

UNIVERSIDADE FEDERAL DO PARANÁ

JOÃO MARCELO CASAGRANDE ADIERS

CÁLCULO DE POSICIONAMENTO PARA ACCESS POINTS EM INSTALAÇÕES
DE CABEAMENTO ESTRUTURADO

CURITIBA – PR

2023

UNIVERSIDADE FEDERAL DO PARANÁ

JOÃO MARCELO CASAGRANDE ADIERS

CÁLCULO DE POSICIONAMENTO PARA ACCESS POINTS EM INSTALAÇÕES
DE CABEAMENTO ESTRUTURADO

Relatório final apresentado para aprovação na disciplina TE 348 - Trabalho de Conclusão de Curso II do Departamento de Engenharia Elétrica, Setor de Tecnologia da Universidade Federal do Paraná. Orientador: Prof. Dr. Carlos Marcelo Pedroso.

CURITIBA – PR

2023

Resumo

Este trabalho apresenta uma solução do problema de determinar o posicionamento ótimo de estações rádio base (Access Points, AP) para redes Wi-Fi em ambientes de instalações de cabeamento estruturado. O problema envolve estabelecer o número mínimo de APs para garantir uma potência mínima recebida considerando a perda de percurso. Foi proposta uma solução ótima utilizando a técnica da programação inteira, com elaboração de um programa em linguagem Python, que permite a configuração de parâmetros do ambiente e dos equipamentos para aplicação do cálculo para diferentes cenários de localização dos APs ao atender aos requisitos das restrições estabelecidas.

Palavras-chave: Telecomunicações, Wi-Fi, Transmissão, Python.

Abstract

This monograph presents a solution to determining the optimum positioning of Access Points (APs) for *Wireless Fidelity (Wi-Fi)* networks in structured cabling installations. The problem involves establishing the minimum number of APs to guarantee minimum received power, taking into account path loss. An optimal solution was proposed using the integer programming technique, with the development of a program in the Python language that allows the configuration of scenario and parameters to meet the requirements of the established restrictions.

Keywords: Telecommunications, Wi-Fi, Transmission, Python.

LISTA DE FIGURAS

- Figura 1 – Faixas de frequência definidas na Resolução 305 da ANATEL.
- Figura 2 – Canais em rede 2,4 GHz.
- Figura 3 – Canais em rede 5 GHz.
- Figura 4 – Bairros da Cidade Exemplo.
- Figura 5 – Representação da área total separada em quadrados de mesma área.
- Figura 6 – Tabela de presença de APs.
- Figura 7 – Exemplo da potência de sinal
- Figura 8 – Parâmetros de cálculo NLos_2m
- Figura 9 – Valores de resposta NLos_2m
- Figura 10 – Parâmetros NLos_700m
- Figura 11 – Valores NLos_700m
- Figura 12 – Tela inicial da interface
- Figura 13 - Ajuste de parâmetros variáveis
- Figura 14 – Tela da interface do método Free-Space
- Figura 15 – Tela da interface do método Site-General
- Figura 16 – Matriz de seleção de área de interesse

LISTA DE TABELAS

- Tabela 1 – Computador Utilizado.
- Tabela 2 – Coeficientes do modelo Site-General

LISTA DE SIGLAS E SÍMBOLOS

- *Wi-Fi* *Wireless Fidelity*, em português, Fidelidade sem fio.
- *AP(s)* *Access Point(s)*.
- *NBR* Norma Brasileira.
- *ANATEL* Agência Nacional de Telecomunicações.
- *IEEE* Instituto de Engenheiros Eletricistas e Eletrônicos.
- *DFS* *Dynamic Frequency Selection*.
- *TPC* *Transmit Power Control*.
- *LOS* *Line of Sight*, em português, Linha de visão, ou visada direta.

SUMÁRIO

1.	INTRODUÇÃO	7
2.	OBJETIVOS E METODOLOGIA	8
2.1.	OBJETIVO GERAL	8
2.2.	OBJETIVOS ESPECÍFICOS	8
2.3.	MATERIAIS E MÉTODOS.....	8
2.3.1.	MATERIAIS	8
3.	FUNDAMENTAÇÃO TEÓRICA.....	11
3.1.	INTRODUÇÃO AO <i>WI-FI</i>	11
3.2.	DESAFIOS DO SINAL DE <i>WI-FI</i> EM AMBIENTES DIVERSOS	13
3.3.	MODELOS DE DESVANECIMENTO	14
3.4.	IMPORTÂNCIA DO POSICIONAMENTO ESTRATÉGICO DE ACCESS POINTS.....	16
3.5.	OTIMIZAÇÃO PARA POSICIONAMENTO DE ACCESS POINTS.....	16
4.	MÉTODO PROPOSTO	1
4.1.	MÉTODO PROPOSTO	4
5.	RESULTADOS.....	21
5.1.	CÁLCULO DE POSICIONAMENTO FREE-SPACE	21
5.2.	CÁLCULO DE POSICIONAMENTO SITE-GENERAL	21
5.3.	INTERFACE	21
6.	CONCLUSÃO	25
7.	BIBLIOGRAFIA	27
8.	ANEXOS.....	29

1. INTRODUÇÃO

A evolução constante da tecnologia e o aumento exponencial no uso de dispositivos móveis e sem fio impulsionam a demanda por redes *Wi-Fi*, especificadas pelo padrão IEEE 802.11 (referência). Em ambientes empresariais, acadêmicos e residenciais, onde a conectividade é um pilar essencial para a produtividade e eficiência, o posicionamento estratégico de *Access Points* (APs) é vital para assegurar uma cobertura de rede abrangente e confiável.

A internet, um recurso indispensável na sociedade contemporânea, desempenha um papel crucial como meio de comunicação, ferramenta de acesso à informação e facilitadora de uma variedade de atividades e compromissos síncronos e assíncronos. O *Wi-Fi*, sendo a tecnologia mais comum para prover acesso à internet, enfrenta desafios significativos relacionados ao desvanecimento de sinal. Este fenômeno é influenciado por vários fatores, incluindo a distância entre o transmissor e o receptor, a frequência de transmissão e a presença de obstáculos físicos que interferem na propagação do sinal.

A complexidade da previsão do número ideal de APs e a determinação de suas localizações ótimas são ampliadas pela variação das configurações de ambientes de cabeamento estruturado, que devem seguir a norma NBR 14565 (Associação Brasileira de Normas Técnicas, 2020). Cada instalação possui características únicas que impactam a propagação do sinal, como a disposição física, materiais de construção e outros elementos ambientais. Este projeto propõe um modelo que utiliza a programação inteira, e modelos de desvanecimento de sinal para maximizar a eficiência da cobertura do sinal com o menor número possível de APs, considerando uma potência mínima recebida a ser configurada pelo usuário.

O dimensionamento do ambiente será definido por um *input* externo de dados utilizando um ambiente genérico em formato de *grid*, com espaços quadrados e permitindo que seja alterado selecionando os quadrados que definem o ambiente desejado. A aplicação de programação inteira permitirá a avaliação eficiente de múltiplas configurações de APs, otimizando a disposição dos mesmos para alcançar o equilíbrio ideal entre cobertura e custo. O resultado foi a implementação de uma ferramenta valiosa para o planejamento e implementação de redes *Wi-Fi* em variadas configurações de cabeamento estruturado.

2. OBJETIVOS E METODOLOGIA

2.1. OBJETIVO GERAL

O objetivo principal é desenvolver uma metodologia para otimizar o posicionamento de APs em redes *Wi-Fi* e minimizando o número de APs necessários, de acordo com uma potência mínima recebida.

2.2. OBJETIVOS ESPECÍFICOS

- Apresentar os principais modelos de desvanecimento de sinal aplicáveis em prédios comerciais no Brasil;
- Definir as variáveis adicionais existentes no cálculo da perda de sinal e sua influência;
- Aplicar um modelo de otimização, utilizando programação inteira, ao modelo de desvanecimento mais adequado ao problema proposto;

2.3. MATERIAIS E MÉTODOS

Para a realização do trabalho foram necessários materiais e aplicação de métodos descritos a seguir:

2.3.1. MATERIAIS

Para o desenvolvimento do cálculo para o posicionamento de Access Points considerando ambientes de cabeamento estruturado foram utilizados os seguintes materiais:

- **Recursos Computacionais:** Foi utilizado o notebook descrito na Tabela 1 para desenvolvimento dos códigos de programação, bem como a realização das pesquisas necessárias ao trabalho.

	Notebook
Modelo	ASUS X555LF
Processador	Intel Core i5-5200U de 2.2GHz
Memória RAM	8 GB
Sistema Operacional	Windows 11

Tabela 1 – Computador utilizado.

- **Linguagem Python:** Python é uma linguagem de programação de alto nível de propósito geral, sendo usada para uma grande variedade de aplicações, com sua primeira versão lançada em 1991. Essa linguagem é muito utilizada pela facilidade de aprendizado e o grande número de bibliotecas para os mais variados tipos de aplicação. A versão da linguagem utilizada foi a Python 3.11.7 (Python Software Foundation, 2023).
- **Biblioteca PuLP:** Biblioteca de Python para modelagem e resolução de problemas de programação linear e programação inteira. Permite definir problemas de otimização dos principais algoritmos conhecidos e resolver computacionalmente esses problemas com o uso. A versão da biblioteca utilizada foi a PuLP 2.8.0 (Mitchell et al., 2024).
- **Framework Streamlit:** Conjunto de bibliotecas de programação para linguagem Python que possibilitam uma fácil implementação de um dashboard web, simplificando o processo de criação e edição de uma interface gráfica implementada em Python.

2.3.2. MÉTODOS

- **Pesquisa sobre Programação Inteira:** Programação inteira é uma derivação de programação linear. Em sua aplicação são usadas uma série de técnicas de otimização para resolução de problemas de programação linear, buscando obter soluções ótimas com o menor gasto de recursos.
Devido ao número de operações necessárias para chegar na solução ótima crescer exponencialmente conforme o número de inequações cresce, a resolução do problema por esforço humano se torna inviável.

- **Pesquisa sobre Path Loss:** A transmissão do sinal passa por perdas para chegar de um ponto a outro. Para verificar o sinal devido a todos os detalhes do ambiente existem modelos de propagação de sinal que foram desenvolvidos e validados. Esses modelos proporcionam um meio de expressar matematicamente a previsão do sinal durante a propagação, tendo os métodos de cálculo de perdas da potência do sinal. A International Telecommunication Union (ITU) publicou recomendações para diversas situações. A ITU-R P.1238-12 (ITU, 2023) apresenta modelos de cálculo de perda de caminho para ambientes indoor que abrangem o caso da rede *Wi-Fi*.
- **Pesquisa sobre *Wi-Fi*:** *Wi-Fi* é uma tecnologia muito utilizada para a transmissão de dados em diversos ambientes. É um tipo de tecnologia de camada 1 e 2 que ainda segue se renovando, conforme novas tecnologias surgem, como a *Wi-Fi 6* ou 802.11ax, a qual já está em funcionamento.
- **Equacionamento do problema:** Para poder ter uma escalabilidade da resolução do problema, foi formulado o equacionamento do mesmo tomando como base um problema do tipo *Set Covering* de minimização.
- **Área de Cobertura:** A implementação foi feita utilizando como base uma área genérica, de formato retangular, separada como uma matriz, tendo o cálculo do sinal recebido em cada porção da área como sendo uma variável binária representando a presença ou ausência da potência mínima no local. A solução de um problema se torna viável se cumprir todas as restrições e se torna ótima se for a melhor dentre as viáveis.
- **Testes:** Foram realizados testes com cenários onde o resultado ótimo é conhecido para validar o método proposto. Para isso, foi usado o link budget calculator do site 5G Tools (VINOGRADOV, 2024), que calcula a perda de sinal com diversos modelos, permitindo a criação de cenários de teste.

3. FUNDAMENTAÇÃO TEÓRICA

3.1. INTRODUÇÃO AO *Wi-Fi*

A administração do espectro de radiofrequência no Brasil, um recurso público de grande valor, é realizada pela Agência Nacional de Telecomunicações (ANATEL). Esta agência é responsável pela regulamentação e supervisão das atividades de telecomunicações no país. Conforme definido pela Resolução 506/08 da ANATEL (referência), o espectro em questão abrange frequências de 8,3 kHz a 3000 GHz, e inclui faixas não licenciadas que são utilizadas para a transmissão de *Wi-Fi*. Conforme pode ser observado na Figura 1 (TELECO, 2023), as frequências relevantes, que não exigem o processo de licenciamento são, na faixa de 2,4 GHz, de 2400 a 2483,5 MHz e na faixa de 5 GHz, de 5150 a 5350 MHz e de 5470 a 5850 MHz. Essas faixas não licenciadas são utilizadas tanto por conta do preço e complexidade do processo e também das exigências que são necessárias para o licenciamento junto a ANATEL.

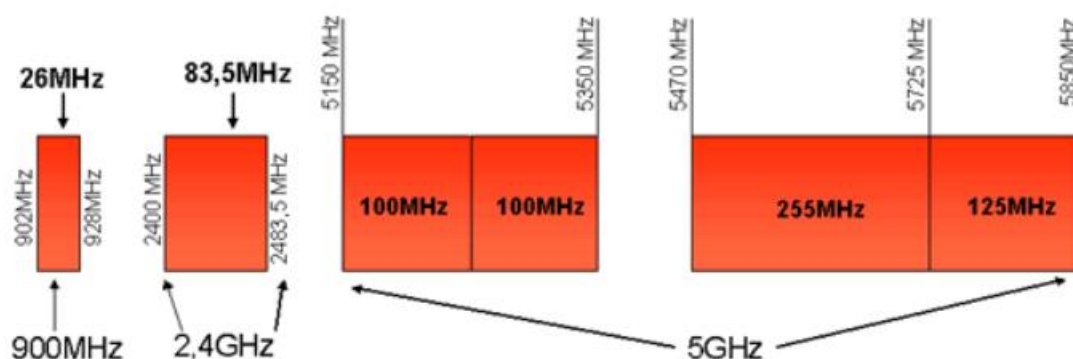


Figura 1 - Faixas de frequência definidas na Resolução 305 da ANATEL - Fonte: (TELECO, 2023)

Os chamados padrões IEEE 802.11 para redes *Wi-Fi*, desenvolvidos pelo Instituto de Engenheiros Eletricistas e Eletrônicos (IEEE), representam um conjunto de normas que definem as características das redes de comunicação sem fio. Cada padrão, variando em frequência e capacidade, contribui de maneira única para a tecnologia *Wi-Fi*. Por exemplo, o 802.11b (*Wi-Fi* 2) e o 802.11g (*Wi-Fi* 3) operam na faixa *Industrial, Scientific and Medical* (ISM) de 2,4 GHz, enquanto o 802.11n (*Wi-Fi* 4) opera nas faixas ISM de 2,4 GHz e 5 GHz. O 802.11ac (*Wi-Fi* 5) é restrito à faixa ISM de 5 GHz.

- 802.11 – *Wi-Fi 1* – ISM 2,4 GHz;
- 802.11b – *Wi-Fi 2* – ISM 2,4 GHz;
- 802.11a – *Wi-Fi 3* – ISM 5 GHz;
- 802.11g – *Wi-Fi 3* – ISM 2,4 GHz;
- 802.11n – *Wi-Fi 4* – ISM 2,4 GHz e ISM 5 GHz;
- 802.11ac – *Wi-Fi 5* – ISM 5 GHz;

As faixas de frequência ISM de 2,4 GHz e 5 GHz foram definidas pela ANATEL e são amplamente utilizadas para a internet *Wi-Fi*. A faixa de 2,4 GHz é dividida em 13 canais com largura de banda de 20 MHz cada, tendo sobreposição parcial entre os canais, ou permitindo o uso simultâneo dos canais 1, 6 e 11 sem interferência, conforme pode ser observado na Figura 2 (EMBARCADOS, 2023).

