



CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
ENGENHARIA DE COMPUTAÇÃO
INTELIGÊNCIA ARTIFICIAL

TRABALHO 2

**ÁLVARO AUGUSTO JOSÉ SILVA
JOÃO PAULO DA CUNHA FARIA**

DIVINÓPOLIS - MG
DEZEMBRO DE 2025

Sumário

Resumo	2
Parte 1: árvore, diagrama mermaid, exemplos de execução	2
Parte 1: Árvore de Decisão Manual (Escala de Glasgow)	2
1.1 Modelagem do Problema	2
1.2 Diagrama de Fluxo (Mermaid)	2
1.3 Exemplos de Execução	3
Cenário A: Paciente com Trauma Leve	3
Cenário B: Paciente com Trauma Grave	4
Parte 2: Aprendizado Supervisionado (Kaggle)	5
2.1 Descrição do Dataset	5
2.2 Pré-processamento e Tratamento de Dados	5
2.3 Resultados Individuais dos Algoritmos	5
I. Support Vector Machine (SVM)	5
II. K-Nearest Neighbors (KNN)	6
III. Árvore de Decisão (Decision Tree)	6
2.4 Análise das Figuras	6
2.5 Compilado e Discussão de Resultados	7
Parte 3: Algoritmo Genético (AG) — Distribuição de Antenas	8
3.1 Modelagem e Abordagem	8
3.2 Operadores Evolutivos e Configuração	8
3.3 Função de Aptidão (Fitness)	9
3.4 Impacto dos Parâmetros do Sistema	9
3.5 Análise do Tempo de Execução	9
3.6 Resultados e Análise Visual	10
Parte 4: Inteligência de Enxame e Sistemas Imunológicos — PSO e CLONALG	10
4.1 Detalhes de Implementação e Cenário Dinâmico	11
4.2 Particle Swarm Optimization (PSO)	11
4.3 Algoritmo de Seleção Clonal (CLONALG)	11
4.4 Análise dos Resultados e Visualização	12
4.5 Comparação Qualitativa entre as Abordagens	13
Conclusões	13

Resumo

Parte 1: árvore, diagrama mermaid, exemplos de execução

Parte 1: Árvore de Decisão Manual (Escala de Glasgow)

1.1 Modelagem do Problema

Nesta etapa, foi implementado um sistema baseado em regras (árvore de decisão manual) para automatizar o cálculo da **Escala de Coma de Glasgow (ECG)**. O objetivo é auxiliar na triagem de pacientes com traumatismo cranioencefálico, classificando a gravidade da lesão com base em três critérios neurológicos distintos:

1. **Abertura Ocular (1 a 4 pontos):** Avalia o nível de consciência visual.
2. **Resposta Verbal (1 a 5 pontos):** Avalia a cognição e capacidade de comunicação.
3. **Resposta Motora (1 a 6 pontos):** Avalia a capacidade de obedecer a comandos e reflexos motores.

O algoritmo soma as pontuações obtidas em cada critério, gerando um resultado final entre 3 e 15. A classificação final segue a lógica implementada no código:

- **13–15 pontos:** Trauma Leve.
- **9–12 pontos:** Trauma Moderado.
- **3–8 pontos:** Trauma Grave (com alerta de intubação).

1.2 Diagrama de Fluxo (Mermaid)

A estrutura lógica do algoritmo foi modelada utilizando a linguagem de diagramação Mermaid. O fluxo consiste em três blocos sequenciais de perguntas binárias (Sim/Não), onde a resposta afirmativa atribui uma pontuação imediata e encerra o bloco, enquanto a negativa leva à próxima pergunta de menor gravidade.

