discutido a técnica de *frequency domain experimentation* e RSM para tratar problemas com variáveis discretas (SCHRUBEN, 1986). No mesmo congresso, vinte e três anos após o trabalho de Schruben (1986), Barton (2009) apresentou uma visão geral sobre as técnicas de metamodelagem em SED. Neste trabalho foram discutidos os principais algoritmos utilizados em metamodelagem na literatura, *i.e.*, RSM, RNA, *Kriging*, *spline*, RBF, provendo um *framework* para condução de projetos de metamodelagem de busca local ou global. No mesmo ano, Kleijnen (2009) revisou a metamodelagem por *Kriging* e sua associação com técnicas como *bootstrapping*, DOE, análise de sensibilidade e otimização.

Em 2010, um artigo comparou sistematicamente cinco algoritmos de metamodelagem (*i.e.*, RNA, RBF, *Kriging*, SVM, e *multivariate regression splines*) para OvS em sistemas de suporte a decisão (LI *et al.*, 2010). Em 2014 foi realizada uma *survey* sobre metamodelagem

por *Kriging* juntamente com *parametric* e *distribution-free bootstrepping* em problemas de otimização via simulação determinística e/ou estocástica (KLEIJNEN, 2014). No mesmo ano, Figueira e Almada-Lobo (2014) estudaram as classificações e taxonomias dos métodos híbridos de OvS, explanando também questões sobre os procedimentos e métodos relacionados a metamodelagem.

Em 2015, Dellino *et al.* (2015) descreveram, em seu capítulo de livro, os conceitos em torno de *robust optimization via Kriging metamodeling*, aplicando-os em um modelo de inventário de *economic order quantity*. Também em 2015, foi realizada uma *survey* sobre o uso prático das técnicas de *search experimentation*, *i.e.*, OvS, DOE, e metamodelagem, aplicados em SED (HOAD; MONKS; O'BRIEN, 2015). Dois anos depois foi desenvolvida uma RL englobando uma revisão geral sobre diversos algoritmos de metamodelagem (*i.e.*, regressão polinomial, *Kriging*, RBF, SVM, e metamodelos mistos), técnicas de DOE (*i.e.*, *geometrical*, *statistical*, e *optimal designs*), e mecanismos de busca globais (*i.e.*, *expected improvement*, *efficient global optimization*, *bumpiness*, etc), aplicadas em problemas com variáveis contínuas (VU *et al.*, 2017). No mesmo ano outras duas RLs revisaram metamodelagem via *Kriging* e regressão polinomial (KLEIJNEN, 2017a, b).

Por fim, apenas um trabalho utilizou a metodologia de RSL ao estudar metamodelagem em *robust black-box* OvS sujeitos a incertezas (PARNIANIFARD *et al.*, 2019). A integridade dos trabalhos mencionados anteriormente foram base para o desenvolvimento desta RSL acerca de OvSM. Vale ressaltar que esta RSL não pretende esgotar a vasta literatura sobre OvSM, mas sim prover a acadêmicos e práticos da área uma visão ampla sobre o tema, ao consolidar e explanar sobre os projetos existentes em OvSM, bem como um melhor delineamento da fronteira do conhecimento na área, identificando e clarificando as oportunidades e desafios presentes na atualidade do tema.

3.2. O Método de Revisão Sistemática de Literatura

A RSL pode ser entendida como sendo a metodologia de geração do embasamento teórico-científico necessário para compreensão de um tema por meio da coleta, compreensão, síntese e avaliação de um conjunto de trabalhos científicos (LEVY; ELLIS, 2006; XIAO; WATSON, 2019). De acordo com Sousa Junior *et al.* (2019), a RSL auxilia o pesquisador na condução de todas as etapas metodológicas, tais como na definição do problema, seleção do método, coleta de dados, e análise, contribuindo para mitigação de falhas nas conclusões do trabalho e permitindo uma maior clareza sobre os procedimentos de pesquisa empregados.

A RSL não deve ser considerada como uma RL tradicional, mas sim como um projeto de pesquisa que visa explorar de forma clara uma questão específica utilizando a literatura existente. A RSL se difere das demais metodologias de RL pelo seu princípio exato, a qual o pesquisador deve pré-especificar e deixar claro aos leitores seus critérios de inclusão/exclusão de estudos (DENYER; TRANFIELD, 2009). Oliveira *et al.* (2016) apontam como uma das grandes vantagens da RSL, sobre os métodos tradicionais de RL, é que ela permite evitar a tendenciosidade e distorções nos dados. O principal intuito da RSL é promover o avanço e desenvolvimento de novos estudos, uma vez que para ultrapassar a fronteira do conhecimento deve-se saber onde está fronteira realmente se encontra.

3.3. Aplicação do Método de Revisão Sistemática de Literatura

Com o propósito de investigar sistematicamente sobre o conhecimento em OvSM, identificando as fronteiras e lacunas da literatura existente e prover novas diretrizes para trabalhos futuros, esta RSL adotará os mesmos procedimentos adotados nos trabalhos de Oliveira *et al.* (2016) e Sousa Junior *et al.* (2019), os quais são conduzidos em quatro fases principais: planejamento; pesquisa/triagem; análise/síntese; e apresentação de resultados.

3.3.1. Planejamento

No intuito de melhor investigar o campo de pesquisa, esta etapa foi desenvolvida a partir de um painel com especialistas da área. Este painel contou com a participação de 3 doutores com experiência acadêmica e prática em projetos de OvSM, além de estudantes de Ph.D. e M.Sc. que pesquisam sobre o mesmo tema.

A motivação principal deste estudo, surge do encontro entre o crescimento expressivo em publicações acerca de OvSM, sua aplicação nas mais diversas áreas da engenharia, e a falta de trabalhos que reúnam e discutam de forma abrangente e sistemática o conhecimento sobre esse assunto. Espera-se que os resultados deste estudo possam auxiliar em todas as etapas de projetos desta área, assim como apresentar o estado da arte em OvSM, compilando o que já foi estudado e, principalmente, as oportunidades de novos trabalhos.

Deste modo, os objetivos desta RSL são:

- Identificar e explorar os principais aspectos relacionados a estudos de OvSM;
- Analisar e sumarizar os resultados encontrados;
- Discutir os resultados, delimitar o estado da arte, lacunas e tendências a serem exploradas.

De modo a atingir este propósito, a formulação das questões de pesquisa (QP) é uma etapa fundamental na extração e análise do conhecimento intrínseco do tema. Em respeito a isto, diversos *frameworks* para estruturação de QPs foram descritos na literatura, tais como: o método PICO, PICOS ou PICOC (*Population, Intervention, Comparison, Outcomes, e Context*) (KLODA; BORUFF; CAVALCANTE, 2020), SPICE (*Setting, Perspective, Intervention, Comparison, e Evaluation*) (BOOTH, 2006), SPIDER (*Sample, Phenomenon of interest, Design, Evaluation, e Research type*) (METHLEY *et al.*, 2014), ECLIPSE (*Expectation, Client, Location, Impact, Professionals, e Service*) (WILDRIDGE; BELL, 2002), entre outros.

Esta pesquisa utilizará o *framework* CIMO-logic proposto por Denyer *et al.* (2008), e envolve a combinação do *Context* do problema que necessita de uma *Intervention* gerada por meio de um *Mechanism* a fim de produzir os *Outcomes*. Essa abordagem já foi utilizada por diversos autores para estudar problemas de engenharia (AOUADNI; AOUADNI; REBAÏ, 2019; BĂLAN, 2018; MAKHASHEN *et al.*, 2020; COSTA; SOARES; DE SOUSA, 2020; ILK; SHANG; GOES, 2020; KOCHAN; NOWICKI, 2018; TANILA; TENHUNEN; HIRVONEN, 2020). Neste sentido, as QPs desta pesquisa foram divididas em quatro elementos delineados pelo método CIMO-logic:

- *Context*: Em quais problemas reais ou teóricos...
- *Intervention*: ...utiliza-se metamodelagem...
- *Mechanism*: ...combinada a técnicas de otimização da simulação...
- *Outcomes*: ...para encontrar a solução de melhor performance dentro do escopo e restrições do projeto.

Dado estes objetivos, as QPs foram formuladas em atendimento ao escopo do estudo e de modo a identificar as lacunas existentes na literatura, e incluem:

- QP1) Qual o papel da metamodelagem em projetos de OvS e quais são suas principais aplicações? (*Context*)
- QP2) Quais as principais técnicas de simulação e otimização utilizadas em conjunto com metamodelagem? (*Mechanism*)
- QP3) Quais os algoritmos de metamodelagem mais utilizados em OvSM? (*Intervention*)
- QP4) Quais as estratégias de amostragem utilizadas? (*Intervention*)
- QP5) Como os resultados são mensurados/validados? (*Outcomes*)
- QP6) Qual é o atual estado da arte da pesquisa sobre OvSM? (*Context*)

No propósito de responder as QPs, as informações a serem extraídas dos artigos foram divididas em três categorias, assim como mostrado na Figura 3.1.



Figura 3.1 – Pilares da pesquisa e dados extraídos dos artigos

A primeira categoria investiga a natureza dos trabalhos, a qual se refere ao setor produtivo foco do estudo, qual tipo de problema está sendo tratado, se o problema é de cunho real ou teórico (quando baseado em dados da literatura), e como são as variáveis de decisão e objetivos dos problemas. A segunda categoria trabalha questões ligadas ao método empregado, tais como: o tipo de simulação, os algoritmos de metamodelagem utilizados, quais mecanismos de busca usados na otimização, estratégias de amostragem, softwares, e como foram medidos os resultados dos estudos. A terceira, e última categoria, abrange o estado da arte, envolvendo uma breve análise dos dados bibliométricos da literatura.

3.3.2. Pesquisa e Triagem dos Artigos

A pesquisa exploratória sobre o tema foi realizada nas bases de dados científicas *Scopus* e *Web of Science* (WoS), uma vez que estas bases são consideradas as principais fontes de dados de citações (MONGEON; PAUL-HUS, 2016) e possuem uma boa representatividade sobre as publicações acerca do tema (AGHAEI *et al.*, 2013). Sendo assim, foi pesquisado nos mecanismos de busca de ambas as bases de dados os termos relacionadas a metamodelagem (*metamodel*, *metamodeling*, *meta-model*, e *machine learning*) e OvS (*Simulation optimization*, *simulation-based optimization*, *optimization via simulation*, e OvS), utilizando-se de Lógica Booleana (AND/OR) como parte do processo. A meta-busca retornou 188 artigos da WoS e 293 da Scopus, gerando um total de 481 trabalhos.

A fase inicial de seleção dos artigos a serem baixados foi dada pela leitura dos títulos/resumos dos artigos em busca daqueles que atendam aos seguintes critérios de inclusão: (a) possuir no título/resumo/palavras-chave os termos de busca para metamodelagem em OvS;

(b) publicados em periódicos com revisão aos pares; (c) escritos na língua inglesa; e (d) possuir o artigo completo disponível. Após a aplicação dos critérios de inclusão, 209 trabalhos foram excluídos da busca por descumprirem pelo menos um dos critérios definidos anteriormente: 3 (0,6%) não possuíam os termos de busca no título, resumo ou palavras-chave, 203 (42,2%) não foram publicados em periódicos com revisão aos pares, 5 (0,8%) não foram escritos na língua inglesa, e 1 não estava disponível para download. No total, 269 foram baixados para etapa de avaliação de qualidade, sendo 144 da Scopus e 125 da WoS. Após isto, 104 artigos foram removidos por duplicidade.

A avaliação de qualidade consiste na leitura do título, resumo, objetivos, métodos, resultados e discussões. Nesta etapa foram excluídos da análise 21 artigos devido aos seguintes motivos: 11 (52,3%) citam o termo “*machine learning*” mas não utiliza nem ML nem metamodelagem; 8 (38,1%) utilizam a metamodelagem/ML para finalidades diferentes de OvS; e 2 (9,5%) não utilizam simulação. Seguindo a avaliação de qualidade 143 artigos foram selecionados para leitura completa, extração de informações e análise dos dados. A Figura 3.2 apresenta os procedimentos de pesquisa/triagem.

Procedimentos de Pesquisa/Triagem			
	Scopus	WoS	Total
Seleção das bases de dados			
Pesquisa por artigos	n = 293	n = 188	n = 481
Triagem	n = 144	n = 125	n = 165
Avaliar qualidade	n = 126	n = 121	n = 143

Figura 3.2 - Processo de busca/triagem

A Figura 3.2 resume os resultados da fase de busca/triagem, sendo que após a fase de pesquisa da literatura foram baixados 293 artigos da Scopus e 188 da WoS. Na triagem foram incluídos 144 artigos da Scopus e 21 da WoS, ressaltando-se que 104 artigos elegíveis da WoS foram excluídos por estarem duplicados em relação a Scopus. Por fim, na fase de avaliação de qualidade, foram selecionados 126 trabalhos da Scopus e 17 da WoS, para análise e extração de dados.

3.3.3. Análise e Síntese

A síntese das informações extraídas dos 143 artigos foi realizada com o auxílio do software *Microsoft Excel*[®] e R. Para responder as QPs os artigos foram avaliados em 15 itens, assim como demonstra a Figura 4.1, e correspondem a: definição do problema; setor produtivo; origem dos dados; tipos de variáveis e objetivos; algoritmos de metamodelagem; estratégias de

amostragem; métodos de simulação e otimização; softwares empregados; indicadores de performance e método de validação; e principais autores, periódicos, nacionalidades, filiação, ano de publicação, e palavras-chave adotados. Posteriormente a extração dos dados foram realizadas padronizações de nomenclaturas similares, *e.g. gaussian process* e *Kriging* foram considerados como o mesmo algoritmo, e *Latin Hypercube Design* e *Latin Hypercube Sampling* dizem respeito a mesma técnica de DOE.

Após a síntese, a estatística descritiva dos dados, tabelas e gráficos foram feitos com auxílio do *Software R*. A análise dos resultados foi realizada em consonância com as QPs da pesquisa e relacionaram as três categorias mencionadas anteriormente, sendo considerada a base de todos os resultados e conclusões que suportaram os autores no desenvolvimento do corpo de evidências sobre as principais perspectivas em OvSM discutidas nesta RSL.

3.3.4. Apresentação dos Resultados

Os resultados foram resumidos em gráficos e tabelas para permitir a discussão sobre este tópico, revelando as percepções mais contundentes desta RSL. Com base nesta análise da interação entre OvS e metamodelagem, foi possível refletir e identificar as lacunas e percepções que cercam o desenvolvimento do tema na comunidade científica.

Após as análises e sínteses, os principais resultados estão resumidos nas Tabelas 4.1 – 4.7 na sequência, na qual serão respondidas as QPs e na qual os autores discutirão o estado da arte sobre OvSM ao apresentarem as práticas presentes na literatura e as possibilidades de futuras pesquisas em OvSM. Os 143 artigos foram divididos em duas categorias quanto a origem dos problemas, podendo ser reais ou teóricos, de modo a elucidar como está sendo a integração entre a academia e problemas práticos da indústria/serviços.

3.4. Resultados e discussões da Revisão Sistemática de Literatura

Esta seção tem o objetivo de apresentar os dados obtidos dos artigos analisados e responder as QPs, sendo discutidos na ordem em que estão dispostos os pilares desta pesquisa, *i.e.*, natureza da pesquisa, métodos adotados, e estado da arte.

3.4.1. Natureza da pesquisa

O propósito desta seção é discutir sobre o pilar “natureza da pesquisa”, o qual está diretamente ligado a QP1 (“Qual o papel da OvSM e quais são suas principais aplicações?”). Neste sentido serão apresentados os principais aspectos relacionados aos problemas e setores produtivos que estão sendo trabalhos na literatura, o setor da economia em que estes se inserem,

e sobre as especificidades do problema, como os objetivos e a sua dimensão. A Tabela 3.1 apresenta o tipo de problema, setor produtivo e econômico.

Tabela 3.1 - Natureza da pesquisa

Tipo de problema	Real	Teórico	Total	Cum. %
Controle de processos	16 (10,4%)	10 (07,2%)	26 (17,2%)	17,2
Controle de estoque	01 (00,6%)	20 (13,1%)	21 (13,7%)	31,2
Design de produto	08 (05,2%)	08 (05,2%)	16 (10,5%)	41,6
Alocação de recursos	11 (07,2%)	05 (03,3%)	16 (10,5%)	52,1
Modelos matemáticos	00 (00,0%)	11 (07,2%)	11 (07,2%)	59,3
Controle de tráfego urbano	08 (05,2%)	01 (00,6%)	09 (5,88%)	65,1
Setor produtivo				
Não especificado	03 (02,3%)	36 (27,9%)	39 (30,2%)	30,2
Gestão de recursos hídricos	19 (14,7%)	10 (07,7%)	29 (22,5%)	52,7
Tráfego urbano	18 (13,9%)	02 (01,5%)	20 (15,5%)	68,2
Geração/distribuição de energia	06 (04,6%)	01 (00,8%)	07 (05,4%)	73,6
<i>Health care</i>	06 (04,6%)	00 (00,0%)	06 (03,9%)	78,3
Indústria química/farmacêutica	03 (02,3%)	02 (1,5%)	05 (03,9%)	82,2
Setor econômico				
Não especificado	03 (02,3%)	36 (27,9%)	39 (30,2%)	30,2
Primário	19 (14,7%)	10 (07,7%)	29 (22,4%)	52,7
Secundário	15 (11,6%)	07 (05,4%)	22 (17,0%)	69,8
Terciário	33 (25,5%)	06 (04,6%)	39 (30,2%)	100,0

Os objetos de estudo dos artigos foram classificados e agrupados em mais de 18 classes de problemas. Devido a ampla variedade de problemas, apenas os seis mais frequentes estão apresentados na Tabela 3.1, sendo eles: controle de processos, controle de estoque, *design* de produtos, alocação de recursos, modelos matemáticos, e controle de tráfego, representando juntos mais de 65% dos problemas encontrados. A questão mais estudada foi o uso das técnicas de OvSM em controle de processos, que diz respeito a otimização dos parâmetros da produção de determinado bem. A maioria dos trabalhos (74,1%) abordaram este tipo de problema na gestão de recursos hídricos, visando encontrar as taxas de bombeamento de poços de extração e injeção d'água que suprem a demanda ao passo que reduzam os custos com energia, desgaste e contaminação das bacias, *e.g.*, (ATAIE-ASHTIANI; KETABCHI; RAJABI, 2013; CANDELIERI; PEREGO; ARCHETTI, 2018; HUSSAIN *et al.*, 2015; ROY; DATTA, 2019a; TIMANI; PERALTA, 2017). Outras aplicações são para determinação dos parâmetros ótimos do processo de injeção plástica (DANG, 2014; VILLARREAL-MARROQUÍN *et al.*, 2011, 2013), parâmetros de corte em fios semicondutores (MONOSTORI; VIHAROS, 2001), e conformação de chapas metálicas (WANG; CHEN; LI, 2018).

A segunda classe de problemas é relacionada à atividades de controle e gestão de estoques, tais como dimensionamento do número de *kanbans* no chão de fábrica, *e.g.*

(ALABAS; ALTIPARMAK; DENGIZ, 2002; DENGIZ; ALABAS-USLU; DENGIZ, 2009; HURRION, 1997; HUYET; PARIS, 2001), e tamanho de lote (SOUSA JUNIOR *et al.*, 2019; GREENWOOD; REES; SIOCHI, 1998; HACHICHA, 2011; HORNG; LIN, 2017; JACKSON *et al.*, 2019; MOGHADDAM; MAHLOOJI, 2016, 2017; NEZHAD; MAHLOOJI, 2014; PARNIANIFARD *et al.*, 2020b; ZAKERIFAR; BILES; EVANS, 2011), ambas tratadas em cases teóricos, onde a fonte de dados provém de outros artigos ou de problemas clássicos da literatura especializada. Nesta RSL apenas um artigo tratou o problema de controle de estoque em um *case* real, em que foi utilizada a metamodelagem por *Kriging* em OvS discreta para otimização do estoque de segurança de combustível em navios (QUAN *et al.*, 2015).

Muitos autores também estudaram as técnicas de OvSM para otimização do *design* de produtos, com exemplos de aplicações ligadas aos setores automobilístico (FU; SAHIN, 2004; IVANOVA; KUHN, 2014; STORK *et al.*, 2008), energia (SHARIF; HAMMAD, 2019; STORTI *et al.*, 2019), hídrico (BANANMAH *et al.*, 2020; GHORBANI MOOSELU *et al.*, 2019), e infraestrutura de trânsito (YIN *et al.*, 2016). Muitos autores também estudaram OvSM no problema de alocação de recursos (*e.g.*, (COELHO; PINTO, 2018; SONG *et al.*, 2005; YOUSEFI; YOUSEFI, 2019; ZEINALI; MAHOOTCHI; SEPEHRI, 2015), em funções matemáticas sintéticas (*e.g.*, BAQUELA; OLIVERA, 2019; KIM; BOUKOUVALA, 2020; GONZALEZ; JALALI; NIEUWENHUYSE, 2020; WANG; YUAN; NG, 2020), e para controle de tráfego urbano (CHEN, XIAO; OSORIO; SANTOS, 2019; CHONG; OSORIO, 2018; OSORIO; BIERLAIRE, 2013; OSORIO; CHONG, 2015).

Outro fator importante é que tanto nestas classes de problemas quanto na grande parte das outras classes encontradas na RSL, observa-se um balanceamento entre estudos teóricos e cases reais, o que indica o interesse e desenvolvimento da área no meio acadêmico e na prática. Entretanto vale ressaltar que para algumas classes de problemas os estudos se concentram em cases teóricos, como no caso de controle de estoque, sendo recomendável uma maior exploração deste problema em um contexto real, uma vez que a gestão de estoque é considerada uma atividade estratégica na maioria das organizações.

O segundo aspecto estudado é o setor produtivo foco dos trabalhos, em que os setores de gestão de recursos hídricos, geração/distribuição de energia, planejamento de tráfego urbano, saúde, e indústria química/farmacêutica representam juntos 52,0% dos artigos. Isto demonstra que a maioria dos esforços em OvSM estão concentrados na produção de bens e serviços considerados essenciais a sociedade, provendo reduções de custo e melhorias que beneficiam grande parte da população. Cerca de 30,2% dos trabalhos não especificaram o setor de atuação, porém ressalta-se que dentre os trabalhos que apresentaram cases reais apenas 4,3% não

especificaram o setor estudado, o que pode ser tomado como algo positivo, uma vez que a indicação mais precisa da problemática abordada auxilia futuros pesquisadores e práticos na seleção das metodologias que melhor se aplicam no seu setor de atuação. Quando se trata do setor econômico em que o objeto de estudo se insere, observa-se uma certa igualdade entre os setores primários, secundários e terciários, o que caracteriza a amplitude e versatilidade em que as técnicas de OvSM vem sendo trabalhadas. A Tabela 3.2 apresenta as variáveis e objetivos trabalhados pelos autores.

Tabela 3.2 - Variáveis e objetivos dos problemas

Variáveis	Real	Teórico	Total	Cum. %
Nº de variáveis (mediana)	10	5	6	-
Contínua	34 (26,7%)	37 (29,1%)	71 (55,9%)	55,9
Discreta	12 (09,4%)	23 (19,1%)	35 (27,6%)	83,5
Híbrido	07 (05,5%)	07 (05,5%)	14 (11,0%)	94,5
Binária	02 (01,6%)	01 (00,8%)	03 (02,4%)	96,9
Não especificado	00 (00,0%)	03 (02,4%)	03 (02,4%)	99,2
Catagórica	01 (00,8%)	00 (00,0%)	01 (00,8%)	100,0
Objetivos da otimização				
Mono-objetivo	53 (35,6%)	54 (36,2%)	107 (71,8%)	71,8
Multi-objetivo	18 (12,1%)	22 (14,8%)	40 (25,8%)	98,7
Não especificado	01 (00,7%)	01 (00,7%)	02 (01,3%)	100,0

Em respeito a dimensão dos problemas estudados, como apresentado na Tabela 3.2, as aplicações em cases reais apresentam uma mediana de 10 variáveis frente a 5 variáveis nos cases teóricos. Além disto, muitos autores estudaram problemas com mais de 100 variáveis (*e.g.*, (CHEN, XIAO; OSORIO; SANTOS, 2019; CHONG; OSORIO, 2018; JACKSON *et al.*, 2019; OSORIO, 2019b; OSORIO; PUNZO, 2019; TIMANI; PERALTA, 2017), chegando a um máximo de 16.200 variáveis em um problema de calibração de modelos de simulação mesoscópica em controle de trafego (OSORIO, 2019a). Quanto ao tipo de variável, destaca-se que 83,5% dos problemas possuíam variáveis contínuas ou discretas, sendo o primeiro tipo com espaço de solução infinito e o segundo delimitado, porém com uma mediana de 203.963 cenários possíveis, variando entre 36 e 1×10^{62} cenários, e com percentil 90% de $2,5 \times 10^{20}$.

Além disto, embora a maioria dos trabalhos estudassem problemas mono-objetivos, 25,8% dos artigos estudaram metamodelagem para problemas de otimização multiobjetivo, envolvendo principalmente a identificação da Fronteira de Pareto. Com isto observa-se a importância do uso da metamodelagem, uma vez que métodos de OvS recursivos tornam-se praticamente inviáveis em problemas com a dimensionalidade e espaço de solução dos problemas que foram encontrados nesta RSL, além de se mostrar uma boa escolha para

otimização multiobjetivo, a qual é considerada uma classe de problemas NP-hard (BAQUELA; OLIVERA, 2019).

Outro fator analisado nos artigos foi a motivação pelo uso da metamodelagem frente a técnicas *model-based*. Embora em 61,3% dos trabalhos não especificaram a motivação pelo escolha da metamodelagem perante outros métodos, em 33,9% dos artigos a metamodelagem assumiu o papel de reduzir o tempo do projeto de otimização, enquanto demais métodos poderiam não convergirem em um tempo computacional dentro do orçamento de simulação estipulado (*e.g.*, PARNIANIFARD *et al.*, 2018; SONG *et al.*, 2005; VILLARREAL-MARROQUÍN *et al.*, 2013). Além destes, em 3,0% o papel da metamodelagem foi prover respostas rápidas em problemas operacionais que necessitam de tomada de decisões em uma janela de tempo de curto prazo (*e.g.*, DUNKE; NICKEL, 2020; CALAHORRANO *et al.*, 2016). A metamodelagem também foi empregada com o propósito de realizar otimizações conjuntas com dados provindos de diversos modelos (*e.g.*, BARTZ-BEIELSTEIN; ZAEFFERER; PHAM, 2018), gerar rapidamente todo espaço de soluções (*e.g.*, STORK *et al.*, 2008), e promover maior compreensão sobre o processo de otimização (*e.g.*, HUYET; PARIS, 2001).

De acordo com as informações apresentadas nas Tabelas 3,1 e 3,2 e nas discussões apresentadas é possível responder a QP1 (“Qual o papel da metamodelagem em projetos de OvS e quais são suas principais aplicações?”): o uso da metamodelagem assume seu papel por meio da redução do tempo computacional em problemas os quais o espaço de solução e complexidade do modelo tornam os métodos recursivos (mais tradicionais) demasiadamente custosos, inviabilizando seu uso. Além disto, com o advento da indústria 4.0 e o uso da simulação como ferramenta de suporte a decisão em tempo real, ou quase real, os métodos recursivos se tornam inviáveis a depender da janela de tempo de tomada de decisão, sendo necessário recorrer ao uso de metamodelos, que uma vez treinados e validados são capazes de gerar bons resultados em segundos.

Diversos autores utilizaram essa estratégia em sistemas de suporte operacional (*e.g.*, DUNKE; NICKEL, 2020; CALAHORRANO *et al.*, 2016; PEDRIELLI *et al.*, 2020; SHIRAZI; MAHDAVI; MAHDAVI-AMIRI, 2011; STEER; WIRTH; HALGAMUGE, 2012). Quanto as principais aplicações de OvSM em cases reais, destacam-se o problema de controle de processos, alocação e dimensionamento de recursos, controle de tráfego urbano, e design de produtos. Além disto, observa-se uma distribuição justa entre os setores da economia, o que permite inferir que a OvSM é um método amplo e capaz de gerar bons resultados frente a diversos problemas, inclusive aqueles considerados NP-hard.

3.4.2. Métodos adotados

O propósito desta seção é apresentar os métodos mais utilizados na literatura no que tange aos algoritmos de metamodelagem, métodos de busca (otimização), tipo de simulação, estratégias de amostragem, softwares e linguagens de programação mais utilizados. Para isto, as Tabelas 3.3 e 3.4 sumarizam os resultados encontrados, apresentando as 6 tecnologias/métodos mais frequentes de cada tópico estudado.

Percebe-se que em determinados tópicos o número de ocorrências supera o total de artigos analisados, isto ocorre pelo fato de alguns trabalhos utilizarem mais de um método, com o intuito de compará-los ou combiná-los. Como exemplo pode-se citar os trabalhos de Li *et al.* (2010), que comparou metamodelos por RNA, RBF, SVM, *Kriging*, e MARS em um problema de alocação de recursos, Bartz-Beielstein *et al.* (2018), que combinou metamodelos por regressão linear, RF, e *Kriging* por um modelo de *stacking* (ensemble), entre outros (*e.g.* LAL; DATTA, 2020; PARNIANIFARD *et al.*, 2020a; RANJBAR; MAHJOURI; CHERUBINI, 2020).

Tabela 3.3 - Método adotado

Tipo de Simulação	Real	Teórico	Total	Cum. %
SED	14 (09,7%)	24 (16,6%)	38 (26,3%)	26,4
Não especificado	12 (08,3%)	24 (16,6%)	36 (25,0%)	51,4
Método de elementos finitos	13 (09,0%)	13 (09,0%)	26 (18,0%)	69,4
Simulação de Tráfego	16 (11,1%)	1 (00,7%)	17 (11,8%)	81,3
Monte Carlo	01 (00,7%)	5 (03,5%)	06 (04,2%)	85,4
Método de volumes finitos	04 (02,8%)	00 (00,0%)	04 (02,8%)	88,2
Método de otimização				
Algoritmo genético	12 (13,1%)	10 (10,9%)	22 (24,1%)	24,2
<i>Trust Region</i>	11 (12,0%)	01 (01,1%)	12 (13,1%)	37,4
NSGAI	08 (08,8%)	02 (02,2%)	10 (10,9%)	48,4
<i>Particle swarm</i>	06 (06,6%)	03 (03,3%)	09 (09,9%)	58,2
<i>Simulated annealing</i>	03 (03,3%)	02 (02,2%)	05 (05,5%)	63,7
Algoritmo de metamodelagem				
<i>Kriging</i>	22 (09,0%)	31 (12,7%)	53 (21,7%)	21,7
Regressão polinomial	20 (08,2%)	19 (07,8%)	39 (16,0%)	37,7
RNA	26 (10,7%)	13 (05,3%)	39 (16,0%)	53,7
<i>Ensembles</i>	06 (02,5%)	15 (05,1%)	21 (08,6%)	62,3
RBF	06 (02,5%)	09 (03,7%)	15 (06,1%)	68,4
SVM	02 (00,8%)	08 (03,3%)	10 (04,1%)	71,5
Estratégia de amostragem				
<i>Latin hypercube design</i>	23 (15,2%)	38 (25,1%)	61 (40,3%)	40,4
Aleatório	10 (06,6%)	10 (06,6%)	20 (13,2%)	53,6
Não especificado	10 (06,6%)	03 (02,0%)	13 (08,6%)	62,3
<i>Factorial design</i>	04 (02,6%)	06 (04,0%)	10 (06,6%)	68,9
<i>Central composite design</i>	03 (02,0%)	03 (02,0%)	06 (04,0%)	72,8
<i>Orthogonal array</i>	01 (00,7%)	04 (02,6%)	05 (03,3%)	76,2

O primeiro aspecto apresentado na Tabela 3.3 foi o tipo de simulação utilizado pelos pesquisadores. Em 25% dos trabalhos não foi possível identificar tipo de simulação empregado, o que surge como uma recomendação que futuros trabalhos deixem claro esta informação, para que pesquisadores consigam identificar na literatura qual tipo de simulação é aplicável aos seus problemas e quais outras ferramentas podem ser associadas a este. Dentre os artigos, 26,3% dos trabalhos utilizaram SED, que de acordo com Law (2013) pode ser definida como a técnica de modelagem e simulação de um sistema que evolui ao longo de uma linha temporal, em que os eventos mudam de estado em pontos discretos no tempo.

Diversos autores utilizaram SED em problemas relacionados apenas ao setor secundário e terciário da economia, tais como em controle de estoque (*e.g.*, DENGIZ; ALABAS-USLU; DENGIZ, 2009; QUAN *et al.*, 2015), planejamento de chão de fábrica (*e.g.*, DUNKE; NICKEL, 2020; GREINACHER *et al.*, 2020), alocação e dimensionamento de recursos (AYDIN *et al.*, 2018; COELHO; PINTO, 2018; CUCKLER; CHANG; HSIEH, 2017), e programação da produção (CHANG; 2016; HASSANNAYEBI *et al.*, 2019). Não foi encontrado nenhuma aplicação da metamodelagem em SED no setor primário da economia, podendo ser relacionado ao fato de este tipo de simulação ser comumente ministrado em cursos específicos as engenharias, o que limitaria a difusão desta ferramenta. Logo uma possível lacuna a ser explorada por futuros pesquisadores é o uso da metamodelagem e SED para otimizar processos voltados ao setor primário.

Em contrapartida, o segundo tipo de simulação mais utilizado, simulação numérica pelo método de elementos finitos (18,0%), possui a maior parte de suas aplicações voltadas para o setor primário (50,0%) relacionados a otimização de taxas de bombeamento na captação d'água (*e.g.*, DHAR; DATTA, 2009; HUSSAIN *et al.*, 2015; ROY; DATTA, 2018a, 2019b). *Traffic simulation* representa 11,8% das ocorrências e corresponde a modelos microscópicos (GUO; ZHANG, 2014; OSORIO; BIERLAIRE, 2013; OSORIO; NANDURI, 2015), mesoscópicos (CHEN; OSORIO; SANTOS, 2019; CHEN *et al.*, 2016; OSORIO, 2019a), e macroscópicos (CHEN *et al.*, 2014; ZHANG; OSORIO; FLÖTTERÖD, 2017). Neste tipo de simulação, a metamodelagem entra como um meio para minimização do erro do modelo de trânsito por meio da otimização dos parâmetros de entrada do modelo (*e.g.*, matriz origem-destino), minimização do tempo médio das viagens pela otimização dos tempos de ciclo de sinais de trânsito, definição de tarifas de transporte público e/ou pedágio, entre outros.

Diversos outros tipos de simulação também foram identificados, porém menos populares que os mencionados anteriormente. Dentre eles pode-se citar a simulação de Monte Carlo (*e.g.*, SONG *et al.*, 2005; WADE, 2019), método de volumes finitos (*e.g.*, MOOSELU *et*

et al., 2019; STORTI *et al.*, 2019), método de elementos diferenciais (*e.g.*, ALIZADEH; NIKOO; RAKHSHANDEHROO, 2017; LI *et al.*, 2020), simulação baseada em agentes (*e.g.*, YOUSEFI; YOUSEFI, 2019; ZEINALI; MAHOOTCHI; SEPEHRI, 2015), entre outros.

O segundo item apresentado na Tabela 3.3 foi o método de otimização utilizado pelos pesquisadores. Vale ressaltar que o metamodelo não é um método de otimização, e sim um modelo simplificado da simulação original, a qual o pesquisador precisa aplicar um método de busca apropriado para performar a otimização. Nos trabalhos analisados foram encontrados uma variedade de algoritmos de otimização, sendo mais frequente o uso de algoritmo genético, principalmente em problemas mono objetivos de planejamento de redes (SHARIATINASAB *et al.*, 2008; VAHIDI; TAVAKOLI; HOSSEINIAN, 2008; YE; YOU, 2015), alocação de recursos (SOUSA JUNIOR *et al.*, 2020; YOUSEFI *et al.*, 2018), controle de processos (ALI *et al.*, 2018; LUO; JI; LU, 2019; OUYANG *et al.*, 2017), e *design* de produtos (ALBANESI *et al.*, 2018; STORTI *et al.*, 2019).

O segundo método de busca mais utilizado é o *trust region*, na sua maioria associado a metamodelos polinomiais quadráticos no problema de controle de tráfego urbano (CHEN; OSORIO; SANTOS, 2019; CHONG; OSORIO, 2018; OSORIO; BIERLAIRE, 2013; OSORIO; CHONG, 2015; OSORIO; NANDURI, 2015), calibração de modelos de tráfego (LI, WEIZI; WOLINSKI; LIN, 2017; OSORIO, 2019b), evacuação de emergência (LV *et al.*, 2015), e planejamento de redes (YE; YOU, 2015). O terceiro método de busca mais aplicado foi o *Non-dominated Sorting Genetic Algorithm II* (NSGA-II), um algoritmo evolutivo destinado a otimização multiobjetivo, sendo o método de maior ocorrência para este tipo de otimização, em destaque para problemas de controle de processos (ALIZADEH; NIKOO; RAKHSHANDEHROO, 2017; DHAR; DATTA, 2009; HUSSAIN *et al.*, 2015; RANJBAR; MAHJOURI; CHERUBINI, 2020; SREEKANTH; DATTA, 2011), e *design* de produtos (GHORBANI MOOSELU *et al.*, 2019; SHARIF; HAMMAD, 2019).

Outros métodos que também foram comumente utilizados na literatura foram o *particle swarm optimization* (*e.g.*, LI *et al.*, 2020; YADAV *et al.*, 2016), e *simulated annealing* (*e.g.*, CHRISTELIS; MANTOGLU, 2016; CIUFFO; PUNZO, 2014). Estratificando pelos principais tipos de simulação, o uso de algoritmos genéticos foi o mais empregado em SED, método de elementos finitos e método de volumes finitos, já na simulação de tráfego o método de busca mais utilizado foi o *trust-region*, e na simulação de Monte Carlo foi empregado os métodos de *Gold Region* e *Ranking* e Seleção

Logo, em respeito a QP2 (“Quais as principais técnicas de simulação e otimização utilizadas em conjunto com metamodelagem?”), pode-se concluir que a SED é a técnica de

simulação mais popular quando considerado o total de publicações, e também a mais aplicada no setor secundário da economia, devendo-se ao fato de que grande parte dos processos industriais possuem variáveis independentes e identicamente distribuídas e podem ser modelados como sistemas discretos. No setor primário, o uso do método de elementos finitos foi o mais comum, impulsionado pelos estudos sobre controle de processos na extração de água, e no setor terciário, o uso de simulação de tráfego (em especial simulação microscópica) no fornecimento de serviços de planejamento e gestão de tráfego urbano foi a técnica de simulação mais frequente. Além disto, os pesquisadores utilizaram, mais frequentemente, o algoritmo genético como mecanismo de busca no processo de otimização mono-objetivo, e o algoritmo NSGA II para otimização multiobjetivo.

