

TDP Roboforge OPEN: CBR 2024 - Categoria Challenge Open

Adriano R. P. Junior, Eric M. Vilela, Kallebe S. Oliveira, Marcella S. Figueredo,
Maria E. N. Andrade, Rebeca L. Graff

Abstract—Neste documento, serão apresentadas propostas, roteiros e reflexões estratégicas utilizadas pela equipe Roboforge OPEN para a CBR 2024 na categoria *Challenge Open*.

I. INTRODUÇÃO

A Equipe Roboforge de Robótica da Universidade Federal de Uberlândia (UFU) é composta por alunos de graduação de diversos cursos da instituição e se dedica à pesquisa e desenvolvimento em robótica autônoma, participando ativamente de competições e também atuando em projetos de extensão. A equipe tem uma trajetória de participações na Competição Brasileira de Robótica (CBR) desde 2018, atuando em categorias de robôs desenvolvidos utilizando kits educacionais padrão, principalmente LEGO, como a categoria *IEEE Standard Educational Kit* até 2023 e a nova categoria *Challenge KIT* em 2024. Neste ano, a equipe se prepara para enfrentar o novo desafio da categoria *Challenge OPEN*, buscando desenvolver robôs que resolvam problemas de nível semelhante ao de competições anteriores, mas que sejam construídos utilizando quaisquer tipos de componentes estruturais e eletrônicos, e não apenas peças de kits educacionais padrão.

No desafio deste ano desta categoria, os times competem entre si em um cenário que simula a locomoção autônoma em cidades, utilizando carros inteligentes. Os robôs devem transportar peças que representam passageiros até seus destinos finais, enfrentando obstáculos como terreno irregular (com a presença de lombadas) e trânsito dinâmico, com possibilidade de acidentes e vias obstruídas.

Este documento traz as ideias a partir das quais a equipe pretende trabalhar até a competição, a fim de projetar, desenvolver, testar e manter um robô construído com peças e componentes de hardware livre que atenda os requisitos das tarefas envolvidas no desafio da categoria *Challenge OPEN* em 2024. São as seções que seguem:

- 1) Desafio: descrição breve das regras do desafio da categoria *Challenge* em 2024;
- 2) Estratégia: descrição geral da estratégia inicial do robô desenvolvido pela equipe;
- 3) Metodologia: envolve aspectos específicos de planejamento, cronograma de atividades e fluxo de trabalho até a competição;
- 4) Mecânica: traz aspectos específicos da mecânica envolvidas no desenvolvimento do robô;
- 5) Eletrônica: aspectos eletrônicos do robô;
- 6) Software: aspectos específicos de software para programação da solução autônoma dos desafios;
- 7) Conclusão.

As seções 2 a 6 também apresentam cada uma as ferramentas utilizadas em cada contexto pela equipe, de forma a constituírem todo o aparato de trabalho necessário para realização das tarefas.

II. DESAFIO

O desafio da categoria *Challenge OPEN*, da RoboCup Brasil, tem como temática o estabelecimento de cidades inteligentes, ou seja, um sistema de locomoção responsivo às circunstâncias da via, trazendo autonomia para o veículo. Isso é abordado, por exemplo, ao simular obstáculos dinâmicos no trânsito com caixas de leite, uma vez que, na vida real, existem inúmeros incidentes que podem tornar uma via inutilizável. Nesses casos, é imprescindível o recálculo de rota para que o veículo ainda possa chegar ao seu destino final. Assim, um robô que consiga realizar essa tarefa sem intervenção de seres humanos é considerado um robô autônomo.

De forma mais detalhada, o desafio aplica essa inspiração da autonomia veicular por meio de regras específicas. Por exemplo, há um condicionamento para cada tipo de pessoa, representado por tubos de 10 cm e 15cm, que simulam respectivamente crianças e adultos, em cores variadas (azul, verde, marrom, vermelho e branco). Esses tubos devem ser posicionados em locais específicos da cidade, que também variam em cores (marrom, azul, vermelho e verde) e tamanhos (24x24 cm, 27x24 cm ou 150x15 cm), e o caminho deve ser realizado sem invadir nenhum local indesejado.

