



RELATÓRIO

Thiago Abreu (nº31523)

Professor Luís Ferreira

Conteúdo

| | |
|--------------------------------------------------------|----|
| Resumo..... | 4 |
| 1. Introdução | 5 |
| 1.1. Motivação..... | 5 |
| 1.2. Enquadramento..... | 5 |
| 1.3. Objetivos | 6 |
| 1.4. Metodologia do Trabalho | 6 |
| 2. Enquadramento Teórico e Prático..... | 7 |
| 2.1. Conceitos Teóricos e Tecnologias a Explorar | 7 |
| 2.2. Soluções (técnicas) Existentes..... | 8 |
| 3. Especificação do Problema | 9 |
| 3.1. Contextualização do Problema | 9 |
| 3.2. Requisitos Funcionais..... | 10 |
| 3.3. Ferramentas utilizadas | 10 |
| 3.3.1. Linguagem C | 10 |
| 3.3.2. Visual Studio Code..... | 10 |
| 3.3.3. Windows Subsystem for Linux (WSL)..... | 11 |
| 3.3.4. Multiple Cursor Case Preserve | 11 |
| 3.3.5. Inteligência Artificial – “ChatGPT” | 11 |
| 3.3.6. Doxygen..... | 11 |
| 3.3.7. WinRAR..... | 11 |
| 3.3.8. Microsoft Word..... | 11 |
| 3.3.9. Inteligência Artificial – “MyMap” | 12 |
| 4. Modelação da Solução..... | 13 |
| 4.1. Representação de Antenas com Listas Ligadas | 13 |
| 4.2. Operações sobre a Lista de Antenas | 14 |
| 4.3. Detecção de Locais com Efeito Nefasto..... | 14 |
| 4.4. Modelação do Grafo..... | 15 |
| 4.5. Algoritmos de Procura | 15 |
| 4.6. Identificação de Caminhos e Interseções | 16 |
| 5. Implementação e Testes | 17 |
| 5.1. Organização Modular | 17 |
| 5.2. Casos de Teste..... | 17 |
| 5.2.1. Leitura e Construção da Lista de Antenas | 17 |
| 5.3.2 - Remoção de Antenas..... | 18 |

| | |
|-----------------------------------------------|----|
| 5.3.3 - Detecção de Locais Nefastos..... | 18 |
| 5.3.4 - Estrutura de Grafo e Ligações..... | 18 |
| 5.3.5 - Algoritmos de Procura e Caminhos..... | 18 |
| 5.3.6 - Interseções entre Frequências..... | 18 |
| 5.4. Resultados Obtidos..... | 18 |
| 5.5 Desafios Encontrados..... | 18 |
| 6. Conclusão..... | 20 |
| 7. Glossário e Siglas..... | 21 |
| 8. Bibliografia..... | 22 |
| 9. Tabela de Figuras..... | 23 |
| 10. Repositório no Github..... | 24 |

Resumo

Este projeto teve como finalidade o desenvolvimento de uma solução informática orientada à identificação e análise de interações entre antenas de ressonância numa cidade, com foco na deteção de locais com potencial efeito nefasto. Partindo de uma matriz textual representativa de um mapa urbano, procurou-se construir uma estrutura de dados dinâmica eficiente capaz de armazenar, manipular e processar informação relativa às antenas e às suas frequências. Foram exploradas listas ligadas e grafos como modelos estruturais fundamentais, permitindo representar com precisão relações espaciais e conexões por frequência. Através de operações de inserção, remoção, leitura de ficheiros e travessias em profundidade e largura, foi possível detetar interferências e caminhos entre dispositivos. A documentação detalhada do código foi realizada com o auxílio da ferramenta Doxygen, promovendo clareza e reusabilidade do software. O projeto concretiza uma abordagem prática ao problema, demonstrando aplicabilidade dos conceitos teóricos da unidade curricular de Estruturas de Dados Avançadas, com resultados funcionais e bem estruturados.

