

ESTRUTURAÇÃO DE HARDWARE E SOFTWARE PARA DESENVOLVIMENTO DE UM AGV

Gabriel Toffanetto França da Rocha¹ (IC), Giovani Bernardes Vitor¹ (PQ), Rafael Francisco dos Santos¹ (PQ), Willian Gomes de Almeida¹ (PQ)

¹Universidade Federal de Itajubá – Campus Itabira.

Palavras-chave: Cidades inteligentes, Percepção, Robótica autônoma, Robótica móvel, Sensoriamento.

Introdução

Com o grande desenvolvimento da tecnologia e a chegada da indústria 4.0, vem se aproximando cada vez mais do meio industrial sistemas autônomos e inteligentes, que utilizam de inteligência artificial, métodos de controle e redes de comunicação para tornarem processos mais eficientes e também com um maior bem-estar e segurança ao operador. Aplicado a robótica, a utilização de veículos autoguiados, AGV (do inglês *Automated Guided Vehicle*), vem sendo aplicados tanto em ambientes urbanos, em carros, ônibus e caminhões autônomos, mas também no meio industrial, como os caminhões autônomos fora de estrada aplicados à mineração. A aplicação desse tipo de sistema autônomo ao meio industrial trás não só um melhor desempenho na produção e otimização de custos, como também uma maior segurança aos operadores, que não são expostos à ambientes de risco e não estão sujeitos a falhas humanas que podem ser causadas por fadiga ou distração. Porém, os AGV's aplicados na indústria do minério utilizam basicamente de rodas pré-definidas com base em um mapeamento prévio do ambiente e sinais de posicionamentos precisos dados por GPS-RTK, o que causa limitações quanto aplicações em ambientes subterrâneos e *indoor*.

Em vista desta problemática quanto a navegação do veículo autônomo, propõem-se que sejam utilizados sensoriamento e algoritmos capazes de mapear e localizar o robô no espaço, conseguindo planejar uma trajetória do ponto de origem até o ponto de destino e chegar até ela sem colidir com nenhum obstáculo, tendo como base um carro de golf elétrico. Para isso, utiliza-se de sensores para a percepção do ambiente, como LiDAR's e câmeras estéreo, para o controle do movimento do robô, como encoders e IMU, e também um sistema embarcado para o comando da plataforma robótica (de ALMEIDA et al., 2019). São utilizados algoritmos de SLAM (do inglês *Simultaneous Localization and Mapping*), para mapeamento e localização (TUKETOMI; UCHIYAMA; IKEDA, 2017), o A* para planejamento de rota e DWA

(do inglês *Dynamic Window Approach*) e tentáculos para a navegação. Todo esse sistema de sensores e algoritmos é embarcado ao *framework* ROS 2 (do inglês *Robot Operation System*), que agrega uma gama de ferramentas e pacotes utilizados na comunicação e no processamento de dados dos sensores, com o intuito de controlar o veículo da forma desejada. É utilizado também o MicroROS, *framework* que traz a possibilidade de embarcar o ROS 2 à um sistema embarcado de tempo real, o que foi utilizado para integrar alto e baixo nível. Para isso, foram realizados estudos sobre robótica autônoma, além do desenvolvimento de modelos simulados da plataforma robótica utilizada. Foram projetados os suportes para embarcar o sensoriamento ao carro de golf, além do projeto da placa de circuito e da case impressa em 3D para sistema embarcado e de segurança do veículo. Uma vez terminada a parte de baixo nível e revisão bibliográfica, foram implementados em ambiente simulado algoritmos de SLAM e de navegação, além do desenvolvimento de um módulo gráfico de diagnóstico para o veículo.

Metodologia

Inicialmente, foi realizada uma revisão bibliográfica com base em sistemas de robótica autônoma e principalmente no *framework* ROS 2. Foram propostos tutoriais e desafios para o aprendizado dos conceitos básicos de robótica móvel que seriam necessários para o desenvolvimento da pesquisa. Os desafios contemplavam a resolução de problemas que seriam enfrentados durante a pesquisa e que já seriam desenvolvidos desde já.

Em seguida, foi construído o modelo simulado da plataforma robótica utilizada, utilizando do *software* de simulação de robôs Gazebo. Nele, foi criado um arquivo de descrição que continha toda a descrição física do robô, dos sensores utilizados e seu posicionamento no robô, sendo utilizado para obtenção das transformações entre referenciais.

