**IOSR Journal of Engineering (IOSRJEN)** **www.iosrjen.org**
ISSN (e): 2250-3021, ISSN (p): 2278-8719
Vol. 05, Issue 07 (July. 2015), ||V4|| PP 53-68

# Simulation study of Markov chain models with an application in Cognitive Radio Networks

Arunkumar B R [1], Raghu M S[2],
*Prof. & Head, Dept. of MCA, BMS Institute of Technology & Management, Bangalore, India,*
*PG student, Dept. of MCA, BMS Institute of Technology, Bangalore, India,*

***Abstract:*** Spectrum management is a crucial task in wireless networks. The research in cognitive radio networks by applying Markov is highly significant suitable model for spectrum management. This research work is the simulation study of variants of basic Markov models with a specific application for channel allocation problem in cognitive radio networks by applying continuous Markov process. The Markov channel allocation model is designed and implemented in MATLAB environment, and simulation results are analyzed.

***Keywords:*** *Cognitive radio networks, channel allocation, Markov Model.*

## I. INTRODUCTION

Andrey Markov began the study of an important new type of chance process. In this process, the outcome of a given experiment can affect the result of the further experiment. This process is called a Markov chain. As per this process, the probability distribution of the next state depends only on the current state and not on the sequence of events that is introduced. This specific kind of memory less (previous state less) is called the Markov property [1]. Markov chains have many applications in diverse fields of science, engineering and technology and in particularly in disciplines of computer science as statistical models of real-world processes. The concept of Markov chains are probability graphs appropriate in computer science and natural sciences as well. Applications of Markov chains can be found everywhere including statistical physics, economy, biology, ecology and even in the stock market, the study of the web based application, cloud computing, cognitive radio networks and in many more. In addition to it, Markov chains are in many combinatorial applications such as approximation, optimization and counting algorithms etc. The three important elements in Markov chains are Probability transition matrix, Steady state vector and Transition diagram [1].

The three important elements in Markov chains are Probability transition matrix, Steady state vector and Transition diagram.

**Probability transition matrix P:**

The switch between states is established in the probability transition matrix P. Each element of it represents the probability that switches or remains in the state. These switches are called transitions. P is a square matrix whose order is the same to the number of states. The below equation shows the structure of probability matrix,

$$P = \begin{pmatrix} P_{00} & P_{01} & P_{02} & \dots \\ P_{10} & P_{11} & P_{12} & \dots \\ \cdot & \cdot & \cdot & \dots \\ P_{i0} & P_{i1} & & \dots \\ & & & \\ P_{ij} \geq 0, & & i,j=0,1,2,3\dots & \end{pmatrix}$$

$\cdot \quad \cdot \quad \cdot$

Where $P_{ij}$ is the probability of the state i given the immediate precedent past time was in state j. i and j are transition states. The probability transition matrix is that the sum of each row must be equal to 1.

The system can be modeled using Markov chain efficiently and can be solved using mathematical methods .For eg, in wireless communication Markov chains can characterize the channel in the best way as it can represents the combination of some phenomena that affects the signal during its wireless transmission, having a better approach to a real channel.

**Transition diagram**

A transition diagram is represented by probability transition matrix. A state of the Markov chain is represented using a node with a number inside. An arrow that connects state m with state n if a transition exists and the transition probability $P_{mn}$ is written on that connecting arrow, even if the transition is to the same state. The state changes depends on the probability. The below diagram depicts the transition diagram for two state Markov chain.

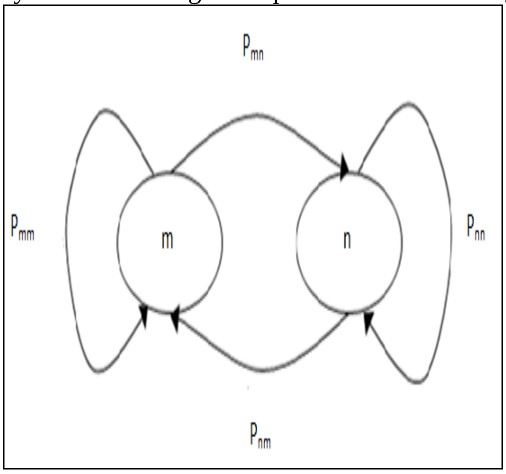


**Figure 1. Transition diagram for Probability matrix**

$$P = \begin{pmatrix} P_{mm} & P_{mn} \\ P_{nm} & P_{nn} \end{pmatrix}$$

**Steady state vector:**

The third important element in Markov chains is the steady-state vector π, which represents the total appearing percentage of a state in a Markov chain. The computation of this vector can be done by raising P to the large power, $P^n \rightarrow 1\pi$ , where P is probability transition matrix, $\pi$ is Steady state probability vector and 1 is column vector.

**1.1.1 Types Of Markov Chains:** The following are the three variants of basic Markov chain: Discrete time Markov chain, Continuous time Markov chain and Markov decision process

**I. Discrete time Markov chain (DTMC):**

In this type, past history of the system is not considered but concentrates on the current history. In the literature, discrete time Markov chain is also referred as the discrete valued Markov process that evolves changes. For eg , in the snake and ladder board game past history of the system is not required and it is discrete time Markov chain process because the changes to the system state can only done on someone's turn.

**II. Continuous time Markov chain:**

The Poisson process drives continuous time Markov chain. In this type, state changes on the continuous interval of time. It takes some finite values for which the time spent for each state that takes the non-negative real values and has the exponential distribution.

**III. Markov decision process:**

It is the process of decision making in address to decision maker where conclusion is partially random below the regulation. The decision changes according to the decision made by the decision maker that affects the growth of the process. Markov decision process is very useful in study of the wide range of optimization problems and this can be solved by the dynamic programming.

**1.1.2 Some applications of Markov chains are:**

i. The representation of both graphs and matrix that can be described randomly.
ii. The characterization of economic correlations is commonly make use of Markov chains.
iii. Used to check the randomness effect of the corporation's financial health.
iv. Google uses Markov chains to rank the pages discovered in a user's search.
v. To represent a class of stochastic processes of great interest for the wide spectrum of practical applications. In exact, discrete time Markov chains (DTMC) permit to model the transition probabilities between discrete states by the aid of matrices.
vi. Stock prices can be predicted very easily with the help of Markov chains.
vii. Used to estimate models that are difficult to tackle with standard tools such as Instrumental Variables regressions, OLS and Maximum Likelihood.
viii. Used to predict the performance of a Divisible Load Application in a Cloud Computing Environment.
ix. Markov chain concepts are used to analyze the behavior in complex distributed systems.