1. Introdução

O presente capítulo pretende enquadrar o problema que serviu de base ao projeto, clarificando a sua motivação, pertinência no contexto atual das tecnologias de comunicação e a proposta de abordagem ao mesmo. Serão ainda apresentados os objetivos definidos, a metodologia adotada e o plano de trabalho executado ao longo do desenvolvimento da solução.

1.1. Motivação

As redes de comunicação modernas tornaram-se elementos essenciais na infraestrutura de qualquer cidade. Com o aumento da densidade de antenas e dispositivos emissores, surgem desafios técnicos como a **interferência de sinais**, especialmente quando antenas operam na mesma frequência. Tal fenómeno pode impactar negativamente serviços críticos, como redes móveis, sistemas de segurança e redes de sensores. A necessidade de compreender e mapear estes efeitos é evidente em diversos contextos, nomeadamente em ambientes urbanos com sobreposição de cobertura. Apesar da existência de ferramentas comerciais para modelação de redes, a implementação de soluções algorítmicas simples e personalizáveis continua a ser um campo relevante de estudo.

1.2. Enquadramento

Este projeto foi desenvolvido no âmbito da Unidade Curricular de Estruturas de Dados Avançadas, integrada no plano de estudos da licenciatura em Engenharia Informática da Escola Superior de Tecnologia do Instituto Politécnico do Cávado e do Ave (EST-IPCA). Inserido no segundo semestre do primeiro ano, o trabalho teve como base os conteúdos lecionados sobre estruturas dinâmicas, alocação de memória, modularização e manipulação de ficheiros. O projeto visa aplicar estes conhecimentos de forma prática, através da criação de uma solução técnica robusta e documentada.

1.3. Objetivos

O projeto teve como principais objetivos:

- Desenvolver uma estrutura de dados dinâmica para representar uma cidade e as antenas instaladas;
- Implementar algoritmos que identifiquem locais com efeito nefasto resultante de interações entre antenas de mesma frequência;
- Modelar o problema através de grafos, permitindo a análise de ligações entre antenas;
- Garantir a modularização do código e documentar automaticamente o sistema;
- Criar uma interface por linha de comandos para leitura de ficheiros, operações sobre as estruturas e visualização de resultados.

1.4. Metodologia do Trabalho

O desenvolvimento seguiu uma metodologia iterativa e incremental, dividida em duas grandes fases:

1. Estruturação das antenas e deteção de locais nefastos com listas ligadas;
2. Conversão da estrutura para grafos, permitindo análise de conectividade entre antenas.

A cada fase, procedeu-se à implementação modular de funcionalidades, realização de testes e documentação das soluções. A ferramenta Doxygen foi integrada desde o início, promovendo a consistência da documentação ao longo do desenvolvimento.

2. Enquadramento Teórico e Prático

O presente capítulo visa enquadrar teoricamente o trabalho desenvolvido, através da análise de conceitos fundamentais, investigações relacionadas e soluções técnicas existentes no domínio da representação e análise de redes com estruturas dinâmicas. Serão abordados os principais fundamentos teóricos aplicáveis à modelação de estruturas espaciais, bem como ferramentas ou abordagens com finalidades semelhantes, procedendo-se a uma comparação crítica entre essas abordagens e a metodologia adotada neste projeto.

2.1. Conceitos Teóricos e Tecnologias a Explorar

A base teórica do projeto assenta em três grandes pilares: **estruturas de dados dinâmicas, representação de grafos e análise de relações espaciais**. As listas ligadas, como estrutura linear, permitem armazenar conjuntos de antenas e coordenadas sem necessidade de alocação estática, sendo ideais para contextos com número variável de elementos. A manipulação destas estruturas permite realizar operações como inserção, remoção e procura, fundamentais para a gestão eficiente da cidade-modelo.