O terceiro item analisado é o algoritmo de metamodelagem utilizado, e assim como disposto na Tabela 3.3, o algoritmo mais frequente é o *Kriging* em 53 artigos (21,7% do total). *Kriging* foi proposto inicialmente por Daniel G. Krige (1951) para aplicações em geoestatística e é um método de interpolação altamente flexível devido a capacidade empregar uma ampla variedade de funções de correlação (PARNIANIFARD *et al.*, 2018). Devido a essa flexibilidade, o *Kriging* tem sido aplicado em uma ampla variedade de problemas, como em controle de estoque (AZIZI; SEIFI; MOGHADAM, 2019; QUAN *et al.*, 2015; SMEW; YOUNG; GERAGHTY, 2013), *design* de produtos (BARTZ-BEIELSTEIN; ZAEFFERER; PHAM, 2018; IVANOVA; KUHNT, 2014; YAOHUI, 2017), controle de processos (BOUKOUVALA; IERAPETRITOU, 2013; CANDELIERI; PEREGO; ARCHETTI, 2018; ROY; DATTA, 2018b), entre outros.

Outra classe de algoritmos clássica da literatura é a RP, representando 16% das ocorrências. RP é uma expressão matemática explícita, linear ou não, que visa aproximar a relação entre as variáveis independentes e dependentes (PARNIANIFARD *et al.*, 2019). A RP é comumente encontrada na literatura na forma de RSM (AYDIN *et al.*, 2018; CUCKLER; CHANG; HSIEH, 2017; HASSANNAYEBI *et al.*, 2019), polinômios quadráticos (XIAO; LEE; CHEN, 2015; ZLATINOV; LASKOWSKI, 2015), e de primeira ordem (HONG; SEO; KIM, 2013; VILLARREAL-MARROQUÍN *et al.*, 2013). Muitos autores também utilizaram polinômios para aproximação da função verdadeira na busca local no método de *trust region* (*e.g.*, CHEN; OSORIO; SANTOS, 2019; CHEN *et al.*, 2014; CHONG; OSORIO, 2018; OSORIO; CHONG, 2015; OSORIO; NANDURI, 2015).

As RNAs foram o terceiro algoritmo mais utilizado pelos pesquisadores. Sendo considerada uma das mais importantes técnicas de inteligência artificial por sua capacidade de resolver problemas com extrema dificuldade e sem soluções analíticas conhecidas. As RNAs

são compostas por elementos interconectados e adaptativos, que visam responder a estímulos de uma maneira similar ao sistema nervoso humano (HACHICHA, 2011). Um fator crucial de sucesso nas aplicações de RNAs é a escolha adequada da sua arquitetura, todavia não é possível determinar uma fórmula padrão, sendo necessário analisar cada caso.

O uso de técnicas de seleção de hiper-parâmetros em metamodelos é uma prática indispensável neste tipo de problema, podendo ser usado desde a tentativa e erro (ATAIE-ASHTIANI; KETABCHI; RAJABI, 2013; SREEKANTH; DATTA, 2011), até métodos de otimização mais eficientes, como os algoritmos genéticos para a seleção dos hiper-parâmetros (HÜLLEN *et al.*, 2020; KIM; JEON; KIM, 2016).

Diversos autores (4^a. posição) também utilizaram-se dos métodos de *ensembles*, algoritmos que combinam diversos metamodelos, semelhantes ou não, para formular um metamodelo com maior capacidade de aprendizagem, tais como *bagging* (CHEN, MINGJIE; IZADY; ABDALLA, 2017; YOUSEFI *et al.*, 2018), *stacking* (BARTZ-BEIELSTEIN; ZAEFFERER; PHAM, 2018), *AdaBoost* (YOUSEFI *et al.*, 2018; YOUSEFI; YOUSEFI, 2019), entre outros.

Os algoritmos que utilizaram *ensembles* baseados em árvores de decisão foram os mais frequentes, sendo comum o uso de RF (*e.g.*, BARTZ-BEIELSTEIN; ZAEFFERER; PHAM, 2018; CANDELIERI; PEREGO; ARCHETTI, 2018; SOUSA JUNIOR *et al.*, 2019; SOUSA JUNIOR *et al.*, 2020; STEER; WIRTH; HALGAMUGE, 2012), GBT (*e.g.*, (SOUSA JUNIOR *et al.*, 2019; SOUSA JUNIOR *et al.*, 2020)). Muitas aplicações também se referem ao uso dos algoritmos RBF (*e.g.*, ALI *et al.*, 2018; CHRISTELIS; MANTOGLOU, 2016; HORNG; LIN, 2018; KRITYAKIERNE; AKHTAR; SHOEMAKER, 2016; STORK *et al.*, 2008), e SVM (*e.g.*, HOU *et al.*, 2019; LAL; DATTA, 2020; LI; TAN; NG, 2019; MIRFENDERESGI; MOUSAVI, 2016).

Com base nos dados apresentados anteriormente é possível responder a QP3 (“Quais os algoritmos de metamodelagem mais utilizados em OvSM?”), em que os principais algoritmos utilizados são o *Kriging*, regressão polinomial (em especial RSM), RNA, métodos *ensembles* (em especial RF e GBT), RBF, e SVM. Estratificando por tipo de simulação tem-se que em SED os algoritmos mais aplicados são regressão polinomial, *Kriging*, RNAs, e árvores de decisão; para o método de elementos finitos destaca-se *Kriging*, RNAs, regressão polinomial, MARS e RBF; e em simulação de tráfego 80% utilizaram regressão polinomial ou *Kriging*; já para nos trabalhos envolvendo simulação de Monte Carlo e método de volumes finitos, o uso de RNAs foi o mais popular. Os principais hiper-parâmetros dos algoritmos discutidos anteriormente e seus valores mais frequentes estão apresentados na Tabela 3.4.

Tabela 3.4 - Principais hiper-parâmetros dos algoritmos de metamodelagem

Algoritmo	Amplitude	Mais frequente
Kriging		
<i>Função de correlação</i>	{ <i>Gaussian</i> , polinomial, exponencial}	<i>Gaussian</i>
<i>Lengthscale (θ)</i>	[10^{-10} ; 10^{10}]	10
<i>Epsilon</i> ¹	[10^{-6} ; 10^{-5}]	-
Tolerância geométrica ¹	[0; 0,3]	-
Nº max. de vetores ¹	[38; 47]	-
RP		
<i>Degree</i>	{1º, 2º, 3º, 4º, 5º}	2º
RNA		
<i>Transfer function</i>	{ <i>linear</i> , <i>sigmoid</i> , <i>quadratic error</i> , <i>hyperbolic tangent</i> , <i>rectified linear unit</i> , RBF}	Linear
<i>Training function</i>	{ <i>backpropagation</i> , <i>levenberg-marquardt</i> , <i>feedforward</i> }	<i>Backpropagation</i>
<i>Learning rate</i>	[0,01; 0,4]	0,1
<i>Epochs</i>	[200; 1500]	1500
Nº de camadas	[1; 4]	1
Nº de neurônios	[1; 66]	4
<i>Momentum</i>	0,1	0,1
RF		
Nº de árvores	[10; 50]	10
<i>Maximum Depth</i>	2	2
Nº de variáveis por nó ³	$\text{int}(p/3) - 1$, $\text{int}(p/3)$, $\text{int}(p/3) + 1$	-
GBT		
Nº de árvores	[1; 100]	100
<i>Learning rate</i> ²	[0,05; 0,30]	-
Min. de Obs. Por folha ²	[0,12; 0,50]	-
RBF		
<i>Função</i>	{ <i>gaussian</i> , <i>splines</i> , <i>multiquadrics</i> , <i>cubic</i> }	<i>Gaussian</i>
<i>Scale (sigma)</i>	[1; 20]	1
<i>Regularization</i>	[10^{-10} ; 10^{10}]	-
SVM		
C^4	[$ \bar{y} - 3\sigma $; $ \bar{y} + 3\sigma $]	20,73
<i>Epsilon</i> ⁴	$\left[0; 3\sigma\sqrt{\ln n/n}\right]$	0,0779
<i>Kernel</i>	{RBF, <i>Gaussian</i> }	<i>Gaussian</i>
Gama	[0,0224; 0,1294]	0,0673

¹Hiper-parâmetro citado, mas não quantificado na RSL, logo adotou-se o sugerido por Rostami, Khaksar Manshad (2014).

²Hiper-parâmetro citado, mas não quantificado na RSL, portanto adotou-se o sugerido por Ganjisaffar, Caruana e Lopes (2011).

³Hiper-parametro citado, mas não quantificado nos artigos da RSL, logo adotou-se como o sugerido por Dai *et al.* (2018), em que p é o número de variáveis de decisão do problema.

⁴Os valores para C e ϵ devem ser calculados a partir de uma execução preliminar do algoritmo com parâmetros default., sendo qual \bar{y} e σ_y são a média e o desvio padrão dos valores de y nos dados de treino, n é o tamanho da amostra e σ a variância da diferença entre o valor previsto e observado, para mais detalhes veja Oliveira *et al.* (2010).

O último item apresentado na Tabela 3.3, e relacionado a QP4 (“Quais as estratégias de amostragem utilizadas?”), foi a estratégia de amostragem utilizada pelos pesquisadores para a construção do metamodelo. Um dos fatores críticos de sucesso no uso da metamodelagem é a base de dados utilizada no treinamento, por isto é fundamental a escolha correta do método de amostragem que seja capaz de reduzir o número de simulações necessárias para sua modelagem, ao mesmo tempo em que tenha uma boa representatividade do espaço de busca (VU *et al.*, 2017). Nos artigos analisados, 61 (40,3%) utilizaram a técnica de DOE conhecida por LHD, seguido pela amostragem aleatória (13,2%), *Fatorial Design* (6,6%), CCD (4,0%), *Orthogonal Array* (3,1%), entre outros, além de 13 (8,6%), que não utilizaram ou não especificaram a técnica de amostragem aplicada.

Proposto inicialmente por McKay *et al.* (1979), o LHD é um método de amostragem estratificada muito popular em estudos de metamodelagem. Dado n amostras e s variáveis, o LHD divide a região de cada variável em n intervalos disjuntos com igual probabilidade e, em seguida, o espaço amostral é selecionado pela construção da matriz $n \times s$, em que as colunas são selecionadas aleatoriamente da permutação de $[1, \dots, n]$ e cada linha corresponde a uma célula de um hiper-retângulo, e após a construção da matriz, um ponto de cada uma das células é amostrado (LUO; JI; LU, 2019).

Além da boa capacidade de gerar amostras representativas do espaço de solução, uma das características do LHD que pode decorrer o seu sucesso para construção de metamodelos é que, diferentemente de técnicas como *orthogonal array* ou CCD, o tamanho da amostra é definido pelo usuário, o qual possui maior poder de decisão perante o *trade-off* entre o tempo para execução dos experimentos e a precisão dos metamodelos. Embora o tamanho da amostra seja definido pelo usuário, existe um consenso na academia que este deve ser, pelo menos, 10 vezes o número de variáveis do problema (KIM; BOUKOUVALA, 2020; PARNIANIFARD *et al.*, 2018; PARNIANIFARD; AZFANIZAM, 2020; SMEW; YOUNG; GERAGHTY, 2013). Ressalta-se que dentre os principais tipos de simulação, o LHD foi o mais utilizado, exceto para a simulação de tráfego que empregou mais frequentemente a amostragem aleatória.

Além disto, em 80 artigos (61,5% dos trabalhos que utilizaram alguma técnica de amostragem) foi utilizada a estratégia de amostragem fixa, ou seja, é retirado apenas um conjunto de amostras de tamanho w no início do projeto e o metamodelo é treinado com, e apenas, este conjunto. Já na estratégia incremental (38,5%) é retirada uma amostra inicial de dados de tamanho ξ , sendo o metamodelo treinado e otimizado, com os resultados de precisão e direção da região em que se encontra o ótimo, incrementando-se a base de dados com um

novo conjunto de amostras de tamanho δ_i . Logo o metamodelo treinado novamente com uma base de tamanho $\xi_{i+1} = \xi_i + \delta_i$, sendo $i \in \mathbb{Z}$ o número de interações do algoritmo.

Na maioria dos trabalhos, esse algoritmo responsável pela aprendizagem recursiva e adaptativa é conhecido por *Efficient Global Optimization* (EGO), e a escolha de qual (quais) ponto(s) deve(m) ser acrescentado(s) a cada interação é geralmente determinado pelo algoritmo *Expected Improvement* (EI) (COELHO; PINTO, 2018; PEDRIELLI *et al.*, 2020; QUAN *et al.*, 2015; ROJAS GONZALEZ; JALALI; VAN NIEUWENHUYSE, 2020; WANG, HU; CHEN; LI, 2018). A Tabela 3.5 apresenta os principais *softwares* e métricas de performance encontradas na literatura.

Tabela 3.5 - Softwares e métricas

Softwares de simulação	Real	Teórico	Total	Cum. %
Não especificado	05 (03,4%)	24 (16,3%)	29 (19,7%)	19,7
Linguagem de programação	07 (04,7%)	15 (10,2%)	22 (15,0%)	34,7
Arena	04 (02,7%)	08 (05,4%)	12 (08,1%)	42,9
FEMWATER	04 (02,7%)	04 (02,7%)	08 (05,4%)	48,3
Aimsun	07 (04,7%)	00 (00,0%)	07 (04,7%)	53,1
MoldFlow	02 (01,3%)	02 (01,4%)	04 (02,7%)	55,8
Software de modelagem				
Matlab	26 (18,5%)	26 (18,5%)	52 (37,1%)	37,1
Não especificado	24 (17,1%)	24 (17,1%)	48 (34,2%)	71,4
Python	03 (02,1%)	03 (02,1%)	06 (04,2%)	75,7
R	03 (02,1%)	03 (02,1%)	06 (04,2%)	80,0
GAMS	03 (02,1%)	03 (02,1%)	06 (04,2%)	84,3
Minitab	02 (01,4%)	02 (01,4%)	04 (02,8%)	87,1
Métricas de performance				
<i>Mean square error</i> (MSE)/ <i>Root MSE</i>	32 (12,6%)	36 (14,1%)	68 (26,8%)	26,8
Coefficientes de correlação (R, R ² , R _{adj})	24 (09,4%)	26 (10,2%)	50 (19,7%)	46,5
Não especificado	24 (09,4%)	13 (05,1%)	37 (14,6%)	61,0
<i>Mean absolute percentage error</i> (MAPE)	14 (05,5%)	12 (04,7%)	26 (10,2%)	71,3
<i>Mean absolute error</i> (MAE)	05 (01,9%)	07 (02,7%)	12 (04,7%)	76,0
<i>Standardized residuals error</i> (SRE)	02 (00,7%)	05 (02,0%)	07 (02,7%)	78,7
Técnica de validação				
Não especificado	28 (22,6%)	20 (16,1%)	48 (38,7%)	38,7
<i>Train/test split</i>	17 (13,7%)	11 (08,9%)	28 (22,6%)	61,3
<i>Leave one out cross-validation</i>	09 (07,3%)	11 (08,9%)	20 (16,1%)	77,4
<i>k-fold cross-validation</i>	06 (04,8%)	11 (08,9%)	17 (13,7%)	91,1
Não usa validação	00 (00,0%)	04 (03,2%)	04 (03,2%)	94,4
Novos dados para teste	02 (01,6%)	02 (01,6%)	04 (03,2%)	97,6

Quanto a tecnologia empregada nos trabalhos, pode-se observar no primeiro tópico da Tabela 3.5 que a maioria dos autores (19,7%) não especificaram o *software* de modelagem e simulação empregados, 14,9% utilizaram linguagens de programação como C++ (*e.g.*, SONG *et al.*, 2011), *Python* (*e.g.*, (SOUSA JUNIOR *et al.*, 2019; KIM; BOUKOUVALA, 2020), e

Matlab (e.g., PARNIANIFARD; AZFANIZAM, 2020; GONZALEZ; JALALI; NIEUWENHUYSE, 2020). Em respeito a softwares proprietários destinados a simulação destaca-se o uso do *Arena*[®], um dos primeiros softwares desenvolvidos com animação 2D para modelagem de sistemas discretos. *FEMWATER* e *MoldFlow* são os mais comuns para o método de elementos finitos, e *Aimsun* para simulação de tráfego.

No que se refere ao software ou linguagem de programação utilizada no desenvolvimento dos metamodelos assim como na estrutura de otimização, o ambiente *Matlab* foi o mais expressivo, sendo utilizado por 37,1% dos pesquisadores. Contudo, embora menos expressivo, *Python* e R, linguagens *open sources* muito usadas no campo de *Data Science*, ganharam visibilidade nos últimos anos, ao passo que todos 6 trabalhos (4,3%) que utilizaram R foram publicados a partir de 2014 e os que usaram *Python* foram publicados ao longo do último biênio (2019-2020). Portanto é possível constatar a popularidade e aplicabilidade do *Matlab* em projetos de OvSM, porém o uso recente e bem sucedido do R e *Python* os coloca como boas alternativas, principalmente pelo crescimento de suas comunidades voltadas a estudos de ML (PEDREGOSA *et al.*, 2011), com a criação e aperfeiçoamento de novos algoritmos que podem vir a incrementar a eficiência e ajuste dos metamodelos.

Para responder a QP5 (“Como os resultados são mensurados/validados?”), o último item analisado nesta seção é a métrica de performance utilizada para avaliação dos metamodelos. As três métricas mais utilizadas pelos autores foram o MSE (ou *root* MSE), coeficientes de determinação (R , R^2 , R^2_{adj}), MAPE e MAE. Mais detalhes sobre estas métricas são apresentados na Seção 3.3.2.

O processo de validação do metamodelo é uma etapa fundamental, pois é com ela que o pesquisador pode avaliar e comparar a eficácia de diversos metamodelos em representar o sistema simulado. Dentre os artigos analisados, 38,3% não especificaram o tipo de validação usado, sendo recomendado a futuras pesquisas uma maior atenção nesta etapa, visto que a validação é parte indispensável do processo metodológico de OvSM e o que fornece suporte e confiabilidade na utilização dos metamodelos. Em 22,9% dos artigos (e.g., SOUSA JUNIOR *et al.*, 2019; HUSSAIN *et al.*, 2015; LUO; JI; LU, 2019; RAEI *et al.*, 2019) foi utilizado a estratégia de divisão da base de dados em duas partes: a primeira para treinamento do metamodelo (em torno de 70%-80% dos dados); a segunda para teste/validação (20%-30% dos dados).

Todavia o uso das técnicas de *cross-validation* estiveram presentes em grande parte dos trabalhos, sendo que 20 (16,4%) dos autores usaram *leave-one-out cross-validation* (e.g., AZIZI; SEIFI; MOGHADAM, 2019; CHEN, XIQUN *et al.*, 2019), e 15 (12,3%) usaram *k-fold*

cross-validation, (e.g., COELHO; PINTO, 2018; HOU *et al.*, 2019; WADE, 2019). Para uma explicação mais detalhada sobre técnicas de *cross-validation* veja (BERGMEIR; BENÍTEZ, 2012).

3.4.3. Estado da arte em OvSM

Esta seção destina-se a responder a QP6 (“Qual é o atual estado da arte da pesquisa sobre MBOS?”), apresentando a origem da pesquisa em OvSM e os principais dados bibliométricos que permitam discutir a direção que o tema está sendo trabalhado nas diversas frentes de estudo. A Tabela 3.6 e as Figuras (3.3-3.7) resumem os principais resultados.

Tabela 3.6 - Número de publicações de acordo com o nome dos autores, nacionalidade e fonte

Posição	Nome – nº de publicações	Posição	Nome – nº de publicações
Pesquisador			
1º	C. Osorio – 9	6º	M. K.A. Ariffin – 5
2º	B. Datta – 8	6º	M. I. S. Ismail – 5
3º	A.S. Azfanizam – 6	8º	X. Chen – 4
3º	S. H. Ng – 6	8º	B. Dengiz – 4
3º	A. Parnianifard – 6	10º	W. Lu – 4
Nacionalidade			
1º	Estados Unidos – 47	5º	Canadá – 8
2º	China – 39	7º	Alemanha – 7
3º	Iran – 19	7º	Singapura – 7
4º	Austrália – 9	7º	Turquia – 7
5º	Brasil – 8	10º	Itália – 6
Periódico			
1º	<i>Journal of Hidrology</i> – 6	6º	<i>Journal of Hydrologic Engineering</i> – 4
1º	<i>Journal of the Operational Research Society</i> – 6	6º	<i>Transportation Research Part B: Methodological</i> – 4
3º	<i>European Journal of Operational Research</i> – 5	6º	<i>Transportation Science</i> – 4
3º	<i>Simulation Modelling Practice and Theory</i> – 5	9º	<i>Applied Soft Computing Journal</i> – 3
3º	<i>Water Resources Management</i> – 5	9º	<i>Communications in Statistics: Simulation and Computation</i> – 3

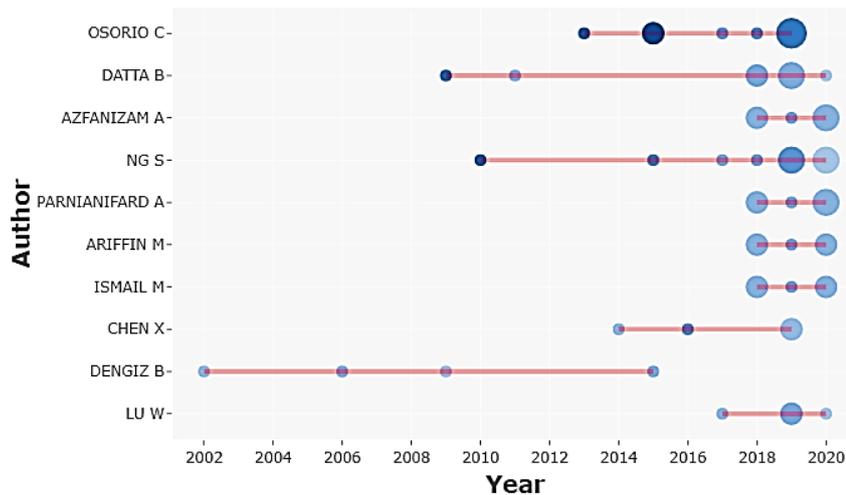


Figura 3.3 - Produção ao longo do tempo dos 10 autores mais produtivos

Fonte - Base de dados da Scopus e WoS

O primeiro item analisado refere-se aos pesquisadores mais influentes na comunidade de OvSM. Dentre os top 10 de autores em OvSM, como mostrado na Tabela 3.6 e Figura 3.3, destaca-se a pesquisadora Carolina Osório, como a autora mais produtiva e citada, com 9 publicações e 209 citações em seus estudos de simulação de tráfego. Bithin Data é o segundo autor mais produtivo, atuando principalmente com modelos de elementos finitos para problemas de gestão hídrica, e Azfanizam, Parnianifard, Ariffin e Ismail possuem uma atuação conjunta em estudos envolvendo otimização robusta. A respeito da filiação dos autores, embora o conhecimento da área não esteja concentrado em um número restrito de localidades, destaca-se a presença da *Jilin Univesity*, *Massachusetts Institute of Technology* e *National Univesity of Singapore* como os principais centros de pesquisa em OvSM. A Figura 3.4 apresenta as 10 instituições mais influentes da área.

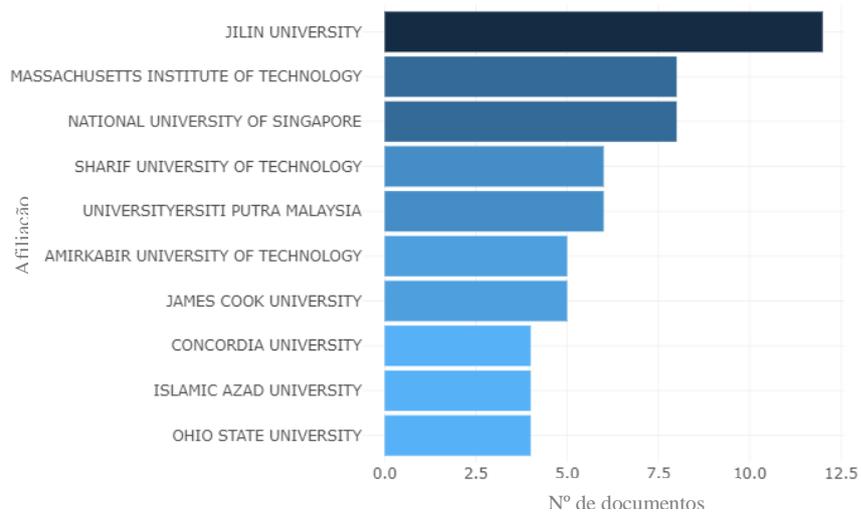


Figura 3.4 - Top 10 centros de pesquisa em OvSM

Fonte - Base de dados da Scopus e WoS

De acordo com a segunda informação da Tabela 3.6, dentre os 32 países que possuíam pelo menos uma publicação em OvSM, os Estados Unidos, China, Irã, Austrália, Brasil, Canadá, Singapura, Turquia e Itália foram países mais produtivos. A Figura 3.5 demonstra que a pesquisa em OvSM, embora presente em 32 países, ocorre de forma isolada em nada nação, apresentando colaboração internacional em apenas 8 trabalhos.

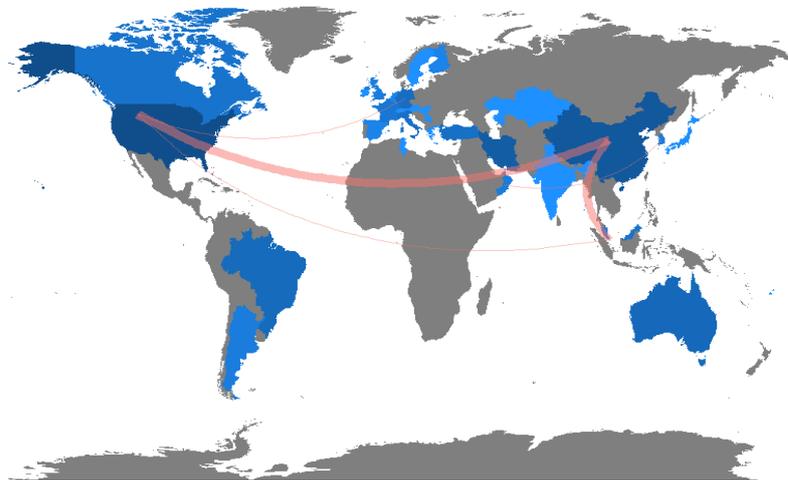


Figura 3.5 - Densidade de publicação e colaboração internacional

Fonte - Base de dados da Scopus e WoS

A última informação contida na Tabela 3.6 é sobre os principais periódicos da área. O *Journal of Hidrology* é o principal periódico da área, com um total de 6 artigos sobre OvSM. Vale ressaltar que os artigos estão pulverizados em 108 periódicos, sendo que o número de publicações no top 10 corresponde apenas a 31,5% do total de artigos, indicando que, se por

um lado não foi identificado a existência de periódicos com uma linha específica para metamodelagem, por outro infere-se a amplitude das técnicas de OvSM, que se encontra presente em periódicos com foco em diferentes áreas e aplicações.

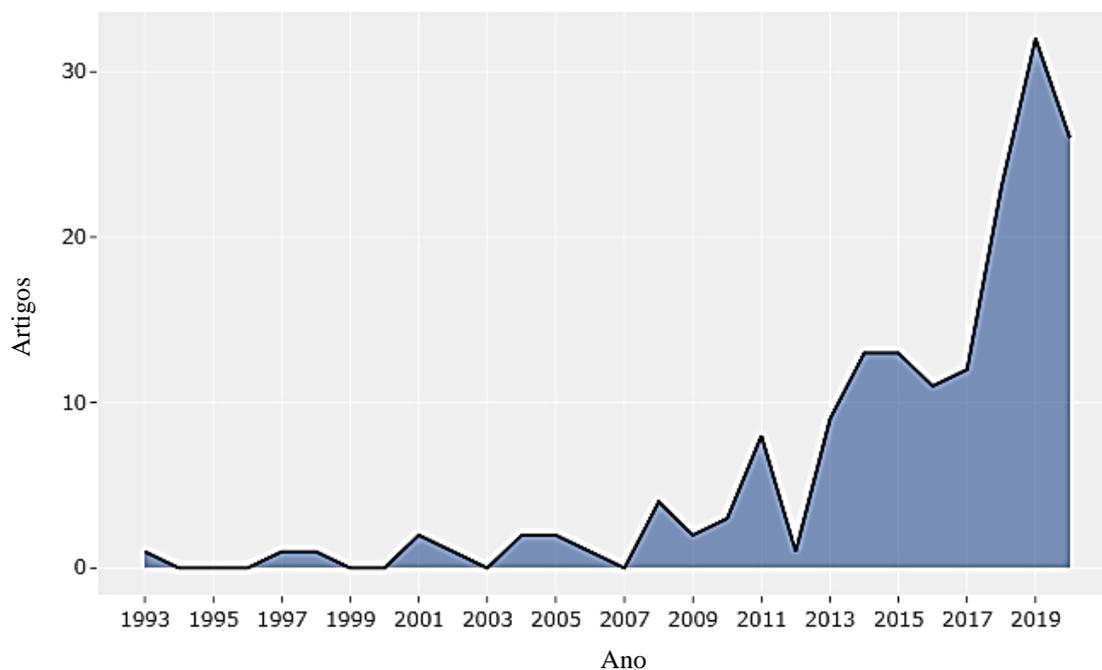


Figura 3.6 - Produção científica anual

Fonte - Base de dados da Scopus e WoS

Quanto a evolução do tema no tempo, como demonstrado na Figura 3.6, observa-se que o interesse sobre o tema ganhou força na última década, devendo-se, à priori, ao desenvolvimento observado no campo da inteligência artificial, poder computacional, e pelo anseio das organizações por ferramentas mais eficientes no processo de tomada de decisões frente a expansão do conceito de indústria e serviços 4.0. Quantitativamente falando, passou-se de uma taxa de crescimento anual de 5,96% entre os anos 2000 e 2010 para 24,10% na última década, o que representa um salto de 20 publicações no período 1993-2010 para 32 artigos apenas em 2019. Estes dados confirmam a forte tendência de crescimento no uso de OvSM tanto em meios práticos quanto acadêmicos, e reforça a necessidade de revisões de literatura, como esta, para compilar os conhecimentos e prover direcionamentos a futuras pesquisas na área.

Por fim, a fim de munir os pesquisadores que desejam iniciar seus estudos na área, ou até mesmo aqueles mais experientes, com os termos científicos mais utilizados e os que estão em ascendência, a Figura 3.7 apresenta as palavras-chaves mais relevantes na pesquisa em OvSM. Esta análise utilizou as 50 principais *Keywords Plus*, pois segundo Zhang *et al.* (2015) esta é mais indicada em análises bibliométricas sobre a estrutura de campo científico.

sistemas de detecção de colisões em navios. O método associava metamodelagem e simulação baseada em agentes para geração de planos de contingência em caso de possível colisão, minimizando os riscos e efeitos negativos da mudança de rota.

Outros *frameworks* também abordaram problemas de engenharia específicos, como nos trabalhos de Dang (2014), que associou metamodelagem, DOE e *Computer Aided Engineering* para otimização de processo de injeção plástica, e Ranjbar, Mahjouri e Cherubini (2020), que desenvolveram um procedimento de otimização multiobjetivo por metamodelagem e NSGA II para gerenciar intrusão de água em aquíferos costeiros.

Parnianifard *et al.* (2020b) propuseram um *framework* para otimização robusta (visa otimizar a FO ao passo que minimiza a variância associada a resposta) por meio de metamodelagem, arranjos ortogonais de Taguchi, Fronteira de Pareto e análise de sensibilidade por *Bootstrapping*.

Barton (2009) propôs um *framework* geral para otimização por metamodelagem em modelos de SED. Em seu trabalho, o processo de OvS foi bifurcado em duas possíveis estratégias, a metamodelagem local ou global, assim como apresentado na Figura 4.8. Na estratégia de metamodelagem global, ao qual é o foco desta dissertação, o metamodelo é ajustado para representar todo o espaço de busca, e esta aproximação global é utilizada no processo iterativo da otimização. Já na estratégia local, as etapas de metamodelagem e otimização se alternam: conforme a otimização se move no espaço de busca, novas regiões locais do espaço θ são exploradas e novas aproximações de metamodelo são ajustadas.

A estratégia de metamodelagem global proposta por Barton (2009) é desenhada em sete etapas principais. Na primeira etapa deve-se identificar a região do espaço de busca por meio da definição das restrições do problema. A segunda etapa é a escolha do tipo de metamodelos a ser utilizado, para esta estratégia o autor sugere o uso de RNA, RBF, *splines* ou *Kriging*. As três etapas seguintes dizem respeito a escolha do tipo de experimentação, a qual pode ser realizada por arranjos fatoriais fracionados ou completos, CCD, LHD, entre outros, a condução dos experimentos no modelo de simulação e o ajuste do metamodelo ao arranjo experimental. Caso as métricas de erro não sejam aceitáveis deve-se voltar a terceira etapa e alterar o tipo de metamodelo ou arranjo experimental. As duas últimas etapas abordam a fase de otimização, a qual deve-se aplicar o algoritmo de otimização e avaliar a resposta ótima encontrada no modelo de simulação.

Ressalta-se que o *framework* para otimização global proposto por Barton (2009), Figura 3.8, foi utilizado como base para o *framework* proposto nesta dissertação. Entretanto, este possui algumas peculiaridades que as diferenciam de Barton (2009), sendo a mais perceptível

a inclusão da etapa de otimização de hiper-parâmetros durante o ajuste do metamodelo, a qual tem o objetivo de minimizar o seu erro. Mais detalhes do *framework* proposto encontram-se no Capítulo 4.

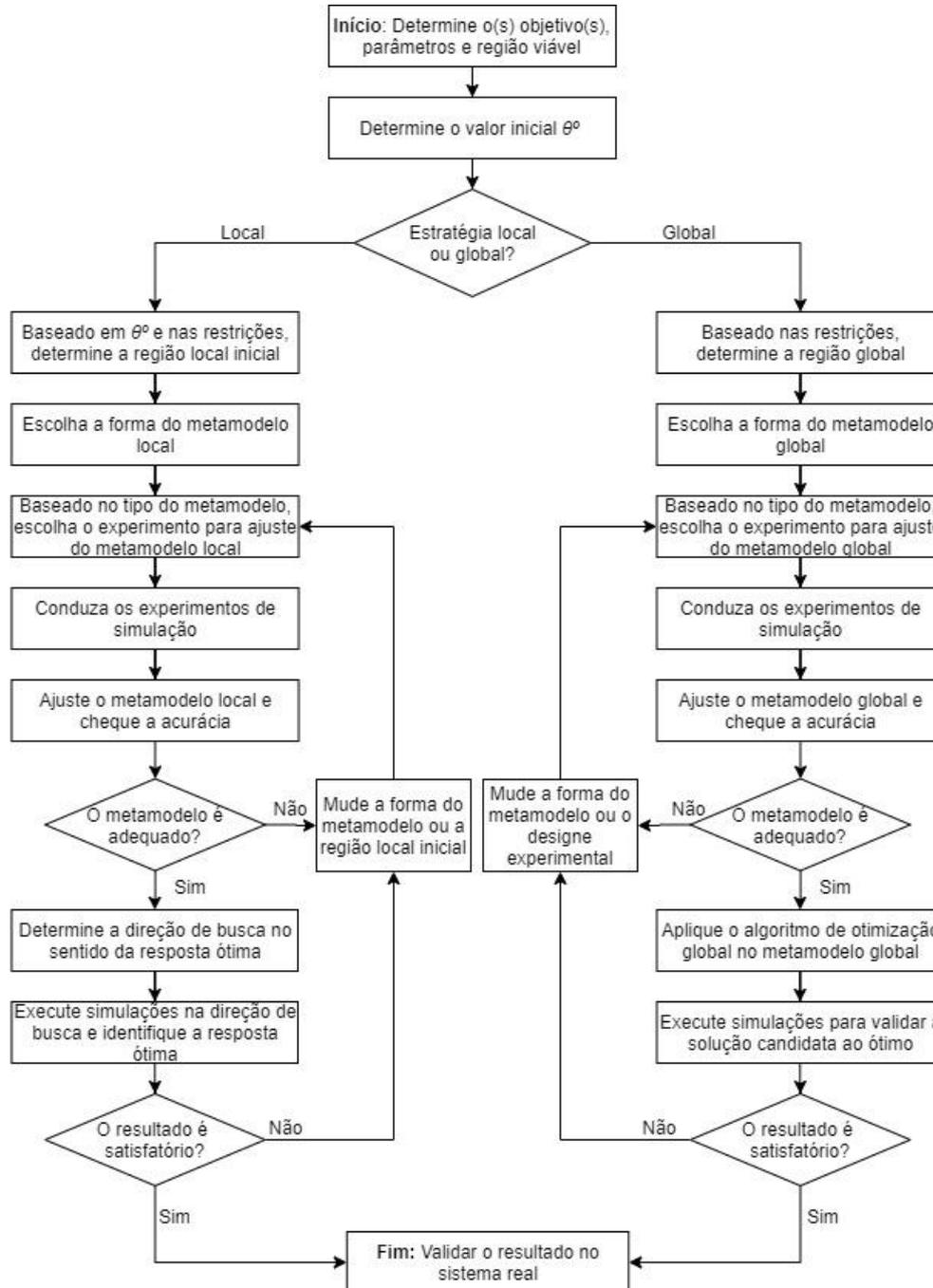


Figura 3.8 - Estratégia de otimização baseada em metamodelo global vs. local, semelhanças e diferenças

Fonte: Adaptado de Barton (2009)

3.4.5. Futuras direções na pesquisa em OvSM

De acordo com os artigos avaliados nesta RSL foi possível identificar as tendências e lacunas presentes na literatura, convertendo-se em oportunidades envolvendo o contexto e os métodos adotados em OvSM. No que tange aos métodos adotados, o desenvolvimento crescente na área de inteligência computacional tem contribuído significativamente com o desenvolvimento e aprimoramento de algoritmos de aprendizagem, que conseqüentemente podem ser explorados na criação de metamodelos mais precisos. Um algoritmo que vem ganhando destaque no campo de ML pela sua flexibilidade e capacidade de aprendizado é o *Deep Learning* (HIGHAM; HIGHAM, 2019; XU *et al.*, 2020), todavia não foi identificado na literatura o seu uso na criação de metamodelos para uso em OvS, tornando-se uma oportunidade promissora a ser explorada em futuras pesquisas.

Além do *Deep Learning*, é possível encontrar oportunidades em algoritmos de ML já trabalhados na literatura, mas que continuam em desenvolvimento. Algoritmos clássicos como o *Kriging* e RNAs apresentaram crescimento considerável nos últimos anos, principalmente no que tange o uso de variações de seus algoritmos, *e.g.*, (ROJAS GONZALEZ; JALALI; VAN NIEUWENHUYSE, 2020; YOUSEFI *et al.*, 2018), ou sua associação com outras técnicas na construção de soluções inovadoras, *e.g.*, (RANJBAR; MAHJOURI; CHERUBINI, 2020; YAOHUI, 2017). Outros algoritmos que ganharam atenção recentemente em OvSM, possuem alta capacidade de aprendizagem e estão em desenvolvimento ativo na área de inteligência artificial são o *extreme learning machines*, *e.g.*, (LI *et al.*, 2020), SVM (LI; TAN; NG, 2019), e métodos *ensemble*, *e.g.*, (LAL; DATTA, 2020).

