Proceedings of the 34th Chinese Control Conference
July 28-30, 2015, Hangzhou, China

Application of Hidden Markov Model to Locate Soccer Robots

QIU Ji[1], ZHU Quanmin[1, 2] ZHAO Dongya[2]

1. Department of Engineering Design and Mathematics, University of the West of England, Frenchay Campus, Coldharbour Lane, Bristol, BS16 1QY, UK
E-mail: Ji2.Qiu@live.uwe.ac.uk quan.zhu@uwe.ac.uk

2. College of Chemical Engineering, China University of Petroleum, Qingdao, China, 266580
E-mail: dyzhao@upc.edu.cn

Abstract: This paper adopts a Hidden Markov Model as a basis for predicting the probabilities in location of soccer robot's trajectories, develops the corresponding algorithms, and then demonstrates the simplicity of the procedure with simulations. The purpose of the initial presentation is to establish a proper platform for the future comprehensive studies of using Hidden Markov Models to assemble critical observations with uncertainties or random measurement errors in stochastic system modelling and control.

Key Words: Hidden Markov Model, Markov Chains, stochastic process, trajectories.

1 Introduction

Due to substitution of human power and intelligence is becoming a great demand from all over the world, robots have come into being the implemented class creatures. Briefly to have intelligence, robots should be installed with proper computational algorithms that can be read as 'thought' as human. Predicting trajectory which the robot might follow has been a classical, and highly widely demanded, issue to study in many realistic situations such soccer robot match, diving robots, mine searching robots and so on. This paper focuses on locating soccer robots with probabilities in moving directions, based on basic Hidden Markov Model theory, and then establishes a procedure in forecasting the robot locations in practical circumstance. Even though there have been amount of algorithms for Markov Model (Ghahramani & Jordan, 1997) based operations, how to tailor the theory of Markov model into condensed pack for an ad hoc application becomes complicated and tedious . To justify the further study on the above challenging problems, a few of research questions are listed below, which subsequently guides the study to provide possible solutions.

Research question one: Which algorithm is suitable for location of soccer robot.

Research question two: How to select and revise a proper parameter estimation algorithm for application in the match reliable design from those existing formulations.

Research question three: How to implement the design through Markov Model to simulate the situation in a match by MATLAB programs.

With such insight, the Hidden Markov Model has been proposed, where some model algorithms (Ghahramani and Jordan, 1997) are suitable for in calculating the probabilities for locating soccer robots. It should be noted that this is the first initial application of Markov Model to locate soccer robot in research, which will summarise/tailor some of the existing algorithms into an ad hoc procedure.

* This work is partially supported by the National Nature Science Foundation of China under Grant 61273188 and Taishan Scholar Funding, Shandong, China.

2 Hidden Markov Model

To establish a basis for the study, the main technical characteristics of Hidden Markov Model are outlined in this chapter.

2.1 Model

A Hidden Markov Model (HMM) is characterised by 5 elements $< S, M, A, B, \pi >$ (Rahman & Zelinsky, 1999), where:

- $S \in R^n$ is a set of state, denoted as $S = \{S_1, S_2, S_3, \dots, S_n\}$, where n is the dimension of the state and is assumed that there exists such a way that any state can be reached from any other state;
- M is a set of distinct observations per state;
- A is a matrix of state transition probability distribution $A = \{a_{ij}\}$ which is the probability of transition from state S_i to state S_j,
- B is a matrix of observation probability distribution at any state j, $B = \{b_j(k)\}$;
- An initial state distribution $\pi = \{\pi_i\}$ which represents the ideal state of the environment.

Applying Hidden Markov Model in locating soccer robot, a discrete time stochastic process $X = \{X(t), t \in T\}$ is required to collect random variables. That means, in the *index set* T, there is random variable $X(t)$ happened in each nonnegative integers t (Dai *et al.*, 1999). If $X_n = i$, the process is said to be in state i at time n. Denote w_{ij} being the connect probability from state i to state j. Then

$$w\{X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \dots, X_0 = i_0\} = w_{ij} \quad (1)$$

where all states $i_0, i_1, \dots, i_{n-1}, i, j$ and all $n \geq 0$. So that $w_{ij} \geq 0$, $\sum_j w_{ij} = 1$. Further let p_i represent the probability in state i. Fig.1 shows such example for the Markov Chain of the state transfer function.