Então, foram realizadas atividades relacionadas às áreas de *hardware*. Uma vez que já tinham sido definidas onde

era desejado que os sensores fossem posicionados, a estrutura física do carro e dos equipamentos foram analisadas com o intuito de projetar suportes que possibilitariam uma leitura confiável e de qualidade do ambiente com o veículo em movimento. Os suportes foram impressos em 3D para validação e em seguida enviados para confecção em aço.

Dessa forma, foram projetados circuitos elétricos de *driver* que foram adicionados à placa de circuito impresso do sistema embarcado que engloba também microcontrolador, circuitos de comunicação e periféricos. A placa foi roteada e a partir das revisões enviada para confecção. Em seguida foi modelada e impressa em 3D uma caixa de proteção para o sistema embarcado. Foi projetada e confeccionada também a placa de segurança para o veículo, com o intuito de assegurar segurança durante a navegação autônoma, onde o controle pode ser retomado em caso de alguma pane.

Assim, foi construído para o monitoramento do funcionamento do robô e detecção de erros um módulo gráfico de diagnóstico, capaz de mostrar em uma página WEB as variáveis de relevância e o estado do veículo de forma remota.

Depois, foram realizadas as simulações com o veículo em ambiente simulado, realizando testes de quais pacotes seriam utilizados para a navegação, mapeamento e controle. Com isso, foram usados pacotes de SLAM para mapear ambientes virtuais diferentes e com isso julgar seu desempenho. Também foram realizados testes com pacotes de navegação, testando a navegação autônoma do robô em simuladores (MACENSKI et al., 2020).

Resultados e discussão

Com os estudos realizados e a revisão bibliográfica, foi possível aprender e entender sobre o funcionamento de um robô autônomo e o que é necessário para que ele funcione com independência e inteligência. Por meio dos desafios realizados, trabalhando com a navegação autônoma ponto-a-ponto com desvio de obstáculo, desenvolveu-se uma melhor noção da implementação das técnicas de controle reativo para a movimentação do veículo.

Com a realização do modelo simulado do carro de Golf no *software* Gazebo, foi utilizado o modelo de *Ackermann* o seu controle, utilizando um *plugin* que foi adicionado ao arquivo de descrição do robô, URDF. Neste arquivo, foram descritas todas as partes do robô que são interessantes para o seu funcionamento, como rodas, esterçamento, sensores, tudo com seu devido tamanho e posicionamento similar ao modelo real, resultando no modelo da Figura 1. Foram adicionados sensores simulados para GPS, IMU, LiDARs, encoders e

câmera, de forma que o robô simulado tivesse o mesmo sensoramento do real, permitindo que fosse realizados ensaios em ambiente virtual que pudessem ser reproduzidos no mundo real. A Figura 2 mostra o modelo no *software* RViz, onde podem ser vistas todas as transformações entre os componentes do veículo, que são utilizadas para as conversões de referencial nos algoritmos de percepção, navegação e mapeamento.



Figura 1 – Modelo simulado do veículo no *software* Gazebo. Fonte: Autor.

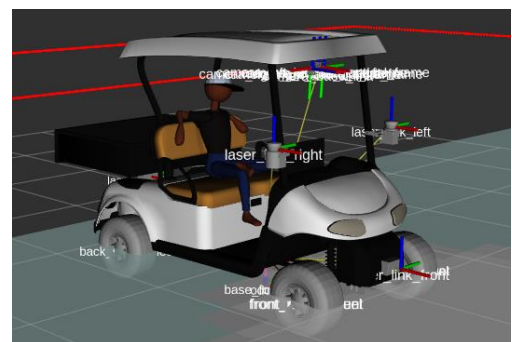


Figura 2 – Modelo simulado do veículo no *software* RViz. Fonte: Autor.

Para a fixação dos sensores reais no veículo, foram modelados utilizando o Autodesk Fusion 360 suportes robustos para suportarem os sensores durante a navegação do robô, minimizando oscilações e tendo boa durabilidade. Eles foram posicionados no carro tentando aproveitar ao máximo a estrutura original, evitando de realizar grandes modificações. Na Figura 1 são exibidos os suportes modelados para o LiDAR de 270° posicionado nas duas laterais do carro de golf. Eles foram projetados de forma que possuam dois graus de liberdade, podendo ser ajustado a posição do ângulo do sensor no eixo vertical e no eixo paralelo à direção do carro. Também na Figura 1, está o suporte projetado para o LiDAR de 120° e 4 camadas, situado na frente do robô, que possui também dois graus de liberdade, para ajuste de altura do sensor de ângulo no eixo perpendicular da direção do carro. Os graus de liberdade são de suma importância para ajustes de medição, obtendo assim uma

melhor percepção.



