

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE COMPUTAÇÃO  
CURSO DE ENGENHARIA DE COMPUTAÇÃO

ANGELO SPINARDI NETTO

**AVALIAÇÃO DA LOCALIZAÇÃO BASEADA NO ROUND-TRIP  
TIME OF FLIGHT PARA DIFERENTES MODELOS DE  
MOBILIDADE EM REDES WI-FI**

TRABALHO DE CONCLUSÃO DE CURSO

CORNÉLIO PROCÓPIO  
2022

ANGELO SPINARDI NETTO

**AVALIAÇÃO DA LOCALIZAÇÃO BASEADA NO ROUND-TRIP  
TIME OF FLIGHT PARA DIFERENTES MODELOS DE  
MOBILIDADE EM REDES WI-FI**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Bacharel em Engenharia de Computação.

Orientadora: Natássya Barlate Floro da Silva  
Universidade Tecnológica Federal do Paraná

CORNÉLIO PROCÓPIO  
2022



4.0 Internacional

Esta licença permite remixe, adaptação e criação a partir do trabalho para fins não comerciais, desde que sejam atribuídos créditos ao(s) autor(es) e que licenciem as novas criações sob termos idênticos. Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.



Ministério da Educação  
**Universidade Tecnológica Federal do Paraná**  
Câmpus Cornélio Procópio  
Nome da Diretoria  
Nome da Coordenação  
Nome do Curso



---

## **TERMO DE APROVAÇÃO**

### **AVALIAÇÃO DA LOCALIZAÇÃO BASEADA NO *ROUND-TRIP* *TIME OF FLIGHT* PARA DIFERENTES MODELOS DE MOBILIDADE EM REDES WI-FI**

**por**

**Angelo Spinardi Netto**

Este Trabalho de Conclusão de Curso de graduação foi julgado adequado para obtenção do Título de “Bacharel em Engenharia de Computação” e aprovado em sua forma final pelo Programa de Graduação em Engenharia de Computação da Universidade Tecnológica Federal do Paraná.

Cornélio Procópio, 23/09/2022

Prof. Doutora Natássya B. F. Da Silva

Prof. Doutor Henrique Yoshikazu Shishido

Prof. Doutor Lucas Dias Hiera Sampaio

Este trabalho é todo dedicado aos meus pais,  
pois me deram a oportunidade de cursar este  
curso.

## **AGRADECIMENTOS**

Em primeiro lugar, a Deus, que fez com que meus objetivos fossem alcançados, durante todos os meus anos de estudos.

Aos meus pais e irmã, que me incentivaram nos momentos difíceis e compreenderam a minha ausência enquanto eu me dedicava à realização deste trabalho.

Aos amigos e namorada, que sempre estiveram ao meu lado, pela amizade incondicional e pelo apoio demonstrado ao longo de todo o período de tempo em que me dediquei a este trabalho.

À professora Natássya, por ter sido minha orientadora e ter desempenhado tal função com dedicação e amizade.

À instituição de ensino Universidade Tecnológica Federal do Paraná, essencial no meu processo de formação profissional, pela dedicação, e por tudo o que aprendi ao longo dos anos do curso.

A todos aqueles que contribuíram, de alguma forma, para a realização deste trabalho.

*Eu denomino meu campo de Gestão do Conhecimento, mas você não pode gerenciar conhecimento. Ninguém pode. O que pode fazer - o que a empresa pode fazer - é gerenciar o ambiente que otimize o conhecimento. (PRUSAK, Laurence, 1997).*

## RESUMO

NETTO, Angelo Spinardi. Avaliação da localização baseada no *Round-Trip Time of Flight* para diferentes modelos de mobilidade em redes Wi-Fi. 2022. 36 f. Trabalho de Conclusão de Curso – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2022.

Devido ao atual avanço da tecnologia, evidencia-se a necessidade da utilização dos sistemas de localização para que as inovações como os veículos autônomos possam desempenhar seu papel de forma mais eficiente. Em razão disso, existem diferentes métodos para localizar um dispositivo, sendo que o mais comum é o GPS (Global Positioning System). Porém, existem algumas limitações que são notadas principalmente em ambientes complexos ou quando o objeto a ser localizado encontra-se em movimento, para esses casos é necessário que seja utilizada uma forma alternativa. Dessa maneira, destaca-se o método RToF (*Round-Trip Time of Flight*) que se baseia no tempo de partida e chegada das mensagens para estimar a posição, sendo o intuito deste trabalho a análise de seu desempenho para diferentes modelos de mobilidade conforme são variadas a velocidade do movimento no nó emissor e a quantidade de nós receptores. Essas avaliações foram realizadas por meio do simulador OMNeT++ com a criação da rede WLAN (Wireless Local Area Network) e configuração dos diferentes tipos de cenários que foram utilizados para realização dos experimentos. Além disso, foi utilizado a técnica *Maximum Likelihood Estimation* (MLE) para o cálculo da trilateração, de forma a obter uma estimativa da posição do nó emissor. Portanto, os resultados da avaliação do algoritmo RToF em conjunto com o MLE em ambiente de simulação demonstram que o tipo de movimento que o nó emissor realiza não interfere na estimativa de posição, diferentemente da velocidade e quantidade de nós que impactaram nitidamente na estimativa de posição do nó emissor.

**Palavras-chave:** Localização, RToF, Mobilidade.

## ABSTRACT

NETTO, Angelo Spinardi. Evaluation of localization based on Round-Trip Time of Flight for different mobility models in Wi-Fi networks. 2022. 36 f. Trabalho de Conclusão de Curso – Curso de Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Cornélio Procópio, 2022.

Due to the current advancement of technology, it is evident the need for localization systems so that innovations such as autonomous vehicles can perform their tasks more efficiently. Consequently, there are several methods to locate a device, and the most common is the GPS. However, there are some limitations that are noticed mainly in complex environments or when the object to be located is in motion, for these cases it is necessary to use an alternative way. Thus, RToF (Round-Trip Time of Flight) method can be highlighted, which is based on the time of departure and arrival of messages to estimate the position, being the purpose of this work to analyze its performance for different mobility models as the speed of movement varies for the sender node and with variation of the number of receiver nodes. These evaluations were performed through the OMNeT++ simulator where is possible to create WLAN networks and to configure the various types of scenarios that were used to perform the experiments. In addition, the Maximum Likelihood Estimation technique was used to calculate the trilateration. Therefore, the results of the evaluation of the RToF algorithm in conjunction with MLE in a simulation environment show that the type of movement that the sender node performs does not interfere with the position estimation, unlike the speed and amount of nodes that clearly impacted the position estimation of the sender node.

**Keywords:** Location, RToF, mobility.



## LISTA DE FIGURAS

Figura 1 – Comunicação entre o nó emissor e o nó âncora com medidas de tempo para as medições do RToF. . . . .	6
Figura 2 – Intersecção dos círculos para cálculo da trilateração. . . . .	6
Figura 3 – Movimento linear do nó. . . . .	8
Figura 4 – Movimento circular do nó. . . . .	9
Figura 5 – Movimento aleatório do nó. . . . .	9
Figura 6 – Trajeto linear do nó emissor. . . . .	15
Figura 7 – Trajeto circular do nó emissor. . . . .	15
Figura 8 – Comparação do erro no cenário utilizando movimento linear sem ruído. . .	21
Figura 9 – Comparação do erro no cenário utilizando movimento linear com ruído. . .	22
Figura 10 – Comparação da posição estimada com a posição real no eixo x no cenário com ruído, movimento linear, velocidade de 1 m/s no nó emissor e com 3 nós âncoras. . . . .	23
Figura 11 – Comparação da posição estimada com a posição real no eixo y no cenário com ruído, movimento linear, velocidade de 1 m/s no nó emissor e com 3 nós âncoras. . . . .	23
Figura 12 – Comparação do erro no cenário utilizando movimento circular sem ruído. .	24
Figura 13 – Comparação do erro no cenário utilizando movimento circular com ruído. .	25
Figura 14 – Comparação da posição estimada com a posição real no eixo x no cenário com ruído, movimento circular, velocidade de 1 m/s no nó emissor e com 3 nós âncoras. . . . .	26
Figura 15 – Comparação da posição estimada com a posição real no eixo y no cenário com ruído, movimento circular, velocidade de 1 m/s no nó emissor e com 3 nós âncoras. . . . .	26
Figura 16 – Comparação do erro no cenário utilizando movimento aleatório sem ruído. .	27
Figura 17 – Comparação do erro no cenário utilizando movimento aleatório com ruído. .	28
Figura 18 – Comparação da posição estimada com a posição real no eixo x no cenário com ruído, movimento aleatório, velocidade de 1 m/s no nó emissor e com 3 nós âncoras. . . . .	29
Figura 19 – Comparação da posição estimada com a posição real no eixo y no cenário com ruído, movimento aleatório, velocidade de 1 m/s no nó emissor e com 3 nós âncoras. . . . .	29
Figura 20 – Comparação entre os movimentos dos nós. . . . .	30
Figura 21 – Comparação entre os movimentos dos nós com a presença do ruído. . . . .	31

## LISTA DE TABELAS

Tabela 1 – Comparação entre os trabalhos relacionados. . . . .	11
Tabela 2 – Exemplo do conteúdo presente no arquivo .csv. . . . .	16
Tabela 3 – Exemplo da tabela para gerar os gráficos. . . . .	19

## LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
ACK	Acknowledgement
AGPS	Assisted GPS
AoA	Angle of Arrival
CSMA/CA	Carrier-Sense Multiple Access with Collision Avoidance
DECOM	Departamento de Computação
EKF	Extended Kalman Filter
FTM	Fine-Timing Measurement
GPS	Global Positioning System
GSM	Global System for Mobile Communications
LLS-RS	Linear Least Square-Reference Selection
MAC	Medium Access Control
MIMO	Multiple Input Multiple Output
MLE	Maximum Likelihood Estimation
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open System Interconnection
RSSI	Received Signal Strength Indication
RTof	Return Time of Flight
RTT	Round-Trip Time
SMACOF	Stress Majorization of a Complicated Function
TA	Trajectory Alignment
TDoA	Time Difference of Arrival
ToA	Time of Arrival
VANTs	Veículos Aéreos Não Tripulados

WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Network
WPAN	Wireless Personal Area Network
WWAN	Wireless Wide Area Network

## LISTA DE SÍMBOLOS

$x_1$	Coordenada x do nó âncora 1.
$y_1$	Coordenada y do nó âncora 1.
$x$	Coordenada x da intersecção dos círculos.
$y$	Coordenada y da intersecção dos círculos.
$x_e$	Coordenada x do nó emissor.
$y_e$	Coordenada y do nó emissor.
$t_{envio}$	Tempo que a mensagem foi enviada
$t_{recepcao}$	Tempo que a mensagem foi recebida
$v$	Velocidade de propagação da luz no ar
$\Delta t_{l,1}$	Tempo de processamento durante envio e recepção nos nós.
$\Delta t_{l,2}$	Tempo de processamento durante envio e recepção nos nós.
$\Delta t_{l,3}$	Tempo de processamento durante envio e recepção nos nós.
$\Delta t_{r,1}$	Tempo para recepção completa da mensagem.
$\Delta t_{r,2}$	Tempo para recepção completa da mensagem.

## LISTA DE ALGORITMOS

Algoritmo 1 – Script para rodar a simulação. . . . .	18
Algoritmo 2 – Função para retirar apenas as médias do <i>dataframe</i> bruto. . . . .	19

## SUMÁRIO

<b>1 – INTRODUÇÃO</b>	<b>1</b>
1.1 Objetivos	2
1.2 Organização do texto	2
<b>2 – FUNDAMENTAÇÃO TEÓRICA</b>	<b>3</b>
2.1 Wi-Fi	3
2.2 RToF	5
2.3 <i>Maximum Likelihood Estimation</i>	6
2.4 Mobilidade	8
2.5 Trabalhos Relacionados	9
<b>3 – DESENVOLVIMENTO</b>	<b>12</b>
3.1 Configurações iniciais	13
3.2 Movimentação	14
3.3 Coleta de dados	16
3.4 Geração dos gráficos	18
<b>4 – RESULTADOS</b>	<b>20</b>
4.1 Movimento Linear	20
4.2 Movimento Circular	22
4.3 Movimento aleatório	24
4.4 Comparação entre os movimentos	25
<b>5 – CONCLUSÃO</b>	<b>32</b>
5.1 TRABALHOS FUTUROS	33
<b>Referências</b>	<b>34</b>

## 1 INTRODUÇÃO

A tecnologia de geolocalização vem sendo utilizada cada vez mais nos dias atuais, sendo muito utilizada em *smartphones* para auxiliar motoristas (KESTRING, 2014), para fins de entretenimento (VIANA, 2011) e em veículos autônomos (KUUTTI et al., 2018). Devido a esses novos serviços, obter a estimativa de posição de um dispositivo de forma precisa se torna muito importante para o desempenho das aplicações.

Dessa forma, a ferramenta de geolocalização nos permite determinar a posição geográfica de um dispositivo baseando-se em um sistema de coordenadas. Para isso, existem diferentes formas de se localizar um dispositivo, sendo os principais GPS (*Global Positioning System*), Radiofrequência ou GSM (*Global System for Mobile Communications*), AGPS (*Assisted GPS*) e Wi-Fi (ROXIN et al., 2007).

O método GPS, muito utilizado por aplicativos através dos *smartphones*, funciona de forma global por meio de 24 satélites disponíveis no espaço. Cada satélite emite sinais de rádio e mediante ao tempo que o sinal levou para se chegar ao destino, é determinado a distância entre o satélite e o dispositivo a ser localizado, sendo assim, é possível realizar a trilateração, a partir de no mínimo 3 satélites, obtendo a posição do objeto (BAJAJ; RANAWEEERA; AGRAWAL, 2002).

Porém, o GPS possui algumas deficiências como ter um alto custo para implementação, alto consumo de energia, possuir um *delay* elevado para se obter a posição do objeto e não tem uma acurácia aceitável em ambientes *indoor* (SINGH et al., 2012). Assim, no método AGPS são utilizados outros métodos de localização fornecendo assistência ao GPS, podendo ser GSM ou Wi-Fi (BUSIC; FILJAR; DESIC, 2005). O GSM funciona de forma parecida com o GPS, com o tempo sendo utilizado como métrica para estimar a distância entre os nós, porém no método GSM não são utilizados satélites como referência, são usadas torres de operadoras de telefonia móvel que por meio de ondas de rádio se calcula a distância entre a torre e o dispositivo a ser localizado (SINGH et al., 2012).

Outra opção é o uso da tecnologia de comunicação Wi-Fi, explorada por Kan, Chan e Ng (2003) utilizando a técnica *RToF* como uma possível solução alternativa para o GPS, uma vez que, em alguns ambientes o GPS apresenta limitações, principalmente em regiões com quantidade elevada de obstáculos por o sinal acabar sofrendo desvios e sendo bloqueado.

O método *RToF* funciona de forma semelhante ao método GPS, necessitando de no mínimo 3 estações fixas com posições conhecidas para receber o sinal, que pode-se denominar de nós âncoras. A partir daí, tem-se o nó emissor que desconhece sua própria localização e cujo objetivo é determiná-la a partir da troca de mensagens com os nós âncoras. Sendo assim, é necessário que haja uma rede onde os nós possam trocar mensagens, sendo proposto em 2016 um protocolo específico para localização chamado FTM (*Fine-Timing Measurement*) inserido no padrão de rede IEEE 802.11mc (MA et al., 2020).



A dificuldade em localizar um dispositivo se torna ainda maior quando o mesmo está em movimento, pois enquanto as mensagens estão sendo transmitidas o objeto pode se mover, e então o algoritmo irá calcular a posição se baseando na posição passada, causando erro na estimativa de posição. Dessa forma, para analisar a performance da técnica é importante considerar diferentes modelos de movimentação do nó emissor, como o movimento linear, circular e aleatório (OMNETPP, 2021c).