**1.1.3 Advantages and Disadvantages:**

The some advantages of Markov chains are as follows:

i. Markov models are comparatively easy to assume from succession data.
i. The Markov model does not need a broad intuition into the mechanisms of dynamic shift, but it can help to reveal areas where such insight would be valuable and hence act as both a guide and substance that invigorates to further research.
ii. The basic transition matrix encapsulates the essential parameters of dynamic change in a way that is produced by very few other types of model.
iii. The results of the Markov chains are readily flexible to graphical presentation and in this form, are commonly more readily presented to, and assumed by, resource managers and decision- makers.
iv. The computational requirements of Markov chains are simple, and can easily be met by small computers and for small numbers of states, by simple calculators.
v. The disadvantages of Markov chains are
vi. The lacks of confidence on functional mechanisms decrease their appeal to the functional orientated ecologist.
vii. The withdrawal from the simple premise of stationary, first-order Markov chains while, conceptually attainable, makes for unreasonable degrees of difficulty in analysis and computation.
viii. In some areas, the data available will be lacking to estimate reliable probability especially for rare transitions. For ex, it may not be possible to observe proportionate transitions from a given transient set of states to a closed state where this transition is dependent on a rare critical event, even though the value of this parameter is of essentially important in the dynamics of the community.
ix. Like all other succession models, the validation of Markov chains rely upon predictions of system behavior over time, and is therefore again and again difficult, and may even be hopeless, for really long periods of time.

**1.2 Statement of problem:**

This paper addresses the channel allocation problem in cognitive wireless networks where multiple channels owned by primary users with multiple cognitive users is considered. Always primary user transmits on the dedicated channel, every cognitive user is intelligent enough to sense one channel at a time and transmit according to a slotted structure, if the channel is currently not being used by the primary user. Cognitive user will be preempted at any point of time if primary user is willing to use the channel. The channel allocation is modeled by a continuous time Markov process. Further, the transmissions of cognitive users on each channel are possible if certain predefined collision constraints are met.

This work assumes the collision constraint as considered in the paper authored by "Shiyao Chen and Lang Tong" where the maximum throughput region is obtained by a policy called as orthogonalized periodic sensing (stiff constraints). When collision constraints are slack the work characterizes the maximum throughput region. The cognitive users are allowed to access to the channels only if the constraint there interference below defined levels.

**1.3 Contributions of the paper:**

The main contributions are :

- Explore the concept an applications of Markov Chain.

- Simulate and investigate the basic Markov model with Poisson distribution.
- Study and simulate hidden Markov model with Poisson distribution.
- Design the algorithm for channel access problem to the model using Markov chain.
- Implementation of new algorithm on a MatLab environment, conduct experiment and analyze the result.

**1.4 Methodology:**

In this section, the work presents the details of the methodology for evaluating the robustness of the cognitive radio networks using Markov chain process. For critical purposes, we create describing the simplest setup in which our method can be applied in computing the problems like channel allocation in cognitive networks and spectrum analysis in wireless network. The continuous time Markov process is one of the solutions for these problems.

To solve the channel access problem in CRN as discussed in problem statement following methodology is adopted.

The steps are described below

I. Choose any two subsets of CRN randomly where each network can choose between two events.
II. Connect these two subsets to form a network which can be of four possible states {X1, X2, X3, X4}.
III. Compute probability transition matrix for the transition given.
IV. Compute P which gives the probability of transition from state X2 to X3. Followed by calculating $P^2$ absorbing the continuous Markov chain to allocate the channels for CRN networks.

## II. LITERATURE SURVEY

A reasonable amount of work is done by applying hidden Markov models (HMMs) to spectrum sensing in the temporal area. While more than a few researchers focus on modeling the state of the primary user and the output process, others think the models are available for their applications. In each case, the model parameter is utilized to improve the channel state estimation, and more significantly, to forecast future activity of the primary user in order to dynamically access the available spectrum. This section briefly evaluate some of the allied work and concentrate on the differences compared to the proposed approach. Most of the existing work has shown that, in common, discrete time HMMs are adequate for channel modeling. However, advanced models are needed in some applications to better signify the channel occupancy of the primary users.

**2.1 Modeling Using Standard Hidden Markov Models**

Modeling the spectrum sensing problem using a standard hidden Markov model signifies the state of the primary user is modeled by moreover a discrete-time or a continuous-time finite-state homogeneous Markov chain while the output process is modeled by either a finite-alphabet or a general-alphabet process. With the finite-alphabet output process, the number of observations observed in every hidden state is finite; and the probability of an observation produced by a particular hidden state is characterized by a probability mass function. In other words the general alphabet output process, the number of observations is not necessary finite and its distribution is usually characterized by a probability density function [3].

In [7], observed measurements taken in the 928 to 948 MHz aging band have validated a Markovian pattern in the spectrum occupancy of the primary user. The main objective of this paper is to recover the hidden state sequence given the observation sequence using the Viterbi algorithm. The spectrum sensing problem was formulated using a simple HMP with two hidden states and two observable states. If the primary user is active, the hidden state is assigned to 0; otherwise, it is assigned to 1. The observation in this model is not the received signal power, but to a certain extent it is a binary value of 0 or 1 depending on the power is higher or lower than a defined threshold, respectively. The probabilities of missed detection and false alarm are used in expressing the observation probabilities given the state. This model is quite simple and cannot be generalized in many cases due to the dependence on the threshold. Moreover, the parameter estimation and the state prediction are not discussed in details.

In [2], an algorithm called Markov-based channel prediction algorithm is proposed for dynamic spectrum allocation in cognitive radio networks. The algorithm is based on a Markov chain with a finite-state observable process, whose parameter is estimated online using the forward part of the Baum-Welch algorithm. Using the estimated parameter, activity of the primary user is estimated based on the joint probability between the observation sequence and the state. The cognitive radio utilizes the likelihood of the state estimation to make channel access decision. The proposed approach shows significant signal-to-interference (SIR) performance compared to the traditional Carrier Sense Multiple Access (CSMA) based approach, where the cognitive radio identifies an empty channel and operates on it until the primary user's signal is detected. The duration in each state of the primary user is assumed to be Poisson distributed. Although prediction algorithm is mentioned in the paper, only state estimation is actually performed. On the other hand, this approach is only applicable to high SNR scenarios without considering the detection errors in modeling.

A channel status predictor based on a Markov chain with a finite-alphabet observable. Process was proposed in [9]. The parameter of the model is estimated from the training data using the Baum-Welch algorithm. The observable process has two states, where 1 represents the OFF state while 2 represents the ON state. The work here focuses on predicting m-step channel state using the forward algorithm. Although the formulation of the model is clearly defined, the performance analysis section in this paper is not well presented. Another problem is that periodic patterns were used as inputs to test the performance of the predictor, which diminishes the advantage of using HMM.

In [5], a modified HMM is proposed to predict the channel state of a single primary user in order to minimize the negative impact of response delays caused by hardware platforms. The modified HMM is developed by shifting the time indexes of the underlying state and the observation to include the maximum possible response delay. Instead of using the Baum re-estimation algorithm to estimate the parameter of the proposed model, the authors of [5] estimate the parameter through a simple statistical process over training sequences. The proposed model is then evaluated using real-world Wi-Fi signal collected in an indoor environment. The performance of the proposed approach is improperly compared with another prediction approach, which assumes the predicted state is the same as the current state.

Unlike the above work, in [10], the spectrum sensing problem in the temporal domain is modeled by a discrete-time Markov chain with general-alphabet observable process. Several sequence detection algorithms derived from the Viterbi and forward-backward algorithms were presented to uncover the state sequence of the primary transmitter given the entire observation sequence observed at the cognitive radio node. The proposed detection sequence algorithms incorporate the probabilities of missed detection and false alarm into the schemes through the use of Bayesian cost factors. Part of our work in [8] is similar to [10] in modeling the primary user's transmission pattern and observation statistics. However, instead of focusing on state sequence detection, we concern ourselves with online single state detection. Although the Baum-Welch algorithm was mentioned in the paper, it has not been implemented to estimate the model parameter. Furthermore, the problem of predicting future activity of the primary transmitter has not been considered in this work.

