

UNIVERSIDADE FEDERAL DE UBERLÂNDIA  
INSTITUTO DE GEOGRAFIA, GEOCIÊNCIAS E SAÚDE COLETIVA  
CURSO DE GRADUAÇÃO EM ENGENHARIA DE AGRIMENSURA E  
CARTOGRÁFICA

LINCON RODRIGUES SILVA

**SAFESTRUCT: SOFTWARE LIVRE PARA IDENTIFICAÇÃO DE PONTOS  
INSTÁVEIS NO MONITORAMENTO GEODÉSICO**

Monte Carmelo

2025

LINCON RODRIGUES SILVA

**SAFESTRUCT: SOFTWARE LIVRE PARA IDENTIFICAÇÃO DE PONTOS  
INSTÁVEIS NO MONITORAMENTO GEODÉSICO**

Trabalho de Conclusão de Curso apresentado à  
Universidade Federal de Uberlândia como  
requisito parcial para obtenção do título de  
bacharel em Engenharia de Agrimensura e  
Cartográfica.

Orientador: Prof. Dr. Vinicius Francisco  
Rofatto

Monte Carmelo

2025

LINCON RODRIGUES SILVA

**SAFESTRUCT: SOFTWARE LIVRE PARA IDENTIFICAÇÃO DE PONTOS  
INSTÁVEIS NO MONITORAMENTO GEODÉSICO**

Trabalho de Conclusão de Curso  
apresentado à Universidade Federal de  
Uberlândia como requisito parcial para  
obtenção do título de bacharel em  
Engenharia de Agrimensura e Cartográfica.

Banca Examinadora:

---

Prof. Dr. Vinicius Francisco Rofatto  
Universidade Federal de Uberlândia

---

Me. Jhonatta Willyan Miato Assunção  
Universidade Federal de Uberlândia

---

Prof. Dr. Leticia Cristina Ribeiro  
Universidade Federal de Uberlândia

## **AGRADECIMENTOS**

Agradeço primeiramente a Deus por me conceder saúde, sabedoria e força para concluir esta etapa tão importante da minha vida. Sua presença constante me guiou e iluminou meus passos durante toda a jornada.

Aos meus pais e minha irmã, minha eterna gratidão pelo amor incondicional, apoio e incentivo em todos os momentos.

A minha companheira, Maria Cecília, pela paciência, compreensão e apoio constantes durante toda a realização deste trabalho e da minha formação.

Ao meu orientador, Prof. Dr. Vinicius Francisco Rofatto, expresso minha mais profunda gratidão pela dedicação, paciência e conhecimento compartilhados. Sua orientação foi fundamental para o sucesso deste trabalho.

Ao meu amigo Me. Jhonatta Wilyan Miato Assunção, agradeço pelos conhecimentos compartilhados ao longo da minha formação, que contribuíram de maneira significativa para o meu aprendizado.

À Universidade Federal de Uberlândia, agradeço pela oportunidade de realizar este trabalho e pelo ensino de excelência que recebi.

## RESUMO

Nos últimos anos, o Brasil enfrentou desafios significativos relacionados ao rompimento de barragens, eventos que têm impactos devastadores na vida das pessoas e no meio ambiente, motivando assim o aumento de pesquisas relacionadas ao monitoramento geodésico de estruturas. Desse modo, foi desenvolvido um método inovador que integra análise combinatória e testes de razão de verossimilhança em um procedimento sequencial, o qual foi denominado de *Sequential and Combinatorial Geometry-Free Identification of Unstable Points for Deformation Monitoring* (SeqCup-Free); por meio de experimentos reais e computacionais, o método se mostrou eficiente, com alta taxa de detecção e identificação de pontos instáveis. Porém, tal método desenvolvido não está disponível em uma interface amigável para usuários em potência. Portanto, neste presente trabalho é proposto um software de código aberto e gratuito para a análise de deslocamento em redes geodésicas. Como resultado, espera-se que o software desenvolvido auxilie os profissionais da área de informações espaciais em termos de otimização e controle de qualidade em redes geodésicas para fins de monitoramento de estruturas.

**Palavras-chave:** Geodésia, estatística, controle de qualidade, monitoramento, teste de hipóteses.

## ABSTRACT

In recent years, Brazil has faced significant challenges related to dam failures, events that have devastating impacts on people's lives and the environment, thereby motivating an increase in research related to the geodetic monitoring of structures. Thus, an innovative method was developed that integrates combinatorial analysis and likelihood ratio tests in a sequential procedure, which was named **Sequential and Combinatorial Geometry-Free Identification of Unstable Points for Deformation Monitoring (SeqCup-Free)**. Through real and computational experiments, the method proved to be efficient, with a high detection rate and identification of unstable points. However, this developed method is not available in a user-friendly interface for potential users. Therefore, this work proposes an open-source and free software for displacement analysis in geodetic networks. As a result, the developed software is expected to assist spatial information professionals in optimizing and ensuring quality control in geodetic networks for structural monitoring purposes.

**Keywords:** Geodesy, statistics, quality control, monitoring, hypothesis testing.

## SUMÁRIO

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>INTRODUÇÃO.....</b>  | <b>12</b> |
| <b>2</b> | <b>OBJETIVOS.....</b>   | <b>13</b> |
|          | 2.1 OBJETIVO GERAL .....  | 13        |
|          | 2.2 OBJETIVOS ESPECÍFICOS.....  | 13        |
| <b>3</b> | <b>REVISÃO BIBLIOGRÁFICA .....</b>  | <b>13</b> |
|          | 3.1 <i>Sequential and Combinatorial Geometry-Free Identification of Unstable Points for Geodetic Deformation Monitoring (SeqCup-Free)</i> ..... | 13        |
| <b>4</b> | <b>MATERIAL E MÉTODOS .....</b>   | <b>21</b> |
|          | 4.1 MATERIAL .....  | 21        |
|          | 4.2 MÉTODOS.....  | 22        |
|          | 4.2.1 FRONT-END.....  | 22        |
|          | 4.2.1.1 ESTRUTURA BÁSICA DA <i>MAIN WINDOW</i> .....  | 22        |
|          | 4.2.1.2 PÁGINA INICIAL.....   | 24        |
|          | 4.2.1.3 PÁGINA DE CONFIGURAÇÕES.....  | 25        |
|          | 4.2.1.4 PÁGINA DE OBSERVAÇÕES .....   | 26        |
|          | 4.2.1.5 PÁGINA SOBRE .....  | 28        |
|          | 4.2.1.7 CONVERTER ARQUIVO .ui .....   | 29        |
|          | 4.2.2 BACK-END .....  | 29        |
|          | 4.2.2.1 MANIPULAÇÃO DE EVENTOS.....   | 29        |
|          | 4.2.2.2 ENTRADA DOS DADOS .....   | 31        |
|          | 4.2.2.2.1 ARQUIVO DE OBSERVAÇÕES.....   | 31        |
|          | 4.2.2.2.2 ARQUIVO DE CONFIGURAÇÃO DA REDE .....   | 32        |
|          | 4.2.2.3 PROCESSAMENTO .....   | 34        |
|          | 4.2.2.4 SAÍDA DE DADOS .....  | 39        |
|          | 4.2.3 TRATAMENTO DE ERROS.....  | 41        |
|          | 4.2.3.1 VERIFICAÇÃO DA PÁGINA DE CONFIGURAÇÕES .....  | 42        |
|          | 4.2.3.2 VERIFICAÇÃO DA CONFIGURAÇÃO DA REDE.....  | 43        |
|          | 4.2.3.3 VERIFICAÇÃO DO ARQUIVO DE OBSERVAÇÕES.....  | 43        |
|          | 4.2.4 TESTES E VALIDAÇÃO .....  | 44        |
| <b>5</b> | <b>RESULTADOS E DISCUSSÕES.....</b>   | <b>44</b> |
|          | 5.1 RESULTADOS REFERENTE AO PROCESSAMENTO .....   | 44        |
|          | 5.2 RESULTADOS REFERENTE À USABILIDADE .....  | 46        |
| <b>6</b> | <b>CONCLUSÃO.....</b>   | <b>46</b> |
|          | REFERÊNCIAS.....  | 48        |

## 1 INTRODUÇÃO

Uma rede geodésica consiste em um conjunto de pontos materializados e considerados estáveis na superfície terrestre, cujas posições são estimadas a partir de medições angulares, lineares e/ou por técnicas espaciais de posicionamento por satélites (Klein, 2014). Essas redes são fundamentais para diversas aplicações, como locação de grandes obras de engenharia, zoneamento urbano, cadastro técnico multifinalitário, definição de limites territoriais e monitoramento de áreas naturais e artificiais.

Entre essas aplicações, destaca-se o monitoramento de grandes estruturas, no qual a rede geodésica deve ser projetada de forma a permitir a detecção de possíveis deslocamentos ao longo do tempo. Uma rede mal projetada ou inexistente pode comprometer a segurança e resultar em prejuízos humanos, ambientais e econômicos, como evidenciado no rompimento da Barragem 1 da Mina Córrego do Feijão, em Brumadinho (MG). Robertson et al. (2019), investigou as causas técnicas deste rompimento. Os autores verificaram a existência de pequenas deformações (inferiores a 36 mm/ano) antes do colapso. Entretanto, infelizmente, não foram detectadas pelos sistemas de monitoramento empregados. Isso reforça ainda mais a necessidade de desenvolver novos procedimentos de monitoramento de estruturas.

A importância do monitoramento geodésico de estruturas fica ainda mais evidente com a regulamentação XXX, conforme disposto no Art. 3º, inciso III, da Lei nº 12.334/2010: que estabelece que o monitoramento e o acompanhamento das ações de segurança de barragens constituem objetivos centrais da Política Nacional de Segurança de Barragens (PNSB).. Tal diretriz evidencia a necessidade de instrumentos técnicos confiáveis para avaliação contínua da integridade estrutural das barragens, o que inclui metodologias geodésicas de alta precisão, como as apresentadas em Rofatto et al. (2025).

Diante desse cenário, torna-se essencial aprimorar os métodos de controle de qualidade e detecção de instabilidades em redes geodésicas. Nesse contexto, Rofatto et al. (2025) propuseram o método *Sequential and Combinatorial Geometry-Free Identification of Unstable Points for Deformation Monitoring* (SeqCup-Free), que combina análise combinatória e testes de razão de verossimilhança em um procedimento sequencial, superando limitações dos métodos tradicionais de identificação ponto a ponto.

Com base no potencial desse método, o presente trabalho tem como objetivo desenvolver um software livre que implemente o SeqCup-Free —o SafeStruct— com o intuito de oferecer aos profissionais da área de informações espaciais uma ferramenta acessível, com interface amigável e eficiente para o monitoramento geodésico de estruturas.



## 2 OBJETIVOS

### 2.1 OBJETIVO GERAL

O presente trabalho tem como objetivo geral desenvolver um software de fácil utilização, dotado de interface amigável, destinado à detecção e identificação de pontos instáveis em estruturas monitoradas por meio de redes geodésicas.

