

UMA METODOLOGIA PARA A RESOLUÇÃO DO DESAFIO DA OLIMPÍADA BRASILEIRA DE ROBÓTICA - O ROBÔ VOLK

André Barbosa¹, Elizandro Gomes Fagundes¹, Jucemar Paes¹, Renan Rocha Darós²,
Werther Alexandre de Oliveira Serralheiro³

¹IFSC Araranguá - Curso Técnico Integrado em Eletromecânica
<barbosa1702;gomes.walker;jocepaes@hotmail.com>

²UFSC Araranguá - Engenharia de Computação - <renandaros@hotmail.com>

³IFSC Araranguá - Departamento de Eletromecânica - Docente - <werther@ifsc.edu.br>

Resumo: O presente trabalho divulga a metodologia empregada por uma equipe do Campus Araranguá do IFSC para a resolução da tarefa a ser executada na Olimpíada Brasileira de Robótica. Uma arena é composta por três salas e o objetivo é que um robô siga por uma linha, desvie de obstáculos e resgate vítimas em um ambiente hostil. O robô foi desenvolvido na plataforma Lego Mindstorm e apresenta peculiaridades na programação que demonstram resultados expressivos na execução da tarefa. Este robô conseguiu o terceiro lugar dentre 25 robôs na Etapa Regional de Santa Catarina.

Palavras-Chave: robótica móvel, olimpíada brasileira de robótica, algoritmo

1 INTRODUÇÃO

A Olimpíada Brasileira de Robótica - OBR (2016a) é uma das olimpíadas científicas brasileiras apoiadas pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq - que utiliza-se da temática da robótica para estimular jovens às carreiras voltadas para áreas tecnológicas. O evento é realizado desde 2007 de forma anual, e é composto por duas modalidades: (i) modalidade teórica é composta por uma prova de conhecimentos curriculares do ensino fundamental e médio, cujo tema central é a robótica e, (ii) modalidade prática é composta por uma tarefa realizada autonomamente por um robô construído pelos próprios alunos.

Para o evento da modalidade prática de 2016, uma equipe formada por três alunos do Curso Técnico Integrado em Eletromecânica do Campus Araranguá do IFSC desenvolveu um robô para realização da tarefa proposta. Mesmo sendo um evento competitivo, o espírito público de difusão da ciência e tecnologia que norteia o tripé ensino, pesquisa e extensão no Instituto Federal de Santa Catarina (IFSC, 2014) motivou os educandos a publicarem a metodologia que desenvolveram para o evento.

O objetivo deste trabalho é apresentar um algoritmo inovador para a resolução da tarefa de seguimento de linha, desvio de obstáculos e resgate da “vítima”. Para tanto, no capítulo 2, o desafio da OBR é apresentado. No capítulo 3, a metodologia desenvolvida é detalhada. Os resultados são discutidos no capítulo 4.

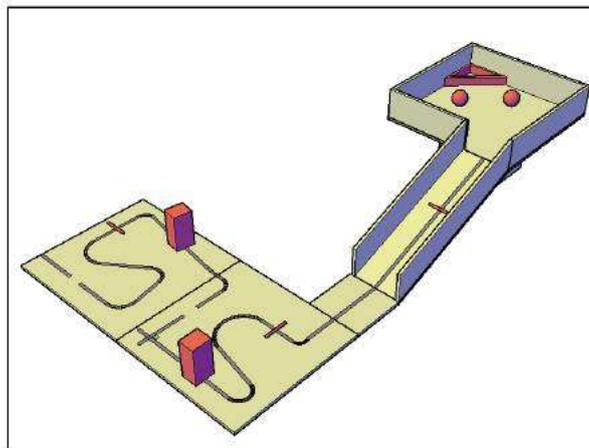
2 O DESAFIO

A missão se caracteriza por simular um ambiente de desastre em mundo real onde o resgate de vítimas precisa ser feito por robôs. Em um ambiente hostil, o robô precisa ser completamente autônomo para cumprir sua missão de seguir uma trilha cheia de obstáculos e desafios, e salvar uma vítima transportando-a para uma área segura. (OBR, 2016b)

Este ambiente simulado é composto por três salas. Nas duas primeiras salas, Sala 1 e Sala 2 que estão ao nível do solo, existem linhas pretas dispostas aleatoriamente para guiarem o caminho do robô. Durante este percurso poderão existir elementos para dificultar seu trajeto: (i) obstáculos, que são caixas retangulares com 25cm de altura, 10cm de comprimento e 12cm de largura dispostas sobre a linha, (ii) redutores de velocidade, barras roliças de 1cm de diâmetro deitadas sobre a linha, (iii) gaps, que são descontinuidades da linha preta e (iv) encruzilhadas marcadas com uma marca verde na qual o robô deverá seguir dobrando pelo lado em que a marca se encontra.