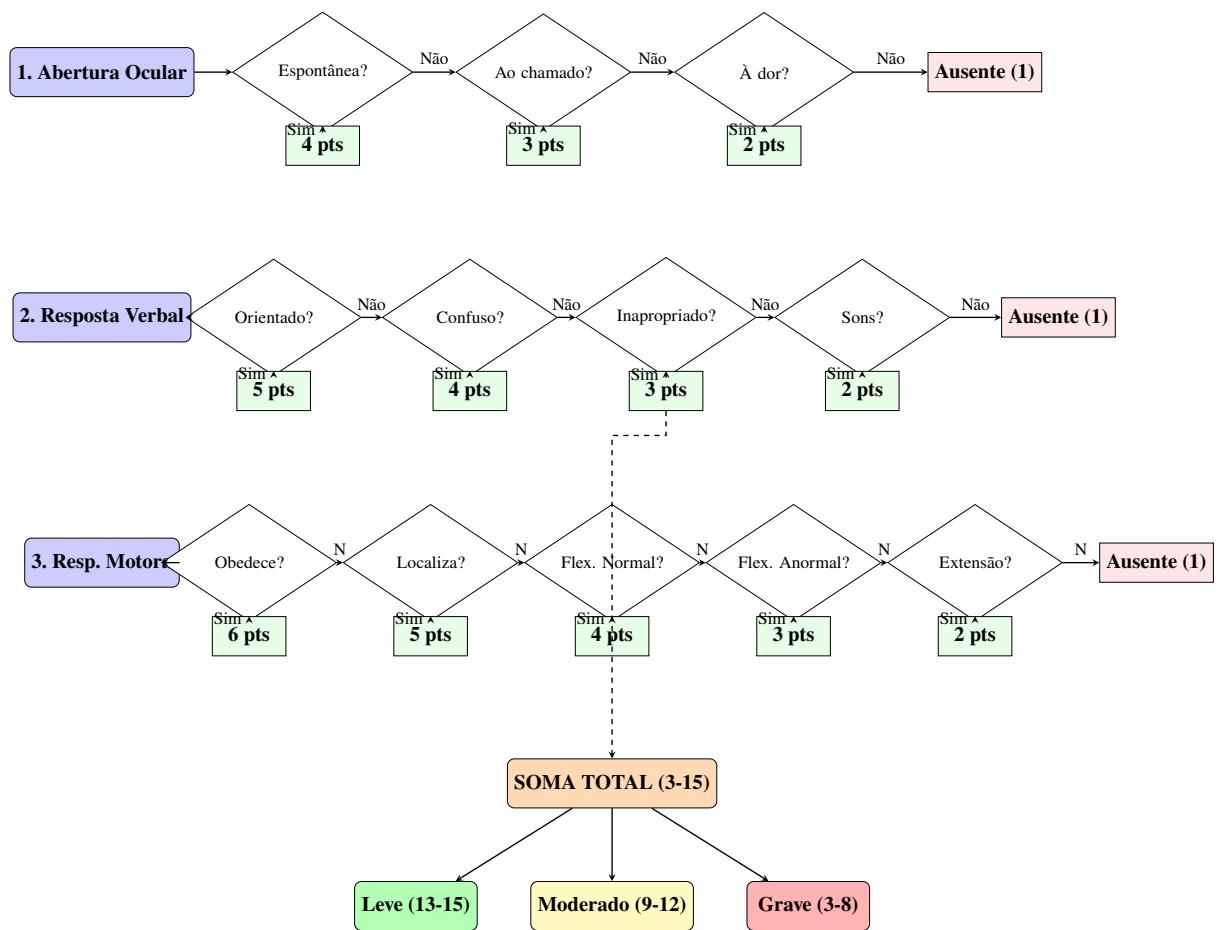


Figura 1 – Fluxograma da Escala de Glasgow (Layout Horizontal).

1.3 Exemplos de Execução

Abaixo são apresentados dois cenários de execução do algoritmo desenvolvido, demonstrando o funcionamento para diferentes gravidades de trauma.

Cenário A: Paciente com Trauma Leve

Neste exemplo, o paciente está consciente, orientado e responde a comandos. O sistema calcula a pontuação máxima (15).

```

1 =====
2   CALCULADORA DA ESCALA DE GLASGOW
3   Responda as perguntas com 's' ou 'n'
4 =====
5
6 --- 1. ABERTURA OCULAR ---
7 Abre os olhos espontaneamente? (s/n): s
8
9 --- 2. RESPOSTA VERBAL ---
10 O paciente est  orientado e conversa? (s/n): s
11

```

```

12 --- 3. RESPOSTA MOTORA ---
13 Obedece a comandos? (s/n): s
14
15 =====
16 RESULTADO FINAL: 15 PONTOS
17 CLASSIFICA 0: TRAUMA LEVE
18
19 Detalhamento: Ocular(4) + Verbal(5) + Motora(6)
20 =====

