# Markov Decision Process, A Dynamic System for Enhancaning the TCP Throughput in CR Network

**Pallavi K. Jadhav[1] Prof. Dr. S.V.Sankpal[2]**
[1]ME Student [2]ME Guide
[1,2]Department of Electronics Engineering
[1,2]D. Y. Patil college of Engg. & Tech. Kolhapur, Maharashtra, India

*Abstract*— In cognitive radio network TCP Throughput is one of the measure issues to improve the performance of the CR network. However most research work concentrated on performance improvement of TCP has weaknesses as follows:-The underlying parameters are only considered to increase the TCP Through-put, keeping the transport layer parameter unchanged. Hence to solve the above problem markov decision process base Algorithm is used in this paper. Formal decision analysis has been increasingly used to address complex problems in cognitive radio Network. The proposed work provides a study on the construction and evaluation of Markov decision processes (MDPs), which are powerful analytical tools used for sequential decision making under uncertainty. In this proposed work each CRN users autonomously decides modulation type and power to be transmitted in PHY layer, channels which is to be selected in MAC layer to get best TCP Throughput. Simulation result shows that the network can learn the optimal policy to improve the TCP Throughput in cognitive radio network.

***Key words:*** CRNs; TCP throughput; MDP; Policy; optimal parameters

## I. INTRODUCTION

The spectrum is being diverse and there is lot of challenges in front of the diverse wireless technology and stream traffic services due to the overcrowded unlicensed bands and the Underutilized license bands.

The challenge such as the reconfiguration communication is to be tackled by implementing different technique. The invention of Cognitive radio by J. Mitola III [12-13] which is from software defined radio (SDR) was considered mainly for the big challenge that is the spectrum underutilization. For physical layer radio transmission the cognitive radio is a link base technology using dynamic access system. The advantages of cognitive radio not only limits to the spectrum usages but also for networking diversity above link layer which is used to bridge the integrated re-configurable system.

This scenario is called cognitive radio network for future wireless network which is the same definition of cognitive radio defined by Haykin's[10]

Cognitive radio is a intelligent wireless radio which has a knowledge of the Surrounding environment and which uses the schemes of understanding the behavior of the environment, Then from this environment learning the behavior and according to the environment statistical variations adapting according to the incoming RF stimuli. By changing the operating parameters the adaptation is done such as the modulation, transmitted power and carrier frequency in real time provided, there must be highly reliable communication when needed and also the radio spectrum must be used efficiently. For this parameter configuration a Markov decision process based decision technique algorithm is used in this paper.

If the system transits to the next state, depending on the current state only than such system is called Markovian system, in that if the system arrives at the same state twice, the behavior of the system is same. Hence there is no need of the memory uses for the agent operating in Markovian system. It is sufficient to observe the current state of the system in order to predict the system's future behavior. In a controlled Markovian system the agent influences the environment through its actions, yet the effect of an action depends solely on the current state. To choose the next action optimally the agent needs to only consider the current world state. This decision technique will efficiently bring change in underlying physical parameter according to the adaption with the environment and also calculate optimal parameter thereby increasing the throughput of TCP.

## II. RELATED WORK

Reinforcement learning is a most closely related work (Hutter, 2009; Nguyen et al., 2011, 2012; Daswani et al., 2012). The agent is defined in the same manner as that of Q-Learning but with a model-based cost function. The cost becomes a penalized likelihood of observation as the state sequence can be useful to recover the observation sequence since the maps used in practice are injective. Comparing with the model base function, the model free criteria are much discriminative as only the expected reward is concerned under optimal policy.

The mapping from observation to histories is used in the internal Policy State Gradient Method by Aberdeen and Jonathan (2002) (in their case finite state controllers (FSC)). In ISPG to find the best policy the FSC is used first to parameterize the policy space and then the gradient ascent algorithms are used.

