Total No. of Pages : 02

Roll No. ☐☐☐☐☐☐☐☐☐☐☐☐

Total No. of Questions : 08

B.Tech. (CSE) PIT (Sem.–6)

ARTIFICIAL INTELLIGENCE

Subject Code : BTCS-602-18

M.Code : 79250

Date of Examination : 28-07-21

Time : 2 Hrs. Max. Marks : 60

INSTRUCTIONS TO CANDIDATES :

1. Attempt any FIVE question(s), each question carries 12 marks.

1. What kinds of problems that humans find difficult, that you think computers are particularly well suited to solve? Are there any such problems that you know of that computers cannot currently solve but which you believe computers will one day be able to solve? What advances in technology or understanding are necessary before those problems can be solved?

2. Consider a state space where the start state is number 1 and each state k has two successors: numbers 2k and 2k + 1.

   a) Draw the portion of the state space for states 1 to 15.

   b) Suppose the goal state is 11. List the order in which nodes will be visited for breadth first search.

3. Suppose you are given a bag containing 'n' unbiased coins. You are told that n - 1 of these coins are normal, with heads on one side and tails on the other, whereas one coin is a fake, with heads on both sides.

   a) Suppose you reach into the bag, pick out a coin uniformly at random, flip it, and get a head. What is the (conditional) probability that the coin you chose is the fake coin?

   b) Suppose you continue flipping the coin for a total of k times after picking it and see k heads. Now what is the conditional probability that you picked the fake coin?

4. Most intelligent systems have some degree of uncertainty associated with them. Uncertainty may occur because of the problems with the data. Briefly discuss the reasons for uncertainty in Data. How this uncertainty can be handled?

5. What is Bayesian-Network? Briefly discuss how a Bayesian Network is constructed and how inference is accomplished in a Bayesian Network.

What is Reinforcement Learning? Briefly discuss active and passive reinforcement learning? List various practical applications of Reinforcement Learning.

What are the main components of a Markov decision process? Briefly discuss value iteration, for calculating an optimal policy.

Question: Give the initial state, goal test, successor function, and cost function for each of the following. Choose a formulation that is precise enough to be implemented.

a. You have to color a planar map using only four colors, in such a way that no two adjacent regions have the same color.

b. A 3-foot-tall monkey is in a room where some bananas are suspended from the 8-foot ceiling. He would like to get the bananas. The room contains two stackable, movable, climbable 3- foot-high crates.

Any student found attempting answer sheet from any other person(s), u
inating material or involved in any wrong activity reported by evaluator sha
d under UMC provisions

ARTIFICIAL INTELLIGENCE
BTCS-602-18

1. **What kind of problems that humans find difficult, that you think computers are particularly well fitted to solve? Are there any such problems that you know of that computers can not currently solve but which you believe computers will one day be able to solve? What advances in tech. or understanding are necessary before those problems can be solved?**

Ans

Speech Recognition: It is also known as automatic speech recognition (ASR), computer speech recognition, or speech-to-text, and it is a capability which uses natural language processing (NLP) to process human speech into a written format. Many mobile devices incorporate speech recognition into their systems to conduct voice search—e.g. Siri—or provide more accessibility around texting.

Customer Service: Online chatbots are replacing human agents along the customer journey. They answer frequently asked questions (FAQs) around topics, like shipping, or provide personalized advice, cross-selling products or suggesting sizes for users, changing the way we think about customer engagement across websites and social media platforms. Examples include messaging bots on e-commerce sites with virtual agents, messaging apps, such as Slack and Facebook Messenger, and tasks usually done by virtual assistants and voice assistants.

Computer Vision: This AI technology enables computers and systems to derive meaningful information from digital images, videos and other visual inputs, and based on those inputs, it can take action. This ability to provide recommendations distinguishes it from image recognition tasks. Powered by convolutional neural networks, computer vision has applications within photo tagging in social media, radiology imaging in healthcare, and self-driving cars within the automotive industry.

Recommendation Engines: Using past consumption behavior data, AI algorithms can help to discover data trends that can be used to develop more effective cross-selling strategies. This is used to make relevant add-on recommendations to customers during the checkout process for online retailers.

Automated stock trading: Designed to optimize stock portfolios, AI-driven high-frequency trading platforms make thousands or even millions of trades per day without human intervention.

