In the name of God

Asfaar Humanoid Soccer Simulation Team Description Paper

Seyyed Alireza Mirzargar, Seyyed Soroush HosseinAlipour, sajjad torabian
seyed mehran tavakolian

Abstract. This paper describes ASFAAR-3D humanoid soccer simulation . We explain some facts about the simulation base, the basic skills that a humanoid robot should contain and some methods that we used for producing our skills.

Keywords: Humanoid Robot, Walking Pattern, Kinematic, RoboCup Soccer Simulation

1 Introduction

Starting from a point to describe 3D soccer simulation, it is notable that a soccer agent thinks and decides about which action (In the set of all possible actions: walking, turning, shooting, standing up, etc.) should be done next. To determine what should be done as the next action, the agent must pay attention to a plenty of parameters (including: agent's position, ball's position, gyro rate, etc) Also, in order to have a good play, there should be some kind of cooperation between the agents. First of all, we produced skills with two methods: Reinforcement Learning and trajectory method. Our walk skill and back-walk skill used trajectory method and other skills used Reinforcement Learning.

2 Walking

Foot Trajectories:

Assuming that the period necessary for one walking step is *Tc* , the time of the *k*th step is from *kTc* to (*k* + 1)*Tc*, *k* = *1,2,...,K* Where *K* is the number of steps.
To simplify our analysis, The *k*th walking step begins when the heel of the right foot leaves the ground at *t* = *kTc* and ends when the heel of the right foot first touches the ground at *t* = *(k + 1)Tc.* In the following, we discuss on only the generation of the right foot trajectory. The left foot trajectory is the same except for a *Tc* delay.
For a sagittal plane, each foot trajectory can be denoted by vector $[x_{\mathrm{a}}(t), z_{\mathrm{a}}(t), \theta_{\mathrm{a}}(t)]^{\mathrm{T}}$ where $(x_{\mathrm{a}}(t), z_{\mathrm{a}}(t))$ is the coordinate of the ankle position, and $\theta_{\mathrm{a}}(t)$ is the angle of the foot.

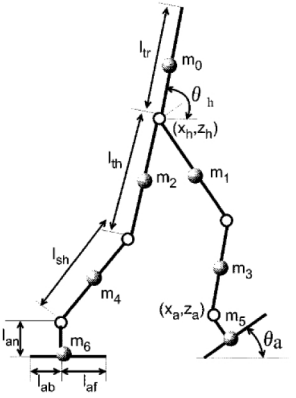
In making the foot trajectory following parameters are involved:

$$\theta_{\mathrm{a}}(t) = \begin{cases} q_{\mathrm{gs}}(k), & t = kT_{\mathrm{c}} \\ q_{\mathrm{b}}, & t = kT_{\mathrm{c}} + T_{\mathrm{d}} \\ -q_{\mathrm{f}}, & t = (k+1)T_{\mathrm{c}} \\ -q_{\mathrm{ge}}(k), & t = (k+1)T_{\mathrm{c}} + T_{\mathrm{d}} \end{cases}$$

$$x_{\mathrm{a}}(t) = \begin{cases} kD_{\mathrm{s}}, & t = kT_{\mathrm{c}} \\ kD_{\mathrm{s}} + l_{\mathrm{an}} \sin q_{\mathrm{b}} + l_{\mathrm{af}}(1 - \cos q_{\mathrm{b}}), & t = kT_{\mathrm{c}} + T_{\mathrm{d}} \\ kD_{\mathrm{s}} + L_{\mathrm{ao}}, & t = kT_{\mathrm{c}} + T_{\mathrm{m}} \\ (k+2)D_{\mathrm{s}} - l_{\mathrm{an}} \sin q_{\mathrm{f}} - l_{\mathrm{ab}}(1 - \cos q_{\mathrm{f}}), & t = (k+1)T_{\mathrm{c}} \\ (k+2)D_{\mathrm{s}}, & t = (k+1)T_{\mathrm{c}} + T_{\mathrm{d}} \end{cases}$$

