



Universidade Estadual de Feira de Santana  
Programa de Pós-Graduação em Ciência da Computação

Otimização de Redes de Sensores Sem Fio:  
Uma Abordagem Baseada em Algoritmo  
Genético para o Posicionamento  
Multiobjetivo em Áreas Urbanas

Gabriel Sá Barreto Alves

Feira de Santana

2025



Universidade Estadual de Feira de Santana

Programa de Pós-Graduação em Ciência da Computação

Gabriel Sá Barreto Alves

**Otimização de Redes de Sensores Sem Fio: Uma  
Abordagem Baseada em Algoritmo Genético para o  
Posicionamento Multiobjetivo em Áreas Urbanas**

Dissertação apresentada à Universidade  
Estadual de Feira de Santana como parte  
dos requisitos para a obtenção do título de  
Mestre em Ciência da Computação.

Orientador: Prof. Dr. Thiago Cerqueira de Jesus

Coorientador: Prof. Dr. Matheus Giovanni Pires

Feira de Santana

2025

Ficha Catalográfica - Biblioteca Central Julieta Carteado - UEFS

A479o

Alves, Gabriel Sá Barreto

Otimização de redes de sensores sem fio: uma abordagem baseada em algoritmo genético para o posicionamento multiobjetivo em áreas urbanas / Gabriel Sá Barreto Alves. – 2025.

139 f.: il.

Orientador: Thiago Cerqueira de Jesus

Coorientador: Matheus Giovanni Pires

Dissertação (mestrado) – Universidade Estadual de Feira de Santana, Programa de Pós-Graduação em Ciência da Computação, Feira de Santana, 2025.

1.Computação inteligente. 2.Internet das coisas. 3.Algoritmo Genético.  
4. Redes de Sensores Sem Fio. 5. Cidades inteligentes. I. Jesus, Thiago Cerqueira de, orient. II. Pires, Matheus Giovanni, coorient. III. Universidade Estadual de Feira de Santana. IV. Título.

CDU 004.421:711

Gabriel Sá Barreto Alves

# Otimização de Redes de Sensores Sem Fio: Uma Abordagem Baseada em Algoritmo Genético para o Posicionamento Multiobjetivo em Áreas Urbanas

Dissertação apresentada à Universidade Estadual de Feira de Santana como parte dos requisitos para a obtenção do título de Mestre em Ciência da Computação.

Feira de Santana, 01 de Outubro de 2025

## BANCA EXAMINADORA

Documento assinado digitalmente  
 MATHEUS GIOVANNI PIRES  
Data: 20/10/2025 10:45:58-0300  
Verifique em <https://validar.iti.gov.br>

---

Matheus Giovanni Pires (Coorientador(a))  
Universidade Estadual de Feira de Santana

Documento assinado digitalmente  
 JOAO PAULO JUST PEIXOTO  
Data: 02/10/2025 10:02:51-0300  
verifique em <https://validar.iti.gov.br>

---

João Paulo Just Peixoto  
Instituto Federal de Educação, Ciência e Tecnologia da Bahia

Documento assinado digitalmente  
 ANGELO CONRADO LOULA  
Data: 16/10/2025 17:46:26-0300  
Verifique em <https://validar.iti.gov.br>

---

Angelo Conrado Loula  
Universidade Estadual de Feira de Santana

# Abstract

Technological advances have expanded the use of computing in various areas, highlighting Wireless Sensor Networks in the context of the Internet of Things. These networks enable continuous collection of structural and environmental data and rapid responses to critical events. However, the efficient placement of sensors is a central challenge, as inadequate arrangement can affect aspects such as coverage, energy consumption, and data reliability. This positioning problem is regarded as multi-objective due to the necessity of optimizing multiple criteria concurrently. In large-scale networks, combinatorial complexity precludes the application of traditional approaches in the pursuit of optimal solutions. In light of these findings, the present study puts forth and examines a novel positioning strategy for RSSFs comprising scalar and visual sensors. This strategy utilizes the Non-dominated Sorting Genetic Algorithm-II (NSGA-II) to optimize the criteria of coverage, sensing quality, and network connectivity. Among the experiments performed, the following should be emphasized: the definition of NSGA-II hyperparameters using Grid Search, the analysis of positioning in an ideal test scenario, the application of the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) for the selection of solutions on the Pareto Frontiers, and the analysis of the variability of the solutions found in one hundred scenarios with random network distribution. The results demonstrated the efficacy of the selected set of hyperparameters in identifying suitable solutions, meeting the established optimization criteria, and underscoring the efficiency of the proposed positioning process. Furthermore, the application of the TOPSIS method indicated that, for the present study, the selection of solutions can be made considering the same level of relevance for all criteria, since the prioritization did not show significant differences. Finally, the low variability observed in successive executions of NSGA-II demonstrated the stability and reproducibility of the proposed methodology in the search process for solutions. Consequently, this research can assist in the optimization of urban applications and facilitate the planning and adaptation of RSSFs in different outdoor environments. This contributes to the improvement of existing studies and provides guidelines for future research in the area.

**Keywords:** Sensor Positioning, Wireless Sensor Networks, Urban Areas, Genetic Algorithm, Multi-objective Optimization.

# Resumo

O avanço tecnológico ampliou o uso da computação em diversas áreas, destacando as Redes de Sensores Sem Fio no contexto da Internet das Coisas, as quais permitem a coleta contínua de dados estruturais e ambientais e respostas rápidas a eventos críticos. Entretanto, a disposição eficiente dos sensores é um desafio central, pois um arranjo inadequado pode afetar aspectos como cobertura, consumo energético e a confiabilidade dos dados. Esse problema de posicionamento é considerado multiobjetivo devido à necessidade de otimizar múltiplos critérios simultaneamente e, em redes de grande escala, a complexidade combinatória inviabiliza o uso de abordagens tradicionais na busca por soluções ótimas. Diante disso, esta pesquisa propõe e analisa uma nova estratégia de posicionamento para RSSF compostas por sensores escalares e visuais, utilizando o *Non-dominated Sorting Genetic Algorithm-II* (NSGA-II) para otimizar os critérios de cobertura, qualidade de sensoriamento e conectividade em rede. Entre os experimentos realizados, destacam-se a definição dos hiperparâmetros do NSGA-II por meio de *Grid Search*, a análise do posicionamento em um cenário ideal de teste, a aplicação do método *Technique for Order of Preference by Similarity to Ideal Solution* (TOPSIS) para a seleção de soluções nas Fronteiras de Pareto e a análise da variabilidade das soluções encontradas em cem cenários de implantação com distribuição aleatória das redes. Os resultados demonstraram que o conjunto de hiperparâmetros escolhido foi eficaz na busca por soluções adequadas, atendendo satisfatoriamente aos critérios de otimização e evidenciando a eficiência do processo de posicionamento proposto. Ademais, a aplicação do método TOPSIS indicou que, para este estudo, a seleção de soluções pode ser feita considerando o mesmo nível de relevância para todos os critérios, uma vez que a priorização não apresentou diferenças significativas. Por fim, a baixa variabilidade observada nas execuções sucessivas do NSGA-II comprovou que a metodologia proposta apresenta comportamento estável e reproduzível no processo de busca por soluções. Desta forma, essa pesquisa não apenas pode auxiliar na otimização de aplicações urbanas, mas também facilita o planejamento e a adaptação das RSSF em diferentes ambientes externos; nesse contexto, contribuindo para o aprimoramento dos estudos existentes e fornecendo diretrizes para futuras investigações na área.

**Palavras-chave:** Posicionamento de Sensores, Redes de Sensores sem Fio, Áreas Urbanas, Algoritmo Genético, Otimização Multiobjetivo.

# Prefácio

Esta dissertação de mestrado foi submetida à Universidade Estadual de Feira de Santana (UEFS) como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

A dissertação foi desenvolvida no Programa de Pós-Graduação em Ciência da Computação (PGCC), tendo como orientador o Prof. Dr. **Thiago Cerqueira de Jesus**. O Prof. Dr. **Matheus Giovanni Pires** foi coorientador(a) deste trabalho.

Esta pesquisa foi financiada pelo CNPq e CAPES.

# Agradecimentos

Após um pouco mais de dois anos, tenho o privilégio de concluir mais uma fase da minha vida acadêmica. E hoje, mentalmente, posso voltar no tempo e relembrar toda a minha trajetória. No segundo semestre de 2016, dei início à jornada para tornar-me um Engenheiro de Computação. Estudando constantemente, buscando aprimoramento, sempre tentando encontrar o meu caminho nessa profissão pela qual fiquei fascinado. Durante esse tempo, tive a honra de conhecer pessoas, estabelecer amizades e criar lembranças incríveis que ficaram marcadas em minha memória, até que, em 2023, alcancei o sonhado diploma.

Agradeço ao meu Deus, por ter dado-me força, ânimo, sabedoria e o privilégio de viver e vencer esses anos de desafio e dedicação. Agradeço aos meus pais, pelo incentivo constante e por todo o esforço que fizeram para que eu e meus irmãos nos concentrássemos nos estudos. Lembro sempre da fala de meu pai: “Enquanto eu tiver forças para trabalhar, vocês vão estudar!! ”. Posso dizer que minha família sempre foi meu apoio e maior incentivo. Também, não poderia deixar de fazer menção ao Prof. Dr. Anfranserai Morais Dias. Pois, durante o início da minha vida acadêmica, abriu minha mente para a pesquisa científica e guiou-me nesse caminho fantástico de descobertas e novos aprendizados.

Então, ainda em 2023, cheguei ao programa de pós-graduação em Ciência da Computação com grande entusiasmo, na expectativa sobre os novos trabalhos a serem desenvolvidos. Agradeço a todos os professores que puderam contribuir para minha formação como pesquisador ao longo desse ciclo, em especial, ao meu orientador Prof. Dr. Thiago Cerqueira de Jesus. Seu apoio, motivação, conselhos e entusiasmo foram essenciais para que eu vencesse todas as etapas que um curso de Mestrado exige, e assim, alcançar o título de Mestre. Agradeço também ao meu coorientador Prof. Dr. Matheus Giovanni Pires. Sua contribuição durante a pesquisa foi significativa, possibilitando o avanço em momentos que me sentia “travado”.

Finalizo esse ciclo com sentimentos de gratidão, alegria e muita expectativa pelos dias que virão. Obrigado a todos que estiveram comigo em todos esses anos de caminhada. Acredito que novas conquistas surgirão, tendo Deus como meu principal orientador nessa jornada de constantes descobertas chamada vida. Como está escrito em Salmos 32.8: “Diz o Senhor: Instruir-te-ei e te guiarei no caminho a seguir; os meus olhos estarão sobre ti para aconselhar-te”.

# Sumário

<b>Abstract</b>	i
<b>Resumo</b>	ii
<b>Prefácio</b>	iii
<b>Agradecimentos</b>	iv
<b>Sumário</b>	vii
<b>Alinhamento com a Linha de Pesquisa</b>	viii
<b>Produções Bibliográficas, Produções Técnicas e Premiações</b>	ix
<b>Lista de Tabelas</b>	x
<b>Lista de Figuras</b>	xii
<b>Lista de Abreviações</b>	xiii
<b>Lista de Símbolos</b>	xv
<b>1 Introdução</b>	1
1.1 Objetivos . . . . .	4
1.1.1 Objetivo geral . . . . .	4
1.1.2 Objetivos Específicos . . . . .	5
1.2 Relevância . . . . .	5
1.3 Contribuições . . . . .	6
1.4 Organização do Trabalho . . . . .	7
<b>2 Fundamentação e Revisão Bibliográfica</b>	8
2.1 Rede de Sensores sem fio (RSSF) . . . . .	9
2.2 Modelos de Detecção . . . . .	11
2.2.1 Modelo de Detecção Determinístico (binário) . . . . .	11
2.2.2 Modelo de Detecção Probabilístico - <i>Elves Sensing Model</i> . . . . .	12

2.2.3	Modelo de Detecção Probabilístico - <i>Shadow-Fading Sensing Model</i>	13
2.3	Classificação das estratégias de posicionamento de sensores	14
2.3.1	Estratégia de posicionamento determinístico e randômico	15
2.3.2	Tipos de Cobertura em RSSF	16
2.3.3	Posicionamento <i>Indoor</i> e <i>Outdoor</i>	16
2.4	Computação Evolutiva	18
2.4.1	Algoritmos Evolutivos	19
2.4.2	Representação dos indivíduos	20
2.4.3	Função de Avaliação	22
2.4.4	Operador de variação - Mutação	22
2.4.5	Operador de variação - Cruzamento	23
2.4.6	Mecanismos de Seleção	24
2.5	Algoritmo Genético	26
2.5.1	Mono Objetivo	26
2.5.2	Multiobjetivo	27
2.6	Seleção de indivíduos nas Fronteiras de Pareto	29
2.6.1	Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)	30
2.6.2	Gray Relational Analysis (GRA)	31
2.6.3	Simple Additive Weighting (SAW)	32
2.7	Indicadores de Qualidade	33
2.7.1	Hypervolume (HV)	33
2.7.2	Two Set Coverage (SC)	34
2.8	Trabalhos relacionados	35
2.8.1	Posicionamento de sensores escalares	35
2.8.2	Posicionamento de sensores visuais (câmeras)	39
2.8.3	Considerações	43
<b>3</b>	<b>Solução Proposta</b>	<b>45</b>
3.1	Modelagem da Região de Interesse (RdI)	45
3.1.1	Processo de mapeamento	46
3.1.2	Representação das regiões de interesse em formato JSON	50
3.2	Representação dos dispositivos de monitoramento	52
3.2.1	Modelo de sensoriamento	52
3.2.2	Região de cobertura	54
3.2.3	Máscaras de cobertura	56
3.2.4	Geração das Máscaras de Cobertura para um Sensor Escalar	58
3.2.5	Geração das Máscaras de Cobertura para as Câmeras	60
3.3	Funções de <i>Fitness</i>	66
3.3.1	Cobertura	66
3.3.2	Dispositivos em regiões de alta conectividade	67
3.3.3	Qualidade de Sensoriamento	68
3.4	Posicionamento através do Algoritmo Genético NSGA-II	69

3.4.1	Codificação: Cromossomo . . . . .	70
3.4.2	Demonstração do método de posicionamento por meio das soluções geradas pelo NSGA-II . . . . .	72
<b>4</b>	<b>Resultados</b>	<b>76</b>
4.1	Aplicação da estratégia Grid Search . . . . .	76
4.2	Análise das soluções geradas pela estratégia <i>Grid Search</i> para a versão V0. . . . .	82
4.3	Análise das soluções geradas pela estratégia <i>Grid Search</i> para a versão V1. . . . .	90
4.4	Avaliação final do método <i>Grid Search</i> . . . . .	94
4.5	Avaliação de cenários aleatórios de teste . . . . .	97
4.5.1	Avaliação da metodologia de escolha de soluções das Fronteiras de Pareto . . . . .	99
4.5.2	Análise de variabilidade das soluções geradas pelo Algoritmo Genético . . . . .	106
<b>5</b>	<b>Conclusões</b>	<b>110</b>
<b>Referências</b>		<b>113</b>

# **Alinhamento com a Linha de Pesquisa**

## **Linha de Pesquisa: Computação Inteligente**

O presente trabalho está inserido na linha de pesquisa Computação Inteligente, e busca contribuir para o campo de pesquisa em Otimização de Redes de Sensores sem Fio (RSSF) por meio de estratégias de posicionamento multiobjetivo.

O objetivo principal desta pesquisa é o desenvolvimento e análise de uma abordagem baseada em Algoritmos Genéticos para a otimização do posicionamento em RSSF. A complexidade combinatória envolvida nesse problema exige o uso de técnicas computacionais avançadas para a busca de soluções eficientes, visto que abordagens tradicionais tornam-se inviáveis em cenários de grande escala. Com isso, a abordagem proposta se alinha aos princípios da Computação Inteligente ao explorar essa busca por meio da Computação Evolutiva, empregando o Non-dominated Sorting Genetic Algorithm-II (NSGA-II), um método evolucionário amplamente utilizado para problemas multiobjetivo, visando aprimorar características consideradas importantes para a eficiência e eficácia das redes de sensores.

Os resultados desta pesquisa podem contribuir significativamente para diversas aplicações práticas, especialmente em Cidades Inteligentes, onde a disposição eficiente de sensores é essencial para o monitoramento de áreas públicas, segurança e gestão de infraestrutura. Além do ambiente urbano, a metodologia desenvolvida pode ser aplicada a outros exemplos de áreas abertas, como regiões agrícolas e reservas ambientais, onde a coleta de dados confiável e eficiente é fundamental para a tomada de decisão e a otimização de recursos. Dessa forma, este estudo não apenas aprimora o estado da arte em técnicas de posicionamento de sensores, mas também fornece uma base metodológica sólida para futuras investigações e aplicações em diferentes domínios.

# Produções Bibliográficas, Produções Técnicas e Premiações

1. BARRETO, GABRIEL S.; JESUS, THIAGO C.; COSTA, DANIEL G.; CATALÃO, JOÃO P. S. Quality-Aware Sensors Positioning in Smart Cities: Enhancing Coverage in IoT-Driven Urban Scenarios, In: *51st IEEE Industrial Electronics Society Conference (IECON2025)*, Madrid, 2025, pp. 1–6.
2. BARRETO, Gabriel S.; PIRES, Matheus G.; JESUS, Thiago C. Posicionamento de sensores para monitoramento colaborativo de emergências urbanas. In: SIMPÓSIO BRASILEIRO DE SISTEMAS COLABORATIVOS (SBSC), 2025, Manaus/AM. p. 52-63. ISSN 2326-2842. DOI: 10.5753/sbsc.2025.8438.
3. BARRETO, GABRIEL S.; OLIVEIRA, P. M.; MARTINS, P. H. A.; JESUS, T. C. Detectabilidade de emergências urbanas usando redes hierárquicas em cidades inteligentes. In: XXV Congresso Brasileiro de Automática (CBA2024), 2024, Rio de Janeiro. Anais do XXV Congresso Brasileiro de Automática (CBA2024), 2024.
4. BARRETO, GABRIEL S.; JESUS, THIAGO C.; COELHO, GUSTAVO A. A.; SILVA, GUSTAVO F.; COSTA, DANIEL G. Plataforma de Crowdsourcing Internet of Things (Crowd-IoT) para detecção de emergências urbanas. In: SIMPÓSIO BRASILEIRO DE SISTEMAS COLABORATIVOS (SBSC), 2024, Salvador/BA. p. 33-38. DOI: 10.5753/sbsc\_estendido.2024.238538.

# Listas de Tabelas

2.1	Classificação de trabalhos de acordo aos modelos de detecção e às categorias associadas ao processo posicionamento. . . . .	18
2.2	Levantamento de trabalhos sobre posicionamento de sensores escalares.	38
2.3	Outras pesquisas sobre posicionamento de câmeras. . . . .	41
3.1	Parâmetros individuais de rede para o cálculo da métrica <i>DPConn</i> . .	49
3.2	Parâmetros gerais de rede para o cálculo da métrica <i>DPConn</i> . . . .	49
3.3	Raio de alcance das redes Wifi, 4G e LoRa. . . . .	50
4.1	Conjunto de valores para os hiperparâmetros. . . . .	77
4.2	Combinações de valores para os hiperparâmetros. . . . .	78
4.3	Versões do NSGA-II em relação aos operadores genéticos. . . . .	78
4.4	Dados paramétricos da RdI ideal apresentada na Figura 4.1. . . . .	79
4.5	Parâmetros para os sensores escalares utilizados no processo de <i>Grid Search</i> . . . . .	80
4.6	Parâmetros para as câmeras utilizadas no processo de <i>Grid Search</i> . .	80
4.7	Valor médio da métrica <i>Time-C</i> para as combinações <i>P4</i> , <i>P5</i> , <i>P6</i> , <i>P11</i> , <i>P12</i> , <i>P18</i> , <i>P23</i> , <i>P24</i> e <i>P30</i> separadas por grupos de acordo ao valor de <i>NG</i> . . . . .	84
4.8	Métrica <i>SC</i> para o agrupamento par a par das combinações <i>P4</i> , <i>P6</i> e <i>P11</i> . . . . .	88
4.9	Versão <i>V0</i> - métricas de performance calculadas a partir das FP. . .	89
4.10	Versão <i>V1</i> - métricas de performance calculadas a partir das FP. . .	93
4.11	Configuração dos cenários com distribuição aleatória de redes. . . .	98
4.12	Configuração dos sensores para o processo de avaliação dos cenários de implantação com distribuição aleatória das redes sem fio. . . .	99
4.13	Níveis de Relevância para escolha de soluções nas Fronteira de Pareto.	101
4.14	Resultados obtidos para os cenários <i>Cfg<sub>1</sub></i> – <i>C14</i> e <i>Cfg<sub>1</sub></i> – <i>C19</i> . . .	103
4.15	Resultados obtidos para os cenários <i>Cfg<sub>1</sub></i> – <i>C31</i> e <i>Cfg<sub>2</sub></i> – <i>C7</i> . . . .	104
4.16	Resultados obtidos para os cenários <i>Cfg<sub>2</sub></i> – <i>C32</i> e <i>Cfg<sub>2</sub></i> – <i>C47</i> . . .	105

# Listas de Figuras

2.1	Modelo Binário de Detecção. . . . .	12
2.2	Representação do <i>Elfes Sensing Model</i> . . . . .	13
2.3	Configuração da curva de decaimento no <i>Elfes Sensing Model</i> com diferentes valores de $\beta$ e $\gamma$ . . . . .	14
2.4	Classificação das estratégias usadas nas pesquisas em posicionamento de sensores. . . . .	15
2.5	Tipos de Cobertura aplicados através das RSSF. . . . .	17
2.6	Comportamento de um AG mono objetivo no aprimoramento dos indivíduos ao longo das gerações. . . . .	27
3.1	Representação da Região de Interesse. . . . .	47
3.2	Mapeamento das redes e níveis de conectividade para uma Região de Interesse. . . . .	49
3.3	Mapeamento dos ABs com conectividade. . . . .	51
3.4	Representação do modelo de sensoriamento. . . . .	54
3.5	Representação da região de cobertura dos sensores escalares e câmeras baseados no modelo de sensoriamento proposto. . . . .	55
3.6	Representação das Máscaras de Sensoriamento. . . . .	57
3.7	Representação das Máscaras de Qualidade de Sensoriamento. . . . .	57
3.8	Ilustração do método de geração da $MS$ para um sensor escalar. . . . .	59
3.9	Exemplo das máscaras de cobertura geradas a partir dos Algoritmos 2 e 3 para um sensor escalar com $r_s = 30$ . . . . .	60
3.10	Ilustração do método de geração da $MS$ e da $MQS$ para uma câmera. . . . .	61
3.11	Exemplo das máscaras de cobertura geradas a partir do Algoritmo 4 para uma câmera com $r_s = 100$ , $\theta = 135^\circ$ , $\alpha = 45^\circ$ . . . . .	62
3.12	Influência do parâmetro $\alpha$ no comportamento da função de <i>fitness</i> $f_2$ . . . . .	68
3.13	Estrutura do Cromossomo. . . . .	71
3.14	Ilustração do processo de posicionamento. . . . .	73
3.15	Resultado da aplicação das $MQS$ sobre o cenário de exemplo. . . . .	74
4.1	Representação da RdI gerada para aplicação da estratégia <i>Grid Search</i> . . . . .	79
4.2	Fluxo de execução do método <i>Grid Search</i> sobre o processo de posicionamento. . . . .	80
4.3	Fronteira de Pareto produzida pela combinação $P12$ na versão $V0$ . . . . .	83

4.4	Fronteira de Pareto produzida pela combinação $P4$ na versão $V0$ . . . . .	85
4.5	Fronteira de Pareto produzida pela combinação $P6$ na versão $V0$ . . . . .	86
4.6	Fronteira de Pareto produzida pela combinação $P11$ na versão $V0$ . . . . .	87
4.7	Fronteira de Pareto produzida pela combinação $P18$ na versão $V1$ . . . . .	91
4.8	Fronteira de Pareto produzida pela combinação $P17$ na versão $V1$ . . . . .	92
4.9	Cenário Ideal ( <i>Grid Search</i> ): disposição dos dispositivos ao longo da RdI. . . . .	95
4.10	Cenário Ideal ( <i>Grid Search</i> ): Taxa de cobertura. . . . .	96
4.11	Cenário Ideal ( <i>Grid Search</i> ): Níveis de sobreposição das áreas monitoradas. . . . .	96
4.12	Cenário Ideal ( <i>Grid Search</i> ): Níveis de qualidade de sensoriamento das áreas monitoradas. . . . .	97
4.13	Cenários com distribuição aleatória das redes sem fio. . . . .	100
4.14	<i>Boxplot</i> : Análise dos dados de posicionamento para os cenários da configuração 1. . . . .	107
4.15	<i>Boxplot</i> : Análise dos dados de posicionamento para os cenários da configuração 2. . . . .	108

# **Lista de Abreviações**

<b>Abreviação</b>	<b>Descrição</b>
AB	Blocos de Área
AE	Algoritmos Evolutivos
AG	Algoritmo Genético
CE	Computação Evolutiva
CFPA	<i>Chaotic Flower Pollination Algorithm</i>
CPMEA	<i>Constrained Pareto-based Multi-objective Evolutionary Approach</i>
DEP	Desvio Padrão
DPConn	<i>Dependable-Quality Connectivity</i>
ED	Evolução Diferencial
ESA	<i>Exhaustive Search Algorithm</i>
FL	<i>Fuzzy Logic</i>
FP	Fronteira de Pareto
HV	<i>Hypervolume</i>
ICS	<i>Improved Cuckoo Search</i>
IoT	<i>Internet of Things</i>
IQ	Indicadores de Qualidade
JSON	<i>JavaScript Object Notation</i>
MDS	Média das Diferenças Sucessivas
MOGA	<i>Multi-objective Genetic Algorithm</i>
MOACO	<i>Multi-objective Ant Colony Optimization</i>
MOEA	<i>Multi-objective Evolutionary Algorithm</i>
MQS	Máscara de Qualidade de Sensoriamento
MS	Máscara de Sensoriamento
NC	Estrutura de Níveis de Conectividade
NG	Hiperparâmetro do NSGA-II: Número de gerações
NSGA-II	<i>Non-dominated Sorting Genetic Algorithm-II</i>
NSND	Número de soluções não-dominadas
PC	Estrutura de Posições de Conectividade
PG	Programação Genética
PSO	<i>Particle Swarm Optimization</i>
RdI	Região de Interesse
RSSF	Rede de Sensores sem Fio

SBX	<i>Simulated Binary Crossover</i>
SC	<i>Set Coverage</i>
SOGA	<i>Single Objective Genetic Algorithm</i>
SPEA-II	<i>Strength Pareto Evolutionary Algorithm-II</i>
TC	Hiperparâmetro do NSGA-II: Taxa de cruzamento
TM	Hiperparâmetro do NSGA-II: Taxa de mutação
Time-C	Tempo computacional
TP	Hiperparâmetro do NSGA-II: Tamanho da População
WSMO	<i>Weighted-Sum Multi Objective</i>

# List of Symbols

<b>Símbolo</b>	<b>Descrição</b>
$AB_{invalid}$	Quantidade de blocos de área monitorados fora da região de interesse
$AB_{sensed}$	Quantidade de blocos de área monitorados por ao menos um sensor
$AB_{total}$	Quantidade de blocos de área dentro de uma RdI
$AB_{wHS}$	Quantidade de blocos de área com alta qualidade de sensoriamento
$AB_{wLS}$	Quantidade de blocos de área com baixa qualidade de sensoriamento
$AB_{wMS}$	Quantidade de blocos de área com média qualidade de sensoriamento
$\vec{AB}$	Vetor dos pontos A e B para delimitação da região de cobertura de uma câmera
$\vec{CA}$	Vetor dos pontos C e A para delimitação da região de cobertura de uma câmera
$f_1$	Função de <i>fitness</i> para avaliar a Cobertura de uma RdI
$f_2$	Função de <i>fitness</i> para avaliar Dispositivos em Alta Conectividade
$f_3$	Função de <i>fitness</i> para avaliar Qualidade de Sensoriamento
$hs_{AB}$	Altura de um bloco de área
$hs_{city}$	Altura da área pertencente à RdI
$nb_C$	Número de blocos de área com conectividade
$Q_S$	Nível da Região de Qualidade de sensoriamento
$r_e$	Limiar da região com média qualidade de sensoriamento
$r_l$	Limiar da região com alta qualidade de sensoriamento
$r_s$	Raio de Sensoriamento de um dispositivo
$S_i$	I-ésima solução gerada por um algoritmo genético
$Sr_{inH}$	Quantidade de dispositivos em regiões de alta conectividade
$Sr_{inL}$	Quantidade de dispositivos em regiões de baixa conectividade
$Sr_{inM}$	Quantidade de dispositivos em regiões de média conectividade
$ws_{AB}$	Largura de um bloco de área
$ws_{city}$	Largura da área pertencente à RdI
$\alpha$	Ângulo (em graus) de orientação de uma câmera
$\theta$	Ângulo (em graus) de abertura do campo de visão de uma câmera

# Capítulo 1

## Introdução

*“In creating, the only hard thing is to begin: a grass blade’s no easier to make than an oak.”*

– James Russell Lowell

O crescimento tecnológico e industrial ao longo das últimas décadas tem transformado profundamente a forma como a sociedade opera, impulsionando o uso da computação em diversos aspectos da vida humana. Desde a Revolução Industrial, a automação tem sido um fator essencial para o aumento da produtividade e da eficiência em processos industriais, evoluindo ao longo dos anos de máquinas mecânicas para sistemas altamente informatizados.

A partir do início da Indústria 4.0, houve uma maior transição em direção à automação e à dependência de inovações e estratégias baseadas em dados. Então, o avanço da microeletrônica e da computação embarcada permitiu a criação de dispositivos inteligentes capazes de processar grandes volumes de dados em tempo real, tornando possível a automação de tarefas antes restritas à intervenção humana. Isso possibilitou maior eficiência operacional, redução de custos e melhoria na qualidade dos produtos desenvolvidos (Rosário e Dias, 2023).

Entretanto, esses avanços não ficaram concentrados apenas no setor industrial; em âmbito geral, a computação aliada à automação tem desempenhado um papel central em áreas como a saúde, educação e a infraestrutura urbana, possibilitando a criação de sistemas interconectados que facilitam o monitoramento e o controle remoto de dispositivos e processos. Com isso, a computação não apenas aprimorou a produtividade industrial, mas também contribuiu significativamente para o avanço da qualidade de vida da população.

A partir desses avanços nas últimas décadas, um novo paradigma de automação tornou-se cada vez mais difundido e tem sido amplamente estudado por pesqui-

sadores ao redor do mundo. Seu objetivo é mudar a forma como monitoramos e controlamos objetos e processos que fazem parte da nossa vida diária. Esse novo paradigma, denominado como Internet das Coisas, ou como é mais conhecido, *IoT - Internet of Things*, tem revolucionado a maneira como as pessoas interagem com diversos ambientes, seja residencial, industrial ou qualquer outro, além de permitir a automação, conexão e troca de informações entre os mais variados dispositivos físicos (Zanella et al., 2014). A IoT une diferentes áreas do conhecimento, tais como programação, redes de computadores, eletrônica e, atualmente, a Inteligência Artificial, proporcionando resultados significativos na resolução de problemas dentro da agricultura de precisão, setores da indústria, na medicina e em ambientes urbanos como as Cidades Inteligentes.

Uma das aplicações mais significativas e abrangentes da IoT é a implementação das Redes de Sensores Sem Fio (RSSF), que consiste em uma rede interligada de dispositivos de monitoramento distribuídos ao longo de uma região específica, capazes de monitorar variáveis ambientais e transmitir os dados coletados por meio de protocolos de comunicação para sistemas de controle externos (Akyildiz et al., 2002). Sua utilização em diferentes estudos e aplicações possibilita a coleta contínua e em tempo real de dados ambientais e estruturais, permitindo respostas rápidas a eventos críticos. Além disso, sua natureza distribuída e sem fio reduz a necessidade de infraestrutura fixa, tornando sua implementação mais flexível e econômica.

Apesar das inúmeras vantagens e aplicações das RSSF, o posicionamento adequado de seus elementos consiste em um aspecto básico, mas que impacta diretamente sua eficácia e eficiência. Um arranjo ineficiente pode comprometer a cobertura dos sensores, aumentar o consumo energético dos dispositivos e reduzir a confiabilidade dos dados coletados (Priyadarshi et al., 2020). Dependendo da estrutura geográfica e da facilidade de acesso associada ao local em que a rede irá operar, diferentes estratégias e abordagens podem ser utilizadas, por exemplo, o posicionamento determinístico ou aleatório (Sharma et al., 2016). O processo de busca por soluções para esse desafio torna-se ainda mais complexo em ambientes dinâmicos, como os cenários urbanos, nos quais obstáculos físicos, ruídos e outras interferências afetam simultaneamente o desempenho das aplicações.

Efetuar o posicionamento adequado para os elementos de uma RSSF consiste basicamente em determinar o local de implantação de cada dispositivo, de forma a aprimorar alguns critérios específicos que determinarão a eficiência da aplicação (Amutha et al., 2020). Entre esses critérios, temos cobertura, consumo energético, conexão entre *nodes*, número total de dispositivos na rede, tolerância a falhas, segurança de rede, dentre outros. Entretanto, devido à variedade de critérios a serem aprimorados, aumenta-se consideravelmente a complexidade e o custo computacional para se encontrar soluções ótimas em um tempo viável através dos métodos tradicionais de modelagem de algoritmos (Fei et al., 2017).

Por isso, pesquisadores em todo o mundo estão alocando esforços para desenvolver pesquisas que proporcionem estratégias de aprimoramento para as diferentes neces-

sidades de uma RSSF. Em todos os estudos realizados, busca-se encontrar soluções que apresentem um equilíbrio entre os diferentes critérios de otimização que, por natureza, são conflitantes. Isso significa que, ao aprimorar um deles, outro sofrerá perda. Logo, o objetivo é maximizar ou minimizar esses critérios tanto quanto for possível até que seja alcançado o nível de balanceamento desejado.

De uma forma geral, encontra-se na literatura uma predominância do uso de abordagens heurísticas/meta-heurísticas nesse tipo de problematização, pois são capazes de obter soluções quase ótimas para questões particularmente difíceis de lidar usando técnicas clássicas de algoritmos (Fei et al., 2017). Nesse sentido, percebe-se que há quase duas décadas, estudos estão sendo publicados sobre esse tópico. Os trabalhos propostos por Jourdan e de Weck (2004), Molina et al. (2008), Navin et al. (2010), Yetgin et al. (2015), Aghdasi e Abbaspour (2016), Domingo-Perez et al. (2016), Hanh et al. (2019), Wu et al. (2020) e Birtane et al. (2024), são apenas alguns exemplos de uma grande variedade de pesquisas realizadas ao longo dos anos.

Após a realização de um levantamento geral de estudos semelhantes, notou-se que diferentes estratégias de posicionamento são propostas para aplicações de monitoramento tanto em áreas internas (prédios, casas, indústrias, etc.) quanto em áreas externas, como em regiões urbanas. Cada um propondo soluções com base no uso e na análise de diferentes algoritmos, não sendo possível estabelecer um padrão metodológico para a resolução do problema, mas características semelhantes de modelagem foram encontradas, como modelos de sensoriamento para os dispositivos da rede e representação das áreas de interesse.

Embora exista essa variedade de pesquisas associadas ao posicionamento de RSSF, nota-se que grande parte preocupa-se com os mesmos critérios de otimização, sendo os principais, cobertura, consumo energético, custo e tempo de vida da rede. Logo, encontra-se na literatura uma saturação de estudos relacionados a esses critérios.

Outra observação é a separação entre estudos de posicionamento envolvendo RSSF com sensores tradicionais e visuais. Essa separação pode levar a abordagens fragmentadas que não consideram as particularidades e desafios da integração entre esses dispositivos em uma única rede, resultando em soluções que não são generalizáveis para diferentes cenários de monitoramento nos quais são consideradas as redes heterogêneas e dinâmicas.

Além disso, não é visto nos estudos prévios levantados nesta pesquisa o uso da distribuição e organização das redes de comunicação (Wifi, LoRa, 4G, dentre outras) como uma característica importante durante o processo de otimização do posicionamento das RSSF. Essas redes podem afetar a eficiência dos processos de comunicação e transferência de dados realizados pela rede de sensores; logo, garantir a alocação dos dispositivos em regiões que apresentam os níveis necessários de conectividade contribui significativamente para a segurança, integridade e confiabilidade desses processos.

Diante disso, e com o intuito de contribuir para o preenchimento dessas lacunas

identificadas na literatura, esta pesquisa propõe a modelagem e a análise de uma nova estratégia de posicionamento de sensores aplicada a áreas abertas, com foco principal em ambientes urbanos. O objetivo consiste em, a partir do mapeamento da região de monitoramento na qual as redes de comunicação estão distribuídas e por meio de uma abordagem que viabiliza o uso de uma RSSF heterogênea, composta por sensores escalares e visuais, determinar, utilizando um algoritmo genético, o local de implantação dos elementos da rede. Essa determinação considera não apenas a maximização da área monitorada, mas também critérios até então não explorados, consistindo na Qualidade do Sensoriamento realizado por dispositivos pré-configurados e na Qualidade da Conectividade em Rede associada a cada sensor da RSSF.

Dentre os resultados alcançados está a definição dos hiperparâmetros via *Grid Search* para o Algoritmo Genético *Non-dominated Sorting Genetic Algorithm-II* (NSGA-II) utilizado como base para o processo de busca de soluções; a análise do posicionamento efetuado sobre um cenário ideal de teste, no qual constatou-se taxas satisfatórias associadas aos critérios de otimização estabelecidos; a utilização do método *Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)* para seleção de soluções nas Fronteiras de Pareto; além da análise de baixa variabilidade para as soluções de um conjunto de 100 cenários de implantação com distribuição aleatória das redes sem fio.

Portanto, a partir desta proposta, diferentemente dos trabalhos existentes na literatura, abre-se outra perspectiva na análise de posicionamento que pode auxiliar diferentes aplicações no aperfeiçoamento do processo de detecção de eventos como, por exemplo, pesquisas direcionadas ao monitoramento de emergências em regiões urbanas (Costa et al., 2022) e análise de ambiente como qualidade do ar (Dhingra et al., 2019). Desta forma, fornecendo soluções que garantem respostas ágeis e eficazes diante da realização de um posicionamento otimizado da rede de sensores.

## 1.1 Objetivos

### 1.1.1 Objetivo geral

O principal objetivo desta pesquisa é, através do uso do Algoritmo Genético NSGA-II, propor uma estratégia de posicionamento que visa auxiliar na determinação dos locais de implantação dos elementos de uma Rede de Sensores sem Fio em áreas urbanas, considerando ambientes bidimensionais, de forma a fornecer uma configuração de sensores que apresente um equilíbrio entre a maximização da Taxa de Cobertura, Qualidade de Sensoriamento e Conectividade para os processos de comunicação e transmissão de dados. Essa estratégia fornece uma modelagem de posicionamento que permitirá o uso de redes heterogêneas compostas por sensores escalares e visuais, mas também considerando a disposição das redes de comunicação alocadas ao longo da região de monitoramento.

### 1.1.2 Objetivos Específicos

- Definir um modelo de sensoriamento que possa ser aplicado tanto em sensores escalares quanto visuais, de forma a representar sua região de cobertura;
- Com base em trabalhos na literatura, definir a forma de representação para as regiões de interesse onde as RSSF serão implantadas;
- Definir a formulação matemática para os critérios de otimização estabelecidos;
- Definir a representação cromossômica para as soluções geradas e processadas pelo Algoritmo Genético;
- Definir os hiperparâmetros do Algoritmo Genético por meio do processo de *Grid Search*;
- Avaliar a eficácia do método de posicionamento proposto através da análise sobre um cenário de monitoramento com configuração ideal de rede;
- Avaliar a variabilidade das soluções produzidas pelo NSGA-II para um conjunto vasto e diversificado de cenários de monitoramento;

## 1.2 Relevância

A crescente evolução das tecnologias em IoT tem impulsionado o desenvolvimento de ambientes cada vez mais interligados e inteligentes. Com relação às áreas urbanas, temos as Cidades Inteligentes, onde redes interconectadas de sensores desempenham um papel fundamental no monitoramento e na gestão de seus recursos (Harrison et al., 2010). As diferentes aplicações implantadas nessas áreas dependem diretamente da coleta eficiente de dados em tempo real, tornando as RSSF componentes essenciais de sua infraestrutura.

Por outro lado, além do ambiente urbano, as RSSF são amplamente aplicáveis em outros cenários, abrangendo desde o monitoramento ambiental em locais remotos ou de difícil acesso até a segurança de fronteiras (Fei et al., 2017). Por essas razões, pesquisas referentes à otimização de critérios associados ao posicionamento de RSSF têm sido amplamente desenvolvidas há mais de duas décadas, tendo em vista a sua importância no aprimoramento da eficácia e eficiência das redes de sensores em inúmeras aplicações.

Existe um grande número de trabalhos nesse campo, pois o posicionamento estratégico de sensores ainda representa um desafio crítico, influenciando diversos aspectos das RSSF (Priyadarshi et al., 2020). Devido à complexidade combinatória desse problema, abordagens tradicionais tornam-se praticamente inviáveis para redes de grande escala. Para superar esse desafio, técnicas avançadas, como as abordagens da Computação Evolutiva, têm-se mostrado promissoras na busca por soluções otimizadas. Desta forma, encontrar métodos e soluções otimizadas para esse problema

é essencial para maximizar o desempenho das aplicações e garantir que funcionem de forma confiável e sustentável.

Por essas razões, esta pesquisa busca contribuir significativamente para esse campo de estudo, propondo uma abordagem eficaz para o posicionamento de sensores em diferentes cenários. Além de otimizar aplicações urbanas, a metodologia desenvolvida facilita o planejamento e a adaptação das RSSF em diversos ambientes externos, ampliando suas possibilidades de aplicação. Assim, os resultados obtidos poderão fortalecer o avanço das pesquisas existentes e fornecer diretrizes para futuras investigações na área.

### 1.3 Contribuições

As principais contribuições deste trabalho são:

- Proposta de uma estratégia que viabiliza o posicionamento de RSSF heterogêneas compostas por sensores escalares e câmeras, aumentando a eficácia e eficiência das aplicações que dispõem dessa tecnologia;
- Proposta de uma estratégia de posicionamento que leva em consideração dois novos aspectos não antes encontrados na literatura, que são: Qualidade de Sensoriamento e Qualidade da Conectividade em Rede;
- Proposta de um processo de mapeamento para localização e identificação das áreas com conectividade presentes nas regiões de monitoramento;
- Construção de uma modelagem que permite simular a região de cobertura de cada elemento da RSSF de acordo às características da região de monitoramento e dos respectivos parâmetros pré-configurados em cada sensor acoplado à rede;
- Construção de uma representação cromossômica para os indivíduos gerados pelo Algoritmo Genético viabilizando a otimização do posicionamento de RSSF heterogêneas e a limitação do espaço de busca, de forma a contribuir para o aumento da eficiência do processo de busca de soluções através da delimitação das regiões com conectividade;
- Proposta de uma representação matemática para avaliação dos critérios de Qualidade de Sensoriamento e Qualidade de Conectividade em Rede;
- Proposta de uma solução que pode servir de auxiliador na implantação de diferentes rede de sensores aplicadas para o monitoramento em cenários urbanos, rurais ou qualquer outro que se caracterize como áreas externas ou de grande escala;

## 1.4 Organização do Trabalho

No Capítulo 2 é apresentado todo o referencial teórico e bibliográfico que serviu de base para o desenvolvimento desta pesquisa, assim como o detalhamento de um conjunto de trabalhos da literatura associados à problematização e criação de soluções para o posicionamento de RSSF em diferentes contextos. No Capítulo 3 é realizada a descrição de cada elemento integrante do processo de posicionamento proposto, bem como a descrição desse processo. No Capítulo 4 são discutidos os testes e procedimentos realizados a fim de validar a metodologia proposta, assim como os resultados alcançados ao longo do desenvolvimento desta pesquisa. E, por fim, no Capítulo 5 é apresentado um breve relato acerca do que foi desenvolvido, bem como a definição de etapas futuras.

## Capítulo 2

# Fundamentação e Revisão Bibliográfica

*“Knowing how things work is the basis for appreciation, and is thus a source of civilized delight.”*

– William Safire

A compreensão dos conceitos e técnicas que norteiam as pesquisas relacionadas ao posicionamento das Redes de Sensores sem Fio (RSSF) constitui um passo fundamental para o desenvolvimento de soluções eficazes nesse domínio. Pois, como dito anteriormente, a disposição espacial dos seus elementos impacta diretamente aspectos como cobertura, consumo energético, conectividade e, consequentemente, a vida útil do sistema (Priyadarshi et al., 2020).