A possible solution for an NPC problem can be checked for correctness very quickly, but finding a good solution takes a very long time. That is because the length of time required to find a solution grows exponentially with the size of the problem, at least as far as we know. For example, with 266 binary variables, solving the problem will require checking more solutions than there are particles in the universe.

This problem points the way toward a scientific test for intelligence that will give us insight into its nature. For example, suppose we identify problems that humans can routinely solve easily but are in the NPC class for computers. We can then pick a problem size that will be impossible for a computer to solve in the lifetime of the universe, even with a universe's worth of computational power. If at the same time a human can still solve the problem, there are a few possible conclusions we can draw:

Something special about the particular problem makes it easy for a computer to solve. While, in general, NPC problems are hard for computers to solve, certain sub groups happen to be easy. We

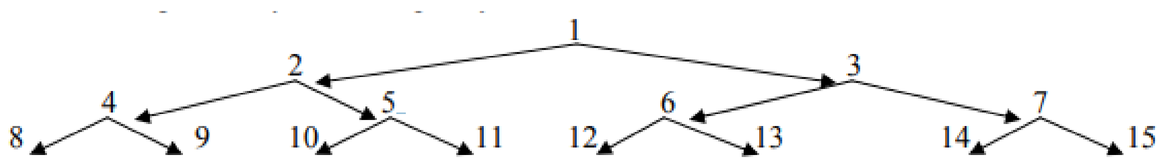
want to avoid confusing the issue by accidentally picking a problem in that subgroup for the human to solve.
The human mind can access the special algorithm that can solve NPC problems.
The human mind operates on a non-physical plane.

The list of new technologies grows every day. Robots, Augmented Reality, algorithms, and machine-to-machine communications help people with a range of different tasks.
(1) These technologies are broad-based in their scope and significant in their ability to transform existing businesses and personal lives. They have the potential to ease people's lives and improve their personal and business dealings.
(2) Technology is becoming much more sophisticated and this is having a substantial impact on the workforce.
(3) These developments can improve the speed, quality, and cost of goods and services, but they also displace large numbers of workers. This possibility challenges the traditional benefits model of tying health care and retirement savings to jobs.

2. Consider a state space where the start state is the number 1 and each state k has two successors: numbers 2k and 2k+1.

a. Draw the portion of the state space for states 1 to 15.

Ans a



b. Suppose the goal state is 11. List the order in which nodes will be visited for breadth-first search.

Ans b Breadth-first: 1 2 3 4 5 6 7 8 9 10 11

3. Suppose you are given a bag containing n unbiased coins. You are told that n-1 of these coins are normal, with heads on one side and tails on the other, whereas one coin is a fake, with heads on both sides. Show your work for the questions below.

a. Suppose you reach into the bag, pick out a coin uniformly at random, flip it, and get a head. What is the conditional probability that the coin you chose is the fake coin?

Ans P(fake | heads) = P(heads | fake)
P(fake) / P(heads) = P(heads | fake) P(fake) / [P(heads | fake) P(fake) + P(heads | −fake)
P(−fake)] = 1 * (1/n) / [1 * (1/n) + 0.5 * (n−1) / n] = 2 / (n+1)

b. Suppose you continue flipping the coin for a total of k times after picking it and see k heads. Now what is the conditional probability that you picked the fake coin?

Ans P(fake | k_heads) = P(k_heads | fake)
P(fake) / P(k_heads) = P(k_heads | fake) P(fake) / [P(k_heads | fake) P(fake) + P(k_heads | −fake) P(−fake)] = 1 * (1/n) / [1 * (1/n) + 2−k * (n−1) / n] = 2k / (2k +n−1)

4. Most intelligent systems have some degree of uncertainty associated with them. Uncertainty may occur because of the problems with the data. Briefly discuss the reasons for uncertainty in data. How this uncertainty can be handled?

Ans In Artificial Intelligence, an agent faces uncertainty in decision making when it tries to perceive the environment for information. Because of this, the agent gets wrong or incomplete data which can affect the results drawn by the agent. This uncertainty is faced by the agents due to the following reasons:

Partially observable environment: The entire environment is not always in reach of the agent. There are some parts of the environment which are out of the reach of the agent and hence they are left unobserved. So, the decisions that the agent makes do not include the information from these areas and hence, the result drawn may vary from the actual case.

