

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL**

**DIMENSIONAMENTO ÓTIMO DE REDES DE
DISTRIBUIÇÃO DE ÁGUA UTILIZANDO ALGORITMOS
GENÉTICOS MULTIOBJETIVOS**

RENATA SAYURI MURANAKA

ORIENTADOR: ALEXANDRE KEPLER SOARES

**DISSERTAÇÃO DE MESTRADO EM TECNOLOGIA AMBIENTAL
E RECURSOS HÍDRICOS**

BRASÍLIA/DF: DEZEMBRO/2024

**UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA CIVIL**

**DIMENSIONAMENTO ÓTIMO DE REDES DE DISTRIBUIÇÃO DE
ÁGUA UTILIZANDO ALGORITMOS GENÉTICOS
MULTIOBJETIVOS**

RENATA SAYURI MURANAKA

**DISSERTAÇÃO SUBMETIDA AO DEPARTAMENTO DE
ENGENHARIA CIVIL E AMBIENTAL DA FACULDADE DE
TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA COMO PARTE
DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU
DE MESTRE EM TECNOLOGIA AMBIENTAL E RECURSOS
HÍDRICOS.**

APROVADA POR:

**Prof. Alexandre Kepler Soares, Dr. (ENC-UnB)
(Orientador)**

**Prof. Sergio Koide (ENC-UnB)
(Examinador Interno)**

**Prof. Klebber Teodomiro Martins Formiga (UFG)
(Examinador Externo)**

BRASÍLIA/DF, DEZEMBRO DE 2024

FICHA CATALOGRÁFICA

MURANAKA, RENATA SAYURI

Dimensionamento Ótimo de Redes de Distribuição de Água Utilizando Algoritmos Genéticos Multiobjetivos [Distrito Federal] 2024.

xi,78p., 210 x 297 mm (ENC/FT/UnB, Mestre, Tecnologia Ambiental e Recursos Hídricos, 2024).

Dissertação de Mestrado – Universidade de Brasília. Faculdade de Tecnologia.

Departamento de Engenharia Civil e Ambiental.

1. Redes de distribuição de água

2. Otimização multiobjetivo

3. Algoritmos genéticos

I. ENC/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

MURANAKA., R. S. (2024). Dimensionamento Ótimo de Redes de Distribuição de Água Utilizando Algoritmos Genéticos Multiobjetivos. Dissertação de Mestrado em Tecnologia Ambiental e Recursos Hídricos, Publicação PPGTARH. DM – 277/2024, Departamento de Engenharia Civil e Ambiental, Universidade de Brasília, Brasília, DF, 78p.

CESSÃO DE DIREITOS

AUTOR: Renata Sayuri Muranaka.

TÍTULO: Dimensionamento Ótimo de Redes de Distribuição de Água Utilizando Algoritmos Genéticos Multiobjetivos.

GRAU: Mestre

ANO: 2024

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.

Renata Sayuri Muranaka
renata.sayuri.m@gmail.com

AGRADECIMENTOS

A Deus, sempre me guiando pelos caminhos da vida com suas infinitas bênçãos.

Aos meus pais, Wilson e Neusa, por toda a dedicação e pelos esforços realizados para prover educação e conhecimento.

Ao Marcelo, meu companheiro incansável e grande incentivador nesta e em todas as minhas jornadas.

Ao Professor Alexandre, meu orientador que com sua calma e paciência me guiou durante a pesquisa.

Aos amigos e colegas do PTARH, sempre apoiando e incentivando o crescimento de todos.

Aos professores do PTARH, por compartilharem conhecimentos e experiências edificantes.

À Universidade de Brasília e ao Decanato de Pós-Graduação, pela concessão de auxílio para desenvolver este trabalho.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES, pela bolsa de mestrado, fundamental para realização do presente trabalho.

RESUMO

DIMENSIONAMENTO ÓTIMO DE REDES DE DISTRIBUIÇÃO DE ÁGUA UTILIZANDO ALGORITMOS GENÉTICOS MULTIOBJETIVOS

A busca por projetos eficientes de redes de distribuição de água (RDA) que refletem as complexidades dos sistemas reais tem estimulado o desenvolvimento e a aplicação de diversas técnicas de otimização. Entre elas, a otimização multi e de muitos objetivos possuem particular importância devido à sua capacidade de melhor representar tanto os sistemas de abastecimento como as necessidades das companhias de saneamento de forma mais realista. Neste contexto, os algoritmos genéticos surgiram como uma técnica de otimização promissora, oferecendo uma ampla gama de metodologias documentadas na literatura que obtiveram resultados satisfatórios em diversos campos de aplicação, entre eles a hidráulica e o saneamento. Para avaliar sistematicamente essas abordagens, foi desenvolvida uma metodologia para comparar seis algoritmos genéticos no contexto de otimização de RDA: NSGA-II, NSGA-III, U-NSGA-III, R-NSGA-III, MOEA/D e RVEA, utilizando duas funções objetivo distintas. A análise comparativa utilizou como métricas principais o critério de eficiência (E), a função de distribuição acumulada (FDA), análises estatísticas de coeficiente de variação e complexidade de algoritmos. As descobertas revelaram que, embora a maioria dos algoritmos tenha convergido com sucesso para o ótimo global conhecido do estudo de caso empregado, o NSGA-II e o NSGA-III exibiram desempenho superior, principalmente na minimização de custos. Estes resultados demonstram a eficácia destes algoritmos em lidar com as complexidades inerentes à otimização de RDA, distinguindo-os neste campo de aplicação.

Palavras-chave: redes de distribuição de água, otimização, multi-objetivo, muitos objetivos, algoritmos genéticos

ABSTRACT

OPTIMAL DESIGN OF WATER DISTRIBUTION NETWORKS USING MULTI-OBJECTIVE GENETIC ALGORITHMS

The pursuit of efficient water distribution network (WDN) projects that reflect the complexities of real systems has driven the development and application of various optimization techniques. Among these, multi and many-objective optimization are particularly significant due to their ability to better represent both supply systems and the needs of sanitation companies in a more realistic manner. In this context, genetic algorithms have emerged as a promising optimization technique, offering a wide range of methodologies documented in literature that have achieved satisfactory results in various application fields, including hydraulics and sanitation.

To systematically evaluate these approaches, a methodology was developed to compare six genetic algorithms in the context of WDN optimization: NSGA-II, NSGA-III, U-NSGA-III, R-NSGA-III, MOEA/D, and RVEA, using two distinct objective functions. The comparative analysis employed key metrics such as the efficiency criterion (E), the cumulative distribution function (CDF), statistical error analyses, and algorithm complexity. The findings revealed that although most algorithms successfully converged to the known global optimum from the employed case study, NSGA-II and NSGA-III demonstrated superior performance, particularly in cost minimization. These results highlight the effectiveness of these algorithms in addressing the complexities inherent to WDN optimization, setting them apart in this field of application.

Keywords: water distribution networks, optimization, multi-objective, many-objective, genetic algorithms

Sumário

1 – INTRODUÇÃO	1
1.1. ESTRUTURA DA DISSERTAÇÃO	3
2 – OBJETIVOS	4
3 – FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA ..	5
3.1 – DIMENSIONAMENTO DE SISTEMAS DE DISTRIBUIÇÃO DE ÁGUA	5
3.1.1 – Reservatórios	5
3.1.2 – Redes de distribuição de água	6
3.1.3 - Modelo hidráulico	7
3.2 – CALIBRAÇÃO DE MODELOS DE REDES DE DISTRIBUIÇÃO DE ÁGUA ...	9
3.3 – OTIMIZAÇÃO DE REDES DE DISTRIBUIÇÃO DE ÁGUA	10
3.4 – ALGORITMOS GENÉTICOS	12
3.4.1. NSGA-II	13
3.4.2. NSGA-III	15
3.4.3. U-NSGA-III	17
3.4.4. R-NSGA-III	18
3.4.5. MOEA/D	19
3.4.6. RVEA	20
3.4 – DIMENSIONAMENTO ÓTIMO DE RDA.....	24
4 – METODOLOGIA	37
4.1 – CENÁRIOS	37
4.3 – MÉTRICAS DE COMPARAÇÃO	41
5 – RESULTADOS E DISCUSSÕES	42
6 – CONCLUSÕES	64

REFERÊNCIAS BIBLIOGRÁFICAS.....	67
APÊNDICE A – ALGORITMO IMPLEMENTADO	72

LISTA DE FIGURAS

FIGURA 1. REDE DE ALPEROVITZ E SHAMIR	39
FIGURA 2. FLUXOGRAMA DE TRABALHO	41
FIGURA 3. FRENTE PARETO DO NSGA-II PARA TODAS AS SIMULAÇÕES	43
FIGURA 4. FRENTE PARETO DO NSGA-III PARA TODAS AS SIMULAÇÕES	43
FIGURA 5. FRENTE PARETO DO R-NSGA-III PARA TODAS AS SIMULAÇÕES	44
FIGURA 6. FRENTE PARETO DO U-NSGA-III PARA TODAS AS SIMULAÇÕES	44
FIGURA 7. FRENTE PARETO DO MOEA/D PARA TODAS AS SIMULAÇÕES	45
FIGURA 8. FRENTE PARETO DO RVEA PARA TODAS AS SIMULAÇÕES	45
FIGURA 9. FRENTE PARETO DO NSGA-II (1 SIMULAÇÃO)	46
FIGURA 10. FRENTE PARETO DO NSGA-III (1 SIMULAÇÃO)	47
FIGURA 11. FRENTE PARETO DO R-NSGA-III (1 SIMULAÇÃO)	47
FIGURA 12. FRENTE PARETO DO U-NSGA-III (1 SIMULAÇÃO)	48
FIGURA 13. FRENTE PARETO DO RVEA (1 SIMULAÇÃO)	48
FIGURA 14. FRENTE PARETO DO MOEA/D (1 SIMULAÇÃO)	49
FIGURA 15. FRENTE PARETO DE CUSTOS MÍNIMOS DO NSGA-II	50
FIGURA 16. FRENTE PARETO DE CUSTOS MÍNIMOS DO NSGA-III	51
FIGURA 17. FRENTE PARETO DE CUSTOS MÍNIMOS DO R-NSGA-III	51
FIGURA 18. FRENTE PARETO DE CUSTOS MÍNIMOS DO U-NSGA-III	52
FIGURA 19. FRENTE PARETO DE CUSTOS MÍNIMOS DO RVEA	52
FIGURA 20. FRENTE PARETO DE CUSTOS MÍNIMOS DO MOEAD	53
FIGURA 21. EFICIÊNCIA (E) POR ALGORITMO	54
FIGURA 22. C.O.V. ASSOCIADOS AOS CUSTOS MÍNIMOS MÉDIOS PARA A POPULAÇÃO DE 10	56
FIGURA 23. C.O.V. ASSOCIADOS AOS CUSTOS MÍNIMOS MÉDIOS PARA A POPULAÇÃO DE 100	57
FIGURA 24. C.O.V. ASSOCIADOS AOS TEMPOS MÉDIOS DE PROCESSAMENTO	60
FIGURA 25. FUNÇÃO DE DISTRIBUIÇÃO ACUMULADA DOS DIÂMETROS	62

LISTA DE TABELAS

TABELA 1. COMPARAÇÃO DOS ALGORITMOS	23
TABELA 3. SUMÁRIO DE ARTIGOS REVISADOS	34
TABELA 3. CUSTOS DAS REDES POR DIÂMETRO	40
TABELA 4. RESULTADOS DAS SIMULAÇÕES PARA O CUSTO MÍNIMO.....	59
TABELA 5. DIÂMETROS ÓTIMOS.....	60
TABELA 6. COMPLEXIDADE DOS ALGORITMOS	64

LISTA DE SÍMBOLOS, NOMENCLATURAS E ABREVIACÕES

ABNT	<i>Associação brasileira de normas técnicas</i>
AG	<i>Algoritmos genéticos</i>
FDA	<i>Função de distribuição acumulada</i>
RDA	<i>Redes de distribuição de água</i>
SDA	<i>Sistema de distribuição de água</i>

1 – INTRODUÇÃO

A proposição de projetos ótimos multiobjetivos ou a reabilitação de redes de distribuição de água (RDA) são temas de grande interesse para redução de custos das empresas de saneamento e para melhoria da infraestrutura dessas redes. O advento da inteligência artificial aplicado a problemas de RDA permitiu que projetos focados anteriormente apenas no menor custo das redes pudessem considerar outros diversos objetivos, como minimização de falhas hidráulicas, redução de perdas por vazamentos (Creaco e Pezzinga, 2015) e melhoria da qualidade da água (Farmani *et al.*, 2006). Na temática de problemas multiobjetivos, busca-se o conjunto de soluções ótimas de Pareto, em que cada solução pode ser melhor que todas as outras em pelo menos um objetivo, ou o balanço (do inglês, *trade-off*) ótimo entre todas elas, aumentando a compreensão sobre as relações entre objetivos de projeto conflitantes e conseqüentemente auxiliando no processo de tomada de decisão sobre os melhores projetos a serem implantados.

Vários métodos que utilizam mais de um objetivo foram desenvolvidos em busca das soluções ótimas que forneçam as frentes de Pareto, como por exemplo algoritmos genéticos (AG), enxames de partículas e colônias de formigas. Dentro da otimização de RDA, os AG se destacam como uma das metaheurísticas mais utilizadas e bem estabelecidas, fundamentadas nos princípios darwinianos da evolução natural (Maier *et al.*, 2014; Garzón *et al.*, 2022). Trabalhos como de Kapelan *et al.* (2003), Fu *et al.* (2013), Creaco e Pezzinga (2015) e Zheng *et al.* (2016) aplicaram os algoritmos genéticos multiobjetivos para minimização dos custos das RDA, redução de perdas através da localização ótima de válvulas de controle de pressão e calibração de modelos de redes utilizando uma localização ótima de sensores de vazão e pressão. A crescente utilização da inteligência artificial fica evidente através de trabalhos como de Rapp *et al.* (2023), que apontam que 24% das grandes companhias de saneamento dos Estados Unidos selecionadas no estudo utilizaram ou utilizam técnicas de inteligência artificial principalmente para diminuição de custos, detecção de vazamentos, melhoria na qualidade de água e aprimoramento da rede hidráulica. Adicionalmente, quando questionadas, 68% das companhias que não utilizam a inteligência artificial, afirmaram que planejam aplicá-la nos próximos cinco anos.

O uso de inteligência artificial em RDA enfrenta desafios devido às características únicas das mesmas, que incluem objetivos conflitantes, restrições hidráulicas e variáveis discretas, impondo obstáculos à otimização (Fu *et al.*, 2013). A esses desafios soma-se a vasta gama de AG disponíveis na literatura, cada qual com suas características inerentes, como função de aptidão, diversidade populacional, convergência e mecanismos de seleção, entre outros. A combinação dos desafios à grande diversidade de AG acarreta diversas incertezas, principalmente sobre o desempenho e a capacidade dos algoritmos em propor soluções aos problemas apresentados. Desta forma, a escolha do algoritmo mais eficiente a ser aplicado ao dimensionamento ótimo de projetos de RDA torna-se uma pergunta de pesquisa a ser investigada.

Neste estudo propõe-se a comparação de seis AG aplicados a um estudo de caso de RDA no intuito de avaliar os diversos conjuntos de soluções buscando elencar os melhores algoritmos. Os desempenhos dos seis AG: NSGA-II, NSGA-III, U-NSGA-III, R-NSGA-III, MOEA/D e RVEA; foram avaliados em termos de eficiência e de custos computacionais utilizando dois objetivos. Além disso, uma análise estatística foi conduzida para avaliar a variabilidade dos resultados finais obtidos e a complexidade dos algoritmos foi investigada para entender o equacionamento matemático de cada algoritmo. Os resultados apontaram as principais diferenças entre os algoritmos e elencaram aqueles que obtiveram os melhores desempenhos frente à métricas adotadas no estudo.

1.1. ESTRUTURA DA DISSERTAÇÃO

A dissertação está estruturada em seis capítulos. O Capítulo 1 apresenta a introdução e aborda as considerações iniciais sobre a pesquisa, incluindo as motivações para desenvolvimento da mesma. No Capítulo 2 são definidos os objetivos gerais e específicos. O Capítulo 3 apresenta a base teórica e a revisão da literatura relevante para a pesquisa. Aborda o dimensionamento de sistemas de distribuição de água, incluindo reservatórios e redes de distribuição. Também discute a calibração de modelos de RDA e os métodos de otimização, com foco em algoritmos genéticos. O capítulo revisa especificamente seis algoritmos genéticos: NSGA-II, NSGA-III, U-NSGA-III, R-NSGA-III, MOEA/D e RVEA.

O Capítulo 4 descreve a metodologia utilizada na pesquisa. Apresenta o modelo hidráulico EPANET, que é usado para simular o comportamento das RDA. O capítulo também detalha os cenários de otimização, incluindo as funções objetivo, restrições e variáveis de decisão. As métricas de comparação para avaliar o desempenho dos algoritmos são discutidas, incluindo o Critério de Eficiência (E) e a Função de Distribuição Acumulada (FDA).

O Capítulo 5 apresenta os resultados das simulações e as discussões sobre o desempenho dos algoritmos. As frentes de Pareto obtidas por cada algoritmo são analisadas, e os resultados são comparados com base nas métricas de Eficiência, FDA e do coeficiente de variação (C.O.V). O capítulo também discute a complexidade dos algoritmos e o tempo de processamento necessário para as simulações.

No Capítulo 6 são apresentadas as conclusões da pesquisa, com o resumo dos principais resultados e a escolha dos melhores algoritmos aplicados ao estudo. Por fim, são apresentadas as perspectivas de trabalhos futuros.

2 – OBJETIVOS

2.1 – OBJETIVO GERAL

O objetivo geral do presente trabalho é comparar o desempenho do dimensionamento ótimo de redes de distribuição de água de seis algoritmos genéticos multiobjetivos: NSGA-II, NSGA-III, U-NSGA-III, R-NSGA-III, MOEA/D e RVEA.

2.2 – OBJETIVOS ESPECÍFICOS

Os objetivos específicos são:

- Formular um problema de otimização multiobjetivo considerando dois objetivos (custo e resiliência da rede) e avaliar as relações perda-ganho entre eles;
- Utilizar métricas de comparação para avaliar o desempenho dos algoritmos;
- Elencar os melhores algoritmos aplicados ao estudo de caso, evidenciando as características únicas de cada um e destacando seus aspectos positivos e negativos.

3 – FUNDAMENTAÇÃO TEÓRICA E REVISÃO BIBLIOGRÁFICA

Neste capítulo são abordados os aspectos relativos ao dimensionamento de sistemas de distribuição de água (SDA), elencando quais componentes são dimensionados e os dados necessários para este dimensionamento. Além disso, são descritos a calibração e a otimização de modelos de RDA, bem como os seis algoritmos genéticos, que são a metaheurística de otimização desenvolvida neste trabalho. Por fim, são levantados os artigos mais relevantes publicados no contexto de otimização de RDA e realizadas as considerações acerca de suas contribuições para esta dissertação.

3.1 – DIMENSIONAMENTO DE SISTEMAS DE DISTRIBUIÇÃO DE ÁGUA

Sistemas de distribuição são dimensionados com o objetivo de fornecer água à população em quantidade, qualidade e pressão adequadas. A concepção desses sistemas compreende o dimensionamento e construção de diversos componentes, tais como estação elevatória de água bruta, adutora da água bruta, estação de tratamento de água, adutora de água tratada, reservatórios, sistemas de bombeamento e as redes de distribuição. O dimensionamento é uma etapa crítica dos projetos de SDA e deve considerar diversos fatores, tais como os tipos de materiais das redes, a demanda de água atual e futura, a topografia da região e a qualidade da água bruta. Habitualmente, são utilizados softwares computacionais para simulação e dimensionamento de SDA com a premissa de atendimento às normas da NBR 12218 da ABNT. Para fins deste trabalho, serão abordados somente os componentes a partir do reservatório com ênfase na otimização do projeto de distribuição de água. Tal abordagem é justificada pela utilização do estudo de caso de Alperovitz e Shamir (Alperovitz e Shamir, 1977), que representa um sistema hipotético de distribuição de água a ser dimensionado a partir do reservatório.

3.1.1 – Reservatórios

Segundo Tsutiya (2006), os reservatórios possuem como finalidade o armazenamento de água potável para distribuição aos consumidores, a regularização da oferta e demanda de água no sistema, a normalização das pressões nas redes e também constituem fonte de reserva de água para combate a incêndios. Adicionalmente, a instalação de reservatórios possibilita que o bombeamento de água seja realizado em períodos fora dos picos de

consumo, reduzindo custos de energia elétrica e que as bombas operem próximas ao rendimento máximo devido à regularidade da vazão e altura manométrica.

Outro aspecto importante é a equalização entre a produção de água nas estações de tratamento e a demanda dos usuários, que varia significativamente ao longo do dia. Durante períodos de menor demanda, como à noite e de madrugada, o reservatório acumula água, enquanto nos horários de pico, fornece o volume necessário para atender à população. Assim, nos horários de menor consumo o nível do reservatório é alto, enquanto ao fim do período de pico o nível é baixo.

No entanto, a instalação de reservatórios depara-se com algumas desvantagens, tais como a localização nas cotas adequadas, os altos custos de implantação e o impacto ambiental que pode ser gerado. O dimensionamento dos reservatórios considera a topografia, as características geológicas da região, projeções das populações a serem atendidas, a quantidade de água demandada, entre outros fatores. Entretanto, a característica mais importante para o dimensionamento de sua capacidade são os volumes de consumo, de combate a incêndios e emergencial. Os reservatórios possuem os níveis máximo, mínimo e útil. O nível máximo representa o maior volume atingido no reservatório, o mínimo é o volume mínimo para que não ocorram cavitação, vórtices ou arrastes de sedimentos do fundo do reservatório e o útil corresponde ao volume entre os níveis máximo e mínimo para atendimento às variações de demanda.

3.1.2 – Redes de distribuição de água

As redes de distribuição compreendem as tubulações e acessórios necessários à alocação de água em quantidade, qualidade e pressão adequadas aos consumidores. Seu dimensionamento compreende estudos de traçados da rede, determinação dos diâmetros e materiais das tubulações, identificação de interferências, como rios e estradas e a análise hidráulica a fim de garantir o pleno atendimento da população. As RDA são usualmente compostas por dois tipos de tubulações: as principais e secundárias. As redes principais têm por objetivo abastecer as redes secundárias e possuem maiores diâmetros. Já as redes secundárias abastecem diretamente os pontos de consumo ou ramais, e possuem diâmetros menores.

Na análise hidráulica das RDA são determinadas as vazões nos trechos, as cotas piezométricas nos nós e os diâmetros das redes a partir das vazões de distribuição no sistema. Na hidráulica, são consideradas três leis fundamentais para o cálculo do escoamento: conservação da massa, conservação da energia e conservação da quantidade de movimento. Com esse propósito, é utilizada a equação da continuidade para estabelecer a condição de equilíbrio. Nesta condição, a soma das vazões em cada nó da rede é nula e nos trechos de redes é aplicada a equação de resistência dada por $\Delta H = r \cdot Q^n$. No cálculo das perdas de carga distribuídas é utilizada a equação de Hazen-Williams, discutida no capítulo 4.2. Durante o dimensionamento, é importante o cumprimento às normas estabelecidas pela ABNT, sendo que a NBR 12218/2017 fixa as condições para elaboração de projetos de redes de distribuição de água e estipula as pressões máxima e mínima. Para a pressão mínima, é fixada a pressão dinâmica de 100 kPa (ou 10 mca), sendo esta a pressão sob condição de consumo máximo. Já para a pressão máxima, é adotado o valor de 400 kPa (ou 40 mca) de pressão estática, ou seja, pressão de consumo nulo.

3.1.3 - Modelo hidráulico

As equações da continuidade e da conservação da energia e a relação entre a vazão e a perda de carga, que caracterizam as condições de equilíbrio hidráulico da rede num dado instante, podem ser resolvidas através do Método do Gradiente, assim designado por Todini e Pilati (1987) e Salgado *et al.* (1988). Este é o método utilizado no EPANET (Rossman, 2000) para cálculo das vazões e pressões na rede.

Sendo uma rede com N nós e NF nós com cota piezométrica fixa, a relação vazão *versus* perda de carga numa tubulação entre os nós i e j pode ser expressa por:

$$H_i - H_j = h_{ij} = r \cdot Q_{ij}^n + m \cdot Q_{ij}^2 \quad (3.1)$$

onde H é a cota piezométrica no nó, h a perda de carga total, r o termo de perda de carga, Q a vazão, n o expoente da vazão e m o coeficiente de perda de carga localizada. O valor do termo de perda de carga r depende da fórmula utilizada.

