CS7641 ML Lecture Summaries
Chapter 3: Reinforcement Learning

Kyle Nakamura

3.1 Markov Decision Processes

3.1.1 Introduction

The lecture is about the introduction to Markov Decision Processes (MDPs) in the context of reinforcement learning. The instructor begins by discussing the topic of decision making, correcting a hyphen error in the term "decision-making." The instructor mentions that the lecture is the first in a series of discussions on reinforcement learning, which the listener, Michael, is interested in. The instructor plans to provide some background information and quizzes before delving into the topic further.

3.1.2 The World 1 Question

In this lecture, the concept of a grid world is introduced as a simplified representation of the universe used in reinforcement learning. The grid world discussed in this lecture is a three by four grid, with specific states and actions defined. The goal is to navigate from the start state to the goal state while avoiding a red spot. Up, down, left, and right actions are available, but if a boundary is reached, certain actions may result in staying in the same state. Additionally, there is a black space that acts as a wall and cannot be entered. The lecture concludes with a quiz asking for the shortest sequence of actions to reach the goal state.

3.1.3 The World 1 Solution

In this lecture on Markov Decision Processes, the speaker discusses the concept of multiple optimal solutions in decision problems. They give an example in which there are two acceptable answers to a quiz question. The first answer is "right, right, up, up, right," and the second answer is "up, up, right, right, right." Both answers require five steps. The speaker emphasizes that in decision problems, there can be multiple correct solutions, similar to randomized optimization discussions. The lecture then introduces a new element to the problem, adding complexity to the scenario.

3.1.4 The World 2 Question

The lecture discusses the introduction of uncertainty and stochasticity into a world by changing the probabilities of actions. The action executes correctly with a probability of 0.8, while 20% of the time it causes the agent to move at a right angle. The lecture poses a question about the reliability of a given sequence of actions in achieving the goal, taking into account the probabilities and uncertainty. The lecture then proceeds to find the answer to this question.

3.1.5 The World 2 Solution

In this lecture on Markov Decision Processes, the speaker discusses the computation of probabilities in a given sequence of actions. They explain that each action in the sequence has a probability of success, and by multiplying these probabilities together, they can compute the overall probability of the sequence being successful. The speaker also explores the possibility of unintended consequences in the sequence and calculates the probability of the first four actions in the sequence going wrong while the last one goes right. This probability is found to be a very small number, leading to the overall probability of the sequence succeeding. The speaker concludes that no matter which of the two sequences is chosen, they both have the same probability of success. They emphasize the importance of incorporating uncertainties and probabilities in decision-making, and introduce a common framework for capturing these uncertainties.

3.1.6 Markov Decision Processes 1

The framework discussed in this lecture is the Markov Decision Process (MDP), which is commonly used in reinforcement learning. MDP involves a single agent and focuses on decision-making. The lecturer introduces the concept of the Markovian property, which will be explained later. The MDP is used to represent different states in a given world. States represent tokens that represent the current state of the agent. In the example discussed, the states correspond to grid positions. The lecturer suggests that there are at least 12 different states in this grid. The states can be represented using coordinates or any arbitrary naming convention. The important point is that states represent something and allow us to determine the current state of the agent.

3.1.7 Markov Decision Processes 2

In this lecture on Markov Decision Processes (MDPs), the professor discusses the components of an MDP model. The model consists of states, actions, and a transition model. States describe the current state of the environment, while actions represent the possible actions that can be taken in each state. The transition model is a function that describes the probabilities of transitioning from one state to another when a specific action is taken. The actions available in a given state determine what the agent or entity can do in that state. The action set can vary depending on the state, and certain actions may not be allowed in certain states. The transition model describes the rules of the game or the physics of the world in the MDP. It takes into account the current state, the action taken, and the resulting state, which can be the same as the current state. The transition model produces probabilities that represent the likelihood of transitioning to a specific state when a particular action is taken. These probabilities must sum up to one when calculated over all possible resulting states. In a deterministic case, where there is no noise, the transition probabilities are either zero or one. However, in a non-deterministic case, where actions may execute with some uncertainty, the transition probabilities can be less than one and can vary depending on the action taken. The professor emphasizes the importance of the transition model as it describes the rules of the game and captures all the knowledge about the dynamics of the environment. The transition model is akin to the physics of the world, although it may not perfectly represent real-world physics. The MDP model, consisting of states, actions, and the transition model, can be seen as a simplified representation of the universe, where states correspond to the positions of atoms and the transition model captures how the universe changes in response to actions.

3.1.8 Markov Decision Processes 3

The Markovian property in Markov Decision Processes (MDPs) states that the transition function, which determines the probability of transitioning to a state given the current state and action, only depends on the current state and not any past states. This property is essential for solving MDPs tractably. Additionally, MDPs assume stationarity, meaning that the rules and physics of the environment do not change over time. Rewards play a crucial role in MDPs, as they provide a scalar value for being in a state, and inform the agent about the usefulness of entering that state. There are different ways to define rewards, including rewards for entering a state, rewards for taking an action in a state, and rewards for transitioning to another state. These definitions are mathematically equivalent and can be combined to form an MDP.

3.1.9 Markov Decision Processes 4

In this lecture, the concept of a policy in Markov Decision Processes (MDPs) is discussed. A policy is defined as a function that takes in a state and returns an action, indicating the action that should be taken in a given state. The optimal policy, denoted as policy star, maximizes the long-term expected reward. In MDPs, we observe sequences of states, actions, and rewards, which is different from being told the correct action to maximize a function. Instead, we need to find the optimal action given a state and the corresponding reward. A policy is a function that maps states to actions, providing guidance on what action to take for any encountered state. It is worth noting that policies do not directly provide a sequence of actions, but rather determine the action to take in a specific state. To compute an optimal policy, the lecture hints at next discussing how to go from defining an MDP to finding a good policy, particularly the optimal policy.

3.1.10 Sequences of Rewards 1

The main assumption being made in this lecture is that of stationarity. The concept of stationarity is illustrated using an example of a grid world game with infinite horizons. It is explained that the optimal policy in this case is to take the long route instead of the short route to maximize rewards. However, if there is a finite time horizon, the policy might change due to the limited time available. The lecture emphasizes that the assumption of stationarity only holds in the context of an infinite horizon and that policies can change even in the same state if the time horizon is finite. The lecturer then introduces the concept of utility of sequences. It is explained that utility refers to the long-term rewards obtained through a sequence of states. The lecture states the assumption of stationarity of preferences, which means that if one sequence of states has higher utility than another, it will remain true even if the initial states are different. This assumption is further elaborated using the concept of adding rewards of sequences. It is argued that adding sequences of rewards follows from the assumption of stationarity and is not an arbitrary choice. The lecture concludes by

stating that the assumption of adding rewards is necessary to maintain the property of stationarity of preferences. This is essential for Markov Decision Processes and helps in evaluating the goodness of states. It is explained that this mathematical property is important to understand and is not just an intuitive concept.

3.1.11 Sequences of Rewards 2

The utility of visiting a sequence of states in a Markov Decision Process is equal to the sum of all the rewards received in those states. This concept is consistent with the grid world example and can be likened to the accumulation of money in a bank account. The lecturer mentions that the mathematical derivation of this concept would take significant time and refers students to read about it. The lecturer introduces the idea that although this approach seems logical, it has limitations, which will be illustrated through a quiz.

3.1.12 Sequences of Rewards 3 Question

The lecturer presents a quiz regarding a sequence of states and rewards in the form of a riverbank scenario. On one side of the bank, there are only plus one rewards, while the other side has both plus one and plus two rewards. The rewards continue indefinitely, with a similar pattern. The lecturer poses the question of whether the listener would prefer to be on the top side or the bottom side of the river bank and invites them to consider their answer before revealing the correct response.

3.1.13 More About Rewards 3 Solution

In this lecture, the instructor and a student discuss the effects of changing rewards in a Markov Decision Process (MDP). The instructor explains that changing the rewards can lead to different optimal strategies in the MDP. The instructor and student discuss specific examples and analyze the optimal strategies for different reward settings. They also discuss the importance of carefully choosing rewards in order to achieve desired behavior in an MDP. The instructor emphasizes that rewards serve as domain knowledge and directly influence the behavior of the agent in the MDP.

