Time de Futebol de Robôs do Centro Universitário da FEI biênio 2003/2004

Eder A. Penharbel, Murilo F. Martins, Murilo dos S. Silva, Ricardo C. Destro, Flavio Tonidandel e Reinaldo A.C. Bianchi

Centro Universitário da FEI - UniFEI
Av. Humberto de A. Castelo Branco, 3972
09850-901 - São Bernardo do Campo - SP - Brazil

e-mails: [fciepenharbel,destro,flaviot,rbianchi]@fei.edu.br

Abstract. *The Biênio03/04 team from the Centro Universitário da FEI has been developing a robot soccer team based on the developed team for the 2003 competition during 6º SBAI. The team works in a simple way. Instead of being driving to achieve an specific point in the x,y coordinate, the team just follows some specific directions. The results has been good and the research is still under development.*

Resumo. *A equipe Biênio03/04 do Centro Universitário da FEI vem desenvolvendo um time de futebol de Robôs baseado no time vice-campeão na competição de 2003 (II Latin American IEEE) realizada durante o 6º SBAI (Simpósio Brasileiro de Automação Inteligente). A equipe opera de forma simples onde cada robô é direcionado para direções específicas ao invés de ser direcionada por um ponto x e y específico. Os resultados tem sido bons e os estudos e desenvolvimento ainda estão em andamento.*

1. Introdução

Para competição de futebol de robôs realizada em Bauru, chamada de II Latin American IEEE Student Robot Competition, a equipe do Centro Universitário da FEI desenvolveu um algoritmo de estratégia que leva em consideração o jogo individual e o jogo em equipe dos robôs. Entretanto, tal algoritmo apresentou problemas em sua implementação, o que acabou resultando na segunda colocação no campeonato da categoria Robot Soccer 1.

Este mesmo algoritmo, para o ano de 2004, foi modificado e novamente implementado sem os probelemas do sistema anterior. Manteve sua estratégia de ao invés de mandar o robô para uma posição (x,y) como alguns times de futebol de robôs fazem [1][2], enviar o robô apenas para uma certa direção.

Este artigo esta estruturado da seguinte forma: seção 2 descreve em detalhes a estratégia da equipe. A seção 3 descreve o hardware do robô para então tudo ser concluído na seção 4.

2. A Equipe de Futebol de Robôs 2003/2004

A equipe 2003 desenvolveu um algoritmo cuja estratégia é dividida em dois grandes níveis: o nível coletivo e o nível individual. Cada um desses níveis são responsáveis pela movimentação do robô em campo, com o objetivo único de conseguir fazer e evitar gols. Esses níveis de estratégia são dependentes da visão computacional, que retorna a posição dos robôs em campo. Segue o detalhamento de cada um dos níveis e da ligação entre visão e estratégia

2.1. Visão Computacional e Estratégia

A parte de visão, desenvolvida pela equipe do Centro Universitário da FEI, detecta a posição e angulo de direção dos robôs da equipe e a posição dos robôs adversários. A visão ainda se encarrega de determinar a posição da bolinha dentro do campo.

Para os robôs da equipe, a visão retorna números aleatórios de 0 a 2 que não correspondem necessariamente os números dos robôs de 1 a 3. Assim, foi desenvolvido um algoritmo em que, inicialmente, os números dos robôs são atribuídos, e posteriormente, esses robôs são seguidos ("tracking") de modo a evitar que os números dos robôs se invertam e assim ficarem impossível de controlar.

Inicialmente, a visão retorna os robôs da equipe em certas posições. Para tanto, executa uma ação de rodar os robôs, um de cada vez, de modo a determinar qual robô está em qual posição

2.2. Trajetória da Bola

A cada instante de análise, o programa calcula a trajetória da bola e os pontos x,y em que ela irá passar nos próximos milisegundos. Essa trajetória serve tanto para posicionar o goleiro para defesa quanto para calcular o ponto de interceptação da bola.

2.3. Goleiro

O goleiro tem tratamento diferente dos demais jogadores. Ele segue regras simples para que possa ser o mais eficiente possível. A frente do gol é dividida em 8 regiões. Cada região tem o seu x,y central anotado. O Goleiro deverá se posicionar em uma dessas regiões para fazer a defesa.

Para determinar qual região o goleiro deve ir, usa-se a trajetória da bola para verificar em qual região a bola irá passar. Além disso, o goleiro ainda segue outras regras:

- Se bola perto, andar em direção a bola
- Se bola for, conforme a trajetória, perto do gol, ir para a posição por onde a bola irá passar.
- Se bola longe e robô fora de um eixo x próximo ao gol então voltar para o eixo x
- Se bola longe e robô próximo ao gol, ficar na vertical

O eixo x que determina a posição próxima ao gol depende do campo de defesa. Se for na esquerda, x será alguns pixels a mais da borda esquerda. Se for na direita, será alguns pixels a menos da borda direita.