Nesse sentido, a análise rudimentar dos tópicos que orientam as pesquisas nessa área e também o aprofundamento em conteúdos mais complexos permitirá estabelecer critérios mais eficientes e eficazes para a criação de novas aplicações, bem como maior confiabilidade para as novas pesquisas desenvolvidas. Além disso, essa revisão fornece o embasamento teórico necessário para compreender os desafios característicos do problema, como sua natureza combinatória e a elevada complexidade associada à busca de soluções ótimas.

Nesse processo de busca, a utilização de algoritmos evolutivos, em especial os algoritmos genéticos, destaca-se como uma abordagem promissora para enfrentar o problema do posicionamento de sensores. Esses métodos permitem explorar o espaço de soluções de forma adaptativa, permitindo a conciliação de múltiplos objetivos; desta forma, alcançando progressivamente soluções mais ajustadas às restrições e condições do problema em estudo (Eiben e Smith, 2015).

Nessa perspectiva, este capítulo tem por objetivo apresentar o referencial teórico e bibliográfico que fundamenta o desenvolvimento desta pesquisa, bem como o levantamento de trabalhos da literatura que discutem a problematização e a proposição

de soluções para o posicionamento das RSSF. Entre os tópicos a serem abordados, destacam-se: o uso das RSSF na coleta de dados em diferentes contextos de aplicação; a definição dos modelos de detecção empregados nos dispositivos de monitoramento; e as estratégias de posicionamento mais frequentemente exploradas em estudos científicos. Além disso, será realizado o detalhamento dos principais conceitos relacionados à Computação Evolutiva, com ênfase nos Algoritmos Genéticos, dado o seu papel de destaque como técnica de otimização para problemas complexos de múltiplos objetivos.

## 2.1 Rede de Sensores sem fio (RSSF)

Uma RSSF consiste na interligação de um conjunto de dispositivos de monitoramento com características iguais ou distintas (homogêneos ou heterogêneos) chamados de “nodes”, onde cada um possui a capacidade de processamento, armazenamento de dados, memória, comunicação sem fio e acoplamento com um ou mais sensores (Akyildiz et al., 2002). Esses dispositivos são posicionados de forma distribuída ao longo de uma região específica a partir da topologia (*Star, Mesh, Tree* ou *Hierárquica*, dentre outras) que melhor se encaixa na aplicação de interesse.

As RSSF podem ser compostas por dezenas a centenas de *nodes*, sendo o seu principal objetivo o de coletar, processar e transmitir dados associados ao ambiente, como temperatura, pressão e movimento. Devido às suas características de flexibilidade e escalabilidade, aliado aos avanços nas tecnologias sem fio, as RSSF são utilizadas em diferentes aplicações direcionadas ao monitoramento ambiental, urbano, industrial, militar, dentre outros (Kuorilehto et al., 2005). Por exemplo, em aplicações de monitoramento ambiental, as RSSF são empregadas para coletar dados sobre condições climáticas, qualidade do ar e do solo, assim facilitando a detecção de diversos eventos como incêndios e enchentes (Mainwaring et al., 2002).

No contexto militar, as RSSF podem ser aplicadas no monitoramento de tropas e equipamentos, na detecção de intrusos em áreas restritas, ou mesmo no reconhecimento de terrenos hostis em um campo de batalha. Como essas redes são implantadas com um grande número de dispositivos, a perda ou destruição de alguns *nodes* não traz danos significativos para as operações e aplicações militares; além disso, a capacidade de comunicação entre si a longas e médias distâncias favorece o monitoramento remoto, um aspecto de suma importância no contexto militar (Akyildiz et al., 2002).

Nas cidades, as RSSF têm sido implementadas para monitorar diversos aspectos dos cenários urbanos, como qualidade do ar, gestão de tráfego e consumo de energia. No trabalho realizado em Filho et al. (2014), pesquisadores propuseram o *NodePM*, um sistema de alerta de monitoramento remoto para consumo de energia elétrica, utilizando RSSF para coletar dados em tempo real e auxiliar na gestão eficiente de recursos energéticos. Aplicações voltadas para monitoramento residencial também são realizadas, como em Saeed et al. (2018), que propõe um sistema de detecção

e alerta de incêndios residenciais por meio da utilização de uma rede de múltiplos sensores sem fio integrados com um sistema para comunicações móveis, visando minimizar falsos alarmes.

Outra possível abordagem está associada aos níveis de poluição do ar em um cenário urbano, uma vez que esses níveis podem afetar a qualidade de vida e a saúde dos cidadãos. Em Dhingra et al. (2019), é proposto um sistema que utiliza sensores de gás (como  $CO$ ,  $CO_2$  e metano) para detectar o nível de poluição do ar em diferentes regiões de uma cidade e determinar um índice de qualidade a partir da concentração dos diferentes gases presentes na região. Após a análise, os resultados podem ser acessados por meio de uma aplicação *Android*. Através desse tipo de projeto, as informações coletadas podem proporcionar às autoridades dados suficientes para auxiliar na mitigação do problema.

Por outro lado, também temos questões de segurança pública. A pesquisa em Baba et al. (2019) propõe um algoritmo baseado em Redes Neurais Profundas para detecção de comportamentos violentos em Cidades Inteligentes. O projeto foi estruturado para tornar possível a execução do algoritmo em uma infraestrutura de redes de sensores, na qual dispositivos com baixo poder computacional acoplados a uma câmera sejam capazes de realizar o processo de detecção e alerta desses eventos.

Através das pesquisas apresentadas e da verificação de levantamentos bibliográficos prévios, como por exemplo, os trabalhos de Kandris et al. (2020) e Sharma et al. (2021), nota-se que as possíveis aplicações e metodologias em RSSF são diversas, trazendo relevantes contribuições para diferentes campos de estudo e problemas de pesquisa. Entretanto, o uso e a implementação dessas redes enfrentam desafios significativos, como limitações de energia, suscetibilidade a interferências, segurança no processo de transmissão de dados, além da limitação no poder computacional. Essas questões impactam diretamente a eficiência e o tempo de vida dos dispositivos (Akyildiz et al., 2002). Por exemplo, a gestão eficiente do consumo de energia é um critério crucial, uma vez que a substituição ou recarga frequente de baterias pode ser inviável em muitas aplicações.

Então, a otimização do posicionamento em RSSF torna-se um fator de grande importância na mitigação dos principais desafios associados ao uso dessa tecnologia. A exemplo disto, um arranjo adequado permite otimizar a distribuição de tarefas entre os nós, reduzindo o número de retransmissões e, consequentemente, o gasto energético com comunicação, que representa uma das principais fontes de consumo em sistemas sem fio. Além disso, a organização equilibrada dos sensores ao longo da região de monitoramento possibilita a formação de rotas de comunicação mais curtas e estáveis, contribuindo para o aumento da robustez da rede, segurança e redução de interferências entre os enlaces de transmissão. De modo complementar, um posicionamento planejado favorece a eficiência do processamento distribuído e o gerenciamento dos recursos disponíveis, o que se reflete em maior tempo de vida útil da rede e maior confiabilidade dos dados coletados.

Diante desses fatores, o posicionamento otimizado dos elementos da rede não apenas

aprimora o desempenho global do sistema, mas também representa uma estratégia essencial para viabilizar aplicações de larga escala em ambientes complexos e com distintas restrições. Técnicas de otimização, como algoritmos evolutivos, têm sido utilizadas para determinar a disposição ideal dos sensores, levando em consideração fatores como a topologia do terreno, obstáculos e requisitos específicos da aplicação (Fei et al., 2017).

Em suma, as RSSF representam uma tecnologia promissora com ampla gama de aplicações. No entanto, para maximizar seu potencial, é essencial abordar os desafios inerentes a essas redes, buscando contribuir para o desenvolvimento de sistemas mais eficientes e robustos.

## 2.2 Modelos de Detecção

Como mencionado na Seção 2.1, as RSSF são compostas por um conjunto homogêneo ou heterogêneo de dispositivos de monitoramento amplamente utilizados em diversas aplicações, sendo empregados para detectar grandezas físicas como temperatura, umidade, pressão, mas também para monitorar e detectar objetos. Logo, durante a etapa de modelagem dos estudos associados à RSSF, é preciso estabelecer modelos que representem o comportamento da detecção desses dispositivos de acordo com a aplicação desejada. Para essa finalidade, foram criados os Modelos de Detecção; estruturas que caracterizam a maneira pela qual os sensores pertencentes à rede percebem ou medem fatores ambientais ao seu redor (Tripathi et al., 2018).

Esses modelos são importantes tanto para o processo de *design* quanto para a avaliação do desempenho da rede, pois auxiliam na identificação das áreas monitoradas pelos sensores e, posteriormente, no cálculo da taxa de cobertura; permitem simular com certa precisão a qualidade de monitoramento dos dispositivos com base na distância; além disso, a modelagem adequada do sensoriamento contribui para um *design* robusto e confiável da rede, aumentando a confiabilidade do estudo em ambientes reais.

No geral, os modelos de detecção são categorizados em Determinístico e Probabilístico. Das várias propostas encontradas na literatura, temos três principais que são: o Modelo Binário de Detecção, e os Modelos Probabilísticos *Elves Sensing Model* e *Shadow-Fading Sensing Model*. Cada um possui suas próprias características, tornando-os adequados para diferentes cenários de estudo e aplicações em uma RSSF (Amutha et al., 2020).

### 2.2.1 Modelo de Detecção Determinístico (binário)

O modelo binário é a representação mais simples encontrada na literatura, na qual o monitoramento de um sensor é percebido apenas através de duas possibilidades: presença ou ausência de um fenômeno ou objeto. Sendo representado da seguinte forma: Seja  $p$  uma posição dentro de uma área de monitoramento,  $sr_i$  o i-ésimo

sensor pertencente à rede,  $r_s$  o seu raio de sensoriamento, e  $d(u, sr_i)$  a distância euclidiana entre  $u$  e  $sr_i$ ; um objeto ou fenômeno qualquer em  $u$  será detectado por  $sr_i$  se a  $d(u, sr_i)$  for menor do que  $r_s$ . A Equação 2.1 e a Figura 2.1(a) demonstram esse comportamento. Caso seja detectada a presença de um fenômeno ou objeto em  $u$ , o valor 1 é utilizado e 0, caso contrário.

$$p(u, sr_i) = \begin{cases} 1, & d(u, sr_i) \leq r_s \\ 0, & c.c \end{cases} \quad (2.1)$$

A partir da forma como a área de monitoramento de um sensor é estabelecida, a Equação 2.1 pode ser utilizada para simular o sensoriamento. Tomando como exemplo os sensores escalares, essas áreas são idealizadas como uma circunferência com raio máximo pré-estabelecido. Veja a Figura 2.1(b). Através de uma busca em pesquisas publicadas, nota-se que, devido à sua baixa complexidade e facilidade de implementação, grande parte dos estudos em posicionamento de sensores utiliza essa modelagem. Pesquisas como Benatia et al. (2017), Binh et al. (2018), El-Sherif et al. (2018), Wu et al. (2020) e Birtane et al. (2024) são alguns exemplos.

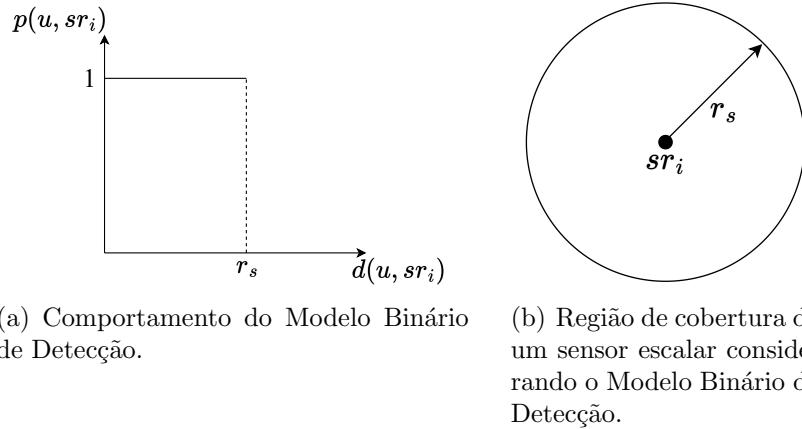


Figura 2.1: Modelo Binário de Detecção.

### 2.2.2 Modelo de Detecção Probabilístico - *Elfes Sensing Model*

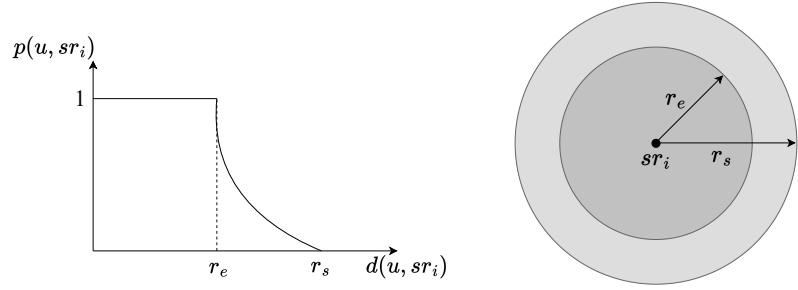
Em cenários reais de implantação dos dispositivos de sensoriamento, o efeito de decaimento na precisão com que eventos e objetos são detectados não pode ser ignorado. Ou seja, a probabilidade de detecção de um sensor diminui conforme a distância entre o alvo e o dispositivo aumenta, pois esse processo de monitoramento torna-se mais suscetível a ruídos externos e imprecisões. Por causa disso, pesquisas como Wang e Tseng (2008), Wang e Chung (2012), Yang et al. (2015) e Chen et al. (2020) investigaram o *Elfes Sensing Model* sob o requisito de alcançar uma representação mais realista do comportamento de um sensor.

Essa nova modelagem é representada da seguinte forma: Considere as mesmas variáveis  $u$ ,  $sr_i$  e  $r_s$  do modelo anterior e a distância euclidiana  $d(u, sr_i)$ . Veja a Figura 2.2(a) e a Equação 2.2a. A precisão com que um sensor detecta um evento ou objeto continua em alta caso o mesmo seja encontrado a uma distância menor do que um limiar ( $r_e$ ) pré-configurado. Caso essa distância seja maior do que  $r_e$ , esse valor começa a diminuir a partir de uma função exponencial onde as variáveis  $\beta$  e  $\gamma$  são usadas como parâmetros extraídos do sensor para definir a configuração da curva de decaimento. Quando o valor da distância estiver muito próximo de  $r_s$ , a precisão estará próxima de zero.

Para as aplicações em sensores escalares, a região de cobertura fica subdividida em duas seções, como mostra a Figura 2.2(b). E, a partir da Equação 2.2a, dependendo do estudo, é possível simular a curva de decaimento de acordo com as características dos sensores utilizados. Veja na Figura 2.3 uma simulação para diferentes valores aplicados aos parâmetros  $\beta$  e  $\gamma$ .

$$p(u, sr_i) = \begin{cases} 1, & d(u, sr_i) \leq r_e, \\ e^{-\gamma * \alpha^\beta}, & r_e < d(u, sr_i) \leq r_s, \\ 0, & \text{otherwise} \end{cases} \quad (2.2a)$$

$$\alpha = d(u, sr_i) - r_e \quad (2.2b)$$



(a) Comportamento do *Elfes Sensing Model*. (b) Região de cobertura de um sensor escalar considerando o *Elfes Sensing Model*.

Figura 2.2: Representação do *Elfes Sensing Model*.

### 2.2.3 Modelo de Detecção Probabilístico - *Shadow-Fading Sensing Model*

Esse modelo consiste na formulação mais complexa dentre os três apresentados. Pode ser utilizado em aplicações de RSSF que consideram em sua modelagem o impacto de obstáculos na capacidade de detecção dos sensores. Tais obstáculos podem ser dos mais variados, como edifícios, muros e carros dentro da área de monitoramento.

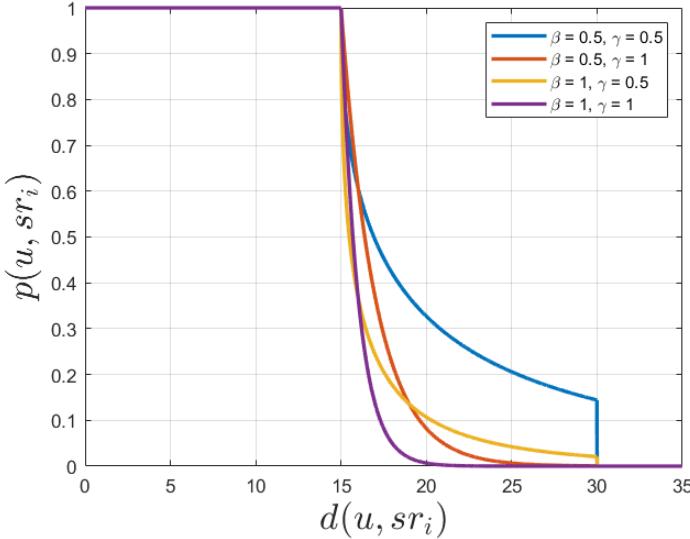


Figura 2.3: Configuração da curva de decaimento no *Elves Sensing Model* com diferentes valores de  $\beta$  e  $\gamma$ .

Essa representação considera que a precisão na detecção não possui um espalhamento uniforme em todas as direções, como ocorre no Modelo Binário ou no *Elves Sensing Model*; desta forma, assemelha-se ao conceito de sombreamento na propagação de ondas de rádio.

A partir da pesquisa apresentada em Hossain et al. (2008), assumindo o modelo de perda de caminho de sombreamento log-normal, a probabilidade de detecção de um evento localizado em  $u$  a uma distância  $x = d(u, sr_i)$  de um sensor é dada pela Equação 2.3a, onde  $n$  é o expoente de perda de percurso ( $2 \leq n \leq 4$ ) e  $\sigma$  é o parâmetro de desvanecimento. Outros trabalhos como Liu et al. (2009) e Kumar e Lobiyal (2013) também abordaram a utilização desse modelo.

$$P_{det}(x) = Q\left(\frac{10n \log_{10}(x/r_s)}{\sigma}\right) \quad (2.3a)$$

$$Q(x) \triangleq \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{-y^2/2} dy \quad (2.3b)$$

## 2.3 Classificação das estratégias de posicionamento de sensores

Os estudos associados ao posicionamento de sensores pertencentes a uma RSSF podem ser classificados em diferentes categorias a depender de alguns aspectos da pesquisa. Dentre as categorias existentes, está a estratégia de posicionamento, o objetivo de uso da aplicação e o domínio de implantação. Veja na Figura 2.4 como

estão estruturadas. Em pesquisas como Amutha et al. (2020) e Sharma et al. (2016) encontra-se uma visão detalhada e abrangente sobre esses tópicos, suas principais características e algoritmos utilizados, pois a implantação eficaz dos elementos da rede é um grande ponto de preocupação, tendo em vista que o desempenho e a vida útil de qualquer RSSF dependem desses aspectos.

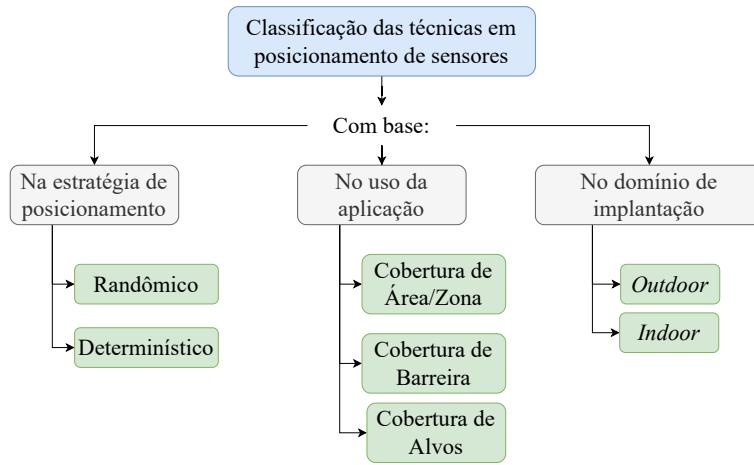


Figura 2.4: Classificação das estratégias usadas nas pesquisas em posicionamento de sensores.

### 2.3.1 Estratégia de posicionamento determinístico e randômico

Pode-se afirmar que o tipo de área em que os dispositivos serão implantados é o fator de maior impacto para determinar a estratégia de posicionamento (Sharma et al., 2016). Dentre essas estratégias, temos o posicionamento determinístico, aplicado em estudos onde a implantação de cada sensor ao longo da região de monitoramento é realizada em locais previamente selecionados, de forma manual ou com baixa complexidade, levando em consideração diversos critérios como cobertura, conectividade, consumo de energia, dentre outros, garantindo um melhor desempenho da rede.

Ao contrário do processo determinístico, temos o posicionamento randômico. Nesse processo, os sensores são dispersos aleatoriamente ao longo da região de monitoramento sem nenhum tipo de critério prévio. É preciso algum método de localização para identificar a posição de cada sensor. Essa estratégia torna-se viável em cenários onde a intervenção humana é complexa ou quando as informações sobre a área de implantação não são previamente conhecidas. Podemos citar florestas, regiões de desastre, campos de batalha, áreas vulcânicas e outras em que o posicionamento randômico é a melhor estratégia a ser utilizada (Amutha et al., 2020). Contudo, com essa estratégia, aspectos como o gerenciamento de falhas, conectividade e maximização da cobertura tornam-se mais complexos.

### 2.3.2 Tipos de Cobertura em RSSF

Com relação à aplicação das RSSF, o posicionamento dos sensores pode ser categorizado conforme o objetivo da rede, dentre os quais a Cobertura de Área/Zona, Cobertura de Barreira e Cobertura de Alvos são os mais comumente encontrados na literatura. Observe suas diferenças através da Figura 2.5.

A Cobertura de Área/Zona busca distribuir os sensores sobre uma área delimitada, buscando alcançar a maior taxa de cobertura possível sobre uma região (ver Figura 2.5(a)). Isso é frequentemente usado para entender ou monitorar as características gerais de eventos, pois alguns podem ocorrer em pontos aleatórios, sendo necessária a cobertura completa da região de monitoramento para identificar suas ocorrências (Birtane et al., 2024).

Já a Cobertura de Barreira consiste no posicionamento de sensores com o objetivo de realizar a detecção de intrusos ou objetos que ultrapassem um limite estabelecido (ver Figura 2.5(b)). Esse tipo de cobertura pode ser utilizado em aplicações militares e de segurança em ambientes internos, incluindo controle de fronteiras ou detecção de movimento, proteção de infraestruturas críticas, vigilância de perímetros, dentre outras (Wang et al., 2011). Comparado à cobertura de área, a cobertura de barreira requer uma quantidade menor de sensores, pois seu monitoramento está associado apenas a uma fatia da região ou na delimitação de marcos na área. Também, segundo Mohamed et al. (2017), a cobertura de barreira pode ser descrita como forte ou fraca; barreiras fortes asseguram a detecção de objetos em qualquer caminho que eles tomem, enquanto barreiras fracas asseguram apenas a detecção de objetos que passam por caminhos congruentes.

Por fim, encontra-se a Cobertura de Alvos, que requer o monitoramento de um conjunto de pontos dentro da região de interesse (ver Figura 2.5(c)); portanto, é mais eficiente em aplicações que exigem a detecção de objetos em movimento (Mohamed et al., 2017). A rede de sensores trabalha no monitoramento de um conjunto limitado de alvos com localização previamente conhecida, de modo que cada um deles seja coberto por pelo menos um sensor, assim garantindo que todos os sensores dentro da região sejam utilizados para monitorar os alvos (Amutha et al., 2020).

### 2.3.3 Posicionamento *Indoor* e *Outdoor*

A depender do local em que os sensores serão implantados, o processo de implantação pode ser classificado como posicionamento *indoor* ou *outdoor*. Desta forma, diferentes estratégias e técnicas podem ser escolhidas a fim de realizar o processo.

O posicionamento *indoor*, também conhecido como implantação em ambientes internos, tem o seu principal foco em pequenas e confinadas áreas, como os andares de um edifício, fábricas e hospitais (Priyadarshi et al., 2020). Nesse tipo de monitoramento, barreiras físicas como paredes e móveis podem afetar o processo de detecção de eventos, além de causar interferências e atenuação do sinal de comunicação dos sensores. Além disso, nesses ambientes, garantir a precisão do posicionamento e

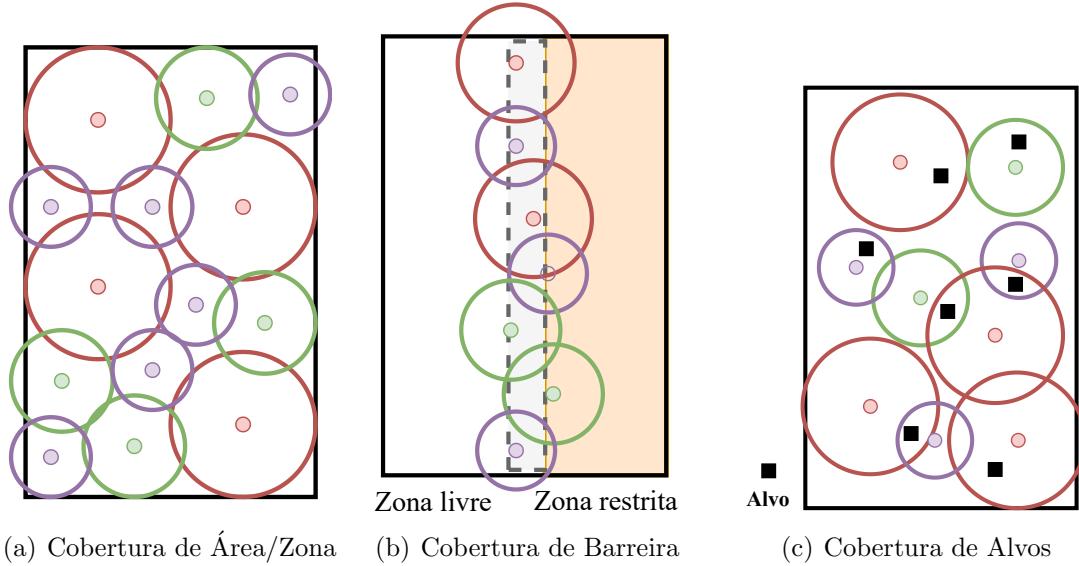


Figura 2.5: Tipos de Cobertura aplicados através das RSSF.

a cobertura uniforme é um desafio, especialmente em cenários dinâmicos onde a disposição dos objetos pode mudar constantemente.

Já o posicionamento *outdoor*, ou implantação em áreas externas/abertas, está associado desde à alocação de sensores em ambientes de fácil acesso, como áreas urbanas e campos agrícolas, até locais remotos onde a implantação randômica de sensores se faz necessária, como áreas afetadas por desastres ambientais e regiões com grande índice de radiação prejudicial à saúde (Priyadarshi et al., 2020). Diferentemente do posicionamento *indoor*, a detecção de eventos é menos obstruída; contudo, está sujeita a interferências ambientais, como variações climáticas, obstáculos naturais e desgaste dos dispositivos devido à exposição prolongada. Um dos principais desafios nesse cenário é garantir a eficiência energética dos sensores, uma vez que a substituição ou recarga das baterias pode ser inviável em certas aplicações remotas ou em aplicações nas quais um grande número de sensores é utilizado.

Assim, tanto no posicionamento *indoor* quanto no *outdoor*, é essencial adotar estratégias eficientes para superar as limitações específicas de cada ambiente e garantir o desempenho adequado da rede de sensores. Então, a partir da descrição dos conceitos apresentados nesta seção e nas subseções anteriores, a Tabela 2.1 busca demonstrar por meio de alguns trabalhos como as pesquisas em posicionamento de sensores podem ser categorizadas de acordo com as técnicas e estratégias mais comuns encontradas na literatura.

Tabela 2.1: Classificação de trabalhos de acordo aos modelos de detecção e às categorias associadas ao processo de posicionamento. **LEGENDA:** Posicionamento: DT(Determinístico) / RD(Randômico); CB(Cobertura); Implantação: In(*Indoor*)/Out(*Outdoor*); MD(Modelo de Detecção)

<b>Pesquisa</b>	<b>DT/RD</b>	<b>CB</b>	<b>In/Out</b>	<b>MD</b>
Cardei et al. (2002)	RD	Área/Zona	Out	Binário
Ghosh e Das (2008)	DT e RD	Área/Zona	Out	Binário e <i>Elfes Sensing</i>
Chen et al. (2010)	RD	Barreira	Out	Binário
Costa et al. (2015)	DT	Alvos	Out	Binário
Binh et al. (2018)	DT	Área/Zona	Out	Binário
Rangel et al. (2019)	DT	Alvos	Out	Binário
Hanh et al. (2019)	RD	Área/Zona	Out	Binário
Fan et al. (2019)	DT e RD	Barreira	Out	Binário
Harizan e Kuila (2020)	DT	Alvos	In	Binário
Wu et al. (2020)	DT	Área/Zona	In	Binário
Chakraborty e Rout (2021)	DT	Barreira	Out	Binário

## 2.4 Computação Evolutiva

Um campo de estudo amplamente empregado nas pesquisas sobre posicionamento de RSSF é a Computação Evolutiva (CE). Consiste em um ramo da inteligência computacional que se inspira no processo de evolução natural para resolver problemas complexos de otimização e busca quando os métodos e estratégias tradicionais não são viáveis (Eiben e Smith, 2015). Em estudos associados às RSSF, a CE, por meio de diversos métodos evolutivos, busca otimizar a distribuição dos dispositivos de forma a maximizar, por exemplo, a cobertura (Hanh et al., 2019), minimizar o consumo energético Yetgin et al. (2015) e garantir a conectividade da rede (Benatia

et al., 2017). A flexibilidade desses métodos permite lidar com as restrições impostas por ambientes dinâmicos e complexos, tornando-os uma abordagem promissora para aplicações práticas em monitoramento e sensoriamento inteligente.

O conceito de evolução natural, oriundo das ideias de Darwin, aplicado à automatização do processo de busca por soluções de problemas começou na década de 40, antes da criação dos computadores. Em 1948, Allan Turing surge com a proposta da pesquisa genética ou pesquisa evolucionária; em 1962, Bremermann (1962) começou a executar experimentos computacionais em otimização através da evolução e recombinação; em 1964, dois alemães chamados Rechenberge e Schwefel desenvolveram as Estratégias Evolutivas; um ano depois, em 1965, surge a Programação Evolutiva através de três pesquisadores americanos (Fogel et al., 1965) (Fogel et al., 1966). Por fim, no início da década de 90, Koza (1992) e Koza (1994) apresentaram outro campo chamado Programação Genética.

Durante vários anos, essas áreas foram desenvolvidas separadamente até que, também no início dos anos 90, notou-se que esses estudos estavam correlacionados. Assim, a Programação Evolutiva, Estratégias Evolutivas, Algoritmos Genéticos e Programação Genética foram categorizados como subáreas de um amplo campo de estudo que ficou conhecido como Computação Evolutiva (Eiben e Smith, 2015). Todos os algoritmos utilizados dentro da CE são denominados como Algoritmos Evolutivos (AE), proporcionando a exploração de grandes espaços de soluções, permitindo encontrar configurações eficientes mesmo em problemas de alta dimensionalidade.

#### **2.4.1 Algoritmos Evolutivos**

Esses algoritmos constituem uma classe de meta-heurísticas inspiradas nos processos de evolução biológica, como seleção natural, mutação e recombinação genética, aplicadas à resolução de problemas de otimização e busca. Durante a evolução conduzida pelos algoritmos, as soluções são avaliadas e selecionadas progressivamente, aprimorando sua qualidade até que um critério de parada seja atingido, seja pela obtenção de uma solução satisfatória, seja pelo esgotamento das tentativas de melhoria. (Eiben e Smith, 2015).

O processo evolutivo aplicado gera um conjunto de soluções progressivamente mais ajustadas às restrições e condições do problema em estudo. Esse aprimoramento reflete o conceito de seleção natural na CE, no qual, em uma população de indivíduos inseridos em um ambiente de recursos limitados, a competição por esses recursos promove, de forma probabilística, a sobrevivência dos mais aptos. Nesse contexto, os indivíduos representam as soluções, enquanto o ambiente e suas restrições correspondem ao problema em análise.

Segundo Eiben e Smith (2015), existem duas estruturas principais que formam a base dos sistemas evolutivos, que são:

- Operadores de Variação: criam a diversidade necessária no conjunto de soluções, assim, facilitando a exploração dentro do espaço de busca.

- Seleção: busca selecionar as melhores soluções durante o processo evolutivo, proporcionando maior qualidade média do conjunto a cada geração.

Os operadores de variação, a seleção e a análise de qualidade são considerados os componentes principais de um AE. Veja no Algoritmo 1, o seu esquema geral. Perceba que esse processo de busca e aprimoramento das soluções é realizado repetidamente e, como dito anteriormente, até que uma condição de parada seja alcançada.

---

**Algorithm 1:** Esquema geral de um Algoritmo Evolutivo (Eiben e Smith, 2015).

---

```

1 begin
2   INITIALISE population with random candidate solutions;
3   EVALUATE each candidate;
4   repeat
5     SELECT parents;
6     RECOMBINE pairs of parents;
7     MUTATE the resulting offspring;
8     EVALUATE new candidates;
9     SELECT individuals for the next generation;
10    until TERMINATION CONDITION is satisfied;
11 end

```

---

Com base na estrutura do Algoritmo 1 podemos detalhar as características dos principais componentes que o compõem. Primeiramente, começamos pelo conceito geral de alguns itens (Eiben e Smith, 2015):

- **Indivíduo:** Termo na CE para referenciar uma solução, também conhecida com solução candidata;
- **População:** É a unidade base do processo evolutivo. Formalmente, consiste em um conjunto de indivíduos, no qual, sua diversidade está associada à medida do número de diferentes indivíduos pertencentes ao mesmo conjunto.
- **Fitness:** Trata-se de uma estimativa heurística da qualidade de um indivíduo com base em sua adaptação ao ambiente em questão. No contexto da computação, isso equivale à sua eficiência na resolução do problema em análise.

Esses conceitos fundamentam a compreensão dos principais componentes de um AE, que incluem a representação dos indivíduos, a função de avaliação, os operadores de variação e os mecanismos de seleção.

#### 2.4.2 Representação dos indivíduos

Essa pode ser considerada uma das etapas mais importantes no processo de uso dos AE, pois define a codificação que traduz o problema original para o espaço de busca onde ocorrerá o processo evolutivo. Funciona como uma ponte entre o mundo real e o domínio dos AE, exigindo, muitas vezes, a simplificação ou abstração de certos aspectos do problema para torná-lo bem definido e passível de avaliação.

Esse processo, essencial para a eficácia do algoritmo, é geralmente conduzido por especialistas na área que capturam a representação das soluções no contexto original do problema, conhecida como fenótipos, e os codificam para uma representação chamada genótipo, que permite a manipulação por parte dos AE. Ao final da execução, a melhor solução é selecionada e decodificada, levando-a ao contexto do problema original, tornando possível sua implantação.

Com a codificação, os AE passam a realizar a busca e o aprimoramento dos indivíduos dentro do Espaço de Genótipo, onde se encontram todas as soluções factíveis. Cada solução é representada por meio de uma estrutura mais conhecida como Cromossomo, oriunda da biologia. Cada cromossomo contém genes, os quais estão em posições chamadas *Locus* - usualmente fixas - e possuem valor atrelado chamado *alelo* (Eiben e Smith, 2015).

Nos AE, a representação de um cromossomo desempenha um papel fundamental na eficiência do algoritmo, influenciando diretamente a forma como as soluções são exploradas e avaliadas. Diferentes estratégias de codificação podem ser adotadas, dependendo da natureza do problema e das características do espaço de busca. Em Eiben e Smith (2015), podemos encontrar a descrição das representações mais comumente usadas, as quais são binária, inteira, real e também no formato de árvores.

A representação binária é uma das mais simples e tradicionais. Ela utiliza uma cadeia de bits (0s e 1s) para descrever soluções. Essa abordagem é amplamente empregada em problemas combinatórios e de otimização discreta, onde cada gene do cromossomo representa uma característica específica da solução. Essa representação facilita a aplicação de operadores genéticos, como mutação e cruzamento, mas pode exigir técnicas adicionais para interpretar os valores codificados em domínios mais complexos. Por exemplo, em Wu et al. (2020), os autores realizaram a otimização do posicionamento de sensores ao longo de uma região, na qual foi definida a codificação binária para o cromossomo. Cada gene representou um bloco de área, e os alelos, a ausência (0) ou presença (1) de um sensor em um bloco específico.

Também temos a representação inteira. Utilizada quando as soluções envolvem valores discretos, como números inteiros que representam índices, alocações ou agrupamentos. Essa abordagem é especialmente útil em problemas onde cada gene do cromossomo pode indicar um identificador ou uma posição em um conjunto de possibilidades.

Para problemas que envolvem variáveis contínuas, a representação real é mais apropriada, permitindo maior precisão na busca por soluções ótimas. Em vez de cadeias binárias ou inteiras, os cromossomos são representados diretamente por números reais, o que elimina a necessidade de conversões e facilita a aplicação de operadores de variação mais sofisticados, como mutações baseadas em distribuições estatísticas e cruzamentos aritméticos.

A maioria dos estudos sobre posicionamento de sensores que empregam algoritmos evolutivos adotam a representação inteira ou real. Os trabalhos Domingo-Perez et al.

(2016), Binh et al. (2018), Bouzid et al. (2020) e Birtane et al. (2024) são alguns exemplos encontrados na literatura.

Em problemas estruturais e mais complexos, como a evolução de programas ou redes neurais, pode-se utilizar a representação em árvores, característica da Programação Genética. Nesse modelo, os cromossomos são organizados hierarquicamente, onde cada nó da árvore representa uma operação ou função, permitindo a evolução de expressões matemáticas, regras de decisão ou mesmo trechos de código-fonte (Koza, 1992).

Segundo Eiben e Smith (2015), a escolha da representação adequada para um problema é um fator crucial. Em alguns casos, mais de uma representação pode ser utilizada; no entanto, a experiência prática e um bom conhecimento do domínio da aplicação são essenciais para a tomada de decisão.

### **2.4.3 Função de Avaliação**

As funções de avaliação, também conhecidas como funções de *fitness* ou funções objetivo, são expressões matemáticas ou procedimentos utilizados para medir a qualidade de cada solução encontrada, isto é, dos indivíduos de uma população (Eiben e Smith, 2015). Do ponto de vista da resolução de problemas, essas funções representam as tarefas a serem resolvidas no contexto evolutivo, de modo que seus respectivos valores reais sejam usados como parâmetros de medição da qualidade das soluções do problema em análise.

### **2.4.4 Operador de variação - Mutação**

Como pode ser visto no Algoritmo 1, o operador de mutação é um dos agentes do processo de aprimoramento e evolução das soluções. Seu objetivo é introduzir variações pequenas e aleatórias nos indivíduos, prevenindo a convergência prematura e assegurando a diversidade genética na população. Essa diversidade proporciona o aumento da exploração dentro do espaço de busca.

Os operadores de mutação atuam sobre um único indivíduo (pai) para gerar outro indivíduo (filho). O tipo de mutação a ser aplicado depende da codificação escolhida, assim como o significado do parâmetro que, geralmente, é introduzido para regular a intensidade ou magnitude da operação (Eiben e Smith, 2015). Esses parâmetros podem ser conhecidos como Probabilidade de Mutação, Taxa de Mutação, tamanho do passo de mutação, dentre outros nomes.

Tendo em vista os tipos de codificação mais usuais aplicadas ao contexto do posicionamento de sensores (binária, inteira e real), podemos abordar os principais métodos de mutação aplicados nesses tipos de representação.

Para a representação binária, a mutação mais usualmente aplicada é a Mutação por Inversão de Bit. Genes do cromossomo são selecionados aleatoriamente através de

um baixo valor de probabilidade e os bits armazenados são invertidos (0 torna-se 1 e vice-versa).

Para a representação real ou inteira, podemos citar dois métodos aplicáveis em ambos os tipos (Michalewicz, 2013):

- **Mutação Uniforme: (*Uniform Mutation*)** Um gene é escolhido aleatoriamente dentro do cromossomo, e o seu valor é substituído por um novo, selecionado aleatoriamente dentro de um intervalo permitido.
- **Mutação Não-Uniforme: (*Non-uniform Mutation*)** É um operador adaptativo, no qual a intensidade da mutação diminui ao longo da execução do processo de busca. Ao selecionar um gene, em vez de substituir seu valor por um aleatório, é aplicada uma pequena perturbação controlada, cujo tamanho diminui conforme o número de gerações cresce. O AE deve explorar amplamente o espaço de busca, mas, conforme se aproxima do critério de parada, as mutações devem se tornar mais delicadas, refinando as soluções.

Na literatura é possível encontrar outros métodos amplamente conhecidos como a Mutação Polinomial (Deb e Deb, 2014) e a Mutação Gaussiana (Higashi e Iba, 2003).

#### 2.4.5 Operador de variação - Cruzamento

Juntamente com a Mutação, o operador de Cruzamento, também conhecido como operador de recombinação, consiste no segundo agente a introduzir modificações em uma população de indivíduos em um AE. Sua função é realizar a troca de informações entre dois cromossomos (pais) para gerar um ou dois novos cromossomos (filhos). A aplicação dessa pequena troca pode gerar indivíduos que estão próximos dos pais, permitindo auxiliar a busca por soluções em regiões promissoras dentro do espaço de busca (Eiben e Smith, 2015).

Embora possam ocorrer combinações indesejáveis de características, como a troca entre soluções com baixo valor de *fitness*, e os descendentes nem sempre sejam superiores aos pais, espera-se que alguns herdem atributos aprimorados, contribuindo para melhores resultados nas gerações futuras produzidas pelo AE.

Assim como a mutação, os operadores de cruzamento são escolhidos com base na representação dos indivíduos. Para a codificação binária, podemos ter os seguintes métodos (Eiben e Smith, 2015):

- **Cruzamento de um ponto (*Single-point Crossover*):** Um ponto de corte é selecionado aleatoriamente na sequência de genes; os segmentos antes e depois desse ponto são trocados entre os pais para formar dois novos filhos.
- **Cruzamento de dois pontos (*Two-point Crossover*):** Dois pontos de corte são selecionados aleatoriamente no cromossomo, em seguida, os seguimentos formados por essa divisão são trocados entre os pais, formando assim dois novos indivíduos.

- **Cruzamento de N pontos (*n-Point Crossover*):** Consiste em uma generalização do cruzamento de um e dois pontos. O cromossomo é dividido em  $n$  pontos de corte, criando  $n + 1$  seguimentos de genes; os seguimentos são trocados entre os pais para formar os dois novos indivíduos.
- **Cruzamento Uniforme: (*Uniform Crossover*)** A origem dos alelos de cada gene dos cromossomos filhos são escolhidos aleatoriamente entre os dois pais. Implementa-se gerando uma sequência de  $n$  (tamanho do cromossomo) variáveis aleatórias de uma distribuição uniforme sobre o intervalo  $[0,1]$ . Para cada gene, se o valor da variável correspondente estiver abaixo de um parâmetro  $p$  (geralmente 0,5), o alelo é herdado do primeiro cromossomo pai; caso contrário, do segundo. O procedimento para geração do segundo indivíduo seguirá o mapeamento inverso.

Na codificação inteira, quando cada gene pode assumir um número finito de valores inteiros, é comum aplicar o mesmo conjunto de operadores utilizados na codificação binária. Já para a codificação real, temos alguns métodos amplamente conhecidos como o:

- *Simulated Binary crossover(SBX)* (Deb et al., 1995)
- *Arithmetic Crossover* (Michalewicz, 2013)
- *Laplace Crossover* (Deep e Thakur, 2007)
- *Blend Crossover* (Eshelman et al., 1989)

#### 2.4.6 Mecanismos de Seleção

Os mecanismos de seleção são responsáveis por escolher os indivíduos mais aptos ou de maior qualidade a fim de serem aplicados no processo de reprodução, mas também, posteriormente estruturar a próxima geração de soluções para o seguimento do processo evolutivo (veja no Algoritmo 1), influenciando diretamente sua eficiência e eficácia.

A função do processo de seleção de indivíduos para reprodução é baseada na distinção por qualidade, buscando a escolha dos melhores indivíduos para possivelmente tornarem-se os pais da próxima geração, pois essa seleção é tipicamente probabilística. Nessa etapa, indivíduos de alta qualidade têm maior probabilidade de serem selecionados, enquanto aqueles de baixa qualidade ainda podem ser escolhidos, porém com menor frequência. Essa natureza estocástica ajuda a evitar o aprisionamento em ótimos locais dentro do espaço de busca (Eiben e Smith, 2015).