A conservação da vazão nos nós é descrita por:

$$\sum_j Q_{ij} - D_i = 0 \quad \text{para } i = 1, \dots, N \quad (3.2)$$

onde D_i é o consumo no nó i . Conhecendo a cota piezométrica em determinados nós (nós de cota piezométrica fixa), pretende-se obter os valores de cota piezométrica H_i , e de vazão Q_{ij} , na rede que satisfaçam as equações (4.1) e (4.2).

O Método do Gradiente arbitra uma primeira distribuição de vazões nas tubulações que não necessariamente satisfazem as equações de continuidade nos nós. Em cada iteração do método, novas cotas piezométricas são obtidas resolvendo a seguinte matriz:

$$A \cdot H = F \quad (3.3)$$

sendo A a matriz Jacobiana ($N \times N$), H o vetor de incógnitas em termos de cota piezométrica ($N \times 1$) e F o vetor dos termos do lado direito da equação ($N \times 1$).

Os elementos da diagonal da matriz Jacobiana são calculados por:

$$A_{ij} = \sum_j p_{ij} \quad (3.4)$$

E os elementos não nulos fora da diagonal são:

$$A_{ij} = -p_{ij} \quad (3.5)$$

em que p_{ij} é o inverso da derivada da perda de carga total no trecho entre os nós i e j em relação à vazão e é calculado por:

$$p_{ij} = \frac{1}{nr|Q_{ij}|^{n-1} + 2m|Q_{ij}|} \quad (3.6)$$

Cada termo do lado direito da matriz é composto por uma parcela referente ao balanço de vazão no nó à qual é adicionado um fator de correção de vazão, conforme a equação a seguir:

$$F_i = \left(\sum_j Q_{ij} - D_i \right) + \sum_j y_{ij} + \sum_f p_{if} H_f \quad (3.7)$$

Sendo o último termo aplicável a qualquer trecho que ligue um nó i a um nó f de cota piezométrica fixa, o fator de correção de vazão, y_{ij} , para tubulações é expresso por:

$$y_{ij} = p_{ij} \left(r|Q_{ij}|^n + m|Q_{ij}|^2 \right) \text{sgn}(Q_{ij}) \quad (3.8)$$

onde $\text{sgn}(x)$ é 1 se $x > 0$ e -1 caso contrário.

Após o cálculo das cotas piezométricas, resolvendo a Eq. (4.3), as novas vazões podem ser obtidas de acordo com a seguinte equação:

$$Q_{ij} = Q_{ij} - \left(y_{ij} - p_{ij}(H_i - H_j) \right) \quad (3.9)$$

Se a soma de todas as variações de vazão (em valor absoluto) relativamente à vazão total em todos os trechos for superior à tolerância especificada (p.ex., 0.001), as equações (3.3) e (3.9) serão resolvidas novamente. A nova vazão obtida a partir da equação (3.9) satisfaz o princípio da continuidade da vazão nos nós após a 1ª iteração.

3.2 – CALIBRAÇÃO DE MODELOS DE REDES DE DISTRIBUIÇÃO DE ÁGUA

A calibração de modelos de RDA é a aproximação entre o que acontece em redes reais e sua modelagem numérica, desenvolvida através de equacionamentos matemáticos e conceitos físicos. Desta forma, a calibração pode ser considerada como um problema de otimização com funções objetivo e restrições específicas, onde devem ser utilizadas as leis de conservação da massa e energia do sistema de água, resultando em um problema de otimização não linear a ser resolvido (Nicolini *et al.*, 2011).

É exatamente a grande quantidade de equações, o caráter probabilístico da demanda dos usuários dependente do tempo e a falta de informações detalhadas sobre o sistema que eleva a complexidade no processo de calibração. A importância de um modelo calibrado adequadamente está diretamente associada a uma previsão confiável como resultado. Assim, apesar da ampla utilização dos modelos de simulação hidráulica para análise e otimização do comportamento do sistema em diversos cenários, o primeiro passo é a correta calibração do modelo (Nicolini *et al.*, 2011).

O primeiro passo para a calibração consiste no levantamento de informações das RDA, englobando os dados de cadastro das redes, reservatórios, acessórios instalados, bombas, estações de tratamento, curvas de demanda de água, perdas no abastecimento, entre outros. Em suma, são todas as informações necessárias para a correta representação do que acontece em um sistema real para a simulação do comportamento hidráulico no computador. A seguir, o software de modelagem calcula as vazões nas tubulações, as pressões nos nós, a altura da água nos reservatórios, a idade da água, entre outros parâmetros.

3.3 – OTIMIZAÇÃO DE REDES DE DISTRIBUIÇÃO DE ÁGUA

A otimização é um tipo de problema matemático que compreende um processo de busca no espaço de possíveis soluções para encontrar a solução ótima, baseado em uma ou mais funções objetivo a serem maximizadas ou minimizadas e em um conjunto de restrições a serem atendidas. Assim, a otimização visa a melhor solução, de acordo com os recursos disponíveis, a partir das variáveis de projeto (entradas). Em redes de distribuição de água o dimensionamento pode ser realizado utilizando técnicas de otimização que aumentem a eficiência do sistema através de modelos matemáticos e algoritmos. Os objetivos principais da otimização geralmente são minimizar custos de instalação, de operação das redes, o consumo de energia elétrica e reduzir as perdas de água na distribuição, entre outros, assegurando a disponibilidade de água aos consumidores de forma quantitativa e qualitativa.

Existem diversas técnicas de otimização que podem ser aplicadas a RDA, onde, nos últimos anos, destacam-se as técnicas de aprendizagem de máquinas, tais como os algoritmos genéticos, para resolução de complexos problemas de dimensionamentos reais através do levantamento e análise dos *trade-offs* entre seus objetivos (Fu *et al.*, 2013; Johns *et al.*, 2020; Marques *et al.*, 2018; Xiong *et al.*, 2018; Walker e Craven, 2019; Zarei *et al.*, 2022). A

otimização é realizada com base nos objetivos elencados, que podem partir de um único objetivo, caracterizando a chamada otimização mono-objetivo, até a consideração de vários objetivos, identificando-a como multiobjetivos, quando há até três objetivos, e muitos objetivos, quando há mais do que quatro objetivos.

Na otimização mono-objetivo o intuito é encontrar a solução ótima que satisfaça tanto a única função objetivo quanto as restrições que governam o comportamento do caso em estudo. Neste caso, em RDA, o exemplo mais abordado é a da minimização da função de custos de implantação e operação do sistema variando diâmetros de tubulações, operação de bombas, posicionamento de reservatórios, entre outros, sujeitos ao atendimento das premissas de dimensionamento, como pressões mínimas e velocidades máximas.

Já nas otimizações multi e de muitos objetivos ocorre a adoção de vários objetivos, muitas vezes conflitantes entre si, simultaneamente. Trabalhos como de Fu et al. (2013) utilizaram como objetivos custo de capital, custo operacional, falha hidráulica, perdas por vazamentos, idade da água e capacidade de combate a incêndios, por exemplo. Nesse cenário, a otimização multiobjetivo busca um conjunto de soluções que melhor identifiquem os *trade-offs*, ou balanços, entre os objetivos, denominado frente Pareto. A frente Pareto representa um conjunto ótimo de soluções que podem ser atingidas, onde um objetivo não pode ser melhorado sem piorar o outro objetivo.

Um ponto importante a se considerar é que à medida em que o número de objetivos considerados na otimização aumenta, a complexidade tanto na análise quanto na convergência das soluções de Pareto tende a se elevar significativamente. Esse aumento na complexidade pode impor limitações práticas ao número máximo de objetivos que podem ser efetivamente estudados e otimizados. Isso ocorre porque cada objetivo adicionado não apenas amplia o espaço de busca das soluções possíveis, mas também aumenta a dificuldade de encontrar soluções que sejam ótimas para todos os objetivos simultaneamente. Portanto, embora a abordagem de multiobjetivo seja extremamente importante para capturar diversas dimensões de um problema, há um ponto crítico em que a escalabilidade da análise se torna um desafio substancial.

Tanto a otimização mono como a multiobjetivo são utilizadas para dimensionamento de RDA. Porém, a busca por soluções que se aproximem de sistemas reais implica em

considerar sua grande complexidade, demandando uma maior quantidade de objetivos a serem analisados. É por esta razão que estudos mais recentes tendem a basear suas soluções em modelos multiobjetivos devido ao melhor entendimento que um conjunto de diversos objetivos atrelados às RDA fornece quando comparado a somente uma solução objetiva (Walski, 2001). Desta forma, há uma melhoria na capacidade de tomada de decisão por parte das empresas de saneamento quanto à melhor estratégia a ser implementada, por exemplo, em um projeto ótimo de redes.

Nesse contexto, destaca-se o desenvolvimento no campo de algoritmos metaheurísticos, que utilizam estratégias de alto nível para fugir de mínimos locais, realizando buscas mais robustas no espaço de soluções complexas do problema, mesmo em situações em que há poucas informações acerca do problema sendo analisado. A maior parte dos algoritmos metaheurísticos possuem inspiração em processos biológicos, tais como enxame de partículas, colônia de formigas e algoritmos genéticos, sendo este último amplamente utilizado nas últimas décadas e compreendendo atualmente a metaheurística mais bem estabelecida para resolver problemas de recursos hídricos (Maier *et al.*, 2014, Garzón *et al.*, 2022).

3.4 – ALGORITMOS GENÉTICOS

Algoritmos genéticos (AG) são uma metaheurística baseada em populações, com inspiração nos processos de evolução biológica desenvolvida em 1975 por J. H. Holland (Holland, J. H., 1975). Desta forma, uma população é composta por várias sequências, denominadas em AG como cromossomos, e através de processos como recombinação e mutação ocorre a combinação dessas sequências, gerando novas populações que evoluem em direção ao conjunto de soluções ótimas.

Geralmente, os AG envolvem três operadores: seleção, recombinação e mutação. O primeiro envolve a seleção do cromossomo mais apto, sendo que este possui mais chance e ser selecionado para recombinação. A recombinação consiste em selecionar dois cromossomos pais para gerar um filho, imitando a recombinação biológica entre dois organismos que possuem somente um cromossomo. Já na mutação ocorrem mudanças aleatórias em um cromossomo para formar um filho. Os AG processam populações de cromossomos, substituindo sucessivamente cada uma das populações por outra, sendo que cada

cromossomo pode ser identificado como um ponto no espaço de busca de soluções. Para avaliar o desempenho do cromossomo dentro da população, é necessária uma função de aptidão que atribua uma pontuação para o mesmo. Tal função de aptidão fornece uma medida da qualidade do cromossomo em relação ao problema abordado, ou seja, seu potencial de solução. Assim, o algoritmo seleciona os cromossomos mais aptos a produzirem um filho e também quais cromossomos serão descartados, aumentando a probabilidade de que a próxima geração contenha os cromossomos mais propícios à solução do problema.

O funcionamento dos algoritmos genéticos compreende os seguintes passos:

1. Criação de uma população inicial aleatória de cromossomos (soluções candidatas ao problema).
2. Cálculo da aptidão de cada cromossomo
3. Seleção dos indivíduos mais aptos da população
4. Aplicação dos operadores de recombinação
5. Aplicação dos operadores de mutação
6. Troca da população atual pelos filhos para formar a próxima geração.
7. Parada do algoritmo ao atingir um dos critérios de parada. Senão, volta ao passo 2.

Uma geração compreende cada iteração realizada no algoritmo. Tipicamente, há entre 50 a 1000 iterações em AG, sendo este intervalo dependente da natureza do problema. Ao final de todas as iterações ocorre uma rodada, gerando os cromossomos mais aptos da população. Basicamente, este é o procedimento para a aplicação de AG. Porém, o sucesso do algoritmo em encontrar a solução ótima depende de diversos fatores, tais como o tamanho da população, das probabilidades de recombinação e mutação, da metodologia utilizada para selecionar os indivíduos mais aptos, entre outros.

Existem diversos tipos de algoritmos genéticos utilizados em problemas de otimização na literatura. Neste trabalho, foram elencados seis algoritmos: NSGA-II, NSGA-III, U-NSGA-III, R-NSGA-III, MOEA/D e RVEA, explicados a seguir.

3.4.1. NSGA-II

O *Non-Dominated Sorting Genetic Algorithm-II* (NSGA-II), desenvolvido por Deb et al. (2002), emprega um procedimento rápido de classificação não-dominada, uma abordagem

de preservação elitista e um operador distância de aglomeração para manter a diversidade da população. A combinação dessas estratégias busca a convergência para as soluções ótimas de Pareto. O método do NSGA-II consiste em criar inicialmente uma população de pais aleatória P_0 , que é classificada com base na não-dominância. Cada solução recebe uma aptidão (ou classificação) de acordo com seu nível de não-dominância, onde 1 é o melhor nível, 2 é o próximo melhor nível, e assim sucessivamente. Desta forma, busca-se a minimização da aptidão, onde os melhores níveis estão associados aos menores valores de aptidão.

No início, os operadores usuais de seleção por torneio binário, recombinação e mutação são usados para criar uma população de descendentes Q_0 de tamanho N . Como o elitismo é introduzido comparando a população atual com as melhores soluções não-dominadas previamente encontradas, o procedimento é diferente após a geração inicial. Primeiro, uma população combinada de R_t é formada por $P_t \cup Q_t$, gerando uma população de tamanho $2N$. Em seguida, a população R_t é classificada de acordo com a não-dominância. Como todos os membros das populações anteriores e atuais estão incluídos, o elitismo é garantido em R_t . As soluções pertencentes ao melhor conjunto não-dominado F_1 são as melhores soluções da população combinada e devem ser mais enfatizadas do que qualquer outra solução na população. Se o tamanho do conjunto F_1 for menor que N , são escolhidos todos os membros desse conjunto F_1 para a nova população P_{t+1} e os membros restantes da população P_{t+1} são escolhidos a partir dos conjuntos não-dominados subsequentes, na ordem de suas classificações. Assim, as soluções do conjunto F_2 são escolhidas, seguidas pelas soluções do conjunto F_3 , e assim por diante. Esse procedimento continua até que não seja possível acondicionar nenhum outro conjunto de dados. Considerando que F_t seja o último conjunto não-dominado, em geral, a quantidade de soluções em todos os conjuntos de F_1 a F_t seria maior do que o tamanho da população. Para escolher exatamente N membros da população F_t , as soluções do último conjunto usando o operador de comparação por aglomeração \prec_n são classificadas em ordem decrescente e escolhem-se as melhores soluções necessárias para preencher todas as vagas da população. Desta forma, o operador \prec_n guia a seleção das soluções para a formação da frente Pareto.

À nova população P_{t+1} de tamanho N são aplicadas a seleção, recombinação e mutação, a fim de criar uma nova população de tamanho Q_{t+1} . É importante salientar que apesar de utilizar um operador de seleção por torneio binário, o critério de seleção neste momento é

baseado no operador de comparação por aglomeração \prec_n . Como esse operador exige tanto o ranking quanto a distância de aglomeração de cada solução na população, ambos são calculados enquanto a população P_{t+1} é formada.

O procedimento de comparação por aglomeração promove a diversidade entre as soluções não dominadas e é aplicado tanto na seleção por torneio quanto durante a fase de redução da população. Assim, as soluções competem com base na sua distância de aglomeração (uma medida da densidade de soluções na vizinhança). Essa associação de estratégias de ordenação não dominadas e o cálculo de distância de aglomeração torna o NSGA-II um algoritmo eficiente para lidar com problemas de otimização caracterizados por objetivos conflitantes, preservando as melhores soluções e mantendo a diversidade das mesmas.

3.4.2. NSGA-III

O algoritmo *Non-Dominated Sorting Genetic Algorithm III* (NSGA-III), desenvolvido por Deb e Jain (2014) e Jain e Deb (2014), oferece uma nova abordagem para lidar com problemas de muitos objetivos, semelhante ao NSGA-II. Um ponto de partida fundamental está em seu operador de seleção, que utiliza pontos de referência estrategicamente distribuídos. Enquanto o NSGA-II utiliza a distância de aglomeração para buscar a diversidade de soluções, o NSGA-III utiliza os pontos de referência. A metodologia de cálculo do NSGA-III consiste em criar inicialmente a população inicial aleatória P_0 de tamanho N . A seguir, a população é classificada em níveis de frentes Pareto não-dominadas F_1, F_2, \dots , como no NSGA-II. As soluções são inseridas em uma nova população P_{t+1} , incluindo suas frentes até que N seja alcançada.

A seguir, a diversidade de soluções é garantida através dos pontos de referência criados em um hiperplano. Tais pontos podem ser definidos de forma sistemática ao utilizar o método de Das e Dennis (Das e Dennis, 1998) ou podem ser fornecidos pelo usuário. O número de pontos de referência H depende do número de divisões p escolhidas para o plano e do número de objetivos M , sendo calculado por:

$$H = \binom{M + p - 1}{p} \quad (3.1)$$

As funções objetivo são normalizadas para lidar com as variações de escalas entre os objetivos, tornando-as comparáveis entre si. Isso é realizado através da construção de um hiperplano a partir dos extremos das funções objetivo, onde os menores valores para cada função objetivo são encontrados, determinando o z_{ideal} . A seguir, cada função objetivo é ajustada em relação aos extremos identificados, criando um hiperplano normalizado. Após a normalização, cada solução é associada a um ponto de referência com base na menor distância perpendicular entre a solução e as linhas de referência traçadas a partir desses pontos.

O operador de preservação de nichos realiza a contagem de nichos e é utilizado para garantir uma distribuição balanceada das soluções. Este método seleciona as soluções mais próximas aos pontos de referência e equilibra a distribuição de soluções. Seu funcionamento consiste em contar quantos membros da população estão associados a cada ponto de referência. Nessa fase, um ponto de referência pode ter um ou mais membros da população associados a ele ou, em alguns casos, não ter nenhum membro associado.

Quando não há nenhum membro de P_{t+1} associado ao ponto de referência \bar{j} , podem ocorrer duas situações. Na primeira, caso existam um ou mais membros na frente que estejam associados ao ponto de referência \bar{j} , o membro com a menor distância perpendicular à linha de referência é adicionado a P_{t+1} . Na segunda, se a frente não tiver nenhum membro associado a \bar{j} , o ponto de referência é excluído da geração atual.

Quando já existe um membro associado ao ponto de referência, escolhe-se um membro associado a \bar{j} aleatoriamente, que é adicionado a P_{t+1} . Após as contagens de nicho serem atualizadas, o procedimento é repetido K vezes para elencar todos os indivíduos da população P_{t+1} . A seguir, os operadores genéticos de recombinação e mutação são então aplicados à população P_{t+1} para gerar a população de descendentes Q_{t+1} . A iteração dos processos continua até atingir o critério de parada especificado, como por exemplo, a quantidade de gerações.

O processo descrito garante que a nova geração de soluções seja diversa, mantendo ao mesmo tempo a capacidade de convergência para a frente Pareto. Assim, o NSGA-III combina operadores genéticos tradicionais com técnicas avançadas de seleção,

características essas que tornam o algoritmo apto a lidar com a complexa otimização de problemas de muitos objetivos.

3.4.3. U-NSGA-III

O *Unified Non-Dominated Sorting Genetic Algorithm III* (U-NSGA-III), desenvolvido por Seada e Deb (2016), apresenta uma abordagem unificada para lidar com problemas de um, vários e muitos objetivos. Ele introduz um operador de seleção baseado em nichos de direção de referência, que ajusta dinamicamente a seleção de soluções com base na dimensionalidade do problema. Essa adaptação decorre do emprego de um tamanho populacional maior em relação às direções de referência e de um operador de seleção que promove a aleatoriedade, a escolha e a identificação de soluções ótimas.

O NSGA-III foi inicialmente desenvolvido para otimização com muitos objetivos (três ou mais). Em problemas de um único objetivo, as dificuldades do algoritmo vão desde o tamanho muito pequeno da população (menor múltiplo de 4), a falta de um operador de seleção específico para escolher as soluções geradas pelos operadores de recombinação e mutação e a não utilização do operador de nicho, pois como todos os valores de função recaem sobre as direções de referência, a distância perpendicular para cada membro da população é igual a zero. Por este motivo, a normalização das funções objetivo também é inutilizada. Já em problemas multiobjetivos, o problema do NSGA-III parece residir na menor pressão de seleção aplicada aos resultados para elencar as soluções não-dominadas, impactando na escolha dos melhores resultados que formam a próxima geração da população.

Assim, o U-NSGA-III foi proposto como uma alternativa para resolução de problemas de um a muitos-objetivos, sendo a diferença entre a quantidade de pontos de referência (H) e o tamanho da população (N) a principal mudança no algoritmo. No NSGA-III, esses dois parâmetros são idênticos, mas no U-NSGA-III, $N \geq H$ e N é um múltiplo de 4. Desta forma, o algoritmo é capaz de ajustar automaticamente o operador de seleção e a manutenção da diversidade conforme o número de objetivos.

Em problemas de um único objetivo, ao utilizar a estratégia de Das e Dennis (Das e Dennis, 1998), é gerada somente uma direção de referência, que agrega todas as soluções. Nessa

condição, o operador de seleção baseado em nichos é degenerado em um mecanismo elitista clássico utilizado em estratégias evolucionárias, denominado seleção por torneio binário, utilizando também operadores de recombinação e mutação, e um operador de preservação elitista.

Já em problemas de dois ou três objetivos, como $N \geq H$, esperam-se múltiplos membros da população para cada direção de referência, dividindo a população em várias frentes não-dominadas. O operador de seleção por nicho proposto no U-NSGA-III enfatiza as soluções não-dominadas sobre soluções dominadas e as soluções mais próximas das direções de referência. A partir desse ponto, as operações realizadas pelo U-NSGA-III para problemas multiobjetivos são as mesmas do NSGA-II.

Finalmente, quando problemas de otimização possuem muitos objetivos, a maioria dos membros da população tende a não ser dominada uns pelos outros. O operador de seleção de torneio por nicho seleciona a solução mais próxima a uma direção de referência, quando estão no mesmo nicho.

Quando a relação N / H é muito maior que 1, isso auxilia na escolha das soluções mais próximas às direções de referência para a próxima geração. Porém, quando N / H é muito próximo de 1, a seleção de torneio por nicho perde sua utilidade, transformando o comportamento do algoritmo no NSGA-III. A última situação é a mais recomendada para problemas com muitos objetivos.

A proficiência do U-NSGA-III em lidar com problemas um-para-muitos, produzindo soluções ótimas de Pareto dependentes do número de funções objetivas empregadas, o posiciona como uma ferramenta valiosa para resolver desafios de otimização e identificar *trade-offs* de forma abrangente.

3.4.4. R-NSGA-III

O algoritmo *Reference Point Based Non-Dominated Sorting Genetic Algorithm III* (R-NSGA-III) foi desenvolvido por Vesikar et al. (2018) aproveitando os múltiplos pontos de referência para pesquisar em segmentos específicos da frente de Pareto e suas populações correspondentes. Essa abordagem direcionada permite a integração das preferências do

usuário em relação aos muitos objetivos sob escrutínio, facilitando a descoberta de soluções desejadas dentro da estrutura algorítmica.

O algoritmo R-NSGA-III é uma extensão do NSGA-III tradicional. A principal diferença entre ambos é a introdução de pontos de aspiração, que são pontos de interesse utilizados pelo algoritmo para otimização em torno dessa região. Desta forma, o primeiro passo do algoritmo é a inserção dos pontos de aspiração pelo usuário como um vetor de coordenadas para cada objetivo. A seguir, os pontos são normalizados para ajuste dos valores máximos e mínimos de cada objetivo, utilizando a mesma técnica empregada no NSGA-III.

Os vetores calculados pelos pontos de referência normalizados e o ponto ideal de cada objetivo (ponto com valor 0 em todas as dimensões) são interceptados no hiperplano unitário de soluções, gerando r^k e $r^{k'}$ e fornecendo as direções de busca do algoritmo. Em seguida, o método Das e Dennis é aplicado para criar pontos no hiperplano, que são “encolhidos” utilizando um fator de redução que varia de 0 a 1. Esses pontos são então movidos no hiperplano através de seu centróide g , transladados para r^k e $r^{k'}$.

A combinação entre os pontos extremos do espaço de soluções e os pontos transladados constitui o hiperplano de soluções, utilizado pelo NSGA-III para guiar a seleção de soluções em cada iteração do algoritmo. O procedimento é repetido a cada geração, pois os valores normalizados mudam conforme novas soluções são encontradas. Ao das iterações, o algoritmo seleciona apenas a solução mais próxima de cada direção de referência, exceto aos que correspondem às direções extremas.

