

eRoboticaLEGO Team

IEEE Standard Education Kits

LARC 2008

Débora D. de Melo*, Carlos A. A. A. Neto*, Marcus A. R. Tenório*, Erivaldo X. L. Filho*, Antonio M. N. Lima*
*eROBOTICA Project

Department of Electrical Engineering
Center for Electrical Engineering and Informatics
Universidade Federal de Campina Grande
Avenida Aprígio Veloso, 882 - Bodocongó, 58109-970, Campina Grande, PB, BRAZIL
{debora.melo, carlos.antonio, marcus.tenorio, amnlima}@ee.ufcg.edu.br
{erivaldo.xavier}@dsc.ufcg.edu.br

Abstract—Esta descrição da equipe, descreve algoritmos e idéias para o desenvolvimento e construção de robôs pelo Time eROBOTICA LEGO para a categoria IEEE Standard Education Kits na Competição Latino-Americana de Robótica, 2008.

Index Terms— Kits de Robótica, NQC, LEGO

I. INTRODUÇÃO

Há anos os módulos educacionais vêm sendo utilizado pelas universidades do mundo a fora, sempre com intuito de incentivar os graduandos a modelarem os problemas reais e suas soluções, através de robôs infinitamente mais baratos. No discorrer deste trabalho, busca-se ilustrar a importância de se “prototipar” e testar novas configurações de robôs antes de construí-los. [1] O plano de fundo para essa ilustração seria O Espaço, onde se tem a cooperação entre dois robôs com a finalidade de restaurar o computador principal de uma nave espacial, que irá colidir com a terra em 5 minutos. Para isso, um dos robôs deverá buscar uma fita de backup (responsável pela restauração do computador central), que só pode ser ativada por senha, que por sua vez deverá ser encontrada por outro robô. Nesse ínterim, paradoxalmente ao tempo em que as ferramentas educacionais parecem simplificadas em relação à construção de um robô real, outros problemas acabam surgindo devido a essa simplificação, tal como o domínio completo e preciso das ferramentas básicas, ou seja, motores e sensores, que muitas vezes mesmo na modelagem exigem uma potência eficiente, no caso dos motores, e uma precisão de leituras, no caso dos sensores. Dessa forma, esse foi um dos grandes desafios: tentar contornar a ineficiência dos elementos de hardware, através do software. Outra grande deficiência, que podemos enumerar, é a falta de rigidez do corpo do robô com módulos educacionais de LEGO Mindstorms. Dessa forma, o grande desafio dessa modelagem é saber utilizar os aspectos positivos da mesma, deixando que os aspectos negativos não prevaleçam.

II. CONSTRUÇÃO DOS ROBÔS

A. Robô A

Para a construção do primeiro robô, responsável pela busca do backup, simulado por um cubo preto, foi necessária a elaboração de uma garra, que deslizasse paralelamente à parede das chamadas cabines (onde se encontram empilhadas várias outras backups, sendo da cor azul), que procure e ache a backup correta: cor preta. A garra dotada desse tipo de deslizamento foi essencial também para que se possa encaixar o cubo no devido lugar da aeronave, representado por um furo na parede da arena. A garra, dotado desse tipo de deslizamento foi essencial também para que se possa encaixar o cubo no devido lugar da aeronave, representado por um furo na parede da arena. Na garra ainda, utiliza-se para a montagem peças do kit LEGO Mindstorms, versão RCX, porém motores, sensores e controlador (“tijolo”) do kit LEGO Mindstorms versão NXT, que será primordial para a futura comunicação com o robô II (responsável pela senha), uma vez que é dotado do sistema Bluetooth de comunicação. Utilizou-se, nessa parte do robô, um sensor de toque, para identificar o momento em que a garra encosta na parede e começa a deslizar, tal como um sensor de luz para identificar a tonalidade das cores dos blocos. Assim como dois motores: um para controlar a abertura e fechadura da garra e outro para o deslizamento paralelo da mesma. Quanto ao movimento, obteve-se um robô dotado de rodas, por privilegiarmos sua estabilidade e eficiência. Utilizou-se peças de LEGO Mindstorms, versão RCX, inclusive motores, sensores e controladores (“tijolo”) dessa mesma versão, respeitando-se sempre a limitação de seis sensores e seis motores. Na movimentação do robô, utilizou-se dois sensores de luz para receber os valores devidos das cores da trilha e do ambiente externo a elas, tal como um sensor de toque, que identifique o momento em que o robô encosta na parede e cesse seu movimento. Utilizou-se também dois motores, para controlar o movimento do robô, efetuando

inclusive curvas.

