

Nexus2D Team Description Paper

Mahdi Abolfazli¹, Masoud Jamili, Mostafa Sayahi,

Saeid Askari, Mehrnaz Ghashgaei, Hadi Kalamati, Sirius Shahini

Nexus Simulating and Modeling Lab, Ferdowsi University of Mashhad, Iran

Abolfazli71@gmail.com¹

Abstract. Dynamic environment of the server caused our research to focus on efficient methods in this environment. Our research resulted in the implementation of a method based on nonlinear and regulated calculation for ball movement and interception model to implement the player's positioning and moving, which are explained in this paper. Noisy environment of the server encouraged us to develop a fuzzy mechanism for passing skill. This mechanism is also explained in the following.

1. Introduction

2D soccer simulation is one of the Robocup leagues which intended to eliminate areas concerning construction of a robot and its difficulties and instead to focus on cooperation between agents and other high-level issues. This paper is based on 2D soccer simulation server which is a physical simulation system. This server is a multi-agent system and has a dynamic environment [1]. Using simple and static methods is inefficient in a dynamic environment. Therefore, it is a good idea to use dynamic algorithms and methods in such an environment. In following, these algorithms and methods are explained.

We have started our research in 2009, and we had participated in national competitions as *Robotoos*, *051*, *Tempux*, and also *Nexus* teams. At this time, all the members feel that our team is ready to participate in Robocup 2013 competitions. Our team is based on Agent-2D base code.

2. Ball interception

One of the important high-level skills is intercepting the ball while moving. Noting that the server is a dynamic environment, which means every object can move, makes the problem more complex [1]. To solve the problem, we need to formulate players and ball movement in the environment.

2.1. Ball movement model

The distance that the ball traverses is calculated from the below formula [1]:

$$s(t) = \sum_{i=1}^t v_0 \mu^{i-1} = v_0 \frac{1 - \mu^t}{1 - \mu} \quad (1)$$

Having the distance that the ball traverses and its initial location and direction, final ball's location can be calculated:

$$P(t) = \vec{a} + s(t) \vec{b} \quad (2)$$

In formula 2 \vec{a} is the initial location of the ball and \vec{b} is its initial direction ($|\vec{b}| = v_0$).

2.2. Player movement model

Assuming that edp is a constant and initial velocity of the player is v_0 , the below recursive formula holds [1]:

$$v(t) = \begin{cases} v_0, & t = 0 \\ v(t) = v(t-1) \cdot \mu + edp, & t > 0 \end{cases} \quad (3)$$

Solving this recursive equation, we get:

$$v(t) = v_0 \mu^t + \mu edp \frac{\mu^{t-1} - 1}{\mu - 1} + edp \quad (4)$$

Having formula 4, the distance that the player traverses can be calculated:

$$s(t) = \sum_{i=1}^t v(t-1) = v_0 \frac{\mu^t - 1}{\mu - 1} + \frac{\mu \cdot edp}{\mu - 1} \left(\mu \frac{\mu^{t-2} - 1}{\mu - 1} - t + 2 \right) + (t-1) edp \quad (5)$$

2.3. Calculating the best location to intercept the ball [2]

Assuming that the player has reached to its maximum velocity (v_1) and can move in any direction, and in the below formula:

$$P(t) = \vec{a} + s(t) \vec{b} \quad (6)$$

\vec{a} is the ball location relative to the player and \vec{b} is the ball initial velocity, we can come to conclusion that the player reaches to the ball when: $|P(t)| \leq v_1 t$. We used Newton method to solve this equation. Consider $f(t)$ as:

$$f(t) = |P(t)| - v_1 t \quad (7)$$

We have:

$$f'(t) = |P(t)|' - v_1 \quad (8)$$

To calculate $|P(t)|'$ first we need $P'(t)$:

$$P'(t) = s'(t) \cdot \vec{b} = v_0 \frac{\mu^t \ln \mu}{\mu - 1} \cdot \vec{b} \quad (9)$$

