

UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
ENGENHARIA ELÉTRICA

BRENO MENESES, GABRIEL VASCONCELOS E MARINA BATISTA
EMAIL: BRENO.MENESES, GABRIEL.VASCONCELOS,
MARINA.BATISTA@EE.UFCG.EDU.BR

ANTEPROJETO - ARARAJUBA

CAMPINA GRANDE - PB

2022

RESUMO

A localização do robô é o processo que determina onde um robô móvel está localizado em relação ao seu ambiente para que seja possível tomar decisões sobre ações futuras de maneira mais precisa. Este documento tem como principal objetivo justificar a metodologia juntamente com o fluxograma do desenvolvimento de uma solução para a localização de uma base robótica de tração diferencial(DDMR-*Differential-Drive Mobile Robot*) desenvolvida e disponibilizada pela BRISA Robotics, e que possui um conjunto de sensores como encoder, LiDAR, câmera RGB-D, sonar, e o IMU que estão disponíveis para utilização. A solução foi idealizada por meio da Localização de Monte Carlo utilizando o método de fusão de sensores a partir do Filtro de Kalman Estendido. A priori os experimentos serão realizados no ambiente de simulação Gazebo.

1 INTRODUÇÃO

A robótica móvel é uma área em crescimento, por exemplo, para aplicações de exploração espacial no envio de veículos projetados para navegar em superfícies extraterrestres na exploração espacial, os rovers. Essa é apenas uma aplicação da área que envolve o controle de robôs móveis com rodas.

Plataformas diferenciais(DDMR - *Differential-Drive Mobile Robot*) são usadas para os mais diversos propósitos, seja pela facilidade de construção, ou pelo fácil controle. Para todas estas aplicações, ou mesmo para a robótica móvel em geral, alguns dos maiores problemas a serem resolvidos seriam Mapeamento, Navegação e Localização, para tal são utilizados sensores que podem possuir os mais diferentes tipos de princípios de funcionamento e aplicações. Neste referido projeto a problemática escolhida a priori foi a localização da plataforma, admitindo a solução prévia dos demais problemas.

Na resolução de localização, em um cenário típico, um mapa do ambiente está disponível e o robô é equipado com sensores que observam o ambiente e monitoram seu próprio movimento. Dessa forma, o problema de localização torna-se então o de estimar a posição e orientação do robô dentro do mapa usando as informações que foram coletadas dos sensores. O conhecimento de uma boa estimativa da localização do robô é um precursor essencial para a tomada de decisões futuras.

Este documento tem o intuito de por meio de uma plataforma bem definida, estudar e por fim solucionar o problema de localização utilizando da fusão de sensores que será feita por meio do Filtro de Kalman Estendido(EKF - *Extend Kalman Filter*).

O documento a seguir está dividido da seguinte forma: primeiro os objetivos do projeto, segundo a fundamentação teórica de cada ferramenta utilizada, terceiro a metodologia a ser seguida no desenvolvimento do projeto e em quarto o cronograma das atividades a serem desenvolvidas.

2 OBJETIVOS

Este plano de atividades propõe o desenvolvimento de um sistema de localização com base nos dados providos pelos sensores inerentes a plataforma Ararajuba da BRISA Robotics.

Objetivos específicos:

- Modelar a plataforma Ararajuba no ambiente de simulação Gazebo
- Realizar e verificar a leitura individual dos sensores no ambiente de simulação
- Realizar a filtragem dos dados dos sensores disponíveis no ambiente de simulação
- Fundir os dados dos sensores disponíveis para melhor estimativa no ambiente de simulação
- Aplicar e verificar algoritmos de localização do tipo probabilístico e correspondência de mapas no ambiente de simulação
- Implementar a solução na plataforma física

3 FUNDAMENTAÇÃO TEÓRICA

3.1 Modelo cinemático

Para iniciar o desenvolvimento da solução, é preciso a modelagem matemática do sistema a fim de entender como se relacionam as entradas e saídas, equações que governam todo o movimento do robô. Em um DDMR as ações de controle são as velocidades angular e linear da base. Nesse contexto as entradas do sistemas são as tensões aplicadas em cada motor e a saída é sua pose.

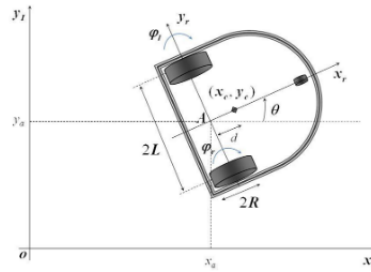


Figura 1 – Representação de um modelo de tração diferencial.