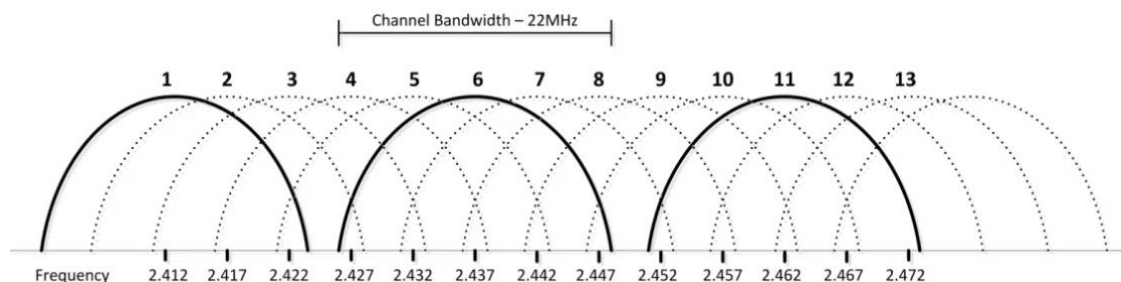


Figura 2 - Canais em rede 2,4 GHz. - Fonte: (EMBARCADOS, 2023)

Na faixa de 5 GHz, por outro lado, os canais têm diferentes possíveis larguras de banda (20, 40, 80 ou 160 MHz) e oferecem a possibilidade de maior número de canais não sobrepostos, conforme pode ser observado na Figura 3 (HUAWEI, 2023), sendo 25 canais de 20 MHz, 12 canais de 40 MHz, 6 canais de 80 MHz e 2 canais de 100 MHz. O aumento na largura de banda de cada canal pode oferecer uma possibilidade de maior velocidade de transmissão de dados, mas em compensação, perde parte do seu alcance, diminui o número de canais não sobrepostos e aumenta a possibilidade de interferência no sinal.

Existem recursos para evitar interferências, principalmente com sistemas de radar, como por exemplo DFS (*Dynamic Frequency Selection*) e TPC (*Transmit Power Control*).

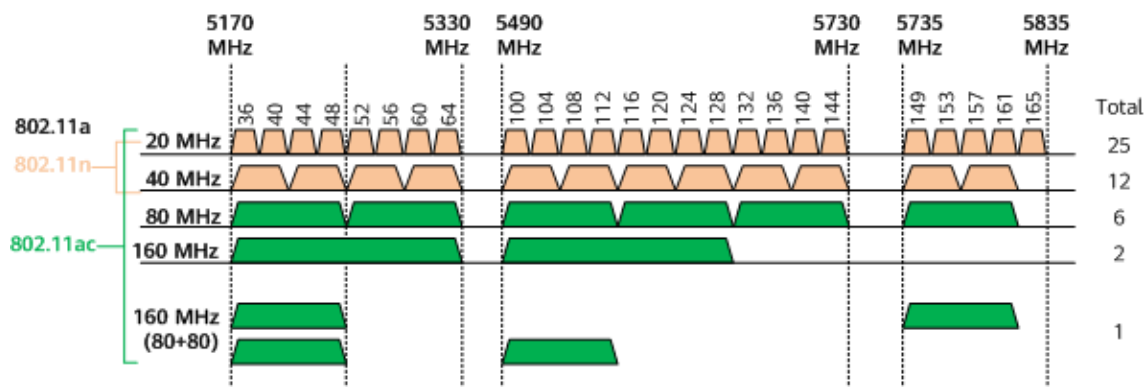


Figura 3 - Canais em rede 5 GHz - Fonte: (HUAWEI, 2023)

Quanto aos modos de rede *Wi-Fi*, eles podem ser configurados tanto em modo *Ad-hoc*, no qual as redes funcionam sem pontos de acesso intermediando as conexões, quanto em modo de Infraestrutura, que requer o uso de APs para intermediar a conexão, atuando como interfaces entre redes sem fio e redes com fio, podendo ter APs adicionais operando como repetidor, para ampliar a cobertura da rede.

Esta revisão destaca a importância de compreender as faixas de frequência e os padrões IEEE 802.11, bem como as regulamentações da ANATEL, no planejamento eficiente de redes *Wi-Fi*. O foco está na seleção apropriada de faixas de frequência, padrões e modos de operação para otimizar a cobertura e o desempenho das redes sem fio.

3.2. DESAFIOS DO SINAL DE *WI-FI* EM AMBIENTES DIVERSOS

A propagação de sinais por ondas de radiofrequência na prática não é ideal, existindo perdas na transmissão de sinais. Para transmissões do tipo *Line-of-Sight* (Los), que é quando não há obstáculos na transmissão, podendo ter uma linha reta sem nenhuma interferência entre o emissor e o receptor do sinal, tendo assim uma linha de visão ou visada direta. Nesse tipo de transmissão as perdas são provenientes principalmente da distância de transmissão do sinal e a frequência do sinal.

Para transmissões do tipo *Non-Line Of Sight* (NLos), que é quando há obstáculos entre o emissor e o receptor do sinal, sem possibilidade de ter uma linha reta livre para propagação do sinal sem interferências, sendo consideradas sem linha de visão ou visada direta, as perdas podem ser muito maiores, devido a obstruções no trajeto, como paredes, objetos, móveis e pessoas, que podem acabar por causar

o chamado *Multipath*, que é quando o sinal é transmitido sem visada direta, sendo difratado e chegando ao receptor por caminhos diferentes, criando interferências destrutivas no sinal ao chegar fora de fase. Devido às interferências, é necessário aumentar a potência do sinal para compensar as perdas. Essa compensação é nomeada Margem de desvanecimento e é calculada pela diferença entre o sinal recebido e a sensibilidade do receptor do sinal (Margem de desvanecimento = P_{rx} – Sensibilidade do Receptor, onde P_{rx} = potência recebida).

Para estimar todos os fatores que afetam a propagação do sinal, é necessário calcular o *Link Budget* (i.e. Orçamento de Link) que é o cálculo dos fatores que dão ganho e dos fatores que causam perdas ao sinal, dado por:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - L_{tx} - L_{path} - L_m - L_{rx} \quad (1)$$

onde P_{rx} é a potência do sinal recebido, P_{tx} é a potência do transmissor, G_{tx} o ganho da antena do transmissor, G_{rx} é o ganho da antena do receptor, L_{tx} é a perda do transmissor, L_{path} a perda no caminho determinada pela equação da perda na transmissão do modelo de perdas escolhido, L_m são as perdas diversas (margem de desvanecimento, perdas nos cabos (*body loss*), perdas por polarização das antenas, entre outras) e L_{rx} a perda no receptor;

3.3. MODELOS DE DESVANECIMENTO

O desvanecimento de sinal devido à transmissão pode ser estimado pelos modelos já criados e validados existentes. Existem modelos para diferentes ambientes e aplicações. A seguir serão descritos os principais modelos.

- Modelo de Espaço Livre ou *Free-space*:

É o modelo que é mais próximo de uma condição ideal para propagação, chamado de Fórmula de Friis, sendo consideradas apenas perdas pela distância e comprimento de onda, considerando que há uma transmissão por um meio sem obstáculos, definido por:

$$P_{rx} = P_{tx} \cdot G_{tx} \cdot G_{rx} \cdot \left(\frac{\lambda}{4\pi d} \right)^2 \quad (2)$$

onde P_{rx} é a potência recebida, P_{tx} a potência transmitida, G_{tx} o ganho da antena do transmissor, G_{rx} o ganho da antena do receptor, λ o comprimento de onda do sinal transmitido, e d a distância tridimensional entre as antenas;

Para poder ser utilizada a distância entre as antenas e a frequência apenas, será utilizada a equação (3) conforme apresentada na recomendação ITU-R P.525-4 (ITU, 2019), a qual traz recomendações para o cálculo de atenuação em transmissões do tipo *free-space*.

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - (32,4 + 20 \cdot \log_{10}(f) + 20 \cdot \log_{10}(d)) \quad (3)$$

onde P_{rx} , P_{tx} , G_{tx} , G_{rx} e d são os mesmos parâmetros utilizados em (2) e f é a frequência da transmissão.

- **Modelo *Site-General***

São modelos utilizados para perda de percurso sem medidas no local desejado, sem especificações mais profundas do ambiente e são mais práticos para serem utilizados em muitos cenários diferentes sem ajustes no modelo. O modelo de *Site-General* sugerido pela ITU-R P.1238-12 (ITU, 2023) para casos em que o receptor e o emissor do sinal estão no mesmo andar, dado por

$$L_b(d, f) = 10\alpha \log_{10}(d) + \beta + 10\gamma \log_{10}(f) \quad (4)$$

onde $L_b(d, f)$ é a perda média na transmissão, d a distância tridimensional entre o emissor e o receptor, f é a frequência da transmissão, α o coeficiente associado ao aumento na perda de transmissão devido a distância, β é o coeficiente associado ao valor de offset na perda de transmissão e γ é o coeficiente associado ao aumento na perda de transmissão devido a frequência;

Para o método, as perdas máximas são dadas pela equação (5):

$$10 \log_{10}(10^{0,1A} + 1)(\text{dB}) \quad (5)$$

onde A é uma variável randômica de distribuição normal $N(\mu, \sigma)$ com média $\mu =$

$L_b(d, f) - L_{fs}$, $L_{fs} = 20 \log_{10}\left(\frac{4 \cdot 10^9 \pi d f}{c}\right)$ e c é a velocidade da luz; e os coeficientes α , β , γ e σ variam de acordo com o ambiente e o tipo de transmissão a ser avaliado, podendo ser com ou sem visada direta (Los ou NLos, respectivamente) conforme a tabela 2:

Ambiente	Los/NLos	Faixa de Frequência (GHz)	Faixa de Distância (m)	α	β	γ	σ
Escritório	Los	0,3—83,5	2—27	1,46	34,62	2,03	3,76
	NLos	0,3—82,0	4—30	2,46	29,53	2,38	5,04
Corredor	Los	0,3—83,5	2—160	1,63	28,12	2,25	4,07
	NLos	0,625—83,5	4—94	2,77	29,27	2,48	7,63
Industrial	Los	0,625—70,28	2—102	2,34	24,26	2,06	2,67
	NLos	0,625—70,28	5—110	3,66	22,42	1,34	9,00
Sala de conferências / auditório	Los	0,625—82,0	2—21	1,61	28,82	2,37	3,28
	NLos	7,075—70,28	4—25	2,07	28,13	2,67	3,67

Tabela 2 – Coeficientes do modelo Site-General

Serão implementados os modelos do tipo *Site-General* e *free-space*, recomendados pelo ITU. O Site-General por poder ser aplicado em diferentes situações com o mesmo método de cálculo, abrangendo com a implementação diferentes casos e o *Free-Space*, por ser um modelo muito simplificado de cálculo.

3.4. IMPORTÂNCIA DO POSICIONAMENTO ESTRATÉGICO DE ACCESS POINTS

Devido às perdas na transmissão do sinal, o alcance em que o sinal ainda é possível ser usado para transmissão de informações é limitado. Com essa limitação, é preciso determinar a localização dos APs que serão planejados para que o sinal ilumine com uma potência mínima todas as áreas do edifício.

3.5. OTIMIZAÇÃO PARA POSICIONAMENTO DE ACCESS POINTS

A programação linear é um método utilizado para maximizar ou minimizar o valor de uma equação linear, dado um conjunto de restrições. Exemplos dos componentes desse método são:

- Equação linear

$$f(x_1, x_2) = c_1x_1 + c_2x_2$$

- Restrições do problema

$$a_{11}x_1 + a_{12}x_2 \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 \leq b_2$$

$$a_{31}x_1 + a_{32}x_2 \leq b_3$$

- Variáveis não negativas e inteiras

$$x_1 \geq 0$$

$$x_2 \geq 0$$

onde c_1 e c_2 são constantes, x_1 e x_2 são as variáveis de interesse, a_{11} a a_{32} são as constantes de cada posição de uma matriz de elementos e b_1 a b_3 o objetivo de otimização das restrições.

Set Covering Problem (SCP) ou problema de área de cobertura é o nome dado para um tipo de problema da programação linear no qual as restrições representam as coberturas parciais do conjunto da equação, garantindo que as soluções sejam viáveis e atendam aos requisitos do problema as variáveis assumem os valores de 0 ou 1. O objetivo é minimizar ou maximizar o valor da equação, que pode representar uma área, um custo, recursos a serem alocados, com a inclusão ou exclusão das variáveis na equação.

O método Simplex (DANTZIG, George B., 1997) é utilizado para encontrar a solução ótima de problemas representados por Programação Linear. Para o método ser resolvido matematicamente, são transformadas as inequações definidas na representação por programação linear adicionando variáveis para ajustá-las para equações.

Após serem mudadas as expressões de inequações para equações, são realizadas operações para serem modificadas as equações e ser definida uma primeira solução, para, a partir dessa solução verificar se é a solução ótima, caso não seja, verificar se alguma chega a um valor mais extremo, substituindo na análise a melhor solução até chegar na solução ótima.

Utilizando o método simplex, são levadas em conta as restrições e utilizadas operações nas inequações para que o problema seja resolvido de maneira ótima. Considere o seguinte exemplo de problema de área de cobertura (TRICK, Michael A, 1997):

Uma cidade está revendo os locais dos seus quartéis de bombeiro. A cidade é constituída de alguns bairros, conforme ilustrado na figura 4:

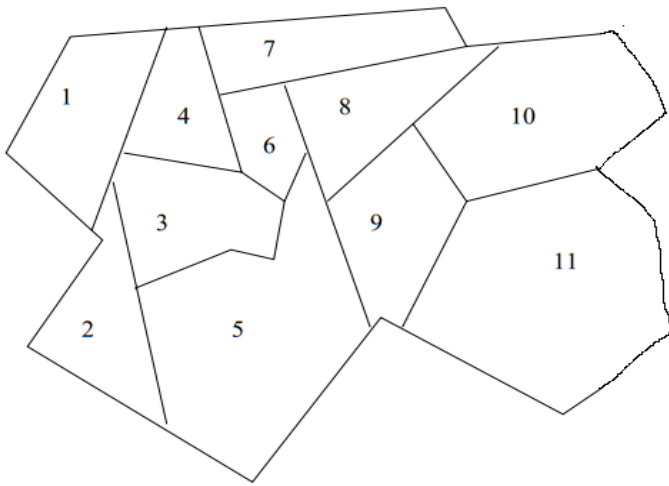


Figura 4 - Bairros da Cidade Exemplo - Fonte: (TRICK, Michael A, 1997)

Um quartel de bombeiros pode ser colocado em qualquer bairro. Ele é capaz de lidar com os incêndios do seu bairro e também dos bairros adjacentes (qualquer bairro com fronteira ao seu bairro de origem). O objetivo é minimizar o número de quartéis de bombeiro usados.