Ao longo dos anos, diversos métodos de seleção para essa finalidade foram desenvolvidos, cada um com características específicas que afetam a diversidade e a convergência da população. Dentre eles, podemos citar a seleção por torneio, seleção por torneio binário, seleção por roleta e a seleção por amostragem universal estocástica.

A seleção por roleta atribui a cada indivíduo uma probabilidade de ser selecionado proporcional ao seu *fitness*. Pode-se imaginar uma roleta dividida em segmentos, onde cada segmento representa um indivíduo e seu tamanho é proporcional ao *fitness*. Ao girar a roleta, indivíduos com maior qualidade possuem chances mais elevadas de serem escolhidos para a próxima geração. Esse método favorece as soluções de alta qualidade, pois ocupam maiores proporções na roleta, mas também permite que, ocasionalmente, indivíduos menos aptos sejam selecionados, mantendo a diversidade genética na população (Coello, 2007).

Por outro lado, a seleção pela amostragem universal estocástica é uma variação do método da roleta que busca reduzir a variância associada ao processo de seleção. Nesse método, são utilizados múltiplos ponteiros igualmente espaçados para selecionar os indivíduos de uma única vez (Eiben e Smith, 2015). Essa abordagem assegura que indivíduos com alta aptidão sejam selecionados proporcionalmente, enquanto mantém a diversidade genética da população. Ao utilizar múltiplos ponteiros, a amostragem universal estocástica evita a possível variância de filhos observada no método da roleta tradicional.

Na seleção por torneio, seleciona-se, aleatoriamente e com igual probabilidade,  $N$  indivíduos da mesma população. Dentre os  $N$  selecionados, aquele com maior *fitness* é escolhido para a próxima geração. Esse processo se repete até que a quantidade necessária de indivíduos seja escolhida (Coello, 2007). Maiores valores de  $N$  aumentam a pressão seletiva, favorecendo os indivíduos mais aptos, enquanto torneios menores permitem maior diversidade genética. O valor de  $N$  deve ser bem definido, de forma a evitar convergências prematuras e baixa diversidade devido à pressão seletiva muito alta, ou mesmo o baixo nível de convergência devido à pressão muito baixa.

Por fim, a seleção por torneio binário é uma variação específica da seleção por torneio, ou seja, quando  $N = 2$ . Devido ao seu tamanho de torneio fixo e pequeno, este método introduz uma pressão seletiva moderada. Isso ajuda a manter a diversidade genética na população e evita a convergência prematura para soluções subótimas.

Similarmente à seleção para reprodução, a seleção de sobrevivência distingue os indivíduos com base em sua qualidade. Contudo, esse mecanismo é aplicado em um estágio diferente dentro do processo evolutivo, pois definirá quais indivíduos permanecerão ativos após o processo de reprodução, podendo adotar estratégias elitistas, nas quais as melhores soluções são preservadas, ou métodos estocásticos buscando manter a diversidade populacional (Eiben e Smith, 2015).

Por último, encontra-se a seleção de indivíduos para a próxima geração, de maneira a estabelecer o conjunto de soluções que irá compor a população subsequente, equilibrando a exploração de novas regiões do espaço de busca com a exploração de soluções promissoras já encontradas.

## 2.5 Algoritmo Genético

Dentre as diversas variantes dos algoritmos evolutivos criadas ao longo do tempo, os Algoritmos Genéticos (AG) são a classe mais conhecida (Eiben e Smith, 2015). Inicialmente, foram introduzidos por um pesquisador conhecido como *John Holland* (Holland, 1992) e, posteriormente, popularizados por Goldberg (1989).

O AG é considerado como uma técnica de pesquisa heurística usada na CE para encontrar soluções exatas ou aproximadas para problemas de busca e otimização (Eiben e Smith, 2015). Sua aplicabilidade se estende por várias áreas do conhecimento, beneficiando pesquisas em Gestão de Operações (Chaudhry e Luo, 2005), Engenharia (Paszkowicz, 2009), Finanças (Aguilar-Rivera et al., 2015) e, como esta pesquisa propõe, na otimização do posicionamento de sensores em uma RSSF.

Dentre as suas vantagens está a capacidade de lidar com extensos e complexos espaços de busca, onde métodos tradicionais são falíveis, e podem ser aplicados a uma variedade de problemas sem a necessidade de realizar adaptações significativas. Entretanto, os AG podem ser computacionalmente custosos em tempo de execução caso necessitem avaliar um número expressivo de soluções até alcançar o resultado desejado, causando uma lenta convergência para a solução ótima. E também, são altamente dependentes da escolha dos seus hiperparâmetros, como taxas de mutação e cruzamento, influenciando diretamente a eficiência do algoritmo.

Dependendo da quantidade de objetivos que um problema de otimização possua, os AG podem ser classificados como Mono-Objetivo ou Multiobjetivo. Essas duas classes possuem suas próprias características que, ao longo dos anos, foram exploradas através da sua aplicação em inúmeras pesquisas.

### 2.5.1 Mono Objetivo

Os AG mono-objetivos, como o próprio nome especifica, têm por função a otimização de um único objetivo, em problemas em que uma única métrica de desempenho é relevante. Por exemplo, as pesquisas de Hanh et al. (2019) e Birtane Akar et al. (2022) buscam realizar o posicionamento de uma RSSF apenas sobre o critério de maximização da cobertura de área. Outras aplicações também são possíveis, como a otimização de funções matemáticas complexas, tendo a finalidade de encontrar o valor mínimo ou máximo de uma função específica.

A estrutura do Algoritmo 1 é a base da modelagem de um AG mono-objetivo. O *Simple Genetic Algorithm (SGA)* ou Algoritmo Genético Canônico é o mais simples já proposto (Eiben e Smith, 2015). Seu esboço é formado pelo uso do método de cruzamento *Single-point crossover*, método de mutação *Bit flip*, seleção por roleta e representação binária para os indivíduos. Posteriormente, técnicas como Elitismo e Modelos Não-Geracionais foram adicionadas para oferecer convergência mais rápida, quando necessário.

Isso nos diz que, a depender dos resultados esperados em uma aplicação, variantes do SGA podem ser criadas. Contudo, tradicionalmente, os AG mono-objetivos sempre terão um fluxo básico a ser seguido.

Inicialmente uma população aleatória de  $m$  indivíduos é fornecida, vistas como possíveis soluções do problema. Aplica-se o processo de seleção para preencher uma população intermediária também de  $m$  indivíduos, permitindo duplicatas. Depois disso, são introduzidos os operadores de cruzamento e mutação. As soluções resultantes desse processo são avaliadas por meio da Função de *Fitness* estabelecida de acordo com o critério de otimização desejado, aplicando a busca pelos melhores indivíduos para formar a próxima geração. Essas etapas de variação e seleção são repetidas até que um determinado critério de parada seja alcançado, como (Eiben e Smith, 2015):

- O número limite de gerações;
- Queda da diversidade populacional abaixo de um determinado limite;
- O aprimoramento dos valores de *fitness* permanece abaixo de um valor limite por um determinado período de tempo, ou seja, por um número de gerações ou soluções avaliadas.

Com o passar das gerações, ao longo do processo de aprimoramento, a qualidade dos indivíduos vai sendo aumentada de forma a maximizar ou minimizar o objetivo desejado. Veja na Figura 2.6, o comportamento típico de um AG mono-objetivo a respeito da qualidade do melhor indivíduo ao longo das gerações produzidas, considerando uma função de *Fitness* com intervalo  $[0, 1]$ .

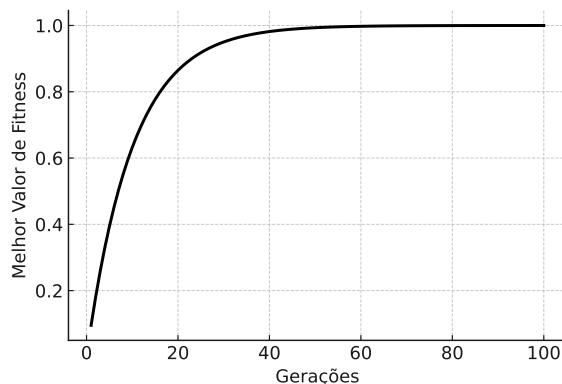


Figura 2.6: Comportamento de um AG mono objetivo no aprimoramento dos indivíduos ao longo das gerações.

### 2.5.2 Multiobjetivo

Os algoritmos e estratégias multiobjetivos são aplicados com a finalidade de buscar possíveis soluções para os Problemas de Otimização Multiobjetivo. Em geral, esses

problemas são formados por múltiplos objetivos a serem otimizados, simultaneamente, a partir de uma série de restrições. Nesse tipo de aplicação, o espaço de busca é maior em comparação aos problemas mono-objetivos, e os critérios a serem otimizados possuem propósitos tipicamente conflitantes, ou seja, o aprimoramento de um resulta na degradação dos outros (Coello, 2007).

Segundo Coello (2007), problemas multiobjetivo são modelados através de um conjunto de variáveis de decisão que satisfazem restrições, buscando otimizar diferentes funções que representam objetivos conflitantes. O uso do termo “otimizar” significa encontrar uma solução ou um conjunto de soluções que apresentem um balanceamento entre os critérios desejados.

Em sua representação, normalmente, encontramos as seguintes definições:

- $x = [x_1, x_2, x_3, \dots, x_n]^T$ : um vetor de variáveis de decisão;
- $f(x) = [f_1(x), f_2(x), f_3(x), \dots, f_m(x)]^T$ : vetor de funções objetivo;
- $g_j(x) \geq 0$  e  $h_k(x) = 0$ : restrições do problema que definem a região factível, tendo  $j = 1, 2, \dots, J$  e  $k = 1, 2, \dots, K$ ;
- $x_i^{(L)} \leq x_i \leq x_i^{(U)}$ : definem os limites para cada  $x_i$ , sendo  $i = 1, 2, \dots, n$ ;

Normalmente, nas estratégias multiobjetivo, encontra-se o conceito de Relação de Dominância entre as soluções, usada para guiar a evolução da população ao longo do processo de otimização. A partir de duas soluções  $A$  e  $B$ , dizemos que  $A$  domina  $B$  ( $A \preceq B$ ) se, somente se:

- $A$  é melhor do que  $B$  em pelo menos um objetivo;
- $A$  não é pior do que  $B$  em nenhum dos objetivos;

Matematicamente, se considerarmos um problema de minimização, os requisitos para  $A \preceq B$  podem ser descritos como na Equação 2.4. Para um problema de maximização, as desigualdades se invertem.

$$A \prec B \iff (\forall i \in \{1, 2, \dots, k\}, f_i(A) \leq f_i(B)) \wedge (\exists j \in \{1, 2, \dots, k\}, f_j(A) < f_j(B)) \quad (2.4)$$

A partir dessas Relações de Dominância, é dito que uma solução dentro do espaço de busca é não-dominada se não for dominada por nenhuma outra solução de uma população. Então, com a formação de um conjunto de soluções não-dominadas temos a criação do **Conjunto de Pareto-Ótimo**, no qual possui a sua representação no espaço de objetivos conhecida como **Fronteira de Pareto-Ótimo** (Coello, 2007).

Dentre as diversas abordagens para solucionar problemas multiobjetivos, podemos citar a estratégia de Soma Ponderada, Lexicográfica e o uso dos Algoritmos Evolutivos. Todas possuem a função de encontrar o conjunto de soluções não-dominadas

para formar a Fronteira de Pareto-Ótimo ou uma solução ótima global, de acordo com os critérios de Convergência e Diversidade.

Podemos começar detalhando a abordagem de Soma Ponderada. Consiste em técnica escalar para resolver problemas multiobjetivo, onde os objetivos são combinados em uma única função através de coeficientes de peso previamente definidos. Dessa forma, o problema multiobjetivo é transformado em um problema mono-objetivo, facilitando a resolução por métodos tradicionais. Esse método é simples e fácil de compreender e implementar. No entanto, apresenta algumas limitações, como a dificuldade de definir pesos adequados e a incapacidade de encontrar soluções em regiões não convexas da Fronteira de Pareto (Fei et al., 2017).

Na abordagem Lexicográfica, os objetivos são ordenados por níveis de prioridade, otimizando o mais importante primeiro e, em seguida, considerando os demais em ordem decrescente de relevância. Esse método é útil quando há uma clara hierarquia entre os critérios de decisão, mas pode levar a soluções insatisfatórias, pois objetivos de menor prioridade podem ser ignorados se entrarem em conflito com objetivos mais importantes. Além disso, essa abordagem não gera um conjunto diversificado de soluções, o que pode ser uma limitação em muitos problemas práticos (Coello, 2007).

Em comparação às abordagens mencionadas, o uso dos AE requer alto custo computacional, uma vez que são avaliados repetidamente diversas populações de indivíduos ao longo de múltiplas gerações, podendo, em alguns casos, inviabilizar a análise de problemas que apresentam elevado número de variáveis ou funções de *fitness* complexas. Além disso, os AE são fortemente dependentes da configuração dos seus parâmetros, os quais influenciam diretamente a convergência e a qualidade das soluções obtidas. Devido a essas características e natureza estocástica, os AE podem apresentar comportamento não reproduzível e convergência prematura, assim exigindo maior esforço de calibração e validação por parte dos pesquisadores a fim de alcançar resultados consistentes.

Por outro lado, os AE não exigem a definição prévia de pesos ou uma hierarquia fixa entre os objetivos, trabalham com um conjunto de soluções simultaneamente, permitindo a obtenção de uma aproximação diversificada da Fronteira de Pareto em uma única execução e, além disso, são menos vulneráveis à característica convexa, não convexa ou descontínua da Fronteira. Por fim, oferecem maior flexibilidade e melhor exploração do espaço de busca. Então, devido a essas vantagens, a pesquisa proposta neste documento adotou a otimização por meio de um AE multiobjetivo da classe dos AG.

## 2.6 Seleção de indivíduos nas Fronteiras de Pareto

Como dito anteriormente, com a finalização do processo evolutivo, os algoritmos multiobjetivos produzem um conjunto de soluções não-dominadas que formam a

Fronteira de Pareto. Entretanto, surge a necessidade da seleção de uma dessas soluções para fins de análise e implementação prática ou simulada. No artigo proposto por Wang e Rangaiah (2017), esse processo seletivo é baseado nos Métodos de Apoio à Decisão Multicritério, os quais buscam ordenar ou escolher a alternativa mais adequada com base nos objetivos previamente definidos.

Dentre vários métodos apresentados em Wang e Rangaiah (2017), desde os mais simples aos mais complexos, os autores interpretam que, devido à simplicidade e à aplicabilidade, os métodos *Technique for Order of Preference by Similarity to Ideal Solution* (TOPSIS), *Gray Relational Analysis* (GRA) e *Simple Additive Weighting* (SAW) são os mais adequados para a seleção de soluções não-dominadas em uma FP.

Como base para a compreensão desses métodos, considera-se a seguinte nomenclatura:

- Matriz de Objetivos: dimensionada por  $m$  linhas (soluções de uma fronteira) e  $n$  colunas (valores objetivos);
- $f_{ij}$ : i-ésimo valor do j-ésimo objetivo na matriz;
- $F_{ij}$ : valor normalizado de  $f_{ij}$ ;
- $V_{ij}$ : valor ponderado de  $F_{ij}$ ;
- $W_j$ : peso atribuído ao j-ésimo objetivo;

### 2.6.1 Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)

Esse método estabelece que a solução ótima deve apresentar menor distância em relação ao ponto ideal positivo (composto pelos melhores valores de cada objetivo) e, simultaneamente, maior distância em relação ao ponto ideal negativo (composto pelos piores valores). Assim, o método busca equilibrar o afastamento das situações indesejáveis com a aproximação dos resultados mais vantajosos. Seu resultado é uma medida de proximidade relativa, sendo escolhida a solução com maior índice de similaridade ao ideal.

O cálculo do TOPSIS segue uma lógica de aproximação, definida primeiramente pela Normalização da Matriz de Objetivos. A Equação 2.5 demonstra como esse passo é realizado. Se todos os valores objetivos estiverem normalizados, essa fase pode ser desconsiderada.

$$F_{ij} = \frac{f_{ij}}{\sqrt{\sum_{i=1}^m f_{ij}^2}} \quad (2.5)$$

Após a normalização, uma nova matriz é criada, na qual o valor de cada objetivo é multiplicado pelo respectivo peso atribuído, representando sua importância relativa (Ver Equação 2.6).

$$V_{ij} = F_{ij} \times W_j \quad (2.6)$$

Com os valores ponderados, são calculadas as coordenadas referentes ao ponto ideal positivo ( $V^+$ ) e negativo ( $V^-$ ). Respectivamente, basta selecionar o melhor e o pior valor encontrado para cada objetivo. Depois disso, mede-se a distância euclidiana entre as soluções da FP e os pontos ideais calculados. As Equações 2.7a e 2.7b mostram como isso é realizado.

$$S_{i+} = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^+)^2} \quad (2.7a)$$

$$S_{i-} = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^-)^2} \quad (2.7b)$$

Por fim, para cada solução da FP, calcula-se o índice de proximidade relativa (ver Equação 2.8). Então, a solução mais recomendada é aquela que mais se aproxima de  $V^+$  e, simultaneamente, a mais distante de  $V^-$ , ou seja, que apresenta maior valor associado a  $C_i$ .

$$C_i = \frac{S_{i-}}{S_{i-} + S_{i+}} \quad (2.8)$$

### 2.6.2 Gray Relational Analysis (GRA)

Outro método de seleção apresentado por Wang e Rangaiah (2017), é o GRA. Fundamentado na teoria dos sistemas cinzentos, o método utiliza o *Gray Relational Coefficient* (GRC) para medir a similaridade entre cada solução da fronteira e uma sequência de referência ideal formada pelo melhor valor de cada objetivo após um processo de normalização. Diferente do TOPSIS, o GRA não faz uso de ponderação, o que, talvez, possa reduzir a subjetividade em algumas aplicações. Particularmente, nos casos em que a definição de pesos é difícil ou controversa.

A primeira etapa desse procedimento consiste na Normalização da Matriz de Objetivos. Aplica-se a Equação 2.9a para situações de maximização dos critérios de otimização ou, semelhantemente, a Equação 2.9b para os processos de minimização.

$$F_{ij} = \frac{f_{ij} - \min_{i \in m}(f_{ij})}{\max_{i \in m}(f_{ij}) - \min_{i \in m}(f_{ij})} \quad (2.9a)$$

$$F_{ij} = \frac{\max_{i \in m}(f_{ij}) - f_{ij}}{\max_{i \in m}(f_{ij}) - \min_{i \in m}(f_{ij})} \quad (2.9b)$$

Após a normalização, define-se a sequência de referência ideal através da seguinte expressão:  $F_j^+ = \max_{i \in m}(F_{ij})$ . Então, por meio da equação  $\Delta_{ij} = |F_j^+ - F_{ij}|$ , mede-se a diferença absoluta entre os valores de cada solução e a sequência de referência. Por fim, deve-se calcular para cada solução da fronteira, o seu respectivo valor de GRC (ver Equação 2.10a), sendo aquela com maior valor, a solução recomendada.

$$GRC_i = \frac{1}{m} \sum_{j=1}^n \frac{\Delta_{min} + \Delta_{max}}{\Delta_{ij} + \Delta_{max}} \quad (2.10a)$$

$$\Delta_{max} = \max_{i \in m, j \in n}(\Delta_{ij}) \quad (2.10b)$$

$$\Delta_{min} = \min_{i \in m, j \in n}(\Delta_{ij}) \quad (2.10c)$$

### 2.6.3 Simple Additive Weighting (SAW)

O método SAW é um dos mais intuitivos, um procedimento aditivo simples baseado na soma ponderada dos objetivos previamente normalizados. Assim como no GRA, a normalização é realizada de acordo com o critério de maximização (ver Equação 2.11a) ou minimização (ver Equação 2.11b). Logo após, os objetivos são ponderados de acordo com os seus respectivos pesos através da expressão:  $V_{ij} = F_{ij} \times W_j$ . Por fim, por meio da Equação 2.12, soma-se, para cada solução, o produto entre o valor normalizado e o peso dos objetivos. Então, a solução recomendada será aquela que apresentar maior escore agregado.

$$F_{ij} = \frac{f_{ij}}{f_j^+}, \text{ onde } f_j^+ = \max_{i \in m}(f_{ij}) \quad (2.11a)$$

$$F_{ij} = \frac{f_j^-}{f_{ij}}, \text{ onde } f_j^- = \min_{i \in m}(f_{ij}) \quad (2.11b)$$

$$(2.11c)$$

$$A_i = \sum_{j=1}^n V_{ij} \quad (2.12)$$

A partir da análise dos três métodos, a pesquisa proposta neste trabalho utilizou o método TOPSIS como mecanismo para selecionar soluções nas FP. Pois, enquanto

o método SAW depende fortemente da atribuição de pesos, o que pode introduzir subjetividade, e o GRA ignora essa etapa ao trabalhar apenas com medidas de similaridade, o TOPSIS integra de forma explícita tanto a busca pelo melhor desempenho quanto a necessidade de evitar resultados indesejados, proporcionando uma análise mais robusta para a seleção da solução ótima.

## 2.7 Indicadores de Qualidade

As métricas de performance, mais conhecidas como Indicadores de Qualidade (IQ), são ferramentas essenciais na avaliação dos AG, pois auxiliam a mensurar a qualidade do conjunto final de soluções em termos de convergência, dominância, diversidade, espalhamento, cardinalidade, uniformidade, dentre outros parâmetros (Coello, 2007).

Uma forma manual e direta de avaliar a qualidade de um conjunto de soluções é através da visualização e comparação gráfica, de forma a verificar intuitivamente a relação de um conjunto ou solução em relação a outro. Contudo, para problemas de otimização com dimensionalidade maior do que 3, torna-se necessário recorrer a ferramentas mais complexas encontradas dentro do campo de estudo da análise de dados. Entretanto, essa alternativa pode não ser suficientemente eficiente em refletir de forma clara todos os aspectos de qualidade desejados (Li e Yao, 2019).

Por esta razão, segundo Li e Yao (2019), os IQ podem superar essas dificuldades através de métodos quantitativos que proporcionam medidas sobre as diferenças entre os conjuntos de soluções. Por exemplo, fornecem medições aproximadas a respeito dos aspectos em que um conjunto de soluções é melhor do que o outro.

Embora o uso dos IQ esteja consolidado na literatura, sua aplicação na validação e análise de soluções em estudos sobre posicionamento de sensores tem sido pouco explorada até o momento desta pesquisa. Uma possível explicação para esse fato seja que esse tipo de problema de otimização não possui soluções ideais. Isso restringe o uso de várias métricas como Erro Relativo (Coello, 2007), e métricas de distância e convergência (Li e Yao, 2019), pois dependem da Fronteira de Pareto (FP) Ideal do problema.

Diante disso, durante a realização desta pesquisa, foi decidido aplicar métricas que não necessitam do conjunto de soluções ideais, e também outros dois parâmetros de avaliação. Temos as métricas *Hypervolume (HV)* e *Set Coverage (SC)*, e como parâmetros, o Número de Soluções Não-Dominadas (NSND) encontradas nas Fronteiras de Pareto e o tempo computacional (Time-C) gasto durante a execução do processo evolutivo.

### 2.7.1 Hypervolume (HV)

O indicador *Hypervolume* foi apresentado primeiramente por Zitzler e Thiele (1998), tendo a função de medir a cobertura e o espalhamento de um conjunto de soluções

dentro do espaço de objetivos de um problema de otimização. Isso é feito através do cálculo do volume ocupado, dentro desse espaço, pelas soluções não dominadas em relação a um ponto de referência pré-definido.

Matematicamente, sua definição pode ser vista na Equação 2.13, sendo  $S$  o conjunto de soluções não dominadas,  $R$  um ponto de referência dominado por todos os elementos de  $S$ , e  $v_i$  o hipercubo formado pelo espaço dominado pela solução  $S_i \in S$ , tendo  $R$  como limite. Então, o HV é calculado através da união do volume de cada hipercubo construído dentro do espaço de objetivos. Quanto maior o valor encontrado, melhor é a qualidade da cobertura e espalhamento da Fronteira de Pareto.

$$HV(S, R) = \bigcup_{i=1}^{|S|} v_i \quad (2.13)$$

Um aspecto importante desse indicador é que não é necessário um conjunto de soluções ideais para serem usadas como referência, assim tornando-o adequado para muitos cenários de otimização do mundo real. Entretanto, suas principais limitações estão associadas ao custo computacional que cresce exponencialmente em relação ao número de objetivos, e à definição do ponto ( $R$ ) de referência (Li e Yao, 2019).

Em relação ao custo computacional, quanto maior a quantidade de objetivos a serem otimizados, maior é o custo para calcular o HV. Então, depende dos critérios que se deseja otimizar. Sobre a escolha de  $R$ , algumas abordagens podem ser aplicadas como:

- 1 - Se o intervalo de cada critério de otimização é conhecido, pode-se definir as coordenadas de  $R$  como os piores valores possíveis em cada objetivo, garantindo que  $R$  seja dominado por todas as soluções factíveis do problema (Zitzler e Thiele, 1998).
- 2 - Em cada fronteira gerada, seleciona-se os piores valores encontrados em cada objetivo, acrescenta-se uma margem de 5 a 10%, então usa-os como coordenadas de  $R$ .

Nesta pesquisa, durante o processo de análise das fronteiras, a primeira abordagem foi aplicada, pois no decorrer do estudo verificou-se que os resultados produzidos pelas duas abordagens não apresentaram diferenças significativas.

### 2.7.2 Two Set Coverage (SC)

O indicador *Two Set Coverage* ou apenas *Set Coverage* proposto por Zitzler et al. (2000) possui a função de medir a cobertura ou dominância entre dois conjuntos de soluções. A partir de duas fronteiras  $A$  e  $B$ , essa métrica avalia a taxa de soluções em  $A$  que são dominadas pelas soluções de  $B$ , e vice-versa.

Matematicamente, sua definição pode ser encontrada na Equação 2.14, sendo  $C(A, B)$  a proporção de soluções em  $B$  que são dominadas por pelo menos uma solução em  $A$ , e o símbolo  $\preceq$  indicando a relação de dominância entre duas soluções.

$$C(A, B) = \frac{|\{b \in B \mid \exists a \in A : a \preceq b\}|}{|B|} \quad (2.14)$$

A partir dos valores calculados, com  $C(A, B)$  variando entre 0 e 1, podemos ter os seguintes resultados:

- $C(A, B) = 1$ : significa que todas as soluções em  $B$  são dominadas por alguma solução em  $A$ .
- $C(A, B) = 0$ : significa que nenhuma solução em  $B$  é dominada por uma solução em  $A$ .
- $C(A, B) > C(B, A)$  podemos inferir que a fronteira  $A$  possui soluções melhores ou mais abrangentes do que  $B$ , pois cobre uma maior proporção das soluções em  $B$ .

Em geral,  $C(A, B)$  e  $C(B, A)$  ambos devem ser considerados devido às interseções de conjuntos não serem vazias. Em relação às suas vantagens, esse indicador é fácil de calcular e fornece uma medida de dominância que pode ser utilizada para fins de comparação entre fronteiras produzidas por diferentes algoritmos, ou avaliação do processo evolutivo através da comparação das fronteiras geradas ao longo do aprimoramento das soluções (Coello, 2007).

## 2.8 Trabalhos relacionados

### 2.8.1 Posicionamento de sensores escalares

Ao analisar projetos na literatura, percebe-se que o tópico sobre posicionamento de sensores escalares não é algo novo, sendo possível encontrar trabalhos publicados há mais de uma década, como por exemplo, Wang et al. (2005), Toumpis e Tassiulas (2006), Vincze et al. (2007) e Liao et al. (2011). Com o passar dos anos, as estratégias utilizadas nos estudos de posicionamento desses dispositivos foram modificadas devido à complexidade dos requisitos que muitas aplicações exigem. Atualmente, com o avanço da tecnologia, tem-se observado a utilização de *hardwares* mais rápidos e eficientes, e algoritmos mais complexos, para fins de otimização dessas estratégias. Dentre a categoria dos algoritmos em uso, estão aqueles pertencentes à classe dos algoritmos evolutivos, como os AG.

Das classes dos algoritmos pertencentes à CE, comumente encontram-se trabalhos que aplicam os AG para otimizar o posicionamento de sensores escalares em uma RSSF sobre diferentes aspectos, dentre os principais sendo a maximização da cobertura de área, conectividade entre os sensores e tempo de vida da rede, além da minimização do consumo energético e custos operacionais.

Dentre todos esses aspectos, a maximização da cobertura de área é considerada a mais importante, existindo assim inúmeras pesquisas voltadas a esse problema. Vejamos o trabalho proposto por Binh et al. (2018). O objetivo principal da pesquisa é propor um modelo de otimização que busca maximizar a cobertura de área através da utilização de um número fixo de sensores heterogêneos a serem implantados em uma região pré-definida. Para este fim, os autores propuseram o aprimoramento de dois algoritmos baseados em evolução natural, o *Cuckoo Search* (Yang e Deb, 2009) e o *Flower Pollination* (Yang et al., 2013) de forma a alcançar maior cobertura em um menor tempo computacional em comparação a outros algoritmos existentes.

Também podemos encontrar o trabalho de Hanh et al. (2019), no qual é apresentado o desenvolvimento e a análise de um AG baseado no algoritmo IGA proposto em (Ly et al., 2015). Com esse novo algoritmo, os autores buscavam realizar a maximização da cobertura de área de uma rede com sensores heterogêneos, de forma a alcançar melhores resultados em termos de qualidade das soluções encontradas e o tempo computacional gasto durante o processo de busca.

Já em Birtane Akar et al. (2022), é realizado inicialmente um posicionamento randômico sobre uma área pré-definida, e logo após, a otimização da cobertura de área é feita através de um AG mono-objetivo. De acordo com os autores, a cobertura máxima alcançada nos resultados iniciais foi de 74.85%, sendo possível aumentar com a realização de outros procedimentos de teste. Em outra pesquisa, o mesmo autor, em Birtane et al. (2024) apresenta um trabalho que busca o mesmo objetivo de otimização, e assim como em Birtane Akar et al. (2022), esse projeto buscou utilizar um AG mono-objetivo para maximizar a cobertura, entretanto, considerando a implantação dos sensores em regiões regulares (formato geométrico) e ambientes irregulares, a exemplo do artigo, um campo universitário.

Entretanto, conforme os requisitos das aplicações aumentam em complexidade, torna-se necessário considerar no processo de posicionamento os outros aspectos mencionados anteriormente. Contudo, um dos maiores desafios desse processo está relacionado à definição de soluções que apresentam um equilíbrio entre os diferentes objetivos devido às suas características conflitantes. Por exemplo, ao maximizar a conexão entre os sensores, realiza-se a sobreposição de suas regiões de conectividade, prejudicando a cobertura de área.

Consequentemente, diante das diferentes características a serem consideradas, torna-se inviável encontrar esse balanceamento através dos métodos tradicionais, fazendo com que pesquisadores invistam mais no uso das estratégias evolutivas com abordagem multiobjetivo a fim de encontrar soluções viáveis que se aproximem do cenário ideal desejado.

Nesse contexto, o trabalho proposto por Khalesian e Delavar (2016) apresenta uma abordagem evolucionária multiobjetivo baseada em restrições na FP, visando alcançar a maximização da cobertura de área em conjunto com a minimização do consumo de energia a fim de prolongar o tempo de vida da RSSF. Com o mesmo foco, a pesquisa descrita por El-Sherif et al. (2018) buscou otimizar os mesmos critérios; no

entanto, utilizando uma abordagem distinta por meio do algoritmo NSGA-II. Já a pesquisa de Domingo-Perez et al. (2016) explorou a maximização da cobertura de área com a minimização do número de sensores escalares implantados. Para tanto, os autores utilizaram uma adaptação do algoritmo NSGA-II.

Também é possível encontrar a pesquisa proposta em Benatia et al. (2017), na qual, através do estudo comparativo entre dois algoritmos evolutivos, buscava-se realizar o posicionamento de uma RSSF em ambientes *Indoor* (ver Subseção 2.3.3) de forma a maximizar a cobertura de área, conectividade entre os sensores e minimizar o custo associado ao preço total dos dispositivos. Para isso, os autores analisaram o uso de uma abordagem multiobjetivo baseada na soma de pesos para os critérios de otimização (*Weighted-Sum Multi Objective (WSMO)*) e outra através do algoritmo NSGA-II, sendo a escolha final da melhor estratégia dependente dos requisitos da aplicação.

Na Tabela 2.2, temos em ordem cronológica as pesquisas descritas e também algumas outras, apresentando um levantamento das duas principais características desses trabalhos associados ao posicionamento de sensores escalares.

Observando os objetivos de otimização a partir do levantamento realizado, percebe-se que os principais critérios estudados ao longo dos anos estão associados à cobertura de área, tempo de vida de uma RSSF, consumo energético e número de sensores posicionados. Para isso, diferentes abordagens e estratégias, sendo em grande parte baseadas no uso de algoritmos evolutivos, são utilizadas dependendo do tipo de aplicação e das restrições do problema.

Diante dessas observações, pode-se concluir que já existe na literatura muitos estudos associados aos principais critérios de otimização mencionados; contudo, eles se diversificam em relação às estratégias utilizadas no processo de modelagem e concepção das soluções. Essa diversidade está relacionada ao uso e performance de diferentes algoritmos já consolidados como o NSGA-II, ICS, CFP, dentre outros, mas também no desenvolvimento de novos; na modelagem dos dispositivos (sensores homogêneos e heterogêneos); e na representação da área de implantação, na qual muitas pesquisas, a exemplo Benatia et al. (2017), Wu et al. (2020) e Bouzid et al. (2020), a representam na forma de blocos de área.

Entretanto, antes de mencionar as contribuições desta pesquisa em relação ao cenário sobre posicionamento de sensores, é necessário avaliar o uso dos sensores visuais (câmeras) neste tipo de aplicação. Com isso, na subseção seguinte serão avaliadas algumas pesquisas sobre a otimização do posicionamento de RSSF baseadas no uso de câmeras.

Tabela 2.2: Levantamento de trabalhos sobre posicionamento de sensores escalares.

Pesquisa	Objetivos	Algoritmos
Jourdan e de Weck (2004)	<b>maximizar:</b> cobertura, tempo de vida;	MOGA
Moscibroda et al. (2006)	<b>minimizar:</b> consumo energético, latência;	Algoritmo Uniforme; Algoritmo de <i>Cluster</i> ;
Molina et al. (2008)	<b>minimizar:</b> número de sensores, consumo energético;	MOEA; NSGA-II;
Le Berre et al. (2011)	<b>maximizar:</b> cobertura, tempo de vida; <b>minimizar:</b> número de sensores;	NSGA-II; SPEA-II; MOACO;
Rani e Devarajan (2012)	<b>maximizar:</b> cobertura, tempo de vida e conectividade;	FL híbrido; PSO;
Yetgin et al. (2015)	<b>maximizar:</b> tempo de vida; <b>minimizar:</b> consumo energético	ESA; SOGA;
Khalesian e Delavar (2016)	<b>maximizar:</b> cobertura; <b>minimizar:</b> consumo energético;	CPMEA
Domingo-Perez et al. (2016)	<b>maximizar:</b> cobertura; <b>minimizar:</b> número de sensores;	NSGA-II
Benatia et al. (2017)	<b>maximizar:</b> cobertura, conectividade; <b>minimizar:</b> custo;	WSMO; NSGA-II;
Binh et al. (2018)	<b>maximizar:</b> cobertura;	ICS; CFPA;
El-Sherif et al. (2018)	<b>maximizar:</b> cobertura, tempo de vida;	NSGA-II
Hanh et al. (2019)	<b>maximizar:</b> cobertura;	MIGA
Harizan e Kuila (2020)	<b>maximizar:</b> cobertura, conectividade; <b>minimizar:</b> número de sensores;	NSGA-II
Bouzid et al. (2020)	<b>maximizar:</b> cobertura e conectividade; <b>minimizar:</b> custo;	WSMO

Pesquisa	Objetivos	Algoritmos
Wu et al. (2020)	<b>maximizar:</b> cobertura; <b>minimizar:</b> custo	NSGA-II
Birtane Akar et al. (2022)	<b>maximizar:</b> cobertura;	Mono-objetivo
Birtane et al. (2024)	<b>maximizar:</b> cobertura;	Mono-objetivo

### 2.8.2 Posicionamento de sensores visuais (câmeras)

Assim como são desenvolvidos trabalhos sobre o posicionamento de sensores escalares, a utilização e o posicionamento eficiente de sensores visuais nas RSSF também se constituem como um tema crucial, pois, aplicações modernas, como monitoramento de tráfego, segurança pública e gestão de emergências, dependem de um posicionamento otimizado para garantir a captação eficiente de imagens relevantes (Tian et al., 2024). Além disso, desafios como sobreposição desnecessária nas regiões de cobertura, pontos cegos e restrições de energia tornam o estudo dessa área essencial para o desenvolvimento de soluções inteligentes e sustentáveis.

Ao realizar a busca por pesquisas na literatura, nota-se que esse tópico já vem sendo discutido há mais de uma década. Por exemplo, o trabalho apresentado em Navin et al. (2010) busca propor uma solução baseada no uso de um AG mono-objetivo para resolver o problema de cobertura de área, bem como a redução do efeito de sobreposição em uma RSSF composta por câmeras. Nessa pesquisa, foram utilizadas apenas câmeras homogêneas, tendo o campo de visão desses dispositivos modelado no formato triangular. Segundo os autores, a metodologia proposta apresentou melhores resultados de cobertura, além de ser mais escalável para outras aplicações e número de câmeras em comparação ao método *self-orientation* utilizado no trabalho de Tezcan e Wang (2008). Contudo, nenhum resultado visual é apresentado para demonstrar a cobertura ou posicionamento das câmeras, além de faltar uma análise comparativa detalhada entre os resultados alcançados e a pesquisa feita por Tezcan e Wang (2008).

Também temos a pesquisa proposta por Aghdasi e Abbaspour (2016). A partir de um pré-posicionamento randômico e com distribuição uniforme ao longo de uma área desejada, o objetivo principal desse trabalho foi propor dois algoritmos de agendamento para gerenciar a ativação de um conjunto de câmeras da RSSF, de forma a aumentar a cobertura de área e o tempo de vida da rede, sendo considerada a energia restante nos dispositivos. Para tanto, dois algoritmos foram modelados como algoritmos genéticos que utilizam o método *Adaptive weighted sum* (Kim e de Weck, 2006) para balancear os critérios de otimização. Sempre que um novo agendamento fosse necessário, o AG era executado com o objetivo de encontrar o melhor conjunto de câmeras a serem ativadas, mantendo o menor valor possível para

as funções de *fitness*.

Entretanto, essa modelagem apresentou algumas limitações. Primeiro, devido ao custo de proporcionar uma infraestrutura que permita a rotação das câmeras já implantadas, os autores impuseram a impossibilidade de rotacionar as câmeras. Além disso, não consideraram a eficiência do método proposto utilizando diferentes configurações de câmeras, ou seja, câmeras heterogêneas. Por fim, não é dito quais critérios foram definidos para o processo de escolha dos hiperparâmetros utilizados nos algoritmos genéticos.

Outro estudo voltado ao monitoramento por câmeras é o de Mehdi e Mejdi (2018). Os autores analisaram a eficácia de um conjunto de técnicas tendo por objetivo solucionar o problema de monitoramento de alvos (pontos) com um nível requerido de qualidade em um ambiente 2D, usando o menor número possível de câmeras. O objetivo é determinar as posições das câmeras entre um conjunto de locais candidatos, juntamente com suas orientações. Dentre as técnicas usadas está uma adaptação do método PSO; um algoritmo guloso; um AG mono-objetivo; a proposta de um método exato a partir de uma formulação matemática; e um método antigo chamado *Simulated Annealing* proposto por van Laarhoven e Aarts (1987). É importante mencionar que o nível de qualidade de monitoramento apresentado pelos autores está associado à precisão com que o alvo é detectado a partir da distância entre a câmera. Quanto maior a distância, menor é a qualidade. Assim, um valor mínimo para essa qualidade tornou-se um parâmetro de entrada para os algoritmos.

Segundo os autores, o objetivo foi alcançado, tendo o PSO, em grande parte das instâncias de teste, melhores resultados em termos de tempo computacional e qualidade das soluções. Contudo, observa-se nas ilustrações apresentadas que existem alvos cobertos por mais de uma câmera; entretanto, os autores não analisaram a influência desse aspecto sobre a cobertura e a qualidade de monitoramento requerida.

Diferentemente de Mehdi e Mejdi (2018), o trabalho proposto por Rangel et al. (2019) leva em consideração essa sobreposição. A pesquisa propõe o uso de dois algoritmos evolutivos multiobjetivo, os quais são, um algoritmo com abordagem lexicográfica e o NSGA-II, para maximizar o número de alvos monitorados por ao menos uma câmera, enquanto minimiza a taxa de dispositivos cobrindo o mesmo alvo, ou seja, a redundância. Os resultados foram comparados com alguns algoritmos gulosos tomados como referência, e constatou-se que o algoritmo NSGA-II apresentou custo computacional e tempo de execução muito superiores aos dos algoritmos gulosos e ao lexicográfico. Ademais, o mesmo demonstrou resultados significativamente melhores na otimização dos objetivos.

Recentemente, ainda encontram-se pesquisas sobre o posicionamento de câmeras, pois, com o avanço da Inteligência Artificial e o desenvolvimento de tecnologias de baixo custo voltadas para projetos em IoT, este campo de estudo continua tendo muita importância. Por exemplo, temos o estudo de Faga et al. (2022), no qual busca maximizar aspectos de cobertura de área associados ao posicionamento de

câmeras em ambientes *indoor*. Para isso, os autores usaram o algoritmo NSGA-II para otimizar a localização em que o dispositivo será implantado e seu ângulo de orientação. Aproximadamente um ano depois, os mesmos autores principais desenvolveram a pesquisa proposta em Faga et al. (2023), na qual, também com o uso do NSGA-II, buscaram realizar o monitoramento de alvos em uma região de interesse, maximizando a cobertura de alvos, cobertura redundante e média geral de redundância. Após o estudo, comparações foram realizadas com o trabalho proposto por Rangel et al. (2019) além de alguns algoritmos já existentes.

No mesmo ano, encontra-se o estudo de Suresh e Menon (2023), que busca propor uma abordagem genérica e escalável que permita gerar o posicionamento de câmeras em ambientes *indoor* e *outdoor* com o objetivo de maximizar a cobertura total de área através da menor quantidade possível de câmeras em locais e orientações ideais. Segundo os autores, a proposta se baseia no desenvolvimento de um algoritmo chamado *Reward Penalty Score*, e na extensão de um algoritmo guloso chamado *Greedy Grid Voting*. Essas soluções realizam a otimização do número de câmeras, suas localizações e ângulo de orientação. Contudo, no estudo não foi considerada a sobreposição das áreas de monitoramento das câmeras, e nenhum critério associado à comunicação desses dispositivos com sistemas externos ou entre si por meio de conexões sem fio.

Por fim, na Tabela 2.3, temos a sintetização de mais alguns trabalhos relacionados ao posicionamento de câmeras.

Tabela 2.3: Outras pesquisas sobre posicionamento de câmeras.

<b>Trabalho</b>	<b>Critérios de Otimização</b>	<b>Descrição</b>
Liu et al. (2008)	Cobertura e Sobreposição	Associado ao tema de detecção de faces humanas, o artigo propõe uma modelagem matemática para o problema de cobertura (K-coverage) na qual cada ponto de observação deve ser monitorado simultaneamente por K câmeras, tendo por objetivo aprimorar a detecção de rostos.
Alaei e Barcelo-Ordinas (2010)	Conservação de energia e Redundância	Através do uso de clusters, é proposto um método para conservação de energia e redução dos níveis de monitoramento redundante com o objetivo de evitar processamento desnecessário e garantir equilíbrio para a taxa de energia restante em cada dispositivo.

<b>Trabalho</b>	<b>Critérios de Otimização</b>	<b>Descrição</b>
Costa et al. (2014)	Redundância	É proposto um algoritmo guloso como solução para o problema de maximização da cobertura redundante em alvos com níveis de relevância distintos. A partir do uso de câmeras, a redundância no monitoramento deve priorizar alvos com níveis mais altos de relevância.
Rebai et al. (2016)	Cobertura e Minimização do custo	A pesquisa busca propor 3 métodos exatos de otimização para abordar o problema de cobertura e custo de implantação de uma rede de câmeras. A cobertura está associada ao monitoramento - em ambientes 3D - de alvos com diferentes níveis de importância.
Zhang e Zhang (2020)	Cobertura e Redundância	O artigo propõe um novo algoritmo evolutivo multi-espécies para otimizar a cobertura de alvos em redes de sensores visuais com campos de visão ajustáveis. Três critérios conflitantes foram considerados: a taxa de cobertura; a redundância mínima, que busca garantir que cada alvo seja monitorado por pelo menos dois sensores; e a redundância média, que avalia a quantidade de sensores monitorando simultaneamente um mesmo alvo.
Zhu e Zhou (2024)	Minimização do custo de implantação	A pesquisa propõe dois métodos heurísticos que visa, a partir de uma rede de câmeras heterogêneas que realiza a cobertura de alvos e extração de características de imagens capturadas, garantir o menor custo de implantação de todos os dispositivos implantados, satisfazendo os requisitos de tempo de vida da rede.