A incorporação de pontos de referência aprimora o processo de pesquisa em diversas regiões, guiando a busca em torno das regiões mais importantes da frente Pareto. O R-NSGA-III mantém as propriedades fundamentais de cálculo do NSGA-III, porém ao incorporar esse mecanismo de busca, obtém-se um aumento de eficiência computacional, uma das necessidades essenciais na busca de soluções de problemas de otimização.

3.4.5. MOEA/D

O *Decomposition-Based Multi-objective Evolutionary Algorithm* (MOEA/D) (Zhang e Li, 2007) decompõe um problema de otimização multiobjetivo em N subproblemas. Tais

subproblemas são resolvidos simultaneamente por meio da evolução e consequente otimização da população de soluções. Notavelmente, as soluções ótimas para subproblemas adjacentes exibem proximidade, com cada subproblema se beneficiando da solução ótima de sua contraparte vizinha. Em essência, o algoritmo explora efetivamente as relações de vizinhança inerentes entre seus subproblemas.

O algoritmo utiliza o método de Tchebycheff para decomposição do problema multiobjetivo em N subproblemas de otimização escalar, onde a função objetivo de cada subproblema é minimizada, conforme a equação:

$$\min g^{te}(x|\lambda^j, z^*) = \max_{1 \leq i \leq m} \{\lambda_i^j |f_i(x) - z_i^*|\} \quad (3.2)$$

Onde $x \in \Omega$, sendo Ω o espaço de decisão das variáveis, λ o vetor de pesos $\lambda = (\lambda_1, \dots, \lambda_m)^T$ e $z^* = (z_1^*, \dots, z_m^*)^T$ o ponto de referência. Desta forma, MOEA/D minimiza a distância ponderada entre a solução x e o ponto de referência z^* , ajustando o vetor de pesos λ para obtenção das soluções ótimas de Pareto.

Uma característica importante do algoritmo é que cada subproblema é resolvido utilizando informações da vizinhança. A vizinhança de um i -ésimo subproblema é definida com base na proximidade de todos os subproblemas com os vetores de peso de λ^i . Assim, a população é composta pela melhor solução encontrada até o momento para cada subproblema. Apenas as soluções atuais para seus subproblemas vizinhos são exploradas na otimização do subproblema.

A abordagem direta do MOEA/D para lidar com problemas multiobjetivos, juntamente com sua exploração hábil de soluções ao longo da frente Pareto por meio da decomposição, o torna aplicável a uma ampla gama de desafios de otimização.

3.4.6. RVEA

O *Reference Vector guided Evolutionary Algorithm* (RVEA) desenvolvido por Cheng *et al.* (2016) emprega uma abordagem de escalarização, denominada distância angular penalizada, para equilibrar dinamicamente a convergência e a diversidade de soluções de acordo com o número de objetivos e gerações. A convergência é avaliada medindo a distância entre as

soluções candidatas e o ponto ideal adaptada ao tipo de problema, por exemplo, problemas de maximização apresentam o valor máximo de cada função objetivo como seu ponto ideal. A diversidade é quantificada pelo ângulo agudo entre as soluções candidatas e os vetores de referência.

O RVEA adota a estratégia de elitismo tradicional, similar ao algoritmo NSGA-II. Sua particularidade reside nas etapas de seleção e adaptação guiadas por vetores de referência. O processo de geração de pais não aplica nenhuma estratégia, sendo que o RVEA utiliza os operadores de mutação e recombinação usuais e os escolhe aleatoriamente a partir da população atual. Essa simplificação é possível devido à estratégia de seleção guiada por vetores de referência, que é capaz de gerenciar eficazmente a convergência e a diversidade de soluções.

A seleção guiada por vetores de referência divide o espaço de objetivos em subespaços e a seleção é realizada em cada subespaço. Para tanto, a estratégia adotada realiza inicialmente a translação dos valores dos objetivos, diminuindo de cada vetor solução seu menor valor obtido. Desta forma, garante-se que todos os valores objetivo estejam no primeiro quadrante, maximizando a cobertura dos vetores de referência e também se determina o ponto ideal de origem do sistema de coordenadas.

Em seguida, ocorre o particionamento da população, onde o espaço de soluções é dividido em subespaços utilizando vetores de referência. Nesse processo, cada indivíduo é associado ao vetor de referência mais próximo baseado no ângulo agudo entre o vetor da solução e o vetor de referência. Realizado o particionamento da população, inicia-se o cálculo da distância penalizada pelo ângulo para equilibrar a convergência, relacionada à distância em relação ao ponto ideal, e a diversidade, representada pelo ângulo agudo entre a solução e outros vetores de referência. A solução com a distância mínima é selecionada como elitista e passada para a população da próxima geração.

A utilização de vetores de referência guia a busca de soluções, mantendo tanto a diversidade como a convergência, posicionando o RVEA como uma escolha viável para lidar efetivamente com problemas multiobjetivos.

Embora os algoritmos discutidos tenham uma base sólida na literatura, é essencial delinear as similaridades e distinções entre os seis algoritmos específicos examinados no estudo. A

Tabela 1 destaca as principais características de cada algoritmo, incluindo: 1) a quantidade de funções objetivo (FO) em que se aplicam; 2) a estratégia de otimização adotada; 3) os parâmetros iniciais e 4 e 5) as principais vantagens e desvantagens de cada um. Esta análise comparativa atua como um orientador, resumindo as características únicas e os parâmetros e recursos compartilhados em cada abordagem, bem como salientando os pontos fortes e limitações de cada algoritmo.

Tabela 1. Comparação dos algoritmos

Algoritmo	Quantidade de FO	Estratégia de Otimização	Parâmetros Iniciais	Vantagens	Desvantagens
NSGA-II (Deb et al., 2002)	2 a 3	Classificação Não-Dominada e Distância de Aglomeração	População Inicial, Taxa de Mutação, Taxa de Cruzamento	É um algoritmo comprovado que utiliza dominância de Pareto	Aplicável a problemas com até 3 objetivos devido ao mecanismo de seleção de soluções não-dominadas
NSGA-III (Deb e Jain, 2014)	≥ 4	Similar ao NSGA-II com adição de pontos de referência para a seleção de soluções não-dominadas	População Inicial, Taxa de Mutação, Taxa de Cruzamento, Pontos de Referência	Atualiza adaptativamente um conjunto bem distribuído de pontos de referência predefinidos, resolvendo eficazmente problemas de otimização com muitos objetivos	A qualidade das soluções depende da seleção dos pontos de referência, que normalmente são definidos com base no conhecimento do problema específico
U-NSGA-III (Seada e Deb, 2016)	≥ 1	Operador de seleção de nicho com função de referência que se adapta à dimensionalidade do problema	População Inicial, Taxa de Mutação, Taxa de Cruzamento, Direções de Referência	Proposto como um único algoritmo para qualquer número de objetivos	A necessidade de atualização adaptativa e manutenção das direções de referência pode introduzir um custo computacional adicional, o que pode ser preocupante em problemas muito grandes ou complexos
R-NSGA-III (Vesikar et al., 2018)	≥ 4	Seleção de pontos de referência para busca de soluções ótimas	População Inicial, Taxa de Mutação, Taxa de Cruzamento, Pontos de Referência	Flexibilidade no tratamento de vários tipos de informação de preferência, tornando-o adaptável a diferentes tipos de problemas de otimização de muitos objetivos	Altamente dependente da qualidade e precisão das informações de preferência fornecidas pelo usuário
MOEA/D (Zhang e Li, 2007)	≥ 4	Decomposição em subproblemas que são resolvidos simultaneamente	Taxa de Mutação, Taxa de Cruzamento, Direções de Referência, Número de Vizinhos, Probabilidade de Combinar Vizinhos	Decomposição do problema de otimização multiobjetivo em um conjunto de problemas de otimização de objetivo único. Introduce relacionamentos de vizinhança para compartilhar informações evolutivas	Falta de diversidade e velocidade de convergência lenta na fase final da evolução
RVEA (Cheng et al., 2016)	≥ 4	Vetores de referência fornecem convergência e diversidade às soluções	População Inicial, Taxa de Mutação, Taxa de Cruzamento, Direções de Referência	Os vetores de referência podem ser usados não apenas para decompor o problema de otimização multiobjetivo em vários subproblemas de objetivo único, mas também para esclarecer as preferências do usuário visando um subconjunto preferido da frente Pareto	Dificuldade em adaptar os vetores de referência à distribuição das soluções candidatas de acordo com as características geométricas estimadas da frente de Pareto

3.4 – DIMENSIONAMENTO ÓTIMO DE RDA

A otimização de projetos em sistemas de distribuição de água deixou de focar somente no objetivo de minimização de custos para abranger multiobjetivos que permitissem a análise e o entendimento dos relacionamentos conflitantes entre as variáveis de projeto e as restrições hidráulicas. Desta forma, é primordial que a otimização seja executada por algoritmos que possam tanto calcular como representar os complexos *trade-offs* entre os variados objetivos de forma assertiva. A presente seção abrange uma revisão da literatura acerca dos variados trabalhos de otimização no campo dos recursos hídricos e na área da computação, apresentando uma breve conceituação do que foi abordado no artigo, a metodologia empregada, os resultados e por fim uma crítica ao trabalho, pontuando similaridades ou possíveis campos a serem mais explorados na dissertação.

No trabalho desenvolvido por Garzón *et al.* (2022) foram revisados 31 artigos recentes sobre modelagem de aprendizagem de máquinas aplicadas a redes de abastecimento de água urbana para levantamento do estado da arte nesse campo de pesquisa, identificação de possíveis lacunas a serem preenchidas e direcionamento de trabalhos futuros. Para cada artigo, foram analisados o propósito do modelo, suas características e o estudo de caso aplicado. O propósito incluiu a classificação do modelo segundo tipo de rede (distribuição ou drenagem) e a sua aplicação (otimização, tempo-real). O estudo de caso compreendeu as informações das redes, sua localização, tamanho, entre outros. Por fim, são trazidas as características do modelo, detalhando o algoritmo utilizado, sua arquitetura, o tipo e número de variáveis de entrada e saída, o desempenho computacional e sua precisão frente à simulação original.

Em relação ao propósito do modelo, a maior parte dos artigos analisados (48%) eram referentes a otimização, empregando usualmente algoritmos baseados na população, como algoritmos genéticos, enxame de partículas, colônia de formigas, entre outros. Algoritmos evolucionários foram apontados como a metaheurística mais bem estabelecida para resolução de problemas de recursos hídricos, com destaque para o dimensionamento de projetos de SDA. Uma crítica apontada foi a de que, embora os modelos baseados em aprendizagem de máquinas sejam utilizados na resolução de vários problemas, os artigos possuíam como entrada poucas variáveis de projeto (por exemplo, diâmetros, dosagem de

cloro, entre outros), e as saídas eram geralmente valores resumidos de pressão crítica, cloro residual, etc.

Como propostas para pesquisas futuras, o artigo aponta a otimização multiobjetivo e a análise de incertezas como fundamentais para tomadas de decisão em modelos de aprendizagem de máquinas. Ademais, a tendência apontada foi a de que a quantidade e complexidade de entradas e saídas aumente, permitindo que mais alternativas sejam exploradas, que mais cenários sejam avaliados e até que incertezas sejam consideradas. Outro campo de pesquisa estaria relacionado à explicabilidade, ou seja, à compreensão da superfície de resposta do modelo original ou espaço de solução. O maior conhecimento de tal superfície poderia facilitar o entendimento do algoritmo, auxiliando na escolha da heurística mais adequada a cada pesquisa em rede de água.

O trabalho de Xiong *et al.* (2018) integrou um modelo de otimização multiobjetivo com avaliação do ciclo de vida a um modelo de simulação de redes de água. Na análise de *trade-offs* foram considerados os objetivos de custo total, emissão de gases de efeito estufa e um impacto no ecossistema aquático. Porém, apesar de diversos estudos terem investigado o efeito das mudanças climáticas nos recursos hídricos e avaliado seu desempenho em climas extremos, o impacto de políticas públicas em otimização de SDA foi pouco abordado. Assim, o trabalho avaliou a influência tanto das mudanças climáticas e como das políticas públicas futuras na otimização, adotando como cenário base o ano de 2012 e como cenários futuros os anos de 2020 e 2030.

Xiong *et al.* (2018) utilizaram o NSGA-II como algoritmo de otimização multiobjetivo para minimização de três objetivos, a saber: custo total, englobando os custos de produção da água e o de distribuição; emissão de gases de efeito estufa, que compreende todas as emissões que ocorrem em materiais, químicos e eletricidade durante a produção e distribuição de água; impacto no ecossistema aquático, que considera os usos de consumo e de degradação da água. Em relação aos cenários, foram investigados três de curto prazo e três de médio prazo em condições de clima normal, semi-árido e árido, bem como em condições de mudanças de políticas (capacidade de tratamento de água, geração de energia elétrica, entre outros) e em disponibilidade de recursos hídricos alternativos (água superficial, subterrânea, importada ou tratamento de águas residuárias). O estudo de caso foi

a área central de Pequim, que possui alta densidade demográfica e escassez de água doce para abastecimento da população.

Os resultados obtidos no trabalho mostraram a correlação entre aumento de emissões de gases de efeito estufa e aumento de custos em SDA. Além disso, observou-se a influência dos cenários climáticos em formas não convencionais de utilização de águas, com a indicação de que importar água é uma alternativa preferível para aumentar a provisão de água em regiões áridas, enquanto o tratamento de águas residuárias é mais apropriado a climas úmidos. Também foi verificada uma redução mais eficiente da emissão de gases de efeito estufa ao adotar estratégias de controle de demanda, enquanto há menos impacto no ecossistema aquático ao implementar estratégias de gestão de abastecimento de água.

O trabalho apresentado por Xiong *et al.* (2018) investigou as relações entre 3 objetivos aplicados a diferentes cenários futuros como forma de auxílio à tomada de decisão sustentável na gestão dos recursos hídricos de forma satisfatória e com resultados pertinentes para os tomadores de decisão. Entretanto, como o intuito era a análise de sustentabilidade, foi utilizado somente 1 algoritmo para otimização do estudo. Portanto, seria interessante explorar diferentes metodologias de otimização para analisar as respostas do modelo, visando obter um conjunto de soluções mais diversificado. Isso enriqueceria a análise dos *trade-offs*, ampliando o entendimento das relações entre os objetivos e proporcionando uma visão mais abrangente das possíveis alternativas de solução.

Sharma *et al.* (2022) apresentaram uma metodologia multietapas para a identificação de Distritos de Medição Controle (DMC) em RDA. A abordagem combinou métodos de detecção de agrupamentos, otimização de quatro objetivos com o algoritmo NSGA-III e tomada de decisão multicritério. O objetivo principal consistia em criar um layout de DMC que fosse eficiente tanto em termos de custo quanto de desempenho hidráulico. A metodologia foi testada em uma rede de água real e em um estudo de caso hipotético, mostrando sua capacidade de decompor a rede em layouts de DMC viáveis hidraulicamente.

Os principais resultados indicam que a metodologia foi capaz de produzir várias soluções considerando os custos de implementação de redes e o desempenho hidráulico. Para tanto, os parâmetros iniciais considerados foram tamanho da população igual a 200, número de gerações igual a 100, taxas de recombinação e mutação de 0,8 e 0,1, respectivamente, e

somente 1 restrição. A abordagem foi eficaz ao fornecer alternativas que equilibram custos, resiliência, desvio de pressão e déficit de fluxo. Desta forma, o conjunto de soluções ótimas de Pareto obtidas permitiu uma melhor compreensão dos *trade-off* entre os diferentes objetivos, auxiliando na tomada de decisão para a implementação de medidores de vazão e válvulas.

Para trabalhos que utilizam o NSGA-III, este artigo oferece uma metodologia robusta aplicável à otimização de RDA com muitos objetivos. Enquanto o estudo de Sharma *et al.* (2022) utilizou quatro objetivos para otimizar suas variáveis de entrada, o presente estudo visa avaliar a eficácia de múltiplos algoritmos em termos de convergência, diversidade das soluções e eficiência computacional. Adicionalmente, Sharma *et al.* aplicaram o NSGA-III utilizando parâmetros fixos, como 200 indivíduos e 100 gerações. Já este trabalho, inclui a comparação de diferentes configurações e metodologias, buscando analisar o desempenho dos algoritmos ao gerar os *trade-offs* entre as soluções.

Yazdi (2016) desenvolveu estratégias de decomposição aplicadas a algoritmos evolucionários multiobjetivos baseados em decomposição (MOEA/D). Tais estratégias integram conceitos de busca harmônica e AG dentro da estrutura do MOEA/D e foram desenvolvidas para otimização de projetos de RDA em larga escala. Os objetivos utilizados foram custo e confiabilidade, sendo que a última relaciona a carga excedente nos nós e os anéis de vários tamanhos das redes.

A metodologia proposta no trabalho aplica os operadores de busca harmônica e AG para aprimorar as capacidades de busca local e usa a estrutura do MOEA/D para a busca global e manutenção da diversidade na população. A eficiência e confiabilidade da metodologia proposta foram comparadas com dois algoritmos baseados em dominância, o NSGA-II e o SPEA2. Adicionalmente, foram utilizados 4 estudos de caso de RDA: os sistemas de Pescara, Saemangeum e Zona Oeste e a rede de irrigação de Balerma.

Os resultados mostraram que os métodos MOEA/D geralmente fornecem melhores frentes Pareto do que os outros dois algoritmos para todas as redes estudadas. O SPEA2 superou o NSGA-II na rede de Pescara, porém seu desempenho decaiu, tornando-se local para problemas maiores, não encontrando as caudas da frente Pareto como o NSGA-II e os dois MOEA/D. As duas estratégias do MOEA/D, principalmente a baseada na busca harmônica,

foi capaz de fornecer uma distribuição grande e uniforme de soluções ótimas de Pareto representativas para todos os estudos de caso considerados.

O estudo demonstrou que as metodologias MOEA/D híbridas podem ser usadas com sucesso para resolver problemas de otimização multiobjetivos em sistemas de distribuição de água de larga escala e também apresentaram maior eficiência do que os algoritmos baseados em não-dominância. Assim, esta dissertação também utilizará o MOEA/D para efeito de comparação com os outros algoritmos, dada a sua eficiência em explorar o espaço de busca atrás das soluções ótimas de Pareto.

O trabalho proposto por Johns *et al.* (2020) aborda a otimização de projetos de RDA utilizando dois algoritmos genéticos multiobjetivos híbridos que utilizam práticas de engenharia para melhoria de desempenho computacional. Ao todo foram utilizadas três funções objetivo de minimização do custo total, da perda de carga e suavidade da rede e a primeira heurística utilizada foi a eliminação de gargalos, ou seja, de tubulações com diâmetros pequenos próximas a reservatórios ou outras fontes de água que introduziam grandes perdas de carga ao sistema. A segunda heurística foi a de transição suave entre diâmetros diferentes nas tubulações, que obedece ao princípio de que o diâmetro de uma rede nunca é maior que a soma dos diâmetros das redes à montante conectados ao mesmo nó. O desempenho dos dois algoritmos foi comparado ao NSGA-II e os resultados obtidos demonstraram que associar conhecimentos de engenharia ao algoritmo resultou em soluções superiores em termos de otimalidade matemática e viabilidade prática.

No trabalho, foi utilizado o algoritmo genético *Multi-Objective Adaptive Locally Constrained Genetic Algorithm* (MOALCO-GA) para a heurística 1, onde, através de um procedimento de roleta, um nó é selecionado e, partindo para a montante deste ponto, busca-se um nó com excesso de perda de carga. Ao identificar esse nó, a tubulação imediatamente a sua jusante sofre um incremento de diâmetro. Já no caso da heurística 2, foi utilizado o algoritmo *Multi-Objective Pipe Smoothing Genetic Algorithm* (MOPS-GA) que também utiliza um procedimento de roleta para selecionar uma tubulação e modifica os diâmetros das tubulações a jusante para atendimento ao princípio citado no parágrafo anterior. Além dos dois algoritmos, o NSGA-II também foi aplicado a três estudos de caso selecionados da literatura e de complexidades distintas (simples, intermediário e complexo), variando de um

SDA com 1 reservatório e 34 trechos de tubulações até um SDA com 4 reservatórios e 317 trechos de rede.

Na maioria dos resultados apresentados por Johns *et al.* (2020), os algoritmos MOALCO-GA e MOPS-GA apresentaram resultados superiores quando comparados ao NSGA-II. Ainda, o MOPS-GA, ao apresentar resultados melhores do que do MOALCO-GA, sugere aplicações mais promissoras, provavelmente devido à possibilidade de sua heurística ser aplicada durante toda a busca, enquanto o MOALCO-GA foi aplicado normalmente aos estágios iniciais da busca. Além disso, a adição da componente de suavização incorporada ao MOPS-GA atingiu a convergência mais rápido e obteve um melhor conjunto de soluções finais do que o NSGA-II. Por fim, a superioridade dos dois algoritmos frente ao NSGA-II foi mais evidente nos estudos de caso de SDA simples e intermediário, porém, no caso do SDA de maior complexidade, os resultados obtidos foram bastante próximos uns dos outros.

O trabalho desenvolvido apresentou a incorporação de práticas de engenharia aplicadas a dois algoritmos e, apesar dos resultados superiores, os resultados ainda se mostraram próximos aos do NSGA-II, o que confirma a eficiência deste último ao lidar com problemas multiobjetivos. Desta forma, o NSGA-II foi um dos algoritmos elencados para este trabalho dada sua vasta utilização no campo de otimização em SDA.

Walker e Craven (2020) propuseram uma abordagem de visualização de parâmetros utilizados em otimização multiobjetivos através de algoritmos evolucionários (AE) ao analisar as características do algoritmo, tais como convergência e diversidade. Como AE são altamente dependentes da configuração correta de seus parâmetros, realizar um *benchmarking* de seus parâmetros permite identificar quais destes apresentam um melhor desempenho para cada problema. No entanto, como a análise do *benchmarking* nem sempre é de fácil compreensão por parte do usuário/leitor, o método de visualização proposto pelos autores visa auxiliar no entendimento dessa problemática. Para tanto, foram utilizados dois algoritmos evolucionários, o NSGA-II e o NSGA-III em dois *benchmarks*, o DTLZ e o CEC 2009.

Para visualização do método, foram adotados quatro indicadores de desempenho: hipervolume (Fleischer, 2003) que calcula o volume do espaço objetivo entre a população e um ponto de referência predefinido, ou seja, sua convergência; distância de aglomeração,

um componente do algoritmo NSGA-II que calcula a distância em relação ao vizinho mais próximo para cada objetivo, fornecendo uma noção de diversidade; espalhamento do espaço de busca objetivo, que calcula o espalhamento de um conjunto de aproximação (aproximação à frente Pareto) no espaço objetivo; diversidade no espaço de decisão, que representa a variedade de soluções na população de busca, importante para assegurar a pesquisa no espaço de busca. A visualização consistiu em um círculo, em que a extensão do arco ocupada pela população era mostrada como uma região sombreada. Essa região preencheu um arco no círculo, que ilustra a melhor e pior pontuação do indicador de desempenho, onde a parte da frente representa o melhor resultado, enquanto a parte de trás, a pior. A distância de multidão foi adicionada às figuras como uma escala de cores para indicar a diversidade da população e todas as ilustrações foram comparadas para diferentes gerações durante a execução da otimização.

Os resultados do trabalho de Walker e Craven (2020) utilizando o algoritmo NSGA-II e 3 objetivos mostraram que, ao analisar o indicador de distância de multidão, a diversidade na população normalizou conforme o avanço na otimização progrediu. Já para o indicador de espalhamento do espaço objetivo, foi observado que à medida em que os valores objetivos reduziam com a minimização das soluções, a medida de diversidade também diminuía. Um padrão similar ao encontrado para o espalhamento do espaço objetivo foi observado no indicador de diversidade do espaço de decisão. Ao utilizar o algoritmo NSGA-III para 4 objetivos, o trabalho concluiu que a otimização não convergiu mesmo após realizar mais rodadas em relação aos outros experimentos e que a diversidade foi reduzida de maneira bem mais acentuada, indicando que o algoritmo otimizador possui preferências em áreas específicas da frente Pareto.

O trabalho de Walker e Craven (2020) propôs uma metodologia para visualização da operação dos algoritmos NSGA-II e NSGA-III com 3 e 5 objetivos fornecendo assim uma melhor compreensão acerca do processo de otimização com a utilização de indicadores de desempenho. Como neste trabalho ambos algoritmos são utilizados, a análise fornecida pelos autores proporciona uma alternativa para avaliar um componente inerente aos algoritmos e que foi utilizada como indicador de desempenho: a distância de aglomeração.