We can simply calculate $|P(t)|'$ by Inner Product of two vectors $P(t)$ and $P'(t)$ and dividing the result by the length of the vector $P(t)$:

$$|P(t)|' = \frac{P'(t) \cdot P(t)}{|P(t)|} \quad (10)$$

Now, Initial guess: [3]

$$t_n = \begin{cases} 0, & n = 0 \\ t_{n-1} - \frac{f(t_{n-1})}{f'(t_{n-1})}, & n > 0 \text{ and } f'(t_{n-1}) < 0 \\ t_{n-1} + 10, & \text{otherwise} \end{cases} \quad (11)$$

We did our computations with the assumption that the player has reached to its final velocity. To have more accuracy, the time that the player needs to reach its final velocity can be considered. This amount of time can be determined by finding the root of this equation:

$$f(t) = v(t) - \text{maxspeed} \quad (12)$$

Finally, ball interception position can be determined by putting the obtained time in formula 2.

3. Passing

Passing is a skill that its success is dependent on both passer and receiver agent. Both of them should have a clear understanding of the environment. Passer must guarantee that the ball will be received to the teammate without any threat. On the other hand, receiver must be aware of that the player who must receive the ball is itself, and it must have the ability to receive it. Since there is noise in the environment, our goal is not setting the ball in a specific point. Our main goal is to move the ball in a direction which the receiver has the potential to receive it in the best position and with the less probability of ball interception by the opponents [1].

3.1. Choosing the best pass mechanism

For finding the best pass, we have considered a score for all teammates. If the kickable agent passes the ball to a teammate, the corresponding score shows the pass success probability. The teammate with the highest score will be chosen for passing the ball to it.

3.2. Fuzzy system

There is noise along with all actions and perceptions of the agents in the simulation environment. On the other hand, fuzzy systems are not sensitive to noise. Therefore, fuzzy systems are a good choice as a decision making system in soccer simulation environment.

The main structure of a fuzzy system is illustrated in Figure 1. Some sets enter the fuzzy system as input sets, and some exit as output sets. In this section, we will study these sets [4].

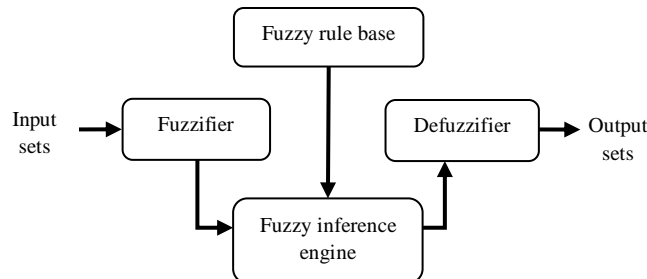


Figure 1- The main structure of the fuzzy systems

3.2.1. Input and output sets of the fuzzy system

One of the inputs of the system is the probability that opponents can interception the ball before it gets to its target. This is one of the important factors, and has a great effect on the performance of our system. If the ball is intercepted by opponent agents before reaching to its target, the pass is completely unsuccessful. Computing this probability is an important issue at this point. Figure 2 shows an algorithm to compute this probability. The algorithm determines the probability by which the ball can be intercepted while moving along its path. In each simulation cycle, opponent movement radius (OMR) is increased by the amount of maximum player speed. If the circle with OMR radius and with opponent agent center includes the ball, this algorithm will return a nonzero value.

```
1 MinVel = 0.1
2 MaxPlayerSpeed = 0.25 // meter per cycle
3 while BallVel > MinVel do
4   if BallVel > 1.5 then
5     BallDecay = 0.78
6   else if BallVel > 1 then
7     BallDecay = 0.80
8   else if BallVel > 0.85 then
9     BallDecay = 0.85
10  else BallDecay = 0.88
11  BallPos = BallPos + BallVel
12  BallVel = BallVel * BallDecay
13  for each opponent player do
14    OMR = OMR + MaxPlayerSpeed
15    if OpponentDistToBall < OMR then
16      return (OpponentDistToBall-OMR)/20
17  end for
18 end while
19 target = BallPos
20 if OpponentDistToTarget - OMR < 4 then
21   return OpponentCanInterceptBall
//Opponent gets the ball near the target
```

