FCPortugal: Simulated Humanoid Robot Team Description Proposal for RoboCup 2008

Nuno Lau[1,3], Luís Paulo Reis[2,4], Hugo Picado[3], and Nuno Almeida[3]

[1] IEETA - Institute of Electronics and Telematics Enginnering of Aveiro, Portugal
[2] LIACC - Artificial Intelligence and Computer Science Lab., Univ. Porto, Portugal
[3] DETI - Dep. of Electronics, Telecommunications and Informatics, Univ. Aveiro, Portugal
[4] FEUP - Faculty of Engineering of the University of Porto, Portugal

Abstract. FC Portugal 2008 3D team is built upon the structure of our previous Simulation league 3D team that won RoboCup 2006 in Bremen, scoring 77 goals without conceding any. Our research is mainly focused on the adaptation of previously developed methodologies from our 2D soccer teams [1–3] to the new 3D environment and on creating new coordination methodologies based on the previously developed ones. In our 2D teams, which participated in RoboCup since 2000 with very good results, we have introduced several concepts and algorithms covering a broad spectrum of the soccer simulation research challenges. From coordination techniques such as Tactics, Formations, Dynamic Positioning and Role Exchange, Situation Based Strategic Positioning and Intelligent Perception to Optimization based low-level skills, Visual Debugging and Coaching, the number of research aspects FC Portugal has been working on is quite extensive [1, 3]. The research-oriented development of our team has been pushing it to be one of the most competitive over the years (World champion of Simulation 2D in 2000 and of Coach Champion in 2002, European champion in 2000 and 2001, Coach 2nd place in 2003 and 2004, European champion in Rescue Simulation and Simulation 3D in 2006 and World Champions in Simulation 3D in 2006). This paper describes some of the main innovations of our team in 2008 3D simulation league competition relating them with previous work developed by our simulated RoboCup teams in 2D and 3D simulation leagues. New skills have been developed for the simulated humanoid agent which includes walk, turn, kick and get up from the ground. It also includes information related to the agent architecture and low-level considerations. The current research is now mainly concerned in improving these skills using optimization algorithms such as tabu search and genetic algorithms.

1 Introduction

The introduction of humanoid agents in the simulation league, in 2007, opened several interesting research topics concerned with biped locomotion, agent architecture, embodied high-level decision, coordination and opponent modelling. Biped locomotion has been studied by years and many solutions are proposed

[4–6]. In the case of a robot, biped locomotion can be described as the control of all of its body joints to achieve a specific goal. This problem depends on the physiognomy of the robot (e.g. geometry, number and types of joints) and the information he can percept from the environment (e.g. touch sensors, and position feedback sensors) and also on the desired action (e.g. walk, turn and kick).

One major problem related with the biped motion generation is the high level control of motion behavior. The walking behavior itself is a basic skill which corresponds to a reactive and regular motion without the need for high level control. On the other side, walking to a specific point, requires some mechanism capable of starting, stopping and adjusting the walk basic skill using sensory feedback information. This mechanism is a higher level mechanism which allows the robot to achieve a more complex task based on the lower level skills.

Another issue to consider is how to integrate all the skills together. The robot must decide autonomostly when he should walk and where he should be, when and how much he should turn, get up when he falls on the ground, and kick the ball with more or less strength, depending on the situation. He should also coordinate his movements with the movements of other robots in the same team, considering their capabilities and current posture, to achieve the best possible result for the team. On the other side, he should be able to extract information on the behaviour of the opponent humanoids, and use that information to adapt its decisions.

2 Research directions

New research directions include research on agent architecture, the humanoid model and its associated restrictions in terms of dynamics, sensing, and decision, will foster the development of new layered architectures for its controlling agents. The lower layers will be responsible for the basic control of the humanoid such as equilibrium while the higher take decisions at a strategic level. Several methods for generation of humanoid behaviors are being evaluated, including tabu search, genetic algorithms, simulated annealing and reinforcement learning and how this behaviors are integreted together. Some new directions of research in FC Portugal will also include developing a model for a strategy for a humanoid game and the integration of humanoids coming from different teams in a inter-team framework to allow the formation of a team with different humanoids. Opponent modelling may be a critical module in humanoid soccer, including the opponent basic behaviors performance, its positioning, its tendency to play at the edge (or beyond) of the fair-play rules, etc. are factors that must be taken into account when selecting a given strategy for a game. Other research that may be performed with humanoids includes intelligent sensing, because the humanoids will not be able to look in all directions at the same time (this may not be the case in 2008, but it will happen with the evolution of the Simulation League). So, it is very important to choose the best looking direction considering all restrictions introduced by the dynamics of humanoid movement. Also heterogeneity will be important because in the future it is expected that not all humanoids will be identical, having humanoids with different capabilities introduces new problems of task assignment that will have to be dealt within humanoid teams. Kinematics is also being studied which allows the agent to know its body bart positions in real-time, using the vision perceptors (forward kinematics) and to know which configuration of the joints are necessary for an end effector achieve a certain position (inverse kinematics).