```

Listing 1 – Output do Terminal - Trauma Leve

Cenário B: Paciente com Trauma Grave

Neste exemplo, o paciente apresenta reações apenas a estímulos dolorosos e extensão anormal, caracterizando um quadro crítico que exige intubação.

```

1 =====
2   CALCULADORA DA ESCALA DE GLASGOW
3   Responda as perguntas com 's' ou 'n'
4 =====
5
6 --- 1. ABERTURA OCULAR ---
7 Abre os olhos espontaneamente? (s/n): n
8 Abre os olhos ao comando verbal? (s/n): n
9 Abre os olhos press o/dor? (s/n): s
10
11 --- 2. RESPOSTA VERBAL ---
12 O paciente est orientado e conversa? (s/n): n
13 A conversa confusa? (s/n): n
14 Usa palavras inapropriadas? (s/n): n
15 Emite sons inintelig veis? (s/n): s
16
17 --- 3. RESPOSTA MOTORA ---
18 Obedece a comandos? (s/n): n
19 Localiza o est mulo doloroso? (s/n): n
20 Apresenta flex o normal (movimento de retirada)? (s/n): n
21 Apresenta flex o anormal (decortica o)? (s/n): n
22 Apresenta extens o anormal (descerebra o)? (s/n): s
23
24 =====
25 RESULTADO FINAL: 6 PONTOS
26 CLASSIFICA 0: TRAUMA GRAVE
27 !!! ATEN 0: CONSIDERAR INTUBA 0 !!!
28
29 Detalhamento: Ocular(2) + Verbal(2) + Motora(2)
30 =====

```

Listing 2 – Output do Terminal - Trauma Grave

Parte 2: Aprendizado Supervisionado (Kaggle)

2.1 Descrição do Dataset

Para esta etapa, o problema escolhido foi de **Classificação**. O conjunto de dados selecionado foi o **Heart Disease Dataset**, obtido via Kaggle.

- **Link de acesso:** <<https://www.kaggle.com/datasets/johnsmith88/heart-disease-dataset>>
- **Objetivo:** Identificar a presença de doenças cardíacas através de indicadores clínicos.
- **Composição:** O dataset original possui 1025 instâncias com variáveis numéricas e categóricas.

2.2 Pré-processamento e Tratamento de Dados

Foram aplicados tratamentos essenciais para garantir a reproduzibilidade e eficácia dos modelos:

1. **Limpeza:** Verificação de integridade e confirmação de ausência de valores nulos.
2. **Escalonamento:** Uso de *StandardScaler* para normalizar os dados, prevenindo que atributos com escalas maiores dominem o cálculo de distância no KNN e SVM.
3. **Divisão da Base:** Separação estratificada em 80

2.3 Resultados Individuais dos Algoritmos

I. Support Vector Machine (SVM)

O SVM foi treinado com o kernel RBF. Embora tenha apresentado estabilidade durante a validação cruzada, obteve os resultados de acurácia mais conservadores do grupo.

- **Acurácia de Teste:** 0.64.
- **Cross-Validation Scores:** [0.67, 0.75, 0.63, 0.72, 0.71].
- **Média CV:** 0.70.

II. K-Nearest Neighbors (KNN)

O modelo KNN foi otimizado através da busca pelo valor ideal de vizinhos (k) que minimizasse o erro de validação.

- **Melhor K identificado:** 17.
- **Acurácia:** 0.73.
- **Precisão e Recall (Macro):** 0.73.

III. Árvore de Decisão (Decision Tree)

A árvore de decisão apresentou o desempenho mais elevado em termos de acurácia bruta, mostrando-se eficaz na separação das classes de risco cardíaco.

- **Acurácia:** 0.76.
- **Precisão:** 0.81.
- **Recall:** 0.75.