### 2.2 Modeling Using Extended Hidden Markov Models

Several researchers have argued that in some applications the channel occupancy of the primary transmitter cannot be properly described by a discrete-time or a continuous-time Markov chain. Therefore, it is mandatory to extend the standard hidden Markov model to a more advanced model such that it can accurately capture the statistics yet possess tolerable computation complexity. In [6], it has been shown that the idle and active periods of the busty transmissions of a wireless local area network (WLAN) are not geometrically distributed but rather have phase type distributions. In another paper [4], a generalized Pareto distribution has been used to statistically model the WLAN data. In both cases, a continuous-time semi-Markov process was proposed as a possibly suitable model for this application but no detailed estimation algorithm was provided. The transition probabilities and the dwell time distributions are approximated in a simple way by matching the statistics of the collected data. Since the work here only focuses on modeling the activity of the primary transmitter, the observable process has been eliminated. The idle and active states of the primary transmitter were identified by using either an energy-based detection or a feature-based detection. It is mentioned in [6] that the state prediction of one-step can be utilized by the secondary user to better access the channel, but no procedure or algorithm was given.

In [11], practical primary user's traffic, whose busty nature is not properly described by a Markovian pattern, was considered for the opportunistic spectrum access in temporal domain. In particular, two source traffic models were investigated for the primary user. The work in this paper focuses on designing an optimal spectrum access scheme which maximizes while keeping the interference to the primary system below a given threshold. Numerical results have shown that the channel access strategy of the secondary user depends on both the elapsed time of idle period and the characteristics of the primary user traffic.

Therefore, it is very important to model the primary traffic accurately. The primary traffic in this paper was not modeled by any hidden Markov model, but instead, by two specific distributions with known parameters. With a different approach, we propose a bivariate Markov chain to model the transmission pattern of the primary transmitter for cognitive radio application. The state durations of the bivariate Markov chain have a discrete phase-type distribution. The parameter estimation algorithms in our approach are derived from the Baum re-estimation algorithm, which seeks to maximize the likelihood of the observation data.

In the papers [2] – [11], the node's intelligence does not implement cognitive users which can immediately give place for primary users as and when required by them. In this work research efforts are put in this direction.

**Research Design**

The design in this work involves the representing of the channel allocation problem in a cognitive framework including unlicensed and licensed users with many traffic conditions applying a continuous-time Markov chain model.

The main objective is to recover the hidden state sequence, given the observation sequence using the Viterbi algorithm. The spectrum sensing problem was formulated using a simple hidden Markov process with two hidden states and two observable states.

Consider a cognitive radio network consisting of two cognitive sub-networks where each sub-network can choose between two events. This network can have four possible states,{X1, X2, X3, X4}. Assume that form as experimental observation, we can create the probability transition matrix for the transition given in Figure 2 where each state is represented as a circle and each transition is represented as a weighted and directed edge labeled with its associated transition probability.

$$P = \begin{array}{c|cccc} & X1 & X2 & X3 & X4 \\ \hline X1 & 1 & 3 & 1 & 5 \\ X2 & 4 & 0 & 3 & 3 \\ X3 & 4 & 1 & 3 & 2 \\ X4 & 1 & 4 & 3 & 2 \end{array}$$

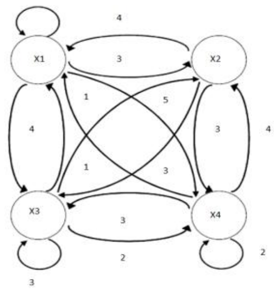


**Figure 2: Digraph Representation for given Transition Matrix**

As shown in the Figure 2, P gives the probability of transition from state X2 to state X3 as 3, followed by calculating $P^2$ as shown by the matrix that follows, one can straight away determine the probability of the system operating in state X4 after two iterations after starting in X3.($P_{34}$=33)

$$P^2 = \begin{array}{c|cccc} & X1 & x2 & x3 & x4 \\ \hline X1 & 22 & 24 & 28 & 26 \\ x2 & 19 & 27 & 22 & 33 \\ X3 & 22 & 23 & 22 & 33 \\ X4 & 31 & 14 & 28 & 27 \end{array}$$

We can readily find the limiting distribution that can model as Ergodic chain for cognitive radio networks. Still the "steady state" to some extent inadequate as the network will not remain same at the single state and in all states will have no zero probability of being occupied and thus the "steady-state" of an Ergodic Markov chain does not adjust to our expectations. This can be solved by absorbing Markov chains.

The following algorithms are used in Markov chain model

4.1 BASIC ALGORITHM FOR MARKOV CHAIN PROCESS:

Step 1 - Initialize the state of the system $x_0$;

Step 2 - For the given state x of the system, calculate the transition rates $\lambda_i(x)$, I = 1, ...,n ;

Step 3 - Calculate the sum of all transition rates, $\lambda = \sum_{i=1}^{l} \lambda i(x)$

Step 4 - Simulate the time T, until the next transition by drawing from an exponential distribution with mean $1/\lambda$;

Step 5 - Simulate the transition type by drawing from the discrete distribution with probability Prob(transition = i) = $\lambda_i(x)/\lambda$. Generate a random number $r_2$ from a uniform distribution and choose the transition as follows: If $0 < r_2 < \lambda_1(x)/\lambda$, choose transition 1, if $\lambda_1(x)/\lambda < r_2 < (\lambda_1(x) + \lambda_2(x))/\lambda$ choose transition 2, and so on

Step 6 - Update the new time t = t + T and the new system state
Step 7 - Iterate steps 2 and step 6 until $t \geq t_{stop}$