### 2.2 OBJETIVOS ESPECÍFICOS

- Implementar no software um método estatístico capaz de realizar testes para detecção e identificação de possíveis pontos instáveis;
- Disponibilizar aos profissionais da área de informação espacial uma ferramenta que automatize etapas do processo de análise, reduzindo o tempo de processamento;
- Garantir que o software desenvolvido seja livre e acessível, favorecendo sua utilização em ambientes acadêmicos, comerciais e operacionais.

## 3 REVISÃO BIBLIOGRÁFICA

### 3.1 *Sequential and Combinatorial Geometry-Free Identification of Unstable Points for Geodetic Deformation Monitoring (SeqCup-Free)*

Um dos principais desafios atuais da análise de dados geodésicos é detectar alterações geométricas de objetos ou áreas sujeitas a deslocamentos e/ou deformações, tanto para estruturas artificiais como barragens ou pontes, bem como estruturas naturais como vulcões ou placas tectônicas. O monitoramento de estruturas pode ser dividido em quatro modelos de deformações: modelo de congruência, modelo cinemático, modelo estático e modelo dinâmico, no qual cada modelo possui seus respectivos conceitos (Rofatto et al. 2025):

O modelo de congruência descreve as deformações por meio de vetores de deslocamento, sem especificar o tempo nem considerar fatores relacionados a forças ou

cargas. Dessa forma, é utilizado para representar deformações de maneira puramente geométrica, sem levar em conta suas causas ou variações temporais.

O modelo cinemático descreve as mudanças geométricas em função das variações ao longo do tempo, como velocidades e acelerações. Esse modelo assume que o objeto está em movimento permanente, porém não aborda as causas físicas responsáveis pelas deformações observadas.

O modelo estático é adequado para investigar a relação entre forças causais e reações geométricas. Nele, as deformações são descritas com base nas propriedades físicas do objeto, como coeficientes de expansão, temperatura e comprimentos, sem considerar explicitamente os aspectos temporais no processo de modelagem.

Por fim, o modelo dinâmico combina elementos dos modelos estático e cinemático, estabelecendo uma relação entre as deformações, os fatores de influência — como forças causais e cargas internas e externas — e as propriedades físicas do objeto.

Entre os diversos modelos de deformações, e considerando seu uso consolidado na literatura (Rofatto et al., 2025), adotou-se o modelo de congruência, que apenas indica se o objeto se moveu ou não. Neste modelo, a estrutura é frequentemente monitorada por uma rede geodésica que é medida em pelo menos duas épocas no tempo, e estas medições de época são então analisadas estatisticamente. O pilar que sustenta tal análise baseia-se firmemente nos fundamentos dos testes de hipóteses estatísticas. O modelo de hipótese nula indica que nenhum deslocamento ocorreu, e este é testado contra uma classe de modelos alternativos, que estipulam diferentes padrões de deslocamentos (Rofatto et al. 2025). No SeqCup-Free a análise do deslocamento é realizada sobre a diferença das observações em diferentes épocas, diferentemente de testes convencionais como o de congruência, utilizado em trabalhos anteriores (Bandeira, A. L.; Klein, I.; Veiga, L. A. K., 2021), no qual a análise é feita na diferença das coordenadas em diferentes épocas.

Como consequência o modelo de equações utilizado para estimar as coordenadas em testes convencionais não é linear, incorrendo assim no problema de linearização das equações, que afeta o poder do teste, compreendido como a capacidade do teste estatístico em aceitar a hipótese alternativa quando de fato ela é verdadeira. O método SeqCup-Free não possui tal deficiência, uma vez que suas equações são todas lineares, desse modo, a análise de deslocamento é livre de parâmetros de Datum. logo não temos

problema de definição de Datum, o que requer em métodos convencionais, como o teste de congruência global.

A equação 1 (Rofatto et al. 2025) corresponde as observações realizadas na época 1, enquanto a equação 2 corresponde às observações realizadas na época 2 (Rofatto et al. 2025).

$$y^{(1)} = y_v + e_v^{(1)} \quad (1)$$

$$y^{(2)} = y_v + e_v^{(2)} \quad (2)$$

No qual:

- $y^{(1)}$  e  $y^{(2)}$  são os vetores das observações em diferentes épocas;
- $y_v$  é o vetor que contém os valores verdadeiro das grandezas medidas, porém desconhecidas;
- $e_v^{(1)}$  e  $e_v^{(2)}$  são os vetores dos erros de medidas desconhecidos.

Ao realizar a diferença das observações em diferentes épocas, ou seja, ao realizar a operação equação 2 – equação 1, resulta na equação 3 (Rofatto et al. 2025).

$$\begin{aligned} y^{(2)} - y^{(1)} &= y_v - y_v + e_v^{(2)} - e_v^{(1)} \\ \Delta_y &= e_{\Delta_y} = e_v^{(2)} - e_v^{(1)} \end{aligned} \quad (3)$$

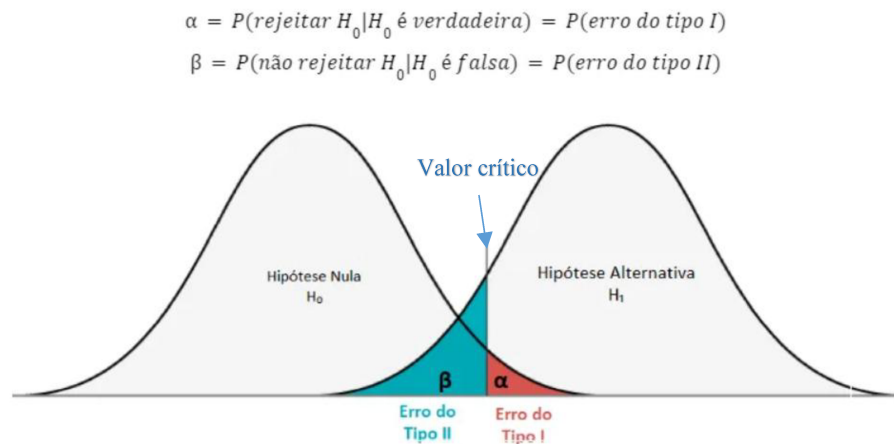
No qual:

- $\Delta_y$  = vetor da diferença das observações;
- $e_{\Delta_y}$  = O vetor desconhecido da diferença dos erros entre as épocas.

É importante destacar que  $\Delta_y = e_{\Delta_y}$ , ou seja, o que se está medindo corresponde à diferença dos erros entre as épocas analisadas.

Como se trata de testes estatísticos, qualquer decisão baseada na amostra está sujeita a erro: ao aceitar ou rejeitar a hipótese nula, pode-se incorrer tanto em erro do Tipo I (rejeitar  $H_0$  quando ela é verdadeira) quanto em erro do Tipo II (não rejeitar  $H_0$  quando ela é falsa). Os níveis de probabilidade de ocorrer o Erro Tipo I e Erro Tipo II são mostrados na Figura 1.

Figura 1 – Níveis de Probabilidade do Erro Tipo I ( $\alpha$ ) e Erro Tipo II ( $\beta$ )



Fonte: Adaptado de Mireia e Ruiz (2022).

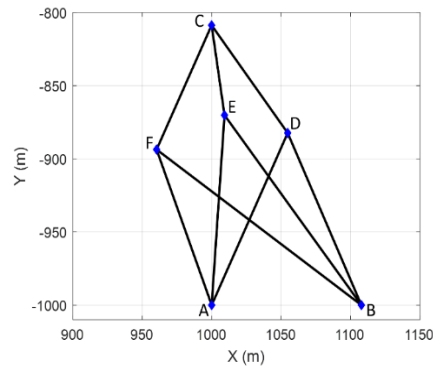
O SeqCup-Free controla com eficiência a taxa de falsos positivos — termo que, em estatística, é conhecido como “Erro Tipo I”, o qual consiste em rejeitar a hipótese nula, quando na verdade ela é verdadeira — em termos de deslocamento: Detectar deslocamento, quando não tem. Esse controle da taxa de falsos positivos deve-se ao fato de o SeqCup-Free calcular o valor crítico por simulação Monte Carlo. Tal simulação computacional utiliza a matriz variância-covariância das observações da época 1 e 2, a partir dela, são geradas 2.000.000 de amostras da distribuição normal multivariada com média zero, para cada vetor de cada época (Equações 1 e 2). Posteriormente, é calculada a estatística de teste para cada uma das 2.000.000 de simulações, gerando uma função densidade de probabilidade, a partir da qual é possível determinar o valor crítico, que depende do nível de significância ( $\alpha$ ). No entanto, diferentemente dos valores críticos tabelados, o valor crítico obtido por simulação de Monte Carlo leva em consideração a correlação entre as observações, permitindo assim controlar de forma mais precisa a taxa de falsos positivos — para os leitores interessados em se aprofundar no método Monte Carlo, recomenda-se consultar a Rofatto, Matsuoka, Klein (2017).

Outro desafio relacionado ao monitoramento geodésico está diretamente associado ao número máximo de pontos a serem considerados no procedimento de teste ( $q_{max}$ ). O SeqCup-Free apresenta uma metodologia inovadora baseada na identificação de um modelo com redundância suficiente para permitir a identificação de pontos instáveis. A determinação de  $q_{max}$  evita o problema da ocorrência de modelos

inseparáveis na identificação de pontos determinantes instáveis (Rofatto et al., 2025). Essa análise é realizada a partir da combinação das matrizes de redundância, conhecida como Matriz R, sendo também analisada a redundância mínima dessas matrizes. O procedimento tem início com  $q_{max} = n_p - 1$  (onde  $n_p$  representa o número de pontos da rede), e esse valor é gradualmente reduzido caso existam matrizes idênticas no conjunto de combinações ou se a redundância mínima entre todas as matrizes for inferior a 0,5. Para os leitores interessados em se aprofundar no cálculo em questão, recomenda-se a consulta a seção 2.4 de **Rofatto et al. (2025)**.

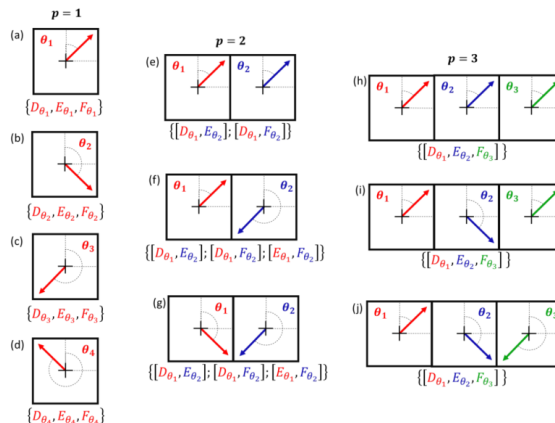
O método foi testado em uma rede real de trilateração (Assunção, 2024) (Figura 2), com diversos padrões de deslocamentos simulados em campo (Figura 3). Os deslocamentos apresentaram uma magnitude constante de 1 cm para todos os casos, com diferentes padrões, variando a direção e/ou sentido, assim como a quantidade de pontos deslocados simultaneamente.