2.4 Análise das Figuras

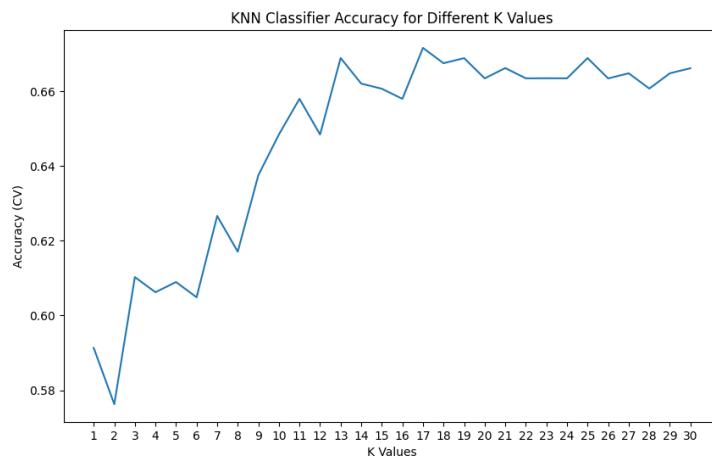


Figura 2 – Acurácia do KNN por valor de K.

Descrição da Figura 2: Este gráfico demonstra o comportamento do KNN conforme o número de vizinhos aumenta. Observa-se que valores muito baixos de k resultam em instabilidade. O ponto ideal foi fixado em $k = 17$, onde a acurácia atinge um platô de estabilidade antes de sofrer possíveis simplificações excessivas.

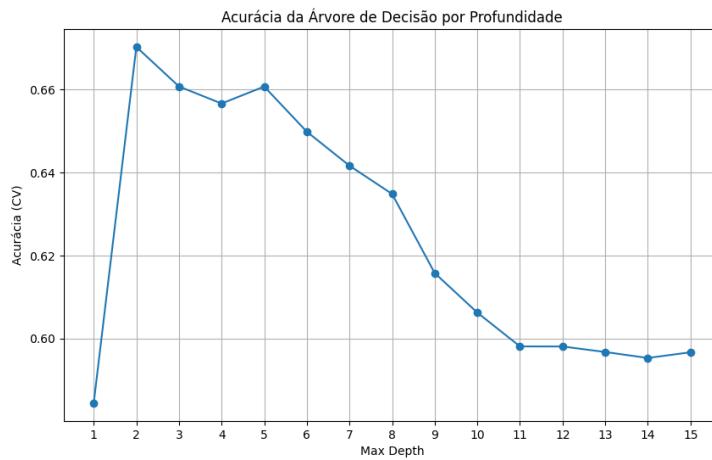


Figura 3 – Acurácia da Árvore de Decisão por Profundidade.

Descrição da Figura 3: A imagem ilustra a relação entre a profundidade máxima da árvore e a acurácia. Nota-se que o pico de desempenho ocorre em profundidade 2. Após esse ponto, a acurácia tende a cair, indicando que árvores mais profundas sofrem de *overfitting*, memorizando ruídos do treino em vez de padrões generalizáveis.

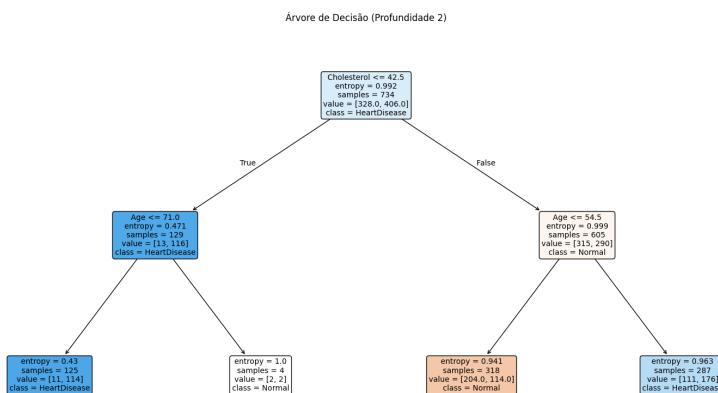


Figura 4 – Visualização da Estrutura da Árvore (Profundidade 2).

Descrição da Figura 4: Apresenta o modelo final de decisão. O nó raiz utiliza o nível de *Cholesterol* como critério primário de divisão, seguido pela *Age* (Idade). Esta visualização permite entender a hierarquia de importância das variáveis no diagnóstico clínico gerado pelo modelo.