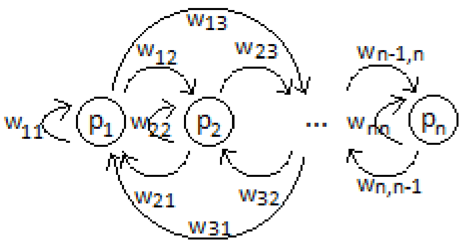

Fig. 1: The Markov Chain

From Fig.1, the dynamic process for the probability in total states can be described as below:

$$\begin{cases} p_1(t+\Delta t) = w_{1,1}p_1(t) + w_{2,1}p_2(t) + \cdots + w_{n,1}p_n(t) \\ p_2(t+\Delta t) = w_{1,2}p_1(t) + w_{2,2}p_2(t) + \cdots + w_{n,2}p_n(t) \\ \cdots \\ p_n(t+\Delta t) = w_{1,n}p_1(t) + w_{2,n}p_2(t) + \cdots + w_{n,n}\, p_n(t) \end{cases} \quad (2)$$

Where $t\,(0, 1, 2\ \ldots)$ is the discrete time index. When $t = 0$, $p_1(0) = 1$, $p_i(0) = 0 (i = 2,3, \ldots, n)$ (Dai *et al.*, 1999). Based on the discrete time Hidden Markov Model, assume that the probability of initial state of the system can be represented by a vector P_s. P_s is always 1 which means the probability of the state where the robot is in at the moment must be established fact. After $\Delta t = 1$ times, the relationship between $P_s$and the expected probability vector $P_s(t+1) = (a_1, a_2, a_3 \ldots a_n)$ could be described as the equation below:

$$P_{trajectory} \in \{P_s(t) | P_s(t+1) = P_s(t)W\}_{max} \quad (3)$$

Where,

$$W = \begin{bmatrix} w_{1,1} & w_{2,1} & \cdots & w_{n,1} \\ w_{1,2} & w_{2,2} & \cdots & w_{n,2} \\ \vdots & \vdots & \vdots & \vdots \\ w_{1,n} & w_{2,n} & \cdots & w_{n,n} \end{bmatrix}, \quad (4)$$

Consequently, formula (3) elaborates the probability of trajectory is the maximum element belong to the set of probabilities of the walkable states after using the iteration method. The iteration process break if the time $t \geq t_{set}$. Where t_{set} is a control parameter which can be operated by designer.

2.2 Model's States

If $w_{ij}^n > 0$, $n > 0$, state j is said to be accessible from state i.

.If j is not accessible from i, then

$$w\{\text{ever enter } j | \text{start in } i\} = w\Big(\bigcup_{n=0}^{\infty} \{X_n = j\} | X_0 = i$$

$$\leq \textstyle\sum_{n=0}^{\infty} w\{X_n = j | X_0 = i\} = 0 \quad (5)$$

Because

$$w_{ii}^0 = w\{X_0 = i | X_0 = i\} = 1 \quad (6)$$

It means that any state is accessible from itself. If state j is accessible from state i and state i is accessible from state j, then state i and state j communicate. Communication between state i and state j is expressed symbolically by $i \leftrightarrow j$ (Ross, 2002).

The communication satisfies 3 properties:

- $i \leftrightarrow i$
- if $i \leftrightarrow j$ then $j \leftrightarrow i$
- if $i \leftrightarrow j$ and $j \leftrightarrow k$ then $i \leftrightarrow k$

The 1st and 2nd properties are derived by definition of communication. To prove 3rd property, suppose that i communicates with j, and j communicates with k. Then there exist integers n and m such that $w_{ij}^n w_{jk}^m > 0$. By the Chapman-Kolmogorov equations (Ross, 2002),

$$w_{ik}^{n+m} = \textstyle\sum_r w_{ir}^n w_{rk}^m \geq w_{ij}^n w_{jk}^m > 0 \quad (7)$$

Hence, state k is accessible from state i.