O uso de técnicas de otimização de hiper-parâmetros permite aumentar a eficácia dos metamodelos em representar corretamente os modelos de simulação. Métodos de otimização como AG, *particle swarm*, *bayesian optimization* e *search grid* para otimização de hiper-parâmetros estão apresentados bons resultados em diversos estudos, *e.g.*, (AMIRI; SHIRAZI; TAJDIN, 2019; HOU *et al.*, 2019; HÜLLEN *et al.*, 2020; KIM; BOUKOUVALA, 2020; LI; TAN; NG, 2019), porem ainda é um tema oportuno na literatura da área.

Outra técnica que vem mostrando bons resultados na construção de metamodelos é o uso de uma estratégia conhecida como método de otimização global, a qual utiliza principalmente das técnicas de EGO e EI para construir metamodelos de forma adaptativa e interativa. Nas aplicações envolvendo este tipo de estratégia, foi predominante o uso do *Kriging* como algoritmo de metamodelagem, *e.g.*, (COELHO; PINTO, 2018; IVANOVA; KUHN, 2014; WANG; YUAN; NG, 2020), sendo recomendado que futuras pesquisas explorem esta estratégia adaptativa na construção de metamodelos com base em outros algoritmos, como RNAs, SVM, *Extreme Learning Machines*, *decision trees based*, entre outros.

Ao passar dos anos, a metamodelagem tem sido usada na tratativa de problemas mais complexos e de grande dimensão. A fim de lidar com base de dados com dimensões cada vez maiores, o uso das técnicas de *feature selection* se tornou uma questão chave para as comunidades de ML e mineração de dados. *Feature selection* pode ser interpretada como um conjunto de técnicas frequentemente usada para redução de dimensões do problema, removendo variáveis irrelevantes ou redundantes do banco de dados (BOMMERT *et al.*, 2020). Todavia, nesta RSL, estas técnicas foram utilizadas apenas nos trabalhos de Kim e Boukouvala (2020), e Sharif e Hammad (2019), surgindo como sugestão para futuros pesquisadores o uso de suas técnicas, tais como *Principal Component Analysis*, *Wrapper*, *Filter*, entre outras, para preparação de dados no processo de OvSM.

Relacionado ao contexto a qual se aplica os estudos, observa-se no setor terciário a maior tendência de crescimento, impulsionado principalmente pelo uso da metamodelagem em problemas logísticos, como transporte e armazenagem de carga, e calibração e otimização de modelos de tráfego urbano. Todavia, vale ressaltar, que não foi encontrado nenhuma aplicação de OvSM para otimização das operações em centros de distribuição, o que pode ser uma potencial aplicação, principalmente no cenário de expansão do *e-commerce* e os desafios da logística reversa. Além disto, o contexto criado pela pandemia da COVID-19 intensificou a tendência pré-existente do uso das técnicas de modelagem e simulação em *healthcare* (CURRIE *et al.*, 2020), o que coloca a metamodelagem como oportunidade na redução da complexidade inerente de modelos *healthcare* e proporcionar uma tomada de decisões mais rápida e consistente.

Por fim, com o advento da indústria & serviço 4.0, estudos e aplicações de Gêmeos Digitais têm sido demandados intensamente por organizações dos mais diversos setores. De acordo com Mueller *et al.* (2017), Gêmeos Digitais são a junção de sistemas físicos e virtuais (simuladores), conectados em autonomia total ou parcial, com controle em tempo real e capaz de cooperar com a interface humana para tomada de decisões. Uma vez que a tomada de decisões é dada em tempo real, ou em uma janela de tempo muitas vezes curta, limita o uso das técnicas de OvS *model-based*, tornando o uso da metamodelagem para substituir modelos de simulações, uma aplicação potencial para otimização via simulação em Gêmeos Digitais. Aplicações de metamodelos para tomada de decisões em tempo quase real, no contexto operacional, podem ser observados nos trabalhos de Dunke e Nickel (2020), Calahorrano *et al.* (2016), Pedrielli *et al.* (2020), Shirazi *et al.* (2011), e Steer *et al.* (2012).

3.5. Considerações Finais da Revisão Sistemática de Literatura

Este Capítulo teve como principais objetivos (a) desenvolver uma RSL sobre OvSM, de modo a (b) identificar e explorar os principais aspectos relacionados ao tema, (c) analisar e sumarizar os resultados a fim de responder as QPs, e (d) delimitar o estado da arte, lacunas e tendências a serem exploradas.

A escolha pelo uso da RSL provou-se uma técnica satisfatória na exploração e análise da literatura científica existente. As diretrizes propostas por Denyer e Tranfield (2009) e Denyer *et al.* (2008) contribuíram significativamente para uma melhor condução desta pesquisa, prevenindo possíveis distorções e parcialidades dos autores sobre os resultados encontrados, atendendo assim ao primeiro objetivo proposto. Vale ressaltar que esta pesquisa não almejou esgotar todos os artigos sobre o tema, porém analisar sistematicamente uma amostra representativa que permitiu responder de forma consistente as QPs, gerando *insights* sobre as práticas em OvSM, de modo a auxiliar futuros pesquisadores e práticos no entendimento e apresentação da fronteira do conhecimento existente na área.

Em atendimento ao segundo e terceiro objetivo, as Seções 3.4.1 e 3.4.2 apresentaram os principais aspectos relacionados aos contextos no qual os estudos de OvSM estão inseridos e os métodos de simulação, otimização, amostragem e metamodelagem adotados. Por fim, a Seção 3.4.3 apresentou os resultados bibliométricos acerca do tema, e a Seção 3.4.4 discutiu as lacunas e tendências identificadas nesta pesquisa, respondendo assim o último objetivo proposto.

Em respeito as QPs, a primeira conclusão deste trabalho é que o papel da metamodelagem está na redução do tempo computacional dispendido em problemas de OvS, para os quais modelos recursivos seriam custosos, ou até inviáveis, de se executar. Com isto, a metamodelagem vem sendo aplicada nos mais diversos problemas, com destaque para o uso da OvSM na otimização de parâmetros de controle de processos, dimensionamento e alocação de recursos, controle de tráfego urbano e *design* de produtos.

Os estudos em OvSM se iniciaram predominantemente na indústria, todavia observa-se uma forte tendência de seu uso nos setores primários, em especial extração e tratamento de recursos hídricos, e terciário, com interesses voltados ao conceito de cidades inteligentes e melhor gestão em *healthcare*.

A segunda conclusão é a respeito dos métodos adotados, para os quais não foi possível chegar a um consenso global sobre qual método ou técnica é a mais indicada para cada tipo de problema, ficando muitas vezes a cargo dos recursos disponíveis e conhecimentos prévio dos pesquisadores. Todavia, quanto as técnicas de simulação utilizadas podem-se perceber a grande aplicabilidade da SED, do método de elementos finitos, e da simulação microscópica em

estudos de tráfego urbano. Quanto aos métodos de otimização mais utilizados, destaca-se o uso de algoritmos genéticos para tratativa de problemas mono-objetivo e do algoritmo NSGA II para problemas multiobjetivos em modelos de SED, elementos finitos e volumes finitos, e do método *trust region* para simulações de tráfego. Além disto, dentre os algoritmos mais utilizados para construção de metamodelos destacam-se o uso do *Kriging*, RP, RNAs, RBF e SVM, com grande aplicabilidade, também, dos métodos de *ensembles* (*Bagging*, *AdaBoost*, entre outros), os quais visam combinar um ou mais algoritmos de modelagem para formular um metamodelo com maior capacidade de aprendizado, *e.g.*, RF e GBT. Não obstante, destaca-se o design LHD e Aleatório como principais estratégias de amostragem do espaço de busca utilizados pelos artigos estudado.

Por fim, durante a realização da RSL, alguns artigos não especificaram o tempo e recursos dispendidos em cada parte do projeto de modelagem (geração da base de treino, treinamento e validação, e otimização), tampouco compará-lo com outros métodos existentes. A clarificação destes pontos em futuras pesquisas permite o leitor avaliar e comparar seus resultados e recursos disponíveis, além de compreenderem as vantagens e desvantagens da escolha pela modelagem frente a outros métodos disponíveis.

4. **FRAMEWORK DE METAMODELAGEM PROPOSTO**

Este Capítulo apresentará a segunda etapa do método de pesquisa utilizado nesta dissertação, a etapa de modelagem, na qual o *framework* para condução de projetos de metamodelagem para OvS será proposto.

4.1. **Técnicas Utilizadas para Proposição do *Framework***

O *framework* proposto nesta dissertação, para condução de projetos de metamodelagem para OvS, faz uso de quatro técnicas principais, a saber:

- 1) Simulação a eventos discretos: utilizada para modelagem dos sistemas reais e obtenção dos dados de entrada e saída;
- 2) *Design of experiments*: utilizado para criação da matriz amostral com os cenários a serem experimentados;
- 3) Algoritmos de *machine learning*: usados para construção dos metamodelos;
- 4) Algoritmo genético: aplicada para otimização dos hiper-parâmetros dos metamodelos.

A opção pela SED deu-se pela sua aplicabilidade aos objetos de estudo analisados, visto que estes apresentam processos sujeitos a estocasticidade e interrelacionados, além da natureza discreta do sistema. Além disto destaca-se que a SED foi identificada pela RSL como a principal técnica de simulação empregada em projetos de OvSM.

A segunda técnica utilizada, o DOE, é responsável pela geração de uma amostra representativa do espaço de solução, valendo-se da prerrogativa que a qualidade dos dados de treinamento (*input*) é fator crítico de sucesso em projetos de ML, e conseqüentemente na eficácia dos metamodelos em emular o modelo de simulação. Diversas técnicas de DOE podem ser encontradas na literatura, tais como *orthogonal array*, CCD, planejamento fatorial, entre outros.

Baseado nos resultados da RSL, este trabalho optou pelo uso do LHD, por se mostrar o mais popular em projetos similares a este. Embora haja um consenso de que o número de pontos amostrais no LHD não deva ser inferior a $10x$, no qual x é a quantidade de variáveis de decisão do problema, não foi encontrado na RSL uma recomendação para o número ideal de amostras para o treinamento de metamodelos.

A fim de comparar a influência do tamanho do arranjo na qualidade do metamodelo optou-se pela aplicação do método com 5 arranjos LHD, $20x$, $40x$, $60x$, $80x$ e $100x$, além de um arranjo aleatório de tamanho $100x$. Os arranjos serão comparados quanto suas métricas de erro e a qualidade da FO gerada após a otimização.

Os algoritmos de ML utilizados na construção dos metamodelos foram escolhidos com base na RSL, sendo eles o GP, RP, RNA, *ensembles* e SVM. Como representante dos métodos *ensembles* foram selecionados os algoritmos RF e GBT, ambos baseados em árvores de decisão. A escolha dos hiper-parâmetros para construção de cada algoritmo é fator primordial para a melhoria do ajuste dos metamodelos e minimizar o risco de *overfitting* dos mesmos. Logo, o *framework* proposto neste trabalho incorpora a etapa de otimização de hiper-parâmetros durante a construção e treino dos metamodelos, sendo esse realizado por meio de AG, uma vez que foi o mais popular e indicado pela RSL.

Como parâmetros iniciais do AG foram adotados o padrão do *software*, visto que não é objetivo deste trabalho discutir a melhor parametrização do AG. Logo, utilizou-se uma população inicial de 5 indivíduos, o torneio será considerado como método de avaliação para determinar a próxima geração da população, o tipo de mutação será a gaussiana, a probabilidade de cruzamento de 0,9, e o número de gerações fixado em 50.

Por fim, devido à natureza puramente discreta das variáveis de decisão, na otimização final do metamodelo para definição do cenário ótimo optou-se pela geração, através da aplicação dos metamodelos, de 100% do espaço de busca, sendo as soluções ranqueadas por ordem de grandeza da FO e selecionada a de melhor performance. Todavia, em problemas com variáveis de decisão contínuas recomenda-se o uso de algoritmos genéticos para problemas mono-objetivos e NSGA II para multiobjetivos.

4.2. *Framework* para Metamodelagem em problemas de Otimização via Simulação

A Figura 4.1 apresenta o *framework* de OvSM proposto neste trabalho, que incorpora tanto atividades estabelecidas no ambiente de simulação, à direita, quanto as desenvolvidas no ambiente de metamodelagem, à esquerda. Esta abordagem assume que o modelo de simulação já foi construído, verificado e validado (Passo 0), etapa fundamental para sua otimização. Vale ressaltar que este *framework* foi proposto com base nos *insights* obtidos na RSL, em especial na Seção 3.4.4, e no trabalho de Miranda *et al.* (2017).

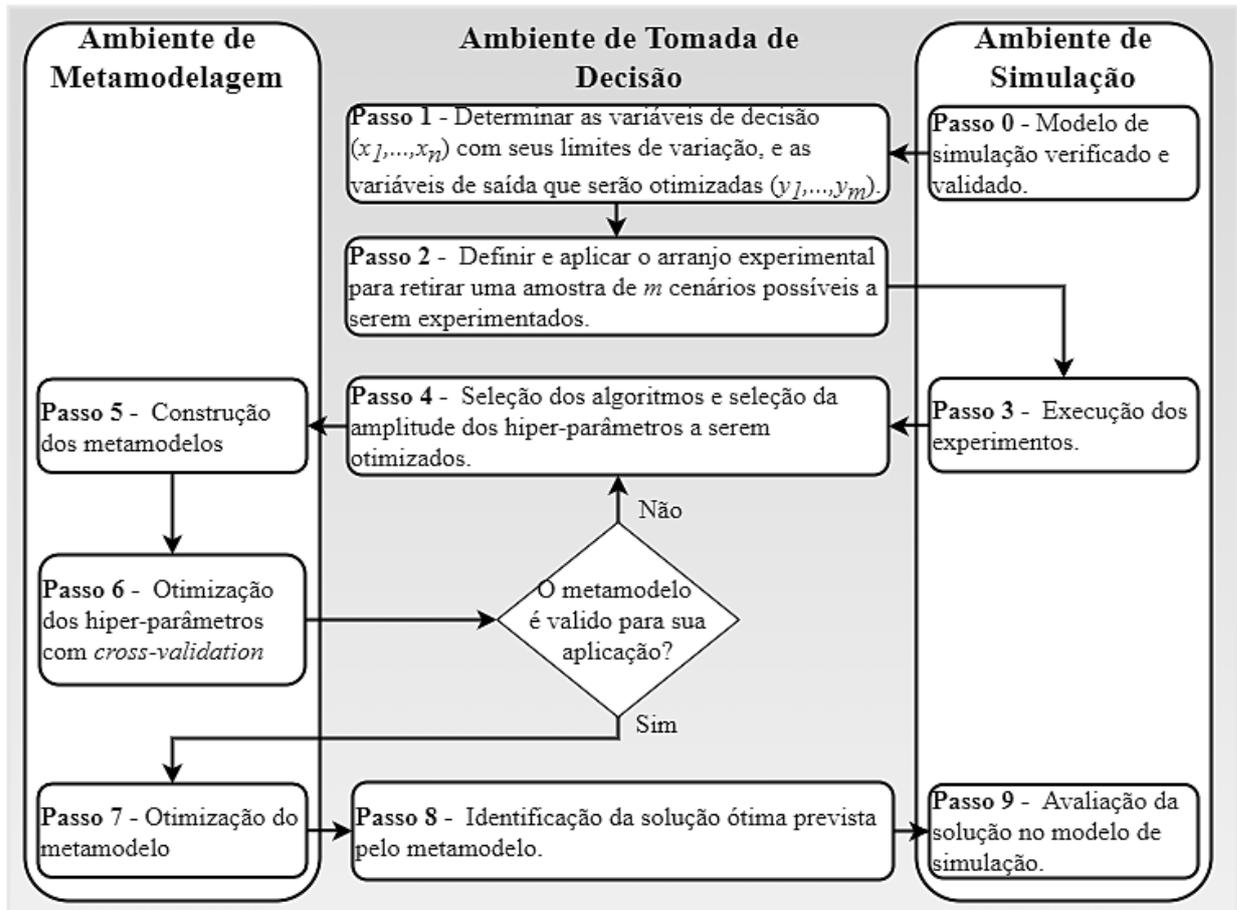


Figura 4.1 - *framework* proposto para otimização via simulação por metamodelagem

Os passos do *framework* são descritos como se segue:

- 1) Determine as variáveis de decisão (x_i, \dots, x_n) e suas amplitudes de variação, bem como a FO a ser otimizada (y) ;
- 2) Defina o arranjo experimental e selecione uma amostra de m cenários que irão compor a base de treinamento dos metamodelos;
- 3) Execute os experimentos no modelo de simulação e armazene os dados simulados;
- 4) Selecione os algoritmos de ML e os seus hiper-parâmetros a serem otimizados, assim como sua amplitude de variação;
- 5) Construa a estrutura do metamodelo no *software* ou linguagem de programação escolhida;
- 6) Execute a otimização dos hiper-parâmetros com validação e teste realizados por meio da técnica de *cross-validation*. Avaliar se as métricas de erro e ajuste são aceitáveis perante os critérios do tomador de decisões, se sim, siga para o passo 7, se não volte ao passo 4;

- 7) Execute a otimização do metamodelo para encontrar os valores das variáveis de decisão que otimizem o valor previsto da FO definida no passo 1. Neste passo, o tomador de decisão poderá utilizar as técnicas de otimização que julgar convenientes, tais como AG, *Ranking* e Seleção, NSGA II, entre outros;
- 8) Identifique a solução que otimiza a FO prevista pelo metamodelo;
- 9) Avalie a solução encontrada no modelo de simulação.

4.3. Procedimentos Metodológicos do Trabalho

A fim de avaliar a aplicabilidade do *framework* proposto foram realizadas três aplicações em problemas reais de OvS. Os objetos de estudo selecionados foram adaptados do trabalho de Miranda *et al.* (2017) e correspondem a uma célula de produção de uma empresa do setor de telecomunicações, controle de qualidade de uma empresa de equipamentos de transmissão, e produção de uma empresa do setor de autopeças.

Neste trabalho será utilizado o software *ProModel* versão 18.2 (PROMODEL 2020), para o desenvolvimento dos modelos de simulação, já para a criação dos metamodelos optou-se pelo uso do software *RapidMiner* versão 9.6.000 (RAPIDMINER, 2020), pois o mesmo apresenta interface amigável, processamento paralelo, e uma ampla variedade de algoritmos de ML e operadores de otimização, ambos necessários neste trabalho, além da possibilidade de integração com R e *Python*, o que aumenta a variedade de algoritmos a serem implementados.

Como comparativo, optou-se pelo uso do otimizador comercial *SimRunner* (SIMRUNNER, 2002). O *SimRunner* é um *software* de otimização integrado ao *ProModel* e baseado em uma metaheurística, que associa AG e estratégias evolutivas, sendo uma variante de algoritmos evolucionários. Este *software* permite o usuário trabalhar com três perfis diferentes, a saber: Cauteloso, Moderado e Agressivo. No perfil Cauteloso será utilizado avaliado um número maior de possíveis soluções para o problema, promovendo assim uma busca mais completa no espaço de soluções. No perfil Agressivo, por outro lado, o *software* opta por utilizar menos avaliações que no perfil anterior para reduzir o tempo computacional gasto na otimização, porém isto pode penalizar a qualidade da solução encontrada. Por fim, o perfil Moderado busca o balanceamento entre os perfis anteriores (SIMRUNNER, 2002).

A máquina escolhida para realização dos experimentos foi um notebook com 8 GB de RAM equipado com processador *Intel i5 7200U* de 4 *threads* e 2 núcleos com velocidade de 2.70 GHz cada.

Para aplicação e validação do *framework* proposto, o procedimento metodológico utilizado neste trabalho será conduzido da seguinte forma:

- Modelagem da SED: Modelar em ambiente de simulação (*software ProModel*) um sistema produtivo real;
 - Definição do *benchmark*: Realizar uma OvS utilizando-se o software *SimRunner* (SIMRUNNER, 2002), a fim de gerar um valor *benchmark* para a solução ótima;
 - Dados para treinamento: Selecionar pela técnica LHD uma amostra de tamanho m para compor a base de treino. A fim de se analisar a influência do tamanho amostral na qualidade dos metamodelos, estes serão treinados com arranjos LHD de tamanho $20x$, $40x$, $60x$, $80x$ e $100x$, sendo x o número de variáveis do problema. Além disto, também será utilizado um arranjo aleatório com tamanho amostral igual a $100x$;
 - Metamodelagem: Construir no *software RapidMiner* os algoritmos GP, RP, ANN, RF, GBT e SVM. Além disto o modelo deverá contemplar uma estrutura de otimização via AG dos hiper-parâmetros de cada algoritmo utilizado, bem como sua validação pela técnica de *k-fold cross-validation*, com $k = 10$. A opção pelo *k-fold cross-validation* deve-se ao ganho de velocidade se comparado ao *leave-one-out cross-validation*, sem perdas consideráveis na confiabilidade;
 - Previsão: Aplicar os algoritmos com hiper-parâmetros ótimos para prever o valor da FO em todo espaço de solução do problema;
 - Encontrar o ótimo via metamodelagem: Em cada algoritmo ranquear as soluções de modo a encontrar a que apresente o maior valor predito e posteriormente testá-la em ambiente de simulação;
 - Avaliação: Avaliar e comparar o resultado dos algoritmos em relação as métricas de erro (RMSE, MAE, MAPE) e ajuste (R^2), e as soluções ótimas encontradas perante o valor do *benchmark*. Além disto será comparado os resultados obtidos entre os arranjos experimentais utilizados e o ganho obtido com o uso da otimização de hiper-parâmetros. Com base nos resultados obtidos com a aplicação deste método pretende-se responder as seguintes perguntas:
- Qual é o ganho de eficiência promovido pelo uso do arranjo LHD frente um arranjo experimental aleatório?
 - Qual o ganho de performance e consequente custo computacional promovido pelo uso da otimização de hiper-parâmetros por AG?
 - O uso da metamodelagem permite ganhos de tempo computacional e/ou qualidade da resposta frente a otimização pelo *software* comercial *SimRunner*?

A fim de responder a primeira questão, comparar-se-á o RSME do arranjo aleatório com o arranjo LHD de mesmo tamanho (100x) obtido em cada OE. Da mesma forma será comparada a performance dos metamodelos construídos com base no *framework* proposto nesta dissertação, com otimização de hiper-parâmetros, e sem esta etapa de otimização, a fim de se comprovar ou refutar a importância desta técnica.

A terceira questão poderá ser respondida pela observação e comparação do tempo computacional e valor da FO entre o metamodelo e o *SimRunner*, em cada OE analisado. Por fim, será discutido, com base nos resultados encontrados com a aplicação do método nos três casos e os *insights* promovidos pela RSL, quando é indicado o uso da metamodelagem.

5. APLICAÇÃO DO *FRAMEWORK* PROPOSTO

Este capítulo trata da aplicação do *framework* proposto, atendendo a fase de solução do método de Mitroff *et al.* (1974).

5.1. Primeiro Objeto de Estudo

5.1.1. Apresentação do Problema

O primeiro objeto de estudo (OE1) representa uma célula produtiva do setor de telecomunicações. A escolha deste caso deu-se por ser um sistema real, complexo e com alto nível de manufatura, esboçando assim alta variabilidade em seus processos produtivos, além de variáveis de decisão com diferentes níveis.

Neste estudo foram utilizadas 7 variáveis de decisão inteiras, a saber: número de operadores (x_1), número de bancadas tipo 1 (x_2), número de bancadas tipo 2 (x_3), variável binária para atividade de organizador material (x_4), e tamanho dos estoques intermediários (x_5 , x_6 , x_7). O objetivo da otimização foi encontrar o valor que maximiza o lucro (y_1) da empresa. A modelagem conceitual desenvolvida sob a ótica da técnica IDEF-SIM (MONTEVECHI *et al.*, 2010) encontra-se no Anexo B deste trabalho.

Dadas as variáveis de decisão e sua amplitude de variação, apresentadas na Tabela 5.1, o problema possui um espaço de solução de 127.000 configurações possíveis, e considerando que para se gerar uma solução (com 30 replicações) demanda-se um tempo computacional de 33 segundos, a geração de todo espaço de solução gastaria aproximadamente 4.191.000 segundos, ou 48,51 dias.

Tabela 5.1 - Variáveis de decisão do problema para OE1

	Variável de Decisão	Tipo	Limite Inferior	Limite Superior
x_1	Número de operadores	Inteiro	1	3
x_2	Número de bancadas com <i>setup</i>	Inteiro	1	4
x_3	Número de bancadas sem <i>setup</i>	Inteiro	1	4
x_4	Atividade organizar material	Binário	0	1
x_5	Tamanho do estoque intermediário 1	Inteiro	5	15
x_6	Tamanho do estoque intermediário 2	Inteiro	5	15
x_7	Tamanho do estoque intermediário 3	Inteiro	5	15

A fim de se estabelecer um valor *benchmark* a ser utilizado na avaliação dos metamodelos, foi construído um modelo de otimização com um horizonte de simulação de um mês e com um número de replicações igual a 30. A formulação matemática do problema

objetiva maximizar o lucro, Eq. (25), sujeito as restrições (Eq. (26) e Eq. (27)) dos valores das variáveis de decisão x_i , com $i \in \mathbb{Z} (1, \dots, 7)$.

$$\text{Max} \mathbb{E}_\omega \left\{ f(\tilde{\phi}(\sum_{i=1}^7 x_i)) \right\} \quad (25)$$

Sujeito a:

$$L_i \leq x_i \leq U_i \quad (26)$$

$$x_i \in \mathbb{Z} + \quad \forall i \in \mathbb{Z} + \quad (27)$$

Sendo:

x_i = Quantidade do recurso i ;

\mathbb{E}_ω = Valor esperado da função objetivo (Lucro) avaliada via simulação;

f = Valor do vetor função objetivo (Lucro);

$\tilde{\phi}$ = Modelo estocástico representado pelo modelo de simulação;

L_i = Limite inferior de variação do recurso i , $L_i = [1 \ 1 \ 1 \ 0 \ 5 \ 5 \ 5]$;

U_i = Limite superior de variação do recurso i , $U_i = [3 \ 4 \ 4 \ 1 \ 15 \ 15 \ 15]$.

Para execução da OvS utilizou-se o *software* comercial *SimRunner*. A otimização deu-se no modo moderado do software. Foram realizadas 245 iterações com um tempo gasto de 8.087 segundos para obtenção de uma solução. A solução encontrada (valor *benchmark*) pode ser observada na Tabela 5.2, com um lucro de \$ 54.933,00. Ressalta-se que, devido o software utilizar técnicas metaheurísticas, não se garante que esta solução seja o ótimo global do problema, porém uma boa solução que pode servir de *benchmark* no momento de avaliação do metamodelo.

Tabela 5.2 - Solução encontrada pelo otimizador *SimRunner* para o OE1

Variável de Decisão	Tipo	Solução Ótima
x_1 Número de operadores	Inteiro	3
x_2 Número de bancadas com <i>setup</i>	Inteiro	3
x_3 Número de bancadas sem <i>setup</i>	Inteiro	2
x_4 Atividade organizar material	Binário	1
x_5 Tamanho do estoque intermediário 1	Inteiro	12
x_6 Tamanho do estoque intermediário 2	Inteiro	13
x_7 Tamanho do estoque intermediário 3	Inteiro	5
Lucro Total (x1000 \$)		54,93
IC 95% para o Lucro (x1000 \$)		(53,08; 56,78)
Tempo Gasto na Otimização (s)		8.087

5.1.2. Aplicação e Comparação dos Metamodelos

Neste estudo os algoritmos utilizados para construção dos metamodelos foram: SVM, RF, GBT, RNA, GP e RP. Os modelos construídos englobam duas fases principais: a primeira

diz respeito ao treinamento e otimização dos hiper-parâmetros do algoritmo de modo a minimizar o RMSE, com base na amplitude de variação dos parâmetros identificados na literatura. A Tabela 5.3 contém amplitude para cada parâmetro a ser otimizado via AG, as quais foram selecionadas com base na RSL e apresentadas na Seção 3.4.2.

Tabela 5.3 - Seleção dos hiper-parâmetros para o OE1

Algoritmo	Amplitude	Algoritmo	Amplitude
GP		RF	
<i>Lengthscale</i> (θ)	$[10^{-10}; 10^{10}]$	Nº de árvores (t)	[10; 50]
<i>Epsilon</i> ^l (e)	$[10^{-6}; 10^{-5}]$	Nº de variáveis por nó(v)	{1; 2; 3}
Tolerância geométrica (G)	[0; 0,3]		
Nº max. de vetores (mbv)	[38; 47]	GBT	
RP		Nº de árvores (t)	[1; 100]
<i>Degree</i> (D)	{1º, 2º, 3º, 4º, 5º}	<i>Learning rate</i> (α)	[0,05; 0,30]
		Min. de Obs. Por folha (l)	[0,12; 0,50]
RNA		SVM	
<i>Learning rate</i> (α)	[0,01; 0,4]	C^*	[18,30; 72,68]
<i>Epochs</i> (E)	[200; 1500]	<i>Epsilon</i> [*] (e)	[0; 1,88]
Nº de camadas (ω)	1	Gama (γ)	[0,0224; 0,1294]
Nº de neurônios (n)	[1; 66]		

* Os valores para C e *epsilon* foram calculados a partir de uma execução preliminar do algoritmo com parâmetros *default*.

Definida a amplitude de cada hiper-parâmetro, o próximo passo é o treinamento e otimização dos hiper-parâmetros para cada um dos arranjos experimentais. Nesta etapa serão comparados 6 arranjos experimentais (5 LHD e 1 aleatório) e 6 algoritmos de ML (GP, RP, RNA, RF, GBT e SVM), totalizando um total de 36 configurações a serem analisadas. A Tabela 6.4 apresenta o resultado dos metamodelos para cada uma destas 36 configurações, destacando o comparativo entre os metamodelos treinados com hiper-parâmetros *default* e otimizados via AG (segundo *framework* proposto) quanto ao RMSE, tempo dispendido para geração dos arranjos experimentais e treinamento e/ou otimização dos metamodelos (Tempo), e o lucro obtido com solução ótima de cada metamodelo, avaliado no modelo de simulação.

Tabela 5.4 - Metamodelos aplicados ao OE1

Arranjo (Metamodelo)	Hiper-parâmetros <i>default</i>			Hiper-parâmetros otimizados		
	RMSE	Tempo(s)	Lucro (mil \$)	RMSE	Tempo(s)	Lucro (mil \$)
LHD 20x						
GP	9,62	6	47,63	5,68	91	46,93
RP	122,02	10	48,90	6,45	205	42,57
RNA	2,73	2	48,53	2,18	228	49,70
RF	4,48	9	52,30	3,98	29	47,57
GBT	8,72	6	54,90	3,40	87	47,40
SVM	6,24	2	47,40	2,34	170	54,93
LHD 40x						
GP	7,99	11	52,30	4,76	106	45,50
RP	81,96	11	46,90	6,63	353	47,40
RNA	2,84	2	52,73	2,48	863	49,70
RF	3,36	12	54,93	3,29	7	54,93
GBT	8,23	7	52,13	3,31	140	52,13
SVM	5,49	2	43,27	2,55	176	54,93
LHD 60x						
GP	6,86	13	54,93	4,94	410	47,27
RP	35,91	10	3,53	6,71	434	32,40
RNA	3,07	3	54,93	2,20	701	54,93
RF	2,65	15	54,93	2,57	61	54,83
GBT	8,03	4	52,30	2,56	69	52,73
SVM	5,42	2	41,67	2,30	255	54,93
LHD 80x						
GP	7,21	20	54,93	5,11	294	47,13
RP	45,07	21	3,20	6,32	952	32,40
RNA	3,10	2	54,93	2,26	1399	55,10
RF	2,73	18	54,83	2,54	51	55,10
GBT	8,18	5	52,13	3,19	75	54,83
SVM	5,67	4	41,67	2,55	512	54,93
LHD 100x						
GP	6,32	27	54,93	4,85	284	48,33
RP	43,42	15	31,60	4,82	535	42,57
RNA	2,89	3	54,93	2,00	3887	49,70
RF	2,61	15	54,93	2,54	69	54,43
GBT	8,21	11	52,13	3,02	58	52,13
SVM	5,45	4	42,10	2,70	693	54,93
Aleatório 100x						
GP	6,95	19	50,70	5,50	199	46,27
RP	53,32	14	43,40	7,25	513	43,50
RNA	3,61	2	54,93	1,88	6453	53,20
RF	2,88	11	54,93	2,80	95	54,93
GBT	9,61	9	52,30	3,58	49	52,73
SVM	6,30	2	43,27	2,78	951	52,30

Com os dados da Tabela 5.4 divide-se a análise em três tópicos principais: quanto aos algoritmos de ML utilizados, quanto ao arranjo experimental e quanto ao uso da otimização de hiper-parâmetros. Considerando a média do lucro em todos os arranjos, o SVM foi o melhor metamodelo, com uma FO média de 54,49, seguida pelo RF, RNA, GBT, GP e RP, com 53,63,

52,05, 51,99, 46,90 e 40,14, respectivamente. Em termos de RMSE, o RNA foi o de melhor resultado, com uma média de 2,16. Já quanto ao valor unitário da FO, o melhor resultado foi encontrado com a utilização do arranjo LHD 80x, onde tanto o RF quanto a RNA obtiveram um lucro de \$ 55,10 mil.

Todavia, resguardando o conceito de eficiência do DOE (máximo resultado com o mínimo de experimentos), este resultado apresentado pelo arranjo LHD 80x não foi estatisticamente superior ao obtido pelo SVM no arranjo LHD 20x ($p\text{-value} = 0,649$ para teste *2-Sample T*), optando-se assim pelo arranjo de menor custo, ou seja, o LHD 20x. Este arranjo demandou um tempo computacional de 4.620 segundos para execução dos experimentos, que acrescidos dos 170 segundos para treinamento e otimização do SVM, esboçou uma redução de 40,76% no tempo computacional se comparado ao *benchmark*.

Em posse das 30 replicações para as soluções propostas pelo melhor metamodelo e pelo *SimRunner*, foi conduzido um teste de normalidade de Anderson-Darling, constatando a normalidade de ambas as amostras ($p\text{-value} = 0.644$ para SVM e para a solução encontrada pelo *SimRunner*). Em seguida foi realizado o Teste F ($p\text{-value} = 1,000$), comprovando que ambas as amostras possuem mesma variância. Por fim, a fim de averiguar a equidade entre as soluções foi realizado um Teste 2-Sample T ($p\text{-value} = 1,000$), e como o $p\text{-value}$ foi maior que o nível de significância adotado ($\alpha = 0.05$) pode-se considerar que ambas as soluções apresentam resultados estatisticamente iguais.

Quanto à métrica de erro, a Figura 5.1 apresenta o RMSE médio dos metamodelos treinados com hiper-parâmetros *default* e otimizados para cada um dos arranjos experimentais.

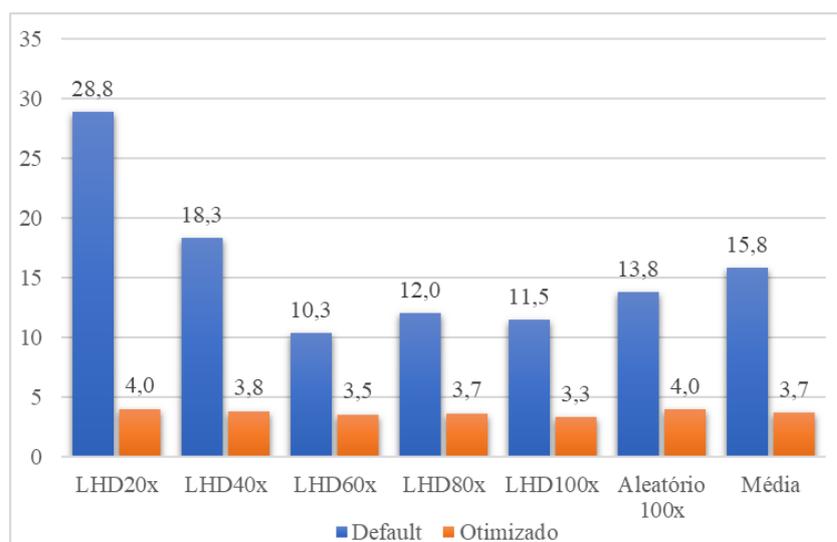


Figura 5.1 - Comparativo do RMSE médio entre arranjos para o OE1

No gráfico da Figura 5.1 é possível fazer três inferências principais. A primeira é que não necessariamente quanto maior for o número de amostras no arranjo LHD menor o erro associado ao metamodelo (considerando os metamodelos com hiper-parâmetros otimizados), como foi o caso do LHD 80x, que obteve um erro maior que o LHD 60x.

Em segundo ponto, pode-se destacar a superioridade do LHD sobre o arranjo aleatório, sendo que este se equiparou em termos de RMSE ao arranjo LHD 20x, mesmo demandando 5 vezes mais tempo computacional e experimentos, e se comparado ao LHD 100x (arranjo com mesmo número de experimentos) teve um RMSE 21,2% maior. Por fim ressalta-se a redução do RMSE promovido pelo uso do AG para otimização dos hiper-parâmetros, que passou de uma média de 15,8 com o *default* para 3,7 com o uso do AG, obtendo uma redução média de 76,58%.

A respeito do lucro, e dado a não normalidade das amostras ($p\text{-value} < 0,005$ para ambos no teste de Anderson-Darling) foi realizado um teste não paramétrico de Man-Whitney, que constatou que, em respeito a FO, os metamodelos *default* e otimizados possuíam a mesma mediana.

Porém destaca-se que valores mais elevados foram obtidos pelos metamodelos otimizados, como foi o caso do RF e RNA com 55,10 mil dólares de lucro. No caso do SVM, o uso da otimização de hiper-parâmetros foi fundamental, pois em todas as ocorrências observou-se uma melhora significativa no lucro, sendo que em média obteve-se uma diferença de 26,04%, sugerindo a sensibilidade deste algoritmo aos hiper-parâmetros escolhidos.

Na Figura 5.2 observa-se que com o uso do AG obteve-se um RMSE menor e com intervalos de confiança mais estreitos, o que aumenta a confiabilidade do método.