2.5 Compilado e Discussão de Resultados

O processo de avaliação demonstrou que, para este dataset, modelos baseados em árvores são ligeiramente mais eficientes que modelos baseados em distância.

Tabela 1 – Tabela Comparativa Final das Métricas (Classificação).

Algoritmo	Acurácia	Precisão	Recall (Revocação)	F1-Score (Macro)
Árvore de Decisão	0.76	0.81	0.75	0.76
KNN ($k = 17$)	0.73	0.75	0.75	0.73
SVM (RBF)	0.64	0.66*	-	-

*Valor referente à média da validação cruzada.

Conclusão da Parte 2: A **Árvore de Decisão** é recomendada como o modelo principal para este problema, pois obteve a melhor precisão (0.81), minimizando falsos positivos. O **KNN** mostrou-se uma alternativa robusta e equilibrada, enquanto o **SVM** exigiria um ajuste de hiperparâmetros (como busca por C e γ) mais agressivo para atingir os patamares dos demais.

Parte 3: Algoritmo Genético (AG) — Distribuição de Antenas

3.1 Modelagem e Abordagem

O problema abordado é uma adaptação do *Maximum Coverage Location Problem* (MCLP). Diferente de problemas de otimização em espaços discretos, este modelo opera em um plano contínuo, onde o objetivo é encontrar a melhor configuração conjunta de até 10 antenas para maximizar o atendimento aos clientes.

A representação cromossômica (genótipo) consiste em um vetor de números reais e booleanos, onde cada antena é definida por três genes: coordenada X, coordenada Y e um bit de ativação. A escolha do Algoritmo Genético justifica-se pela natureza não linear da função de aptidão e pela alta dimensionalidade do espaço de busca (10 antenas×3 genes cada=30 dimensões).

3.2 Operadores Evolutivos e Configuração

Para garantir a exploração (*exploration*) e o aproveitamento (*exploitation*) do espaço de busca, foram implementados os seguintes operadores:

- **Seleção:** Foi utilizado o método de **Seleção por Torneio** (tamanho 3). Este método foi escolhido por manter uma pressão seletiva constante e evitar a dominância prematura de indivíduos “super-fit”, o que é comum em problemas de localização espacial.
- **Cruzamento (Crossover):** Aplicou-se o **Cruzamento de Ponto Único** com uma taxa de **80%**. Dada a estrutura do cromossomo organizada por blocos de antenas, este operador permite que subconjuntos de posicionamentos bem-sucedidos sejam transmitidos integralmente para os descendentes.
- **Mutação:** Utilizou-se a **Mutação Gaussiana** para as coordenadas (ajuste fino de posição) e **Bit-flip** para o estado de ativação, com uma taxa de **10%** por gene. Essa taxa relativamente

alta é necessária para evitar que o algoritmo fique preso em ótimos locais, especialmente quando as antenas convergem para o mesmo cluster de clientes.

- **Critério de Parada:** O algoritmo encerra sua execução após um limite fixo de **100 gerações** ou caso não haja melhoria no melhor indivíduo da população (**estagnação**) por 20 gerações consecutivas.

3.3 Função de Aptidão (Fitness)

A função de *fitness* foi desenhada para equilibrar três objetivos conflitantes: $\text{Fitness} = C(S)(N) \text{Penalidades}$
Onde:

- C: Número de clientes cobertos (distância 30m de uma antena ativa).
- S: Sobreposição de sinal entre antenas (penalidade para evitar redundância).
- N: Número de antenas ativas (incentivo à economia de recursos).
- **Penalidades:** Atribuídas a antenas posicionadas fora do raio limite de 100m para garantir a viabilidade geográfica.

3.4 Impacto dos Parâmetros do Sistema

A dinâmica de convergência é diretamente afetada pelas constantes do sistema:

- **Quantidade de Clientes (N=80):** Define a "resolução" do problema. Uma densidade maior de clientes permite que o AG identifique *clusters* populacionais com mais clareza.
- **Raio de Alcance (30.0m):** Determina a sensibilidade da solução. Um raio curto exige precisão cirúrgica no posicionamento, enquanto um raio longo facilita a cobertura, mas aumenta o risco de sobreposições ineficientes.
- **Raio do Campo (100.0m):** Atua como a restrição de contorno, definindo o grau de dificuldade do "empacotamento" espacial.