Dynamic Environment: As we all know that the environment is dynamic, i.e. there are always some changes that keep taking place in the environment. So, the decision or calculations made at any instant may not be the same after some time due to the changes that have occurred in the surroundings by that time. So, if the observations made at any instance are considered later, then there can be an ambiguity in the decision making.

Incomplete knowledge of the agent: If the agent has incomplete knowledge or insufficient knowledge about anything, then it cannot produce correct results because the agent itself does not know about the situation and the way in which the situation is to be handled.

Inaccessible areas in the environment: There are areas in the environment which are observable, but not in reach of the agent to access. In such situations. The observation made is correct, but the as an agent cannot act on these parts of the environment, these parts will remain unchanged by the actions of the agent. This will not affect the current decision but can affect the estimations made by the agent in the future.

Uncertainty is handled in Artificial Intelligence as follows:

•Fuzzy Logic

–Logic that extends traditional 2-valued logic to be a continuous logic (values from 0 to 1)

•while this early on was developed to handle natural language ambiguities such as “you are *very* Tall” it instead is more successfully applied to device controllers

•Probabilistic Reasoning

–Using probabilities as part of the data and using Bayes theorem or variants to reason over what is most likely

•Hidden Markov Models

–A variant of probabilistic reasoning where internal states are not observable (so they are called hidden)

•Certainty Factors and Qualitative Fuzzy Logics

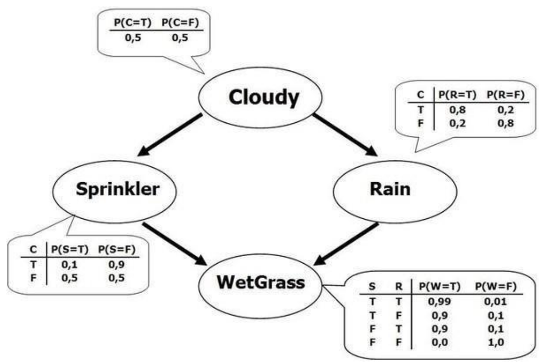
–More ad hoc approaches (non formal) that might be more flexible or at least more human-like

•Neural Networks

–We will skip these in this lecture as we want to talk about NNs more with respect to learning

5: What is Bayesian Network? Briefly discuss how a Bayesian Network is constructed and how inference is accomplished in a Bayesian Network.

Ans A Bayesian network is a directed acyclic graph in which each edge corresponds to a conditional dependency, and each node corresponds to a unique random variable. Formally, if an edge (A, B) exists in the graph connecting random variables A and B, it means that P(B|A) is a **factor** in the joint probability distribution, so we must know P(B|A) for all values of B and A in order to conduct inference. In the below example, since Rain has an edge going into WetGrass, it means that P(WetGrass|Rain) will be a factor, whose probability values are specified next to the WetGrass node in a conditional probability table.



Bayesian networks satisfy the **local Markov property**, which states that a node is conditionally independent of its non-descendants given its parents. In the above example, this means that P(Sprinkler|Cloudy, Rain) = P(Sprinkler|Cloudy) since Sprinkler is conditionally independent of its non-descendant, Rain, given Cloudy. This property allows us to simplify the joint distribution, obtained in the previous section using the chain rule, to a smaller form. After simplification, the joint distribution for a Bayesian network is equal to the product of P(node|parents(node)) for all nodes, stated below:

In larger networks, this property allows us to greatly reduce

$$P(X_1,\ldots,X_n) = \prod_{i=1}^{n} P\left(X_i \mid X_1,\ldots,X_{i-1}\right) = \prod_{i=1}^{n} P\left(X_i \mid Parents(X_i)\right)$$

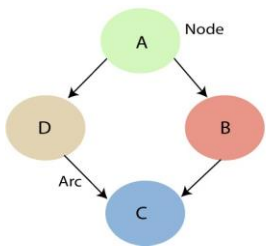
The amount of required computation, since generally, most nodes will have few parents relative tothe overall size of the network.

Bayesian Network can be used for building models from data and experts opinions, and it consists of two parts:

- Directed Acyclic Graph
- Table of conditional probabilities.

The generalized form of Bayesian network that represents and solve decision problems under uncertain knowledge is known as an **Influence diagram**.