Locais indesejados são as áreas que delimitam o tapete, seja internamente, com as linhas pretas, ou externamente, como a faixa azul que divide a área disponível para cada equipe competidora cumprir seu desafio. As linhas vermelhas que delimitam a borda da pista também não podem ser ultrapassadas.

Por outro lado, as linhas amarelas representam possíveis locais de acesso do robô para o desembarque das pessoas embarcadas na área azul. Um desembarque será considerado um sucesso se o robô não tiver abandonado momentaneamente nenhum passageiro ou derrubado algum outro na área de embarque; se o robô tenha entrado pela linha amarela sem invadir a linha preta, deixando a pessoa na área correspondente ao seu destino e retornado ao corredor branco pela mesma faixa amarela. Caso contrário, a pontuação não será integral, e o juiz pode recolocar os tubos ou o robô no tapete nos lugares designados, se forem considerados como perdidos.

Também será considerada uma penalidade se houver algum obstáculo na pista – lembrando que este recurso não pode ultrapassar três por lado do tapete – e o robô não

conseguir refazer sua rota. Além disso, a competição de 2024 conta com o recurso das lombadas. Elas podem ser colocadas em pontos do corredor que não obstruam a passagem dos robôs, mas que, de fato, configuram um desafio adicional para que os robôs sejam guiados pelos seus sensores.

Por último, as regras da categoria determinam limitações específicas ao design do robô, que devem ser respeitadas por todos os competidores. Uma delas diz respeito às dimensões dos veículos, que não devem passar de um cubo de 25 cm de aresta. Outra é uma limitação de software: não é permitido nenhum tipo de processamento de imagem, então a engenharia dos protótipos também não pode conter nenhum tipo de câmera ou similar.

III. ESTRATÉGIA

Inicialmente a equipe decidiu dividir as tarefas necessárias para alcançar o objetivo proposto pelo campeonato, de forma que grupos menores de pessoas foram designadas a executar subtarefas ao longo do tempo que seriam compostas no projeto final. Dentre essas subtarefas existentes apresentam: mecânica do robô, eletrônica, autolocalização do robô no mapa, determinação de rotas eficientes entre quaisquer dois pontos do cenário, busca e embarque de passageiros, desembarque de passageiros e detecção de obstáculos.

Como o desafio da categoria *Standard Educational Kit* em 2023 e o *Challenge Open* de 2024, apresentam propostas semelhantes por parte da programação, algumas das subtarefas exigidas neste desafio já vinham sendo trabalhadas e estão sendo aprimoradas pela equipe e adaptadas da melhor forma possível para o novo robô que será construído. Tais trabalhos envolvem, por exemplo: autolocalização do robô no mapa, determinação de rotas eficientes entre quaisquer dois pontos do cenário, busca e embarque de passageiros, desembarque de passageiros e detecção de obstáculos.

Contudo, a área em que são necessários maiores estudos e testes seria a parte de hardware do robô, o que envolve a realização de vários protótipos, e pesquisas para sua confecção final. A princípio, foi proposto a alternativa da realização de um chassi feito de impressão 3D, que seria modelado por meio do software *Fusion 360*. Outra alternativa levantada foi a de estruturar um “esqueleto” do chassi através de estruturas de metal, e elaborar outras peças através de impressão 3D, para compor o todo. De qualquer forma, a estrutura deverá ser construída de modo que consiga ser modular e capaz de comportar diversos tipos de sensores necessários, além do microcontrolador, motores, bateria e também uma garra, que seria juntamente confeccionada através de manufatura aditiva. Esta teria a tarefa de agarrar e transportar os passageiros desde a plataforma de embarque e soltá-los apenas nas áreas propostas, levantando-os do chão e sem deixá-los cair.

Além disso, para a realização da parte eletrônica necessária do robô, serão utilizados softwares para testar ideias, montar circuitos, e planejar a confecção de todo sistema elétrico que será usado no robô como LTSpice, EasyEDA, KiCad, Proteus, etc. Também serão estudados pela equipe os melhores sensores e microcontroladores para a

realização do desafio: serão realizados testes e construídos protótipos que mostrarão quais e quantos dos modelos de sensores de cor, ultrassônicos, infravermelhos, e giroscópios serão usados na versão final a fim de obter o melhor resultado possível na competição.