Considere criar uma variável x_j para cada bairro j . Essa variável terá valor 1 quando o bairro tiver um quartel no bairro, caso contrário, terá valor 0. Com isso obtem-se a formulação do problema:

$$\text{Minimize } x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11}$$

Sujeito à:

$$\begin{array}{rcl} x_1 + x_2 + x_3 + x_4 & & \geq 1 \\ x_1 + x_2 + x_3 + x_5 & & \geq 1 \\ x_1 + x_2 + x_3 + x_4 + x_5 + x_6 & & \geq 1 \\ x_1 + x_3 + x_4 + x_6 + x_7 & & \geq 1 \\ x_2 + x_3 + x_5 + x_6 + x_8 + x_9 & & \geq 1 \\ x_3 + x_4 + x_5 + x_6 + x_7 + x_8 & & \geq 1 \\ x_4 + x_6 + x_7 + x_8 & & \geq 1 \\ x_5 + x_6 + x_7 + x_8 + x_9 + x_{10} & & \geq 1 \\ x_5 + x_8 + x_9 + x_{10} + x_{11} & & \geq 1 \\ x_8 + x_9 + x_{10} + x_{11} & & \geq 1 \\ x_9 + x_{10} + x_{11} & & \geq 1 \end{array}$$

Para $x_j \in \{0,1\}$ $j = 1, \dots, 11$

A primeira restrição afirma que deve existir um quartel no bairro 1 ou nos bairros adjacentes ao 1. A próxima restrição é a respeito do bairro dois e assim por diante. Pode ser notado que os coeficientes a_{ij} das restrições é igual a 1 se o bairro i é adjacente ao bairro j ou se $i = j$ e 0 caso contrário. A coluna de número j da matriz de restrições apresenta o conjunto dos bairros atendidos caso seja colocado um quartel seja colocado no bairro j . Para resolver o problema, é necessário encontrar o conjunto de subconjuntos que possibilite cobrir todos os bairros de modo que cada bairro j esteja associado a pelo menos um quartel de bombeiros.

Para resolver o problema, é possível escrever um programa em linguagem Python utilizando a biblioteca PuLP (MITCHELL et al., 2024), conforme o trecho de código a seguir. No código é definido o tipo de problema a ser resolvido na linha 2, definindo como sendo um problema do tipo *set covering*, e que o objetivo é minimizar a expressão que determina se os bairros possuem um quartel de bombeiros ou não, são definidas as restrições de resolução nas linhas 5 a 17 e é chamado o solucionador na linha 20.

```
1  from pulp import *
2  problema = LpProblem("SetCoveringProblem", LpMinimize)
3  x = LpVariable.dicts("x", range(1, 12), cat=LpBinary)
4  problema += lpSum(x[i] for i in range(1, 12))
5  Constraints = [
6      [1, 2, 3, 4],
7      [1, 2, 3, 5],
8      [1, 2, 3, 4, 5, 6],
9      [1, 3, 4, 6, 7],
10     [2, 3, 5, 6, 8, 9],
11     [3, 4, 5, 6, 7, 8],
12     [4, 6, 7, 8],
13     [5, 6, 7, 8, 9, 10],
14     [5, 8, 9, 10, 11],
15     [8, 9, 10, 11],
```

```
16         [9, 10, 11]
17     ]
18     for i, c in enumerate(constraints, start=1):
19         problema += lpSum(x[j] for j in c) >= 1
20     problema.solve()
21     print("Status:", LpStatus[problema.status])
22     print("Valor_da_função_objetivo:", value
(problema.objective))
23     print("Valores das variáveis:")
24     for v in problema.variables():
25         print(v.name, "=", v.varValue
```

4. MÉTODO PROPOSTO

Para identificar o número mínimo e local ideal de APs para se obter um sinal mínimo ao longo da planta, será utilizada inicialmente uma matriz para a entrada de dados da área do local de interesse por seleção das frações de uma matriz dividida em quadrados de área fixa e pré-definida, conforme a Figura 5. Após a definição da área, será feito o cálculo das perdas de sinal, utilizando o modelo de propagação escolhido.

LINHAS	COLUNAS							
	1	2	3	4	5	6	...	m
1	y_{11}	y_{12}	y_{13}	y_{14}	y_{15}	y_{16}	...	y_{1m}
2	y_{21}	y_{22}	y_{23}	y_{24}	y_{25}	y_{26}	...	y_{2m}
3	y_{31}	y_{32}	y_{33}	y_{34}	y_{35}	y_{36}	...	y_{3m}
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\ddots	\vdots
n	y_{n1}	y_{n2}	y_{n3}	y_{n4}	y_{n5}	y_{n6}	...	y_{nm}

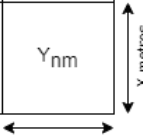


Figura 5 – Matriz de seleção de área de interesse

onde y_{11} a y_{nm} são as variáveis de validação da célula da matriz, assumindo valores 1 para a presença da fração de área na área de interesse e 0 para a ausência da fração de área na área de interesse, representada pela planta do local a ser avaliado o sinal.

Para ser obtido um valor mínimo de potência de sinal recebido em cada subárea, estabelecido por entrada de dados, será considerado a área selecionada anteriormente e calculada a influência do sinal para cada fração dessa área, com a existência ou não de um AP em cada fração da área, sujeito às restrições dadas pelo cálculo das inequações dadas pelas restrições.

		COLUNAS							
		1	2	3	4	5	6	...	m
LINHAS	1	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	...	x_{1m}
	2	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	...	x_{2m}
	3	x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	...	x_{3m}
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	↘	⋮
	n	x_{n1}	x_{n2}	x_{n3}	x_{n4}	x_{n5}	x_{n6}	...	x_{nm}
								↕ X metros	
									↔ X metros

Figura 6 - Tabela de presença de APs

onde x_{11} a x_{nm} indicam a presença ou não de um AP na célula da linha 1 e coluna 1 a linha n e coluna m e X indica um tamanho genérico do lado de cada célula em metros, indicando que elas são quadradas.

O problema será definido por:

Minimize:

$$x_{11} + x_{12} + x_{13} + \dots + x_{21} + x_{22} + x_{23} \dots + x_{nm} \quad (4)$$

onde x_{11} a x_{nm} são os mesmos utilizados na Figura 6.

Sujeito a:

$$y_{11} \cdot (y_{11} \cdot x_{11} \cdot P_{11} + y_{12} \cdot x_{12} \cdot P_{12} + y_{13} \cdot x_{13} \cdot P_{13} + \dots + y_{nm} \cdot x_{nm} \cdot P_{nm} \geq 1) \quad (5.1)$$

$$y_{12} \cdot (y_{11} \cdot x_{11} \cdot P_{11} + y_{12} \cdot x_{12} \cdot P_{12} + y_{13} \cdot x_{13} \cdot P_{13} + \dots + y_{nm} \cdot x_{nm} \cdot P_{nm} \geq 1) \quad (5.2)$$

$$y_{13} \cdot (x_{11} \cdot P_{11} + y_{12} \cdot x_{12} \cdot P_{12} + y_{13} \cdot x_{13} \cdot P_{13} + \dots + y_{nm} \cdot x_{nm} \cdot P_{nm} \geq 1) \quad (5.3)$$

...

$$y_{nm} \cdot (x_{11} \cdot P_{11} + y_{12} \cdot x_{12} \cdot P_{12} + y_{13} \cdot x_{13} \cdot P_{13} + \dots + y_{nm} \cdot x_{nm} \cdot P_{nm} \geq 1) \quad (5.n)$$

onde cada restrição apresentada nas equações 5.x representam a cobertura do sinal em cada célula da matriz indicada pela Figura 5. Por exemplo, a Equação (5.1) representa quais células são iluminadas com uma potência maior que a potência mínima definida, considerando o modelo de perda de caminho utilizada. Neste caso,

supondo uma matriz 3x3, com a colocação na célula (1,1), e uma potência de sinal ilustrada pela Figura 7. Desta forma o valor das variáveis P_{nm} serão 0 ou 1.

	1	2	3
1	20 dBm	-30 dBm	-78 dBm
2	-30 dBm	-40 dBm	-80 dBm
3	-78 dBm	-80 dBm	-100 dBm

Figura 7 - Exemplo da potência de sinal

Considerando que a potência limite configurada seja de -70dBm, a inequação (5.1) será escrita como $y_{11} \cdot x_{11} + y_{21} \cdot x_{21} + y_{12} \cdot x_{12} + y_{22} \cdot x_{22} \geq 1$

As demais restrições são obtidas alterando-se a posição do AP e repetindo-se o procedimento.

As variáveis y_{nm} foram inseridas para facilitar a implementação da interface gráfica. O valor possível dessas variáveis serão 0 ou 1, habilitando o posicionamento do AP em uma determinada posição da matriz de interesse.

O número de restrições é definido pelo número de posições habilitadas para posicionamento de APs, sendo que o número máximo de restrições é dado pela multiplicação do número total de linhas pelo número total de colunas da matriz, ou seja, para uma matriz de 3x3, o número máximo de restrições é 9, para uma matriz de 4x4 o número máximo é 16 e assim por diante. Para cada posição desabilitada, diminui-se uma variável e uma restrição, diminuindo a complexidade do problema.

Quanto maior o número de restrições, é exigido maior tempo para poder chegar à solução do problema, até o limite em que o recurso computacional não consiga dar uma solução sem precisar de um tempo muito grande, o que torna inviável a resolução do problema. Para mitigar esse problema, as células da matriz de interesse podem variar de tamanho (exemplo: 1m x 1m, 5m x 5m ou 10m x 10m por célula).

4.1.MÉTODO PROPOSTO

Para validação dos resultados foram utilizados os cálculos do site 5G tools, que implementa o cálculo de perda de sinal, com parâmetros ajustáveis de distância em números inteiros, apresentando um bom valor aproximado para cálculo de perdas para o modelo de perdas desejado. Além disso, foi feita validação manual com o cálculo das perdas pela definição matemática da perda para os modelos. Para a área na posição do AP, será considerado o sinal recebido como -30 dBm. Para o sinal fora da área da posição do AP, será considerado um bom sinal como sendo entre -30 a -70 dBm.

Na validação, foram consideradas algumas situações específicas para possibilitar a validação dos cálculos, nas quais foram considerados valores diferentes de distâncias entre as células, mas tendo um formato de área em comum, uma matriz quadrada, de 4 linhas e 4 colunas, com cada célula tendo uma área quadrada.

Cada distância entre o ponto em que fica o AP e a célula de interesse em saber o valor do sinal recebido é calculado com base na distância em linha reta entre esses pontos. Será considerado o valor de o cálculo das distâncias como um cálculo genérico para facilitar a sua representação, em que serão considerados padrões de distância a partir de uma célula específica, sendo eles:

Padrão 1 – Células adjacentes na vertical ou horizontal;

Padrão 2 – Duas células de distância na vertical ou horizontal;

Padrão 3 – Três células de distância na vertical ou horizontal;

Padrão 4 – Células adjacentes na diagonal;

Padrão 5 – Duas células de distância na vertical e uma na horizontal ou vice-versa;

Padrão 6 – Três células de distância na vertical e uma na horizontal ou vice-versa;

Padrão 7 – Duas células de distância na vertical e na horizontal;

Padrão 8 – Três células de distância na vertical e duas na horizontal ou vice-versa;

Padrão 9 – Três células de distância na vertical e na horizontal;

Local do AP	Padrão 1	Padrão 2	Padrão 3
Padrão 1	Padrão 4	Padrão 5	Padrão 6
Padrão 2	Padrão 5	Padrão 7	Padrão 8

Padrão 3	Padrão 6	Padrão 8	Padrão 9
----------	----------	----------	----------

Cálculo de perdas na transmissão:

Potência receptor

$$= P_{tx} + G_{tx} + G_{rx} - \text{Perdas percurso} - \text{Perdas transmissor} \\ - \text{Perdas receptor}$$

onde $P_{tx} = 20$, $G_{tx} = G_{rx} = \text{Perdas transmissor} = \text{Perdas receptor} = 1$

4.2. Situação 1 - Lado de cada célula de 2m

Distâncias:

0	2	4	6
2	2,828427125	4,472135955	6,32455532
4	4,472135955	5,656854249	7,211102551
6	6,32455532	7,211102551	8,485281374

Perda da transmissão tipo *Free-space*

5G Tools:

X	52,44	58,4606	61,9824
52,44	55,4503	59,4297	62,44
58,4606	59,4297	61,4709	63,5794
61,9824	62,44	63,5794	64,9927

Cálculo matemático das perdas, utilizando a perda da transmissão dada pela expressão:

$$32,44 + 20 \cdot \log_{10}(f) + 20 \cdot \log_{10}(d)$$

Padrões:

$$1 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2}{1000}\right) = 52,44$$

$$2 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{4}{1000}\right) = 58,46059991$$

$$3 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{6}{1000}\right) = 61,98242509$$

$$4 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2\sqrt{2}}{1000}\right) = 55,45029996$$

$$5 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2\sqrt{5}}{1000}\right) = 59,42970004$$

$$6 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2\sqrt{10}}{1000}\right) = 62,44$$

$$7 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2\sqrt{8}}{1000}\right) = 61,47089987$$

$$8 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2 \cdot \sqrt{13}}{1000}\right) = 63,57943352$$

$$9 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2 \cdot \sqrt{18}}{1000}\right) = 64,99272505$$

30	52,44	58,4606	61,9824
52,44	55,4503	59,4297	62,44
58,4606	59,4297	61,4709	63,5794
61,9824	62,44	63,5794	64,9927

Potência no receptor para cada padrão de distância:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - \text{Perdas percurso} - \text{Perdas transmissor} - \text{Perdas receptor}$$

$$1 - P_{rx} = 20 + 1 + 1 - 52,44 - 1 - 1 = -32,44$$

$$2 - P_{rx} = 20 + 1 + 1 - 58,4606 - 1 - 1 = -38,4606$$

$$3 - P_{rx} = 20 + 1 + 1 - 61,9824 - 1 - 1 = -41,9824$$

$$4 - P_{rx} = 20 + 1 + 1 - 55,4503 - 1 - 1 = -35,4503$$

$$5 - P_{rx} = 20 + 1 + 1 - 59,4297 - 1 - 1 = -39,4297$$

$$6 - P_{rx} = 20 + 1 + 1 - 62,44 - 1 - 1 = -42,44$$

$$7 - P_{rx} = 20 + 1 + 1 - 61,4709 - 1 - 1 = -41,4709$$

$$8 - P_{rx} = 20 + 1 + 1 - 63,5794 - 1 - 1 = -43,5794$$

$$9 - P_{rx} = 20 + 1 + 1 - 64,9927 - 1 - 1 = -44,9927$$

-30	-32,44	-38,4606	-41,9824
-32,44	-35,4503	-39,4297	-42,44
-38,4606	-39,4297	-41,4709	-43,5794
-41,9824	-42,44	-43,5794	-44,9927

Perda na transmissão do tipo *site-general*:

Para o modelo, serão consideradas duas situações para ser avaliada cada perda, com a combinação dos valores a seguir:

Situação Los -> $\alpha = 1.46$, $\beta = 34.62$, $\gamma = 2.03$ e $\sigma = 3.76$;

Situação NLos -> $\alpha = 2.46$, $\beta = 29.53$, $\gamma = 2.38$ e $\sigma = 5.04$; sendo considerada a perda máxima na transmissão como sendo $10 \log_{10}(10^{0.1A} + 1)$, em que A é uma variável aleatória de distribuição normal $N(\mu, \sigma)$ com média $\mu = L_b(d, f) - (\text{Perda básica de free-space})$, sendo $L_b(d, f) = 10 \cdot \alpha \cdot \log_{10}(d) + \beta + 10 \cdot \gamma \cdot \log_{10}(f)$ e a Perda básica de *free-space* = $20 \log_{10}(4 \times 10^9 \pi d f / c)$, onde d é a distância 3D entre o transmissor e o receptor em metros, f sendo a frequência em GHz e c sendo a velocidade da luz em m/s.

Para serem consideradas as maiores variações, irá ser considerada para a distribuição normal, a cobertura de 99.7% dos valores, tendo variação do valor da média equivalente a 3 desvios padrões, tendo os limites de $\mu \pm 3\sigma$.