A representação de **grafos por listas de adjacência** foi explorada como forma de modelar ligações entre antenas com a mesma frequência. Este modelo é amplamente estudado na literatura (Cormen, 2009) e permite executar algoritmos clássicos de procura, como *breadth-first search* e *depth-first search*, essenciais para detetar caminhos possíveis e relações diretas entre vértices.

A componente de deteção de **efeitos nefastos** baseia-se na análise geométrica entre pares de antenas. Foram considerados alinhamentos e relações de proporcionalidade de distância, para determinar locais de potencial interferência. A deteção é feita através de operações de cálculo sobre as coordenadas, recorrendo a aritmética básica para verificar se um ponto médio entre duas antenas satisfaz as condições estipuladas.

Paralelamente, foi explorada a ferramenta **Doxygen** como tecnologia de apoio à documentação automática do projeto. Esta ferramenta interpreta as *tags* de documentação inseridas no código e gera, automaticamente, documentação técnica navegável em formato *HTML*. Esta prática, comum no desenvolvimento profissional de software, permite assegurar clareza, organização e facilidade de leitura para outros programadores ou avaliadores.

2.2. Soluções (técnicas) Existentes

A análise de interferências entre antenas e a representação de redes espaciais é um tema com aplicações em diversas áreas, nomeadamente em sistemas de telecomunicações, planeamento urbano, redes de sensores e computação gráfica. No domínio académico, muitos projetos relacionados com este tipo de problema recorrem ao uso de linguagens de alto nível como Python ou Java. No entanto, estas soluções, apesar de bastante acessíveis e poderosas, abstraem grande parte da gestão de memória e das estruturas internas, o que limita a compreensão profunda dos mecanismos de funcionamento.

Em ambientes industriais, é comum o uso de software para mapeamento e análise espacial de infraestruturas de comunicação. Como se conclui, embora existam soluções com mais capacidades técnicas e suporte gráfico, o projeto desenvolvido em C destaca-se por privilegiar a construção manual e estruturada de todo o sistema, permitindo o controlo e a compreensão total dos mecanismos envolvidos.

3. Especificação do Problema

Este capítulo pretende enunciar formalmente o problema estudado e implementado ao longo do projeto. Será apresentada a descrição detalhada do contexto, das restrições a considerar e dos requisitos funcionais esperados. A estrutura segue uma abordagem incremental, acompanhando as fases de desenvolvimento e os desafios progressivamente resolvidos. A formalização do problema permite clarificar os objetivos a atingir, bem como validar a adequação das soluções técnicas propostas.

3.1. Contextualização do Problema

Num cenário urbano onde várias antenas estão distribuídas geograficamente, cada uma sintonizada numa determinada frequência de ressonância, existe a possibilidade de interferência quando múltiplas antenas operam na mesma frequência e encontram-se alinhadas ou próximas em determinadas condições. Este fenómeno pode resultar em zonas de instabilidade eletromagnética, apelidadas de locais com efeito nefasto, cuja identificação se reveste de particular importância em sistemas críticos de comunicação e controlo. A modelação computacional deste cenário exige a representação da cidade sob a forma de uma estrutura dinâmica, onde cada antena é um elemento com propriedades associadas: frequência, posição X e Y, e ligação a outros elementos. A análise do comportamento coletivo das antenas e das suas interações implica a capacidade de percorrer a estrutura, verificar condições espaciais entre pares de antenas e deduzir implicações decorrentes da sua disposição relativa. (Rocha, 2020)

3.2. Requisitos Funcionais

A solução proposta deve garantir os seguintes comportamentos:

- Leitura de ficheiro com matriz de antenas e conversão para estrutura de dados dinâmica;
- Inserção e remoção de antenas mantendo ordenação e integridade dos dados;
- Detecção automática de posições com efeito nefasto com base em critérios geométricos;
- Representação das antenas como vértices num grafo com ligações apenas entre antenas da mesma frequência;
- Execução de algoritmos de procura em profundidade e largura para identificar caminhos entre antenas;
- Listagem de todos os caminhos possíveis entre dois vértices;
- Detecção de pares de antenas com frequências diferentes que coabitam o mesmo espaço lógico (interseções);