A Bayesian network graph is made up of nodes and Arcs (directed links), where:



- Each **node** corresponds to the random variables, and a variable can be **continuous** or discrete.
- Arc or directed arrows represent the causal relationship or conditional probabilities between random variables. These directed links or arrows connect the pair of nodes in the graph.
  These links represent that one node directly influence the other node, and if there is no directed link that means that nodes are independent with each other
  - In the above diagram, A, B, C, and D are random variables represented by the nodes of the network graph.
  - If we are considering node B, which is connected with node A by a directed arrow, then node A is called the parent of Node B.
  - Node C is independent of node A.

6. What is Reinforcement Learning? Briefly discuss Active and Passive reinforcement learning? List various practical applications of reinforcement learning.

Ans Reinforcement Learning is a feedback-based Machine learning technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or penalty.

In Reinforcement Learning, the agent learns automatically using feedbacks without any labeled data, unlike supervised learning.

Since there is no labeled data, so the agent is bound to learn by its experience only.

RL solves a specific type of problem where decision making is sequential, and the goal is long-term, such as game-playing, robotics, etc.

The agent interacts with the environment and explores it by itself. The primary goal of an agent in reinforcement learning is to improve the performance by getting the maximum positive rewards.

The agent learns with the process of hit and trial, and based on the experience, it learns to perform the task in a better way. Hence, we can say that "Reinforcement learning is a type of machine learning method where an intelligent agent (computer program) interacts with the environment and learns to act within that."

Active reinforcement learning (ARL) is a variant on reinforcement learning where the agent does not observe the reward unless it chooses to pay a query cost $c > 0$. The central question of ARL is how to quantify the long-term value of reward information.

Passive reinforcement learning is when we want an agent to learn about the utilities of various states under a *fixed* policy. Since the choices for each state are predetermined, passive reinforcement learning is not particularly useful for letting an agent learn how it should behave in an environment, but it's useful for us to learn as one step on the way to active reinforcement learning.

Practical applications of Reinforcement Learning:

1. RL in Marketing: Marketing is all about promoting and then, selling the products or services either of your brand or someone else's. In the process of marketing, finding the right audience which yields larger returns on investment you or your company is making is a challenge in itself. And, it is one of the reasons companies are investing dollars in managing digitally various marketing campaigns. Through real-time bidding supporting well the fundamental capabilities of RL, your and other companies, smaller or larger, can expect: more display ad impressions in Real-time.

2. RL in Broadcast Journalism: Through different types of Reinforcement Learning, attracting likes and views along with tracking the reader's behavior is much simpler. Besides, recommending news that suits the frequently-changing preferences of readers and other online users can possibly be achieved since journalists can now be equipped with an RL-based system that keeps an eye on intuitive news content as well as the headlines. Take a look at other advantages too which Reinforcement Learning is offering to readers all around the world.

3. RL in Healthcare: Healthcare is an important part of our lives and through DTRs (a sequence-based use-case of RL), doctors can discover the treatment type, appropriate doses of drugs, and timings for taking such doses.. Then, they optimally propose treatments that can Diagnose diseases like diabetes, HIV, Cancer, and mental illness too. If required, these DTRs (i.e.Dynamic Treatment Regimes) can reduce or remove the delayed impact of treatments through their multi-objective healthcare optimization solutions.

4. RL in Robotics: Robotics without any doubt facilitates training a robot in such a way that a robot can perform tasks – just like a human being can. But still, there is a bigger challenge the robotics industry is facing today – Robots aren't able to use common sense while making various

Moral, social decisions. Here, a combination of Deep Learning and Reinforcement learning i.e. Deep Reinforcement Learning comes to the rescue to enable the robots with, “Learn How to Learn” model. With this, the robots can now: –manipulate their decisions by grasping well various objects visible to them. Solve complicated tasks which even humans fail to do as robots now know what and how to learn from different levels of abstractions of the types of datasets available to them.

5. RL in Gaming: Gaming is something nowadays without which you, me, or a huge chunk of people can’t live. With games optimization through Reinforcement Learning algorithms, we may expect better performances of our favorite games related to adventure, action, or mystery.

7. What are the main components of a Markov decision process? Briefly discuss value iteration, for calculating an optimal policy.