Cada movimento visa simular possíveis trajetos que o dispositivo possa realizar em um ambiente real, necessitando de mais estudos sobre a técnica utilizando movimentação no nó emissor, visto que, os trabalhos presentes na literatura dão enfoque na análise em técnicas para se calcular a trilateração de forma mais precisa, ou a utilização de outras técnicas em conjunto.

## 1.1 Objetivos

Avaliar o impacto dos diferentes modelos de mobilidade do nó emissor para a solução de localização do RToF com o método *Maximum Likelihood Estimation* utilizando um ambiente de simulação OMNeT++. Nessa ferramenta é possível configurar uma rede WLAN com no mínimo 4 dispositivos inseridos dentro dela, sendo um nó emissor que se move de forma linear, circular ou aleatória, e três nós âncoras fixos. Também são analisados diferentes velocidades para cada modelo de mobilidade e um número maior de nós âncoras fixos de 6 e 9.

## 1.2 Organização do texto

O texto se encontra organizado em 5 capítulos: Introdução, Fundamentação Teórica, Desenvolvimento, Resultado e Conclusão. O Capítulo 2 é responsável por explicar os principais conceitos que foram utilizados durante este trabalho, como as redes Wi-Fi, a técnica RToF, o método de trilateração *Maximum Likelihood Estimation*, os modelos de mobilidade e por fim os trabalhos relacionados. Já no Capítulo 3 é apresentado como foi desenvolvido o trabalho, sendo subdivido em 4 subseções sendo uma que trará informações sobre como foi configurado o ambiente de simulação, outra se referindo a implementação dos movimentos no nó emissor, outra subseção explicando como os dados foram coletados e a última subseção desse capítulo falando sobre como os gráficos foram gerados. Em seguida, tem-se a discussão dos gráficos gerados pela simulação no Capítulo 4. Por fim, o Capítulo 5 relata o que se concluiu com o trabalho e possíveis trabalhos que poderão ser feitos para melhor analisar essa técnica.

## 2 FUNDAMENTAÇÃO TEÓRICA

Para melhor entendimento do projeto é importante ter conhecimento sobre algumas tecnologias utilizadas, portanto nesta seção será explicado os principais conceitos pertencente ao projeto, através da subseção 2.1 que explicará como se dá a transmissão de dados por meio de rede sem fio, a subseção 2.2 que detalhará a técnica RToF, outra para explicar a técnica de trilateração *Maximum Likelihood Estimatio* 2.3 e por fim as subseções 2.4 e 2.5 explicando a mobilidade no nó emissor e os trabalhos relacionados a técnica RToF em conjunto com o MLE, respectivamente.

### 2.1 Wi-Fi

Com o avanço da tecnologia as redes sem fio se tornaram muito comum em nosso cotidiano, sendo responsável por permitir a troca de dados pela transmissão de sinais de rádio sem a necessidade de cabos entre dispositivos dentro de uma mesma área. Sendo assim, a transmissão de dados pode ser feita por radiação infravermelha e por radiofrequência, podendo ser ou não auxiliada por satélites. (TANENBAUM; WETHERALL, 2010).

A tecnologia de redes sem fio se divide em 4 tipos (MACEDO; FRANCISCATTO; BERTOLINI, 2018):

- **WPAN (Wireless Personal Area Network):** são caracterizadas por uma transmissão de dados que ocorre somente em um raio de dezenas de metros. Um exemplo de tecnologia WPAN são as conexões Bluetooth muito utilizadas para conectar impressoras, fones de ouvidos e periféricos pessoais.
- **WLAN (Wireless Local Area Network):** é uma rede local que realiza transmissão de dados dentro de uma pequena área em torno de 100 e 300 metros.
- **WMAN (Wireless Metropolitan Area Network):** nessa categoria as conexões podem ser realizadas entre dispositivos distantes a dezenas de quilômetros. Pode ser usada para interconectar redes WLAN, sendo muito usada em Universidades e empresas com departamentos distintos em uma mesma área.
- **WWAN (Wireless Wide Area Network):** essa rede é a que consegue cobrir uma grande área e conectar grandes quantidades de dispositivos, sendo muito utilizada pelos dispositivos móveis através das redes 3G, 4G e 5G.

Toda comunicação entre duas máquinas ou mais só é possível devido aos protocolos de redes que especificam a estrutura dos pacotes adicionando informações detalhadas para o seu funcionamento em cabeçalhos. Assim, as máquinas que recebem esses pacotes conseguem identificar os conteúdos de cada campo do cabeçalho e realizar o processamento necessário para a troca de mensagens com sucesso (TANENBAUM; WETHERALL, 2010).

No início da implementação desses protocolos, cada empresa possuía o seu próprio

protocolo, o que ocasionava alguns problemas na rede devido à falta de compatibilidade entre os diferentes equipamentos. Com o intuito de solucionar esse impasse criado, surge o modelo OSI (*Open System Interconnection*), que define um modelo de referência para comunicação de redes divididas em 7 camadas: Física, Enlace de Dados, Rede, Transporte, Sessão, Apresentação e Aplicação (LOPES; BENEVENUTO; OLIVEIRA, 2015).

Dessa forma, para as comunicações de redes sem-fio, o IEEE (Instituto de Engenheiros Eletricistas e Eletrônicos) criou o padrão de comunicação IEEE 802.11, o qual especifica o conjunto de protocolos das camadas de enlace e física da rede. Possuindo diferentes variações conforme a tecnologia foi evoluindo, sendo algumas delas a 802.11b, 802.11g, 802.11n, 802.11c e 802.11ac. (HIERTZ et al., 2010)

A partir do padrão 802.11a foi implementado na camada física, a técnica de transmissão OFDM (*Orthogonal Frequency Division Multiplexing*), que divide sua banda em múltiplas portadoras ortogonais, permitindo que a transmissão ocorresse a uma taxa de transmissão de até 54 Mb/s (HIERTZ et al., 2010).

Essa técnica de transmissão foi expandida para os novos padrões, incluindo o 802.11n. Nesse padrão, além da técnica OFDM, também foi aumentado a largura de banda dos canais para até 40 Mhz e implementado a tecnologia MIMO (*Multiple Input Multiple Output*), que permite múltiplas transmissões de dados acontecendo de forma simultânea, resultando no aumento da capacidade do canal por meio do emprego de mais de uma antena de transmissão e recepção. Essas melhorias na camada física do padrão 802.11n permitiram que a taxa de transmissão chegasse até 600 Mb/s nominalmente, superando os padrões anteriores 802.11g e 802.11a (PERAHIA, 2008).

Na camada de enlace é implementado o protocolo de acesso ao meio (*Medium Access Control – MAC*) CSMA/CA (*Carrier-Sense Multiple Access with Collision Avoidance*), que é importante para que durante a transmissão dos dados não exista colisões entre os pacotes, prejudicando a comunicação entre os dispositivos. Dessa forma, o CSMA/CA atua como prevenção, ouvindo o meio em que o dispositivo irá transmitir as mensagens e transmitindo apenas se estiver livre. Caso contrário, é selecionado um *backoff timer* aleatoriamente para aguardar a retransmissão do pacote (BIANCHI; FRATTA; OLIVERI, 1996).

Como o CSMA/CA não consegue detectar se na transmissão do pacote houve colisões, então para garantir que o pacote realmente foi entregue, ao enviar o sinal é enviado também em seguida um pacote ACK, que é um pacote de reconhecimento. Assim, se o transmissor não receber de volta esse pacote de reconhecimento dentro de um tempo específico (*ACK\_Timeout*) ou receber um pacote ACK diferente do que foi enviado, então o transmissor assume que a mensagem não foi entregue ao destino reprogramando-se para enviá-la novamente (BIANCHI; FRATTA; OLIVERI, 1996).

Utilizando as redes de comunicação sem-fio surgiram diferentes aplicações, sendo uma delas rastrear um dispositivo dentro de uma WLAN. Para isso, existem métodos baseados na própria troca de mensagens do sistema de comunicação, calculando-se o tempo de viagem dos

sinais transmitidos. Dessa forma, para o auxílio da localização através dos tempos dos sinais, surgiu recentemente o FTM (*Fine Timing Measurement*), um protocolo que foi introduzido pelo padrão IEEE 802.11mc (IEEE, 2016) com intuito de auxiliar na localização, permitindo o cálculo do tempo médio que os sinais levam para sair do nó de origem e chegar no destino. Com esse tempo pode ser estimada a distância entre os nós ao multiplicá-la pela velocidade de propagação do sinal e, assim, é possível realizar os cálculos de trilateração para localizar o dispositivo (HUILLA et al., 2019).

## 2.2 RToF

Visto que, a localização está se tornando uma tecnologia muito utilizada nos dias atuais é necessário que suas técnicas evoluam para que haja maior precisão nos resultados. Dessa forma, além da utilização de GPS, há novas implementações utilizando redes Wi-Fi, com a finalidade de complementar a tecnologia GPS, principalmente em ambientes fechados, onde sua performance é prejudicada (ZAFARI; GKELIAS; LEUNG, 2019).

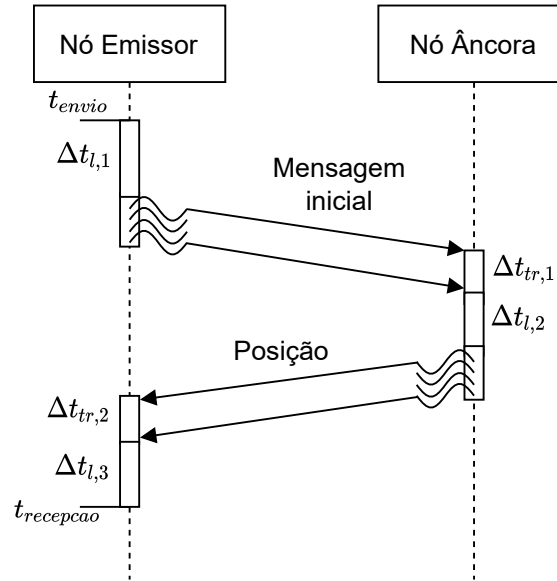
A localização de dispositivo por redes sem-fio pode ser realizada através de diferentes métodos, utilizando a potência do sinal (RSSI - *Received Signal Strength Indication*), o tempo de viagem das mensagens (RTT/RToF - *Round-Trip Time/Round-Trip Time of Flight*, TDoA - *Time Difference of Arrival* - e ToA - *Time of Arrival*) e pelo ângulo de chegada das mensagens (AoA - *Angle of Arrival*). O RToF, em relação aos métodos ToA e TDoA, se beneficia por não necessitar que os nós estejam sincronizados pelo tempo, porém, para cada nó  $n$ , a complexidade de seu algoritmo é incrementado de ordem  $O(n^2)$ , sendo assim, torna-se muito custoso o algoritmo para uma rede com um número elevado de nós (ZHANG et al., 2010).

O método RToF se baseia no tempo em que a mensagem leva para sair do dispositivo que emite o sinal (nó emissor), chegar no dispositivo que recebeu o sinal (nó âncora), e retornar novamente ao nó emissor, como pode ser observado na Figura 1. Sendo assim, a distância entre os nós pode ser calculada por (1), onde  $v$  corresponde à velocidade da luz no ar, que é a velocidade em que o sinal se propaga,  $t_{envio} - t_{recepcao}$  é o tempo que o sinal leva para ir e voltar,  $\Delta t_{l,1}$ ,  $\Delta t_{l,2}$  e  $\Delta t_{l,3}$  são os tempos de processamento durante o envio e a recepção nos nós,  $\Delta t_{tr,1}$  e  $\Delta t_{tr,2}$  são os tempos para a recepção completa da mensagem, já que o sinal é composto por mais de um bit.

$$D = \frac{v(t_{envio} - t_{recepcao} - (\Delta t_{l,1} + \Delta t_{l,2} + \Delta t_{l,3} + \Delta t_{tr,1} + \Delta t_{tr,2}))}{2} \quad (1)$$

Essa distância calculada pelo RToF pode ser influenciada pelo atraso ou adiantamento das medições dos tempos de chegada e saída do sinal realizada com o *clock* da unidade de processamento responsável pelas medições de tempo, resultando em um ruído dependente do hardware.

Figura 1 – Comunicação entre o nó emissor e o nó âncora com medidas de tempo para as medições do RToF.

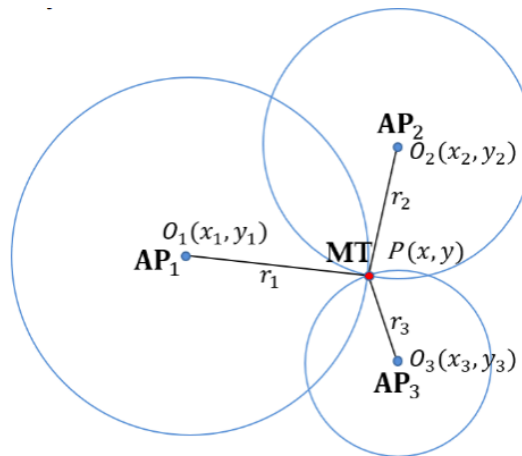


Fonte: Autoria própria.

### 2.3 Maximum Likelihood Estimation

A partir das distâncias entre o nó emissor e os nós âncoras é possível encontrar a posição do nó emissor por meio do método *Maximum Likelihood Estimation*, que consiste em localizar o objeto através da intersecção dos círculos demonstrados pela Figura 2, que tem seus centros nos nós âncoras. Assim, para cada distância entre o nó emissor e nó receptor tem-se o círculo definido por (2), onde  $x_1$  e  $y_1$  são as coordenadas do nó âncora 1 e  $x, y$  são os pontos de intersecção dos círculos.

Figura 2 – Intersecção dos círculos para cálculo da trilateração.



Fonte: (MA et al., 2020).

$$D = \sqrt{(x - x_1)^2 + (y - y_1)^2} \quad (2)$$

O mesmo se aplica para os demais nós  $n$ , chegando ao sistema expresso por (3) (MURPHY; HEREMAN, 1995). A solução em comum para esse sistema fornece a localização do nó emissor, definida como  $x_e$  e  $y_e$ .

$$\begin{cases} (x - x_1)^2 + (y - y_1)^2 = D_1^2 \\ (x - x_2)^2 + (y - y_2)^2 = D_2^2 \\ \vdots \\ (x - x_n)^2 + (y - y_n)^2 = D_n^2 \end{cases} \quad (3)$$

Assim, para realizar a trilateração é possível aplicar a técnica *Maximum Likelihood Estimation* a fim de obter melhor precisão e eficiência nos resultados. Para utilização dessa técnica em um ambiente 2D são necessários no mínimo três dispositivos com posições já conhecidas. A partir daí, dentro do sistema demonstrado por (3) pode-se subtrair cada equação por sua subsequente resultando em (4) (ZHANG et al., 2010).

$$\begin{cases} x_1^2 - x_n^2 - 2x_e(x_1 - x_n) + y_1^2 - y_n^2 - 2y_e(y_1 - y_n) = D_1^2 - D_n^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 - 2x_e(x_{n-1} - x_n) + y_{n-1}^2 - y_n^2 - 2y_e(y_{n-1} - y_n) = D_{n-1}^2 - D_n^2 \end{cases} \quad (4)$$

Transformando esse sistema em matrizes é possível reescrever o problema como (5), onde  $\mathbf{b}$  é o vetor definido por (6), a matriz  $\mathbf{A}$  é definida por (7) e o vetor  $\mathbf{X}$  é a posição do nó emissor representado por (8).

$$\mathbf{b} = \mathbf{A}\mathbf{X} \quad (5)$$

$$\mathbf{b} = \begin{bmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + D_n^2 - D_1^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + D_{n-1}^2 - D_n^2 \end{bmatrix} \quad (6)$$

$$\mathbf{A} = \begin{bmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ \vdots & \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{bmatrix} \quad (7)$$

$$\mathbf{X} = \begin{bmatrix} x_e \\ y_e \end{bmatrix} \quad (8)$$

Assim, utilizando o método de *Minimum Variance Estimation*, a solução para  $\mathbf{X}$  pode ser obtida por meio de (9) (ZHANG et al., 2010).

$$\mathbf{X} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} \quad (9)$$

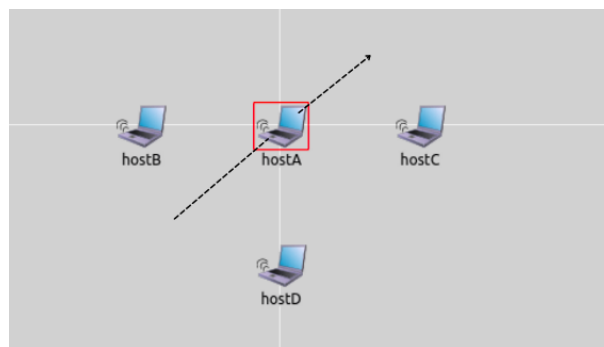
## 2.4 Mobilidade

A mobilidade do nó também pode influenciar na estimativa de posição, uma vez que existe um atraso para se realizar o cálculo da trilateração. Isso se deve ao fato de que no momento que é obtida a posição do dispositivo, existe a possibilidade do mesmo já ter se deslocado, influenciando na acurácia do algoritmo (HALDER; GHOSAL, 2016).