**4.1.1 Flowchart For Basic Markov Chain**

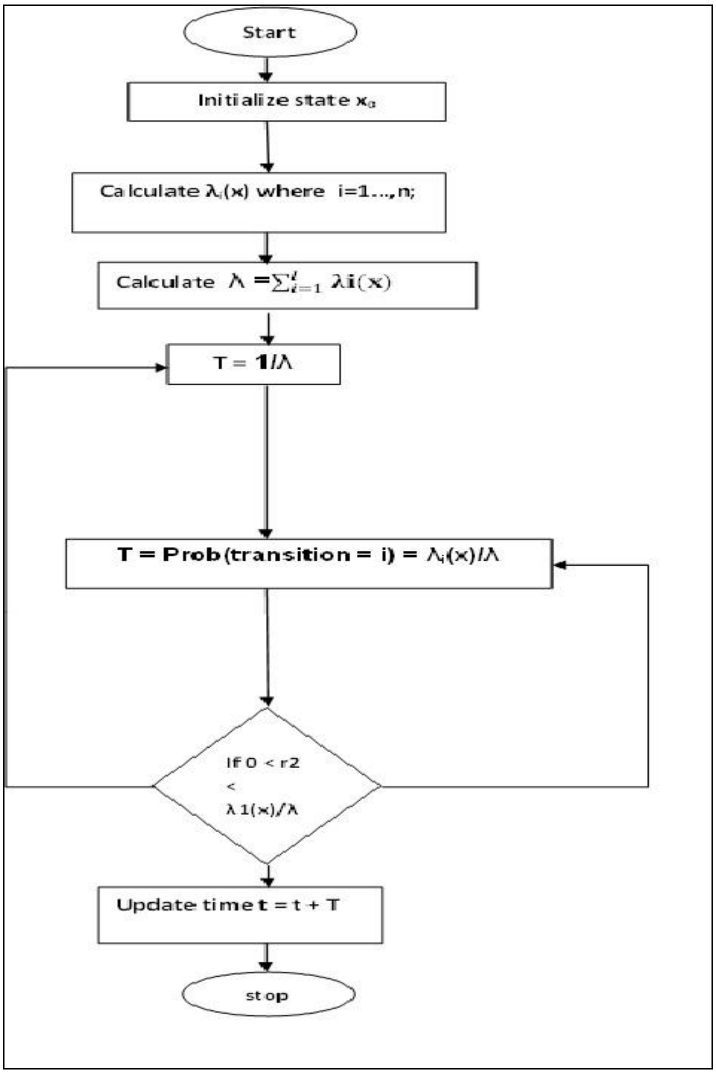


**Figure 3. Flow chart of Basic Markov Chain**

**4.2 Poisson Distribution In Markov Chains**
Step 1 – Generate packet arrival vector for Poisson distribution.
Step 2 – Generate service time vector for exponential distribution.
Step 3 – Initialize n = 0 and i = 1.
Step 4 –If i <= m-1, A=arrival (i) + service (i). if A<(arrival(i+1)) , N=n+1
Step 5 – Calculate  p_1 = n/m, p_0 = 1-p_1.

### 4.2.1 Flowchart For The Poisson Distribution In Markov Chains

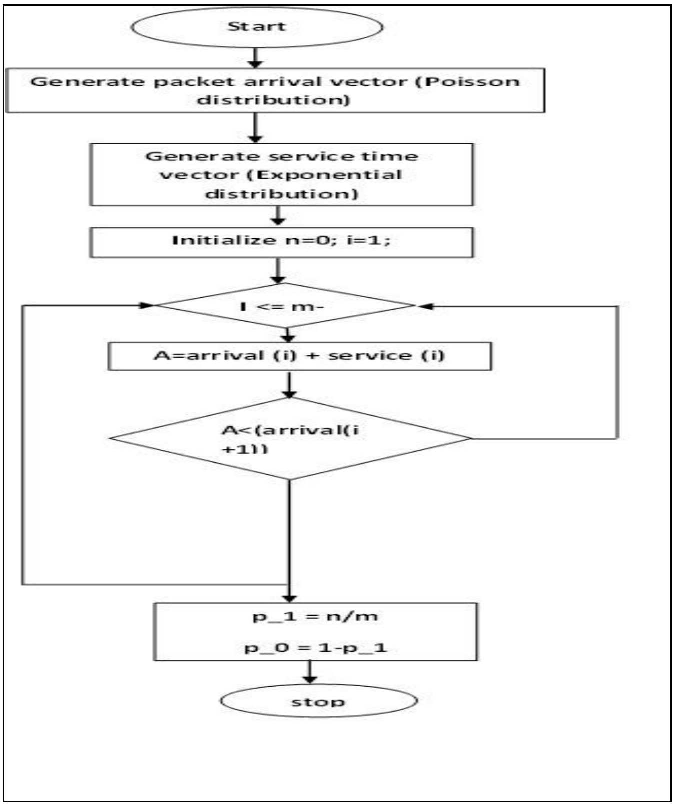


**Figure 4. Flowchart for Poisson Distribution**

### 4.3 Poisson Hidden Markov Model

The algorithms are proposed in Poisson hidden Markov model to solve the three basic computational problems of hidden Markov modeling.

**The three basic problems are:** Compute the likelihood, the optimal estimate of the state or state sequence and Hidden Markov model (HMM) parameter. More precisely, compute the maximum-likelihood (ML).

To compute likelihood the forward backward algorithm is used. In fact, the forward backward algorithm consists of two separate algorithms, the forward algorithm and the backward algorithm.

### 4.3.1 Forward Algorithm:

In forward algorithm the frame of reference of hidden Markov model is used. This is used to estimate the amount of the belief state. In forward algorithm the confirmation of the history or previous state is given as probability at certain state. The forward algorithm is based on recursive relation. Let $\alpha_n(i)$, $0 \leq n \leq N$, $1 \leq i \leq M$ be the forward variable defined as $\alpha_n(i)$, $0 = p(y_0^n, X_{n=}i; \lambda)$.

Step 1 – Initialize $\alpha_0(i)$

Step 2 – Simulate the Poisson distribution $\alpha_0(i) = \pi_i b_i(y_0)$,if $1 \leq i \leq M$.

Step 3 – Initialize n and j where n = 0,1….,N-1.

Step 4 – for n = 0,1….,N-1,

$$\alpha_{n+1}(j) = \left(\sum_{i=1}^{M} \alpha_n(i) a_{ij}\right) b_j(y_{n+1}), \quad 1 \leq j \leq M.$$

Step 5 – terminate

$$p(y_0^N; \lambda) = \sum_{i=1}^{M} \alpha_N(i).$$

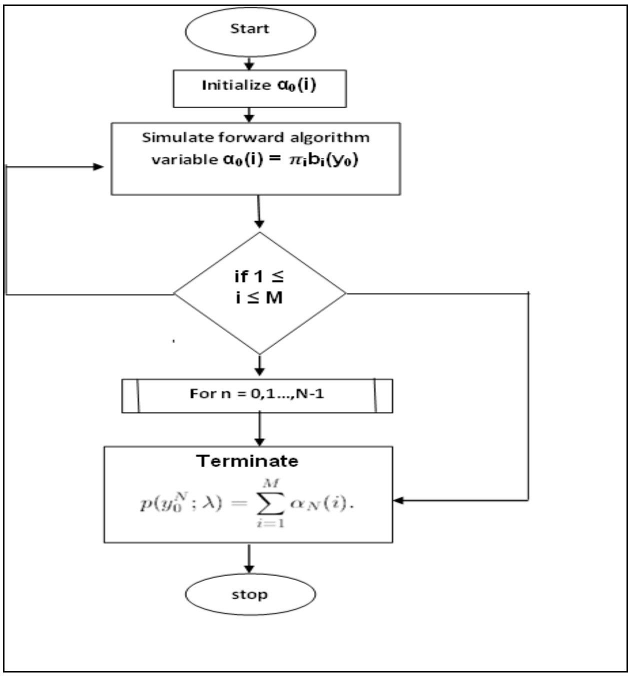


**Figure 5. Flowchart for Forward Algorithm**

**4.4.2 The Backward Algorithm**

The forward algorithm has the forward variable $\alpha_n(i)$. Similarly, the backward algorithm has the backward variable $\beta_n(i)$ that can be computed recursively.

Step 1 – Initialize the backward variable $\beta_n(i)$.