IV. METODOLOGIA

Nesta seção do trabalho a equipe se propõe a explicitar o modelo de trabalho que pretende adotar até a competição, assim como as formas que pretende gerir imprevistos e possíveis desvios da estratégia inicial. Para isso, a equipe pretende se inspirar em técnicas de metodologias ágeis, como *Kanban* e *Scrum*, por exemplo, e outros modelos de planejamento de projetos.

A organização geral do trabalho da equipe se dá organizando cada um dos membros em uma subequipe designada de acordo com sua área de atuação. As áreas definidas para a categoria *Challenge OPEN* em 2024 são: engenharia, que contempla os trabalhos de mecânica e eletrônica; programação, que contempla os trabalhos de desenvolvimento de software e gestão de pessoas, responsável pela manutenção das estruturas descritas nesta seção.

Para definição explícita das atividades a serem realizadas por cada uma das áreas, a equipe buscou se aproximar do modelo *Kanban*. Este modelo pode ser definido a partir de alguns conceitos chave, como:

- Sistema visual, que fica explícito a todos da equipe, e mostra de forma clara e objetiva quais tarefas estão sendo realizadas, quais ainda não foram iniciadas e quais já foram concluídas.
- Cada tarefa é representada como um “cartão”, e eles ficam dispostos em colunas que indicam o status daquela tarefa no fluxo de trabalho.
- Limitação do número de itens que pode estar em andamento em cada estado do fluxo de trabalho, controlando a possibilidade de sobrecarga e/ou negligenciamento de uma tarefa sobre outra.

Dentro da equipe pretende-se utilizar um sistema de gerenciamento online como o *Trello*, ou o que existe dentro do *GitHub*, por exemplo, para que o quadro *Kanban* fique disponível a cada uma das áreas de forma virtual. Nesses sistemas é possível explicitar datas de entrega e qual membro fica responsável por cada tarefa, além de possibilitar que as áreas atualizem o quadro de forma colaborativa. A equipe pretende eleger, no início dos trabalhos, uma pessoa responsável pela manutenção do quadro virtual em cada área, além dos membros da gestão, que ficam responsáveis pelo suporte geral a todas as áreas.

A conferência, atualização e levantamento de novas tarefas pode ser realizada inclusive (mas não exclusivamente) dentro das reuniões semanais de cada área, que são realizadas justamente com este fim, para que todos se atualizem do progresso, das dificuldades, e das tarefas que serão desenvolvidas a seguir. A ideia é que cada semana seja um ciclo de desenvolvimento *Scrum*, e revisões maiores com a presença de todos os membros de todas as áreas seriam marcadas a cada 2 ou 3 semanas.

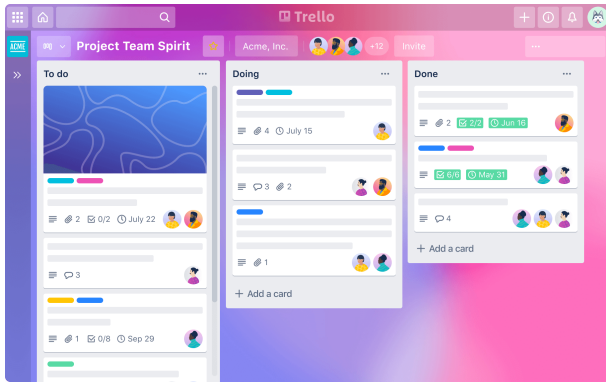


Fig. 1. Exemplo de tarefas associadas ao método Kanban na interface do Trello.

Inicialmente, a equipe trabalha com uma projeção das tarefas a serem realizadas, a fim de manter um ritmo de trabalho adequado, e de forma a entender quanto tempo seria empregado no desenvolvimento de cada uma delas, e tentando deixar mais explícito quais tarefas podem ser realizadas ao mesmo tempo por áreas ou membros diferentes. Para isso, foi utilizado o Diagrama de Gantt, uma ferramenta de gestão de projetos que relaciona trabalhos concluídos em certo tempo com o tempo reservado para eles. O diagrama de Gantt também pode incluir as datas de início e término de tarefas, marcos, dependências entre tarefas, responsáveis e muito mais. Abaixo o diagrama previsto para a equipe, destacando as responsabilidades das áreas de engenharia e programação. Os membros da área de gestão são responsáveis por todo o acompanhamento e gestão das tarefas em geral.