Two states are said to be in the same *class* when they can communicate. It is a simple consequence of 1st, 2nd and 3rd properties that any two classes of states are either identical or disjoint. In other words, the concept of communication divides the state space up into a number of separate classes. The Markov chain is said to be irreducible if there is only one class, that is, if all states communicate with each other (Ross, 2002).

A Markov process of order two would depend on the two preceding states, a Markov process of order three would depend on the three preceding states, and so on. In any case, "the memory" is finite (Stamp, 2012).

In this approach, movements are described by general terms (states), which are a representation easily interpretable by humans and amenable to be exchanged among robots (Patel *et al.*, 2012).

3 Methodologies

In this paper, a discrete Hidden Markov Model is used for segmentation of the observed trajectories which requires the recorded continuous trajectories to be mapped into a codebook of discrete values (Vakanski *et al.*, 2012). To simplify the beginning study, it is assumed that only the situation is happened while a soccer robot to move with carrying the soccer ball. Moreover, the states numbers in chapter 3.1 and chapter 3.2 are both assumptions. In practical situation, the states number can be set as more as required.

3.1 Foundations

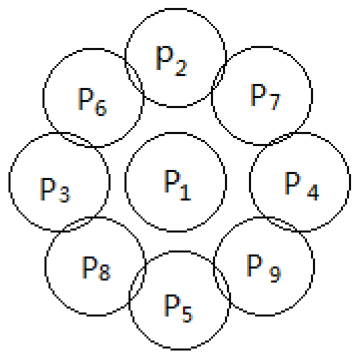

Fig. 2: The original situation for $t = 0$

To explain the above principle with an example, it can be supposed that there are 8 states which a soccer robot can follow when the soccer robot is in state 1 at $t = 0$ and its probability is $P_1(0) = 1$ (Fig. 2). According to normal match, the soccer robot goes in the direction to goal in high possibility. Hence the connect probability $w_{1,2}$ could be set as the largest one because the direction of state 2 is assumed towards to the goal. The connect probability $w_{1,2}$ becomes 0.6 as required. Then, for states 6, 7, 3 and 4, the connect probabilities $w_{1,6}, w_{1,7}, w_{1,3}$ and $w_{1,4}$ could be assumed as equal values like 0.05 respectively.

So that,

$$p_2(1) = w_{1,2} \times p_1(0) = 0.6 \times 1 = 0.6 \quad (8)$$

$$p_3(1) = w_{1,3} \times p_1(0) = 0.05 \times 1 = 0.05 \quad (9)$$

For the most probability to go to state 2, the situation is shown in fig.3. The connect probability $w_{2,10}$ from state 2 to state 10 is the same as state 1 to state 2.

Hence,

$$\begin{aligned} p_{10}(2) = w_{2,10} \times p_2(1) &\approx w_{2,10} \times [w_{1,2} \times p_1(0)] \\ &= 0.6 \times 0.6 = 0.36 \end{aligned} \quad (10)$$

$$p_{11}(2) = w_{2,11} \times p_2(1) \approx 0.05 \times 0.6 = 0.03 \quad (11)$$

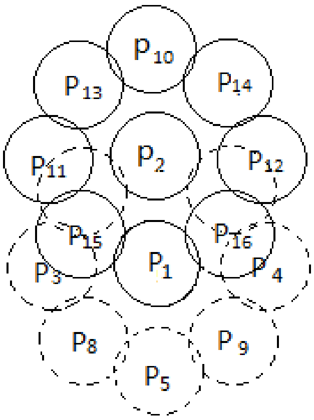

Fig. 3: The setting in state 2 at $t = 1$

3.2 New Development --- Rolling-back Algorithm

From chapter 3.1, the foundations of how to calculate the estimation probabilities are cleared. To simplify the application, but not losing generality, and make the results more visible, there is a 4 direction state example elucidated the application.