3.3. Ferramentas utilizadas

Neste trabalho, foram utilizadas diversas ferramentas para alcançar o resultado apresentado, sendo que cada uma delas desempenhou uma função específica, contribuindo de forma conjunta na realização do projeto. Entre estas ferramentas, destacam-se:

3.3.1. Linguagem C

A linguagem C foi a linguagem de programação utilizada para a implementação deste projeto. Considerada uma das linguagens mais fundamentais e poderosas no desenvolvimento de software, C oferece um controlo elevado sobre a gestão de memória, o que a torna especialmente adequada para a criação de estruturas de dados dinâmicas e a manipulação eficiente de recursos.

3.3.2. Visual Studio Code

O Visual Studio Code foi utilizado como o ambiente de desenvolvimento integrado (IDE) para a escrita, edição e debug do código-fonte do projeto. Com a sua interface leve e extensível, o VS Code permitiu a utilização de extensões essenciais, como o suporte para C/C++. Além disso, o terminal embutido facilitou a execução de comandos diretamente dentro da plataforma, assim permitindo um desenvolvimento mais eficiente e organizado.

3.3.3. Windows Subsystem for Linux (WSL)

O Windows Subsystem for Linux (WSL) foi utilizado neste projeto para fornecer um ambiente de desenvolvimento Linux dentro do Windows. O WSL permite a execução de ferramentas e utilitários nativos do Linux sem a necessidade de uma máquina virtual ou dual boot, garantindo maior compatibilidade com bibliotecas e compiladores frequentemente usados no desenvolvimento em C. Neste projeto, o WSL foi essencial para compilar e executar o código utilizando o GCC (GNU Compiler Collection) e para utilizar o GDB (GNU Debugger) no debug do programa.

3.3.4. Multiple Cursor Case Preserve

O Multiple Cursor Case Preserve foi uma funcionalidade utilizada no Visual Studio Code para facilitar a edição de código. Esta ferramenta permite editar múltiplas ocorrências de um texto simultaneamente, mantendo a formatação original das palavras. Isto foi útil para a correção de erros em diferentes partes do código sem a necessidade de edições repetitivas.

3.3.5. Inteligência Artificial – “ChatGPT”

A ferramenta de Inteligência Artificial – “ChatGPT” foi utilizada como um suporte no desenvolvimento do projeto, auxiliando com dúvidas, como o debug do código, explicação de conceitos complexos e na otimização do código.

3.3.6. Doxygen

O Doxygen foi a ferramenta escolhida para gerar a documentação do projeto de forma automatizada. Ele permitiu a extração de comentários estruturados do código-fonte e a conversão destes em documentação bem organizada.

3.3.7. WinRAR

O WinRAR foi utilizado para a compactação de arquivos durante o desenvolvimento do projeto. Isto foi essencial visto que como eu, pessoalmente, vivo sozinho numa residência universitária, tenho de em alguns fins de semana visitar a família, assim não conseguindo levar o meu computador principal, a minha solução foi compactar os arquivos e assim enviar para o meu portátil para dar seguimento ao meu trabalho.

3.3.8. Microsoft Word

O Microsoft Word foi utilizado para a escrita e formatação do relatório do projeto. Através das suas ferramentas, consegui obter um relatório bem estruturado e organizado para entregar juntamente do meu projeto.

3.3.9. Inteligência Artificial – “MyMap”

A ferramenta de Inteligência Artificial – “MyMap” foi utilizada para organizar e estruturar as ideias do projeto por meio de mapas mentais. Esta abordagem ajudou a visualizar a lógica do código e ao compreender o código como um conjunto.