### 2.8.3 Considerações

Após a análise das pesquisas descritas nas Seções 2.8.1 e 2.8.2, tendo como base trabalhos antigos e outros mais recentes, foi possível observar algumas características importantes. Percebe-se que tanto no posicionamento de sensores escalares quanto em câmeras, existem alguns critérios de otimização que podem ser considerados como principais nesses estudos, que são: cobertura, redundância ou sobreposição nas regiões de monitoramento, custo de implantação, além do consumo de energia. Como pode ser visto nas pesquisas apresentadas, esses critérios são abordados em diferentes estudos, dependendo da necessidade da aplicação ou lacuna de pesquisa encontrada.

No geral, temos que o aspecto de cobertura sempre é maximizado e está em grande parte das pesquisas por ser a característica mais importante em uma RSSF, tendo por objetivo aumentar a eficácia do monitoramento da rede. O aprimoramento da redundância ou sobreposição é aplicado em algumas pesquisas visando garantir um nível de robustez para a rede em caso de falhas de sensores. O consumo energético também é outro importante critério de otimização pois busca prolongar o tempo de vida/funcionamento dos dispositivos da rede. E por fim, temos o custo de implantação. Mesmo com a diminuição dos preços associados aos dispositivos de monitoramento, implantar e manter uma rede com um número elevado de elementos ainda é custoso. Por essa razão, algumas pesquisas buscam minimizar os custos através da busca por soluções que utilizem o menor número possível desses dispositivos.

Em relação aos métodos e estratégias utilizadas durante o processo de otimização, percebe-se uma grande variedade, sem ser possível definir um padrão metodológico para o estudo de posicionamento. Nessa pluralidade, o uso de abordagens heurísticas como os algoritmos evolutivos (mono ou multiobjetivo), além de algoritmos gulosos, destacam-se em muitos estudos.

Diante dessas observações, esta pesquisa busca propor contribuições em relação ao cenário geral encontrado através da análise dos diversos trabalhos na literatura. Primeiramente, esta pesquisa propõe uma solução que permite realizar o posicionamento de sensores escalares e visuais simultaneamente. Até o presente momento, não foram encontrados trabalhos que permitissem esse tipo de posicionamento. Como visto, normalmente, as aplicações são realizadas de maneira separada.

Outro diferencial está no uso de câmeras para cobertura de área, pois, diversos trabalhos como Liu et al. (2008), Costa et al. (2014), Costa et al. (2015), Mehdi e Mejdi (2018), Rangel et al. (2019), Faga et al. (2023), Zhu e Zhou (2024) e outros, abordam apenas o monitoramento de alvos. O aspecto de cobertura de área associado ao uso de câmeras pode ser de extrema importância para o monitoramento em áreas urbanas onde há grande circulação de pessoas e necessidade de segurança, como centros comerciais, avenidas e ruas, praças, parques, áreas esportivas, dentre outras.

E, por fim, considerando que os mesmos critérios de otimização vêm sendo estuda-

dos há duas décadas tanto em estudos sobre sensores escalares quanto em câmeras, esta pesquisa busca avaliar três critérios, sendo dois deles ainda não encontrados em pesquisas prévias. O primeiro será a cobertura de área. Mesmo sendo um aspecto amplamente estudado, é crucial que seja mantido, tendo em vista o objetivo principal das RSSF, que consiste basicamente em monitorar o máximo de área possível. Em seguida, apresenta-se o critério associado ao posicionamento de sensores em regiões de alta capacidade de conectividade. Isso é possível mediante o mapeamento da região de interesse, no qual estão distribuídas as redes sem fio. Por último, realiza-se a avaliação da qualidade de sensoriamento nas regiões monitoradas pelos dispositivos. Esses novos objetivos de otimização, em conjunto com a modelagem do processo de posicionamento proposto, serão devidamente descritos no Capítulo 3.

# **Capítulo 3**

## **Solução Proposta**

O desenvolvimento de uma solução eficiente para o problema de posicionamento das RSSF exige a definição de uma metodologia capaz de articular os fundamentos teóricos discutidos anteriormente com estratégias práticas de implementação. Considerando a natureza combinatória do problema e a necessidade de conciliar múltiplos objetivos, torna-se imprescindível adotar um processo sistemático que oriente desde a definição da rede de sensores até a aplicação do método de otimização selecionado. Nesse sentido, este capítulo apresenta a metodologia concebida para estruturar e conduzir a pesquisa, descrevendo as etapas que permitem a construção e validação da solução proposta.

A metodologia adotada fundamenta-se na utilização do algoritmo NSGA-II (Deb et al., 2002) como mecanismo central de otimização, dada a sua contínua aplicação em trabalhos da literatura, evidenciando assim, sua eficácia nos estudos de posicionamento. Diante disso, neste capítulo, são apresentados os critérios de mapeamento das regiões de interesse, a modelagem da rede de sensores, definição das funções de avaliação usadas para medir a qualidade das soluções produzidas, bem como o detalhamento da estratégia de posicionamento proposta neste trabalho. Dessa forma, busca-se evidenciar a estrutura lógica da pesquisa, assegurando transparência metodológica e fornecendo as bases necessárias para a análise crítica dos experimentos conduzidos.

### **3.1 Modelagem da Região de Interesse (RdI)**

Como mencionado no Capítulo 1, este estudo tem como foco a otimização do posicionamento de sensores escalares e visuais em grandes áreas abertas que disponham de acesso a redes de comunicação sem fio. Por meio dessas redes, os dispositivos de monitoramento podem transmitir, para outros dispositivos ou sistemas externos, dados relacionados às variáveis monitoradas e/ou eventos detectados, além de receber informações de configuração e controle. Para viabilizar a otimização do posicionamento desses dispositivos, é essencial conhecer previamente a localização, a

distribuição e a cobertura geográfica dessas redes, de forma a definir a estratégia mais adequada para o processo.

Neste contexto, assume-se como premissa que uma RdI é de fácil acesso, possibilitando a implantação dos sensores quando necessário. Essas regiões serão consideradas essencialmente como áreas urbanas, tais como as Cidades Inteligentes. Regiões de difícil acesso, contudo, não estão contempladas no escopo desta pesquisa.

Com o intuito de levantar as restrições e características associadas ao ambiente de monitoramento, torna-se indispensável realizar um mapeamento detalhado da RdI, a fim de extrair e aplicar todas as informações relevantes no desenvolvimento da estratégia de posicionamento. Para tal, este estudo se baseia no mapeamento proposto por Jesus et al. (2023), em que os autores apresentam um método analítico para mensurar a qualidade da conectividade das redes sem fio implantadas em uma área específica — no caso do estudo mencionado, as Cidades Inteligentes. Esse método avalia conjuntamente os parâmetros de Custo, *Throughput*, Segurança e Confiabilidade para cada rede presente na região. Além disso, o trabalho introduz uma estratégia de mapeamento da área de interesse que pode ser utilizada para os objetivos desta pesquisa.

### 3.1.1 Processo de mapeamento

Como discutido na Seção 2.8, diversos estudos realizam o mapeamento de regiões de interesse por meio da subdivisão da área total em pequenas células de tamanho fixo, geralmente em formato quadrado ou retangular. Exemplos dessa abordagem podem ser encontrados em trabalhos como os de Benatia et al. (2017), Bouzid et al. (2020) e Jesus et al. (2023). Essa metodologia permite representar tais regiões como a união de células disjuntas, que neste estudo serão denominadas Blocos de Área (*Area Blocks* - AB), com altura ( $hs_{AB}$ ) e largura ( $ws_{AB}$ ) fixas.

De maneira semelhante aos trabalhos relacionados, podemos representar as áreas selecionadas através de uma aproximação em um sistema de coordenadas X e Y. Essa representação facilita a modelagem da estratégia de posicionamento, o cálculo da localização de elementos dentro da RdI e reduz o custo computacional associado à manipulação de um sistema tridimensional.

A Figura 3.1 ilustra a representação de uma área genérica mapeada por essa abordagem. Toda RdI é delimitada pelo retângulo externo, sendo subdividida em ABs com dimensões  $ws_{AB} \times hs_{AB}$ . Com essa delimitação e utilizando a estratégia proposta por Jesus et al. (2023), pode-se definir um conjunto de redes sem fio  $NT = \{net_1, net_2, \dots, net_n\}$  presentes na RdI, cada uma caracterizada pelos seguintes parâmetros:

- $(X_{AP_j}, Y_{AP_j})$  - coordenadas X e Y do ponto de acesso (*access point*) da j-ésima rede;
- $R_j$  - raio de cobertura da j-ésima rede;

- $T_j$  - tipo da j-ésima rede (e.g., WiFi, 4G, LoRa, entre outros).

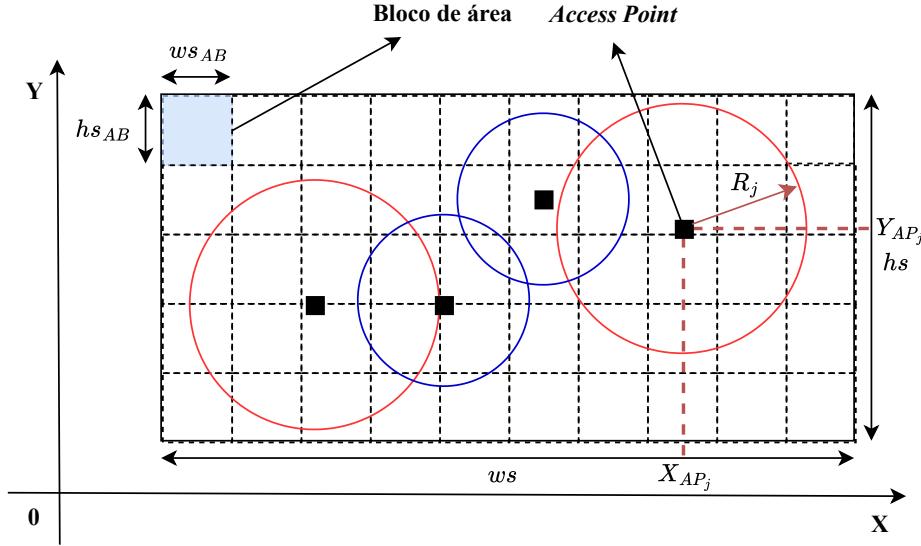


Figura 3.1: Representação da Região de Interesse.

Com essas informações, o i-ésimo AB é considerado coberto pela j-ésima rede se o ponto central do AB estiver dentro do alcance definido por  $R_j$ . Formalmente, a distância euclidiana entre as coordenadas do ponto de acesso da rede e o ponto central do  $AB_i$  deve ser menor que  $R_j$ , conforme descrito na Equação 3.1.

$$cov_{AB}(AB_i, net_j) = \begin{cases} 1, & d(AB_i, netAP_j) \leq R_j \\ 0, & c.c \end{cases} \quad (3.1)$$

É importante destacar que um AB pode ser coberto por mais de uma rede simultaneamente, o que aumenta a conectividade na área e possibilita uma comunicação mais robusta (Jesus et al., 2023).

Uma vez posicionadas as redes, Jesus et al. (2023) propõem uma métrica chamada *Dependable-Quality Connectivity (DPCConn)* para avaliar a qualidade da conectividade em cada AB. Essa métrica permite verificar se a disposição das redes atende aos requisitos relacionados ao tráfego de dados, integridade, autenticidade e tolerância a falhas dos dispositivos de monitoramento. A definição formal da variável *DPCConn* é apresentada na Equação 3.2.

$$DPCConn(AB_i) = \frac{\sum_{j=1}^n [cov_{AB}(AB_i, net_j) \cdot (S.s_j + T.t_j + R.r_j - C.c_j)]}{\sum_{j=1}^{|type|} [S.s_j + T.t_j + R.r_j - C.c_j]} \quad (3.2)$$

As seguintes especificações estão relacionadas à Equação 3.2 (Jesus et al., 2023):

- $n$  - número de redes implantadas;
- $|type|$  - número de diferentes tipos de redes;
- $cov_{AB}(AB_i, net_j)$  - Veja a Equação 3.1
- $S$  - peso associado ao parâmetro de nível de Segurança;
- $T$  - peso associado ao parâmetro de *Throughput*;
- $R$  - peso associado ao parâmetro de Confiabilidade;
- $C$  - peso associado ao parâmetro de Custo;
- $s_j, t_j, r_j, c_j$  - indicam os níveis de Segurança, *Throughput*, Confiabilidade e Custo, respectivamente, para a  $j$ -ésima rede;

Com base na Equação 3.2, Jesus et al. (2023) define intervalos para a variável  $DPCon$ , permitindo classificar três níveis de conectividade:  $\{1, 2, 3\}$ , que correspondem, respectivamente, aos níveis Baixo, Médio e Alto. Esses intervalos estão detalhados na Equação 3.3. Por ser uma definição simples, intuitiva e que atende aos objetivos deste estudo, decidiu-se que, para esta pesquisa, a classificação inicial seria mantida. Entretanto, diferentes intervalos podem ser classificados, gerando novos níveis de conectividade a depender das necessidades da aplicação.

$$\begin{aligned} connLevelRange(1) &= [0, 0.33] = \text{nível baixo} \\ connLevelRange(2) &= [0.33, 0.66] = \text{nível médio} \\ connLevelRange(3) &= [0.66, +\infty] = \text{nível alto} \end{aligned} \quad (3.3)$$

A Figura 3.2 apresenta um exemplo gerado através do *software* Matlab, no qual redes representadas pelas áreas em círculo foram implantadas aleatoriamente. Esse cenário possui dimensões de  $4000 \times 4000$  metros, subdividida em ABs de  $15 \times 15$  metros, totalizando 71.289 ABs. As áreas em vermelho, amarelo e verde representam, respectivamente, regiões com níveis de conectividade baixo, médio e alto, enquanto áreas em branco correspondem a regiões sem acesso a nenhuma rede. Essa representação permite uma análise precisa da localização das redes implantadas e dos locais mais adequados para a transferência de dados por comunicação sem fio, considerando os níveis de conectividade.

Observa-se também que, no exemplo da Figura 3.2, existem poucas áreas com alta conectividade, enquanto as regiões com nível baixo se destacam por predominar sobre grande parte da área coberta. Isso se deve à sobreposição das regiões de cobertura das redes. Nota-se que todas as áreas com média ou alta conectividade estão sujeitas a mais de uma rede, levando-nos a perceber que a sobreposição das regiões de cobertura beneficia o aumento da conectividade, pois garante que os dispositivos implantados nessas áreas tenham diferentes caminhos para trocas de mensagens, possibilitando mais segurança e robustez para as tarefas de comunicação.

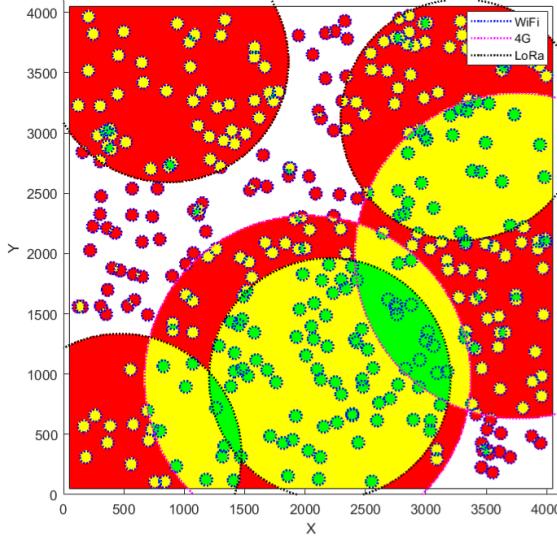


Figura 3.2: Mapeamento das redes e níveis de conectividade para uma Região de Interesse.

Na geração do cenário da Figura 3.2, foram utilizados os dados das Tabelas 3.1, 3.2 e 3.3. A Tabela 3.1 apresenta os pesos individuais atribuídos aos parâmetros de cada tipo de rede. Na Tabela 3.2, os pesos gerais indicam o impacto relativo de cada parâmetro no cálculo do *DPCConn*, sendo todos iguais, essa característica reflete igual relevância entre os parâmetros. Por fim, foram implantadas 350 redes WiFi, 2 redes 4G e 4 redes LoRa, cujos raios de alcance estão descritos na Tabela 3.3. Para uma descrição detalhada do cálculo do *DPCConn*, incluindo implementações, simulações e resultados, recomenda-se consultar (Jesus et al., 2023).

Tabela 3.1: Parâmetros individuais de rede para o cálculo da métrica *DPCConn*.

<b>Rede</b>	<b>Segurança (s)</b>	<b>Throughput (t)</b>	<b>Confiabilidade (r)</b>	<b>Custo (c)</b>
<b>WiFi</b>	2	4	2	1
<b>4G</b>	3	3	3	3
<b>LoRa</b>	2	1	3	1

Tabela 3.2: Parâmetros gerais de rede para o cálculo da métrica *DPCConn*.

<b>Segurança (S)</b>	<b>Throughput (T)</b>	<b>Confiabilidade (R)</b>	<b>Custo (C)</b>
1	1	1	1

Ao longo desta pesquisa, será adotado o método de mapeamento baseado na métrica *DPCConn*, uma vez que auxilia na modelagem computacional da RdI, articulando sua estrutura de rede e os diferentes níveis de conectividade. Além disso, essa métrica

Tabela 3.3: Raio de alcance das redes Wifi, 4G e LoRa.

<b>Rede</b>	<b>Raio</b>
<b>WiFi</b>	<i>50m</i>
<b>4G</b>	<i>1350m</i>
<b>LoRa</b>	<i>1000m</i>

permite identificar de forma simples e precisa as regiões com disponibilidade de conexão, possibilitando o posicionamento de sensores e câmeras apenas em áreas onde a transmissão de dados e a comunicação entre dispositivos sejam efetivamente viáveis.

### 3.1.2 Representação das regiões de interesse em formato JSON

A visualização das RdI por meio de exemplos gráficos, como ilustrado na Figura 3.2, constitui uma abordagem eficaz para analisar aspectos fundamentais relacionados às redes, incluindo localização, cobertura, sobreposição das áreas de alcance e o nível de conectividade em diferentes regiões. Contudo, é necessário armazenar informações específicas que serão utilizadas no processo de posicionamento dos dispositivos. A seguir, será apresentado o método proposto para essa finalidade.

Conforme observado na Figura 3.1, uma RdI, independentemente de ser representada em formato quadrado ou retangular, pode ser modelada computacionalmente como uma matriz bidimensional, na qual cada posição corresponde a um AB. Essa representação permite que cada posição armazene informações diversas, como a presença ou ausência de cobertura por rede ou sensor, bem como o nível de conectividade associado ao AB.

Com base nessa abordagem, suponha que o exemplo da Figura 3.1 possua  $ws_{AB} = 10m$ ,  $hs_{AB} = 20m$ ,  $w_s = h_s = 100m$ , e também tenha sido realizado o cálculo da métrica  $DPC_{Conn}$  para cada AB coberto por alguma rede. Isso proporciona a ilustração vista na Figura 3.3. Devido às dimensões estabelecidas, a RdI está sendo representada como uma matriz de  $5 \times 10$  posições. Ao término do processo de geração da RdI, todos os ABs que possuem algum nível de conectividade são mapeados em duas estruturas unidimensionais denominadas PC (*Posições de Conectividade*) e NC (*Níveis de Conectividade*).

A estrutura PC armazena as posições de todos os ABs com conectividade, enquanto a estrutura NC registra o respectivo nível de conectividade de cada posição. Os ABs sem conectividade não são incluídos nessas estruturas. Dessa maneira, esse mapeamento proporciona acesso rápido aos dados de conectividade, aumentando a eficiência das consultas realizadas durante o processo de posicionamento.

Após o mapeamento, um arquivo no formato JSON (*JavaScript Object Notation*) é gerado, como exemplificado na Listagem 1. Esse arquivo contém informações

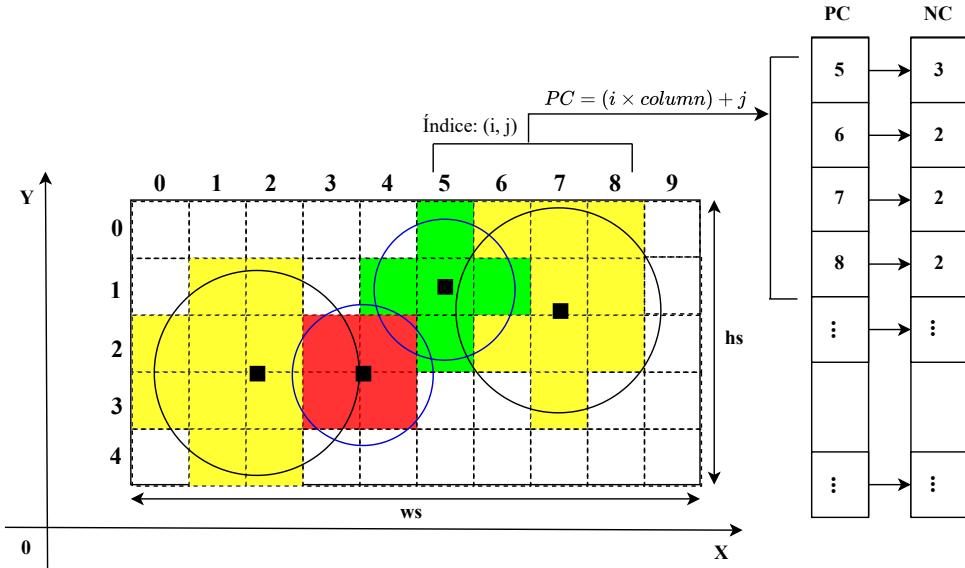


Figura 3.3: Mapeamento dos ABs com conectividade.

detalhadas, como as dimensões de largura e altura dos ABs e da RdI, a quantidade de ABs com conectividade ( $nb_C$ ), bem como o conteúdo das estruturas PC e NC.

Com essa modelagem, todas as informações necessárias sobre a RdI estão devidamente organizadas e acessíveis para utilização durante as próximas etapas deste estudo.

```

1   {
2       "city": {
3           "ws_AB": 10,
4           "hs_AB": 20,
5           "ws_city": 100,
6           "hs_city": 100,
7           "nb_C": 28
8       },
9       "positions": [5,6,7,8,11,12,14,15,16,17,18,20,21,22,23,24,25,26,
10      27,28,30,31,32,33,34,37,41,42],
11      "connectivity": [3,2,2,2,2,2,3,3,3,2,2,2,2,2,1,1,3,2,2,2,2,2,3,
12      3,2,2,2]
13  }

```

Listing 1: Arquivo JSON relacionado às informações do mapeamento da RdI ilustrada na Figura 3.3.

## 3.2 Representação dos dispositivos de monitoramento

Com a finalização do processo de modelagem das RdIs, torna-se necessário definir um modelo representativo para os dispositivos de monitoramento, que, neste caso, correspondem aos sensores escalares e sensores visuais (câmeras). Para isso, dependendo da aplicação, é imprescindível considerar informações como o tipo do dispositivo, raio de sensoriamento, raio de comunicação, modelo de sensoriamento e quantidade, de forma a estabelecer uma representação adequada para o conjunto de dispositivos que serão posicionados ao longo de uma RdI.

Nos estudos expostos na Seção 2.8, como em (Benatia et al., 2017), (Binh et al., 2018), (Wu et al., 2020) e (Birtane Akar et al., 2022), são adotadas formas semelhantes de representação das características desses dispositivos. Nos trabalhos propostos por Binh et al. (2018) e Birtane Akar et al. (2022), todos os sensores posicionados são homogêneos, ou seja, possuem as mesmas características, como tipo e raio de sensoriamento; e o modelo de sensoriamento é representado através do modelo binário de detecção (ver Subseção 2.2.1).

Por outro lado, em estudos como o de Harizan e Kuila (2020), o raio de comunicação também é considerado no processo de modelagem. Usado como um parâmetro essencial para definir a conectividade entre os sensores, estabelece um limiar no qual um dispositivo pode trocar informações com outros dispositivos próximos. Já em (Bouzid et al., 2020), os autores modelam esse parâmetro de conectividade analisando a topologia da rede de sensores e o tipo de protocolo de comunicação empregado.

Em contrapartida, Rangel et al. (2019) aborda o posicionamento de câmeras, buscando otimizar a cobertura total e redundante de pontos específicos. Nesse estudo, as áreas de monitoramento das câmeras são representadas como áreas triangulares, cuja orientação pode ser ajustada para melhorar a cobertura.

Observa-se, portanto, que, tanto para sensores escalares quanto para sensores visuais, há padrões amplamente utilizados por pesquisadores em seus projetos. Contudo, esses modelos podem ser adaptados para atender às necessidades específicas de cada estudo ou aplicação. A seguir, serão apresentados os modelos e adaptações adotados neste trabalho para representar os dispositivos de monitoramento.

### 3.2.1 Modelo de sensoriamento

Inicialmente, com relação ao modelo de detecção aplicado aos dispositivos, o Modelo Binário (ver Subseção 2.2.1) foi escolhido por ser o mais simples e de baixa complexidade; contudo, tal padrão não se aproxima com exatidão do monitoramento real de um sensor. Então, ao analisar o Modelo Probabilístico *Elfes Sensing* (ver Subseção 2.2.2), foi visto que esse padrão é considerado superior ao Binário, pois modela com mais precisão o comportamento real de um sensor, além de possuir mais flexibilidade, permitindo simular diferentes curvas que emulam a precisão com a qual um sensor pode detectar um evento a partir do aumento ou diminuição da distância.

É importante informar que o modelo *Shadow-Fading Sensing Model* (ver Subseção 2.2.3) não foi considerado para este estudo pois a inclusão de obstáculos e suas consequências na modelagem da RdI e no processo de posicionamento não fazem parte do escopo desta pesquisa.

Logo, a partir dessas observações, este estudo propõe a utilização de uma modelagem baseada nos padrões Binário e Probabilístico para representar a área de sensoriamento dos sensores escalares e das câmeras. Formalmente, temos sua definição através da Equação 3.4a. Comparado ao comportamento do modelo probabilístico original, em vez de termos um decaimento exponencial a partir de um limiar estabelecido, temos um decaimento linear (ver Figura 3.4(a)), se aproximando do comportamento dos sensores lineares. Desta forma, quanto maior a distância euclidiana entre um sensor e um ponto específico (ver Equação 3.4b), menor será a precisão na detecção de um evento ou no monitoramento de uma variável do ambiente. Ao se aproximar do alcance máximo ( $r_s$ ) do sensor, essa precisão estará perto de zero.

$$DP(sr_i, p) = \begin{cases} 1, & d(sr_i, p) \leq r_l \\ (r_s - d)/(r_s - r_l), & r_l < d(sr_i, p) \leq r_s \\ 0, & c.c \end{cases} \quad (3.4a)$$

$$d(sr_i, p) = \sqrt{(p_x - sr_x)^2 + (p_y - sr_y)^2} \quad (3.4b)$$

Com base no alcance  $r_s$  do dispositivo, a área sob a curva da Figura 3.4(a) foi dividida em três subáreas definidas como Regiões de Qualidade de Sensoriamento ( $Q_S$ ), sendo elas: qualidade Alta, Média e Baixa, sendo representadas, respectivamente, pelos valores  $\{1, 2, 3\}$ . A divisão dos intervalos pode ser compreendida através da Figura 3.4(b) e pelas Equações 3.5a e 3.5b.

$$Q_S(sr_i, p) = \begin{cases} 1, & d(sr_i, p) \leq r_l \\ 2, & r_l < d(sr_i, p) \leq r_e \\ 3, & r_e < d(sr_i, p) \leq r_s \\ 0, & c.c \end{cases} \quad (3.5a)$$

$$r_e = r_l + \frac{(r_s - r_l)}{2} \quad (3.5b)$$

Essa representação, que caracteriza a precisão ou qualidade com que um dispositivo pode medir ou detectar um evento, pode ser aplicada não apenas a sensores escalares, mas também às câmeras atuando como sensores visuais. Essa qualidade está

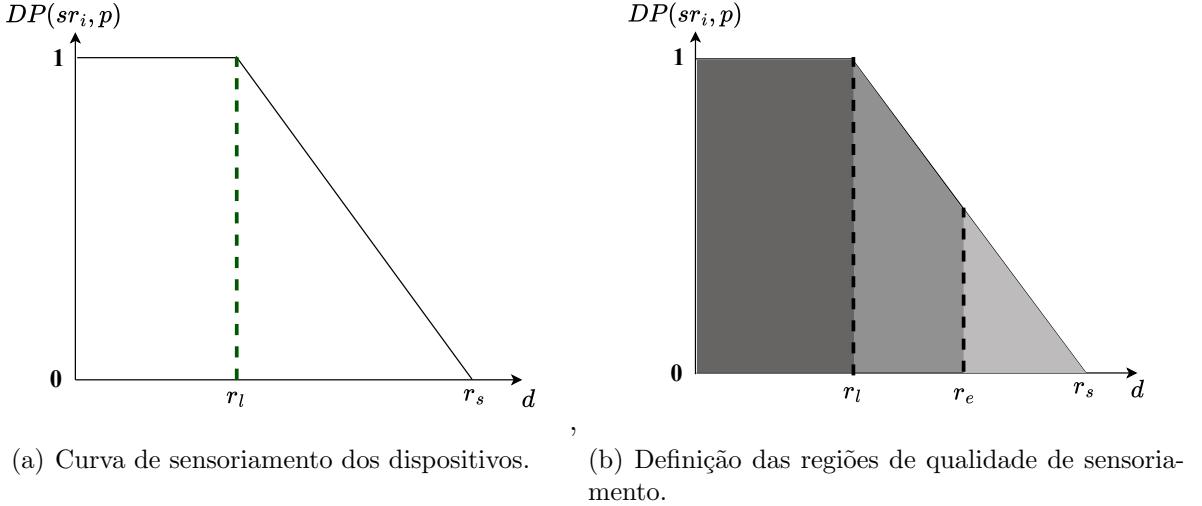


Figura 3.4: Representação do modelo de sensoriamento.

diretamente relacionada ao nível de nitidez com que a câmera é capaz de capturar um objeto. Nesse contexto, Jesus et al. (2022) afirma que a maneira mais simples de avaliar esse parâmetro de nitidez é considerar que, de maneira geral, quanto mais próximo o objeto estiver da câmera, mais nítida será a imagem capturada; por outro lado, à medida que a distância aumenta, a nitidez do objeto diminui.

Por essa razão, Jesus et al. (2022) observa que, na literatura, o campo de visão de uma câmera é frequentemente dividido em três regiões de qualidade (alta, média e baixa). Assim, ao considerar um raio máximo de visão para a câmera, essa divisão é análoga ao modelo de Qualidade de Sensoriamento apresentado na Figura 3.4(b).

Cabe ressaltar que a modelagem de sensoriamento e cobertura das câmeras adotada neste estudo restringe-se à análise bidimensional do campo de visão, não incluindo a avaliação da profundidade das imagens capturadas.

### 3.2.2 Região de cobertura

Nos estudos relacionados ao posicionamento de sensores, os pesquisadores frequentemente buscam modelar formas de representar a área de cobertura desses dispositivos de forma a simular sua implantação em coordenadas específicas. Por exemplo, em trabalhos como os de Wu et al. (2020) e Bouzid et al. (2020), assim como nesta pesquisa, as RdIs são representadas em formato de matrizes bidimensionais, permitindo o cálculo de métricas como a taxa de cobertura dos sensores a partir das áreas contidas dentro de seus respectivos raios de sensoriamento.

Com base na análise desses trabalhos, este estudo propõe o uso de estruturas chamadas de Máscaras de Cobertura para sensores escalares e câmeras, com o objetivo de criar uma aproximação da representação da área de cobertura desses dispositivos de

acordo com a configuração dos AB das RdI. Foram definidos dois tipos de máscaras: a Máscara de Sensoriamento ( $MS$ ), que delimita a área coberta por um sensor, e a Máscara de Qualidade de Sensoriamento ( $MQS$ ), que define os diferentes níveis de qualidade dentro da área coberta. No entanto, antes de detalhar essas máscaras, é necessário analisar a região de cobertura associada a cada tipo de dispositivo.

Como ilustrado na Figura 3.5(a), a região de cobertura de um sensor escalar apresenta uma estrutura simples e intuitiva, uma vez que as delimitações dos níveis de qualidade de sensoriamento são definidas exclusivamente pelo raio de sensoriamento ( $r_s$ ) e as variáveis  $r_l$  e  $r_e$ . Em contrapartida, a modelagem da região de cobertura para as câmeras é mais complexa.

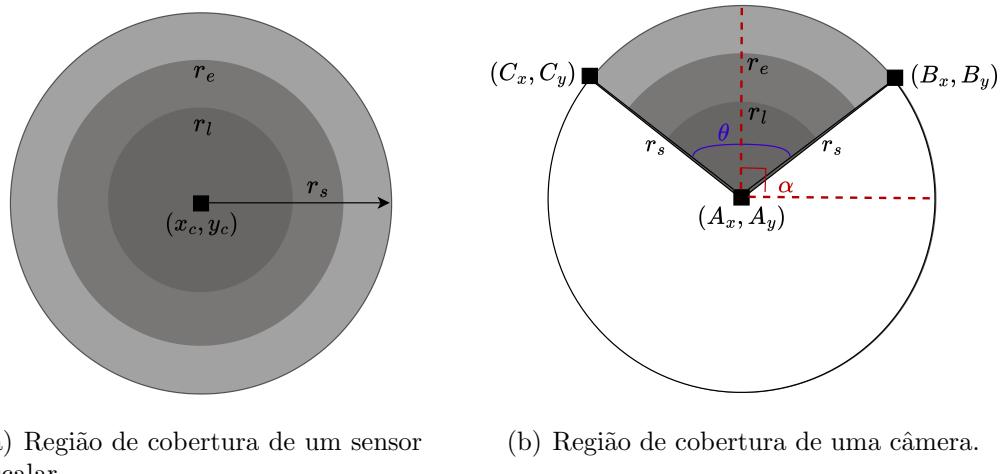


Figura 3.5: Representação da região de cobertura dos sensores escalares e câmeras baseados no modelo de sensoriamento proposto.

Na Figura 3.5(b), observa-se que, devido à geometria do campo de visão das câmeras, a modelagem inclui as variáveis  $\theta$  e  $\alpha$ , que representam, respectivamente, em graus, o ângulo de abertura do campo de visão da câmera e sua orientação. Essas variáveis estão definidas nos intervalos  $0^\circ < \theta \leq 180^\circ$  e  $0^\circ \leq \alpha \leq 360^\circ$ . Essa parametrização permite que a câmera seja configurada com diferentes campos de visão, além de possibilitar ajustes na sua orientação para cobrir pontos específicos ou outras áreas de interesse.

Adicionalmente, são consideradas as coordenadas do ponto  $A$ , que corresponde à localização da câmera, e dos pontos  $B$  e  $C$ , que representam os vértices utilizados no cálculo da delimitação da região de cobertura. A formalização desses pontos está detalhada na Equação 3.6c a Equação 3.6h.

$$\alpha_1 = (\alpha - \frac{\theta}{2}).mod2\pi \quad (3.6a)$$

$$\alpha_2 = (\alpha + \frac{\theta}{2}).mod2\pi \quad (3.6b)$$

$$A_x = sr_x; \quad (3.6c)$$

$$A_y = sr_y; \quad (3.6d)$$

$$B_x = A_x + \cos(\alpha_1).r_s \quad (3.6e)$$

$$B_y = A_y + \sin(\alpha_1).r_s \quad (3.6f)$$

$$C_x = A_x - \cos(\alpha_2).r_s \quad (3.6g)$$

$$C_y = A_y + \sin(\alpha_2).r_s \quad (3.6h)$$

Com base na descrição dessas regiões e no modelo de sensoriamento apresentado na Subseção 3.2.1, foi desenvolvida uma estratégia para o cálculo e a geração das máscaras de cobertura correspondentes.

### 3.2.3 Máscaras de cobertura

As máscaras de cobertura são fundamentais durante o processo de posicionamento para identificar todos os ABs monitorados por um sensor escalar ou câmera, além de indicar o nível de qualidade de sensoriamento atribuído a cada local. Essas estruturas são representadas como uma matriz bidimensional de dimensões  $(r_s/hs_{AB}) \times (r_s/ws_{AB})$ , onde cada posição da matriz corresponde a uma área equivalente às dimensões de um AB da RdI.

Na Figura 3.6, ilustra-se a *MS* para os dois tipos de dispositivos utilizados. Cada posição da matriz é preenchida com os valores 0 ou 1, indicando, respectivamente, a ausência ou presença de monitoramento. De forma similar, a Figura 3.7 apresenta a *MQS*, onde as posições são preenchidas com os níveis de qualidade de sensoriamento: alto - 1, médio - 2 ou baixo - 3. Vale ressaltar que a definição das regiões de qualidade, bem como as dimensões das máscaras, depende diretamente dos limiares  $r_s$ ,  $r_l$  e  $r_e$  do modelo de sensoriamento e do tamanho configurado para os AB da RdI.

Note nas Figuras 3.6 e 3.7 que existem posições das máscaras que não são completamente monitoradas pela região de cobertura, o que pode ser explicado por dois fatores principais. Primeiro, como cada posição representa um bloco com as mesmas dimensões de um AB, considera-se que um AB está sendo monitorado se a coordenada do seu ponto central estiver dentro do limite de  $r_s$  (ver Equação 3.7), mesmo que a área total do bloco não esteja completamente contida no raio de alcance do sensor.

$$covBySensor_{AB}(AB_i, sr_j) = \begin{cases} 1, & d(p_{AB_i}, sr_j) \leq r_{s_j} \\ 0, & c.c \end{cases} \quad (3.7)$$

Segundo, as máscaras são aproximações das regiões de cobertura representadas na Figura 3.5. Dessa forma, as dimensões de um AB impactam diretamente na precisão dessa representação. Quanto menor for a dimensão de um AB, maior será a precisão no mapeamento tanto das regiões de cobertura dos sensores quanto das RdI. No entanto, esse aumento na precisão implica em um custo computacional mais elevado, devido ao tamanho das matrizes utilizadas nas máscaras e das estruturas PC e NC no mapeamento das RdIs (ver Figura 3.3).

Portanto, é responsabilidade do pesquisador definir as dimensões dos ABs de maneira a equilibrar a precisão do modelo com o custo computacional associado, que nesse caso, pode ser observado pelo tempo gasto para realizar o processo de posicionamento. Portanto, vale salientar que este estudo não está focado na análise matemática do custo computacional relacionado à metodologia proposta.

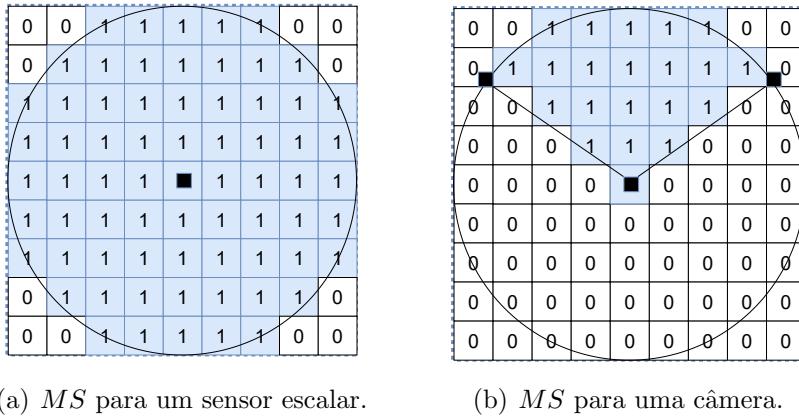
(a)  $MS$  para um sensor escalar.(b)  $MS$  para uma câmera.

Figura 3.6: Representação das Máscaras de Sensoriamento.

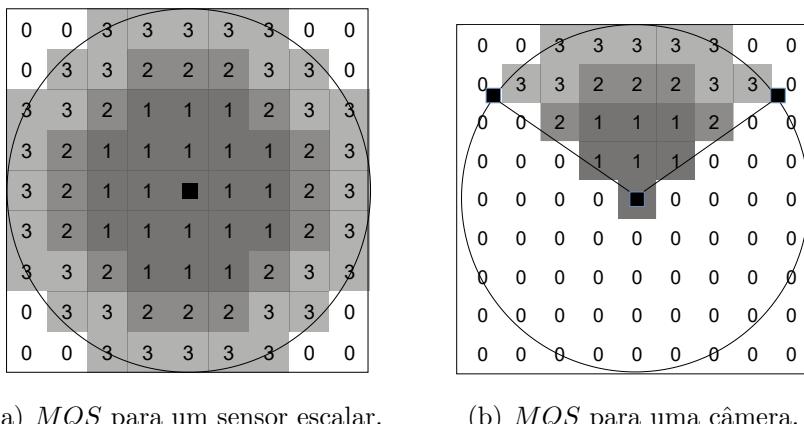
(a)  $MQS$  para um sensor escalar.(b)  $MQS$  para uma câmera.

Figura 3.7: Representação das Máscaras de Qualidade de Sensoriamento.

Com base nas ilustrações das máscaras de cobertura apresentadas nas Figuras 3.6 e 3.7, torna-se necessário estabelecer um método para a geração dessas máscaras

considerando os parâmetros específicos de cada tipo de sensor. A Listagem 2 apresenta o modelo de arquivo JSON utilizado para realizar a pré-configuração desses parâmetros. Ressalta-se que os dados fornecidos no exemplo são fictícios, não correspondendo a dispositivos disponíveis no mercado. A seguir, temos a descrição de cada campo:

- *type* - tipo do dispositivo: 1 - sensor escalar ou 2 - câmera;
- *qtd* - quantidade de dispositivos para um modelo específico;
- *radius* - raio de sensoriamento;
- *limitOfHighQuality* e *limitOfMediumQuality* - representam, respectivamente o limiar da regiões de alta e média qualidade de sensoriamento associadas às variáveis  $r_l$  e  $r_e$  (ver Figura 3.4(b));
- *viewing\_angle* - ângulo de abertura do campo de visão de uma câmera;

```

1   {
2       "sensors": [
3           {
4               "type": 1,
5               "qtd": 3000,
6               "radius": 25,
7               "limitOfHighQuality": 12.5,
8               "limitOfMediumQuality": 18.75
9           },
10          {
11              "type": 2,
12              "qtd": 1000,
13              "radius": 200,
14              "limitOfHighQuality": 100,
15              "limitOfMediumQuality": 150,
16              "viewing_angle": 135
17          }
18      ]
19  }
```

Listing 2: Modelo do arquivo de configuração dos sensores.

Com as informações da RdI e os parâmetros dos dispositivos, foram desenvolvidas estratégias para a geração automatizada das máscaras de cobertura.

### 3.2.4 Geração das Máscaras de Cobertura para um Sensor Escalar

Com o objetivo de gerar as máscaras de cobertura para cada sensor escalar especificado no arquivo JSON de configuração, foram desenvolvidos dois algoritmos. O primeiro deles, o Algoritmo 2, é responsável pela geração da  $MS$  para os sensores

escalares. A estratégia adotada pode ser entendida a partir da ilustração apresentada na Figura 3.8 e da análise do algoritmo. Nesse processo, considera-se que o sensor está posicionado na origem do plano cartesiano ( $X_c = 0$  e  $Y_c = 0$ ). Conforme mostrado na Figura 3.8, inicialmente, calcula-se a distância euclidiana ( $d$ ) entre as coordenadas  $(X_c, Y_c)$  do sensor e o ponto central do primeiro AB mais à direita encontrado na linha imediatamente superior à do sensor, representado pelo ponto destacado em negrito.

Se a distância  $d$  for menor que  $r_s$ , todos os ABs anteriores até o centro da matriz são considerados como pertencentes à região de cobertura. Caso contrário, testa-se o AB à esquerda para verificar se ele atende ao critério. Nas etapas subsequentes, outras posições são analisadas seguindo a mesma abordagem, até que todo o quadrante seja verificado e os ABs correspondentes sejam alocados (marcados em cinza). Esse processo está descrito entre as linhas 10 e 25 do Algoritmo 2. As coordenadas de cada ponto central são calculadas com base na posição (linha e coluna) da matriz e nas dimensões  $ws_{AB}$  e  $hs_{AB}$  de um AB.