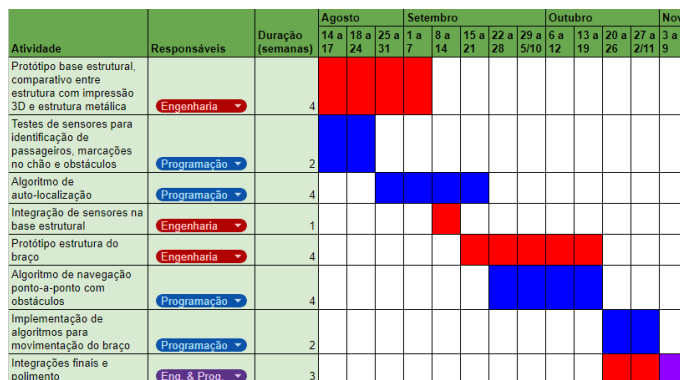


Fig. 2. Diagrama de Gantt com possível cronograma a ser seguido pela equipe até a competição.

V. MECÂNICA

Na RoboCup Brasil Challenge 2024, a construção de um robô eficiente para a modalidade *OPEN* exige uma cuidadosa seleção de materiais e uma abordagem metódica no design da estrutura. O desafio é garantir que o robô, com suas dimensões restritas a 25x25x25 cm, seja robusto, leve e versátil o suficiente para cumprir as tarefas da competição.

A escolha do material para a estrutura do robô é crucial. Uma das estratégias da equipe é a de utilizar perfis estruturais de alumínio para o esqueleto do robô. O alumínio se destaca por sua combinação de resistência e leveza, dois atributos essenciais para navegar de forma ágil e eficaz.

A montagem dos perfis de alumínio é facilitada pela sua modularidade. Pode-se utilizar cantoneiras e parafusos com porcas martelo, o que simplifica o processo de construção e permite ajustes rápidos e fáceis durante a fase de prototipagem. Esta abordagem modular é particularmente vantajosa, pois possibilita a personalização e a modificação da estrutura conforme necessário, sem a necessidade de ferramentas especializadas ou processos complexos.

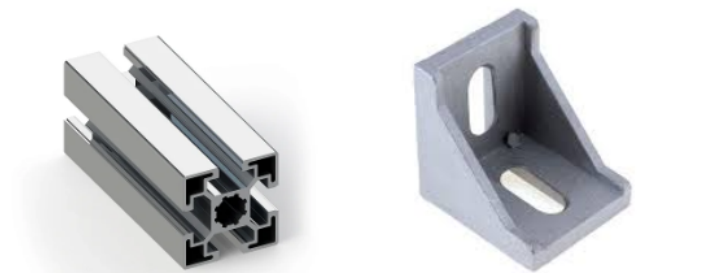


Fig. 3. Perfil estrutural de alumínio e cantoneira equivalente.



Fig. 4. Porca martelo e seu funcionamento no perfil de alumínio.

As demais peças (fixação dos motores, sensores e garra) poderiam ser impressas utilizando filamentos de plástico como o PLA, e tendo uma estrutura sólida e com “slots” para conectar mais peças o processo de prototipagem seria mais fluido, uma vez que poderíamos trocar facilmente a posição dos componentes.

A garra do robô é um componente crítico que pode ser projetada com dois graus de liberdade para atender às exigências do desafio. O primeiro grau de liberdade é a flexão e extensão do cotovelo, que permite a elevação e a descida do tubo. O segundo grau de liberdade é a flexão e extensão do pulso, que garante que a base do tubo permaneça paralela ao solo durante o transporte. Por fim, o efetor final deve ser capaz de agarrar e soltar os tubos de PVC com precisão, garantindo que estes não se percam no caminho ou “tombem” no desembarque.

A decisão de não incluir movimentos de rotação na garra foi baseada na constatação de que o robô pode girar em

todas as direções utilizando as rodas omnidirecionais, o que elimina a necessidade de rotação adicional no braço robótico.

Para garantir a eficácia do projeto, empregamos uma série de ferramentas e softwares especializados. Utilizamos o *Autodesk Fusion 360* e o *Autodesk Tinkercad* para o design e modelagem dos componentes. O software *Ultimaker Cura* é utilizado para preparar os arquivos de impressão 3D, garantindo que as peças sejam produzidas com a qualidade necessária. Além disso, ferramentas práticas como multímetros e ferros de solda são empregadas para a montagem e ajuste dos componentes eletrônicos e mecânicos.