3.1.14 Sequences of Rewards 1

The main assumption discussed in this lecture is that of stationarity. It is assumed that there is an infinite time horizon and that the policy remains the same regardless of the time step. However, if there is a finite time horizon, the policy may change depending on the time remaining. The concept of utility of sequences is introduced, which refers to the rewards obtained through a sequence of states. It is assumed that if the utility of one sequence is greater than another, it will continue to hold true in the future. The notion of adding rewards is discussed, as it follows from this assumption. Mathematically, it is shown that adding rewards is necessary to maintain the property of stationary preferences.

3.1.15 Sequences of Rewards 2

The lecturer begins by explaining that the utility of visiting a sequence of states can be represented mathematically as the sum of all the rewards received for visiting those states. This concept aligns with the Grid World example previously discussed. The lecturer mentions that it is possible to derive this formula, but it would be time-consuming to do so. The utility as the sum of rewards is consistent with the idea of accumulating money in a bank account. However, the lecturer also states that this approach has limitations and proceeds to introduce a quiz.

3.1.16 Sequences of Rewards 3 Question

The lecture introduces a quiz scenario involving a riverbank with different rewards on each side. The rewards consist of plus ones and plus twos, with the top side having only plus ones and the bottom side having a combination of plus ones and plus twos. The question posed is whether one would prefer to be on the top side or the bottom side of the riverbank. The lecture implies that there is a correct answer to this question.

3.1.17 Sequences of Rewards 3 Solution

The lecturer discusses two sequences of rewards and asks the student to determine which one is better. The student initially chooses the bottom sequence because it provides occasional higher rewards. However, the lecturer explains that neither sequence is better than the other because the utility of both sequences is infinity. This is because both sequences accumulate rewards and continually provide positive rewards. The lecturer then discusses the existential dilemma of living forever and how, if rewards are always attainable, it ultimately does not matter what actions are taken. The lecturer mentions that the student's intuition of never regretting taking the second path compared to the first is valid, but it is not reflected in the current utility scheme. The lecturer suggests a simple change to the utility scheme to incorporate this intuition.

3.1.18 Sequences of Rewards 4

The lecturer discusses the concept of discounted rewards in Markov Decision Processes. By adding the rewards of future states multiplied by a discount factor called gamma, the total reward can be bounded. This is equivalent to a geometric series and can be represented as the maximum reward divided by (1 - gamma). When gamma is close to 0, the rewards quickly diminish, and when gamma is close to 1, the rewards are amplified. Discounted rewards allow for infinite sequences to be added and still result in a finite value. The lecturer also briefly mentions the singularity, where infinite computation is possible in finite time, as an analogy for the concept of infinite distance in finite time. The lecture ends by introducing a future math-focused segment.

3.1.19 Assumptions

In this lecture, the speaker discusses the assumptions underlying the equation for calculating the sum of gammas (discount factors) multiplied by maximum rewards (R max). They suggest taking out R max and representing the equation as an infinite sequence, which can be denoted as x. By shifting this sequence one step over in time and multiplying it by gamma, the equation can be written as $x = \gamma^0 + \gamma * x$. Solving for x, the speaker subtracts $gamma * x$ from both sides and simplifies to find that $x = \gamma 0/(1 - \gamma)$. They note that multiplying this equation by R max gives the original formula. The speaker concludes that this exercise highlights the simplicity of geometry and prepares the audience for the more complex math to follow.

3.1.20 Policies 1

The lecture begins with a discussion on the optimal policy in Markov Decision Processes (MDPs), which maximizes long-term expected reward. The lecturer introduces the equation for the optimal policy and explains that it is derived by considering the expected sum of discounted rewards at each time step, given a specific policy. The lecturer emphasizes that the policy's utility is important and depends on the sequence of states generated by the policy. The utility of a state accounts for both immediate rewards and future rewards, providing a measure of long-term feedback. The lecturer gives examples to differentiate between immediate rewards and long-term utility, highlighting the importance of considering delayed rewards in decision-making. This notion of delayed rewards is essential for addressing the credit assignment problem. The lecture concludes with the intention to discuss more equations.

3.1.21 Policies 2

The lecture discusses the concept of the optimal policy in Markov Decision Processes (MDPs) and introduces the Bellman Equation as the key equation for solving MDPs and reinforcement learning. The equation calculates the true utility of a state by taking into account the reward received at that state, as well as the discounted future rewards and transition probabilities. The lecturer emphasizes that solving the Bellman Equation allows us to determine the optimal policy, which maximizes the utility. The name "Bellman Equation" is attributed to the equation's inventor, and it encompasses all the components of an MDP, making it a powerful tool for solving MDPs. The lecture concludes by discussing the next steps of solving the equation and its significance in the field of machine learning.

3.1.22 Finding Policies 1

The lecture discusses the challenge of solving Bellman's equation for Markov Decision Processes (MDPs), which involves N equations for N unknowns. The problem is that the equations are nonlinear due to the

presence of the max operator. Despite this nonlinearity, there is an algorithm that allows us to solve the equations iteratively. The algorithm involves starting with arbitrary utilities and updating them based on the utilities of neighboring states. This update is performed iteratively until convergence. The lecture explains the equation for updating utilities and emphasizes that the update relies on the estimates of utilities at the previous time step. The lecture also mentions that the equations are intertwined due to the expectation term, requiring a summation over all possible states.

3.1.23 Finding Policies 2

In this lecture, the instructor discusses the process of finding policies in Markov Decision Processes (MDPs). The goal is to update the utility of a state by considering the utilities of other states and weighting them based on the probability of reaching those states. The chosen action is the one that maximizes expected utility. The instructor explains that the process works because the true value of entering a state propagates through all states until convergence. The instructor also mentions that the discount factor helps in discounting the initial arbitrary utilities. This process is referred to as value iteration and it steadily gets closer to the true value of states. Eventually, the computed utilities can be used to define the optimal policy.

3.1.24 Finding Policies 3 Question

The quiz is to figure out how value iteration would work for a particular state in a grid world. The state is marked with an X. Gamma is equal to one half. The rewards for all states except the two goal states are minus 0.04. The initial utilities for all states are 0, except for the two absorbing states which are 1 and minus 1 respectively. The task is to determine how the utility for the given state will evolve after one step and two steps of value iteration.

3.1.25 Finding Policies 3 Solution

In this lecture on Markov Decision Processes (MDP), the speaker discusses finding policies for solving MDPs. They begin by discussing a specific example and calculating the utility values (U) for different states. The speaker emphasizes that the optimal policy is to take the action that leads to the highest chance of achieving a positive reward. They go on to explain that the values of certain states depend on the values of other states, and demonstrate the calculations involved. The speaker also discusses the concept of policies, which map states to actions, and explains that the goal is to find a policy that is good enough to achieve the desired outcome, rather than seeking absolute convergence in utilities. The speaker suggests that there are alternative approaches to finding policies that may be more efficient in practice.

3.1.26 Finding Policies 4

In this lecture, the focus is on finding policies that are just as effective as using value iteration. The algorithm presented starts with an initial policy and evaluates its utility. The policy is then improved based on the calculated utility. The utility is computed using Doman's equation, where the utility at time t is equal to the true reward plus gamma times the expected utility. Unlike value iteration, the policy iteration algorithm does not involve the max operator and instead uses linear equations, making it easier to solve. The process involves iterating through policies and taking advantage of specific policies to convert the nonlinear equations into linear ones. While this inversion step can be computationally expensive, there are tricks to speed up the process. The algorithm is guaranteed to converge due to the finite number of policies and the fact that it constantly improves.

3.1.27 Wrapping up

In this lecture on Markov Decision Processes (MDPs), the professor and Michael discuss the key concepts covered. They define MDPs as consisting of states, rewards, actions, transitions, and discounts. The discount is debated as either a problem definition or an algorithm parameter. The professor views it as something that can be adjusted. They emphasize that MDPs represent the underlying process of the world, while rewards and discounts represent the task at hand. Policies and value functions (called utilities) are introduced as important concepts. Utilities account for long-term aspects, while rewards focus on moment-to-moment feedback. Discounting is used to assign value to infinite sequences of rewards without resulting in infinitely large sums. Stationarity and the Bellman equation are central to understanding MDPs. The

lecture concludes with a mention of value iteration and policy iteration as methods to solve the Bellman equation, and the distinction between MDPs and reinforcement learning. It is noted that reinforcement learning involves unknown rewards, transitions, and possibly states and actions. The lecture concludes with the assignment for Michael to learn about reinforcement learning for the next session.