Ans Reinforcement Learning is defined by a specific type of problem, and all its solutions are classed as Reinforcement Learning algorithms. In the problem, an agent is supposed to decide the best action to select based on his current state. When this step is repeated, the problem is known as a **Markov Decision Process**.

A **Markov Decision Process (MDP)** model contains:

- A set of possible world states S.
- A set of Models.
- A set of possible actions A.
- A real-valued reward function R(s,a).
- A policy the solution of **Markov Decision Process**.

A **State** is a set of tokens that represent every state that the agent can be in.

A **Model** (sometimes called Transition Model) gives an action’s effect in a state. In particular, T(S, a, S’) defines a transition T where being in state S and taking an action ‘a’ takes us to state S’ (S and S’ may be the same). For stochastic actions (noisy, non-deterministic) we also define a probability P(S’|S,a) which represents the probability of reaching a state S’ if action ‘a’ is taken in state S. Note Markov property states that the effects of an action taken in a state depend only on that state and not on the prior history.

An **Action** A is a set of all possible actions. A(s) defines the set of actions that can be taken being in state S.

A **Reward** is a real-valued reward function. R(s) indicates the reward for simply being in the state S. R(S,a) indicates the reward for being in a state S and taking an action ‘a’. R(S,a,S’) indicates the reward for being in a state S, taking an action ‘a’ and ending up in a state S’.

A **Policy** is a solution to the Markov Decision Process. A policy is a mapping from S to a. It indicates the action ‘a’ to be taken while in state S.

Value Iteration for calculating an optimal policy

Value iteration is a method of computing an optimal MDP policy and its value.

Value iteration starts at the "end" and then works backward, refining an estimate of either Q^* or V^*. There is really no end, so it uses an arbitrary end point. Let V_k be the value function assuming there are k stages to go, and let Q_k be the Q-function assuming there are k stages to go. These can be defined recursively. Value iteration starts with an arbitrary function V_0 and uses the following equations to get the functions for k+1 stages to go from the functions for k stages to go:

$Q_{k+1}(s,a) \quad = \sum_{s'} P(s'|s,a) (R(s,a,s')+ \gamma V_k(s'))$ for $k \geq 0$

$V_k(s) = \max_a Q_k(s,a)$ for $k>0$.

It can either save the V[S] array or the Q[S,A] array. Saving the V array results in less storage, but it is more difficult to determine an optimal action, and one more iteration is needed to determine which action results in the greatest value.Value iteration is a method of computing an optimal MDP policy and its value.

Value iteration starts at the "end" and then works backward, refining an estimate of either Q^* or V^*. There is really no end, so it uses an arbitrary end point. Let V_k be the value function assuming there are k stages to go, and let Q_k be the Q-function assuming there are k stages to go. These can be defined recursively. Value iteration starts with an arbitrary function V_0 and uses the following equations to get the functions for k+1 stages to go from the functions for k stages to go:

$Q_{k+1}(s,a) \quad = \sum_{s'} P(s'|s,a) (R(s,a,s')+ \gamma V_k(s'))$ for $k \geq 0$

$V_k(s) = \max_a Q_k(s,a)$ for $k>0$.

It can either save the V[S] array or the Q[S,A] array. Saving the V array results in less storage, but it is more difficult to determine an optimal action, and one more iteration is needed to determine which action results in the greatest value.

8. Give a complete problem formulation for each of the following. Choose a formulation that is precise enough to be implemented.

a. Using only four colors, you have to color a planar map so that no two adjacent regions have the same color.

Ans a. Initial state: No regions colored.

Actions (3rd ed.)/Successors (2nd ed.): Assign a color to an uncolored region.

Transition model (3rd ed.): The previously uncolored region has the assigned color.

Goal test: All regions colored, and no two adjacent regions have the same color.

Cost function: Number of assignments.

b. A 3-foot tall monkey is in a room where some bananas are suspended from the 8-foot ceiling. He would like to get the bananas. The room contains two stackable, movable, climbable 3-foothigh crates.

Ans b. Initial state: As described in the text.

Actions/Transition model/Successors: Hop on crate; Hop off crate; Push crate from one spot to another; Stack one crate on another; Walk from one spot to another; Grab bananas (if standing on crate).

Goal test: Monkey has bananas.

Cost function: Number of actions.