VI. ELETRÔNICA

Com relação aos componentes eletrônicos do robô, foi bastante discutido o uso de placas de circuito impresso ilhadas (furadas) e/ou protoboards no estágio de prototipagem dos circuitos antes de manufaturar a placa personalizada definitiva utilizando placas de fenolite e uma máquina de CNC para fazer as trilhas e furações necessárias.

Com relação a isso, as placas de circuito impresso ilhadas são uma opção tradicional e eficaz para a prototipagem, permitindo a criação de circuitos com conexões sólidas. No entanto, para a fase inicial, onde a flexibilidade e a facilidade de modificação são essenciais, as protoboards oferecem uma solução mais prática. Estas placas permitem a montagem rápida e a experimentação com diferentes arranjos de componentes sem a necessidade de soldagem permanente, facilitando a identificação e correção de problemas antes da fabricação final.

Antes de proceder com a manufatura da placa personalizada definitiva, é preciso simular os circuitos eletrônicos em ambientes virtuais para validar seu funcionamento. É possível utilizar ferramentas de simulação como *LTSpice* e *Proteus* para testar e ajustar os circuitos de forma virtual. Estes softwares permitem a modelagem detalhada dos circuitos, possibilitando a análise de seu comportamento sob diferentes condições e a identificação de possíveis falhas sem a necessidade de montar fisicamente o circuito.

Após a simulação e a validação dos circuitos, o próximo passo é a criação da placa de circuito impresso final. Para isso, é possível empregar softwares como *EasyEDA* e *KiCad*, que auxiliam na elaboração do layout da PCB, como a correta configuração dos *footprints* dos componentes e a trajetória das trilhas. Estes softwares garantem que os circuitos estejam corretamente projetados, evitando problemas comuns como trilhas com curvas de 90 graus, que podem causar interferências, perdas de sinal e sobreaquecimento.

O circuito de alimentação do robô seria conectado a uma bateria LiPo de 11,1 V, que proporciona a potência necessária para o funcionamento dos motores e demais componentes eletrônicos. A escolha da bateria deve levar em conta a capacidade de corrente e a autonomia desejada para o robô, garantindo que a fonte de alimentação seja adequada para as demandas do sistema.

A integração dos componentes eletrônicos na placa de circuito impresso final inclui a utilização de uma ponte H quádrupla embutida, que permite o controle dos motores,

e um suporte para um controlador, como o Raspberry Pi Pico W, que pode ser montado como SMD (*Surface-Mount Device*). Esta configuração proporciona a flexibilidade necessária para a programação e controle do robô, além de permitir a manipulação dos pinos GPIO para a conexão de sensores e atuadores.

VII. SOFTWARE

A. Ferramentas utilizadas

Esta subseção descreve as ferramentas utilizadas pela equipe RoboForge durante a execução do projeto de software, tratando-se de plataformas e serviços essenciais para o desenvolvimento. Estas ferramentas envolvem, mas não se limitam a: desenvolvimento de código, auxílio na tomada de decisões, etc. São elas:

1) *Git/GitHub*: *Git* e a plataforma *Github* são ferramentas de versionamento de código. A equipe segue o conjunto de práticas *Git Workflow* para realizar os trabalhos de forma estável. Neste modelo, contamos com várias *branches* (“versões, correntes alternativas, ramificações”) onde os membros poderão trabalhar e rascunhar etapas diferentes ao mesmo tempo, otimizando o trabalho da equipe, e podendo gerar e testar novas ideias a partir da mesma base do código principal. Estas convenções são comprovadamente úteis durante competições anteriores, auxiliando no uso de versões alternativas, sem o risco de alterações que causariam a descaracterização do que já havia sido consolidado e funcional.