3.1.28 Reinforcement Learning

This lecture is about reinforcement learning in the context of Markov decision processes (MDPs). The speaker begins by discussing a reinforcement learning API, which consists of a transition function (T) and a reward function (R) that are used to generate a policy (pi) from an MDP model. This process is called planning. The speaker then introduces a different setup where instead of using a model, the learner receives transitions as input, which include the current state, action, reward, and the resulting state. Using these transitions, the learner learns a policy through reinforcement learning. The speaker acknowledges the question of what makes it reinforcement learning but does not provide a clear answer. They suggest discussing it further on the next slide.

3.1.29 Rat Dinosaurs

In the early days of reinforcement learning, researchers observed that animals, such as rats, learned to associate certain stimuli with rewards. For example, a rat would learn to associate a red light with the presence of cheese and would be more likely to go investigate the area associated with the red light. Computer scientists adopted this concept and called it reinforcement learning, but instead of focusing on strengthening connections, they focused on maximizing rewards. This led to the development of algorithms for solving similar problems. Interestingly, psychologists have also turned to computer scientists to understand how the brain may solve these problems, and they have borrowed the term "reinforcement learning" to describe their own research on reward maximization. So, in a way, computer scientists have had an impact on the field of psychology.

3.1.30 API

In this lecture on Markov Decision Processes (MDPs), the instructor discusses the concept of an applications program interface (API) in relation to planning and learning. They introduce two sub-components, a modeler and a simulator, which connect transitions and generate transitions, respectively. The instructor explains that modeling can be seen as a machine learning problem, where information is mapped into models. They suggest that one approach to learning could be to first perform modeling and simulation to obtain models and the reinforcement function, and then use planning. The lecture ends with the promise to explore different ways of integrating these components.

3.1.31 API Quiz Question

The lecture discusses two approaches to solving the reinforcement learning problem. The first approach involves using a modeler to convert turn transitions into a model, and then using a planner (such as value iteration or policy iteration) to generate a policy based on the model. This approach is referred to as mapping transitions to model to policy. The second approach involves mapping a model through a simulator into transitions, and then using reinforcement learning to convert these transitions into a policy. This approach is referred to as turning a model into a policy, and it incorporates a learner that uses a planner inside. The lecture concludes by asking the audience what they would call these approaches.

3.1.32 API Quiz Solution

In this lecture on Markov Decision Processes, the speaker discusses two approaches to reinforcement learning. The first approach is called model-based reinforcement learning, where a model is built and used for planning. The second approach is model-free reinforcement learning, where a model is not used and the learner interacts with a simulator to generate transitions and derive a policy. The speaker mentions the success of Jerry Tassara's backgammon playing program, which used a simulator to generate transitions and applied reinforcement learning. The speaker also mentions other related works, including Justin Boyin's master's thesis on backgammon with RL and Rob Schapire's work on boosting for his PhD.

3.1.33 Three Approaches to RL

In this lecture, the professor discusses three different approaches to solving reinforcement learning problems: policy search algorithms, value function based approaches, and model based reinforcement learners. Policy search algorithms focus on directly learning the policy, which maps states to actions. However, learning the policy is indirect because the data does not provide information on which action to choose in a given state. On the other hand, value function based approaches target the learning of a function that maps states to values. By observing the values that result from taking actions in different states, this approach allows for more direct learning. The challenge with value function based approaches is turning the learned values into a policy. The professor also mentions model based reinforcement learners, which learn the transition and reward functions to make decisions. Learning the transition and reward functions is relatively direct because it can be treated as a supervised learning problem. However, planning and optimization are required to develop a policy. The professor advocates for focusing on value function based approaches because they strike a balance between direct learning and usage and have been extensively studied.

3.1.34 A New Kind of Value Function

Summary: - A new kind of value function is introduced to make optimization and learning easier in Markov Decision Processes (MDPs). - The value function U is defined for each state in terms of the long-term value of being in that state, which includes the reward for arriving in that state and the discounted reward of the future. - The policy in a state is determined by considering the expected values of all possible actions weighted by their probabilities of reaching a specific state. - The new value function, called the Q function, is defined as the value for arriving in a state, plus the discounted expected value of taking an action and proceeding optimally thereafter. - The Q function allows for comparing the values of different actions without requiring the model to be evaluated.

3.1.35 Value Function Quiz Question

The lecture discusses the concept of the Q function, which encapsulates the necessary information for dealing with utility (U) and policy (pi) in Markov Decision Processes (MDPs) without requiring knowledge of the transition function (T) or reward function (R). The lecturer poses a question to the audience, asking them to rewrite the equations for U and pi using Q instead of references to T and R. The audience is given time to think about the question before providing an answer.

3.1.36 Value Function Quiz Solution

In this lecture, the instructor discusses the relationship between the value function (U) and the action-value function (Q) in a Markov Decision Process (MDP). The value function represents the expected cumulative reward for an agent in a given state, while the action-value function represents the expected cumulative reward for an agent taking a specific action in that state. The instructor explains that the value function can be defined as the maximum action-value over all possible actions in a given state (Max over a of Q(s,a)). Similarly, the policy, which represents the optimal action to take in a given state, can be defined as the action that maximizes the action-value (argmax_a(Q(s,a))). The instructor concludes by introducing the concept of Q-learning, which involves finding the optimal action-value function Q.

3.1.37 Q Learning Question

Q-learning is a type of reinforcement learning algorithm. It involves evaluating the Bellman equations from data. This is the correct answer to the quiz question. The other options - figuring out the best line to wait in, discovering when to come in for your line in a play, and practicing the best bank shot - do not describe Q-learning.

3.1.38 Q Learning Solution

The lecture discusses different examples of cue-learning, which involve figuring out the best line to wait in, practicing a bank shot, and evaluating Bellman equations from data. Q-learning is described as the process of using transitions, or data, to directly produce the solution to the Q equations.

3.1.39 Estimating Q From Transitions

In this lecture, the focus is on estimating the Q function from transitions in Q learning. The Q equation cannot be directly solved because the reward (R) and transition probabilities (T) are unknown. Instead, transitions are observed, which involve starting in a state (S), taking an action (A), landing in a new state (S'), receiving a reward, and determining the next state. The Q function estimate (Q hat) is updated using these transitions. The update is a combination of the immediate reward and the discounted estimated value of the next state. The learning rate (alpha) determines how much the estimate is updated. The Q learning equation combines these elements to calculate the utility of the current state and the next state. The notation alpha arrow is used to represent updating a value towards another value. A learning rate of 0 means no learning, while a learning rate of 1 means complete replacement of the old value. A learning rate between 0 and 1 averages the old and new values.

3.1.40 Learning Incrementally Question

The lecture begins with a quiz introducing the concept of learning rates in Q-learning equations. The learning rates, denoted as alpha sub t, need to satisfy two properties: the sum of learning rates summed up to infinity, but the sum of squares of learning rates, as t goes to infinity, should be less than infinity. One example of a learning rate sequence that satisfies this property is alpha sub t equals one over t. This is because the sum of the values up to t, when summed up over all values of t, acts like the natural logarithm, which goes to infinity as t goes to infinity. However, the sum of squares converges to pie squared over six, a finite value. The question posed is what do you think the learning sequence converges to - does it converge to the expected value of X, converge to the variance of X, converge to infinity, or not converge at all?

3.1.41 Learning Incrementally Solution

The lecture discusses the concept of learning incrementally and the importance of choosing the parameter alpha, which approaches zero over time. The speaker notes that this convergence is necessary for the learning process to be effective. The speaker eliminates two potential answers and decides that the expected value of x (the mean) is the correct choice. The speaker explains that computing the average involves continuously sampling and updating values, and that fluctuations above and below the average eventually cancel out, leading to the expected value of the random variable. The speaker emphasizes that the order of the samples doesn't matter as long as they are independently and identically distributed (iid). The weighted average in Q-learning is also mentioned.

3.1.42 Estimating Q From Transitions Two

The lecture discusses the concept of estimating the Q value in a Markov Decision Process (MDP). The Q value represents the average value obtained by following an optimal policy after taking a particular action. The lecture emphasizes the importance of updating the learning rates over time. The linearity of expectation is used to break down the sum in order to calculate the expected value of the reward. The distribution over the next state is determined by the transition function. The speaker acknowledges that the Q value is a moving target and explains that the Q-learning update rule solves MDPs.

