Human Robot Interaction: Coaching to Play Soccer via Spoken-Language

Alfredo Weitzenfeld, *Senior Member, IEEE,* Abdel Ejnioui, and Peter Dominey

***Abstract*— In this paper we describe our current work in the development of a human-robot interaction architecture to enable robot coaching by humans on how to play soccer. This approach is analogous to human coaches training soccer players to improve their skills and learn advance game strategies prior to a game while optimizing those strategies during actual games. Our goal is to distinguish between hardwired robot skills and higher level abilities learned from a coach. This is analogous to walking, running and kicking that are basic human skills in contrast to advanced soccer strategies that are learned from a coach. While higher level robot abilities could be acquired by direct software programming, this approach would limit the interaction with human soccer coaches having limited or no software programming experience. To achieve this goal, we exploit recent developments in cognitive science, particularly notions of shared intentions as distributed plans for interaction and collaboration between humans and robots. We define different sets of voice-driven commands for human-robot interaction: (a) action commands requiring robots to perform certain behaviors, (b) interrogation commands, i.e. queries, requiring a response from the robot, and (c) control structure to enable more advanced interaction with the robot including if, if-else, while and specialized training expressions. The human robot interaction architectures is based on the Aldebaran NAO robot platform used in the context of RoboCup soccer standard platform league. This platform interacts with the human coach via CSLU RAD spoken language system. While preliminary work has been previously presented using Sony AIBO, we currently describe more advanced human robot interaction initially developed using the Webots simulated environment before actual experimenting with NAO robots.**

I. Introduction

We expect interaction between humans and robots to be as natural as interaction among humans. To achieve this goal robots need to be capable of high level language processing comparable to humans. For this purpose our current work emphasizes the development of a domain independent language processing system that can be applied to arbitrary domains while having psychological validity based on knowledge from social cognitive science. In particular our architecture exploits: (i) language and meaning correspondence relevant to both neurological and behavioral aspects of human language developed by Dominey et al. [1], and (ii) perception and behavior correspondence based on the notion of shared intentions developed by Tomasello et al. [2, 3]. The particular domain chosen to test our hypotheses is coaching robots to play soccer. While initially robots are taught to kick the ball towards the goal at the first available opportunity, a simple cognitive task for the robot is to decide when to kick and when to pass the ball as shown in Figure 1. While this ability may be directly programmed into the robot, training instead by a human coach requires higher level language processing.

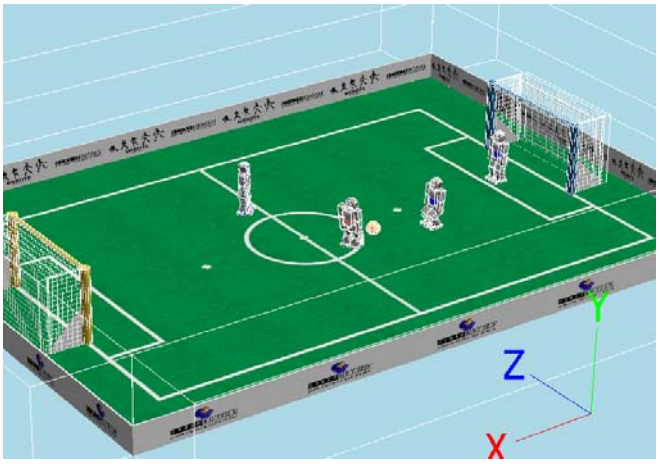

Fig. 1. The image shows a typical game scene where an offensive player controls the ball but is blocked by a defender from the other team. The offensive player needs to decide whereas to kick the ball towards the goal even if blocked or pass it to a teammate.

.