Figura 2 – Rede real de trilateração



Fonte: Assunção (2024).

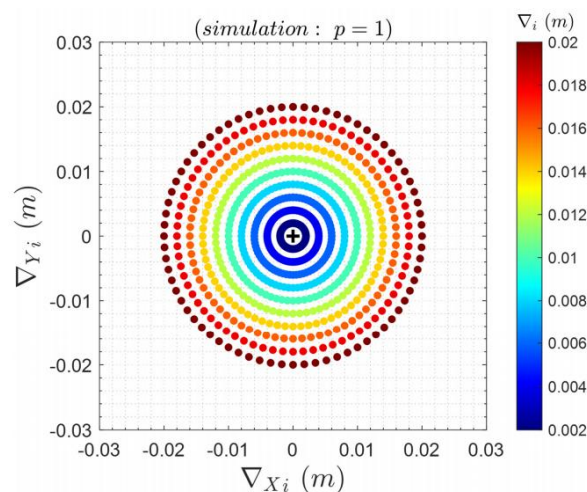
Figura 3 – Padrões de deslocamentos simulado em campo



Fonte: Assunção (2024).

Também foi realizada uma simulação numérica em uma rede de trilateração sintética, utilizando 72 orientações, variando de  $0^\circ$  a  $355^\circ$ , com incremento de  $5^\circ$ , e utilizando 10 magnitudes dentro do intervalo de 1 a 10, com o incremento de 1 desvio padrão da observação (2mm) para cada iteração (Figura 4).

Figura 4 – Deslocamentos simulados para cada ponto (caso  $p = 1$ )



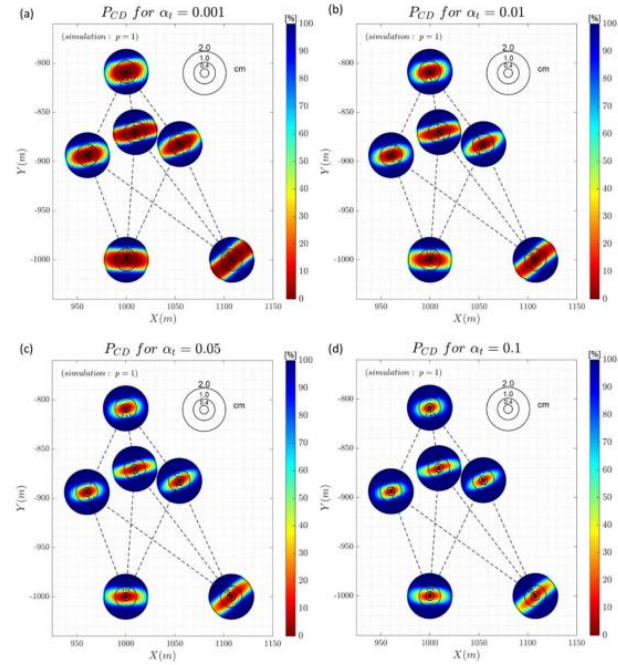
Fonte: Rofatto et al. (2025).

Segundo Rofatto et al. (2025), em geral, os resultados de ambas as redes mostram que o sucesso do método proposto depende da interação entre o padrão de deslocamento e a geometria da rede; deslocamentos no sentido da linha de visada das observações tornam-se mais propícios para uma correta detecção e identificação, por outro lado, deslocamentos perpendiculares à linha de visada das observações tornam-se menos propícios para uma correta detecção e identificação. Outro fator que prejudica a correta detecção e identificação de pontos deslocados está diretamente ligado ao número de pontos deslocados simultaneamente, pois quanto mais pontos deslocados simultaneamente mais complexo fica o modelo .

Na simulação numérica para o caso de um ponto deslocado por vez, observa-se uma maior taxa de detecção (Figura 5 ) e identificação (Figura 6) no sentido da linha de visada e Figura 6). O aumento do nível de significância ( $\alpha$ ) sempre melhora a detecção, como pode ser observado na Figura 5, mas nem sempre melhora a identificação, como pode ser observado na Figura 6. Quando aumenta-se o nível de significância ( $\alpha$ ), a taxa de correta identificação para deslocamentos de pequena

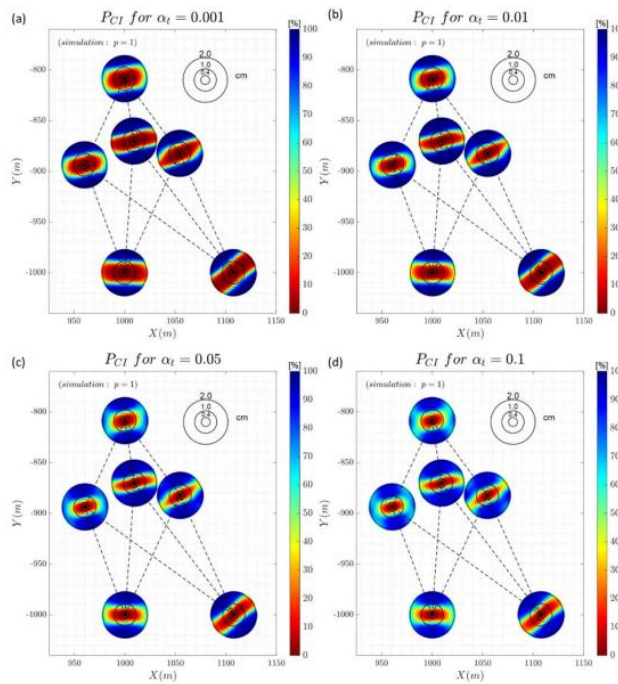
magnitude aumenta; entretanto, a taxa de correta identificação para deslocamentos de grande magnitude diminui.

Figura 5 – Taxa de correta detecção para o caso de 1 ponto deslocado por vez



Fonte: Rofatto et al. (2025).

Figura 6 – Taxa de correta identificação para o caso de 1 ponto deslocado por vez



Fonte: Rofatto et al. (2025).

No caso da rede de trilateração real a taxa de sucesso foi de aproximadamente 83% para correta identificação (localização) de pontos instáveis (Tabela 1 – Frequência de ocorrência para os dados reais da rede de trilateração). A detecção apenas informa se ocorreu ou não deslocamentos, mas não diz a sua localização. É preferível que a detecção ocorra mesmo que a identificação não ocorra da forma esperada. A taxa de correta detecção do método proposto para a rede geodésica utilizada foi de 100%, enquanto a taxa de correta identificação foi de 82,61 %.

Tabela 1 – Frequência de ocorrência para os dados reais da rede de trilateração.

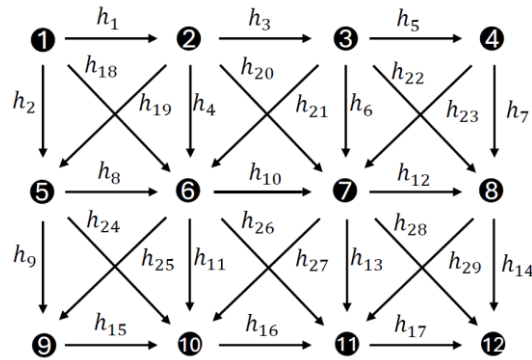
| Padrão de deslocamento | Correta Detecção | Correta Identificação | Errada Identificação | Sub-identificação |
|------------------------|------------------|-----------------------|----------------------|-------------------|
| (a)                    | 3                | 3                     | 0                    | 0                 |
| (b)                    | 3                | 3                     | 0                    | 0                 |
| (c)                    | 3                | 3                     | 0                    | 0                 |
| (d)                    | 3                | 3                     | 0                    | 0                 |
| (e)                    | 2                | 1                     | 0                    | 1                 |
| (f)                    | 3                | 1                     | 0                    | 2                 |
| (g)                    | 3                | 3                     | 0                    | 0                 |
| (h)                    | 1                | 0                     | 1                    | 0                 |
| (i)                    | 1                | 1                     | 0                    | 0                 |
| (j)                    | 1                | 1                     | 0                    | 0                 |
| Total                  | 23               | 19                    | 1                    | 3                 |
| Taxa %                 | 100              | 82,61                 | 4,35                 | 13,04             |

Fonte: Rofatto et al. (2025).

O método também foi testado em uma rede de nivelamento, com 12 pontos e 29 observações (Figura 7). Foram simulados computacionalmente 4 cenários referente a deslocamentos de pontos: um ponto deslocado ( $q_1$ ), dois pontos deslocados ( $q_2$ ), quatro pontos deslocados ( $q_4$ ), seis pontos deslocados ( $q_6$ ). Para cada cenário, realizaram-se 1.000 simulações, variando aleatoriamente a magnitude do deslocamento no intervalo de [2 – 25] mm. Cada simulação foi testada para diferentes níveis de significância:  $\alpha = 0,001$ ,  $\alpha = 0,01$ ,  $\alpha = 0,05$  e  $\alpha = 0,1$ , no qual o valor crítico para cada nível de significância foi calculado por simulação Monte Carlo.



Figura 7 – Rede de nivelamento.



Fonte: Rofatto et al. (2025).

O resultado da aplicação do método SeqCup-Free foram bastantes significativos, como pode ser observado na Tabela 2, no qual como resultado foi aplicado a taxa média de sucesso referente a cada simulação, para cada  $\alpha$ . Segundo Rofatto et al. (2025), baixos valores de  $\alpha$  são mais suscetíveis a identificação de menos pontos deslocados simultaneamente ( $q_1$  e  $q_2$ ), enquanto altos valores de  $\alpha$  são mais suscetíveis a identificação de mais pontos deslocados simultaneamente ( $q_4$  e  $q_6$ ).

Tabela 2 – Taxa média de sucesso para os dados simulados na rede de nivelamento.

| Alpha | Valor Crítico | $q_1$          | $q_2$          | $q_4$          | $q_6$          |
|-------|---------------|----------------|----------------|----------------|----------------|
| 0,001 | 15,4055       | $96,6 \pm 0,6$ | $93,1 \pm 0,8$ | $83,5 \pm 1,2$ | $61,3 \pm 1,6$ |
| 0,01  | 11,1467       | $97,1 \pm 0,5$ | $94,8 \pm 0,7$ | $87,9 \pm 1,0$ | $67,3 \pm 1,3$ |
| 0,05  | 8,1480        | $94,1 \pm 0,8$ | $93,3 \pm 0,7$ | $88,6 \pm 1,0$ | $70,1 \pm 1,4$ |
| 0,1   | 6,8213        | $89,7 \pm 1,0$ | $89,8 \pm 1,0$ | $86,5 \pm 1,1$ | $70,5 \pm 1,1$ |

Fonte: Rofatto et al. (2025).