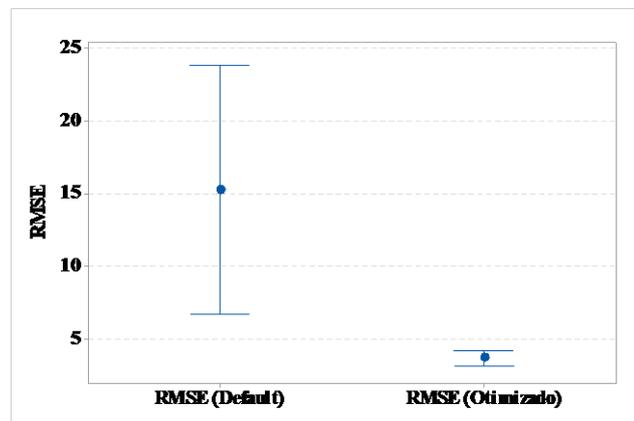


Figura 5.2 - IC95% para o RMSE

A Tabela 5.5 apresenta o comparativo dos metamodelos quanto ao lucro previsto por estes para a melhor solução, o lucro observado pela avaliação desta no modelo de simulação, e

o tempo dispendido na otimização e treinamento, ambos para o arranjo LHD 20x, que foi escolhido por ter atingido um lucro estatisticamente igual ao arranjo 80x, porém com 4 vezes menos experimentos.

Tabela 5.5 - Metamodelos aplicados ao OE1

Metamodelo	RMSE	MAE	MAPE	R ²	T – Opt.(s)	Lucro Observado (mil \$)	Hiper-parâmetros
SVM	2,34	1,70	8,73%	0,94	170	54,93	$C = 69,24; e = 0,10; \gamma = 0,09$
GBT	3,40	2,20	10,92%	0,92	87	47,40	$t = 94; \alpha = 0,06; l = 0,38$
RNA	2,18	1,64	9,73%	0,98	228	49,70	$\alpha = 0,16; E = 830; n = 16$
RF	3,98	2,59	14,34%	0,90	29	47,57	$t = 35; v = 4$
GP	5,68	4,33	27,07%	0,84	91	46,93	$\theta = 15,09; e = 1 \times 10^{-6}; G = 0; mbv = 42$
RP	6,45	5,28	31,46%	0,75	205	42,57	$D = 1^\circ$

Como observado na tabela, o metamodelo de melhor FO foi o SVM, sendo também o que obteve o menor MAPE. Quanto ao RSME, MAE e R² o metamodelo de melhor performance foi o RNA, todavia obteve um lucro de apenas 49,70 mil dólares. A Tabela 5.6 apresenta o comparativo dos metamodelos quanto lucro previsto por estes para a melhor solução, o lucro observado pela avaliação desta no modelo de simulação, e o tempo dispendido na execução dos experimentos amostrais, otimização e treinamento.

Tabela 5.6 - Comparativo dos metamodelos para o OE1

Metamodelo	Solução ($x_1; x_2; x_3; x_4; x_5; x_6; x_7$)	Solução Final			Tempo (s)	
		Lucro Previsto (mil \$)	Lucro Observado (mil \$)	Rank	Valor	Rank
SVM	(3; 3; 2; 1; 9; 11; 11)	54,75	54,93	1,00	4790	0,97
GBT	(3; 3; 2; 1; 15; 12; 9)	55,69	47,40	0,86	4707	0,99
ANN	(3; 3; 4; 1; 15; 15; 5)	58,42	49,70	0,90	4848	0,96
RF	(3; 3; 1; 1; 7; 6; 12)	49,10	47,57	0,87	4649	1,00
GP	(3; 4; 1; 1; 9; 8; 9)	46,96	46,96	0,85	4711	0,99
RP	(3; 1; 1; 0; 5; 5; 5)	42,57	42,57	0,77	4825	0,96
Referência: <i>SimRunner</i>	(3; 3; 2; 1; 12; 13; 5)	-	54,93	1,00	8087	0,57

O metamodelo de melhor desempenho, SVM, demandou um tempo de 4.790 segundos para otimização, treinamento e previsão de 100% do espaço de solução do problema, como mostrado na Figura 5.3 (a). Atrelado a isto, ressalta-se que, uma vez treinado o metamodelo pode ser utilizado inúmeras vezes demandando apenas o tempo de previsão, que para este caso foi de 13 segundos para previsão de 100% do espaço de solução. Isto torna-se importante em aplicações que necessitam de uma boa resposta em um intervalo de tempo que torna proibitivo

o uso de técnicas convencionais de OvS, como os emergentes gêmeos digitais. Além disto, destaca-se que o RF foi o metamodelo mais rápido, permitindo a tomada de decisão em um tempo de 4.649 segundos.

Outro importante indicador é o tempo gasto para treinamento e otimização dos hiper-parâmetros, seu comparativo é mostrado na Figura 5.3 (b). Neste quesito, o metamodelo mais veloz foi o RF que demandou 20 segundos, porém deve-se levar em consideração a quantidade de hiper-parâmetros e suas amplitudes, que influencia diretamente no tempo demandado para otimização.

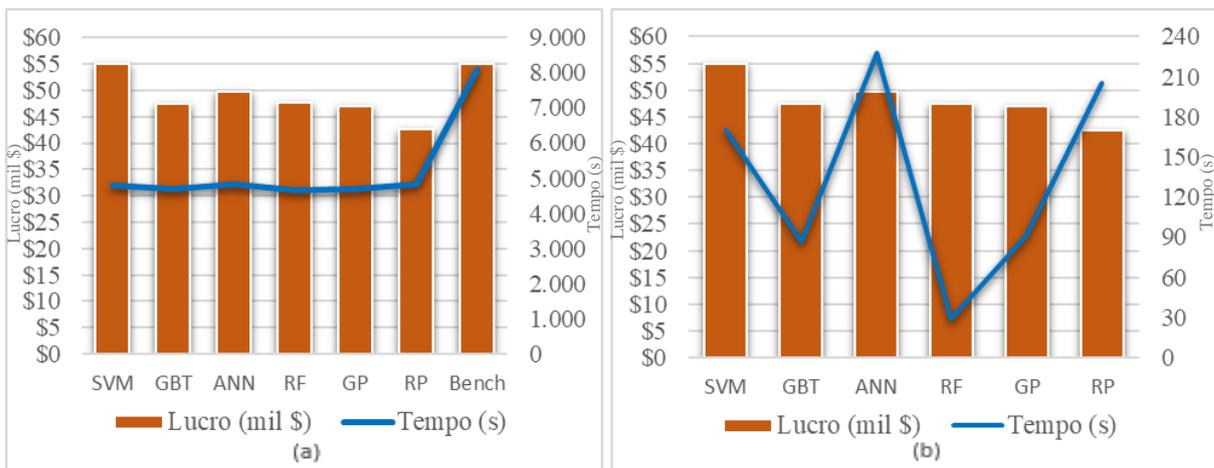


Figura 5.3 - (a) Comparativo de FO e tempo para otimização. (b) Comparativo de FO e tempo para treinar e otimizar os metamodelos para o OE1

Além disto, comparando-se este tempo ao demandado pelo otimizador comercial *SimRunner* para atingir o ótimo local (8.087 segundos), o uso do *framework* proposto neste trabalho proporcionou uma redução de 40,76% no tempo computacional, atingindo a mesma FO.

5.2. Segundo Objeto de Estudo

5.2.1. Apresentação do Problema

O segundo objeto de estudo (OE2) representa uma célula de controle de qualidade de uma empresa produtora de transmissores, cujo principal objetivo é a maximização da função lucro.

Este case possui 6 variáveis de decisão, correspondendo ao número de operadores responsáveis por cada teste de qualidade (x_1, x_2, x_3) e ao número de equipamentos para realização dos testes dos tipos 1, 2 e 3 (x_4, x_5, x_6). Considerando os limites inferiores de

variação dos recursos, como mostrado na Tabela 5.7, o OE2 possui um total de 1.000.000 soluções possíveis.

Tabela 5.7 - Variáveis de decisão do problema para o OE2

	Variável de Decisão	Tipo	Limite Inferior	Limite Superior
x_1	Operadores tipo 1	Inteiro	1	10
x_2	Operadores tipo 2	Inteiro	1	10
x_3	Operadores tipo 3	Inteiro	1	10
x_4	Número de equipamentos tipo 1	Inteiro	1	10
x_5	Número de equipamentos tipo 2	Inteiro	1	10
x_6	Número de equipamentos tipo 3	Inteiro	1	10

A formulação matemática do problema objetiva maximizar o lucro, Eq. (28), sujeito as restrições (Eq. (29) e Eq. (30)) dos valores das variáveis de decisão x_i , com $i \in \mathbb{Z} (1, \dots, 6)$.

$$\text{Max} \mathbb{E}_\omega \left\{ f(\tilde{\phi}(\sum_{i=1}^6 x_i)) \right\} \quad (28)$$

Sujeito a:

$$L_i \leq x_i \leq U_i \quad (29)$$

$$x_i \in \mathbb{Z} + \quad \forall i \in \mathbb{Z} + \quad (30)$$

Sendo:

x_i = Quantidade do recurso i ;

\mathbb{E}_ω = Valor esperado da função objetivo (Lucro) avaliada através da simulação;

f = Valor do vetor função objetivo (Lucro);

$\tilde{\phi}$ = Modelo estocástico representado pelo modelo de simulação.

L_i = Limite inferior de variação do recurso i , $L_i = [1 \ 1 \ 1 \ 1 \ 1 \ 1]$;

U_i = Limite superior de variação do recurso i , $U_i = [10 \ 10 \ 10 \ 10 \ 10 \ 10]$.

Dada estas variáveis e suas amplitudes, o problema possui um espaço de soluções de 1.000.000 de cenários possíveis. Para este caso, o modelo de simulação demanda 152 segundos para avaliar cada cenário, com 30 replicações, desta forma, estima-se um total de 1.759,26 dias para a avaliação completa do espaço de solução pelo modelo de simulação. Foram realizadas 451 iterações com um tempo gasto de 68.552 segundos para obtenção de uma solução. A solução encontrada (valor *benchmark*) pode ser observada na Tabela 5.8, com um lucro de \$ 330,12.

Tabela 5.8 - Solução encontrada pelo otimizador *SimRunner* para o OE2

Variável de Decisão	Tipo	Solução Ótima
x_1 Operadores tipo 1	Inteiro	8
x_2 Operadores tipo 2	Inteiro	5
x_3 Operadores tipo 3	Inteiro	6
x_4 Número de equipamentos tipo 1	Inteiro	3
x_5 Número de equipamentos tipo 2	Inteiro	4
x_6 Número de equipamentos tipo 3	Inteiro	2
Lucro Total (mil \$)		330,12
IC 95% para o Lucro		(328,01; 332,51)
Tempo Gasto na Otimização (s)		68.552

5.2.2. Aplicação e Comparação dos Metamodelos

A amplitude para os hiper-parâmetros dos metamodelos e o código em XML encontram-se nos anexos deste trabalho. A Tabela 5.9 apresenta o resultado dos metamodelos treinados com hiper-parâmetros *default* e para os com hiper-parâmetros otimizados pelo AG.

Tabela 5.9 - Metamodelos aplicados ao OE2

Arranjo (Metamodelo)	Hiper-parâmetros <i>default</i>			Hiper-parâmetros otimizados		
	RMSE	Tempo(s)	Lucro (mil \$)	RMSE	Tempo(s)	Lucro (mil \$)
LHD 20x						
GP	131743	27	301,77	29006	232	294,38
RP	138296	17	154,52	130964	567	201,44
RNA	32826	15	223,56	16933	493	292,77
RF	22194	84	311,25	21569	211	311,25
GBT	41041	32	240,56	13566	207	295,59
SVM	54730	15	245,49	23908	101	300,91
LHD 40x						
GP	97131	36	312,58	22631	329	307,38
RP	141959	14	187,71	128465	267	170,22
RNA	27925	18	268,26	10228	1603	312,53
RF	20269	82	316,61	19001	87	317,45
GBT	42755	23	318,56	9875	88	313,30
SVM	58625	22	187,71	21566	675	316,05
LHD 60x						
GP	54332	33	175,66	45553	315	221,39
RP	158379	14	135,82	156099	829	114,19
RNA	50416	15	240,56	48654	1145	115,72
RF	46142	76	242,15	44834	99	207,22
GBT	49808	27	175,05	44285	234	198,12
SVM	57355	14	187,71	44712	179	190,78
LHD 80x						
GP	50689	29	164,34	45479	186	227,26
RP	160081	32	217,30	151386	477	109,52
RNA	47193	17	221,52	45027	552	217,73
RF	43992	80	242,15	43436	44	192,52
GBT	47436	27	234,90	42325	78	169,69
SVM	54477	12	217,30	42779	162	175,88

LHD 100x						
GP	76671	31	307,17	19841	284	306,42
RP	133172	21	131,59	139042	362	154,52
RNA	20942	15	255,69	7735	3437	317,11
RF	14872	83	319,39	14830	160	319,76
GBT	38444	25	254,98	7850	52	315,77
SVM	55678	13	245,49	18977	634	312,11
Aleatório 100x						
GP	87447	38	296,52	20540	408	306,07
RP	137248	19	131,59	135289	578	154,52
RNA	19181	15	271,70	7961	1602	316,11
RF	15664	77	310,05	15243	94	316,15
GBT	35506	26	254,98	7309	102	326,88
SVM	49044	14	187,71	18653	1801	205,70

Ao analisar cada metamodelo separadamente, o GBT gerado a partir do arranjo aleatório 100x foi o de melhor desempenho, apresentando um RMSE de 7.309 e um lucro de 326,15 mil dólares. Todavia, em média, os arranjos que performaram melhor em termos de lucro foram o LHD 100x e LHD 40x, com 287,65 e 287,15 mil dólares, respectivamente. Resguardando o princípio da eficiência do DOE, optou-se pelo uso do arranjo LHD 40x.

Na Figura 5.4, que apresenta o RMSE médio dos metamodelos treinados com hiperparâmetros *default* e otimizados para cada um dos arranjos experimentais, observa-se que a métrica de erro do arranjo LHD 40x é 3,22% superior ao arranjo aleatório 100x, porém demandando 60% menos experimentos. Logo, optou-se pelo arranjo LHD 40x, que demandou um total de 240 experimentos e um tempo computacional de 36.480 segundos.

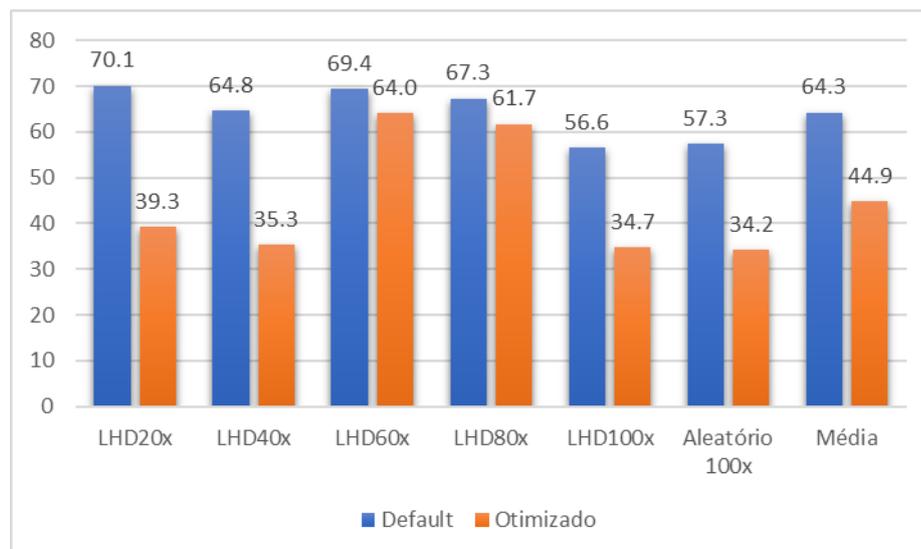


Figura 5.4 - Comparativo do RMSE médio entre arranjos para o OE2

De acordo com a Figura 5.4 destaca-se que, para este OE2, o arranjo aleatório superou o arranjo LHD de mesmo tamanho (LHD 100x), refutando que o LHD é a melhor opção para todos os projetos de metamodelagem, porém é capaz de gerar bons resultados em diferentes cenários, resguardando a peculiaridade de cada problema. Além disto, para esse OE não foi possível estabelecer uma relação direta entre tamanho amostral e qualidade do metamodelo, uma vez que os arranjos LHD 60x e LHD 80x possuíam métricas de erro piores que os arranjos LHD 20x e LHD 40x. Outro ponto é que a otimização de hiper-parâmetros permitiu uma redução no RMSE de, em média, 30,17%, e para o arranjo escolhido (LHD 40x) de 45,51%.

A Tabela 5.10 apresenta as principais métricas e os hiper-parâmetros ótimos para os metamodelos do arranjo LHD 40x.

Tabela 5.10 - Metamodelos aplicados ao OE2

Metamodelo	RMSE	MAE	MAPE	R ²	T – Opt.(s)	Lucro Observado (mil \$)	Hiper-parâmetros
RF	19001	14867	9,71%	0,90	87	317,45	$t = 33; v = 3$
SVM	21566	17323	10,30%	0,85	675	316,05	$C = 103832; e = 4076; \gamma = 0,10$
GBT	9875	7567	4,48%	0,97	88	313,30	$t = 63; \alpha = 0,26; l = 0,32$
RNA	10228	8271	4,52%	0,97	1603	312,53	$\alpha = 0,13; E = 1100; n = 60$
GP	22631	18206	10,64%	0,84	329	307,38	$\theta = 15,09; e = 1 \times 10^{-5}; G = 0; mbv = 39$
RP	128465	111760	51,70%	0,04	267	170,22	$D = 3^\circ$

Dentre os metamodelos, o RF foi o que apresentou o melhor lucro e menor tempo de otimização, todavia as melhores métricas de erro foram obtidas pelo GBT. A Tabela 5.11 apresenta as soluções apontadas como ótimas em cada metamodelo e um comparativo entre eles e o valor obtido pelo *SimRunner*.

Tabela 5.11 - Comparativo dos metamodelos para o OE2

Metamodelo	Solução Final			Tempo (s)		
	Solução ($x_1; x_2; x_3; x_4; x_5; x_6$)	Lucro Previsto (mil \$)	Lucro Observado (mil \$)	Rank	Valor	Rank
RF	(9; 8; 4; 3; 4; 2)	303,40	317,45	0,96	36.567	1,00
SVM	(7; 7; 6; 5; 5; 3)	348,34	316,05	0,96	37.155	0,98
GBT	(4; 10; 9; 4; 4; 2)	305,69	313,30	0,95	36.568	1,00
RNA	(8; 9; 5; 4; 4; 4)	322,06	312,53	0,95	38.083	0,96
GP	(7; 7; 6; 6; 6; 3)	339,75	307,38	0,93	36.809	0,99
RP	(10; 10; 10; 1; 1; 10)	399,91	170,22	0,52	36.747	1,00
Referência: <i>SimRunner</i>	(8; 4; 6; 3; 4; 2)	-	330,12	1,00	68.552	0,53

A fim de comparar as soluções encontradas pelo RF e o *benchmark*, a normalidade das amostras foi comprovada pelo teste de Anderson-Darling ($p\text{-value} = 0,391$ para o RF e $p\text{-value} = 0,167$ para o *SimRunner*) e, com o Teste F, pode-se comprovar a equidade das variâncias das amostras ($p\text{-value} = 0,568$). Considerando o nível de significância adotado ($\alpha = 0,05$), a realização do Teste 2-Sample *T* ($p\text{-value} < 0,001$) refutou a hipótese de igualdade dos resultados. Todavia, o valor de lucro encontrado pelo RF foi 3,83% menor que o *benchmark*, podendo ainda ser considerado um bom resultado dada a complexidade do problema.

Outro importante fator é o tempo dispendido, que se somando o tempo de geração dos cenários amostrais, treinamento e otimização do metamodelo, o RF demandou 36.567 segundos, uma redução de 46,65% frente ao tempo gasto pelo *SimRunner*. O comparativo entre os metamodelos podem ser observados na Figura 5.5.

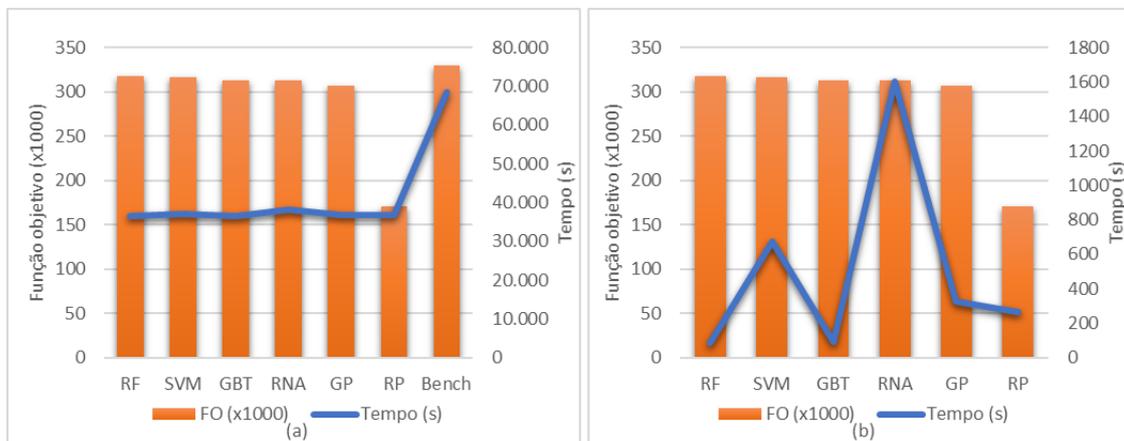


Figura 5.5 - (a) Comparativo de FO e tempo para otimização. (b) Comparativo de FO e tempo para treinar e otimizar os metamodelos para o OE2

5.3. Terceiro Objeto de Estudo

5.3.1. Apresentação do Problema

O terceiro OE3 visa a maximização da quantidade de peças produzidas em uma célula de produção de autopeças.

Neste sistema foram estudadas 11 variáveis de decisão, que correspondem ao número de operadores do tipo 1, 2, e 3 (x_1, x_2, x_3) e a quantidade de máquinas de cada um dos 8 tipos existentes ($x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}$). A Tabela 5.12 apresenta as variáveis de decisão do problema.

Tabela 5.12 - Variáveis de decisão para o OE3.

	Variável de Decisão	Tipo	Limite Inferior	Limite Superior
x_1	Operadores tipo 1	Inteiro	1	5
x_2	Operadores tipo 2	Inteiro	1	4
x_3	Operadores tipo 3	Inteiro	1	6
x_4	Número de equipamentos tipo 1	Inteiro	1	3
x_5	Número de equipamentos tipo 2	Inteiro	1	3
x_6	Número de equipamentos tipo 3	Inteiro	1	3
x_7	Número de equipamentos tipo 4	Inteiro	1	3
x_8	Número de equipamentos tipo 5	Inteiro	1	4
x_9	Número de equipamentos tipo 6	Inteiro	1	4
x_{10}	Número de equipamentos tipo 7	Inteiro	1	4
x_{11}	Número de equipamentos tipo 8	Inteiro	1	4

A formulação matemática do problema objetiva maximizar a produção, Eq. (31), sujeito as restrições (Eq. (32) e Eq. (33)) dos valores das variáveis de decisão x_i , com $i \in \mathbb{Z}(1, \dots, 11)$.

$$\text{Max} \mathbb{E}_\omega \left\{ f(\tilde{\phi}(\sum_{i=1}^{11} x_i)) \right\} \quad (31)$$

Sujeito a:

$$L_i \leq x_i \leq U_i \quad (32)$$

$$x_i \in \mathbb{Z} + \quad \forall i \in \mathbb{Z} + \quad (33)$$

Sendo:

x_i = Quantidade do recurso i ;

\mathbb{E}_ω = Valor esperado da função objetivo (produção) avaliada através da simulação;

f = Valor do vetor função objetivo (produção);

$\tilde{\phi}$ = Modelo estocástico representado pelo modelo de simulação.

L_i = Limite inferior de variação do recurso i , $L_i = [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1]$;

U_i = Limite superior de variação do recurso i , $U_i = [5 \ 4 \ 6 \ 3 \ 3 \ 3 \ 3 \ 4 \ 4 \ 4 \ 4]$.

Devido a amplitude das variáveis, o problema possui um total de 2.488.320 soluções possíveis, que multiplicados aos 96 segundos gastos para avaliação de cada solução com 30 replicações, demandaria em torno de 7,57 anos para avaliação todo espaço de solução pelo modelo de simulação. Com a utilização do *SimRunner* foram realizadas 325 iterações com um tempo gasto de 31.200 segundos para obtenção de uma solução. A solução encontrada (valor *benchmark*) pode ser observada na Tabela 5.13, com uma produção de 859.300 peças.

Tabela 5.13 - Solução encontrada pelo otimizador *SimRunner* para o OE3

Variável de Decisão	Tipo	Solução Ótima
x_1 Operadores tipo 1	Inteiro	1
x_2 Operadores tipo 2	Inteiro	3
x_3 Operadores tipo 3	Inteiro	5
x_4 Número de equipamentos tipo 1	Inteiro	3
x_5 Número de equipamentos tipo 2	Inteiro	3
x_6 Número de equipamentos tipo 3	Inteiro	1
x_7 Número de equipamentos tipo 4	Inteiro	2
x_8 Número de equipamentos tipo 5	Inteiro	3
x_9 Número de equipamentos tipo 6	Inteiro	4
x_{10} Número de equipamentos tipo 7	Inteiro	1
x_{11} Número de equipamentos tipo 8	Inteiro	4
Produção Total (peças)		859.300
IC 95% para a produção		(848.607; 869.992)
Tempo Gasto na Otimização (s)		31.200

5.3.2. Aplicação e Comparação dos Metamodelos

A Tabela 5.14 apresenta o resultado dos metamodelos treinados com hiper-parâmetros *default* e para os com hiper-parâmetros otimizados pelo AG.

Tabela 5.14 - Metamodelos aplicados ao OE3

Arranjo (Metamodelo)	Hiper-parâmetros <i>default</i>			Hiper-parâmetros otimizados		
	RMSE	Tempo(s)	Produção	RMSE	Tempo(s)	Produção
LHD 20x						
GP	113943	31	781300	89705	63	277333
RP	436888	22	537766	389809	219	767000
RNA	89678	7	858866	88367	222	802533
RF	75316	50	757900	72594	55	824633
GBT	96594	19	574600	69449	99	260433
SVM	117928	4	154266	79471	988	606233
LHD 40x						
GP	104303	34	786933	95663	55	270833
RP	424817	18	790833	403964	201	514800
RNA	90056	7	448066	83762	631	568100
RF	71775	90	761366	71042	106	790400
GBT	91034	29	579033	70640	97	697666
SVM	112426	4	224033	84717	702	730166
LHD 60x						
GP	89732	37	506113	72653	309	300733
RP	431107	19	502666	383796	101	438100
RNA	58179	8	791266	55629	1090	776966
RF	55224	101	844133	54568	115	879200
GBT	83793	36	640033	55560	223	35366
SVM	114368	4	270833	67652	1058	795166
LHD 80x						
GP	88492	42	693766	77472	300	270400
RP	442754	48	755300	393069	122	744466
RNA	63555	8	762666	54423	3246	743166

RF	58934	95	832433	58300	117	859300
GBT	85323	45	519566	63245	94	615333
SVM	111130	4	334533	73321	1488	711100
LHD 100x						
GP	92480	48	747933	86041	128	219266
RP	442754	48	755300	401381	1112	663433
RNA	55241	9	350133	48559	3476	439833
RF	56210	120	643066	55367	129	815100
GBT	84065	33	661700	61741	65	422933
SVM	113612	4	301166	74411	1867	633533
Aleatório 100x						
GP	105320	47	761366	106214	292	478833
RP	378738	35	501366	344078	266	564200
RNA	78123	8	673833	66699	6264	861466
RF	70044	194	793433	68573	208	801666
GBT	96370	26	664733	81099	98	676000
SVM	118570	3	297700	82749	1698	737533

De acordo com a Tabela 5.14, o melhor resultado em termos de FO foi encontrado pelo RF otimizado no LHD 60x (879.200 peças), seguido pela RNA otimizada no Aleatório 100x (861.466 peças), RF otimizada no LHD 80x (859.300 peças) e RNA com hiper-parâmetros *default* no LHD 20x (858.866 peças). Segundo o critério de menor RMSE, como pode ser observado na Figura 5.6, o arranjo com menor erro foi o LHD 60x, sendo este o escolhido para a continuidade do estudo.

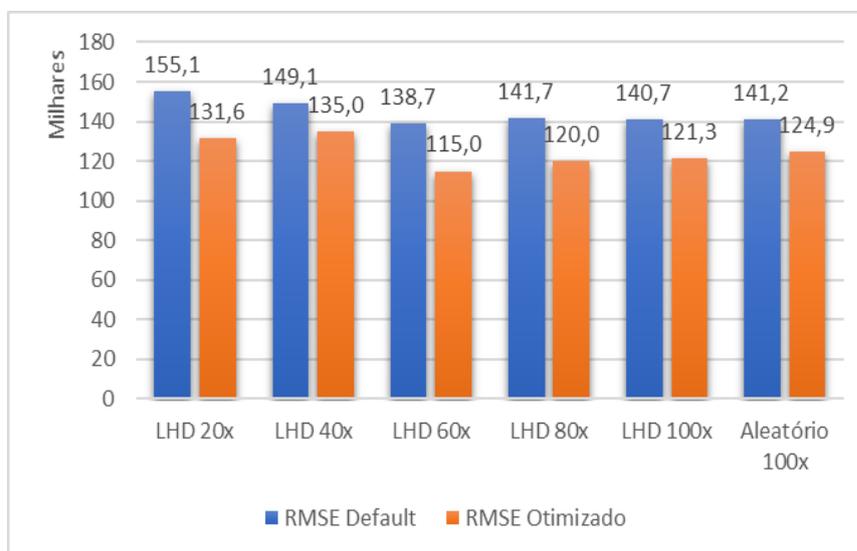


Figura 5.6 - Comparativo do RMSE médio entre arranjos para o OE3

Como já observado nos casos anteriores, o tamanho amostral não possui uma relação direta com a qualidade do metamodelo, uma vez que os arranjos LHD 80x, LHD 100x e Aleatório 100x possuíram métricas de erro piores que LHD 60x. Outro ponto a ser destacado é que, embora menos expressivo que nos demais OEs, o uso da otimização de hiper-parâmetros

permitiu uma redução de 13,7% no RMSE, comprovando a importância desta etapa em um projeto de metamodelagem. A Tabela 5.15 apresenta as métricas dos metamodelos no arranjo LHD 60x.

Tabela 5.15 - Metamodelos aplicados ao OE3

Metamodelo	RMSE	MAE	MAPE	R ²	T – Opt.(s)	Lucro Observado (mil \$)	Hiper-parâmetros
RF	54568	40787	7,68%	0,79	115	879200	$t = 33; v = 3$
SVM	67652	52480	9,87%	0,64	1058	795166	$C = 235356; e = 3061; \gamma = 0,06$
RNA	55629	40983	7,61%	0,77	1090	770900	$\alpha = 0,07; E = 830; n = 5$
RP	383796	358242	67,15%	0,22	101	438100	$D = 5^\circ$
GBT	55560	40388	7,49%	0,75	223	353166	$t = 63; \alpha = 0,28; l = 0,29$
GP	72653	60327	11,57%	0,59	309	300733	$\theta = 15,07; e = 1 \times 10^{-5}; G = 3,78; mbv = 42$

A Tabela 5.15 foi ordenada conforme maior lucro, e neste contexto, o metamodelo que obteve o melhor desempenho foi o RF, com produção de 879.200 peças, apresentando também o menor RMSE e maior R². Considerando a métrica MAE e MAPE, o GBT foi o que se mostrou mais preciso, porém não esboçou um bom resultado da FO, com produção de 353.166 peças.

Embora o R² não tenha atendido as expectativas e valores alvo propostos pela literatura, a análise conjunta das métricas de erro podem ser consideradas aceitáveis, tendo em vista a complexidade do problema e a discrepância entre o tamanho amostral dos dados de treino e o tamanho total do espaço de soluções.

A fim de comparar a solução encontrada pelo RF com o *benchmark*, os dados das replicações para ambas as soluções não puderam ser considerados normais pelo teste de Anderson- Darling ($p\text{-value} < 0,005$ para ambas as amostras). Neste sentido foi conduzido o teste de hipótese não-paramétrico de Mann-Whitney unilateral, que atestou, com 95% de confiança, que a solução encontrada pelo RF é, na mediana, estatisticamente superior a solução *benchmark* encontrada pelo *SimRunner* ($p\text{-value} = 0,000$).

A Tabela 5.16 apresenta o comparativo dos metamodelos quanto a produção prevista por estes para a melhor solução, o lucro observado pela avaliação desta no modelo de simulação, e o tempo dispendido na otimização e treinamento.

Tabela 5.16 - Comparativo dos metamodelos para o OE3

Metamodelo	Solução Final			Tempo de Treino e Otimização (s)		
	Solução ($x_1; x_2; x_3; x_4; x_5; x_6;$ $x_7; x_8; x_9; x_{10}; x_{11}$)	Prod. Prevista (mil)	Prod. Observada (mil)	Rank	Valor	Rank
RF	(3; 4; 6; 1; 3; 2; 1; 3; 4; 1; 1)	813642	879200	1,00	63.475	0,49
SVM	(4; 3; 6; 1; 2; 2; 2; 3; 4; 1; 1)	834192	795166	0,90	64.418	0,48
RNA	(5; 4; 6; 1; 1; 1; 1; 1; 4; 4; 2)	767715	770900	0,88	64.450	0,48
RP	(1; 1; 6; 3; 3; 1; 1; 4; 1; 1; 1)	922811	438100	0,50	63.461	0,49
GBT	(5; 2; 6; 1; 3; 3; 2; 4; 4; 1; 3)	855846	353166	0,40	63.583	0,49
GP	(5; 3; 6; 1; 1; 1; 2; 4; 4; 1; 3)	813627	300733	0,34	63.669	0,49
Referência 01: <i>SimRunner</i>	(1; 3; 5; 3; 3; 1; 2; 3; 4; 1; 4)	-	859.300	0,98	31.200	1,00

Comparando o metamodelo de melhor desempenho com o *benchmark*, obteve-se uma produção adicional de 19.900 peças, todavia demandando 103,40% mais tempo computacional. A Figura 5.7 apresenta o comparativo entre os metamodelos.

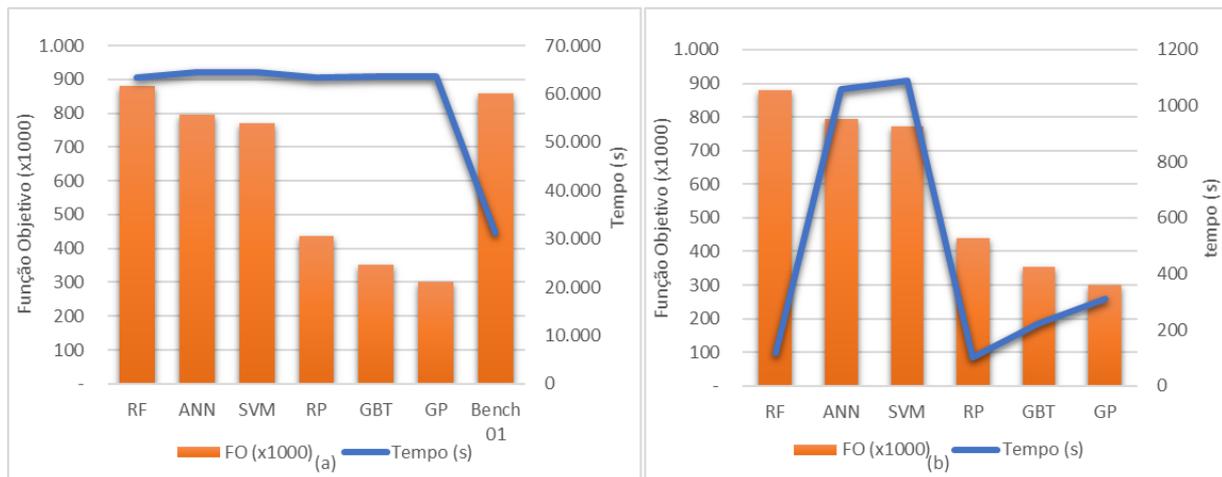
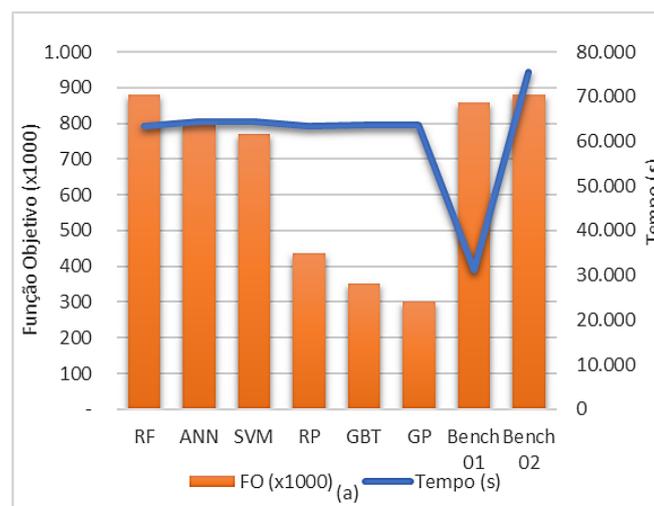


Figura 5.7 - (a) Comparativo de FO e tempo para otimização. (b) Comparativo de FO e tempo para treinar e otimizar os metamodelos para o OE3

Na expectativa de se atingir a mesma FO encontrada pelo metamodelo e apenas para efeito comparativo, executou-se outra otimização pelo *Simrunner*, porém agora no modo cauteloso do *software*. Como resultado, o *Simrunner* alcançou como solução uma FO de 880.966 peças produzidas, valor igual estatisticamente ao encontrado com o metamodelo (*p-value* de 0,971 para o teste de Mann-Whitney). Todavia, o *software* demandou 75.552 segundos para realizar a otimização, tempo 16,04% maior que o metamodelo. A Tabela 5.17 e Figura 5.8 apresentam o comparativo dos metamodelos com este novo valor de referência.