Preliminary work in human robot coaching was described in Weitzenfeld and Dominey [4, 5] where Sony AIBO robots learned individual "go" and "shoot" skills corresponding to searching for the ball and then kicking towards the goal in the context of RoboCup [6], a well documented and standardized robot environment that provides a quantitative domain for evaluation of success. In the Standard Platform League (SPL) two teams of fully autonomous robots play soccer on a 4m x 6m carpeted soccer field using Aldebaran NAO robots. NAO robots use two color-based cameras as primary sensor and include wireless communication capabilities to interact with a game controller and other robots in the field. The field includes two colored goals, yellow and cyan, in addition to lines used for robot localization and for human refereeing. The ball is of orange ball color with robots having different colored "shirts", blue and red. As with human soccer, players need to outperform the opponents by moving faster,

Alfredo Weitzenfeld is with the Division of Information Technology at the University of South Florida Polytechnic, Lakeland, FL, 33180, USA, e-mail: aweitzenfeld@poly.usf.edu.

Abdel Ejnioui is with the Division of Information Technology at the University of South Florida Polytechnic, Lakeland, FL, 33180, USA, e-mail: aejnioui@poly.usf.edu

Peter Dominey is with the INSERM U846 Stem Cell and Brain Research Institut, Robot Cognition Laboratory, 69675 Bron, France, email: peter.dominey@inserm.fr

processing external information more efficiently, localizing and kicking the ball more precisely, in addition to having more advanced individual and team behaviors. In general, approaches to robot programming vary from direct programming to advanced learning approaches. Weitzenfeld's Eagle Knights team has regularly competed in the prior four-legged league [7] and now in the two-legged league [8].

While no human intervention is allowed during a game, in the future humans could play a decisive role analogous to real soccer coaches adjusting in real-time their team playing characteristics according to the state of the game, individual or group performance, or the playing style of the opponent. Furthermore, a software-based coach may become incorporated into the robot program analogous to the RoboCup simulated coaching league where coaching agents can learn during a game and then advice virtual soccer agents how to optimize their behavior accordingly (see [9, 10]). Our human-robot interaction approach is intended to enable human coaches to train robots to play soccer individually and in groups.

In the rest of the paper we describe the human robot interaction architecture (Section II), the robot commands developed for human interaction (Section III), spoken language architecture providing an interface between human and robot commands (Section IV), robot training example describing the pass or shoot coaching by a human (Section V), and conclusions and discussion (Section VI).

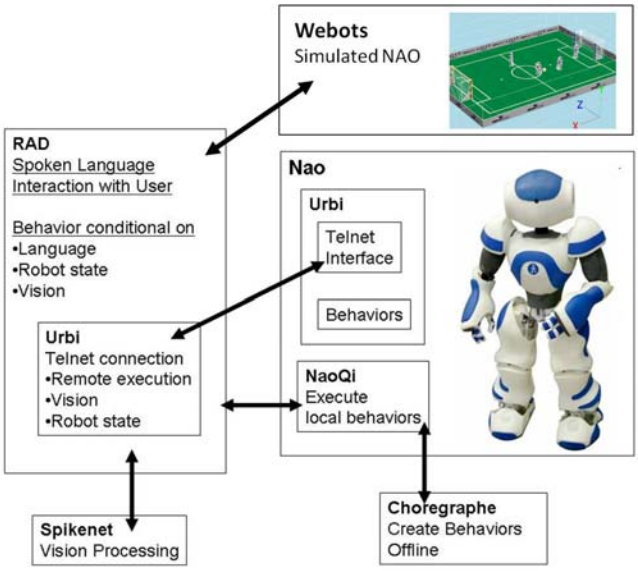

Fig. 2. Human robot interaction architecture.

II. Human Robot Interaction Architecture

The human-robot interaction architecture, shown in Figure 2, consists of the Rapid Application Development (RAD) CSLU Speech Tools system [11] connected via Urbi or NaoQi to Aldebaran NAO robot or alternatively to the Webots simulated environment. Additional components integrated to the architecture include Spikenet for advanced vision processing and Choregraphe used to design basic arm and leg motions. The human coach interacts with RAD through voice commands to control the behavior of the NAO. These voice commands are translated into regular text commands and then transmitted by the external computer system to remotely control the behavior of the NAO robot.

