Pardis: A Fuzzy Extension to the 3D Soccer Simulation Server

Saeed Bagheri[1], Reza Hesamifard[1], Vahid M. Hashemi[1], Sara Ramezani[1], Amir Avani[1], Jafar Habibi[1]

[1]Sharif University of Technology
Tehran, Iran, {bagheri, hesami, habibi}@ce.sharif.edu,
{sramezani, vhashemi, aavani}@mehr.sharif.edu.

Abstract: The Pardis 3D Soccer Server Development Team has aimed at using a fuzzy approach for extending the current server to incorporate fuzzy communication with the agents. Fuzzy communication is closer to natural reasoning, so it can be used more conveniently to model human intelligence. We have designed a fuzzy environment where agents can define their own fuzzy sets for different concepts, thus they can acquire skills by modifying their fuzzy sets, and manipulating skills becomes much easier. This server can be beneficial both for agents using fuzzy logic, and for other agents.

1 Introduction

RoboCup has been launched in 1994 as a platform and standard problem for testing, applying, and comparing various research methods in artificial intelligence. The soccer game has been proposed as a unifying framework for these activities. The various different properties of soccer (e.g. communication in a multi-agent system, strategy planning, real-time inference, etc.) shows how carefully chosen this framework is.

As a guideline for leading the research society, a long-term goal has been set for RoboCup, that is: “By the year 2050, develop a team of fully autonomous humanoid robots that can win against the human world soccer champion team”. RoboCup has been divided into various branches (leagues) so that different branches of research could proceed in parallel. The simulation league is one of these branches that have been introduced for concentrating on implementing ideas in artificial intelligence that are independent of the physical robot design.

In the past 10 years, not only the teams, but also the simulation servers have undergone many changes and improvements. These changes are usually caused by two different reasons; one is better knowledge of the environment that is gradually acquired. The second and more important reason is that many of the simplifying assumptions that were first considered for making the environment more convenient are no longer needed due to improvements of methods and technology, so they can be eliminated to make the simulation environment resemble a real environment more closely. A good example of such an improvement is the “3D Simulation League”. Since the first year of the RoboCup competitions, 2D simulation was practiced, until last year that the “3D Soccer Server” was introduced. This new server is a great improvement in bringing the simulation closer to real world simulation.

As an improvement to the current “3D Simulation Server”, we propose using fuzzy tools to extend the current server, so that it may simulate the real world more closely. Our idea arises from the observation that the sensory information and control commands of human agents are expressed in qualitative form (linguistic labels), and not in quantitative form (numbers). Before discussing the features of the Pardis environment, we shall briefly review the basic concepts of fuzzy logic.

2 Fuzzy Logic

The notion of fuzzy sets was first introduced by Zadeh in 1965 [1]. In this section, brief general introduction to the concepts of fuzzy sets and fuzzy logic will be presented.

2.1 Basic Definitions

Let X be the universal set, a set A in classical set theory can be defined by a function $F_A(x)$ from X onto {0,1} such that $F_A(x) = 1$ if and only if x is a member of A. Such a function is known as the *characteristic function* of A. In fuzzy set theory, the range of the characteristic function is the interval [0, 1], so that $F_A(x)$ shows the degree of membership of x in A.

Each fuzzy set represents a *linguistic label* such as cold, warm, far, short, large, etc. These labels do not have a precise interpretation in a natural language; the fuzzy set representing them reflects this impreciseness or fuzziness by assigning membership degrees to its members. Some of the basic notions regarding fuzzy sets are defined below.

Two fuzzy sets A and B are *equal* ($A = B$) if for each x in X, $F_A(x) = F_B(x)$.

The complement of a fuzzy set A is A^c defined by $F_{A^c}(x) = 1 - F_A\ (x)$ for each x in X.

For two fuzzy sets A and B , A is a subset of B ($A \subseteq B$) if for each x in X, $F_A(x) \leq F_B(x)$.

The operators normally used for *union* and *intersection* of fuzzy sets are:

The *union* of two fuzzy sets A and B is a fuzzy set C such that $F_c(x) = \max(F_A(x), F_b(x))$ for each x in X.

The *intersection* of two fuzzy sets A and B is a fuzzy set C such that $F_c(x) = \min(F_A(x), F_b(x))$ for each x in X.

2.2 Computing with words

Fuzzy agents are usually used in environments that do not have a fuzzy structure, so two layers are needed, one for converting the crisp inputs from the environment (e.g. the outputs from a sensor) into fuzzy data for the system (fuzzy sets), and another for converting the fuzzy output of the system back to crisp data or applying in the environment (fig1); these stages are called *fuzzification* and *defuzzification* stages respectively[2].