Figura 3 – Suportes para os LiDAR's laterais (esquerda) e frontal (direita). Fonte: Autor.

Já a Figura 4 mostra o suporte modelado para fixar os encoders às rodas do carro. Ele foi construído de forma à facilitar a transmissão de movimento da roda para o sensor, de forma pouco invasiva e discreta. Além desses suportes, foram realizados os projetos dos suportes para a câmera estéreo, LiDAR traseiro e para o encoder situado no volante.

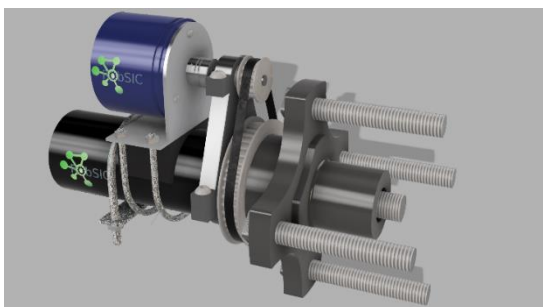


Figura 4 – Suportes para os encoders das rodas. Fonte: Autor.

Para o comando do carro, o sistema embarcado deve ser capaz de ler e escrever sinais analógicos e digitais de diferentes níveis e faixas de tensão. Para isso, foram feitos *drivers* utilizando amplificadores operacionais, opto acopladores e relés para o interfaceamento com a elétrica do carro de golf, permitindo controlá-lo. Juntamente com esses componentes, existem também circuitos para a comunicação com o alto nível (computador), por meio de comunicação serial, e também com outros sensores, utilizando o barramento CAN. A placa realiza também a leitura dos pulsos dos encoders e de outros sensores, disponibilizando todos esses dados para o alto nível através do uso do MicroROS, que por meio de um agente serial, consegue se comunicar com o ROS 2 no computador, interligando alto e baixo nível dentro do escopo do ROS. O projeto da placa de circuito para o sistema embarcado é mostrado na Figura 5.

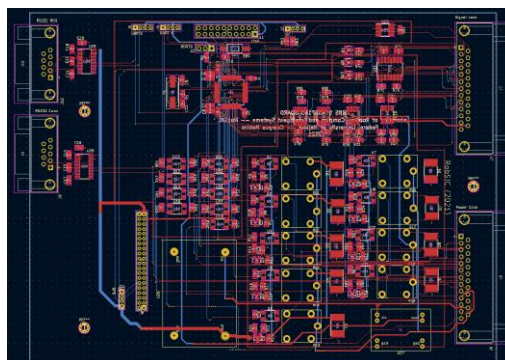


Figura 5 – Placa do sistema embarcado. Fonte: Autor.

Porém, uma vez que é possível controlar o veículo de forma eletrônica, utilizando a placa mostrada anteriormente, é necessário que exista um circuito elétrico de segurança que possa ser acionado para retomar o controle manual do veículo no caso de qualquer falha do sistema de navegação. Para isso, foi projetada e confeccionada a placa de circuito impresso mostrada na Figura 6, onde ao ser pressionado uma botoeira de emergência, os contatos dos relés são comutados, devolvendo o controle dos pedais e do volante ao motorista.

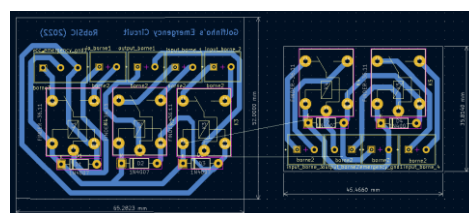


Figura 6 – Placa de segurança. Fonte: Autor.

A arquitetura mostrada na Figura 7 foi utilizada como modelo para o desenvolvimento do módulo de diagnóstico, onde, com base nas informações disponíveis publicadas via ROS 2 (alto nível) e MicroROS (baixo nível), foi criado um nó do ROS 2 para fundir esses dados de interesse em um arquivo JSON, que é enviado via FTP para uma página WEB disponível no servidor local.

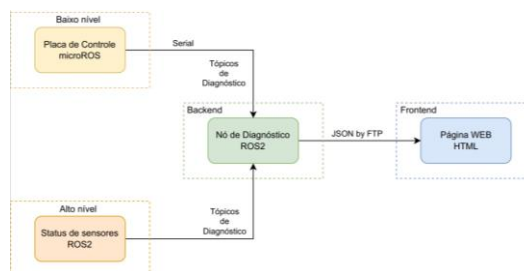


Figura 7 – Diagrama de funcionamento do módulo de diagnóstico. Fonte: Autor.