III. Robot Commands

Robots are programmed with a basic set of soccer playing behaviors that continuously process external environmental information, primarily vision, on order to decide on the next action. Additionally, robots need to consider the state of the game provide by a referee box common to all robots. Since robots are programmed to perform their behaviors autonomously, it is necessary to develop voice language command to access basic behaviors. We distinguish among action commands, interrogation commands or queries and control expressions giving the language more structure by including, e.g. if-else and do-while statements.

A. Action Commands

Action commands take the form described in Table 1. The user requests certain action command via the RAD interface that immediately requests the robot to perform the corresponding behavior.

Table 1. General form for action commands and robot behavior.

User	**Robot**
Action Command	Behavior

Table 2 describes action commands and the corresponding behaviors in the robot. Note that certain actions such as *Go to Ball* depend on perceptions, in this case seeing the ball.

Table 2. Action commands and corresponding robot behavior.

Action Commands	**Behavior**
Stop	Stop moving
Walk	Walk forward
Kick	Kick the ball forward
Block	Block the ball
Go to Ball	Go to ball and stop in front of it
Hold	Keep the ball near the robot
Turn Left	Turn left
Turn Right	Turn right
Turn Left Hold	Turn left while holding the ball
Turn Right Ball	Turn right while holding the ball
Orient To Goal	Orient towards the goal
Shoot	Shoot ball towards goal
Pass Left	Left with ball and kick the ball
Pass Right	Right with ball and kick the ball

B. Interrogation Commands

Interrogation commands or queries take the form described in Table 3. The user requests certain query via the RAD interface that immediately requests the robot to reply with an appropriate response.

Table 3. General form for interrogation commands and robot response.

User	**Robot**
Query	Response

Table 4 describes action commands and the corresponding behaviors in the robot. Note that certain actions such as *Go to*

Ball depend on perceptions, in this case seeing the ball.

Table 4. Interrogation commands and corresponding robot behavior.

Queries	Description	Response
Ball?	Does the robot see the ball?	yes = 1, no = 0
Ball near?	Is the robot near the ball?	yes = 1, no = 0
Blue goal?	Does the robot see the blue (cyan) goal?	yes = 1, no = 0
Yellow goal?	Does the robot see the yellow goal?	yes = 1, no = 0
Blue goal near?	Is the robot near the blue (cyan) goal?	yes = 1, no = 0
Yellow goal near?	Is the robot near the yellow goal?	yes = 1, no = 0
Blocked to blue goal?	Are you blocked from the blue (cyan) goal?	yes = 1, no = 0
Blocked to yellow goal?	Is the robot blocked from the yellow goal?	yes = 1, no = 0

C. Control Expressions

While initial version of our human robot interaction system were based on action and interrogation commands, we have been incorporating basic control structures to the spoken language to enable more sophisticated user interaction.

In Table 5 and Table 6 we describe the basic If and If-Else command structures correspondingly.

Table 5. If-Else control expressions.

User	Robot
If Query	response
Then Action Command	behavior

Table 6. If-Else control expressions.

User	Robot
If Query	response
Then Action Command 1	behavior 1
Else Action Command 2	behavior 2

In Table 7 we describe the basic Do-While command structures.

Table 7. Do-While control expressions.

User	Robot
While Query	response
Do Action Command	behavior

In Table 8 we describe the basic train command structure. The train command is important since the train sequence will be recorded by the system and stored under the specified command name. Later on the trained sequence can be recalled in a similar way to other commands.

Table 8. Train control expressions.

User	Robot
Train Command Name	
Training Sequence	behavior-response sequences
End Train	

IV. Spoken Language Architecture

Having human users control and interrogate robots through spoken language results in the ability to naturally teach robots individual action sequences conditional on perceptual values or even more sophisticated shared intention tasks involving multiple robots such as passing the ball between robots when one of them is blocked or far away from the goal.