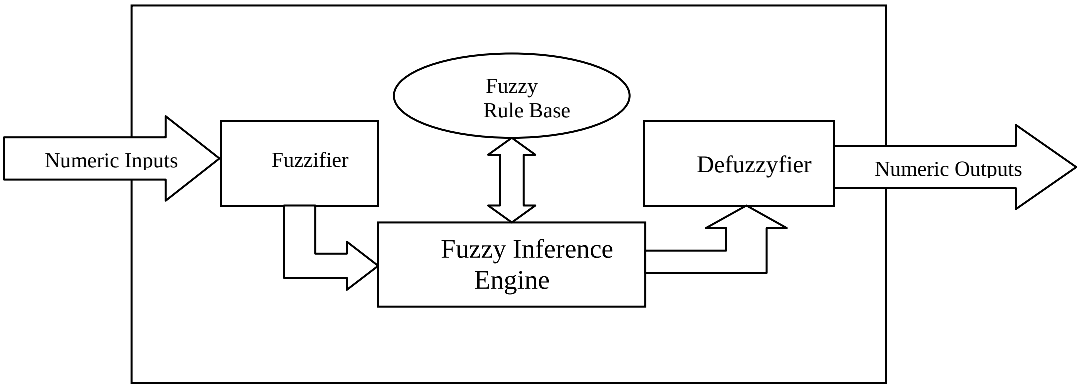


Fig. 1: The structure of a fuzzy system.

So the expertise of a fuzzy system is basically incorporated into its rule base and inference system, because the fuzzification and defuzzification stages only make the system compatible with the environment. In other words, if we had an environment in which the agents' perceptions could be naturally fuzzy and the agent could execute fuzzy actions, then there would be no need for the fuzzification and defuzzification stages, and the agent could be modelled as in Fig. 2.

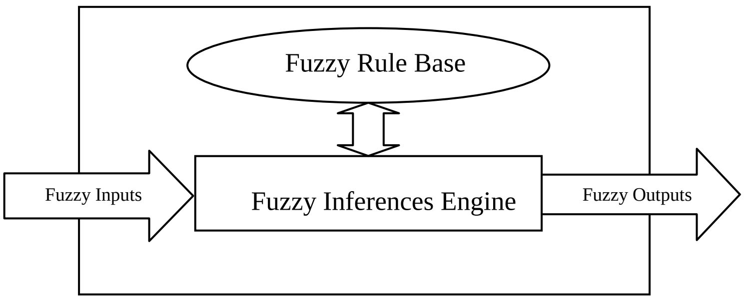


Fig. 2: A fuzzy agent in a fuzzy environment

The inference in such a system is an inference with words, and the knowledge is represented and stored as *qualitive data*, and not quantities. Since the early 1990's, Zadeh has introduced a new concept that he calls *"Computing with Words"* (CW) [3]. The main idea behind this concept, which he believes is the next step in the natural evolution of fuzzy systems, is that natural languages essentially incorporate most of human intelligence. Humans store and transmit information and even reason and compute using natural languages. So if we could capture the power of natural languages, we may come closer to simulating human intelligence. The inputs and outputs of such a system shall be words, which are imprecise in nature; the inference is also performed using words.

CW can be chosen as a computational core for designing a fuzzy agent. In such an agent, all computation and inference could be done using words, and each word corresponds with a fuzzy set. The principles of CW are described in [3]. There are still many obstacles in implementing CW, and the non-existence of a real fuzzy environment is part of them.

Finally, fuzzy agents try to achieve simpler and more efficient models by acknowledging the natural ambiguousness of real systems. These systems are usually complex systems for which no precise model is known.

3 Description of the Pardis Environment

In the Pardis environment, the current 3D Simulation Server [5] is used as a core, so we have designed a new layer that can be added to the current server. This layer enables fuzzy communication of agents with the server (see fig.3). Each agent can inform the server of what fuzzy sets it shall use for each of the designated notions, e.g. the agent can say that by a “gentle kick”, I mean this specific fuzzy set, etc.

By adding this fuzzy layer, the previous functions of the server are preserved, and each agent can notify the server at initialization whether it wants to use the fuzzy layer or not. The agents using the fuzzy tools should communicate with the server via a special communication protocol that shall be introduced. In such an environment, fuzzy agents will only consist of a fuzzy rule base and inference system (Fig 2).