Situação Los

Padrões:

$$1 - 10 \cdot 1,46 \cdot \log_{10}(2) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 59,01405315$$

$$2 - 10 \cdot 1,46 \cdot \log_{10}(4) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 63,40909109$$

$$3 - 10 \cdot 1,46 \cdot \log_{10}(6) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 65,98002347$$

$$4 - 10 \cdot 1,46 \cdot \log_{10}(2,828427125) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 61,21157212$$

$$5 - 10 \cdot 1,46 \cdot \log_{10}(4,472135955) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 64,11653418$$

$$6 - 10 \cdot 1,46 \cdot \log_{10}(6,32455532) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 66,31405315$$

$$7 - 10 \cdot 1,46 \cdot \log_{10}(5,656854249) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 65,60661006$$

$$8 - 10 \cdot 1,46 \cdot \log_{10}(7,211102551) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 67,14583962$$

$$9 - 10 \cdot 1,46 \cdot \log_{10}(8,485281374) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 68,17754244$$

30	59,014	63,409	65,98
59,014	61,211	64,116	66,314
63,409	64,116	65,606	67,146
65,98	66,314	67,146	68,177

Potência no receptor para cada padrão de distância:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - \text{Perdas percurso} - \text{Perdas transmissor} - \text{Perdas receptor}$$

$$1 - P_{rx} = 20 + 1 + 1 - 59,014 - 1 - 1 = -39,014$$

$$2 - P_{rx} = 20 + 1 + 1 - 63,409 - 1 - 1 = -43,409$$

$$3 - P_{rx} = 20 + 1 + 1 - 65,980 - 1 - 1 = -45,980$$

$$4 - P_{rx} = 20 + 1 + 1 - 61,211 - 1 - 1 = -41,211$$

$$5 - P_{rx} = 20 + 1 + 1 - 64,116 - 1 - 1 = -44,116$$

$$6 - P_{rx} = 20 + 1 + 1 - 66,314 - 1 - 1 = -46,314$$

$$7 - P_{rx} = 20 + 1 + 1 - 65,606 - 1 - 1 = -45,606$$

$$8 - P_{rx} = 20 + 1 + 1 - 67,145 - 1 - 1 = -47,145$$

$$9 - P_{rx} = 20 + 1 + 1 - 68,177 - 1 - 1 = -48,177$$

-30	-39,014	-43,409	-45,98
-39,014	-41,211	-44,116	-46,314
-43,409	-44,116	-45,606	-47,146
-45,98	-46,314	-47,146	-48,177

Situação Nlos

Perda básica de *free-space*:

$$20 \cdot \log_{10}(4 \cdot 10^9 \pi d f / 3 \cdot 10^8)$$

Padrões:

$$1 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 2 \cdot \frac{5}{3 \cdot 10^8} \right) = 52,442177$$

$$2 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 4 \cdot \frac{5}{3 \cdot 10^8} \right) = 58,4623721$$

$$3 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 6 \cdot \frac{5}{3 \cdot 10^8} \right) = 61,98419728$$

$$4 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 2\sqrt{2} \cdot \frac{5}{3 \cdot 10^8} \right) = 55,45207214$$

$$5 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 2\sqrt{5} \cdot \frac{5}{3 \cdot 10^8} \right) = 59,43147223$$

$$6 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 2\sqrt{10} \cdot \frac{5}{3 \cdot 10^8} \right) = 62,44177219$$

$$7 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 2\sqrt{8} \cdot \frac{5}{3 \cdot 10^8} \right) = 61,47267206$$

$$8 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 2\sqrt{13} \cdot \frac{5}{3 \cdot 10^8} \right) = 63,58120571$$

$$9 - 20 \cdot \log_{10} \left(4 \cdot 10^9 \cdot \pi \cdot 2\sqrt{18} \cdot \frac{5}{3 \cdot 10^8} \right) = 64,99449724$$

20	52,44	58,46	61,98
52,44	55,45	59,43	62,44
58,46	59,43	61,47	63,58
61,98	62,44	63,58	64,99

Perda dada por $L_b(d, f)$:

$$L_b(d, f) = 10 \cdot \alpha \cdot \log_{10}(d) + \beta + 10 \cdot \gamma \cdot \log_{10}(f)$$

Padrões:

$$1 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 53,570824$$

$$2 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(4) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 60,976162$$

$$3 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(6) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 65,30800686$$

$$4 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2\sqrt{2}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 57,27349294$$

$$5 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2\sqrt{5}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 62,16815505$$

$$6 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2\sqrt{10}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 65,870824$$

$$7 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2\sqrt{8}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 64,67883084$$

$$8 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2\sqrt{13}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 67,27232723$$

$$9 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2\sqrt{18}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 69,01067581$$

Tabela de médias $\mu = L_b(d, f) - (\text{Perda básica de free - space})$:

X	11,072	18,477	22,809
11,072	14,774	19,669	23,372
18,477	19,669	22,18	24,773
22,809	23,372	24,773	26,512

Excesso de perdas:

Perda para os padrões de distância:

$$1 - 10 \log_{10}(10^{0,1(N(0,997; 11,072; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (-4,048)} + 1) = 1,44$$

$$10 \log_{10}(10^{0,1 \cdot (26,192)} + 1) = 26,20$$

$$2 - 10 \log_{10}(10^{0,1(N(0,997; 18,447; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (3,3574)} + 1) = 5,005$$

$$10 \log_{10}(10^{0,1 \cdot (33,5974)} + 1) = 33,60$$

$$3 - 10 \log_{10}(10^{0,1(N(0,997; 22,8092; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (7,6892)} + 1) = 8,372$$

$$10 \log_{10}(10^{0,1 \cdot (37,9292)} + 1) = 37,93$$

$$4 - 10 \log_{10}(10^{0,1(N(0,997; 14,7747; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (-0,3453)} + 1) = 2,84$$

$$10 \log_{10}(10^{0,1 \cdot (29,8947)} + 1) = 29,89$$

$$5 - 10 \log_{10}(10^{0,1(N(0,997; 19,6694; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (4,5494)} + 1) = 5,85$$

$$10 \log_{10}(10^{0,1 \cdot (34,7894)} + 1) = 34,79$$

$$6 - 10 \log_{10}(10^{0,1(N(0,997; 23,3720; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (8,2520)} + 1) = 8,85$$

$$10 \log_{10}(10^{0,1 \cdot (38,4920)} + 1) = 38,49$$

$$7 - 10 \log_{10}(10^{0,1(N(0,997; 22,1801; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (7,0601)} + 1) = 7,84$$

$$10 \log_{10}(10^{0,1 \cdot (37,3001)} + 1) = 37,30$$

$$8 - 10 \log_{10}(10^{0,1(N(0,997; 24,7736; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (9,6536)} + 1) = 10,10$$

$$10 \log_{10}(10^{0,1 \cdot (39,8936)} + 1) = 39,89$$

$$9 - 10 \log_{10}(10^{0,1(N(0,997; 26,5119; 5,04))} + 1)$$

$$10 \log_{10}(10^{0,1 \cdot (11,3919)} + 1) = 11,69$$

$$10 \log_{10}(10^{0,1 \cdot (41,6319)} + 1) = 41,63$$

Perda mínima:

X	1,44	5,005	8,372
1,44	2,84	5,85	8,85
5,005	5,85	7,84	10,10
8,372	8,85	10,10	11,69

Perda máxima:

X	26,20	33,60	37,93
26,20	29,89	34,79	38,49
36,60	34,79	37,30	39,89
37,93	38,49	39,89	41,63

Potência no receptor:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - \text{Perdas percurso} - \text{Perdas transmissor} - \text{excesso de perdas} \\ - \text{Perdas receptor}$$

em que, excesso de perdas é a perda considerada como sendo $N(0, \sigma)$, portanto, para considerarmos os valores mais extremos, temos um desvio da média de 3 desvios padrões.

$$1 - P_{rx} = 20 + 1 + 1 - 53,570824 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 53,570824 - 1 - 1 - 15,12 = -48,69$$

$$20 + 1 + 1 - 53,570824 - 1 - 1 + 15,12 = -18,69$$

$$2 - P_{rx} = 20 + 1 + 1 - 60,976162 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 60,976162 - 1 - 1 - 15,12 = -56,09$$

$$20 + 1 + 1 - 60,976162 - 1 - 1 + 15,12 = -25,85$$

$$3 - P_{rx} = 20 + 1 + 1 - 65,3080068 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 65,3080068 - 1 - 1 - 15,12 = -60,428$$

$$20 + 1 + 1 - 65,3080068 - 1 - 1 + 15,12 = -30,188$$

$$4 - P_{rx} = 20 + 1 + 1 - 57,27349294 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 57,27349294 - 1 - 1 - 15,12 = -52,393$$

$$20 + 1 + 1 - 57,27349294 - 1 - 1 + 15,12 = -22,153$$

$$5 - P_{rx} = 20 + 1 + 1 - 62,16815505 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 62,16815505 - 1 - 1 - 15,12 = -57,288$$

$$20 + 1 + 1 - 62,16815505 - 1 - 1 + 15,12 = -27,048$$

$$6 - P_{rx} = 20 + 1 + 1 - 65,870824 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 65,870824 - 1 - 1 - 15,12 = -60,99$$

$$20 + 1 + 1 - 65,870824 - 1 - 1 + 15,12 = -30,75$$

$$7 - P_{rx} = 20 + 1 + 1 - 64,67883084 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 64,67883084 - 1 - 1 - 15,12 = -59,798$$

$$20 + 1 + 1 - 64,67883084 - 1 - 1 + 15,12 = -29,558$$

$$8 - P_{rx} = 20 + 1 + 1 - 67,27232723 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 67,27232723 - 1 - 1 - 15,12 = -62,392$$

$$20 + 1 + 1 - 67,27232723 - 1 - 1 + 15,12 = -32,152$$

$$9 - P_{rx} = 20 + 1 + 1 - 69,01067581 - 1 - 1 - N(0,997; 0; 5,04)$$

$$20 + 1 + 1 - 69,01067581 - 1 - 1 - 15,12 = -64,13$$

$$20 + 1 + 1 - 69,01067581 - 1 - 1 + 15,12 = -33,89$$

Para esse caso, podemos concluir que mesmo no pior caso considerado pelo método, considerando os valores na extremidade negativa da distribuição normal, o sinal ainda tem um nível aceitável para valores de qualidade de sinal, ao ser posicionado apenas um AP em qualquer das posições no local avaliado.

Perdas mínimas | máximas:

-30	-18,69 -48,69	-25,85 -56,09	-30,188 -60,428
-18,69 -48,69	-22,153 -52,393	-27,048 -57,288	-30,75 -60,99
-25,85 -56,09	-27,048 -57,288	-29,558 -59,798	-32,152 -62,392
-30,188 -60,428	-30,75 -60,99	-32,152 -62,392	-33,89 -64,13

Para o cálculo realizado pelo programa, utilizando os mesmos parâmetros, foram obtidos valores dentro dos limites calculados anteriormente, conforme as figuras 8 e 9.

Modelo de cálculo de perda de sinal em ambientes de comunicação sem fio (LoS & NLoS)

Parâmetros de Entrada ⚙️

Escolha o Ambiente

Escritório ▾

Escolha a Condição

☐ LoS

☒ NLoS

Distância (m)

2 - +

Frequência (GHz)

5,00 - +

Potência do AP (dBm)

20,00 - +

Ganho da Antena TX (dB)

1,00 - +

Ganho da Antena RX (dB)

1,00 - +

Perdas no Cabo TX (dB)

1,00 - +

Perdas no Cabo RX (dB)

1,00 - +

Coefficiente de Perda de Potência N

20,00 - +

Número de Linhas da Matriz

4 - +

Número de Colunas da Matriz

4 - +

Calcular Alcance

Figura 8 – Parâmetros de cálculo NLoS_2m

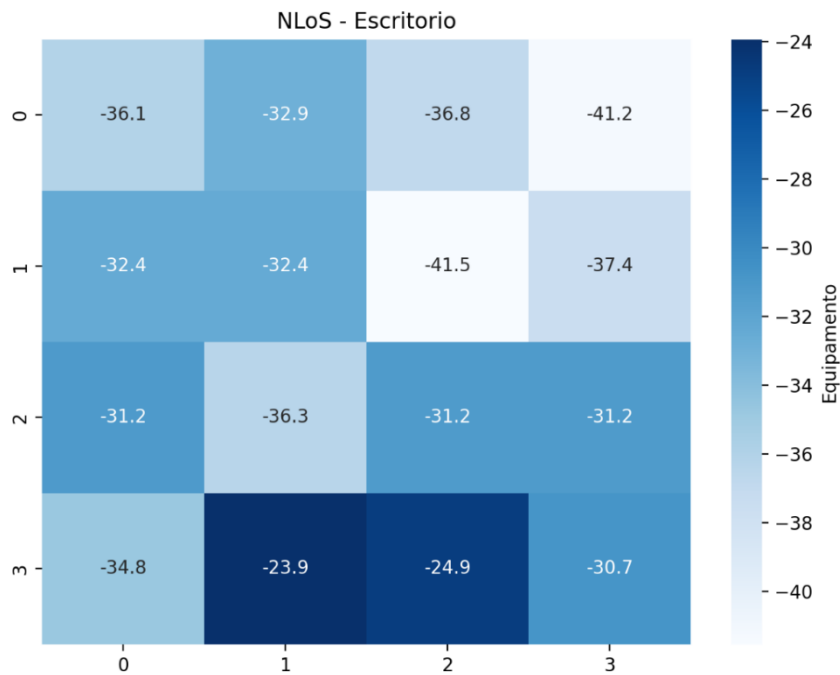


Figura 9 – Valores de resposta NLos_2m

Situação 2:

Lado de cada célula de 200m

Distâncias:

X	700	1400	2100
700	989,9494937	1565,247584	2213,594362
1400	1565,247584	1979,898987	2523,885893
2100	2213,594362	2523,885893	2969,848481

Perda da transmissão tipo *Free-space*

5G Tools:

X	99,9970	106,0176	109,5395
99,9970	103	106,9867	109,997
106,0176	101,980	109,0279	111,1365
109,5395	109,997	111,1365	112,5498

Cálculo matemático das perdas, utilizando a perda da transmissão dada pela expressão $32,44 + 20 \cdot \log_{10}(f) + 20 \cdot \log_{10}(d)$:

Padrões:

$$1 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{700}{1000}\right) = 103,32$$

$$2 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{1400}{1000}\right) = 109,34$$

$$3 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{2100}{1000}\right) = 112,86$$

$$4 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{2}}{1000}\right) = 106,33$$

$$5 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{5}}{1000}\right) = 110,31$$

$$6 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{10}}{1000}\right) = 113,32$$

$$7 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{8}}{1000}\right) = 112,35$$

$$8 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{13}}{1000}\right) = 114,46$$

$$9 - 32,44 + 20 \cdot \log_{10}(5 \cdot 1000) + 20 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{18}}{1000}\right) = 115,87$$

X	103,3213609	109,3419608	112,863786
103,3213609	106,3316608	110,3110609	113,3213609
109,3419608	110,3110609	112,3522608	114,4607944
112,863786	113,3213609	114,4607944	115,8740859

Potência no receptor para cada padrão de distância:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - \text{Perdas percurso} - \text{Perdas transmissor} - \text{Perdas receptor}$$

$$1 - P_{rx} = 20 + 1 + 1 - 103,3213609 - 1 - 1 = -83,3213609$$

$$2 - P_{rx} = 20 + 1 + 1 - 109,3419608 - 1 - 1 = -89,3419608$$

$$3 - P_{rx} = 20 + 1 + 1 - 112,863786 - 1 - 1 = -92,863786$$

$$4 - P_{rx} = 20 + 1 + 1 - 106,3316608 - 1 - 1 = -86,3316608$$

$$5 - P_{rx} = 20 + 1 + 1 - 110,3110609 - 1 - 1 = -90,3110609$$

$$6 - P_{rx} = 20 + 1 + 1 - 113,3213609 - 1 - 1 = -93,3213609$$

$$7 - P_{rx} = 20 + 1 + 1 - 112,3522608 - 1 - 1 = -92,3522608$$

$$8 - P_{rx} = 20 + 1 + 1 - 114,4607944 - 1 - 1 = -94,4607944$$

$$9 - P_{rx} = 20 + 1 + 1 - 115,8740859 - 1 - 1 = -95,8740859$$

-30	-83,3213609	-89,3419608	-92,863786
-83,3213609	-86,3316608	-90,3110609	-93,3213609
-89,3419608	-90,3110609	-92,3522608	-94,4607944
-92,863786	-93,3213609	-94,4607944	-95,8740859

Perda na transmissão do tipo *site-general*:

Para o modelo, serão consideradas duas situações para ser avaliada cada perda, com a combinação dos valores a seguir:

Situação Los -> $\alpha = 1.46$, $\beta = 34.62$, $\gamma = 2.03$ e $\sigma = 3.76$;

Situação NLos -> $\alpha = 2.46$, $\beta = 29.53$, $\gamma = 2.38$ e $\sigma = 5.04$; sendo considerada a perda máxima na transmissão como sendo $10 \log_{10}(10^{0.1A} + 1)$, em que A é uma variável aleatória de distribuição normal $N(\mu, \sigma)$ com média $\mu = L_b(d, f) - (\text{Perda básica de free-space})$, sendo $L_b(d, f) = 10 \cdot \alpha \cdot \log_{10}(d) + \beta + 10 \cdot \gamma \cdot \log_{10}(f)$ e a Perda básica de free-space = $20 \log_{10}(4 \times 10^9 \pi d f / c)$, onde d é a distância 3D entre o transmissor e o receptor em metros, f sendo a frequência em GHz e c sendo a velocidade da luz em m/s.