Com isso, é mostrada na Figura 8 uma página WEB básica construída para exibir os dados do diagnóstico

enviados via FTP, que pode ser acessada por diversos dispositivos, como computadores e *smartphones*, desde que estejam conectados à rede.



Figura 8 – Página WEB básica criada para teste do módulo de diagnóstico. Fonte: Autor.

Prosseguindo para o teste de algoritmos de navegação e SLAM, foi utilizado em ambiente simulado o pacote Cartographer para realizar o mapeamento e localização utilizando os dados dos LiDAR's, encoders e IMU. O mesmo obteve um bom desempenho após de alguns ajustes finos no modelo e parametrização. Pode-se ver na Figura 9 o ambiente que foi percorrido pelo carro e o mapa que foi gerado por meio do algoritmo.

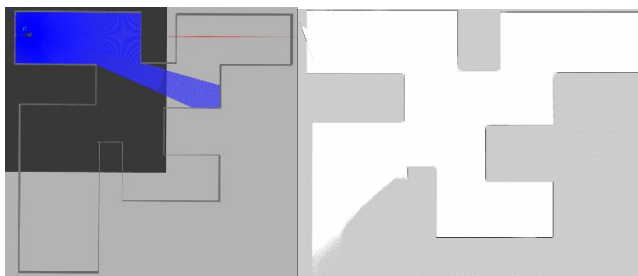


Figura 9 – Mapeamento utilizando SLAM (esquerda ambiente simulado, direita mapa criado). Fonte: Autor.

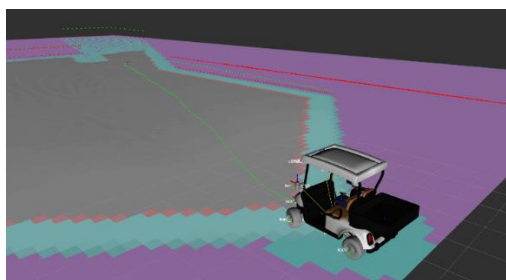


Figura 10 – Navegação utilizando o *Navigation2*. Fonte: Autor.

Já na Figura 10, com base no mapa do ambiente que estava sendo gerado via SLAM, foi aplicado o pacote *Navigation2* para realizar a navegação do ponto de origem até o ponto de destino, por meio da trajetória planejada mostrada em verde. Dessa forma, o robô teve a capacidade de a partir da percepção do ambiente, se locomover pela melhor rota até o ponto de destino, e

atualizando sua rota sempre que necessário para fins de desvio de obstáculo ou de otimização do caminho.

Conclusões

Com base na problemática de realizar a navegação de um veículo autônomo com base em sensoriamento e percepção local, vê-se que é um processo que utiliza de muita interdisciplinaridade, misturando diversas áreas para que o produto final seja alcançado. Foi visto todos os recursos de baixo nível necessários para realizar o controle do robô e conseguir medir todas as variáveis utilizadas em algoritmos de navegação. Comunicando com o alto nível, são enviados dados que aplicados em algoritmos de SLAM e navegação, são capazes de mapear o ambiente, localizar o robô no mapa, planejar a rota do veículo até o ponto de destino e fazer a navegação do mesmo sem haver colisão.

De acordo com os resultados obtidos, foi possível realizar a utilização dos sensores para a percepção do ambiente e a navegação e mapeamento em ambientes simulados. Foram realizados também testes em bancada com o sistema embarcado que apresentou bons resultados. A aplicação da placa de emergência no veículo real cumpriu com a função esperada e os suportes projetados propuseram uma ótima fixação dos sensores no carro de golf.

Como trabalhos futuros, é proposto a continuação dos testes em ambiente real, utilizando a plataforma robótica real, validando os algoritmos simulados. Além disso, propor novos algoritmos e desafios, ampliando a capacidade de inteligência do veículo e o tornando cada vez mais eficiente aplicável à diferentes ambientes e tarefas.

Agradecimento

Agradeço à empresa Vale, pelo apoio recebido para o desenvolvimento deste trabalho.

Referências

- de ALMEIDA, W. G. et al. Conception of an electric vehicle's robotic platform developed for applications on cts. In: IANO, Y. et al. (Ed.). Proceedings of the 3rd Brazilian Technology Symposium. Cham: Springer International Publishing, 2019. p. 123–129.
- MACENSKI, S. et al. The marathon 2: A navigation system. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). [s.n.], 2020.
- TAKETOMI, T.; UCHIYAMA, H.; IKEDA, S. Visual slam algorithms: a survey from 2010 to 2016. *IPSJ Transactions on Computer Vision and Applications*, v. 9, n. 1, p. 16, Jun 2017.