Na figura acima é possível notar que a posição do robô está descrita por meio de dois sistemas de coordenadas distintos. O primeiro é o sistema de coordenadas inercial onde a posição do robô, levando em conta o ponto médio entre as rodas, é definida como $[x_a, y_a]$. Em seguida tomando este ponto como origem de um segundo sistema de coordenadas do próprio modelo, esses eixos coordenadas são denominados de $[x_r, y_r]$, onde y_r está na direção do eixo das rodas, e x_r está apontando na direção frontal do modelo. Assim a orientação em relação aos eixos inerciais pode ser descrita como a rotação do sistema do robô em relação ao sistema inercial.

Em relação a esse sistema o centro de massa está a uma distancia d na direção x_r , e cada roda está a uma distancia de L em relação a origem, tanto na direção y_r , quanto na direção $-y_r$. Cada roda possui um raio R e possuem velocidades angulares dadas por $\dot{\varphi}_r$ e $\dot{\varphi}_l$. Sendo assim, as equações básicas para descrição do movimento do DDMR são dadas por:

$$\begin{bmatrix} \dot{x}_a \\ \dot{y}_a \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{R}{2} \cos(\theta) & \frac{R}{2} \cos(\theta) \\ \frac{R}{2} \sin(\theta) & \frac{R}{2} \sin(\theta) \\ \frac{R}{2L} & \frac{R}{2L} \end{bmatrix} \begin{bmatrix} \dot{\varphi}_r \\ \dot{\varphi}_l \end{bmatrix} \quad (3.1)$$

3.2 Encoder

É um sensor que realiza a edição de rotação de cada roda, por meio de interrupção sucessivas. Com essa medição é possível realizar a estimativa da velocidade de roda, e integrando tal dado se encontra valores de posição. Este tipo de medição de posição é chamado de odometria,

isso será o primeiro passo na predição dos estados da posição, por ser em teoria a forma de medição com mais erros seus dados serão revisados pelos demais sensores.

3.3 Inertial Measurement Unit (IMU)

É um dispositivo composto por dois sensores são eles: o acelerômetro e o giroscópio. Sendo capaz de medir as acelerações lineares ao longo dos eixos e as velocidades angulares em torno de cada eixo, se tornando assim um sensor com seis graus de liberdade (6 DOF).

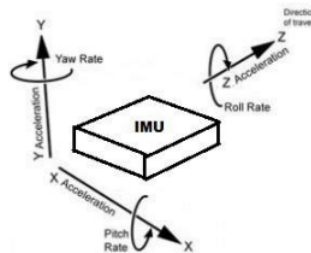


Figura 2 – Dados adquiridos por meio do IMU em cada um dos eixos coordenados.

Este tipo de sensor é altamente sensível a erros, além disso por vezes os dados devem ser integrados aumentando ainda mais as incertezas. Devido a isso nesta aplicação o IMU não será usado de forma única, e sim associando suas leituras com as dos demais sensores. Para propósitos de simulação o IMU escolhido foi Razor's IMU 9 DOF, por ser o mais utilizado em termos de pacotes, possuindo uma quantidade de documentação referente a aplicação mais elevada. Por isso, o pacote utilizado foi o IMU tolls.

3.4 Light Detection And Ranging (LiDAR)

O princípio operacional de um LiDAR típico é por meio da emissão de ondas de luz pulsadas no ambiente. Os pulsos emitidos são refletidos pelos objetos que encontram-se no alcance, e a partir do tempo de voo (TOF) que cada pulso demorou a retornar ao sensor é calculado a distância percorrida. Uma outra forma é por meio da medição de desvio de fase.

A partir do LiDAR 3D da empresa Velodyne será possível realizar a localização por meio da nuvem de pontos gerada e sua correspondência no mapa, no entanto a priori será utilizado um LiDAR 2D da Hokuyo por simplificação.

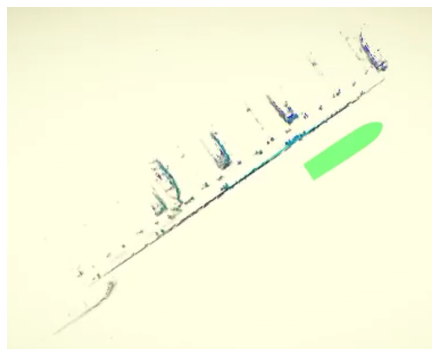


Figura 3 – Aplicação de uma solução simulada para atracar um navio em uma marina por meio de técnicas de localização com um LiDAR 3D da Velodyme realizado pela empresa BRISA Robotics.