Subindo uma rampa, que poderá conter redutores de velocidade e/ou gaps, o robô chegará à Sala 3. Nesta sala, entre uma e três vítimas (uma bola de isopor coberta com papel alumínio) são dispostas aleatoriamente em qualquer posição, assim como um local de resgate, disposto necessariamente em um dos cantos da sala. A Figura 1 ilustra uma possível configuração do ambiente simulado.

Figura 01 – Uma possível configuração da Arena da Etapa Regional



Fonte: os autores

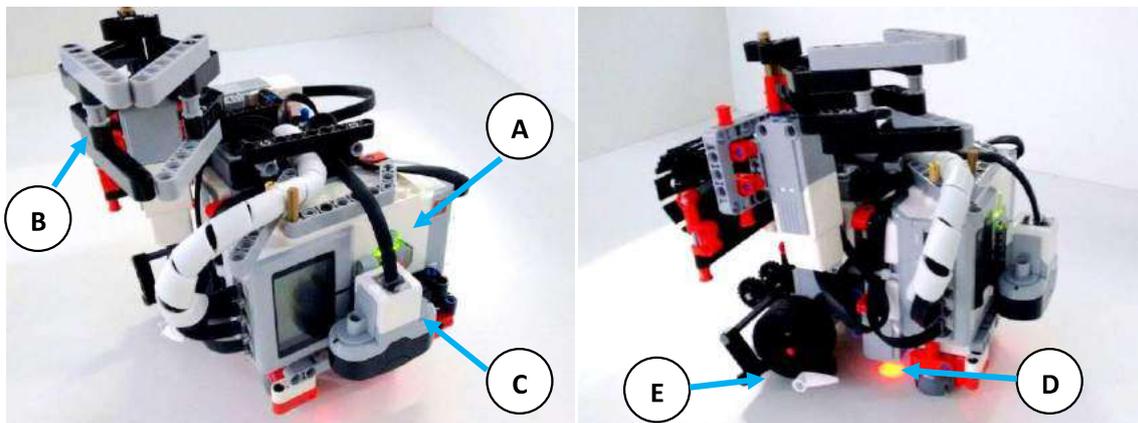
O objetivo do robô é percorrer as salas 1 e 2 seguindo a linha e desviando dos possíveis obstáculos, subir a rampa, identificar vítimas e levá-las ao local de resgate.

3 METODOLOGIA

3.1 Montagem mecânica

Um robô foi desenvolvido utilizando peças do kit Lego Mindstorm Education - EV3. Este kit possui 541 peças para montagem mecânica, um *brick* microcontrolador com 4 canais de entrada e 4 de saída, 3 servomotores, além de sensores de rotação giroscópica, de distância por ultrassom e de luminosidade. A Figura 2 ilustra o robô montado, em duas perspectivas diferentes, indicando algumas de seus principais componentes: (A) o *brick* microcontrolador, (B) a garra, (C) o sensor ultrassônico, (D) o sensor de luminosidade e (E) roda com seu apêndice.

Figura 02 –O robô Volk



Fonte: os autores

3.2 Programação para as Salas 1 e 2 e para a Rampa

A programação do robô foi desenvolvida na plataforma do Lego Mindstorm EV3 (ROLLINS, 2014). O sensor de luminosidade (Figura 02, D) opera em dois modos: sensor de cor ou reflexivo. No modo reflexivo, após calibração, o sensor retorna valores inteiros entre 0 e 100, diretamente proporcional à quantidade de luz recebida. Usando este sensor, foi desenvolvido um procedimento para o seguimento de linha, ilustrado no Algoritmo 01.

A função *potencia_do_motor* possui dois argumentos: o identificador do motor (esquerdo ou direito) e a velocidade, que pode variar entre -100 e 100. Um mapeamento é realizado entre o valor lido pelo sensor e a velocidade dos motores.