Para serem consideradas as maiores variações, irá ser considerada para a distribuição normal, a cobertura de 99.7% dos valores, tendo variação do valor da média equivalente a 3 desvios padrões, tendo os limites de $\mu \pm 3\sigma$.

Situação Los

Padrões:

$$1 - 10 \cdot 1,46 \cdot \log_{10}(700) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 90,34752247$$

$$2 - 10 \cdot 1,46 \cdot \log_{10}(1400) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 94,74256$$

$$3 - 10 \cdot 1,46 \cdot \log_{10}(2100) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 97,31349279$$

$$4 - 10 \cdot 1,46 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{2}}{1000}\right) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 92,54504144$$

$$5 - 10 \cdot 1,46 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{5}}{1000}\right) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 95,450004$$

$$6 - 10 \cdot 1,46 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{10}}{1000}\right) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 97,64752247$$

$$7 - 10 \cdot 1,46 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{8}}{1000}\right) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 96,940079$$

$$8 - 10 \cdot 1,46 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{13}}{1000}\right) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 98,47930894$$

$$9 - 10 \cdot 1,46 \cdot \log_{10}\left(\frac{700 \cdot \sqrt{18}}{1000}\right) + 34,62 + 10 \cdot 2,03 \cdot \log_{10}(5) = 99,51101176$$

X	90,34752247	94,74256	97,31349279
90,347522	92,54504144	95,450004	97,64752247
94,74256	95,4500035	96,940079	98,47930894
97,313493	97,64752247	98,479309	99,51101176

Potência no receptor para cada padrão de distância:

$$P_{rx} = P_{tx} + G_{tx} + G_{rx} - \text{Perdas percurso} - \text{Perdas transmissor} - \text{Perdas receptor}$$

$$1 - P_{rx} = 20 + 1 + 1 - 90,34752247 - 1 - 1 = -70,3475$$

$$2 - P_{rx} = 20 + 1 + 1 - 94,74256 - 1 - 1 = -74,74256$$

$$3 - P_{rx} = 20 + 1 + 1 - 97,31349279 - 1 - 1 = -77,31349279$$

$$4 - P_{rx} = 20 + 1 + 1 - 92,54504144 - 1 - 1 = -72,54504144$$

$$5 - P_{rx} = 20 + 1 + 1 - 95,450004 - 1 - 1 = -75,450004$$

$$6 - P_{rx} = 20 + 1 + 1 - 97,64752247 - 1 - 1 = -77,64752247$$

$$7 - P_{rx} = 20 + 1 + 1 - 96,940079 - 1 - 1 = -76,940079$$

$$8 - P_{rx} = 20 + 1 + 1 - 98,47930894 - 1 - 1 = -78,47930894$$

$$9 - P_{rx} = 20 + 1 + 1 - 99,51101176 - 1 - 1 = -79,51101176$$

-30	-70,3475	-74,74256	-77,31349279
-70,347522	-72,54504144	-75,450004	-77,64752247
-74,74256	-75,4500035	-76,940079	-78,47930894
-77,313493	-77,64752247	-78,479309	-79,51101176

Situação Nlos

Perda básica de *free-space*:

$$20 \cdot \log_{10}(4 \cdot 10^9 \pi d f / 3 \cdot 10^8)$$

Padrões:

$$1 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 700 \cdot \frac{5}{3 \cdot 10^8}\right) = 103,3231331$$

$$2 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 1400 \cdot \frac{5}{3 \cdot 10^8}\right) = 109,34373$$

$$3 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 2100 \cdot \frac{5}{3 \cdot 10^8}\right) = 112,8655582$$

$$4 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 700\sqrt{2} \cdot \frac{5}{3 \cdot 10^8}\right) = 106,333433$$

$$5 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 700\sqrt{5} \cdot \frac{5}{3 \cdot 10^8}\right) = 110,31283$$

$$6 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 700\sqrt{10} \cdot \frac{5}{3 \cdot 10^8}\right) = 113,3231331$$

$$7 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 700\sqrt{8} \cdot \frac{5}{3 \cdot 10^8}\right) = 112,35403$$

$$8 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 700\sqrt{13} \cdot \frac{5}{3 \cdot 10^8}\right) = 114,4625666$$

$$9 - 20 \cdot \log_{10}\left(4 \cdot 10^9 \cdot \pi \cdot 700\sqrt{18} \cdot \frac{5}{3 \cdot 10^8}\right) = 115,8758581$$

X	103,3231331	109,34373	112,8655582
103,32313	106,333433	110,31283	113,3231331
109,34373	110,3128331	112,35403	114,4625666
112,86556	113,3231331	114,46257	115,8758581

Perda dada por $L_b(d, f)$:

$$L_b(d, f) = 10 \cdot \alpha \cdot \log_{10}(d) + \beta + 10 \cdot \gamma \cdot \log_{10}(f)$$

Padrões:

$$1 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(700) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 116,1548979$$

$$2 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(1400) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 123,56024$$

$$3 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(2100) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 127,8920808$$

$$4 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(700\sqrt{2}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 119,8575668$$

$$5 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(700\sqrt{5}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 124,75223$$

$$6 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(700\sqrt{10}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 128,4548979$$

$$7 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(700\sqrt{8}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 127,2629$$

$$8 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(700\sqrt{13}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 129,8564011$$

$$9 - L_b(d, f) = 10 \cdot 2.46 \cdot \log_{10}(700\sqrt{18}) + 29.53 + 10 \cdot 2.38 \cdot \log_{10}(5) = 131,5947497$$

Tabela de médias $\mu = L_b(d, f) - (\text{Perda básica de free - space})$:

X	22,7748	30,1801	34,5119
22,7748	26,4774	31,3721	35,0748
30,1801	31,3721	33,8828	36,4763
34,5119	35,0748	36,4763	38,2146

Excesso de perdas:

Perda para os padrões de distância:

$$1 - 10 \log_{10}(10^{0.1(N(0,997; 22,7748; 5,04))} + 1)$$

$$10 \log_{10}(10^{0.1 \cdot (7,6548)} + 1) = 1,44$$

$$10 \log_{10}(10^{0.1 \cdot (37,8948)} + 1) = 26,20$$

$$2 - 10 \log_{10}(10^{0.1(N(0,997; 30,1801; 5,04))} + 1)$$

$$10 \log_{10}(10^{0.1 \cdot (15,0601)} + 1) = 5,005$$

$$10 \log_{10}(10^{0.1 \cdot (45,3001)} + 1) = 33,60$$

$$3 - 10 \log_{10}(10^{0.1(N(0,997; 34,5119; 5,04))} + 1)$$

$$10 \log_{10}(10^{0.1 \cdot (19,3919)} + 1) = 8,372$$

$$10 \log_{10}(10^{0.1 \cdot (49,6319)} + 1) = 37,93$$

$$4 - 10 \log_{10}(10^{0.1(N(0,997; 26,4774; 5,04))} + 1)$$

$$\begin{aligned}
&10 \log_{10}(10^{0,1 \cdot (11,3574)} + 1) = 2,84 \\
&10 \log_{10}(10^{0,1 \cdot (41,5974)} + 1) = 29,89 \\
5 - &10 \log_{10}(10^{0,1 \cdot (N(0,997; 31,3721; 5,04))} + 1) \\
&10 \log_{10}(10^{0,1 \cdot (16,2521)} + 1) = 5,85 \\
&10 \log_{10}(10^{0,1 \cdot (46,4921)} + 1) = 34,79 \\
6 - &10 \log_{10}(10^{0,1 \cdot (N(0,997; 35,0748; 5,04))} + 1) \\
&10 \log_{10}(10^{0,1 \cdot (19,9548)} + 1) = 8,85 \\
&10 \log_{10}(10^{0,1 \cdot (50,1948)} + 1) = 38,49 \\
7 - &10 \log_{10}(10^{0,1 \cdot (N(0,997; 33,8828; 5,04))} + 1) \\
&10 \log_{10}(10^{0,1 \cdot (18,7628)} + 1) = 7,84 \\
&10 \log_{10}(10^{0,1 \cdot (49,0028)} + 1) = 37,30 \\
8 - &10 \log_{10}(10^{0,1 \cdot (N(0,997; 36,4763; 5,04))} + 1) \\
&10 \log_{10}(10^{0,1 \cdot (21,3563)} + 1) = 10,10 \\
&10 \log_{10}(10^{0,1 \cdot (51,5963)} + 1) = 39,89 \\
9 - &10 \log_{10}(10^{0,1 \cdot (N(0,997; 38,2146; 5,04))} + 1) \\
&10 \log_{10}(10^{0,1 \cdot (23,0946)} + 1) = 11,69 \\
&10 \log_{10}(10^{0,1 \cdot (53,3346)} + 1) = 41,63
\end{aligned}$$

Perda mínima:

X	8,3426	15,1935	19,4416
8,3426	11,6641	16,3538	19,9984
15,1935	16,3538	18,8201	21,3879
19,4416	19,9984	21,3879	23,1159

Perda máxima:

X	37,8955	45,3002	49,6320
37,8955	41,5977	46,4922	50,1948
45,3002	46,4922	49,0028	51,5963
49,6320	50,1948	51,5963	53,3346

Potência no receptor:

$$\begin{aligned}
P_{rx} = &P_{tx} + G_{tx} + G_{rx} - \text{Perdas percurso} - \text{Perdas transmissor} - \text{excesso de perdas} \\
&- \text{Perdas receptor}
\end{aligned}$$

em que, excesso de perdas é a perda considerada como sendo $N(0, \sigma)$, portanto, para considerarmos os valores mais extremos, temos um desvio da média de 3 desvios padrões.

$$\begin{aligned}
1 - P_{rx} = &20 + 1 + 1 - 116,1548979 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 116,1548979 - 1 - 1 - 15,12 = -81,0348979
\end{aligned}$$

$$\begin{aligned}
&20 + 1 + 1 - 116,1548979 - 1 - 1 + 15,12 = -111,2748979 \\
2 - P_{rx} &= 20 + 1 + 1 - 123,56024 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 123,56024 - 1 - 1 - 15,12 = -88,44023578 \\
&20 + 1 + 1 - 123,56024 - 1 - 1 + 15,12 = -118,6802358 \\
3 - P_{rx} &= 20 + 1 + 1 - 127,8920808 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 127,8920808 - 1 - 1 - 15,12 = -92,7720808 \\
&20 + 1 + 1 - 127,8920808 - 1 - 1 + 15,12 = -123,0120808 \\
4 - P_{rx} &= 20 + 1 + 1 - 119,8575668 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 119,8575668 - 1 - 1 - 15,12 = -84,7375668 \\
&20 + 1 + 1 - 119,8575668 - 1 - 1 + 15,12 = -114,9775668 \\
5 - P_{rx} &= 20 + 1 + 1 - 124,7522289 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 124,7522289 - 1 - 1 - 15,12 = -89,6322289 \\
&20 + 1 + 1 - 124,7522289 - 1 - 1 + 15,12 = -119,8722289 \\
6 - P_{rx} &= 20 + 1 + 1 - 128,4548979 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 128,4548979 - 1 - 1 - 15,12 = -93,3348979 \\
&20 + 1 + 1 - 128,4548979 - 1 - 1 + 15,12 = -123,5748979 \\
7 - P_{rx} &= 20 + 1 + 1 - 127,2629047 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 127,2629047 - 1 - 1 - 15,12 = -92,1429047 \\
&20 + 1 + 1 - 127,2629047 - 1 - 1 + 15,12 = -122,3829047 \\
8 - P_{rx} &= 20 + 1 + 1 - 129,8564011 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 129,8564011 - 1 - 1 - 15,12 = -94,7364011 \\
&20 + 1 + 1 - 129,8564011 - 1 - 1 + 15,12 = -124,9764011 \\
9 - P_{rx} &= 20 + 1 + 1 - 131,5947497 - 1 - 1 - N(0,997; 0; 5,04) \\
&20 + 1 + 1 - 131,5947497 - 1 - 1 - 15,12 = -96,4747497 \\
&20 + 1 + 1 - 131,5947497 - 1 - 1 + 15,12 = -126,7147497
\end{aligned}$$

Para esse caso, em todas as outras áreas fora a do AP ficam fora da faixa de um sinal aceitável, necessitando ser colocado um AP em cada fração de área.

-30	-81,03 - 111,27	-88,44 -118,68	-92,77 - 123,01
-81,03 -111,27	-84,73 - 114,97	-89,63 - 119,87	-93,33 -123,57
-88,44 -118,68	-89,63 - 119,87	-92,14 - 122,38	-94,73 - 124,97
-92,77 - 123,01	-93,33 -123,57	-94,73 - 124,97	-96,47 -126,71

Para o cálculo realizado pelo programa, utilizando os mesmos parâmetros, foram obtidos valores, em sua maioria, dentro dos limites calculados anteriormente, conforme as figuras 10 e 11.

Escolha a Condição

☐ LoS

☒ NLoS

Distância (m)

700 - +

Frequência (GHz)

5,00 - +

Potência do AP (dBm)

20,00 - +

Ganho da Antena TX (dB)

1,00 - +

Ganho da Antena RX (dB)

1,00 - +

Perdas no Cabo TX (dB)

1,00 - +

Perdas no Cabo RX (dB)

1,00 - +

Coefficiente de Perda de Potência N

20,00 - +

Número de Linhas da Matriz

4 - +

Número de Colunas da Matriz

4 - +

Calcular Alcance

Figura 10 – Parâmetros NLoS_700m

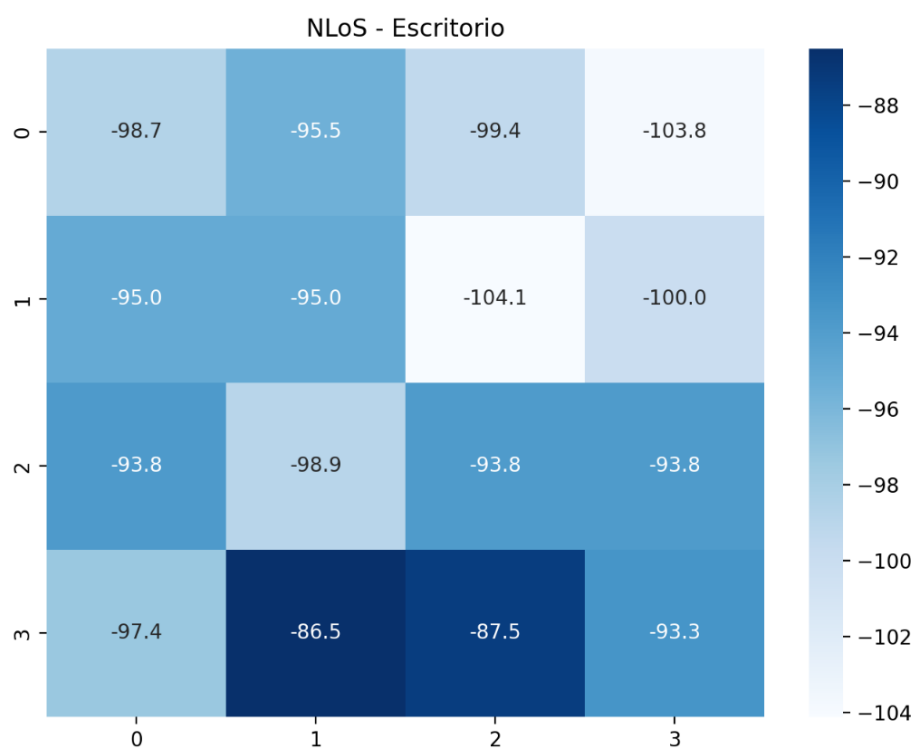


Figura 11 – Valores NLoS_700m

5. RESULTADOS

5.1. CÁLCULO DE POSICIONAMENTO FREE-SPACE

Com o cálculo implementado, foi possível definir os locais ideais para os APs nesse modelo de propagação, chegando ao objetivo da implementação. Na aplicação, foram obtidos conhecimentos práticos que auxiliam na complementação da formação acadêmica.