3.5 Sonar (Sound Navigation and Ranging)

É um sensor de distância baseado no tempo de voo (TOF) que uma determinada onda sonora percorre. Assim quando essa onda é rebatida por algum objeto ocorre a medição da distância do mesmo.

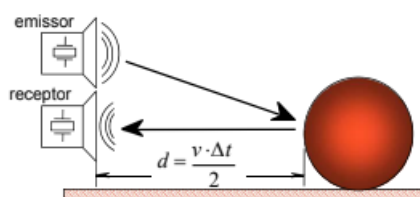


Figura 4 – Princípio de funcionamento de um sonar com emissor e receptor distintos.

A distância medida por cada um dos sonares será correspondida com o mapa, obtendo assim uma estimativa da posição. Essa estimativa por sua vez irá compor o fluxo de aproximação do valor correto da posição do modelo. Como o sensor sonar possui uma grande variedade de modelos, será utilizado um modelo generalista afim de alterar apenas alguns parâmetros para adaptação com a plataforma física. Com o intuito de se obter uma nuvem de pontos será implementado 10 sonares ao redor da plataforma, fornecendo assim parâmetros para fundir com o LiDAR e a câmera RGB-D, gerando assim uma nuvem de pontos para a correspondência de mapa com uma maior taxa de confiabilidade.

3.6 Câmera RGB-D (Red, Blue, Green, Depth)

É um dispositivo de detecção de profundidade que funciona comumente com a associação de uma câmera RGB (Red, Blue, Green) e uma câmera infravermelha (IR) e que é capaz de aumentar a imagem convencional com informações de profundidade (relacionadas com a distância ao sensor) dos objetos presentes efetuando-se uma triangulação entre as imagens das duas. Para propósitos de simulação, o Microsoft Kinect RGB-D será utilizado no auxílio de aquisição de dados para localização.

3.7 Kalman Filter Extended (EKF)

O filtro de Kalman (KF) é capaz de estimar com certa precisão os estados, tendo como ponto de partida os dados corrompidos que sensores podem coletar. O EKF é baseado na teoria do KF, porém levando em consideração sistemas não lineares. Nesse sentido, o EKF é um modelo de filtro probabilístico que será implementado para diminuição de ruídos inerentes aos sensores, por exemplo no IMU. Ademais, o EKF será utilizado para realizar a fusão de dados dos sensores disponíveis devido a não linearidade. Dessa forma, o pacote ROS Robot-Localization-EKF será utilizado.

3.8 Robot Operating System (ROS)

É um sistema de código aberto para robôs que atua como uma ponte entre um sistema operacional ou banco de dados e aplicativos, especialmente em uma rede. Ele fornece serviços que se esperaria de um sistema operacional incluindo abstração de hardware, controle de dispositivo de baixo nível, implementação de funcionalidades comumente usadas, troca de mensagens entre processos e gerenciamento de pacotes.

O principal objetivo do ROS é apoiar a reutilização de código na pesquisa e desenvolvimento de robótica, por ser uma estrutura distribuída de processos (Nós) que permite que diferentes dispositivos possam se comunicar. Neste projeto, a princípio será utilizado o ROS Noetic.

3.9 Gazebo

É um ambiente de simulação de código aberto, que permite a simulação dinâmica de sistemas com multi robôs. Ele se baseia em sistemas de modelagem física como o ODE(Open Dynamics Engine) para representar o funcionamento dinâmico, e para construir a interface gráfica uma como sistema de renderização OGRE (Object-oriented Graphics Rendering Engine). Utilizar o Gazebo em conjunto com o ROS flexibiliza a passagem do ambiente de simulação para a plataforma real, já que leva em consideração os parâmetros físicos do sistema real.

3.10 EtherCAT

É um protocolo de comunicação apriori instalado na plataforma real. Este sistema é uma tecnologia Ethernet industrial de alto desempenho, baixo custo e topologia flexível em relação ao padrão Ethernet, características essas que se adequa a aplicações em malhas de automação e controle.

Comumente implementado em sistemas de controle em tempo real. Normalmente, na extração de informações de plataformas robóticas para a tomada de decisões, tendo suma importância uma maior latência. Sendo assim, a partir do pacote ROS Simple Open EtherCAT Master(SOEM) será possível implementar uma rede EtherCAT IO simples.