É relevante destacar que, como a região de cobertura é representada por um círculo perfeito, aproveita-se sua característica simétrica para preencher as posições correspondentes nos outros quadrantes. Para cada posição alocada (AB em cinza), as posições simétricas nos demais quadrantes também são alocadas, até que toda a região esteja preenchida. Esse processo é implementado pela função descrita entre as linhas 27 e 37 do Algoritmo 2.

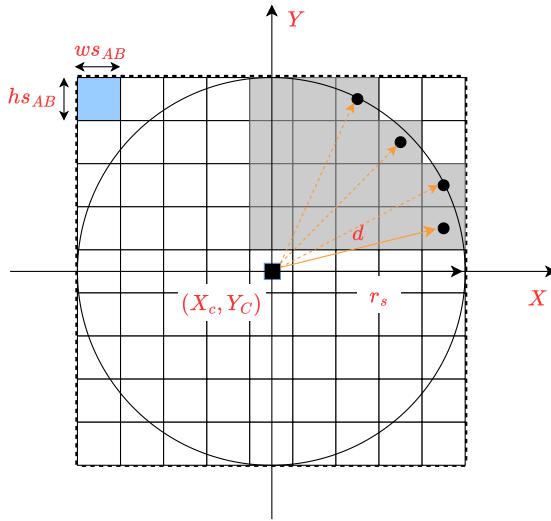
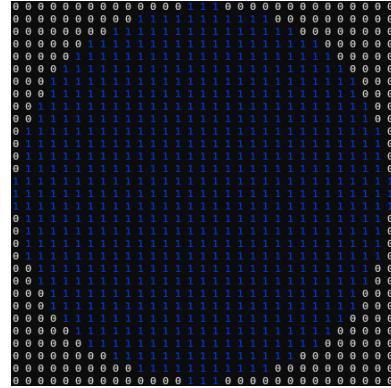


Figura 3.8: Ilustração do método de geração da  $MS$  para um sensor escalar.

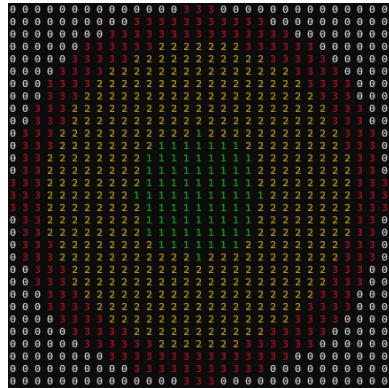
Após a formação da  $MS$ , o Algoritmo 3 realiza a geração da  $MQS$ , analisando cada posição da  $MS$  com valor 1. Nesse algoritmo, percorre-se a matriz da  $MS$ , calculando a coordenada do ponto central de cada AB correspondente em relação à origem ( $X_c = 0$  e  $Y_c = 0$ ) do plano cartesiano, conforme indicado na linha 9. Com

base nessa coordenada, calcula-se a distância euclidiana entre o ponto e a origem para determinar o nível de qualidade de sensoriamento a ser aplicado à posição, conforme os intervalos estabelecidos na Equação 3.5a.

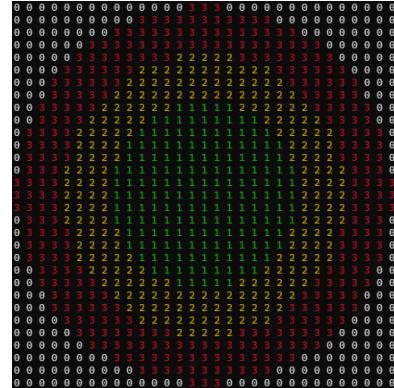
Na Figura 3.9, apresentam-se as máscaras de cobertura geradas a partir desses algoritmos para um sensor escalar com  $r_s = 30m$ , sendo cada posição das matrizes correspondente a um AB de  $2 \times 2m$ . A Figura 3.9(a) exibe a  $MS$ , enquanto as Figuras 3.9(b) e 3.9(c) ilustram as  $MQS$  para diferentes valores de  $r_l$  e  $r_e$ , demonstrando a flexibilidade do método em simular sensores escalares com características de sensoriamento distintas.



(a)  $MS$  gerada a partir do Algoritmo 2.



(b)  $MQS$  gerada a partir do Algoritmo 3 considerando  $r_l = 15$  e  $r_e = 22.5$ .



(c)  $MQS$  gerada a partir do Algoritmo 3 considerando  $r_l = 10$  e  $r_e = 25$ .

Figura 3.9: Exemplo das máscaras de cobertura geradas a partir dos Algoritmos 2 e 3 para um sensor escalar com  $r_s = 30$ .

### 3.2.5 Geração das Máscaras de Cobertura para as Câmeras

Para os sensores visuais, ou seja, as câmeras, tanto a  $MS$  quanto a  $MQS$  são geradas simultaneamente. Na Figura 3.5(b), observa-se que a região de cobertura

de uma câmera corresponde a um recorte da área de um círculo, definido pelos valores do ângulo de abertura do campo de visão e sua orientação, representados, respectivamente, pelas variáveis  $\theta$  e  $\alpha$ . Assim, considerando que as máscaras de cobertura são representadas como matrizes, conforme ilustrado nas Figuras 3.6(b) e 3.7(b), é necessário estabelecer um método para determinar quais posições ou ABs estão dentro da região especificada.

Para isso, pode-se utilizar uma estratégia baseada na colinearidade de pontos no plano cartesiano. Conforme ilustrado na Figura 3.10, considera-se os vetores  $\vec{AB}$  e  $\vec{CA}$  orientados no sentido anti-horário. Com base nessa definição, é necessário verificar se a coordenada do ponto central de um AB, representado pelo ponto  $P$ , atende a dois critérios:

1. A distância entre  $P$  e  $A$  deve ser menor que  $r_s$ ;
2. Respeitando a direção estabelecida pelos vetores,  $P$  deve estar localizado à esquerda de  $\vec{AB}$  e  $\vec{CA}$  ou sobre as retas formadas por eles;

Se ambos os critérios forem atendidos, a posição correspondente a  $P$  na matriz é marcada como pertencente à região de cobertura da câmera.

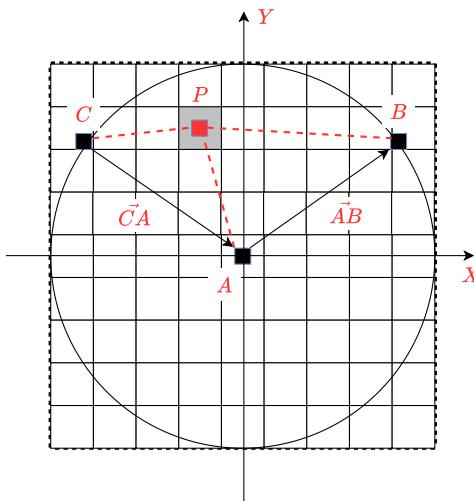


Figura 3.10: Ilustração do método de geração da *MS* e da *MQS* para uma câmera.

O primeiro critério é verificado por meio do cálculo da distância euclidiana entre as coordenadas  $(A_x, A_y)$  e  $(P_x, P_y)$ . Já para o segundo critério, é necessário calcular o determinante das seguintes composições:  $\Delta(A, B, P)$  e  $\Delta(C, A, P)$ , conforme mostrado nas Equação 3.8a e 3.8b.

$$\Delta(A, B, P) = (P_y - A_y).(B_x - A_x) - (P_x - A_x).(B_y - A_y) \quad (3.8a)$$

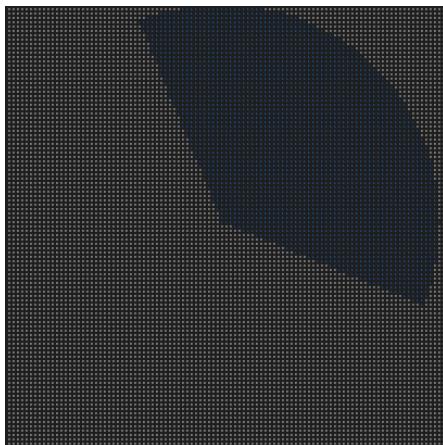
$$\Delta(C, A, P) = (P_y - C_y).(A_x - C_x) - (P_x - C_x).(A_y - C_y) \quad (3.8b)$$

A análise do sinal de  $\Delta$  permite identificar os seguintes casos:

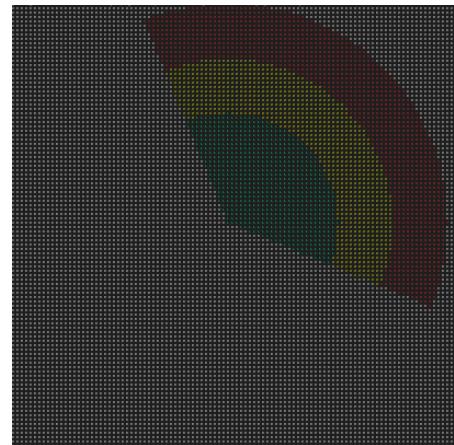
- 1 - Se  $\Delta = 0$ , então  $P$  pertence à reta formada pelos dois outros pontos;
- 2 - Se  $\Delta < 0$ , então  $P$  está à esquerda da reta formada pelos outros dois pontos;
- 3 - Se  $\Delta > 0$ , então  $P$  está à direita da reta formada pelos outros dois pontos;

Dessa forma, para que o segundo critério seja atendido, os determinantes calculados nas Equação 3.8a e 3.8b devem produzir resultados correspondentes aos casos  $\Delta = 0$  ou  $\Delta < 0$ . Com ambos os critérios satisfeitos, a posição na  $MS$  é marcada como pertencente à região de cobertura. Além disso, a distância calculada entre  $P$  e  $A$  é avaliada conforme a Equação 3.5a, a fim de determinar o respectivo nível de qualidade de sensoriamento.

Esse processo é realizado para todas as posições das matrizes até que toda a máscara seja gerada. A estratégia também pode ser analisada através do Algoritmo 4. Na Figura 3.11, são apresentadas as máscaras de cobertura geradas para uma câmera utilizando esse método, considerando  $\theta = 135^\circ$ ,  $\alpha = 45^\circ$ ,  $r_s = 100$ ,  $r_l = 50$  e  $r_e = 75$ . Cada posição nas matrizes representa um AB de  $2 \times 2m$ . Assim como nos sensores escalares, é possível definir diferentes características de monitoramento para as câmeras.



(a)  $MS$  gerada a partir do Algoritmo 4.



(b)  $MQS$  gerada a partir do Algoritmo 4 considerando  $r_l = 50$  e  $r_e = 75$ .

Figura 3.11: Exemplo das máscaras de cobertura geradas a partir do Algoritmo 4 para uma câmera com  $r_s = 100$ ,  $\theta = 135^\circ$ ,  $\alpha = 45^\circ$ .

---

**Algorithm 2:** Algoritmo de geração da MS para um sensor escalar.

---

**Data:**  $ws_{AB}$ ;  $hs_{AB}$ ;  $r_s$

**Result:** Matriz representando a  $MS$

```

1   $MS \leftarrow \emptyset;$ 
2  /* Global and control variables */
3   $x\_maskDimension \leftarrow (2 * (\text{int})(r_s/ws_{AB})) + 1;$ 
4   $y\_maskDimension \leftarrow (2 * (\text{int})(r_s/hs_{AB})) + 1;$ 
5   $x\_halfDimension \leftarrow (\text{int})x\_dimension \div 2;$ 
6   $y\_halfDimension \leftarrow (\text{int})y\_dimension \div 2;$ 
7   $a \leftarrow x\_halfDimension;$ 
8   $b \leftarrow 0;$ 
9   $j \leftarrow x\_maskDimension;$ 
10 for  $i \leftarrow y\_halfDimension$  to 0 do
11   | if  $\text{EuclidianDistance}(0, 0, (a \times ws_{AB}), (b \times hs_{AB})) \leq r_s$  then
12   |   |  $MS \leftarrow \text{InsertValues}(MS, i, j);$ 
13   |   |  $b++;$ 
14   | else
15   |   | while  $a > 0$  do
16   |   |   |  $a--; j--;$ 
17   |   |   | if  $\text{EuclidianDistance}(0, 0, (a \times ws_{AB}), (b \times hs_{AB})) \leq r_s$  then
18   |   |   |   |  $MS \leftarrow \text{InsertValues}(MS, i, j);$ 
19   |   |   |   |  $b++;$ 
20   |   |   |   | break;
21   |   |   | end
22   |   | end
23   | end
24   |  $MS[y\_halfDimension][x\_halfDimension] \leftarrow 1;$ 
25 end
26 return  $MS;$ 
27 Function  $\text{InsertValues}(mask, line, j):$ 
28   for  $column \leftarrow j$  to  $x\_halfDimension$  do
29   |   | # Set the Sensing Mask
30   |   |  $mask[line][column] \leftarrow 1;$ 
31   |   | # Replicates on the left
32   |   |  $mask[line][column - a] \leftarrow 1;$ 
33   |   | # Replicates in the bottom right corner
34   |   |  $mask[y\_halfDimension + b][column] \leftarrow 1;$ 
35   |   | # Replicates in the bottom left corner
36   |   |  $mask[y\_halfDimension + b][column - a] \leftarrow 1;$ 
37   | end
38   | return  $mask;$ 

```

---

---

**Algorithm 3:** Algoritmo de geração da MQS para um sensor escalar.

---

**Data:**  $MS$  com  $L$  linhas e  $C$  colunas;  $r_l$ ;  $r_e$ ;  $r_s$ ;  $ws_{AB}$ ;  $hs_{AB}$

**Result:** Matriz representando a  $MQS$

```

1  $MQS \leftarrow \emptyset;$ 
2  $maskDimension \leftarrow \{(2 * (int)(r_s / hs_{AB})) + 1, (2 * (int)(r_s / ws_{AB})) + 1\};$ 
3 # Mask center position coordinate
4  $p\_mask\_center \leftarrow \{maskDimension[0] \div 2, maskDimension[1] \div 2\};$ 
5 # It runs through the MS checking the positions with sensing
6 for  $line \leftarrow 0$  to  $maskDimension[0]$  do
7   for  $column \leftarrow 0$  to  $maskDimension[1]$  do
8     if  $MS[line][column] == 1$  then
9        $point \leftarrow point\_inRelatedTo\_center(p\_mask\_center[0],$ 
10       $p\_mask\_center[1], line, column);$ 
11       $d \leftarrow EuclidianDistance(0,0, point[0], point[1]);$ 
12      # Set the sensing quality level
13      if  $d \leq r_l$  then
14         $| MQS[line, column] \leftarrow 1;$ 
15      else
16        if  $d > r_l \ \& \ d \leq r_e$  then
17           $| MQS[line, column] \leftarrow 2;$ 
18        else
19           $| MQS[line, column] \leftarrow 3;$ 
20        end
21      end
22    else
23       $| MQS[line, column] \leftarrow 0;$ 
24    end
25  end
26 return  $MQS;$ 

```

---

---

**Algorithm 4:** Algoritmo de geração da MS e MQS para uma câmera.

---

**Data:** Parâmetros  $r_s$ ,  $r_l$ ,  $r_e$ ,  $ws_{AB}$  e  $hs_{AB}$

**Result:** Matrizes representando a  $MS$  e a  $MQS$

```

1 # Coordinates of vertices A, B and C
2  $A_v \leftarrow \{0, 0\}; B_v \leftarrow \emptyset; C_v \leftarrow \emptyset;$ 
3  $maskDimension \leftarrow calculateMaskDimension(r_s, hs_{AB}, ws_{AB});$ 
4 # Mask center position coordinate
5  $p\_mask\_center \leftarrow \{maskDimension[0] \div 2, maskDimension[1] \div 2\};$ 
6 /* Vertex calculation
7  $\alpha_1 \leftarrow (\alpha - (\theta \div 2)).mod2\pi;$ 
8  $\alpha_2 \leftarrow (\alpha + (\theta \div 2)).mod2\pi;$ 
9  $B_v[0] \leftarrow A_v[0] + \cos(\alpha_1).r_s;$ 
10  $B_v[1] \leftarrow A_v[1] + \sin(\alpha_1).r_s;$ 
11  $C_v[0] \leftarrow A_v[0] - \cos(\alpha_2).r_s;$ 
12  $C_v[1] \leftarrow A_v[1] + \sin(\alpha_2).r_s;$ 
13 for  $i \leftarrow 0$  to  $maskDimension[0]$  do
14   for  $j \leftarrow 0$  to  $maskDimension[1]$  do
15     point  $\leftarrow point\_inRelatedTo\_center(p\_mask\_center[0], p\_mask\_center[1],$ 
        line, column);
16      $d \leftarrow EuclidianDistance(0,0, point[0], point[1]);$ 
17     if  $d \leq r_s$  then
18        $detV_{AB} \leftarrow DeterminantCalculation(A_v, B_v);$ 
19        $detV_{CA} \leftarrow DeterminantCalculation(C_v, A_v);$ 
20       if  $detV_{AB} \leq 0$  &  $detV_{CA} \leq 0$  then
21         # Set the MS
22          $MS[i][j] \leftarrow 1;$ 
23         # Set the MQS
24         if  $d \leq r_l$  then
25            $MQS[i][j] \leftarrow 1;$ 
26         else
27           if  $d > r_l$  &  $d \leq r_e$  then
28              $MQS[i][j] \leftarrow 2;$ 
29           else
30              $MQS[i][j] \leftarrow 3;$ 
31           end
32         end
33       else
34          $MS[i][j] \leftarrow 0;$ 
35          $MQS[i][j] \leftarrow 0;$ 
36       end
37     end
38   end
39 end

```

---

### 3.3 Funções de *Fitness*

Após a descrição do modelo representativo para as RdI e para os dispositivos de monitoramento, nesta seção são descritas as funções de *fitness* utilizadas para a maximização dos critérios de *cobertura*, *alta conectividade* e *qualidade de sensoriamento* de uma RSSF construída a partir de um conjunto de sensores escalares e visuais. Essas funções são processadas posteriormente pelo algoritmo genético para avaliar a qualidade das soluções produzidas.

#### 3.3.1 Cobertura

Segundo Fei et al. (2017), a taxa de cobertura consiste em uma das mais importantes métricas quando se estuda as RSSF. Ela define a eficácia da rede de sensores no monitoramento de uma região, pois, quanto maior o alcance de sensoriamento da rede, melhor será sua eficácia na detecção de eventos adversos ao longo de vários pontos da RdI.

Entre os três tipos de monitoramento de área associados às RSSF, os quais são, Cobertura de Área, Cobertura de Pontos e Cobertura de Barreira, este trabalho adota a Cobertura de Área como o primeiro objetivo de otimização, por ser uma aplicação mais abrangente, utilizada em pesquisas nos mais diversos contextos. Um planejamento adequado da cobertura em grandes regiões de monitoramento favorece a economia de energia e o prolongamento da vida útil da rede, minimizando a necessidade de realocação dos sensores ou implantação de nós adicionais (Fei et al., 2017).

Diante disso, ao implantar um conjunto de dispositivos  $Sr = \{sr_1, sr_2, sr_3, \dots, sr_N\}$  em uma RdI subdividida em AB, busca-se maximizar a quantidade de blocos monitorados por ao menos um sensor. Para isso, após o posicionamento do conjunto  $Sr$ , pode-se calcular a quantidade de ABs monitorados ( $AB_{sensed}$ ) em relação ao total ( $AB_{total}$ ). Entretanto, é necessário levar em consideração um aspecto específico: quando os sensores são posicionados próximos ao limiar ou à borda da RdI, os sensores podem cobrir áreas que não são relevantes para a aplicação, ou seja, que estão fora da RdI.

Para tratar esse problema, calcula-se o número de AB inválidos e aplica-se uma penalidade à função de *fitness* final. Os AB inválidos correspondem aos blocos de área monitorados fora da RdI. Com isso, a penalidade é definida da seguinte forma:

- Para cada três ABs inválidos, um AB válido é desconsiderado na taxa de cobertura.

A partir disso, sendo  $S_i$  a i-ésima solução gerada pelo algoritmo genético, a função de otimização  $f_1$  avalia a cobertura de área dessas soluções, conforme descrito na Equação 3.9. É importante mencionar que o intervalo de  $f_1$  corresponde a  $[0, 1]$ .

$$f_1(S_i) = \frac{(AB_{sensed} - \frac{AB_{invalid}}{3})}{AB_{total}} \quad (3.9)$$

### 3.3.2 Dispositivos em regiões de alta conectividade

Com o cálculo da métrica *DPConn* (explicado na Subseção 3.1.1), foi possível definir, como um parâmetro de qualidade de rede, os níveis de conectividade de cada AB presente em uma RdI. Para inúmeras aplicações de monitoramento, é imprescindível que os dispositivos de uma RSSF possuam acesso a um sinal de comunicação robusto, garantindo maior eficiência e integridade no processo de transmissão e recebimento de dados ao longo do tempo.

Diante desse fato, a segunda função de otimização ( $f_2$ ) busca maximizar a taxa de dispositivos posicionados em regiões de alta conectividade ( $Sr_{inH}$ ), ou seja, em AB com o nível de conectividade igual a 3, garantindo também que a seguinte expressão seja atendida:  $Sr_{inH} > Sr_{inM} > Sr_{inL}$ . Essa expressão busca assegurar que, em conformidade com as características da RdI, o menor número possível de dispositivos seja posicionado em regiões de baixa conectividade ( $Sr_{inL}$ ), seguidas das regiões de média conectividade ( $Sr_{inM}$ ), enquanto a maioria é alocada em regiões de alta conectividade.

A partir disso, temos a formulação de  $f_2$  na Equação 3.10a, sendo o seu intervalo variando entre [0, 1]. Após a geração da solução  $S_i$  pelo algoritmo genético, verifica-se a taxa de dispositivos implantados nas regiões de alta, média e baixa conectividade em relação ao total pré-estabelecido através do arquivo de configuração (ver Listagem 2). Essas taxas são representadas, respectivamente, pelas Equações 3.10b, 3.10c e 3.10d.

$$f_2(S_i) = \begin{cases} Sr_{inH} * e^{-\alpha * Sr_{inH}}, & Sr_{inL} > Sr_{inH} \mid Sr_{inL} > Sr_{inM}, \\ Sr_{inH}, & \text{caso contrário} \end{cases} \quad (3.10a)$$

$$Sr_{inH} = Sensor_{NetHigh} \div Total_{sensors} \quad (3.10b)$$

$$Sr_{inM} = Sensor_{NetMedium} \div Total_{sensors} \quad (3.10c)$$

$$Sr_{inL} = Sensor_{NetLow} \div Total_{sensors} \quad (3.10d)$$

A Equação 3.10a estabelece que, caso a taxa  $Sr_{inL}$  seja superior a  $Sr_{inH}$  ou  $Sr_{inM}$ , aplica-se uma penalização exponencial sobre o valor de  $Sr_{inH}$ . Essa função tem como propósito intensificar a penalidade atribuída à variável  $Sr_{inH}$  conforme seu valor aumenta, de modo que será aplicada caso a proporção de dispositivos em regiões de baixa conectividade ultrapasse aquela verificada nas áreas de média ou alta conectividade. O grau dessa penalização é controlado pelo parâmetro  $\alpha$ , cuja influência foi avaliada por meio de um experimento considerando uma rede de sensores composta por 1000 dispositivos.

Na Figura 3.12 apresenta-se o comportamento de  $f_2$  a partir de  $\alpha = 0.3, 0.4, 0.5, 0.6$ . Observa-se que, quanto maior o valor de  $Sr_{inH}$ , maior é o decaimento na curva da função  $f_2$ , em consequência dessa penalidade. Além disso, quanto maior o valor de  $\alpha$ , o grau de penalização cresce em conformidade. Então, para não ser empregado um grau elevado de penalidade ou mesmo muito baixo, o valor de 0.5 foi escolhido para ser utilizado nos testes subsequentes.

O uso dessa abordagem de penalização foi adotado com o objetivo de impulsionar o algoritmo genético a manter uma alta quantidade de dispositivos em regiões de alta conectividade, enquanto aloca o menor número possível de dispositivos em regiões de baixa conectividade, buscando, se possível, respeitar a expressão  $Sr_{inH} > Sr_{inM} > Sr_{inL}$ .

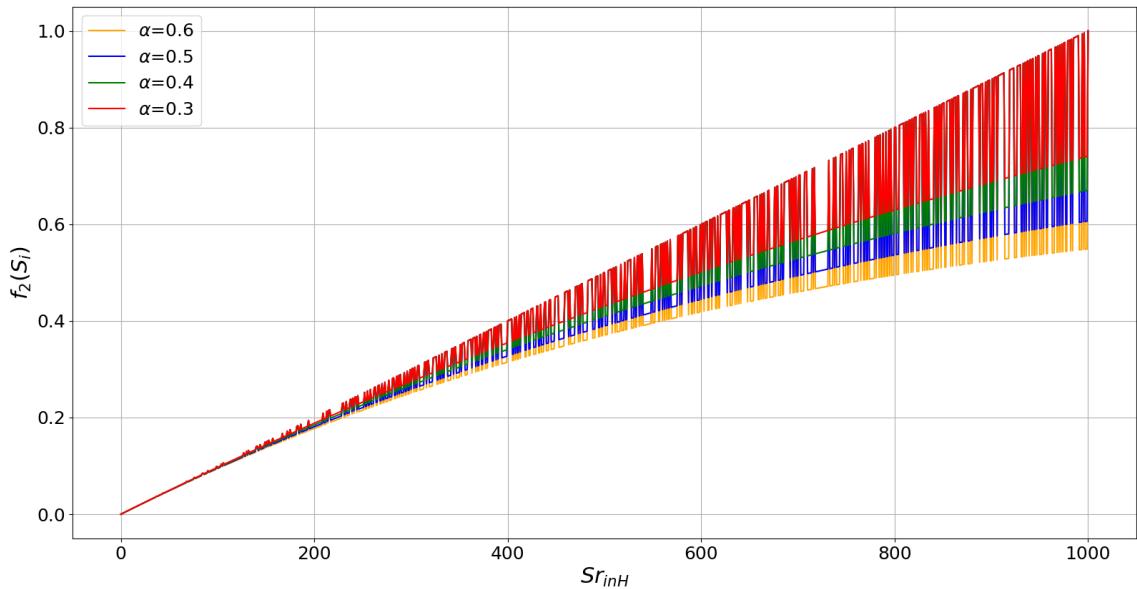


Figura 3.12: Influência do parâmetro  $\alpha$  no comportamento da função de fitness  $f_2$ .

### 3.3.3 Qualidade de Sensoriamento

Além da busca pela maximização da cobertura e do posicionamento dos dispositivos em regiões de alta conectividade, também é necessário garantir a qualidade de sensoriamento ao longo da RdI. De acordo com o modelo proposto na Subseção 3.2.2 para a região de cobertura de um sensor, é possível que um determinado evento esteja ocorrendo em uma área com baixa qualidade de sensoriamento. Isso implica que, nessa região, a precisão com que o evento é detectado é baixa devido à distância ou outros aspectos, como ruídos associados ao ambiente. Por essa razão, é necessário que se busque maximizar o número de áreas com alta qualidade de sensoriamento.

Para alcançar esse objetivo, é realizada a sobreposição das regiões de cobertura que estão próximas umas das outras. Desta forma, considerou-se que o nível  $Q_S$  do

i-ésimo AB é dado pelo maior nível de qualidade sobreposto a essa área. Portanto, seja  $Sr = \{sr_1, sr_2, sr_3, \dots, sr_N\}$  o conjunto de sensores que monitoram o mesmo  $AB_i$ , o  $Q_S(AB_i)$  é dado pelo maior nível de qualidade imposto por um sensor do conjunto  $Sr$ . A Equação 3.11 apresenta formalmente essa definição.

$$Q_S(AB_i) = \max(Q_S(sr_j, p_{AB_i}), Q_S(sr_{j+1}, p_{AB_i}), \dots, Q_S(sr_n, p_{AB_i})) \quad (3.11)$$

A partir disso, como pode ser visto na Equação 3.12a, 3.12b e 3.12c, para cada AB monitorado verifica-se o seu respectivo nível  $Q_S$  e realiza-se o cálculo da taxa de AB em cada um dos níveis em relação ao total de blocos monitorados. As funções *BlocksHS*, *BlocksMS* e *BlocksLS* percorrem o registro de blocos monitorados verificando se os mesmos possuem o nível  $Q_S$  desejado (*HS* = sensoriamento alto; *MS* = sensoriamento médio; *LS* = sensoriamento baixo).

$$AB_{wHS} = \frac{\sum_{i=1}^N BlocksHS(AB_{i_sensed})}{AB_{sensed}} \quad (3.12a)$$

$$AB_{wMS} = \frac{\sum_{i=1}^N BlocksMS(AB_{i_sensed})}{AB_{sensed}} \quad (3.12b)$$

$$AB_{wLS} = \frac{\sum_{i=1}^N BlocksLS(AB_{i_sensed})}{AB_{sensed}} \quad (3.12c)$$

Depois desse processo, pode-se calcular a função de *fitness*  $f_3$ , demonstrada na Equação 3.13. Seu intervalo corresponde a  $[0, 1]$ . Perceba que  $f_3$  possui a mesma estrutura da função  $f_2$ , pois, como originalmente o comportamento de ambas é linear, então pode-se aproveitar a mesma composição, incluindo a função exponencial para a penalidade com  $\alpha = 0.5$ . Também, assim como em  $f_2$ , mas agora em relação a  $f_3$ , deseja-se que, sempre que possível, a seguinte expressão seja alcançada:  $AB_{wHS} > AB_{wMS} > AB_{wLS}$ . Garantindo que a maioria dos ABs sejam monitorados por regiões de cobertura com alta qualidade de sensoriamento, seguido pelos ABs com média, e por último, com baixa.

$$f_3(S_i) = \begin{cases} AB_{wHS} * e^{-\alpha * AB_{wHS}}, & AB_{wLS} > AB_{wHS} \mid AB_{wLS} > AB_{wMS}, \\ AB_{wHS}, & \text{caso contrário} \end{cases} \quad (3.13)$$

### 3.4 Posicionamento através do Algoritmo Genético NSGA-II

A partir de toda a modelagem apresentada, que inclui o mapeamento da RdI, a representação dos dispositivos de monitoramento e as funções de *fitness*, foi decidido

utilizar uma abordagem heurística baseada no uso do algoritmo genético multiobjetivo NSGA-II, a fim de realizar o balanceamento entre os objetivos de interesse que estão expressos através dos critérios de otimização.

A abordagem multiobjetivo se torna a mais adequada para este estudo devido à característica conflitante das funções de *fitness*. Por exemplo, a função para análise de cobertura ( $f_1(S_i)$ ) visa maximizar o número de áreas monitoradas dentro da RdI; para isso, os dispositivos devem ser espalhados ao máximo, e assim, minimizando também a sobreposição das regiões de cobertura. Enquanto isso, a função  $f_2(S_i)$  busca implantar o maior número possível desses dispositivos em regiões de alta conectividade que, na maioria das vezes, pode causar a aglomeração de muitos dispositivos em regiões específicas, desta forma, prejudicando a taxa de cobertura da RdI.

Por esse motivo, a utilização da abordagem multiobjetivo garantirá que sejam encontradas soluções que apresentam um balanceamento ou equilíbrio entre esses critérios. Tendo isso em vista, o algoritmo NSGA-II, proposto por Deb et al. (2002), foi escolhido para ser utilizado neste estudo, pois, na literatura, ainda é muito utilizado por pesquisadores em diversos projetos, que também incluem o posicionamento de sensores, como mostrado na Seção 2.8.

A primeira etapa na utilização de um algoritmo genético consiste em representar o problema de interesse através do processo de codificação. Por isso, nas subseções subsequentes será detalhado como as soluções candidatas foram codificadas, bem como o seu processo de decodificação a fim de avaliar as funções de *fitness*.

### 3.4.1 Codificação: Cromossomo

O principal elemento de um algoritmo genético é a estrutura chamada Cromossomo. Ele representa a codificação de uma solução candidata (indivíduo) que será avaliada pelo algoritmo. É de extrema importância que esse cromossomo expresse corretamente a solução para o problema, pois toda a análise e manipulação realizadas pelo algoritmo genético são feitas sobre esse elemento, de forma que problemas na representação podem produzir impactos negativos no desempenho do algoritmo, podendo levar a soluções sub-ótimas ou até mesmo à convergência prematura para resultados inadequados. Assim, a escolha de uma representação cromossômica adequada é um fator crítico para a eficácia do algoritmo genético, influenciando diretamente sua capacidade de explorar o espaço de soluções e encontrar respostas otimizadas para o problema em questão.

Como este estudo se concentra no posicionamento de sensores, então, a estrutura do cromossomo precisa expressar a localização de cada dispositivo dentro da RdI. Para tanto, alguns detalhes precisam ser considerados. Primeiramente, os dispositivos serão idealmente posicionados no centro de um AB pertencente à RdI, sendo permitido que mais de um dispositivo seja alocado no mesmo bloco. Além disso, os dispositivos são alocados apenas em ABs com nível de conectividade diferente

de zero, de forma a garantir que exista algum sinal de conexão disponível para os processos de comunicação.

Como as RdI são conceitualmente visualizadas no formato de matrizes bidimensionais onde cada posição representa um AB, então, através das variáveis  $hs_{AB}$  e  $ws_{AB}$  em conjunto com a delimitação da área pertencente à RdI, torna-se simples calcular a coordenada do seu ponto central. Além disso, analisando os AB que apresentam conectividade mapeados através das estruturas PC e NC (ver Figura 3.3) o algoritmo genético poderá escolher e avaliar as posições de alocação com base nos valores armazenados nessas estruturas.

A partir dessas definições, a estrutura do cromossomo utilizada no algoritmo genético pode ser vista na Figura 3.13. Considerando  $Sr = \{sr_1, sr_2, sr_3, \dots, sr_n\}$  o conjunto de sensores escalares e  $Cr = \{cr_1, cr_2, cr_3, \dots, cr_m\}$  o conjunto de câmeras, os genes do cromossomo foram divididos da seguinte forma: os primeiros  $n$  genes, sendo  $n$  igual ao tamanho do conjunto  $Sr$ , são utilizados para representar a posição de cada sensor escalar do conjunto  $Sr$ ; os próximos  $m$  genes, sendo  $m$  igual ao tamanho do conjunto  $Cr$ , são utilizados para representar a posição das câmeras. No entanto, de acordo com a representação da região de cobertura das câmeras (ver Figura 3.5(b)), é permitido que elas sejam rotacionadas através da variável  $\alpha$ . Por essa razão, a terceira parte do cromossomo é alocada para armazenar o ângulo de orientação para cada uma das câmeras posicionadas.

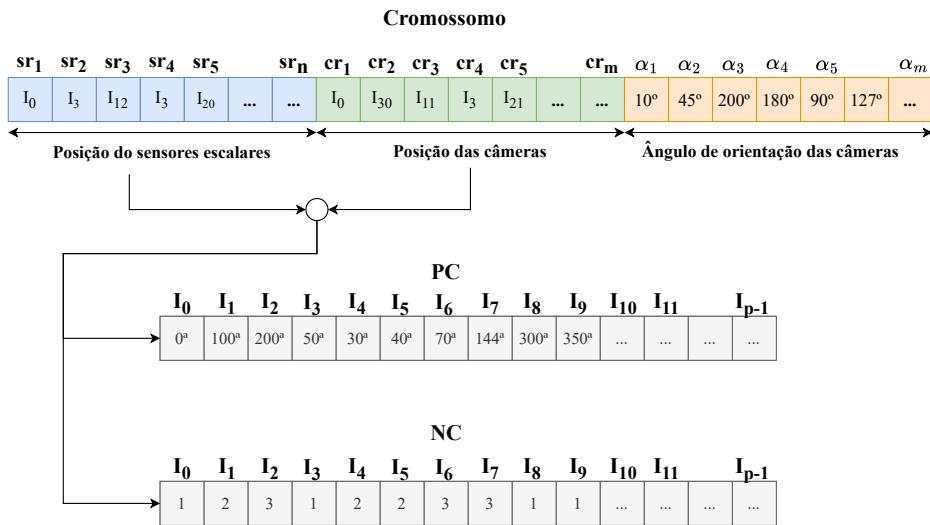


Figura 3.13: Estrutura do Cromossomo.

Os genes que representam as posições estão conectados às estruturas PC e NC, e seus valores são representados pelo índice  $I_p$  dessas estruturas. Desta forma, o algoritmo genético gera novas soluções com base apenas na posição dos AB que possuem algum nível de conectividade, sendo possível saber não somente a posição em que o sensor será alocado, mas também o nível de conectividade do respectivo local.

É importante ressaltar que os genes de um cromossomo podem ser representados com diferentes codificações, que são do tipo: inteiro, real, binário ou nominal. Para este estudo, todos os genes estão codificados com valor inteiro, pois as posições dos dispositivos são representadas através do índice de um *array*, e por simplicidade, o grau de orientação das câmeras também constitui-se de valores inteiros.

Com isso, a partir dessa representação, é possível calcular e avaliar todas as funções de *fitness* sobre cada solução gerada pelo algoritmo genético.

### 3.4.2 Demonstração do método de posicionamento por meio das soluções geradas pelo NSGA-II

Como dito anteriormente, as soluções geradas pelo algoritmo genético são representadas através da estrutura do cromossomo ilustrada na Figura 3.13. Esse cromossomo precisa ser processado visando avaliar a qualidade da solução por meio das funções de *fitness*.

A cada nova solução, temos todas as informações necessárias para alocar os dispositivos pré-configurados em sua respectiva posição dentro da RdI. Para isso, são lidos os valores dos genes do cromossomo que indicam os índices de posicionamento dos sensores escalares e das câmeras, e com esses índices verifica-se na estrutura PC, qual é o AB correspondente na RdI. Com o bloco identificado, é necessário aplicar as máscaras *MS* e *MQS* sobre o local definido de forma a determinar os ABs da RdI que serão monitorados pelo dispositivo.

Entretanto, é importante mencionar que as máscaras de cobertura dos sensores escalares são geradas antes da execução do algoritmo genético, pois não se alteram ao longo do processo de avaliação das soluções. Em relação às câmeras, a cada nova solução gerada, as respectivas máscaras de cobertura precisam ser recalculadas devido à mudança no seu ângulo de orientação.

Para que o processo de avaliação das soluções seja melhor compreendido, será feita uma análise da ilustração apresentada na Figura 3.14. Esse exemplo é meramente fictício para fins de esclarecimento. Primeiramente, toma-se como base a RdI representada na Figura 3.3 tendo suas posições com conectividade mapeadas por meio das estruturas PC e NC (ver Listagem 1).

Em seguida, deve-se configurar os parâmetros associados aos dispositivos de monitoramento. Nesse exemplo, um conjunto  $Sr = \{sr_1, sr_2, sr_3\}$  de sensores escalares e outro  $Cr = \{cr_1, cr_2\}$  de câmeras. Por enquanto, pode-se ignorar os aspectos de parametrização relacionados ao algoritmo genético. Então, a partir dessas informações, inicia-se o processo de geração de possíveis soluções ótimas para o problema de posicionamento.

Na parte superior da Figura 3.14 pode-se averiguar o cromossomo que está representando uma possível solução gerada pelo algoritmo. Os genes em cor azul e verde estão indicando os índices para as estruturas PC e NC que podem ser vistas na

parte inferior da figura. Quando esses índices são verificados, encontra-se a posição em que cada dispositivo será posicionado na RdI. Na estrutura PC, essas posições estão marcadas na cor cinza, e logo abaixo, em NC, apresenta-se o respectivo nível de conectividade da posição selecionada.

A partir dessas posições, é realizado o processo de posicionamento. Convertendo os valores dos genes lidos em uma representação de linha e coluna, temos o local de implantação de cada dispositivo. Com cada um possuindo suas respectivas máscaras de cobertura, elas são sobrepostas à região de implantação. Pode-se perceber que um dos dispositivos é alocado próximo ao limite de área da RdI, então parte da região de monitoramento se perde, fazendo com que os blocos das máscaras que se encontram fora da RdI sejam considerados como AB inválidos. Nesse exemplo, está sendo ilustrada a aplicação da *MS*. Então, ainda na Figura 3.14, observa-se a alocação dos elementos dos conjuntos *Sr* e *Cr*, e o resultado final produzido.

Após a aplicação das *MS*, cada AB da RdI está sendo monitorado por um, dois ou mesmo nenhum dispositivo. Esse resultado é armazenado em uma estrutura unidimensional e posteriormente utilizado para a avaliação da qualidade da solução por meio das funções de *fitness*. O processo de aplicação das *MQS* é semelhante, e o resultado final pode ser observado na Figura 3.15. Contudo, esse resultado é armazenado em outra estrutura unidimensional, também para fins de avaliação da qualidade da solução.

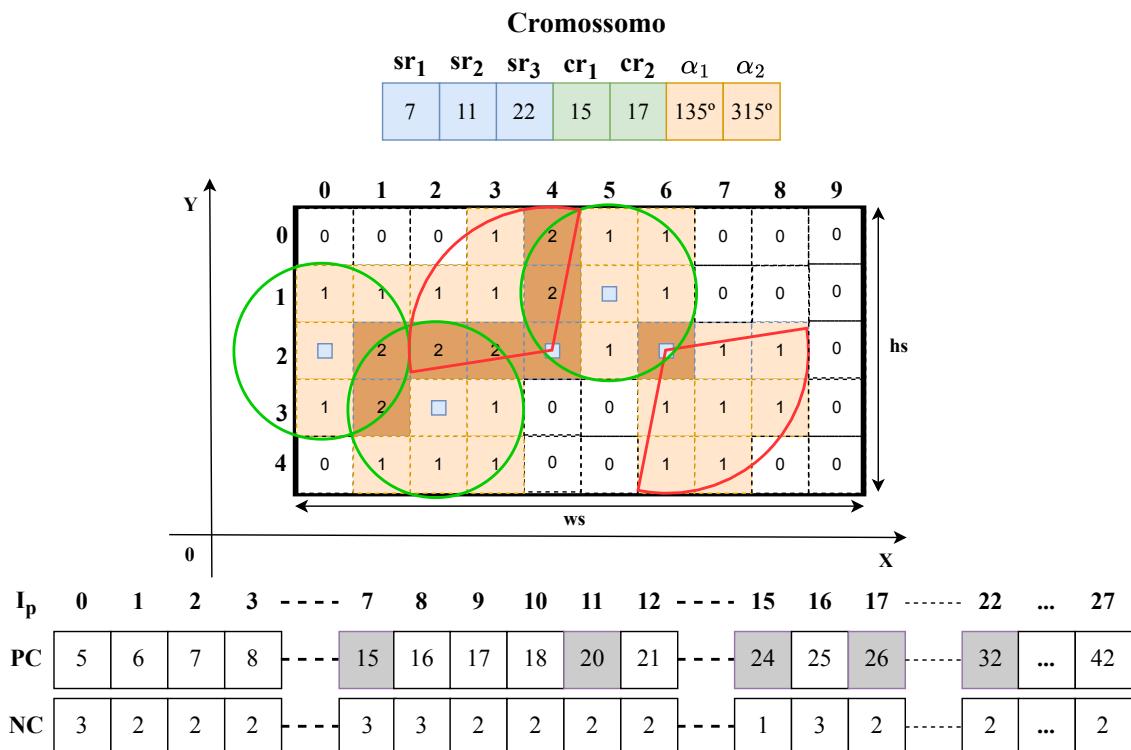


Figura 3.14: Ilustração do processo de posicionamento.

Depois que o processo de posicionamento é finalizado, segue-se a avaliação da solução através das funções de *fitness*. Primeiramente, podemos verificar a  $f_1(s)$  para análise de cobertura. A RdI utilizada está subdividida em 50 blocos, logo,  $AB_{total} = 50$ . Do total, 32 são monitorados, então,  $AB_{sensed} = 32$ . Após a sobreposição de todas as  $MS$ , 3 blocos ficaram fora da RdI, então,  $AB_{invalid} = 3$ . Com isso, de acordo com a Equação 3.9, encontra-se o valor para  $f_1(s)$  na Equação 3.14.

$$f_1(s) = \frac{(AB_{sensed} - \frac{AB_{invalid}}{3})}{AB_{total}} = \frac{(32 - \frac{3}{3})}{50} = 0.62 = 62\% \quad (3.14)$$

Para o cálculo da função  $f_2(s)$  (ver Equação 3.10a a 3.10d) é necessário verificar as taxas de dispositivos alocados nas regiões de alta, média e baixa conectividade. Analisando na Figura 3.14 o cromossomo com a solução gerada e as estruturas PC e NC, percebe-se que 1 dispositivo foi implantado em uma região de alta conectividade, 3 em regiões de média e 1 em região de baixa. Sendo  $Total_{sensors} = 5$ , logo,  $Sr_{inH} = 1/5 = 0.2$ ,  $Sr_{inM} = 3/5 = 0.6$  e  $Sr_{inL} = 1/5 = 0.2$ . Como pode ser visto na Equação 3.10a, temos a expressão  $Sr_{inL} > Sr_{inH} | Sr_{inL} > Sr_{inM}$  que determina se uma penalidade deve ser aplicada. De acordo com os valores calculados, nenhum dos termos da penalidade é atendido, logo a  $f_2(s) = Sr_{inH} = 0.2$ .