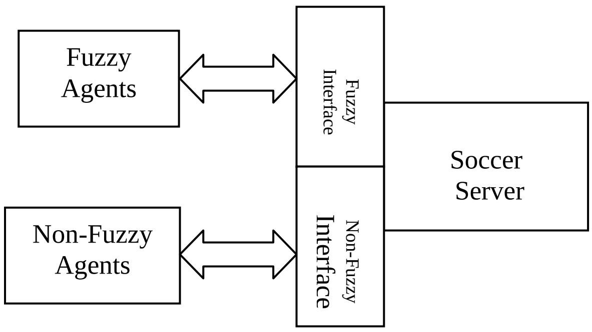


Fig 3: The Pardis 3Dserver

The internal structure of the fuzzy interface is shown in Fig4. The agent profiler component maintains the details of the fuzzy sets that the agent sends to the server. The fuzzifier and defuzzifier use this information for interpreting the data between the server and each agent.

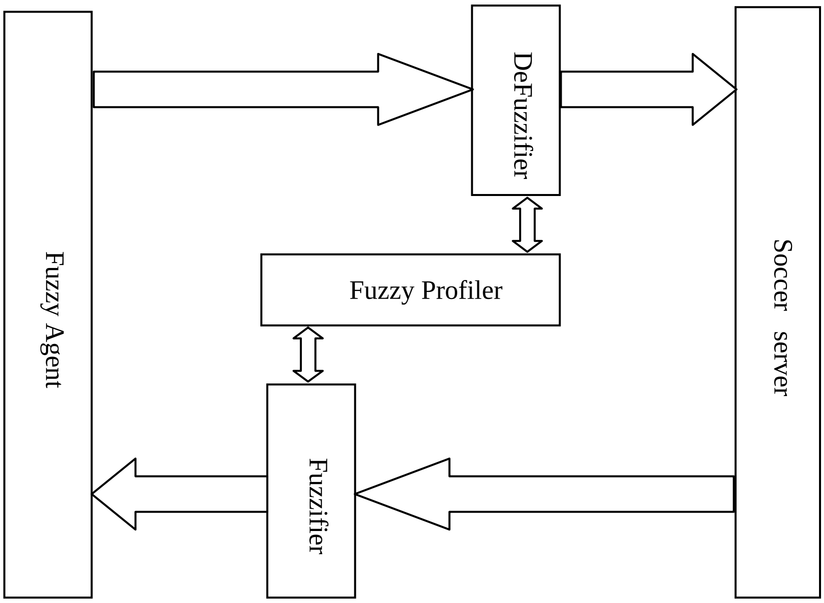


Fig 4 A more detailed view of the Fuzzy Interface

3.1 Fuzzifying and Defuzzifying Process

Fuzzification and defuzzification are the most important problems that a fuzzy simulation system is encountered with. Because it's usually not possible to implement the environment and agents on fuzzy hardware, normal (crisp) hardware is used. Still, we wish to communicate with the agents with information that is fuzzy enough to encourage them to use fuzzy information processing. Some possible methods for fuzzification are described below:

1- **Fuzzy labels only:** For a notion in the environment, say x, by comparison among related fuzzy sets, a linguistic label A is chosen and sent to the agent such that A(x) is maximized. In this method, although linguistic labels are used to communicate information, the ambiguousness of natural language is eliminated. In other words, in usual communication in a real world system (e.g. human agents in the real environment), the information acquired is more than just fixing an interval (membership in a fuzzy set); it also involves the degree of membership in the designated set.

2- **Fuzzy labels and membership degrees:** A second method that can be considered is to communicate the degree of membership of x in A, as well as the linguistic label A. A possible expression of this type may be "x is 0.8 degree A". Expressions of this sort are less problematic than the previous. Also, either only the set with maximum membership for x can be announced as before, or all the sets which x is a member of can be announced to the agent along with their corresponding degrees. This method solves the problem of too much ambiguity that occurred in the first method. The transmitted data contains more information, and is also still fuzzy. But since the agents know the exact fuzzy sets corresponding to different notions, the data can easily be transformed to exact crisp information. Consequently, the agents may tend to process the numerical membership degrees, instead of fuzzy information processing. Considering our aim in designing a system with completely fuzzy data transmition, this method will not suffice; thus we have resorted to a third method which is more appropriate and has been used in the Pardis environment.

3- **Fuzzy labels and linguistic hedges:** In this method, information is communicated as labels of fuzzy sets, along with hedges that approximately express the membership degree of x in A. examples of this kind of data are “x is very A”, or “x is slightly A”. Again, either only the set with maximum membership for x can be announced, or all the sets which the degree of membership of x in them is higher than a specific threshold.

Defuzzification can also be performed by different methods corresponding to the above fuzzification methods. Since we use the third method for fuzzification, the corresponding method shall be used for defuzzification.