4. Modelação da Solução

O presente capítulo descreve o processo de conceção e organização da solução técnica implementada. A abordagem foi modular, com separação clara por ficheiros, e focada em garantir a representatividade dos dados através de estruturas dinâmicas. Foram utilizados conceitos fundamentais como listas ligadas, grafos e percursos em profundidade e largura. A modelação evoluiu progressivamente, começando com a representação de antenas numa cidade e culminando com a análise da conectividade e interferência entre estas.

A cada fase do desenvolvimento, a estrutura de dados foi refinada para acomodar funcionalidades mais complexas, mantendo-se sempre a integridade dos dados e a legibilidade do código.

4.1. Representação de Antenas com Listas Ligadas

As antenas são a unidade fundamental da solução. Cada antena é caracterizada por uma frequência (representada por um carácter) e por coordenadas no espaço bidimensional. A estrutura escolhida para representar este conjunto de dados foi a lista ligada simples, de forma a permitir:

- Inserção ordenada por coordenadas;
- Remoção direta com controlo de integridade;
- Percursos lineares para análise e deteção de padrões.

Cada nó da lista é representado por uma estrutura Antena, com os campos frequência, x, y e um apontador para a próxima antena.

A figura 1 representa a estrutura sequencial da lista ligada que armazena todas as antenas da cidade, gerada pela Inteligência Artificial MyMap.



Figura 1 - Representação da lista ligada de antenas

Antenna (Antena) – Estrutura que representa cada antena e os seus detalhes.

Frequency (Frequência) – Guarda o caractere da frequência associada com a antena.

Coordinates (Coordenadas) [Antena] - Contem as integrais x e y que equivalem a localização das antenas.

Linked List (Lista Ligada) – Aponta para a próxima antena para a gestão de listas.

4.2. Operações sobre a Lista de Antenas

Foram implementadas as seguintes operações principais:

Inserção ordenada (`inserirAntena`) – garante que as antenas são armazenadas em ordem crescente de coordenadas (primeiro X, depois Y). A função verifica duplicações de posição.

Remoção (`removerAntena`) – elimina antenas com base em coordenadas. A função retorna uma nova versão da lista sem a antena eliminada.

Leitura de ficheiros (`carregarAntenas`) – converte uma matriz textual em lista ligada, identificando cada antena válida.

Libertação de memória (`libertarAntenas`) – elimina todos os nós da lista após a conclusão do programa.

4.3. Detecção de Locais com Efeito Nefasto

A análise de locais com efeito nefasto foi baseada na verificação de alinhamento e proporcionalidade entre pares de antenas com a mesma frequência. Um ponto é considerado nefasto se:

- Encontra-se exatamente entre duas antenas (ponto médio);
- As antenas estão alinhadas (em linha reta nas direções ortogonal ou diagonal);
- A distância entre uma e o ponto nefasto é igual à distância entre este ponto e a segunda antena.

O algoritmo percorre todos os pares possíveis da lista e calcula o ponto médio sempre que os critérios de alinhamento e distância são satisfeitos.

4.4. Modelação do Grafo

Com o objetivo de analisar a conectividade entre antenas, foi criada uma estrutura de grafo. Cada antena torna-se um vértice (Vertice) e cada ligação possível, baseada em igualdade de frequência, é representada como uma aresta (Aresta).

- A estrutura do grafo consiste numa lista ligada de vértices, onde cada vértice contém:
- Dados da antena (frequência, X, Y);

Lista ligada de arestas com os apontadores para outros vértices conectados.

Esta estrutura permite adicionar novas antenas e estabelecer ligações dinâmicas sem restrições de tamanho.

A figura 2 mostra como as antenas da mesma frequência interligam-se no grafo construído.