## 4 MATERIAL E MÉTODOS

### 4.1 MATERIAL

A linguagem de programação escolhida para o desenvolvimento do software foi o Python 3.11.4, por ser de alto nível e oferecer diversas bibliotecas já implementadas, no qual foi utilizada a IDE (*Integrated Development Environment*) Pycharm. Foram utilizadas as seguintes bibliotecas:

- Pandas (Python Data Analysis Library): para a manipulação dos dados de entrada do usuário.
- NumPy (Numerical Python): para o tratamento de vetores e matrizes;
- Itertools (Python Itertools Module): para criar e manipular iteradores, realizando operações como combinações e permutações;
- PyQt5: para o desenvolvimento do *Front-end*.

Além disso, a ferramenta Qt Designer foi empregada na criação do *Front-end*. Trata-se de um framework que possibilita o desenvolvimento visual e interativo de interfaces, em conjunto com a biblioteca PyQt5.

## 4.2 MÉTODOS

O desenvolvimento do software foi dividido em quatro etapas principais: ***Front-end***, ***Back-end***, **Tratamento de erros** e **Testes e validação**.

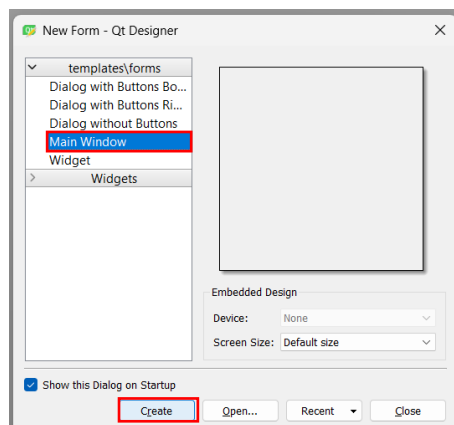
### 4.2.1 FRONT-END

O desenvolvimento do *Front-end*, foi dividido em 7 subetapas: **Estrutura Básica da *Main Window***, **Página inicial**, **Página de configurações**, **Página de observações**, **Página sobre**, **Página contato** e **Converter arquivo .ui**.

#### 4.2.1.1 ESTRUTURA BÁSICA DA MAIN WINDOW

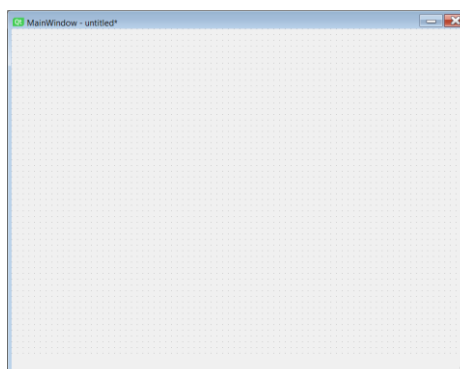
No Qt Designer, foi criada uma ***Main Window*** para dar início ao desenvolvimento do *Front-end* (Figura 8 e Figura 9).

Figura 8 – Criação da *Main Window*.



Fonte: O autor (2025).

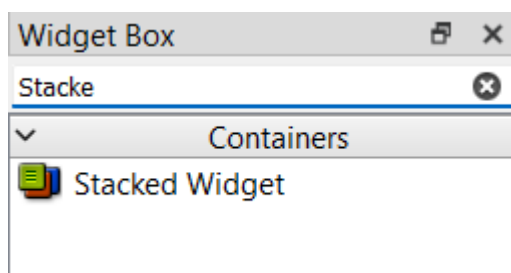
Figura 9 – *Main Window* criada.



Fonte: O autor (2025).

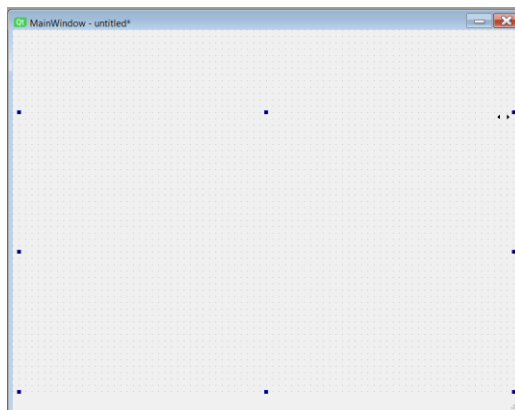
Posteriormente, por meio do *Widget Box*, foi adicionado um *Stacked Widget* com cinco páginas na *Main Window* (Figura 10 e Figura 11): uma página de **Início**, outra de **Configurações**, uma de **Observações**, outra de **Sobre** e, por fim, uma de **Contato**.

Figura 10 – Adicionado um *Stacked Widget*.



Fonte: O autor (2025).

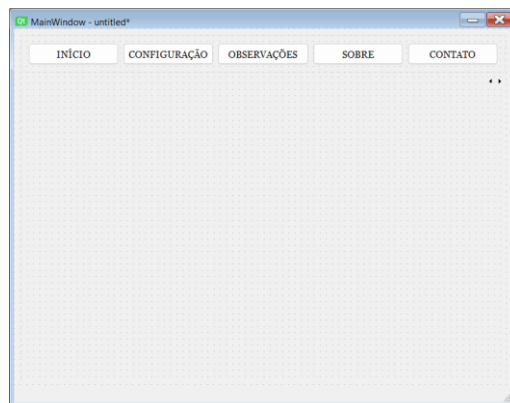
Figura 11 – *Stacked Widget* adicionado a *Main Window* .



Fonte: O autor (2025).

Com as páginas já criadas, por meio do *Widget Box*, foram adicionados à *Main Window* cinco *PushButtons* (Figura 12), que proporcionam acesso a cada uma das páginas.

Figura 12 – Criação dos *PushButtons*.



Fonte: O autor (2025).

Com a estrutura básica da *Main Window* criada, a próxima etapa foi a criação dos elementos correspondentes a cada página.

#### 4.2.1.2 PÁGINA INICIAL

Na **página inicial**, por meio do *Widget Box*, foram adicionados cinco *Labels* (Figura 13), destinados à inserção de imagens e textos. Um deles faz referência ao ícone do software, outro ao nome do software e os três restantes às logos institucionais.

Figura 13 – Página Inicial.



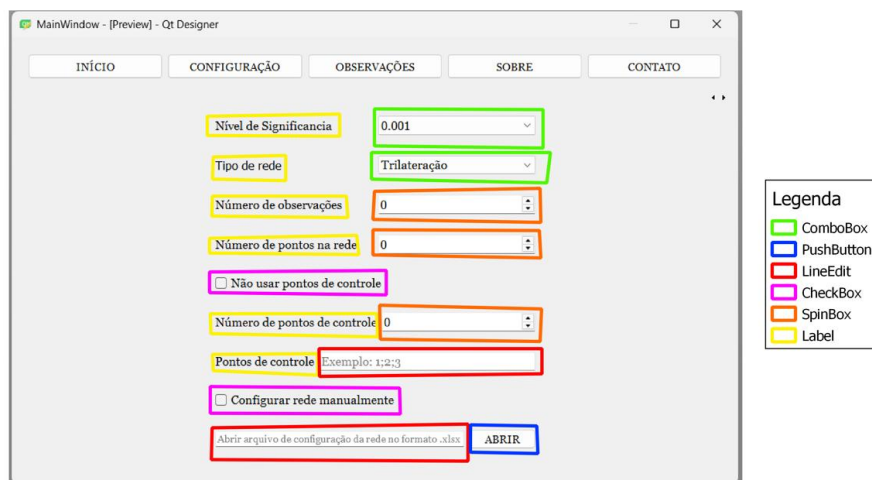
Fonte: O autor (2025).

#### 4.2.1.3 PÁGINA DE CONFIGURAÇÕES

Na página de configurações, por meio do **Widget Box**, foram adicionados os seguintes elementos (Figura 14):

- Seis **Labels**, destinados à inserção de textos;
- Dois **LineEdits**, um para a especificação dos pontos de controle e outro para o caminho do arquivo de configuração da rede;
- Dois **CheckBoxes**, um referente ao uso de pontos de controle e outro à configuração manual da rede;
- Dois **ComboBoxes**, um para a escolha do Nível de Significância e outro para o tipo de rede.
- Três **SpinBoxes**, um para o número de observações da rede, outro para o número de pontos da rede e outro para o número de pontos de controle;
- Um **PushButton**, para abrir o arquivo de configurações da rede.

Figura 14 – Página de Configurações.



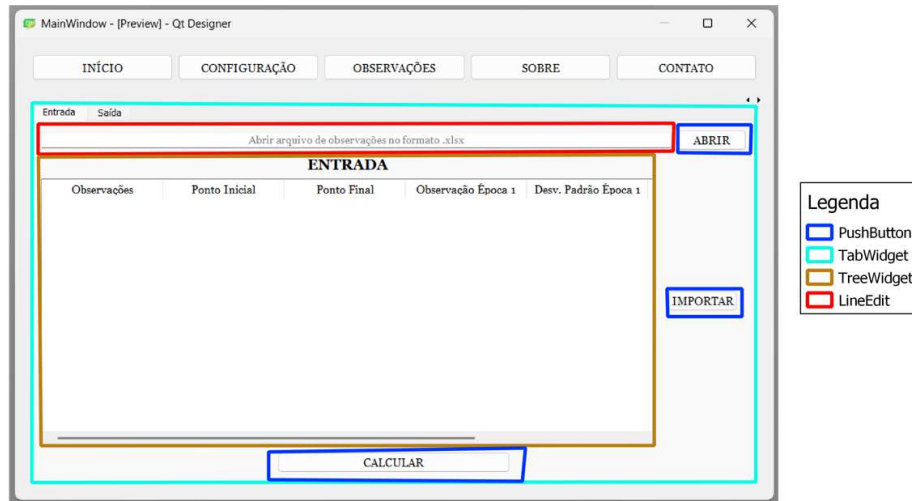
Fonte: O autor (2025).

#### 4.2.1.4 PÁGINA DE OBSERVAÇÕES

Na **página de observações**, por meio do *Widget Box*, foi adicionado uma *TabWidget*, com duas tabelas: a tabela **Entrada** e a tabela **Saída**. Na tabela **Entrada**, por meio do *Widget Box*, foram adicionados os seguintes elementos (Figura 15):

- Uma *TreeWidget*, com sete colunas: **Observações**, **Ponto Inicial**, **Ponto Final**, **Observação Época 1**, **Desv. Época 1**, **Observação Época 2**, **Desv. Época 2**, ;
- Um *LineEdit*, referente ao caminho do arquivo das observações;
- Três *PushButton*, um para abrir o arquivo das observações, outro para importar o arquivo para a *TreeWidget* e outro para calcular possíveis pontos deslocados.

Figura 15 – Página de Observações, tabela entrada.



Fonte: O autor (2025).

