UI-AI3D 2008 Team Descripton Paper

Saeid Akhavan, Hamid Reza Hasani, Kamal Jamshidi,
Arefeh Kazemi, Hoda Safaeipour, Mohammad Hoseyn Sheykholeslam,
Hedayat Vatankhah, Mohammad Reza Yazdchi, Zahra Zojaji, and Zainab
Zolaktaf
http://eng.ui.ac.ir/uisim

Computer Engineering Department, University of Isfahan,
81745 Isfahan, Iran
ui_3dsoccer@eng.ui.ac.ir

Abstract. Due to the critical role of midlevel actions such as walk, kick, etc in the decision making processes of a soccer agent, our main efforts have been concentrated on the design and implementation of an adequate midlevel decision and action management layer .In this paper we proposed a five layered hierarchical structure for the implementation of midlevel actions. Higher levels concentrate on determining the agents appropriate action and movement path using the proposed Most Similar State Recognizer (MSSR), while the lower ones are responsible for the elicitation of transitions through the use of Adaptive Neural Fuzzy Inference System (ANFIS).

1 Introduction

The soccer 3D agent is in its initial stages of development. In addition to the relatively high level processes that were required in the previous server, new actions such as walking, standing up, running, etc must be modeled in terms of body joints movement. As a consequence, the design of a new structure for the agent which takes advantage of all available facilities, especially ones concerning its physical motions, is unavoidable. Therefore designing a highly qualified midlevel structure is essential. We have employed the idea of dividing each agent action into states and transitions [1], and introduced a flexible midlevel structure which eradicates the inefficiencies of the model. The rest of this paper is organized as follows. Section 2 gives a brief review of the midlevel decision model. Section 3 describes the function of the State Selector, section 4 describes the Most Similar State Recognizer and section 5 explains the ANFIS controller system.

2 Midlevel Decision Model

Due to the complex nature of the agent and its environment, the following four layered hierarchy is defined as the agents midlevel decision model. Assuming that a high level decision such as pass or dribble is made, the midlevel action selector