Dessa forma, é importante avaliar as possíveis movimentações dos nós para as soluções de localização. Existem três modelos de mobilidade que se destacam na literatura: movimento aleatório, predefinido e controlado. No movimento aleatório o nó se move de forma aleatória dentro da área em que está inserido. Já no movimento predefinido, a forma com que o nó se move, assim como sua velocidade são definidas antes do início da atividade e permanecem até o encerramento. Por fim, no modo controlado, o nó se move conforme as especificações inseridas por uma entidade externa em tempo real (SILVA; SILVA; BOAVIDA, 2014).

No movimento predefinido a trajetória pode ser definida de forma linear ou circular. Na forma linear para ambientes 2D, é definido a posição  $(x_{in}, y_{in})$  em que o nó irá iniciar a sua trajetória, assim como o tamanho máximo da área em que o dispositivo será permitido andar no eixo. A partir daí, é informado um ângulo inicial para o nó começar a se mover em linha reta com velocidade geralmente definida em  $m/s$  pelo usuário (OMNETPP, 2021b). Cada movimento do nó sendo demonstrado pelas Figuras 3, 4 e 5.

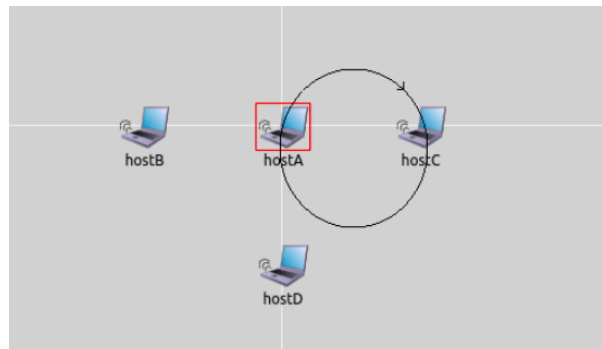
Figura 3 – Movimento linear do nó.



Fonte: Autoria própria.

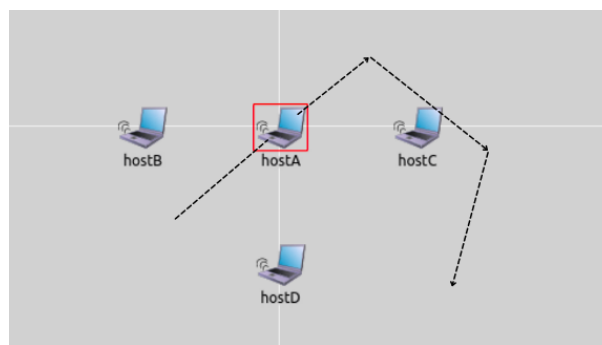
Além da trajetória linear, o modelo predefinido implementa também o circuito circular. No caminho circular inicia-se determinando as coordenadas  $(x_{raio}, y_{raio})$  do centro do círculo,

Figura 4 – Movimento circular do nó.



Fonte: Autoria própria.

Figura 5 – Movimento aleatório do nó.



Fonte: Autoria própria.

sendo possível traçar a trajetória do dispositivo inserindo o raio do círculo e sua velocidade (OMNETPP, 2021a).

O percurso do nó também pode ser determinado de forma aleatória. Deste modo, o nó é iniciado na posição  $(x_{in}, y_{in})$ , podendo ser definida pelo usuário, e assim irá se mover em linha reta para um ponto alvo definido de forma aleatória, sendo que, no momento da chegada no destino, o mesmo espera um tempo de intervalo e em seguida é definido um novo ponto para o dispositivo se mover, tendo sua velocidade também sendo definida de forma aleatória (OMNETPP, 2021d).

## 2.5 Trabalhos Relacionados

Assim como este trabalho realizado, o artigo Han et al. (2021) também visa avaliar a técnica RToF realizando experimentos com o dispositivo se movendo de forma linear e aleatória, porém utilizando o método *Linear Least Square-Reference Selection* (LLS-RS) para o cálculo da trilateração determinando a posição do nó. Dessa forma, o trabalho é realizado comparando a técnica RToF pura e em conjunto com técnicas de linearização como o *Trajectory Alignment* (TA) e *Extended Kalman Filter* (EKF), demonstrando que o tipo de movimento do nó não tem grande influência no resultado, diferentemente do auxílio das técnicas de linearização. O



uso da técnica TA possibilitou uma ligeira melhora na precisão do algoritmo.

Já no trabalho de Saeed, Naseem e Zaidi (2007), a posição do nó em movimento é estimada utilizando o modelo auto-regressivo AR-1, sendo os valores dos parâmetros da função determinada através da aplicação das equações de *Yule-Walker* em um *dataset* de treino. Tendo esses parâmetros definidos, o método *Kalman Filter* define o estado da mobilidade, ou seja, a posição, velocidade e aceleração do nó tornando-se possível monitorar o deslocamento do dispositivo. Dessa forma, a técnica RToF é comparada com a RSSI em ambientes em que os sinais sofrem interferência por obstáculos, concluindo-se que a técnica RToF para ambientes que haja muitas paredes ou empecilhos para o sinal, apresentou um erro quadrático médio de 55,316 m, sendo inferior ao erro quadrático médio do RSSI.

Ainda se tratando de trabalhos que relatam a performance do RToF com o dispositivo em movimento como o trabalho apresentado, há o Yan et al. (2019) que propõem minimizar os erros que ocorrem devido aos múltiplos sinais de rádio que alcançam o nó receptor. Dessa forma, este artigo realiza o estudo do método de localização com auxílio da estratégia de otimização SMACOF (*Stress Majorization of a Complicated Function*) aplicando o protocolo de comunicação FTM e inserindo a mobilidade linear no nó. Os resultados demonstram que, mesmo em ambientes complexos como shoppings e aeroportos, o algoritmo desenvolvido ainda fornece uma acurácia abaixo de 3 m semelhante aos algoritmos propostos em ambientes simples, conforme comparado com artigos da literatura.

Entretanto existem pesquisas que focam somente na técnica de localização RToF, não utilizando o movimento do nó como métrica para avaliação, como pode ser visto em Park, Ahn e Yu (2007). Os autores avaliaram o método com os nós estando em posições estáticas e empregando o procedimento *Least Square* para cálculo da trilateração, sendo possível perceber um erro médio menor se comparado com o nó em movimento, visto que, o erro apresentado para o nó estático foi de 0,76 m. Esta técnica ainda pode ser aprimorada se utilizada em conjunto com outras técnicas conforme visto em Bahillo et al. (2010) que analisa a técnica RSSI em conjunto com a técnica RToF com o dispositivo se movendo de forma aleatória, resultando em uma performance melhor do que o RToF puro e o RSSI puro.

A Tabela 1 apresenta um resumo das principais características dos trabalhos relacionados e deste trabalho. Este trabalho é responsável por analisar a influência dos diferentes tipos de mobilidade quando considerada a localização obtida pela técnica RToF, algo que ainda não foi descrito nos trabalhos encontrados na literatura, uma vez que analisam apenas um tipo de mobilidade ou, no caso em que foram considerados dois tipos de mobilidade, por exemplo em Han et al. (2021), não é avaliada a questão da mobilidade, mas sim as técnicas de linearização para aprimoramento da estimativa de posição. Outros aspectos que se diferenciam é a avaliação de diferentes números de nós âncoras para se obter a posição com o RToF com suporte do CSMA/CA, que permite incorporar os tempos do funcionamento do CSMA/CA na solução.

Tabela 1 – Comparação entre os trabalhos relacionados.

<b>Autores</b>	<b>Movimento do dispositivo</b>	<b>Método para cálculo da trilateração</b>	<b>Estratégia para obter as distâncias entre os nós</b>	<b>Quantidade nós âncoras</b>	<b>Área delimitada para mobilidade</b>	<b>Técnica utilizada em conjunto com o RToF</b>
Han et al. (2021)	Linear e aleatório	LLS-RS	FTM	n/a	n/a	TA e EKF
Saeed, Naseem e Zaidi (2007)	n/a	Auto-regressivo AR-1	n/a	4	n/a	n/a
Yan et al. (2019)	Linear	<i>Least Square</i>	FTM	8	96 m <sup>2</sup>	Estratégia de otimização <i>SMACOF</i>
Park, Ahn e Yu (2007)	Estática	<i>Least Square</i>	RToF	3	2500 m <sup>2</sup>	n/a
Bahillo et al. (2010)	Aleatório	n/a	Técnica que combina as estimativas do RToF e RSSI	4	1200 m <sup>2</sup>	Técnica de localização RSSI
Este trabalho	Linear, circular e aleatória	<i>Maximum Likelihood Estimation</i>	RToF com suporte do CSMA/CA	3/6/9	100 m <sup>2</sup>	CSMA/CA modificado

Fonte: Autoria própria.

### 3 DESENVOLVIMENTO

Sabendo da importância de determinar a localização de forma mais precisa possível, a técnica RToF pode ser considerada como uma possível solução de aprimoramento para localizar dispositivos. Apesar de apresentar resultados promissores, ainda é necessário analisar diferentes cenários para determinar a eficiência desta técnica.

Esta técnica para ser executada deve conter no mínimo 3 nós âncoras, que correspondem aos dispositivos de posição fixa e conhecida e que também são considerados nós receptores nesse cenário, além do nó emissor, que é o dispositivo de posição desconhecida. O nó emissor será responsável por enviar as mensagens para os nós âncoras e os nós âncoras devolvem a mensagem para o nó emissor. Após o nó emissor receber a mensagem de todos os nós âncoras, o mesmo realiza os cálculos através do tempo de transmissão da mensagem para determinar sua localização.

Sendo assim, neste estudo foi desenvolvido uma análise da técnica RToF em conjunto com o método de trilateração MLE (*Maximum Likelihood Estimation*) com o nó emissor se movendo, a fim de verificar como o movimento do nó interfere na estimativa de posição.

Para isso, foi utilizado a ferramenta OMNeT++ que permitiu a criação do ambiente de simulação, em conjunto com o *framework* INET que possibilitou a criação de uma rede WLAN, além de ser um software de código aberto que possibilitou a implementação do protocolo de comunicação CSMA/CA. Desta maneira, foram configurados 27 cenários variando a velocidade (1 m/s, 15 m/s e 30 m/s), quantidade de nós âncoras (3, 6 e 9) e a trajetória no nó emissor (Linear, Circular ou Aleatório).

Os parâmetros de velocidades foram definidos para todos os cenários onde existe a mobilidade, como para uma pessoa caminhando (1 m/s) (HIGHWAY... , 2000), na velocidade de um carro em uma avenida (15 m/s) (HIGHWAY... , 2000) e um VANT de asa fixa (30 m/s) (ZHANG et al., 2018) com intuito de avaliar a interferência da velocidade na estimativa de posição.

Já para a quantidade de nós, o valor mínimo foi definido de acordo com a quantidade mínima de nós âncoras para o algoritmo funcionar (3) e então foi aumentado esse número por um fator igual a 3 para avaliar sua performance.

Todos os nós estão inseridos dentro de uma rede WLAN utilizando o padrão IEEE 802.11n , sendo inserido o protocolo CSMA/CA na camada de enlace para que não haja colisões entre os pacotes que estão sendo transmitidos, porém foram realizadas algumas modificações nesse protocolo. Como visto anteriormente, o CSMA/CA aguarda um tempo aleatório e logo após escuta o meio para saber se pode enviar ou não a mensagem. Este tempo tem que ser retirado na hora de realizar o cálculo das medidas do RToF, pois poderia acarretar em um erro muito grande na estimativa de posição. Sendo assim, é necessário saber o valor deste tempo, sendo o mesmo enviado pela mensagem dentro dos cabeçalhos do próprio CSMA/CA

modificado.

Os cálculos das distâncias do RToF entre o nó emissor e os nós âncoras são implementados pela aplicação do nó emissor e calculadas continuamente enquanto o nó se move em uma área de 50 m<sup>2</sup>. A cada iteração, também foi obtida a estimativa da posição do nó emissor com o *Maximum Likelihood Estimation* a partir das posições conhecidas dos nós âncoras, que também são transmitidas nas mensagens de resposta. Assim, coleta-se as informações sobre a posição com a finalidade de analisar a influência de cada tipo de movimento sobre a estimativa de posição.

A implementação também se baseou inicialmente no projeto de SOUZA (2022) como referência, já que ele oferece o estudo do algoritmo RToF utilizando posições fixas do nó emissor com diferentes distâncias entre o nó emissor e nós âncoras, além de implementar o protocolo de comunicação CSMA/CA entre eles.

Dessa forma, logo após baixado o projeto do repositório <[https://github.com/allan-silveira/RToF\\_1](https://github.com/allan-silveira/RToF_1)> foram realizadas modificações para implementar o protocolo CSMA/CA e foi utilizado a biblioteca “*Node Mobility*” para poder aplicar a movimentação no nó emissor. Também foi configurado a distância entre o nó emissor e os nós âncoras para um valor inicial fixo, uma vez que, quando o nó emissor realizar a movimentação, essa distância entre os nós já irá ser alterada.

Portanto, as configurações iniciais serão melhor detalhadas na subseção 3.1, assim como, consecutivamente, na subseção 3.2 será abordada a implementação do movimento do nó. Os processos de coleta e manipulação de dados também foram demonstrados na sequência, nas subseções 3.3 e 3.4, sendo todos os códigos disponibilizados no repositório <[https://github.com/angelospi/RToF\\_TCC](https://github.com/angelospi/RToF_TCC)>.

### 3.1 Configurações iniciais

Para início do desenvolvimento da simulação foi necessário instalar o programa OM-NeT++ em conjunto com o pacote do *framework* INET. Com os devidos programas instalados foi baixado o projeto do repositório de referência, no qual já estava configurado as comunicações entre os nós, porém ainda faltava a implementação do movimento dos nós, dos cálculos da estimativa de posição e da comunicação se repetir enquanto o nó se movia.

Ao rodar o projeto já baixado, ainda foi necessário algumas alterações na configuração do projeto, como o *download* da biblioteca Eigen, que permite realizar cálculos utilizando matrizes. Também foi preciso alterar dois arquivos para adicionar o protocolo de comunicação CSMA/CA modificado. Para isso, foi necessário acrescentar a linha “*const Protocol Protocol::csmaCaMacRToF(“csmaCaMacRToF ”, “CSMA/CA MA”);*” no arquivo *inet/common/Protocol.cc* e também a linha “*static const Protocol csmaCaMacRToF;*” no arquivo *inet/common/Protocol.h*.

Após a correta configuração do ambiente de desenvolvimento, o foco do trabalho foi a implementação da movimentação do nó emissor de acordo com os modelos apresentados na

seção 2.4.