Para finalizar, é calculado o valor de  $f_3(s)$  que mede a qualidade de sensoriamento sobre a RdI após a implantação dos dispositivos. Para isso, precisa-se do resultado apresentado na Figura 3.15. Esse é o resultado da aplicação das *MQS* sobre a RdI. Cada AB que é monitorado por um ou mais dispositivos recebe um nível de qualidade de sensoriamento de acordo à definição descrita através da Equação 3.11.

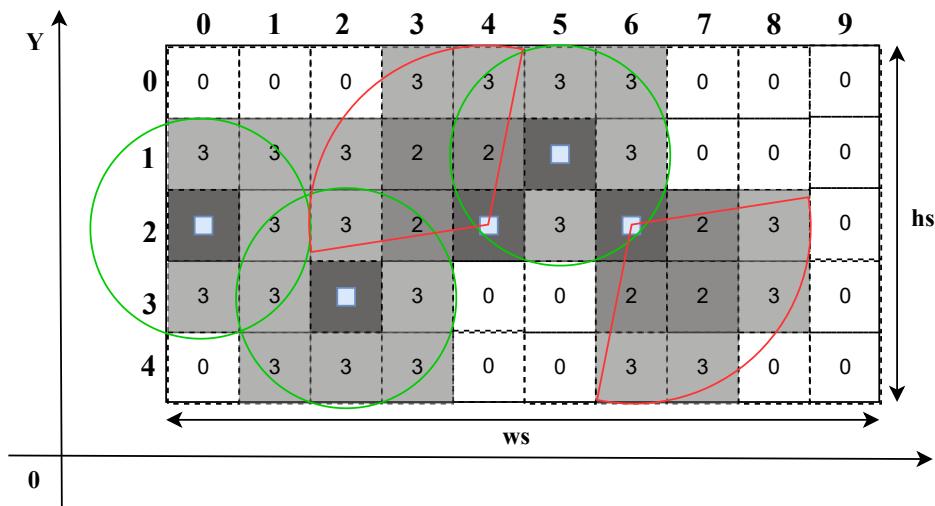


Figura 3.15: Resultado da aplicação das *MQS* sobre o cenário de exemplo.

Com a análise da Figura 3.15 verifica-se quantos AB estão sendo monitorados dentro de cada nível de qualidade: alto, médio e baixo. Aplicando as Equações 3.12a

a 3.12c temos que  $AB_{wHS} = 0,1562 = 15.62\%$ ,  $AB_{wHM} = 0,1875 = 18.75\%$  e  $AB_{wHL} = 0.6562 = 65.62\%$ , pois 21 blocos estão em nível baixo de qualidade de sensoriamento, 6 em nível médio e 5 em alto. É possível notar que a expressão  $AB_{wLS} > AB_{wHS} \mid AB_{wLS} > AB_{wMS}$  para aplicação da penalidade foi atendida, o que resulta na aplicação do termo  $e^{-\alpha * AB_{wHS}}$  sobre o valor de  $AB_{wHS}$ . Então, o resultado final para  $f_3(s)$  pode ser visto na Equação 3.15.

$$f_3(s) = AB_{wHS} * e^{-\alpha * AB_{wHS}} = 0.1562 * e^{-0.5 \times 0.1562} \approx 0.144 \quad (3.15)$$

É importante lembrar que toda essa análise de posicionamento e qualidade das soluções geradas é realizada para cada nova solução encontrada pelo algoritmo genético. Nesse exemplo ilustrativo, as funções  $f_2(s)$  e  $f_3(s)$  apresentaram valores muito baixos, enquanto a função  $f_1(s)$  obteve um valor mais elevado. Isso demonstra que essa solução exibe uma característica significativa apenas no objetivo relacionado à cobertura, sendo ineficaz em relação aos outros critérios.

Com essa análise, finaliza-se a descrição da estratégia de posicionamento proposta neste estudo, que busca posicionar um conjunto de sensores dentro de áreas previamente delimitadas e com redes sem fio que proporcionam conectividade à RdI. No próximo capítulo, será descrito o processo e os resultados alcançados durante a etapa de definição dos parâmetros de configuração do algoritmo NSGA-II, a avaliação do método TOPSIS como mecanismo de seleção de soluções dentro de uma FP, bem como a análise do posicionamento efetuado sobre um conjunto vasto e diversificado de cenários de teste.

# **Capítulo 4**

## **Resultados**

Após a elucidação, no Capítulo 2, dos conceitos fundamentais para a compreensão e o desenvolvimento desta pesquisa, e tendo sido estabelecidos no capítulo anterior os elementos que compõem a estratégia de posicionamento baseada na utilização do algoritmo NSGA-II para a otimização dos critérios de cobertura, conectividade e qualidade de sensoriamento nas RSSF, este capítulo dedica-se à apresentação e análise dos experimentos realizados. O objetivo é validar a metodologia proposta por meio da avaliação de seu desempenho em diferentes cenários simulados, permitindo observar sua eficácia frente às restrições e condições previamente definidas.

Para tanto, primeiramente, será descrita a análise do método Grid Search, usado para definir os hiperparâmetros e operadores genéticos a serem aplicados no algoritmo NSGA-II, aumentando sua eficiência na busca por soluções ótimas. Logo após, serão descritos os procedimentos e resultados experimentais adotados durante a etapa de análise do método proposto sobre um conjunto diversificado de cenários de implantação, de modo a fornecer um exame crítico e fundamentado sobre a aplicabilidade e eficácia da solução desenvolvida.

### **4.1 Aplicação da estratégia Grid Search**

Antes que se iniciasse a análise de diferentes cenários de teste, foi preciso estabelecer corretamente os hiperparâmetros do algoritmo NSGA-II de forma a proporcionar o aumento do seu desempenho relacionado à geração de soluções ótimas da Fronteira de Pareto, assim aprimorando a pesquisa por soluções dentro do espaço de busca.

Para alcançar esse objetivo, foi utilizada uma abordagem sistemática conhecida como *Grid Search*. Essa abordagem consiste em definir um conjunto de valores para cada hiperparâmetro e, em seguida, avaliar a combinação desses diferentes valores a partir dos resultados produzidos pela execução do algoritmo genético. Com essa avaliação, é possível escolher qual combinação proporciona melhor performance ao algoritmo genético. Essa técnica encontra-se consolidada na literatura e, tanto sua análise

quanto a utilização em projetos de pesquisa, podem ser encontradas em diferentes trabalhos como de Bergstra et al. (2011), Khalesian e Delavar (2016), Rangel et al. (2019) e Yang e Shami (2020).

Durante a aplicação do *Grid Search* foram avaliados diferentes valores para os seguintes hiperparâmetros: tamanho da população (TP), taxa de cruzamento (TC), taxa de mutação (TM) e o número de gerações (NG). Na Tabela 4.1 é apresentado o conjunto de valores definidos empiricamente para cada hiperparâmetro. Realizando a combinação desses valores, temos um conjunto de 54 possíveis combinações que pode ser visto na Tabela 4.2.

Tabela 4.1: Conjunto de valores para os hiperparâmetros.

Parâmetros	Valores
<b>TP</b>	100,200
<b>TC</b>	70%, 80%, 90%
<b>TM</b>	10%, 20%, 30%
<b>NG</b>	250,500,750

Tendo em vista que na literatura não se encontra um consenso de quais operadores genéticos melhor se adéquam ao problema de posicionamento, então, além de realizar a variação dos hiperparâmetros, também foi decidido incluir na análise do *Grid Search*, o uso de dois métodos de cruzamento – *Simulated Binary Crossover (SBX)* e *Two-Point Crossover* – conjuntamente com o método *Polynomial Mutation* e Seleção por Torneio Binário, formando assim duas versões de configuração para a execução do NSGA-II.

Como apenas um método de mutação e seleção está sendo incluído na análise, logo, o impacto maior dessa alternância dos operadores está sobre os dois métodos de cruzamento. Além disso, devido ao aumento de combinações e custo computacional gerados pela inclusão da análise de diferentes métodos de mutação e seleção, então apenas a etapa de cruzamento foi escolhida para variação.

A Tabela 4.3 apresenta a configuração dos operadores para cada versão. Como a solução para o problema proposto é modelada por meio de cromossomos representados por codificação do tipo inteiro, então o método *SBX* é referenciado como *Integer SBX*, pois consiste em uma adaptação do método original de forma a permitir a manipulação das soluções geradas de acordo à codificação estabelecida.

Após o estabelecimento dos operadores genéticos e da combinação de hiperparâmetros, foi necessário definir um cenário de teste sobre o qual o algoritmo NSGA-II fosse executado. Inicialmente, utilizando o *software Matlab*, foi gerado o mapeamento de uma RdI que apresentasse aspectos ideais de conectividade em uma região urbana. Tais aspectos podem ser descritos como:

Tabela 4.2: Combinações de valores para os hiperparâmetros.

ID	TM	TC	TP	NG	ID	TM	TC	TP	NG	ID	TM	TC	TP	NG
<b>P1</b>	0.1	0.7	100	250	<b>P19</b>	0.2	0.7	100	250	<b>P37</b>	0.3	0.7	100	250
<b>P2</b>	0.1	0.7	100	500	<b>P20</b>	0.2	0.7	100	500	<b>P38</b>	0.3	0.7	100	500
<b>P3</b>	0.1	0.7	100	750	<b>P21</b>	0.2	0.7	100	750	<b>P39</b>	0.3	0.7	100	750
<b>P4</b>	0.1	0.7	200	250	<b>P22</b>	0.2	0.7	200	250	<b>P40</b>	0.3	0.7	200	250
<b>P5</b>	0.1	0.7	200	500	<b>P23</b>	0.2	0.7	200	500	<b>P41</b>	0.3	0.7	200	500
<b>P6</b>	0.1	0.7	200	750	<b>P24</b>	0.2	0.7	200	750	<b>P42</b>	0.3	0.7	200	750
<b>P7</b>	0.1	0.8	100	250	<b>P25</b>	0.2	0.8	100	250	<b>P43</b>	0.3	0.8	100	250
<b>P8</b>	0.1	0.8	100	500	<b>P26</b>	0.2	0.8	100	500	<b>P44</b>	0.3	0.8	100	500
<b>P9</b>	0.1	0.8	100	750	<b>P27</b>	0.2	0.8	100	750	<b>P45</b>	0.3	0.8	100	750
<b>P10</b>	0.1	0.8	200	250	<b>P28</b>	0.2	0.8	200	250	<b>P46</b>	0.3	0.8	200	250
<b>P11</b>	0.1	0.8	200	500	<b>P29</b>	0.2	0.8	200	500	<b>P47</b>	0.3	0.8	200	500
<b>P12</b>	0.1	0.8	200	750	<b>P30</b>	0.2	0.8	200	750	<b>P48</b>	0.3	0.8	200	750
<b>P13</b>	0.1	0.9	100	250	<b>P31</b>	0.2	0.9	100	250	<b>P49</b>	0.3	0.9	100	250
<b>P14</b>	0.1	0.9	100	500	<b>P32</b>	0.2	0.9	100	500	<b>P50</b>	0.3	0.9	100	500
<b>P15</b>	0.1	0.9	100	750	<b>P33</b>	0.2	0.9	100	750	<b>P51</b>	0.3	0.9	100	750
<b>P16</b>	0.1	0.9	200	250	<b>P34</b>	0.2	0.9	200	250	<b>P52</b>	0.3	0.9	200	250
<b>P17</b>	0.1	0.9	200	500	<b>P35</b>	0.2	0.9	200	500	<b>P53</b>	0.3	0.9	200	500
<b>P18</b>	0.1	0.9	200	750	<b>P36</b>	0.2	0.9	200	750	<b>P54</b>	0.3	0.9	200	750

- A RdI deve ser altamente conectada, ou seja, a taxa de área coberta por redes sem fio precisa ser alta;
- As redes deve estar uniformemente distribuídas ao longo da RdI;

A Figura 4.1 apresenta a RdI idealizada. Os dados paramétricos associados a essa RdI podem ser encontrados na Tabela 4.4. Na área delimitada, foram implantadas 1000 redes do tipo WiFi, 3 do tipo 4G e 28 do tipo LoRa, nas quais tais tipos foram previamente mencionados na Tabela 3.3 e usados anteriormente no exemplo da Figura 3.2. Os parâmetros utilizados no cálculo da métrica *DPConn* de forma a definir o nível de conectividade de cada AB foram os mesmos definidos nas Tabelas 3.1 e 3.2.

Tabela 4.3: Versões do NSGA-II em relação aos operadores genéticos.

Versão	Seleção	Mutação	Cruzamento
<b>V0</b>	Seleção por torneio	<i>Polynomial Mutation</i>	<i>Integer SBX</i>
<b>V1</b>	Seleção por torneio	<i>Polynomial Mutation</i>	<i>Two-Point Crossover</i>

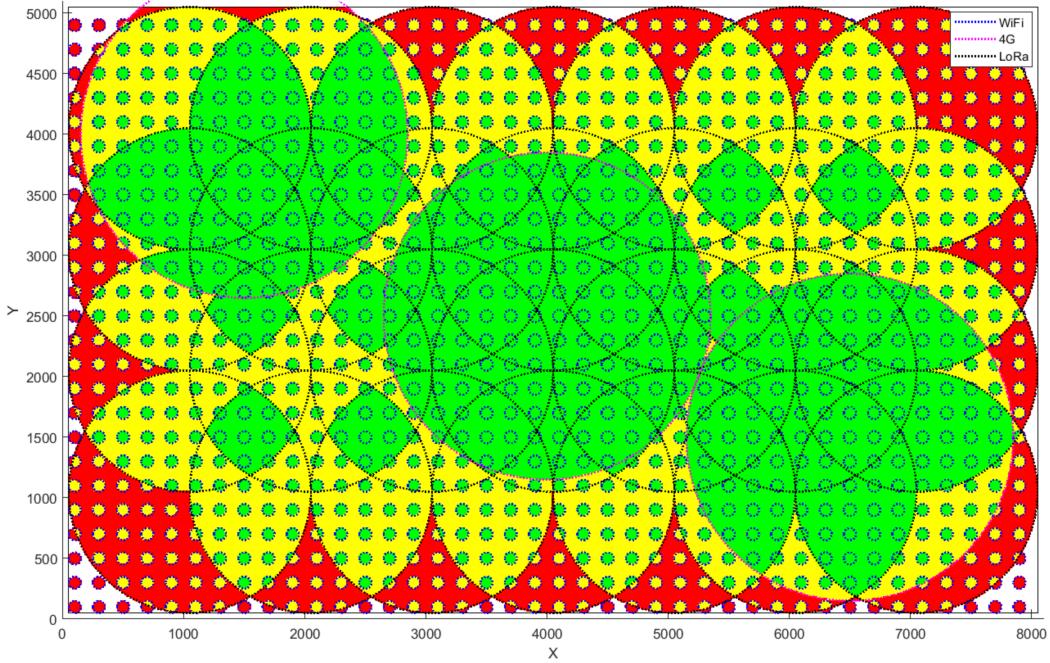


Figura 4.1: Representação da RdI gerada para aplicação da estratégia *Grid Search*.

Tabela 4.4: Dados paramétricos da RdI ideal apresentada na Figura 4.1.

$ws_{city}$	$hs_{city}$	$ws_{AB}$	$hs_{AB}$	$AB_{total}$	$nb_C$
8000m	5000m	15	15	177.489	171.961-(96.88%)

Observa-se que a RdI apresentada na Figura 4.1 está de acordo com os dois critérios anteriormente estabelecidos que definem uma área com aspectos ideais de conectividade. Pois, além de serem encontradas redes uniformemente distribuídas, a taxa de área coberta corresponde a 96.88%, caracterizando uma região altamente conectada.

Com a RdI estabelecida, se faz necessário definir as características do conjunto de dispositivos que serão implantados. As Tabelas 4.5 e 4.6 apresentam essas características de acordo com as variáveis para modelagem da região de cobertura (ver Figura 3.4), além da quantidade de dispositivos alocados, correspondente a 6000 sensores escalares e 2000 câmeras. A partir dessa definição, essas informações foram transcritas para o arquivo de configuração encontrado na Listagem 3. Elas permitirão gerar as respectivas máscaras de cobertura de acordo com a representação e metodologia descritas desde a Subseção 3.2.2 a 3.2.5.

De forma a ilustrar todo o processo do *Grid Search*, um fluxo de execução foi criado e pode ser visto na Figura 4.2. As etapas até aqui descritas fazem parte das ações da fase de configuração inicial. Com o processo de configuração finalizado, deu-se início à fase de processamento, que consiste, em um âmbito geral, no processo de geração e avaliação das soluções geradas pelo NSGA-II, conforme descrito na Seção 3.4.

Tabela 4.5: Parâmetros para os sensores escalares utilizados no processo de *Grid Search*.

Quantidade	$r_s$	$r_l$	$r_e$
3000	25m	12.5	18.75
3000	50m	25	37.5

Tabela 4.6: Parâmetros para as câmeras utilizadas no processo de *Grid Search*.

Quantidade	$r_s$	$r_l$	$r_e$	$\theta$
1000	150m	75	112.5	90°
1000	200m	100	150	135°

O algoritmo foi configurado de acordo com as combinações dos operadores genéticos (ver Tabela 4.3), e cada uma dessas versões foi submetida a todas as 54 possíveis combinações para os hiperparâmetros (ver Tabela 4.2). Isso significa que o número total de FP geradas é de 108, sendo 54 da versão V0 mais 54 da versão V1. Para cada execução do NSGA-II baseadas nas versões e hiperparâmetros, são calculadas as métricas de performance *HV*, *NSND* e *Time-C* descritas na Seção 2.7. Os valores são armazenados em um arquivo do tipo .csv e, posteriormente, usados durante o processo de avaliação das FP. Nesse processo avaliativo, como o problema de posicionamento proposto não possui uma FP ideal que possa ser utilizada como referência, logo, outras métricas não foram encontradas para apoiar essa análise.

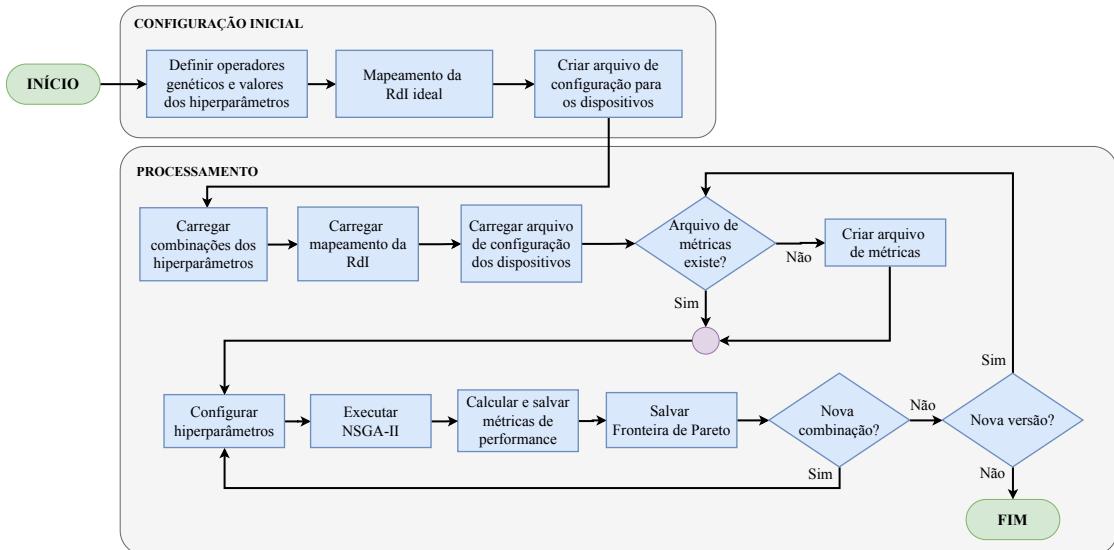


Figura 4.2: Fluxo de execução do método *Grid Search* sobre o processo de posicionamento.

Com a execução do algoritmo NSGA-II, foram geradas tanto as 108 FP, como tam-

```

1      {
2          "sensors": [
3              {
4                  "type": 1,
5                  "qtd": 3000,
6                  "radius": 25,
7                  "limitOfHighQuality": 12.5,
8                  "limitOfMediumQuality": 18.75
9              },
10             {
11                 "type": 1,
12                 "qtd": 3000,
13                 "radius": 50,
14                 "limitOfHighQuality": 25,
15                 "limitOfMediumQuality": 37.5
16             },
17             {
18                 "type": 2,
19                 "qtd": 1000,
20                 "radius": 150,
21                 "limitOfHighQuality": 75,
22                 "limitOfMediumQuality": 112.5,
23                 "viewing_angle": 90
24             },
25             {
26                 "type": 2,
27                 "qtd": 1000,
28                 "radius": 200,
29                 "limitOfHighQuality": 100,
30                 "limitOfMediumQuality": 150,
31                 "viewing_angle": 135
32             }
33         ]
34     }

```

Listing 3: Arquivo de configuração para os dispositivos implantados durante o processo de *Grid Search*.

bém os arquivos de métricas para as versões *V0* e *V1*. Nas Tabelas 4.9 e 4.10 é possível observar os valores calculados das métricas de performance para cada combinação de hiperparâmetros apresentada na Tabela 4.2 aplicada às duas diferentes versões.

Para a implementação e execução desse experimento, foi utilizado o *framework* JMetal na linguagem Java, que proporciona o código-fonte para diversos algoritmos multiobjetivos, incluindo o NSGA-II, além das métricas de performance mencionadas. Também foi usado um *notebook* Dell com arquitetura de 64 bits, 16GB de memória

RAM, processador Intel Core i5-1135G7 da 11º geração com *clock* de  $2.40GHz$  e sistema operacional Windows 11.

É importante ressaltar que as características da rede de sensores e da RdI considerada ideal, aplicadas nessa etapa de *Grid Search*, podem variar a depender do estudo ou da aplicação. Consequentemente, os resultados que serão apresentados em relação a esse processo estão diretamente associados às características dos elementos em questão; logo, eventualmente, a variação desses elementos implicaria na obtenção de diferentes parâmetros para o algoritmo genético. Por essa razão, é aconselhado a modificação e reexecução do processo caso os elementos indicados não se enquadrem nos objetivos da pesquisa.

Na próximas seções, serão apresentados os resultados da análise do método *Grid Search* sobre as FP produzidas a partir das versões e combinações dos hiperparâmetros. Assim, os operadores genéticos e hiperparâmetros definidos ao final desse processo serão aplicados em todos os outros cenários de posicionamento abordados neste estudo.

## 4.2 Análise das soluções geradas pela estratégia *Grid Search* para a versão V0.

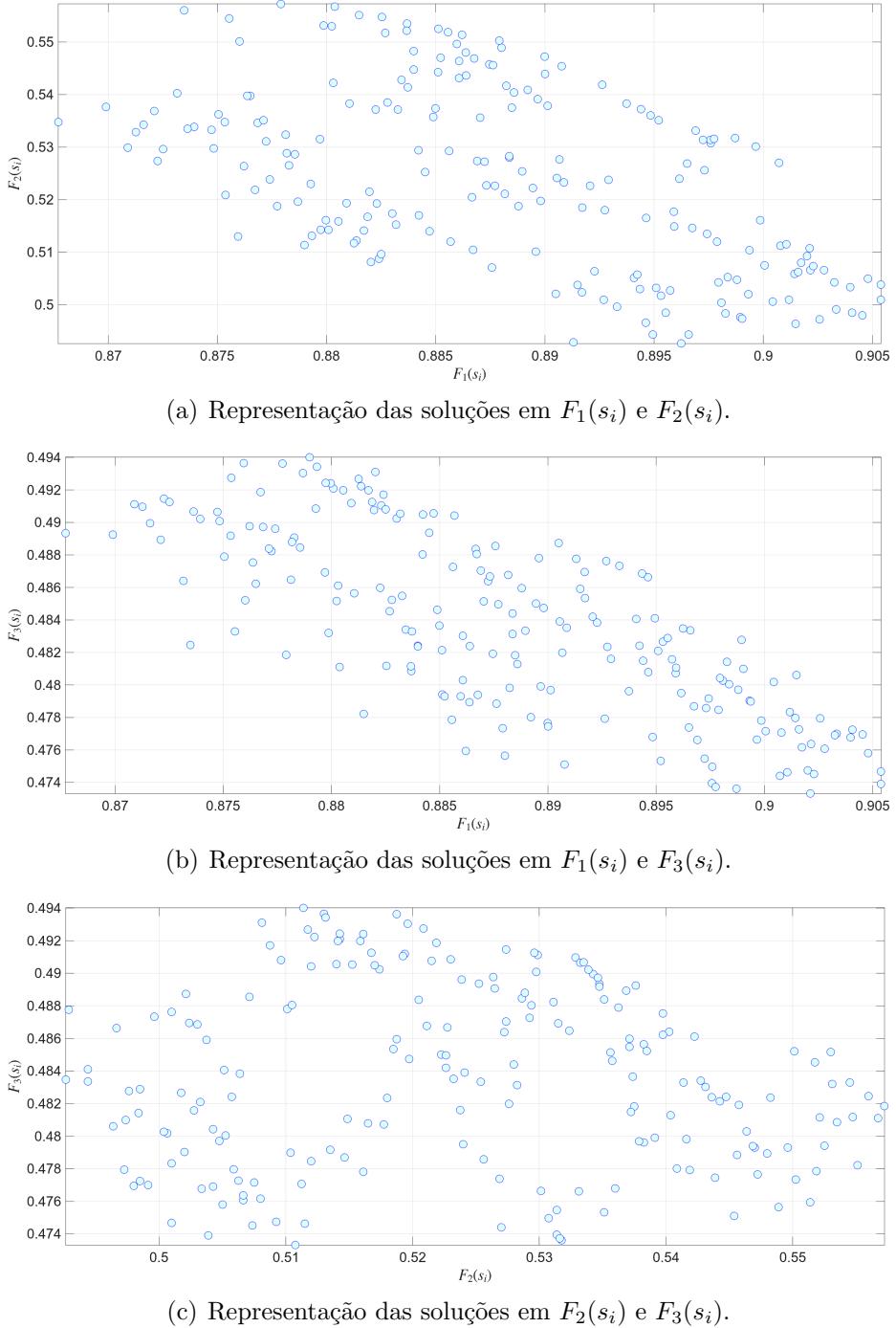
Na Tabela 4.9, temos os valores das métricas de performance para a versão *V0*. Analisando esses valores, algumas conclusões foram alcançadas. Primeiramente, as FP produzidas pelas combinações dos hiperparâmetros estão cobrindo, aproximadamente, o mesmo volume dentro do espaço de busca, pois os valores de *HV* estão bem próximos uns dos outros, sendo o maior valor gerado pela FP da combinação *P12*. Na Figura 4.3 é apresentada a disposição das soluções dessa fronteira em relação aos valores de *fitness*.

Em relação à métrica *NSND*, as combinações *P4*, *P5*, *P6*, *P11*, *P12*, *P18*, *P23*, *P24* e *P30* apresentam o maior número de soluções, com  $NSND = 200$ , e percebe-se que a combinação *P12*, além de se destacar pelo valor de *HV*, também está entre as melhores dessa métrica. Contudo, como os valores de *HV* para essas combinações estão muito próximos e possuem o mesmo valor de *NSND*, ainda não é possível destacar qual produziu melhores soluções. Por causa disso, deu-se seguimento à análise a partir das métricas *Time-C* e *SC*.

Sendo o hiperparâmetro *NG* utilizado como critério de parada, observou-se que as combinações *P4*, *P5*, *P6*, *P11*, *P12*, *P18*, *P23*, *P24* e *P30* foram executadas a partir da configuração de diferentes critérios de parada. Com isso, levanta-se a seguinte hipótese:

- Quanto maior o valor de *NG* maior será o valor de *Time-C*;

Separando as combinações mencionadas em grupos de acordo ao valor de *NG*, temos que:

Figura 4.3: Fronteira de Pareto produzida pela combinação  $P12$  na versão  $V0$ .

- $P4$  foi executada a partir de  $NG = 250$ ;
- $P23$ ,  $P11$  e  $P5$  foram executadas a partir de  $NG = 500$ ;
- $P24$ ,  $P30$ ,  $P18$ ,  $P6$  e  $P12$  foram executadas a partir de  $NG = 750$ ;

A partir da observação da métrica *Time-C* calculada para cada uma das combinações mencionadas acima, conclui-se que a hipótese levantada é verdadeira. Na Tabela 4.7 é visto que o tempo médio de *Time-C* para o grupo com  $NG = 750$  é superior aos outros grupos, bem como o grupo com  $NG = 500$  é superior ao grupo com  $NG = 250$  que é formado apenas pela combinação  $P4$ . Por causa da natureza desse comportamento, não faz sentido realizar um *ranking* com os valores de *Time-C*, pois o grupo com  $NG = 250$  sempre se destacaria por apresentar os menores valores. Diante disso, a partir das combinações que formam os grupos da Tabela 4.7, foi decidido capturar a combinação que produziu o menor valor de *Time-C* em cada grupo e aplicar a métrica *SC* com o objetivo de verificar entre eles o nível de dominância das soluções.

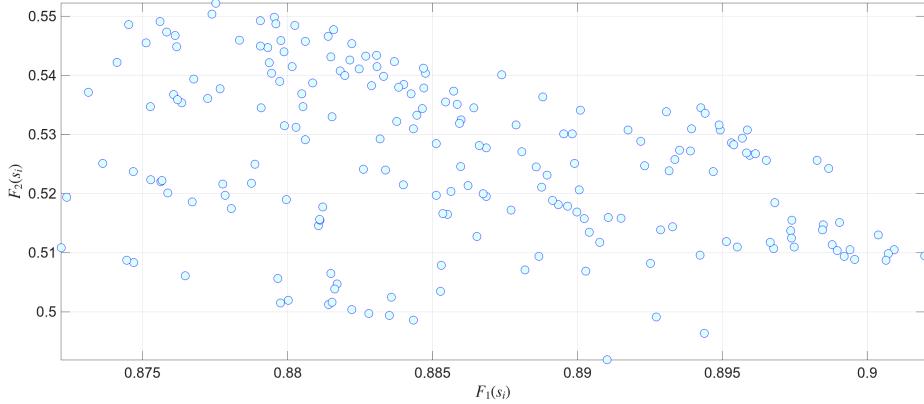
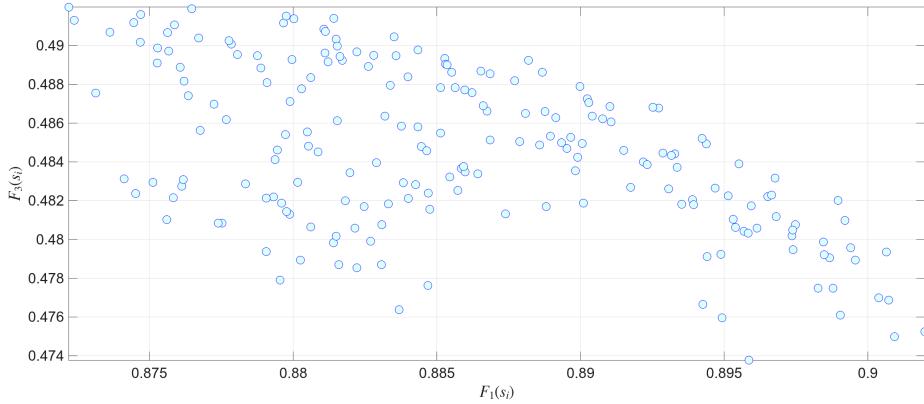
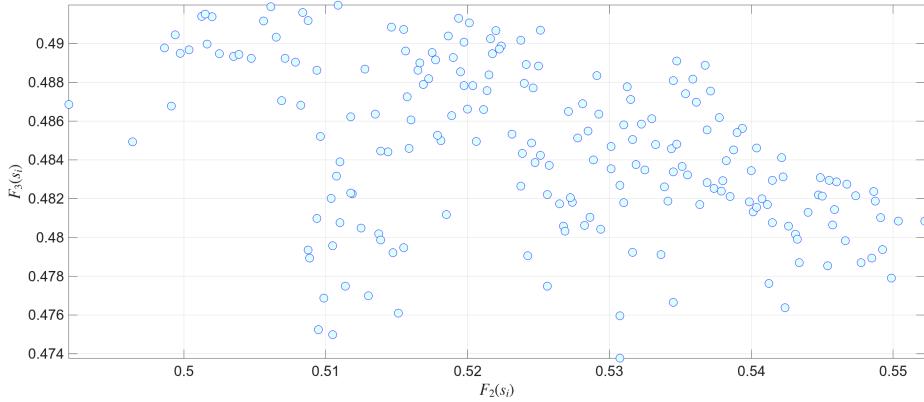
Tabela 4.7: Valor médio da métrica *Time-C* para as combinações  $P4$ ,  $P5$ ,  $P6$ ,  $P11$ ,  $P12$ ,  $P18$ ,  $P23$ ,  $P24$  e  $P30$  separadas por grupos de acordo ao valor de  $NG$ .

<b>NG</b>	<b>Time-C (min)</b>
250	18.4201
500	29.8722
750	45.6206

Para o grupo com  $NG = 250$ , a combinação  $P4$  foi selecionada. Para  $NG = 500$ , a combinação  $P11$ , e para  $NG = 750$ , a combinação  $P6$  foi selecionada. Nas Figuras 4.4, 4.5 e 4.6 é possível observar a disposição das soluções nas respectivas fronteiras geradas a partir dessas 3 combinações. Na Tabela 4.8 são apresentados os valores calculados da métrica *SC* para o agrupamento par a par das FP produzidas pela aplicação das combinações  $P4$ ,  $P6$  e  $P11$  durante a etapa de execução do NSGA-II.

A partir da fundamentação descrita na Subseção 2.7.2 e dos valores da Tabela 4.8, podemos observar a característica de um *ranking* de dominância, pois a  $FP_6$  apresentou superioridade em relação a  $FP_4$  e  $FP_{11}$ , destacando-se em 1<sup>a</sup> lugar. Seguida da  $FP_{11}$  que apresentou superioridade em relação a  $FP_4$ , logo, ficando em 2<sup>a</sup> lugar, e por fim, temos a  $FP_4$  em 3<sup>a</sup> lugar. Então, a  $FP_6$  destacou-se como a melhor dentre as três fronteiras selecionadas.

Entretanto, existe um último ponto a ser levado em consideração. Observou-se que a  $FP_{12}$  produzida pela combinação  $P12$ , mencionada inicialmente, se destacou na métrica *HV*, possui o mesmo *NSND* da  $FP_6$  e as duas foram produzidas a partir de  $NG = 750$ . Devido a essas características, foi aplicada a métrica *SC* sobre essas duas fronteiras, assim, destacando a melhor combinação para a versão *V0*.

(a) Representação das soluções em  $F_1(s_i)$  e  $F_2(s_i)$ .(b) Representação das soluções em  $F_1(s_i)$  e  $F_3(s_i)$ .(c) Representação das soluções em  $F_2(s_i)$  e  $F_3(s_i)$ .Figura 4.4: Fronteira de Pareto produzida pela combinação  $P4$  na versão  $V0$ .

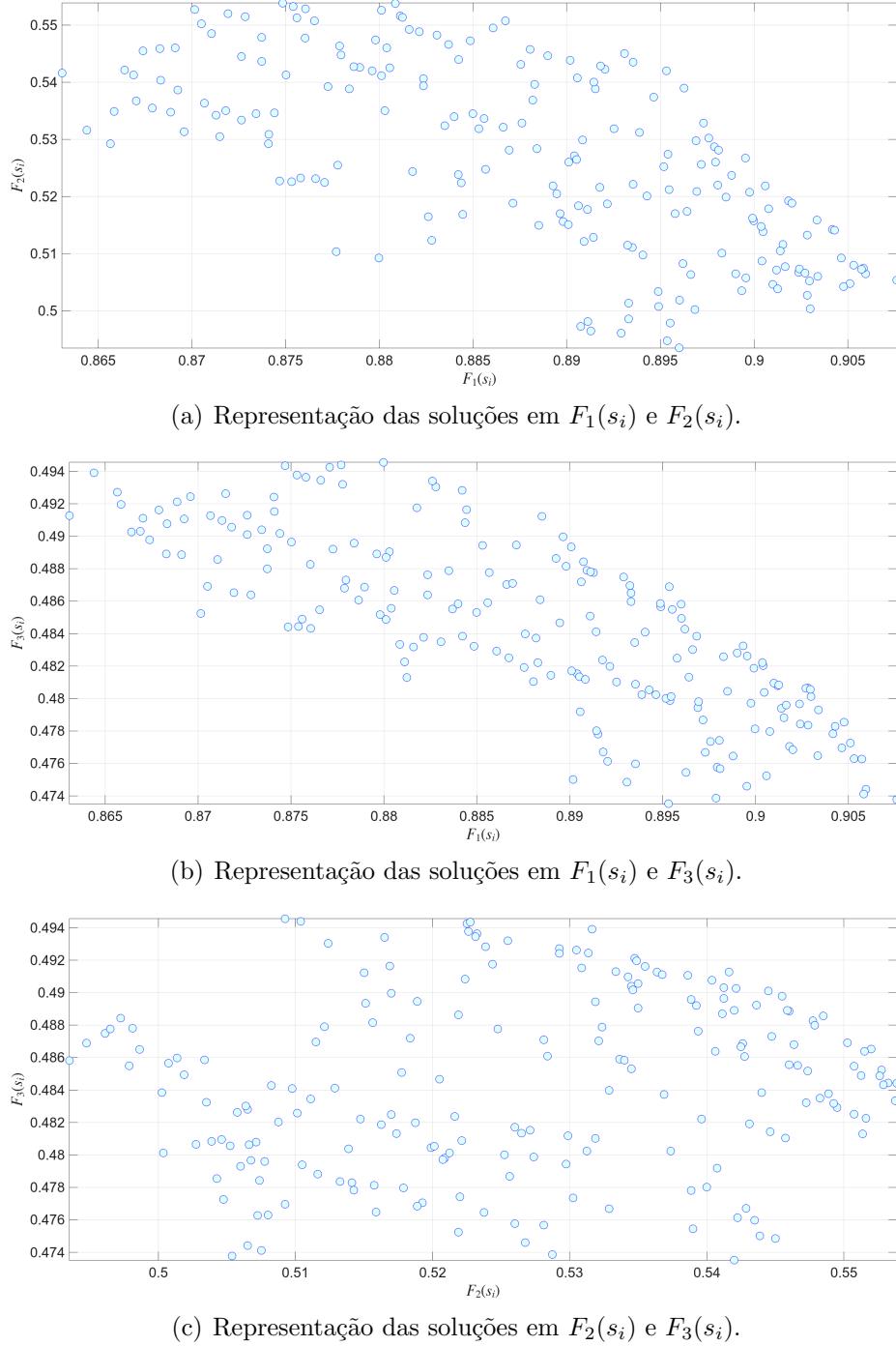


Figura 4.5: Fronteira de Pareto produzida pela combinação  $P6$  na versão  $V0$ .

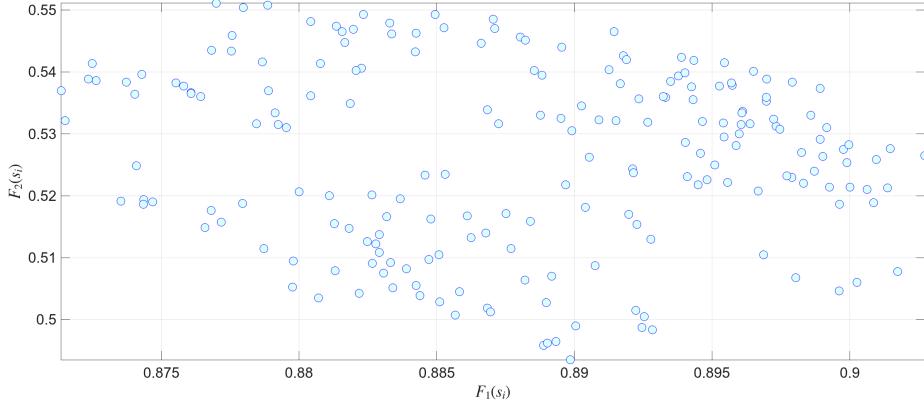
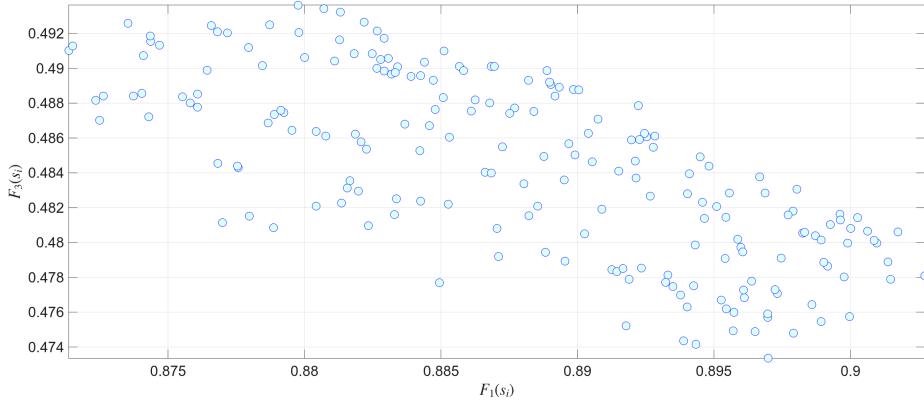
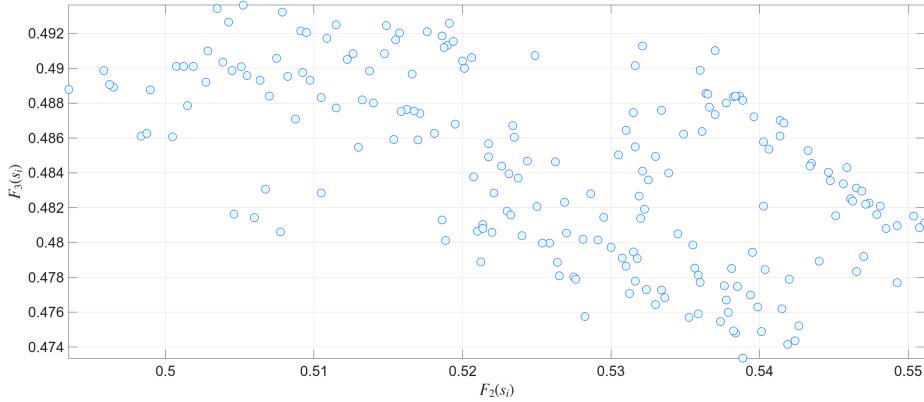
(a) Representação das soluções em  $F_1(s_i)$  e  $F_2(s_i)$ .(b) Representação das soluções em  $F_1(s_i)$  e  $F_3(s_i)$ .(c) Representação das soluções em  $F_2(s_i)$  e  $F_3(s_i)$ .

Figura 4.6: Fronteira de Pareto produzida pela combinação P11 na versão V0.

Tabela 4.8: Métrica  $SC$  para o agrupamento par a par das combinações  $P4$ ,  $P6$  e  $P11$ .

Agrupamento	Valor
$SC(FP_4, FP_6)$	0.085
$SC(FP_6, FP_4)$	0.79
$SC(FP_4, FP_{11})$	0.085
$SC(FP_{11}, FP_4)$	0.64
$SC(FP_6, FP_{11})$	0.535
$SC(FP_{11}, FP_6)$	0.23

Então, sendo  $SC(FP_{12}, FP_6) = 0.13 < SC(FP_6, FP_{12}) = 0.645$ , logo, mesmo que a combinação  $P12$  tenha gerado uma FP com melhor  $HV$ , a FP gerada por  $P6$  apresenta melhores resultados, tanto no tempo computacional - lembrando que as duas estão no grupo de  $NG = 750$  - quanto na relação de dominância. Por essas razões, a combinação de hiperparâmetros utilizada por  $P6$  se destacou como a melhor na versão  $V0$ .

Tabela 4.9: Versão V0 - métricas de performance calculadas a partir das FP.