5.2. CÁLCULO DE POSICIONAMENTO SITE-GENERAL

Implementado o cálculo do modelo *Site-General*, foram obtidos os locais ideais dos APs para diferentes considerações de ambientes, previstos nas considerações do modelo de propagação, chegando ao objetivo da implementação.

5.3. INTERFACE

A interface foi implementada de maneira a ser possível selecionar, na mesma interface, o método desejado de cálculo, em uma página web local, tendo uma página inicial e uma página para cada método, conforme a Figura 12.

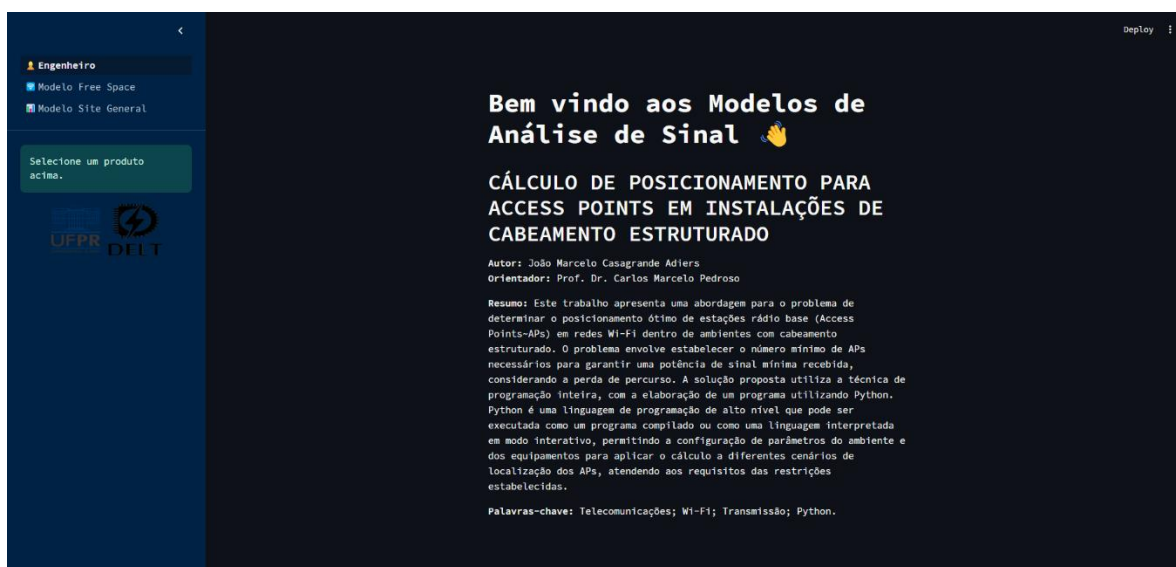


Figura 12 – Tela inicial da interface

Na página de cada método, o usuário pode ajustar os valores dos parâmetros variáveis para os desejados, conforme a Figura 13:

The image displays two side-by-side screenshots of a software interface for signal loss calculation. Both interfaces are in Portuguese and feature a dark blue theme.

Left Screenshot: Modelo Free Space

- Engenheiro** (User profile)
- Modelo Free Space** (Selected tab)
- Modelo Site General** (Available tab)
- Modelo de cálculo de perda de sinal em um ambiente idealizado onde não há obstáculos** (Title)
- Parâmetros de Entrada** (Section header)
- Frequência (MHz)**: 5000,00
- Potência do AP (dBm)**: 28,00
- Ganho da Antena TX (dB)**: 1,00
- Ganho da Antena RX (dB)**: 1,00
- Perdas no Cabo TX (dB)**: 1,00
- Perdas no Cabo RX (dB)**: 1,00
- Potência Limite (dBm)**: -38,00
- Número de Linhas Matriz**: 4
- Número de Colunas Matriz**: 4
- Calcular Alcance** (Button)

Right Screenshot: Modelo Site General

- Modelo Free Space** (Available tab)
- Modelo Site General** (Selected tab)
- Modelo de cálculo de perda de sinal em ambientes de comunicação sem fio (LoS & NLoS)** (Title)
- Parâmetros de Entrada** (Section header)
- Escolha o Ambiente**: Escritório (Dropdown)
- Escolha a Condição**: ☒ LoS, ☐ NLoS
- Distância (m)**: 27
- Frequência (GHz)**: 2,50
- Potência do AP (dBm)**: 28,00
- Ganho da Antena TX (dB)**: 1,00
- Ganho da Antena RX (dB)**: 1,00
- Perdas no Cabo TX (dB)**: 1,00
- Perdas no Cabo RX (dB)**: 1,00
- Coefficiente de Perda de Potência N**: 20,00
- Número de Linhas da Matriz**: 4
- Número de Colunas da Matriz**: 4
- Calcular Alcance** (Button)

Figura 13 - Ajuste de parâmetros variáveis

Após o ajuste, o usuário pode clicar no botão para calcular o alcance e a interface chamará o código para fazer o cálculo do posicionamento dos APs e retornará a apresentação gráfica dos resultados, em uma matriz, conforme as Figuras 14 e 15:

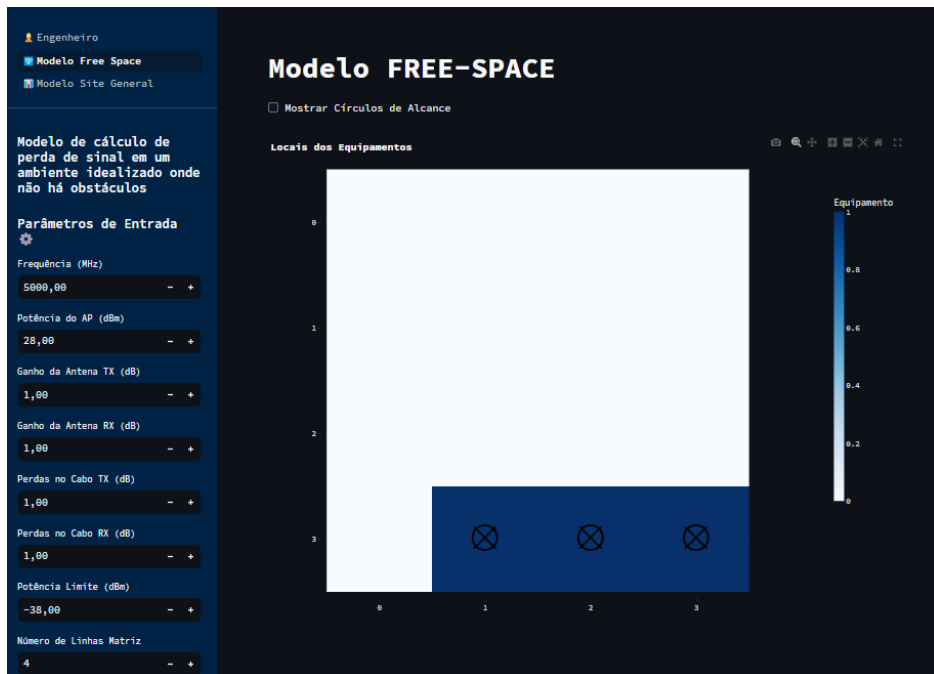


Figura 14 – Tela da interface do método Free-Space

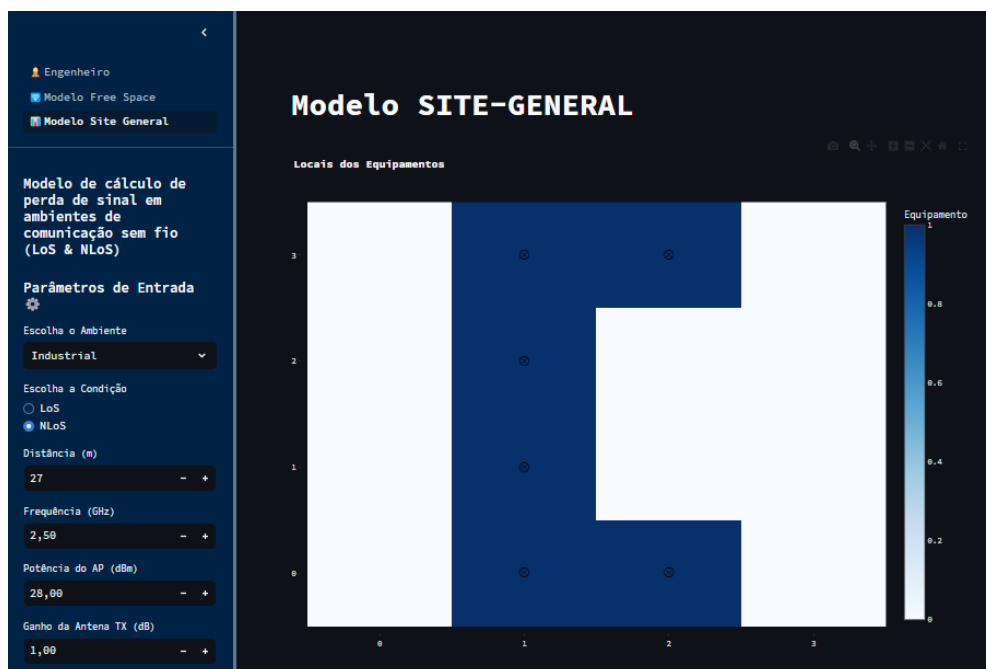


Figura 15 – Tela da interface do método Site-General

onde as posições das matrizes ocupadas por um quadrado azul escuro representam as posições necessárias de ter APs para poder satisfazer as restrições do problema.

Após o cálculo, há um *checkbox* que pode ser selecionado para mostrar a matriz de seleção de área de interesse, apresentada na Figura 5, na seção 4. As

células da matriz são selecionadas por meio de botões, selecionando as áreas para estarem ativas e deixando sem selecionar as áreas inativas, conforme a Figura 16:

☒ **Mostrar Matriz Interativas**

Escolha o Matriz:

(0, 0) ▾

	0	1	2	3
0	0.63	-31.91	-37.93	-41.45
1	-31.91	-34.92	-38.90	-41.91
2	-37.93	-38.90	-40.94	-43.05
3	-41.45	-41.91	-43.05	-44.46

Selecionar Matriz (0, 0)

Selecionar Matriz (0, 1)

Selecionar Matriz (0, 2)

Selecionar Matriz (0, 3)

Selecionar Matriz (1, 0)

Selecionar Matriz (1, 1)

Selecionar Matriz (1, 2)

Selecionar Matriz (1, 3)

Selecionar Matriz (2, 0)

Selecionar Matriz (2, 1)

Selecionar Matriz (2, 2)

Selecionar Matriz (2, 3)

Selecionar Matriz (3, 0)

Selecionar Matriz (3, 1)

Selecionar Matriz (3, 2)

Selecionar Matriz (3, 3)

Figura 16 – Matriz de seleção de área de interesse

6. CONCLUSÃO

Em conclusão, este trabalho apresentou uma metodologia para otimização do posicionamento de *Access Points* (APs) em redes *Wi-Fi*, com o objetivo de minimizar o número de APs necessários enquanto se garante uma potência mínima recebida em ambientes de cabeamento estruturados. Utilizando técnicas de programação inteira, foi possível desenvolver um modelo que considera diferentes cenários de localização de APs e suas restrições. Foram implementados modelos de desvanecimento de sinal, como o modelo *Free-space* e o *Site-General* (*NLoS* e *LoS*), que permitiram calcular a perda de sinal em diversos ambientes. A aplicação prática do modelo mostrou-se eficiente, alcançando os objetivos propostos e proporcionando uma ferramenta valiosa para o planejamento e implementação de redes *Wi-Fi*.

A metodologia desenvolvida neste trabalho mostrou-se eficaz na otimização do posicionamento de APs, permitindo uma cobertura de rede mais eficiente e econômica. A utilização de programação inteira e os modelos de desvanecimento de sinal aplicáveis contribuíram significativamente para a precisão dos cálculos e a obtenção de resultados satisfatórios. A implementação prática dos algoritmos e o desenvolvimento da interface em *Python*, utilizando o *framework Streamlit*, facilitaram a aplicação e visualização dos resultados, tornando o processo acessível e intuitivo, na construção da aplicação dos modelos.

Apesar dos resultados positivos, algumas limitações foram identificadas. A principal limitação está relacionada à simplificação dos modelos de desvanecimento de sinal, que podem não capturar todas as variáveis e peculiaridades dos ambientes reais. Além disso, a necessidade de entrada manual de dados na forma de matriz pode tornar o processo de configuração inicial mais demorado e suscetível a erros humanos. Outro ponto a ser considerado é a limitação do modelo em lidar com ambientes altamente dinâmicos, onde mudanças frequentes na disposição física ou nas condições de propagação do sinal podem exigir reconfigurações constantes.

Para aprimorar a metodologia desenvolvida, futuras pesquisas podem explorar diversas direções. Uma melhoria na interface, possibilitando a importação direta de plantas baixas em formatos digitais (como CAD ou imagens) e sua conversão automática em matrizes para cálculo, é uma prioridade. A implementação de modelos de desvanecimento mais complexos e específicos para diferentes tipos de ambientes, levando em consideração fatores como a presença de móveis, densidade de pessoas e materiais de construção, pode aumentar a precisão do modelo.

Além disso, o desenvolvimento de algoritmos adaptativos que possam re-calibrar a posição dos APs em tempo real, conforme mudanças nas condições do ambiente ou nas necessidades de cobertura da rede, tornaria o sistema mais robusto. Por fim, a aplicação da metodologia em estudos de caso reais em diferentes tipos de instalações (residenciais, escritórios, comerciais, industriais) validaria e refinaria o modelo proposto. Com estas melhorias, espera-se que a metodologia desenvolvida possa se tornar uma ferramenta ainda mais robusta e versátil para o planejamento e otimização de redes *Wi-Fi*, atendendo às demandas crescentes por conectividade eficiente e confiável.

7.BIBLIOGRAFIA

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 14565: Procedimento básico para elaboração de projetos de cabeamento de telecomunicações para rede interna estruturada. Rio de Janeiro. 2000.

CISCO. O que é um ponto de acesso e como ele funciona? Disponível em: <https://www.cisco.com/c/pt_br/solutions/small-business/resource-center/networking/what-is-access-point.html#~tipos-de-access-points>. Acesso em: 21 nov. 2023.