3.1.43 Q Learning Convergence

The Q-learning rule is a remarkable fact in which, regardless of the starting position of Q hat, if it is updated according to the given rule, the estimate q hat S A will converge to Q S A, the actual solution to the Bellman equation. The caveat is that for this convergence to happen, SA must be visited infinitely often, and the algorithm must run for a long time and visit all state-action pairs. The learning rates, next states, and rewards must also be updated and drawn according to certain conditions. Although this convergence takes a long time, it is reassuring that the update rule leads to the optimal solution.

3.1.44 Choosing Actions

Q-learning is a family of reinforcement learning algorithms that vary in the initialization of the Q-hat estimate, the decay of learning rates, and the choice of actions during learning. Choosing actions is important in achieving convergence and utilizing learned knowledge. Randomly choosing actions can visit all states

and actions, but does not effectively use learned knowledge. Always choosing the same action or a suboptimal action also hinders learning. Using the Q-hat estimate to choose actions can learn something, but may not converge to optimal results if the initial estimate favors suboptimal actions. Careful initialization, such as random restarts, can address this issue.

3.1.45 Choosing Actions Two

Random restarts were useful in optimization to overcome local optima, and the same idea may be helpful in this setting. However, there are some problems with random restarts, such as the time it takes to find a good answer. Simulated annealing is a different approach that combines random and informed choices by occasionally taking random downhill steps while mostly taking uphill steps. This approach can be adapted for choosing actions in Markov Decision Processes (MDPs) by occasionally taking random actions while mostly following the estimated best action based on the current state. This allows for exploration of the whole space and learning the true Q values. The probability of taking a random action is determined by epsilon, and as long as epsilon is greater than zero, the MDP is connected and all state-action pairs can be visited infinitely. This approach is the first idea that combines learning and usage of what is learned.

3.1.46 Greedy Exploration

The lecture discusses the concept of Epsilon Greedy Exploration, which is a method of action selection in reinforcement learning. The speaker explains that as the exploration decreases and the exploitation increases over time, the policy followed by the agent becomes more similar to the optimal policy. This trade-off between exploration and exploitation is known as the exploration-exploitation dilemma. The lecture also mentions the exploration-exploitation lemma, which focuses on the choice between exploring and exploiting actions. Different approaches to exploration and exploitation are discussed, with emphasis on the benefits of model-based approaches. It is emphasized that finding a balance between exploration and exploitation is crucial for effective reinforcement learning, as failure to do so can result in a lack of learning or getting stuck in local minima. The lecture concludes by highlighting the role of reinforcement learning in integrating model learning and planning processes, and the importance of information exchange between these two processes.

3.2 Game Theory 3.2.1 Game Theory

3.1.47 What Have We Learned

In this lecture, the speaker summarizes the main insights gained from the topic of reinforcement learning. The speaker emphasizes the significance of being able to solve Markov Decision Processes (MDPs) without knowing the transition and reward functions, highlighting the power of learning in decision-making with delayed rewards. Specifically, the speaker discusses Q-learning, exploration vs exploitation tradeoff, convergence of Q-learning, and the concept of optimism in the face of uncertainty. The speaker mentions two methods for achieving exploration in Q-learning: randomly choosing actions and manipulating the initialization of the Q function to be optimistic. The latter approach, referred to as optimism in the face of uncertainty, allows the algorithm to try new actions that it hasn't explored much and gradually form a realistic understanding of the environment. Additionally, the speaker briefly mentions other topics discussed in the course, such as function approximation, overfitting, policy search, and model-based reinforcement learning, which will be explored in a later lesson. The lecture concludes with a discussion on the next topic: game theory. The speaker explains that game theory is a natural extension of reinforcement learning and mentions its relevance to understanding multi-agent interactions. Overall, the lecture provides an overview of the key concepts and ideas covered in the reinforcement learning section of the course, highlighting the importance of learning in decision-making and introducing the upcoming topic of game theory.

3.2.2 What Is Game Theory

Game theory is described as the mathematics of conflict and involves making optimal choices in conflicts of interest. It is related to reinforcement learning and decision making for single agents, but it also considers the desires and goals of other agents. Game theory allows us to explicitly take into account the goals of all agents involved and can be tied back to reinforcement learning and the Bellman equation. Economics is the field where game theory originated, as it deals with the interactions of many agents with their own

goals. Other fields such as sociology and biology can also benefit from game theory, as they involve interactions between individual agents. Game theory helps us understand what happens when there are multiple agents with conflicting or aligned goals and how to incorporate these goals into decision making. It has become increasingly important in artificial intelligence and machine learning.

3.2.3 A Simple Game 1

In this lecture on game theory, the professor introduces a simple game with two agents, A and B. The game is represented by a game tree, where circles represent states and edges represent actions. A starts in state 1 and can choose to go left or right. If A goes right, she ends up in state 3 and receives a reward of +2. If she goes left, she ends up in state 2. B then gets to make a choice. The game is a two-player zero-sum finite deterministic game of perfect information, where the sum of the rewards is always a constant. Strategies in game theory are similar to policies in reinforcement learning, and they map states to actions for each player. In this game, A has a strategy of going left in state 1 and left in state 4. There are multiple strategies for each player, and the professor quizzes the class on how many other strategies there are for A.

3.2.4 A Simple Game 2 Question

In this lecture on game theory, the speaker presents a quiz to determine the number of different strategies for players A and B in a two-player zero-sum, finite, and deterministic game of perfect information. The speaker clarifies that they are referring to deterministic mappings only. However, they acknowledge the potential usefulness of stochastic strategies. The speaker introduces the concept of pure strategies and emphasizes the importance of purity, specifically in relation to chocolate and bacon. The quiz focuses on determining the number of pure strategies for players A and B.

3.2.5 A Simple Game 2 Solution

The lecture discusses a simple game with two states: state one and state four. In state one, there are two possible choices: left or right. Similarly, in state four, there are also two possible choices: left or right. This can be represented as a 2x2 matrix. The lecturer explains that when considering the strategy, it is important to think about all possible states. Even though going right from state one would eliminate the need for another choice, it is still necessary to consider all states. The lecturer then introduces another variable, B, which complicates the game. With B, the number of reachable strategies depends on the choice made by player A. If player A chooses left, there are three possible choices. If player A chooses right, there is only one possible choice. In this case, the number of reachable strategies is three. Overall, the lecture focuses on understanding the number of strategies in a simple game and emphasizes the importance of considering all possible states and choices.

3.2.6 A Simple Game 3 Question

The lecturer discusses a simple game with two players, A and B. They explain that by considering all possible strategies for A and B, a matrix can be formed, where each cell represents the value of taking a particular strategy for A and a particular strategy for B. The lecturer suggests making it a quiz and asks the students to fill in the numbers in the matrix. They provide an example calculation to demonstrate how to determine the value of the game for player A. They mention that since this is a two-player zero-sum game, if player A gets a value of 7, player B gets a value of -7. The goal is to fill out the entire table.

3.2.7 A Simple Game 3 Solution

In this lecture, the speaker discusses a simple game with two players, A and B, and explains how to determine the optimal strategies for each player. The speaker presents a matrix that captures the payoffs for different combinations of strategies chosen by A and B. The speaker emphasizes that this matrix contains all the information needed to understand the game and that other aspects, such as the rules and the tree representation, are irrelevant. The main goal of the game is to optimize long-term expected rewards. The speaker poses the question of what strategies A and B will choose given the matrix of values. It is noted that A wants to maximize rewards while B wants to minimize A's rewards, as this is a two-player, zero-sum, finite game of perfect information. Through a discussion of the different possibilities, the speaker concludes that the optimal strategy for A is to choose the top row of the matrix and for B to choose the middle column.

This results in a payoff of 7 for A and -3 for B. The speaker explains that this outcome is not a coincidence but the expected result based on the characteristics of the game. Overall, the lecture focuses on the process of determining optimal strategies for a simple game using a matrix representation.

3.2.8 Minimax

In the lecture on Minimax, the concept of worst case counter strategies in game theory is discussed. Both participants, A and B, must consider the worst case counter strategy of the other. The values for A are set up so that A always tries to maximize, while B always tries to minimize A, effectively maximizing itself. This approach is known as Minimax. The lecture also mentions that Minimax is the algorithm used for game search and is applicable in two-player zero-sum games with perfect information. Alpha-beta pruning is a more efficient way of finding the answer but does not change the outcome. The lecture concludes by stating that the value of the game in this case is three, assuming rational play from both A and B.