### 3.2 Movimentação

Dessa forma, para implementar a movimentação foi modificado o arquivo *.ini* definindo o tipo de movimentação (linear, circular ou aleatória), a trajetória e a velocidade do nó. A trajetória é estabelecida pelos parâmetros de área máxima e mínima nos casos do movimento linear e aleatório, já no caso do circular foi necessário definir o ponto central e o raio da circunferência.

Todas essas configurações foram possíveis através do módulo *Node Mobility*. Logo, para implementar o movimento linear foi definido o *typename* do nó 1 que corresponde ao nó emissor acrescentando a linha *\*.host1.mobility.typename = "RectangleMobility"*. O mesmo se repetiu para os outros modelos mudando apenas o argumento do *typename* para *"CircleMobility"* e *"RandomWaypointMobility"*.

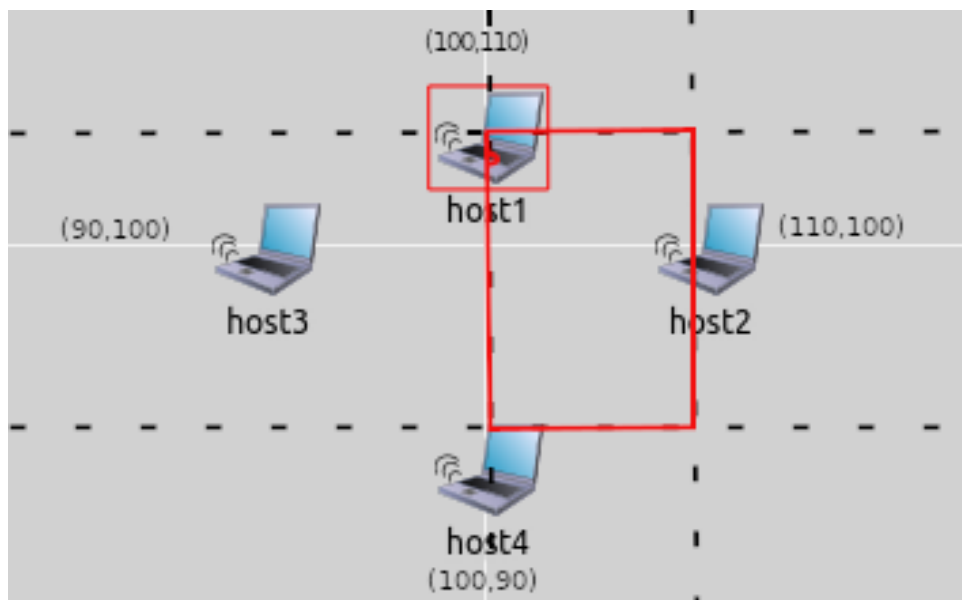
Para os modelos *RectangleMobility* e *RandomWaypointMobility* apenas foi necessário configurar a área máxima e mínima definindo os argumentos *constraintAreaMaxX* para 105 m e *constraintAreaMinX* para 100 m. Esse procedimento foi repetido para o eixo y configurando a área máxima para 105 e mínima 95 m. No caso do eixo z foi ajustado uma posição única de 0 m para trabalhar apenas em duas dimensões. A diferença entre os dois modelos é que no *RandomWaypointMobility* é definido um ponto aleatório para o nó se mover em linha reta, ao chegar no ponto destino, um novo ponto é definido dentro da área máxima e mínima, e assim segue-se até o fim da simulação. Já no *RectangleMobility* o nó se move até chegar no limite em que é permitido no eixo que está se movendo e assim muda-se de direção e navega-se até o limite novamente. Ou seja, o nó se move por cima das arestas do retângulo que delimitam a área, conforme demonstrado pela Figura 6, onde as linhas pontilhadas correspondem aos limites das áreas e a linha em vermelho o percurso do *host1*.

No modelo *"CircleMobility"* foi configurado um ponto central definindo o argumento *cx* (posição no eixo x) para 104 m e *cy* (posição no eixo y) para 100 m, enquanto no eixo z a posição foi a mesma das configurações anteriores de 0 m. Para esse modelo também foi necessário configurar o raio do círculo que o nó percorreu através da variável *r* sendo 4 m. A decisão de colocar o centro do círculo na posição 104 m no eixo x foi devido ao fato de que quando o nó emissor realiza sua trajetória, o mesmo passe pelo centro de um dos nós âncoras. Dessa forma sua trajetória pode ser melhor visualizada pela Figura 7.

Dentro do arquivo *.ini* foram criadas diversas seções, sendo que cada uma correspondia a configuração de um cenário. Portanto, foram configuradas 27 seções com suas respectivas características de acordo com o cenário.

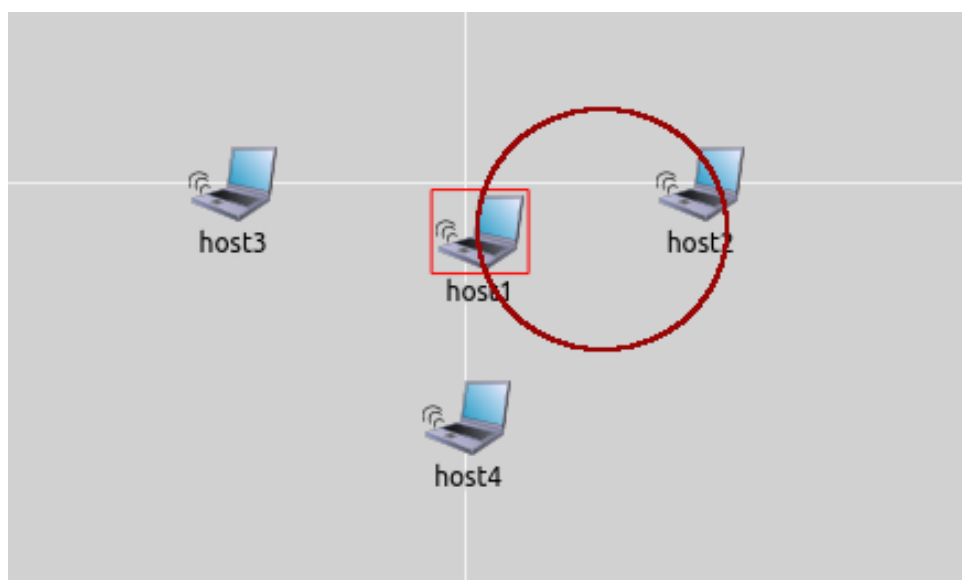
As configurações do posicionamento dos nós âncoras foram de 10 m de distância em relação ao centro. O trajeto linear foi configurado com um rota no formato de um retângulo, sendo que o nó passa pelo centro e percorre as extremidades onde os nós receptores foram colocados.

Figura 6 – Trajeto linear do nó emissor.



Fonte: Autoria própria.

Figura 7 – Trajeto circular do nó emissor.



Fonte: Autoria própria.

Assim, o movimento do nó já estava implementado, porém havia o problema de que os nós realizavam apenas uma comunicação, ou seja, o nó emissor enviava uma mensagem e essa mensagem era respondida. Mas, depois dessa comunicação, o nó emissor parava de emitir a mensagem, e então o único evento que ocorria era a movimentação do nó.

Portanto, após a implementação da movimentação foi configurado no nó emissor para que quando ele recebesse a mensagem de todos os nós receptores, reenviasse outra mensagem. Com isso, a comunicação entre os nós emissores e receptores ocorre de forma contínua, mantendo-se a comunicação enquanto o nó se move.

Para isso, foi implementado uma verificação no nó emissor se a quantidade de mensagens recebidas é igual ao número de nós âncoras, assim pode-se saber se todas as mensagens já tinham sido recebidas. Dessa forma, no momento em que todas as mensagens são captadas pelo nó emissor são realizados os cálculos da trilateração, são inseridos os dados no arquivo .csv e, por fim, é realizada uma nova transmissão *broadcast* (para todos os nós âncoras).

### 3.3 Coleta de dados

Após realizadas as configurações anteriores, a simulação já está ocorrendo da forma adequada. Sendo assim, precisou ser realizado a coleta dos dados para cada cenário dos experimentos. Para isso, foi implementado no nó emissor uma verificação se o mesmo já teria recebido as mensagens de todos os receptores, sinalizando que a comunicação entre emissor e receptor para aquela mensagem teria finalizado. Caso a comunicação realmente estivesse finalizada, então as informações de posição real do nó emissor no eixo x, posição real do nó emissor no eixo y, posição estimada do nó emissor no eixo x, posição estimada do nó emissor no eixo y, erro e tempo de simulação foram salvas em um arquivo .csv, conforme exemplificado pela Tabela 2.

Tabela 2 – Exemplo do conteúdo presente no arquivo .csv.

Likelihood position X (m)	Likelihood position Y (m)	Real position X (m)	Real position Y (m)	Erro	Tempo de Simulação
107,384	102,131	107,385	102,131	0,00125165	53,3450815
107,034	102,605	107,035	102,605	0,00164193	53,9361768
106,615	103,026	106,617	103,025	0,00151103	54,5286358
.	.	.	.	.	.
.	.	.	.	.	.
<b>Média</b>				0,0015449	

Fonte: Autoria própria.

Para cada cenário foi necessário definir um número de *seed* diferente, já que essas

*seeds* determinam os valores aleatórios gerados durante a execução da simulação. Ou seja, para uma *seed* de valor 1 por exemplo, todas as simulações feitas com essa *seed* terão as variáveis aleatórias de valores sempre iguais.

Sendo assim, cada cenário foi rodado com 10 *seeds* diferentes, sendo que para cada *seed* a simulação rodava até 60 segundos, um tempo que foi considerado adequado para que o nó emissor percorresse toda sua trajetória no momento em que está mais lento há 1 m/s. Assim, no momento em que a simulação finaliza, para cada *seed* foi calculada a média de todos os valores de erros do MLE pertencente a mesma *seed* e inserido essa média no arquivo também, podendo ser visualizado pela última linha da Tabela 2. Os valores das médias foram capturados para posteriormente realizar a comparação entre os diferentes cenários por meio de gráficos.

Após inseridos esses dados, foi verificado que houveram alguns erros na estimativa de posição, pois os valores em alguns casos estavam muitos incoerentes com o esperado. Assim, foi percebido que o nó emissor quando passa por cima de um receptor, por eles ocuparem exatamente a mesma posição, o que seria impossível em um cenário real, acontecia um erro no cálculo da distância, gerando um valor muito discrepante. Para resolução desse problema foi definida a trajetória do nó emissor para que não passe por cima de algum nó âncora e passe somente entre eles, por esse motivo que o modelo circular a posição máxima de  $x$  dele é no ponto 108 m e 100 m.

Todo esse processo também foi realizado incluindo o ruído proveniente das medições de tempo baseadas no *clock* do processador. O tempo é medido de acordo com o período do *clock* do processador, sendo assim, em um cenário real, quando o nó enviar ou receber a mensagem pode acontecer de enviar no tempo  $t$  e por conta da velocidade do processador o tempo medido e utilizado para o cálculo ser  $t + 1$ , no caso o tempo da máquina estaria atrasado em relação ao tempo real.

Como têm-se dois tempos para o cálculo da distância, o tempo inicial e final, foi configurado o ruído gerando um número discreto a partir de uma distribuição normal entre  $-2$  e  $2$  vezes o período do *clock* do hardware ESP32<sup>1</sup>, que é de 0,0000000125 s. Essa distribuição foi escolhida para representar as diferentes situações no momento da medição de tempo, uma vez que os dois tempos podem estar adiantados em relação ao tempo real, ou os dois podem estar atrasados, ou um pode estar adiantado apenas ou o outro atrasado apenas ou os dois podem estar corretos.

Por fim, para facilitar a execução de todos esses cenários com diversas *seeds*, foi criado um shell script contendo as linhas para executar a simulação via terminal demonstrado pelo Algoritmo 1. Dessa forma, não foi necessário configurar manualmente a execução da simulação.

---

<sup>1</sup>ESP32 é uma placa de desenvolvimento com vários microcontroladores e possui tecnologias WiFi e Bluetooth.



```

1 velocidade=('1ms' '15ms' '30ms')
2 qtde_nos=('3h' '6h' '9h')
3 tipo_movimento=('linear' 'Circle' 'Random')
4
5 for((i_velocidade=0; i_velocidade<${#velocidade[@]}; i_velocidade++));
6 do
7   for((i_qtde=0; i_qtde<${#qtde_nos[@]}; i_qtde++)); do
8     for((i_movimento=0; i_movimento<${#tipo_movimento[@]}; i_movimento
9       ++)); do
10      comando='./rtof1 -u Cmdenv -c Wireless_`${tipo_movimento[
11        i_movimento]}`${qtde_nos[i_qtde]}`${velocidade[i_velocidade]}
12      echo $comando
13      done;
14    done;
15  done;
16 done;

```

Algoritmo 1 – Script para rodar a simulação.

### 3.4 Geração dos gráficos

Após os arquivos .csv criados com os dados das posições dos nós âncoras, da estimativa de posição do nó emissor, da posição real do nó emissor, o tempo de simulação e o erro do MLE (*Maximum Likelihood Estimation*) na estimativa de posição do nó emissor. Dessa forma, foram gerados os gráficos de comparação das médias entre os cenários utilizando a linguagem *Python* com auxílio da plataforma *Jupyter Notebook*.

Sendo assim, através da plataforma *Jupyter Notebook* foram implementados os códigos para gerar os gráficos adequados. Para isso, de início foram importados todos os arquivos .csv utilizando a biblioteca *Pandas*, obtendo-se assim os *dataframes*, que são estruturas de dados tabular bidimensionais, ou seja, os dados retirados poderiam ser manipulados dentro do código através de uma tabela.

Como cada *dataframe* contém informações de todas as repetições com diferentes *seeds* e as médias do erro euclidiano, foi necessário retirar de cada *dataframe* apenas as médias, montando uma nova tabela com todos os cenários conforme exemplificado pela Tabela 3. Para cada tipo de movimento foi gerado uma nova tabela utilizando o mesmo processo.

Para que não seja necessária a realização de todo o processo novamente para os próximos movimentos, criou-se também uma função dentro do *Jupyter Notebook* em que a tabela com os dados brutos é passada com o intuito de retornar uma lista com todas as médias daquele movimento, assim com essa lista pode ser criada uma nova tabela apenas com estas médias, de acordo com o Algoritmo 2.

A partir da tabela contendo os dados tratados foram gerados os gráficos de barras com o auxílio da biblioteca *Seasborn* e *Matplotlib*. A biblioteca *Seasborn* foi responsável pela geração dos gráficos e a *Matplotlib* permitiu editar fontes e criar na mesma imagem a tabela com a legenda do gráfico. A barra de intervalo de confiança foi gerada pela própria biblioteca apenas programando a porcentagem do nível de confiança de 95% através do parâmetro *ci*.

Tabela 3 – Exemplo da tabela para gerar os gráficos.

Wireless3h 1ms Linear	Wireless3h 15ms Linear	...	Wireless9h 15ms Linear	Wireless9h 30ms Linear
0,001590	0,022216	...	0,051178	0,101404
0,001407	0,022916	...	0,055475	0,102469
0,001555	0,022158	...	0,051071	0,101810
0,001557	0,022825	...	0,054303	0,107634
0,001550	0,022363	...	0,054999	0,110844
0,001501	0,021619	...	0,053791	0,099256

Fonte: Autoria própria.

```

1 def media_erro_df(df):
2     vetor_media=[]
3     somas_media=0
4     cont=0
5
6     for i in range(len(df)):
7         if(df.loc[i,"Likelihood position X (m)"] == 'Media'):
8             vetor_media.append(df.loc[i,"Erro"])
9             somas_media=somas_media+df.loc[i,"Erro"]
10            cont=cont+1
11    media=somas_media/cont
12    return media, vetor_media

```

Algoritmo 2 – Função para retirar apenas as médias do *dataframe* bruto.