D. Bertsimas e J. N. Tsitsiklis. Introduction to Linear Optimization, Athena Scientific, 1997

DANTZIG, George B.; THAPA, Mukund N. The simplex method. Springer New York, 1997.

EMBARCADOS. A evolução do protocolo *Wi-Fi* IEEE 802.11. Disponível em: <<https://embarcados.com.br/a-evolucao-do-protocolo-wi-fi-ieee-802-11/>>. Acesso em: 21 nov. 2023.

HUAWEI. What Is 802.11ac?. Disponível em: <<https://info.support.huawei.com/info-finder/encyclopedia/en/802.11ac.html>>. Acesso em: 25 nov. 2023.

Recommendation ITU-R P.1238-12, Propagation data and prediction methods for the planning of indoor radiocommunication systems and radio local area networks in the frequency range 300 MHz to 450 GHz, Electronic Publication Geneva, 2023. Disponível em: <<https://www.itu.int/rec/R-REC-P.1238-12-202308-I/BR>>. Acesso em: 20 nov. 2023.

Kagan, N., Schmidt, H. P., de Oliveira, C. C. B., & Kagan, H. (2009). *Métodos de otimização aplicados a sistemas elétricos de potência*. Editora Blucher.

Recommendation ITU-R P.525-4, Calculation of free-space attenuation, Electronic Publication Geneva, 2019. Disponível em: <<https://www.itu.int/rec/R-REC-P.525/en>>. Acesso em: 18 jun. 2024.

MITCHELL, Stuart; KEAN, Anita; MASON Andrew; O'SULIVAN, Michael; PHILLIPS, Antony; PESCHIERA, Franco. *PuLP: Version 2.8.0*. Disponível em: <<https://github.com/coin-or/pulp>>. Acesso em: 21 mar. 2024.

PYTHON SOFTWARE FOUNDATION. Python 3.11.7. 2023. Disponível em: <<https://www.python.org/>>. Acesso em: 18 jun. 2024.

TELECO. WLAN X Sistemas Móveis Celulares: Faixas de Frequências. Disponível em: <https://www.teleco.com.br/tutoriais/tutorialwlanx/pagina_3.asp>. Acesso em: 23 nov. 2023.

TRICK, Michael A. A tutorial on integer programming. The Operations Research Faculty of GSIA, 1997.

VINOGRADOV, Oleg. 5G NR Link budget calculator. Disponível em: <https://5g-tools.com/5g-nr-link-budget-calculator/>. Acesso em: 20 jun. 2024.

8. ANEXOS

ANEXO A

CÓDIGO EM *PYTHON* DO PROBLEMA EXEMPLO

```
from pulp import *
problema = LpProblem("SetCoveringProblem", LpMinimize)
x = LpVariable.dicts("x", range(1, 12), cat=LpBinary)
problema += lpSum(x[i] for i in range(1, 12))
constraints = [
    [1, 2, 3, 4],
    [1, 2, 3, 5],
    [1, 2, 3, 4, 5, 6],
    [1, 3, 4, 6, 7],
    [2, 3, 5, 6, 8, 9],
    [3, 4, 5, 6, 7, 8],
    [4, 6, 7, 8],
    [5, 6, 7, 8, 9, 10],
    [5, 8, 9, 10, 11],
    [8, 9, 10, 11],
    [9, 10, 11]
]
for i, c in enumerate(constraints, start=1):
    problema += lpSum(x[j] for j in c) >= 1
problema.solve()
print("Status:", LpStatus[problema.status])
print("Valor_da_função_objetivo:", value (problema.objective))
print("Valores das variáveis:")
for v in problema.variables():
    print(v.name, "=", v.varValue)
```

ANEXO B

CÓDIGO FONTE EM *PYTHON*

```
import math
import numpy as np
from pulp import LpProblem, LpMinimize, LpVariable, LpBinary, lpSum, value,
LpStatus
from statistics import NormalDist

def criar_matriz_tridimensional(x: int, y: int, z: int, valor_inicial=0):
    """Cria uma matriz tridimensional.

    Parameters
    -----
    x : int
        Entrada
    y : int
        Entrada
    z : int
        Entrada
    valor_inicial : int, optional
        Valor inicial: 0.

    Returns
    -----
    list
        Matriz tridimensional.
    """
    return [[[valor_inicial] * z for _ in range(y)] for _ in range(x)]

def calcular_alcance(LINHAS: int, COLUNAS: int,
                    freq: float, P_tx: float,
                    G_tx: float, G_rx: float,
                    Cable_tx_loss: float, Cable_rx_loss: float,
                    dist_cel: int):
    #Pen_loss: float):
    """Calcula o alcance do sinal.

    Parameters
    -----
    LINHAS : int
        Linhas na matriz.
    COLUNAS : int
        Colunas na matriz.
    freq : float
        Frequência (MHz).
```

```

P_tx : float
    Potência (dBm).
G_tx : float
    Ganho da antena (dB).
G_rx : float
    Ganho da antena (dB).
Cable_tx_loss : float
    Perda transmissor (dB).
Cable_rx_loss : float
    Perda receptor (dB).
#Pen_loss : float
    Perdas queda (dB). # não vai mais usar
dist_cel : int
    distancia da celula
Returns
-----
list
    Matriz com o alcance calculado.
"""
alcance = criar_matriz_tridimensional(LINHAS*COLUNAS,
                                      LINHAS, COLUNAS)

for i in range(LINHAS*COLUNAS):
    for row in range(LINHAS):
        for column in range(COLUNAS):
            cat_1 = abs(column-i % COLUNAS)
            cat_2 = abs(row-i // COLUNAS)
            dist_m = math.hypot(cat_1, cat_2)
            #teste dist
            dist_m_km = (dist_m * dist_cel) / 1000
            if dist_m == 0:
                alcance[i][row][column] = 10*math.log10(pow(10, -3) *
pow(10, (P_tx / 10))/10)
            else:
                #path_loss = 32.4 + 20 *(math.log10(freq)) + 20
                *(math.log10(dist_m_km))
                #alcance[i][row][column] = P_tx + G_tx + G_rx - path_loss
                - Cable_tx_loss - Cable_rx_loss
                # alcance[i][row][column] = P_tx*pow(10,((G_tx + G_rx
                #
                #                                     - (32.4 + 20*(math.log10(freq))+
                #                                     20*(math.log10(dist_m_km))) -
                #                                     Cable_rx_loss -
                Cable_tx_loss)/10))
                alcance[i][row][column] =
                10*math.log10((P_tx*pow(10,((G_tx + G_rx
                #                                     - (32.44 + 20*(math.log10(freq))+
                #                                     20*(math.log10(dist_m_km))) -
                #                                     Cable_rx_loss -
                Cable_tx_loss)/10))))

```

```

        #alcance[i][row][column] =
10*math.log10((P_tx*pow(10,(32.44 +
20*(math.log10(freq))+20*(math.log10(dist_m_km)))/10)))
    return alcance

def resolver_cobertura(LINHAS: int, COLUNAS: int,
                       alcance: list, P_lim: float):
    """Resolve o problema de cobertura.

    Parameters
    -----
    LINHAS : int
        Linhas na matriz.
    COLUNAS : int
        Colunas na matriz.
    alcance : list
        Matriz de alcances.
    P_lim : float
        Potência mínima.

    Returns
    -----
    tuple
    """
    problema = LpProblem("SetCoveringProblem", LpMinimize)
    x = LpVariable.dicts("x", range(1, LINHAS*COLUNAS+1), cat=LpBinary)
    problema += lpSum(x[i] for i in range(1, LINHAS*COLUNAS+1))

    constraints = []
    for i in range(LINHAS*COLUNAS):
        linha = []
        for j in range(LINHAS):
            for k in range(COLUNAS):
                if alcance[i][j][k] >= P_lim:
                    linha.append(j * COLUNAS + k + 1)
            constraints.append(linha)

    for i, c in enumerate(constraints, start=1):
        problema += lpSum(x[j] for j in c) >= 1

    problema.solve()

    return LpStatus[problema.status], value(problema.objective), [v.varValue
for v in problema.variables()]

# Perda de caminho em espaço livre (FSPL)
def free_space_path_loss(d:float, f:float, ):
    """

```

Calcula a perda de caminho em espaço livre (FSPL).

Parameters

d : float

Distância entre transmissor e receptor em metros.

f : float

Frequência do sinal em Hz.

Returns

float

Perda de caminho em espaço livre em dB.

"""

c = 3 * 10**8 # velocidade da luz em metros/segundo

return 20 * np.log10((4 * np.pi * d * f * 10**9) / c)

Perda de transmissão básica (BTL)

```
def basic_transmission_loss(d:float, f:float,  
                             alpha: float, beta:float, gamma: float):
```

"""

Calcula a perda de transmissão básica (BTL).

Parameters

d : float

Distância entre transmissor e receptor em metros.

f : float

Frequência do sinal em Hz.

alpha : float

Parâmetro alfa do modelo.

beta : float

Parâmetro beta do modelo.

gamma : float

Parâmetro gama do modelo.

Returns

float

Perda de transmissão básica em dB.

"""

return 10 * alpha * np.log10(d) + beta + 10 * gamma * np.log10(f)

NLoS Monte Carlo

```
def nlos_monte_carlo(d: float, f: float,  
                     alpha: float, beta: float, gamma: float, sigma: float,  
                     P_tx: float, G_tx: float, G_rx: float, Cable_tx_loss:  
float, Cable_rx_loss: float,
```

```

        LINHAS: int, COLUNAS: int, condition: str,
seed_value=42):
    """
    Calcula a perda de sinal em ambiente NLoS ou LoS usando o método de Monte
    Carlo com a fórmula da imagem.

    Parameters
    -----
    condition : str
        Condição escolhida ('LoS' ou 'NLoS'). Se 'LoS', o ruído gaussiano não
        será adicionado.
        (demais parâmetros idênticos...)
    """

    np.random.seed(seed_value)

    # Perda em espaço livre (L_FS)
    c = 3 * 10**8
    L_FS = 20 * np.log10((4 * np.pi * d * f * 10**9) / c) # dB

    # Perda de transmissão básica (L_b) usando a fórmula da imagem
    L_b = 10 * alpha * np.log10(d) + beta + 10 * gamma * np.log10(f)

    if condition == 'NLoS':
        gaussian_noise = np.random.normal(0, sigma, size=(int(LINHAS),
int(COLUNAS)))
        excess_loss = gaussian_noise #- L_FS + L_b
    else:
        mean_excess_loss = 10 * np.log10(P_tx + (G_tx + G_rx - L_FS -
Cable_tx_loss - Cable_rx_loss) / 10)
        # excess_loss = np.random.normal(mean_excess_loss, sigma,
size=(int(LINHAS), int(COLUNAS)))
        excess_loss = np.random.normal((L_b - L_FS), sigma, size=(int(LINHAS),
int(COLUNAS)))
        print(excess_loss)

    # Potência recebida (P_rx)
    P_rx = P_tx + G_tx + G_rx - excess_loss - L_b - Cable_tx_loss -
Cable_rx_loss
    return P_rx

```

ANEXO C

CÓDIGO INTERFACE – PÁGINA PRINCIPAL

```
"""Landing page of WI-FI TCC."""
```

```
import streamlit as st
from PIL import Image
import base64
from io import BytesIO
```

```
def get_image_as_base64(image_path):
    img = Image.open(image_path)
    buffered = BytesIO()
    img.save(buffered, format="PNG")
    return base64.b64encode(buffered.getvalue()).decode()
```

```
st.set_page_config(
    page_title="Wi-Fi TCC",
    page_icon="👋",
)
```

```
st.sidebar.success("Selecione um produto acima.")
```

```
ufpr_image_base64 =
get_image_as_base64("../codigo_com_interface/src/marca_UFPR.png")
delt_image_base64 = get_image_as_base64("../codigo_com_interface/src/delt_UFPR.png")
```

```
st.sidebar.markdown(
    f"""
    <div style="display: flex; justify-content: center; align-items: center;">
        
        
    </div>
    """,
    unsafe_allow_html=True,
)
```

```
st.markdown(
    """
    <style>
    div[data-testid='stSidebarNav'] ul {max-height:none}</style>
    """,
    unsafe_allow_html=True,
)
```

```
st.write("# Bem vindo aos Modelos de Análise de Sinal 🙌")
```

```
st.markdown(
```

```
    ""
```

```
    ## CÁLCULO DE POSICIONAMENTO PARA ACCESS POINTS EM INSTALAÇÕES DE  
    CABEAMENTO ESTRUTURADO
```

```
    **Autor:** João Marcelo Casagrande Adiers
```

```
    **Orientador:** Prof. Dr. Carlos Marcelo Pedroso
```

```
    **Resumo:**
```

```
    Este trabalho apresenta uma abordagem para o problema de determinar o  
    posicionamento ótimo de estações rádio base (Access Points~APs) em redes Wi-Fi dentro  
    de ambientes com cabeamento estruturado. O problema envolve estabelecer o número  
    mínimo de APs necessários para garantir uma potência de sinal mínima recebida,  
    considerando a perda de percurso. A solução proposta utiliza a técnica de programação  
    inteira, com a elaboração de um programa utilizando Python. Python é uma linguagem de  
    programação de alto nível que pode ser executada como um programa compilado ou como  
    uma linguagem interpretada em modo interativo, permitindo a configuração de parâmetros  
    do ambiente e dos equipamentos para aplicar o cálculo a diferentes cenários de localização  
    dos APs, atendendo aos requisitos das restrições estabelecidas.
```

```
    **Palavras-chave:** Telecomunicações; Wi-Fi; Transmissão; Python.
```

```
    ""
```

```
)
```


ANEXO D

CÓDIGO INTERFACE – *FREE-SPACE*

```
import math
import numpy as np
from pulp import LpProblem, LpMinimize, LpVariable, LpBinary, lpSum, value, LpStatus
from statistics import NormalDist

def criar_matriz_tridimensional(x: int, y: int, z: int, valor_inicial=0):
    """Cria uma matriz tridimensional.

    Parameters
    -----
    x : int
        Entrada
    y : int
        Entrada
    z : int
        Entrada
    valor_inicial : int, optional
        Valor inicial: 0.

    Returns
    -----
    list
        Matriz tridimensional.
    """
    return [[[valor_inicial] * z for _ in range(y)] for _ in range(x)]

def calcular_alcance(LINHAS: int, COLUNAS: int,
                    freq: float, P_tx: float,
                    G_tx: float, G_rx: float,
                    Cable_tx_loss: float, Cable_rx_loss: float,
                    dist_cel: int):
    """Calcula o alcance do sinal.

    Parameters
    -----
    LINHAS : int
        Linhas na matriz.
    COLUNAS : int
        Colunas na matriz.
    freq : float
        Frequência (MHz).
```

```

P_tx : float
    Potência (dBm).
G_tx : float
    Ganho da antena (dB).
G_rx : float
    Ganho da antena (dB).
Cable_tx_loss : float
    Perda transmissor (dB).
Cable_rx_loss : float
    Perda receptor (dB).
#Pen_loss : float
    Perdas queda (dB). # não vai mais usar
dist_cel : int
    distancia da celula
Returns
-----
list
    Matriz com o alcance calculado.
"""

alcance = criar_matriz_tridimensional(LINHAS*COLUNAS,
                                      LINHAS, COLUNAS)
for i in range(LINHAS*COLUNAS):
    for row in range(LINHAS):
        for column in range(COLUNAS):
            cat_1 = abs(column-i % COLUNAS)
            cat_2 = abs(row-i // COLUNAS)
            dist_m = math.hypot(cat_1, cat_2)
            #teste dist
            dist_m_km = (dist_m * dist_cel) / 1000
            if dist_m == 0:
                alcance[i][row][column] = 10*math.log10(pow(10, -3) * pow(10, (P_tx / 10))/10)
            else:
                #path_loss = 32.4 + 20 *(math.log10(freq)) + 20 *(math.log10(dist_m_km))
                #alcance[i][row][column] = P_tx + G_tx + G_rx - path_loss - Cable_tx_loss -
Cable_rx_loss
                # alcance[i][row][column] = P_tx*pow(10,((G_tx + G_rx
                #             - (32.4 + 20*(math.log10(freq))+
                #             20*(math.log10(dist_m_km)))) -
                #             Cable_rx_loss - Cable_tx_loss)/10))
                alcance[i][row][column] = 10*math.log10((P_tx*pow(10,((G_tx + G_rx
                - (32.44 + 20*(math.log10(freq))+
                20*(math.log10(dist_m_km)))) -
                Cable_rx_loss - Cable_tx_loss)/10))))
                #alcance[i][row][column] = 10*math.log10((P_tx*pow(10,(32.44 +
                20*(math.log10(freq))+20*(math.log10(dist_m_km))))/10)))
            return alcance

def resolver_cobertura(LINHAS: int, COLUNAS: int,

```

```

        alcance: list, P_lim: float):
    """Resolve o problema de cobertura.

    Parameters
    -----
    LINHAS : int
        Linhas na matriz.
    COLUNAS : int
        Colunas na matriz.
    alcance : list
        Matriz de alcances.
    P_lim : float
        Potência mínima.

    Returns
    -----
    tuple
    """
    problema = LpProblem("SetCoveringProblem", LpMinimize)
    x = LpVariable.dicts("x", range(1, LINHAS*COLUNAS+1), cat=LpBinary)
    problema += lpSum(x[i] for i in range(1, LINHAS*COLUNAS+1))

    constraints = []
    for i in range(LINHAS*COLUNAS):
        linha = []
        for j in range(LINHAS):
            for k in range(COLUNAS):
                if alcance[i][j][k] >= P_lim:
                    linha.append(j * COLUNAS + k + 1)
            constraints.append(linha)

    for i, c in enumerate(constraints, start=1):
        problema += lpSum(x[j] for j in c) >= 1

    problema.solve()

    return LpStatus[problema.status], value(problema.objective), [v.varValue for v in
problema.variables()]

# Perda de caminho em espaço livre (FSPL)
def free_space_path_loss(d:float, f:float, ):
    """
    Calcula a perda de caminho em espaço livre (FSPL).

    Parameters
    -----
    d : float
        Distância entre transmissor e receptor em metros.