Na tabela **Saída**, por meio do *Widget Box*, foram adicionados os seguintes elementos (Figura 16):

- Uma *TreeWidget*, com oito colunas: **Ponto Identificado**, **Deslocamento (m)**, **Desvio Padrão (m)**, **Qmax**, **Nível de Significância (%)**, **Valor Crítico**, **Data**, **Horário**;
- Dois *PushButton*, um para mudar o Alpha do teste estatístico, outro para gerar um relatório final, no formato .xlsx.

Figura 16 – Página de Observações, tabela saída.



Fonte: O autor (2025).

#### 4.2.1.5 PÁGINA SOBRE

Na página sobre, por meio do *Widget Box*, foram adicionados os seguintes elementos (Figura 17):

- Duas *Labels*, destinados à inserção de textos;
- Dois *PushButton*, um para acessar o artigo do método aplicado (Sec - UP), outro para acessar um tutorial de como utilizar o software.

Figura 17 – Página sobre.



Fonte: O autor (2025).

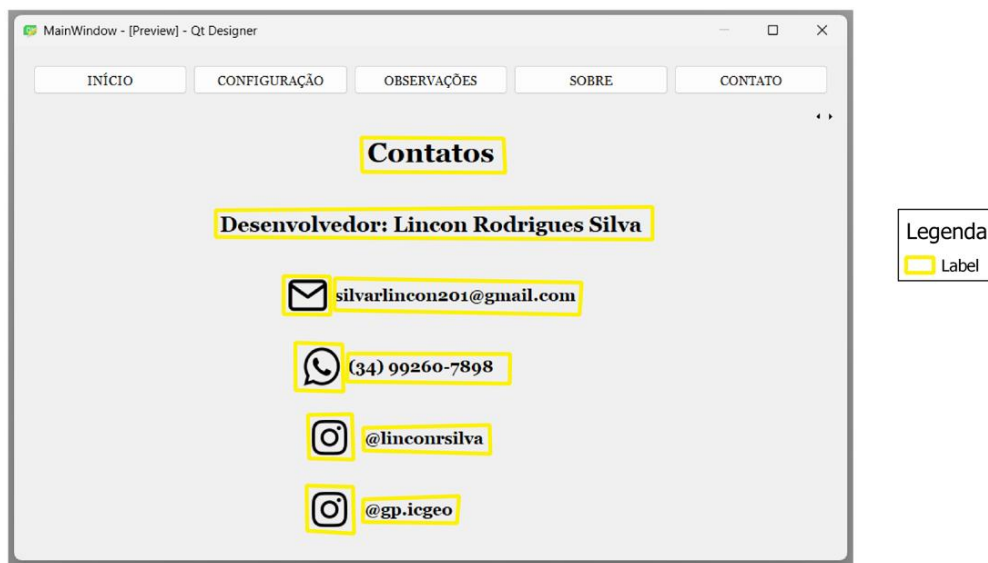
#### 4.2.1.6 PÁGINA CONTATO

Na página contato, por meio do *Widget Box*, foram adicionados os seguintes elementos (Figura 18):

- Dez *Labels*, destinados à inserção de textos e imagens.



Figura 18 – Página sobre.



Fonte: O autor (2025).

#### 4.2.1.7 CONVERTER ARQUIVO .ui

A ferramenta Qt Designer gera um arquivo de interface gráfica no formato .ui. Esse arquivo, utilizando a biblioteca PyQt5 e por meio do comando: “pyuic5 -o nome\_do\_arquivo.py nome\_do\_arquivo.ui”, inserido no terminal do PyCharm, foi convertido para o formato .py, um arquivo de código Python tradicional.

#### 4.2.2 BACK-END

O desenvolvimento do *Back-end*, foi dividido em 4 subetapas: Manipulação de eventos, Entrada de dados, Processamento, Saída de dados.

##### 4.2.2.1 MANIPULAÇÃO DE EVENTOS

A manipulação de eventos resume-se em adicionar interatividade e funcionalidade a uma interface gráfica (*Front-end*). Quando a interface é criada no Qt Designer e convertida para o formato .py, é gerada apenas a classe do objeto com seus atributos, resultando em uma interface sem

funcionalidade. Para adicionar interatividade, é necessário criar sinais que acionem eventos por meio dos métodos da classe.

Para o acesso às páginas **Início**, **Configurações**, **Observações**, **Sobre** e **Contato**, foram criados sinais (Figura 19), que após o clique nos respectivos botões, o usuário é redirecionado para a página correspondente.

Figura 19 – Sinal após o click.

```
# Conectar a pagina
self.btn_home.clicked.connect(lambda: self.pages.setCurrentWidget(self.pg_home))
self.btn_config.clicked.connect(lambda: self.pages.setCurrentWidget(self.pg_config))
self.btn_obs.clicked.connect(lambda: self.pages.setCurrentWidget(self.pg_table))
self.btn_sobre.clicked.connect(lambda: self.pages.setCurrentWidget(self.pg_sobre))
self.btn_contato.clicked.connect(lambda: self.pages.setCurrentWidget(self.pg_contato))
```

Fonte: O autor (2025).

Para capturar os elementos informados pelo usuário na Página de Configurações (Figura 14), foram criados sinais que, após o usuário inserir os dados, acionam os métodos (Figura 20) responsáveis por obter os seguintes elementos:

- Nível de Significância;
- Tipo de rede;
- Número de observações;
- Número de pontos na rede;
- Número de pontos de controle (se tiver pontos de controle);

Figura 20 – Métodos para capturar elementos.

```
# Método pra pegar o número de pontos de controle
1 usage
def ponto_controle(self, i):
    self.pts_contro = i

# Método pra pegar o número de pontos na rede
1 usage
def ponto_rede(self, i):
    self.pts_rede = i

# Método pra pegar o número de observação
1 usage
def observa(self, i):
    self.Nobs = i

# Método pra pegar o nível de significancia
1 usage
def nivel_sig(self, s):
    self.sig_level = float(s) / 100

# Método pra pegar o tipo de rede
1 usage
def tip_rede(self, s):
    self.tipo_rede = s
```

Fonte: O autor (2025).

#### 4.2.2.2 ENTRADA DOS DADOS

A entrada dos dados é realizada por meio de dois arquivos no formato **.csv**, sendo eles: um arquivo contendo as observações das duas épocas, com suas respectivas incertezas, e outro referente às configurações do designer da rede a ser utilizada. Esse arquivo é opcional, ficando a critério do usuário utilizá-lo ou configurar a rede manualmente na **Tabela Entrada**.

##### 4.2.2.2.1 ARQUIVO DE OBSERVAÇÕES

O arquivo de observações deve conter 4 colunas (Figura 21):

- “E1”, referente às observações da época 1;
- “D1”, referente ao desvio padrão de cada observação da época 1;
- “E2”, referente às observações da época 2;

- “D2”, referente ao desvio padrão de cada observação da época 2.

Figura 21 – Arquivo de observações.

|    | A       | B       | C     | D     |
|----|---------|---------|-------|-------|
| 1  | E1      | E2      | D1    | D2    |
| 2  | 129,806 | 129,807 | 0,002 | 0,002 |
| 3  | 130,212 | 130,203 | 0,002 | 0,002 |
| 4  | 113,542 | 113,541 | 0,002 | 0,002 |
| 5  | 129,141 | 129,141 | 0,002 | 0,002 |
| 6  | 163,059 | 163,056 | 0,002 | 0,002 |
| 7  | 181,857 | 181,858 | 0,002 | 0,002 |
| 8  | 91,789  | 91,787  | 0,002 | 0,002 |
| 9  | 62,178  | 62,184  | 0,002 | 0,002 |
| 10 | 93,744  | 93,741  | 0,002 | 0,002 |

Fonte: O autor (2025).

Posteriormente, utilizando a biblioteca Pandas, as informações contidas no Arquivo de Observações (Figura 21) são armazenadas em estruturas de dados, especificamente vetores e matrizes. As colunas “E1” e “E2” representam os vetores das observações referentes às respectivas épocas. Já as colunas “D1” e “D2” têm seus elementos elevados ao quadrado, formando as diagonais principais das matrizes de variância e covariância de cada época,  $\Sigma_y^{\{1\}}$  e  $\Sigma_y^{\{2\}}$ .

#### 4.2.2.2.2 ARQUIVO DE CONFIGURAÇÃO DA REDE

O arquivo de configuração da rede deve conter 3 colunas (Figura 22):

- “Obs”, referente a cada observação;
- “PI”, referente ao ponto inicial da observação;
- “PF”, referente ao ponto final da observação.

Figura 22 – Arquivo de configuração da rede.

|    | A   | B  | C  |
|----|-----|----|----|
| 1  | Obs | PI | PF |
| 2  | 1   | 1  | 4  |
| 3  | 2   | 1  | 5  |
| 4  | 3   | 1  | 6  |
| 5  | 4   | 2  | 4  |
| 6  | 5   | 2  | 5  |
| 7  | 6   | 2  | 6  |
| 8  | 7   | 3  | 4  |
| 9  | 8   | 3  | 5  |
| 10 | 9   | 3  | 6  |

Fonte: O autor (2025).

O usuário também pode configurar manualmente a rede, mantendo marcada a opção “**Configurar rede manualmente**” na página de configurações (Figura 14). Dessa forma, após a importação do arquivo de observações, através do ComboBox, é possível selecionar manualmente os pontos inicial e final correspondentes a cada observação, conforme ilustrado na Figura 23.

Figura 23 – Configuração Manual da Rede.

The screenshot shows the 'SafeStruct' application window. The 'CONFIGURAÇÃO' tab is selected. A table titled 'ENTRADA' is displayed with the following data:

| Observações  | Ponto Inicial | Ponto Final | Observação Época 1 | Desv. Padrão Época 1 |    |
|--------------|---------------|-------------|--------------------|----------------------|----|
| observação 1 | 1             | 1           | 129.806            | 0.002                | 12 |
| observação 2 | 1             | 1           | 130.212            | 0.002                | 13 |
| observação 3 | 5             | 1           | 113.542            | 0.002                | 11 |
| observação 4 | 4             | 1           | 129.141            | 0.002                | 12 |
| observação 5 | 5             | 1           | 163.059            | 0.002                | 16 |
| observação 6 | 1             | 1           | 181.837            | 0.002                | 18 |
| observação 7 | 1             | 1           | 91.789             | 0.002                | 91 |
| observação 8 | 1             | 1           | 62.177749999999996 | 0.002                | 62 |
| observação 9 | 1             | 1           | 93.743500000000001 | 0.002                | 93 |

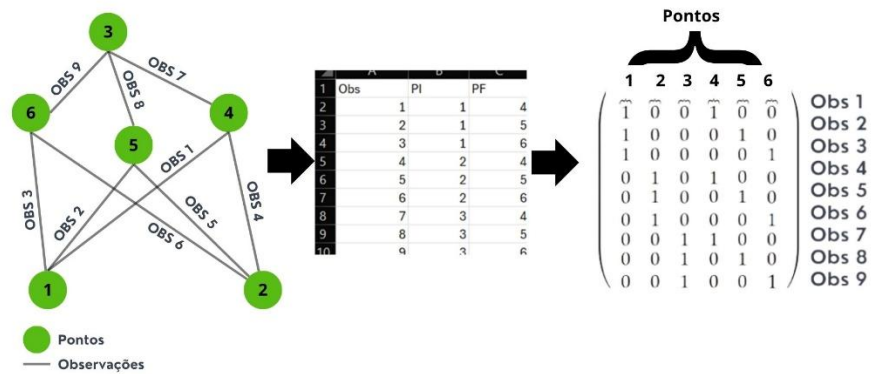
Buttons: INÍCIO, CONFIGURAÇÃO, OBSERVAÇÕES, SOBRE, CONTATO, ABRIR, CALCULAR, IMPORTAR.