Step 2 – Calculate the backward variable $\beta_n(i)$,Where $\beta_n(i)=1$, $1 \le I \le M$.

Step 3 – Initialize n, where n = N-1,N-2,…..,0.

Step 4 – Simulate the backward variable by below equation

$$\beta_n(i) = \sum_{j=1}^{M} a_{ij} b_j(y_{n+1}) \beta_{n+1}(j), \quad 1 \le i \le N.$$

Step 5 – Terminate the poisson distribution by below equation

$$p(y_0^N;\lambda) = \sum_{i=1}^{M} \beta_N(i)\pi_i.$$

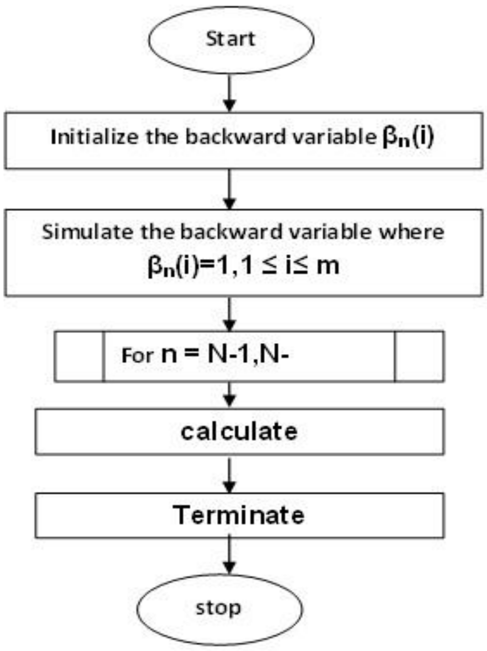


**Figure 6. Flowchart for Backward Algorithm**

### 4.4.3 Viterbi Algorithm

The forward backward algorithm is similar to the Viterbi algorithm. The Viterbi algorithm is also called as Viterbi decoder. The maximization of the HMM can be performed efficiently by the dynamic programming algorithm called Viterbi algorithm.

Step 1 – Initialize the variable $\delta_0$(i)

Step 2 – Where $\delta_0(i) = \pi_i b_i(y_0)$, $1 \leq i \leq M$.

Step 3 –Initialize n , where n = 0

Step 4 – for n = 0,2,….N-1. Calculate the maximization by the equation give below :

$$\delta_{n+1}(j) = b_j(y_{n+1}) \max_{1\leq i\leq M} [\delta_n(i)a_{ij}], \quad 0 \leq j \leq M,$$

$$\psi_{n+1}(j) = \arg\max_{1\leq i\leq M} [\delta_n(i)a_{ij}], \quad 0 \leq j \leq M.$$

Step 5 – After the maximization the termination takes place. For termination the below given equation is used.

$$\hat{P} = \max_{1\leq i\leq M} \delta_N(i),$$

$$\hat{x}_N = \arg\max_{1\leq i\leq M} \delta_N(i).$$

Step 6 – Backtracking is also done in Viterbi algorithm, for n= N-1,N-2,….0

$$\hat{x}_n = \psi_{n+1}(\hat{x}_{n+1}).$$

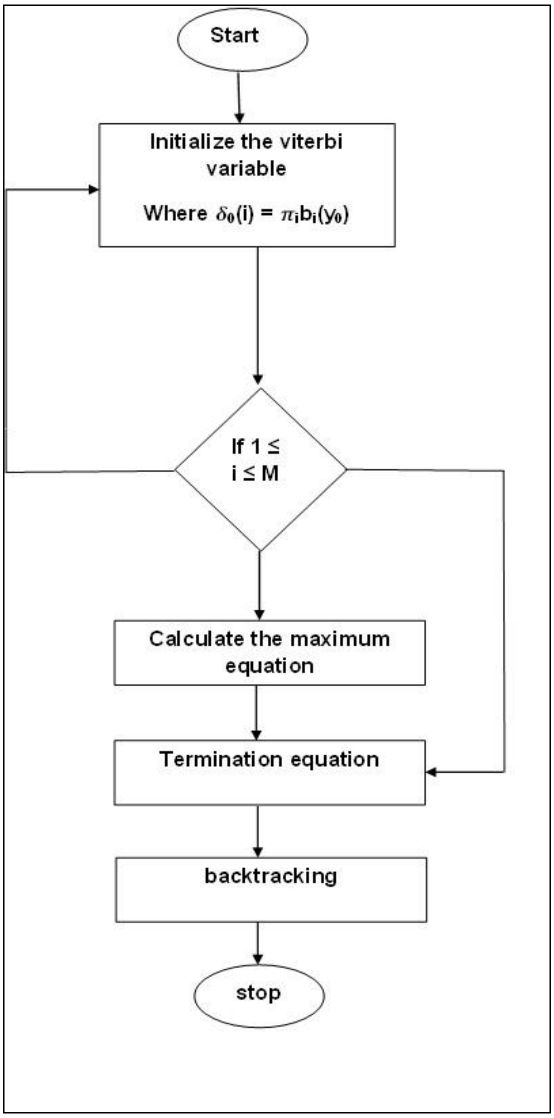


**Figure 7. Flowchart of Viterbi Algorithm**

### 4.4.4 The Baum-Welsh Algorithm

The estimation of the parameters of a hidden Markov model can easily be casted as a missing data problem. In HMM that is observed the complete data and incomplete data. In this algorithm the likelihood(forward backward algorithm) can be maximized by the EM algorithm.

Step 1 – Find the initial estimation $\lambda^{(0)}$ of $\hat{\lambda}$

Step 2 – Set $\lambda = \lambda^{(0)}$

Step 3 – Compute $\check{\lambda}$ by the estimation formula given below

$$\check{\pi}_i = \gamma_0(i) \quad 1 \le i \le M.$$

$$\check{a}_{ij} = \frac{\sum_{n=0}^{N-1} \xi_n(i,j)}{\sum_{n=0}^{N-1} \gamma_n(i)} \quad 1 \le i,j \le M,$$

$$\check{\theta}_i \in \arg\max_{\bar{\theta}\in\Theta} \sum_{n=0}^{N} \gamma_n(i) \ln f(y_n;\bar{\theta}), \quad 1 \le i \le M.$$

Computed with respect to $\lambda$

Step 4 – Set $\lambda = \check{\lambda}$

Step 5 - Go to step 3 unless some convergence is occur

Step 6 –Set $\hat{\lambda} = \check{\lambda}$.

**Simulation Results**

By following the methodology explained in section 1.4, MATLAB codes are written[12][13][14] and run, a simulation environment is created and experiments results are shown in the Figures below.