ID	HV	NSND	Time-C (min)	ID	HV	NSND	Time-C (min)	ID	HV	NSND	Time-C (min)
P1	0.2414	100	11.3633	P19	0.2342	100	9.5706	P37	0.2333	100	7.0508
P2	0.2445	100	22.1750	P20	0.2389	100	14.2255	P38	0.2347	100	21.6299
P3	0.2476	100	30.9315	P21	0.2408	100	20.5560	P39	0.2376	100	31.7503
P4	0.2446	200	18.4201	P22	0.2370	158	16.1495	P40	0.2338	118	20.0828
P5	0.2475	200	30.2977	P23	0.2406	200	29.8036	P41	0.2367	191	28.4232
P6	0.2481	200	42.8208	P24	0.2400	200	43.2833	P42	0.2365	157	43.3865
P7	0.2399	100	6.8839	P25	0.2364	100	6.7891	P43	0.2335	96	7.2034
P8	0.2468	100	13.3188	P26	0.2375	100	13.8753	P44	0.2345	100	14.5701
P9	0.2443	100	20.7049	P27	0.2394	100	21.1745	P45	0.2345	100	20.6281
P10	0.2405	197	14.6258	P28	0.2367	149	14.6648	P46	0.2345	109	15.3534
P11	0.2452	200	29.5154	P29	0.2392	198	28.6763	P47	0.2370	172	32.0247
P12	0.2486	200	43.2198	P30	0.2418	200	43.1523	P48	0.2370	154	44.3797
P13	0.2391	100	7.0424	P31	0.2383	100	7.0706	P49	0.2325	83	6.9435
P14	0.2436	100	14.0049	P32	0.2380	100	14.1227	P50	0.2328	95	14.0458
P15	0.2459	100	20.9854	P33	0.2384	100	20.3729	P51	0.2386	100	21.5732
P16	0.2391	199	14.4911	P34	0.2390	150	14.5339	P52	0.2328	100	15.1625
P17	0.2447	199	28.9333	P35	0.2379	131	30.0143	P53	0.2380	160	29.5771
P18	0.2456	200	55.6276	P36	0.2424	196	46.9776	P54	0.2380	159	44.4674

### 4.3 Análise das soluções geradas pela estratégia *Grid Search* para a versão V1.

Na Tabela 4.10, temos os valores das métricas de performance para a versão V1. Primeiramente, realizando um *ranking* dos valores obtidos para a métrica *HV*, percebe-se que a combinação *P18* apresentou maior resultado. Na Figura 4.7 é apresentada a FP produzida por essa combinação, e nota-se que as soluções estão bem próximas umas das outras. Além disso, notou-se o mesmo comportamento anterior visto na versão *V0* que, devido à proximidade dos valores de *HV*, as FP produzidas pela versão *V1* estão cobrindo, aproximadamente, o mesmo volume dentro do espaço de busca.

Em relação à métrica *NSND*, temos um conjunto variado do número de soluções geradas em cada fronteira, diferente dos resultados produzidos na versão *V0*. Essa variação demonstra que as combinações que geraram maior *NSND* proporcionaram maior característica de diversidade. Realizando um *ranking* dessa métrica, temos que a combinação *P17* se destaca com  $NSND = 87$ , contudo, a *P18* vem em seguida com  $NSND = 81$ , sendo a diferença de apenas 6 soluções. A Figura 4.8 apresenta a FP produzida pela combinação *P17*.

Até o momento, constata-se o destaque de diferentes combinações nas duas primeiras métricas. Com isso, analisando os hiperparâmetros associados às combinações *P18* e *P17*, é possível observar que ambas foram executadas com os mesmos valores para *TM*, *TC* e *TP*, tendo divergência apenas no valor para *NG*. A combinação *P18* foi executada com  $NG = 750$  e *P17* com  $NG = 500$ . Ou seja, o algoritmo NSGA-II avaliou mais soluções candidatas no uso da combinação *P18* do que a *P17*.

Analizando o tempo computacional para essas duas combinações, a hipótese levantada anteriormente continua sendo verdadeira, pois *P18* apresentou um valor de  $Time - C = 45.8956min$ , enquanto *P17*, um valor de  $Time - C = 27.8518min$ . Por essa razão, não podemos definir a melhor combinação apenas observando o tempo computacional, tendo em vista que essa diferença está relacionada aos valores aplicados a *NG*.

Diante disso, foi aplicada a métrica *SC* com o objetivo de verificar qual das duas combinações apresenta maior taxa de dominância em relação à outra. Então, sendo  $SC(FP_{17}, FP_{18}) = 0.210 < SC(FP_{18}, FP_{17}) = 0.690$ , a FP produzida por *P18* apresenta maior dominância em relação à FP produzida por *P17*.

Consequentemente, diante de todas essas observações, é possível concluir que, mesmo que a FP gerada por *P17* apresente 6 soluções a mais em relação à FP gerada por *P18*, e também tenha sido gerada em um menor tempo computacional devido ao valor de *NG*, esses fatores não proporcionaram base suficiente para definir a melhor combinação. Contudo, com a FP de *P18* apresentando melhor valor de *HV* e maior taxa na relação de dominância, foi estabelecido que essa é a melhor combinação para a versão *V1*.

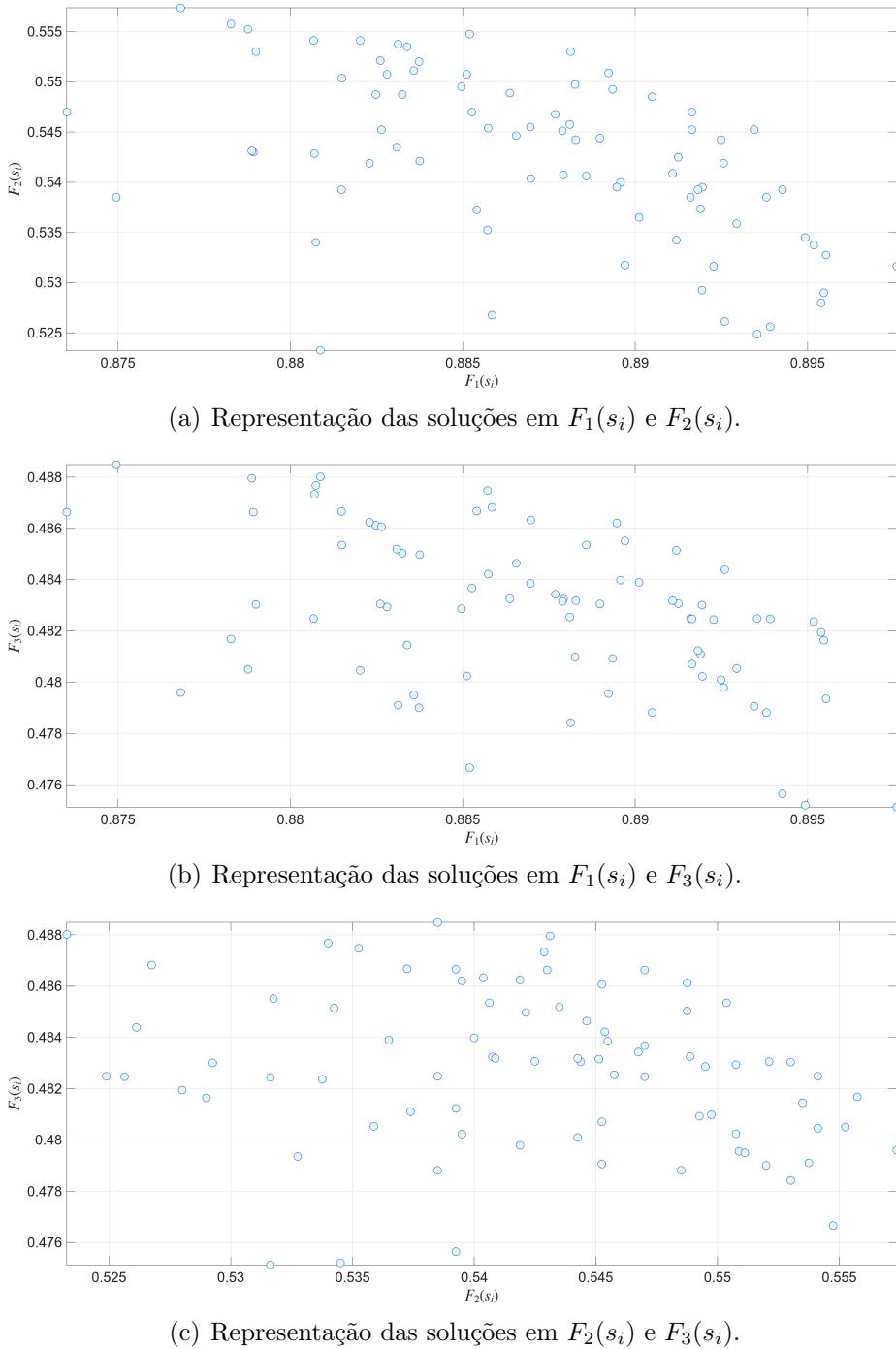


Figura 4.7: Fronteira de Pareto produzida pela combinação P18 na versão V1.

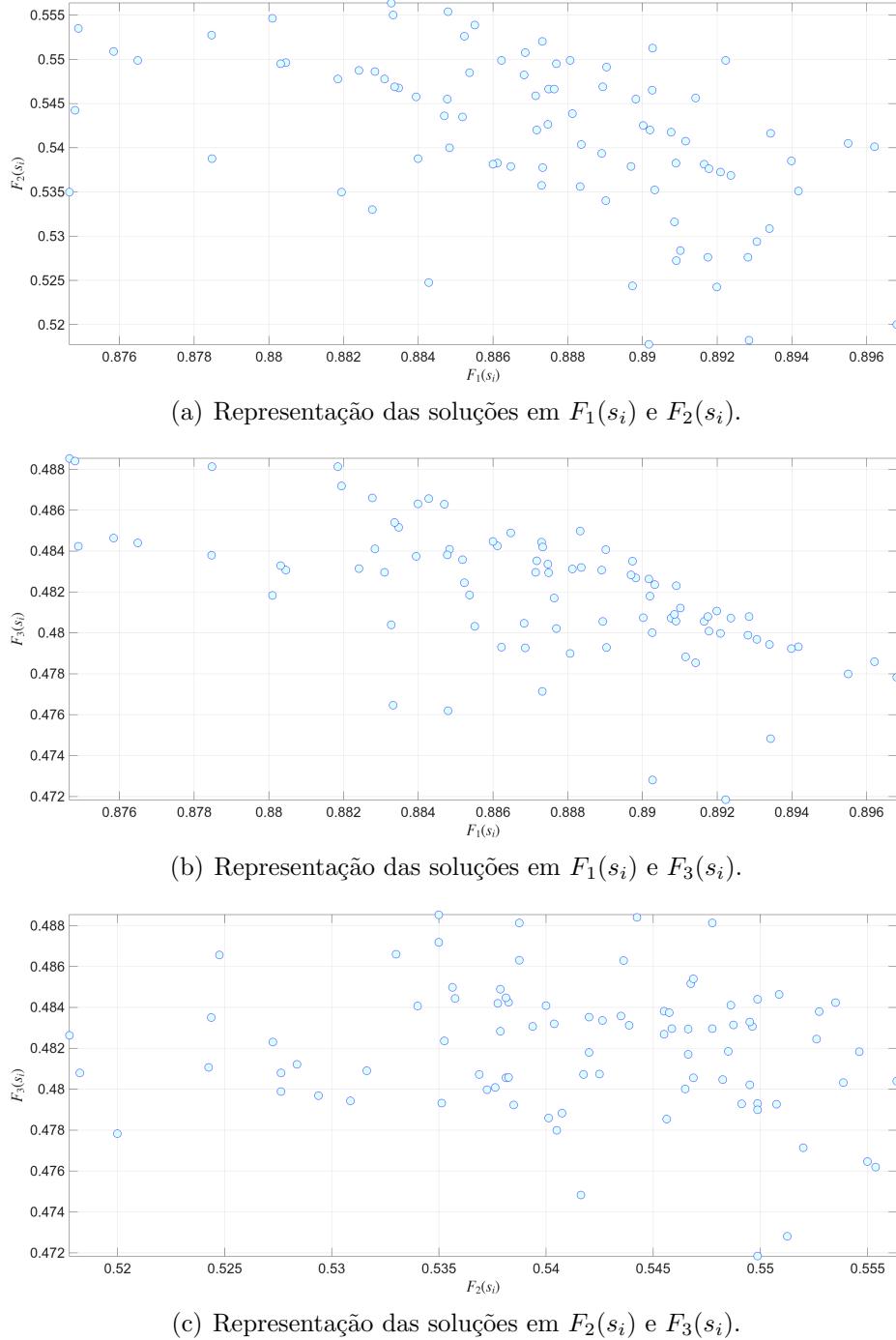
Figura 4.8: Fronteira de Pareto produzida pela combinação  $P17$  na versão  $V1$ .

Tabela 4.10: Versão V1 - métricas de performance calculadas a partir das FP.

ID	HV	NSND	Time-C (min)	ID	HV	NSND	Time-C (min)	ID	HV	NSND	Time-C (min)
P1	0.2304	40	7.1820	P19	0.2256	18	7.0062	P37	0.2227	27	16.7971
P2	0.2308	15	13.8234	P20	0.2273	33	14.2102	P38	0.2259	43	33.6745
P3	0.2341	35	20.7732	P21	0.2256	46	20.9841	P39	0.2255	22	49.2654
P4	0.2356	46	13.9906	P22	0.2279	33	14.3240	P40	0.2267	24	23.3299
P5	0.2336	56	28.0961	P23	0.2301	44	28.7404	P41	0.2257	44	31.0086
P6	0.2348	55	46.0281	P24	0.2292	53	43.3445	P42	0.2276	38	46.4513
P7	0.2345	34	6.9557	P25	0.2298	38	7.0786	P43	0.2279	23	7.4130
P8	0.2364	45	13.8682	P26	0.2285	35	14.0776	P44	0.2254	29	14.8128
P9	0.2338	43	27.4737	P27	0.2309	44	21.3339	P45	0.2276	25	22.2135
P10	0.2372	64	32.8307	P28	0.2312	42	14.4281	P46	0.2280	42	16.4047
P11	0.2382	66	56.3372	P29	0.2310	62	28.7292	P47	0.2271	45	32.6797
P12	0.2389	73	42.1703	P30	0.2321	43	42.8919	P48	0.2285	21	46.0766
P13	0.2385	61	6.9763	P31	0.2336	39	7.0773	P49	0.2302	35	7.4536
P14	0.2389	54	13.9951	P32	0.2349	45	14.2487	P50	0.2285	45	14.6232
P15	0.2386	65	21.0526	P33	0.2322	42	21.3964	P51	0.2279	38	22.0349
P16	0.2434	81	14.2203	P34	0.2356	47	14.3719	P52	0.2299	42	15.2690
P17	0.2436	87	27.8518	P35	0.2340	69	31.5206	P53	0.2334	44	30.2049
P18	0.2442	81	45.8956	P36	0.2332	37	51.7766	P54	0.2314	66	45.1409

## 4.4 Avaliação final do método *Grid Search*

Com a execução e análise dos resultados produzidos pelo algoritmo NSGA-II sobre as versões  $V0$  e  $V1$  em conjunto com as 54 combinações de hiperparâmetros, conclui-se que, a partir das métricas avaliadas:

- Na versão  $V0$  a combinação  $P6$  apresentou melhores resultados;
- Na versão  $V1$  a combinação  $P18$  apresentou melhores resultados;

Comparando através da Tabela 4.2, os valores dos hiperparâmetros dessas duas combinações, nota-se que a única diferença está na taxa de cruzamento, com  $P6$  definida com  $TC = 0.7$  e  $P18$  com  $TC = 0.9$ . Além disso,  $P6$  apresentou maior diversidade de soluções em comparação a  $P18$ , pois seus valores de  $NSND$  correspondem, respectivamente, a  $NSND = 200$  e  $NSND = 81$ . Como todas as combinações de hiperparâmetros executadas sobre a versão  $V1$  produziram um número de soluções inferior ao valor de TP, acredita-se que esse comportamento esteja vinculado ao operador de cruzamento *Two-Point Crossover*, tendo em vista que o operador de mutação e seleção são os mesmos nas duas versões.

Observando as outras métricas, também para essas combinações, observa-se que as duas FP geradas obtiveram valores muito próximos de  $HV$  e  $Time-C$ . Diante desses fatores, foi decidido utilizar a métrica  $SC$  para definir qual combinação apresenta melhor taxa de dominância. Realizado o cálculo, obteve-se que  $SC(V0_{FP_6}, V1_{FP_{18}}) = 0.123 < SC(V1_{FP_{18}}, V0_{FP_6}) = 0.215$ , concluindo que  $P18$  executado sobre a versão  $V1$  produziu melhores soluções, mesmo sendo em menor quantidade.

Desta forma, finalizou-se o processo de *Grid Search*, sendo definido que os operadores genéticos *Polynomial Mutation*, *Two-Point Crossover* e Seleção por Torneio Binário seriam utilizados em todos os outros cenários avaliados ao longo desta pesquisa, em conjunto com os seguintes valores de hiperparâmetros associados à combinação  $P18$ :

- Taxa de Mutação: 0.1;
- Taxa de Cruzamento: 0.9;
- Tamanho da População: 200;
- Número de Gerações (critério de parada): 750;

Com o objetivo de avaliar a qualidade do posicionamento efetuado pelo algoritmo NSGA-II a partir dos operadores genéticos e hiperparâmetros escolhidos, foi selecionada uma solução presente na respectiva  $FP_{18}$ . Analisando a disposição das soluções na Figura 4.7, buscou-se selecionar aquela que apresentasse maior equilíbrio entre as três funções de *fitness*. Diante disso, foi escolhida a  $44^\circ$  solução gerada na  $FP_{18}$ , com  $F_1(s) = 0.88792$ ,  $F_2(s) = 0.54075$  e  $F_3(s) = 0.48324$ .

Como forma de avaliar o posicionamento a partir do processamento da solução escolhida, foram definidos os seguintes aspectos a serem examinados: disposição dos

dispositivos ao longo da RdI; taxa de cobertura; níveis de sobreposição das áreas monitoradas; e níveis de qualidade de sensoriamento. Na Figura 4.9, encontra-se a implantação dos dispositivos de monitoramento da Listagem 3 sobre a RdI ideal. Os pontos em cor laranja representam os sensores escalares, enquanto os pontos em azul, as câmeras. Nota-se que o NSGA-II conseguiu dispersar satisfatoriamente os 8000 dispositivos configurados, abrangendo assim toda a área da RdI.

A partir desse posicionamento, 54.075% dos dispositivos foram implantados em regiões de alta conectividade, 30.925% em regiões de média e 15% em regiões de baixa conectividade. Isso mostra que a expressão  $Sr_{inH} > Sr_{inM} > Sr_{inL}$  descrita na Subseção 3.3.2 foi alcançada, garantindo que mais da metade dos dispositivos possua um nível robusto de conectividade de forma a realizar suas tarefas de comunicação com eficiência, segurança e integridade dos dados.

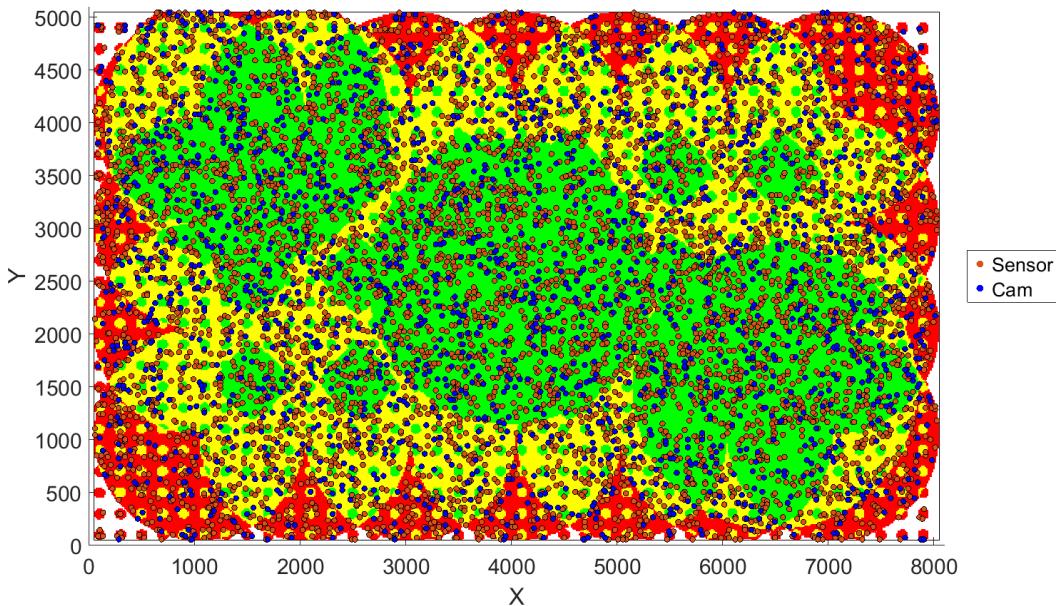


Figura 4.9: Cenário Ideal (*Grid Search*): disposição dos dispositivos ao longo da RdI.

Levando em consideração que, segundo Fei et al. (2017), a cobertura de uma RSSF é o aspecto mais importante a ser avaliado, podemos dizer que, especificamente para este cenário, o posicionamento efetuado alcançou um bom resultado, pois 89.55% da área pertencente à RdI está sendo monitorada. Isso pode ser visto na Figura 4.10. As regiões em cor azul representam áreas que possuem conectividade, seja em nível alto, médio ou baixo, mas não estão sendo monitoradas por nenhum dispositivo, enquanto as pequenas regiões em branco nos limiares da RdI são áreas sem conexão e sem monitoramento.

Também foi analisado os níveis de sobreposição em cada AB da RdI. Na Figura 4.11 encontra-se a dispersão desses níveis ao longo da área monitorada. Foi verificado que 25.130% dos ABs estão sendo monitorados 1 dispositivo; 28.732% por 2 dispo-

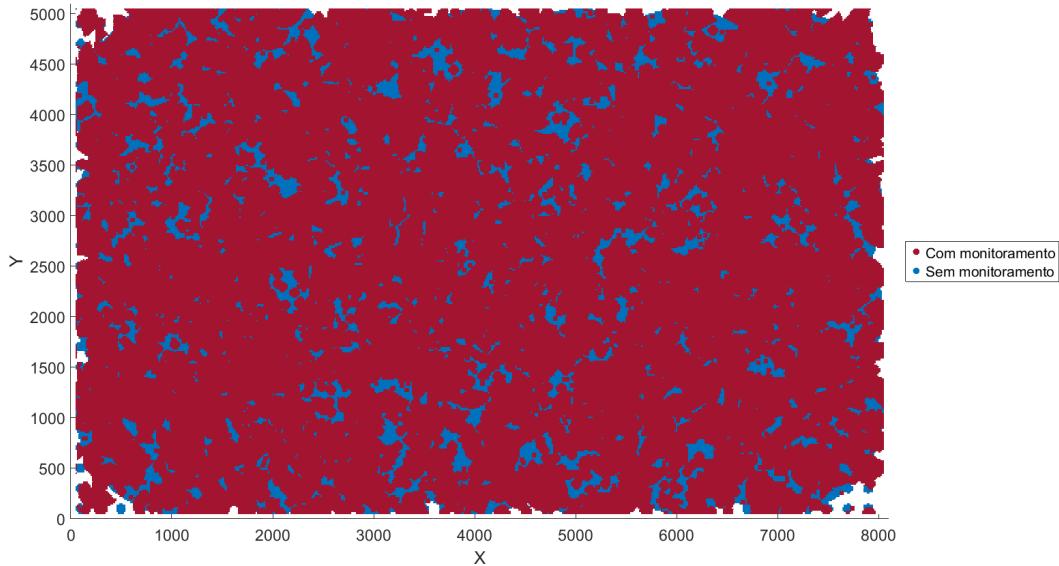


Figura 4.10: Cenário Ideal (*Grid Search*): Taxa de cobertura.

sitivos; 22.946% por 3 dispositivos; e 23.192% por mais de 3. Devido à proximidade das taxas, percebe-se que, sem prejudicar a taxa de cobertura, o NSGA-II conseguiu alcançar uniformidade para o número de dispositivos cobrindo cada AB. Implementar diferentes dispositivos monitorando o mesmo bloco pode favorecer os níveis de qualidade de sensoriamento, pois, de acordo com a Equação 3.11, esse nível está associado à região de cobertura de maior qualidade sobreposta ao respectivo bloco.

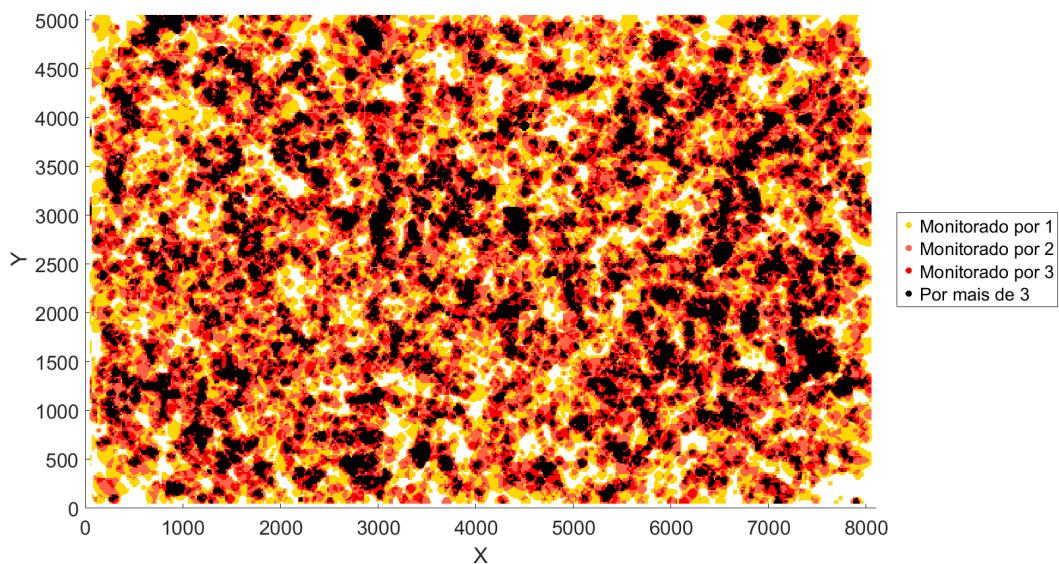


Figura 4.11: Cenário Ideal (*Grid Search*): Níveis de sobreposição das áreas monitoradas.

Analizando através da Figura 4.12 a qualidade de sensoriamento imposta pelo posicionamento dos dispositivos, constatou-se que 48.324% dos ABs apresentaram nível

alto de qualidade, enquanto 33.794% apresentaram nível médio e apenas 17.881% nível baixo. Isso demonstra que a expressão  $AB_{wHS} > AB_{wMS} > AB_{wLS}$  descrita na Subseção 3.3.3 foi alcançada, garantindo que, em grande parte da área monitorada, a detecção de eventos não seja prejudicada por aspectos como distância, ruídos do ambiente, temperaturas extremas ou mesmo o fim da vida útil de um dos dispositivos que monitora, em conjunto com outros, o respectivo bloco.

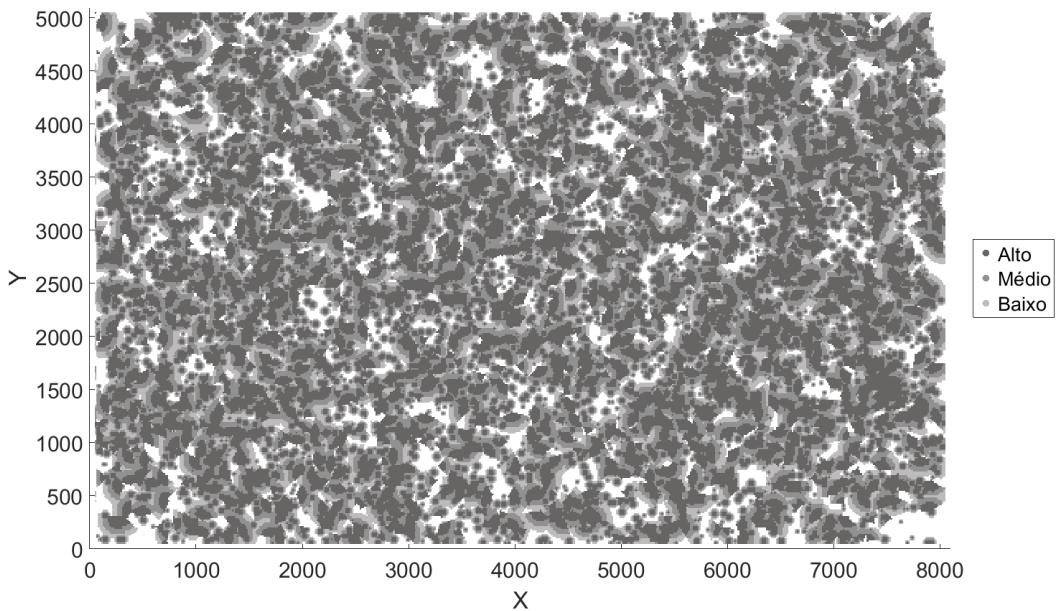


Figura 4.12: Cenário Ideal (*Grid Search*): Níveis de qualidade de sensoriamento das áreas monitoradas.

Após a aplicação do método *Grid Search* e da análise dos resultados obtidos, foi possível estabelecer para o algoritmo NSGA-II um conjunto de operadores genéticos e hiperparâmetros que, diante dessa análise inicial, comprovou ser eficaz na busca por soluções que atendem satisfatoriamente aos critérios de otimização impostos. A partir do conjunto de soluções encontradas, foi escolhida aquela que apresentou maior equilíbrio entre os 3 critérios. Então, constatou-se a eficiência do processo de posicionamento sobre um cenário ideal de teste, no qual os aspectos de qualidade examinados foram apresentados nas Figuras 4.9, 4.10, 4.11 e 4.12.

Diante disso, a próxima etapa deste estudo estará focada na análise de posicionamento em diferentes cenários de teste. O objetivo é avaliar a eficácia do método proposto diante de uma variedade de configurações possíveis para os cenários de implantação e rede de sensores.

## 4.5 Avaliação de cenários aleatórios de teste

O uso das RSSF no monitoramento de áreas urbanas pode ser aplicado em diversos cenários de implantação, nos quais possuem suas próprias características como a taxa

de regiões com conectividade e seus respectivos níveis de conexão, a distribuição das redes sem fio ao longo da RdI, e também o tamanho total da região delimitada. Com isso, diante dessa variabilidade e visando avaliar a eficácia da estratégia de posicionamento proposta neste trabalho, foi decidido realizar a avaliação de 100 cenários de implantação com distribuição aleatória das redes sem fio. Durante esse processo, foi avaliada a metodologia para escolha automatizada de soluções nas FP com base em critérios de relevância para os objetivos, de forma a proporcionar uma estratégia para determinar qual solução da fronteira será avaliada. Também foi averiguado o nível de homogeneidade das soluções encontradas pelo AG a partir de um número predefinido de execuções.

Em todo o processo, foram utilizadas duas configurações para os cenários de implantação. Na Tabela 4.11, encontra-se a definição de suas características e quantos cenários foram gerados a partir de cada uma. E na Tabela 4.12, são apresentadas as configurações dos sensores utilizados na rede de monitoramento. É importante mencionar que as características dos cenários e sensores são dados hipotéticos descrevendo regiões ou dispositivos hipotéticos, com a finalidade de validar a metodologia proposta. Por essa razão, nenhum critério específico foi aplicado a fim de determinar as características dos elementos utilizados nesse experimento.

Os sensores do grupo  $Sr_{cfg_1}$  foram posicionados nos cenários gerados a partir da configuração  $Cfg_1$ , enquanto o grupo  $Sr_{cfg_2}$  foi posicionado nos cenários gerados a partir de  $Cfg_2$ . As características dos dispositivos utilizados são idênticas, diferenciando-se apenas na quantidade de elementos da rede, pois uma RdI com maiores delimitações requer redes de monitoramento mais extensas e densas, capazes de garantir a cobertura adequada da área observada.

Tabela 4.11: Configuração dos cenários com distribuição aleatória de redes.

	Redes			Características		
	WiFi	4G	LoRa	$hs_{AB} \times ws_{AB}$	$hs \times ws$	Cenários
$Cfg_1$	300	1	2	$20 \times 20$	$3000 \times 3000$	50
$Cfg_2$	500	2	4	$20 \times 20$	$5000 \times 5000$	50

Tabela 4.12: Configuração dos sensores para o processo de avaliação dos cenários de implantação com distribuição aleatória das redes sem fio.

<b>Sensor</b>	$Sr_{cfg_1}$				$Sr_{cfg_2}$			
	Qtd	$r_s$	$r_l$	$r_e$	Qtd	$r_s$	$r_l$	$r_e$
<b>Escalar</b>	1000	25	12.5	18.75	1700	25	12.5	18.75
<b>Escalar</b>	1000	50	25	37.5	1700	50	25	37.5
<b>Cam</b> ( $\theta = 90^0$ )	200	150	75	112.5	360	150	75	112.5

#### 4.5.1 Avaliação da metodologia de escolha de soluções das Fronteiras de Pareto

Devido à característica dos AG multiobjetivos de proporcionarem um conjunto de soluções não-dominadas após o término do processo de busca, é necessário que o pesquisador ou especialista realize a escolha de uma ou mais soluções desse conjunto para fins de avaliação e, possivelmente, a implementação em um contexto real. Entretanto, essa escolha depende do contexto em que a aplicação está inserida, sendo possível sua determinação com base em níveis de relevância para cada objetivo. Por exemplo, escolhendo as soluções da FP que apresentam maior cobertura de área ou maior taxa de dispositivos em regiões de alta conectividade. Um método a ser aplicado para esta finalidade é o *Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)* (ver Seção 2.6), o qual também foi utilizado nessa avaliação.

Na fase de análise do *Grid Search*, foi constatado baixos valores de HV, revelando grande proximidade entre as soluções das FP para uma configuração de cenário ideal. Portanto, nesta etapa do estudo, é necessário verificar se, em cenários com distribuição aleatória de redes, a seleção de soluções da fronteira com base em critérios de relevância para os objetivos resulta em diferenças significativas em relação aos parâmetros de posicionamento. Caso essa diferença não seja comprovada, a estratégia mais adequada consiste em adotar soluções que apresentem o maior índice de balanceamento entre os objetivos. Para tanto, dos 100 cenários gerados, foram escolhidos 3 de cada configuração. Na Figura 4.13 é possível observá-los. Cada um se adequando a um dos seguintes critérios:

1. Um cenário com alta quantidade de regiões de alta conectividade;
2. Um cenário com baixa quantidade de regiões de alta conectividade;
3. Um cenário com distribuição intermediária em relação à conectividade;

Uma das etapas do método TOPSIS consiste na ponderação dos valores objetivos normalizados, permitindo atribuir pesos para os diferentes dados. Então, foram estabelecidos 4 níveis de relevância caracterizados pela associação de pesos para cada critério de otimização. Na Tabela 4.13 é possível averiguar como cada nível

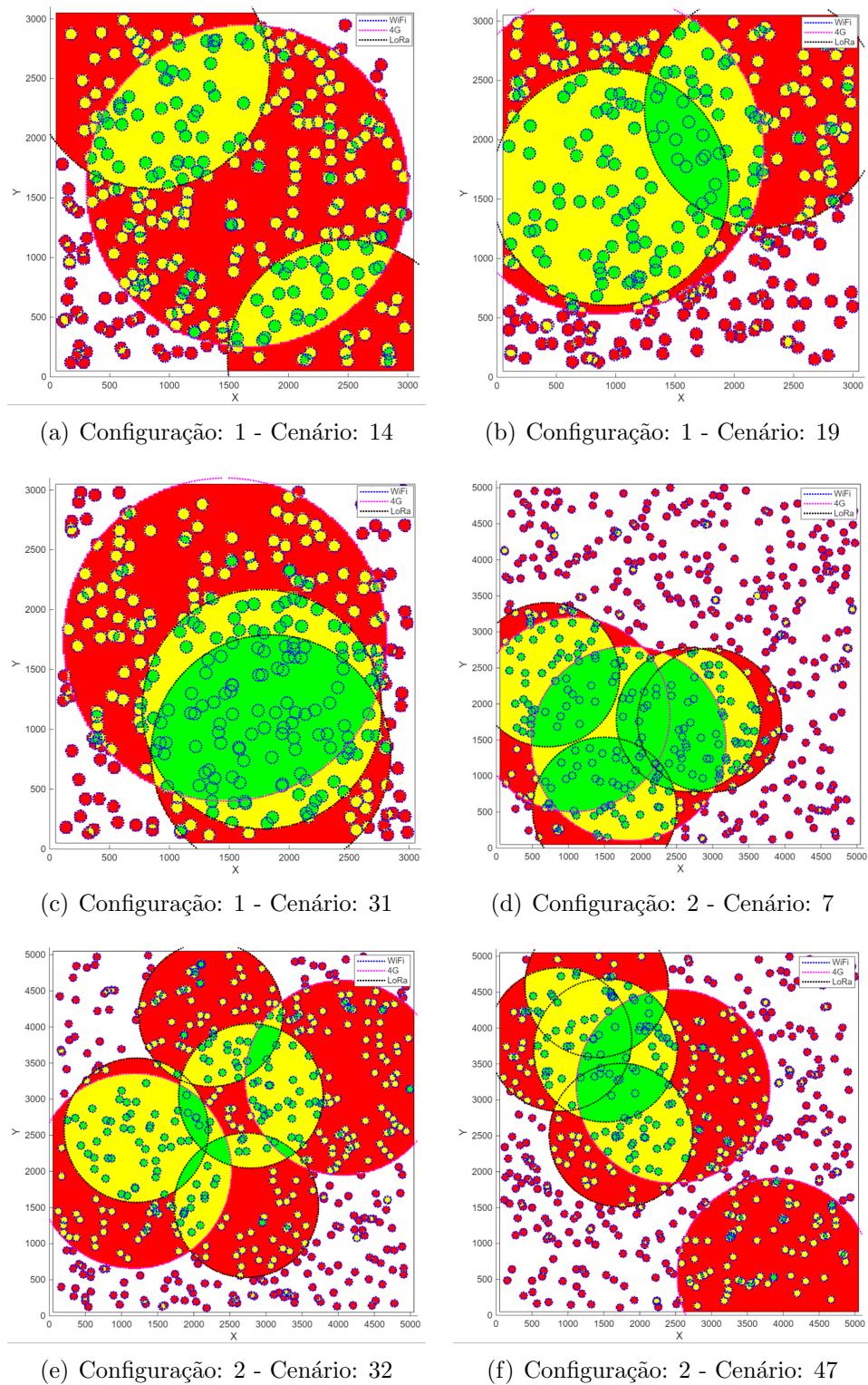


Figura 4.13: Cenários com distribuição aleatória das redes sem fio.

está composto. Os níveis  $R1$ ,  $R2$  e  $R3$  serão aplicados para encontrar a solução que mais se destaca em cada critério de otimização, enquanto o nível  $R4$ , através da atribuição de pesos iguais, possui a finalidade de localizar a solução que apresenta maior equilíbrio entre os três objetivos. Com essas definições, os próximos passos dessa análise consistem em:

1. Executar o processo de posicionamento através do NSGA-II sobre os cenários da Figura 4.13;
2. Após a execução, escolher 4 soluções de cada FP. Cada uma de acordo com um dos níveis de relevância estabelecidos;
3. Calcular e analisar os parâmetros de posicionamento para cada solução escolhida;
4. Averiguar se existe diferenças significativas entre os parâmetros calculados;

Tabela 4.13: Níveis de Relevância para escolha de soluções nas Fronteira de Pareto.

Nível de Relevância	$W_{f_1}$	$W_{f_2}$	$W_{f_3}$
<b>R1</b>	0.8	0.1	0.1
<b>R2</b>	0.1	0.8	0.1
<b>R3</b>	0.1	0.1	0.8
<b>R4</b>	0.3333	0.3333	0.3333

Após a finalização dos passos 1 a 3, temos o levantamento dos dados de posicionamento sobre cada cenário de teste. As Tabelas 4.14, 4.15 e 4.16 apresentam esses dados, levando em consideração os 4 níveis de relevância aplicados para a escolha das soluções nas fronteiras.

Começando a análise dos dados referentes aos cenários da configuração 1, realiza-se a separação pelos níveis de relevância definidos. Então, observa-se que em todos os cenários ( $Cfg_1 - C14$ ,  $Cfg_1 - C19$  e  $Cfg_1 - C31$ ) a solução escolhida através do nível  $R1$  manifestou maior valor de Cobertura em comparação aos outros níveis; entretanto, a diferença é pequena. A maior diferença encontra-se em relação ao nível  $R3$ , sendo 3.58% para o  $Cfg_1 - C14$ ; 4.77% para o  $Cfg_1 - C19$ ; e 5.87% para o  $Cfg_1 - C31$ .

A mesma característica de proximidade pode ser observada ao comparar os parâmetros de Conectividade das soluções escolhidas através do nível  $R2$  e também os parâmetros de Qualidade de Sensoriamento apresentados pelas soluções escolhidas por meio do nível  $R3$ . Quanto aos níveis de sobreposição, igualmente é possível observar a proximidade entre os dados.

Verificando os dados para os cenários da configuração 2 ( $Cfg_2 - C7$ ,  $Cfg_2 - C32$  e  $Cfg_2 - C47$ ) essa proximidade também é percebida. Evidenciando o mesmo comportamento visto nas soluções escolhidas para os cenários da configuração 1.

Portanto, como os níveis de relevância  $R1$ ,  $R2$  e  $R3$  priorizam um critério de otimização, nota-se que cada solução, associada a cada nível, apresentou o comportamento de acentuação que era esperado. Entretanto, o valor dessa acentuação foi pequeno. Deixando evidente que o NSGA-II está gerando em sua FP, soluções muito próximas umas das outras, tanto considerando um cenário com distribuição ideal (ver Figura 4.1) das redes sem fio quanto com distribuição aleatória.

Diante dessas observações, foi concluído que, para o contexto deste estudo de posicionamento, a escolha e análise de soluções da FP a partir da priorização dos critérios de otimização não apresentam vantagens significativas. Portanto, para os próximos testes, será escolhida a solução da fronteira que apresenta maior balanceamento entre os objetivos. Para tanto, o nível de relevância  $R4$  será aplicado, no qual cada critério de otimização está associado ao mesmo valor de ponderação.

Tabela 4.14: Resultados obtidos para os cenários  $Cfg_1 - C14$  e  $Cfg_1 - C19$ . **Relevância(R)**, **Cobertura(CB)**, **Conectividade( CON)**, **Q.Sensoriamento(QS)**, **Sobreposição(SB)**.

$Cfg_1 - C14$					$Cfg_1 - C19$				
R	CB	CON	QS	SB	R	CB	CON	QS	SB
<b>R1</b>	75.38%	H: 13.23% M: 33.05% L: 53.73%	H: 40.90% M: 22.96% L: 36.14%	one: 39.97% two: 32.41% three: 17.68% more: 9.93%	<b>R1</b>	73.30%	H: 24% M: 40.23% L: 35.77%	H: 41.71% M: 23.19% L: 35.11%	one: 38.87% two: 31.23% three: 18.18% more: 11.71%
<b>R2</b>	72.68%	H: 16.00% M: 33.59% L: 50.41%	H: 41.99% M: 22.26% L: 35.75%	one: 39.64% two: 30.32% three: 17.59% more: 12.45%	<b>R2</b>	69.55%	H: 27.59% M: 39.82% L: 32.59%	H: 42.80% M: 22.68% L: 34.52%	one: 36.44% two: 30.23% three: 19.49% more: 13.84%
<b>R3</b>	71.80%	H: 12.91% M: 35.14% L: 51.95%	H: 43.15% M: 22.30% L: 34.55%	one: 37.00% two: 32.14% three: 18.22% more: 12.63%	<b>R3</b>	68.53%	H: 25% M: 40.73% L: 34.27%	H: 44.19% M: 21.82% L: 33.99%	one: 35.11% two: 31.05% three: 19.50% more: 14.34%
<b>R4</b>	74.04%	H: 14.95% M: 32.86% L: 52.18%	H: 41.88% M: 22.37% L: 35.75%	one: 38.98% two: 32.23% three: 17.71% more: 11.08%	<b>R4</b>	72.96%	H: 25.32% M: 41.32% L: 33.36%	H: 42.03% M: 22.87% L: 35.09%	one: 38.52% two: 31.35% three: 18.60% more: 11.53%

Tabela 4.15: Resultados obtidos para os cenários  $Cfg_1 - C31$  e  $Cfg_2 - C7$ . **Relevância(R)**, **Cobertura(CB)**, **Conectividade( CON)**, **Q.Sensoriamento(QS)**, **Sobreposição(SB)**.