Algoritmo 01 – Pseudo-código do seguidor de linha**PROCEDIMENTO** Seguimento de Linha

```

1:  Enquanto não (chegou na sala 3)
2:      potencia_do_motor(esquerdo, 2*sensor_esq - 100)
3:      potencia_do_motor(direito, 2*sensor_dir - 100)
4:      Se (sensor_ultrasom < dist_obstaculo)    desviar_obstaculo
5:      Se (sensor_esq = verde)                  gira_para_esquerda
6:      Se (sensor_dir = verde)                  gira_para_direita
7:      Se (sensor_esq < 50 e sensor_dir < 50 e sensor_redutor < 5)
8:          pulso_para_frente
9:      Fim-Se
10: Fim-Enquanto

```

Fonte: os autores

Quando a linha da trajetória está entre os dois sensores reflexivos, os sensores estão sobre a superfície branca da arena e retornam o valor máximo de luminosidade. Então, os dois motores moverão o robô pra frente à velocidade máxima. À medida que a linha da trajetória começar a sair do centro do robô, o sensor de um dos lados começará a receber valores de luminosidade cada vez menores, forçando o motor do mesmo lado a diminuir a velocidade, colocando assim o robô novamente sobre a linha. Em casos extremos, quando um sensor estiver completamente sobre a linha preta, o motor irá receber valores negativos de velocidade e irá girar no sentido oposto, corrigindo a direção do robô e o forçando a voltar para cima da linha da trajetória.

Caso o sensor ultrasônico (Figura 02, C) encontrar um obstáculo a uma distância mínima <dist_obstaculo>, o procedimento *desviar_obstaculo* é chamado. Seu pseudocódigo é ilustrado no Algoritmo 02. O parâmetro <v_desvio> determina a velocidade que o robô irá realizar a tafera, e <delta_desvio> define o raio de curvatura na qual o robô irá desviar o obstáculo. Os valores dos parâmetros são definidos de forma empírica em testes na arena.

Algoritmo 02 – Pseudo-código para o desvio de obstáculos**PROCEDIMENTO** desviar_obstaculo

```

1:  girar 90° direita
2:  Enquanto não (sensor_esq=100 ou sensor_dir=100)
3:      potencia_do_motor(esquerdo, v_desvio)
4:      potencia_do_motor(direito, v_desvio+delta_desvio)
5:  Fim-Enquanto

```

Fonte: os autores

Para resolver o problema do redutor de velocidade foi utilizado um sensor infravermelho virado para baixo para identificar se o robô estava numa pequena elevação. Assim, se ambos sensores de refletância estiverem lendo um valor menor que cinquenta e o sensor de infravermelho estiver lendo um valor maior cinco, o robô sabe que está no redutor e para ultrapassá-lo ele dá um pequeno pulso para frente.

Além disso, foi acoplado em ambas as rodas do robô apêndices (Figura 02, E). Estes aumentam a sustentação da roda quando encontram os redutores de velocidade, facilitando sua ultrapassagem.

3.2 Programação para a Sala 3

Entre a Rampa e a Sala 3 existe uma faixa prateada delimitando os ambientes. Quando o sensor de cor encontra esta faixa, o procedimento ilustrado no Algoritmo 03 é executado.