3.5 Análise do Tempo de Execução

O tempo de execução é influenciado pela complexidade computacional da função de aptidão, estimada em $O(GPCA)$, onde G são gerações, P a população, C os clientes e A as antenas. A necessidade de calcular a distância euclidiana entre cada cliente e cada antena ativa para todos os indivíduos de todas as gerações é a principal causa do custo computacional elevado.

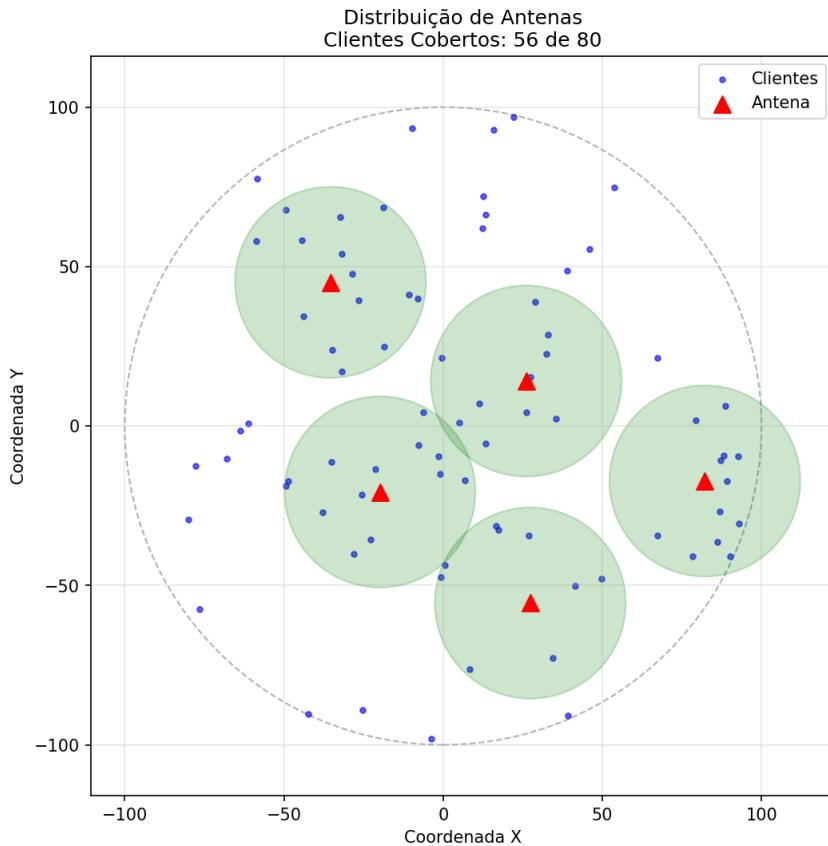


Figura 5 – Distribuição final das antenas e clientes no plano operacional.

3.6 Resultados e Análise Visual

O experimento resultou na cobertura de 56 dos 80 clientes (70%

A análise da Figura 5 revela comportamentos estratégicos do algoritmo:

- **Clusterização Tática:** As antenas foram posicionadas em zonas de alta densidade. O AG optou por ignorar clientes isolados nas bordas para maximizar o retorno sobre o uso de cada antena.
- **Otimização de Ativos:** Mesmo com um limite de 10 antenas, o algoritmo utilizou apenas 5 unidades ativas, demonstrando que a penalidade por excesso de equipamentos foi eficaz.
- **Conformidade Geométrica:** Todas as antenas foram mantidas dentro da fronteira operacional, validando o sistema de penalidades por coordenadas inválidas.

Parte 4: Inteligência de Enxame e Sistemas Imunológicos — PSO e CLONALG

Nesta etapa, o problema de otimização de cobertura foi abordado através de duas meta-heurísticas distintas: o *Particle Swarm Optimization* (PSO), baseado no comportamento social de bandos, e o *CLONALG*, inspirado na teoria da seleção clonal do sistema imunológico biológico.

4.1 Detalhes de Implementação e Cenário Dinâmico

Diferente das abordagens estáticas, a implementação foi projetada para testar a robustez dos algoritmos em cenários mutáveis.