Figure 2- The algorithm for computing ball interception probability

Another input parameter is the distance between the target and opponent goal. This parameter has an inverse relation with the target score- the less this distance, the more score assigned to that target. In this case, the ball is passed to a teammate which is closer to the opponent goal. Therefore, the chance of obtaining a goal will be increased. The distance between the passer and target is considered as another input parameter of the system. The more this distance, the probability of reaching the ball to the target is decreased. It has reverse relation the pass success or in other words, the more this distance, the less pass success. The last input is the number of opponent agents around the target. The more number of opponents causes the pass success to decrease. Table 1 shows a list of inputs and output of the system.

Table 1- Inputs and output of the fuzzy system

Parameter	Input / Output
Ball Interception Probability	Input
Distance Between Target and Opponent Goal	Input
Distance Between Passer and Target	Input
Number of Opponents Around Target	Input
Pass Success	Output

4.2.2. Knowledge base

Fuzzy systems are based on knowledge and rules. The heart of a fuzzy system is its knowledge base which consists of a set of if-then rules. An if-then rule is a simple if-then statement that some of its words are described by membership functions [4].

The starting point of making a fuzzy system is finding a set of if-then rules from the knowledge of expert people or science of that special field. The next step is to combine these rules in a unique system. Different fuzzy systems use different methods and principles for combining these rules. Figure 3 shows the knowledge base of the system.

1. **If** OpponentCanIntercept is high **then** PassSuccess is low
2. **If** OpponentCanIntercept is medium **then** PassSuccess is medium
3. **If** OpponentCanIntercept is low **then** PassSuccess is high
4. **If** DistanceToGoal is far away **then** PassSuccess is low
5. **If** DistanceToGoal is far **then** PassSuccess is medium
6. **If** DistanceToGoal is near **then** PassSuccess is high
7. **If** DistanceToTarget is far away **then** PassSuccess is low
8. **If** DistanceToTarget is far **then** PassSuccess is medium
9. **If** DistanceToTarget is near **then** PassSuccess is high
10. **If** OpponentsNumber is many **then** PassSuccess is low
11. **If** OpponentsNumber is medium **then** PassSuccess is medium
12. **If** OpponentsNumber is few **then** PassSuccess is high

Figure 3- Our fuzzy knowledge base

In Figure 3, the words high, medium, low, and ... are described by the membership functions of Figure 4, 5, and 6. Membership functions of *DistanceToGoal* and *DistanceToTarget* are the same, because both of them are distance.

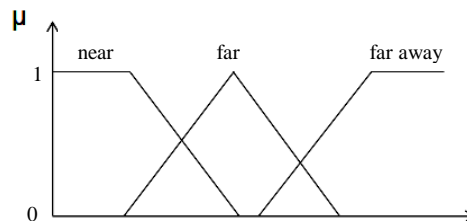


Figure 4- Membership function of *DistanceToGoal* and *DistanceToTarget*

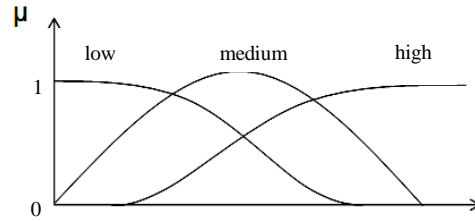


Figure 5- Membership function of *OpponentCanIntercept*

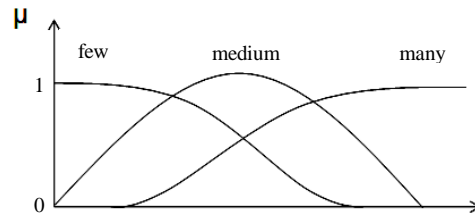


Figure 6- Membership function of *OpponentsNumber*

4. Genetic algorithm for blocking

Genetic algorithm (GA) is a variant of local search. We have used this algorithm for blocking an opponent agent. Our goal is to find the best eight coefficients which are used in blocking procedure. GA is a useful algorithm for optimization problems, and our objective is to find eight optimized coefficients. Therefore, it sounds a good choice to use GA in this case. [5]