In terms of language processing, Dominey and Boucher [12, 13] have developed a system that can adaptively acquire a limited grammar by training with human narrated video events. An image processing algorithm extracts the meaning of the narrated events translating them into action descriptors, detecting physical contacts between objects, and then using the temporal profile of contact sequences in order to categorize the events (see [14]). The visual scene processing system is similar to related event extraction systems that rely on the characterization of complex physical events (e.g. give, take, stack) in terms of composition of physical primitives such as contact (e.g. [15, 16]). The visual scene processing system was able to perform: (a) scene processing for event recognition, (b) sentence generation from scene description and response to questions, (c) speech recognition for posing questions, (d) speech synthesis for responding, and (e) sending and receiving textual communications with the robot. We have incorporated some of these capabilities into the current system to provide more natural language interaction between coach and robot.

A. Language Mappings

In terms of language mapping, each narrated event generates a well formed <*sentence*, *meaning*> pair that is used as input to a model that learns the sentence-to-meaning mappings as a form of template where nouns and verbs can be replaced by new arguments in order to generate the corresponding new meanings. Each grammatical construction corresponds to a mapping from sentence to meaning. This information is also used to perform the inverse transformation from meaning to sentence. These templates or grammatical constructions (see [17]) are identified by the configuration of grammatical markers or function words within the sentences [18]. The construction set provides sufficient linguistic flexibility. For example, in Table 9, the sentence translates into a set of two robot action commands as described in the previous section.

Table 9. Sentence-meaning mapping example.

Sentence	Meaning
Kick ball towards goal	Orient to goal, kick

Additionally, new <*percept, response*> constructions can be acquired into the language by binding together perceptual and behavioral capabilities. Three components are involved in <*percept, response*> constructions: (i) the percept, either a verbal command or a sensory system state, e.g. external visual information; (ii) the response to this percept, either a verbal response or a motor response from the existing behavioral repertoire; and (iii) the binding together of the <*percept, response*> construction and its subsequent validation that it was correctly learned. The system then links and saves the <*percept, response*> pair so that it can be used in the future. This is achieved by using the "train" control command previously described storing a sequence of behavior-response sequence. An example of such constructions is shown in Table 10.

Table 10. Percept-response mapping.

Percept	Response
Ball	Kick

B. *Spoken Language Processing*

Spoken language processing is done via CSLU-RAD. The system defines a directed graph where each node in the graph links voice commands to specific behaviors and queries sent to the robot as shown in Figure 3. The "select" node separates action and interrogation commands. Action commands are represented by the "behaviors" node while interrogation commands are represented by the "questions" node. Behavior nodes include 'Stop', `Walk', 'Kick', 'Go->ball', 'Hold, 'TurnL', 'TurnR', 'TLH', and TRH; while question nodes are 'Ball?' ('Do you see the ball?'), 'BallN? ('Is the ball near?'), 'BGoal' ('Do you see the blue goal?') and 'YGoal´ ('Do you see the yellow goal?'). Behavior commands are processed by the "exec" node while questions are processed by the question mark "*?*" node that waits for a 'Yes' or 'No' response from the robot. Finally, the "Return" node goes back to the top of the hierarchy corresponding to the "select" node. The "goodbye" node exits the system.

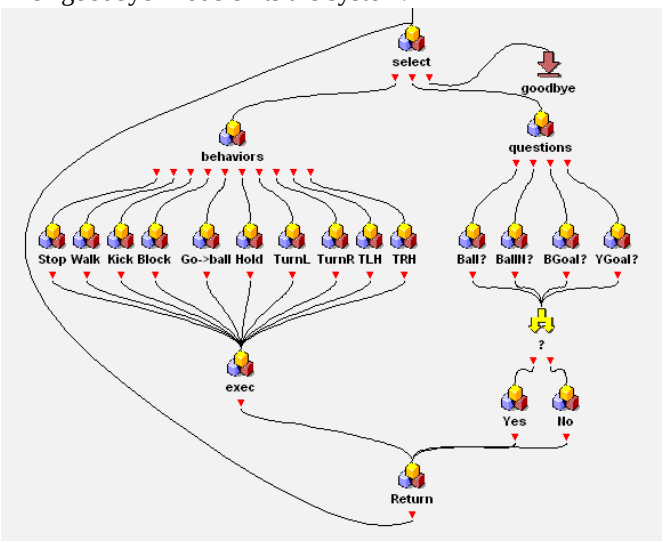

Fig. 3. The CLSU-RAD diagram describes the basic set of behaviors and questions that can be sent as voice commands to the robot.