A grande diversidade de algoritmos evolucionários na literatura, combinada às diversas possibilidades de aplicação em diferentes áreas, demandou a necessidade de comparar o

desempenho dos mesmos. No trabalho de El-Ghandour e Elbeltagi (2018) foram comparados os desempenhos de cinco AE na otimização de RDA, sendo eles: algoritmos genéticos, enxame de partículas, colônia de formigas, algoritmo memético e uma modificação do algoritmo *shuffled frog leaping*. O objetivo era determinar qual algoritmo era mais eficiente na otimização de dois objetivos, o custo de projeto e reabilitação de RDA.

Os critérios de comparação eram a taxa de sucesso para encontrar a solução ótima, o número de avaliações da função objetivo, a influência das sementes iniciais na solução e a qualidade geral dos resultados. Os cinco algoritmos foram aplicados a dois estudos de caso conhecidos, Alperovitz e Shamir e Nova Iorque e a uma rede real de distribuição de água da cidade de El-Mostakbal, no Egito.

O estudo determinou que o enxame de partículas foi o algoritmo que apresentou melhor desempenho em várias métricas, incluindo qualidade da solução, eficiência, avaliações de funções objetivas e velocidade de convergência. Os autores não utilizaram algoritmos específicos encontrados na literatura, e sim os implementaram para investigar o desempenho dos parâmetros mais comuns envolvidos no processo de otimização. Ao contrário de El-Ghandour e Elbeltagi (2018), este trabalho emprega AG que utilizam técnicas aprimoradas para garantir a convergência e diversidade do conjunto de soluções.

Zhao *et al.*, (2019) conduziram uma análise abrangente de comparação de seis algoritmos de muitos objetivos, categorizando-os com base na qualidade das soluções, velocidade de convergência e aproximação da frente de Pareto. O trabalho utilizou os algoritmos MOEA/D, DBEA, RVEA, NSGA-III, ϵ -MOEA e IBEA e detalhou as estratégias de otimização e suas aplicabilidades a diferentes tipos de testes.

O estudo considerou algoritmos pertencentes a quatro tipos de categorias, classificadas de acordo com sua estratégia de busca e seleção de soluções, sendo eles: baseados em decomposição, dominância, redução de objetivos e indicadores. Os tipos de testes utilizados foram o Deb–Thiele–Laumanns–Zitzler ou DTLZ (Deb *et al.*, 2006) e o Walking-Fish-Group ou WFG (Huband *et al.*, 2006), considerando 3, 5, 10 e 15 objetivos. A avaliação da qualidade dos conjuntos de soluções não-dominadas apresentados por cada algoritmo e sua convergência foi realizada pelo indicador IGD+.

A análise dos resultados destacou o desempenho superior dos algoritmos RVEA e NSGA-III em termos de qualidade de soluções e estabilidade, principalmente com o aumento do número de objetivos. O RVEA apresentou melhor convergência e diversidade de soluções no DTLZ e o NSGA-III se destacou por seu desempenho no WFG. O RVEA supera o MOEA/D na convergência de resultados, especialmente quando o número de objetivos é maior que dez. Porém, embora a frente Pareto dos dois algoritmos seja bem distribuída, conjunto de soluções do MOEA/D é geralmente mais uniforme do que do RVEA.

O trabalho de Zhao *et al.* (2019) ressalta o potencial de algoritmos baseados em dominância e em indicadores para estudos futuros. Um ponto positivo do trabalho foi a abrangência e a variedade de testes utilizados, variando a quantidade de objetivos e permitindo uma avaliação robusta dos algoritmos. Embora o trabalho de Zhao *et al.* (2019) também tenha avaliado seis algoritmos, com destaque para o NSGA-III e o RVEA, os algoritmos considerados neste estudo diferem, envolvendo funções objetivas únicas e restrições específicas para a área de RDA.

No trabalho de Zou *et al.* (2021), os autores exploraram 8 algoritmos para encontrar soluções para 27 problemas de otimização de muitos objetivos. Os autores verificaram que nesses tipos de problemas, quando um objetivo é otimizado, os demais objetivos têm seus valores rapidamente deteriorados, logo, é necessário verificar e encontrar um conjunto de soluções que, embora estejam deterioradas, sejam também eficientes para serem analisadas. Ou seja, soluções não necessariamente ótimas a um único objetivo, mas com grande potencial de se explorar *trade-offs* entre os objetivos. Entretanto, balancear a diversidade e alcançar uma boa convergência com uma solução do espaço de soluções torna-se um desafio. Nesse caso, os autores propuseram a exploração de algoritmos evolucionários baseados em dominância e decomposição dada sua grande popularidade.

Na análise de Zou *et al.* (2021), os algoritmos evolucionários de maior impacto são aqueles capazes de decompor um problema multiobjetivo em vários problemas de um único objetivo. Nesse caso, várias possibilidades de algoritmos e técnicas podem ser exploradas como o uso de MOEA/D (Zhang; Li, 2007), NSGA-III (Deb; Jain, 2014), A-NSGAIII (Jain e Deb, 2013), I-DBEA (Asafuddoula *et al.*, 2015), MOEA/D-CMA (Li; Zhang; Deng, 2017), RVEA (Cheng *et al.*, 2016), entre outros. O principal problema dessas várias possibilidades é que em sua grande maioria a solução ótima é extremamente dependente do formato da

curva de Pareto, ou ainda, da frente Pareto. Essa dependência gera problemas quando a curva se apresenta disforme, comprometendo as soluções que utilizam apenas MOEA ou NSGA-II. Nesse caso, os autores propõem a combinação de ambas as técnicas, criando o que chamaram de MaOEADRA, na qual será utilizado o mecanismo de elitismo, típico dos NSGA-II, baseado na dominação e decomposição de Pareto para equilibrar a distribuição e a convergência da população gerada no MOEA/D.

Os principais resultados encontrados por Zou *et al.* (2021) foram comparados dentro de um ambiente de simulação chamado PlatEMO (<https://github.com/BIMK/PlatEMO>) e comparado com o DDEANS (He *et al.*, 2017), NSGA-III (Deb; Jain, 2014), MaOEAIGD (Sun; Yen; Yi, 2019), SPEAR Jiang; Yang, 2017), ANSGAIII (Jain; Deb, 2013), MOEADD (Li *et al.*, 2015) e RVEA (Cheng *et al.*, 2016). Além disso, os autores verificaram seus resultados utilizando como cenários DTLZ1–DTLZ7 (Deb *et al.*, 2005), IDTLZ1- IDTLZ2 (Jain; Deb, 2013), WFG4–WFG9 (Huband *et al.*, 2006), MaF1-MaF13 (Cheng *et al.*, 2017), *Pareto-Box problem* MLDMP (Li *et al.*, 2018) e MPDMP (Köppen; Yoshida, 2007). Os resultados consideraram as métricas IGD e HV (Ishibuchi *et al.*, 2010). O algoritmo proposto pelos autores mostrou-se superior a todos os demais algoritmos comparados, alcançando uma alta adaptabilidade mesmo sob a presença de frentes Pareto disformes.

O trabalho de Zou *et al.* (2021) mostrou-se como uma proposta bastante adaptável e com boa convergência para trabalhar com problemas multiobjetivos. Para tanto, foram realizadas comparações detalhadas entre o desempenho do algoritmo proposto e 7 outros algoritmos, destacando as características de cada um nos cenários utilizados. Essa análise permitiu a melhor compreensão do desempenho dos algoritmos em diferentes cenários, aumentando o entendimento do funcionamento do RVEA, NSGA-III e MOEA/D, também utilizados neste estudo.

A seguir, é apresentada a Tabela 3 apresentando os principais pontos de interesse dos artigos revisados.

Tabela 2. Sumário de artigos revisados

Autores	Ano	Título	Local	Objetivo	Técnica de otimização	Quantidade de funções objetivo
Garzon <i>et al.</i>	2022	Machine Learning-Based Surrogate Modeling for Urban Water Networks: Review and Future Research Directions	Water Resources Research	Revisão de 31 artigos recentes sobre modelagem de aprendizagem de máquinas aplicadas a redes de abastecimento de água urbana		
Xiong <i>et al.</i>	2018	Integrated multi-objective optimization framework for urban water supply systems under alternative climates and future policy	Journal of Cleaner Production	Integrar um modelo de otimização multiobjetivo com avaliação do ciclo de vida a um modelo de simulação de redes de água	NSGA-II	3
Sharma <i>et al.</i>	2022	Multiphase Procedure for Identifying District Metered Areas in Water Distribution Networks Using Community Detection, NSGA-III Optimization, and Multiple Attribute Decision Making	Journal of Water Resources Planning and Management	Identificar Distritos de Medição e Controle utilizando detecção de agrupamentos, otimização de muitos objetivos com o algoritmo NSGA-III e tomada de decisão multicritério.	NSGA-III	4
Yazdi	2016	Decomposition based Multi Objective Evolutionary Algorithms for Design of Large-Scale Water Distribution Networks	Water Resources Management	Desenvolver estratégias de decomposição aplicadas a algoritmos evolucionários multiobjetivos baseados em decomposição (MOEA/D)	Busca harmônica e AG	2
Johns <i>et al.</i>	2020	Knowledge-based multi-objective genetic algorithms for the design of water distribution networks	Journal of Hydroinformatics	Otimizar projetos de RDA utilizando dois algoritmos genéticos multiobjetivos híbridos que utilizam práticas de engenharia para melhoria de desempenho computacional	MOALCO-GA, MOPSGA e NSGA-II	3
Walker e Craven	2020	Identifying good algorithm parameters in evolutionary multi- and many-objective optimisation: A visualisation approach	Applied Soft Computing Journal	Proposta de visualização de parâmetros utilizados em otimização multiobjetivos através de algoritmos evolucionários	NSGA-II e NSGA-III	3 e 4
El Ghandour e Elbeltagi	2018	Comparison of five evolutionary algorithms for optimization of water distribution networks	Journal of Computing in Civil Engineering	Comparar cinco algoritmos para otimização de RDA	AG, Enxame de partículas, Colônia de	2

					formigas, Algoritmo memético e <i>Shuffled frog leaping</i> modificado	
Zhao et al.	2019	A comparative study of the evolutionary many-objective algorithms	Progress in Artificial Intelligence	Comparar de seis algoritmos de muitos objetivos	MOEA/D, DBEA, RVEA, NSGA-III, ϵ - MOEA e IBEA	3, 5, 10, 15
Zou et al.	2021	A many-objective evolutionary algorithm based on dominance and decomposition with reference point adaptation	Knowledge-Based Systems	Propor o algoritmo MaOEADRA, comparando-o a sete outros algoritmos estabelecidos na literatura	MaOEADRA, DDEANS, NSGA-III, MaOEAIGD, SPEAR, ANSGA-III, MOEADD, RVEA	3, 5, 8, 10, 15

Todos os artigos revisados apresentam pontos de interesse que contribuíram para esta dissertação, sejam através de propostas de novas metodologias e algoritmos, pela concepção de novas formas de visualização de resultados ou até mesmo pela difusão de conhecimentos que auxiliem no desenvolvimento deste trabalho. Desta forma, buscou-se trazer o estado da arte dos principais estudos sendo conduzidos no âmbito de otimização multi e de muitos objetivos, com o intuito de identificar lacunas e oportunidades para avanços metodológicos que possam ser incorporados ao desenvolvimento desta pesquisa.

4 – METODOLOGIA

A metodologia consiste na comparação do desempenho de seis algoritmos genéticos para selecionar um conjunto de diâmetros que atendam às funções objetivo e suas restrições, utilizando o software de modelagem hidráulica EPANET. Na subseção 4.1 são discutidas as formulações das funções objetivo, restrições impostas, variáveis de entrada de simulação e estudos de caso adotados. Em seguida, a subseção 4.2 explora as métricas utilizadas para comparar o desempenho dos algoritmos.

4.1 – CENÁRIOS

Problemas de otimização de RDA são comumente relacionados ao dimensionamento dos diâmetros das redes para atingir objetivos específicos. Nesse cenário, são consideradas informações adicionais como a topologia da rede, a pressão mínima exigida, as demandas nos nós e o funcionamento dos reservatórios, entre outros. A otimização multiobjetivo foi implementada na linguagem de programação *Python* e seu algoritmo de implementação encontra-se no Apêndice A brevemente explicado. O computador utilizado para simulação possui sistema operacional Arch Linux, processador Intel i5 11ª geração e 8GB de memória RAM. O projeto ótimo desenvolvido neste estudo busca maximizar o Índice de Resiliência Modificado (IRM) ao mesmo tempo em que minimiza os custos das redes, conforme descrito a seguir.

Objetivo 1: Índice de Resiliência Modificado

O Índice de Resiliência Modificado (IRM), proposto por Jayaram e Srinivasan (2008), representa a porcentagem de excesso de carga nos nós em relação à carga necessária nos nós, e é uma função de maximização, calculada por:

$$Max\ IRM = \frac{\sum_{j=1}^n q_j (ha_j - hr_j)}{\sum_{j=1}^n q_j \cdot hr_j} \cdot 100 \quad (4.1)$$

onde q_j é a demanda no nó j , ha_j é a pressão disponível no nó j e hr_j é a carga necessária no nó j .

Objetivo 2: Custos

A função de custo é determinada pela minimização do produto do preço do diâmetro do tubo pelo comprimento da seção:

$$\text{Min Custo} = \sum_{t=1}^{np} C_i(D_i^k) \cdot L_i \quad (4.2)$$

onde np é o número total de tubos; C_i o custo por unidade de comprimento do tubo i relacionado ao diâmetro D_i , L_i o comprimento do tubo e k é o coeficiente de regressão.

A perda de carga é a diferença entre as cargas de dois nós em um tubo, calculada através da fórmula de Hazen-Williams, elencada pelo ganho de tempo na simulação e calculada por:

$$\Delta H_k = H_{1,k} - H_{2,k} = w \frac{L_k}{C_k^\beta D_k^\gamma} Q_k Q_k^{\beta-1}, \forall k \in NP \quad (4.3)$$

onde w é uma constante numérica que depende das unidades utilizadas, C é o coeficiente de rugosidade do material e β e γ são os coeficientes de regressão. Os cálculos que consideram as leis da conservação de massa e de energia são realizados pelo EPANET, conforme descrito anteriormente.

As outras restrições do problema estão relacionadas às pressões mínimas nas redes, ao diâmetro mínimo e aos diâmetros comerciais disponíveis:

$$H_n \geq H_{n \min}, \forall n \in NN \quad (4.4)$$

onde $H_{n \min}$ é a pressão mínima em cada nó.

$$D_k \geq D_{\min}, \forall n \in NN \quad (4.5)$$

onde D_{\min} é o diâmetro mínimo.

$$D_k \in D, \forall k \in NP \quad (4.6)$$

com D sendo o conjunto de diâmetros comerciais.

Estudo de caso:

Há diversos estudos de caso de RDA na literatura, tanto de sistemas reais como hipotéticos. Dentre os sistemas hipotéticos, destacam-se o *Anytown* (Walski et al. 1987), que possui características do mundo real, tais como como programação de bombas, armazenamento em reservatórios e provisão de capacidade de combate a incêndios e o sistema de Alperovits e Shamir (Alperovits e Shamir, 1977) ou sistema de dois anéis.

Neste trabalho foi utilizado o estudo de caso de Alperovits e Shamir, que possui 6 nós, 8 trechos de rede de 1000 m de comprimento e 1 reservatório de altura fixa de 210 m. A pressão mínima em todos os nós é de 30 m, as informações de demanda e cota nos nós podem ser visualizadas na Figura 1 e os custos unitários dos diâmetros são apresentados na Tabela 3.

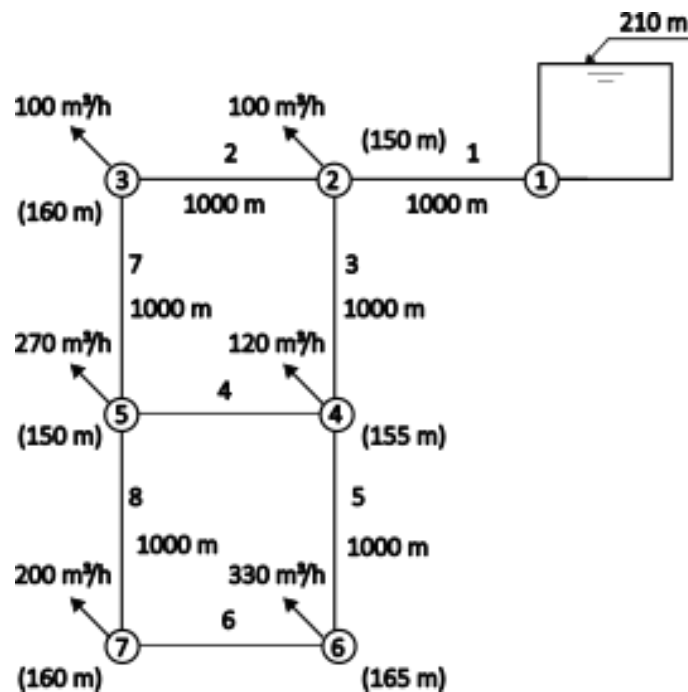


Figura 1. Rede de Alperovitz e Shamir

Tabela 3. Custos das redes por diâmetro

Diâmetro (mm)	Custo (unidade)
25,4	2,0
50,8	5,0
76,20	8,0
101,6	11,0
152,4	16,0
304,80	50,0
355,60	60,0
406,40	90,0
457,20	130,0
508,00	170,0
558,80	300,0
609,60	550,0

O cenário de simulação foi elaborado para permitir uma comparação eficiente entre os seis algoritmos, variando parâmetros como o tamanho inicial da população e as probabilidades de recombinação e mutação. O número de gerações foi estabelecido em 1000, e cada algoritmo utilizou a mesma semente aleatória inicial em cada execução. Para avaliar o impacto das sementes aleatórias nos resultados, seis simulações adicionais foram realizadas, cada uma com uma semente diferente. As quantidades de partições e pontos de referência, presentes em todos os algoritmos, exceto para o NSGA-II, também foram mantidas fixas ao longo das simulações.

O fluxograma da metodologia pode ser visualizado na Figura 2, onde primeiramente é definida a população de indivíduos com os diâmetros iniciais. A seguir, tem início a modelagem hidráulica no EPANET e a otimização utilizando os algoritmos genéticos, onde são aplicadas as taxas de recombinação e mutação e calculadas as funções objetivo sujeitas às restrições do modelo. Caso a simulação não atinja os critérios de parada, como por exemplo a quantidade de gerações, são gerados novos diâmetros que serão inseridos novamente no modelo hidráulico. Caso o critério de parada seja atendido, a simulação é finalizada com as populações calculadas.

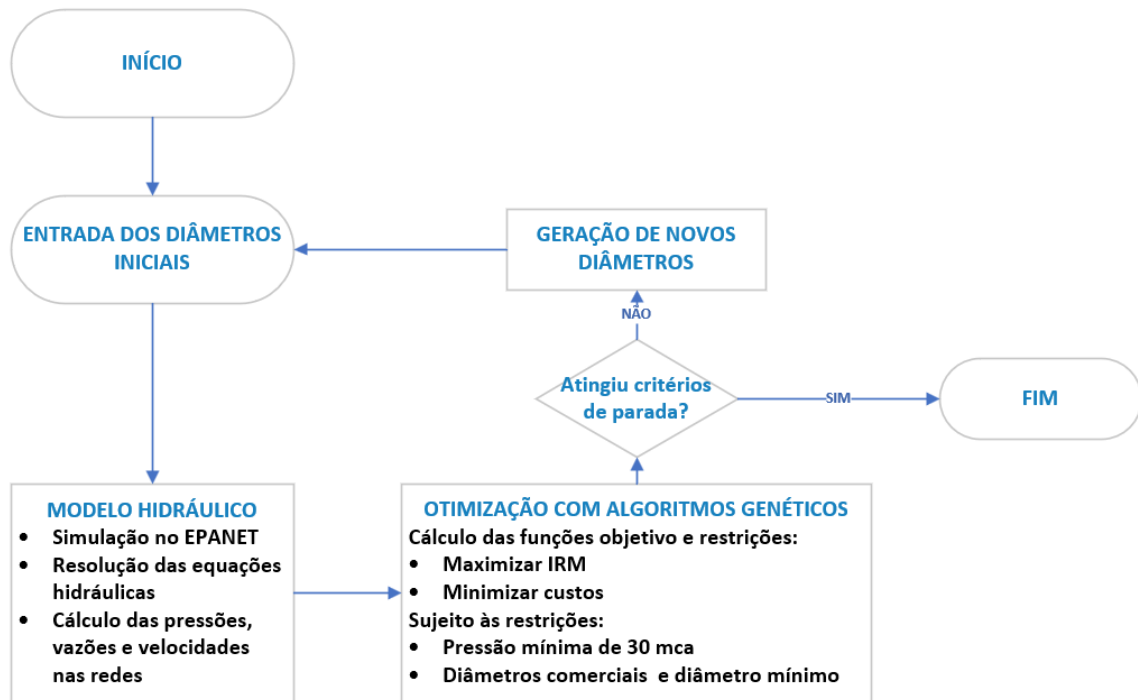


Figura 2. Fluxograma de trabalho

4.3 – MÉTRICAS DE COMPARAÇÃO

A primeira métrica de comparação utilizada neste estudo foi o Critério de Eficiência (E), proposto por Mora-Melia *et al.* (2015). Essa métrica mede o desempenho de um algoritmo ao equilibrar a qualidade das soluções obtidas com o esforço computacional necessário para obtê-las. O cálculo do E é dado pela Equação:

$$E = \frac{n_q}{n_c} \quad (4.7)$$

onde n_c representa a velocidade de convergência do algoritmo, medida pelo número total de avaliações da função objetivo até a solução final, e n_q refere-se à proporção de soluções bem-sucedidas de alta qualidade em relação ao número total de soluções geradas. As boas soluções são aquelas que atingem um valor limite predeterminado, que no caso de problemas de minimização, indica a quantidade de resultados abaixo do limite inferior definido pelo usuário. O cálculo de n_q segue a Equação:

$$n_q = \frac{n_s}{n_{sim}} \quad (4.8)$$

onde n_s é o número de boas soluções e n_{sim} representa o total de soluções obtidas.

A segunda métrica aplicada foi a Função de Distribuição Acumulada (FDA), utilizada para avaliar a probabilidade de uma variável ser menor ou igual a um valor específico. No estudo, ela foi empregada para analisar a variabilidade dos diâmetros em relação aos custos. Esse método permite examinar os diâmetros necessários para alcançar os custos mínimos nas simulações, ao mesmo tempo em que fornece uma análise da variabilidade entre os diâmetros resultantes.

5 – RESULTADOS E DISCUSSÕES

Em cada simulação, os tamanhos de população inicial considerados foram de 10 e 100 indivíduos, com probabilidades de mutação de 0,01, 0,05 e 0,1, e probabilidades de crossover de 0,1, 0,5 e 0,9. Todos os seis algoritmos utilizaram a mesma semente aleatória inicial, resultando em seis simulações distintas com diferentes sementes aleatórias. As análises foram organizadas com base nos mesmos parâmetros de população, mutação e crossover, totalizando 18 análises por algoritmo.

Finalizadas todas as seis simulações, foi obtida a frente Pareto de custo *versus* IRM (Índice de Resiliência Modificado) para os seis algoritmos avaliados, considerando todos os resultados obtidos nas seis simulações, apresentada nas Figuras 3 a 8. Para fins de análise visual, nas Figuras 9 a 14 constam apenas os resultados da segunda simulação, divididos entre as populações de 10 e 100 para destacar a variabilidade, convergência e a quantidade de soluções geradas para cada tamanho de população.