$Cfg_1 - C31$						$Cfg_2 - C7$					
R	CB	CON	QS	SB		R	CB	CON	QS	SB	
<b>R1</b>	73.46%	H: 40.73% M: 20.09% L: 39.18%	H: 42.23% M: 22.59% L: 35.19%	one: 38.20% two: 32.17% three: 18.56% more: 11.07%		<b>R1</b>	47.66%	H: 46.76% M: 26.97% L: 26.28%	H: 40.09% M: 22.72% L: 37.19%	one: 44.43% two: 28.82% three: 15.89% more: 10.87%	
<b>R2</b>	70.16%	H: 46.18% M: 17.05% L: 36.77%	H: 42.93% M: 22.81% L: 34.26%	one: 36.89% two: 31.43% three: 18.31% more: 13.38%		<b>R2</b>	45.15%	H: 50.27% M: 24.92% L: 24.81%	H: 41.44% M: 22.41% L: 36.16%	one: 41.68% two: 27.66% three: 17.33% more: 13.33%	
<b>R3</b>	67.59%	H: 44.05% M: 18.95% L: 37.00%	H: 44.55% M: 21.37% L: 34.09%	one: 35.39% two: 29.78% three: 19.00% more: 15.84%		<b>R3</b>	44.25%	H: 49.55% M: 26.38% L: 24.07%	H: 42.48% M: 22.03% L: 35.49%	one: 40.24% two: 27.53% three: 18.36% more: 13.87%	
<b>R4</b>	71.95%	H: 45.00% M: 18.00% L: 37.00%	H: 41.74% M: 22.73% L: 35.53%	one: 38.71% two: 29.59% three: 18.80% more: 12.91%		<b>R4</b>	46.66%	H: 48.51% M: 25.93% L: 25.56%	H: 40.88% M: 22.53% L: 36.59%	one: 43.23% two: 27.91% three: 16.98% more: 11.88%	

Tabela 4.16: Resultados obtidos para os cenários  $Cfg_2 - C32$  e  $Cfg_2 - C47$ . **Relevância(R)**, **Cobertura(CB)**, **Conectividade( CON)**, **Q.Sensoriamento(QS)**, **Sobreposição(SB)**.

$Cfg_2 - C32$						$Cfg_2 - C47$					
R	CB	CON	QS	SB		R	CB	CON	QS	SB	
<b>R1</b>	55.93%	H: 11.91% M: 33.14% L: 54.95%	H: 36.02% M: 22.38% L: 41.61%	one: 53.42% two: 29.47% three: 12.49% more: 4.62%		<b>R1</b>	53.58%	H: 17.05% M: 25.00% L: 57.95%	H: 37.16% M: 22.58% L: 40.26%	one: 51.12% two: 29.65% three: 13.16% more: 6.07%	
<b>R2</b>	53.96%	H: 13.94% M: 33.11% L: 52.95%	H: 36.70% M: 22.25% L: 41.05%	one: 51.16% two: 29.93% three: 13.04% more: 5.87%		<b>R2</b>	50.55%	H: 19.87% M: 23.51% L: 56.62%	H: 38.62% M: 21.93% L: 39.45%	one: 48.18% two: 29.07% three: 14.39% more: 8.36%	
<b>R3</b>	53.22%	H: 12.55% M: 33.43% L: 54.02%	H: 37.63% M: 22.05% L: 40.32%	one: 49.67% two: 30.56% three: 13.67% more: 6.09%		<b>R3</b>	50.04%	H: 18.01% M: 25.00% L: 56.99%	H: 39.38% M: 21.88% L: 38.74%	one: 46.36% two: 30.46% three: 15.18% more: 7.99%	
<b>R4</b>	55.59%	H: 13.22% M: 33.11% L: 53.67%	H: 35.87% M: 22.36% L: 41.77%	one: 53.24% two: 30.06% three: 11.82% more: 4.89%		<b>R4</b>	52.74%	H: 18.62% M: 24.63% L: 56.76%	H: 37.52% M: 22.34% L: 40.13%	one: 50.19% two: 29.52% three: 13.56% more: 6.73%	

### 4.5.2 Análise de variabilidade das soluções geradas pelo Algoritmo Genético

O último experimento a ser realizado nesta pesquisa consiste em verificar a variabilidade das soluções encontradas pelo Algoritmo Genético a partir de execuções sucessivas do NSGA-II. Essa análise tem por objetivo verificar se, sobre o mesmo cenário de implantação e rede de sensores, o AG tende a convergir para uma configuração específica de soluções, mesmo sem possuir uma FP ideal para fins de comparação.

Para isso, cada um dos 100 cenários gerados a partir da distribuição aleatória das redes foi processado 50 vezes consecutivas pelo NSGA-II sobre o mesmo cenário de implantação e rede de sensores. Os cenários da  $Cfg_1$  executados sobre a configuração  $Sr_{Cf g_1}$  de sensores, e os cenários  $Cfg_2$  sobre a configuração  $Sr_{Cf g_2}$ . Para avaliar a variabilidade das soluções, foi calculada a métrica de Desvio Padrão (DEP) e a Média das Diferenças Sucessivas (MDS) sobre os parâmetros de posicionamento.

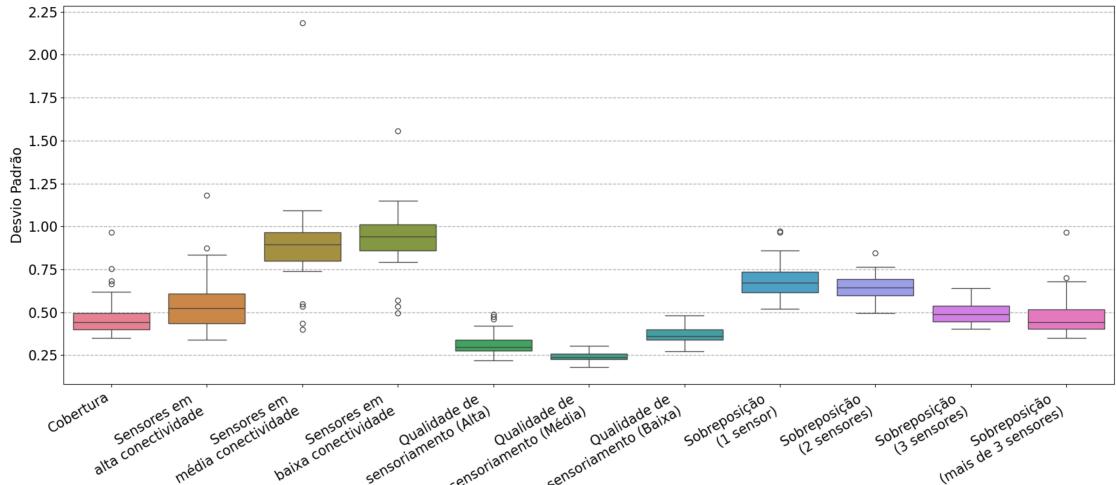
As seguintes etapas foram definidas para esse experimento. Para cada cenário de implantação:

1. Executar 50 vezes consecutivas o NSGA-II, salvando as respectivas FP;
2. Em cada fronteira, escolher uma solução através do método TOPSIS aplicando os pesos do nível de relevância  $R4$ ;
3. Para cada solução, calcular os parâmetros de posicionamento;
4. Juntar os dados das 50 execuções e calcular os valores para métricas DEP e MDS para cada parâmetro de posicionamento;

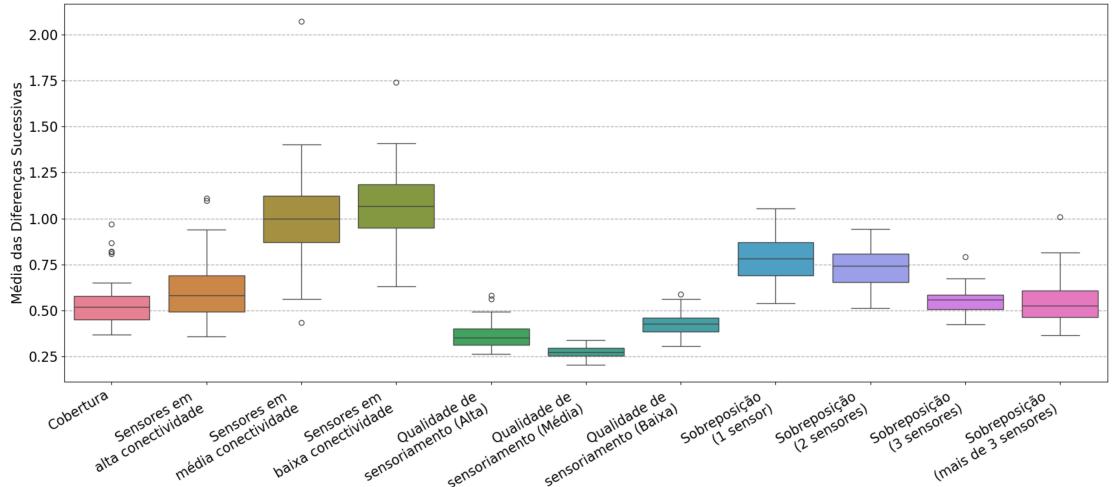
Essa análise foi iniciada a partir dos resultados alcançados para os cenários da configuração  $Cfg_1$ . Na Figura 4.14(a), é possível observar que, em grande parte dos dados, o desvio padrão calculado está abaixo de 1.25, tendo uma pequena variação. Também podemos encontrar alguns *outliers* nessa representação, revelando que, em relação ao conjunto de cenários, alguns deles obtiveram maior ou menor variabilidade em parâmetros específicos após a finalização do número de execuções pré-definidas. Os baixos valores encontrados para o DEP são um indício da comprovação da baixa variabilidade dos dados produzidos.

Contudo, como medida de reforço à análise do DEP, a MDS foi calculada sobre cada par de execuções em cada cenário. A Figura 4.14(b) exibe os resultados alcançados. A partir desses dados, notou-se que todos os parâmetros estão com um valor de diferença abaixo de 1.5%, tendo apenas dois cenários com uma maior variação em relação à taxa de sensores em média e baixa conectividade.

Então, em conformidade com a análise do DEP e da MDS, é possível concluir que, para cada cenário executado 50 vezes sobre as mesmas configurações ( $Cfg_1$  e  $Sr_{cfg_1}$ ), o NSGA-II está convergindo para uma região específica do espaço de objetivos e produzindo soluções muito próximas umas das outras; desta forma, proporcionando



(a) Configuração: 1 - Desvio Padrão dos dados de posicionamento.



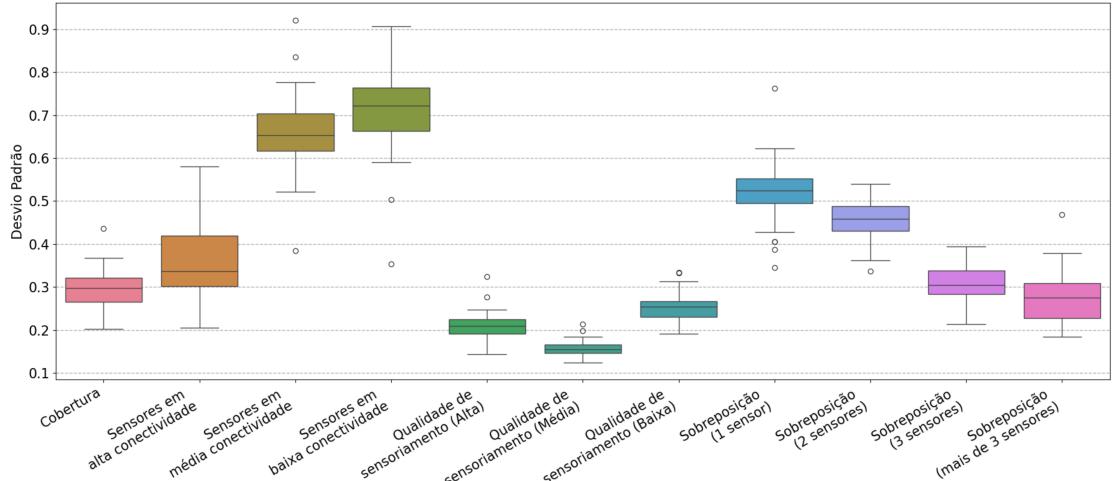
(b) Configuração: 1 - Média das Diferenças Sucessivas dos dados de posicionamento.

Figura 4.14: *Boxplot*: Análise dos dados de posicionamento para os cenários da configuração 1.

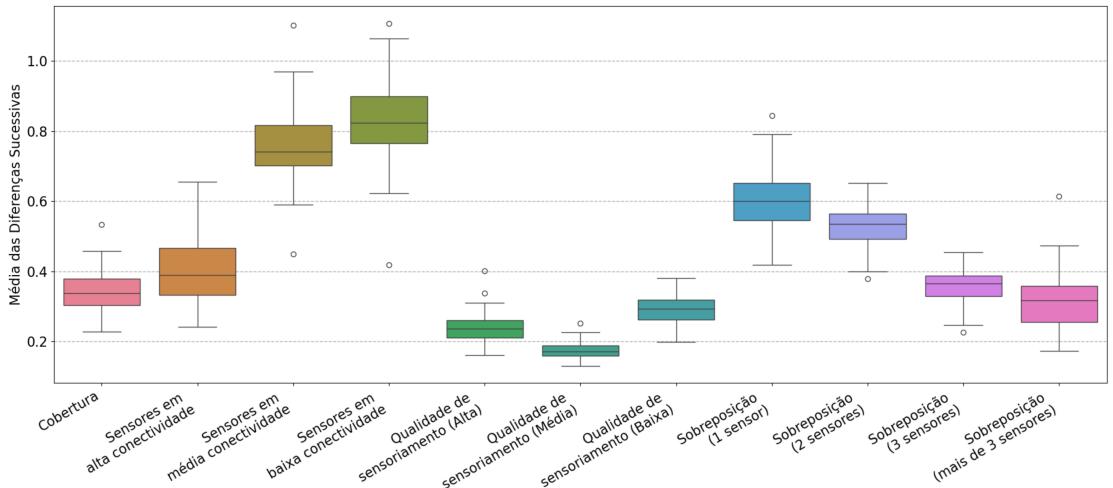
a característica de baixa variabilidade ou alta homogeneidade com relação às soluções para cada cenário.

Com respeito aos dados de posicionamento produzidos a partir dos cenários da configuração  $Cfg_2$  e da rede de sensores  $Sr_{cfg_2}$ , também foram averiguados os valores para o DEP e a MDS. Nas Figuras 4.15(a) e 4.15(b) são exibidos os dados calculados. Comparando os valores apresentados na Figura 4.14(a) com relação à disposição vista em 4.15(a), nota-se que, para os dados na Figura 4.15(a), encontra-se um DEP relativamente mais baixo, com um alcance máximo um pouco acima de 0.9. Embora exista uma grande variação dos valores do DEP entre os parâmetros de posicionamento, a baixa variabilidade pode ser notada. Observando a MDS na

Figura 4.15(b) pode-se comprovar essa característica. Consta-se que grande parte dos dados está apresentando uma diferença abaixo de 1%, demonstrando que, entre cada execução do NSGA-II, a proximidade vista entre os parâmetros calculados é considerável.



(a) Configuração: 2 - Desvio Padrão dos dados de posicionamento.



(b) Configuração: 2 - Média das Diferenças Sucessivas dos dados de posicionamento.

Figura 4.15: *Boxplot*: Análise dos dados de posicionamento para os cenários da configuração 2.

Logo, concluímos que, assim como no primeiro caso, o NSGA-II está convergindo para uma região específica do espaço de objetivos e produzindo soluções próximas umas das outras. Essa característica indica que o algoritmo é capaz de gerar um conjunto consistente de alternativas de posicionamento, mesmo quando executado repetidamente, independente da caracterização do cenário de implantação. A baixa variabilidade observada nas métricas DEP e MDS reforça a robustez do modelo.

proposto, garantindo que os resultados obtidos apontam para uma convergência estável e reproduzível.

Embora este experimento final tenha considerado as métricas descritivas MDS e o DEP, não foram aplicados testes estatísticos formais para verificar a significância das variações observadas entre as execuções. Essa limitação implica que as diferenças identificadas refletem apenas uma análise exploratória da variabilidade, sem confirmação estatística quanto à sua relevância. A aplicação de testes não paramétricos poderia fornecer evidências adicionais sobre a consistência e robustez das soluções obtidas, permitindo uma interpretação mais sólida dos resultados. Assim, é possível que, com a utilização de tais testes, as conclusões sobre a estabilidade e convergência do NSGA-II pudessem apresentar variações ou reforçar de forma mais rigorosa as tendências observadas empiricamente.

Então, levando em consideração os resultados observados, do ponto de vista da eficiência, demonstram que a metodologia é capaz de explorar o espaço de busca de forma abrangente e, ao mesmo tempo, concentrar-se em soluções de alta qualidade, mesmo na ausência de uma fronteira de Pareto ideal para comparação. Isso é especialmente relevante para cenários reais de implantação, nos quais é difícil obter um “padrão ouro” para validação. Mesmo sem conhecer a complexidade computacional do método proposto, a suposta convergência consistente obtida com execuções sucessivas por cenário indica que o método apresenta baixo custo computacional incremental para rodadas adicionais, podendo ser utilizado em situações nas quais é necessário recalcular rapidamente o posicionamento em tempo quase real, como no caso de redes adaptativas ou ambientes urbanos sujeitos a mudanças dinâmicas.

Por fim, quanto à aplicabilidade, os resultados corroboram que a estratégia baseada no NSGA-II é adequada para apoiar a tomada de decisão no planejamento de redes de sensores em áreas urbanas e outras regiões de grande escala. A baixa dispersão das soluções permite que engenheiros de rede selecionem, com maior confiança, configurações de sensores que equilibrem cobertura, qualidade de sensoriamento e conectividade, reduzindo o risco de sub ou superdimensionamento da rede. Além disso, a metodologia proposta pode ser integrada a ferramentas de simulação e plataformas de cidades inteligentes, servindo como um módulo de otimização que se adapta a diferentes objetivos de monitoramento, por exemplo, vigilância ambiental, detecção precoce de emergências ou suporte a aplicações de mobilidade urbana. Assim, este trabalho contribui não apenas com resultados teóricos consistentes, mas também com um caminho viável para aplicações práticas e escaláveis em Internet das Coisas e Cidades Inteligentes, por exemplo.

# **Capítulo 5**

## **Conclusões**

Com o crescente aumento das áreas urbanas, temos o advento de novos desafios decorrentes da complexidade do processo de monitoramento e gestão eficiente dos recursos disponíveis. Nesse contexto, o uso das RSSF tem sido crucial na construção de aplicações para coleta e transmissão de dados em tempo real. Sua natureza distribuída e sem fio contribui significativamente para a redução de uma infraestrutura fixa, tornando sua implementação mais flexível e econômica. Entretanto, a eficiência dessa rede de monitoramento depende do posicionamento adequado dos seus elementos. Por esta razão, ao longo dos anos, pesquisadores em todo o mundo alocam esforços para desenvolver pesquisas que proporcionem estratégias de posicionamento mais eficientes e que estejam alinhadas com as diferentes necessidades de monitoramento.

Com o objetivo de contribuir para esse campo de estudo, o presente trabalho propôs uma estratégia de posicionamento baseada na utilização do algoritmo genético multiobjetivo NSGA-II para determinar o local de implantação dos elementos de uma RSSF composta tanto por sensores escalares quanto visuais. Nesse processo, três objetivos foram otimizados: a extensão da área monitorada dentro das RdI, a qualidade de sensoriamento realizada pelos dispositivos pré-configurados e o posicionamento desses elementos em regiões de maior conectividade a fim de garantir robustez aos processos de comunicação.

Durante o processo de modelagem da solução proposta, novas contribuições em relação a pesquisas prévias foram abordadas, tendo destaque o processo de mapeamento da RdI para localização e identificação de áreas com conectividade; a modelagem do modelo de sensoriamento e algoritmo para construção das máscaras de cobertura associadas aos sensores escalares e câmeras; construção de uma representação cromossômica para o AE que viabiliza a otimização do posicionamento de RSSF heterogêneas, aumentando a eficiência do processo de busca de soluções através da delimitação das regiões com conectividade; além da construção de uma representação matemática para avaliação de dois novos critérios de otimização para as RSSF.

As fases iniciais desta pesquisa foram compostas pela modelagem e representação das RdI e dispositivos de monitoramento, pela definição das funções de *fitness* usadas para avaliar a qualidade das soluções produzidas pelo NSGA-II, definição da estrutura do cromossomo usado para representar as soluções dentro do contexto dos algoritmos evolutivos, além da modelagem do processo de posicionamento através dessas estruturas.

Dentre os resultados obtidos, temos a definição dos hiperparâmetros do algoritmo genético através do método de busca exaustiva *Grid Search*. Esse método permitiu escolher dentro de um conjunto finito de combinações de parâmetros, aquela que possibilitou alcançar melhores soluções para o contexto de posicionamento em um cenário com distribuição ideal das redes. A partir disso, foi avaliado um dos posicionamentos realizados sobre esse cenário, comprovando a sua eficácia ao atender aos critérios estabelecidos através das funções de *fitness*.

O método proposto foi avaliado por meio de um conjunto vasto e diversificado de cenários com distribuição aleatória das redes sem fio. Para tanto, o método TOPSIS foi definido como mecanismo de escolha de soluções em uma Fronteira de Pareto, a fim de permitir calcular os parâmetros de posicionamento a partir da melhor solução encontrada para cada cenário. Como resultado, observou-se que, mesmo com execuções sucessivas sobre um mesmo cenário e rede de sensores, a metodologia proposta proporciona soluções com baixa variabilidade entre si, corroborando para uma estratégia de posicionamento estável e convergente.

Diante disso, os resultados obtidos demonstraram a eficácia da metodologia, indicando que a estratégia apresentada pode ser aplicada em diferentes domínios, não apenas em áreas urbanas, mas em qualquer RdI que atenda aos critérios do mapeamento proposto. Em um contexto real, essa abordagem pode aumentar significativamente a resiliência das aplicações de monitoramento, criando meios para respostas mais rápidas a incidentes, tomada de decisões e expansão dos serviços disponíveis.

Ademais, visando o aprimoramento desta pesquisa, alguns pontos foram elencados, possibilitando a continuidade do trabalho a partir de novas modelagens e testes a serem realizados, por exemplo:

- A substituição do sistema de coordenadas  $(x, y)$  para a utilização de coordenadas GPS (Sistema de Posicionamento Global). Essa mudança possibilitará alcançar maior precisão na localização das redes sem fio e no posicionamento dos sensores, permitindo a construção de um sistema mais preciso e adaptado para o contexto do mundo real. Como por exemplo, a realização do processo de posicionamento sobre RdI baseadas em áreas urbanas reais.
- Incluir no processo de posicionamento a modelagem de alguns tipos de obstáculos e restrições presentes em áreas urbanas. Isso permitirá analisar e criar soluções para diferentes contextos em que a capacidade de sensoriamento nas RdI é afetada por objetos (carros, prédios, casas, dentre outros) normalmente encontrados nessas regiões.

- Como este problema de posicionamento não possui uma solução ideal para ser usada como referência, necessita-se definir uma métrica que possibilite avaliar a qualidade do posicionamento efetuado a partir das características intrínsecas da RdI, distribuição das redes sem fio, extensão da rede de sensores e as taxas dos parâmetros de posicionamento. Desta forma, permitindo evidenciar se o posicionamento está adequado ou pode ser melhorado.
- Construção de uma aplicação gráfica que facilite a utilização do processo de posicionamento proposto neste trabalho. Por meio de um *design* simples e intuitivo, essa ferramenta poderá facilitar a configuração da rede de sensores, a delimitação da RdI, visualização da Fronteira de Pareto após o processamento do AG, além da visualização das estatísticas associadas aos parâmetros de posicionamento.

Dessa forma, este trabalho representa uma base sólida para a evolução de estratégias de posicionamento em Redes de Sensores Sem Fio, permitindo o desenvolvimento de soluções mais robustas e aderentes a cenários reais. A continuidade dessa pesquisa, com a integração de coordenadas geográficas, modelagem de obstáculos urbanos e implementação de ferramentas gráficas de apoio, tende a ampliar sua aplicabilidade em projetos de cidades inteligentes, sistemas de monitoramento adaptativos e aplicações de resposta a emergências. Com isso, estabelece-se um caminho concreto para a implementação de métodos mais precisos, escaláveis e eficientes, capazes de atender às crescentes demandas por monitoramento em tempo real em ambientes complexos e dinâmicos.

# Referências

- Aghdasi, H. S. e Abbaspour, M. (2016). Energy efficient area coverage by evolutionary camera node scheduling algorithms in visual sensor networks. *Soft Computing*, 20(3):1191–1202.
- Aguilar-Rivera, R., Valenzuela-Rendón, M., e Rodríguez-Ortiz, J. (2015). Genetic algorithms and darwinian approaches in financial applications: A survey. *Expert Systems with Applications*, 42(21):7684–7697.
- Akyildiz, I., Su, W., Sankarasubramaniam, Y., e Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422.
- Alaei, M. e Barcelo-Ordinas, J. M. (2010). Node clustering based on overlapping fovs for wireless multimedia sensor networks. In *2010 IEEE Wireless Communication and Networking Conference*, páginas 1–6.
- Amutha, J., Sharma, S., e Nagar, J. (2020). Wsn strategies based on sensors, deployment, sensing models, coverage and energy efficiency: Review, approaches and open issues. *Wireless Personal Communications*, 111(2):1089–1115.
- Baba, M., Gui, V., Cernazanu, C., e Pescaru, D. (2019). A sensor network approach for violence detection in smart cities using deep learning. *Sensors*, 19(7).
- Benatia, M. A., Sahnoun, M., Baudry, D., Louis, A., El-Hami, A., e Mazari, B. (2017). Multi-objective wsn deployment using genetic algorithms under cost, coverage, and connectivity constraints. *Wireless Personal Communications*, 94(4):2739–2768.
- Bergstra, J., Bardenet, R., Bengio, Y., e Kégl, B. (2011). Algorithms for hyper-parameter optimization. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., e Weinberger, K., editores, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc.
- Binh, H. T. T., Hanh, N. T., Quan, L. V., e Dey, N. (2018). Improved cuckoo search and chaotic flower pollination optimization algorithm for maximizing area coverage in wireless sensor networks. *Neural Computing and Applications*, 30(7):2305–2317.

- Birtane, S., Sahingoz, O. K., e Korkmaz, H. (2024). Vibrational Genetic Algorithm-Based Deployment of Wireless Sensor Networks With Heterogeneous Nodes in Irregularly Shaped Areas. *IEEE Access*, 12:62926–62941.
- Birtane Akar, S., Korkmaz, H., e Sahingoz, O. K. (2022). Efficient Deployment of Wireless Sensor Nodes with Evolutionary Approaches. In *2022 International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, páginas 1–7, Ankara, Turkey. IEEE.
- Bouzid, S. E., Serrestou, Y., Raoof, K., Mbarki, M., Omri, M. N., e Dridi, C. (2020). Wireless sensor network deployment optimisation based on coverage, connectivity and cost metrics. *International Journal of Sensor Networks*, 33(4):224–238.
- Bremermann, H. J. (1962). Optimization through evolution and recombination. *Self-organizing systems*, páginas 93–106.
- Cardei, M., MacCallum, D., Cheng, M. X., Min, M., Jia, X., Li, D., e Du, D.-Z. (2002). Wireless sensor networks with energy efficient organization. *Journal of Interconnection Networks*, 03(03n04):213–229.
- Chakraborty, M. e Rout, M. (2021). Optimal barrier coverage in randomly scattered wireless sensor networks. In *2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, páginas 1–6.
- Chaudhry, S. e Luo, W. (2005). Application of genetic algorithms in production and operations management: A review. *International Journal of Production Research*, 43(19):4083 – 4101.
- Chen, A., Kumar, S., e Lai, T. H. (2010). Local barrier coverage in wireless sensor networks. *IEEE Transactions on Mobile Computing*, 9(4):491–504.
- Chen, Y.-N., Lin, W.-H., e Chen, C. (2020). An effective sensor deployment scheme that ensures multilevel coverage of wireless sensor networks with uncertain properties. *Sensors*, 20(7).
- Coello, C. A. C. (2007). *Evolutionary algorithms for solving multi-objective problems*. Springer.
- Costa, D. G., Peixoto, J. P. J., Jesus, T. C., Portugal, P., Vasques, F., Rangel, E., e Peixoto, M. (2022). A survey of emergencies management systems in smart cities. *IEEE Access*, 10:61843–61872.
- Costa, D. G., Silva, I., Guedes, L. A., Portugal, P., e Vasques, F. (2014). Enhancing redundancy in wireless visual sensor networks for target coverage. In *Proceedings of the 20th Brazilian Symposium on Multimedia and the Web, WebMedia '14*, página 31–38, New York, NY, USA. Association for Computing Machinery.

- Costa, D. G., Silva, I., Guedes, L. A., Vasques, F., e Portugal, P. (2015). Optimal sensing redundancy for multiple perspectives of targets in wireless visual sensor networks. In *2015 IEEE 13th International Conference on Industrial Informatics (INDIN)*, páginas 185–190.
- Deb, K., Agrawal, R. B., et al. (1995). Simulated binary crossover for continuous search space. *Complex systems*, 9(2):115–148.
- Deb, K. e Deb, D. (2014). Analysing mutation schemes for real-parameter genetic algorithms. *International Journal of Artificial Intelligence and Soft Computing*, 4(1):1–28. PMID: 59280.
- Deb, K., Pratap, A., Agarwal, S., e Meyarivan, T. (2002). A fast and elitist multi-objective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197.
- Deep, K. e Thakur, M. (2007). A new crossover operator for real coded genetic algorithms. *Applied Mathematics and Computation*, 188(1):895–911.
- Dhingra, S., Madda, R. B., Gandomi, A. H., Patan, R., e Daneshmand, M. (2019). Internet of things mobile–air pollution monitoring system (iot-mobair). *IEEE Internet of Things Journal*, 6(3):5577–5584.
- Domingo-Perez, F., Lazaro-Galilea, J. L., Bravo, I., Gardel, A., e Rodriguez, D. (2016). Optimization of the coverage and accuracy of an indoor positioning system with a variable number of sensors. *Sensors*, 16(6).
- Eiben, A. E. e Smith, J. E. (2015). *Introduction to evolutionary computing*. Springer.
- El-Sherif, M., Fahmy, Y., e Kamal, H. (2018). Lifetime maximisation of disjoint wireless sensor networks using multiobjective genetic algorithm. *IET Wireless Sensor Systems*, 8(5):200–207.
- Eshelman, L. J., Caruana, R. A., e Schaffer, J. D. (1989). Biases in the crossover landscape. In *Proceedings of the Third International Conference on Genetic Algorithms*, página 10–19, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.
- Faga, Y., Abdou, W., e Dubois, J. (2022). An optimised indoor deployment of visual sensor networks. In *2022 16th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, páginas 479–485.
- Faga, Y., Tossa, F., Abdou, W., e Dubois, J. (2023). Deployment of visual sensor networks for redundant data acquisition. In *2023 17th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, páginas 274–280.
- Fan, F., Ji, Q., Wu, G., Wang, M., Ye, X., e Mei, Q. (2019). Dynamic barrier coverage in a wireless sensor network for smart grids. *Sensors*, 19(1).

- Fei, Z., Li, B., Yang, S., Xing, C., Chen, H., e Hanzo, L. (2017). A survey of multi-objective optimization in wireless sensor networks: Metrics, algorithms, and open problems. *IEEE Communications Surveys & Tutorials*, 19(1):550–586.
- Filho, G. P. R., Ueyama, J., Villas, L. A., Pinto, A. R., Gonçalves, V. P., Pessin, G., Pazzi, R. W., e Braun, T. (2014). Nodepm: A remote monitoring alert system for energy consumption using probabilistic techniques. *Sensors*, 14(1):848–867.
- Fogel, L. J., Owens, A. J., e Walsh, M. J. (1965). Artificial intelligence through a simulation of evolution. In Callahan, A., Maxfield, M., e Fogel, L. J., editores, *Biophysics and Cybernetic Systems*, páginas 131–156. Spartan, Washington, DC.
- Fogel, L. J., Owens, A. J., e Walsh, M. J. (1966). *Artificial Intelligence through Simulated Evolution*. Wiley, Chichester, UK.
- Ghosh, A. e Das, S. K. (2008). Coverage and connectivity issues in wireless sensor networks: A survey. *Pervasive and Mobile Computing*, 4(3):303–334.
- Goldberg, D. (1989). Genetic algorithms in search, optimization, and machine learning, lu: ading. MA: Addison W.. ley.
- Hanh, N. T., Binh, H. T. T., Hoai, N. X., e Palaniswami, M. S. (2019). An efficient genetic algorithm for maximizing area coverage in wireless sensor networks. *Information Sciences*, 488:58–75.
- Harizan, S. e Kuila, P. (2020). A novel NSGA-II for coverage and connectivity aware sensor node scheduling in industrial wireless sensor networks. *Digital Signal Processing*, 105:102753.
- Harrison, C., Eckman, B., Hamilton, R., Hartswick, P., Kalagnanam, J., Paraszczak, J., e Williams, P. (2010). Foundations for smarter cities. *IBM Journal of Research and Development*, 54(4):1–16.
- Higashi, N. e Iba, H. (2003). Particle swarm optimization with gaussian mutation. In *Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No.03EX706)*, páginas 72–79.
- Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.
- Hossain, A., Biswas, P. K., e Chakrabarti, S. (2008). Sensing models and its impact on network coverage in wireless sensor network. In *2008 IEEE Region 10 and the Third international Conference on Industrial and Information Systems*, páginas 1–5.
- Jesus, T. C., Costa, D. G., Portugal, P., e Vasques, F. (2022). A survey on monitoring quality assessment for wireless visual sensor networks. *Future Internet*, 14(7).

- Jesus, T. C., Costa, D. G., Portugal, P., Vasques, F., e Ferreira, W. A. (2023). Dependability and quality-aware connectivity in smart cities applications. In *2023 IEEE International Smart Cities Conference (ISC2)*, páginas 1–7.
- Jourdan, D. e de Weck, O. (2004). Layout optimization for a wireless sensor network using a multi-objective genetic algorithm. In *2004 IEEE 59th Vehicular Technology Conference. VTC 2004-Spring (IEEE Cat. No.04CH37514)*, volume 5, páginas 2466–2470 Vol.5.
- Kandris, D., Nakas, C., Vomvas, D., e Koulouras, G. (2020). Applications of wireless sensor networks: An up-to-date survey. *Applied System Innovation*, 3(1).
- Khalesian, M. e Delavar, M. R. (2016). Wireless sensors deployment optimization using a constrained pareto-based multi-objective evolutionary approach. *Engineering Applications of Artificial Intelligence*, 53:126–139.
- Kim, I. Y. e de Weck, O. L. (2006). Adaptive weighted sum method for multiobjective optimization: a new method for pareto front generation. *Structural and Multidisciplinary Optimization*, 31(2):105–116.
- Koza, J. R. (1992). *Genetic Programming*. MIT Press, Cambridge, MA.
- Koza, J. R. (1994). *Genetic Programming II*. MIT Press, Cambridge, MA.
- Kumar, S. e Lobiyal, D. K. (2013). Sensing coverage prediction for wireless sensor networks in shadowed and multipath environment. *The Scientific World Journal*, 2013(1):565419.
- Kuorilehto, M., Hännikäinen, M., e Hämäläinen, T. D. (2005). A survey of application distribution in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2005(5):859712.
- Le Berre, M., Hnaien, F., e Snoussi, H. (2011). Multi-objective optimization in wireless sensors networks. In *ICM 2011 Proceeding*, páginas 1–4.
- Li, M. e Yao, X. (2019). Quality evaluation of solution sets in multiobjective optimisation: A survey. *ACM Comput. Surv.*, 52(2).
- Liao, W.-H., Kao, Y., e Wu, R.-T. (2011). Ant colony optimization based sensor deployment protocol for wireless sensor networks. *Expert Systems with Applications*, 38(6):6599–6605.
- Liu, L., Ma, H., e Zhang, X. (2008). On directional k-coverage analysis of randomly deployed camera sensor networks. In *2008 IEEE International Conference on Communications*, páginas 2707–2711.
- Liu, T., Li, Z., Xia, X., e Luo, S. (2009). Shadowing effects and edge effect on sensing coverage for wireless sensor networks. In *2009 5th International Conference on Wireless Communications, Networking and Mobile Computing*, páginas 1–4.

- Ly, D. T. H., Hanh, N. T., Binh, H. T. T., e Nghia, N. D. (2015). An improved genetic algorithm for maximizing area coverage in wireless sensor networks. In *Proceedings of the 6th International Symposium on Information and Communication Technology*, SoICT '15, página 61–66, New York, NY, USA. Association for Computing Machinery.
- Mainwaring, A., Culler, D., Polastre, J., Szewczyk, R., e Anderson, J. (2002). Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*, WSNA '02, página 88–97, New York, NY, USA. Association for Computing Machinery.
- Mehdi, R.-S. e Mejdi, K. (2018). On optimizing the deployment cost of visual sensor networks. *Journal of High Speed Networks*, 24(1):17–30.
- Michalewicz, Z. (2013). *Genetic algorithms+ data structures= evolution programs*. Springer Science & Business Media.
- Mohamed, S. M., Hamza, H. S., e Saroit, I. A. (2017). Coverage in mobile wireless sensor networks (m-wsn): A survey. *Computer Communications*, 110:133–150.
- Molina, G., Alba, E., e Talbi, E.-G. (2008). Optimal sensor network layout using multi-objective metaheuristics. *J. Univers. Comput. Sci.*, 14(15):2549–2565.
- Moscibroda, T., Rickenbach, P. v., e Wattenhofer, R. (2006). Analyzing the energy-latency trade-off during the deployment of sensor networks. In *INFOCOM 2006: 25th Annual Joint Conference of the IEEE Computer and Communications Societies, Barcelona, Spain*.
- Navin, A. H., Asadi, B., Pour, S. H., e Mirnia, M. (2010). Solving coverage problem in wireless camera-based sensor networks by using genetic algorithm. In *2010 International Conference on Computational Intelligence and Communication Networks*, páginas 226–229.
- Paszkowicz, W. (2009). Genetic algorithms, a nature-inspired tool: Survey of applications in materials science and related fields. *Materials and Manufacturing Processes*, 24(2):174–197.
- Priyadarshi, R., Gupta, B., e Anurag, A. (2020). Deployment techniques in wireless sensor networks: a survey, classification, challenges, and future research issues. *The Journal of Supercomputing*, 76(9):7333–7373.
- Rangel, E. O., Costa, D. G., e Loula, A. (2019). On redundant coverage maximization in wireless visual sensor networks: Evolutionary algorithms for multi-objective optimization. *Applied Soft Computing*, 82:105578.
- Rani, K. S. S. e Devarajan, N. (2012). Multiobjective sensor node deployment in wireless sensor networks. *International Journal of Engineering Science and Technology*, 4(4):1262–1266.

- Rebai, M., le Berre, M., Hnaien, F., e Snoussi, H. (2016). Exact biobjective optimization methods for camera coverage problem in three-dimensional areas. *IEEE Sensors Journal*, 16(9):3323–3331.
- Rosário, A. T. e Dias, J. C. (2023). How industry 4.0 and sensors can leverage product design: Opportunities and challenges. *Sensors*, 23(3).
- Saeed, F., Paul, A., Rehman, A., Hong, W. H., e Seo, H. (2018). IoT-based intelligent modeling of smart home environment for fire prevention and safety. *Journal of Sensor and Actuator Networks*, 7(1).
- Sharma, H., Haque, A., e Blaabjerg, F. (2021). Machine learning in wireless sensor networks for smart cities: A survey. *Electronics (Switzerland)*, 10(9).
- Sharma, V., Patel, R., Bhaduria, H., e Prasad, D. (2016). Deployment schemes in wireless sensor network to achieve blanket coverage in large-scale open area: A review. *Egyptian Informatics Journal*, 17(1):45–56.
- Suresh, M. S. S. e Menon, V. (2023). A generic and scalable approach to maximize coverage in diverse indoor and outdoor multicamera surveillance scenarios. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 53(2):1172–1182.
- Tezcan, N. e Wang, W. (2008). Self-orienting wireless multimedia sensor networks for occlusion-free viewpoints. *Computer Networks*, 52(13):2558–2567. (1) Research and Trials for Reliable VoIP Applications (2) Wireless Multimedia Sensor Networks.
- Tian, W., Li, H., Zhu, H., Wang, Y., Liu, X., Yang, R., Xie, Y., Zhang, M., Zhu, J., e Wang, X. (2024). A review of smart camera sensor placement in construction. *Buildings*, 14(12).
- Toumpis, S. e Tassiulas, L. (2006). Optimal deployment of large wireless sensor networks. *IEEE Transactions on Information Theory*, 52(7):2935–2953.
- Tripathi, A., Gupta, H. P., Dutta, T., Mishra, R., Shukla, K. K., e Jit, S. (2018). Coverage and connectivity in wsns: A survey, research issues and challenges. *IEEE Access*, 6:26971–26992.
- van Laarhoven, P. J. M. e Aarts, E. H. L. (1987). *Simulated annealing*, páginas 7–15. Springer Netherlands, Dordrecht.
- Vincze, Z., Vida, R., e Vidacs, A. (2007). Deploying multiple sinks in multi-hop wireless sensor networks. In *IEEE International Conference on Pervasive Services*, páginas 55–63.
- Wang, C., Wang, B., e Liu, W. (2011). Movement strategies for improving barrier coverage in wireless sensor networks: A survey. In *2011 IEEE 13th International Conference on Communication Technology*, páginas 938–943.

- Wang, H.-L. e Chung, W.-H. (2012). The generalized k-coverage under probabilistic sensing model in sensor networks. In *2012 IEEE Wireless Communications and Networking Conference (WCNC)*, páginas 1737–1742.
- Wang, Y.-C., Hu, C.-C., e Tseng, Y.-C. (2005). Efficient deployment algorithms for ensuring coverage and connectivity of wireless sensor networks. In *First International Conference on Wireless Internet (WICON'05)*, páginas 114–121.
- Wang, Y.-C. e Tseng, Y.-C. (2008). Distributed deployment schemes for mobile wireless sensor networks to ensure multilevel coverage. *IEEE Transactions on Parallel and Distributed Systems*, 19(9):1280–1294.
- Wang, Z. e Rangaiah, G. P. (2017). Application and Analysis of Methods for Selecting an Optimal Solution from the Pareto-Optimal Front obtained by Multiobjective Optimization. *Industrial & Engineering Chemistry Research*, 56(2):560–574.
- Wu, H., Liu, Z., Hu, J., e Yin, W. (2020). Sensor placement optimization for critical-grid coverage problem of indoor positioning. *International Journal of Distributed Sensor Networks*, 16(12):1550147720979922.
- Yang, L. e Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, 415:295–316.
- Yang, Q., He, S., Li, J., Chen, J., e Sun, Y. (2015). Energy-efficient probabilistic area coverage in wireless sensor networks. *IEEE Transactions on Vehicular Technology*, 64(1):367–377.
- Yang, X.-S. e Deb, S. (2009). Cuckoo search via lévy flights. In *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, páginas 210–214.
- Yang, X.-S., Karamanoglu, M., e He, X. (2013). Multi-objective flower algorithm for optimization. *Procedia Computer Science*, 18:861–868. 2013 International Conference on Computational Science.
- Yetgin, H., Cheung, K. T. K., El-Hajjar, M., e Hanzo, L. (2015). Network-lifetime maximization of wireless sensor networks. *IEEE Access*, 3:2191–2226.
- Zanella, A., Bui, N., Castellani, A., Vangelista, L., e Zorzi, M. (2014). Internet of things for smart cities. *IEEE Internet of Things Journal*, 1(1):22–32.
- Zhang, D. e Zhang, J. (2020). Multi-species evolutionary algorithm for wireless visual sensor networks coverage optimization with changeable field of views. *Applied Soft Computing*, 96:106680.
- Zhu, X. e Zhou, M. (2024). Optimal deployment of heterogeneous visual sensor networks for target coverage and feature aggregation with lifetime guarantee. In *2024 International Conference on Networking, Sensing and Control (ICNSC)*, páginas 1–6.

- Zitzler, E., Deb, K., e Thiele, L. (2000). Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary Computation*, 8(2):173–195.
- Zitzler, E. e Thiele, L. (1998). Multiobjective optimization using evolutionary algorithms — a comparative case study. In Eiben, A. E., Bäck, T., Schoenauer, M., e Schwefel, H.-P., editores, *Parallel Problem Solving from Nature — PPSN V*, páginas 292–301, Berlin, Heidelberg. Springer Berlin Heidelberg.