## III. PROPOSED WORK

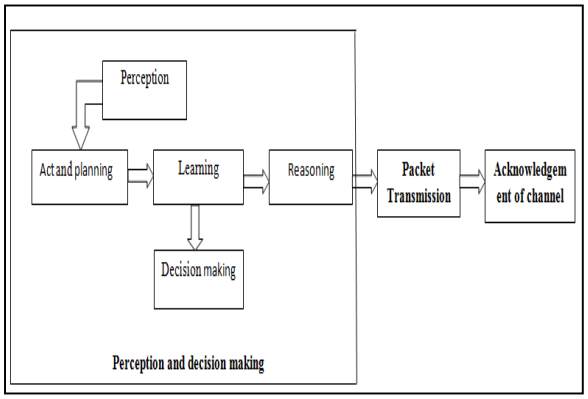


Fig. 1: Block schematic of proposed system

The proposed work consists of three stages, first is the perception and decision making stage than the packet

 

Transmission stage and the last stage is Acknowledgement stage.

*A. Decision Making and Perception Stage:*

Reinforcement learning is a machine learning that specifies how an agent takes an action in an environment and gets a cumulative reward. Reinforcement learning is mapping a situation to a action so that the numerical reward signal will be maximum. The Action to be taken is not told to the learner as in machine learning, but instead must discover which actions yield the most reward by trying them. In the worst condition the action can affect not only the immediate reward but also the situation and all subsequent reward. For example consider the dog is taught to take a ball. The positive reward is when the dog fetches the ball and he is awarded with the cookies and negative award with a scolding when he does not. Hence the dog learns the best to fetch the ball as many times he wants cookies. As many times he fetches the ball he will get knowledge of fetching the ball by experience and the drive to do so was the positive reward.

Reinforcement Learning is a machine learning technique whereby an agent interacts with an environment in the hope of achieving a goal. This interaction with the hope of the agent occurs on a continual basis being able to learn to function in an optimal fashion within that environment. The agent interact with the environment is through the series of actions that can be performed. Hence such action can have positive reward or negative reward over time that is used to determine how best to work in the current environment. At each time, an agent can be in a particular state, with the ability to choose an action based on what it has learned in previous iterations. Another thing that makes the Q-Learning algorithm suitable for this type of problem is because that Q-Learning will converge with a probability of 1 as long as each state action pair is visited infinitely as the learning rate approaches zero. The electromagnetic spectrum environment that the agent is working in is very unpredictable, making it suitable to use an off-policy RL algorithm such as Q-Learning so as to allow a period of random exploration before following the target policy of the agent. The policy used in selecting which action to take is dependent on the type of policy used. The simplest example is to select the action with the greatest reward for that state, although this may not always lead to an optimal solution as it would lead to a totally greedy policy that would not explore parts of the state space that would not appear to be advantageous but could lead to an optimal solution in the future. The discount factor decides how important future rewards are for the agent.

The overall goal is to find an optimal policy that maps each state to an action an agent should take in those states [S].

*1) Mdp Algorithm:*

This section will give the formal definition of MDP algorithm and the description of value iteration. Then we will describe the Q-learning MDP Algorithm.

a) Description:

1) s Є S A finite state space.
2) a Є A a finite set of action.
3) T(s,a,s') Transition function
4) R(s,a) a reward function

Where the transition function specify the probability of taking an action in state s and reaching in state s' and the reward function specifies the reward the agent will receive after performing action in state s and transition in state s'.

MDP framework assumes that the agent has full knowledge of environment and treats time and set S and action A as discrete. For reinforcement learning algorithm, the MDP does not have to be known. The Markov property says that the state of the environment and the reward the agent receive at time t+1 is stochastically determined by the state of the agent at time t and the action the agent takes. This is the first order Markov process.

$$P(st, rt|s0, a0, \ldots, St-1, At-1) = P(st, rt| St-1, At-1) \quad (1)$$

Long term reward is maximized by the task of agent. A mapping is required from states to actions as the problem is stochastic. We call such a mapping a policy and denote it as $\pi(s)$.The long term reward intake is maximized by the optimal policy $\pi^*$. Certain value is assigned to the agent to compute the optimal policy for being in a state or performing some action in a state.