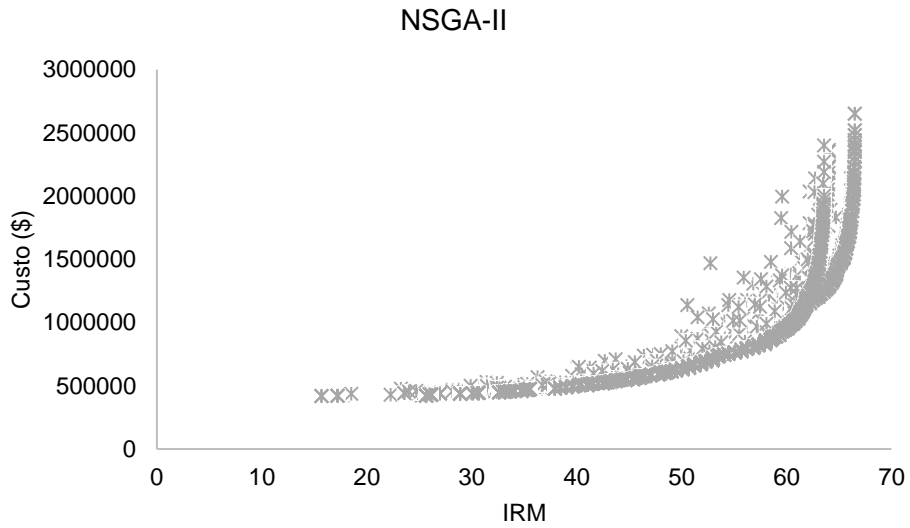


Figura 3. Frente Pareto do NSGA-II para todas as simulações

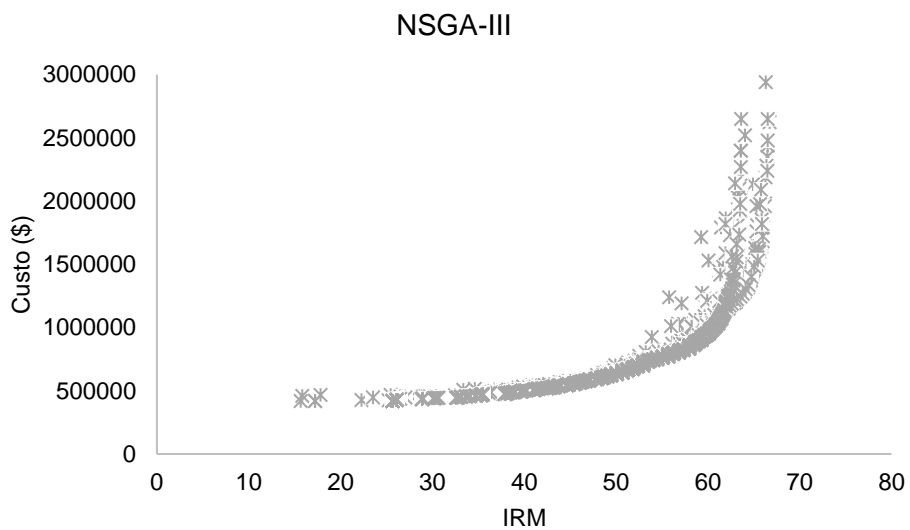


Figura 4. Frente Pareto do NSGA-III para todas as simulações

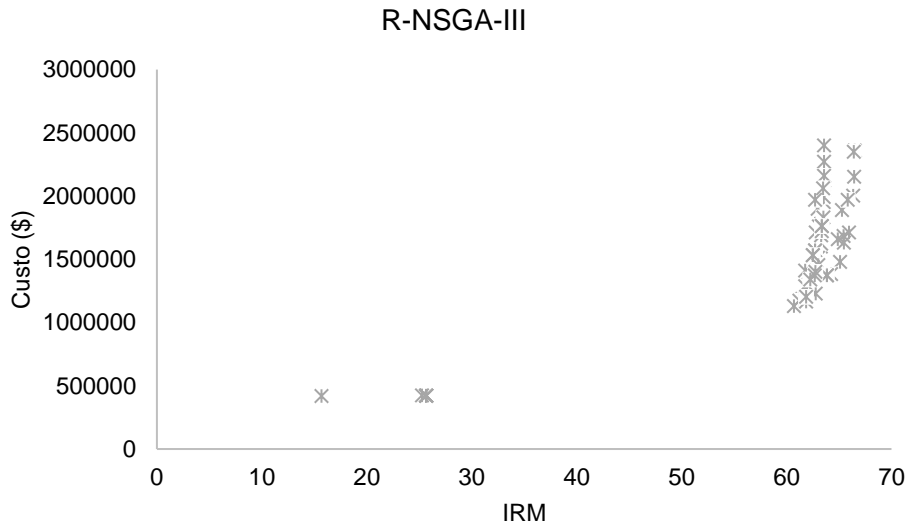


Figura 5. Frente Pareto do R-NSGA-III para todas as simulações

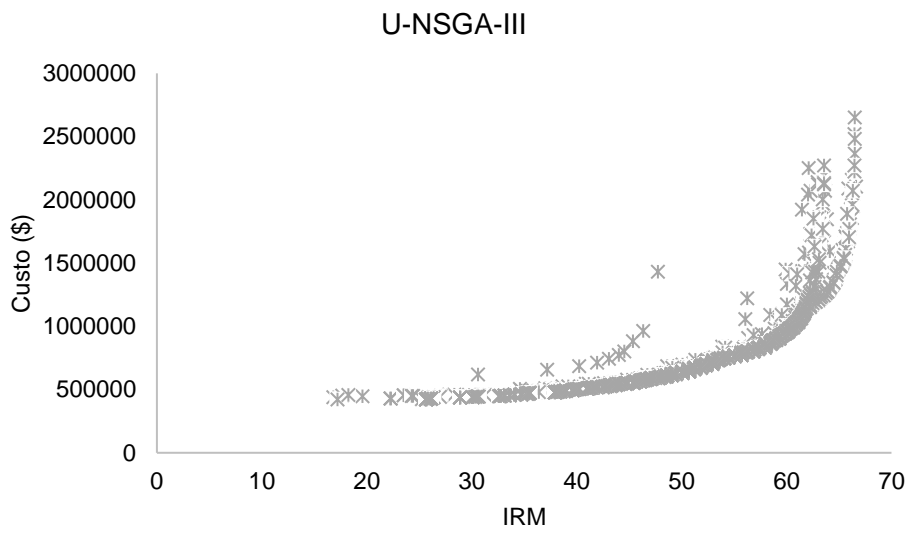


Figura 6. Frente Pareto do U-NSGA-III para todas as simulações

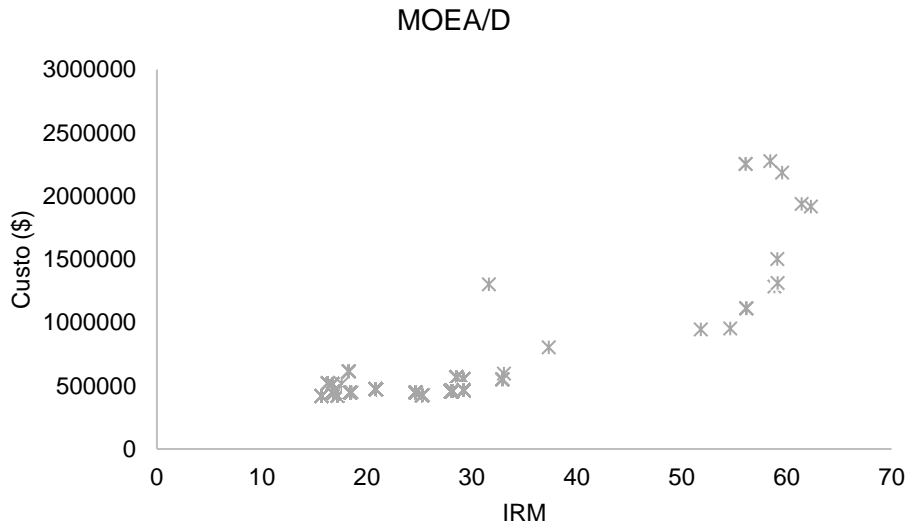


Figura 7. Frente Pareto do MOEA/D para todas as simulações

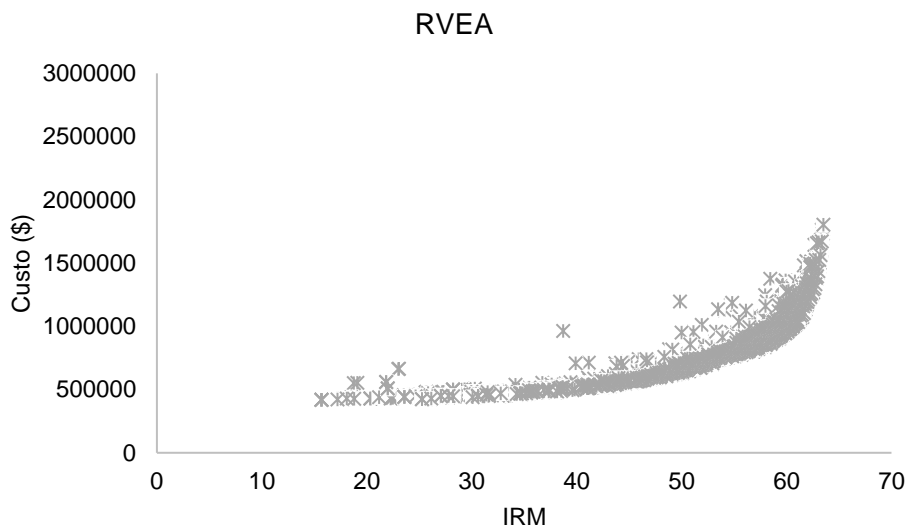


Figura 8. Frente Pareto do RVEA para todas as simulações

Analisando os resultados de somente uma simulação, o R-NSGA-III foi o algoritmo que produziu menos soluções em ambas as populações, com 18 resultados em cada caso. Além disso, foram obtidas as mesmas soluções para as populações de 10 e 100. Embora o MOEA/D também pareça ter gerado menos soluções, ele produziu um número semelhante ao dos outros algoritmos (cerca de 1000), mas com soluções concentradas nos extremos, ou seja, otimizando ou o custo ou o IRM. Em contrapartida, os demais algoritmos apresentaram um comportamento mais equilibrado, gerando uma diversidade maior de soluções distribuídas ao longo da frente Pareto, especialmente fora das regiões extremas.

O objetivo da frente de Pareto é identificar os melhores *trade-offs* entre diferentes soluções, permitindo a escolha das opções com a melhor relação custo-benefício para a tomada de decisão. Essa relação é exatamente o objetivo principal da frente, pois convergir para soluções de menor custo resultam em uma redução do IRM, que é um indicador da qualidade da rede. Um IRM mais baixo implica em maior vulnerabilidade a flutuações de demanda ou a eventos inesperados em SDA, podendo comprometer a capacidade de manter a pressão mínima nas redes. Assim, algoritmos que apresentam uma maior diversidade de soluções são mais vantajosos do que aqueles que convergem apenas para soluções de baixo custo ou de IRM máximo. Com base nesses critérios, o NSGA-II, NSGA-III, U-NSGA-III e o RVEA demonstraram um desempenho superior ao R-NSGA-III e ao MOEA/D.

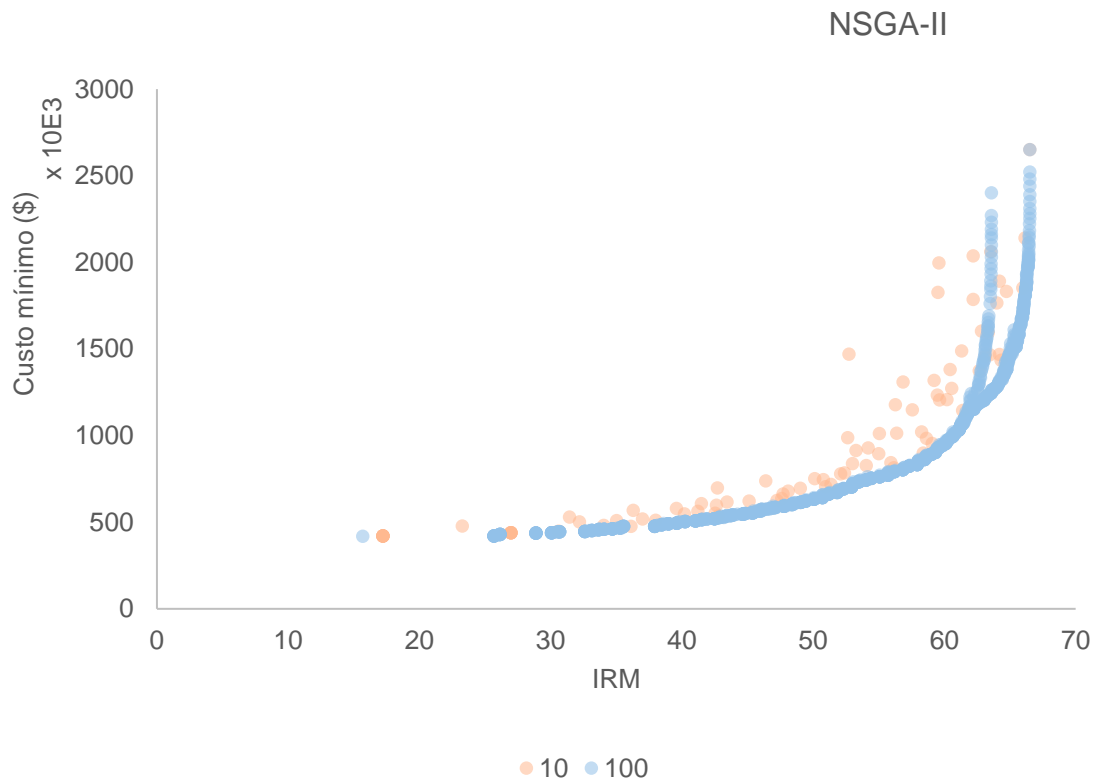


Figura 9. Frente Pareto do NSGA-II (1 simulação)

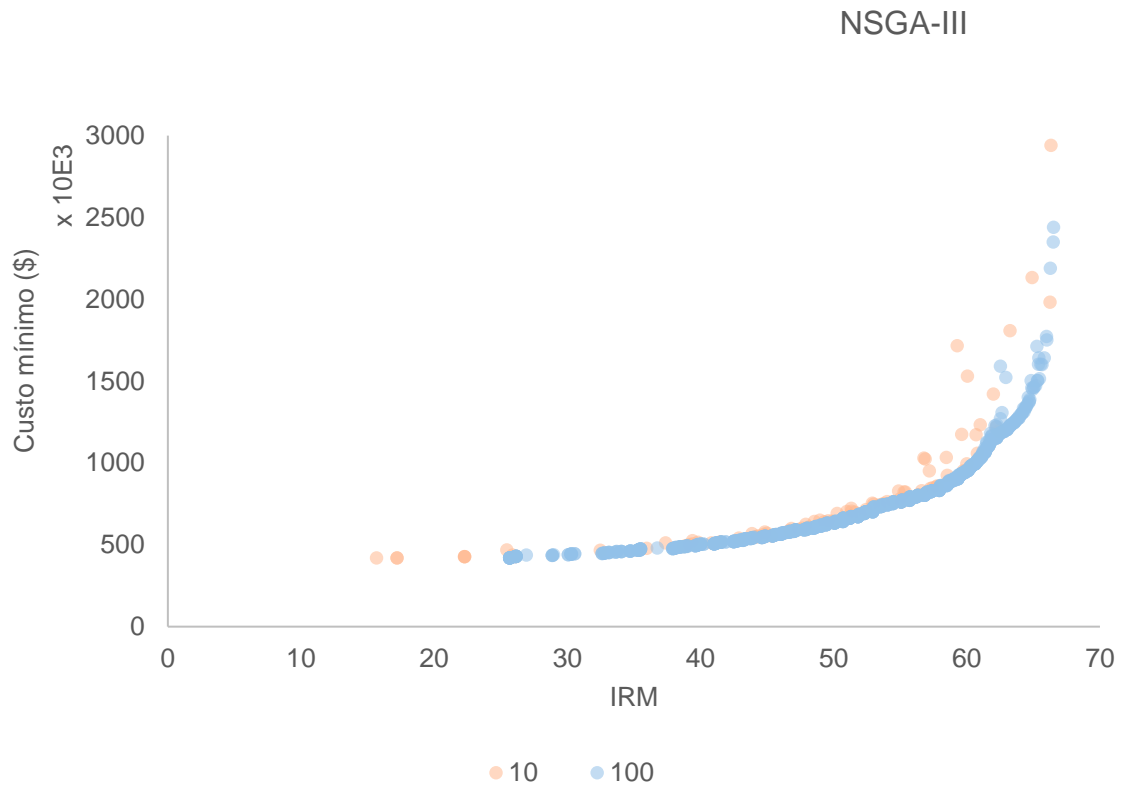


Figura 10. Frente Pareto do NSGA-III (1 simulação)

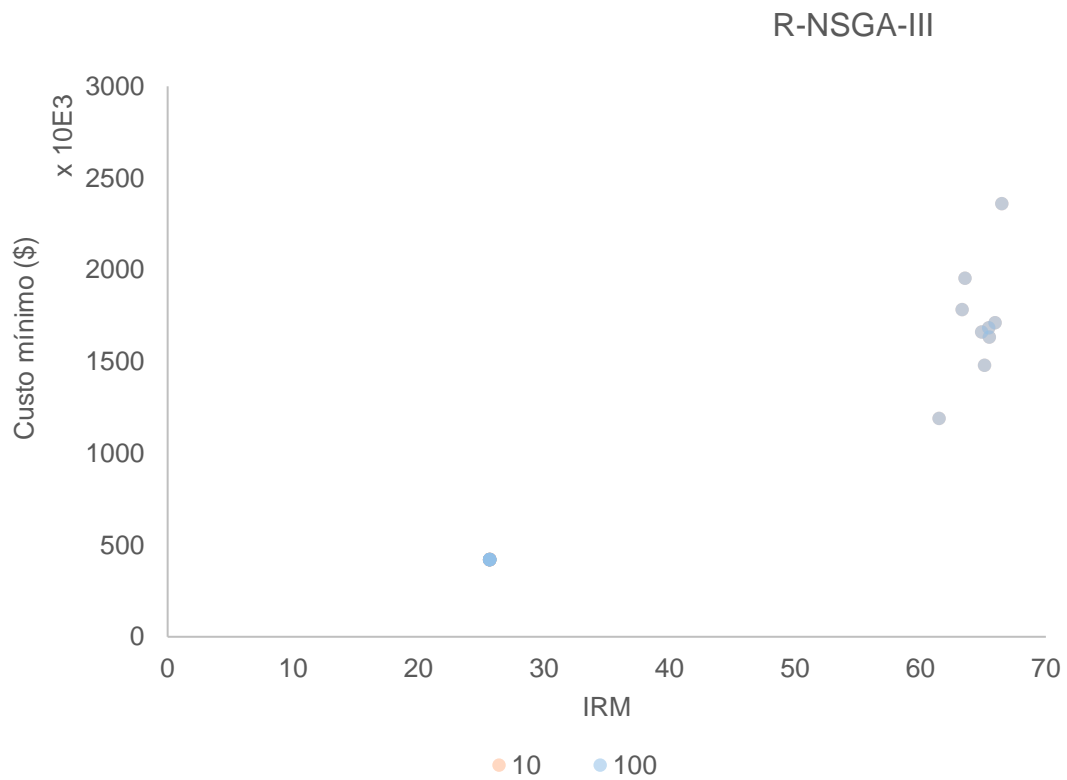


Figura 11. Frente Pareto do R-NSGA-III (1 simulação)

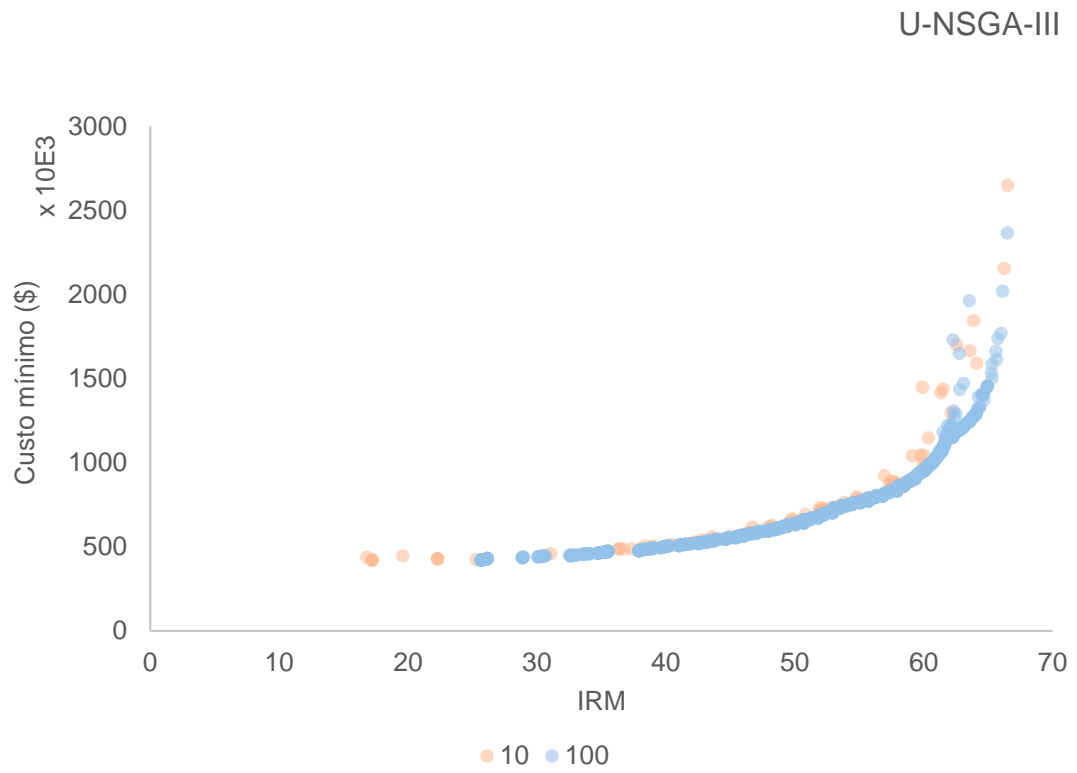


Figura 12. Frente Pareto do U-NSGA-III (1 simulação)

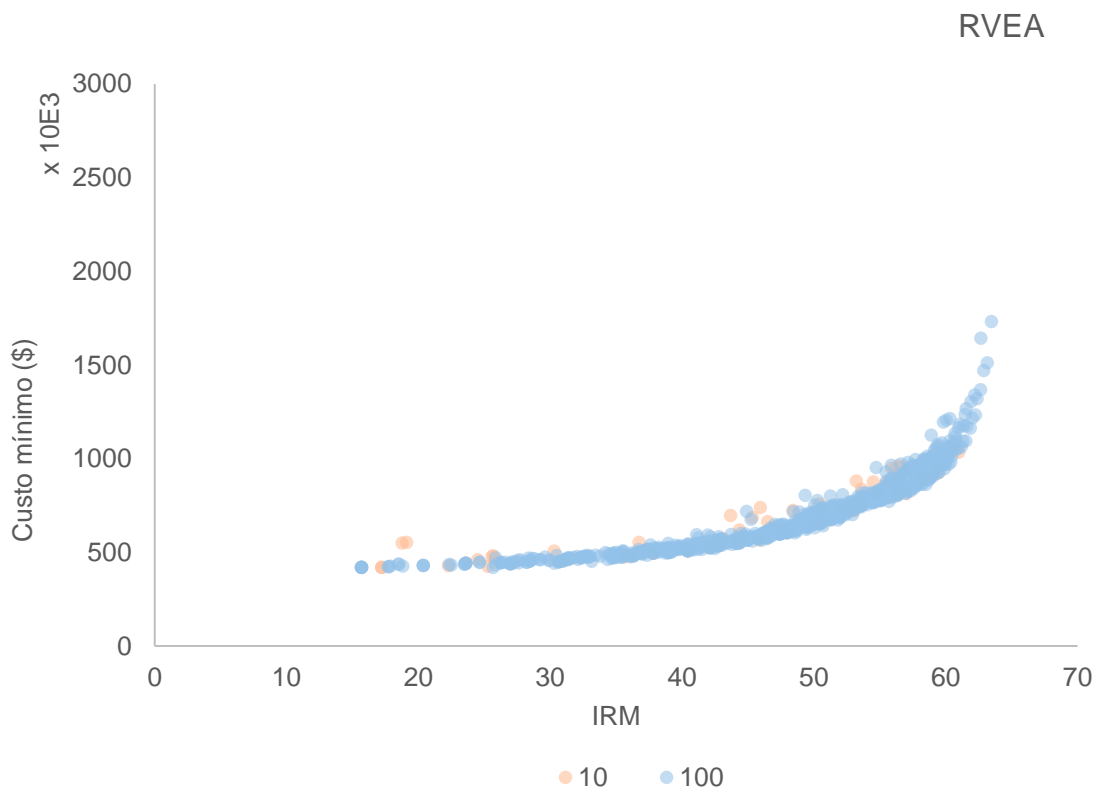


Figura 13. Frente Pareto do RVEA (1 simulação)

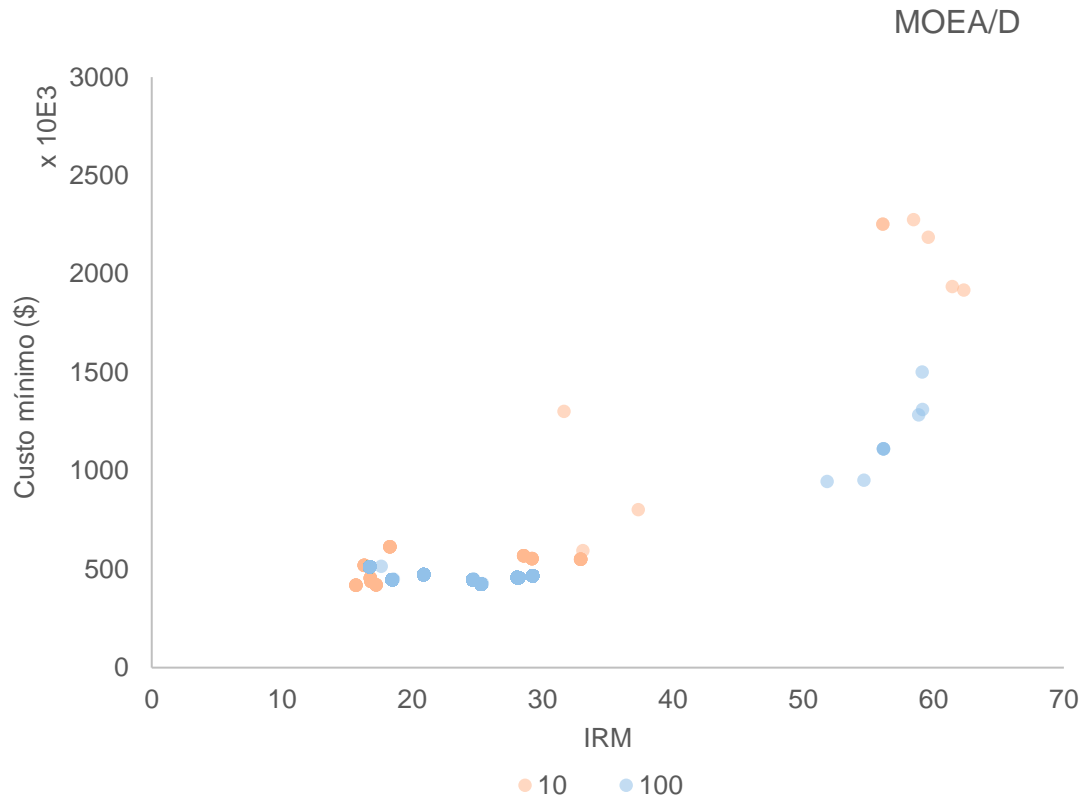


Figura 14. Frente Pareto do MOEA/D (1 simulação)

Embora destacar a frente Pareto seja fundamental para revelar os *trade-offs* entre os objetivos, a grande quantidade de soluções – cerca de 1000 na maioria dos algoritmos – apresenta desafios para uma representação gráfica mais compreensível. Por isso, para cada conjunto de parâmetros de população, recombinação e mutação, apenas os valores de custo mínimo e seus respectivos IRM foram selecionados. Essa abordagem permite analisar o comportamento dos algoritmos na busca pelos menores custos em meio às variações dos parâmetros. Dada a ocorrência de pares de resultados repetidos, utilizou-se um gráfico de bolhas, cujos tamanhos variam conforme a frequência dos resultados, para garantir uma visualização mais informativa.