Action and interrogation commands form the basis for teaching new behaviors in the system. In particular, we are interested in teaching soccer-related tasks at two levels: (i) basic behaviors linking interrogations to actions such as "if you see the ball then go to the ball" ("Go"), or "if the ball is near then kick the ball" ("Shoot"); and (ii) hierarchical behaviors composed of previously learnt behaviors such as "Go and Shoot". To achieve such learning, we have extended the CSLU-RAD interface previously shown in Figure 3 to enable creation of new behavior sequences as shown in Figure 4. The main difference with the previous diagram is that after the "questions" node the model saves the response and continues directly to the "behaviors" node where actions are taken and the sequence stored as part of the teaching process. Additionally, all sequences learnt are included as new behaviors in the system, e.g. "GO", "SHOOT", and "GO&SHOOT" nodes. As shown with this example, a teaching conversation is represented by a sequence of action and interrogation commands: (i) '*GO*' telling the robot to go towards the ball; and (ii) '*SHOOT*' telling the robot to kick the ball towards the goal.

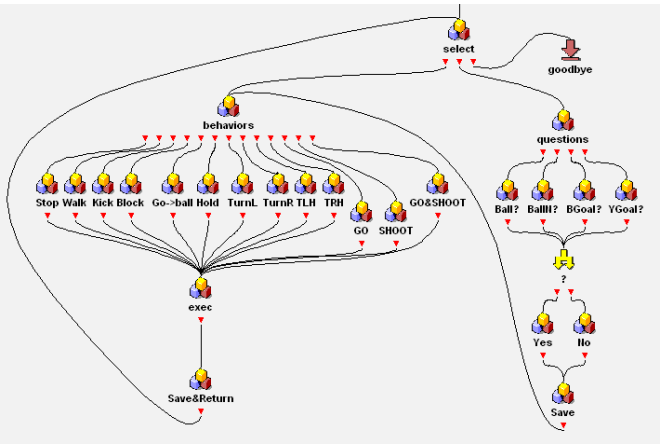

Fig. 4. The CLSU-RAD diagram describes the extended set of commands for training the robot to Go and Shoot the ball.

In [5] we describe this training sequence in more detail. The greatest benefit from this hierarchical training is that previously learned skills can be accessed through compact expressions as opposed to the full set of training sequences. From the robot perspective both basic and hierarchical forms perform comparably.

V. Multiple Robot Training Example

In [5] we have described a very basic set of individual training of robot behaviors using the CSLU-RAD spoken language interface. In this section we describe our current work in training multiple robots to perform more advanced soccer strategies. Possibly the most basic decision in soccer is whether to pass the ball or to shoot it towards the goal. This simple decision making can make players and thus teams much more effective than their opponents. We have thus developed a "Ball Pass" strategy involving two attacking robots, a left forward and a right forward, in addition to a defender and goalie in the opposing team. The two forward robots have two corresponding strategies analyzing whether to shoot or pass the ball the companion player:

- "Ball Pass Right" applied to the left offensive player.
- "Ball Pass Left" applied to the right offensive player.

In the next section we describe how we train the two offensive players to decide whether to pass or shoot the ball towards the goal.

A. *Ball Pass Strategy*

The Ball Pass strategy requires the individual robots to: (a) go to the ball, (b) orient towards the goal, and (c) when ready to shoot decide if to actually shoot or pass the ball to its accompanying offensive player. To initialize the strategy

both offensive players must be correctly positioned in the field as shown in Figure 1. Additionally the two players must be able to perform correct passes and most important, they must be able to recognize when they are blocked by a defender when trying to shoot towards the goal. Thus, the actual passing or shooting behavior is decided depending on whether the robot can perceive an opening for shooting towards the goal. In real soccer there is also the possibility to dribble the ball away from the defender, something we are not considering in our strategy. The state diagram for the "Ball Pass" strategy is shown in Figure 5.