*2) Value Functions:*

The return Rt of a state is defined as the cumulative reward the agent can expect to receive after reaching the given state at time step t. The sum of all reward the agent received is written as Rt for each time step mathematically weighted by a discount factor γ, where $0 < \gamma < 1$:

$$R_t=r_t+1, Y_{rt}+2, \gamma 2rt+3+\ldots=\sum_0^\alpha \Upsilon krt+k+1\ldots.(2)$$

There are two purposes for introducing a discount factor (1) it models the preference of the agent to immediate rewards as opposed to those received in the future, and (2) ensures the infinite sum is finite as long as $\gamma < 1$ and the rewards are bounded. When the discount factor is set close to 1, the agent will value future rewards greatly, whereas one close to 0 will make the agent focus on immediate rewards and value the future less. The expected discount cumulative reward is defined as the value of state s under policy π and is given by

$$(V)^\pi=E\left[\sum_{k=0}^{\infty} \gamma^k r_t + k+1|s_t=s\right]\ldots\ldots\ldots(3)$$

In most situations it is desired to have knowledge of the value of an action in a certain state, we call this the Q-value, with Q(s, a) providing the value of taking a in s, it is defined as:

$$Q^\pi(s.a) =E\left[\sum_{k=0}^{\infty} \gamma^k r_t + k+1|s_t=s,a_t=a\right]$$
$$\ldots\ldots\ldots\ldots\ldots..(4)$$

Assuming the values of all successor states s′ are known to the agent, "Eq. (4)" can be rewritten as the reward the agent receives plus the discounted value of s′, weighted by the probability of ending in s′, after taking action a in s:

$$Q^\pi\ (s,a) = \sum_{s'} T(s,a,s')[R(s,a) + \gamma V\pi(s')]$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots(5)$$

This formula is a form of the Bellman equation named after Richard Bellman, who introduced it in 1957 [3]. With this function, we can iteratively update the value of all states, until it reaches a convergence criterion, resulting in an optimal state-value function V*(s), from which we can derive an optimal state action-value function Q*(s, a). Knowing the value of all states, the agent can select the action with the highest utility in every state, which will lead to an optimal policy. Value iteration is an algorithm that uses this concept.

*3) Value Iteration:*

Value iteration is a (truncated) dynamic programming algorithm for computing optimal value functions and

 

provides an exact solution for solving MDPs. The main idea behind this method is to compute the value of all s 2 S iteratively, and to truncate the algorithm as soon as the difference in value of a state between two iterations: Δ= maxs ∈S |Vi(s)− Vi−1(s)| drops below a threshold, where Δ is typically referred to as the Bellman residual. To approximate the value of a state, value iteration uses the Bellman equation in "Eq. (5)" as an update rule.

Formally, the algorithm would need an infinite number of sweeps through the state space to converge to an optimal value function, but the optimal value can be approximated by aborting the algorithm if Δ is sufficiently small. A major drawback is that each iteration requires updating the value of every s 2 S, resulting in a computational complexity of O(|A||S|2) per iteration. This is time consuming for problems with a large state space. Once the algorithm is finished, the agent can use the values of state action pairs to select the action with the best expected outcome:

$$\pi^* (s) = \arg\max_{a \in A} Q* (s, a) \ldots\ldots(6)$$

*B. Packet Transmission Stage:*

Packet transmission in cognitive radio network is carried out through transport control protocol (tcp). TCP has been designed under the assumption that packet losses are caused almost exclusively by network congestion, so TCP packet losses invoke congestion avoidance mechanisms [1, 2] incorporating rate reduction and multiplicative increase of their transmission timeout. In a high and correlated radio link involved TCP connection, misinterpretation of packet losses over radio links as congestion losses leads to significant throughput degradation.

For flow control TCP uses a sliding window protocol such that the window size can vary with respect to time. The minimum of the congestion window and advertised window can be transmitted by sender. For flow control at the sender the congestion window is imposed and for the receiver flow control the advertised window is imposed. The congestion window is related with the network congestion perceived by sender assessment and the advertised window is the amount of buffer space for connection at the receiver.