## 4 RESULTADOS

Após gerados os gráficos, foi analisado o comportamento do algoritmo *RToF* e da solução de trilateração MLE comparando os diversos cenários dentro de um único movimento, incluindo os resultados da simulação sem ruídos, que, por mais que sejam ideais, estão longe de acontecer em ambiente real e dos resultados da simulação com ruídos. Tal análise foi realizada com a finalidade de conferir a influência da velocidade e da quantidade de nós âncoras na estimativa de posição. Verificou-se também o comparativo entre os movimentos linear, circular e aleatório.

### 4.1 Movimento Linear

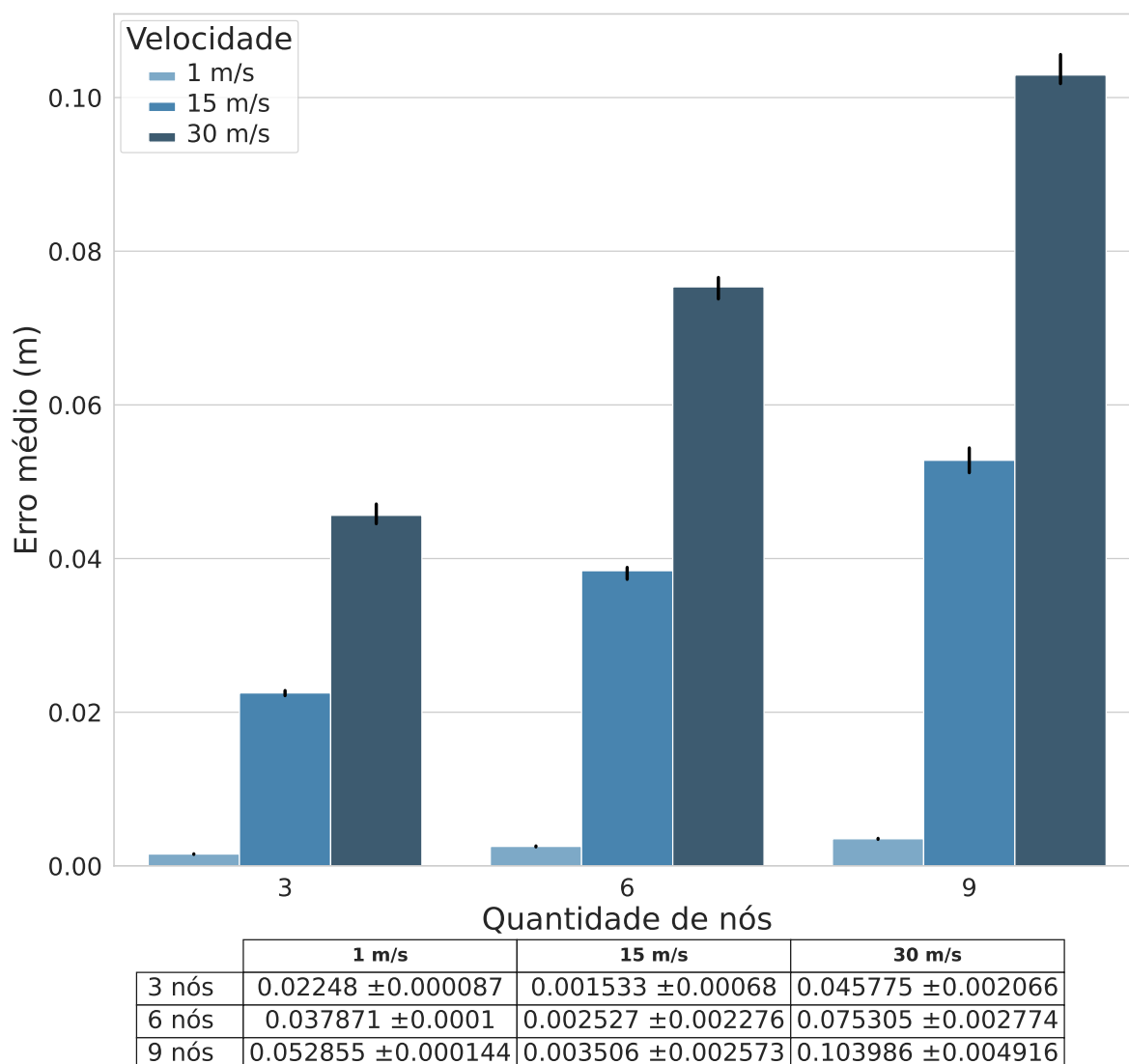
Analisando o comportamento da performance do algoritmo utilizando o movimento linear sem ruído, pode-se perceber pela Figura 8 que o erro médio foi aumentando conforme aumentou-se a velocidade do nó emissor. Isso aconteceu devido ao fato de que a posição do nó emissor no momento do início da transmissão das mensagens é diferente da posição no momento do cálculo para se estimar a posição, já que o nó está se movimentando. O cálculo de estimativa de posição só acontece depois que todos os nós receptores respondem a mensagem. Então, como o nó emissor envia a mensagem e continua se movendo, a mensagem de resposta leva um tempo para ser recebida ocasionando o erro demonstrado, visto que nesse caso não tem ruído de outras fontes. O único fator que causa esse erro é a velocidade, ou seja, esse erro é proporcional ao tempo em que a mensagem de resposta demorou para ser recebida multiplicado pela velocidade que o nó está se movendo, resultando na distância percorrida pelo nó enquanto aguardava as mensagens de resposta.

Outro fato que percebeu-se nesse cenário sem ruído foi o aumento do erro em consequência da quantidade de nós, resultado que não era esperado. Isso ocorreu por cada nó apresentar um erro contido no cálculo da distância devido ao fator da velocidade explicado anteriormente, a qual no momento do cálculo da estimativa de posição esses erros são associados.

Comparando com o resultado da estimativa de posição sendo incluído o ruído, já pode-se perceber através da Figura 9 que o erro diminui conforme aumenta a quantidade de nós, e percebe-se que a velocidade praticamente não influencia para o erro do algoritmo. Isso se deve pelo erro causado pelo ruído ser muito maior que o erro causado pela variação da velocidade, sendo imperceptível o erro causado pela movimentação do nó. Dessa forma, conforme aumenta a quantidade de nós o erro da estimativa de posição diminui, podendo concluir que para essa causa do erro, quanto maior for a quantidade de nós, menor vai ser o erro por ter mais medidas de distâncias para se realizar o cálculo da trilateração levando o algoritmo a uma melhor precisão.

Por fim, foi levantada a hipótese de que a posição do nó emissor poderia influenciar

Figura 8 – Comparação do erro no cenário utilizando movimento linear sem ruído.

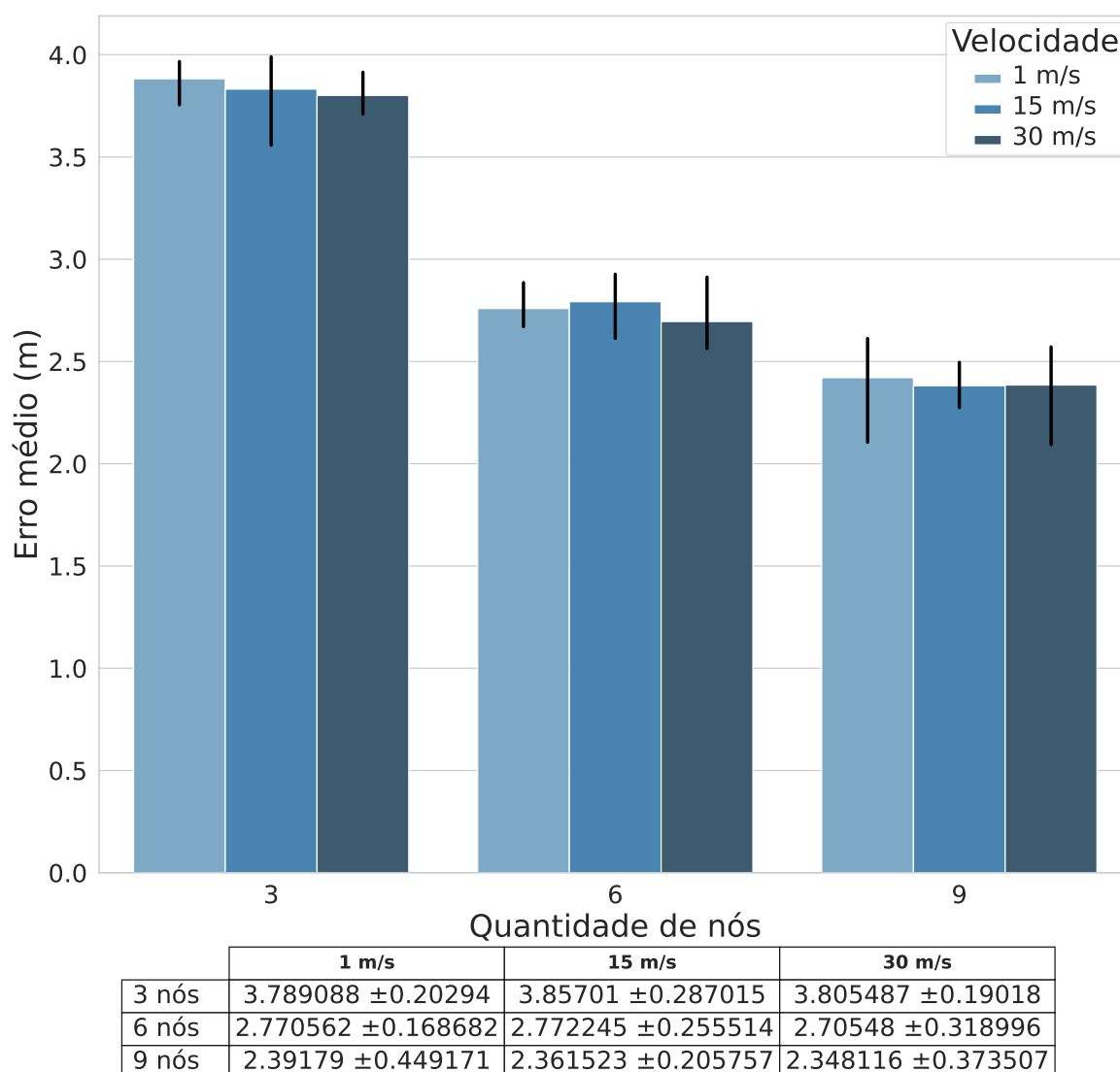


Fonte: Autoria própria.

na estimativa de posição, acreditou-se que talvez no momento em que o nó emissor passe pelo centro dos nós âncoras o algoritmo iria apresentar um erro menor. Por isso, foi plotado dois gráficos de linhas do cenário com 3 nós na velocidade 1 m/s com movimento linear e com ruído.

Sendo assim, conforme ilustrado pelas Figuras 10 e 11 nota-se a variação da posição de acordo com o decorrer do tempo em que a simulação está sendo executada, percebendo-se que não existe um ponto em que o erro é minimizado nitidamente, e que o ruído não cresce de acordo com o tempo de simulação. Porém percebe-se que o ruído interfere bastante na estimativa de posição. Por estes motivos a hipótese inicial acabou não sendo verdadeira para essa configuração do experimento.

Figura 9 – Comparação do erro no cenário utilizando movimento linear com ruído.



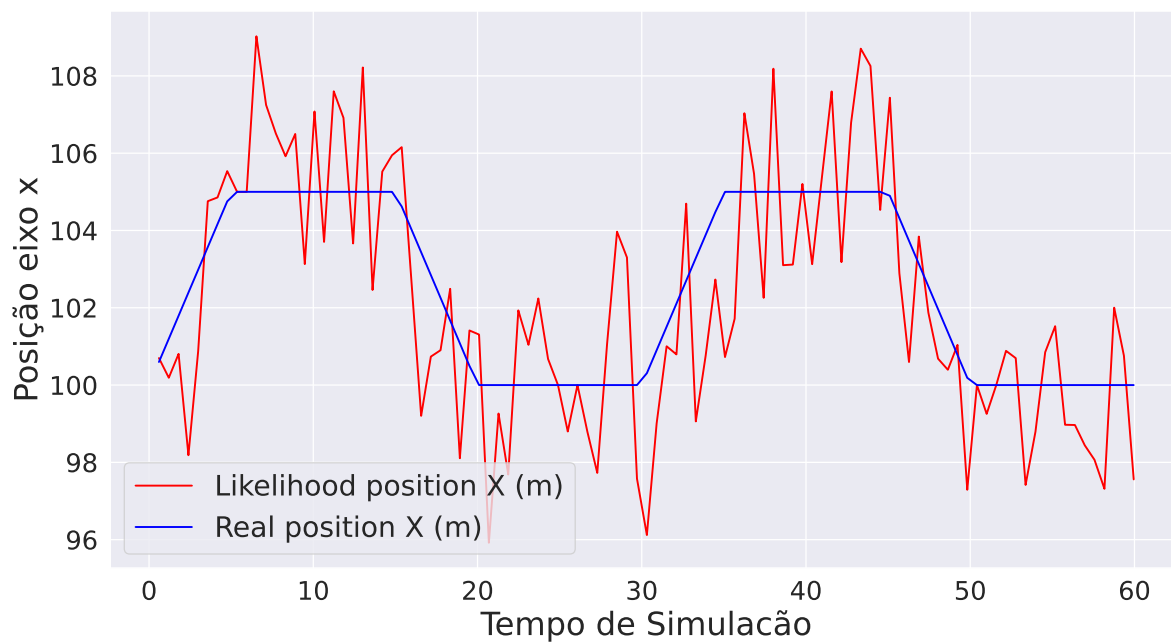
Fonte: Autoria própria.

## 4.2 Movimento Circular

A análise do comportamento do algoritmo quando o nó emissor realiza o trajeto circular passando pelo centro dos nós âncoras, percebe-se pela Figura 12 que para o cenário que não contém ruído, o único erro que aparece visivelmente é gerado pela alteração das velocidades, conforme discutido na seção 4.1, sendo variáveis diretamente proporcionais a velocidade e o erro.

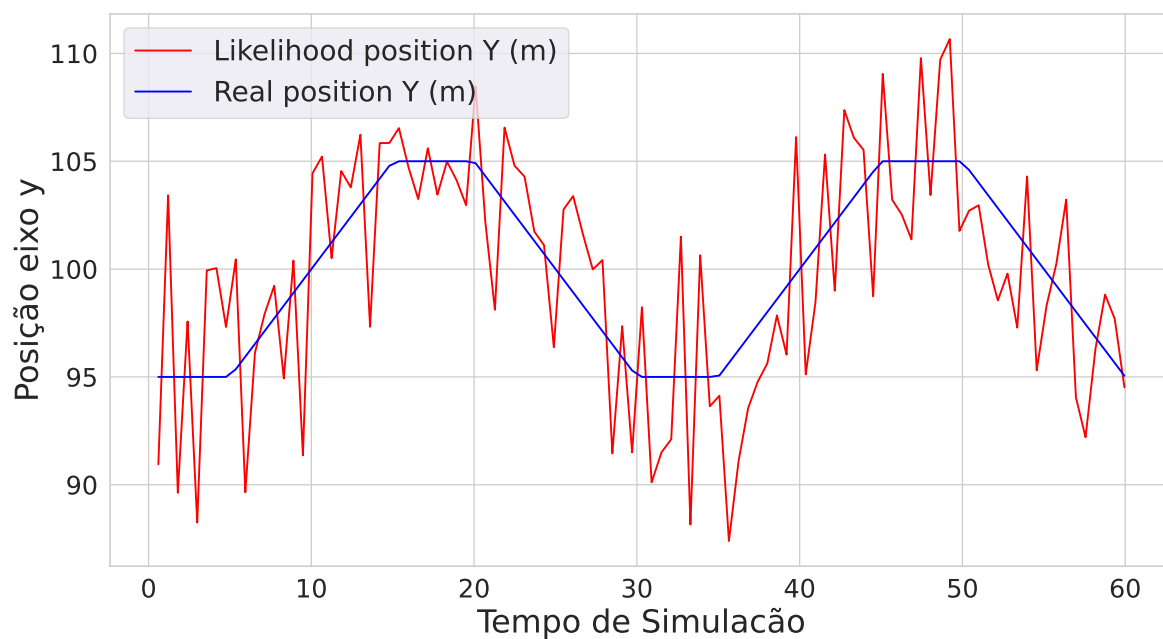
Já no cenário que contém ruído pode-se perceber pela Figura 13 que a influência da velocidade diminui, visto que, comparando cenários em que a quantidade de nós são iguais a diferença do erro entre elas é muito baixa, sendo praticamente iguais pelas médias dos erros dos cenários de mesma quantidade de nós estarem dentro do intervalo de confiança das outras, tornando sua eficiência estatisticamente iguais. Já a quantidade de nós tem grande influência na estimativa de posição, pois conforme aumenta-se a quantidade dos nós âncoras, diminui-se

Figura 10 – Comparação da posição estimada com a posição real no eixo x no cenário com ruído, movimento linear, velocidade de 1 m/s no nó emissor e com 3 nós âncoras.