Suppose that the original situation is shown in fig. 4. Assume that up move is the direction to goal. That is, the connect probability from the beginning state to up state is the largest one. Generally, left and right directions are the same probability and down direction has the least probability. Blue and red rectangles mean different teams, while the circle one represents the robot carrying the soccer ball under analysis. As the other robots are obstacles, blue and red rectangles are unwalkable.

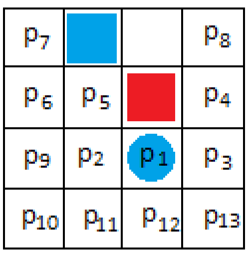

Fig. 4: The transition probabilities at $t = 0$

Presume that $w_{2,5} = w_{6,7} = w_{3,4} = w_{4,8} = 0.6$ and $w_{1,2} = w_{13} = w_{5,6} = 0.15$. $p_1(0) = 1$ as p_1 is the original state.

Then the step by step procedure is listed below.

Step 1: Calculate the probability by formula (3). Substitute the value of connect probability into the formula (3), it is given that $t_{set} = 3$,

$$\begin{aligned} p_{12}(3) = w_{1,12} \times p_1(2) + w_{11,12} \times p_{11}(2) + w_{13,12} \\ \times p_{13}(2) = 0.0195 \end{aligned} \quad (12)$$

$$p_{10}(3) = 0.00675 \quad (13)$$

$$\begin{aligned} p_2(3) = w_{1,2} \times p_1(2) + w_{5,2} \times p_5(2) + w_{9,2} \times p_9(2) \\ + w_{11,2} \times p_{11}(2) \\ = w_{1,2} \\ \times [w_{2,1} \times p_2(1) + w_{12,1} \times p_{12}(1) + w_{3,1} \\ \times p_3(1)] + w_{5,2} \times \left(w_{1,2} \times p_2(1)\right) + w_{9,2} \\ \times \left(w_{2,9} \times p_2(1)\right) + w_{11,2} \\ \times \left(w_{2,11} \times p_2(1)\right) = w_{1,2} \\ \times [w_{2,1} \times (w_{1,2} \times p_1(0)) + w_{12,1} \\ \times (w_{1,12} \times p_1(0)) + w_{3,1} \times (w_{1,3} \\ \times p_1(0))] + w_{5,2} \\ \times \left[w_{2,5} \times \left(w_{1,2} \times p_1(0)\right)\right] + w_{9,2} \\ \times \left(w_{2,9} \times w_{1,2} \times p_1(0)\right) + w_{11,2} \\ \times [w_{2,11} \times w_{1,2} \times p_1(0)] = 0.46125 \end{aligned} \quad (14)$$

$$\begin{aligned} p_8(3) = w_{4,8} \times p_4(2) = w_{4,8} \times \left(w_{3,4} \times p_3(1)\right) \\ = w_{4,8} \times \left[w_{3,4} \times \left(w_{1,3} \times p_1(0)\right)\right] \\ = 0.054 \end{aligned} \quad (15)$$

$$\begin{aligned} p_3(3) = w_{1,3} \times p_1(2) + w_{4,3} \times p_4(2) + w_{13,3} \times p_{13}(2) \\ = w_{1,3} \\ \times \left(w_{2,1} \times p_2(1) + w_{12,1} \times p_{12}(1) + w_{3,1} \\ \times p_3(1)\right) + w_{4,3} \times \left(w_{3,4} \times p_3(1)\right) \\ + w_{13,3} \\ \times \left(w_{3,13} \times p_3(1) + w_{12.13} \times p_{12}(1)\right) \\ = w_{1,3} \\ \times \left(w_{2,1} \times \left(w_{1,2} \times p_1(0)\right) + w_{12,1} \\ \times \left(w_{1,12} \times p_1(0)\right) + w_{3,1} \\ \times \left(w_{1,3} \times p_1(0)\right)\right) + w_{4,3} \\ \times \left(w_{3,4} \times \left(w_{1,3} \times p_1(0)\right)\right) + w_{13,3} \\ \times \left(w_{3,13} \times \left(w_{1,3} \times p_1(0)\right) + w_{12.13} \\ \times \left(w_{1,12} \times p_1(0)\right)\right) = 0.04275 \end{aligned} \quad (16)$$