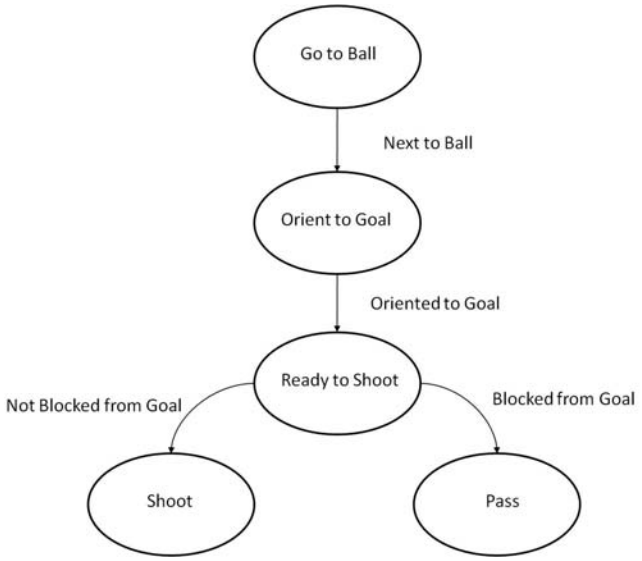

Fig. 5. State diagram for the "Ball Pass" strategy.

B. Ball Pass Training Sequence

The "Ball Pass" training sequence is shown in Table 11. The left column corresponds to the left offensive player while the right column corresponds to the right offensive player. Note how both sequences are initiated by the "train" command.

Table 11. Left offensive (left column) and right offensive (right column) player training sequences.

Left Offensive 'Ball Pass Right' Training Sequence	**Right Offensive 'Ball Pass Left' Training Sequence**
RAD: Select option	RAD: Select option
User: Train Ball Pass Right	User: Train Ball Pass Left
RAD: Select option	RAD: Select option
User: Go to Ball	User: Go to Ball
RAD: Select option	RAD: Select option
User: Orient to Goal	User: Orient to Goal
RAD: Select option	RAD: Select option
User: If blocked from blue goal	User: If blocked from blue goal
User: Then Shoot	User: Then Shoot
User: Else Pass Right	User: Else Pass Left
RAD: Select option	RAD: Select option
User: End Train	User: End Train
RAD: Select Option	RAD: Select option
User: Goodbye	User: Goodbye

The actual execution of the "Ball Pass" strategy is shown in Table 12. Again, the left column corresponds to the left offensive player while the right column corresponds to the right offensive player.

Table 12. Left offensive (left column) and right offensive (right column) players execution commands.

Left Attacker 'Ball Pass Right' Command	**Right Attacker 'Ball Pass Left' Command**
RAD: Select option	RAD: Select option
User: Ball Pass Right	User: Ball Pass Left
RAD: Select Option	RAD: Select option
User: Goodbye	User: Goodbye

Figures 6-9 show snapshots of the "Ball Pass" strategy.

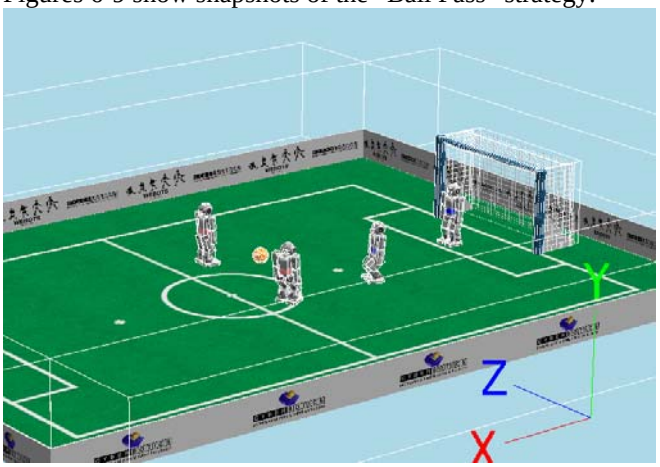

Fig. 6. Right offensive player passes ball to left offensive player.