In our implementation, each gene consists of eight positive integers (coefficients). To find the fitness of each gene, we put the ball, an opponent who owns the ball, and a teammate in eight different situations. Then by modeling the blocking procedure, we compute the time the teammate needs to intercept the ball for each situation. The sum of the eight different time computed by modeling is considered as the fitness of the gene.

First, we start with a hundred randomly generated genes. To produce the next generation, fifty best genes are chosen from the previous generation. Then, fifty genes are produced by mating pairs of the fifty best. Thirty of them go into the next generation directly, but we make mutation in twenty remaining and form the next twenty genes for next generation. Figure 7 illustrates the production of the next generation.

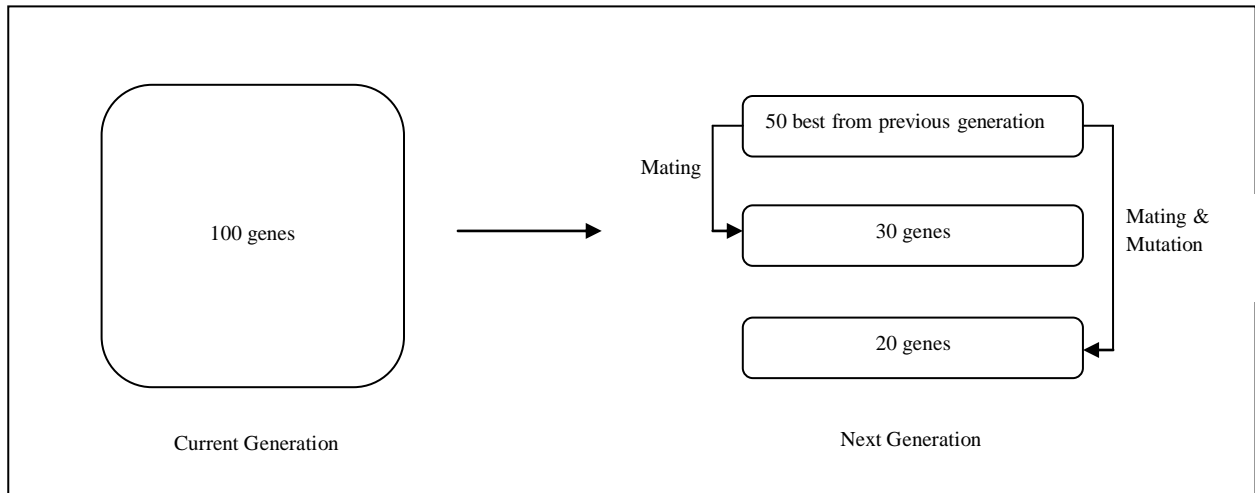


Figure 7- The production of next generation

5. References

1. Mao Chen, Klaus Dorer, Ehsan Foroughi, Fredrik Heintz, ZhanXiang Huang, Spiros Kapetanakis, Kostas Kostiadis, Johan Kummeneje, Jan Murray, Itsuki Noda, Oliver Obst, Pat Riley, Timo Steffens, Yi Wang, Xiang Yin, *Robocup Soccer Server: User Manual*, 2003
2. Frieder Stolzenburg, OliverObst, Jan Murry, *Qualitative Velocity and ball Interception*, 2002
3. Karen A. Kopecky, *Root-Finding*, 2007.
4. Li-Xin Wang, *A Course In Fuzzy Systems and Control*, New Jersey: Prentice Hall, 1996.
5. Stuart J. Russell, Peter Norving, *Artificial Intelligence: A Modern Approach*, 2nd edition, Practice Hall.