$$\begin{aligned} p_6(3) = w_{9,6} \times p_9(2) + w_{5,6} \times p_5(2) \\ = w_{9,6} \times \left(w_{2,9} \times p_2(1)\right) + w_{5,6} \\ \times \left(w_{2,5} \times p_2(1)\right) \\ = w_{9,6} \times [w_{2,9} \times \left(w_{1,2} \times p_1(0)\right] + w_{5,6} \\ \times \left[w_{2,5} \times \left(w_{1,2} \times p_1(0)\right)\right] = 0.027 \end{aligned}$$

(17)

Step 2: Find the best trajectory by compare these probabilities above. The biggest probability, which is the most probable trajectory is to $p_2(3)$. Sort out that trajectory and disposal the data from the trajectory every unit t.

Step 3: At $t = 0$, the robot could make a decision for next step at $t = 1$ is $p_2(1)$. It is obviously that the other robots may also move to adjacent states at $t = 1$.

Hence, the available state should be detected again at the same time and calculate the initial probabilities originally from $p_2(1)$ to next step in expected state probability $p_{es}(2)$ as step 1.

In this calculation, time unit could be defined by actual situation in matches. For the simple example in Fig. 4, time unit is 3, that means, judge next step $(t + 1)$ according to the most probable trajectory after $(t + 1 + 3)$. Since the available states are variable, the state is uploaded every time t and it is irrelevant with the history states. What could affect the next step are the connect probability and the actual available states.

When the soccer robot in a state at t, the best trajectory for next step at $(t + 1)$ can be obtained from comparing the state for $(t + 1 + n)$ and back to judge the next step at $(t + 1)$. Hence, it is a rolling-back procedure. This rolling-back algorithm adopts the characteristics of Hidden Markov Model which the state is only accessible from itself and every factors affect the next state in future.

In this approach to the recognition problem, each behaviour is modelled using a separate Hidden Markov Model. Each model is first trained on example trajectories of a specific behaviour. The trained model is then given to check trajectories to determine the log-likelihood that the checked trajectory is generated by that behaviour (Novitzky *et al*., 2014).

3.3 Computational demonstration by MATLAB simulations

In general, a soccer robot has to predict its location and judge the trajectory in the field. Otherwise, it might lose itself and cannot accomplish the specific task in match. Therefore, this chapter is going to apply the Rolling-back algorithm (discussed in section 3.2) to forecast and simulate the actual system which soccer robot will move in a 13*15 soccer field with two Blue team members (blue rectangles) and three Red opposite team members (red rectangles). This application can be simulated in MATLAB. To simplify the complicated problem in programming, assume that the connect probabilities w_{ij} are equal (Wu, 2010). With reference to the flow chart in Appendix, this simulation generally contains three parts as follows:

- Field specification

This 13*15 field is specified based on the match, which can denote the state space of this modelling system. In this simulation, in order to build the field, the soccer robot walks around until it has explored all walkable states. Noticeably, the soccer robot (shown as blue circle) carrying the soccer ball should evade the states which other soccer robots (shown as blue and red rectangles) located. With the purpose of building the state space, the field is discretized into $(13 \times 15 =)195$ state points with green colour. For the other five robots (blue and red rectangles) are obstacles, there are $(13 \times 15 - 5 =)190$ state points left for soccer robot to detect. These state points are used to specify the possible positions that the soccer robot can go.

- Determine transition probability matrix

Calculation of the transition probability matrix is the most central part of the whole procedure. Until the matrix is known, the prediction can be ensured. This matrix can be established based on the given facts. In this case, the soccer robot can explore all the obtainable space to make decision. And then system can calculate the transition probability matrix. For calculating the transition probabilities, the moving law of the soccer robot have to be worked out. Let Dl externalize the distance from the position on the left of soccer robot to all positions. Similarly, Dr, Du and Dd are set for the right, upper and lower position, respectively. If Dl=0, the position on its left is walkable. The Dl, Dr, Du and Dd are mainly used to update the counter N, which represent the number of walkable states by left, right, up, and down side. Therefore, the connect probability is equal to $1/N$ ($1/N$ is only chose to simplify the calculation in this MATLAB simulation program. In fact, the connect probabilities should be modify by the goal's direction and the actual situation in the match).