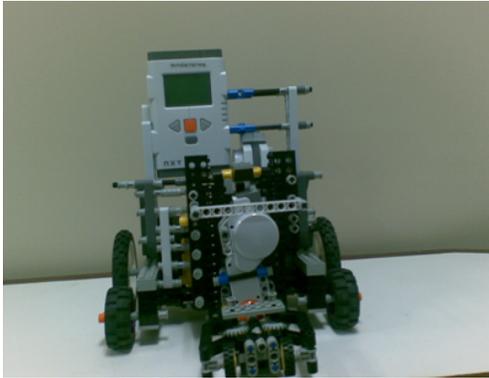


Fig. 1. Robô A

B. Robô B

Na construção do segundo robô, responsável pela busca da senha no segundo andar do prédio manteve-se o esquema de montagem do primeiro robô, isso é: o movimento da garra dotada de deslizamento paralelo. Pois, mesmo não sendo necessária a busca do bloco paralelamente ao robô, se faz necessário o mesmo esquema para que o mesmo encaixasse o bloco na parede a uma altura de 0,1m. Mantendo-se também o esquema de montagem, sendo utilizadas peças de LEGO Mindstorms, versão RCX, motores e sensores de LEGO Mindstorms, versão NXT. Nessa partição do robô, utilizou-se um sensor de luz para buscar-se o cubo da cor preta. Com relação ao movimento, dotou-se esse robô com esteiras, uma vez que são indicadas para a subida de escadas, como é o caso: duas escadas de 0,03 m cada. Utilizou-se, assim como para o primeiro robô, motores e sensores do LEGO Mindstorms, versão RCX. Já nessa partição buscou-se utilizar dois sensor para que o robô siga a trilha, tal como dois motores, para controlador o movimento do robô, efetuando inclusive curvas.

III. LINGUAGEM NQC

Not Quite C é uma linguagem simples com uma sintaxe semelhante a linguagem C, que pode ser usada no controlador programável da lego o Lego RCX(do Mindstorms set). Para aqueles que estão sendo inicializados agora em programação, um envolvimento gráfico como aqueles usados no software da Mindstorms(RIS) ou no Robolab são melhores opções. Entretanto, em competições de grande nível, as quais requerem tarefas bem mais minuciosas do que esses programas anteriormente citados conseguiriam fazer, sendo assim o NQC, tem várias bibliotecas com n funções úteis ao lego RCX. [2] Not Exactly C é uma linguagem também criada com o mesmo propósito, que seria, uma programação mas robusta e eficiente para os Bricks da lego(NXT e RCX), ela também tem uma semelhança com a sintaxe da linguagem C, além de possuir funções para a comunicação entre dois bricks. Nós do erobotica, usamos o a linguagem nqc e a nxt para uma alta precisão que esta competição necessita, podemos dar um exemplo básico disso, quando citamos o famoso código de

seguir varias linhas, que feito com qualquer dos programas criado pela lego para os kits, não conseguiram distinguir exatamente, pois o espectro muito, já isso é rapidamente resolvido, quando usamos a linguagem nqc ou nqc, [3] usamos os famosos arrays para assim calcularmos uma média entre vários espectros de cor e darmos as direções exatas aos robôs.

IV. ALGORITMOS

A. Seguimento das Linhas Azul e Preta, e Captura dos Blocos

O robô que procura o bloco nas pilhas é posicionado com seu primeiro sensor sobre a trilha azul. Uma rotina lê este sensor e classifica o valor lido. As classificações usadas são branco, branco azul, azul, azul preto e preto. Quando a leitura do sensor é branco azul, que significa que o sensor está entre a linha azul e o fundo branco, o robô segue em frente. Quando a leitura é azul, que significa que ele está desviando para o lado da faixa, ele faz a curva corrigindo. Quando a leitura é branco, ele faz a curva para o outro lado. Um segundo sensor, um pouco afastado do outro, fica sobre o fundo branco, esperando pelo estímulo da faixa preta que indica quando ele deverá fazer a curva em direção a pilha de blocos. Quando sua leitura difere de branco, o robô realiza a curva. O algoritmo para seguir a linha preta é semelhante ao usado para seguir a linha azul, apenas troca-se os valores, e espera-se que o sensor de toque indique a colisão com a lateral da arena. Então o robô alinha com a pilha de blocos, através do sensor de luminosidade e realiza quatro leituras do mesmo sensor, alterando a altura da garra onde está fixado, para a altura de cada um dos blocos. Usando essas leituras, a média dos valores lidos é calculada e é verificado qual o bloco cuja leitura foi a mais díspar da média. Esse deve ser o bloco preto, distinto dos demais azuis da mesma pilha. Usando uma porcentagem para calcular se a diferença de um bloco da média é significativa, também é decidido se o bloco preto está ou não naquela pilha. Então a garra é novamente posicionada na altura do provável bloco preto, e é fechada, capturando-o. Então o robô faz uma rotação de 180° graus e faz o caminho de volta pela linha preta até a azul, usando a mesma estratégia usada para detectar a linha preta para a linha azul neste procedimento de volta. E prossegue para a próxima pilha de blocos, caso o bloco preto não tenha sido encontrado.