Each agent indicates the number of hedges it wants to use for each notion (e.g. speed of player, angle of ball, etc.) when connecting to the server. In the fuzzification stage, the server finds the best hedge and sends it to the agent. Among the different sets that x is a member of, only two of them that have the highest membership degree for x are chosen, and the information corresponding to them (linguistic label and hedge) is sent to the agent.

Agents can define three different types of membership functions: triangular, trapezoidal, and Gaussian. Each of these functions is identified with the data specified in Table1.

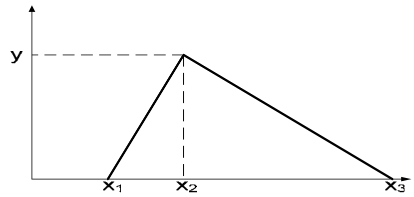
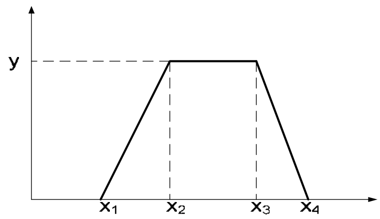
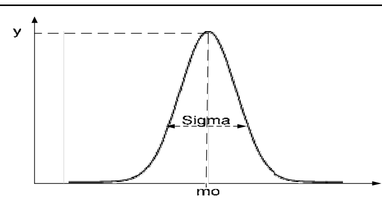
Membership function	**Parameters**	**Shape**
Triangular	**(x_1, x_2, x_3 , y)**	
Trapezoidal	**(x_1, x_2, x_3, x_4, y)**	
Gaussian	**(mo, sigma, y)**	

Table 1 : Valid membership function types in the Pardis simulation system

Agents can express their control commands with respect to the number of hedges they have specified. Hedges indicate the degree of membership in the desired set. This level of preciseness suffices for a natural (granular) approach [4].

4 Properties of the Pardis Environment

In a fuzzy football environment, all perceptions of an agent are fuzzy sets. For example the notion of distance can be expressed by a number of fuzzy sets each of them representing a notion (e.g. close, far, etc.). It should be emphasized here that a notion may have different interpretations regarding different concepts, for example, 30 Km/h is very fast for a player to move, but it's not a very high speed for the ball.

The actions (control commands) of the agent are also expressed by fuzzy notions. An example of the sensory information of an agent may be "a team-mate of yours with number 3 is far away from you and a bit to the left"; a control command of an agent may be "kick the ball very hard and a small amount upward". Thus the environment will allow the agent to compute using only fuzzy notions.

Enabling an agent to define its own fuzzy sets for different notions, gives it the ability to define the environment as it perceives it. Thus as the agent's expertise gradually increases (by a learning method), it can define new fuzzy sets according to its new perceptions. Consequently this system incorporates the skills into the agents, by defining specific fuzzy sets for each agent, thus skills will become an important feature, complementing team strategy. Such a tool can be very beneficial not only for fuzzy agents, but also for agents that do not use a fuzzy inference system.

5 Conclusions

Some of the basic benefits of the Pardis environment are:

- Agents that worked with the previous server will still be supported, but also tools for the easy implementation of (completely) fuzzy agents will be added.
- The capability of fuzzy and crisp agents playing against each other gives rise to new comparisons and evaluations of the two types of systems.
- As it is observed that with the increase of expertise in natural systems, the control of the agents becomes more precise; by enabling an agent to define its own fuzzy sets, a new and more naturally skilled class of RoboCup agents will emerge.
- Human experts can transfer their knowledge more easily to agents, because fuzzy systems have closer resemblance to the human mind.
- The fuzzy system can be enhanced using linguistic hedges, such as "very", "somewhat", "fairly", to implement various effects on the fuzzy sets (both of the sensory information and the control commands.)

6 References

[1] L.A. Zadeh, "Fuzzy Sets", Information and control, 8(3), pp 338-353.
[2] G.J. Klir and Bo Yuan, "Fuzzy Sets and Fuzzy Logic: Theory and Applications", Prentice Hall, Upper Saddle River, NJ, 1995.

[3] L.A. Zadeh, "Fuzzy logic = computing with words", IEEE Trans. on Fuzzy Systems 4 (1996) 103-111.

[4] Kreinovich, V.; Nguyen, H.T.; "1st order, 2nd order, what next? We do not really need third-order descriptions: a view from a realistic (granular) viewpoint", IFSA World Congress and 20th NAFIPS International Conference, Volume 4, 25-28 July 2001 Page(s):1908 - 1913 vol.4

[5] Marco K. and Oliver Obst, 'Simulation League: The Next Generation', Springer, Vol. 3020, 2004,page(s) 458-469.