Fig. 7. Since left offensive player is now blocked by the defender it passes the ball back to the right offensive player instead of shooting towards goal.

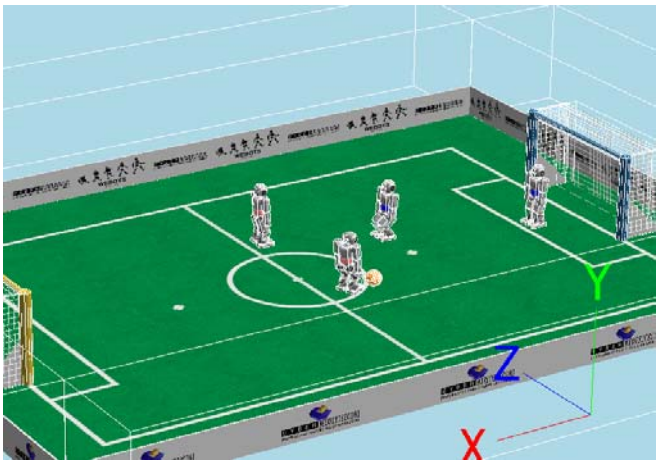

Fig. 8. Right offensive player is now open to shoot the ball towards the blue (cyan) goal.

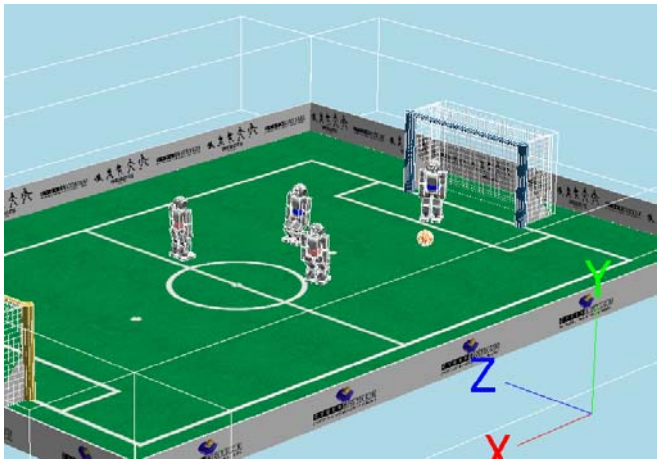

Fig. 9. Right offensive player shoots the ball towards the blue (cyan) goal.

VI. Conclusions and Discussion

We have described in this paper our current research in the development of a generalized approach to human-machine interaction via spoken language in the context of robot soccer that may be extended to other domains. The coaching architecture described in the paper exploits recent developments in cognitive science - particularly notions of grammatical constructions as form-meaning mappings in language, and notions of shared intentions as distributed plans for interaction and collaboration binding perceptions to actions. With respect to social cognition, shared intentions represent distributed plans in which two or more collaborators have a common representation of an action plan in which each plays specific roles with specific responsibilities with the aim of achieving some common goal. In the current study, the common goals were well defined in advance (e.g. teaching the robots new relations or new behaviors). As such, the shared intentions could be built into the dialog management system. Training sequences were developed in the context of RoboCup soccer standard platform league where we have been competing for many years both in the four-legged and two-legged leagues. We used the CSLU-RAD environment for spoken voice human interaction with the robots.

As technical sequences become more complex it is important to be able to teach robots them using a more natural interaction between humans and robots. In particular the dialog pathways are somewhat constrained, with several levels of hierarchical structure in which the user has to navigate the control structure with several single word commands in order to teach the robot a new relation, and then to demonstrate the knowledge, rather than being able to do these operations in more natural single sentences. In order to address this issue, we are reorganizing the dialog management where context changes are made in a single step.