Firstly, set the initial transition matrix as the original state (where the robot is in at the moment) is equal to 1. And then, calculate the distance from the position around soccer robot to the other points. Also, If Dl (Dr, Dd or Du) = 0, the left (right, down or up) position is walkable.

- Position prediction

As long as the transition probability matrix is obtained, the soccer robot's position can be evaluated after it transfers numerous times on the foundation of the Markov Model. The initial probability distribution can be changed by inputting the integer between 1 and 190. That is, if the 'Initial position' module is input integer 1, the first element of the initial probability matrix is equal to 1 and other elements are equal to 0. In the program, let Ps indicate state probabilities distribution after t transitions. As long as the step time (t) (the number of transitions) is known, its probabilities $P_s(t)$ can be estimated by calculating formula (3). The largest probability can be referred as the estimation of the soccer robot's position. The possible states are coloured with dark to light according to their probabilities large to small.

- Examples of Simulation

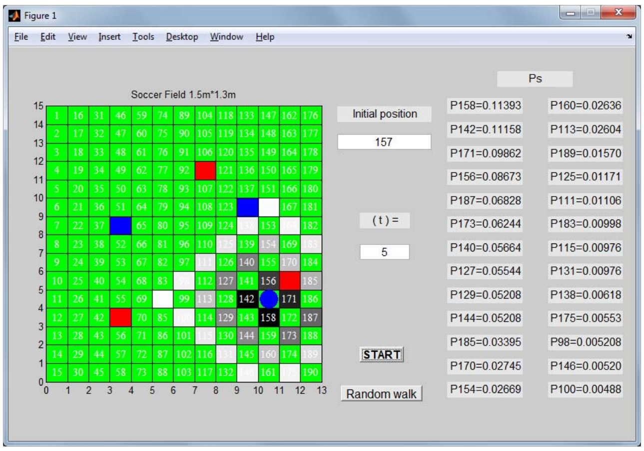

Fig. 5: Example simulation

Fig. 5 is an example of simulation with the first 26 probabilities in a soccer field. Initial position and the step time (t) are input by 157 and 5 respectively.

3.4 Practical implementation

Because this research is only theoretically discussed, examples mentioned above are preliminary and simple. For the purpose of improving the precision and the application scope of prediction in practical implement, formula (3) is the key formula to calculate the probability, and the further work is to detect all factors which would affect the connect probabilities.

For a practical RoboCup situation, the time unit (t), state number and all direct connect probabilities should be delimited by encoding the programme while the soccer field was detecting by sensor. Deserve to be mentioned, the time unit (t) is in accordance with the robot's walking velocity. If robot's walking velocity is lower, the time for it to judge the next step could be longer, and the time unit (t) could be greater. Also, the direct connect probabilities rely on the sensor detected the goal for itself and then let the direct-to-goal connect probability is the greatest one compared to the other connect probability. After that, utilise the predict calculation introduced above to get the most possible direction and to make the decision in next step. When next step is complied, the anew detection is update to the state. Meanwhile, calculate the probability and refresh the next direction.

The procedure can be expressed by five stages:

- Begin by regarding the phenomenon as a system and dividing this system into states according to the actual situation and requirement of system. These states are the contents of forecast.
- Next is to compute the initial probability distribution. That is to find out the current state of the system.
- Thirdly is to calculate the probabilities of moving from one state to another. That is called connect probability and is to determine how the system makes transitions.
- Fourth is to extrapolate the trajectory of reaching the most probable states after several transitions according to the initial state and then move to next state.
- Finally, upload the state and detect the adjacent states is available or not. Refresh all the calculation and take a start from the first stage.

This progress can be organised as an excellent self-adaptation control system.