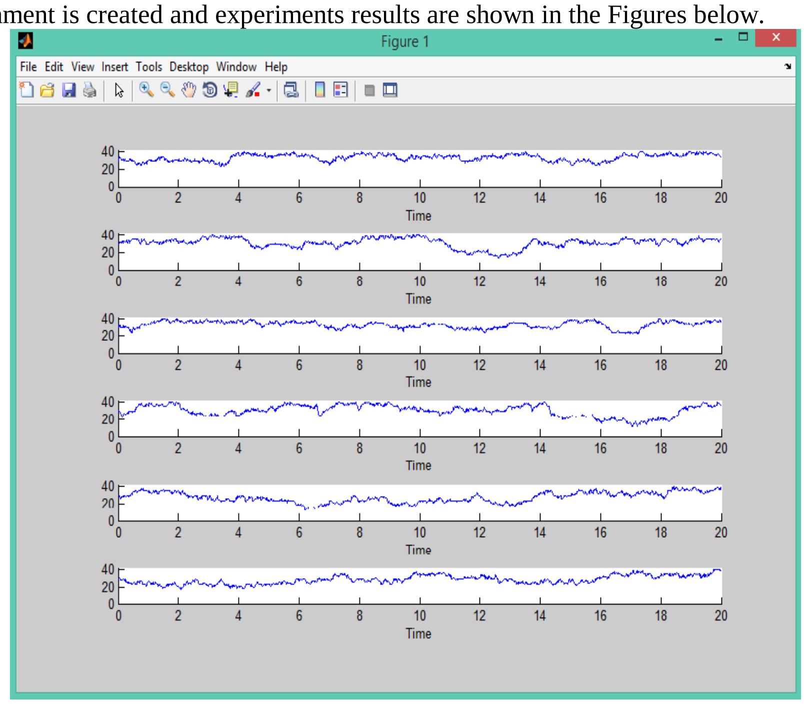


**Figure 8: Simulation Results- general Markov Process**

The above graph depicts the creation of multiple simulations using the general Markov process on given an initial distribution, models parameter and a transition matrix for a three state Markov process.

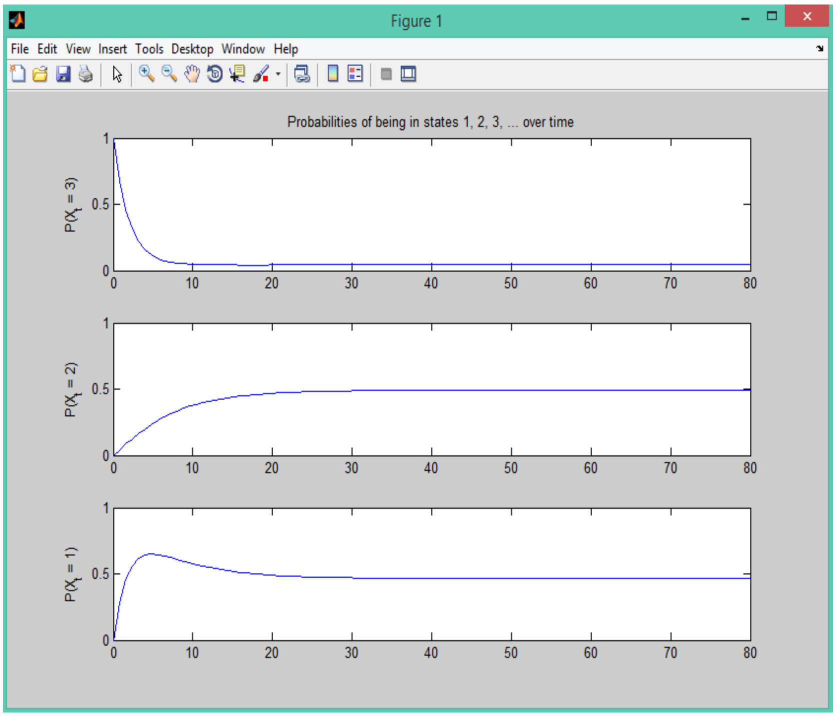


**Figure 9. Distribution of Markov process**

The above graph depicts the distribution of Markov process over time with the probability of state space for three state Markov process.

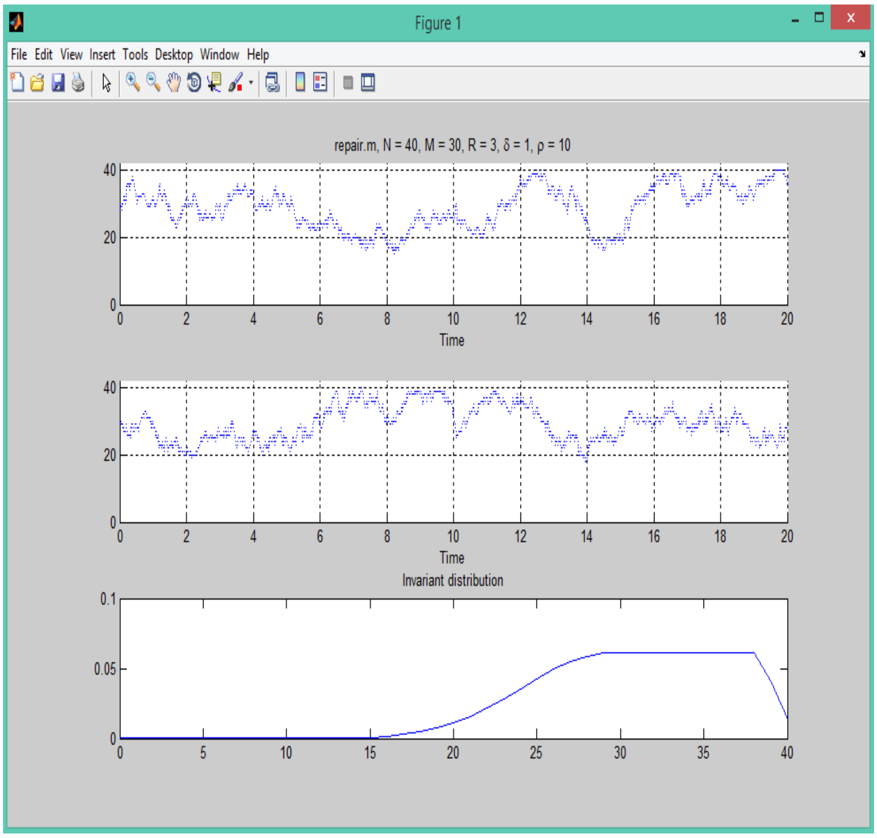


**Figure 10. Invariant distribution Graph**

The above graph depicts the number of machines working and the number of machines under repair and also it shows the invariant distribution corresponding to the given matrix in Markov chain process.

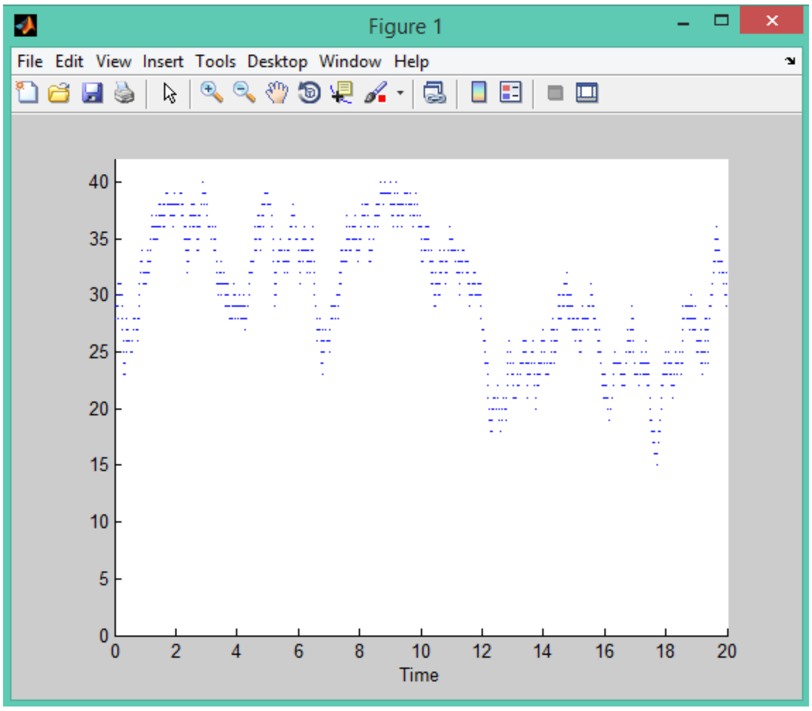


**Figure 11. General Markov with single state**

The graph depicts the simulation of Markov process on a values given an initial distribution, model parameter and a transition matrix.