*C. Tcp Throughput Model:*

The channel is shared by the CR users using the CSMA protocol for our assumption. Let Xln be the channel allocation strategy of nth slot, then Xln satisfies the following relationship as Xln={x∈{0,1}C , x•Yn=0,That is, each CR user can choose a set of free channels from its action space, with bandwidth of Band(i) (1≤i≤C) for each channel. Based on channel allocation, we assume each user chooses frame size from set LF= (Lfr1, Lfr2,…, LfrK), then the frame error rate can be depicted by bit error rate BERln and frame size LF.

$$F^n_{e\ (c,l)=1-(1-BER^n(c,l)^{Lfr}} \ldots\ldots(7)$$

Where Lfr∈LF represents frame size of l. Let Lfrh be the frame header size and Ltcp be the TCP packet size. so each packet can be divided into Nfr=ceil(Ltcp/(Lfr+Lfrh)) frames (ceil(x) rounds x to the nearest integer). By ARQ protocol and setting maximum retransmission number Re in TCP layer, a packet can be successfully received under the condition of all frames being successfully received by the destination node. Hence the packet error rate can be expressed by "Eq.(9)".

$$P^n_e(c,l) = 1(1 - F^n_e(c,l)^{Re+1})^{Nfr} \ldots..(8)$$

Simultaneously, TCP sliding window protocol is employed for flow control and incidental recognition mechanism is adopted to ensure transmission reliability. At the destination node, by assuming that the maximum TCP congestion window length is cwnd, the number of successfully received packets per unit time, i.e. TCP throughput, can be derived from [8][9].

$$Th^n(c,l) = \min(cwd|Tr, Thl^n(c,l) \ldots..(10)$$

$$Th^n(c,l) = \frac{1}{Tr.\sqrt{2bper^n_e(c,l)3} + T_0.\ \min(1,3\sqrt{3bper^n_e(c,l)/8per^n_e(c,l)(1+32per^n_e(c,l)^2)}} \ldots\ldots\ldots\ldots(11)$$

*D. Stimulation Result:*

Under NS2 platform, we assume 70 CR users are randomly distributed in a square area of 1000m× 1000m. Let channels be independent of each other. Each channel occupancy probability is given by *pu* for the registered users. The TCP packet length $L_{tcp}$ is set to be 1500 bytes and the maximum number of retransmission *Nre* is 5. The CR users node range is set to 250m. The maximum congestion window *cwnd* is given by 6000 bytes and initialize timeout *T0* is 5*s*. The carrier power is set to 10W. The modulation scheme is set to BPSK, and the flatgrid topology of 1000*1000 area is selected. The ARQ protocol is selected in MAC layer and the maximum frame retransmission $N_{fr}$ is 10, of which header length $L_{frh}$ is set to be 20 bits. The ACK frame length $L_{ack}$ is 24 bits and the bandwidth is assumed to be 1MHz. This paper assumes that each CR user can either be a sender or a receiver in a certain slot, while all of them are working abidingly.

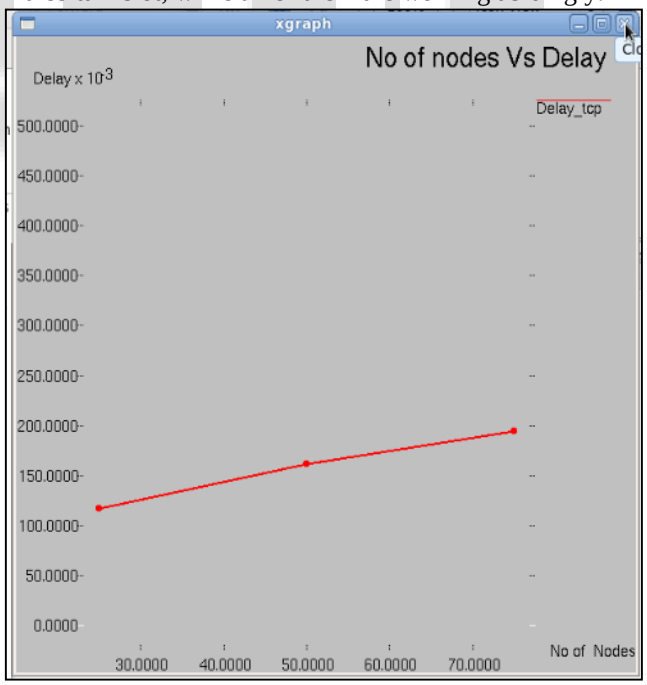


Fig. 2: Graph of no. of nodes vs dealy

 