3.2.9 Fundamental Result

In a two-player zero-sum deterministic game of perfect information, the minimax strategy is equivalent to the maximin strategy. The maximin strategy aims to maximize the minimum outcome, while the minimax strategy aims to minimize the maximum outcome. The order in which players choose their moves does not affect the outcome. Furthermore, there always exists an optimal pure strategy for each player, meaning that the games can be solved by applying either the minimax or maximin strategy. The assumption in game theory is that players are rational and seek to maximize their rewards. The term "optimal" refers to maximizing one's expected reward while assuming that everyone else is doing the same. This theorem holds in the case of a two-player zero-sum deterministic game of perfect information, but may not hold in more complex scenarios. The concept of pure strategies is important because as games become more complicated, impure strategies may need to be considered. The process of analyzing games using game trees and search algorithms in AI is consistent with what is expected in game theory when players have complete information. Transforming a game tree into a matrix can be done, and the same principles apply, although further analysis may be needed to prove this.

3.2.10 Game Tree 1

The lecture discusses a game tree with two players, A and B. Player A makes the initial choice to go left or right, and player B then has a choice to go left or right as well. There are chance nodes represented by square boxes where randomness occurs. If player A goes left, there is a 50% chance of ending up in one node and a 50% chance of ending up in another node. If player A goes right and player B goes left, there is an 80% chance of ending up in one node and a 20% chance of ending up in another node. If player B goes right, the outcome is always in the same node. The lecturer explains that this game has transitioned from a deterministic game of perfect information to a non-deterministic game of perfect information. The stochasticness occurs at the leaves of the tree, where no choices are left for the players. However, the lecturer notes that the tree could continue with more chance nodes and choices. The value of the game is then discussed, and the lecturer suggests using a quiz format to determine the value. The lecture emphasizes the importance of writing down a matrix to calculate the value and identifies the strategies for players A (left and right) and players B (left and right).

3.2.11 Game Tree 2 Question

The quiz question asks for the values to be filled in a matrix in a zero-sum game scenario. It is specified that the values are from A's perspective, while B's values are known implicitly. The question prompts the values to be determined based on the available information about A and B.

3.2.12 Game Tree 2 Solution

In this lecture on game theory, the lecturer and Michael discuss the value of a game and how to calculate the matrix. They determine the values for different scenarios, such as if A goes left and B goes right or if both A and B go right. They highlight the importance of matrices in representing the information of the game and how they can be used to make decisions, regardless of the original game tree structure. They

conclude that A would choose to go right and B would choose to go left in order to obtain a value of -2 for the game.

3.2.13 Von Neumann

This lecture discusses von Neumann's theorem on non-deterministic games of perfect information. The theorem states that in such games, the only thing that matters is the matrix, and the minimax or maximin strategies can be used to determine the value of the game and the optimal policy. Von Neumann, known for von Neumann architectures in computer science, contributed to the basic design of microprocessors. The lecture concludes by mentioning the relaxation of the current topic and the possibility of constructing non-zero sum matrices.

3.2.14 Minipoker

In this lecture, the instructor introduces a new game called Minipoker which falls under the category of two-player games with hidden information. The game involves a set of red and black cards, where red is considered bad for Player A and black is considered good. Player A is dealt a card, and there is an equal probability of it being red or black. If Player A resigns with a red card, they lose 20 cents. If Player A holds the card, Player B can choose to resign and Player A gets 10 cents regardless of the card's color. Alternatively, Player B can demand to see the card. If the card is red, Player A loses 40 cents, and if it is black, Player A gains 30 cents. The game is zero-sum, meaning whatever Player A wins, Player B loses and vice versa. The lecture mentions that there is no point in Player A resigning with a black card since black is always good for them. Overall, Minipoker is a simplified version of poker with hidden information and betting dynamics.

3.2.15 Minipoker Tree Question

The lecture discusses a game called Minipoker and presents a tree diagram representation of the game. Chance nodes are represented by squares, and the lecture explains the different possible states and decisions for player A and player B. Player A can be in a red state or a black state, while player B does not know which state A is in. The lecture goes through the possible outcomes for each state and decision, including the options to hold or resign for both players. The lecture also mentions the hidden information aspect of the game, where player B is unaware of the state he is in. The lecturer mentions that the game was obtained from Andrew Moore. The lecture then moves on to discussing the strategies for each player, stating that A has two strategies (resign or hold), while B has two strategies related to A's decision to hold. The lecture concludes by mentioning that the next step is to determine the values for different strategies using a matrix. The lecture ends with a quiz to test understanding.

3.2.16 Minipoker Tree Solution

In this lecture on Game Theory, the lecturer and Michael discuss the solutions for different scenarios in a minipoker game. They calculate the values for each possible outcome based on the decisions made by the players and the cards they receive. They also note that the value of the game is not determined solely by the minimax or maximin strategy due to the presence of hidden information. This introduces complexity and the need for mixed strategies instead of pure strategies.

3.2.17 Mixed Strategy Question

A mixed strategy in game theory refers to a distribution of probabilities over different strategies, as opposed to a pure strategy where only one strategy is chosen. In this context, player A can choose to either be a "holder" or a "resigner," and the mixed strategy for A is represented by the probability P of choosing to be a holder. A pure strategy is also considered a mixed strategy if all the probability mass is on a single strategy. In the quiz presented, player B is assumed to always choose to resign. The task is to determine player A's expected profit when A chooses to be a holder with probability P. Another scenario is introduced where player B always chooses to see the card, and player A's expected profit is to be determined when A chooses to be a holder with probability P as well.

3.2.18 Mixed Strategy Solution

In this lecture on mixed strategy solution in game theory, the speaker discusses the calculation of expected profit in a game. The speaker analyzes two players, A and B, and their strategies of holding or resigning. The speaker provides the equation for player A's profit which is 15P - 1, where P is the probability of holding. Similarly, for player B, the profit equation is -10P + 5. The speaker confirms the validity of these equations by checking the values at P=0 and P=1. The speaker emphasizes that this analysis is based on mixed strategies against deterministic strategies. The speaker mentions that the equations for profit are linear, and suggests drawing lines to visualize them.

3.2.19 Lines Question

The lecture discusses two lines: 15p-5 and -10b+5. The lines intersect, and the objective is to determine the point of intersection.

3.2.20 Lines Solution

In this lecture on Game Theory, the speaker discusses finding the solution to a system of equations representing two lines. The process involves setting the equations equal to each other and solving for the value of p where they are equal. The speaker demonstrates this process by finding that p is equal to 0.4 or $2/5$. The speaker emphasizes the importance of approaching the problem by setting the equations equal to each other and using simple algebra. It is noted that if p had ended up not being a probability, the lines would not have crossed inside.

3.2.21 Center Game

The lecture covers the concept of game theory and specifically focuses on the center game. The speaker discusses the value of the game at a given probability and explains that for the center game, the value is $0.01. The speaker also highlights that B's strategy choice does not change the value of the game and that no matter how B weights the average, the outcome will still be the same. The lecture further delves into the rationality of players and how they choose their strategies. It is explained that if A announces a mixed strategy, B can determine the optimal strategy to minimize A's payoff. The lecture also clarifies that the intersection of the lines in the game does not necessarily hold importance; rather, it is the maximum point that should be considered. The lecture outlines a method to determine the optimal strategy for A by plotting the lines, finding the minimum at each point, and selecting the maximum point. Additionally, the lecture addresses the question of why A should be the one to choose a random strategy instead of B. The speaker explains that both players are in the same situation and that the choice is based on the belief that the other player will try to minimize their payoff. Lastly, it is mentioned that the concept of game theory can be extended to more than two options, but it becomes more complex and involves searching for intersections. Overall, the lecture provides an overview of the center game in game theory, discusses strategies, and explores the rationality of players' choices.

3.2.22 Snitch 1

The lecture discusses two-player non-zero sum, non-deterministic games of hidden information. It introduces a game involving two criminals who have been captured by the police and put in separate jails. Each criminal is given the opportunity to implicate the other in order to avoid jail time. If one criminal defects before the other, they will walk free while the other serves a nine-month sentence. If both criminals remain silent, they will each serve a lesser sentence. The lecture explains that there are four possible outcomes: both criminals can defect, both can cooperate, one can defect while the other cooperates, or both can remain silent. The lecturer proposes drawing a matrix to represent the costs associated with each outcome.