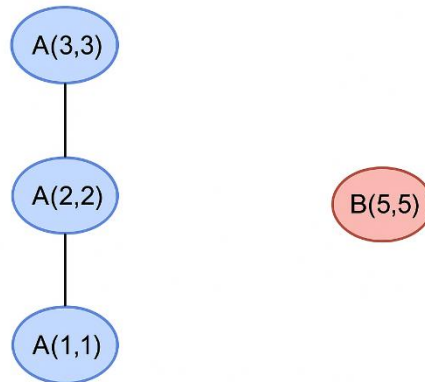


Figura 2 - Grafo dinâmico com ligações por frequência

4.5. Algoritmos de Procura

Foram implementados dois algoritmos fundamentais para explorar o grafo:

Procura em profundidade (*Depth-First Search*) – usada para explorar conexões profundas entre antenas, útil para confirmar ligação total.

Procura em largura (*Breadth-First Search*) – utilizada para identificar caminhos mínimos em número de ligações. Ambos os algoritmos operam sobre listas ligadas.

Estes algoritmos seguem os princípios clássicos descritos em (Wayne, 2011)

A identificação de vértices visitados é feita por listas auxiliares de coordenadas, garantindo que não se repetem visitas.

4.6. Identificação de Caminhos e Interseções

Com o grafo completo, implementaram-se duas funcionalidades avançadas:

- Todos os caminhos entre duas antenas – através de recursividade, são percorridos todos os ramos entre dois vértices dados, armazenando as sequências encontradas.
- Interseções entre frequências distintas – pares de antenas com frequências diferentes são agrupados para análise espacial. Os seus pares de coordenadas são listados como potenciais zonas de sobreposição.

A figura 3 representa as duas funções utilizadas para tal, gerada por meio da Inteligência Artificial MyMap.

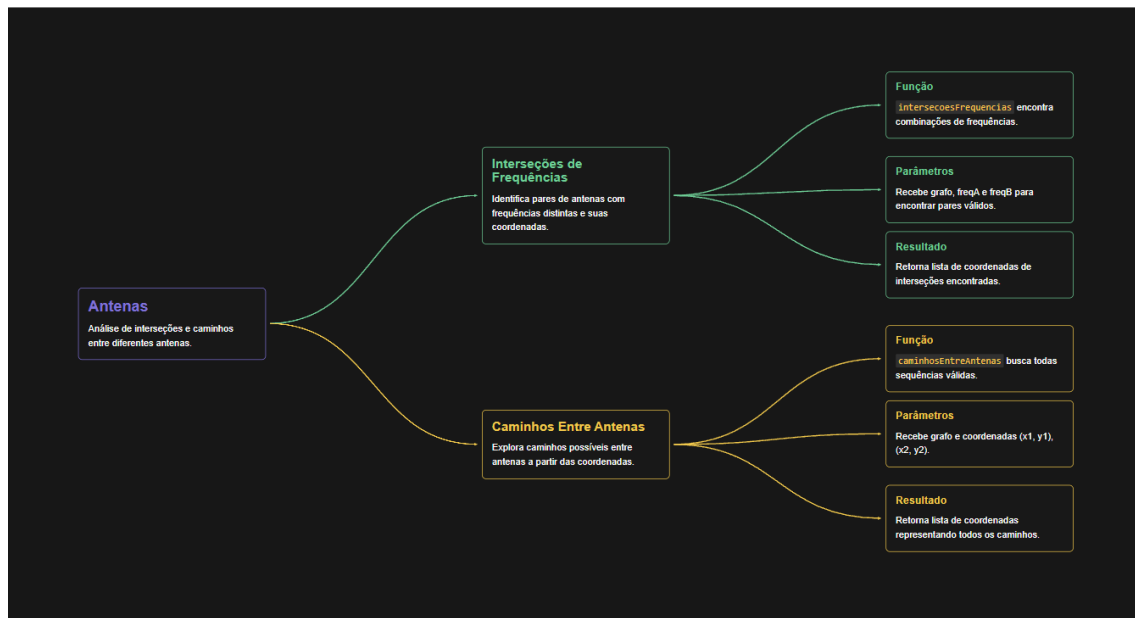


Figura 3 - Funções `intersecoesFrequencias` e `caminhosEntreAntenas`

5. Implementação e Testes

Este capítulo descreve o processo de implementação da solução proposta, desde a organização modular do código até à validação funcional de cada componente. A secção inclui ainda os testes realizados para garantir o correto funcionamento das operações essenciais, nomeadamente a leitura de dados, manipulação das listas e grafos, procura de caminhos, e deteção de locais com efeito nefasto.