To demonstrate the interaction model we described how to coach a robot to play soccer by teaching new behaviors at two levels: (i) individual basic behaviors trained from a sequence of existing actions and interrogations, and (ii) hierarchical multi-robot strategies trained from newly trained sequences. In prior work [5] we describe individual basic training such as 'GO' and 'SHOOT' tasks. In this paper we extend the training to hierarchical multi-robot strategies to "Ball Pass" where the robot needs to decide whether to shoot or pass the ball. This task is being initially developed using Webots simulation environment to be finally tested using Aldebaran NAO robots hopefully under real game constraints.

Finally, our long term goal in human-robot coaching is to be able to positively affect team performance during a real game similarly to human soccer coaches.

References

[1] Dominey PF, Hoen M, Lelekov T and Blanc JM, Neurological basis of language in sequential cognition: Evidence from simulation, aphasia and ERP studies, Brain and Language, 86(2):207-25, 2003.

[2] Tomasello M, Constructing a language: A usage-based theory of language acquisition. Harvard University Press, Cambridge, 2003.

[3] Tomasello M, Carpenter M, Call J, Behne T, Moll H, Understanding and sharing intentions: The origins of cultural cognition, Behavioral and Brain Sciences, 2006.

[4] Weitzenfeld, A., and Dominey, P., 2007, Cognitive Robotics: Command, Interrogation and Teaching in Robot Coaching, RoboCup 2006: Robot Soccer World Cup X, G. Lakemeyer et al. (Eds.), LNCS 4434, pp. 379–386, Springer-Verlag.

[5] Weitzenfeld, A., Ramos, C, and Dominey, P., 2009, Coaching Robots to Play Soccer via Spoken-Language, RoboCup Symposium, RoboCup 2008: Robot Soccer World Cup XII, L. Iocchi et al. (Eds.), LNCS/LNAI 5399, pp. 379–390, Springer-Verlag.

[6] Kitano H, Asada M, Kuniyoshi Y, Noda I, and Osawa, E.,. Robocup: The robot world cup initiative. In Proceedings of IJCAI-95 Workshop on Entertainment and AI/ALife, 1995.

[7] Weitzenfeld, A., Martínez, A., Muciño, B., Serrano, G., Ramos, C., and Rivera C., 2007, EagleKnights 2007: Four-Legged League, Team Description Paper, ITAM, Mexico.

[8] Ramos, C., and Rivera C., Rios, G., Herrera, E., Morales, M., and Weitzenfeld, A., 2009, EagleKnights 2009: Two-Legged Standard Platform League, Team Description Paper, ITAM, Mexico.

[9] Riley P, Veloso M, and Kaminka G, An empirical study of coaching. In: Distributed Autonomous Robotic Systems 6, Spring-Verlag, 2002.

[10] Kaminka G, Fidanboylu M, Veloso M, Learning the Sequential Coordinated Behavior of Teams from Observations. In: RoboCup-2002 Symposium, Fukuoka, Japan, June, 2002.

[11] CSLU Speech Tools Rapid application Development (RAD), http://cslu.cse.ogi.edu/toolkit/index.html

[12] Dominey PF and Boucher JD, Developmental stages of perception and language acquisition in a perceptually grounded robot, Cognitive Systems Research, Volume 6, Issue 3, Pages 243-259, September 2005.

[13] Dominey PF and Boucher JD, Learning to talk about events from narrated video in a construction grammar framework, AI, Vol 167, No 1-2, pp 31-61, Sept 2005.

[14] Kotovsky L and Baillargeon R, The development of calibration-based reasoning about collision events in young infants, Cognition, 67, 311-351, 1998.

[15] Siskind JM, Grounding the lexical semantics of verbs in visual perception using force dynamics and event logic. Journal of AI Research (15) 31-90, 2001.

[16] Steels L and Baillie JC. Shared Grounding of Event Descriptions by Autonomous Robots. Robotics and Autonomous Systems, 43(2-3):163—173, 2002.

[17] Goldberg A, Constructions. U Chicago Press, Chicago and London, 1995.

[18] Bates E, McNew S, MacWhinney B, Devescovi A, and Smith S, Functional constraints on sentence processing: A cross linguistic study, Cognition (11): 245-299, 1982.