4 METODOLOGIA

A metodologia a ser seguida para o desenvolvimento do projeto consiste na divisão de etapas, que se inicia com a implementação do modelo DDMR Pioneer-3DX e dos respectivos sensores que farão parte do projeto no ambiente de simulação Gazebo. Além disso, serão incluídas as configurações necessárias para que o sistema seja controlado utilizando ROS. Com o ambiente configurado, pode-se dar início a aquisição de dados dos sensores para solucionar o problema de localização, utilizando a fusão dos dados coletados e o processo de filtragem com EKF e o filtro de partículas. Por fim, será feita a migração e validação da solução simulada para a plataforma real.

Devido a indisponibilidade da documentação da plataforma Ararajuba da BRISA Robotics, serão realizados experimentos no ambiente de simulação Gazebo utilizando a plataforma Pioneer 3-DX por caracterizar um DDMR. Sendo assim, após a aquisição da plataforma Ararajuba e suas informações será realizado a implementação do seu modelo no gazebo.

A implementação dos sensores no ambiente gazebo será feita por meio dos arquivos de descrição .xacro e .gazebo. No arquivo .xacro relativo ao modelo será incluído a descrição física dos sensores. Já no arquivo .gazebo será implantado o plugin relativo a leitura que possibilita a comunicação, entre o ROS e o gazebo. Ainda no arquivo .gazebo será incluído parâmetros necessários como taxa de comunicação e especificação do tipo e da intensidade do ruído.

Combinando as informações oriundas dos sensores disponíveis, será aplicada a técnica de fusão de dados devido a baixa confiabilidade dos dados dos sensores dispostos individualmente. Além disso, será relevante para um maior aproveitamento das informações que estarão disponíveis. A fusão será feita partir do EKF, devido a sua compreensão de modelos não-lineares, podendo ser alterado futuramente para o filtro de partículas a depender dos resultados.

Inicialmente a abordagem escolhida para solucionar o problema de localização foi a Localização de Monte Carlo (MCL), esse é um algoritmo que tem como princípio o filtro de partículas, estimando a pose do robô conforme a sua movimentação e detecção do ambiente, sem necessidade da informação da pose inicial, motivo que determinou a sua escolha a priori.

Para a etapa de implementação da solução na plataforma física, primeiramente será realizado um estudo da estrutura de comunicação já implementada na plataforma. A princípio, a comunicação é composta por um Computador industrial instalado na própria plataforma, que realiza a comunicação com drives, sensores e demais atuadores por meio de conexão Ethernet usando o protocolo denominado EtherCAT. O qual será possível realizar implementações da topologia de barramento em uma placa JetsonNano e sensores como o Sensor Tag 2 CC2650STK, da Texas Instrument.

Com a migração da interface simulada para a plataforma real há algumas mudanças que

devem ser consideradas, como a configuração dos sensores físicos para que escrevam nos tópicos ROS utilizados durante as simulações utilizando os devidos drives. Após essa configuração, será feita a validação dos dados dos sensores utilizando filtragem e fusão de dados disponíveis em pacotes ROS para obter uma estimativa de localização do robô no mapa.

| | Semanas | | | | | | | | |
|--------|---------|---|---|---|---|---|---|---|---|
| Etapas | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 1 | x | | | | | | | | |
| 2 | | x | x | | | | | | |
| 3 | | | x | x | x | | | | |
| 4 | | | | | x | x | | | |
| 5 | | | | | | x | x | | |
| 6 | | | | | | | x | x | x |

REFERÊNCIAS

C. Jing, J. Potgieter, F. Noble and R. Wang, *"A comparison and analysis of RGB-D cameras' depth performance for robotics application, 2017"*. 24th International Conference on Mechatronics and Machine Vision in Practice (M2VIP), 2017, pp. 1-6, doi: 10.1109/M2VIP.2017.8211432.

L. Velodyn *"What is lidar?"*. Lidar 101. 2022. Disponível em: <<https://velodynelidar.com/what-is-lidar/>>. Acesso em: 01 de Maio de 2022.

R. E. KALMAN, *"A New Approach to Linear Filtering and Prediction Problems, 1960"*. Transactions of the ASME–Journal of Basic Engineering, Baltimore, 1960.

D. Rached, A. A. Hatab, *"Dynamic Modelling of Differential-Drive Mobile Robots using Lagrange and Newton-Euler Methodologies: A Unified Framework, 2013"*. Advances in Robotics Automation, College of Engineering, American University of Sharjah, 2013.