5.1. Organização Modular

O projeto foi dividido em sete ficheiros principais, cada um com uma responsabilidade bem definida.

| Ficheiro | Descrição |
|--------------|--------------------------------------------------------------------------|
| Main.c | Função principal e execução de testes |
| Antenas.c/h | Operações sobre a lista ligada de antenas |
| Funcoes.c/h | Carregamento de antenas, remoções, inserções, memória e validações |
| Grafo.c/h | Estrutura de grafo, ligações, procura com profundidade/largura, caminhos |
| Nefastos.c/h | Lógica de deteção de locais com efeito nefasto |

Figura 4 - Tabela com cada objetivo por ficheiro)

5.2. Casos de Teste

A validação foi dividida por blocos funcionais, tendo sido utilizados ficheiros de entrada com diferentes distribuições de antenas.

5.2.1. Leitura e Construção da Lista de Antenas

- **Input** - uploadantenas.txt;
- **Objetivo** – Garantir a conversão correta de uma matriz textual para lista ligada ordenada;
- **Resultado esperado** – Lista com todos os nós inseridos corretamente, sem duplicações.

5.3.2 - Remoção de Antenas

- **Caso 1** - Remoção de antena existente
- **Caso 2** - Tentativa de remoção de antena inexistente
- **Validação** - Impressão de mensagem de confirmação e atualização da lista

5.3.3 - Detecção de Locais Nefastos

- **Cenário** - Antenas alinhadas e equidistantes com mesma frequência
- **Resultado esperado** - Lista de coordenadas nefastas gerada dinamicamente

5.3.4 - Estrutura de Grafo e Ligações

- **Objetivo** - Verificar se apenas antenas com mesma frequência são ligadas
- **Figura 2** como suporte ilustrativo
- **Validação** - Impressão das conexões estabelecidas

5.3.5 - Algoritmos de Procura e Caminhos

- **Profundidade** - Teste com ponto inicial e verificação dos vértices visitados
- **Largura** - Teste com ponto inicial para caminhos mínimos
- **Caminhos entre antenas** - Validação com origem e destino fixos

5.3.6 - Interseções entre Frequências

Frequências - A e B

Resultado esperado - Lista de pares de coordenadas encontradas

5.4. Resultados Obtidos

Todos os módulos funcionaram conforme esperado. A implementação respeitou os critérios de desempenho e modularidade. Os algoritmos de procura foram eficientes para o volume de dados testado, e a detecção de locais nefastos respondeu corretamente aos critérios geométricos estabelecidos.

5.5 Desafios Encontrados

A maior dificuldade sentida ao longo do desenvolvimento deste projeto foi certamente a interpretação do enunciado, especialmente na parte relativa à identificação das posições nefastas causadas pelas antenas. A definição das regras para determinar estas posições exigiu uma análise detalhada e cuidadosa para garantir a correta implementação. Além disso, o planeamento do trabalho revelou-se um desafio, uma vez que foi necessário estruturar a

abordagem de forma eficiente. No entanto, com uma análise mais aprofundada e a divisão do problema em partes mais simples, foi possível superar estas dificuldades e desenvolver uma solução funcional.

6. Conclusão

Este capítulo apresenta uma reflexão final sobre o trabalho desenvolvido, avaliando os resultados alcançados face aos objetivos inicialmente propostos. São revistos os principais contributos do projeto no contexto da unidade curricular de Estruturas de Dados Avançadas, tanto ao nível técnico como formativo.