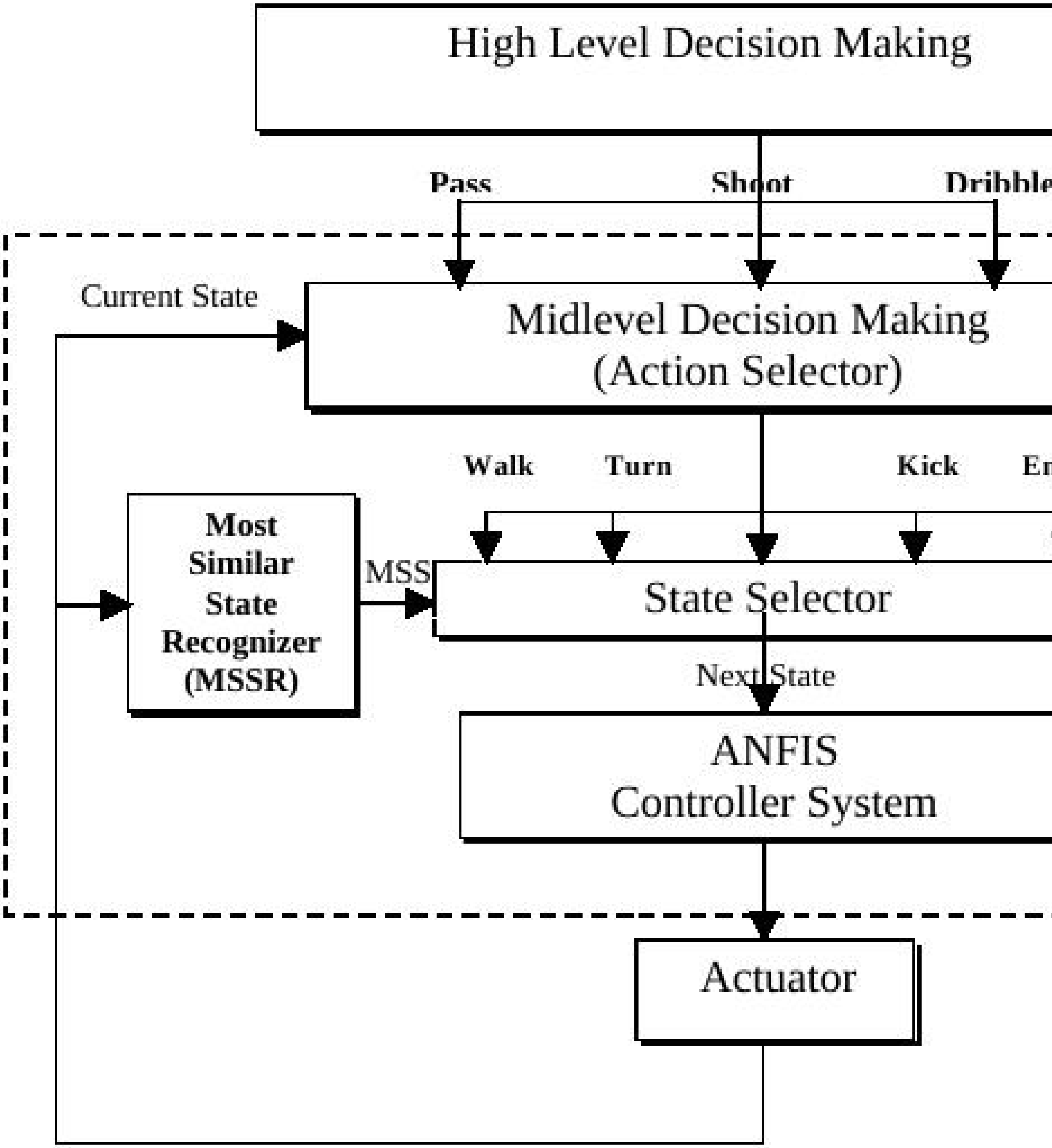

Fig. 1. agent midlevel decision model hierarchy

selects an appropriate action to be performed. This selection is based upon the previous layers output and the sensorial information. For example if the decision is to pass the ball, the midlevel action selector selects the kick action.
The next layer in the hierarchy, the state selector layer, is responsible for the design and elicitation of the actions. Because each action requires an intricate and subtle set of sub actions to be performed, and also because of the complex and uncertain nature of the environment, it is important to model the action using an approximate and general description of the parameters involved; in our model such a description was obtained by dividing each action in to a set of sub actions, steps, and the required transitions.
In this layer, a state graph using the method deployed in [1] is implemented. This graph provides us with a general description of states and possible direct transitions between them. However, two design issues arise. One is the method of identifying the agents next state in order to perform the action. The other is the precise explanation of the transition required to move to the next state.
To work out the first issue, the MSSR algorithm, described below, is used to identify the agents current state with respect to the graph. As a consequence the next step in the agents movement path is recognized.

The second issue is resolved through the use of ANFIS networks. Using the agents current state and its desired next state as inputs, the Anfis controller selector determines the Anfis structure which yields the transition. By the end of this process the generated command is sent to the server[1]

3 State Selector

The MSSR-most similar state recognizer- provides this unit with the MSS-most similar state- and its corresponding similarity degree. If this degree is less than a certain threshold it demonstrates that the agents current state similarity degree is not enough and it should try to reach an acceptable MSS in order to proceed. Otherwise, the agents current state similarity degree is acceptable and its next state can be identified according to its MSS and the overall action that has to be performed. For example, consider the two states of the walk action, S1 is the standing state and S2 is the right leg forward state. A and B represent the agents status in two different situations; suppose that the output of MSSR is listed in Figure 4 ,State Selector chooses S2 for A and S1 for B as the next state.

4 Most Similar State Recognizer (MSSR)

This unit provides the state selector with the MSS and a corresponding similarity degree, according to aset of predefined states and the also the agents current state. The definition of the similarity is based on transitions instead of states because of the usage of MSS in the subsequent units. To be precise, the MSS is used by the state selector to identify best next step in the agents movement path, therefore, when calculating the MSS, it is essential to look ahead and consider the transitions that would be made possible if the given state was selected.