$$z_{\mathrm{a}}(t) = \begin{cases} h_{\mathrm{gs}}(k) + l_{\mathrm{an}}, & t = kT_{\mathrm{c}} \\ h_{\mathrm{gs}}(k) + l_{\mathrm{af}} \sin q_{\mathrm{b}} + l_{\mathrm{an}} \cos q_{\mathrm{b}}, & t = kT_{\mathrm{c}} + T_{\mathrm{d}} \\ H_{\mathrm{ao}}, & t = kT_{\mathrm{c}} + T_{\mathrm{m}} \\ h_{\mathrm{ge}}(k) + l_{\mathrm{ab}} \sin q_{\mathrm{f}} + l_{\mathrm{an}} \cos q_{\mathrm{f}}, & t = (k+1)T_{\mathrm{c}} \\ h_{\mathrm{ge}}(k) + l_{\mathrm{an}}, & t = (k+1)T_{\mathrm{c}} + T_{\mathrm{d}} \end{cases}$$

$$\begin{cases} \dot{\theta}_{\mathrm{a}}(kT_{\mathrm{c}}) = 0 \\ \dot{\theta}_{\mathrm{a}}((k+1)T_{\mathrm{c}} + T_{\mathrm{d}}) = 0 \end{cases}$$
$$\begin{cases} \dot{x}_{\mathrm{a}}(kT_{\mathrm{c}}) = 0 \\ \dot{x}_{\mathrm{a}}((k+1)T_{\mathrm{c}} + T_{\mathrm{d}}) = 0 \end{cases}$$
$$\begin{cases} \dot{z}_{\mathrm{a}}(kT_{\mathrm{c}}) = 0 \\ \dot{z}_{\mathrm{a}}((k+1)T_{\mathrm{c}} + T_{\mathrm{d}}) = 0. \end{cases}$$



Hip Trajectory:

The hip trajectory can be denoted by a vector $[x_{\mathrm{h}}(t), z_{\mathrm{h}}(t), \theta_{\mathrm{h}}(t)]^{\mathrm{T}}$, where $(x_{\mathbf{h}}(t), z_{\mathbf{h}}(t))$ is the coordinate of the hip position and $\theta_{\mathbf{h}}(t)$ denotes the angle of the hip
In making the hip trajectory following parameters are involved:

$$z_{\mathrm{h}}(t) = \begin{cases} H_{\mathrm{h\,min}}, & t = kT_{\mathrm{c}} + 0.5T_{\mathrm{d}} \\ H_{\mathrm{h\,max}}, & t = kT_{\mathrm{c}} + 0.5(T_{\mathrm{c}} - T_{\mathrm{d}}) \\ H_{\mathrm{h\,min}}, & t = (k+1)T_{\mathrm{c}} + 0.5T_{\mathrm{d}}. \end{cases}$$

$$x_{\mathrm{h}}(t) = \begin{cases} kD_{\mathrm{s}} + x_{\mathrm{ed}}, & t = kT_{\mathrm{c}} \\ (k+1)D_{\mathrm{s}} - x_{\mathrm{sd}}, & t = kT_{\mathrm{c}} + T_{\mathrm{d}} \\ (k+1)D_{\mathrm{s}} + x_{\mathrm{ed}}, & t = (k+1)T_{\mathrm{c}}. \end{cases}$$

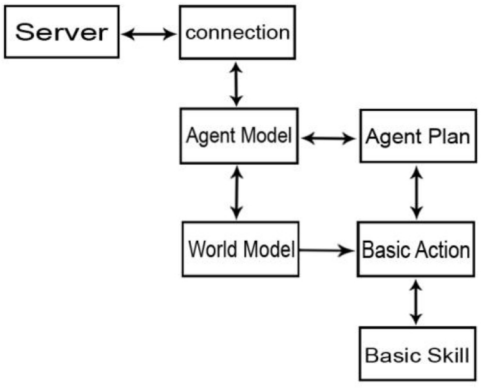
$$\begin{cases} 0.0 < x_{\mathrm{sd}} < 0.5D_{\mathrm{s}} \\ 0.0 < x_{\mathrm{ed}} 0.5D_{\mathrm{s}}. \end{cases}$$

$$\begin{cases} \dot{x}_{\mathbf{h}}(kT_{\mathrm{c}}) = \dot{x}_{\mathbf{h}}(kT_{\mathrm{c}} + T_{\mathrm{c}}) \\ \ddot{x}_{\mathbf{h}}(kT_{\mathrm{c}}) = \ddot{x}_{\mathbf{h}}(kT_{\mathrm{c}} + T_{\mathrm{c}}). \end{cases}$$

3 Behaviors

Soccer simulation server(rcssserver3d) sends some data to the agent. The agent should analyze the data and make decision to perform the best action.
Then the agent sends the specified data back to the server.
The agents must contain some procedures which manage the their behaviors.
Obviously, a normal human soccer player has got these abilities. For our purpose, the agent must have some basic skills.