3.2.23 Snitch 2

In Game Theory, two individuals, A and B, are faced with a choice to cooperate or defect. The outcomes and rewards for each choice are represented in a matrix. If A cooperates and B defects, A receives -9 (months in jail) and B receives 0. If A defects and B cooperates, A walks free and B receives -9. If both A and B confess, they both receive -6 months in jail. If both keep quiet, they each receive -1 month. The best outcome for

them is mutual cooperation, as they both spend a lesser amount of time in jail. However, the decision to cooperate depends on what each individual believes the other will choose. If A knows B will cooperate, it is advantageous for A to defect and go free. Similarly, if A knows B will defect, A should also defect. Thus, the overall outcome is dependent on the beliefs and choices of each player.

3.2.24 Snitch 3

In this lecture, the professor discusses the concept of dominance in game theory using the example of the Prisoner's Dilemma. The professor explains that in the Prisoner's Dilemma, both players have a dominant strategy to defect rather than cooperate, resulting in a less favorable outcome for both parties. The professor points out that the only way to break this dilemma is if the players can communicate and collude with each other. However, in a simple version of the game, where communication is not possible, the dilemma remains unresolved. The professor then introduces the concept of strict dominance and explains that although it may not work in all cases, it is often used to solve complex games and determine their true value.

3.2.25 A Beautiful Equilibrium 1

The lecture introduces the concept of Nash Equilibrium in game theory. Nash Equilibrium occurs when each player in a game chooses a strategy that maximizes their utility given the strategies chosen by all other players. In other words, no player has an incentive to change their strategy if given the chance. Nash Equilibrium can be applied to both pure strategies (specific choices) and mixed strategies (probabilistic distributions). The concept is named after John Nash, a Nobel Prize-winning mathematician portrayed in the movie "A Beautiful Mind." Understanding Nash Equilibrium is essential in analyzing strategic decision-making in multi-player scenarios.

3.2.26 A Beautiful Equilibrium 2 Question

The lecturer presents a quiz to find the Nash equilibrium for two matrices: the prisoner's dilemma and a symmetric-looking but not quite matrix. The lecturer explains that each row and column represents a choice for one of the players, and the numbers in each pair represent the payoffs for players A and B. The lecturer mentions that probability distributions may or may not be needed to find the Nash equilibrium. The lecturer also assures the student that there are pure Nash equilibria present in the matrices. Finally, the student is asked to circle or underline the answer to the quiz.

3.2.27 A Beautiful Equilibrium 2 Solution

The lecture discusses finding Nash Equilibria in two different games. In the first game, called the prisoner's dilemma, the lecturer demonstrates that one Nash Equilibrium is when Player A chooses the second row and Player B chooses the second column, resulting in a payoff of -6 for both players. The lecturer also explains that this strategy is a Nash Equilibrium because neither player can benefit by switching their strategy. In the second game, the lecturer explores whether there are any dominant strategies that can help identify the Nash Equilibria. The lecture concludes that there are no dominant strategies, but points out that there is a strategy where both players can receive a payoff of 6, making it the Nash Equilibrium. The lecturer notes that finding the Nash Equilibrium in both games was easier than expected.

3.2.28 A Beautiful Equilibrium 3

The lecture discusses three fundamental theorems related to Nash equilibrium. The first theorem states that in an n-player pure strategy game, if elimination of all strictly dominated strategies leaves only one combination of strategies, then that combination is the unique Nash equilibrium. This is illustrated by the example of the prisoner's dilemma. The elimination process is done iteratively, getting rid of whatever can be eliminated in each round. The second theorem states that any Nash equilibrium will still exist after the iterated elimination of strictly dominated strategies. This makes sense because if a strategy is strictly dominated, it would never be chosen and therefore cannot be a Nash equilibrium. The third theorem states that in a finite game with a finite number of players and finite sets of strategies, there will always exist at least one Nash equilibrium, which may involve mixed strategies. This implies that there is always a Nash

equilibrium in any finite game. The lecture concludes by emphasizing the importance of understanding Nash equilibrium in solving complex games and the concept of equilibrium in general.

3.2.29 The Two-Step

In this lecture on game theory, the speaker discusses the potential impact of communication on the outcome of the prisoner's dilemma game. Despite considering the possibility of communication between the players, it is determined that going first in the game still puts one at a disadvantage. The speaker introduces the idea of playing the game multiple times and using information from previous games to inform future decisions. However, when considering a two-step version of the game, it becomes apparent that there are eight possible combinations of actions for each player, resulting in a complex decision matrix. Despite the complexity, the speaker concludes that filling out the matrix is unnecessary, as it ultimately does not change the outcome of the game.

3.2.30 2Step2Furious

The speaker discusses the concept of playing multiple repeated games and how strategies for cooperation or defection may change depending on the position in the sequence of games. The equilibrium strategy in the final game is always to defect, regardless of any trust that may have been built up in previous games. The speaker mentions that this result is not unique to their discussion, but rather a theorem derived from Nash equilibrium. The speaker acknowledges that if there are multiple Nash equilibria, choosing among them is a separate problem. The lecture concludes by addressing the idea that knowing the end of the world, even without knowing when, may affect behavior in repeated games. The speaker suggests further exploration of this topic in the next lecture.

3.3 Game Theory Continued 3.3.1 The Sequencing

3.2.31 What Have We Learned

In this lecture, the speaker and Michael discuss what they have learned about game theory. They talk about how game theory can be depressing and how changing the numbers in the matrix can affect outcomes in the prisoner's dilemma. They also discuss how changing the game can be done by altering everyone's utilities and creating a system where snitches are punished. They mention mechanism design as a way to encourage certain behaviors through changing payment structures. They also touch on topics like strategies, different types of games (perfect and hidden information, zero-sum and non-zero sum, deterministic and non-deterministic), and Nash equilibria. The speaker mentions that the examples used in the lecture come from Andrew Moore's slides and recommend looking at them. They also mention that there is more to learn about game theory, including other equilibrium concepts and repeated games. They plan to explore these topics in future lessons.

3.3.2 Iterated Prisoners Dilemma

The lecture discusses the iterated prisoners dilemma, a scenario involving two criminals, Smooth and Curly, deciding whether to cooperate or defect against each other after being arrested. If they have multiple rounds to interact, they always defect since it is irrational to do otherwise. The lecture explores the scenario of having more than one round and concludes that it leads to the same outcome as a single round, where both players defect. This argument can be extended to any number of rounds. However, the lecture raises the question of what happens if the number of rounds is unknown. It initially seems like it should have the same outcome as a known finite number of rounds, but it actually makes a difference. The lecture teases that this difference is interesting and connects to other topics previously discussed.

3.3.3 Uncertain End

The lecture discusses the concept of an uncertain ending in game theory. The idea is represented using a generic probability distribution over the number of rounds in a game. The example used is a game of prisoner's dilemma, where after each round, a coin is flipped. With a probability of 1 minus gamma, the game ends, but with a probability of gamma, the game continues. Gamma represents the probability of continuing the game, and it is equated to the discount factor used in previous discussions. The expected

number of rounds in this game can be expressed as 1 over (1 minus gamma). As gamma approaches 1, the expected number of rounds approaches infinity. This is related to the concept of discount factors used in previous discussions.

3.3.4 Tit-for-Tat 1

In this lecture, the professor introduces the concept of strategies in a game with an uncertain ending. They explain that traditional sequences of actions or decision trees are not sufficient and that a different representation is needed to play for an unbounded number of rounds. The professor then presents an example strategy called "tit for tat" for the iterated prisoner's dilemma game. The strategy starts by cooperating on the first round and then copies the opponent's previous move in all future rounds. They clarify that if the opponent's move pattern is "cooperate, defect, cooperate, defect, cooperate, defect, defect, defect, defect, cooperate, cooperate, cooperate," the tit for tat agent will show a similar pattern. The professor represents the strategy as a finite state machine, showing how it proceeds by observing the opponent's move and following corresponding arrows to determine its own move in each round. The black letters represent the agent's move, while the green letters represent the observation of the opponent's move in the tit for tat strategy.

3.3.5 Tit-for-Tat 2 Question

This lecture discusses the concept of Tit for Tat strategy in game theory. The speaker poses the question of what happens if we follow Tit for Tat, and provides a set of opponent strategies for analysis. The goal is to determine Tit for Tat's response to each strategy. The speaker encourages the audience to engage in the exercise by checking the corresponding boxes for each scenario.