Algoritmo 03 – Pseudo-código para o resgate da vítima

PROCEDIMENTO Sala_3

```

1:  parar
2:  girar 45° direita
3:  SE (detectar obstáculo) girar 90° esquerda \\ entrou pela direita
4:  andar 60cm frente                          \\ até centro da sala
5:  dist_vitima = 1000cm
6:  Enquanto (houver_vitima)
7:  Enquanto (angulo < 360)
8:      girar direita                          \\ girar guardando
9:      Se (sensor_ultrasom < dist_vitima)     \\ menor valor (vitima)
10:         dist_vitima = sensor_ultrasom
11:     Fim-Se
12: Fim-Enquanto
13: Enquanto (sensor_ultrasom > dist_vitima)
14:     girar direita                          \\ gira até a vítima
15: Fim-Enquanto
16: andar dist_vitima frente
17: girar 180° direita                        \\ garra na traseira
18: abrir_garra; descer_garra
19: fechar_garra; subir_garra                \\ pegou vítima
20: Enquanto não (identifica_area_resgate)
21:     girar direita                          \\ gira até area resgate
22: Fim-Enquanto
23: andar até dist_area_resgate                \\ vai até area resgate
24: girar 180° direita
25: descer_garra; abrir_garra
26: subir_garra; fechar_garra                \\ resgatou vítima
27: andar 60cm tras                          \\ voltar ao centro da sala
28: Fim-Enquanto
29: FIM

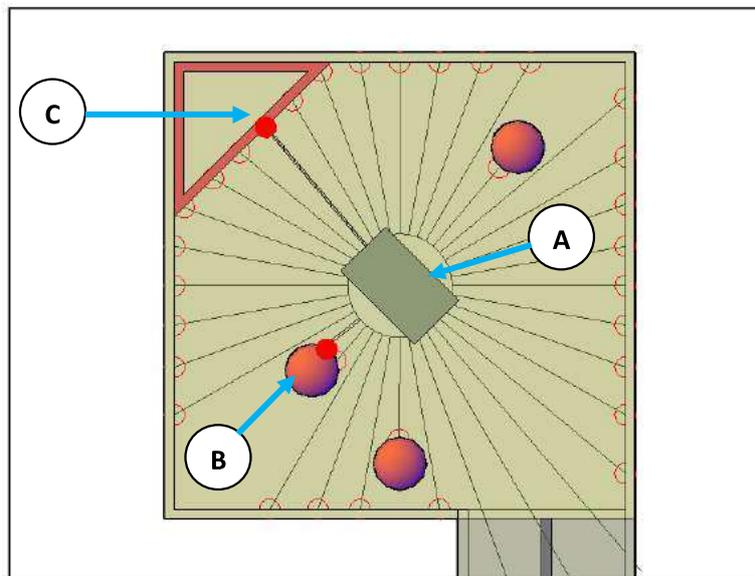
```

Fonte: os autores

A Figura 03 ilustra o cenário da Sala 3. O robô verifica o lado em que ele entrou, corrige a sua posição e então se desloca até o centro da sala (Figura 03, A). Nesta posição, realiza um rastreamento de uma vítima, que se encontra necessariamente na menor distância entre o robô e qualquer uma das leituras de distância lidas pelo sensor ultrassônico (Figura 03, B). Então, vai até a vítima e a pega com sua garra (Figura 02, B), retornando ao centro da Sala 3.

Então, realiza novamente um rastreamento, desta vez em busca de um padrão reconhecido como o da Área de Resgate (Figura 03, C). O robô se desloca até a a Área de Resgate e solta a vítima e, em seguida, retorna até o centro da sala para realizar novo rastreamento de vítima.

Figura 03 –O robô na Sala 3



Fonte: os autores

4 RESULTADOS E DISCUSSÃO

Uma arena com as mesmas características da OBR foi construída. Nesta arena, foram realizados dez testes completos, com a pontuação máxima de 250 pontos. Durante estes testes foram observados a pontuação obtida na tarefa, bem como o tempo necessário para completá-la.

Todos os dez testes foram completados dentro do tempo limite de 5 minutos. A média de tempo foi de 3 minutos e 22 segundos, e o desvio padrão de 46 segundos. Vale frisar que o critério de tempo é classificatório apenas para a condição de empate em pontos.

A pontuação média nos dez testes foi de 154 pontos (61,6% dos pontos válidos), com desvio padrão de 54 pontos (ou 21% de desvio). Em um dos testes, o robô conquistou a pontuação máxima (sem erros de percurso e com resgate de todas as vítimas), e em quatro delas apenas com desconto de pontos por erro em percurso, mas restagatando algumas vítimas.

5 CONSIDERAÇÕES FINAIS

O robô Volk demonstrou ser competitivo, robusto e repetitivo nos testes realizados. Como resultado, conquistou o terceiro lugar dentre 25 equipes na Olimpíada Brasileira de Robótica de 2016.

AGRADECIMENTOS

Este trabalho é um dos resultados parciais do projeto "Nucleação de Equipes de Robótica Móvel no IFSC Araranguá", com apoio do CNPq (nº 467927/2014-6), o qual os autores oferecem gratidão.

REFERÊNCIAS

ROLLINS, Mark. **Beginning Lego Mindstorm EV3**. Nova Iorque: Springer Verlag, 2014.

OLIMPÍADA BRASILEIRA DE ROBÓTICA. **O que é a OBR?**. Disponível em <www.obr.org.br/?page_id=9>. Acesso em 02 de agosto de 2016.

_____. **Regras e Instruções – Provas Regionais/Estaduais Modalidade Prática 2016**. Campinas: OBR, 2016. 46p.

IFSC. **Plano de Desenvolvimento Institucional do IFSC (2015 - 2019)**. Disponível em <pdi.ifsc.edu.br>. Acesso em 02 de agosto de 2016.