We have provided our agents with two kinds of BasicSkills and AdvancedSkills. Each AdvancedSkill uses some BasicSkills, and each BasicSkill sets the properties of the joints directly and then the agent can move. The agent can be a GoalKeeper, Defender, MidFielder and Striker.In each situation, we have a plan. GoalKeeper plan, Defender plan, MidFielder plan and Striker Plan. It runs a plan according to the player type, and the plan chooses which AdvancedSkill and where it should be used for.



Some of agent's basic plans are listed below:

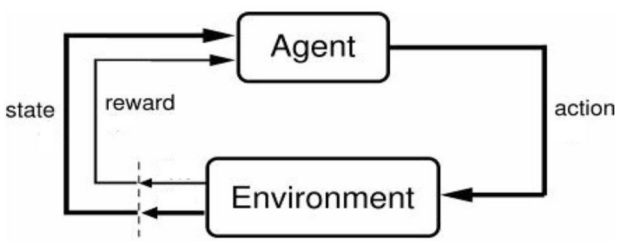
1. (GoalKeeper): The most important duty of GoalKeeper is saving the ball.
For GoalKeeper position we draw an arc. Then we draw a line from
ball to middle of the goal. As we have line and arc equations, their intersection point will be the position of our GoalKeeper.
Our Goalkeeper uses Sidewalk and diving skills. If the ball approaches the goalkeeper too much, she should dive to clear the goal. Then after some cycles it must stand up.

2. (Defender): a Defender must always be in front of the ball. So we find the ball and then sidewalk to ball position until our Y position equals the ball's Y position.
If the ball gets so close to defender she must clear it.

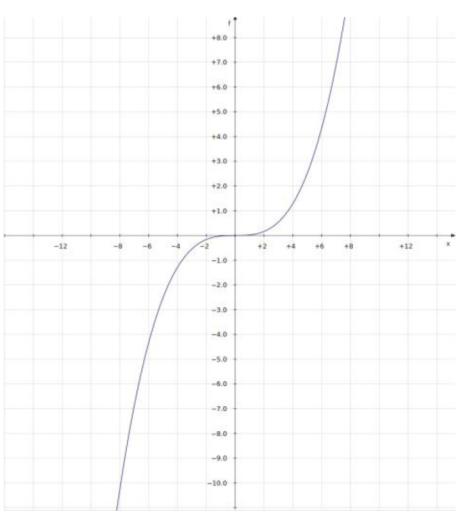
3. (Striker): At first it should find the ball. Then it should turn toward the ball.
It should stop walking near the ball and align the body with the ball and at last shoot it to the goal.

4 Humanoid BasicSkills

BasicSkills are some basic movements. Our program has some extra movements.
As we said, we used Reinforcement Learning for producing these skills.
Reinforcement learning is a sub-area of machine learning concerned with how an agent ought to take actions in an environment so as to maximize some notion of long-term reward.



We needed to have a value function because we wanted to reward the robot.
our rated range was from 0 to 255. So our value function was like this one:



We define a first action for robot and then the robot try to move better. For doing so we gave reward to robot and punished it by value function.

Some of agent's BasicSkills are listed below:

1. (Turn): A robot needs to turn because it needs to walk in different paths. during team attack, agents should walk to the opponent's goal and during the defense agents should walk to our goal. In this situation we need
to turn. So All of our players need to turn. Especially Striker uses this class so much because it needs shoot to goal of opponent in right angle.
We define some turns for some different angles.

2. (SideWalk): Our goalkeeper needs to walk left and right. So we had to make class. This class do this work. It means our agent can walk left and right.

3. (BehindStandUp): Sometimes our agent fall down. If we don't have the "standing up" skill our agent can't continue the game. We also have "front stanup" skill.