Tabela 5.17 - Comparativo dos metamodelos para o OE3 considerando o *SimRunner* no modo Cauteloso

Metamodelo	Solução Final			Tempo de Treino e Otimização (s)		
	Solução ($x_1; x_2; x_3; x_4; x_5; x_6;$ $x_7; x_8; x_9; x_{10}; x_{11}$)	Prod. Prevista (mil)	Prod. Observada (mil)	Rank	Valor	Rank
RF	(3; 4; 6; 1; 3; 2; 1; 3; 4; 1; 1)	813642	879200	1,00	63.475	0,49
SVM	(4; 3; 6; 1; 2; 2; 2; 3; 4; 1; 1)	834192	795166	0,90	64.418	0,48
RNA	(5; 4; 6; 1; 1; 1; 1; 1; 4; 4; 2)	767715	770900	0,88	64.450	0,48
RP	(1; 1; 6; 3; 3; 1; 1; 4; 1; 1; 1)	922811	438100	0,50	63.461	0,49
GBT	(5; 2; 6; 1; 3; 3; 2; 4; 4; 1; 3)	855846	353166	0,40	63.583	0,49
GP	(5; 3; 6; 1; 1; 1; 2; 4; 4; 1; 3)	813627	300733	0,34	63.669	0,49
Referência 01: <i>SimRunner</i>	(1; 3; 5; 3; 3; 1; 2; 3; 4; 1; 4)	-	859300	0,98	31.200	1,00
Referência 02: <i>SimRunner</i>	(5; 4; 5; 1; 2; 1; 1; 2; 4; 2; 1)	-	880966	1,00	75.552	0,41

Figura 5.8 - Comparativo de FO e tempo considerando o *SimRunner* no modo Cauteloso (*bench 02*)

5.4. Análise dos Resultados

Esta seção apresentará e discutirá os principais resultados obtidos no capítulo. O primeiro ponto a ser destacado é que o uso da metamodelagem pode contribuir significativamente no processo de OvS. Ressalta-se que o principal objetivo deste trabalho não foi concorrer em tempo com o *software SimRunner*, uma vez que o metamodelo, quando treinado e validado, pode ser utilizado inúmeras vezes pelo tomador de decisões para performar otimizações em um curto espaço de tempo. Neste sentido o resultado obtido com o *software*

SimRunner foi tomado como uma referência para determinar se o metamodelo obteve, ou não, bons resultados. A Tabela 5.18 resume os principais resultados dos três casos analisados.

Tabela 5.18 - Resultado dos testes estatísticos para os objetos de estudo

OE	Método	Arranjo	Nº de experimentos	FO (mil)	Teste de Anderson-Darling	Teste F	Teste T	Teste Mann-Whitney
1	SVM	LHD20x	140	54,93	0,644	1,000	1,000	-
	Referência	-	245	54,93	0,644			
2	RF	LHD40x	240	317,45	0,391	0,568	0,001	-
	Referência	-	451	330,12	0,167			
3	RF	LHD60x	660	879,20	<0,005	-	-	0,000
	Referência 1	-	325	859,30	<0,005	-	-	-
	Referência 2	-	787	880,97	<0,005	-	-	0,971

De acordo com a Tabela 5.18, o primeiro OE foi capaz de atingir o valor referência demandando 42,85% menos experimentos. Já o OE2 não foi capaz de atingir a FO referência, obtendo um lucro 3,83% menor, porém demandando 46,78% menos experimentos. Por outro lado, o OE3 superou a FO referência, produzindo 19.900 peças a mais, porém demandando mais que o dobro de experimentos. Para este último caso, vale ressaltar que, devido o metamodelo ter alcançado um valor acima do encontrado pelo *SimRunner*, este foi executado novamente, agora no modo cauteloso, e encontrou uma solução com FO estatisticamente igual ao encontrado pelo metamodelo, porém demandando 16,04% mais tempo.

Outro ponto a ser ressaltado é que no segundo e terceiro OE, o metamodelo que melhor performou foi o RF, obtendo também bons resultados para o OE1 nos arranjos LHD 40x, 60x, 80x, 100x e aleatório 100x. Estes resultados apontam o RF como o algoritmo mais adaptável perante as diferentes características dos objetos de estudo. Os resultados de FO e RMSE (média dos três OEs padronizados entre 0 e 1) dos metamodelos obtidos perante cada arranjo estudado estão apresentados nas Tabelas 5.19 e 5.21 (para metamodelos com hiper-parâmetros *default*) e Tabelas 5.20 e 5.22 (para metamodelos com a otimização de hiper-parâmetros).

Tabela 5.19 - FO média: metamodelo vs. arranjo com hiper-parâmetros *default*

Arranjo	GP	RP	RNA	RF	GBT	SVM	Média Arranjo
LHD 20x	0,892	0,657	0,847	0,921	0,795	0,596	0,785
LHD 40x	0,933	0,775	0,762	0,944	0,860	0,538	0,802
LHD 60x	0,703	0,350	0,878	0,899	0,738	0,546	0,686
LHD 80x	0,763	0,527	0,847	0,894	0,752	0,601	0,731
LHD 100x	0,929	0,612	0,726	0,902	0,826	0,619	0,769
Aleatório 100x	0,898	0,587	0,865	0,949	0,828	0,566	0,782
Média Metamodelo	0,853	0,585	0,821	0,918	0,800	0,578	

Tabela 5.20 - FO média: metamodelo vs. arranjo com hiper-parâmetros otimizados

Arranjo	GP	RP	RNA	RF	GBT	SVM	Média Arranjo
LHD 20x	0,689	0,754	0,903	0,918	0,687	0,869	0,803
LHD 40x	0,691	0,656	0,835	0,956	0,899	0,931	0,828
LHD 60x	0,626	0,479	0,745	0,876	0,534	0,828	0,681
LHD 80x	0,619	0,590	0,837	0,855	0,738	0,781	0,737
LHD 100x	0,688	0,667	0,791	0,964	0,798	0,891	0,800
Aleatório 100x	0,774	0,635	0,971	0,959	0,909	0,806	0,842
Média Metamodelo	0,681	0,630	0,847	0,921	0,761	0,851	

Tabela 5.21 - RMSE média: metamodelo vs. arranjo com hiper-parâmetros *default*

Arranjo	GP	RP	RNA	RF	GBT	SVM	Média Arranjo
LHD 20x	0,352	0,817	0,116	0,092	0,153	0,184	0,286
LHD 40x	0,271	0,710	0,106	0,084	0,152	0,187	0,252
LHD 60x	0,172	0,621	0,139	0,128	0,163	0,185	0,235
LHD 80x	0,165	0,655	0,135	0,125	0,159	0,179	0,236
LHD 100x	0,218	0,594	0,076	0,063	0,140	0,182	0,212
Aleatório 100x	0,248	0,601	0,085	0,072	0,143	0,172	0,220
Média Metamodelo	0,238	0,666	0,110	0,094	0,152	0,182	

Tabela 5.22 - RMSE média: metamodelo vs. arranjo com hiper-parâmetros otimizados

Arranjo	GP	RP	RNA	RF	GBT	SVM	Média Arranjo
LHD 20x	0,116	0,465	0,081	0,088	0,069	0,092	0,152
LHD 40x	0,103	0,467	0,066	0,080	0,061	0,090	0,144
LHD 60x	0,141	0,515	0,132	0,125	0,124	0,130	0,195
LHD 80x	0,143	0,509	0,124	0,124	0,125	0,129	0,192
LHD 100x	0,093	0,482	0,043	0,063	0,052	0,380	0,186
Aleatório 100x	0,105	0,456	0,052	0,070	0,061	0,084	0,138
Média Metamodelo	0,117	0,482	0,083	0,092	0,082	0,151	

Ao analisar as Tabelas 5.19 e 5.20, cujos dados representam a média da FO dos metamodelos dos três casos estudados, percebe-se que dentre os metamodelos treinados com hiper-parâmetros *default* o RF, RNA e GP foram os que atingiram os melhores valores médios de FO, sendo o SVM o de pior desempenho. Todavia quando se trata dos metamodelos com hiper-parâmetros otimizados, o SVM torna-se o segundo melhor, ficando claro o impacto desta etapa no desempenho deste algoritmo. Vale ressaltar que o RF possuiu o melhor desempenho nas duas situações analisadas e maior flexibilidade entre os arranjos experimentais utilizados.

Ao comparar o RMSE médio dos metamodelos apresentados nas Tabelas 5.21 e 5.22, é possível observar que os metamodelos baseados em RF, GBT e RNA foram os de melhores resultados, ou seja, aqueles que melhor representaram os modelos de simulação estudados. Destaca-se que os algoritmos RP apresentaram, em todos os arranjos, os piores resultados em

termos de FO e RMSE. Em contrapartida, o RF, RNA, e GBT apresentaram, em média, bons resultados.

Um ponto a ser destacado é que, considerando o RMSE dos metamodelos com hiper-parâmetros otimizados, houve uma boa performance nos arranjos LHD 20x, LHD 40x, LHD 100x e aleatório 100x, sendo o LHD 60x e 80x os arranjos de piores desempenhos médios. Já quando se observa os metamodelos com hiper-parâmetros *default*, as melhores métricas foram encontradas nos experimentos dos arranjos LHD 100x e Aleatório 100x. As Figuras 5.9 e 5.10 apresentam o comparativo entre os arranjos para o FO e RMSE.

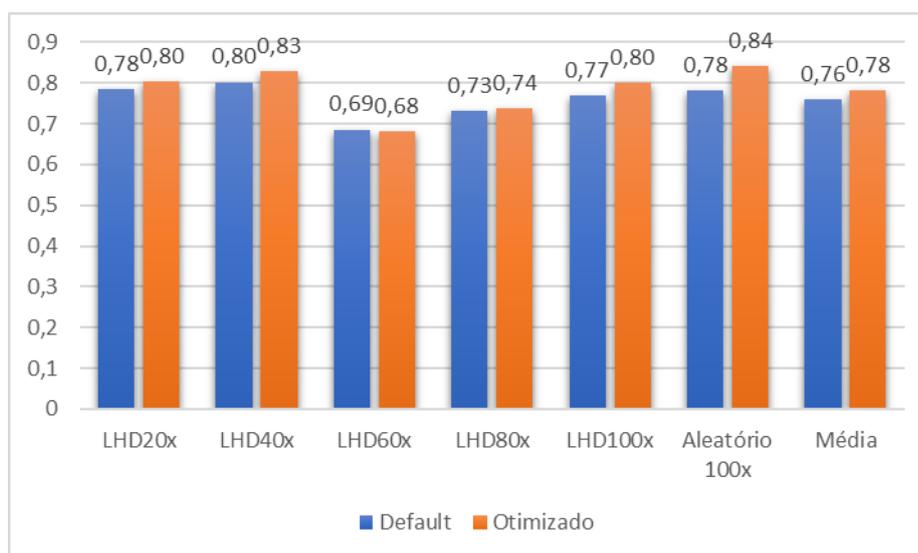


Figura 5.9 – Comparativo do FO médio entre os arranjos

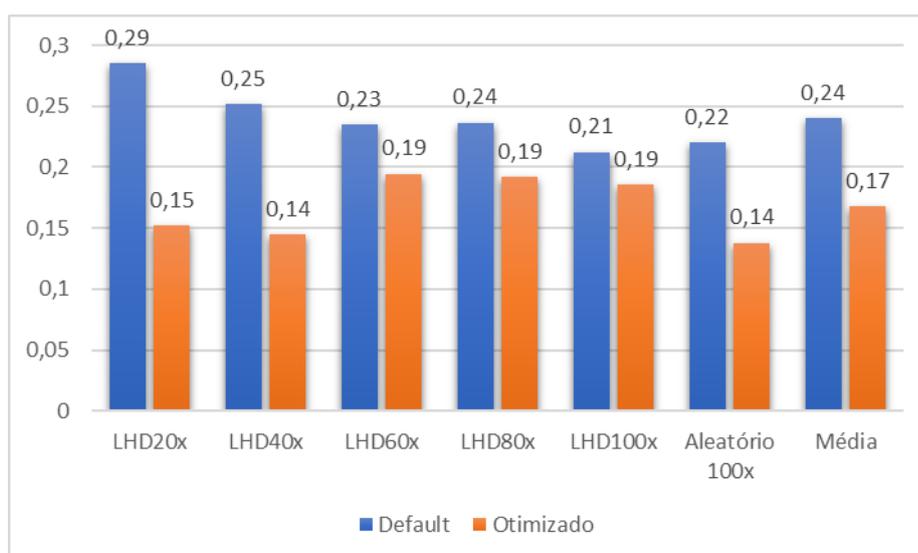


Figura 5.10 – Comparativo do RMSE médio entre os arranjos

Como apresentado nas Figuras 5.9 e 5.10, o arranjo Aleatório 100x foi o que esboçou o melhor resultado em termos de FO e RMSE, todavia equiparou-se estatisticamente ao LHD 40x

(p -value = 0,887 para o teste de Mann-Whitney), que utiliza 60% menos experimentos. Não obstante, comparou-se o arranjo Aleatório 100x com o LHD 100x (arranjo de mesmo tamanho amostral), sendo possível inferir que não houve diferença estatisticamente significativa quando se tratou do valor da mediana da FO (p -value de 0,937 para hiper-parâmetros *default* e 0,788 para hiper-parâmetros otimizados para o teste de Mann-Whitney). Em respeito ao RMSE, embora possa ser observado uma diferença na Figura 5.9, não foi encontrado uma diferença estatisticamente significativa do RMSE obtido com o arranjo LHD 100x e o Aleatório 100x (p -value de 0,812 para hiper-parâmetros *default* e 0,887 para hiper-parâmetros otimizados para o teste de Mann-Whitney). Portanto, para os casos analisados, não se encontrou evidências que atestem a superioridade ou inferioridade do LHD frente ao arranjo aleatório, todavia observou-se que o LHD 40x atingiu resultados similares ao aleatório, mesmo executando 60% menos experimentos, podendo ser um indicativo que LHD consegue amostrar satisfatoriamente o espaço de busca, mesmo com poucos experimentos.

Outro ponto a ser destacado é que cada problema apresenta peculiaridades que podem demandar diferentes tamanhos amostrais na construção de metamodelos, não sendo possível determinar um tamanho amostral que sirva de referência para projetos desta natureza. Todavia, embora com base em apenas três casos, foi possível estabelecer uma correlação entre o tamanho do espaço de soluções dos problemas e o tamanho N do arranjo LHD. O modelo de regressão linear para esta relação obteve um R^2 de 0,982 e é apresentado na Eq. (34).

$$N = (19,675 + 1,728 \times 10^{-5}D)x \quad (34)$$

Sendo N o tamanho da amostra, D a tamanho do espaço de soluções e x o número de variáveis de decisão do problema.

Quanto aos resultados da otimização de hiper-parâmetros, a Figura 5.11 apresenta a redução média no RMSE proporcionada pelo uso desta técnica em cada metamodelo e a Figura 5.12 o comparativo de RMSE e FO para metamodelos com hiper-parâmetro *default* e otimizados.

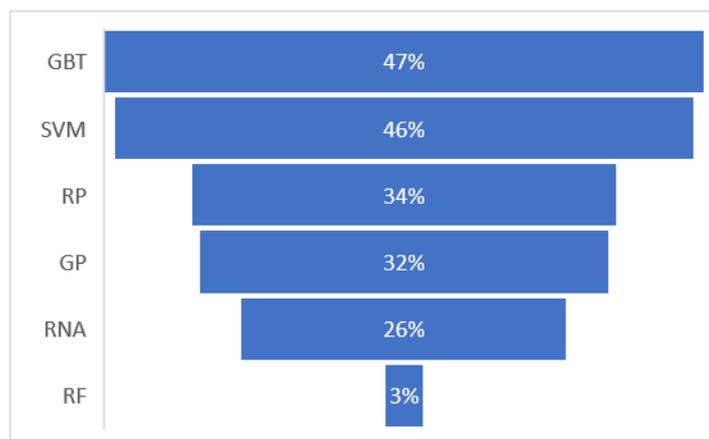


Figura 5.11 – Redução no RMSE com o uso da otimização de hiper-parâmetros

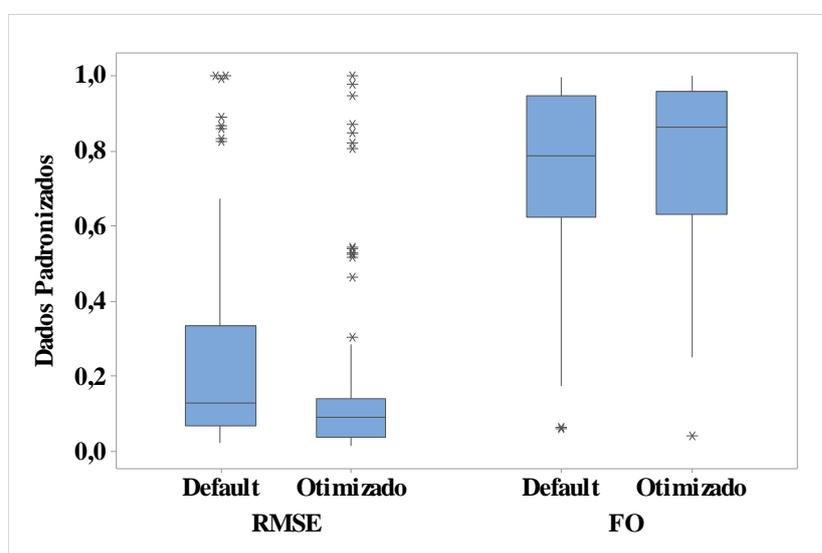


Figura 5.12 – Comparativo do RMSE e FO entre os metamodelos com hiper-parâmetros *default* e otimizados

Para os problemas estudados, esta etapa permitiu uma redução média de 31,28% no RMSE dos metamodelos, reforçando a importância da escolha adequada dos hiper-parâmetros. Dentre os metamodelos, o GBT e SVM foram os que esboçaram maior redução do RMSE, enquanto o RF apresentou uma redução de apenas 3% com o uso da otimização. Embora em termos de FO não tenha sido encontradas melhorias significativas com o uso da otimização ($p\text{-value} = 0,265$ para o teste de Mann-Whitney), a mediana do RMSE dos metamodelos otimizados foi estatisticamente menor que o RMSE dos metamodelos com hiper-parâmetros *default* ($p\text{-value} = 0,000$ para o teste de Mann-Whitney), ressaltando que esta técnica permite a criação de metamodelos mais precisos, e conseqüentemente, mais confiáveis. Vale ressaltar que o teste não paramétrico de Mann-Whitney foi escolhido devido as amostras não serem consideradas normais pelo teste de Anderson-Darling (ambas com $p\text{-value} < 0,005$) e as

amostras foram compostas pelos valores padronizados (entre 0 e 1) de todos metamodelos, arranjos e OEs.

A aplicação do *framework* proposto visou responder três questões de pesquisa principais, a saber: (i) Qual é o ganho de eficiência promovido pelo uso do arranjo LHD frente ao aleatório? (ii) Qual o ganho de performance e consequente custo computacional promovido pelo uso da otimização de hiper-parâmetros por AG? (iii) O uso da metamodelagem permite ganhos de tempo computacional e/ou qualidade da resposta frente a otimização pelo software comercial *SimRunner*?

Em resposta a primeira pergunta e ao contrário do que sugeria em parte a literatura, para os casos analisados não se encontrou evidências que comprovassem a superioridade ou inferioridade do LHD sobre o arranjo aleatório, uma vez que, em média, o aleatório se igualou estatisticamente em RMSE e FO ao arranjo LHD de mesmo número de experimentos (100x). Todavia, ao analisar separadamente os OEs, observa-se que o arranjo aleatório obteve resultados piores ou equiparáveis a arranjos LHD de menor tamanho, como no OE1, no qual se equiparou ao LHD 20x, arranjo cinco vezes menor, e na média dos três casos se equiparou ao LHD 40x, arranjo 60% menor.

Quanto a segunda pergunta de pesquisa, pode-se afirmar que o uso da otimização de hiper-parâmetros permitiu a criação de metamodelos mais precisos, uma vez que, em média, obteve-se uma redução de 31,28% no RMSE. Além disto, o tempo computacional demandado nesta etapa foi, em média, de 624,37 segundos, sendo considerado um tempo baixo se comparado ao tempo total de execução dos experimentos na simulação.

A respeito da terceira pergunta, dentre os três problemas estudados, o primeiro atingiu o mesmo valor estatístico da solução *benchmark*, o segundo obteve um resultado de FO apenas 3,83 % inferior, e o terceiro superou o *benchmark* em 2,32%. Quanto ao tempo computacional demandado, o OE1 atingiu o *benchmark* com 42,85% menos tempo e o OE2, embora não tenha conseguido atingir *benchmark*, demandou um tempo 46,78% menor. Já no OE3, o melhor valor de FO foi atingido pelo RF no arranjo LHD 60x, que superou o *benchmark* em 19.900 peças, porém demandando 103,4% mais tempo computacional. Vale ressaltar que, para efeito comparativo, para o *SimRunner* atingir a mesma FO encontrada pelo metamodelo foram necessários 787 experimentos, levando um tempo 16,04% maior em relação ao RF no arranjo LHD 60x.

6. CONCLUSÕES

Este capítulo apresenta a fase de implantação do método de Mitroff *et al.* (1974), visando concluir sobre a validade do *framework* proposto para problemas de OvS. Nas últimas décadas, o uso das técnicas de OvS ganhou forte popularidade entre práticos e acadêmicos, no entanto ainda enfrenta desafios, principalmente quanto ao tempo computacional demandado neste tipo de solução. O avanço das pesquisas na área de ML, com desenvolvimento e aperfeiçoamento de algoritmos com alta capacidade de aprendizado, tornou o uso da metamodelagem um campo de pesquisa promissor. Todavia, a investigação destes algoritmos de ML para metamodelagem em problemas de OvS ainda representa uma lacuna na literatura.

Vale ressaltar a importância do uso do método de RSL para a condução da análise da literatura existente, que permitiu delinear com mais efetividade o *framework* proposto, premissas e parâmetros utilizados nesta dissertação. Por fim, com base nos resultados obtidos com a aplicação do método, referencial teórico apresentado e *insights* retirados da RSL é possível inferir que o papel da metamodelagem está na redução do tempo computacional dispendido em problemas de OvS, para os quais modelos recursivos seriam custosos, ou até inviáveis, de se executar. Este papel é evidenciado pelos anseios da indústria 4.0, por ferramentas de simulação para apoio e/ou tomada de decisões em tempo real, como é o caso dos gêmeos digitais, nos quais a otimização por métodos recursivos inviabilizaria seu uso, sendo necessário a utilização de metamodelos para geração de bons resultados em tempo hábil.

O *framework* de otimização da simulação proposto neste trabalho incorpora o uso de algoritmos de ML e otimização de hiper-parâmetros via AG para metamodelagem em problemas reais de SED. Com base nos resultados apresentados no Capítulo 5, existem fortes evidências que o *framework* proposto é uma ferramenta de suporte a decisão eficiente em aplicações de OvS. Destaca-se que, embora o *framework* proposto tenha obtido bons resultados frente ao *benchmark*, o objetivo deste trabalho não foi confrontar os resultados da metamodelagem com os do *SimRunner*, apenas tomou-se este como um valor de referência para o estudo devido a inviabilidade de se simular todo o espaço de solução dos problemas. Ressalta-se ainda que, diferentemente do *SimRunner*, o metamodelo, uma vez que treinado e validado, permite ao decisor performar otimizações quantas vezes forem necessárias, no contexto operacional e estratégico, demandado apenas poucos segundos para obtenção de um bom resultado.

Quanto a performance individual dos metamodelos, os que exibiram os melhores resultados foram o SVM para o OE1 e o RF para o OE2 e OE3, indicando que embora não

exista uma hegemonia entre os algoritmos, o RF apresenta uma boa flexibilidade e resultados entre diferentes problemas. Um ponto a ser destacado nas experimentações foi que os metamodelos construídos com o algoritmo GP (ou *kriging*) e RP, alguns dos mais citados na literatura, obtiveram as piores performances em todos os casos analisados. Isto pode-se dever ao fato de que o GP desenvolvido para problemas com poucas dimensões (KLEIJNEN, 2017b), perdendo sua efetividade a medida que o problema tem suas dimensões aumentadas. Além disto, Barton (2009) enfatiza que o RP é eficaz para estratégias de modelagem local, não sendo indicado para criação de metamodelos globais.

Tendo em vista os pontos destacados acima e os resultados obtidos no Capítulo 5, o *framework* proposto obteve um desempenho satisfatório, ao passo atingiu ou se aproximou dos valores ótimos referência com uma redução média de 35,22% no tempo computacional. Embora não possa ser estabelecida uma hegemonia entre os metamodelos e arranjos experimentais, para novos projetos de OvSM recomenda-se o uso, em primeira instância, do algoritmo RF devido a sua eficiência, tanto em tempo quanto em ajuste e FO, além de uma boa flexibilidade perante diferentes problemas. Quanto ao arranjo experimental, recomenda-se inicialmente uso do LHD 40x, aumentando o tamanho amostral até que seja atingido um ajuste aceitável. Outra recomendação é o uso da etapa de otimização de hiper-parâmetros, uma vez que o ganho de qualidade do metamodelo justifica o tempo demandado nesta etapa, que no geral, representa uma parcela pequena, ou até irrisória, do tempo total de um projeto de OvSM.

6.1. Verificação dos Objetivos Específicos

Esta Seção apresenta a verificação dos objetivos específicos propostos no Capítulo 1:

- 1) Propor um *framework* para projetos de OvSM que integre SED, ML, e otimização de hiper-parâmetros.

Este objetivo começou a ser explorado a partir do levantamento do referencial teórico apresentado no Capítulo 3 e da RSL no Capítulo 3, que promoveram o suporte teórico necessário para o desenvolvimento do *framework* proposto. Por fim, o Capítulo 4 apresenta premissas assumidas, técnicas empregadas e a estrutura do *framework* proposto. Mais detalhes sobre as etapas do método podem ser verificados no Capítulo 4.

- 2) Desenvolver, treinar, otimizar e validar metamodelos para cada um dos objetos de estudo.

Para cada OE foram construídos seis metamodelos baseados em diferentes algoritmos de aprendizado supervisionado, a saber: GP, RP, GBT, RF, SVM e RNA, para cada um dos seis arranjos experimentais estudados, totalizando 36 combinações avaliadas em cada OE. Para fim

de comparar a importância da etapa de otimização de hiper-parâmetros, os mesmos metamodelos também foram construídos sem esta etapa. Os passos da construção, treinamento, otimização e validação dos metamodelos podem ser observados no decorrer do Capítulo 5.

- 3) Aplicar o *framework* proposto na otimização de três modelos de SED.

O método desenvolvido foi aplicado para construção de metamodelos em três modelos de SED, os quais representavam problemas reais de alocação de recursos industriais. O primeiro caso tratou de uma célula produtiva do setor de telecomunicações cujo objetivo era a maximização do lucro por meio do dimensionamento do número de funcionários, bancadas de trabalho e tamanho de estoques intermediários, totalizando 127.000 soluções possíveis.

Também com o objetivo de maximização do lucro, o segundo OE visou dimensionar a quantidade de operadores e equipamentos em uma célula de controle de qualidade de uma empresa produtora de transmissores, tendo um espaço de soluções com 1.000.000 de possibilidades. Já o terceiro caso possui 2.488.320 cenários possíveis e visou a maximização da quantidade de peças produzidas em uma célula de produção de autopeças. Os detalhes e resultados envolvendo este objetivo podem ser observados no Capítulo 5.

- 4) Avaliar os resultados obtidos e comparar os resultados da otimização via simulação por metaheurística com os resultados da otimização via metamodelagem.

A fim de cumprir com este objetivo, o Capítulo 5, em especial a Seção 5.4, apresentou comparativos entre o *framework* proposto neste trabalho e o *framework* sem a etapa de otimização de hiper-parâmetros. Além disto, para eficácia dos metamodelos em atingir bons resultados de FO, os modelos de SED foram submetidos à otimização via o *software* comercial *SimRunner*, e seus resultados foram comparados aos obtidos pelos metamodelos em termos de FO e tempo computacional.

6.2. Sugestões para Futuras Pesquisas

De acordo com as respostas encontradas para as questões de pesquisa na RSL e o entendimento adquirido durante este estudo, sugere-se a futuras pesquisas:

Uma oportunidade encontrada na literatura é a respeito dos algoritmos utilizados na metamodelagem. Seria interessante que futuras pesquisas se desenvolvessem em paralelo com a área de ML, explorando algoritmos com alta capacidade de aprendizado, como o *Deep Learning*, para criar metamodelos mais precisos em problemas de OvS. Além disto, o uso de algoritmos como *Extreme Learning Machines*, SVM, RNAs e variantes de *Kriging* apresentam oportunidades de aprimoramentos e novas aplicações.

Uma outra oportunidade a ser explorada em futuras pesquisas é o estudo aprofundado do tamanho do espaço amostral a ser utilizado em problemas de metamodelagem. Este estudo pode ser conduzido pela aplicação do *framework* proposto nesta dissertação em uma gama maior de OEs, de modo a buscar identificar a correlação entre o tamanho amostral e as características do problema, tais como dimensão, espaço de soluções, variabilidade, entre outras.

Sugere-se também realizar uma comparação sistemática entre os arranjos LHD e aleatório, avaliando suas performances em diferentes tamanhos amostrais. Além disto, seria interessante a expansão desta comparação para outras estratégias de amostragem, tais como *factorial design*, *CCD*, *orthogonal arrays*, entre outros.

A utilização de estratégias adaptativas, como o EGO, apresentou resultados consistentes em diversas aplicações, porém ainda apresenta oportunidades de novos estudos. Os estudos envolvendo este tipo de estratégia utilizaram o *Kriging* como algoritmo base de metamodelagem, que apesar das vantagens de seu uso existem outros algoritmos que, no geral, possuem melhor performance, sendo um ponto a ser explorado em futuras pesquisas. Além disto, a maior parte dos estudos utilizam o algoritmo EI para delinear quais novos pontos serão inseridos a cada interação do EGO, sendo interessante estudar alternativas ou aprimoramentos do código de modo a incrementar a eficiência desta estratégia.

A combinação de técnicas pré-existentes na formulação de uma solução inovadora é uma prática bem vista por acadêmicos e práticos. Neste sentido sugere-se como futuros estudos a integração de ferramentas de *feature selection* e *generation* nos projetos de OvSM, de modo a treinar os metamodelos com base em variáveis relevantes e não-redundantes. Poucos artigos utilizam essa abordagem, consistindo em uma lacuna na literatura.

Na nova era da indústria e serviços, também existem oportunidades promissoras ao uso de OvSM. Uma possibilidade é uma maior exploração de metamodelos na construção de gêmeos digitais, pois uma vez treinado e validado o metamodelo é capaz de promover otimizações em poucos segundos. Além disto, a integração das técnicas de OvSM com o conceito incipiente de Big Data DEA, *e.g.*, (ZELENYUK, 2020), pode ser um caminho interessante e inovador para futuras pesquisas.

Os pontos destacados derivam apenas de um conjunto de *insights* observados pelo autor no decorrer desta pesquisa em meio a uma vasta gama de oportunidades e desafios presentes na literatura em OvSM. Doravante, neste trabalho foram discutidos três problemas de dimensionamento de recursos no setor industrial, logo novos casos poderão ser estudados de modo a validar e expandir a aplicação do *framework* proposto para outras classes de problemas.

REFERÊNCIAS BIBLIOGRÁFICAS

AGHAEI CHADEGANI, Arezoo *et al.* A comparison between two main academic literature collections: Web of science and scopus databases. **Asian Social Science**, v. 9, n. 5, p. 18–26, 2013.

ALABAS, C.; ALTIPARMAK, F.; DENGIZ, B. A comparison of the performance of artificial intelligence techniques for optimizing the number of kanbans. **Journal of the Operational Research Society**, v. 53, n. 8, p. 907–914, 2002.

ALBANESI, Alejandro *et al.* A metamodel-based optimization approach to reduce the weight of composite laminated wind turbine blades. **Composite Structures**, v. 194, n. April, p. 345–356, 2018. Disponível em: <<https://doi.org/10.1016/j.compstruct.2018.04.015>>.

ALI, Wahid *et al.* Surrogate-assisted modeling and optimization of a natural-gas liquefaction plant. **Computers and Chemical Engineering**, v. 118, p. 132–142, 2018. Disponível em: <<https://doi.org/10.1016/j.compchemeng.2018.08.003>>.

ALIZADEH, Mohammad Reza; NIKOO, Mohammad Reza; RAKHSHANDEHROO, Gholam Reza. Developing a Multi-Objective Conflict-Resolution Model for Optimal Groundwater Management Based on Fallback Bargaining Models and Social Choice Rules: a Case Study. **Water Resources Management**, v. 31, n. 5, p. 1457–1472, 2017.

AMARAN, Satyajith *et al.* Simulation optimization: a review of algorithms and applications. **Annals of Operations Research**, v. 240, n. 1, p. 351–380, 2016.

AMIRI, Farbod; SHIRAZI, Babak; TAJDIN, Ali. Multi-objective simulation optimization for uncertain resource assignment and job sequence in automated flexible job shop. **Applied Soft Computing Journal**, v. 75, p. 190–202, 2019. Disponível em: <<https://doi.org/10.1016/j.asoc.2018.11.015>>.

AOUADNI, Sourour; AOUADNI, Ismahene; REBAÏ, Abdelwaheb. A systematic review on supplier selection and order allocation problems. **Journal of Industrial Engineering International**, v. 15, n. 2009, p. 267–289, 2019. Disponível em: <<https://doi.org/10.1007/s40092-019-00334-y>>.

ARTHUR, Samuel. Some Studies in Machine Learning Using the Game of Checkers. **IBM Journal of Research and Development**, v. 3, n. 3, p. 210, 1959.

ATAIE-ASHTIANI, Behzad; KETABCHI, Hamed; RAJABI, Mohammad Mahdi. Optimal management of a freshwater lens in a small Island using surrogate models and evolutionary algorithms. **Journal of Hydrologic Engineering**, v. 19, n. 2, p. 339–354, 2013.

AYDIN, Cem Savas *et al.* Response surface approach to robust design of assembly cells

through simulation. **Assembly Automation**, v. 38, n. 4, p. 450–464, 2018.

AZIZI, Mohammad Javad; SEIFI, Farshad; MOGHADAM, Samira. A robust simulation optimization algorithm using kriging and particle swarm optimization: Application to surgery room optimization. **Communications in Statistics: Simulation and Computation**, v. 0, n. 0, p. 1–17, 2019. Disponível em: <<https://doi.org/10.1080/03610918.2019.1593452>>.

BĂLAN, Carmen. The disruptive impact of future advanced ICTs on maritime transport: a systematic review. **Supply Chain Management**, v. 25, n. 2, p. 157–175, 2018.

BANANMAH, Mohammad *et al.* Optimizing chute-flip bucket system based on meta-modelling approach. **Canadian Journal of Civil Engineering**, v. 47, n. 5, p. 584–595, 2020.

BANKS, Jerry *et al.* **Discrete-Event System Simulation**. 5th. ed. London: Pearson, 2014.

BAQUELA, Enrique Gabriel; OLIVERA, Ana Carolina. A novel hybrid multi-objective metamodel-based evolutionary optimization algorithm. **Operations Research Perspectives**, v. 6, n. April 2018, p. 100098, 2019. Disponível em: <<https://doi.org/10.1016/j.orp.2019.100098>>.

BARTON, Russell R.; MECKESHEIMER, Martin. Chapter 18 Metamodel-Based Simulation Optimization. **Handbooks in Operations Research and Management Science**, v. 13, n. C, p. 535–574, 2006.

BARTON, Russell R. **Simulation Optimization Using Metamodel**. **Winter Simulation Conference**. [S.l: s.n.], 2009.

BARTZ-BEIELSTEIN, Thomas; ZAEFFERER, Martin; PHAM, Quoc Cuong. Optimization via multimodel simulation: A new approach to optimization of cyclone separator geometries. **Structural and Multidisciplinary Optimization**, v. 58, n. 3, p. 919–933, 2018.

BASHIRI, M.; GERANMAYEH, A. Farshbaf. Tuning the parameters of an artificial neural network using central composite design and genetic algorithm. **Scientia Iranica**, v. 18, n. 6, p. 1600–1608, 2011. Disponível em: <<http://dx.doi.org/10.1016/j.scient.2011.08.031>>.

BATUR, Demet; BEKKI, Jennifer M.; CHEN, Xi. Quantile regression metamodeling: Toward improved responsiveness in the high-tech electronics manufacturing industry. **European Journal of Operational Research**, v. 264, n. 1, p. 212–224, 2018.

BELGIN, Onder. Data Envelopment Analysis Based Metamodeling for Multi Objective Simulation Optimization in a Manufacturing Line. **Sigma Journal of Engineering and Natural Sciences**, v. 37, n. 4, p. 1435–1449, 2019.

BERGMEIR, Christoph; BENÍTEZ, José M. On the use of cross-validation for time series predictor evaluation. **Information Sciences**, v. 191, p. 192–213, 2012. Disponível em:

<<http://dx.doi.org/10.1016/j.ins.2011.12.028>>.

BERTRAND, Will M.; FRANSOO, Jan C. Operations management research methodologies using quantitative modeling. **International Journal of Operations and Production Management**, v. 22, n. 2, p. 241–264, 2002.

BIN MAKHASHEN, Yousef *et al.* Exploring the role of ambidexterity and coopetition in designing resilient fashion supply chains: a multi-evidence-based approach. **Journal of Enterprise Information Management**, 2020.

BOMMERT, Andrea *et al.* Benchmark for filter methods for feature selection in high-dimensional classification data. **Computational Statistics and Data Analysis**, v. 143, p. 106839, 2020. Disponível em: <<https://doi.org/10.1016/j.csda.2019.106839>>.

BOOTH, Andrew. Clear and present questions: formulating questions for evidence based practice. **Library Hi Tech**, v. 24, n. 3, p. 355–368, 2006.

BOUKOUVALA, Fani; IERAPETRITOU, Marianthi G. Surrogate-based optimization of expensive flowsheet modeling for continuous pharmaceutical manufacturing. **Journal of Pharmaceutical Innovation**, v. 8, n. 2, p. 131–145, 2013.

BREIMAN, Leo. Random forests. **Machine Learning**, v. 45, p. 5–32, 2001.

CAI, Xiwen *et al.* An efficient surrogate-assisted particle swarm optimization algorithm for high-dimensional expensive problems. **Knowledge-Based Systems**, v. 184, p. 104901, 2019. Disponível em: <<https://doi.org/10.1016/j.knosys.2019.104901>>.

CANDELIERI, A.; PEREGO, R.; ARCHETTI, F. Bayesian optimization of pump operations in water distribution systems. **Journal of Global Optimization**, v. 71, n. 1, p. 213–235, 2018. Disponível em: <<https://doi.org/10.1007/s10898-018-0641-2>>.

CAO, Yuzhou; XU, Yitian; DU, Junling. Multi-variable estimation-based safe screening rule for small sphere and large margin support vector machine. **Knowledge-Based Systems**, v. 191, n. xxxx, p. 105223, 2020. Disponível em: <<https://doi.org/10.1016/j.knosys.2019.105223>>.

CARNEIRO, Vitor Hugo Sousa. Abordagens híbridadas de Machine Learning / Simulação para sistemas logísticos dinâmicos. 2019.

CARSON, Yolanda; MARIA, Anu. Simulation optimization: Methods and applications. **Winter Simulation Conference Proceedings**, p. 118–126, 1997.

CHAI, T.; DRAXLER, R. R. Root mean square error (RMSE) or mean absolute error (MAE)? -Arguments against avoiding RMSE in the literature. **Geoscientific Model Development**, v. 7, n. 3, p. 1247–1250, 2014.

CHANG, Kuo Hao; HSIEH, Liam Y. Determination of Wafer Start Mix in

Semiconductor Manufacturing during New Technology Ramp-Up: Model, Solution Method, and an Empirical Study. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, v. 46, n. 2, p. 294–302, 2016.