O projeto permitiu aplicar, de forma prática, os conceitos de estruturas dinâmicas estudados ao longo do semestre, destacando-se a utilização exclusiva de listas ligadas simples como base para a representação de antenas, locais nefastos e caminhos.

A modelação do problema urbano através de um grafo dinâmico, com vértices representando antenas e arestas condicionadas à igualdade de frequência, demonstrou a viabilidade de estruturar uma rede de interações complexas de forma eficiente. A implementação de algoritmos clássicos de procura em profundidade e largura confirmou a aplicabilidade dos mesmos em cenários reais de análise de conectividade e propagação de sinal.

A funcionalidade de detecção de locais com efeito nefasto trouxe uma componente analítica adicional ao projeto, permitindo não só representar a infraestrutura, mas também antecipar os impactos gerados por certas configurações. A sua integração como parte do sistema reforçou a lógica de construção incremental da solução, centrada na modularidade e clareza funcional.

Conclui-se que o projeto atingiu com sucesso os objetivos propostos, oferecendo uma solução funcional, extensível e tecnicamente sólida para a representação e análise de antenas numa malha urbana. Além disso, reforçou competências cruciais como abstração algorítmica, gestão de memória dinâmica, e organização modular de sistemas em linguagem C.

7. Glossário e Siglas

Este capítulo apresenta os principais termos técnicos e siglas utilizados ao longo do documento, com o objetivo de clarificar a terminologia empregue e garantir consistência na interpretação dos conceitos abordados.

| Termo/Sigla | Definição |
|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Aresta | Estrutura que representa uma ligação entre dois vértices num grafo. Contém um apontador para o vértice de destino e outro para a próxima aresta na lista. |
| Estrutura Dinâmica | Tipos de estrutura de dados cuja memória é alocada em tempo de execução, como listas ligadas. |
| Ficheiro de Entrada | Documento de texto contendo uma matriz de caracteres com a disposição das antenas, utilizado para gerar a estrutura de dados inicial. |
| Grafo | Conjunto de vértices e arestas que representam as ligações entre antenas com a mesma frequência. Implementado com listas ligadas neste projeto. |
| Lista Ligada | Estrutura de dados linear onde cada elemento (nó) aponta para o seguinte. Usada para armazenar antenas, arestas, coordenadas, etc. |
| Local Nefasto | Ponto do mapa urbano identificado como sujeito a interferência, por estar exatamente entre duas antenas da mesma frequência e alinhadas. |
| Modularidade | Princípio de desenvolvimento que divide o sistema em componentes independentes, facilitando a organização, manutenção e reutilização do código. |
| Ponto Médio | Coordenada central entre duas outras coordenadas, calculada quando as diferenças entre os valores são pares. Utilizada para identificar locais nefastos. |
| Procura em Largura | Algoritmo de busca de grafos que explora os vértices por níveis. |
| Procura em Profundidade | Algoritmo de busca de grafos que explora os vértices em profundidade antes de recuar. |
| Vértice | Estrutura que representa uma antena no contexto de um grafo, contendo os dados da antena e as ligações para outros vértices com a mesma frequência. |

Figura 5 - Tabela Glossário

8. Bibliografia

Cormen, T. H. (2009). Introduction to Algorithms. Em T. H. Cormen, *Introduction to Algorithms* (p. 1313).

Rocha, A. M. (2020). Estruturas de Dados e Algoritmos em C.

Wayne, S. &. (2011). Algorithms.

9. Tabela de Figuras

| | |
|------------------------------------------------------------------------|----|
| Figura 1 - Representação da lista ligada de antenas | 13 |
| Figura 2 - Grafo dinâmico com ligações por frequência..... | 15 |
| Figura 3 - Funções intersecoesFrequencias e caminhosEntreAntenas | 16 |
| Figura 4 - Tabela com cada objetivo por ficheiro)..... | 17 |
| Figura 5 - Tabela Glossário | 21 |

10. Repositório no Github

<https://github.com/notunked/Projeto-EDA-ThiagoAbreu>