2) *Visual Studio Code*, *PlatformIO* e *Live Share*: A programação será desenvolvida utilizando as Linguagem C/C++, pois as linguagens de microcontroladores são baseadas nestas. E para isso, utilizaremos o editor de código fonte *Visual Studio Code* junto da extensão *PlatformIO* que foi criada para desenvolvimento de sistemas embarcados e microcontroladores. Como o *VS Code* se trata apenas de um editor de texto a extensão *PlatformIO* se faz necessária pois o torna um ambiente de desenvolvimento eficiente e versátil para compilar e fazer o upload dos códigos para os microcontroladores. Também poderemos usar a extensão *Live Share* do *VS Code* que possibilita que mais de um membro da equipe trabalhem na mesma versão do código ao mesmo tempo, auxiliando outros membros mesmo a distância em tempo real. Isto é algo que se faz útil para programação em duplas, na correção de erros ao vivo e auxílio de membros independente da localização física.

3) *Arduino IDE*: O ambiente de desenvolvimento integrado (IDE) do Arduino é um dos softwares mais utilizados para desenvolver projetos em Arduino e outros microcontroladores. Por possuir uma interface simplificada, ele se torna muito eficiente como editor de texto e ambiente onde se pode compilar e fazer upload do código.

B. Cinemática de movimento omnidirecional

Para possibilitar maior liberdade de movimento, a equipe considera adicionar ao robô a ser desenvolvido, um conjunto de rodas omnidirecionais, ou seja, rodas que permitam com

que o robô se mova em qualquer direção, incluindo lateralmente e diagonalmente. É esperado obter grande valor nessas características do robô, dadas as condições de espaço do mapa da competição. As tratativas em software para controle de movimentações do mesmo ficam, então, um pouco mais complexas do que num robô com rodas tradicionais.

Cada roda omnidirecional é composta por um conjunto de pequenas rodas rolantes inclinadas em um ângulo específico e dispostas através de sua circunferência exterior. Quando a roda principal gira, essas pequenas rodas rolam em direções diferentes, e a composição dos movimentos das quatro rodas presentes no robô podem gerar movimentos mais complexos. Essas relações podem ser representadas através de vetores, como na imagem abaixo.

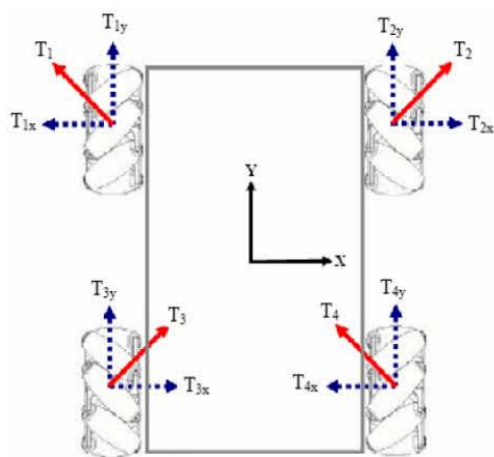


Fig. 5. Esquema de um robô com rodas omnidirecionais indicando os vetores de força em cada roda.

Nomeando os motores como 1, 2, 3 e 4 sendo, respectivamente, os motores da frente-esquerda, frente-direita, trás-esquerda e trás-direita, alguns movimentos podem ser programados da seguinte forma:

- Movimento frontal: intuitivamente, é obtido com todas as rodas girando na mesma direção. O sinal da velocidade determina se o robô se moverá para frente ou para trás.
- Movimento lateral: motores 1 e 4 na mesma direção, motores 2 e 3 na direção oposta, todos com a mesma velocidade. Ou seja, motores diametralmente opostos giram com a mesma velocidade para o mesmo lado. O vetor resultante dos motores promove o movimento lateral do robô. Dependendo do sinal da velocidade, o robô se move para a direita ou para a esquerda.
- Movimento diagonal: os motores de determinada diagonal se mantêm parados, enquanto os motores da outra diagonal se movem na mesma direção. O sinal da velocidade aplicada determina em qual direção sobre a diagonal determinada o robô se move. Por exemplo: os motores 1 e 4 permanecem imóveis enquanto 2 e 3 se movem para frente - o robô se move na diagonal frente-esquerda.

- Curva em torno do próprio eixo: de forma análoga a um robô com rodas convencionais, as rodas de um lado do robô se movimentam em uma direção, e as do outro lado na direção oposta. A troca de direções provoca uma curva no sentido contrário.

Além desses movimentos básicos citados, outros mais elaborados podem ser alcançados pela equipe explorando outras técnicas mais complexas de controle e cinemática.