3 FCPortugal3D Agent Architecture

The FC Portugal Agent 3D [7] is composed by 5 main packages: WorldState, Physics, Geometry, Skills and Utils (Figure 1).

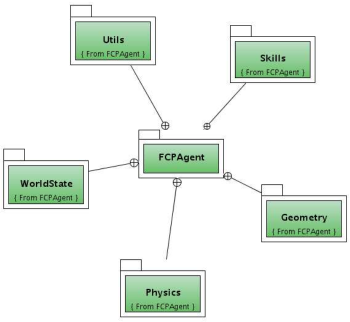

Fig. 1. Agent Architecture

The world state package (Figure 2) is probably the most complex one. It has all the information that the FCPAgent needs to decide which action it should take. There are three kinds of information that the WorldState needs: information about the objects (like players, landmarks and the ball), information about the conditions of the game (like field length, goal width, etc) and the state of the game (like the current play mode, the result, the time, etc). The physics package aims to reproduce the physical interactions between the bodies in the world as accurate as possible in the way the server does.

On the geometry package, three classes are implemented - the Vector3f, Vector and Matrix4D. Each of them provides methods to manipulate and to produce calculations with 3D vectors, 2D vectors and 4×4 matrices respectively.

The package Utils was made to contain classes that do not have a direct relevance on the agent behavior but help to make some tasks easier. Samples of the operations of those classes are the creation of log files, communication with the simulator, a message parser and a message composer to send the actions to the simulator.

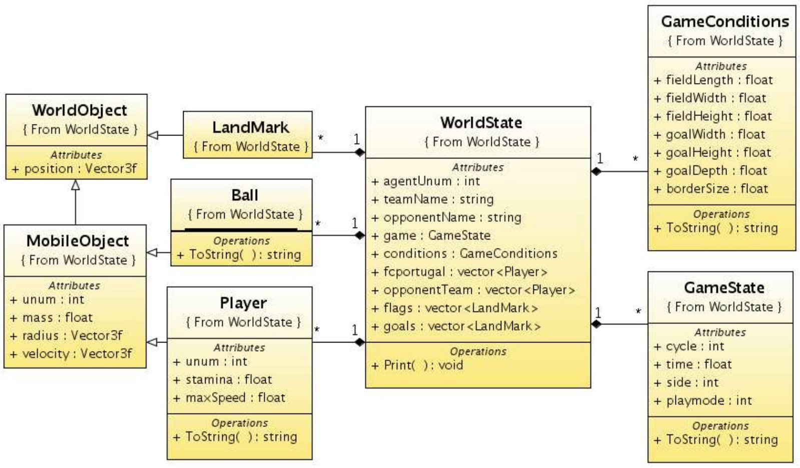

Fig. 2. WorldState Architecture

4 Low-level joint control

A very important issue when designing a humanoid robot is to improve low level joint controllers to achieve smooth joint movements. Our approach allows us to control not only joint positions, but also the initial and final angular velocity of each joint movement. In this way, based on the idea of Virtual Werder 3D Robocup 2007 Team [8], we developed a joint trajectory plan that is an interpolation between the previous and the desired target angle. For each move command a period of time, δ is provided, and a cyclic function is projected between 0 and δ:

$$\Theta(t_i) = A + sin(t_i/T + \phi) + \alpha, \tag{1}$$

where A represents the amplitude, T the period, ϕ the phase and α the offset of the sine function. All of these parameters are, calculated for each Move command, based on the previous target angle, the desired target angle and the δ parameter. On each simulation time step t_i, a new target angle, Θ is calculated, until the δ time is reached, when the corresponding Move finishes.

A feedback control loop is used to maintain the current obtained joint angles as close as possible to the desired joint angles, tuning the gain parameters accordingly. The feedback control loop used is based on PID[5] controllers. The PID controller model is described on figure 3. The PID control loop allows for improving the joint response time avoiding the overshoot which could be the main cause of robot instability at low-level control.