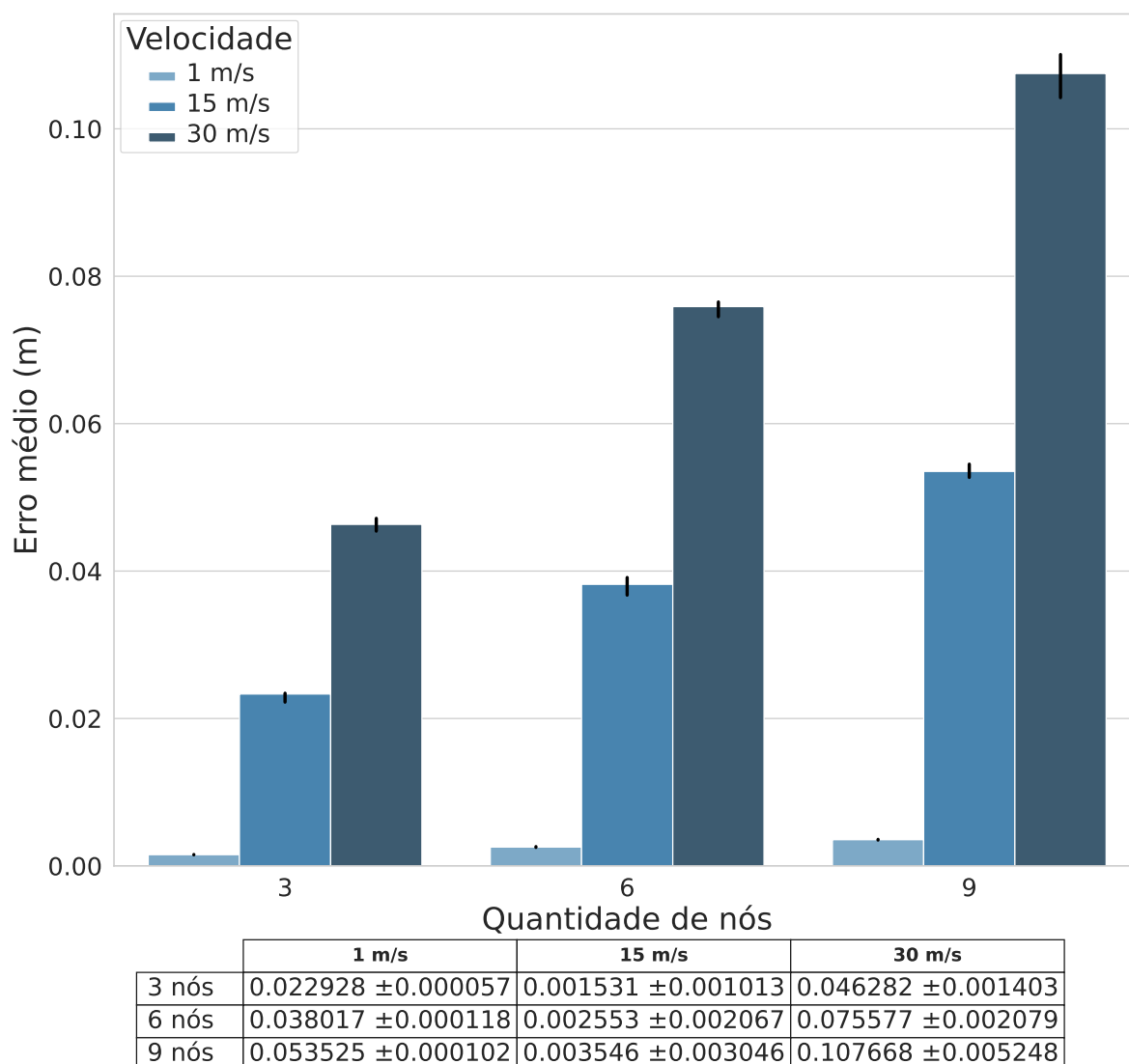
Fonte: Autoria própria.

Figura 11 – Comparação da posição estimada com a posição real no eixo y no cenário com ruído, movimento linear, velocidade de 1 m/s no nó emissor e com 3 nós âncoras.



Fonte: Autoria própria.

Figura 12 – Comparação do erro no cenário utilizando movimento circular sem ruído.



Fonte: Autoria própria.

o erro da estimativa de posição.

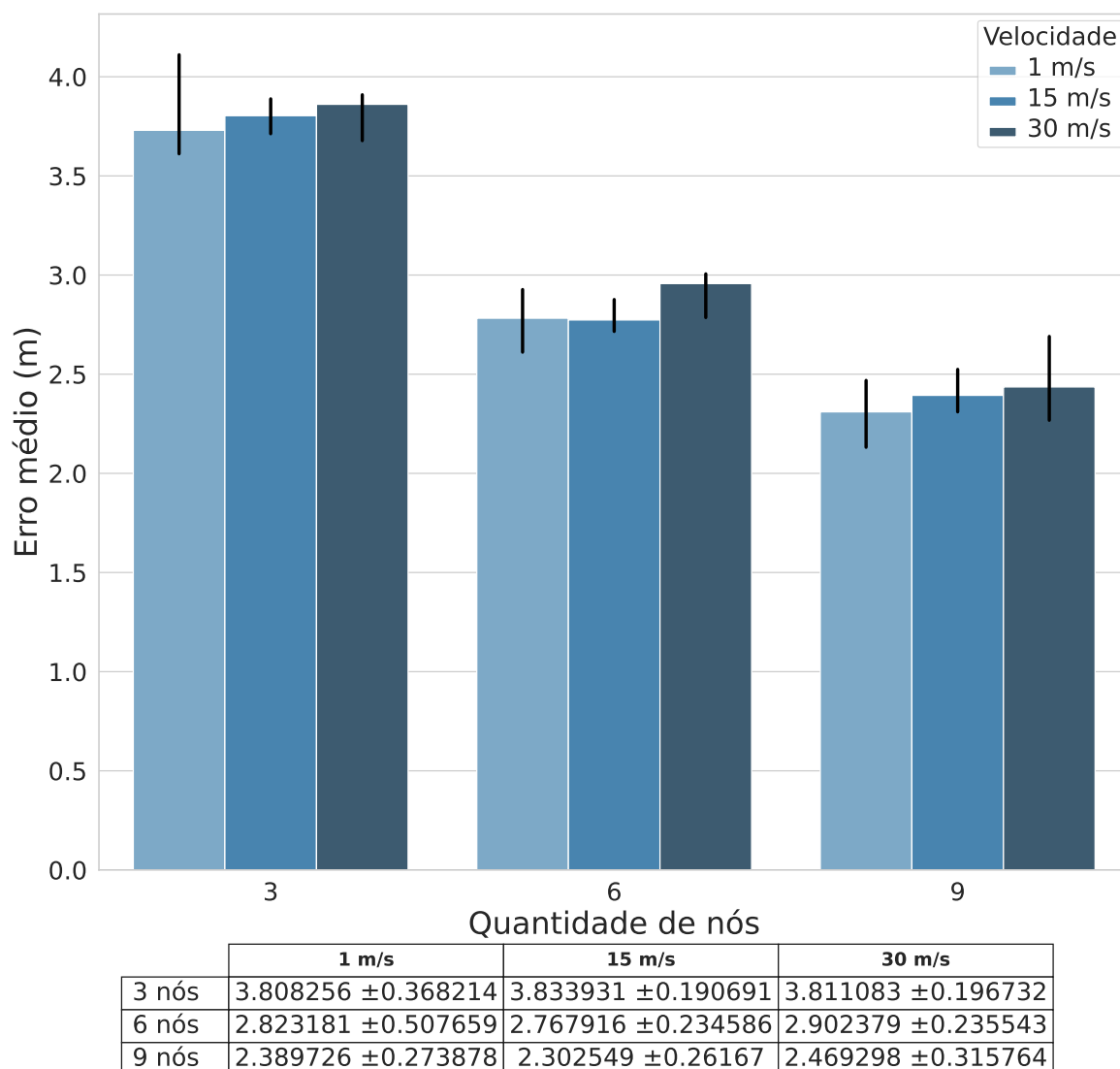
Analisando o percurso do nó emissor durante o tempo de simulação utilizando o movimento circular no nó emissor, pode-se perceber pelas Figuras 14 e 15 que apesar do ruído influenciar na estimativa de posição, a mesma está sempre dentro de um intervalo específico, demonstrando ser uma boa solução mesmo após passado um longo tempo de simulação.

#### 4.3 Movimento aleatório

Observando o movimento aleatório com ausência do ruído, cujos resultados são apresentados na Figura 16, percebe-se que acontece o fenômeno semelhante ao ocorrido nas seções 4.1 e 4.2, em que o erro aumenta de acordo com o crescimento da velocidade e da quantidade de nós âncoras.

Visualizando o gráfico do cenário em que o nó se movimenta aleatoriamente contendo

Figura 13 – Comparação do erro no cenário utilizando movimento circular com ruído.



Fonte: Autoria própria.

o ruído, representado pela Figura 17 nota-se que ele tem grande influência na estimativa de posição, visto que o erro diminui conforme o aumento da quantidade de nós, de forma semelhante aos movimentos linear e circular.

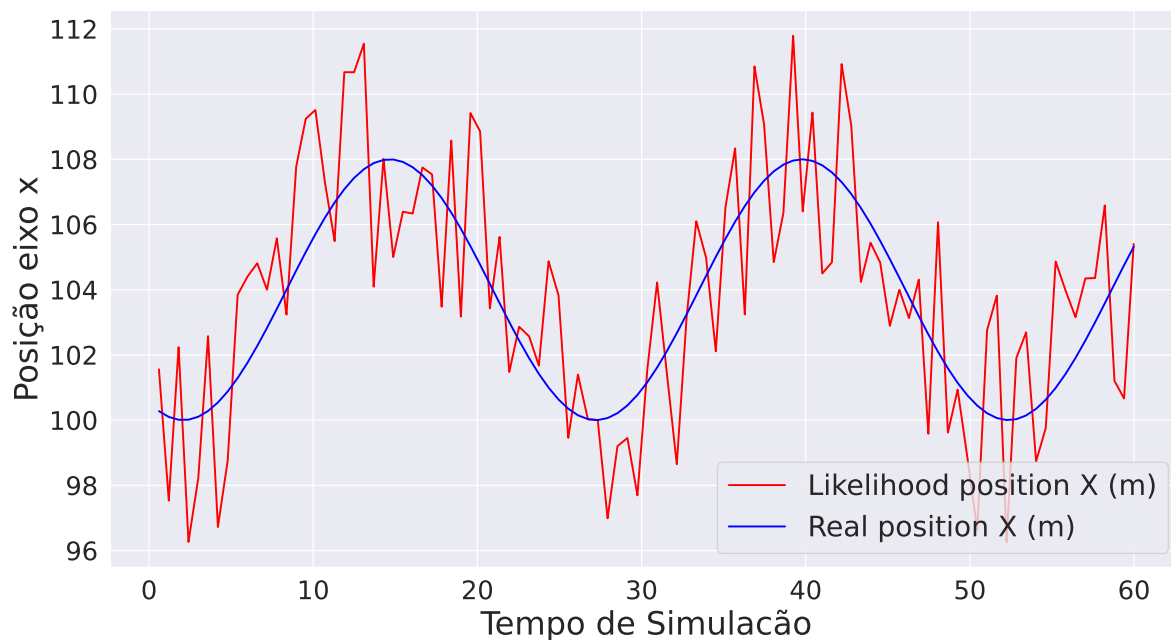
Verificando o percurso do nó emissor ao longo do tempo de simulação utilizando o movimento aleatório pode-se notar pelas Figuras 18 e 19 que a estimativa de posição com ruído não diminui ao longo do tempo, ou seja, não há uma certa posição que justifique uma melhor performance do algoritmo.

#### 4.4 Comparação entre os movimentos

Realizando a comparação entre os tipos de movimentos diferentes com intuito de verificar se o tipo de movimento interfere na estimativa de posição foi utilizado todos no cenário com 3 nós âncoras sendo o nó emissor com velocidade de 30 m/s sem a utilização do ruído,

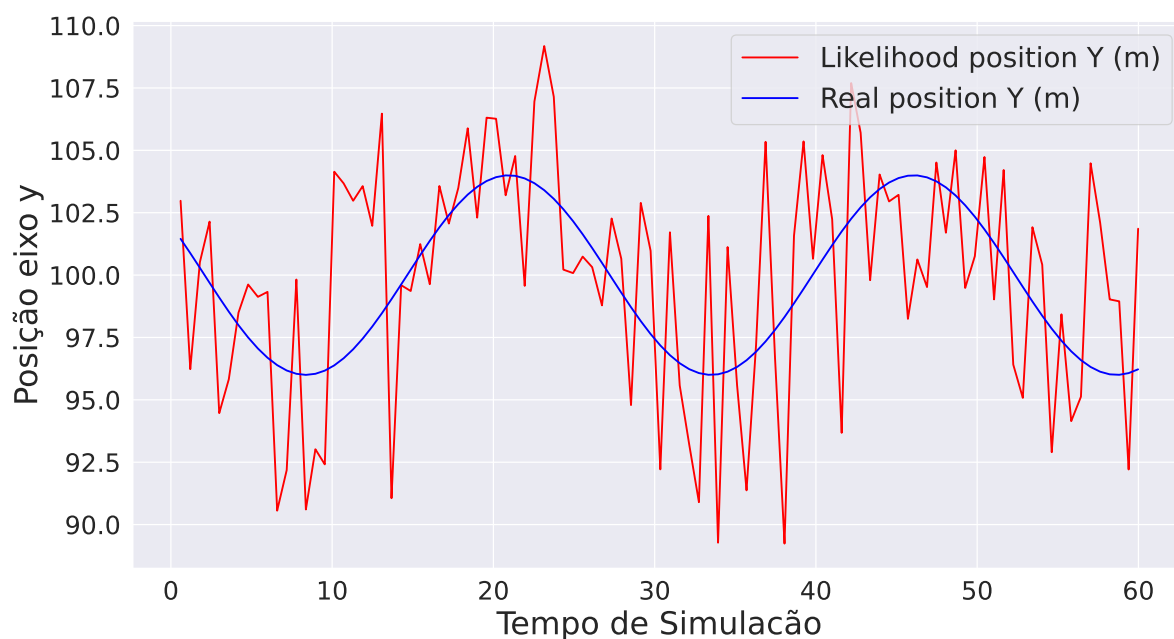


Figura 14 – Comparação da posição estimada com a posição real no eixo x no cenário com ruído, movimento circular, velocidade de 1 m/s no nó emissor e com 3 nós âncoras.