- **Dinamismo dos Clientes:** Em cada execução do código, as posições dos 80 clientes são geradas de forma aleatória dentro do campo de 100×100 metros. Isso garante que o algoritmo não esteja "viciado" em uma topologia específica, exigindo uma capacidade real de generalização.
- **Representação do Espaço de Busca:** O vetor de decisão (partícula ou anticorpo) é estruturado como um arranjo unidimensional de tamanho $2N$ (onde N é o número de antenas), contendo as coordenadas (x, y) de cada equipamento.
- **Restrição de Margem:** Foi implementada uma zona de exclusão de 10 metros nas bordas do terreno ($LIMITE_MIN = 10$, $LIMITE_MAX = 90$). Essa restrição força os algoritmos a encontrar soluções que cubram clientes periféricos sem posicionar as antenas fora da área útil permitida, simulando limitações reais de instalação em propriedades privadas ou áreas de preservação.

4.2 Particle Swarm Optimization (PSO)

O PSO baseia-se na premissa de que a inteligência emerge da colaboração social. Cada "partícula" voa pelo espaço de busca ajustando sua trajetória com base em:

1. Sua própria inércia ($w = 0.7$);
2. Sua melhor experiência individual ($pbest$);
3. A melhor experiência compartilhada pelo grupo ($gbest$).

A aplicação do *clipping* (função `np.clip`) foi fundamental para manter a viabilidade das soluções dentro da margem de 10m. O PSO demonstrou uma capacidade excepcional de "sugar" as antenas em direção aos centros de massa dos clientes, apresentando uma convergência fluida devido à natureza vetorial contínua do movimento das partículas.

4.3 Algoritmo de Seleção Clonal (CLONALG)

O CLONALG aborda o problema tratando cada configuração de antenas como um "anticorpo". A eficácia (aptidão) da solução determina sua taxa de reprodução:

- **Clonagem:** Soluções com alta cobertura são clonadas extensivamente.
- **Hipermutação Somática:** Os clones sofrem mutações com uma taxa inversamente proporcional à sua aptidão. Isso significa que soluções quase perfeitas sofrem ajustes finos

(mutação baixa), enquanto soluções ruins são radicalmente alteradas para explorar novas áreas do mapa.

Essa dualidade entre exploração e aproveitamento permitiu que o CLONALG mantivesse uma diversidade populacional alta, evitando que todas as antenas se aglomerassesem em um único ponto de alta densidade de clientes.

4.4 Análise dos Resultados e Visualização

Ambos os algoritmos atingiram a marca de **100,00% de cobertura** nos testes realizados, demonstrando uma superioridade prática em relação ao Algoritmo Genético convencional para este número específico de sensores.

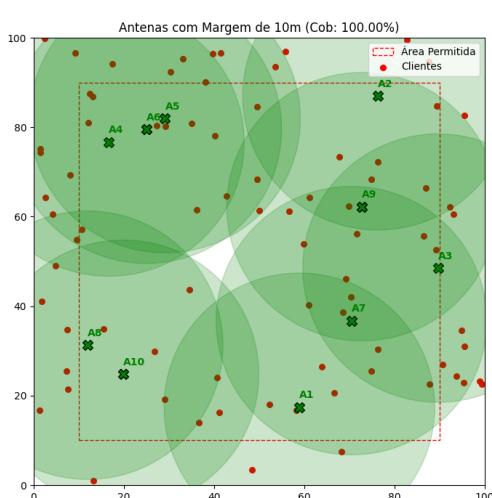


Figura 6 – Resultado final da otimização via PSO.

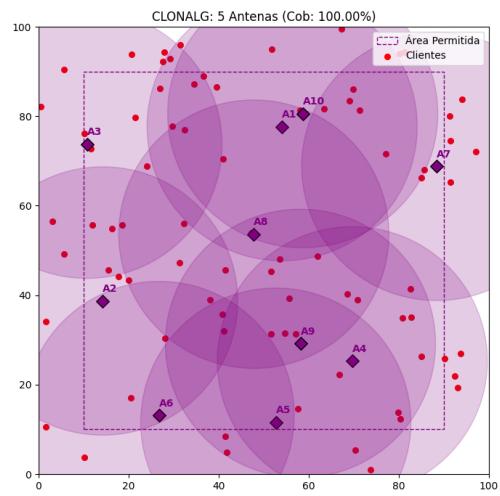


Figura 7 – Resultado final da otimização via CLONALG.