Nas Figuras 15 a 20, a convergência para o custo mínimo dos algoritmos NSGA-II, R-NSGA-III e U-NSGA-III torna-se evidente, com seus custos girando em torno de \$ 420.000, independentemente dos parâmetros iniciais utilizados. Notavelmente, esses três algoritmos forneceram apenas dois pares de resultados, mesmo com as seis simulações. Em contrapartida, o NSGA-III, RVEA e, especialmente, o MOEA/D apresentaram maior

diversidade de custos mínimos, sendo que as ocorrências próximas a \$ 420.000 foram mais frequentes em comparação aos outros valores.

Essa simplificação da frente Pareto funciona como uma análise visual de sensibilidade, mostrando como os algoritmos respondem às variações nos parâmetros iniciais de população, recombinação e mutação. Assim, o NSGA-III, RVEA e MOEA/D demonstram maior sensibilidade, exibindo um espectro maior de resultados, conforme pode ser visualizado nas Figuras 16, 19 e 20. Isso reforça a importância de selecionar cuidadosamente os parâmetros iniciais ao utilizar esses algoritmos, uma vez que pequenas mudanças podem impactar significativamente na obtenção das frentes Pareto.

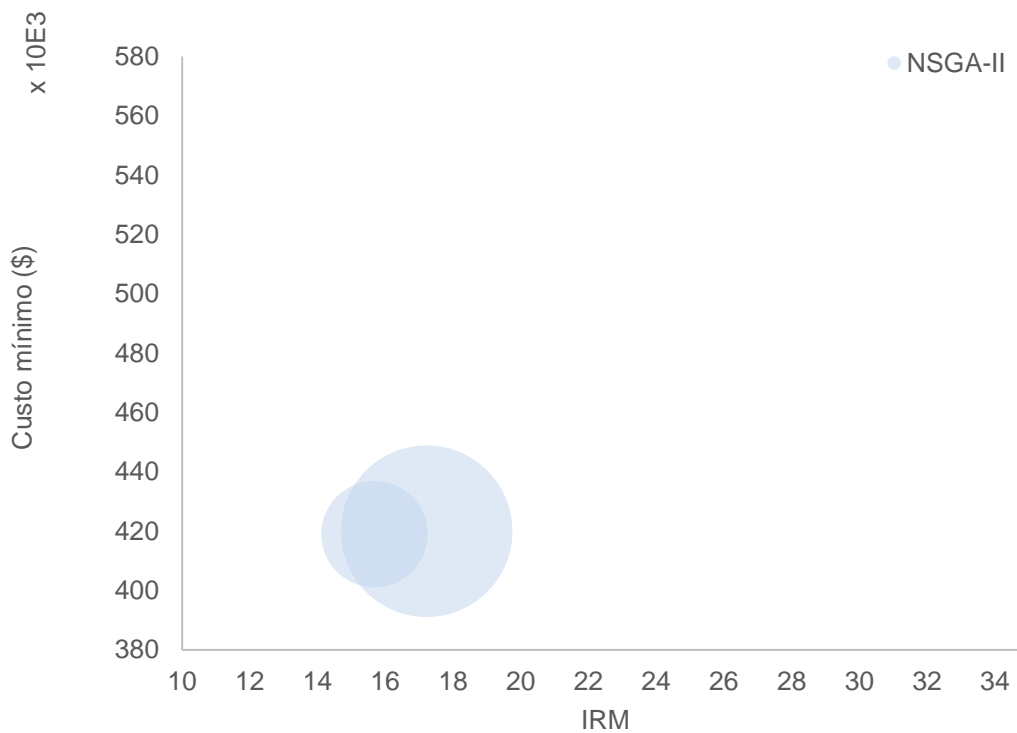


Figura 15. Frente Pareto de custos mínimos do NSGA-II

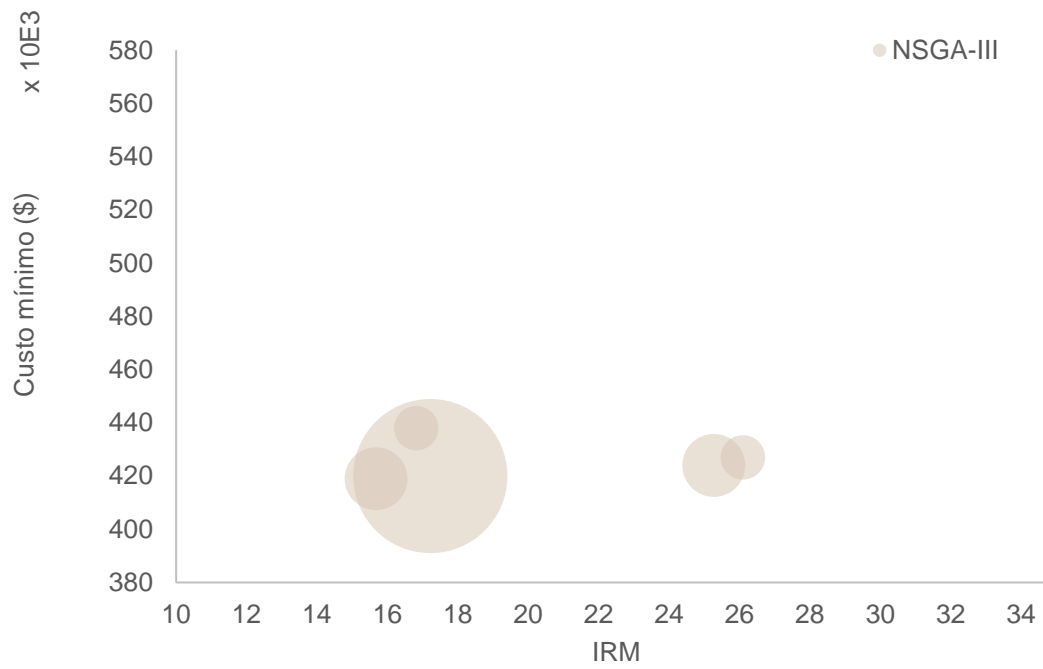


Figura 16. Frente Pareto de custos mínimos do NSGA-III

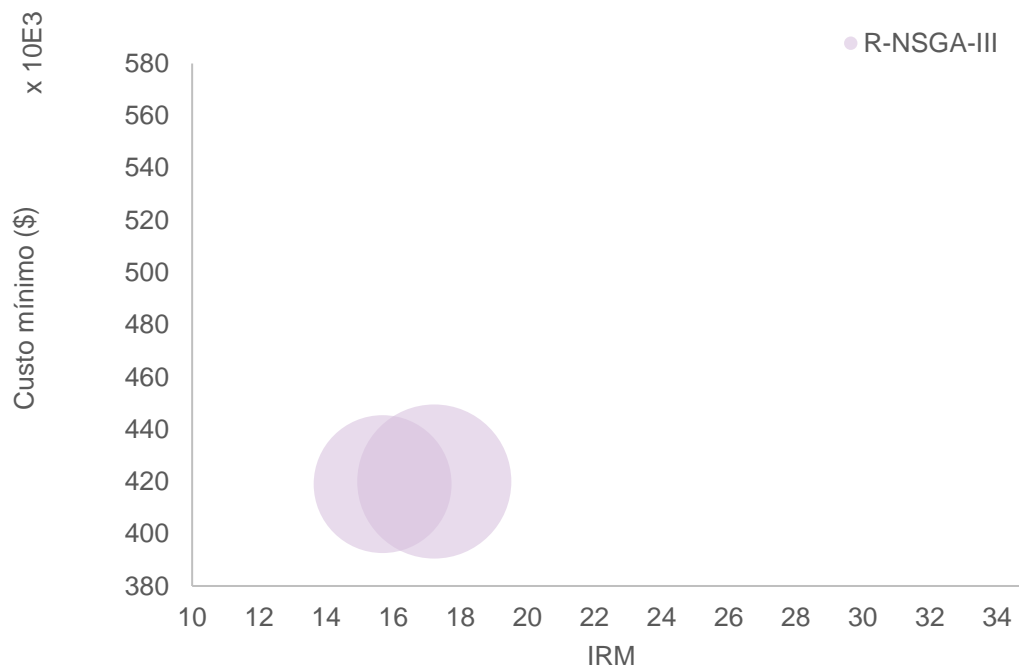


Figura 17. Frente Pareto de custos mínimos do R-NSGA-III

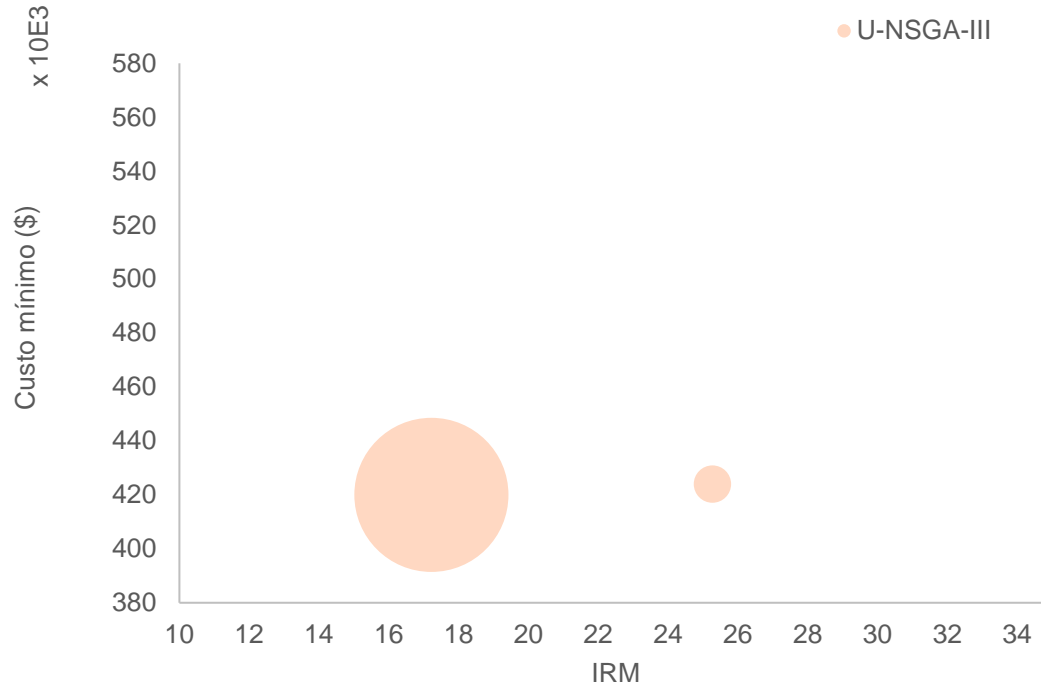


Figura 18. Frente Pareto de custos mínimos do U-NSGA-III

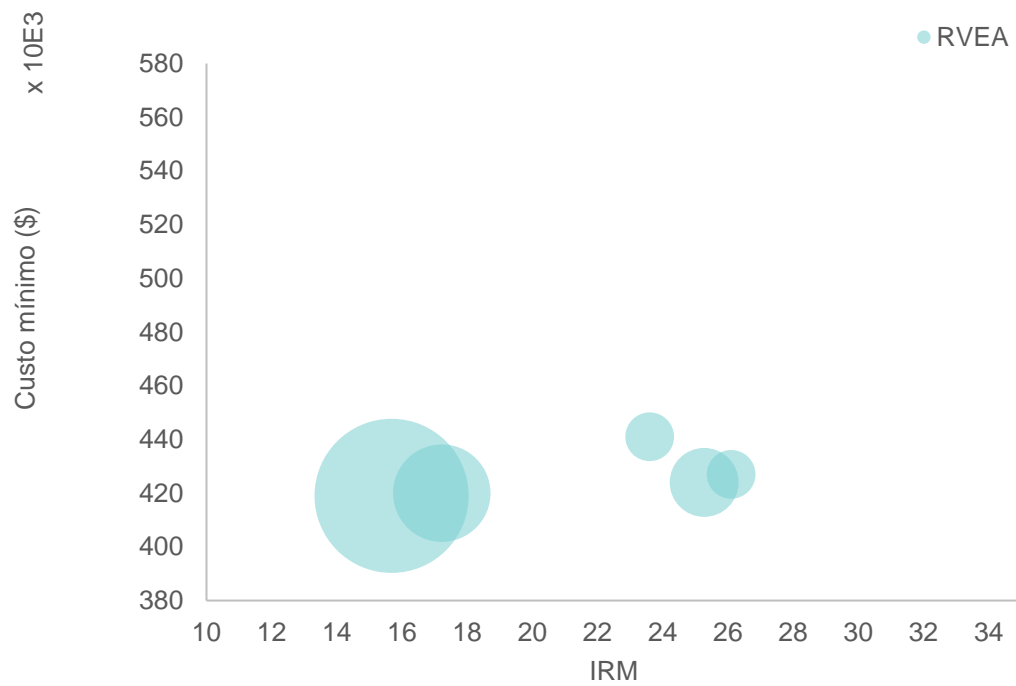


Figura 19. Frente Pareto de custos mínimos do RVEA

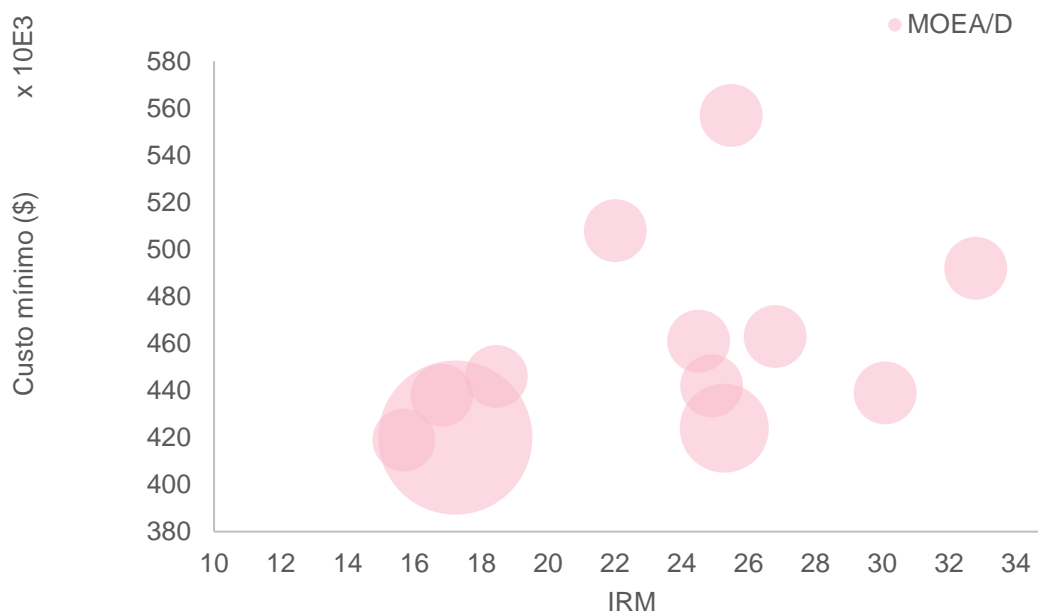


Figura 20. Frente Pareto de custos mínimos do MOEA/D

A Figura 21 ilustra o critério de Eficiência (E) obtido para os seis algoritmos ao considerar todas as simulações. Isso significa que os parâmetros referentes ao número de sucessos, simulações e avaliações das funções objetivo foram somados para cada agrupamento de população-recombinação-mutação. Os algoritmos MOEA/D, RVEA e R-NSGA-III apresentaram os maiores valores de E. No caso do MOEA/D, os picos de E são atribuídos ao elevado número de sucessos alcançados, ou seja, custos iguais ou inferiores a \$ 430.000.

Para o RVEA, o pico de E está relacionado ao baixo número de simulações e, conseqüentemente, de avaliações das funções objetivo. Em contraste, o R-NSGA-III demonstrou o mesmo número de sucessos, simulações e avaliações das funções objetivo para todos os agrupamentos, resultando em um valor constante de E. Os demais algoritmos apresentaram um comportamento bastante semelhante, especialmente para a população inicial de 100.

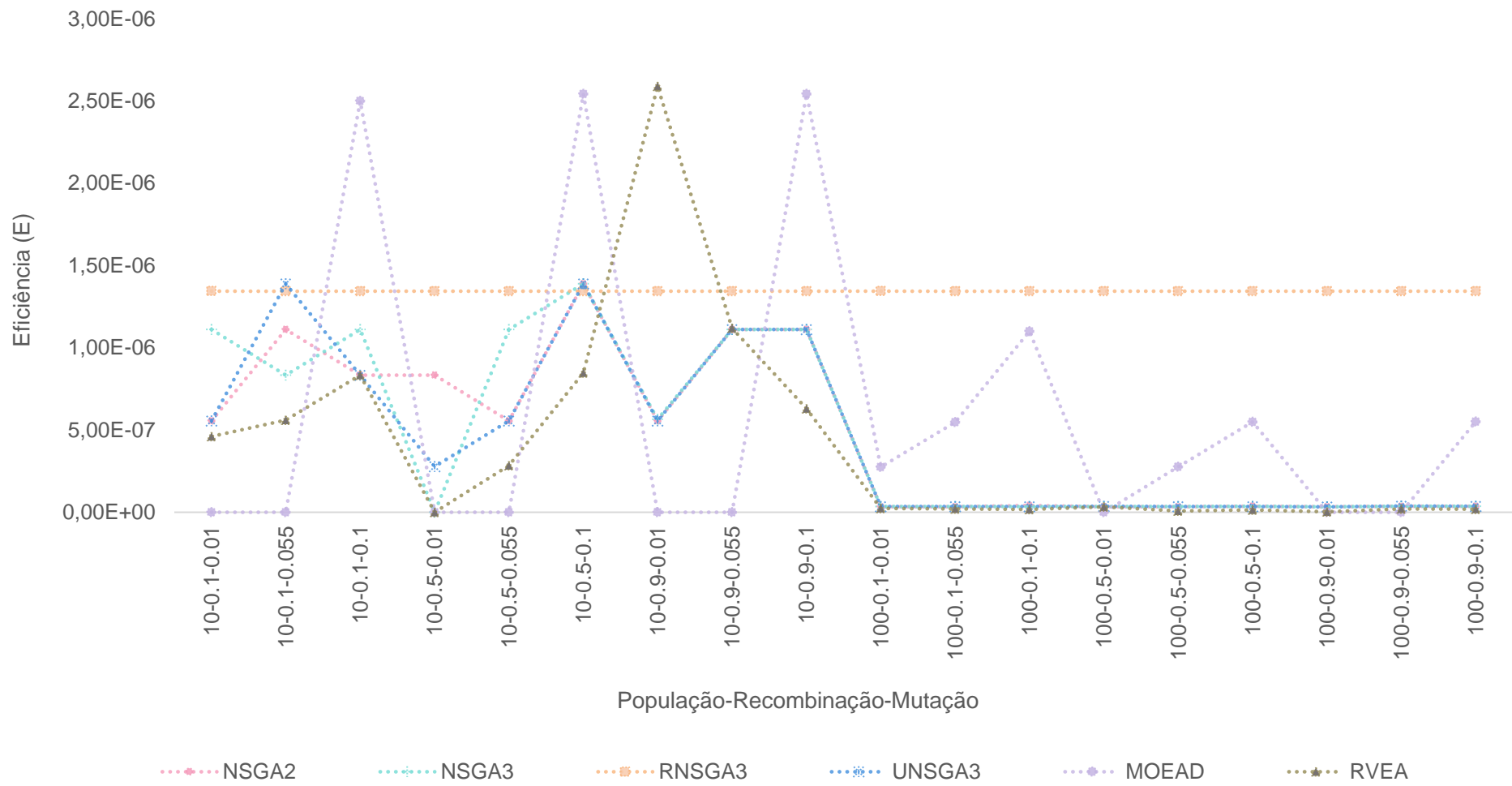


Figura 21. Eficiência (E) por algoritmo

O teste T de Student foi utilizado para avaliar os desvios em relação aos valores mínimos de custo obtidos através do coeficiente de variação amostral (C.O.V.). O C.O.V. representa a variabilidade dos custos mínimos em relação às suas respectivas médias, mostrada graficamente. Assim, a média dos custos mínimos em cada simulação para cada grupo foi calculada juntamente com o C.O.V. e o número de amostras necessário para realizar o teste foi igual ao número de sementes utilizadas.

As Figuras 22 e 23 ilustram os resultados categorizados pelas populações de 10 e 100, respectivamente. Quando a simulação hidráulica não atingiu a pressão mínima de 30 mca, resultando na aplicação de uma penalidade de custo de \$100.000, os resultados foram excluídos do cálculo do desvio padrão para não o comprometer.

Todos os algoritmos apresentaram C.O.V. maiores para a população de 10 em comparação com a população de 100, indicando maior sensibilidade deste parâmetro na obtenção de valores de custo mais baixos. Outro fator que corrobora essa evidência é que os custos mínimos médios foram menores para a população de 100 em comparação com a de 10. Já as variações nos parâmetros de taxa de mutação e crossover não foram significativas na redução de custos ou C.O.V..

Embora os algoritmos MOEA/D e RVEA tenham apresentado os maiores valores de E , seus C.O.V. foram maiores em comparação aos outros algoritmos, indicando maior variabilidade nos resultados em ambas as populações. Isso sugere que, embora esses algoritmos frequentemente atinjam valores mínimos de custo, também apresentam uma notável dispersão nas soluções encontradas.