C. Cinemática do braço robótico

A cinemática inversa é uma técnica empregada em robótica para calcular os ângulos necessários nas articulações de um braço robótico, de modo que o efector final atinja uma localização específica desejada. Diferentemente da cinemática direta, que determina a posição do efector a partir das configurações das articulações, a cinemática inversa oferece a vantagem de, a partir da localização final desejada do efector, permitir que o software gere comandos para todos os motores e/ou atuadores do braço robótico.

No contexto do desafio apresentado, observa-se que o apêndice não requer movimentos complexos, como a manipulação de objetos em um espaço tridimensional, sendo a principal tarefa do braço robótico coletar os tubos a uma altura superior à da entrega. Assim, o braço robótico deve possuir duas articulações com juntas de revolução, além do efector final, responsável por segurar e soltar os tubos de PVC.

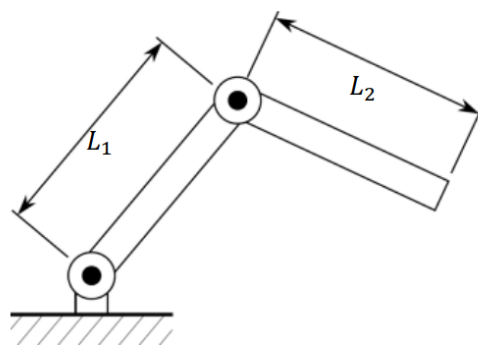


Fig. 6. Diagrama de um braço robótico com duas juntas de revolução.

Como o braço possui apenas dois elos (L_1 e L_2 , conforme mostrado na imagem acima), é possível formar um triângulo ao traçar uma linha entre a articulação na origem do sistema de coordenadas e o efector final do braço, cuja posição é determinada pelo operador. A partir disso, podemos utilizar ferramentas matemáticas de fácil implementação, como a lei dos cossenos, para determinar os ângulos a serem aplicados nas articulações.

D. Localização e Navegação

Para programar um robô autônomo com capacidade de cumprir o desafio da competição, a habilidade de se localizar com precisão no mapa é essencial, especialmente quando a posição inicial é aleatória e há a possibilidade da adição

de obstáculos durante a partida. Para isso, modelos probabilísticos, como os *Hidden Markov Models* (HMMs) e suas variações, podem ser extremamente úteis para alcançar essa precisão, lidando eficazmente com incertezas e ruídos nas leituras dos sensores.

Este tipo de abordagem modela a localização do robô como um processo estocástico, que pode ser definido como “uma sequência de variáveis aleatórias indexadas ao tempo e também a eventos, sendo uma variável que se desenvolve no tempo de maneira parcialmente aleatória e imprevisível.” Os estados ocultos representam a posição verdadeira do robô e as observações correspondem às leituras dos sensores. Essa modelagem permite ao robô considerar múltiplas hipóteses sobre sua localização e atualizar essas hipóteses conforme novas observações são feitas. Desta forma, a ideia é aperfeiçoar a navegação e a localização do robô, tornando-as mais sofisticadas.

Uma das principais vantagens desta estratégia é a capacidade de integrar informações de múltiplas leituras de sensores ao longo do tempo. Cada nova observação ajuda a refinar a estimativa da posição do robô. Em ambientes dinâmicos, onde a precisão das leituras dos sensores pode variar significativamente, algoritmos baseados neste tipo de modelo probabilístico oferecem uma abordagem robusta para a navegação, pois permitem que o robô se adapte a mudanças no ambiente.

Apesar de suas vantagens, esta abordagem também apresenta algumas desvantagens. A complexidade computacional pode ser significativa, especialmente em ambientes com muitos estados e observações possíveis, o que pode levar a um maior tempo de processamento e consumo de recursos. A intenção é analisar a aplicabilidade da prática, bem como explorar as diversas variações de modelos probabilísticos existentes e entender como eles podem ser úteis para alcançar um bom desempenho na resolução do desafio proposto.

E. Processamento de sinais dos sensores

Dado os ruídos presentes nas leituras dos sensores, é fundamental que seja feito o tratamento dos sinais coletados. Em função disso, estratégias para o processamento desses sinais devem ser traçadas. Uma dessas estratégias é a utilização de árvores de decisão. No entanto, acredita-se que para aumentar a robustez do modelo, é viável utilizar uma floresta aleatória, isto é, o conjunto de árvores de decisão.