2.4. Nível Individual

Foi definido que cada robô pode se movimentar em 8 direções:

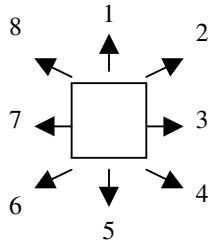

Figura 1 – direções possíveis para cada robô

Essas direções norteiam os movimentos de cada robô. Inicialmente, cada robô pode se mover em qualquer uma das 8 direções. Isso é feito por uma *vetor de inteiros* de oito elementos. Cada elemento representa uma direção. Ex: Elemento 1 → direção 1, etc. Quando o valor do elemento for '2' significa que ele pode ir para a direção. Portanto, inicialmente, ao vetor estará: 22222222

O programa de estratégia deverá decidir em qual delas ele deve se mover, eliminando as direções que ele não pode ir seguindo algumas regras:

- Para cada direção, se houver um robô naquela direção faça:
- Coloque 1 no elemento relativo a direção se o robô estiver longe
- Coloque 0 no elemento relativo a direção se o robô estiver próximo

Um 0 em um elemento significa que o robô não pode ir para a direção relacionada ao elemento.

Do mesmo modo, se a bola estiver atrás do robô, ele não poderá interceptar a bola, logo, a direção que iria interceptar a bola deverá ser 0.

Se a bola estiver a frente, o robô recebe a permissão de interceptar a bola. Conforme descrito no nível coletivo a seguir.

2.5. Nível Coletivo

Os jogadores devem se posicionar de acordo com o andamento do jogo. Foi definido assim um comportamento coletivo:

- Para cada robô, exceto o goleiro, se bola atrás então voltar. Isso significa que as direções de ataque serão postas como 0.
- Para cada robô, exceto o goleiro, se bola a frente, então zerar todas as direções de defesa.

Feito isso, tem-se apenas algumas direções possíveis para cada robô. No entanto, um refinando maior ainda faz-se necessário:

- Se houver algum robô com permissão de interceptar a bola, então, define qual robô irá interceptar a bola e em qual posição (usando a trajetória para um tempo e posição que o robô chegue mais rápido que a bola). O robô que interceptará a bola mais rápido será o escolhido para interceptar.
- O robô que irá interceptar a bola deverá ter a direção de interceptação anotada com número 3 no vetor de inteiros.
- Ao decidir qual robô irá interceptar a bola, os demais perdem a permissão de interceptação.

- Para cada robô sem permissão de interceptação, fica decidido que ele irá para uma posição de "abertura de jogo", ou seja, irá para uma direção onde não haja robô. Assim, no vetor de inteiros, mantem-se todos os numeros 2 e zera todos os numeros 1 em que haja robô a uma distância menor que um certo limite.

Com base nessas regras, espera-se que cada robô tenha uma direção ou nenhuma direção possível de movimento. Se houver nenhuma, o robô ficará parado. Se houver uma, toma-se a ação. Se houver mais de uma, escolhe-se a que tiver maior numero e aparecer primeiro no vetor de inteiros.

Decidida a direção a tomar, aplica-se a ação em cada robô, inclusive o goleiro. As ações possíveis são:

- Ficar parado
- Girar tantos graus para direita ou esquerda
- Andar para frente ou para trás uma certa distância

Quando decide-se ir para uma certa direção, seja de interceptação da bola (x,y definido) ou apenas uma direção, o programa decide se irá girar ou andar. Se o robô não estiver na direção correta, ele então decide girar, senão decide andar.

Para cada ação, uma velocidade é definida dependendo da quantidade de giro ou a distância a percorrer. Cada ciclo da estratégia, chamado de *frame*, é de aproximadamente 100 milisegundos. Para cada *frame* o programa reinicia executando sempre a seguinte ordem de comandos:

- Pega visão
- Determina posição dos robôs (nossos e adversários)
- Determina posição da bolinha
- Traça trajetória da bolinha com relação a posição anterior
- Determina ação do goleiro;
- Determina direções possíveis;
- Determina robô intercepta;
- Determina ação de cada robô com a determinada velocidade;
- Recomeça o processo

3. Hardware do Robô

Os robôs usam pequenos motores de 5V. Um estágio único de redução é utilizado para cada montagem motor/roda, com uma redução de 5:1. Em cada roda existe um reticulado reflexivo estampado, de onde um sensor infravermelho detecta informações de rotação que realimentam a unidade de controle permitindo a regulagem da velocidade. Os robôs podem desenvolver uma velocidade máxima de aproximadamente 1 m/s. A unidade de controle é composta de um microcontrolador (Atmel 89C2051) executando o software de controle em 14,75 Mhz de clock. O microcontrolador controla a tensão nos motores através de duas pontes H's completas ligadas aos pinos de E/S, permitindo o controle de velocidade e direção por PWM (*pulse width modulation*). A figura abaixo mostra os motores e as rodas com o estágio único de redução:

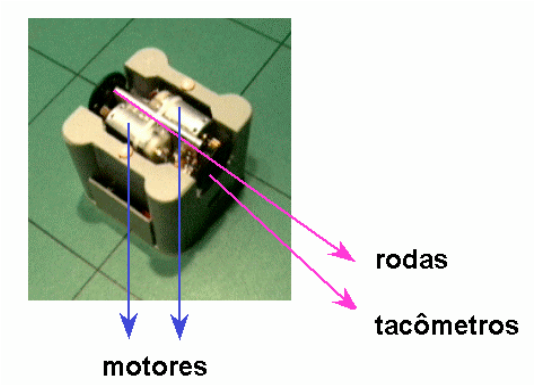

Figura 2 - Motores e rodas do robô

O sistema de comunicação que transmite os comandos do computador aos robôs é composto por um transmissor e três receptores utilizando duas frequências selecionáveis por um jumper (905 Mhz e 915 Mhz). Utilizam codificação FSK Manchester, a 9600 bps, com 2 stop bits e sem paridade. O transmissor envia mensagens aos robôs obedecendo o seguinte protocolo: 2 bits de sincronismo, 2 bits de preâmbulo, 8 bits para o robô 1, 8 bits para o robô 2, 8 bits para o robô 3 e 4 bits de verificação (check-sum), totalizando uma palavra de 32 bits por transmissão. A figura abaixo mostra o interior do robô:

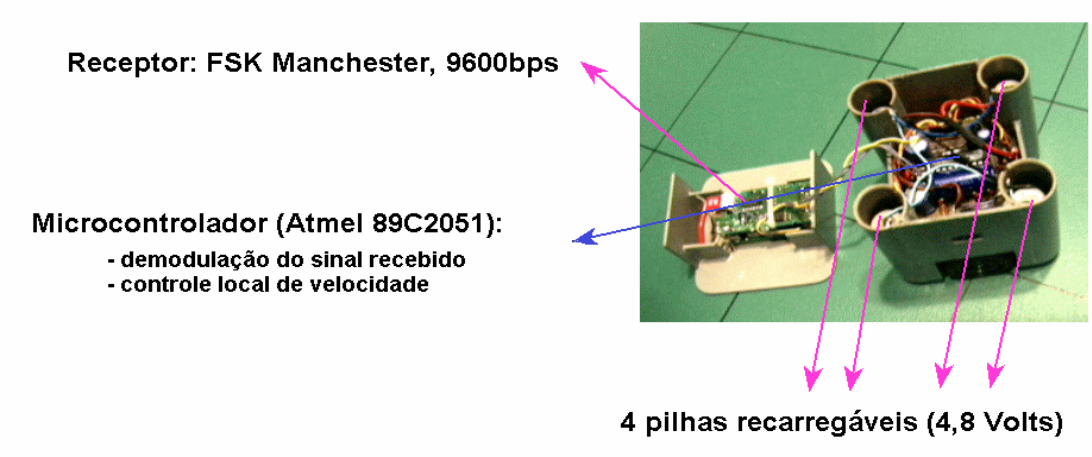

Figura 3 - módulo receptor (tampa) e microcontrolador e pilhas (interior)

4. Conclusão

O programa funcionando corretamente atualmente, embora na competição de Bauru não tenha funcionado muito bem. O sistema de visão foi reformulado e está descrito no artigo [3] e usado em ambos projetos de futebol de robôs do Centro universitário da FEI

Bibliografia

[1] . Reali, A. H.,; Pegoraro, R.; Stolfi, G. Sichman, J. Pait, F.M., Ferasoli, H. GUARANA Robot Soccer Team: Some Architectural Issue. Anais do 4° SBAI. São Paulo. 1999.

[2] . Fassi, H.; Scarpettini, F.; Santos, J. Development of the UBASOT Simulation Team. Proceedings of FIRA Robot World Congress 2003. Austria. 2003.

[3]. Penharbel, E; Destro, R.;Tonidandel, F.; Bianchi, R.; Filtro de Imagem Baseado em Matriz RGB de Cores-Padrão para Futebol de Robôs. Submetido ao I Encontro de Robótica Inteligente. 2004