```

f : float
Frequência do sinal em Hz.

Returns

float
Perda de caminho em espaço livre em dB.
"""

```
c = 3 * 10**8 # velocidade da luz em metros/segundo
return 20 * np.log10((4 * np.pi * d * f * 10**9) / c)
```

Perda de transmissão básica (BTL)

```
def basic_transmission_loss(d:float, f:float,
                             alpha: float, beta:float, gamma: float):
    """
```

"""

Calcula a perda de transmissão básica (BTL).

Parameters

d : float
Distância entre transmissor e receptor em metros.

f : float
Frequência do sinal em Hz.

alpha : float
Parâmetro alfa do modelo.

beta : float
Parâmetro beta do modelo.

gamma : float
Parâmetro gama do modelo.

Returns

float
Perda de transmissão básica em dB.
"""

```
return 10 * alpha * np.log10(d) + beta + 10 * gamma * np.log10(f)
```

NLoS Monte Carlo

```
def nlos_monte_carlo(d: float, f: float,
                     alpha: float, beta: float, gamma: float, sigma: float,
                     P_tx: float, G_tx: float, G_rx: float, Cable_tx_loss: float, Cable_rx_loss: float,
                     LINHAS: int, COLUNAS: int, condition: str, seed_value=42):
    """
```

Calcula a perda de sinal em ambiente NLoS ou LoS usando o método de Monte Carlo com a fórmula da imagem.

Parameters

```

condition : str
    Condição escolhida ('LoS' ou 'NLoS'). Se 'LoS', o ruído gaussiano não será adicionado.
    (demais parâmetros idênticos...)
"""

np.random.seed(seed_value)

# Perda em espaço livre (L_FS)
c = 3 * 10**8
L_FS = 20 * np.log10((4 * np.pi * d * f * 10**9) / c) # dB

# Perda de transmissão básica (L_b) usando a fórmula da imagem
L_b = 10 * alpha * np.log10(d) + beta + 10 * gamma * np.log10(f)

if condition == 'NLoS':
    gaussian_noise = np.random.normal(0, sigma, size=(int(LINHAS), int(COLUNAS)))
    excess_loss = gaussian_noise #- L_FS + L_b
else:
    mean_excess_loss = 10 * np.log10(P_tx + (G_tx + G_rx - L_FS - Cable_tx_loss -
Cable_rx_loss) / 10)
    # excess_loss = np.random.normal(mean_excess_loss, sigma, size=(int(LINHAS),
int(COLUNAS)))
    excess_loss = np.random.normal((L_b - L_FS), sigma, size=(int(LINHAS),
int(COLUNAS)))
    print(excess_loss)

# Potência recebida (P_rx)
P_rx = P_tx + G_tx + G_rx - excess_loss - L_b - Cable_tx_loss - Cable_rx_loss
return P_rx

```

ANEXO E

CÓDIGO INTERFACE *SITE-GENERAL*

```
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.graph_objects as go
from libs.free_space_sinal_em_mw import nlos_monte_carlo
import streamlit as st

def plot_nlos_monte_carlo(coefficients, environment, condition, d, f, P_tx, G_tx, G_rx,
Cable_tx_loss, Cable_rx_loss, LINHAS, COLUNAS, seed_value=42):
    params = coefficients[environment][condition]
    alpha, beta, gamma, sigma = params['alpha'], params['beta'], params['gamma'],
params['sigma']

    matrix = nlos_monte_carlo(d, f, alpha, beta, gamma, sigma, P_tx, G_tx, G_rx,
Cable_tx_loss, Cable_rx_loss, LINHAS, COLUNAS, condition, seed_value)

    # Se quiser fazer a perda pelo médio e o desvio
    #mean_value = np.mean(matrix)
    #std_value = np.std(matrix)
    #variables_matrix = np.where((matrix >= (mean_value - std_value)) & (matrix <=
(mean_value + std_value)), 1, 0)

    variables_matrix = np.where((matrix >= -72) & (matrix <= -30), 1, 0)

    return matrix, variables_matrix

st.title('Modelo SITE-GENERAL')


coefficients = {
    'Escritorio': {
```

```

        'LoS': {'f_range': (2.4, 2.5), 'd_range': (2, 27), 'alpha': 1.46, 'beta': 34.62, 'gamma': 2.03,
'sigma': 3.76},
        'NLoS': {'f_range': (2.4, 2.5), 'd_range': (4, 30), 'alpha': 2.46, 'beta': 29.53, 'gamma':
2.38, 'sigma': 5.04},
    },
    'Corredor': {
        'LoS': {'f_range': (2.4, 2.5), 'd_range': (2, 160), 'alpha': 1.63, 'beta': 28.12, 'gamma': 2.25,
'sigma': 4.07},
        'NLoS': {'f_range': (5.1, 5.8), 'd_range': (4, 94), 'alpha': 2.77, 'beta': 29.27, 'gamma':
2.48, 'sigma': 7.63},
    },
    'Industrial': {
        'LoS': {'f_range': (5.1, 5.8), 'd_range': (2, 102), 'alpha': 2.34, 'beta': 24.26, 'gamma': 2.06,
'sigma': 2.67},
        'NLoS': {'f_range': (5.1, 5.8), 'd_range': (5, 110), 'alpha': 3.66, 'beta': 22.42, 'gamma':
1.34, 'sigma': 9.00},
    },
    'Sala de conferências/palestras': {
        'LoS': {'f_range': (2.4, 2.5), 'd_range': (2, 21), 'alpha': 1.61, 'beta': 28.82, 'gamma': 2.37,
'sigma': 3.28},
        'NLoS': {'f_range': (5.1, 5.8), 'd_range': (4, 25), 'alpha': 2.07, 'beta': 28.13, 'gamma':
2.67, 'sigma': 3.67},
    }
}

```

```

st.sidebar.header('Modelo de cálculo de perda de sinal'
    ' em ambientes de comunicação sem fio (LoS & NLoS)')
st.sidebar.header("Parâmetros de Entrada )
environment = st.sidebar.selectbox('Escolha o Ambiente', list(coefficients.keys()))
condition = st.sidebar.radio('Escolha a Condição', ['LoS', 'NLoS'])
d = st.sidebar.number_input('Distância (m)', 2, 1000, 27)
f = st.sidebar.number_input('Frequência (GHz)', 2.4, 5.8, 2.5)
P_tx = st.sidebar.number_input('Potência do AP (dBm)', 0.0, 100.0, 28.0)

```

```

G_tx = st.sidebar.number_input('Ganho da Antena TX (dB)', 0.0, 30.0, 1.0)
G_rx = st.sidebar.number_input('Ganho da Antena RX (dB)', 0.0, 30.0, 1.0)
Cable_tx_loss = st.sidebar.number_input('Perdas no Cabo TX (dB)', 0.0, 10.0, 1.0)
Cable_rx_loss = st.sidebar.number_input('Perdas no Cabo RX (dB)', 0.0, 10.0, 1.0)
N = st.sidebar.number_input('Coeficiente de Perda de Potência N', 15.0, 40.0, 20.0)
LINHAS = st.sidebar.number_input('Número de Linhas da Matriz', 1, 12, 4)
COLUNAS = st.sidebar.number_input('Número de Colunas da Matriz', 1, 12, 4)

calc_button = st.sidebar.button('Calcular Alcance')

# Colunas principais para visualização
col1, col2 = st.columns([2, 1])

with col1:
    if calc_button:
        matrix, variables_matrix = plot_nlos_monte_carlo(coefficients, environment, condition,
d, f, P_tx, G_tx, G_rx, Cable_tx_loss, Cable_rx_loss, LINHAS, COLUNAS)
        fig_initial = go.Figure(data=go.Heatmap(
            z=variables_matrix,
            colorscale='Blues',
            zmin=0, zmax=1,
            showscale=True,
            colorbar=dict(title='Equipamento')
        ))

        equipment_idx = np.argwhere(variables_matrix == 1)

        for idx in equipment_idx:
            row, col = idx
            fig_initial.add_trace(go.Scatter(
                x=[col],
                y=[row],
                text=["⊗"],
                mode="text",

```



```
    textfont=dict(color="black", size=18),
    showlegend=False
))
```

```
fig_initial.update_layout(
    title='Locais dos Equipamentos',
    xaxis=dict(tickmode='array', tickvals=list(range(COLUNAS))),
    yaxis=dict(tickmode='array', tickvals=list(range(LINHAS))),
    width=1000, height=800,
    coloraxis=dict(colorbar=dict(outlinewidth=0)))
```

```
st.plotly_chart(fig_initial)
```

with col2:

```
    plot_heatmap = st.checkbox('Mostrar Matriz Interativa')
    if plot_heatmap and calc_button:
        plt.figure(figsize=(8, 6))
        sns.heatmap(matrix, annot=True, fmt=".1f", cmap='Blues', cbar_kws={'label':
'Equipamento'})
        plt.title(f'{condition} - {environment}')
        st.pyplot(plt)
```

ANEXO F

CÓDIGO INTERFACE – PARTE GRÁFICA

```
import streamlit as st
import pandas as pd
import numpy as np
import plotly.express as px
import plotly.graph_objects as go
import seaborn as sns
import matplotlib.pyplot as plt
from free_space_sinal_em_mw import calcular_alcance, resolver_cobertura

st.title('Análise de Configurações de Sinal Wi-Fi 📶')

# Para não ficar reiniciando a página na hora de mudar os plots
if 'alcance' not in st.session_state:
    st.session_state.alcance = None
    st.session_state.status = None
    st.session_state.obj_value = None
    st.session_state.variables = None

# Parâmetros
with st.sidebar:
    st.header("Parâmetros de Entrada ⚙️")
    freq = st.number_input('Frequência (MHz)',
                           min_value=100.0, max_value=10000.0,
                           value=5000.0, step=100.0)
    P_tx = st.number_input('Potência do AP (dBm)',
                           min_value=0.0, max_value=100.0, value=28.0)
```

```

G_tx = st.number_input('Ganho da Antena TX (dB)',
                        min_value=0.0, max_value=30.0, value=1.0)
G_rx = st.number_input('Ganho da Antena RX (dB)',
                        min_value=0.0, max_value=30.0, value=1.0)
Cable_tx_loss = st.number_input('Perdas no Cabo TX (dB)',
                                 min_value=0.0, max_value=10.0, value=1.0)
Cable_rx_loss = st.number_input('Perdas no Cabo RX (dB)',
                                 min_value=0.0, max_value=10.0, value=1.0)
P_lim = st.number_input('Potência Limite (dBm)',
                         min_value=0.0, max_value=100.0, value=0.0)
LINHAS = st.number_input('Número de Linhas Matriz',
                          min_value=1, max_value=12, value=4)
COLUNAS = st.number_input('Número de Colunas Matriz',
                           min_value=1, max_value=12, value=4)
calc_button = st.button('Calcular Alcance')

```

```

if calc_button:

```

```

    alcance = calcular_alcance(LINHAS, COLUNAS,
                               freq, P_tx, G_tx,
                               G_rx, Cable_tx_loss,
                               Cable_rx_loss)
    status, obj_value, variables = resolver_cobertura(LINHAS,
                                                       COLUNAS,
                                                       alcance,
                                                       P_lim)

```

```

    st.session_state.alcance = alcance
    st.session_state.status = status
    st.session_state.obj_value = obj_value
    st.session_state.variables = variables

```

```

if st.session_state.alcance is not None:

```

```

selected_matrix = st.selectbox('Escolha o Cenário:',
                               options=range(len(st.session_state.alcance)),
                               format_func=lambda x: f"Cenário {x+1}")
df_alcance = pd.DataFrame(st.session_state.alcance[selected_matrix])

max_idx = np.argmax(df_alcance.values)
max_row, max_col = divmod(max_idx, COLUNAS)

fig = px.imshow(df_alcance, color_continuous_scale='OrRd',
                labels={'color': 'Equipamento'},
                zmin=-31, zmax=-1,)
fig.update_layout(title='Locais dos Equipamentos',
                  xaxis=dict(tickmode='array', tickvals=list(range(COLUNAS))),
                  yaxis=dict(tickmode='array', tickvals=list(range(LINHAS))),
                  width=1000, height=800)

fig.add_trace(go.Scatter(
    x=[max_col],
    y=[max_row],
    text=["⊗"],
    mode="text",
    textfont=dict(color="black", size=50)
))

num_rings = 5
for i in range(1, num_rings + 1):
    x0 = max(max_col - i, 0)
    y0 = max(max_row - i, 0)
    x1 = min(max_col + i, COLUNAS - 1)
    y1 = min(max_row + i, LINHAS - 1)

    if x0 < COLUNAS and y0 < LINHAS and x1 >= 0 and y1 >= 0:

```

```
fig.add_shape(type="circle",  
              xref="x", yref="y",  
              x0=x0, y0=y0,  
              x1=x1, y1=y1,  
              line=dict(color="blue", dash="dot"))
```

```
st.plotly_chart(fig)
```

```
st.write('Status do Problema:', st.session_state.status)  
st.write('Valor da Função Objetivo:', st.session_state.obj_value)  
st.write('Variáveis do Problema:', st.session_state.variables)
```

```
show_heatmap = st.checkbox('Mostrar Matriz de Alcance - Cenário')
```

```
if show_heatmap:
```

```
    plt.figure(figsize=(10, 8))  
    sns.heatmap(df_alcance, annot=True, cmap='Blues', fmt=".4f")  
    plt.title(f'Matriz de Alcance - Cenário {selected_matrix + 1}')  
    st.pyplot(plt)
```