Fonte: O autor (2025).

Posteriormente, utilizando a biblioteca Pandas, as informações contidas no arquivo de configuração da rede (Figura 22), ou inseridas manualmente (Figura 23) são armazenadas em estruturas de dados, especificamente em uma matriz. Essa matriz é responsável por conter as conexões das observações, com seus

respectivos pontos, denominada de Matriz A (Figura 24), em que cada linha representa uma observação e cada coluna corresponde a um ponto existente na rede a ser monitorada. Por exemplo, na rede de trilateração apresentada por Assunção 2024, a observação 1 interliga os pontos 1 e 4. Dessa forma, insere-se o número 1 nas posições  $a_{1,1}$  e  $a_{1,4}$  da Matriz A (Figura 24).

Figura 24 – Construção da Matriz de Conexões.



Fonte: Adaptado de Rofatto (2025).

#### 4.2.2.3 PROCESSAMENTO

Com as estruturas de dados criadas, o próximo passo consiste em aplicar o teste **SeqCup-Free**, proposto por **Rofatto et al. 2025**, com o objetivo de detectar e identificar estatisticamente possíveis pontos deslocados com base nas observações de entrada (Figura 21).

Na aplicação do teste SeqCup-Free é necessário calcular a diferença entre os vetores “E2” e “E1”, que correspondem às observações em épocas distintas, resultando no vetor  $\Delta_e$  (vetor dos erros), conforme mostra a equação (3) (Adaptado de Rofatto et al. 2025)

Com o vetor das diferenças das observações (Equação 3) e com a Matriz A (Figura 24), é possível construir a Matriz dos Deslocamentos (Figura 25). Nessa etapa, caso a rede a ser utilizada seja de trilateração, é necessário verificar o sentido do deslocamento, ou seja, se as distâncias

entre cada ponto na época 2 sofreram estiramento e/ou compressão quando comparadas às distâncias da época 1. Essa verificação permite calcular a Matriz C (Figura 25), que representa os deslocamentos considerando o sinal associado a cada um deles, no qual essa matriz contém o padrão de deslocamento referente a cada ponto da rede a ser monitorada. Para esse cálculo, utiliza-se a função  $\text{sgn}$ , cuja função está ilustrada na Equação (4) (Rofatto et al. 2025).

Figura 25 – Matriz C.

$$C = \begin{pmatrix} \overbrace{-1}^1 & \overbrace{0}^2 & \overbrace{0}^3 & \overbrace{-1}^4 & \overbrace{0}^5 & \overbrace{0}^6 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 \end{pmatrix}$$

Fonte: Adaptado de Rofatto (2025).

$$\text{sgn}[\Delta_{e(d)}] := \begin{cases} +1, \text{ se } \frac{\Delta_{e(d)}}{|\Delta_{e(d)}|} > 0, \\ -1, \text{ se } \frac{\Delta_{e(d)}}{|\Delta_{e(d)}|} < 0, \\ 0, \text{ se } \Delta_{e(d)} = 0, \end{cases} \quad \begin{matrix} \text{para } d = 1, \\ \dots, n \end{matrix} \quad (4)$$

A matriz variância – covariância (MVC) da diferença das observações (equação 5) (Rofatto et al. 2025), é obtida aplicando a lei de propagação de variância – covariância, resultado assim na soma da MVC da época 1 com a MVC da época 2.

$$\Sigma_{\Delta_e} = \Sigma_y^{\{1\}} + \Sigma_y^{\{2\}} \quad (5)$$

No qual:

- $\Sigma_{\Delta_e} = \text{MVC da diferença das observações};$
- $\Sigma_y^{\{1\}} = \text{MVC da época 1};$
- $\Sigma_y^{\{2\}} = \text{MVC da época 2}.$

Com a MVC da diferença das observações calculada, a matriz peso será a inversa da MVC da diferença das observações (Equação 6) (Rofatto et al. 2025).

$$W = \theta_o^2 (\Sigma_{\Delta_y})^{-1} \quad (6)$$

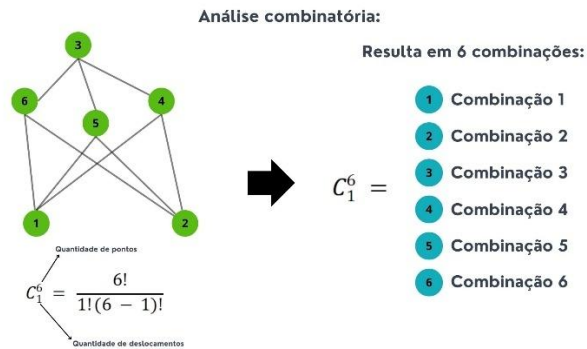
No qual:

- $\Sigma_{\Delta_y} = \text{MVC da diferença das observações};$
- $\theta_o^2 = \text{sigma a priori};$
- $W = \text{matriz peso}.$

São formuladas duas hipóteses: a hipótese nula ( $H_0$ ), de que não houve deslocamento significativo estatisticamente, testada contra uma classe de hipóteses alternativas ( $H_A$ ), através da análise combinatória (Figura), que estipulam diversos padrões de deslocamentos (Figura 27). Quando o modelo da hipótese nula for verdadeiro, o parâmetro  $x$  pode ser estimado pelo cálculo simples pelo Método dos Mínimos Quadrados (MMQ); por outro lado, caso haja incertezas associadas à estabilidade dos pontos, é proposto um modelo alternativo, com parâmetros adicionais que descreve a perturbação nas medições em função do deslocamento de um subconjunto de pontos (Rofatto et al, 2025). A priori, parte-se do pressuposto de que apenas 1 ponto foi deslocado, testando assim todos os modelos para o deslocamento de 1 ponto e gerando a estatística de teste, para cada modelo de deslocamento testado.

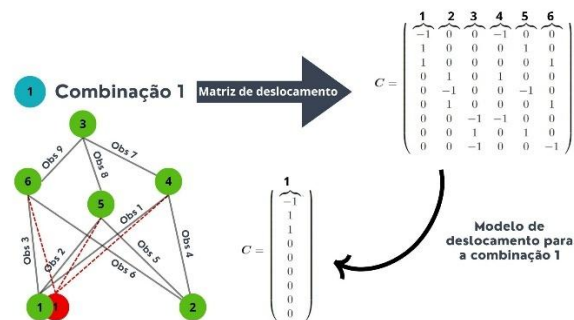


Figura 26 – Combinações.



Fonte: O autor (2025).

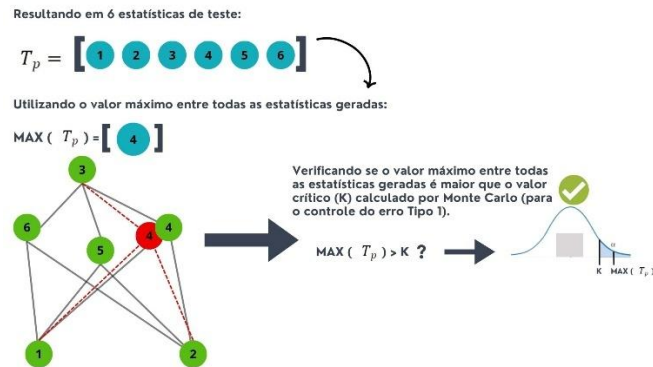
Figura 27 – Padrão de deslocamento.



Fonte: O autor (2025).

Dentre todas as estatísticas de teste calculadas para cada modelo de deslocamento, seleciona-se aquela que representou o valor máximo ( $\max_{T_{p=1}}$ ) dentre todas as analisadas e confronta-se a estatística de teste com o valor crítico que será calculado por simulação Monte Carlo — para os leitores interessados em se aprofundar no método Monte Carlo, recomenda-se consultar a Rofatto, V. F.; Matsuoka, M. T.; Klein, I.(2017) —, com o nível de significância à escolha do usuário. Se a estatística de teste for menor que o valor crítico, a hipótese nula prevalece, indicando que não há nenhum deslocamento significativo a um certo nível de significância. Caso contrário, se a estatística de teste for maior ou igual ao valor crítico, a hipótese alternativa prevalece, e o conjunto de ponto que gerou o  $\max_{T_{q=1}}$  é identificado como deslocado, conforme exemplificado na Figura 28.

Figura 28 – Identificação de máxima estatística.



Fonte: O autor (2025).

As hipóteses são reformuladas, a nova hipótese nula passa a ser que apenas o ponto que gerou o  $\max_{T_{q=1}}$  foi deslocado. Esta hipótese é testada contra uma classe de hipóteses alternativas ( $H_a$ ) em que, a priori, parte-se do pressuposto que apenas 2 pontos foram deslocados, testando assim todos os modelos para o deslocamento de 2 pontos simultâneos e gerando a estatística de teste, para cada modelo de deslocamento testado.

Em sequência é escolhida a estatística de teste que gerou o valor máximo ( $\max_{T_{q=2}}$ ) dentre todas as analisadas e confronta-se a estatística de teste com o valor crítico que será calculado por simulação Monte Carlo com o nível de significância à escolha do usuário. Se a estatística de teste for menor que o valor crítico, a hipótese nula prevalece, indicando que apenas o ponto que gerou  $\max_{T_{q=1}}$  foi deslocado. Caso contrário, se a estatística de teste for maior ou igual ao valor crítico, verifica-se se a hipótese nula é subconjunto da hipótese alternativa, se esta condição for falsa a hipótese nula prevalece, caso contrário o teste continua com a reformulação das hipóteses.

O teste termina quando a hipótese alternativa for negada, ou quando atingir o número máximo de pontos deslocados simultaneamente ( $q_{max}$ ). A definição do número máximo de pontos deslocados simultaneamente ( $q_{max}$ ). O processo de definição de  $q_{max}$  envolve a identificação de um modelo com redundância suficiente para possibilitar a identificação de pontos instáveis. Essa análise é realizada sobre a combinação de matrizes de redundâncias, conhecida como **Matriz R**.