[5] PID: Proportional-Derivative-Integral is a generic control loop feedback mechanism that attemps to decrese the error between the actual value and the desired value

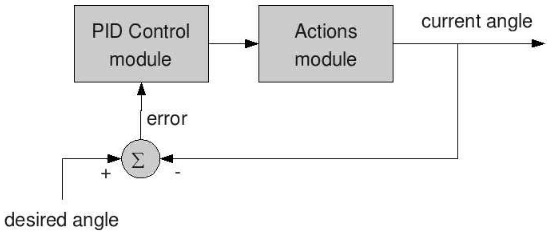

Fig. 3. Feedback control loop

5 Skills description model

The skills of a humanoid agent describe what he can really do in the environment. A soccer humanoid agent should walk, turn, kick the ball, get up from the ground and behave as a goalie when he have to, defending its own goal from the opponent attack. A skill description model was developed to easily implement this skills, being flexible enough to be adaptative to changes on the robot. This description model has the main goal of apply the strategy defined on section 4 and also easily allow for integration of different behaviors, avoiding discontinuities when passing from one behavior to another.

The structure of the skills description model is presented in figure 4.

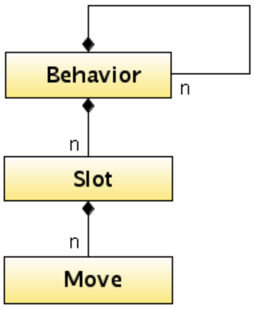

Fig. 4. Skills description model

A Move is the lowest level of the description model and represents what we will do with a joint. It has a parameter *jointId* which represents the joint identifier and a set of another parameters that will move the DOFs[6] of the corresponding joint to a specific position. It is also possible to include the PID parameters (See section 4) inside the Move. A brief description of all parameters are in the table 1.

[6] DOF: Degree of Freedom represents each independent direction about which a joint can perform a movement

Parameter	Description
θ_1	The desired angle for the fist DOF of the joint (valid for hinge and universal joints)
θ_2	The desired angle for the second DOF of the joint (valid for universal joints only)
k_P	PID proportional gain
k_I	PID integral gain
k_D	PID derivative gain

Table 1. Move additional parameters

A Slot can contain several Moves. All Moves inside a slot will be executed in parallel. Each slot has a parameter *delta*, which represents the period of time, δ, described on section 4. Each slot will be divided in a number of steps necessary to interpolate the cyclic function described on section 4 based on the *delta* parameter and on previous and desired target angles of the joints.

A Behavior is the higher level of the description model. Each Behavior has a parameter *name* for identify it. It can contain several Slots or another Behaviors. At least one Slot or one other Behavior.

The model is described using XML (Extensible Markup Language), which is a standard markup language. The advantages of using XML is that it is a standard all over the world, human readable and its inherent tree-like structure makes it be adequate to our description model.

6 Skills optimization

Our main research is actually concerned in improving the actual defined skills using optimization algorithms such as genetic algorithms, tabu search, reinforcement learning and simulated annealing. A solution is currently being developed and it will take into account constraints such as Center of Mass, Center of Pressure and Zero-Moment-Point [9] to keep the humanoid agent stable while performing the skills. The main goal of this approach is to develop fast and stable skills in order to get the best performance of the humanoid.

7 Kinematics

To study actual movements and improve them there are two things that we need: to know where are agent's body parts during the movements and what movement should the agent do to put some body parts in a specific position. The first one is achieved by a "Forward Kinematics" mechanism. The second one, by an "Inverse Kinematics" mechanism.

7.1 Forward Kinematics

As was said, one thing that is useful to improve movements or even during the simulation is to know where are some agent's bodies parts. To achieve that we implement a Forward Kinematics module. If we assume that θ are the joint angular values and χ are the coodinates of a point, the idea is to find T so that:

$$\chi = T\left(\theta\right)$$

We use Denavit-Hartenberg Convention [10] to represent the robotic model of the agent. The complexity of that model is low since all the joints are revolute joints and the model is a kinematic serial chain, that is: all the joints are connected in a serial way.