A Tabela 4 apresenta as configurações dos parâmetros iniciais e os resultados para alcançar o custo mínimo de cada algoritmo. Como todas as simulações foram consideradas, os resultados de tempo, ns , $nsim$, nc , nq e E representam seus valores cumulativos para as seis simulações. O U-NSGA-III foi o único que não atingiu o custo mínimo de \$ 419.000 e sim de \$ 420.000. Adicionalmente, todos os algoritmos utilizaram uma taxa de recombinação de 0,1 para alcançar o valor mínimo, exceto o NSGA-II, que demandou uma taxa de recombinação de 0,9.

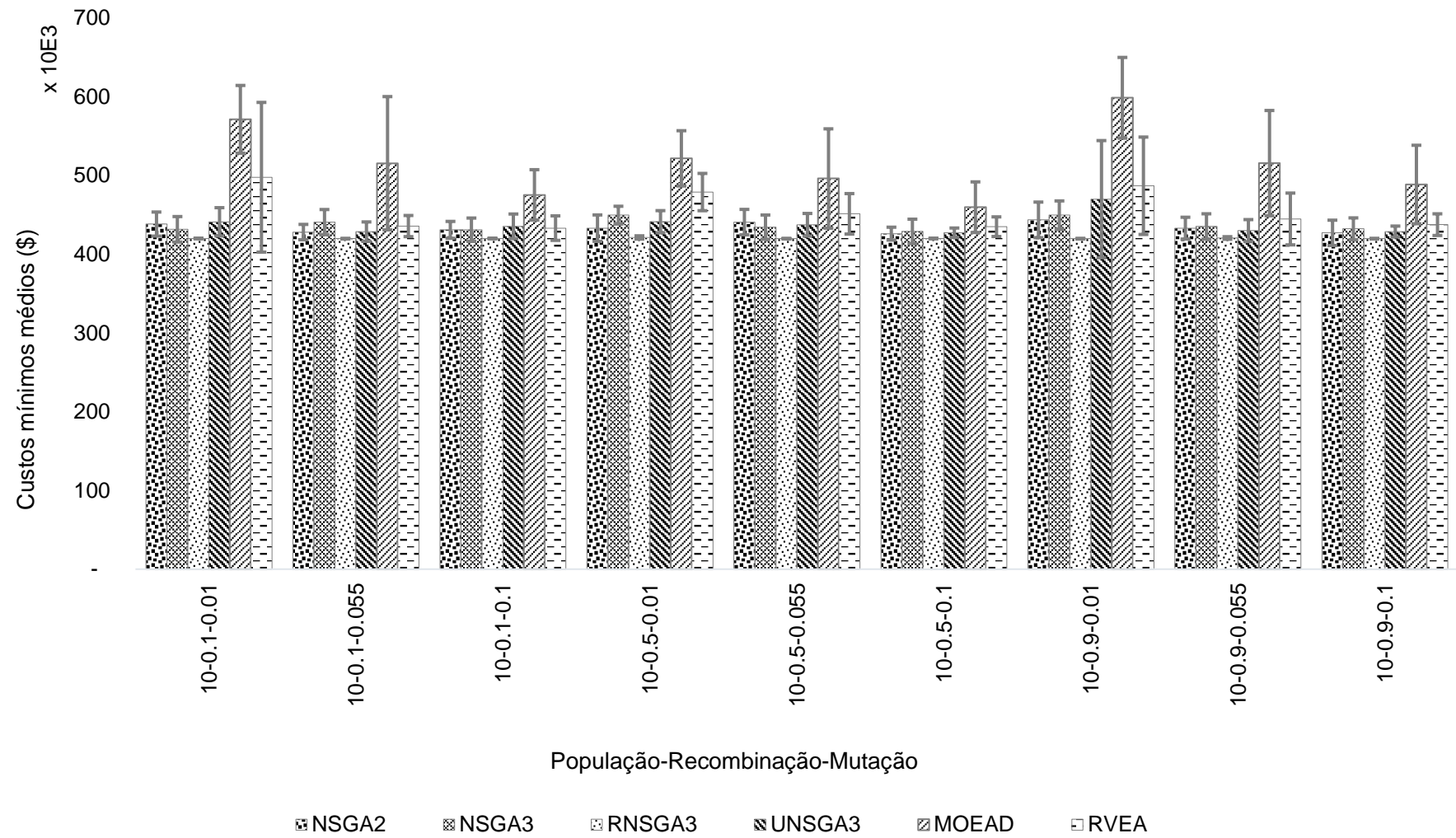


Figura 22. C.O.V. associados aos custos mínimos médios para a população de 10

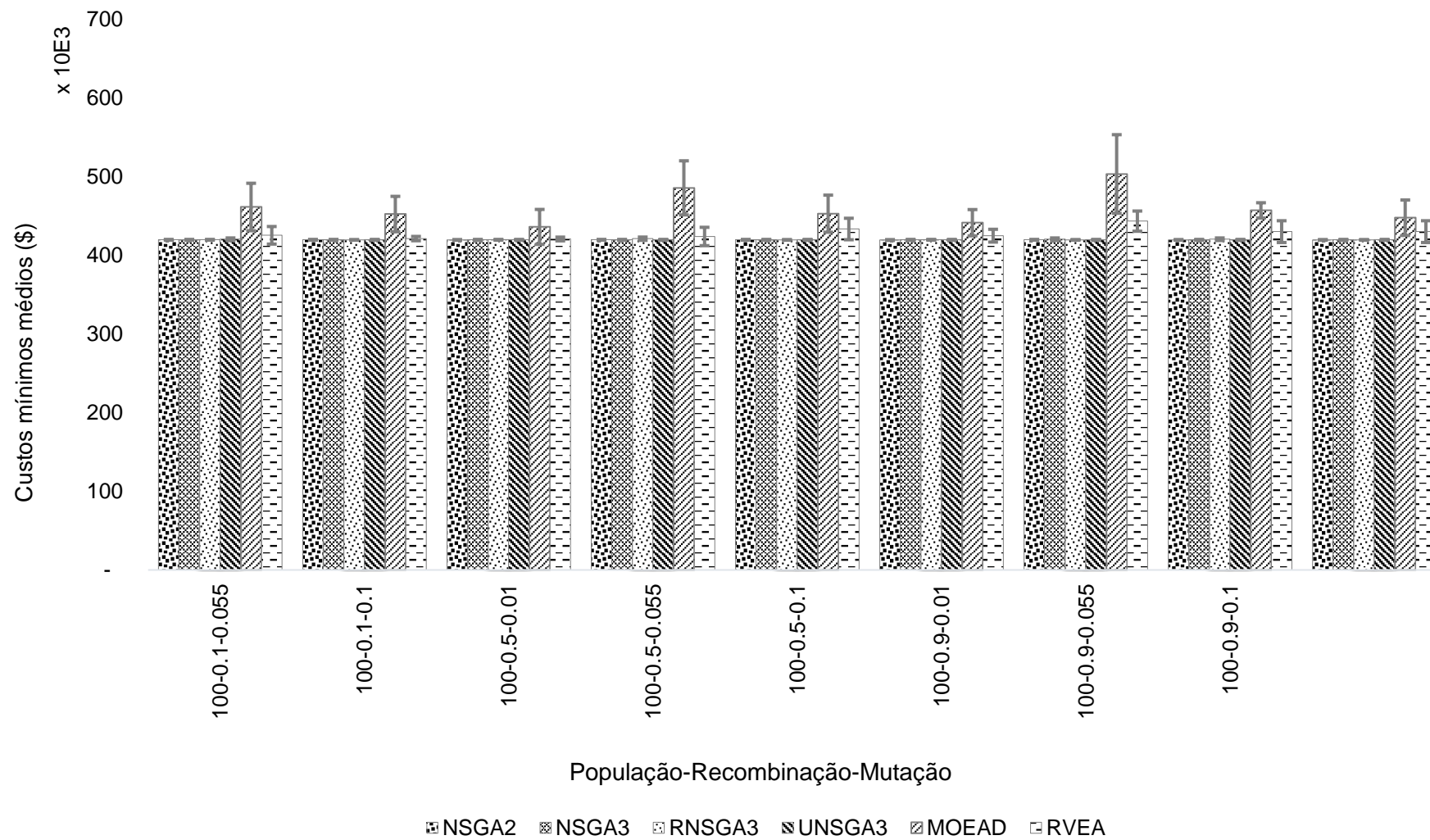


Figura 23. C.O.V. associados aos custos mínimos médios para a população de 100

Em relação ao tempo de processamento, NSGA-II, NSGA-III, U-NSGA-III e RVEA apresentaram tempos idênticos de 23 minutos, enquanto o MOEA/D demandou 44 minutos e o R-NSGA-III 154 minutos. Apesar do elevado tempo de processamento, o R-NSGA-III foi o algoritmo que apresentou a maior taxa de sucesso (n_q) devido ao menor número de simulações realizadas e à segunda maior Eficiência (E), sendo superado apenas pelo MOEA/D neste quesito e considerando essa configuração específica de população-recombinação-mutação. A Tabela 5 exibe os diâmetros ótimos calculados por cada algoritmo, juntamente com seus respectivos custos e valores de IRM.

Como o tempo de processamento na Tabela 4 é a somatória da duração das seis simulações para as configurações elencadas, a Figura 24 ilustra o tempo médio necessário para executar as 6 simulações completas, categorizada pelo tamanho da população. Além disso, foi calculado o C.O.V. utilizando o teste t de Student para o tempo médio, semelhante ao mostrado nas Figuras 22 e 23, anteriormente.

O MOEA/D apresentou um tempo médio de processamento significativamente maior em comparação aos outros algoritmos para a população de 100, enquanto o R-NSGA-III fez o mesmo para a população de 10. Em relação ao C.O.V. representado pelas barras, ele foi geralmente maior para a população de 100 em todos os algoritmos, exceto para o R-NSGA-III, que apresentou um C.O.V. maior para a população de 10 em comparação à população de 100.

É importante salientar que todos os algoritmos (à exceção do U-NSGA-III) atingiram o custo mínimo de \$419.000, indicando a adoção de um número de gerações excessivo para a resolução do problema. Isso ocorre devido à pequena dimensionalidade do estudo de caso adotado, onde os algoritmos convergem para o já conhecido mínimo global do problema. Logo, uma outra estratégia de comparação seria adotar a métrica do tempo médio somente até a obtenção do valor de custo mínimo de \$419.000, ou adotar estudos de caso maiores, onde os resultados de custo mínimo obtidos pelos algoritmos fossem diferentes.

Tabela 4. Resultados das simulações para o custo mínimo

NSGA-II										
População	Recombinação	Mutação	Custo mínimo (\$)	IRM	Tempo (min)	n _s	n _{sim}	n _c	n _q	E
10	0.9	0.01	419.000	15.68	23	2	60	60000	3.3%	5.56E-07
NSGA-III										
População	Recombinação	Mutação	Custo mínimo (\$)	IRM	Tempo (min)	n _s	n _{sim}	n _c	n _q	E
10	0.1	0.01	419.000	15.68	23	4	60	59993	6.7%	1.11E-06
U-NSGA-III										
População	Recombinação	Mutação	Custo mínimo (\$)	IRM	Tempo (min)	n _s	n _{sim}	n _c	n _q	E
10	0.1	0.01	420.000	17.22	23	2	60	59996	3.3%	5.56E-07
R-NSGA-III										
População	Recombinação	Mutação	Custo mínimo (\$)	IRM	Tempo (min)	n _s	n _{sim}	n _c	n _q	E
10	0.1	0.055	419.000	15.68	154	6	12	371996	50.0%	1.34E-06
MOEA/D										
População	Recombinação	Mutação	Custo mínimo (\$)	IRM	Tempo (min)	n _s	n _{sim}	n _c	n _q	E
10	0.1	0.1	419.000	15.68	44	9	60	60000	15.0%	2.50E-06
RVEA										
População	Recombinação	Mutação	Custo mínimo (\$)	IRM	Tempo (min)	n _s	n _{sim}	n _c	n _q	E
10	0.1	0.055	419.000	15.68	23	2	60	59518	3.3%	5.60E-07

Tabela 5. Diâmetros ótimos

Trecho	NSGA-II, NSGA-III, R-NSGA-III, RVEA e MOEA/D	U-NSGA-III
	Diâmetros (mm)	
1	457,2	508
2	254	254
3	406,4	406,4
4	101,6	25,4
5	406,4	355,6
6	254	254
7	254	254
8	25,4	25,4
Custo (\$)	419.000	420.000
IRM	15,68	17,22

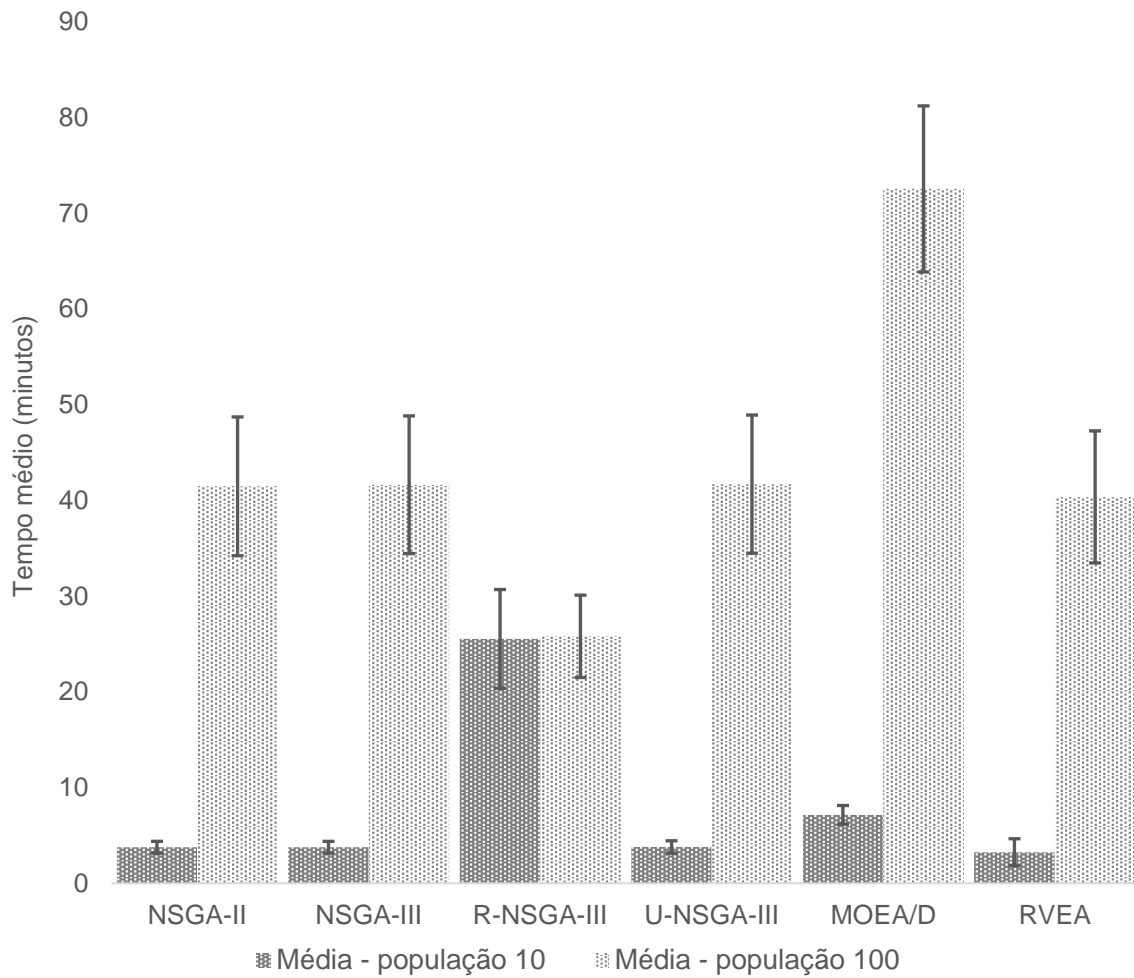


Figura 24. C.O.V. associados aos tempos médios de processamento

A FDA foi calculada utilizando os resultados obtidos a partir dos parâmetros descritos na Tabela 4 para analisar a variabilidade dos diâmetros listados. Os valores da FDA para os algoritmos NSGA-II, NSGA-III e U-NSGA-III são muito próximos, sugerindo que diâmetros semelhantes foram obtidos no resultado final das simulações. Já a FDA do RVEA, embora também semelhante aos algoritmos mencionados, diverge dos outros no início do gráfico devido à maior utilização do diâmetro de 25,4 mm (aproximadamente 14% do total). Adicionalmente, o MOEA/D é o único que elencou significativamente os diâmetros de 254 mm e 304,8 mm, evidenciado pela inclinação acentuada de seu gráfico nesses pontos. Por fim, nos resultados do R-NSGA-III houve a predominância de diâmetros maiores, como 355,6 mm e principalmente 558,8 mm, em detrimento de diâmetros menores.

A análise gráfica da FDA possibilita a visualização da ocorrência dos diâmetros nos resultados obtidos. Embora as seis curvas dos algoritmos sejam distintas, não é possível afirmar que um algoritmo tenha apresentado um resultado superior a outro considerando apenas essa métrica. Isso porque os algoritmos geraram aleatoriamente os resultados, sem restrições específicas para priorizar a menor variação de diâmetros, o que seria melhor do ponto de vista operacional para reduzir as perdas de carga nas redes. A FDA pode ser visualizada na Figura 25.

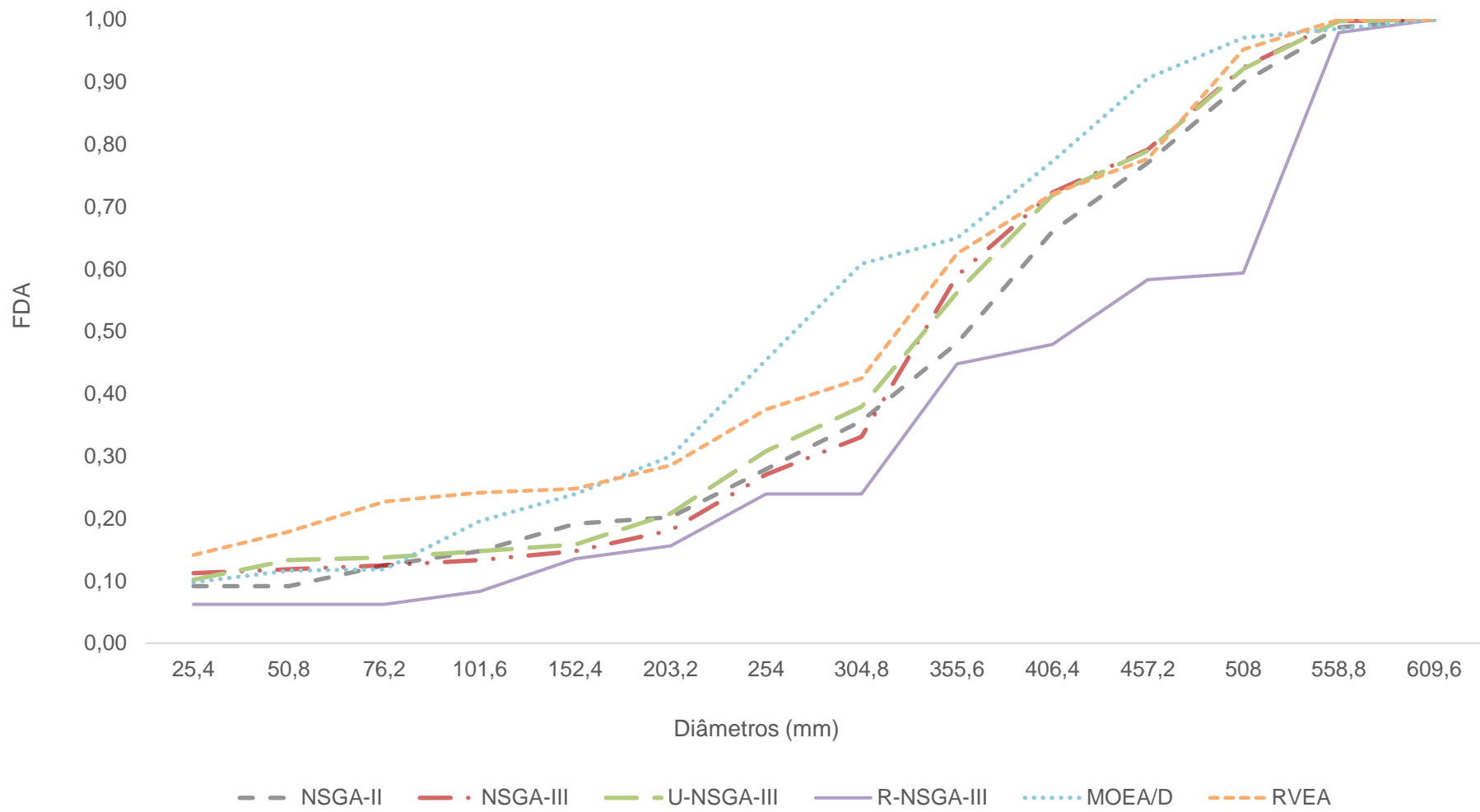


Figura 25. Função de distribuição acumulada dos diâmetros

Outro aspecto significativo explorado neste trabalho foi a complexidade dos algoritmos, relacionado à análise da formulação matemática de cada algoritmo ao invés da avaliação dos resultados gerados. A análise de complexidade quantifica o esforço computacional necessário por um algoritmo como uma função do tamanho dos dados de entrada. Essa análise oferece informações valiosas para comparar e selecionar os algoritmos mais adequados com base em fatores como o tamanho dos dados de entrada, os recursos computacionais disponíveis e o desempenho desejado, por exemplo.

Convencionalmente, a notação O é utilizada para denotar a complexidade de tempo, representando-o como uma função do tamanho dos dados de entrada. A Tabela 6 descreve a complexidade dos seis algoritmos, onde m representa o número de funções objetivo, N é o tamanho da população inicial, e T representa o número de soluções. NSGA-II, U-NSGA-III e RVEA apresentam complexidade idêntica, expressa em termos de m e de uma relação quadrática com N . Por outro lado, MOEA/D demonstra uma relação linear entre m , N e T , enquanto NSGA-III e R-NSGA-III compartilham a mesma complexidade, com uma relação quadrática com N e uma relação logarítmica com $(M-2)$.

Considerando o tempo médio mostrado na Figura 24, torna-se evidente que algoritmos com complexidade similar, como NSGA-II, U-NSGA-III e RVEA, exibem tempos e C.O.V. comparáveis. No entanto, NSGA-III e R-NSGA-III apresentam tempos e C.O.V. marcadamente diferentes nas duas populações. Uma possível explicação para essas disparidades pode ser atribuída aos pontos de referência utilizados no R-NSGA-III, que necessitam de pré-processamento para o ponto de partida. Esse pré-processamento geraria uma sobrecarga computacional adicional, especialmente evidente em populações menores, como 10. Entretanto, para populações maiores que excedem 100, o pré-processamento se tornaria mais eficiente, classificando o R-NSGA-III como o algoritmo mais rápido entre todos.

Outro fator que contribui para os atrasos nos algoritmos com funções de complexidade logarítmicas pode ser a alocação de pontos de referência em regiões subótimas do espaço de soluções, resultando em processamento ineficiente. Vesikar et al. (2008) discutem esse problema e sugerem realocar os pontos para regiões onde soluções não dominadas já foram descobertas como uma melhoria para a eficiência do algoritmo.

O MOEA/D foi o único algoritmo que apresentou tanto complexidade linear quanto dependência da variável T , relativa ao número de soluções. Embora a linearidade pressuponha um algoritmo mais rápido, sua relação com T posiciona o MOEA/D como o que apresenta o pior desempenho em termos de eficiência de tempo. Essa observação levanta a hipótese de que pode haver um domínio de T em detrimento de outras variáveis no desempenho do algoritmo.

Tabela 6. Complexidade dos algoritmos

	Algoritmo					
	NSGA-II	NSGA-III	R-NSGA-III	U-NSGA-III	MOEA/D	RVEA
Complexidade	$O(m \cdot N^2)$	$O(m \cdot N^2)$ ou $O(N^2 \cdot \log N^{(M-2)})$	$O(m \cdot N^2)$ ou $O(N^2 \cdot \log N^{(M-2)})$	$O(m \cdot N^2)$	$O(m \cdot N \cdot T)$	$O(m \cdot N^2)$

6 – CONCLUSÕES

Entre as diversas técnicas de otimização aplicadas a projetos de redes de distribuição de água, os algoritmos evolucionários se destacam como técnicas de otimização versáteis, cada um com suas características e metodologias únicas. O desenvolvimento e a disseminação do uso desses algoritmos motivaram a necessidade de elaborar estudos comparativos para discernir suas vantagens e desvantagens, buscando guiar a escolha do algoritmo às necessidades dos usuários.