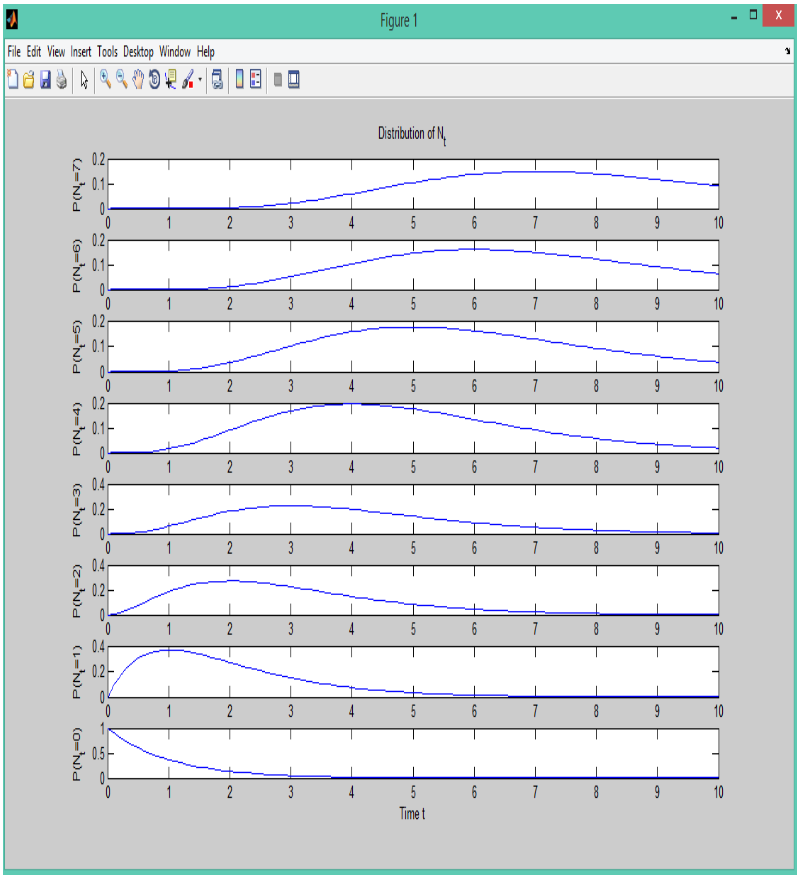


**Figure 12. Probability Distribution - Poisson process**

Displays the probability distribution of the Poisson process as the function of time.

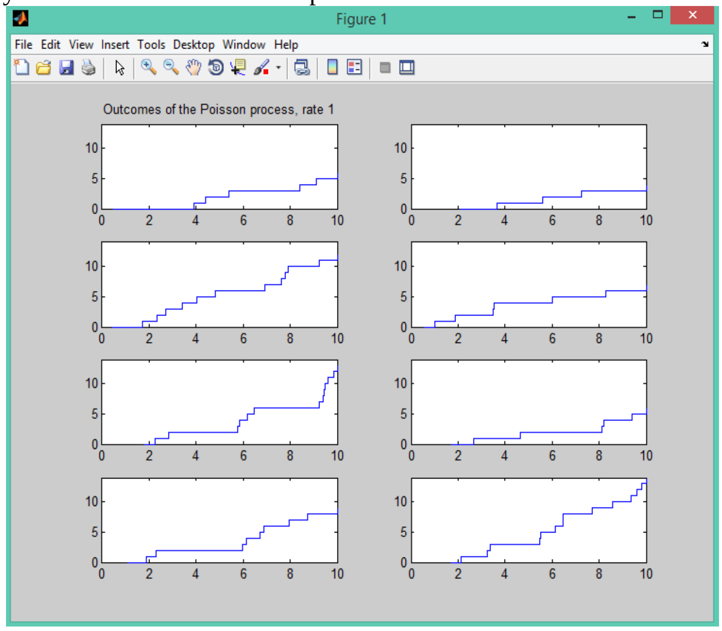


**Figure 13. Poisson Process with many outcomes.**

The above graph depicts the Poisson process which generates the many outcomes.

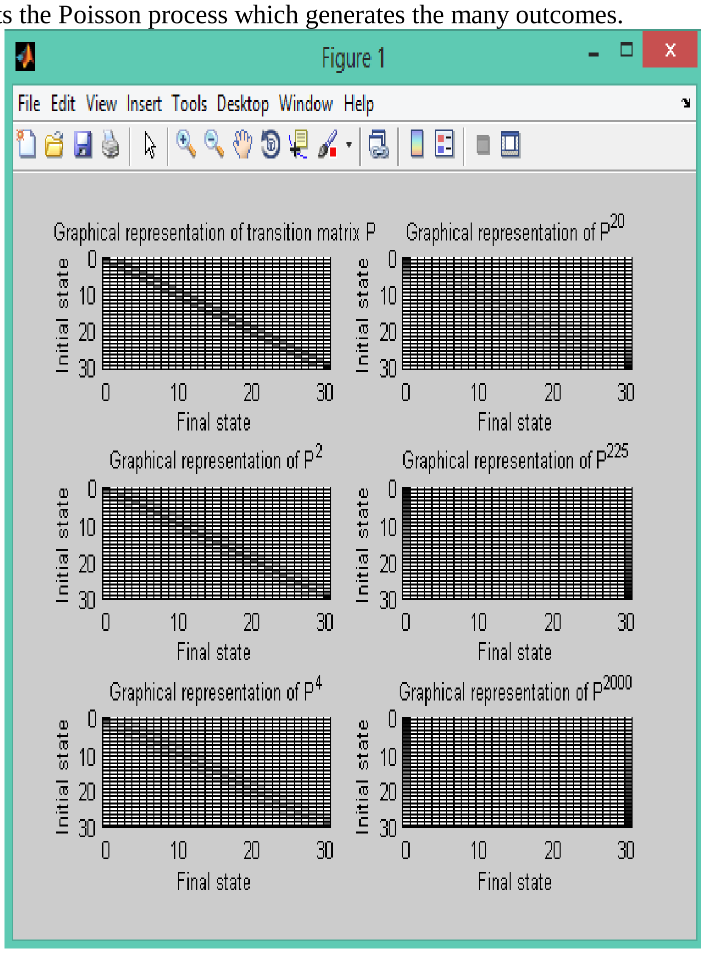


**Figure 14. Graphical representation of transition matrix**

The above graph is the graphical representation of transition matrix with given initial states