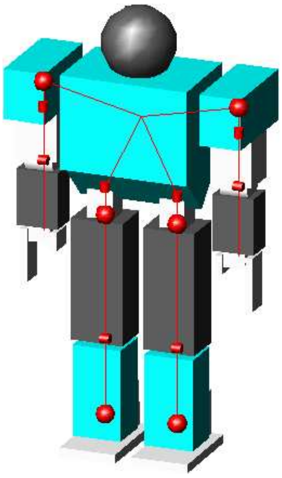

Fig. 5. Kinematic Model of the agent

After the transformation matrices have been determined, it is easy to calculate the position of the points at any time, knowing the angle of the joints at the time. For example: if we want to know the position of the elbow relatively to the vision[7], since the method is systematic we just take 3 transformation matrices:

- "Vision to Shoulder" Transformation matrix
- "Shoulder to Upper Arm Joint" Transformation matrix
- "Upper Arm Joint to Elbow" Transformation matrix

[7] Remember the vision is positioned on Upper Torso

The position of the elbow is determined as follows:

$$\begin{bmatrix}\text{“Vision to}\\ \text{Shoulder”}\\ \text{Transformation}\end{bmatrix} \times \begin{bmatrix}\text{“Shoulder to}\\ \text{Upper Arm Joint”}\\ \text{Transformation}\end{bmatrix} \times \begin{bmatrix}\text{“Upper Arm Joint}\\ \text{to Elbow”}\\ \text{Transformation}\end{bmatrix} \times \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

Then, the absolute position of the elbow will be:

$$\begin{bmatrix}\text{“World to}\\ \text{Vision”}\\ \text{Transformation}\end{bmatrix} \times \begin{bmatrix}\text{“Vision to}\\ \text{Shoulder”}\\ \text{Transformation}\end{bmatrix} \times \\ \times \begin{bmatrix}\text{“Shoulder to}\\ \text{Upper Arm Joint”}\\ \text{Transformation}\end{bmatrix} \times \begin{bmatrix}\text{“Upper Arm Joint}\\ \text{to Elbow”}\\ \text{Transformation}\end{bmatrix} \times \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

7.2 Inverse Kinematics

The basis of a movement is to put a body part in a specific point, then another body part in a different point, and so on...

But how do we put the agent's body parts in that specific position? What are the values of the joints so the agent assume that position? To find the angular values of the joints related to a specific position of a body part we implemented an Inverse Kinematics module.

Basically there are two ways to implement Inverse Kinematics: an algebraic one and a numeric one. We studied both and both needed Forward Kinematics' Transformation Matrices.

The algebraic method is simpler to understand than the numeric one. The problem is that, for some body parts, we have a redundant system: we have several ways to achieve that point. In that cases we need to eliminate some DOF so that we have 3 DOF as long as we're working on a 3D space.

Like in Forward Kinematics, if we assume that θ are the joint angular values and χ are the coodinates of a point, the idea is to find T^{-1} so that:

$$\theta = T^{-1}(\chi)$$

After calculating T of a body part's position (Forward Kinematics), as we're working on a 3D space, T will have three equations. Assuming that $\chi = (x, y, z)$ and $\theta = (\alpha, \beta, \gamma)$:

$$T = \begin{cases} x = T_1(\alpha, \beta, \gamma) \\ y = T_2(\alpha, \beta, \gamma) \\ z = T_3(\alpha, \beta, \gamma) \end{cases}$$

T^{-1} (Inverse Kinematics Transformation Matrix) will also have 3 equations that will be:

$$T^{-1} = \begin{cases} \alpha = T_1^{-1}\left(x, y, z\right) \\ \beta = T_2^{-1}\left(x, y, z\right) \\ \gamma = T_3^{-1}\left(x, y, z\right) \end{cases}$$

So, the Inverse Kinematics algebraic method is not difficult to implement, although it has the referred redundancy problems.

The numeric method does have that redundancy problem as it solves it by itself. We don't need to eliminate DOF as long as this method chooses one of the ways to achieve the desired position and can even be chosen an equation that minimizes some parameters (the distance travelled by the body part, for example).

We still studying and implementing the numeric method.