Neste estudo, seis algoritmos evolucionários - NSGA-II, NSGA-III, R-NSGA-III, U-NSGA-III, MOEA/D e RVEA - foram submetidos a uma comparação usando uma variedade de métricas. Essas métricas incluíram Eficiência, Função de Distribuição Acumulada (FDA), análise de frente de Pareto, complexidade de algoritmos e C.O.V. relacionados ao custo mínimo e tempo de execução. O estudo de caso utilizado para avaliação dos resultados foi a rede de Alperovitz e Shamir.

Com base em todas as métricas utilizadas, observou-se que, à exceção do U-NSGA-III, todos os algoritmos atingiram o custo mínimo de \$419.000, demonstrando sua bem-sucedida convergência para o mínimo encontrado em outros estudos que aplicaram a mesma RDA.

Além disso, constatou-se que, entre todas as variáveis iniciais, apenas a população causou mudanças significativas na minimização dos custos e do C.O.V..

A investigação das métricas utilizadas gerou conclusões sobre o desempenho dos algoritmos. Verificou-se que nenhum algoritmo se destacou como universalmente superior em todas as métricas; cada algoritmo exibiu superioridade em algumas métricas, mas foi inferior em outras. Esse entendimento detalhado enfatiza a importância de utilizar análises diversas para obter uma visão completa do desempenho dos algoritmos.

Além disso, a exploração da complexidade algorítmica destacou as formulações matemáticas que regem cada algoritmo. Curiosamente, algoritmos com níveis de complexidade semelhantes apresentaram resultados divergentes em termos de tempo devido a variações nos parâmetros iniciais e nas metodologias de busca por soluções não dominadas dentro do espaço de soluções. Adicionalmente, o algoritmo com complexidade linear, pressuposto como o mais rápido devido à sua estrutura matemática mais simples, revelou-se o menos eficiente em termos de tempo.

Na avaliação geral, NSGA-II e NSGA-III surgiram como algoritmos superiores, demonstrando melhor desempenho em várias métricas. Eles exibiram picos satisfatórios de Eficiência, atingiram baixos custos mínimos e obtiveram tempos de processamento eficientes, minimizando o C.O.V..

Por outro lado, o U-NSGA-III não atingiu o limiar de custo mínimo, o R-NSGA-III apresentou o maior tempo de processamento para a população de 10, e MOEA/D e RVEA exibiram altos custos e C.O.V..

Em conclusão, a análise comparativa conduzida neste estudo oferece informações valiosas sobre o desempenho dos AG no contexto da otimização de RDA. Ao avaliar sistematicamente várias métricas e considerar as complexidades dos algoritmos, os tomadores de decisão podem adquirir uma compreensão detalhada dos aspectos positivos e negativos de cada algoritmo. Com esse conhecimento, companhias de saneamento podem desenvolver processos de tomadas de decisão mais completos, escolhendo uma técnica de inteligência artificial que aumente a probabilidade de alcançar melhores resultados dentro das restrições desejadas, como por exemplo o NSGA-III.

Pesquisas futuras poderiam expandir a análise incorporando mais objetivos aos SDA, bem como utilizar estudos de caso maiores e mais complexos para avaliar o ganho ao aplicar uma técnica de otimização robusta. Sistemas simplificados usualmente não representam as complexas interações existentes em redes reais, como variações de demanda, perdas por vazamentos nas redes e requisitos hidráulicos operacionais, entre outros. Assim, a comparação de algoritmos que utilizassem dados reais poderia fornecer maior compreensão sobre a eficiência dessas técnicas, aliando o desenvolvimento acadêmico às necessidades das companhias de saneamento.

REFERÊNCIAS BIBLIOGRÁFICAS

- ALPEROVITS, E., SHAMIR, U. (1977). “Design of optimal water distribution systems”. In: *Water Resources Research*, 13(6).
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. (2017). “NBR 12218: Projeto de rede de distribuição de água para abastecimento público – Procedimento”.
- Asafuddoula, M., Ray, T., & Sarker, R. (2015). “A decomposition-based evolutionary algorithm for many objective optimization”. In: *IEEE Transactions on Evolutionary Computation*, 19(3), 445–460.
- Cheng, R., Jin, Y., Olhofer, M., & Sendhoff, B. (2016). “A Reference Vector Guided Evolutionary Algorithm for Many-Objective Optimization”. In: *IEEE Transactions on Evolutionary Computation*, 20(5), 773–791.
- Cheng, R., Li, M., Tian, Y., Zhang, X., Yang, S., Jin, Y., & Yao, X. (2017). “A benchmark test suite for evolutionary many-objective optimization”. In: *Complex & Intelligent Systems*, 3(1), 67–81.
- Creaco, E., & Pezzinga, G. (2015). “Embedding linear programming in multi objective genetic algorithms for reducing the size of the search space with application to leakage minimization in water distribution networks”. In: *Environmental Modelling and Software*, 69, 308–318.
- Das, I., & Dennis, J. E. (1998). “Normal-boundary intersection: A new method for generating the pareto surface in nonlinear multicriteria optimization problems”. In: *SIAM Journal on Optimization*, vol. 8, no. 3, pp. 631–657.
- Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A. (2002). “A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II”. In: *IEEE Transactions on Evolutionary Computation*, 6(2), 182-197.
- Deb, K., Thiele, L., Laumanns, M., Zitzler, E. (2005). *Scalable Test Problems for Evolutionary Multiobjective Optimization*. In: Abraham, A., Jain, L., Goldberg, R. (eds) *Evolutionary Multiobjective Optimization*. Advanced Information and Knowledge Processing. Springer, London. https://doi.org/10.1007/1-84628-137-7_6
- Deb, K., & Jain, H. (2014). “An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: Solving problems with box constraints”. In: *IEEE Transactions on Evolutionary Computation*, 18(4), 577–601.

- El-Ghandour, H. A., Elbeltagi, E. (2018). “Comparison of five evolutionary algorithms for optimization of water distribution networks”. In: *Journal of Computing in Civil Engineering*, 32(1):04017066.
- Farmani, R.; Walters, G.; Savic, D. (2006). “Evolutionary multi-objective optimization of the design and operation of water distribution network: total cost vs. reliability vs. water quality”. In: *Journal of Hydroinformatics*, v. 8, n. 3, p. 165–179.
- Fu, G., Kapelan, Z., Kasprzyk, J. R., & Reed, P. (2013). “Optimal Design of Water Distribution Systems Using Many-Objective Visual Analytics”. In: *Journal of Water Resources Planning and Management*, 139(6), 624–633.
- Garzón, A., Kapelan, Z., Langeveld, J., & Taormina, R. (2022). “Machine Learning-Based Surrogate Modeling for Urban Water Networks: Review and Future Research Directions.” In: *Water Resources Research*, 58(5).
- He, X., Zhou, Y., Chen, Z., & Zhang, Q. (2017). “Evolutionary Many-objective Optimization based on Dynamical Decomposition.” In: *IEEE Transactions on Evolutionary Computation*, vol. XX.
- Holland, J. H. (1975). *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press.
- Huband, S., Hingston, P., Barone, L., & While, L. (2006). “A review of multiobjective test problems and a scalable test problem toolkit.” In: *IEEE Transactions on Evolutionary Computation*, Vol. 10, Número 5, p. 477–506.
- Ishibuchi, H., Hitotsuyanagi, Y., Tsukamoto, N., & Nojima, Y. (2010). “Many-objective test problems to visually examine the behavior of multiobjective evolution in a decision space”. In: *International Conference on Parallel Problem Solving from Nature*, Springer, pp. 91–100.
- Jain, H., & Deb, K. (2013). “An Evolutionary Many-Objective Optimization Algorithm Using Reference-point Based Non-dominated Sorting Approach, Part II: Handling Constraints and Extending to an Adaptive Approach”. In: *IEEE Transactions on Evolutionary Computation*. 18(4):602–622. <http://www.egr.msu.edu/>
- Jayaram, N., Srinivasan, K. (2008). “Performance-based optimal design and rehabilitation of water distribution networks using life cycle costing”. In: *Water Resources Research*, 44(1), 1–15.
- Jiang, S., & Yang, S. (2017). “A strength pareto evolutionary algorithm based on reference direction for multiobjective and many-objective optimization”. In: *IEEE Transactions on Evolutionary Computation*, 21(3), 329–346.

- Johns, M. B., Keedwell, E., & Savic, D. (2020). “Knowledge-based multi-objective genetic algorithms for the design of water distribution networks”. In: *Journal of Hydroinformatics*, 22(2), 402–422.
- Kapelan, Z., Savic, D. A., & Walters, G. A. (2003). “Multi-objective sampling design for water distribution model calibration”. In: *ASCE Journal of Water Resources Planning and Management*, 129 (6) : 466 – 479 .
- Köppen, M., & Yoshida, K. 2007). “Substitute Distance Assignments in NSGA-II for Handling Many-objective Optimization Problems.” In: *Evolutionary Multi-Criterion Optimization*, Springer, 727--741.
- Li, K., Deb, K., Zhang, Q., & Kwong, S. (2015). “An evolutionary many-objective optimization algorithm based on dominance and decomposition.” In: *IEEE Transactions on Evolutionary Computation*, 19(5), 694–716.
- Li, M., Grosan, C., Yang, S., Liu, X., & Yao, X. (2018). “Multiline Distance Minimization: A Visualized Many-Objective Test Problem Suite.” In: *IEEE Transactions on Evolutionary Computation*, 22(1), 61–78.
- Maier, H. R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L. S., Cunha, M. C., Dandy, G. C., Gibbs, M. S., Keedwell, E., Marchi, A., Ostfeld, A., Savic, D., Solomatine, D. P., Vrugt, J. A., Zecchin, A. C., Minsker, B. S., Barbour, E. J., Kuczera, G., Pasha, F., Castelletti, A., & Reed, P. M. (2014). “Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions.” In: *Environmental Modelling and Software*, 62, 271–299.
- Marques, J., Cunha, M., & Savić, D. (2018). “Many-objective optimization model for the flexible design of water distribution networks.” In: *Journal of Environmental Management*, 226, 308–319.
- Mora-Melia, D, Iglesias-Rey, P. L., Martinez-Solano, F. J., Ballesteros-Pérez, P. (2015). “Efficiency of evolutionary algorithms in water network pipe sizing”. In: *Water Resources Management*, 29:4817–4831
- Nicolini, M., Giacomello, C., and Deb, K. (2011). “Calibration and optimal leakage management for a real water distribution network.” In: *Journal of Water Resources Planning and Management*, 137(1), 134–142.
- Rapp, A.H., Capener, A.M., Sowby, R.B. (2023). “Adoption of Artificial Intelligence in Drinking Water Operations: A Survey of Progress in the United States”. In: *Journal of Water Resources Planning and Management*, 149, 06023002.

- Rossman, L. A. (2000). *EPANET 2: Users manual*. US EPA: US Environmental Protection Agency National Risk Management Research Laboratory.
- Salgado, R.; Todini, E.; O'Connell, P.E. (1988). "Extending the gradient method to include pressure regulating valves in pipe networks". In: *Proc. Inter. Symposium on Computer Modeling of Water Distribution Systems*, University of Kentucky, May 12-13.
- Seada, H., & Deb, K. (2016). "A Unified Evolutionary Optimization Procedure for Single, Multiple, and Many Objectives." In: *IEEE Transactions on Evolutionary Computation*, 20(3), 358–359.
- Sharma, A. N., Dongre S. R., Gupta R. & Ormsbee L. (2022). "Multiphase procedure for identifying district metered areas in water distribution networks using community detection, NSGA-III optimization, and multiple attribute decision making". In: *Journal of Water Resources Planning and Management*, 148(8), 04022040.
- Sun, Y., Yen, G. G., & Yi, Z. (2019). "IGD Indicator-Based Evolutionary Algorithm for Many-Objective Optimization Problems." In: *IEEE Transactions on Evolutionary Computation*, 23(2), 173–187.
- Todini, E.; Pilati, S. (1987). "A gradient method for the analysis of pipe networks". In: *International Conference on Computer Applications for Water Supply and Distribution*, Leicester Polytechnic, UK, September 8-10.
- Tsutiya, M. T. (2006). *Abastecimento de Água (4a edição)*. Departamento de Engenharia Hidráulica e Sanitária da Escola Politécnica da Universidade de São Paulo.
- Vesikar, Y., Deb, K., Blank, J. (2018). "Reference point based NSGA-III for preferred solutions". In: *IEEE Symposium Series on Computational Intelligence (SSCI)*, 1587–1594. doi:10.1109/SSCI.2018.8628819.
- Walker, D. J., & Craven, M. J. (2020). "Identifying good algorithm parameters in evolutionary multi- and many-objective optimisation: A visualisation approach." In: *Applied Soft Computing Journal*, 88.
- Walski, T. M., Brill, E. D., Gessler, J., Goulter, I. C., Jeppson, R. M., Lansey, K., Han-Lin Lee, , Liebman, J. C., Mays, L., Morgan, D. R., & Ormsbee, L. (1987). "Battle of the network models: epilogue." In: *Journal of Water Resources Planning and Management*, ASCE, 113 (2), 191-203.
- Walski, T.M. (2001). "The wrong paradigm—Why water distribution optimization doesn't work". In: *Journal of Water Resources Planning and Management*, 127, 203–205.

- Xiong, W., Li, Y., Zhang, W., Ye, Q., Zhang, S., & Hou, X. (2018). “Integrated multi-objective optimization framework for urban water supply systems under alternative climates and future policy”. In: *Journal of Cleaner Production*, 195, 640–650.
- Yazdi, J. (2016). “Decomposition based Multi Objective Evolutionary Algorithms for Design of Large-Scale Water Distribution Networks”. In: *Water Resources Management*, 30, 2749–2766.
- Zarei, N., Azari, A., & Heidari, M. M. (2022). “Improvement of the performance of NSGA-II and MOPSO algorithms in multi-objective optimization of urban water distribution networks based on modification of decision space.” In: *Applied Water Science*, 12(6).
- Zhao, H., Zhang, C., Ning, J., Zhang, B., Sun, P., Feng, Y. (2019). “A comparative study of the evolutionary many-objective algorithms”. In: *Progress in Artificial Intelligence*, 8:15–43
- Zhang, Q., & Li, H. (2007). “MOEA/D: A multiobjective evolutionary algorithm based on decomposition.” In: *IEEE Transactions on Evolutionary Computation*, 11(6), 712–731.
- Zheng, F., Zecchin, A. C., Maier, H. R., & Simpson, A. R. (2016). “Comparison of the Searching Behavior of NSGA-II, SAMODE, and Borg MOEAs Applied to Water Distribution System Design Problems.” In: *Journal of Water Resources Planning and Management*, 142(7).
- Zou, J., Zhang, Z., Zheng, J., & Yang, S. (2021). “A many-objective evolutionary algorithm based on dominance and decomposition with reference point adaptation.” In: *Knowledge-Based Systems*, 231.

1 APÊNDICE A – ALGORITMO IMPLEMENTADO

2
3 A seguir, é declarado um trecho de pseudocódigo que contém o algoritmo dos experimentos
4 executados nessa dissertação.

5 As linhas 21 a 32 definem as variáveis iniciais. A escolha do algoritmo utilizado é realizada
6 entre as linhas 37 e 59. A seguir, nas linhas 66 a 97 são declaradas as variáveis necessárias
7 para o cálculo do EPANET, bem como o contador para o cálculo da taxa de sucesso n_q . Nas
8 linhas 106 a 163 são calculadas as funções objetivo bem como é realizada a simulação
9 hidráulica do estudo de caso, considerando as restrições e a penalidade de aumento de custo
10 frente à pressão mínima menor que 30 mca. Por fim, nas linhas 171 a 226 são declaradas as
11 chamadas necessárias para execução do algoritmo, bem como é realizada a exportação dos
12 resultados.

13
14 ##### Algorithm Selection #####

15 // This section defines the general function that calls the execution of a specific learning
16 algorithm.

17 Import necessary libraries and modules

18 Define dictionary REF with available algorithms:

19 - "UNSGA3" mapped to UNSGA3 class

20 - "RNSGA3" mapped to RNSGA3 class

21 - "NSGA2" mapped to NSGA2 class

22 - "NSGA3" mapped to NSGA3 class

23 - "MOEAD" mapped to MOEAD class

24 - "CTAEA" mapped to CTAEA class

25 - "RVEA" mapped to RVEA class

26
27 Function SelectAlgorithm(name, pop_size = 10, samp = IntegerRandomSampling(), co =
28 SBX(prob = 0.8, vtype = int), mt = PM(prob = 0.050, vtype = int), no = 2, nd = 2, rd
29 = get_reference_directions("das-dennis", 2, n_partitions = 10 - 1, seed = 1)):

30
31 Set population_size to pop_size

32 Set sampling to samp

33 Set crossover to co

34 Set mutation to mt

```

35     Set n_objectives to no
36     Set n_dimensions to nd
37     Set ref_dirs to rd
38     Set ref_points to array([[0.1, 0.5], [0.7, 0.3], [0.5, 0.5]])
39
40     If name is not in REF:
41         Raise Exception "Reference directions factory not found."
42
43     If name is "NSGA3" or "UNSGA3":
44         return REF[name] with parameters: ref_dirs, population_size, sampling, crossover,
45             mutation
46     Else if name is "RNSGA3":
47         return REF[name] with parameters: ref_points, population per reference point, mu,
48             sampling, crossover, mutation
49     Else if name is "NSGA2":
50         return REF[name] with parameters: population_size, sampling, crossover, mutation
51     Else if name is "MOEAD":
52         return REF[name] with parameters: ref_dirs, number of neighbors, probability of
53             neighbor mating, sampling, crossover, mutation
54     Else if name is "SMSEMOA":
55         return REF[name] with parameters: population_size, sampling, crossover, mutation
56     Else if name is "AGEMOEAO":
57         return REF[name] with parameters: population_size, sampling, crossover, mutation
58     Else if name is "AGEMOEAO2":
59         return REF[name] with parameters: population_size, sampling, crossover, mutation
60     Else if name is "CTAEA":
61         return REF[name] with parameters: ref_dirs, sampling, crossover, mutation
62     Else if name is "RVEA":
63         return REF[name] with parameters: ref_dirs, population_size, sampling, crossover,
64             mutation
65 End function
66
67 #####     EPANET Problem     #####

```

```

68 // Especification of the problem object inhereting from ElementwiseProblem from the Epyt
69 library. The object calls the UDP Client and TCP Server to communicate with the containers
70 that will execute the objective function.
71
72 Class EPANETProblem (inherits from ElementwiseProblem)
73     Method __init__(counter = False, other_arguments):
74
75         Set Xmin to empty list
76         Set Xmax to empty list
77         Set NumberOfVariables to Globals.NUMBER_OF_PIPES
78         Set allowcounter to counter
79         Set overallSuccesses to 0
80
81         If allowcounter is True:
82             Set counter to 0
83         For each variable in NumberOfVariables:
84             Append 0 to Xmin
85             Append Globals.AVAILABLE_DIAMETERS - 1 to Xmax
86         For each port in Globals.availablePorts:
87             Set Globals.threadState[port] to 0
88         Call superclass initializer with n_var, n_obj, xl, xu, vtype, and other_arguments
89     End method
90
91     Method _evaluate(X, out, other_arguments):
92         Set diameter_pattern to string with the first element of X
93         For each element in X starting from the second element:
94             Append "," and element as string to diameter_pattern
95         Send UDP signal as client to ("localhost", port) with diameter_pattern as message
96         Set result to the response from TCPserver("localhost", thisPort + 500), split by spaces
97         If currentRes[0] <= 430000:
98             Increment overallSuccesses by 1
99         If allowcounter is True:
100             Increment counter by 1
101         Set out["F"] to currentRes

```

```

102     End Method
103 End Class
104
105 #####Container:Objective function#####
106
107 // The objective function where communication with the EPANET program is done to analyse
108 the received values
109
110 Import necessary libraries
111 Function ObjectiveFunction(splitDpInput):
112     Initialize diameter_pattern as empty list
113     For each index i from 0 to the length of splitDpInput - 1:
114         Append integer value of splitDpInput[i] to diameter_pattern
115     Initialize total_cost as 0
116     Initialize SUM_RI as 0
117     Initialize A as 0.0
118     Initialize B as 0.0
119     Initialize Hmin as 30.0
120
121     Load epanet network using the file './app/EPANET/Hanoi.inp'
122     Open hydraulic analysis
123     Initialize hydraulic analysis with time parameter 0
124     Get the number of links (Nlinks)
125     Get the number of nodes (Nnodes)
126     Get the number of tanks and reservoirs (Nres_tanks)
127     Calculate the number of junctions (Njunctions) as Nnodes - Nres_tanks
128     Initialize diameter as empty list
129     Initialize pipe_cost as empty list
130     Open file './app/Tabela_Custos_Hanoi.txt' for reading
131     Read the first line and get the number of diameters
132     Initialize diameter_base as empty list
133     Initialize cost_base as empty list
134     Skip the next line in the file
135     For each line IN the file:

```

```

136     Split line into current[]
137     Extract obs1, obs2, and obs3 from current
138     Append float value of obs2 to diameter_base
139     Append float value of obs3 to cost_base
140     Close the file
141
142     For index i from 0 to Nlinks - 1:
143         Set aux to diameter_pattern[i]
144         Append diameter_base[aux] to diameter
145         Set the link diameter for link i + 1 to diameter[i]
146         Get the pipe length for link i + 1
147         Append the product of cost_base[aux] and pipe_lenth to pipe_cost
148         ADDpipe_cost[i] to total_cost
149     Run hydraulic analysis and store the result in t
150
151     For each index i in the range of Njunctions:
152         Get the junction pressure for junction i + Nres_tanks
153         If junction_pressure is less than Hmin:
154             Add (Hmin - junction_pressure) * 10000000.0 to total_cost
155             Subtract (Hmin - junction_pressure) * 100.0 from SUM_RI
156         Else:
157             Get the junction demand for junction i + Nres_tanks
158             Set aux1 to junction_demand * (junction_pressure - Hmin)
159             Add aux1 to A
160             Set aux2 to junction_demand*Hmin
161             Add aux2 to b
162         If B is not equal to 0:
163             Add 100 * (A / B) to sum_RI
164     Close hydraulic analysis
165     Delete the network objective
166     Print the tuple (total_cost, -1 * sum_RI)
167 End function
168
169 #####      main      #####

```

```

170 // Main file, the one that will be executed. Calls the execution of the algorithms and writes
171 the results in CSV files
172
173 Import required libraries and modules
174 Set allOfThem to True
175 Set selectedAlgorithm to "NSGA3"
176 Set counter to True
177 Set currentRound to 0
178 Function SingleExecution(seed, populationSize, mutationRate, mutation, crossoverRate,
179     crossover, currentAlgorithm, runner, ref_dirs, seedRound):
180     Create EPANETProblem instance with counter and runner
181     Print currentAlgorithm
182     Set algorithm based on parameters:
183     - name, population size, sampling, crossover, mutation, objectives, dimensions, and
184     reference directions
185     Record start time of algorithm execution
186     Run optimization using minimize(problem, algorithm, Globals.stop_criteria, seed)
187     Calculate execution time
188     SAVE state to .savestate file
189 End function
190
191 Function ExecuteAlgorithms(**kwargs):
192     Declare global variables: currentRound, counter, diametersLabels, allOfThem,
193     selectedAlgorithm
194     Retrieve input parameters from kwargs (e.g., seed, seedRound, mutationRate, etc.)
195     If Globals.saveState is True:
196         Open and read .savestate file
197         Compare saved parameters with current parameters
198         If any mismatch, return to caller
199     Generate reference directions using Globals.n_dimensions, populationSize, etc.
200     Set mutation and crossover operators with provided rates
201     Increment currentRound by 1
202     If allOfThem is True:
203         For each algorithm in AlgorithmSelection.REF:

```



```
204     If Globals.saveState is True:
205         Open and read .savestate file
206         Check if currentAlgorithm matches saved algorithm
207         If match, disable saveState
208     Else:
209         Call SingleExecution with parameters
210
211 Else:
212     Set currentAlgorithm to selectedAlgorithm
213     Disable Globals.saveState
214     Call SingleExecution with parameters
215     Close thread pool
216 End function
217
218 If script is executed as main program:
219     Call Globals.initialize() to set up global variables
220     Record start time of execution
221     Print "Running..."
222     Set Globals.saveState based on presence of .savestate file
223     Execute Loops.SeedLoop with nested loops (PopulationLoop, MutationRateLoop,
224         CrossoverRateLoop), calling ExecuteAlgorithms
225     Calculate and print total execution time in minutes and seconds
226     If .savestate file exists:
227         Delete .savestate file
228 End main
```