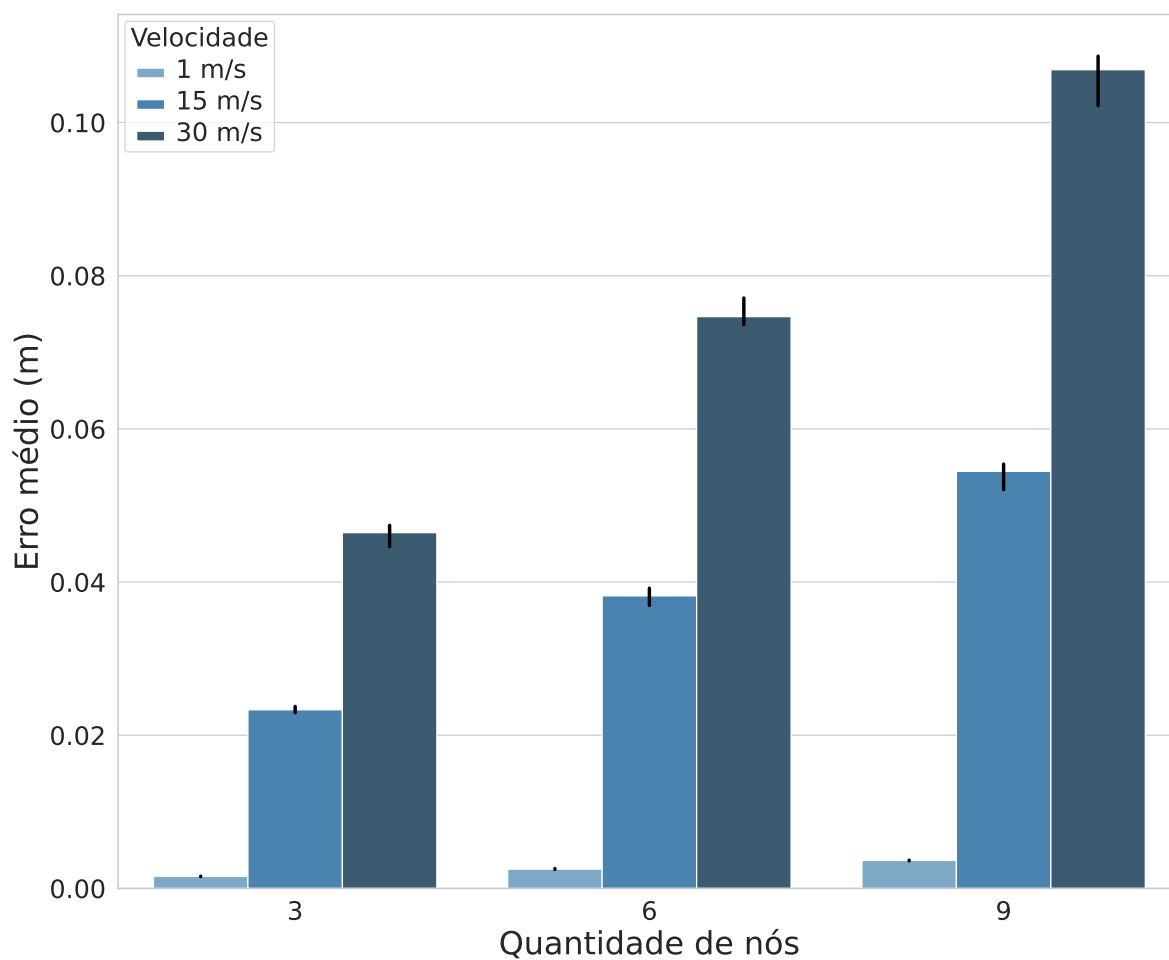
Fonte: Autoria própria.

Figura 15 – Comparação da posição estimada com a posição real no eixo y no cenário com ruído, movimento circular, velocidade de 1 m/s no nó emissor e com 3 nós âncoras.



Fonte: Autoria própria.

Figura 16 – Comparação do erro no cenário utilizando movimento aleatório sem ruído.

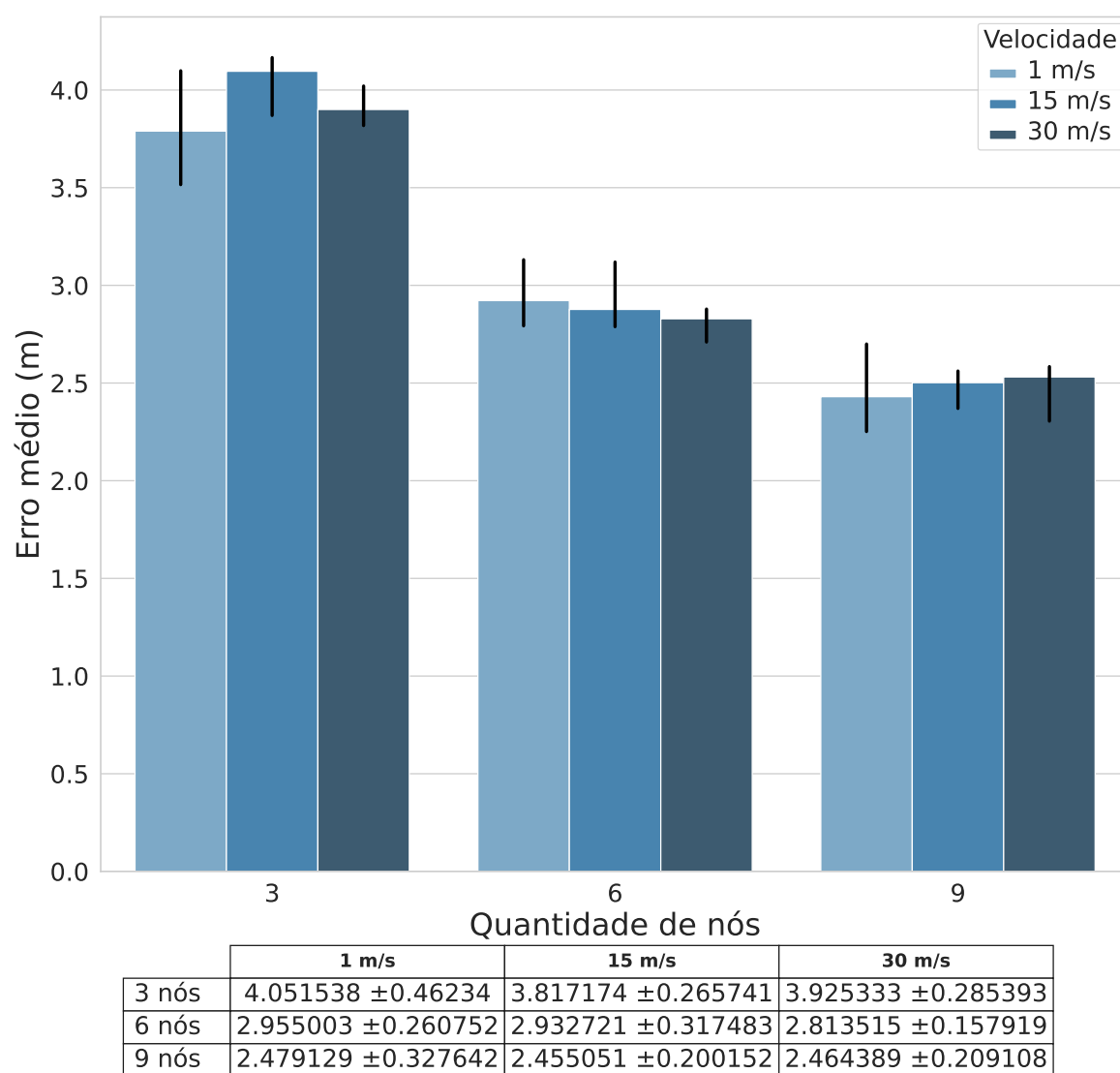


	1 m/s	15 m/s	30 m/s
3 nós	0.023419 ±0.000058	0.001564 ±0.000815	0.046228 ±0.001974
6 nós	0.038185 ±0.000113	0.002545 ±0.002083	0.075235 ±0.003246
9 nós	0.053909 ±0.000189	0.003622 ±0.002899	0.105623 ±0.005743

Fonte: Autoria própria.

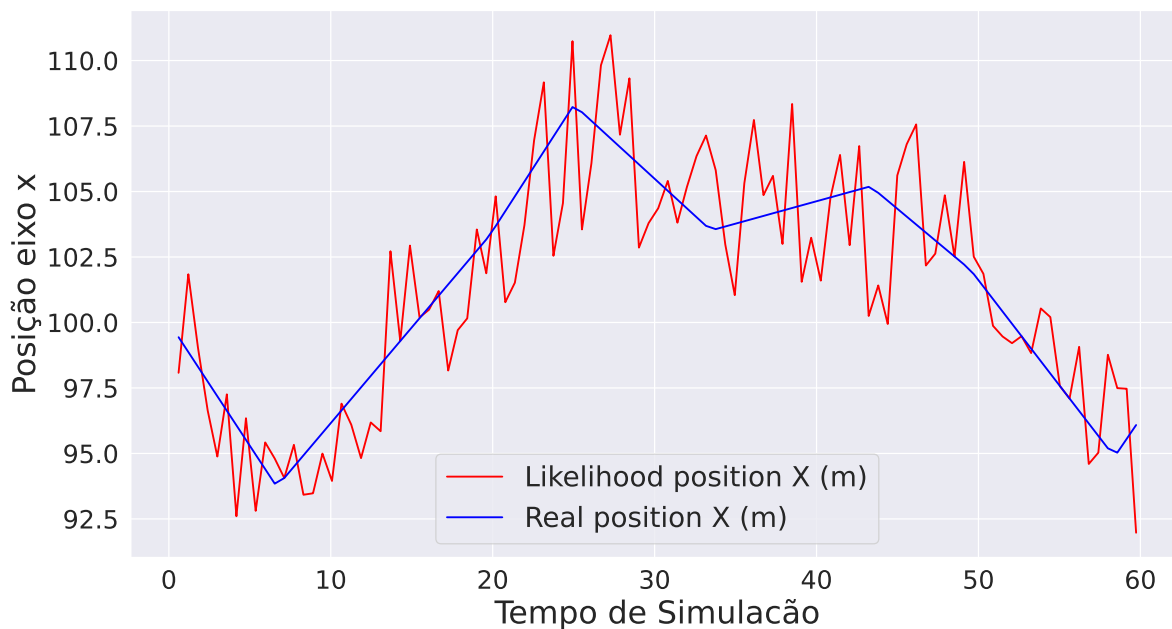
percebe-se através da Figura 20 que não há uma nítida influência do tipo de movimento na estimativa de posição, uma vez que, os erros se aproximam muito comparando diferentes tipos de movimentos. O mesmo acontece quando o ruído é incluído, conforme nota-se pela Figura 21, porém o erro acaba sendo maior do que sem ruído.

Figura 17 – Comparação do erro no cenário utilizando movimento aleatório com ruído.



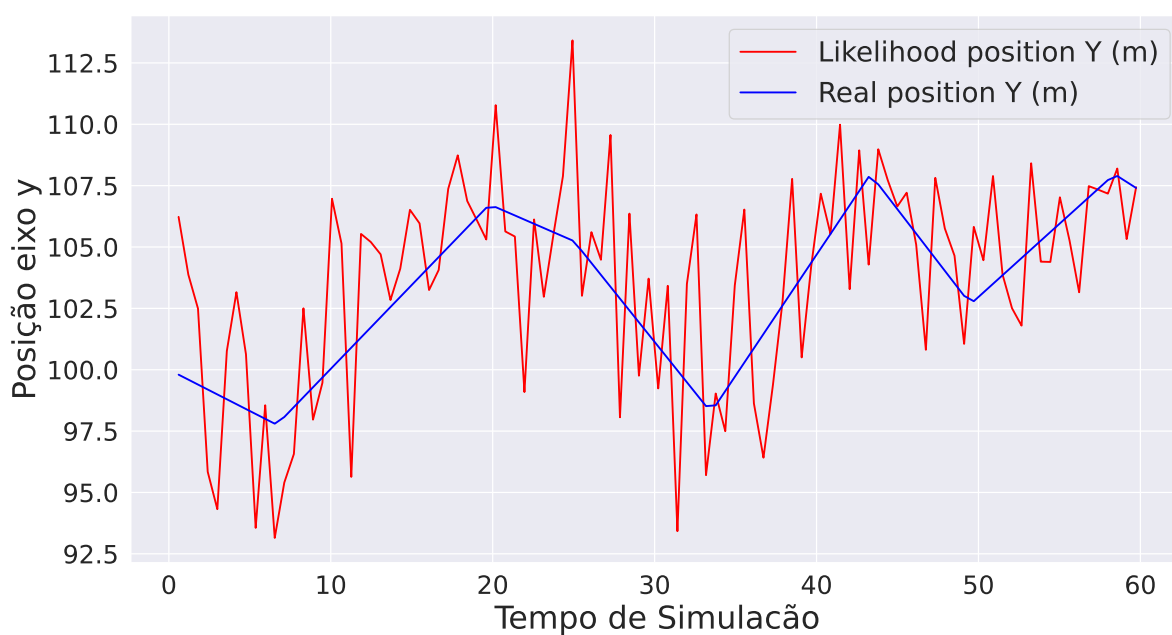
Fonte: Autoria própria.

Figura 18 – Comparação da posição estimada com a posição real no eixo x no cenário com ruído, movimento aleatório, velocidade de 1 m/s no nó emissor e com 3 nós âncoras.



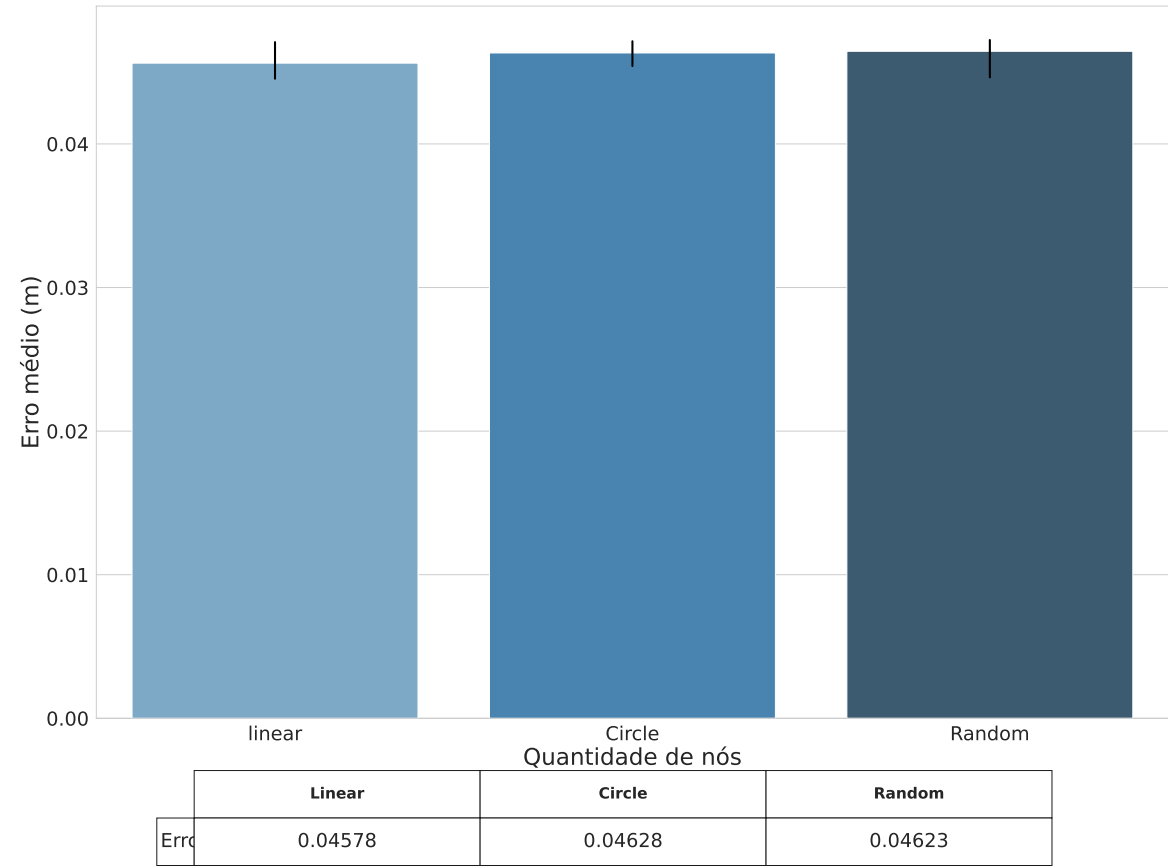
Fonte: Autoria própria.