8 High-Level Decisions and Coordination

Flexible Tactics has always been one of the major assets of FC Portugal teams. FC Portugal 3D is capable of using several different formations and for each formation players may be instantiated with different player types. The management of formations and player types is based on SBSP Situation Based Strategic Positioning algorithm [1,11]. Players abandon their strategic positioning when they enter a critical behavior: Ball Possession or Ball Recovery. This enables the team to move in a quite smooth manner, keeping the field completely covered. The high-level decision uses the infrastructure presented in the section 3. Several new types of actions are currently being considered taking in consideration the new opportunities opened by the 3D environment of the new simulator. We also have adapted our previous researched methodologies to the new 3D environment:

- Strategy for a Competition with a Team with Opposite Goals [1,2,11];
- Concepts of Tactics, Formations and Player Types [1,2,11];
- Distinction between Active and Strategic Situations [1,11];
- Situation Based Strategic Positioning (SBSP) [1,11];
- Dynamic Positioning and Role Exchange (DPRE) [1,11];
- Visual Debugging and Analysis Tools [1,3];
- Optimization based Low-Level Skills [1,3].
- COACH UNILANG A Standard Language to Coach a (Robo)Soccer Team [2,3];
- ADVCOM Intelligent Communication using a Communicated World State [1,3].

With the new humanoid agent, our research is most concerned in developing basic low level skills for the humanoid agent which should be capable to reason about which skill to execute at any time, depending on its situation.

The high-level layers of the team are ready to be used in the new simulator (as they are the same used in our Simulation 2D, small-size, middle-size and rescue teams).

9 Conclusion

In 2006 we proved that almost all of our research on high-level flexbile coordination methodologies was directly applicable to 3D league. The results achieved proved this and the team easily won the European championship and the world championship, scoring 77 goals without conceding any in this last competition. Robust low-level skills are being currently developed for the new humanoid model, using optimization techniques, enabling us to continue the research in strategical reasoning and coordination methodologies that should be the focus of the simulation leagues inside RoboCup.

A new low-level joint control has been showing good results in achieving smooth joint movements since it allows us to control not only the position of the joints, but also its initial and final angular velocity for each movement. The movements developed are now being optimized by using metaheuristics: tabu search and genetic algorithms, aiming at creating very fast and robust humanoid low-level skills.

Acknowledgements

This work was partially supported by the Portuguese National Foundation for Science and Technology FCT/PTDC/EIA/70695/2006 ACORD: Adaptive Coordination of Robotics Teams, (LIACC-FEUP, Univ. Aveiro-IEETA).

References

1. Luís Paulo Reis and Nuno Lau. Fc portugal team description: Robocup 2000 simulation league champion. In *RoboCup 2000: Robot Soccer World Cup IV*, pages 29–40, London, UK, 2001. Springer-Verlag.
2. Luís Paulo Reis and Nuno Lau. Coach unilang - a standard language for coaching a (robo)soccer team. In *RoboCup 2001: Robot Soccer World Cup V*, pages 183–192, London, UK, 2002. Springer-Verlag.
3. Nuno Lau and Luís Paulo Reis. *FC Portugal Homepage, http://www.ieeta.pt/robocup.* March 2008.
4. L. Righetti and Ijspeert A.J. Programmable central pattern generators: an application to biped locomotion control. In *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, 2006.
5. J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato. a framework for learning biped locomotion with dynamic movement primitives. In *ieee-ras/rsj international conference on humanoid robots (humanoids 2004)*. ieee, 2004.
6. Y. Fujimoto and A. Kawamura. Simulation of an autonomous biped walking robot including environmental force interaction. *IEEE Robotics and Automation Magazine*, 5(2):33–42, 1998.
7. Hugo Marques, Nuno Lau, and Luís Paulo Reis. *FC Portugal 3D Simulation Team: Architecture, Low-Level Skills and Team Behaviour Optimized for the New RoboCup 3D Simulator*, volume 14, pages 31–37. Reis, L.P. et al. eds, Proc. Scientific Meeting of the Portuguese Robotics Open 2004, FEUP Editions, Colecção Colectâneas, April 2004.

8. Andreas D. Lattner, Cord Niehaus, Carsten Rachuy, Arne Stahlbock, Ubbo Visser, and Tobias Warden. *Virtual Werder 3D Robocup 2007 Team Description Paper*, 2007.
9. Dirk Wollherr. Design and control aspects of humanoid walking robots. Master's thesis, Technische Universität München, 2005.
10. R. S. Hartenberg and J. Denavit. *Kinematic Synthesis of Linkages.* McGraw-Hill, New York, 1964.
11. Luís Paulo Reis, Nuno Lau, and Eugenio Oliveira. Situation based strategic positioning for coordinating a team of homogeneous agents. In *Balancing Reactivity and Social Deliberation in Multi-Agent Systems, From RoboCup to Real-World Applications (selected papers from the ECAI 2000 Workshop and additional contributions)*, pages 175–197, London, UK, 2001. Springer-Verlag.