CHEN, Mingjie; IZADY, Azizallah; ABDALLA, Osman A. An efficient surrogate-based simulation-optimization method for calibrating a regional MODFLOW model. **Journal of Hydrology**, v. 544, p. 591–603, 2017. Disponível em: <<http://dx.doi.org/10.1016/j.jhydrol.2016.12.011>>.

CHEN, Xiao; OSORIO, Carolina; SANTOS, Bruno Filipe. Simulation-based travel time reliable signal control. **Transportation Science**, v. 53, n. 2, p. 523–544, 2019.

CHEN, Xiqun *et al.* A Bayesian stochastic kriging optimization model dealing with heteroscedastic simulation noise for freeway traffic management. **Transportation Science**, v. 53, n. 2, p. 545–565, 2019.

CHEN, Xiqun *et al.* Social welfare maximization of multimodal transportation: Theory, metamodel, and application to Tianjin Ecocity, China. **Transportation Research Record**, v. 2451, n. 2451, p. 36–49, 2014.

CHEN, Xiqun Michael *et al.* Time-of-day vehicle mileage fees for congestion mitigation and revenue generation: A simulation-based optimization method and its real-world application. **Transportation Research Part C: Emerging Technologies**, v. 63, p. 71–95, 2016.

CHERKASSKY, Vladimir; MA, Yunqian. Practical selection of SVM parameters and noise estimation for SVM regression. **Neural Networks**, v. 17, n. 1, p. 113–126, 2004.

CHONG, Linsen; OSORIO, Carolina. A simulation-based optimization algorithm for dynamic large-scale urban transportation problems. **Transportation Science**, v. 52, n. 3, p. 637–656, 2018.

CHRISTELIS, Vasileios; MANTOGLOU, Aristotelis. Pumping Optimization of Coastal Aquifers Assisted by Adaptive Metamodelling Methods and Radial Basis Functions. **Water Resources Management**, v. 30, n. 15, p. 5845–5859, 2016. Disponível em: <<http://dx.doi.org/10.1007/s11269-016-1337-3>>.

CIUFFO, Biagio; PUNZO, Vincenzo. “No free lunch” Theorems applied to the calibration of traffic simulation models. **IEEE Transactions on Intelligent Transportation Systems**, v. 15, n. 2, p. 553–562, 2014.

COELHO, Guilherme F.; PINTO, Luiz R. Kriging-based simulation optimization: An emergency medical system application. **Journal of the Operational Research Society**, v. 69, n. 12, p. 2006–2020, 2018. Disponível em: <<https://doi.org/10.1080/01605682.2017.1418149>>.

COSTA, Eric; SOARES, António Lucas; DE SOUSA, Jorge Pinho. Industrial business associations improving the internationalisation of SMEs with digital platforms: A design science research approach. **International Journal of Information Management**, v. 53, n. January, p. 102070, 2020. Disponível em: <<https://doi.org/10.1016/j.ijinfomgt.2020.102070>>.

CRISCI, C.; GHATTAS, B.; PERERA, G. A review of supervised machine learning algorithms and their applications to ecological data. **Ecological Modelling**, v. 240, p. 113–122, 2012. Disponível em: <<http://dx.doi.org/10.1016/j.ecolmodel.2012.03.001>>.

CUCKLER, Robert; CHANG, Kuo Hao; HSIEH, Liam Y. Optimal Parallel Machine Allocation Problem in IC Packaging Using IC-PSO: An Empirical Study. **Asia-Pacific Journal of Operational Research**, v. 34, n. 6, p. 1–20, 2017.

CURRIE, Christine S.M. *et al.* How simulation modelling can help reduce the impact of COVID-19. **Journal of Simulation**, v. 14, n. 2, p. 83–97, 2020. Disponível em: <<https://doi.org/10.1080/17477778.2020.1751570>>.

DAI, Bo *et al.* **Statistical model optimized random forest regression model for concrete dam deformation monitoring. Structural Control and Health Monitoring**. [S.l: s.n.], 2018

DANG, Xuan Phuong. General frameworks for optimization of plastic injection molding process parameters. **Simulation Modelling Practice and Theory**, v. 41, p. 15–27, 2014. Disponível em: <<http://dx.doi.org/10.1016/j.simpat.2013.11.003>>.

DE JONG, Kenneth Alan. **Analysis of the Behavior of a Class of Genetic Adaptive Systems**. . [S.l: s.n.], 1975

DE LA FUENTE, Rodrigo; ERAZO, Ignacio; SMITH, Raymond L. **Enabling intelligent processes in simulation utilizing the tensorflow deep learning resources. Winter Simulation Conference**. [S.l.]: IEEE, 2019.

DE LA FUENTE, Rodrigo; SMITH, Raymond. Metamodeling a system dynamics model: a contemporary comparison of methods. 2017, Las Vegas, NY: [s.n.], 2017. p. 1689–1699.

DE SOUSA JUNIOR, W. T. *et al.* Economic lot-size using machine learning, parallelism, metaheuristic and simulation. **International Journal of Simulation Modelling**, v. 18, n. 2, p. 205–216, 2019.

DELLINO, Gabriella; KLEIJNEN, Jack P.C.; MELONI, Carlo. Metamodel-Based Robust Simulation-Optimization: An Overview. **Operations Research/ Computer Science Interfaces Series**. [S.l: s.n.], 2015. v. 59. p. 101–122. Disponível em: <http://www.researchgate.net/publication/261512645_A_review_on_global_sensitivity_analy>

sis_methods>.

DENGIZ, B.; ALABAS-USLU, C.; DENGIZ, O. Optimization of manufacturing systems using a neural network metamodel with a new training approach. **Journal of the Operational Research Society**, v. 60, n. 9, p. 1191–1197, 2009.

DENGIZ, Berna; BEKTAS, Tolga; ULTANIR, A. Eren. Simulation optimization based DSS application: A diamond tool production line in industry. **Simulation Modelling Practice and Theory**, v. 14, n. 3, p. 296–312, 2006.

DENGIZ, Berna; IÇ, Yusuf Tansel; BELGIN, Onder. A meta-model based simulation optimization using hybrid simulation-analytical modeling to increase the productivity in automotive industry. **Mathematics and Computers in Simulation**, v. 120, p. 120–128, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.matcom.2015.07.005>>.

DENYER, D.; TRANFIELD, D. Producing a systematic review. In **D. A. Buchanan & A. Bryman (Eds.), The Sage handbook of organizational research methods**. [S.l.: s.n.], 2009. p. 671–689. Disponível em: <https://www.mendeley.com/catalogue/0ff0533c-01f2-338e-a1a9-6f60a28fa801/?utm_source=desktop&utm_medium=1.19.4&utm_campaign=open_catalog&userDocumentId=%7B8b821bd1-4e6e-4cce-8efd-130b29a0bf5b%7D>.

DENYER, David; TRANFIELD, David; VAN AKEN, Joan Ernst. Developing design propositions through research synthesis. **Organization Studies**, v. 29, n. 3, p. 393–413, 2008.

DHAR, Anirban; DATTA, Bithin. Saltwater intrusion management of coastal aquifers. I: Linked simulation-optimization. **Journal of Hydrologic Engineering**, v. 14, n. 12, p. 1263–1272, 2009.

DUNKE, Fabian; NICKEL, Stefan. Neural networks for the metamodeling of simulation models with online decision making. **Simulation Modelling Practice and Theory**, v. 99, p. 102016, 2020. Disponível em: <<https://doi.org/10.1016/j.simpat.2019.102016>>.

FIGUEIRA, Gonçalo; ALMADA-LOBO, Bernardo. Hybrid simulation-optimization methods: A taxonomy and discussion. **Simulation Modelling Practice and Theory**, v. 46, p. 118–134, 2014.

FRIEDMAN, Jerome H. Stochastic gradient boosting. **Computational Statistics and Data Analysis**, v. 38, n. 4, p. 367–378, 2002.

FU, Michael C. Feature Article: Optimization for simulation: Theory vs. Practice. **INFORMS Journal on Computing**, v. 14, n. 3, p. 192–215, 2002.

FU, Yan; SAHIN, Kemal H. Better Optimization of Nonlinear Uncertain Systems (BONUS) for vehicle structural design. **Annals of Operations Research**, v. 132, n. 1–4, p. 69–

84, 2004.

GABRIEL, Gustavo Teodoro *et al.* Good practices and deficiencies in conceptual modelling: A systematic literature review. **Journal of Simulation**, v. 00, n. 00, p. 1–17, 2020. Disponível em: <<https://doi.org/10.1080/17477778.2020.1764875>>.

GANJISAFFAR, Yasser; CARUANA, Rich; LOPES, Cristina Videira. Bagging gradient-boosted trees for high precision, low variance ranking models. **SIGIR'11 - Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval**, n. c, p. 85–94, 2011.

GÉRON, Aurélien. **Hands-On Machine Learning with Scikit-Learn and TensorFlow**. [S.l.]: O'Reilly Media, 2017.

GHORBANI MOOSELU, Mehrdad *et al.* Fuzzy Multi-Objective Simulation-Optimization of Stepped Spillways Considering Flood Uncertainty. **Water Resources Management**, v. 33, n. 7, p. 2261–2275, 2019.

GIL, Antonio Carlos. **Como Elaborar Projetos de Pesquisa**. 4. ed. São Paulo: Editora Atlas, 2002.

GREASLEY, Andrew. The case for the organisational use of simulation. **Journal of Manufacturing Technology Management**, v. 15, n. 7, p. 560–566, 2004.

GREENWOOD, Allen G.; REES, Loren Paul; SIOCHI, Fernando C. An investigation of the behavior of simulation response surfaces. **European Journal of Operational Research**, v. 110, n. 2, p. 282–313, 1998.

GREINACHER, Sebastian *et al.* Multi-objective optimization of lean and resource efficient manufacturing systems. **Production Engineering**, v. 14, n. 2, p. 165–176, 2020. Disponível em: <<https://doi.org/10.1007/s11740-019-00945-9>>.

GUO, Rui; ZHANG, Yu. Exploration of correlation between environmental factors and mobility at signalized intersections. **Transportation Research Part D: Transport and Environment**, v. 32, p. 24–34, 2014. Disponível em: <<http://dx.doi.org/10.1016/j.trd.2014.05.011>>.

HACHICHA, W. A simulation metamodelling based neural networks for lot-sizing problem in MTO sector. **International Journal of Simulation Modelling**, v. 10, n. 4, p. 191–203, 2011.

HANNAH, Lauren A.; POWELL, Warren B.; DUCSON, David B. Semiconvex regression for metamodeling-based optimization. **SIAM Journal of Optimization**, v. 24, n. 2, p. 573–597, 2014.

HASSANNAYEBI, Erfan *et al.* Train schedule optimization in a high-speed railway

system using a hybrid simulation and meta-model approach. **Computers and Industrial Engineering**, v. 138, n. October, p. 106110, 2019. Disponível em: <<https://doi.org/10.1016/j.cie.2019.106110>>.

HASSOUN, M.H. **Fundamentals of Artificial Neural Networks**. Cambridge: [s.n.], 1995.

HASTIE, Trevor; TIBSHIRANI, Robert; FRIEDMAN, Jerome. **The Elements of Statistical Learning**. [S.l.: s.n.], 2008. v. 2.

HELLENO, A. L. *et al.* Integrating value stream mapping and discrete events simulation as decision making tools in operation management. **International Journal of Advanced Manufacturing Technology**, v. 80, n. 5–8, p. 1059–1066, 2015.

HIGHAM, Catherine F.; HIGHAM, Desmond J. Deep learning: An introduction for applied mathematicians. **SIAM Review**, v. 61, n. 4, p. 860–891, 2019.

HOAD, Kathryn; MONKS, Thomas; O'BRIEN, Frances. The use of search experimentation in discrete-event simulation practice. **Journal of the Operational Research Society**, v. 66, n. 7, p. 1155–1168, 2015.

HOLLAND, John H. **Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence**. 1^o ed. Cambridge, Massachusetts: MIT Press, 1975.

HONG, Jeong Hee; SEO, Kyung Min; KIM, Tag Gon. Simulation-based optimization for design parameter exploration in hybrid system: A defense system example. **Simulation**, v. 89, n. 3, p. 362–380, 2013.

HORNG, Shih Cheng; LIN, Shieh Shing. Embedding Advanced Harmony Search in Ordinal Optimization to Maximize Throughput Rate of Flow Line. **Arabian Journal for Science and Engineering**, v. 43, n. 2, p. 1015–1031, 2018.

HORNG, Shih Cheng; LIN, Shieh Shing. Ordinal optimization based metaheuristic algorithm for optimal inventory policy of assemble-to-order systems. **Applied Mathematical Modelling**, v. 42, p. 43–57, 2017. Disponível em: <<http://dx.doi.org/10.1016/j.apm.2016.10.002>>.

HOU, Zeyu *et al.* Application of Mixed-Integer Nonlinear Optimization Programming Based on Ensemble Surrogate Model for Dense Nonaqueous Phase Liquid Source Identification in Groundwater. **Environmental Engineering Science**, v. 36, n. 6, p. 699–709, 2019.

HU, Bo; PALTA, Mari; SHAO, Jun. Properties of R2 statistics for logistic regression. **Statistics in Medicine**, v. 25, n. 8, p. 1383–1395, 2006.

HÜLLEN, Gordon *et al.* Managing uncertainty in data-driven simulation-based

optimization. **Computers and Chemical Engineering**, v. 136, p. 106519, 2020. Disponível em: <<https://doi.org/10.1016/j.compchemeng.2019.106519>>.

HURRION, R. D. An example of simulation optimisation using a neural network metamodel: Finding the optimum number of kanbans in a manufacturing system. **Journal of the Operational Research Society**, v. 48, n. 11, p. 1105–1112, 1997.

HUSSAIN, Mohammed S. *et al.* A surrogate model for simulation-optimization of aquifer systems subjected to seawater intrusion. **Journal of Hydrology**, v. 523, p. 542–554, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.jhydrol.2015.01.079>>.

HUYET, A. L.; PARIS, J. L. Optimizing simulated manufacturing systems using machine learning coupled to evolutionary algorithms. **IEEE International Conference on Emerging Technologies and Factory Automation, ETFA**, v. 1, p. 17–21, 2001.

ILK, Noyan; SHANG, Guangzhi; GOES, Paulo. Improving customer routing in contact centers: An automated triage design based on text analytics. **Journal of Operations Management**, v. 66, n. 5, p. 553–577, 2020.

IVANOVA, Momchil; KUHNT, Sonja. A parallel optimization algorithm based on FANOVA decomposition. **Quality and Reliability Engineering International**, v. 30, n. 7, p. 961–974, 2014.

JACKSON, Ilya *et al.* Metamodelling of Inventory-Control Simulations Based on a Multilayer Perceptron. **Transport and Telecommunication**, v. 20, n. 3, p. 251–259, 2019.

JALALI, Hamed; NIEUWENHUYSE, Inneke Van. Simulation optimization in inventory replenishment: A classification. **IIE Transactions**, v. 47, n. 11, p. 1217–1235, 2015.

JENG, Jin Tsong. Hybrid approach of selecting hyperparameters of support vector machine for regression. **IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics**, v. 36, n. 3, p. 699–709, 2006.

KARA, Ahmet; DOGAN, Ibrahim. Reinforcement learning approaches for specifying ordering policies of perishable inventory systems. **Expert Systems with Applications**, v. 91, p. 150–158, 2018. Disponível em: <<http://dx.doi.org/10.1016/j.eswa.2017.08.046>>.

KIM, Sun Hye; BOUKOUVALA, Fani. Machine learning-based surrogate modeling for data-driven optimization: a comparison of subset selection for regression techniques. **Optimization Letters**, v. 14, n. 4, p. 989–1010, 2020. Disponível em: <<https://doi.org/10.1007/s11590-019-01428-7>>.

KIM, Wonuk; JEON, Yongseok; KIM, Yongchan. Simulation-based optimization of an integrated daylighting and HVAC system using the design of experiments method. **Applied Energy**, v. 162, p. 666–674, 2016. Disponível em:

<<http://dx.doi.org/10.1016/j.apenergy.2015.10.153>>.

KLEIJNEN, Jack P. C. Simulation Optimization Through Regression or Kriging Metamodels. **SSRN Electronic Journal**, 2017a.

KLEIJNEN, Jack P.C. Regression and Kriging metamodels with their experimental designs in simulation: A review. **European Journal of Operational Research**, v. 256, n. 1, p. 1–16, 2017b. Disponible em: <<http://dx.doi.org/10.1016/j.ejor.2016.06.041>>.

KLEIJNEN, Jack P C. Simulation-optimization via Kriging and bootstrapping: a survey. **Journal of Simulation**, v. 8, n. 4, p. 241–250, 2014. Disponible em: <<http://dx.doi.org/10.1057/jos.2014.4>>.

KLODA, Lorie A.; BORUFF, Jill T.; CAVALCANTE, Alexandre Soares. A comparison of patient, intervention, comparison, outcome (PICO) to a new, alternative clinical question framework for search skills, search results, and self-efficacy: A randomized controlled trial. **Journal of the Medical Library Association**, v. 108, n. 2, p. 185–194, 2020.

KOCHAN, Cigdem Gonul; NOWICKI, David R. Supply chain resilience: a systematic literature review and typological framework. **International Journal of Physical Distribution and Logistics Management**, v. 48, n. 8, p. 842–865, 2018.

KRIGE, D. G. A Statistical Approach to Some Basic Mine Valuation Problems on the Witwatersrand. **Journal of the Chemical Metallurgical & Society of South Mining Africa**, v. 52, n. 6, p. 119–139, 1951. Disponible em: <http://journals.co.za/content/saimm/52/6/AJA0038223X_4792>.

KRITYAKIERNE, Tupaluck; AKHTAR, Taimoor; SHOEMAKER, Christine A. SOP: parallel surrogate global optimization with Pareto center selection for computationally expensive single objective problems. **Journal of Global Optimization**, v. 66, n. 3, p. 417–437, 2016.

LAL, Alvin; DATTA, Bithin. Performance Evaluation of Homogeneous and Heterogeneous Ensemble Models for Groundwater Salinity Predictions: a Regional-Scale Comparison Study. **Water, Air, and Soil Pollution**, v. 231, n. 6, 2020.

LAW, Averill M. **Simulation Modeling and Analysis, FIFTH EDITION**. [S.l.: s.n.], 2013. Disponible em: <www.averill-law.com>.

LESSMANN, Stefan; STAHLBOCK, Robert; CRONE, Sven F. Genetic algorithms for support vector machine model selection. **IEEE International Conference on Neural Networks - Conference Proceedings**, p. 3063–3069, 2006.

LEVY, Yair; ELLIS, Timothy J. A systems approach to conduct an effective literature review in support of information systems research. **Informing Science**, v. 9, n. May 2014, p.

181–211, 2006.

LI, Guilin; HWAI-YONG TAN, Matthias; HUI NG, Szu. Metamodel-based optimization of stochastic computer models for engineering design under uncertain objective function. **IISE Transactions**, v. 51, n. 5, p. 517–530, 2019. Disponível em: <<https://doi.org/10.1080/24725854.2018.1504355>>.

LI, Jiahui *et al.* Groundwater contamination source identification based on a hybrid particle swarm optimization-extreme learning machine. **Journal of Hydrology**, v. 584, p. 124657, 2020. Disponível em: <<https://doi.org/10.1016/j.jhydrol.2020.124657>>.

LI, Weizi; WOLINSKI, David; LIN, Ming C. City-scale traffic animation using statistical learning and metamodel-based optimization. **ACM Transactions on Graphics**, v. 36, n. 6, p. 1–12, 2017.

LI, Y. F. *et al.* A systematic comparison of metamodeling techniques for simulation optimization in Decision Support Systems. **Applied Soft Computing Journal**, v. 10, n. 4, p. 1257–1273, 2010.

LI, Yu Sheng *et al.* A novel random forest approach for imbalance problem in crime linkage. **Knowledge-Based Systems**, v. 195, n. xxxx, p. 105738, 2020. Disponível em: <<https://doi.org/10.1016/j.knosys.2020.105738>>.

LIN, James T.; CHEN, Chien Ming. Simulation optimization approach for hybrid flow shop scheduling problem in semiconductor back-end manufacturing. **Simulation Modelling Practice and Theory**, v. 51, p. 100–114, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.simpat.2014.10.008>>.

LUO, Jiannan; JI, Yefei; LU, Wenxi. Comparison of Surrogate Models Based on Different Sampling Methods for Groundwater Remediation. **Journal of Water Resources Planning and Management**, v. 145, n. 5, p. 1–9, 2019.

LUPERA CALAHORRANO, Gicela J. *et al.* Application of the Meta-Multiparametric methodology to the control of emissions in the industry under continuous and discrete uncertain parameters. **Chemical Engineering Research and Design**, v. 115, p. 365–373, 2016. Disponível em: <<http://dx.doi.org/10.1016/j.cherd.2016.09.006>>.

LV, Yisheng *et al.* Managing Emergency Traffic Evacuation with a Partially Random Destination Allocation Strategy: A Computational-Experiment-Based Optimization Approach. **IEEE Transactions on Intelligent Transportation Systems**, v. 16, n. 4, p. 2182–2191, 2015.

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**, v. 21, n. 2, p. 239–245, 1979.

METHLEY, Abigail M. *et al.* PICO, PICOS and SPIDER: A comparison study of specificity and sensitivity in three search tools for qualitative systematic reviews. **BMC Health Services Research**, v. 14, n. 1, 2014.

MIGUEL, PAC *et al.* **Metodologia de Pesquisa em Engenharia de Produção e Gestão de Operações**. 3^o ed. Rio de Janeiro: Elsevier, 2018.

MIRANDA, Rafael de Carvalho *et al.* Increasing the efficiency in integer simulation optimization: Reducing the search space through data envelopment analysis and orthogonal arrays. **European Journal of Operational Research**, v. 262, n. 2, p. 673–681, 2017.

MIRANDA, Rafael De Carvalho *et al.* A new approach to reducing search space and increasing efficiency in simulation optimization problems via the fuzzy-DEA-BCC. **Mathematical Problems in Engineering**, v. 2014, 2014.

MIRFENDERESGI, Golnazalsadat; MOUSAVI, S. Jamshid. Adaptive meta-modeling-based simulation optimization in basin-scale optimum water allocation: A comparative analysis of meta-models. **Journal of Hydroinformatics**, v. 18, n. 3, p. 446–465, 2016.

MITROFF, Ian I. *et al.* On Managing Science in the Systems Age: Two Schemas for the Study of Science as a Whole Systems Phenomenon. **Interfaces**, v. 4, n. 3, p. 46–58, 1974.

MOGHADDAM, Samira; MAHLOOJI, Hashem. A new metamodel-based method for solving semi-expensive simulation optimization problems. **Communications in Statistics: Simulation and Computation**, v. 46, n. 6, p. 4795–4811, 2017.

MOGHADDAM, Samira; MAHLOOJI, Hashem. Robust simulation optimization using ϕ -divergence. **International Journal of Industrial Engineering Computations**, v. 7, n. 4, p. 517–534, 2016.

MOHAMMAD NEZHAD, Ali; MAHLOOJI, Hashem. An artificial neural network meta-model for constrained simulation optimization. **Journal of the Operational Research Society**, v. 65, n. 8, p. 1232–1244, 2014. Disponível em: <<http://dx.doi.org/10.1057/jors.2013.73>>.

MONGEON, Philippe; PAUL-HUS, Adèle. The journal coverage of Web of Science and Scopus: a comparative analysis. **Scientometrics**, v. 106, n. 1, p. 213–228, 2016.

MONOSTORI, L.; VIHAROS, Zs J. Hybrid, al- and simulation-supported optimisation of process chains and production plants. **CIRP Annals - Manufacturing Technology**, v. 50, n. 1, p. 353–356, 2001.

MONTEVECHI, Jose Arnaldo B *et al.* Identification of the main methods used in simulation projects. 2015, [S.l.]: Proceedings of the 2015 Winter Simulation Conference, 2015. p. 3469–3480.

MONTEVECHI, José Arnaldo Barra *et al.* **Application of design of experiments on the simulation of a process in an automotive industry. Winter Simulation Conference.** [S.l.: s.n.], 2007.

MONTEVECHI, José Arnaldo Barra *et al.* **Conceptual modeling in simulation projects by mean adapted IDEF: an application in a brazilian tech company. Proceedings - Winter Simulation Conference.** [S.l.: s.n.], 2010.

MUELLER, Egon; CHEN, Xiao Li; RIEDEL, Ralph. Challenges and Requirements for the Application of Industry 4.0: A Special Insight with the Usage of Cyber-Physical System. **Chinese Journal of Mechanical Engineering (English Edition)**, v. 30, n. 5, p. 1050–1057, 2017.

MYERS, Raymond H.; MONTGOMERY, Douglas C.; ANDERSON-COOK, Christine M. **Response surface methodology : process and product optimization using designed experiments.** 4^o ed. Hoboken, New Jersey: John Wiley & Sons, 2016.

NGUYEN-TUONG, Duy; SEEGER, Matthias; PETERS, Jan. Model learning with local Gaussian process regression. **Advanced Robotics**, v. 23, n. 15, p. 2015–2034, 2009.

OLIVEIRA, Adriano L.I. *et al.* GA-based method for feature selection and parameters optimization for machine learning regression applied to software effort estimation. **Information and Software Technology**, v. 52, n. 11, p. 1155–1166, 2010. Disponível em: <<http://dx.doi.org/10.1016/j.infsof.2010.05.009>>.

OLIVEIRA, Josenildo Brito; LIMA, Renato Silva; MONTEVECHI, José Arnaldo Barra. Perspectives and relationships in Supply Chain Simulation: A systematic literature review. **Simulation Modelling Practice and Theory**, v. 62, p. 166–191, 2016.

OSORIO, Carolina. Dynamic origin-destination matrix calibration for large-scale network simulators. **Transportation Research Part C: Emerging Technologies**, v. 98, n. December 2018, p. 186–206, 2019a. Disponível em: <<https://doi.org/10.1016/j.trc.2018.09.023>>.

OSORIO, Carolina. High-dimensional offline origin-destination (OD) demand calibration for stochastic traffic simulators of large-scale road networks. **Transportation Research Part B: Methodological**, v. 124, p. 18–43, 2019b. Disponível em: <<https://doi.org/10.1016/j.trb.2019.01.005>>.

OSORIO, Carolina; BIERLAIRE, Michel. A simulation-based optimization framework for urban transportation problems. **Operations Research**, v. 61, n. 6, p. 1333–1345, 2013.

OSORIO, Carolina; CHONG, Linsen. A computationally efficient simulation-based optimization algorithm for large-scale urban transportation problems. **Transportation Science**,

v. 49, n. 3, p. 623–636, 2015.

OSORIO, Carolina; NANDURI, Kanchana. Urban transportation emissions mitigation: Coupling high-resolution vehicular emissions and traffic models for traffic signal optimization. **Transportation Research Part B: Methodological**, v. 81, p. 520–538, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.trb.2014.12.007>>.

OSORIO, Carolina; PUNZO, Vincenzo. Efficient calibration of microscopic car-following models for large-scale stochastic network simulators. **Transportation Research Part B: Methodological**, v. 119, p. 156–173, 2019. Disponível em: <<https://doi.org/10.1016/j.trb.2018.09.005>>.

ØSTERGÅRD, Torben; JENSEN, Rasmus Lund; MAAGAARD, Steffen Enersen. A comparison of six metamodeling techniques applied to building performance simulations. **Applied Energy**, v. 211, n. October 2017, p. 89–103, 2018.

OUYANG, Qi *et al.* Conservative strategy-based ensemble surrogate model for optimal groundwater remediation design at DNAPLs-contaminated sites. **Journal of Contaminant Hydrology**, v. 203, n. April, p. 1–8, 2017. Disponível em: <<http://dx.doi.org/10.1016/j.jconhyd.2017.05.007>>.

PARNIANIFARD, Amir *et al.* Comparative study of metamodeling and sampling design for expensive and semi-expensive simulation models under uncertainty. **Simulation**, v. 96, n. 1, p. 89–110, 2020a.

PARNIANIFARD, Amir *et al.* Crossing weighted uncertainty scenarios assisted distribution-free metamodel-based robust simulation optimization. **Engineering with Computers**, v. 36, n. 1, p. 139–150, 2020b. Disponível em: <<http://dx.doi.org/10.1007/s00366-018-00690-0>>.

PARNIANIFARD, Amir *et al.* Kriging and Latin hypercube sampling assisted simulation optimization in optimal design of PID controller for speed control of DC motor. **Journal of Computational and Theoretical Nanoscience**, v. 15, n. 5, p. 1471–1479, 2018.

PARNIANIFARD, Amir *et al.* Recent developments in metamodel based robust black-box simulation optimization: An overview. **Decision Science Letters**, v. 8, n. 1, p. 17–44, 2019.

PARNIANIFARD, Amir; AZFANIZAM, Ahmad Siti. Metamodel-based robust simulation-optimization assisted optimal design of multiloop integer and fractional-order PID controller. **International Journal of Numerical Modelling: Electronic Networks, Devices and Fields**, v. 33, n. 1, p. 1–20, 2020.

PEDREGOSA, Fabian *et al.* Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, v. 12, n. 1, p. 2825–2830, 2011.

PEDRIELLI, Giulia *et al.* A real time simulation optimization framework for vessel collision avoidance and the case of singapore strait. **IEEE Transactions on Intelligent Transportation Systems**, v. 21, n. 3, p. 1204–1215, 2020.

QUAN, Ning *et al.* Simulation optimization via kriging: A sequential search using expected improvement with computing budget constraints. **IIE Transactions (Institute of Industrial Engineers)**, v. 45, n. 7, p. 763–780, 2015.

RAEI, Ehsan *et al.* Multi-objective decision-making for green infrastructure planning (LID-BMPs) in urban storm water management under uncertainty. **Journal of Hydrology**, v. 579, n. August, p. 124091, 2019. Disponível em: <<https://doi.org/10.1016/j.jhydrol.2019.124091>>.

RANJBAR, Ali; MAHJOURI, Najmeh; CHERUBINI, Claudia. Development of an efficient conjunctive meta-model-based decision-making framework for saltwater intrusion management in coastal aquifers. **Journal of Hydro-Environment Research**, v. 29, n. November, p. 45–58, 2020. Disponível em: <<https://doi.org/10.1016/j.jher.2019.11.005>>.

ROJAS GONZALEZ, Sebastian; JALALI, Hamed; VAN NIEUWENHUYSE, Inneke. A multiobjective stochastic simulation optimization algorithm. **European Journal of Operational Research**, v. 284, n. 1, p. 212–226, 2020. Disponível em: <<https://doi.org/10.1016/j.ejor.2019.12.014>>.

ROSTAMI, Habib; KHAKSAR MANSHAD, Abbas. Application of evolutionary Gaussian processes regression by particle swarm optimization for prediction of dew point pressure in gas condensate reservoirs. **Neural Computing and Applications**, v. 24, n. 3–4, p. 705–713, 2014.

ROY, Dilip Kumar; DATTA, Bithin. Adaptive management of coastal aquifers using entropy-set pair analysis-based three-dimensional sequential monitoring network design. **Journal of Hydrologic Engineering**, v. 24, n. 3, 2019a.

ROY, Dilip Kumar; DATTA, Bithin. An Ensemble Meta-Modelling Approach Using the Dempster-Shafer Theory of Evidence for Developing Saltwater Intrusion Management Strategies in Coastal Aquifers. **Water Resources Management**, v. 33, n. 2, p. 775–795, 2019b.

ROY, Dilip Kumar; DATTA, Bithin. Influence of sea level rise on multiobjective management of saltwater intrusion in coastal aquifers. **Journal of Hydrologic Engineering**, v. 23, n. 8, p. 1–17, 2018a.

ROY, Dilip Kumar; DATTA, Bithin. Trained meta-models and evolutionary algorithm based multi-objective management of coastal aquifers under parameter uncertainty. **Journal of Hydroinformatics**, v. 20, n. 6, p. 1247–1267, 2018b.

SALAM, Mohammad Asif; KHAN, Sami A. Simulation based decision support system for optimization: A case of Thai logistics service provider. **Industrial Management and Data Systems**, v. 116, n. 2, p. 236–254, 2016.

SCHAPIRE, Robert E. The Strength of Weak Learnability. **Machine Learning**, v. 5, n. 2, p. 197–227, 1990.

SCHRUBEN, Lee. Simulation optimization using frequenct domain methods. 1986, [S.l: s.n.], 1986. p. 366–369.

SHARIATINASAB, Reza *et al.* Optimization of surge arrester's location on EHV and UHV power networks using simulation optimization method. **IEEJ Transactions on Power and Energy**, v. 128, n. 12, p. 1465–1472, 2008.

SHARIF, Seyed Amirhosain; HAMMAD, Amin. Developing surrogate ANN for selecting near-optimal building energy renovation methods considering energy consumption, LCC and LCA. **Journal of Building Engineering**, v. 25, n. October 2018, p. 100790, 2019. Disponível em: <<https://doi.org/10.1016/j.jobe.2019.100790>>.

SHIRAZI, Babak; MAHDAVI, Iraj; MAHDAVI-AMIRI, Nezam. ICoSim-FMS: An intelligent co-simulator for the adaptive control of complex flexible manufacturing systems. **Simulation Modelling Practice and Theory**, v. 19, n. 7, p. 1668–1688, 2011. Disponível em: <<http://dx.doi.org/10.1016/j.simpat.2011.04.003>>.

SIEBERS, P. O. *et al.* Discrete-event simulation is dead, long live agent-based simulation! **Journal of Simulation**, v. 4, n. 3, p. 204–210, 2010.

SMEW, W.; YOUNG, P.; GERAGHTY, J. Supply chain analysis using simulation, gaussian process modelling and optimisation. **International Journal of Simulation Modelling**, v. 12, n. 3, p. 178–189, 2013.

SONG, Chen *et al.* Machine learning approach for determining feasible plans of a remanufacturing system. **IEEE Transactions on Automation Science and Engineering**, v. 2, n. 3, p. 262–275, 2005.

SONG, Chen *et al.* Remanufacturing planning based on constrained ordinal optimization. **Frontiers of Electrical and Electronic Engineering in China**, v. 6, n. 3, p. 443–452, 2011.

SOUSA JUNIOR, W. T. De *et al.* Economic Lot-Size Using Machine Learning, Parallelism, Metaheuristic and Simulation. **International Journal of Simulation Modelling**, v. 18, n. 2, p. 205–216, 2019.

SOUSA JUNIOR, W. T. De *et al.* Shop floor simulation optimization using machine learning to improve parallel metaheuristics. **Expert Systems with Applications**, 2020.

SOUSA JUNIOR, Wilson Trigueiro De *et al.* Discrete simulation-based optimization methods for industrial engineering problems: A systematic literature review. **Computers and Industrial Engineering**, v. 128, n. January, p. 526–540, 2019. Disponível em: <<https://doi.org/10.1016/j.cie.2018.12.073>>.

SOUSA JUNIOR, Wilson Trigueiro *et al.* Discrete simulation-based optimization methods for industrial engineering problems: A systematic literature review. **Computers and Industrial Engineering**, v. 128, n. January, p. 526–540, 2019. Disponível em: <<https://doi.org/10.1016/j.cie.2018.12.073>>.

SREEKANTH, J.; DATTA, Bithin. Optimal combined operation of production and barrier wells for the control of saltwater intrusion in coastal groundwater well fields. **Desalination and Water Treatment**, v. 32, n. 1–3, p. 72–78, 2011.

STEER, Kent C.B.; WIRTH, Andrew; HALGAMUGE, Saman K. Decision tree ensembles for online operation of large smart grids. **Energy Conversion and Management**, v. 59, p. 9–18, 2012. Disponível em: <<http://dx.doi.org/10.1016/j.enconman.2012.01.010>>.

STORK, André *et al.* Towards interactive simulation in automotive design. **Visual Computer**, v. 24, n. 11, p. 947–953, 2008.

STORTI, Bruno A. *et al.* Improving the efficiency of a Savonius wind turbine by designing a set of deflector plates with a metamodel-based optimization approach. **Energy**, v. 186, p. 115814, 2019. Disponível em: <<https://doi.org/10.1016/j.energy.2019.07.144>>.

TANILA, Tuomo; TENHUNEN, Henni; HIRVONEN, Petteri. Value mechanisms in the implementation of intelligent patient flow management system - A multiple case study. **Studies in Health Technology and Informatics**, v. 270, p. 708–712, 2020.

TIMANI, Bassel; PERALTA, Richard. Aggregated surrogate simulator for groundwater-surface water management via simulation-optimization modeling: Theory, development and tests. **Environmental Modelling and Software**, v. 96, p. 210–231, 2017. Disponível em: <<http://dx.doi.org/10.1016/j.envsoft.2017.06.014>>.

TURRIONI, J.; MELLO, C. Metodologia De Pesquisa Em Engenharia De Produção: Estratégias, Métodos E Técnicas Para Condução De Pesquisas Quantitativas E Qualitativas. **Programa de Pós-graduação em Engenharia de Produção**, v. 1, n. Programa Pós-graduação em Eng. Produção, p. 191, 2012.

VAHIDI, B.; TAVAKOLI, M. R. Bank; HOSSEINIAN, S. H. Determining arresters best positions in power system for lightning shielding failure protection using simulation optimization approach. **EUROPEAN TRANSACTIONS ON ELECTRICAL POWER**, v. 20, n. 3, p. 255–276, 2008. Disponível em:

<http://linkinghub.elsevier.com/retrieve/pii/S1568494604000596%5Cnhttp://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4137655%5Cnhttp://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5512904%5Cnhttp://ieeexplore.ieee.org/xpls/abs_all.jsp?>>.

VAPNIK, Vladimir N. **The Nature of Statistical Learning Theory**. New York: [s.n.], 1999.

VILLARREAL-MARROQUÍN, María G. *et al.* A comparison of two metamodel-based methodologies for multiple criteria simulation optimization using an injection molding case study. **Journal of Polymer Engineering**, v. 33, n. 3, p. 193–209, 2013.

VILLARREAL-MARROQUÍN, María G. *et al.* Selecting process parameter in injection molding via simulation optimization. **Journal of Polymer Engineering**, v. 31, n. 5, p. 387–395, 2011.

VU, Ky Khac *et al.* Surrogate-based methods for black-box optimization. **International Transactions in Operational Research**, v. 24, n. 3, p. 393–424, 2017.

WADE, Brian M. Creating surrogate models for an air and missile defense simulation using design of experiments and neural networks. **Journal of Defense Modeling and Simulation**, 2019.

WAN, Hua Ping; REN, Wei Xin. Parameter selection in finite-element-model updating by global sensitivity analysis using gaussian process metamodel. **Journal of Structural Engineering**, v. 141, n. 6, p. 1–11, 2015.

WANG, Haowei; YUAN, Jun; NG, Szu Hui. Gaussian process based optimization algorithms with input uncertainty. **IISE Transactions**, v. 52, n. 4, p. 377–393, 2020. Disponible em: <<http://dx.doi.org/10.1080/24725854.2019.1639859>>.

WANG, Hu; CHEN, Lei; LI, Enying. Time dependent sheet metal forming optimization by using Gaussian process assisted firefly algorithm. **International Journal of Material Forming**, v. 11, n. 2, p. 279–295, 2018.

WILDRIDGE, Valerie; BELL, Lucy. How CLIP became ECLIPSE: a mnemonic to assist in searching for health policy/ management information. **Health Information and Libraries Journal**, v. 19, n. 1, p. 113–115, 2002.

WOLDEMARIAM, Endashaw Tesfaye; LEMU, Hirpa G. A machine learning based framework for model approximation followed by design optimization for expensive numerical simulation-based optimization problems. **Proceedings of the International Offshore and Polar Engineering Conference**, v. 1, p. 943–950, 2019.

WOLLMANN, Dewey; TORTATO, Ubiratã. Proposal for a model to hierarchize strategic decisions according to criteria of value innovation, sustainability and budgetary

constraint. **Journal of Cleaner Production**, v. 231, p. 278–289, 2019.

XIA, Yufei *et al.* A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. **Expert Systems with Applications**, v. 78, p. 225–241, 2017. Disponível em: <<http://dx.doi.org/10.1016/j.eswa.2017.02.017>>.

XIAO, Hui; LEE, Loo Hay; CHEN, Chun Hung. Optimal budget allocation rule for simulation optimization using quadratic regression in partitioned domains. **IEEE Transactions on Systems, Man, and Cybernetics: Systems**, v. 45, n. 7, p. 1047–1062, 2015.

XIAO, Yu; WATSON, Maria. Guidance on Conducting a Systematic Literature Review. **Journal of Planning Education and Research**, v. 39, n. 1, p. 93–112, 2019.

XU, Jie *et al.* Simulation optimization in the era of Industrial 4.0 and the Industrial Internet. **Journal of Simulation**, v. 10, n. 4, p. 310–320, 2016.

XU, Lizhong *et al.* Simulation and prediction of hydrological processes based on firefly algorithm with deep learning and support vector for regression. **International Journal of Parallel, Emergent and Distributed Systems**, v. 35, n. 3, p. 288–296, 2020. Disponível em: <<https://doi.org/17445760.2019.1593409>>.

YADAV, Basant *et al.* Estimation of in-situ bioremediation system cost using a hybrid Extreme Learning Machine (ELM)-particle swarm optimization approach. **Journal of Hydrology**, v. 543, p. 373–385, 2016. Disponível em: <<http://dx.doi.org/10.1016/j.jhydrol.2016.10.013>>.

YANG, Li; SHAMI, Abdallah. On hyperparameter optimization of machine learning algorithms: Theory and practice. **Neurocomputing**, v. 415, p. 295–316, 2020. Disponível em: <<https://doi.org/10.1016/j.neucom.2020.07.061>>.

YAOHUI, Li. A Kriging-based global optimization method using multi-points infill search criterion. **Journal of Algorithms and Computational Technology**, v. 11, n. 4, p. 366–377, 2017.

YE, Wenhe; YOU, Fengqi. A simulation-based optimization method for solving the integrated supply chain network design and inventory control problem under uncertainty. **Chemical Engineering Transactions**, v. 45, p. 499–504, 2015.

YIN, Hanfeng *et al.* Design optimization of a MASH TL-3 concrete barrier using RBF-based metamodels and nonlinear finite element simulations. **Engineering Structures**, v. 114, p. 122–134, 2016. Disponível em: <<http://dx.doi.org/10.1016/j.engstruct.2016.02.009>>.

YOUSEFI, Milad *et al.* Chaotic genetic algorithm and Adaboost ensemble metamodeling approach for optimum resource planning in emergency departments. **Artificial Intelligence in Medicine**, v. 84, p. 23–33, 2018. Disponível em:

<<http://dx.doi.org/10.1016/j.artmed.2017.10.002>>.

YOUSEFI, Milad; YOUSEFI, Moslem. Human resource allocation in an emergency department: A metamodel-based simulation optimization. **Kybernetes**, v. 49, n. 3, p. 779–796, 2019.

ZAKERIFAR, Mehdi; BILES, William E.; EVANS, Gerald W. Kriging metamodeling in multiple-objective simulation optimization. **Simulation**, v. 87, n. 10, p. 843–856, 2011.

ZEINALI, Farzad; MAHOOTCHI, Masoud; SEPEHRI, Mohammad Mehdi. Resource planning in the emergency departments: A simulation-based metamodeling approach. **Simulation Modelling Practice and Theory**, v. 53, p. 123–138, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.simpat.2015.02.002>>.

ZELENYUK, Valentin. Aggregation of inputs and outputs prior to Data Envelopment Analysis under big data. **European Journal of Operational Research**, v. 282, n. 1, p. 172–187, 2020.

ZHANG, Chao; OSORIO, Carolina; FLÖTTERÖD, Gunnar. Efficient calibration techniques for large-scale traffic simulators. **Transportation Research Part B: Methodological**, v. 97, p. 214–239, 2017. Disponível em: <<http://dx.doi.org/10.1016/j.trb.2016.12.005>>.

ZHANG, Juan *et al.* Comparing Keywords Plus of WOS and Author Keywords: A Case Study of Patient Adherence Research. **Journal of the American Society for Information Science and Technology**, v. 64, n. July, p. 967–972, 2015. Disponível em: <<http://onlinelibrary.wiley.com/doi/10.1002/asi.22883/abstract>>.

ZLATINOV, Metodi; LASKOWSKI, Greg. Hybrid large-eddy simulation optimization of a fundamental turbine blade turbulated cooling passage. **Journal of Guidance, Control, and Dynamics**, v. 38, n. 6, p. 1292–1297, 2015.

ANEXO A – Símbolos do IDEF-SIM

Elementos	Simbologia	Técnica de Origem	
Entidade		IDEF3 (modo de descrição de transições)	
Funções		IDEF0	
Fluxo de Entidade		IDEF0 e IDEF3	
Recursos		IDEF0	
Controles		IDEF3	
Regras para fluxos alternativos E/OU paralelos	 &	Regra E	IDEF3
	 X	Regra OU	
	 O	Regra E/OU	
Movimento		Fluxograma	
Informação		IDEF0 e IDEF3	
Fluxo de entrada em sistema modelado			
Fim do sistema			
Conexão para outra figura			

Figura A.1 - Simbologia utilizada no IDEF-SIM

Fonte: Adaptado de Montevechi *et al.* (2010)

ANEXO B – Modelos Conceituais em IDEF-SIM

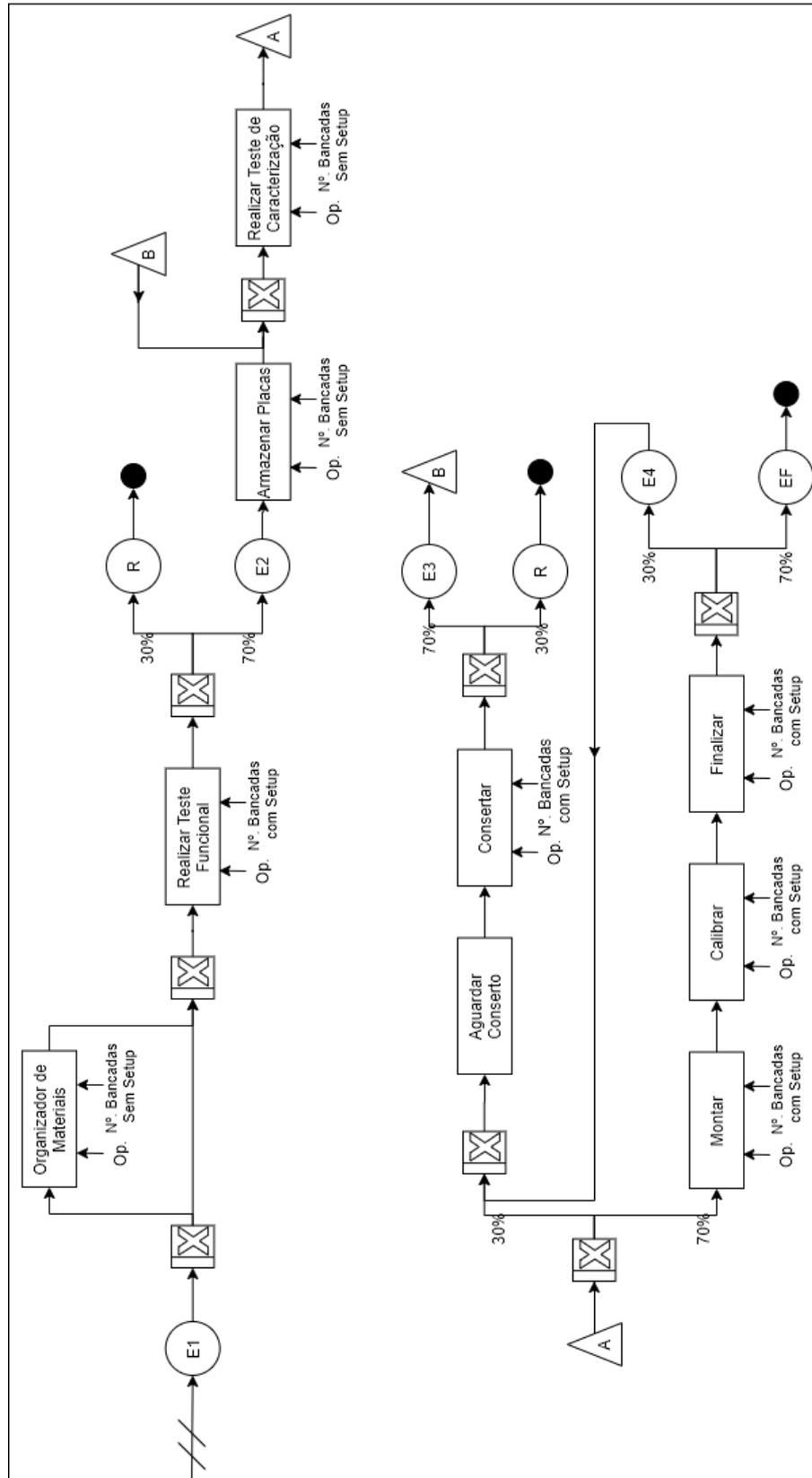


Figura B.2 – IDEF-SIM para o OE1

Fonte: Adaptado de Miranda *et al.* (2017)

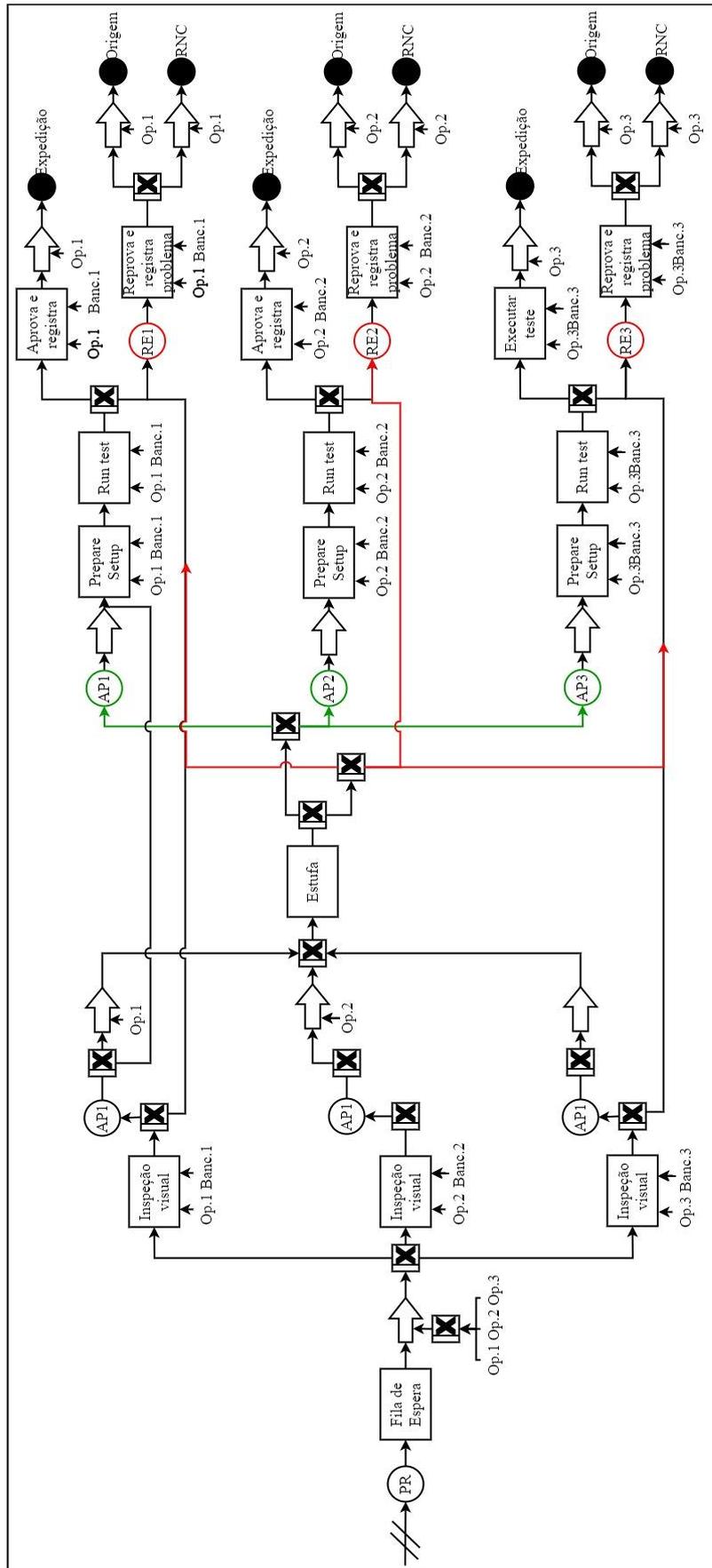


Figura B.3 - IDEF-SIM para o OE2

Fonte: Adaptado de Miranda *et al.* (2017)

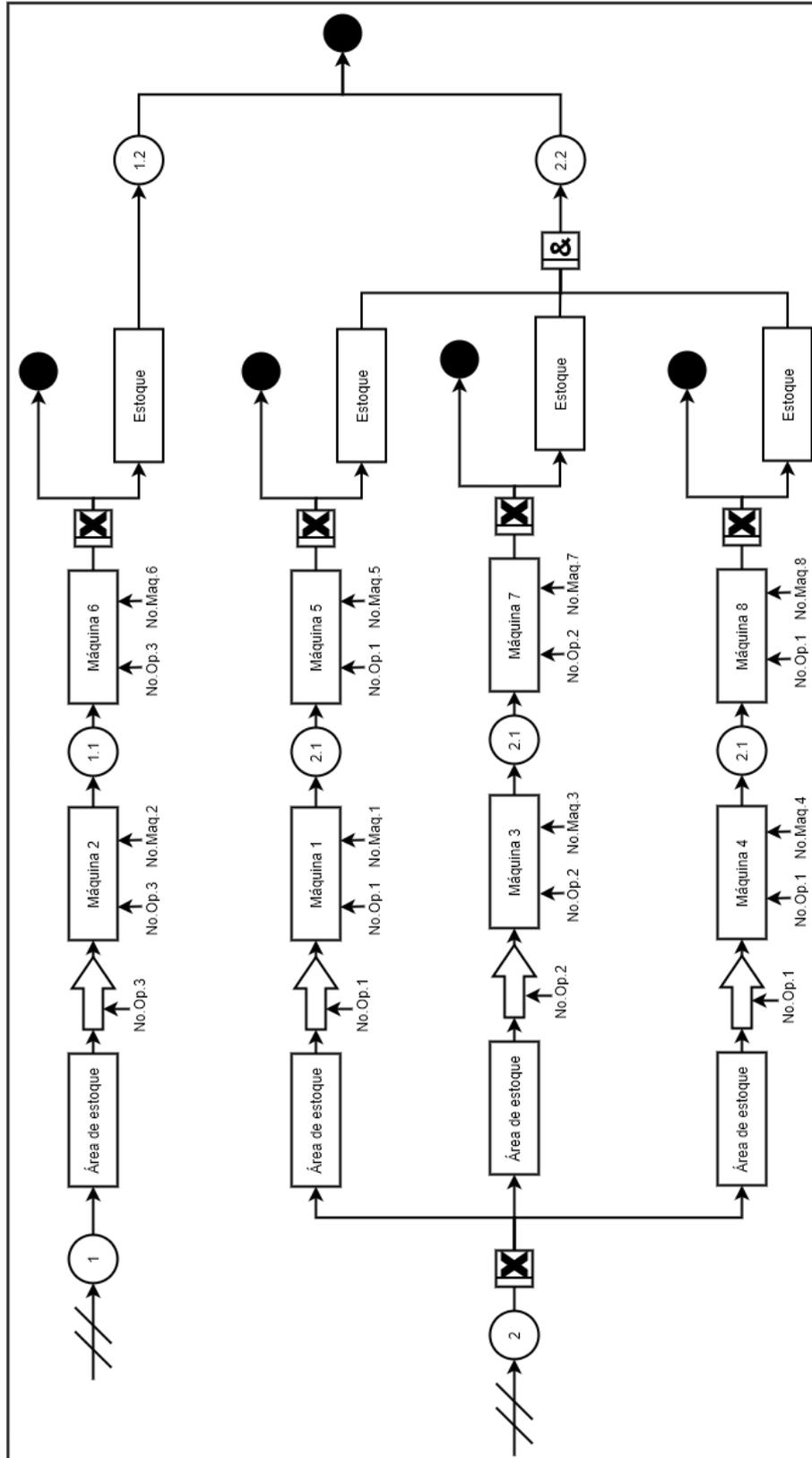


Figura B.4 - Modelo conceitual para o OE3

Fonte: Adaptado de Miranda *et al.* (2017)

ANEXO C – Hiper-parâmetros

Tabela C.1 - Seleção dos hiper-parâmetros para o OE2

Algoritmo	Amplitude	Algoritmo	Amplitude
GP		RF	
<i>Lengthscale</i> (θ)	[10 ⁻¹⁰ ; 10 ¹⁰]	Nº de árvores (t)	[10; 50]
<i>Epsilon</i> ^l (e)	[10 ⁻⁶ ; 10 ⁻⁵]	Nº de variáveis por nó(v)	{ 1; 2; 3 }
Tolerância geométrica (G)	[0; 0,3]		
Nº max. de vetores (mbv)	[38; 47]	GBT	
RP		Nº de árvores (t)	[1; 100]
<i>Degree</i> (D)	{ 1º, 2º, 3º, 4º, 5º }	<i>Learning rate</i> (α)	[0,05; 0,30]
		Min. de Obs. Por folha (l)	[0,12; 0,50]
RNA		SVM	
<i>Learning rate</i> (α)	[0,01; 0,4]	C^*	[39.255; 393.189]
<i>Epochs</i> (E)	[200; 1500]	<i>Epsilon</i> [*] (e)	[0; 4.208]
Nº de camadas (ω)	1	Gama (γ)	[0,0224; 0,1294]
Nº de neurônios (n)	[1; 66]		

* Os valores para C e $epsilon$ foram calculados a partir de uma execução preliminar do algoritmo com parâmetros default.

Tabela C.2 - Seleção dos hiper-parâmetros para o OE3

Algoritmo	Amplitude	Algoritmo	Amplitude
GP		RF	
<i>Lengthscale</i> (θ)	[10 ⁻¹⁰ ; 10 ¹⁰]	Nº de árvores (t)	[10; 50]
<i>Epsilon</i> ^l (e)	[10 ⁻⁶ ; 10 ⁻⁵]	Nº de variáveis por nó(v)	{ 3; 4; 5 }
Tolerância geométrica (G)	[0; 0,3]		
Nº max. de vetores (mbv)	[38; 47]	GBT	
RP		Nº de árvores (t)	[1; 100]
<i>Degree</i> (D)	{ 1º, 2º, 3º, 4º, 5º }	<i>Learning rate</i> (α)	[0,05; 0,30]
		Min. de Obs. Por folha (l)	[0,12; 0,50]
RNA		SVM	
<i>Learning rate</i> (α)	[0,01; 0,4]	C^*	[186.520; 896.737]
<i>Epochs</i> (E)	[200; 1500]	<i>Epsilon</i> [*] (e)	[0; 13.882]
Nº de camadas (ω)	1	Gama (γ)	[0,0224; 0,1294]
Nº de neurônios (n)	[1; 66]		

* Os valores para C e $epsilon$ foram calculados a partir de uma execução preliminar do algoritmo com parâmetros default.

ANEXO D – Código para Treinamento, Otimização e Aplicação dos Metamodelos

```

<?xml version="1.0" encoding="UTF-8"?><process version="9.8.000">
  <context>
    <input/>
    <output/>
    <macros/>
  </context>
  <operator activated="true" class="process" compatibility="9.8.000" expanded="true" name="Process">
    <parameter key="logverbosity" value="init"/>
    <parameter key="random_seed" value="2001"/>
    <parameter key="send_mail" value="never"/>
    <parameter key="notification_email" value=""/>
    <parameter key="process_duration_for_mail" value="30"/>
    <parameter key="encoding" value="SYSTEM"/>
    <process expanded="true">
      <operator activated="true" class="read_excel" compatibility="9.8.000" expanded="true" height="68" name="Solution Space" width="90"
x="112" y="34">
        <parameter key="excel_file" value="espaço de solução case padtec.xlsx"/>
        <parameter key="sheet_selection" value="sheet number"/>
        <parameter key="sheet_number" value="1"/>
        <parameter key="imported_cell_range" value="C1:K10485776"/>
        <parameter key="encoding" value="SYSTEM"/>
        <parameter key="first_row_as_names" value="true"/>
        <list key="annotations"/>
        <parameter key="date_format" value=""/>
        <parameter key="time_zone" value="SYSTEM"/>
        <parameter key="locale" value="English (United States)"/>
        <parameter key="read_all_values_as_polynomial" value="false"/>
        <list key="data_set_meta_data_information">
          <parameter key="0" value="ID.true.integer.id"/>
          <parameter key="1" value="X1.true.integer.attribute"/>
          <parameter key="2" value="X2.true.integer.attribute"/>
          <parameter key="3" value="X3.true.integer.attribute"/>
          <parameter key="4" value="X4.true.integer.attribute"/>
          <parameter key="5" value="X5.true.integer.attribute"/>
          <parameter key="6" value="X6.true.integer.attribute"/>
          <parameter key="7" value="X7.true.integer.attribute"/>
          <parameter key="8" value="lucro.true.polynomial.label"/>
        </list>
        <parameter key="read_not_matching_values_as_missings" value="false"/>
        <parameter key="datamanagement" value="double_array"/>
        <parameter key="data_management" value="auto"/>
      </operator>
      <operator activated="true" class="subprocess" compatibility="9.8.000" expanded="true" height="187" name="Training Database (2)"
width="90" x="112" y="238">
        <process expanded="true">
          <operator activated="true" class="read_excel" compatibility="9.8.000" expanded="true" height="68" name="20x (2)" width="90"
x="246" y="85">
            <parameter key="excel_file" value="Base de treinamento OE1.xlsx"/>
            <parameter key="sheet_selection" value="sheet number"/>
            <parameter key="sheet_number" value="1"/>
            <parameter key="imported_cell_range" value="A1"/>
            <parameter key="encoding" value="SYSTEM"/>
            <parameter key="first_row_as_names" value="true"/>
            <list key="annotations"/>
            <parameter key="date_format" value=""/>
            <parameter key="time_zone" value="SYSTEM"/>
            <parameter key="locale" value="English (United States)"/>
            <parameter key="read_all_values_as_polynomial" value="false"/>
            <list key="data_set_meta_data_information">
              <parameter key="0" value="ID.true.polynomial.id"/>
              <parameter key="1" value="X1.true.integer.attribute"/>
              <parameter key="2" value="X2.true.integer.attribute"/>
              <parameter key="3" value="X3.true.integer.attribute"/>
              <parameter key="4" value="X4.true.integer.attribute"/>
              <parameter key="5" value="X5.true.integer.attribute"/>
            </list>
          </operator>
        </process>
      </operator>
    </operator>
  </operator>
</process>

```

```

    <parameter key="6" value="X6.true.integer.attribute"/>
    <parameter key="7" value="X7.true.integer.attribute"/>
    <parameter key="8" value="lucro.true.real.label"/>
  </list>
  <parameter key="read_not_matching_values_as_missings" value="false"/>
  <parameter key="datamanagement" value="double_array"/>
  <parameter key="data_management" value="auto"/>
</operator>
<operator activated="true" class="read_excel" compatibility="9.8.000" expanded="true" height="68" name="40x (2)" width="90"
x="246" y="187">
  <parameter key="excel_file" value="Base de treinamento OE1.xlsx"/>
  <parameter key="sheet_selection" value="sheet number"/>
  <parameter key="sheet_number" value="3"/>
  <parameter key="imported_cell_range" value="A1"/>
  <parameter key="encoding" value="SYSTEM"/>
  <parameter key="first_row_as_names" value="true"/>
  <list key="annotations"/>
  <parameter key="date_format" value=""/>
  <parameter key="time_zone" value="SYSTEM"/>
  <parameter key="locale" value="English (United States)"/>
  <parameter key="read_all_values_as_polynomial" value="false"/>
  <list key="data_set_meta_data_information">
    <parameter key="0" value="ID.true.polynomial.id"/>
    <parameter key="1" value="X1.true.integer.attribute"/>
    <parameter key="2" value="X2.true.integer.attribute"/>
    <parameter key="3" value="X3.true.integer.attribute"/>
    <parameter key="4" value="X4.true.integer.attribute"/>
    <parameter key="5" value="X5.true.integer.attribute"/>
    <parameter key="6" value="X6.true.integer.attribute"/>
    <parameter key="7" value="X7.true.integer.attribute"/>
    <parameter key="8" value="lucro.true.real.label"/>
  </list>
  <parameter key="read_not_matching_values_as_missings" value="false"/>
  <parameter key="datamanagement" value="double_array"/>
  <parameter key="data_management" value="auto"/>
</operator>
<operator activated="true" class="read_excel" compatibility="9.8.000" expanded="true" height="68" name="60x" width="90" x="246"
y="289">
  <parameter key="excel_file" value="Base de treinamento OE1.xlsx"/>
  <parameter key="sheet_selection" value="sheet number"/>
  <parameter key="sheet_number" value="4"/>
  <parameter key="imported_cell_range" value="A1"/>
  <parameter key="encoding" value="SYSTEM"/>
  <parameter key="first_row_as_names" value="true"/>
  <list key="annotations"/>
  <parameter key="date_format" value=""/>
  <parameter key="time_zone" value="SYSTEM"/>
  <parameter key="locale" value="English (United States)"/>
  <parameter key="read_all_values_as_polynomial" value="false"/>
  <list key="data_set_meta_data_information">
    <parameter key="0" value="ID.true.polynomial.id"/>
    <parameter key="1" value="X1.true.integer.attribute"/>
    <parameter key="2" value="X2.true.integer.attribute"/>
    <parameter key="3" value="X3.true.integer.attribute"/>
    <parameter key="4" value="X4.true.integer.attribute"/>
    <parameter key="5" value="X5.true.integer.attribute"/>
    <parameter key="6" value="X6.true.integer.attribute"/>
    <parameter key="7" value="X7.true.integer.attribute"/>
    <parameter key="8" value="lucro.true.real.label"/>
  </list>
  <parameter key="read_not_matching_values_as_missings" value="false"/>
  <parameter key="datamanagement" value="double_array"/>
  <parameter key="data_management" value="auto"/>
</operator>
<operator activated="true" class="read_excel" compatibility="9.8.000" expanded="true" height="68" name="80x" width="90" x="246"
y="391">
  <parameter key="excel_file" value="Base de treinamento OE1.xlsx"/>
  <parameter key="sheet_selection" value="sheet number"/>
  <parameter key="sheet_number" value="5"/>
  <parameter key="imported_cell_range" value="A1"/>
  <parameter key="encoding" value="SYSTEM"/>
  <parameter key="first_row_as_names" value="true"/>
  <list key="annotations"/>
  <parameter key="date_format" value=""/>
  <parameter key="time_zone" value="SYSTEM"/>
  <parameter key="locale" value="English (United States)"/>

```

```

<parameter key="read_all_values_as_polynomial" value="false"/>
<list key="data_set_meta_data_information">
  <parameter key="0" value="ID.true.polynomial.id"/>
  <parameter key="1" value="X1.true.integer.attribute"/>
  <parameter key="2" value="X2.true.integer.attribute"/>
  <parameter key="3" value="X3.true.integer.attribute"/>
  <parameter key="4" value="X4.true.integer.attribute"/>
  <parameter key="5" value="X5.true.integer.attribute"/>
  <parameter key="6" value="X6.true.integer.attribute"/>
  <parameter key="7" value="X7.true.integer.attribute"/>
  <parameter key="8" value="lucro.true.real.label"/>
</list>
<parameter key="read_not_matching_values_as_missings" value="false"/>
<parameter key="datamanagement" value="double_array"/>
<parameter key="data_management" value="auto"/>
</operator>
<operator activated="true" class="read_excel" compatibility="9.8.000" expanded="true" height="68" name="100x (2)" width="90"
x="246" y="493">
  <parameter key="excel_file" value="Base de treinamento OE1.xlsx"/>
  <parameter key="sheet_selection" value="sheet number"/>
  <parameter key="sheet_number" value="6"/>
  <parameter key="imported_cell_range" value="A1"/>
  <parameter key="encoding" value="SYSTEM"/>
  <parameter key="first_row_as_names" value="true"/>
  <list key="annotations"/>
  <parameter key="date_format" value=""/>
  <parameter key="time_zone" value="SYSTEM"/>
  <parameter key="locale" value="English (United States)"/>
  <parameter key="read_all_values_as_polynomial" value="false"/>
  <list key="data_set_meta_data_information">
    <parameter key="0" value="ID.true.polynomial.id"/>
    <parameter key="1" value="X1.true.integer.attribute"/>
    <parameter key="2" value="X2.true.integer.attribute"/>
    <parameter key="3" value="X3.true.integer.attribute"/>
    <parameter key="4" value="X4.true.integer.attribute"/>
    <parameter key="5" value="X5.true.integer.attribute"/>
    <parameter key="6" value="X6.true.integer.attribute"/>
    <parameter key="7" value="X7.true.integer.attribute"/>
    <parameter key="8" value="lucro.true.real.label"/>
  </list>
  <parameter key="read_not_matching_values_as_missings" value="false"/>
  <parameter key="datamanagement" value="double_array"/>
  <parameter key="data_management" value="auto"/>
</operator>
<operator activated="true" class="read_excel" compatibility="9.8.000" expanded="true" height="68" name="Aleatório_100x"
width="90" x="246" y="595">
  <parameter key="excel_file" value="Base de treinamento OE1.xlsx"/>
  <parameter key="sheet_selection" value="sheet number"/>
  <parameter key="sheet_number" value="7"/>
  <parameter key="imported_cell_range" value="A1:I10485776"/>
  <parameter key="encoding" value="SYSTEM"/>
  <parameter key="first_row_as_names" value="true"/>
  <list key="annotations"/>
  <parameter key="date_format" value=""/>
  <parameter key="time_zone" value="SYSTEM"/>
  <parameter key="locale" value="English (United States)"/>
  <parameter key="read_all_values_as_polynomial" value="false"/>
  <list key="data_set_meta_data_information">
    <parameter key="0" value="n.true.integer.id"/>
    <parameter key="1" value="X1.true.integer.attribute"/>
    <parameter key="2" value="X2.true.integer.attribute"/>
    <parameter key="3" value="X3.true.integer.attribute"/>
    <parameter key="4" value="X4.true.integer.attribute"/>
    <parameter key="5" value="X5.true.integer.attribute"/>
    <parameter key="6" value="X6.true.integer.attribute"/>
    <parameter key="7" value="X7.true.integer.attribute"/>
    <parameter key="8" value="lucro.true.real.label"/>
  </list>
  <parameter key="read_not_matching_values_as_missings" value="false"/>
  <parameter key="datamanagement" value="double_array"/>
  <parameter key="data_management" value="auto"/>
</operator>
<connect from_op="20x (2)" from_port="output" to_port="out 1"/>
<connect from_op="40x (2)" from_port="output" to_port="out 2"/>
<connect from_op="60x" from_port="output" to_port="out 3"/>
<connect from_op="80x" from_port="output" to_port="out 4"/>

```

```

<connect from_op="100x (2)" from_port="output" to_port="out 5"/>
<connect from_op="Aleatório_100x" from_port="output" to_port="out 6"/>
<portSpacing port="source_in 1" spacing="0"/>
<portSpacing port="sink_out 1" spacing="0"/>
<portSpacing port="sink_out 2" spacing="0"/>
<portSpacing port="sink_out 3" spacing="0"/>
<portSpacing port="sink_out 4" spacing="0"/>
<portSpacing port="sink_out 5" spacing="0"/>
<portSpacing port="sink_out 6" spacing="0"/>
<portSpacing port="sink_out 7" spacing="0"/>
</process>
</operator>
<operator activated="true" class="subprocess" compatibility="9.8.000" expanded="true" height="166" name="Learning" width="90"
x="313" y="238">
  <process expanded="true">
    <operator activated="false" class="optimize_parameters_evolutionary" compatibility="9.8.000" expanded="true" height="166"
name="Optimize SVM" width="90" x="246" y="34">
      <list key="parameters">
        <parameter key="SVM.C" value="[18.30;72.68]"/>
        <parameter key="SVM.epsilon" value="[0.0;1.88]"/>
        <parameter key="SVM.kernel_gamma" value="[0.0224;0.1294]"/>
      </list>
      <parameter key="error_handling" value="fail on error"/>
      <parameter key="max_generations" value="50"/>
      <parameter key="use_early_stopping" value="true"/>
      <parameter key="generations_without_improval" value="2"/>
      <parameter key="specify_population_size" value="true"/>
      <parameter key="population_size" value="5"/>
      <parameter key="keep_best" value="true"/>
      <parameter key="mutation_type" value="gaussian_mutation"/>
      <parameter key="selection_type" value="tournament"/>
      <parameter key="tournament_fraction" value="0.25"/>
      <parameter key="crossover_prob" value="0.9"/>
      <parameter key="use_local_random_seed" value="false"/>
      <parameter key="local_random_seed" value="1992"/>
      <parameter key="show_convergence_plot" value="false"/>
      <process expanded="true">
        <operator activated="true" class="concurrency:cross_validation" compatibility="9.8.000" expanded="true" height="145"
name="Cross Validation" width="90" x="179" y="34">
          <parameter key="split_on_batch_attribute" value="false"/>
          <parameter key="leave_one_out" value="false"/>
          <parameter key="number_of_folds" value="10"/>
          <parameter key="sampling_type" value="automatic"/>
          <parameter key="use_local_random_seed" value="false"/>
          <parameter key="local_random_seed" value="1992"/>
          <parameter key="enable_parallel_execution" value="true"/>
          <process expanded="true">
            <operator activated="true" class="support_vector_machine" compatibility="9.8.000" expanded="true" height="124"
name="SVM" width="90" x="112" y="34">
              <parameter key="kernel_type" value="radial"/>
              <parameter key="kernel_gamma" value="0.08976590503770446"/>
              <parameter key="kernel_sigma1" value="1.0"/>
              <parameter key="kernel_sigma2" value="0.0"/>
              <parameter key="kernel_sigma3" value="2.0"/>
              <parameter key="kernel_shift" value="1.0"/>
              <parameter key="kernel_degree" value="2.0"/>
              <parameter key="kernel_a" value="1.0"/>
              <parameter key="kernel_b" value="0.0"/>
              <parameter key="kernel_cache" value="200"/>
              <parameter key="C" value="52.19937630105207"/>
              <parameter key="convergence_epsilon" value="0.001"/>
              <parameter key="max_iterations" value="100000"/>
              <parameter key="scale" value="true"/>
              <parameter key="calculate_weights" value="true"/>
              <parameter key="return_optimization_performance" value="true"/>
              <parameter key="L_pos" value="1.0"/>
              <parameter key="L_neg" value="1.0"/>
              <parameter key="epsilon" value="1.6519903508043345"/>
              <parameter key="epsilon_plus" value="0.0"/>
              <parameter key="epsilon_minus" value="0.0"/>
              <parameter key="balance_cost" value="false"/>
              <parameter key="quadratic_loss_pos" value="false"/>
              <parameter key="quadratic_loss_neg" value="false"/>
              <parameter key="estimate_performance" value="false"/>
            </operator>
            <connect from_port="training set" to_op="SVM" to_port="training set"/>
          </process>
        </operator>
      </process>
    </operator>
  </operator>

```

```

    <connect from_op="SVM" from_port="model" to_port="model"/>
    <portSpacing port="source_training set" spacing="0"/>
    <portSpacing port="sink_model" spacing="0"/>
    <portSpacing port="sink_through 1" spacing="0"/>
  </process>
  <process expanded="true">
    <operator activated="true" class="apply_model" compatibility="9.8.000" expanded="true" height="82" name="Apply Model (2)"
width="90" x="45" y="34">
      <list key="application_parameters">
        <parameter key="create_view" value="false"/>
      </operator>
      <operator activated="true" class="performance_regression" compatibility="9.8.000" expanded="true" height="82"
name="Performance" width="90" x="179" y="34">
        <parameter key="main_criterion" value="first"/>
        <parameter key="root_mean_squared_error" value="true"/>
        <parameter key="absolute_error" value="true"/>
        <parameter key="relative_error" value="true"/>
        <parameter key="relative_error_lenient" value="false"/>
        <parameter key="relative_error_strict" value="false"/>
        <parameter key="normalized_absolute_error" value="true"/>
        <parameter key="root_relative_squared_error" value="true"/>
        <parameter key="squared_error" value="true"/>
        <parameter key="correlation" value="true"/>
        <parameter key="squared_correlation" value="true"/>
        <parameter key="prediction_average" value="false"/>
        <parameter key="spearman_rho" value="false"/>
        <parameter key="kendall_tau" value="false"/>
        <parameter key="skip_undefined_labels" value="true"/>
        <parameter key="use_example_weights" value="true"/>
      </operator>
      <connect from_port="model" to_op="Apply Model (2)" to_port="model"/>
      <connect from_port="test set" to_op="Apply Model (2)" to_port="unlabelled data"/>
      <connect from_op="Apply Model (2)" from_port="labelled data" to_op="Performance" to_port="labelled data"/>
      <connect from_op="Performance" from_port="performance" to_port="performance 1"/>
      <connect from_op="Performance" from_port="example set" to_port="test set results"/>
      <portSpacing port="source_model" spacing="0"/>
      <portSpacing port="source_test set" spacing="0"/>
      <portSpacing port="source_through 1" spacing="0"/>
      <portSpacing port="sink_test set results" spacing="0"/>
      <portSpacing port="sink_performance 1" spacing="0"/>
      <portSpacing port="sink_performance 2" spacing="0"/>
    </process>
  </operator>
  <connect from_port="input 1" to_op="Cross Validation" to_port="example set"/>
  <connect from_op="Cross Validation" from_port="model" to_port="result 1"/>
  <connect from_op="Cross Validation" from_port="example set" to_port="result 2"/>
  <connect from_op="Cross Validation" from_port="test result set" to_port="result 3"/>
  <connect from_op="Cross Validation" from_port="performance 1" to_port="performance"/>
  <portSpacing port="source_input 1" spacing="0"/>
  <portSpacing port="source_input 2" spacing="0"/>
  <portSpacing port="sink_performance" spacing="0"/>
  <portSpacing port="sink_result 1" spacing="0"/>
  <portSpacing port="sink_result 2" spacing="0"/>
  <portSpacing port="sink_result 3" spacing="0"/>
  <portSpacing port="sink_result 4" spacing="0"/>
</process>
</operator>
<operator activated="false" class="optimize_parameters_evolutionary" compatibility="9.8.000" expanded="true" height="166"
name="Optimize ANN" width="90" x="246" y="238">
  <list key="parameters">
    <parameter key="Extract Macro - Layer 1.example_index" value="[1;64]"/>
    <parameter key="Neural Net.learning_rate" value="[0.01;0.4]"/>
    <parameter key="Neural Net.training_cycles" value="[200;1500]"/>
  </list>
  <parameter key="error_handling" value="ignore error"/>
  <parameter key="max_generations" value="50"/>
  <parameter key="use_early_stopping" value="true"/>
  <parameter key="generations_without_improval" value="2"/>
  <parameter key="specify_population_size" value="true"/>
  <parameter key="population_size" value="5"/>
  <parameter key="keep_best" value="true"/>
  <parameter key="mutation_type" value="gaussian_mutation"/>
  <parameter key="selection_type" value="tournament"/>
  <parameter key="tournament_fraction" value="0.25"/>
  <parameter key="crossover_prob" value="0.9"/>
  <parameter key="use_local_random_seed" value="false"/>

```

```

<parameter key="local_random_seed" value="1992"/>
<parameter key="show_convergence_plot" value="false"/>
<process expanded="true">
  <operator activated="true" class="read_excel" compatibility="8.1.000" expanded="true" height="68" name="Read Excel"
width="90" x="45" y="238">
  <parameter key="excel_file" value="layer_size.xlsx"/>
  <parameter key="sheet_selection" value="sheet number"/>
  <parameter key="sheet_number" value="1"/>
  <parameter key="imported_cell_range" value="A1"/>
  <parameter key="encoding" value="SYSTEM"/>
  <parameter key="first_row_as_names" value="true"/>
  <list key="annotations">
    <parameter key="0" value="Name"/>
  </list>
  <parameter key="date_format" value=""/>
  <parameter key="time_zone" value="SYSTEM"/>
  <parameter key="locale" value="English (United States)"/>
  <parameter key="read_all_values_as_polynomial" value="false"/>
  <list key="data_set_meta_data_information">
    <parameter key="0" value="LayerSize.true.integer.attribute"/>
  </list>
  <parameter key="read_not_matching_values_as_missings" value="false"/>
  <parameter key="datamanagement" value="double_array"/>
  <parameter key="data_management" value="auto"/>
</operator>
  <operator activated="true" class="extract_macro" compatibility="9.8.000" expanded="true" height="68" name="Extract Macro -
Layer 1" width="90" x="179" y="238">
  <parameter key="macro" value="layer1"/>
  <parameter key="macro_type" value="data_value"/>
  <parameter key="statistics" value="average"/>
  <parameter key="attribute_name" value="LayerSize"/>
  <parameter key="example_index" value="60"/>
  <list key="additional_macros"/>
</operator>
  <operator activated="true" class="extract_macro" compatibility="9.8.000" expanded="true" height="68" name="Extract Macro -
Layer 2" width="90" x="313" y="238">
  <parameter key="macro" value="layer2"/>
  <parameter key="macro_type" value="data_value"/>
  <parameter key="statistics" value="average"/>
  <parameter key="attribute_name" value="LayerSize"/>
  <parameter key="example_index" value="12"/>
  <list key="additional_macros"/>
</operator>
  <operator activated="false" class="extract_macro" compatibility="9.8.000" expanded="true" height="68" name="Extract Macro -
Layer 3" width="90" x="447" y="238">
  <parameter key="macro" value="layer3"/>
  <parameter key="macro_type" value="data_value"/>
  <parameter key="statistics" value="average"/>
  <parameter key="attribute_name" value="LayerSize"/>
  <parameter key="example_index" value="9"/>
  <list key="additional_macros"/>
</operator>
  <operator activated="false" class="extract_macro" compatibility="9.8.000" expanded="true" height="68" name="Extract Macro -
Layer 4" width="90" x="581" y="238">
  <parameter key="macro" value="layer4"/>
  <parameter key="macro_type" value="data_value"/>
  <parameter key="statistics" value="average"/>
  <parameter key="attribute_name" value="LayerSize"/>
  <parameter key="example_index" value="14"/>
  <list key="additional_macros"/>
</operator>
  <operator activated="true" class="concurrency:cross_validation" compatibility="9.8.000" expanded="true" height="145"
name="Cross Validation (2)" width="90" x="447" y="34">
  <parameter key="split_on_batch_attribute" value="false"/>
  <parameter key="leave_one_out" value="false"/>
  <parameter key="number_of_folds" value="10"/>
  <parameter key="sampling_type" value="automatic"/>
  <parameter key="use_local_random_seed" value="false"/>
  <parameter key="local_random_seed" value="1992"/>
  <parameter key="enable_parallel_execution" value="true"/>
</process expanded="true">
  <operator activated="true" class="neural_net" compatibility="9.8.000" expanded="true" height="82" name="Neural Net"
width="90" x="179" y="34">
  <list key="hidden_layers">
    <parameter key="hd1" value="% {layer1 }"/>
  </list>

```

```

<parameter key="training_cycles" value="1100"/>
<parameter key="learning_rate" value="0.11544755211599754"/>
<parameter key="momentum" value="0.1"/>
<parameter key="decay" value="false"/>
<parameter key="shuffle" value="true"/>
<parameter key="normalize" value="true"/>
<parameter key="error_epsilon" value="1.0E-4"/>
<parameter key="use_local_random_seed" value="false"/>
<parameter key="local_random_seed" value="1992"/>
</operator>
<connect from_port="training set" to_op="Neural Net" to_port="training set"/>
<connect from_op="Neural Net" from_port="model" to_port="model"/>
<portSpacing port="source_training set" spacing="0"/>
<portSpacing port="sink_model" spacing="0"/>
<portSpacing port="sink_through 1" spacing="0"/>
</process>
<process expanded="true">
<operator activated="true" class="apply_model" compatibility="9.8.000" expanded="true" height="82" name="Apply Model (3)"
width="90" x="112" y="34">
<list key="application_parameters"/>
<parameter key="create_view" value="false"/>
</operator>
<operator activated="true" class="performance_regression" compatibility="9.8.000" expanded="true" height="82"
name="Performance (2)" width="90" x="246" y="34">
<parameter key="main_criterion" value="root_mean_squared_error"/>
<parameter key="root_mean_squared_error" value="true"/>
<parameter key="absolute_error" value="true"/>
<parameter key="relative_error" value="true"/>
<parameter key="relative_error_lenient" value="false"/>
<parameter key="relative_error_strict" value="false"/>
<parameter key="normalized_absolute_error" value="false"/>
<parameter key="root_relative_squared_error" value="false"/>
<parameter key="squared_error" value="true"/>
<parameter key="correlation" value="true"/>
<parameter key="squared_correlation" value="true"/>
<parameter key="prediction_average" value="false"/>
<parameter key="spearman_rho" value="false"/>
<parameter key="kendall_tau" value="false"/>
<parameter key="skip_undefined_labels" value="true"/>
<parameter key="use_example_weights" value="true"/>
</operator>
<connect from_port="model" to_op="Apply Model (3)" to_port="model"/>
<connect from_port="test set" to_op="Apply Model (3)" to_port="unlabelled data"/>
<connect from_op="Apply Model (3)" from_port="labelled data" to_op="Performance (2)" to_port="labelled data"/>
<connect from_op="Performance (2)" from_port="performance" to_port="performance 1"/>
<connect from_op="Performance (2)" from_port="example set" to_port="test set results"/>
<portSpacing port="source_model" spacing="0"/>
<portSpacing port="source_test set" spacing="0"/>
<portSpacing port="source_through 1" spacing="0"/>
<portSpacing port="sink_test set results" spacing="0"/>
<portSpacing port="sink_performance 1" spacing="0"/>
<portSpacing port="sink_performance 2" spacing="0"/>
<description align="center" color="yellow" colored="false" height="105" resized="false" width="180" x="237" y="463">Type
your comment</description>
</process>
</operator>
<connect from_port="input 1" to_op="Cross Validation (2)" to_port="example set"/>
<connect from_op="Read Excel" from_port="output" to_op="Extract Macro - Layer 1" to_port="example set"/>
<connect from_op="Extract Macro - Layer 1" from_port="example set" to_op="Extract Macro - Layer 2" to_port="example set"/>
<connect from_op="Extract Macro - Layer 2" from_port="example set" to_op="Extract Macro - Layer 3" to_port="example set"/>
<connect from_op="Extract Macro - Layer 3" from_port="example set" to_op="Extract Macro - Layer 4" to_port="example set"/>
<connect from_op="Cross Validation (2)" from_port="model" to_port="result 1"/>
<connect from_op="Cross Validation (2)" from_port="example set" to_port="result 2"/>
<connect from_op="Cross Validation (2)" from_port="test result set" to_port="result 3"/>
<connect from_op="Cross Validation (2)" from_port="performance 1" to_port="performance"/>
<portSpacing port="source_input 1" spacing="0"/>
<portSpacing port="source_input 2" spacing="0"/>
<portSpacing port="sink_performance" spacing="0"/>
<portSpacing port="sink_result 1" spacing="0"/>
<portSpacing port="sink_result 2" spacing="0"/>
<portSpacing port="sink_result 3" spacing="0"/>
<portSpacing port="sink_result 4" spacing="0"/>
</process>
</operator>
<operator activated="true" class="optimize_parameters_evolutionary" compatibility="9.8.000" expanded="true" height="166"
name="Optimize GBT" width="90" x="246" y="442">

```

```

<list key="parameters">
  <parameter key="Gradient Boosted Trees.number_of_trees" value="[1.0;100]"/>
  <parameter key="Gradient Boosted Trees.learning_rate" value="[0.05;0.3]"/>
  <parameter key="Gradient Boosted Trees.min_rows" value="[0.12;0.5]"/>
</list>
<parameter key="error_handling" value="fail on error"/>
<parameter key="max_generations" value="50"/>
<parameter key="use_early_stopping" value="false"/>
<parameter key="generations_without_improval" value="2"/>
<parameter key="specify_population_size" value="true"/>
<parameter key="population_size" value="5"/>
<parameter key="keep_best" value="true"/>
<parameter key="mutation_type" value="gaussian_mutation"/>
<parameter key="selection_type" value="tournament"/>
<parameter key="tournament_fraction" value="0.25"/>
<parameter key="crossover_prob" value="0.9"/>
<parameter key="use_local_random_seed" value="false"/>
<parameter key="local_random_seed" value="1992"/>
<parameter key="show_convergence_plot" value="false"/>
<process expanded="true">
  <operator activated="true" class="concurrency:cross_validation" compatibility="9.8.000" expanded="true" height="145"
name="Cross Validation (3)" width="90" x="179" y="34">
  <parameter key="split_on_batch_attribute" value="false"/>
  <parameter key="leave_one_out" value="false"/>
  <parameter key="number_of_folds" value="10"/>
  <parameter key="sampling_type" value="automatic"/>
  <parameter key="use_local_random_seed" value="false"/>
  <parameter key="local_random_seed" value="1992"/>
  <parameter key="enable_parallel_execution" value="true"/>
  <process expanded="true">
    <operator activated="true" class="h2o:gradient_boosted_trees" compatibility="9.8.000" expanded="true" height="103"
name="Gradient Boosted Trees" width="90" x="112" y="34">
      <parameter key="number_of_trees" value="63"/>
      <parameter key="reproducible" value="false"/>
      <parameter key="maximum_number_of_threads" value="4"/>
      <parameter key="use_local_random_seed" value="false"/>
      <parameter key="local_random_seed" value="1992"/>
      <parameter key="maximal_depth" value="2"/>
      <parameter key="min_rows" value="0.3399025675576721"/>
      <parameter key="min_split_improvement" value="1.0E-5"/>
      <parameter key="number_of_bins" value="20"/>
      <parameter key="learning_rate" value="0.25677029122716966"/>
      <parameter key="sample_rate" value="1.0"/>
      <parameter key="distribution" value="AUTO"/>
      <parameter key="early_stopping" value="false"/>
      <parameter key="stopping_rounds" value="1"/>
      <parameter key="stopping_metric" value="AUTO"/>
      <parameter key="stopping_tolerance" value="0.001"/>
      <list key="monotone_constraints"/>
      <parameter key="max_runtime_seconds" value="0"/>
      <list key="expert_parameters"/>
    </operator>
    <connect from_port="training set" to_op="Gradient Boosted Trees" to_port="training set"/>
    <connect from_op="Gradient Boosted Trees" from_port="model" to_port="model"/>
    <portSpacing port="source_training set" spacing="0"/>
    <portSpacing port="sink_model" spacing="0"/>
    <portSpacing port="sink_through 1" spacing="0"/>
  </process>
  <process expanded="true">
    <operator activated="true" class="apply_model" compatibility="9.8.000" expanded="true" height="82" name="Apply Model (4)"
width="90" x="45" y="34">
      <list key="application_parameters"/>
      <parameter key="create_view" value="false"/>
    </operator>
    <operator activated="true" class="performance_regression" compatibility="9.8.000" expanded="true" height="82"
name="Performance (3)" width="90" x="179" y="34">
      <parameter key="main_criterion" value="first"/>
      <parameter key="root_mean_squared_error" value="true"/>
      <parameter key="absolute_error" value="true"/>
      <parameter key="relative_error" value="true"/>
      <parameter key="relative_error_lenient" value="false"/>
      <parameter key="relative_error_strict" value="false"/>
      <parameter key="normalized_absolute_error" value="false"/>
      <parameter key="root_relative_squared_error" value="false"/>
      <parameter key="squared_error" value="true"/>
      <parameter key="correlation" value="true"/>
    </operator>
  </process>
</process>

```

```

    <parameter key="squared_correlation" value="true"/>
    <parameter key="prediction_average" value="false"/>
    <parameter key="spearman_rho" value="false"/>
    <parameter key="kendall_tau" value="false"/>
    <parameter key="skip_undefined_labels" value="true"/>
    <parameter key="use_example_weights" value="true"/>
  </operator>
  <connect from_port="model" to_op="Apply Model (4)" to_port="model"/>
  <connect from_port="test set" to_op="Apply Model (4)" to_port="unlabelled data"/>
  <connect from_op="Apply Model (4)" from_port="labelled data" to_op="Performance (3)" to_port="labelled data"/>
  <connect from_op="Performance (3)" from_port="performance" to_port="performance 1"/>
  <connect from_op="Performance (3)" from_port="example set" to_port="test set results"/>
  <portSpacing port="source_model" spacing="0"/>
  <portSpacing port="source_test set" spacing="0"/>
  <portSpacing port="source_through 1" spacing="0"/>
  <portSpacing port="sink_test set results" spacing="0"/>
  <portSpacing port="sink_performance 1" spacing="0"/>
  <portSpacing port="sink_performance 2" spacing="0"/>
</process>
</operator>
<connect from_port="input 1" to_op="Cross Validation (3)" to_port="example set"/>
<connect from_op="Cross Validation (3)" from_port="model" to_port="result 1"/>
<connect from_op="Cross Validation (3)" from_port="example set" to_port="result 2"/>
<connect from_op="Cross Validation (3)" from_port="test result set" to_port="result 3"/>
<connect from_op="Cross Validation (3)" from_port="performance 1" to_port="performance"/>
<portSpacing port="source_input 1" spacing="0"/>
<portSpacing port="source_input 2" spacing="0"/>
<portSpacing port="sink_performance" spacing="0"/>
<portSpacing port="sink_result 1" spacing="0"/>
<portSpacing port="sink_result 2" spacing="0"/>
<portSpacing port="sink_result 3" spacing="0"/>
<portSpacing port="sink_result 4" spacing="0"/>
</process>
</operator>
<operator activated="false" class="optimize_parameters_evolutionary" compatibility="9.8.000" expanded="true" height="166"
name="Optimize RF" width="90" x="246" y="646">
  <list key="parameters">
    <parameter key="Random Forest.number_of_trees" value="[10;50]"/>
    <parameter key="Random Forest.minimal_leaf_size" value="[2;4]"/>
  </list>
  <parameter key="error_handling" value="fail on error"/>
  <parameter key="max_generations" value="50"/>
  <parameter key="use_early_stopping" value="false"/>
  <parameter key="generations_without_improval" value="2"/>
  <parameter key="specify_population_size" value="true"/>
  <parameter key="population_size" value="5"/>
  <parameter key="keep_best" value="true"/>
  <parameter key="mutation_type" value="gaussian_mutation"/>
  <parameter key="selection_type" value="tournament"/>
  <parameter key="tournament_fraction" value="0.25"/>
  <parameter key="crossover_prob" value="0.9"/>
  <parameter key="use_local_random_seed" value="false"/>
  <parameter key="local_random_seed" value="1992"/>
  <parameter key="show_convergence_plot" value="false"/>
  <process expanded="true">
    <operator activated="true" class="concurrency:cross_validation" compatibility="9.8.000" expanded="true" height="145"
name="Cross Validation (4)" width="90" x="179" y="34">
      <parameter key="split_on_batch_attribute" value="false"/>
      <parameter key="leave_one_out" value="false"/>
      <parameter key="number_of_folds" value="10"/>
      <parameter key="sampling_type" value="automatic"/>
      <parameter key="use_local_random_seed" value="false"/>
      <parameter key="local_random_seed" value="1992"/>
      <parameter key="enable_parallel_execution" value="true"/>
    </process expanded="true">
    <operator activated="true" class="concurrency:parallel_random_forest" compatibility="9.8.000" expanded="true" height="103"
name="Random Forest" width="90" x="45" y="34">
      <parameter key="number_of_trees" value="33"/>
      <parameter key="criterion" value="least_square"/>
      <parameter key="maximal_depth" value="10"/>
      <parameter key="apply_pruning" value="false"/>
      <parameter key="confidence" value="0.1"/>
      <parameter key="apply_prepruning" value="false"/>
      <parameter key="minimal_gain" value="0.01"/>
      <parameter key="minimal_leaf_size" value="2"/>
      <parameter key="minimal_size_for_split" value="4"/>
    </operator>
  </process>
</operator>

```

```

    <parameter key="number_of_prepruning_alternatives" value="3"/>
    <parameter key="random_splits" value="false"/>
    <parameter key="guess_subset_ratio" value="true"/>
    <parameter key="subset_ratio" value="0.2"/>
    <parameter key="voting_strategy" value="confidence vote"/>
    <parameter key="use_local_random_seed" value="false"/>
    <parameter key="local_random_seed" value="1992"/>
    <parameter key="enable_parallel_execution" value="true"/>
  </operator>
  <connect from_port="training set" to_op="Random Forest" to_port="training set"/>
  <connect from_op="Random Forest" from_port="model" to_port="model"/>
  <portSpacing port="source_training set" spacing="0"/>
  <portSpacing port="sink_model" spacing="0"/>
  <portSpacing port="sink_through 1" spacing="0"/>
</process>
<process expanded="true">
  <operator activated="true" class="apply_model" compatibility="9.8.000" expanded="true" height="82" name="Apply Model (5)"
width="90" x="45" y="34">
    <list key="application_parameters"/>
    <parameter key="create_view" value="false"/>
  </operator>
  <operator activated="true" class="performance_regression" compatibility="9.8.000" expanded="true" height="82"
name="Performance (4)" width="90" x="179" y="34">
    <parameter key="main_criterion" value="first"/>
    <parameter key="root_mean_squared_error" value="true"/>
    <parameter key="absolute_error" value="true"/>
    <parameter key="relative_error" value="true"/>
    <parameter key="relative_error_lenient" value="false"/>
    <parameter key="relative_error_strict" value="false"/>
    <parameter key="normalized_absolute_error" value="false"/>
    <parameter key="root_relative_squared_error" value="false"/>
    <parameter key="squared_error" value="false"/>
    <parameter key="correlation" value="true"/>
    <parameter key="squared_correlation" value="true"/>
    <parameter key="prediction_average" value="false"/>
    <parameter key="spearman_rho" value="false"/>
    <parameter key="kendall_tau" value="false"/>
    <parameter key="skip_undefined_labels" value="true"/>
    <parameter key="use_example_weights" value="true"/>
  </operator>
  <connect from_port="model" to_op="Apply Model (5)" to_port="model"/>
  <connect from_op="test set" to_op="Apply Model (5)" to_port="unlabelled data"/>
  <connect from_op="Apply Model (5)" from_port="labelled data" to_op="Performance (4)" to_port="labelled data"/>
  <connect from_op="Performance (4)" from_port="performance" to_port="performance 1"/>
  <connect from_op="Performance (4)" from_port="example set" to_port="test set results"/>
  <portSpacing port="source_model" spacing="0"/>
  <portSpacing port="source_test set" spacing="0"/>
  <portSpacing port="source_through 1" spacing="0"/>
  <portSpacing port="sink_test set results" spacing="0"/>
  <portSpacing port="sink_performance 1" spacing="0"/>
  <portSpacing port="sink_performance 2" spacing="0"/>
</process>
</operator>
  <connect from_port="input 1" to_op="Cross Validation (4)" to_port="example set"/>
  <connect from_op="Cross Validation (4)" from_port="model" to_port="result 1"/>
  <connect from_op="Cross Validation (4)" from_port="example set" to_port="result 2"/>
  <connect from_op="Cross Validation (4)" from_port="test result set" to_port="result 3"/>
  <connect from_op="Cross Validation (4)" from_port="performance 1" to_port="performance"/>
  <portSpacing port="source_input 1" spacing="0"/>
  <portSpacing port="source_input 2" spacing="0"/>
  <portSpacing port="sink_performance" spacing="0"/>
  <portSpacing port="sink_result 1" spacing="0"/>
  <portSpacing port="sink_result 2" spacing="0"/>
  <portSpacing port="sink_result 3" spacing="0"/>
  <portSpacing port="sink_result 4" spacing="0"/>
</process>
</operator>
  <operator activated="false" class="optimize_parameters_evolutionary" compatibility="9.8.000" expanded="true" height="166"
name="Optimize GP" width="90" x="246" y="850">
    <list key="parameters">
      <parameter key="Gaussian Process.kernel_lengthscale" value="[1;100]"/>
      <parameter key="Gaussian Process.max_basis_vectors" value="[38;47]"/>
      <parameter key="Gaussian Process.geometrical_tol" value="[0.0;0.3]"/>
      <parameter key="Gaussian Process.epsilon_tol" value="[0.000001;0.00001]"/>
    </list>
    <parameter key="error_handling" value="fail on error"/>
  </operator>

```

```

<parameter key="max_generations" value="50"/>
<parameter key="use_early_stopping" value="false"/>
<parameter key="generations_without_improval" value="2"/>
<parameter key="specify_population_size" value="true"/>
<parameter key="population_size" value="5"/>
<parameter key="keep_best" value="true"/>
<parameter key="mutation_type" value="gaussian_mutation"/>
<parameter key="selection_type" value="tournament"/>
<parameter key="tournament_fraction" value="0.25"/>
<parameter key="crossover_prob" value="0.9"/>
<parameter key="use_local_random_seed" value="false"/>
<parameter key="local_random_seed" value="1992"/>
<parameter key="show_convergence_plot" value="false"/>
<process expanded="true">
  <operator activated="true" class="concurrency:cross_validation" compatibility="9.8.000" expanded="true" height="145"
name="Cross Validation (5)" width="90" x="179" y="34">
    <parameter key="split_on_batch_attribute" value="false"/>
    <parameter key="leave_one_out" value="false"/>
    <parameter key="number_of_folds" value="10"/>
    <parameter key="sampling_type" value="automatic"/>
    <parameter key="use_local_random_seed" value="false"/>
    <parameter key="local_random_seed" value="1992"/>
    <parameter key="enable_parallel_execution" value="true"/>
    <process expanded="true">
      <operator activated="true" class="gaussian_process" compatibility="9.8.000" expanded="true" height="82" name="Gaussian
Process" width="90" x="45" y="34">
        <parameter key="kernel_type" value="rbf"/>
        <parameter key="kernel_lengthscales" value="15.06911050456206"/>
        <parameter key="kernel_degree" value="2.0"/>
        <parameter key="kernel_bias" value="1.0"/>
        <parameter key="kernel_sigma1" value="1.0"/>
        <parameter key="kernel_sigma2" value="0.0"/>
        <parameter key="kernel_sigma3" value="2.0"/>
        <parameter key="kernel_shift" value="1.0"/>
        <parameter key="kernel_a" value="1.0"/>
        <parameter key="kernel_b" value="0.0"/>
        <parameter key="max_basis_vectors" value="39"/>
        <parameter key="epsilon_tol" value="1.0E-5"/>
        <parameter key="geometrical_tol" value="8.614388370087028E-4"/>
      </operator>
      <connect from_port="training set" to_op="Gaussian Process" to_port="training set"/>
      <connect from_op="Gaussian Process" from_port="model" to_port="model"/>
      <portSpacing port="source_training set" spacing="0"/>
      <portSpacing port="sink_model" spacing="0"/>
      <portSpacing port="sink_through 1" spacing="0"/>
    </process>
  </operator>
  <process expanded="true">
    <operator activated="true" class="apply_model" compatibility="9.8.000" expanded="true" height="82" name="Apply Model (6)"
width="90" x="45" y="34">
      <list key="application_parameters"/>
      <parameter key="create_view" value="false"/>
    </operator>
    <operator activated="true" class="performance_regression" compatibility="9.8.000" expanded="true" height="82"
name="Performance (5)" width="90" x="179" y="34">
      <parameter key="main_criterion" value="first"/>
      <parameter key="root_mean_squared_error" value="true"/>
      <parameter key="absolute_error" value="true"/>
      <parameter key="relative_error" value="true"/>
      <parameter key="relative_error_lenient" value="false"/>
      <parameter key="relative_error_strict" value="false"/>
      <parameter key="normalized_absolute_error" value="false"/>
      <parameter key="root_relative_squared_error" value="false"/>
      <parameter key="squared_error" value="false"/>
      <parameter key="correlation" value="true"/>
      <parameter key="squared_correlation" value="true"/>
      <parameter key="prediction_average" value="false"/>
      <parameter key="spearman_rho" value="false"/>
      <parameter key="kendall_tau" value="false"/>
      <parameter key="skip_undefined_labels" value="true"/>
      <parameter key="use_example_weights" value="true"/>
    </operator>
    <connect from_port="model" to_op="Apply Model (6)" to_port="model"/>
    <connect from_port="test set" to_op="Apply Model (6)" to_port="unlabelled data"/>
    <connect from_op="Apply Model (6)" from_port="labelled data" to_op="Performance (5)" to_port="labelled data"/>
    <connect from_op="Performance (5)" from_port="performance" to_port="performance 1"/>
    <connect from_op="Performance (5)" from_port="example set" to_port="test set results"/>
  </process>

```

```

    <portSpacing port="source_model" spacing="0"/>
    <portSpacing port="source_test set" spacing="0"/>
    <portSpacing port="source_through 1" spacing="0"/>
    <portSpacing port="sink_test set results" spacing="0"/>
    <portSpacing port="sink_performance 1" spacing="0"/>
    <portSpacing port="sink_performance 2" spacing="0"/>
  </process>
</operator>
<connect from_port="input 1" to_op="Cross Validation (5)" to_port="example set"/>
<connect from_op="Cross Validation (5)" from_port="model" to_port="result 1"/>
<connect from_op="Cross Validation (5)" from_port="example set" to_port="result 2"/>
<connect from_op="Cross Validation (5)" from_port="test result set" to_port="result 3"/>
<connect from_op="Cross Validation (5)" from_port="performance 1" to_port="performance"/>
<portSpacing port="source_input 1" spacing="0"/>
<portSpacing port="source_input 2" spacing="0"/>
<portSpacing port="sink_performance" spacing="0"/>
<portSpacing port="sink_result 1" spacing="0"/>
<portSpacing port="sink_result 2" spacing="0"/>
<portSpacing port="sink_result 3" spacing="0"/>
<portSpacing port="sink_result 4" spacing="0"/>
</process>
</operator>
<operator activated="false" class="optimize_parameters_evolutionary" compatibility="9.8.000" expanded="true" height="166"
name="Optimize PR" width="90" x="246" y="1054">
  <list key="parameters">
    <parameter key="Polynomial Regression.max_degree" value="[1;3]"/>
  </list>
  <parameter key="error_handling" value="fail on error"/>
  <parameter key="max_generations" value="50"/>
  <parameter key="use_early_stopping" value="false"/>
  <parameter key="generations_without_improval" value="2"/>
  <parameter key="specify_population_size" value="true"/>
  <parameter key="population_size" value="5"/>
  <parameter key="keep_best" value="true"/>
  <parameter key="mutation_type" value="gaussian_mutation"/>
  <parameter key="selection_type" value="tournament"/>
  <parameter key="tournament_fraction" value="0.25"/>
  <parameter key="crossover_prob" value="0.9"/>
  <parameter key="use_local_random_seed" value="false"/>
  <parameter key="local_random_seed" value="1992"/>
  <parameter key="show_convergence_plot" value="false"/>
  <process expanded="true">
    <operator activated="true" class="concurrency:cross_validation" compatibility="9.8.000" expanded="true" height="145"
name="Cross Validation (6)" width="90" x="179" y="34">
      <parameter key="split_on_batch_attribute" value="false"/>
      <parameter key="leave_one_out" value="false"/>
      <parameter key="number_of_folds" value="10"/>
      <parameter key="sampling_type" value="automatic"/>
      <parameter key="use_local_random_seed" value="false"/>
      <parameter key="local_random_seed" value="1992"/>
      <parameter key="enable_parallel_execution" value="true"/>
      <process expanded="true">
        <operator activated="true" class="polynomial_regression" compatibility="9.8.000" expanded="true" height="82"
name="Polynomial Regression" width="90" x="112" y="34">
          <parameter key="max_iterations" value="5000"/>
          <parameter key="replication_factor" value="1"/>
          <parameter key="max_degree" value="1"/>
          <parameter key="min_coefficient" value="-100.0"/>
          <parameter key="max_coefficient" value="100.0"/>
          <parameter key="use_local_random_seed" value="false"/>
          <parameter key="local_random_seed" value="1992"/>
        </operator>
        <connect from_port="training set" to_op="Polynomial Regression" to_port="training set"/>
        <connect from_op="Polynomial Regression" from_port="model" to_port="model"/>
        <portSpacing port="source_training set" spacing="0"/>
        <portSpacing port="sink_model" spacing="0"/>
        <portSpacing port="sink_through 1" spacing="0"/>
      </process>
      <process expanded="true">
        <operator activated="true" class="apply_model" compatibility="9.8.000" expanded="true" height="82" name="Apply Model (7)"
width="90" x="45" y="34">
          <list key="application_parameters">
            <parameter key="create_view" value="false"/>
          </operator>
        <operator activated="true" class="performance_regression" compatibility="9.8.000" expanded="true" height="82"
name="Performance (6)" width="90" x="179" y="34">

```

```

<parameter key="main_criterion" value="first"/>
<parameter key="root_mean_squared_error" value="true"/>
<parameter key="absolute_error" value="true"/>
<parameter key="relative_error" value="true"/>
<parameter key="relative_error_lenient" value="false"/>
<parameter key="relative_error_strict" value="false"/>
<parameter key="normalized_absolute_error" value="false"/>
<parameter key="root_relative_squared_error" value="false"/>
<parameter key="squared_error" value="true"/>
<parameter key="correlation" value="true"/>
<parameter key="squared_correlation" value="true"/>
<parameter key="prediction_average" value="false"/>
<parameter key="spearman_rho" value="false"/>
<parameter key="kendall_tau" value="false"/>
<parameter key="skip_undefined_labels" value="true"/>
<parameter key="use_example_weights" value="true"/>
</operator>
<connect from_port="model" to_op="Apply Model (7)" to_port="model"/>
<connect from_port="test set" to_op="Apply Model (7)" to_port="unlabelled data"/>
<connect from_op="Apply Model (7)" from_port="labelled data" to_op="Performance (6)" to_port="labelled data"/>
<connect from_op="Performance (6)" from_port="performance" to_port="performance 1"/>
<connect from_op="Performance (6)" from_port="example set" to_port="test set results"/>
<portSpacing port="source_model" spacing="0"/>
<portSpacing port="source_test set" spacing="0"/>
<portSpacing port="source_through 1" spacing="0"/>
<portSpacing port="sink_test set results" spacing="0"/>
<portSpacing port="sink_performance 1" spacing="0"/>
<portSpacing port="sink_performance 2" spacing="0"/>
</process>
</operator>
<connect from_port="input 1" to_op="Cross Validation (6)" to_port="example set"/>
<connect from_op="Cross Validation (6)" from_port="model" to_port="result 1"/>
<connect from_op="Cross Validation (6)" from_port="example set" to_port="result 2"/>
<connect from_op="Cross Validation (6)" from_port="test result set" to_port="result 3"/>
<connect from_op="Cross Validation (6)" from_port="performance 1" to_port="performance"/>
<portSpacing port="source_input 1" spacing="0"/>
<portSpacing port="source_input 2" spacing="0"/>
<portSpacing port="sink_performance" spacing="0"/>
<portSpacing port="sink_result 1" spacing="0"/>
<portSpacing port="sink_result 2" spacing="0"/>
<portSpacing port="sink_result 3" spacing="0"/>
<portSpacing port="sink_result 4" spacing="0"/>
</process>
</operator>
<connect from_port="in 1" to_op="Optimize GBT" to_port="input 1"/>
<connect from_op="Optimize GBT" from_port="performance" to_port="out 1"/>
<connect from_op="Optimize GBT" from_port="parameter" to_port="out 2"/>
<connect from_op="Optimize GBT" from_port="result 1" to_port="out 3"/>
<connect from_op="Optimize GBT" from_port="result 2" to_port="out 4"/>
<connect from_op="Optimize GBT" from_port="result 3" to_port="out 5"/>
<portSpacing port="source_in 1" spacing="294"/>
<portSpacing port="source_in 2" spacing="252"/>
<portSpacing port="sink_out 1" spacing="0"/>
<portSpacing port="sink_out 2" spacing="0"/>
<portSpacing port="sink_out 3" spacing="0"/>
<portSpacing port="sink_out 4" spacing="0"/>
<portSpacing port="sink_out 5" spacing="0"/>
<portSpacing port="sink_out 6" spacing="0"/>
</process>
</operator>
<operator activated="true" class="apply_model" compatibility="9.8.000" expanded="true" height="82" name="Apply Model" width="90"
x="447" y="34">
  <list key="application_parameters"/>
  <parameter key="create_view" value="false"/>
</operator>
<connect from_op="Solution Space" from_port="output" to_op="Apply Model" to_port="unlabelled data"/>
<connect from_op="Training Database (2)" from_port="out 1" to_op="Learning" to_port="in 1"/>
<connect from_op="Learning" from_port="out 1" to_port="result 5"/>
<connect from_op="Learning" from_port="out 2" to_port="result 6"/>
<connect from_op="Learning" from_port="out 3" to_op="Apply Model" to_port="model"/>
<connect from_op="Learning" from_port="out 4" to_port="result 3"/>
<connect from_op="Learning" from_port="out 5" to_port="result 4"/>
<connect from_op="Apply Model" from_port="labelled data" to_port="result 1"/>
<connect from_op="Apply Model" from_port="model" to_port="result 2"/>
<portSpacing port="source_input 1" spacing="0"/>
<portSpacing port="sink_result 1" spacing="0"/>

```

```
<portSpacing port="sink_result 2" spacing="0"/>  
<portSpacing port="sink_result 3" spacing="0"/>  
<portSpacing port="sink_result 4" spacing="0"/>  
<portSpacing port="sink_result 5" spacing="0"/>  
<portSpacing port="sink_result 6" spacing="0"/>  
<portSpacing port="sink_result 7" spacing="0"/>  
</process>  
</operator>  
</process>
```

APÊNDICE - Artigos Publicados em Congressos e Periódicos

Artigos publicados em anais de congressos:

- AMARAL, J. V. S.; MONTEVECHI, J. A. B.; MIRANDA, R. C. Gradient-Boosted Trees como estratégia para metamodelagem em problemas de otimização via simulação. In: XL Encontro Nacional de Engenharia de Produção, *Anais...* Online. 2020.
- AMARAL, J. V. S.; LIMA, R. S.; MIRANDA, R. C. Modelagem e simulação na cadeia de suprimentos do petróleo: revisão sistemática de literatura. In: XL Encontro Nacional de Engenharia de Produção, *Anais...* Online. 2020.
- AMARAL, J. V. S.; LEAL, G. S.; OLIVEIRA, G. F.; MONTEVECHI, J. A. B.; BALESTRASSI, P. P. Otimização de portfólio de ações do Ibovespa: uma comparação entre métodos de programação linear e programação não linear. In: XL Encontro Nacional de Engenharia de Produção, *Anais...* Online. 2020.
- AMARAL, J. V. S.; MIRANDA, R. C.; MONTEVECHI, J. A. B. Support Vector Machine para metamodelagem em problemas de otimização via simulação. In: LII Simpósio Brasileiro de Pesquisa Operacional, *Anais...* Online. 2020.
- AMARAL, J. V. S.; MONTEVECHI, J. A. B.; Vilela, Flávio F.; NOGUEIRA, T. L.; GOMES, K. H. P. Mapeamento do fluxo de valor: desenvolvimento do estado futuro em uma indústria de capas para bancos automotivos. In: XXXIX Encontro Nacional de Engenharia de Produção, *Anais...* Santos, SP. 2019.
- AMARAL, J. V. S.; MACHADO, R. H. C.; BARROSO, L.G.C; SILVA, W. D.; MENDES, M.P. Relayout como estratégia de redução de custos: Estudo de caso em uma confecção de uniformes. In: XXXIX Encontro Nacional de Engenharia de Produção, *Anais...* Santos, SP. 2019.

Artigos submetidos para publicação em periódicos:

- AMARAL, J. V. S.; MIRANDA, R. C.; MONTEVECHI, J. A. B. Metamodeling-based simulation optimization: an approach to discrete event problems. Artigo submetido a Revista *Applied Soft Computing Journal*. Data da submissão: 21/12/2020.
- AMARAL, J. V. S.; MIRANDA, R. C.; MONTEVECHI, J. A. B. Metamodel-based simulation optimization: A systematic literature review. Artigo submetido a Revista *Simulation Modelling Practice and Theory*. Data da submissão: 30/10/2020.