The key point of Markov Model is the dependability of rolling-back algorithm. That is why the model involves numbers of statistical data with high accuracy. Only by doing so can guarantee the accuracy of prediction. Such statistical data request comprehensive records of changes in the long-term. However, in practical situation, it is very tough to acquire these data no matter in terms of time or in terms of space. The defect of this algorithm is that it needs a large amount of work. The probability of each state after a great number of transitions mainly not only depends on the connect probabilities, but also depends on the initial state of system. Thence, in order to precisely predict the future states, a large number of key work is to determine the connect probabilities based on the judgment of system's present state and the actual situation of RoboCup events. Accompanied with the development of system, the calculations of matrix probabilities will increase the workload. In order to decrease the workload, factors should be focused on. But the factors consolidation should be informed.

The rolling-back algorithm defaults that the Markov Model discussed here satisfies homogeneity. But in actual practical application, the stochastic sequence may not satisfy the homogeneity. Under this situation, perhaps overlying the predicting probabilities of different orders should be considered together, because these probabilities may donate to the prediction. Furthermore, the influences of these probabilities of different orders on the prediction can be different in fact. (Wu, 2010) In order to improve the system, this kind of considerations should be taken into account. That is to say, these predicted results of different orders should be superimposed with different weighting factors.

All in all, the discussion and summarise above provides a better algorithm based on Hidden Markov Model to locate and forecast the localisation of soccer robots in practical situation. With examples shown in chapter 3.1 and chapter 3.2, the algorithm can be obviously implement in practical situation.

4 Conclusions

This paper has briefly formulated a research problem (location of soccer robots), proposed a solution procedure (based on Hidden Markov Model), and finally used computational experiment approach to demonstrate the analytical results with MATLAB simulations. With the research understanding, Markov Model has been a very useful framework to accommodate predicting problems encountered in many stochastic systems, which its applications in soccer robot location could promote many other interesting research issues as well as increasing the robot intelligence. Once again this is the beginning stage presented in the paper and comprehensive future studies have been planned accordingly.

References

[1] Bae, J., and Tomizuka, M. (2011) Gait phase analysis based on a Hidden Markov Model. *Mechatronics*, 21(6), 961-970.

[2] Dai Kui, Q. G., & Shouren, H. (1999). A Discrete Markov Model Based Neural Network Method for Reliability Design [J]. *COMPUTER ENGINEERING & SCIENCE*, *3*.

[3] Ghahramani, Z., & Jordan, M. I. (1997). Factorial hidden Markov models. *Machine learning*, 29(2-3), 245-273.

[4] Novitzky, M., Pippin, C., Collins, T. R., Balch, T. R., & West, M. E. (2014). AUV behavior recognition using behavior histograms, HMMs, and CRFs.*Robotica*, *32*(02), 291-304.

[5] Patel, M., Miro, J. V., & Dissanayake, G. (2012, June). A hierarchical hidden markov model to support activities of daily living with an assistive robotic walker. In *Biomedical Robotics and Biomechatronics (BioRob), 2012 4th IEEE RAS & EMBS International Conference on* (pp. 1071-1076). IEEE.

[6] Rahman, S., & Zelinsky, A. (1999). Mobile Robot Navigation based on localisation using Hidden Markov Models. In *Australasian Conference on Robotics and Automation, ACRA*.

[7] RoboCup. (1998) *Introduction of RoboCup*. Available from: http://www.robocup.org/about-robocup/objective/ [Accessed on 16/11/2014]

[8] Ross, S. M. (2002). *Probability models for computer science*. San Diego: Harcourt Academic Press.
[9] Stamp, M. (2004). A revealing introduction to hidden Markov models. *Department of Computer Science San Jose State University*.
[10] Vakanski, A., Mantegh, I., Irish, A., & Janabi-Sharifi, F. (2012). Trajectory learning for robot programming by demonstration using hidden Markov model and dynamic time warping. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, *42*(4), 1039-1052.
[11] Wu, H.Q. (2010) *Discrete Time Markov Model Parameter Estimation with Application to Position Estimation for Mobile Robots in Unknown Environments*. MSc dissertation. University of the West of England, Bristol, UK.

Appendix: