UNIVERSIDADE TECNOLOGICA FEDERAL DO PARANÁ

DEPARTAMENTO ACADÊMICO DE INFORMÁTICA

ENGENHARIA DE COMPITAÇÃO

CAMPUS CURITIBA

Renato Girardi Gasoto

Renan Victor Emilio Coimbra

Crenças e ambiente para sistema de futebol de robôs

Curitiba

2011

Introdução 3

Objetivo 3

Descrição do Ambiente 3

Movimentação possível dos agentes 4

Crenças iniciais 4

Objetivo Inicial 4

Ações possíveis 4

Goleiro 4

Atacante 5

Referencias 7

# Introdução

Em um time de futebol o existem diversas técnicas para se fazer gols. A mais comum é uma estratégia formada por mais de um jogador com a intenção de desorientar o goleiro, e facilitar marcar o gol. Por isso, será implementado, em futebol de robôs, um time de robôs com três jogadores, tendo como oponente um goleiro . O time deverá elaborar uma sequencia de ações para marcar um gol.

# Objetivo

O objetivo desta etapa é apresentar, em linguagem Jason [HÜBNER, 2010], a descrição do ambiente onde os jogadores estão situados, as crenças iniciais de cada um dos agentes e as ações possíveis de serem tomadas pelos atacantes e pelo goleiro.

# Descrição do Ambiente

O campo de futebol no simulador Tewnta 1.3 [DETONI, 2008] possui o tamanho de 490x338 pixels. Como o robô possui aproximadamente 20 pixels de diâmetro, inicialmente pensou-se em dividir o campo em uma grade 25x17, de modo que em cada campo da grade seja aceito somente um robô. Como a bola não é um agente, mas sim um objeto do mundo, seria possível que esta estivesse presente em um mesmo campo da grade, junto com outro robô. Apesar de não ser uma divisão exata com o tamanho do robô, estes valores foram decididos para que haja um bloco exatamente no centro do campo.

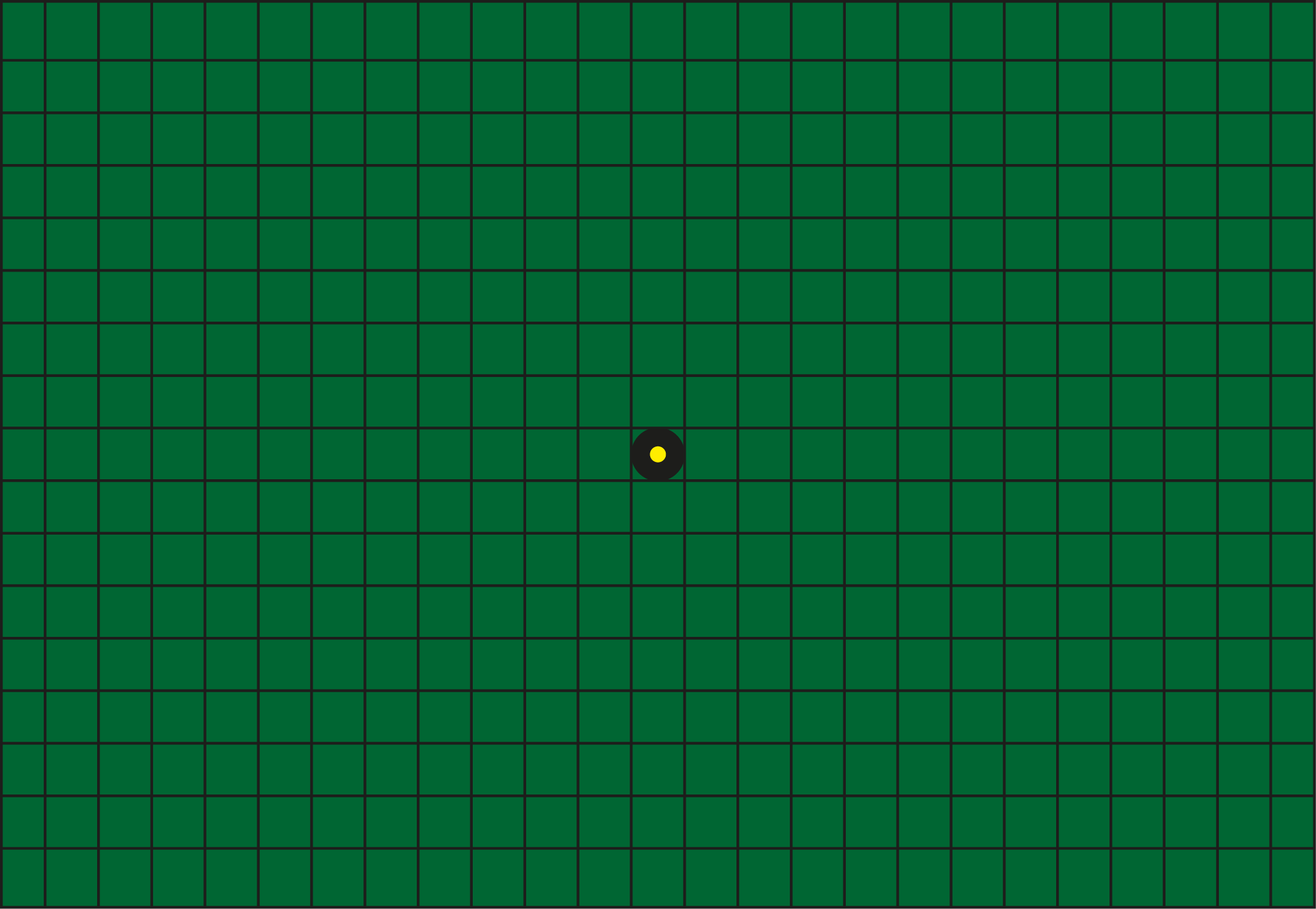


Figura Imagem do campo dividido na grade escolhida com um robô no centro

Como isto limita a movimentação do robô este modelo foi descartado.

# Movimentação possível dos agentes

O robô poderá se deslocar em qualquer direção, em linha reta, dadas as coordenadas do ponto destino.

O robô poderá, também rotacionar sobre seu próprio eixo, para poder mirar em diversas direções.

Para obter controle sobre a bola, o robô possui uma função que mantém a bola consigo em sua parte frontal, e quando necessário, chuta.

# Crenças iniciais

As crenças iniciais são os estados que o agente receberá assim que inicializar a partida. Os pontos fundamentais destacados pela equipe são:

* Posição inicial;
* A qual time pertence;

# Objetivo Inicial

O objetivo inicial dos jogadores é entrar em campo, e assim se posicionarem no mesmo e receberem o time para o qual vão jogar.

# Ações possíveis

## Goleiro

O goleiro possui como meta defender o gol. Para tanto, é definido que sua posição inicial é em frente ao gol e ao meio:

// Agente: Goleiro

/\* Initial beliefs and rules \*/

posicao(505, 200).

time(team\_b).

/\* Initial goal \*/

// Objetivo inicial: entrar em campo

!entrarEmCampo.

Após entrar em campo, o goleiro deve iniciar a estratégia de defesa:

Trecho de código 1 Declaração das ações de initial Beliefs e initial Goal para o Goleiro

Trecho de código 2 Ação Entrar em Campo para goleiro

+!entrarEmCampo : true

<- ?posicao(X, Y); ?time(Z);

createPlayer(X, Y,Z);//Cria jogador no Tewnta na posição [x,y], no time z

!defesa //inicia estratégia de defesa

A estratégia de defesa consiste em duas situações diferentes: a padrão, em que o goleiro se posiciona entre a bola e o gol, e outra, quando o goleiro está próximo da bola, que ele a chuta com a força necessária:

//caso o goleiro esteja com a bola, chutar

+!defesa : com(bola)

<- chutar(100);

!defesa.

//caso padrão

+!defesa : true

<- ?posBola(XBola,YBola);

?posicao(XGoleiro, YGoleiro);

!!olheBola;

defender(XBola, YBola, XGoleiro, YGoleiro);

!defesa.

Trecho de código 3 Planos de defesa para o goleiro

## Atacante

Apesar de no sistema haver três agentes atacantes, todos possuem o mesmo plano, doravante mesmas ações. O que muda entre eles é o initial belief, que define posição diferenciada para cada atacante:

//Atacante do meio

/\* Initial beliefs and rules \*/

posicaoIni(240, 200).

time(team\_a).

//Inclui os comportamentos de um atacante

{ include("atacante.asl") }

Trecho de código 4 Initial Beliefs para o atacante do meio, em conjunto com o include para o conjunto de ações, crenças e objetivos

Assim como o goleiro, o atacante possui como initial goal entrar em campo:

Trecho de código 5 Initial goal dos atacantes

+!entrarEmCampo : true

<- ?posicaoIni(X,Y); ?time(Z); //Obtem a posicao inicial e o time no qual vai jogar(do initial belief)

createPlayer(X, Y, Z);// cria o jogador no tewnta

!ataque. //começa estrategia de ataque.

A

O plano de ataque possui nove situações diferentes, descritas no Trecho de código 6

//caso o atacante seja o jogador mais proximo da bola, então domine a bola

+!ataque: maisPerto(bola) & not com(bola)

<- !!verBola;

?posBola(X,Y);

irLinhaReta(X,Y);

!ataque.

//se sem bola e longe do gol, vá para perto do gol

+!ataque: not com(bola) & not perto(gol)

<- !!verBola;

?posicao(X,Y);

?posicaoIni(A,B);

posicionaAtaque(420,B);

!ataque.

//se sem bola e perto do gol, posição de ataque

+!ataque: not com(bola) & perto(gol)

<- !!verBola;

?posicao(X,Y);

?posicaoIni(A,B);

posicionaAtaque(420,B);

!ataque.

//se com a bola e perto do gol, ache melhor posição para chute

+!ataque: com(bola) & perto(gol)

<- posicaoChute;

!ataque.

//se com a bola e longe do gol, passar para o companheiro mais próximo.

+!ataque: com(bola) & not perto(gol)

<- !olharCompanheiro;

passar;

!ataque.

//caso padrão: reposiciona

+!ataque: true

<- ?posicaoIni(X,Y);

irLinhaReta(X,Y);

!ataque.

Trecho de código 6 Plano de ataque

Para auxiliar em alguns destes planos, há planos secundários para determinados objetivos, descritos no Trecho de código 7

Trecho de código Planos auxiliares para o atacante

//Resgate a posicao da bola (percepcao) e rotacione olhando pra bola

+!verBola : true

<- ?posBola(X,Y);

rotacionePara(X,Y).

+!olharCompanheiro : true //pega a posicao do companheiro

<- ?companheiroMaisProximo(X,Y);

rotacionePara(X,Y). //rotacione olhando pro companheiro

# Referencias

BORDINI, R. H.; HÜBNER, J. F.; WOOLDRIDGE, M. Programming Multi-Agent

Systems in AgentSpeak using Jason. [S.l.]: Wiley, 2007.

COELHO, H. Teoria da Agência: Arquitectura e Cenograa. [S.l.]: LabMAg e ICC,

FCUL, 2008.

DETONI, G. G. Tewnta: Robocup Small Size League (SSL) F180 Simulator. 2008.

Disponível em: <http://code.google.com/p/tewnta/>.

HÜBNER, J. F.; BORDINI, R. H. Jason: a Java-based interpreter for an extended

version of AgentSpeak. 2010. Disponível em: <http://jason.sourceforge.net/Jason/Jason.html>