3.3.6 Tit-for-Tat 2 Solution

Tit for Tat is a cooperative strategy that starts by cooperating. If it plays against someone who always cooperates, it will continue to cooperate. If it plays against someone who always defects, it will cooperate once and then defect for the rest of the game. If it plays against another Tit for Tat, both players will always cooperate. If it plays against a strategy that alternates between defecting and cooperating, Tit for Tat will copy the opponent's previous move. This pattern of responses is consistent across various scenarios.

3.3.7 Facing TfT Question

Facing the tit for tat strategy, there are two possible approaches: always defect or always cooperate. The total discounted reward for always cooperating against tit for tat is -1/(1-Gamma), while the total reward for always defecting is -6/(1-Gamma) * Gamma. For high values of gamma, always cooperate is more favorable, while for low values of gamma, always defect is preferable. There is a specific value of gamma for which these two strategies are equally good, but the answer is not provided in the text.

3.3.8 Facing TfT Solution

The speaker discusses the calculation of the optimal strategy for iterated prisoner's dilemma in the context of tit for tat (TfT) solution. They determine that for gamma values less than 1/6, defecting is the best strategy, as the game won't last long enough to form a coalition. However, for values greater than 1/6, it is better to cooperate than to defect against TfT.

3.3.9 Finite State Strategy

The lecture discusses the computation of a best response to a finite-state strategy in game theory, using the example of the tit for tat strategy. The lecturer explains that a finite-state strategy like tit for tat can be represented as a finite state machine. The lecturer introduces the concept of an MDP (Markov Decision Process) and explains that the game can be seen as a discounted MDP, with the opponent's strategy serving as states, the payoffs as rewards, and the player's choices as actions. The lecturer then highlights the importance of considering future decisions and maximizing payoff over time. It is stated that solving the MDP can help determine an optimal strategy against the finite-state strategy. The lecture concludes by discussing the three strategies that can be effective against tit for tat: always cooperate, always defect, and

taking the loop between defect and cooperate. The lecturer emphasizes that these strategies are the only ones that matter because they do not require remembering past actions, which is impossible in an MDP.

3.3.10 Best Responses in IPD Question

In this lecture, the instructor discusses the concept of computing a best response against a finite state strategy. The instructor then gives a quiz to determine the best response to different strategies. The first question asks for the best response when playing against a strategy of always cooperating. The second question asks for the best response when playing against a strategy of always defecting. Finally, the third question asks for the best response when playing against a strategy of tit for tat. The instructor clarifies that the opponent's strategies are represented by the rows, and the best possible responses are represented by the columns. The quiz is meant to determine which column will yield the maximum reward.

3.3.11 Best Responses in IPD Solution

In this lecture, the concept of best responses in the Iterated Prisoner's Dilemma (IPD) game is discussed. The instructor first clarifies that cooperating is better than defecting when the discount factor gamma is greater than 1/6. However, this only applies when playing against an opponent who always cooperates, not the general case. The instructor then explores different scenarios, such as playing against an opponent who always defects or against tit for tat. It is concluded that always defecting is the best response in the former scenario, while cooperating or using tit for tat strategy leads to a Nash equilibrium in the latter scenario. The concept of Nash equilibrium is defined as a pair of strategies where each is a best response to the other. The lecture also mentions the possibility of modifying the reward structure or playing multiple rounds to encourage cooperation in the IPD game. This is only possible in the repeated game setting and involves a subtle change in the game dynamics. The concept of changing the game is emphasized, with the reminder to "hate the game, not the player".

3.3.12 Folk Theorem

In this lecture on game theory, the concept of repeated games and the folk theorem are discussed. In repeated games, the possibility of retaliation can lead to cooperation. The tit for tat strategy is mentioned as an example. The term "folk theorem" is explored, with two different meanings given. In mathematics, a folk theorem is a known result but not formally published. In game theory, the folk theorem refers to a specific result that describes the set of payoffs from Nash strategies in repeated games. The idea of building up to the folk theorem using basic concepts is mentioned.

3.3.13 Repeated Games 1

The lecture introduces the concept of a two-player plot, also known as a folk plot, to understand the folk theorem in game theory. The plot is used to visualize the outcomes of different joint actions in a prisoner's dilemma game. Each joint outcome is represented by a dot on a two-dimensional plot, with Smoove's actions on the x-axis and Curly's actions on the y-axis. The lecture acknowledges that the plot loses some information compared to the matrix representation of the game, as it does not capture the relationship between players' actions when one player changes while the other remains the same.

3.3.14 Repeated Games 2 Question

The lecturer introduces a quiz regarding average payoffs in a repeated game setting. The payoffs for different strategies are presented, and the question is posed: which of these payoffs can be the average payoff for some joint strategy? The goal is not to maximize reward or achieve Nash equilibrium, but simply to find a strategy that results in a specific average payoff. The lecturer asks the audience to check the box if the specified average payoff is possible for the given payoffs. The audience is given time to think about and answer the question.

3.3.15 Repeated Games 2 Solution

In this lecture, the concept of repeated games in game theory is discussed. The speaker begins by addressing a question about the joint strategy of getting a payoff of -1, -1, and explains that both players would

cooperate to achieve this. The speaker then analyzes other possible payoffs and concludes that all possible achievable averages form a convex hull. They explain that outside points cannot be achieved, and the point (-4, -4) is inside the convex hull, suggesting a combination of two-thirds D,D and one-third C,C. The speaker mentions a more general result involving the convex hull. They present a graphical representation of the feasible region, where colluding joint strategies can achieve payoffs anywhere inside the region. The speaker and a student discuss the concept of feasible regions and its application to game theory. The graphical representation is considered a useful way to understand the concept.

3.3.16 Minmax Profile Question

A minmax profile in game theory is a pair of payoffs, one for each player, that represents the payoffs they can achieve by defending themselves from a malicious adversary. In this context, a malicious adversary is someone who is trying to give the lowest score to the other player. The concept of a minmax profile is similar to zero-sum games, where one player's payoff is the negative of the other player's payoff. In the example of the Bach and Stravinsky game, Smooth and Curly are choosing independently between attending a Backstreet Boys concert or a Sting concert. If they choose different concerts, they both get zero payoff. If they choose the same concert, they both get a positive payoff, but Smooth prefers Backstreet Boys and Curly prefers Sting. The minmax profile for this game is the payoff that each player can guarantee themselves even if the other player is trying to minimize their score.

3.3.17 Minmax Profile Solution

In this lecture, the concept of a minmax profile solution in game theory is discussed. The lecturer uses an example involving two players, Curly and Smoove, who are trying to maximize their own score while minimizing the other player's score. Curly is choosing among rows and Smoove is choosing among columns. The lecturer demonstrates that Smoove can choose a strategy that forces Curly to get a score of $2/3$, even if Curly behaves optimally. The lecturer also introduces the concept of a security level profile, which allows for the possibility of mixed strategies. Both the minmax profile solution and the security level profile can be used to analyze game scenarios, and the lecturer prefers the latter. The lecture concludes with the suggestion to move on to the next example, where these concepts align.

3.3.18 Security Level Profile

The lecture discusses the concept of the minmax or security level profile in the context of the prisoner's dilemma. The minmax value in this case is "d, d", which represents the guaranteed outcome against a malicious adversary. The lecture introduces a new region defined by this minmax point and proposes calling it the acceptable or preferable region. This region represents outcomes that are better than what can be guaranteed against an adversarial situation. The intersection of the acceptable region and the existing feasible region is referred to as the feasible preferable acceptable region. The lecture then mentions that the Folk Theorem will be discussed next.

3.3.19 Folksy Theorem

The Folk Theorem states that any feasible payoff profile that is better than the minimum security level can be achieved as a Nash equilibrium payoff profile. This can be done by following instructions given by the other player, otherwise they will adopt a strategy to force the player down to their minimum security level. The only way this can be stable is if the feasible payoff is better than the minimum security level. The Folk Theorem can also be called the folksey theorem in a more casual sense.