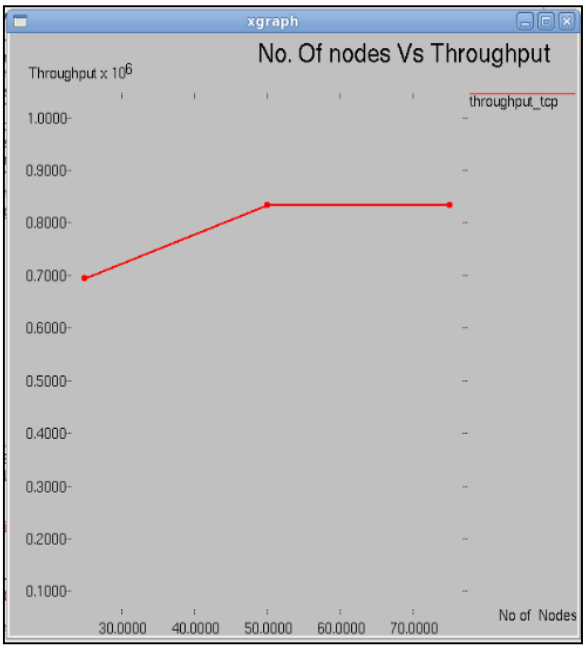


Fig. 3: Graph of no. of nodes vs throughput

End to end TCP performance is one of the main criteria to measure the network performance in distributed network, but most previous studies are based on complete knowledge, without taking into account the perception error. This paper presents a MDP based optimal parameters configuration scheme in CRN. It optimally selects the modulation type and transmitted power in the physical layer and employs Q-value iteration to search the optimal parameters to maximize TCP throughput. Due to the optimal parameter the losses will be reduce and hence it will ultimately enhance the performance of TCP.

## REFERENCES

[1] Jacobson, "Congestion avoidance and control," Computer Communication Review, vol. 18, no. 4, pp. 314–329,1988.
[2] P. Karn and C. Partridge, "Improving round-trip time estimates in reliable transport protocols,"
[3] S. Hykin, "Cognitive Radio: brain-empowered wireless communications," IEEE J.Sel. Areas Commun, Vol. 23, no. 2, pp. 201-220, 2005.
[4] C. S. R. Murthy and B. S. Manoj. Ad Hoc WirelessNetworks: Architecture and Protocols. Prentice Hall, 2004.
[5] Akyildiz, I. F., Wang, X., and Wang, W. 2005. Wirelessmesh networks: a survey. Comput. Netw. ISDN Syst. 47, 4(Mar. 2005), 445-487. DOI=http://dx.doi.org/10.1016/j.comnet.2004.12.001
[6] F. M. Aduljalil and S. K. Bodhe. A survey of integrating IP mobility protocols and mobile ad hoc networks. IEEE Communications Surveys & Tutorials, vol.9, no.1, pp.14-30,First Quarter 2007.
[7] Ojanpera, T. 2006. Convergence Transforms Internet. Wirel.Pers. Commun. 37, 3-4 (May. 2006), 167-185. DOI=http://dx.doi.org/10.1007/s11277-006-9072-3
[8] W. Stevens, TCP/IP Illustrated Volume I. Addison-Wesley, 1994.
[9] X. Gao, G. Wu and T. Miki. End-to-end QoS provisioning in mobile heterogeneous networks. IEEE Wireless Communications, vol.11, no.3, pp. 24-34, June 2004.
[10] S. Haykin. Cognitive radio: brain-empowered wireless communications. IEEE Journal on Selected Areas in Communications, vol.23, no.2, pp. 201-220, Feb. 2005.
[11] J. Mitola III, G. Q. Maguire. Cognitive Radio: Making Software Radios More Personal. Mitola, J., III; Maguire, G.Q., Jr., "Cognitive radio: making software radios more personal," IEEE Personal Communications, vol.6, no.4, pp.13-18, Aug 1999
[12] J. Mitola III. Cognitive Radio Architecture. Wiley, 2006.

 