B. Comunicação e sincronização entre os bricks

Após se conseguir êxito, em criar, uma comunicação via código Morse entre o brick do rcx e do nxt, pois só havia como material dois kits nxt, vimos que tal maneira, não era viável pelo grande uso de sensores, e como a organização da competição limitava os mesmos, preferimos fazer a comunicação em um robô entre sua parte inferior que usava um brick rcx e sua parte superior um brick nxt, fez-se tal comunicação pelo tempo. A comunicação entre os dois bricks, para se encontrarem no meio da arena foi feita por um algoritmo de mensagens criado usando o sistema bluetooth presente no brick nxt, esse mesmo sistema foi feito para criar a sincronização exata entre eles, baseada em um sistema de sonar, no qual o mais rápido baseado em testes, se posiciona

em um ponto Y na arena, no qual o outro necessariamente irá passar, baseado nisso começa a jogar sem parar um sinal Bluetooth o qual só cessará, quando o outro estiver chegando perto ao Receber tal sinal, ele também para e sincroniza até um ponto, pois um fica mais atrás por ser mais rápido e baseado em testes, começam a andar até chegar juntos ao local de colocar os backups. [4]

V. PERFORMANCE NOS TESTES

Ao realizar alguns testes com os robôs, obtive-se ótimos resultados apesar de terem acontecidos pequenos erros em alguns aspectos, como: * Problemas nas curvas durante o percurso cabine-linha azul: Havia uma perda no sentido ao fazer a curva para voltar ao caminho azul, em direção à outra cabine. * Problemas na comunicação entre os bricks do RCX e NXT: Não é permitido o uso de um adaptador que facilita a comunicação entre o RCX e NXT, pois havia poucos sensores de um só kit, por este motivo, tive-se que montar os robôs usando uma parte do RCX e outra parte do NXT. * Problemas na comunicação entre os dois robôs: Na etapa final da tarefa, tive-se que fazer uma sincronização entre os robôs para cumprir-la corretamente, para isso, precisava comunicar os dois robôs para que eles pudessem avisar um ao outro que tinham a fita de backup e a senha, e estavam preparados para colocar-la no computador central. Ao término dos testes, corrigiu-se o problema nas curvas, utilizando um sensor de luz extra e aprimorando a lógica de programação do robô. Com isso, obteve-se um resultado bastante satisfatório.

VI. PROJETOS FUTUROS

O eRobotica é um grupo de estudo das diversas áreas da robótica. Esta competição será usada como palanque intelectual no qual a partir da efusão e difusão do ensino da robótica e da amostragem do uso da robótica em competições de modo inteligente mostrando coisas simples como os kits educacionais de robótica podem ser usados como um grande incentivo no ensino da robótica. Pois baseados em experiências dentro do próprio laboratório vimos que o LEGO, serve como alicerce teórico e prático do ensino da robótica, sendo assim, a idéia com esta competição é a criação do espírito competidor e desafiador para os alunos ingressantes e a mostra da robótica como ciência útil ao engenheiro eletricista e ao ser humano em geral. Desenvolver novos modelos de robôs e situações reais, tais como a da competição, para estudo e desenvolvimento dos nossos graduandos para futuras participações em competições utilizando o kit educacional LEGO.

VII. CONCLUSÃO

Temos como conclusão, que a competição nos deu plena ajuda, para o desenvolvimento da robótica em nossa universidade, dado que a partir do momento, que houve interesse de alunos de outros cursos, e que, o laboratório em si cresceu, ministrando aulas, e cursos, quando a alguns meses atrás existia somente boa vontade, então esperamos obter um bom resultado, pois apesar da baixa verba, onde tivemos que utilizar a comunicação entre dois kits pelo tempo como citado acima

neste TDP, pela falta de kits nxc onde faríamos facilmente a comunicação bluetooth entre eles,, porém caso isto não ocorra, não será o fim, mas apenas um recomeço com mais esforço e dedicação, para o desenvolvimento da robótica, como área de estudo importante dentro de nosso curso.

VIII. AGRADECIMENTOS

Os autores gostariam de agradecer o aporte financeiro e o apoio dado pelo Departamento de Engenharia Elétrica(UFCG) e pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), respectivamente.

REFERENCES

- [1] T. de Freitas Oliveira Araújo, M. A. de Sousa Falcão, A. M. N. Lima, and C. F. C. L. Loureiro, "Maerobot: an open source test platform for prototyping robots," To be published: Proceedings of the 5th Latin American Robotic Symposium, 2008.
- [2] D. Baum, "Nqc programmer's guide," Versão 3.1.r5, 2002.
- [3] J. Hausen, "Programming lego robots using nqc," Versão 3.4.
- [4] —, "Not exactly c (nxc) programmer's guide," Versão 1.0.1, 2007.