Para os leitores interessados em se aprofundar no cálculo em questão, recomenda-se a consulta a seção 2.4 de **Rofatto et al. (2025)**.

Toda a etapa de processamento está contida no método “calcular”, com exceção do cálculo do valor crítico por Monte Carlo, que está implementado no método “kslrtmo”, e do cálculo de  $q\_max$ , presente no método “num\_max\_qmax”( Figura 29).

Figura 29 – Métodos utilizados no processamento.

```
# Método para realizar o calculo
2 usages
def calcular(self):...

# Método para calcular o número máximo de pontos deslocado
1 usage
def num_max_qmax(self):...

# Método para o cálculo do valor crítico por Monte Carlo
2 usages
def kslrtmo(self):...
```

Fonte: O autor (2025).

#### 4.2.2.4 SAÍDA DE DADOS

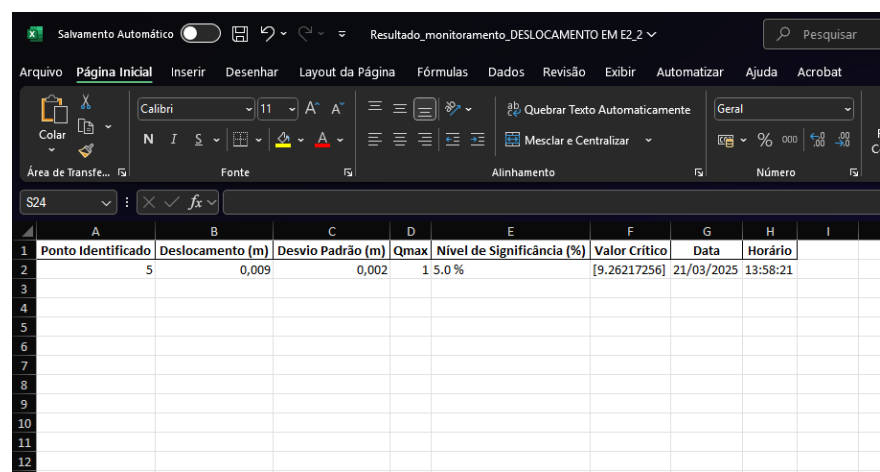
Após a etapa de processamento, na qual foi selecionado o conjunto de pontos que gerou a máxima estatística de teste, é estimado o deslocamento desses pontos, bem como a incerteza associada a esse(s) deslocamento(s). Esses resultados estão apresentados na tabela “saída” (Figura 30), acompanhados de informações relevantes, como: ponto(s) deslocado(s), valor crítico,  $q\_max$ , nível de significância, data e horário do teste. Além disso, é possível gerar um relatório em formato **.xlsx** contendo esses resultados, bastando clicar no botão “**GERAR RELATÓRIO**” (Figura 31).

Figura 30 – Resultado após o processamento.



Fonte: O autor (2025).

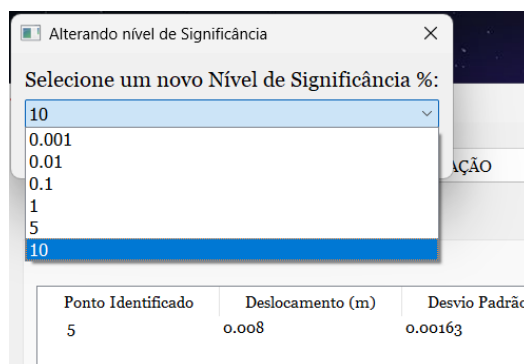
Figura 31 – Relatório no formato .xlsx.



Fonte: O autor (2025).

Também é possível testar outro nível de significância, bastando clicar no botão “MUDAR ALPHA” (Figura 32). Nessa opção, o usuário pode selecionar um novo nível de significância, conforme exemplificado na Figura 32.

Figura 32 – Mudança no valor de alpha.



Fonte: O autor (2025).

### 4.2.3 TRATAMENTO DE ERROS

Na etapa de tratamento de erros, foram desenvolvidos três métodos (Figura 33) com o objetivo de evitar possíveis inconsistências de entrada e garantir o correto funcionamento do software. Essa etapa é subdividida em três subetapas: **verificação da página de configurações**, **verificação da configuração da rede** e **verificação do arquivo de observações**. É importante mencionar que, caso o usuário cometa algum dos erros descritos a seguir, será exibida uma janela *pop-up* (Figura 34) contendo uma mensagem de aviso, indicando o tipo de erro e orientando o usuário sobre como corrigi-lo.

Figura 33 – Métodos para Tratamento de Erros.

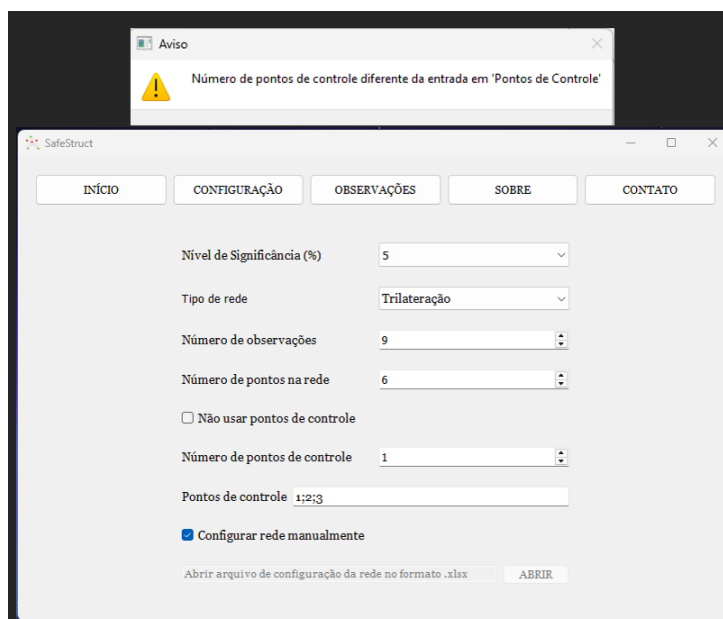
```
# Método para verificar a entrada de pontos iniciais e finais na QTreeWidget
1 usage
> def verifica_pts_iguais(self):...

# Método para verificar pag das configurações
1 usage
> def verifica_configura(self):...

# Método que verifica o arquivo de configuração da rede
1 usage
> def verifica_arq_conf_rede(self):...
```

Fonte: O autor (2025).

Figura 34 – Janela pop-up.



Fonte: O autor (2025).

#### 4.2.3.1 VERIFICAÇÃO DA PÁGINA DE CONFIGURAÇÕES

Para verificar os dados inseridos na página de configurações (Figura 14), foi criado um sinal (Figura 35) que, após o clique no botão “**Observações**”, aciona o método “**verifica\_configura**”, o qual realiza as seguintes verificações:

- Se os “**Pontos de controle**” estão formatados corretamente (por exemplo: 1;2;3);
- Se o “**Número de pontos de controle**”, está condizente com a entrada em “**Pontos de controle**” (por exemplo: Número de pontos de controle = 3; Pontos de controle = 1;2;3);
- Se o ponto escolhido como de controle existe dentro do “**Número de pontos na rede**” (por exemplo: Número de pontos na rede = 6; Pontos de controle = 1;2;3);
- Se os dados contidos no “**Arquivo de configuração da rede**”, estão condizentes com o “**Número de observações**” e com o “**Número de pontos na rede**” (apenas se o usuário **não** for configurar a rede manualmente).

Figura 35 – Sinal para verificar as configurações.

```
# Métodos para verificação
self.btn_obs.clicked.connect(self.verifica_configura)
```

Fonte: O autor (2025).

#### 4.2.3.2 VERIFICAÇÃO DA CONFIGURAÇÃO DA REDE

Para verificar os dados inseridos no arquivo de configuração da rede — apenas quando o usuário **não** opta por configurar a rede manualmente —, foi criado o método “**verifica\_arq\_conf\_rede**”, que é acionado após a etapa de verificação da página de configurações. Esse método é responsável por verificar se o arquivo de entrada está corretamente formatado e se contém as colunas das observações (“**Obs**”) e as colunas referentes aos pontos inicial e final de cada observação (“**PI**” e “**PF**”).

Com o preenchimento da tabela de entrada (Figura 15), seja por meio do arquivo de configuração da rede (Figura 22) ou pela configuração manual (Figura 23), é acionado o método “**verifica\_pts\_iguais**”. Esse método tem como objetivo identificar se existe alguma observação que contenha o mesmo ponto inicial e final.

#### 4.2.3.3 VERIFICAÇÃO DO ARQUIVO DE OBSERVAÇÕES

Em relação ao arquivo de observações (Figura 21), a verificação da formatação correta e dos dados nele contidos não é realizada por um método separado. Essa verificação é executada pelo método “**entrada\_tabela**” (Figura 36). Antes de preencher a tabela de entrada (Figura 15), esse método executa as seguintes verificações:

- Se existem as colunas “E1”, “D1”, “E2” e “D2”, correspondentes às observações da época 1, ao desvio padrão da época 1, às observações da época 2 e ao desvio padrão da época 2;

- Se o tamanho das colunas “E1”, “D1”, “E2” e “D2” é igual;
- Se o número de observações configurado na página de configurações (Figura 14) corresponde ao número de observações no arquivo de observações (Figura 21).

Figura 36 – Método entrada tabela.

```
# Método para alimentar a tabela entrada
1 usage
def entrada_tabela(self):...
```

Fonte: O autor (2025).

#### 4.2.4 TESTES E VALIDAÇÃO

Para verificar se o programa está aplicando corretamente o método SeqCup-Free, foram utilizados 23 dados provenientes de uma rede real de trilatação (Figura 2) (Assunção, 2024) e 4.000 dados simulados da rede de nivelamento (Figura 7), os mesmos dados utilizados no trabalho de Rofatto et al. (2025).

## 5 RESULTADOS E DISCUSSÕES

### 5.1 RESULTADOS REFERENTE AO PROCESSAMENTO

Nesta seção são apresentados e discutidos os resultados obtidos a partir da aplicação do método SeqCup-Free aos conjuntos de dados reais e simulados. O objetivo é comparar os resultados provenientes do software SafeStruct e avaliar se estes estão condizentes com aqueles apresentados por Rofatto et al. (2025), tanto para os dados da rede de trilatação (Tabela 1 – Frequência de ocorrência para os dados reais da rede de trilatação) quanto para os dados da rede de nivelamento (Tabela 2 – Taxa média de sucesso para os dados simulados na rede de nivelamento).