As Figuras 6 e 7 ilustram as estratégias adotadas:

- **Geometria de Cobertura:** Observa-se que os algoritmos posicionam as antenas de forma a "envelopar" os aglomerados de clientes. As áreas de sobreposição (tonalidades mais escuras de verde/roxo) ocorrem apenas onde a densidade de clientes é tão dispersa que uma única antena não seria suficiente.
- **Uso da Margem:** É visível o respeito à linha tracejada vermelha/roxa. Mesmo com clientes localizados próximos às coordenadas (0,0) ou (100,100), as antenas permanecem dentro do limite de 10m, utilizando o alcance do sinal ($RAIO = 15m$) para "alcançar" os clientes externos sem violar a restrição de posicionamento físico.
- **Distribuição de Esforço:** No CLONALG, as antenas rotuladas (A1 a A10) mostram uma distribuição bem distribuída, preenchendo vazios espaciais de forma quase geométrica, o que minimiza o desperdício de sinal em áreas sem clientes.

4.5 Comparação Qualitativa entre as Abordagens

Apesar de compartilharem o mesmo objetivo e restrições geométricas, as três abordagens apresentam comportamentos distintos na forma como "enxergam" e exploram o plano operacional:

- **Exploração vs. Refinamento:** O **Algoritmo Genético (AG)** baseia sua busca na recombinação de soluções. Isso permite saltos globais no espaço de busca, mas pode causar a fragmentação de boas configurações geográficas durante o *crossover*. Em contraste, o **PSO** apresenta um movimento "fluido" e vetorial; as antenas deslizam pelo plano como um bando, o que facilita o ajuste fino em torno dos aglomerados de clientes. Já o **CLONALG** utiliza a hipermutação como uma ferramenta de precisão, refinando localmente cada antena após identificar uma área de interesse.
- **Tratamento de Zonas de Sombra:** O AG demonstrou uma tendência a priorizar grandes concentrações de clientes, aceitando deixar indivíduos isolados em "zonas de sombra" para manter a economia de recursos e evitar penalidades. Por outro lado, as técnicas da Seção 4 (PSO e CLONALG) mostraram-se mais resilientes na busca pela cobertura total. Elas conseguem "esticar" a configuração das antenas até os limites da margem permitida, garantindo que mesmo os clientes periféricos sejam atendidos sem violar as restrições de borda.
- **Coordenação Coletiva:** No PSO, existe uma consciência social imediata (através do *gbest*), onde todas as antenas de todas as partículas colaboram para cobrir o vácuo deixado pelas outras. No AG, essa coordenação é indireta e baseada na sobrevivência do mais apto, o que pode levar a uma convergência prematura em soluções que cobrem bem o centro, mas negligenciam as extremidades.
- **Adaptabilidade ao Cenário Dinâmico:** Devido ao fato de os clientes serem reposicionados a cada execução, o PSO e o CLONALG demonstraram maior robustez. Enquanto o AG exige um ajuste mais fino de suas taxas de mutação e seleção para não se "viciar" em uma topologia, a memória imunológica do CLONALG e a inércia social do PSO permitem uma reconfiguração espacial mais rápida e eficiente diante de novas distribuições de clientes.

Esta comparação evidencia que, para problemas de localização contínua com alta exigência de cobertura (100

Conclusões

O trabalho evidenciou que, para problemas de localização contínua com restrições de margem, algoritmos baseados em enxames e sistemas imunes oferecem uma estabilidade de convergência superior. Enquanto o AG tradicional pode sofrer com a "quebra" de boas soluções durante o

crossover, o movimento contínuo do PSO e a hipermutação controlada do CLONALG permitem um refinamento mais preciso das coordenadas, resultando em cobertura total do cenário proposto.

Referências

Repositório do Projeto:

- SANTOS, Alvaro. *IA Trabalho 2025.2 - Implementação de Algoritmo Genético e Otimização*. GitHub, 2025. Disponível em: <<https://github.com/alvaroajs/ia-trabalho-2025-2>>. Acesso em: 21 dez. 2025.