Figura 19 – Comparação da posição estimada com a posição real no eixo y no cenário com ruído, movimento aleatório, velocidade de 1 m/s no nó emissor e com 3 nós âncoras.



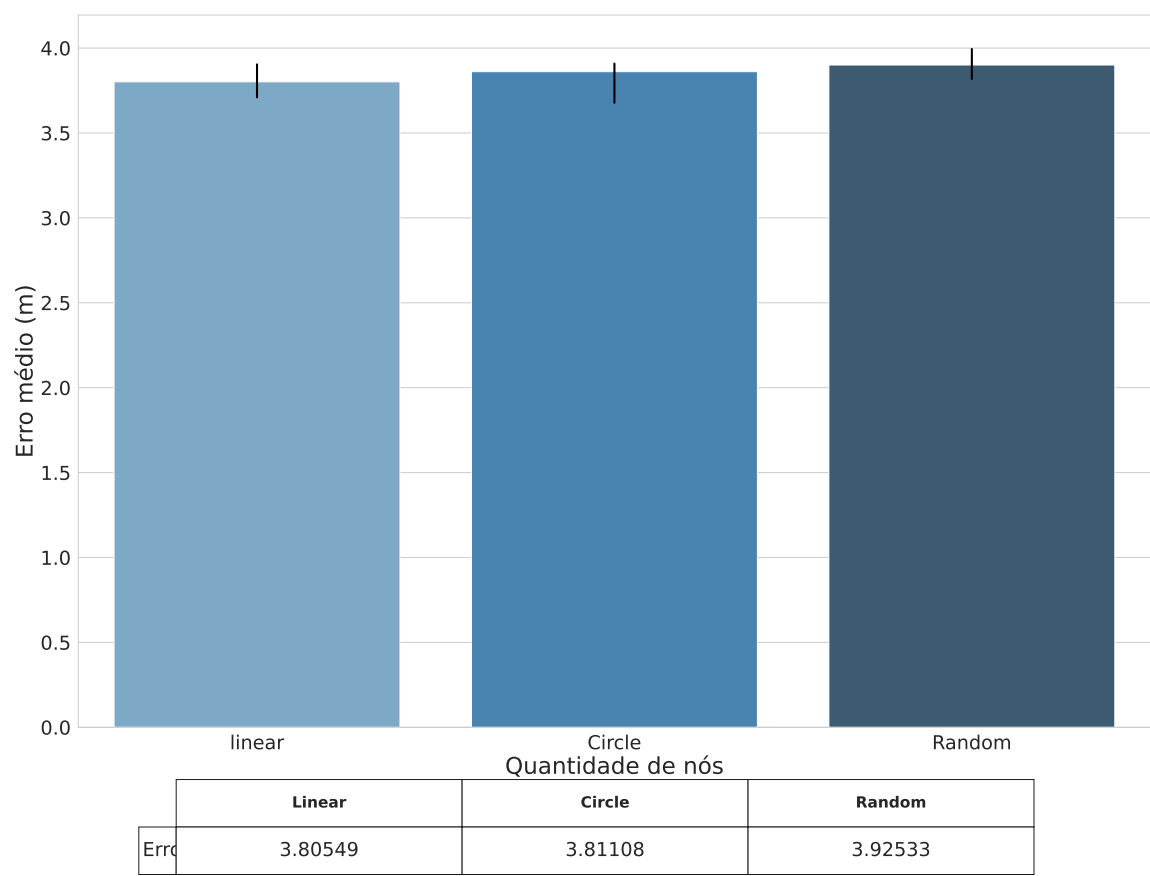
Fonte: Autoria própria.

Figura 20 – Comparação entre os movimentos dos nós.



Fonte: Autoria própria.

Figura 21 – Comparação entre os movimentos dos nós com a presença do ruído.



Fonte: Autoria própria.

## 5 CONCLUSÃO

Conforme apresentado no decorrer do presente trabalho, o RToF é uma importante técnica de geolocalização, se fazendo necessário o estudo aprofundado especialmente focado em sua performance quando o objeto a ser localizado encontra-se em movimento, o que devido a variação de posição acarreta em um erro muito grande, uma vez que, a velocidade interfere diretamente na estimativa de posição.

Em conjunto com o Omnet++ também foi utilizado o *framework* INET que possibilitou a criação da rede WLAN. Apesar da comunicação através do *framework* ocorrer de forma adequada, foi necessária a realização de alguns ajustes no protocolo CSMA/CA para a mensagem poder enviar a informação do tempo de espera dentro do *header*.

Apesar de já ter sido feito o estudo da implementação da mobilidade através deste trabalho, o mesmo contemplou a aplicação da mobilidade apenas em ambiente simulado possuindo algumas limitações em relação ao ambiente real.

Dessa forma, a verificação da viabilidade técnica em ambiente real se torna imprescindível para a validação da utilização como auxílio na geolocalização de VANTs em aplicativos que demandam saber a posição do objeto de forma precisa, ou até mesmo para fins de entretenimento.

Portanto, através dos resultados obtidos percebeu-se que o tipo de movimento não influencia no algoritmo *RToF* utilizando a técnica de trilateração MLE, o que já não é o caso da velocidade, já que no cenário sem ruído apresentou bastante alterações no cálculo de estimativa de posição. A quantidade de nós também causou interferências nos resultados, sendo que no cenário sem ruído o algoritmo apresentou uma pior performance no aumento de nós âncoras, ao contrário do cenário com ruído, em que o aumento de nós âncoras melhorou a acurácia do algoritmo.

Notou-se também que o algoritmo no cenário com pior performance, ou seja, o cenário com 3 nós âncoras, velocidade de 30 m/s e se movendo linearmente, que o erro no cálculo da trilateração realizado pela técnica MLE obteve valor máximo de 10,274 m aproximadamente mas com uma média de 3,8055 aproximadamente, tornando a técnica RToF em conjunto com o MLE uma solução viável para aplicações com fins de entretenimento. Por outro lado, para veículos autônomos que requerem uma maior precisão, visto que lidam com vidas humanas, talvez seja necessário a utilização de outras tecnologias em conjunto com a que foi estudada, sendo essa conclusão realizada analisando o RToF e o MLE apenas em ambiente simulado.

O presente trabalho foi realizado no ambiente de simulação Omnet++, que apesar de ser uma boa ferramenta para simular os eventos, apresenta algumas deficiências, como por exemplo, não ter um ruído já implementado.

## 5.1 TRABALHOS FUTUROS

Utilizando o simulador pode-se realizar outros experimentos para avaliar a técnica RToF em conjunto com o MLE tentando realizar a correção do erro causado pela velocidade via software, já que tem-se a velocidade e tempo em que a mensagem é enviada pelo nó emissor e uma resposta é recebida. Sendo assim, pode ser calculada a distância percorrida pelo nó enquanto transmitia a mensagem, se também for conhecida a direção do seu movimento. Os valores de distância entre os nós fixos também podem ser extrapolados para estarem dentro de uma área de  $1000m^2$  por exemplo.

Outro trabalho que poderá ser realizado é avaliar a técnica de trilatação MLE, por ela utilizar matrizes para seu cálculo e em uma das etapas utiliza-se a matriz inversa, essa operação é muito custosa podendo causar interferência na estimativa de posição em um ambiente real, uma vez que, conforme o aumento de número de nós âncoras, aumenta-se a complexidade da operação. Sendo assim, a operação irá levar mais tempo para ser realizada podendo ocasionar em um erro maior na estimativa de posição.

Por fim, seria importante reaplicar os experimentos realizados em ambiente de simulação para ambientes reais, afim de avaliar se os mesmos comportamentos são observados em cenário real, uma vez que, o ambiente simulado está longe de ser um caso real.



## Referências

- BAHILLO, A. et al. Hybrid rss-rtt localization scheme for wireless networks. In: **2010 International Conference on Indoor Positioning and Indoor Navigation**. [S.l.: s.n.], 2010. p. 1–7. Citado 2 vezes nas páginas 10 e 11.
- BAJAJ, R.; RANAWEERA, S. L.; AGRAWAL, D. P. Gps: location-tracking technology. **Computer**, IEEE, v. 35, n. 4, p. 92–94, 2002. Citado na página 1.
- BIANCHI, G.; FRATTA, L.; OLIVERI, M. Performance evaluation and enhancement of the csma/ca mac protocol for 802.11 wireless lans. In: IEEE. **Proceedings of PIMRC'96-7th International Symposium on Personal, Indoor, and Mobile Communications**. [S.l.], 1996. v. 2, p. 392–396. Citado na página 4.
- BUSIC, L.; FILJAR, R.; DESIC, S. Assisted satellite positioning. In: IEEE. **47th International Symposium ELMAR, 2005**. [S.l.], 2005. p. 263–267. Citado na página 1.
- HALDER, S.; GHOSAL, A. A survey on mobility-assisted localization techniques in wireless sensor networks. **Journal of Network and Computer Applications**, Elsevier, v. 60, p. 82–94, 2016. Citado na página 8.
- HAN, K. et al. Exploiting user mobility for wifi rtt positioning: A geometric approach. **IEEE Internet of Things Journal**, v. 8, n. 19, p. 14589–14606, 2021. Citado 3 vezes nas páginas 9, 10 e 11.
- HIERTZ, G. R. et al. The ieee 802.11 universe. **IEEE Communications Magazine**, IEEE, v. 48, n. 1, p. 62–70, 2010. Citado na página 4.
- HIGHWAY Capacity Manual. [S.l.]: Transportation Research Board, National Research Council, 2000. Citado na página 12.
- HUILLA, S. et al. Smartphone-based indoor positioning using wi-fi fine timing measurement protocol. **MSc Thesis**, University of Turku, 2019. Citado na página 5.
- IEEE. IEEE standard for information technology – telecommunications and information exchange between systems local and metropolitan area networks – specific requirements – part 11: Wireless lan medium access control (MAC) and physical layer (PHY) specifications. **IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012)**, p. 1–3534, 2016. Citado na página 5.
- KAN, K.; CHAN, S.; NG, J.-Y. A dual-channel location estimation system for providing location services based on the gps and gsm networks. In: **17th International Conference on Advanced Information Networking and Applications, 2003. AINA 2003**. [S.l.: s.n.], 2003. p. 7–12. Citado na página 1.
- KESTRING, B. A. Sistema móvel na plataforma android para compartilhamento de geolocalização usando mapas e notificações da google. 2014. Citado na página 1.
- KUUTTI, S. et al. A survey of the state-of-the-art localization techniques and their potentials for autonomous vehicle applications. **IEEE Internet of Things Journal**, IEEE, v. 5, n. 2, p. 829–846, 2018. Citado na página 1.

LOPES, I. M.; BENEVENUTO, F. M.; OLIVEIRA, F. M. de. As camadas do modelo osi: Revisitando suas funcionalidades e respectivos protocolos. **Revista Interdisciplinar Pensamento Científico**, v. 1, n. 2, 2015. Citado na página 4.

MA, C. et al. Wi-fi rtt ranging performance characterization and positioning system design. **IEEE Transactions on Mobile Computing**, IEEE, 2020. Citado 2 vezes nas páginas 1 e 6.

MACEDO, R. T.; FRANCISCATTO, G. B. d. C. R.; BERTOLINI, C. **Redes de Computadores**. [S.l.]: UNIVERSIDADE FEDERAL DE SANTA MARIA, 2018. Citado na página 3.

MURPHY, W.; HEREMAN, W. Determination of a position in three dimensions using trilateration and approximate distances. **Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado, MCS-95**, v. 7, p. 19, 1995. Citado na página 7.

OMNETPP. **CircleMobility**. 2021. Disponível em: <<https://doc.omnetpp.org/inet/api-current/neddoc/inet.mobility.single.CircleMobility.html>>. Acesso em: 15 de setembro de 2021. Citado na página 9.

OMNETPP. **LinearMobility**. 2021. Disponível em: <<https://doc.omnetpp.org/inet/api-current/neddoc/inet.mobility.single.LinearMobility.html>>. Acesso em: 14 de setembro de 2021. Citado na página 8.

OMNETPP. **Node Mobility**. 2021. Disponível em: <<https://inet.omnetpp.org/docs/users-guide/ch-mobility.html>>. Acesso em: 08 de agosto de 2021. Citado na página 2.

OMNETPP. **RandomWaypointMobility**. 2021. Disponível em: <<https://doc.omnetpp.org/inet/api-current/neddoc/inet.mobility.single.RandomWaypointMobility.html>>. Acesso em: 17 de setembro de 2021. Citado na página 9.

PARK, S.; AHN, H.-S.; YU, W. Round-trip time-based wireless positioning without time synchronization. In: **2007 International Conference on Control, Automation and Systems**. [S.l.: s.n.], 2007. p. 2323–2326. Citado 2 vezes nas páginas 10 e 11.

PERAHIA, E. Ieee 802.11 n development: History, process, and technology. **IEEE Communications Magazine**, IEEE, v. 46, n. 7, p. 48–55, 2008. Citado na página 4.

ROXIN, A. et al. Survey of wireless geolocation techniques. In: IEEE. **2007 IEEE Globecom Workshops**. [S.l.], 2007. p. 1–9. Citado na página 1.

SAEED, A.; NASEEM, S. F.; ZAIDI, Z. R. Mobility estimation for wireless networks using round trip time (rtt). In: **2007 6th International Conference on Information, Communications Signal Processing**. [S.l.: s.n.], 2007. p. 1–5. Citado 2 vezes nas páginas 10 e 11.

SILVA, R.; SILVA, J. S.; BOAVIDA, F. Mobility in wireless sensor networks—survey and proposal. **Computer Communications**, Elsevier, v. 52, p. 1–20, 2014. Citado na página 8.

SINGH, B. et al. A survey of cellular positioning techniques in gsm networks. In: **National Conference in Mobile Computing-NCMC'12**. [S.l.: s.n.], 2012. Citado na página 1.

SOUZA, A. S. M. D. Comparação do uso do min-max e maximum likelihood com o tempo de voo para estimar posições utilizando o ieee 802.11. In: . [S.l.: s.n.], 2022. Citado na página 13.

TANENBAUM, A. S.; WETHERALL, D. J. **Computer Networks**. 5th. ed. USA: Prentice Hall Press, 2010. ISBN 0132126958. Citado na página 3.

VIANA, J. A. A mobilidade como aventura na cidade: jogos baseados em geolocalização (gps) e apropriação urbana. **Movimento**, Escola de Educação Física, v. 17, n. 1, p. 237–251, 2011. Citado na página 1.

YAN, S. et al. Wi-fi rtt based indoor positioning with dynamic weighted multidimensional scaling. In: **2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)**. [S.l.: s.n.], 2019. p. 1–8. Citado 2 vezes nas páginas 10 e 11.

ZAFARI, F.; GKELIAS, A.; LEUNG, K. K. A survey of indoor localization systems and technologies. **IEEE Communications Surveys Tutorials**, v. 21, n. 3, p. 2568–2599, 2019. Citado na página 5.

ZHANG, D. et al. Localization technologies for indoor human tracking. In: IEEE. **2010 5th international conference on future information technology**. [S.l.], 2010. p. 1–6. Citado 3 vezes nas páginas 5, 7 e 8.

ZHANG, X. et al. Safety assessment and risk estimation for unmanned aerial vehicles operating in national airspace system. **Journal of Advanced Transportation**, v. 2018, p. 1–11, 10 2018. Citado na página 12.