Tabela 3 – Comparação dos resultado na rede de trilateração.

| Padrão de deslocamento | Correta Detecção |         | Correta Identificação |         | Errada Identificação |         | Sub-identificação |         |
|------------------------|------------------|---------|-----------------------|---------|----------------------|---------|-------------------|---------|
|                        | SafeStruct       | Rofatto | SafeStruct            | Rofatto | SafeStruct           | Rofatto | SafeStruct        | Rofatto |
| (a)                    | 3                | 3       | 3                     | 3       | 0                    | 0       | 0                 | 0       |
| (b)                    | 3                | 3       | 3                     | 3       | 0                    | 0       | 0                 | 0       |
| (c)                    | 3                | 3       | 3                     | 3       | 0                    | 0       | 0                 | 0       |
| (d)                    | 3                | 3       | 3                     | 3       | 0                    | 0       | 0                 | 0       |
| (e)                    | 2                | 2       | 1                     | 1       | 0                    | 0       | 1                 | 1       |
| (f)                    | 3                | 3       | 1                     | 1       | 0                    | 0       | 2                 | 2       |
| (g)                    | 3                | 3       | 3                     | 3       | 0                    | 0       | 0                 | 0       |
| (h)                    | 1                | 1       | 0                     | 0       | 1                    | 1       | 0                 | 0       |
| (i)                    | 1                | 1       | 1                     | 1       | 0                    | 0       | 0                 | 0       |
| (j)                    | 1                | 1       | 1                     | 1       | 0                    | 0       | 0                 | 0       |
| Total                  | 23               | 23      | 19                    | 19      | 1                    | 1       | 3                 | 3       |
| Taxa %                 | 100              | 100     | 82,61                 | 82,61   | 4,35                 | 4,35    | 13,04             | 13,04   |

Fonte: O autor (2025).

Tabela 4 – Comparação dos resultado na rede de nivelamento.

| Alpha | Valor Crítico |         | $q_1$      |                | $q_2$      |                | $q_4$      |                | $q_6$      |                |
|-------|---------------|---------|------------|----------------|------------|----------------|------------|----------------|------------|----------------|
|       | SafeStruct    | Rofatto | SafeStruct | Rofatto        | SafeStruct | Rofatto        | SafeStruct | Rofatto        | SafeStruct | Rofatto        |
| 0,001 | 15,4055       | 15,4055 | 96,9       | $96,6 \pm 0,6$ | 93,2       | $93,1 \pm 0,8$ | 84,7       | $83,5 \pm 1,2$ | 63,1       | $61,3 \pm 1,6$ |
| 0,01  | 11,1467       | 11,1467 | 97,2       | $97,1 \pm 0,5$ | 95,2       | $94,8 \pm 0,7$ | 89,7       | $87,9 \pm 1,0$ | 68,6       | $67,3 \pm 1,3$ |
| 0,05  | 8,148         | 8,1480  | 94,5       | $94,1 \pm 0,8$ | 92,5       | $93,3 \pm 0,7$ | 89,4       | $88,6 \pm 1,0$ | 71,2       | $70,1 \pm 1,4$ |
| 0,1   | 6,8213        | 6,8213  | 89,5       | $89,7 \pm 1,0$ | 88,9       | $89,8 \pm 1,0$ | 86,8       | $86,5 \pm 1,1$ | 71,4       | $70,5 \pm 1,1$ |

Fonte: O autor (2025).

Em ambos os cenários, tanto os resultados da rede de trilateração (Tabela 3 – Comparação dos resultados na rede de trilateração) quanto os resultados da rede de nivelamento (Tabela 4 – Comparação dos resultados na rede de nivelamento) foram condizentes com aqueles apresentados por Rofatto et al. (2025), apresentando comportamento semelhante. Para a rede de trilateração, a taxa de correta identificação depende da interação entre o padrão de deslocamento e a geometria da rede, conforme também observado por Rofatto et al. (2025). Já para a rede de nivelamento, baixos valores de  $\alpha$  são mais suscetíveis à identificação de um menor número de pontos deslocados simultaneamente ( $q_1$  e  $q_2$ ), enquanto altos

valores de  $\alpha$  são mais suscetíveis à identificação de um maior número de pontos deslocados simultaneamente ( $q_4$  e  $q_6$ ), tal como afirma Rofatto et al. (2025). Por fim, em ambos os casos, a taxa de correta detecção e identificação tende a diminuir à medida que novos parâmetros (deslocamentos) são inseridos para estimativa no modelo matemático.

## 5.2 RESULTADOS REFERENTE À USABILIDADE

O software SafeStruct apresenta uma interface amigável, conforme demonstrado na seção *Front-end*, com instruções simples e objetivas (Figura 14). Além disso, o usuário conta com uma página de ajuda (Figura 17), que contém uma breve descrição do software.

Nessa mesma página, há dois botões adicionais: o botão “Artigo SeqCup”, que redireciona o usuário para o artigo utilizado como embasamento teórico do SafeStruck, e o botão “Como utilizar o software?”, que direciona para um tutorial passo a passo, explicando como montar os arquivos de configuração da rede e de observações, bem como configurar corretamente a página de configurações (Figura 14).

Observou-se, contudo, que a etapa de cálculo do valor crítico, realizada por meio da simulação de Monte Carlo, demanda considerável poder computacional, apresentando, um tempo de execução relativamente elevado, o qual pode variar conforme a capacidade da máquina utilizada. Essa limitação está associada à natureza do método, que requer a geração de 2.000.000 de amostras para assegurar a precisão estatística do valor crítico — valor de grande importância no método SeqCup-Free.

## 6 CONCLUSÃO

O trabalho em questão cumpriu os objetivos propostos, fornecendo aos profissionais da área de informações espaciais um software de fácil utilização e com interface amigável, capaz de realizar a detecção e identificação de pontos instáveis em estruturas monitoradas. O desenvolvimento do SafeStruck contribui para tornar mais acessível a

aplicação do método SeqCup-Free, automatizando etapas do processo e ampliando seu potencial de uso em ambientes comerciais e operacionais.

Como proposta de aprimoramento futuro, sugere-se a otimização do tempo de execução da etapa de cálculo do valor crítico — realizada via simulação de Monte Carlo — por meio da implementação da biblioteca Numba. Essa biblioteca permite a compilação Just-In-Time (JIT) de funções Python, proporcionando ganhos significativos de desempenho computacional, especialmente em rotinas iterativas e de alto custo computacional, como as envolvidas neste método.

Além disso, outra alternativa é a utilização da rede neural, apresentada por Rofatto et al. (2021) para o cálculo do valor crítico, o que pode representar uma alternativa promissora para reduzir o tempo de processamento.

O software **SafeStruck** está disponível publicamente através do google drive, permitindo o acesso, utilização e aprimoramento por outros pesquisadores e profissionais da área. O código-fonte, bem como o manual de utilização, podem ser acessados por meio do seguinte link: [https://drive.google.com/drive/folders/1Xpjpq44-SelghsRz\\_oV1SZ0\\_MT4-d7tm?usp=drive\\_link](https://drive.google.com/drive/folders/1Xpjpq44-SelghsRz_oV1SZ0_MT4-d7tm?usp=drive_link)

## REFERÊNCIAS

ASSUNÇÃO, Jhonatta Willyan Miato. **Aplicação do teste sequencial de razão de verossimilhança para identificação de pontos instáveis em uma rede de trilateração**. 2024. 42 f. Dissertação (Mestrado em Agricultura e Informações Geoespaciais) - Universidade Federal de Uberlândia, Monte Carmelo, 2024. DOI <http://doi.org/10.14393/ufu.di.2024.62>. Disponível em: <https://repositorio.ufu.br/handle/123456789/41127>. Acesso em: 29 out. 2025.

BANDEIRA, A. L.; KLEIN, I.; VEIGA, L. A. K. **O Papel das Covariâncias na Análise de Deformação Aplicada ao Monitoramento Geodésico de Estruturas**. Revista Brasileira de Cartografia, [S. l.], v. 73, n. 3, p. 723–735, 2021. DOI: 10.14393/rbcv73n3-57873. Disponível em: <https://seer.ufu.br/index.php/revistabrasileiracartografia/article/view/57873>. Acesso em: 30 jan. 2024.

BRASIL. **Lei nº12.334, de 20 de setembro de 2010**. Política Nacional de Segurança de Barragens destinadas à acumulação de água para quaisquer usos [...]. Brasília, DF: Presidência da República, 2010. Disponível em: [https://www.planalto.gov.br/ccivil\\_03/\\_ato2007-2010/2010/lei/l12334.htm](https://www.planalto.gov.br/ccivil_03/_ato2007-2010/2010/lei/l12334.htm). Acesso em: 29 out. 2025.

KLEIN, I. **Proposta de um novo método para o planejamento de redes geodésicas**. 2014. Dissertação (Doutorado em Sensoriamento Remoto) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2014. Disponível em: <https://lume.ufrgs.br/handle/10183/115554>. Acesso em: 30 jan. 2024.

MIREIA, I. **Teste de Hipóteses**: o que significa, quais são os tipos e como realizar um teste de hipóteses. Disponível em: <https://medium.com/turing-talks/teste-de-hip%C3%B3teses-o-que-significa-quais-s%C3%A3o-os-tipos-e-como-realizar-um-teste-de-hip%C3%B3teses-1decda841bcd>.

Robertson, P. K. et al. **Report of the Expert Panel on the Technical Causes of the Failure of Feijão Dam I**. [S. l.], 2019. Portal: B1technicalinvestigation. Disponível em: <http://www.b1technicalinvestigation.com/>. Acesso em: 30 jan. 2024.

Rofatto, V. F. et al. SeqCup-Free: Sequential and Combinatorial Geometry-Free Identification of Unstable Points for Geodetic Deformation Monitoring. SciELO Preprints, 2025. DOI: 10.1590/SciELOPreprints.11415. Disponível em: <https://preprints.scielo.org/index.php/scielo/preprint/view/11415>. Acesso em: 29 out. 2025.

Rofatto, V. F.; Matsuoka, M. T.; Klein, I. **MÉTODO MONTE CARLO APLICADO À ANÁLISE DE CONFIABILIDADE EM GEODÉSIA**. In: XXVII Congresso Brasileiro de Cartografia e XXVI Expositcarta. Rio de Janeiro – RJ, 6 a 9 de novembro de 2017. Disponível em: <https://www.cartografia.org.br/cbc/2017/trabalhos/2/201.html>

Rofatto, VF, Matsuoka, MT, Klein, I., Bonimani, MLS, Rodrigues, BP, de Campos, CC, ... da Silveira Jr., LG (2021). **Valores críticos baseados em redes neurais artificiais para testes de hipóteses múltiplas: caso de espionagem de dados**. *Survey Review* , 54 (386), 440–455. <https://doi.org/10.1080/00396265.2021.1968176> Disponível em: <https://www.tandfonline.com/doi/full/10.1080/00396265.2021.1968176>. Acesso em: 13 nov. 2025.

Souza, L. **Brumadinho**: combinação entre deformações causou rompimento da barragem. São Paulo, 12 dez. 2019. Portal: Agência Brasil. Disponível em: <https://agenciabrasil.ebc.com.br/geral/noticia/2019-12/brumadinho-combinacao-entre-deformacoes-causou-rompimento-da-barragem>. Acesso em: 30 jan. 2024.