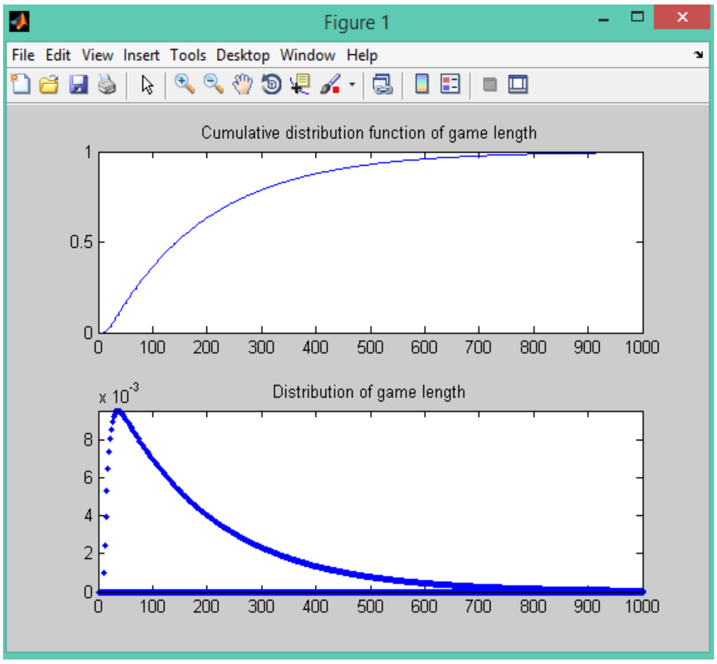


**Figure 15. Distribution of game length.**

The above graph is the distribution of game length using continuous Markov chain.

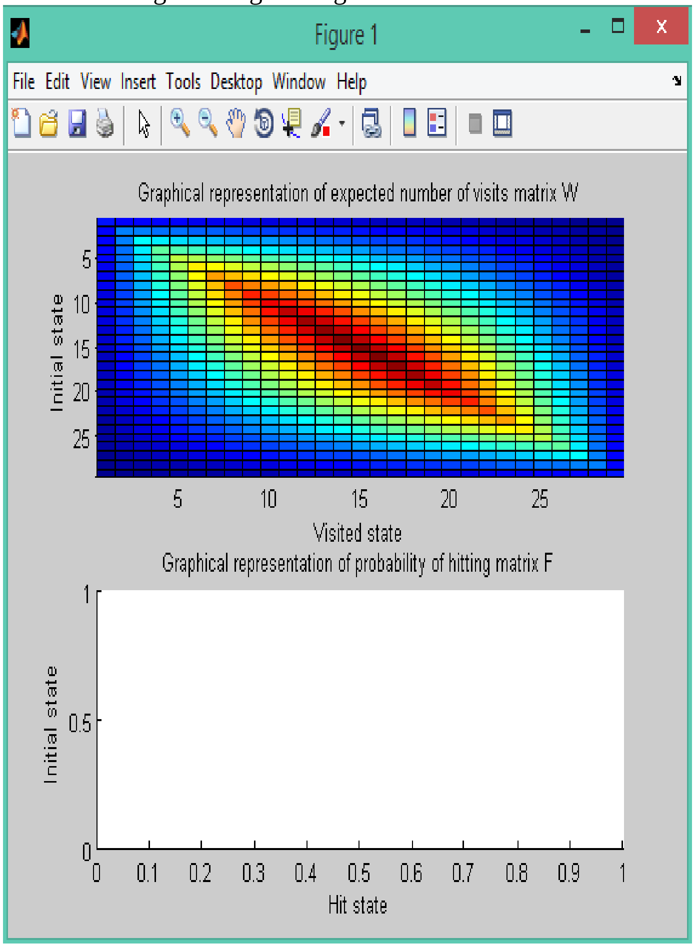


**Figure 16: Graphical representation visits matrix and hitting matrix.**

The above graph depicts the graphical representation of expected number of visits matrix and a graphical representation of probability hitting a matrix.

## III. Conclusions

In this paper, we focused on simulation on Markov chains for the channel allocation of primary users in cognitive radio networks. There are many other research areas which can be extended from our work. This requires a new estimation algorithm that can perform the updating quickly without putting a burden on the processing time. The transmission pattern does vary with time and the parameter estimate should be updated periodically by incorporating new observations into the current estimate. In this paper, we focused on channel allocation for primary user and the spectrum access problem is the future work.

## References

[1]. www.dartmouth.edu/~chance/teaching_aids/books_articles/probability_book/Chapter11.pdf

[2]. I. A. Akbar and W. H. Tranter. Dynamic spectrum allocation in cognitive radio using hidden Markov models: Poisson distributed case. Proceedings of IEEE Southeast Conference, pages 196-201, March 2007.

[3]. Z. Chen, N. Guo, Z. Hu, and R. C. Qiu. Channel state prediction in cognitive radio, Part II: Single-user prediction. Proceedings of IEEE SoutheastCon, pages 50-54, March2011.

[4]. Y. Ephraim and N. Merhav. Hidden Markov processes. IEEE Transactions on InformationTheory, 48(6):1518-1569, June 2002.

[5]. S. Geirhofer, L. Tong, and B. M. Sadler.A measurement-based model for dynamic spectrum access in WLAN channels.Proceedings IEEE Military Communication Conference, pages 1-7, Oct. 2006.

[6]. S. Geirhofer, L. Tong, and B. M. Sadler. Dynamic spectrum access in the time domain: Modeling and exploiting white space. IEEE Communications Magazine, 45:66-72, May 2007.

[7]. C. Ghosh, C. Cordeiro, D. P. Agrawal, and M. B. Rao.Markov chain existence and hidden Markov models in spectrum sensing.IEEE Pervasive Computing and Communications, pages 1-6, March 2009.

[8]. T. Nguyen, B. L. Mark, and Y. Ephraim. Hidden Markov process based dynamic spectrum access for cognitive radio. Conference on Information Sciences and Systems (CISS), pages 1-6, March 2011.

[9]. Z. Sun, G. Bradford, and N. Laneman. Sequence detection algorithms for PHY-layer sensing in dynamic spectrum access networks.IEEE Journal on Selected Topics in Signal Processing, 5(1):97-109, February 2011.

[10]. K. W. Sung, S.-L. Kim, and J. Zander. Temporal spectrum sharing based on primary user activity prediction. IEEE Transactions on Wireless Communications, 9(12):3848-3855, December 2010.

[11]. Q. Zhao, S. Geirhofer, L. Tong, and B. M. Sadler.Optimal dynamic spectrum access via periodic channel sensing.IEEE Wireless Communications and Networking Conference, pages 33-37, March 2007.

[12]. http://www.mes.edu.mn/files/matlab_tutorial.pdf

[13]. http://passat.crhc.illinois.edu/rakeshk/asilomar13_cam.pdf

[14]. http://digilib.gmu.edu/xmlui/bitstream/handle/1920/8288/Nguyen_gmu_0883E_10364.pdf?sequence=1