“Florestas aleatórias”, ou *Random Forest*, é um método de aprendizado de máquina supervisionado usado principalmente para classificação. Uma das vantagens de usar este método é a sua precisão. Isso se dá pois a combinação de múltiplas árvores de decisão é capaz de captar uma grande variação de padrões nos dados e analisá-los de forma eficaz.

Devido à característica de *overfitting* deste modelo, que muitas vezes é vista como uma desvantagem, ele pode oferecer alta precisão em contextos específicos. Isso ocorre porque, neste caso, não é necessário generalizar o modelo, mas sim aplicá-lo a um conjunto específico de dados observados pelo sensor em questão.

A priori, a implementação deste método será realizada utilizando os dados dos sensores de cor. Novas abordagens serão consideradas para os demais sensores. Cabe ressaltar que, como grande parte da estratégia do desafio depende da identificação correta dos passageiros e dos locais de destino, a necessidade de um bom processamento dos dados dos sensores de cor torna-se ainda mais crucial.

Em relação aos sensores de cor candidatos a serem utilizados, é importante destacar que, além de retornarem valores para RGB, eles também permitem capturar a intensidade da luz refletida. Isso é fundamental, pois ajuda a determinar as cores considerando também o nível de saturação. Os valores são retornados em uma faixa de 0 a 255, e serão analisados pelo modelo.

Para utilizar as variáveis R, G e B, é necessário fazer uma combinação de leitura dos pinos S2, S3 e OUT. A tabela abaixo ajuda a visualizar melhor essas combinações:

S2	S3	Filtros do Fotodiodo (OUT)
Baixo/Low	Baixo/Low	Vermelho
Baixo/Low	Alto/High	Azul
Alto/High	Baixo/Low	Claro (sem filtro)
Alto/High	Alto/High	Claro (Verde)

Ao fazer uso deste modelo de machine learning, espera-se uma melhora na identificação das cores, bem como uma melhor performance no desafio como um todo, visto que, como ressaltado anteriormente, grande parte do problema proposto envolve a distinção adequada das cores do mapa e dos passageiros.

VIII. CONCLUSÃO

Portanto, é desta forma que a equipe de robótica Roboforge se propõe a projetar, desenvolver, testar e manter um robô construído com peças e componentes de hardware livre para resolver as tarefas envolvidas no desafio colocado na categoria Challenge OPEN da CBR 2024. Estão descritos neste documento a estratégia inicial geral, assim como alguns dos aspectos de mecânica, eletrônica, programação e gestão do projeto já identificados em fase inicial. Dessa forma, a fim de buscar o melhor resultado possível, até a competição a equipe fica responsável por testar as possibilidades de componentes, construção e programação do robô, seguindo as diretrizes aqui colocadas, e permanece aberta a novas possibilidades e ideias que possam surgir no futuro como resultado das iterações deste projeto.

REFERENCES

- [1] “Regras Challenge 2024”, versão 1.1 - Agosto, 2024.
- [2] “Kanban: conceito, como funciona, vantagens e implementação”. Disponível em <https://www.totvs.com/blog/negocios/kanban/>. Acesso em agosto de 2024.
- [3] “Processos Estocásticos”. Disponível em https://www.maxwell.vrac.puc-rio.br/15321/15321_4.PDF. Acesso em agosto de 2024.

- [4] Mohd Salih, J., Rizon, M., Yaacob, S., Adom, A. & Mamat, M. *Designing Omni-Directional Mobile Robot with Mecanum Wheel. American Journal Of Applied Sciences.* **3** (2006,5)
- [5] “The Brilliant Engineering of Mecanum Wheels!”. Disponível em <https://youtu.be/noqBUEgyQ8A>. Acesso em agosto de 2024.
- [6] “Introdução à robótica: Cinemática Inversa”. Disponível em <https://homepages.dcc.ufmg.br/~doug/cursos/lib/exe/fetch.php?media=cursos:introrobotica:2017-1:aula10-cinemática-inversa.pdf>. Acesso em agosto de 2024.
- [7] Craig, J. *Introduction to robotics.* (Pearson Education,2006)