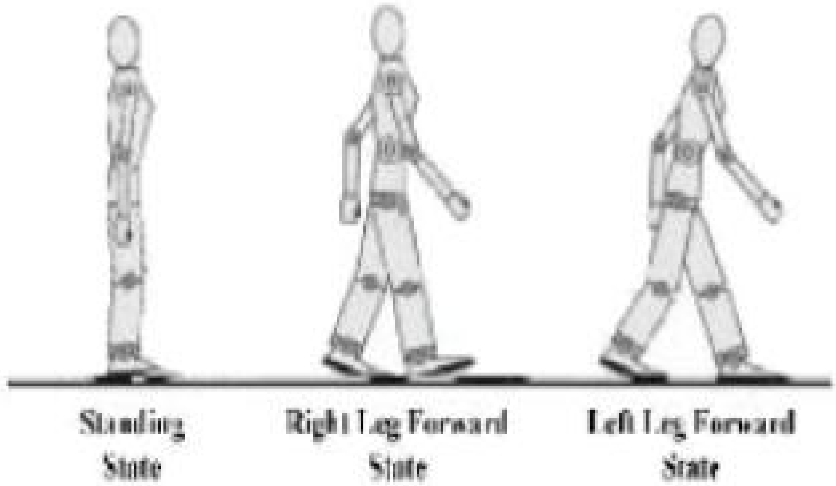

Fig. 2. walk states

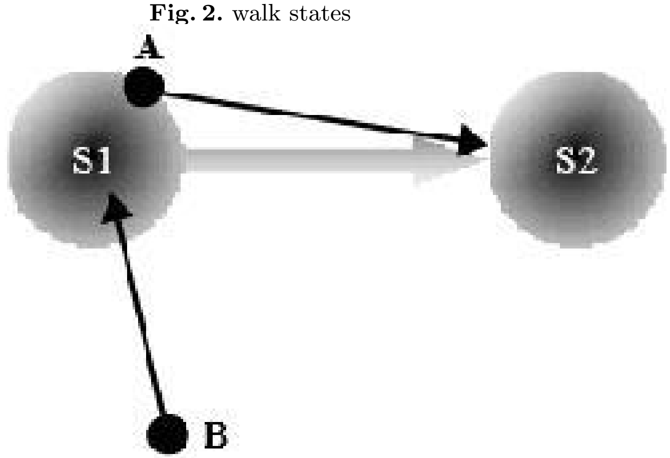

Fig. 3. State Selector selects S2 for A and S1 for B as the next state

Agent status	MSS	Similarity degree
A	S1	0.7
B	S1	0.2

Fig. 4. Sample MSSR output

Based on this hypothesis, we have designed a forward looking method to identify this kind of similarity. Consider we have a crisp current state of the agent, state C, and need to recognize the MSS in the predefined walk states. For each walk state W1 that reaches W2 by transition T, C is similar to W1 if its transition to W2 1is similar to T. To approach this task pragmatically, a similarity measurement mechanism is required for the transitions. Since each transition is defined by a certain number of variables (i.e. joint angular velocity etc), it can be mapped to a vector in n-dimensional space. The similarity in vector space model is determined by using associative coefficients based on the inner product of two transitions. The inner product is normalized to prevent the effect of length separately.

5 ANFIS controller system

After the identification of the next state to which the agent has to move, using the MSSR algorithm mentioned earlier, the next phase is calculating the movement that makes this transformation possible.

As mentioned previously, the imprecision and ambiguity of the environment make crisp action modeling impossible; it is important to define the states and the transitions between two adjacent states in an approximate and general manner. One possible solution is the use of a fuzzy inference method, employing fuzzy if-then rules, which enables approximate reasoning and deals with imprecision and information granularity. However an effective method for tuning the membership functions (MFs) so as to minimize the output error measure or maximize performance index, is required.

As a result, the fundamental concept and method used in our approach is based on ANFIS- Adaptive Neuro Fuzzy Inference System- which provides both approximate reasoning and the capability of learning and adaptation [2].

A separate ANFIS controller is used for every two states defined in the movement graph in the previous layer. The agents current state and its desired next state are given as inputs to the ANFIS controller selector, the output of the selector is the controller which subsequently yields the transition according to the agents current state.

5.1 Training ANFIS

ANFIS can learn fuzzy rules from input-output data pairs, incorporate prior knowledge of fuzzy rules, fine tune the membership functions, and act as a self-learning fuzzy controller by automatically generating the fuzzy rules needed [4].

For the ANFIS controller to be able to predict the transition first it has to be trained with the suitable input and output data. The training data of the ANFIS structures consists of the initial state information as input and the successful transition needed to move to the next state as output. In order to reduce the complexity of the problem, only six of the agents joints have been considered so far. Therefore the input consists of the agents current state information i.e. its knees and hips angles in two axis, while the output consists of the joints angles and velocities in the next desired state.

6 Future Works

Our future work is based on fixing the short comings of the model and obtaining a complete implementation of the structure discussed earlier. At present, initial steps of different parts of the agents midlevel decision model have been covered independently. The integration phase is yet to be fulfilled;The state selector unit still has a lot of potential for improvement;in addition an applicable high level action selection structure must be designed and implemented.

References

1. Akhavan S, Babaeizadeh M, Hasani H, Kazemi A, Safaeipour H, Sheykholeslam M, Vatankhah H, Zojaji Z, Zolaktaf Z: UI-AI3D 2007 Team Description
2. J.-S.R. Jang, ANFIS: Adaptive-network- based fuzzy inference systems, IEEE Trans. Syst., Man, and Cybem., vol. 23, pp. 665-685
3. Katic D, Vukobratovic M: Control Algorithm for HumanoidWalking Based on Fuzzy Reinforcement Learning.
4. Changjiu Z, Kanniah J: Adaptive Network Based Fuzzy Control of a Dynamic Biped Walking Robot.
5. Parseghian A: Control of a Simulated, Three-Dimensional Biped Walking, Continue Walking, Rock Side-to-Side, and Balance.
6. Stone P: Layered Learning in Multi-agent Systems.