3.3.20 Grim Trigger

The grim trigger strategy is a way to prove the "folk theorem" in game theory. It involves starting with a mutually beneficial action or pattern of actions that will continue as long as cooperation continues. However, if cooperation is ever broken, the strategy is to deal out vengeance forever. In the context of the prisoner's dilemma, cooperation is the mutually beneficial action, and as long as one person cooperates, the other person will cooperate as well. But if one person defects, the other person will defect indefinitely. The purpose of this strategy is to create a Nash Equilibrium system where neither player has an incentive to defect. However, there is a problem with the strategy that needs to be addressed.

3.3.21 Implausible Threats

The lecture discusses the concept of implausible threats in game theory. An implausible threat refers to a situation where a player threatens a certain action, but it is not in their best interest to follow through with the threat. A plausible threat, on the other hand, corresponds to a subgame perfect equilibrium, where each player always takes a best response regardless of the history. The lecture uses the example of the Grim Trigger and Tit for Tat strategies. These strategies are in Nash equilibrium with each other since deviating from them would not result in better outcomes. However, the question is whether they are in a subgame perfect equilibrium. To test this, one needs to examine the history of actions and determine if one player can deviate from the strategy and achieve better results. In the case of Grim Trigger and Tit for Tat, a sequence of moves can be identified where Grim could choose to cooperate and achieve a better outcome. This indicates that these strategies are not in a subgame perfect equilibrium because an implausible threat exists. The lecture concludes by acknowledging that the concept of implausible threats potentially undermines the effectiveness of achieving cooperation in game theory. However, the lecturer suggests that there may still be ways to address this issue.

3.3.22 TfT vs. TfT Question

The lecture discusses the concept of subgame perfect equilibrium between tit-for-tat strategies in a game. The instructor asks students to evaluate whether tit-for-tat strategies are subgame perfect and provides two choices: yes or no. Students are asked to give a sequence of actions to demonstrate if the strategies are not subgame perfect, meaning that the machines would not be willing to follow through on their threat in the long run. They are also asked to consider if altering the sequence of actions would change their decision to follow tit-for-tat in the next time step.

3.3.23 TfT vs. TfT Solution

In this lecture, the discussion revolves around the concept of subgame perfection in the context of the Prisoner's Dilemma. The lecturer and the person they are conversing with analyze the strategy of "tit for tat" (TfT) in this game. They explore the scenario where one of the TfT agents defects at the beginning, leading to a sequence of alternating defections and cooperations. Through calculations of expected rewards and average scores, they determine that it is not advantageous for one agent to defect at the beginning. They conclude that this strategy is not subgame perfect. The conversation then turns to the question of whether there is a way to achieve subgame perfection in the Prisoner's Dilemma. The lecturer proposes a machine and suggests further investigation.

3.3.24 Pavlov

The Pavlov machine is a strategy in game theory that is similar to tit for tat, but with a twist. It starts off cooperating and continues to cooperate as long as the opponent also cooperates. However, if the opponent defects, the machine will defect against them. The unique aspect of Pavlov is that if the opponent defects, the machine will then cooperate with them. This strategy aims to take advantage of the opponent until they "pull the trigger" by defecting, at which point the machine will cooperate again. The strategy is characterized by a sequence of actions: cooperate if both players cooperate, defect if the opponent defects, and cooperate again if the opponent cooperates after defecting. The lecturer raises the question of whether Pavlov is a Nash equilibrium.

3.3.25 Pavlov vs. Pavlov Question

The lecture presents a quiz regarding the concept of Pavlov v. Pavlov and whether it aligns with Nash equilibrium. The answer is not provided immediately to allow the audience time to consider the question.

3.3.26 Pavlov vs. Pavlov Solution

According to the lecture, the solution to the Pavlov vs. Pavlov game is for both players to co-operate. This leads to a Nash equilibrium where cooperation is maintained.

3.3.27 Pavlov Is Subgame Perfect

Pavlov is shown to be subgame perfect, unlike tit for tat. By examining different sequences and states, it is observed that Pavlov machines will always end up in mutual cooperation, regardless of the initial state. The ability to return to mutual cooperation regardless of the sequence makes Pavlov a plausible and stable strategy. The significance of subgame perfection lies in the fact that a player can make a threat or punish the opponent, knowing that it will not ultimately cost them and will stabilize the opponent's behavior.

3.3.28 Computational Folk Theorem

The computational folk theorem is a concept that applies to any two-player bimatrix game, including the prisoner's dilemma and iterated prisoner's dilemma. This theorem allows for the construction of a Pavlov-like machine that can generate a subgame-perfect Nash equilibrium for these games in polynomial time. If the game allows for a mutually beneficial relationship, this machine can quickly solve it and produce a Nash equilibrium. Alternatively, if the game is zero-sum, a linear program can be solved to determine the strategies for each player. In some cases, at most one player can improve their behavior, and by taking the best response against the other player in a zero-sum sense, a Nash equilibrium can be identified. Overall, there are three possible forms of Nash equilibria, and they can be determined in polynomial time. This result was discovered by Peter Stone and the lecturer. The lecturer mentions that this will conclude the discussion on the folk theorem and repeated games, and the next topic will be stochastic games, which are a generalization of repeated games and relate to concepts like queue learning and MDPs. The lecturer jokingly suggests that the students demand more classes as the course comes to an end.

3.3.29 Stochastic Games and Multiagent RL

Stochastic games, also known as Markov games, are a generalization of both MDPs and repeated games. They provide a formal model for multiagent reinforcement learning. In a stochastic game, players take turns moving on a grid, with both deterministic and stochastic transitions. The goal is to reach a specific state and receive a reward. Players cannot occupy the same square and as soon as one player reaches the goal, the game ends. They can collaborate to reach the goal together, but there is contention over who gets the reward. The concept of Nash Equilibrium is introduced, which is a pair of strategies that neither player wants to deviate from. However, finding Nash Equilibria in stochastic games can be challenging. Learning in these environments is also discussed.

3.3.30 Stochastic Games

Stochastic games, introduced by Shapley, consist of states (s), actions for player one (a) and player two (b), transition probabilities, rewards for both players, and a discount factor. The transition function determines the probability of reaching a next state given a joint action (a,b). The discount factor is a universal quantity in stochastic games. It is sometimes included in the definition of problems or algorithms, but it is generally preferred to include it in the game definition. Shapley's model is a generalization of Markov Decision Processes (MDPs), which were introduced by Bellman later. MDPs can be seen as a subset of stochastic games.

3.3.31 Models and Stochastic Games Question

Stochastic Games are more general models compared to others. They can be constrained in three ways: 1. Opposing reward functions for both players. 2. Transition function unaffected by player two's actions, and player two's rewards are always zero. 3. Only one state in the environment. These constraints result in specific models: Markov Decision Processes, zero-sum stochastic games, and repeated games. Students are instructed to place the letters A, B, and C in the appropriate boxes.

3.3.32 Models and Stochastic Games Solution

The lecture discusses different scenarios in game theory and their implications. The first scenario is a zero-sum stochastic game, where the payoffs of the players are equal and opposite, resulting in a sum of zero. The second scenario is a regular Markov decision process (MDP), where one player is essentially the only agent involved, making the other player irrelevant. The lecturer explains that as long as the rewards of the

second player are constant, they can be ignored by the first player. However, it is noted that the actions of the first player can still be influenced by the second player, so the rewards can be set equal to each other to account for that. The third scenario is a repeated game, where actions impact rewards but not transitions, and the discount factor determines the stochastic end of the game. The lecturer suggests that it would be beneficial to generalize techniques like Q learning and value iteration from MDPs to this more general setting.

3.3.33 Zero-Sum Stochastic Games 1

In this lecture on zero-sum stochastic games, the professor discusses the challenges of incorporating the impact of player actions on future states. To address this, they introduce the Belmont Equation which is used to define the value function. However, the equation is not suitable for zero-sum stochastic games. The professor suggests modifying the equation to consider a matrix game and to use a min operator rather than a max operator. This approach assumes that all players maximize their benefit and disregards the assumption of a zero-sum game. To fix this, the professor proposes solving the zero-sum game in the Q values and incorporating the value into the equation. This modified equation, known as mini-max Q, is analogous to Q learning with the min operator. The professor concludes that zero-sum games are generally limited to two players due to the presence of a third player disrupting the balance.

3.3.34 Zero-Sum Stochastic Games 2

In zero-sum stochastic games, value iteration can be used to solve the system of equations and find the unique Q star values. The minimax Q algorithm converges similarly to Q learning. The policies for each player can be computed independently and will converge to minimax optimal policies. The update equation can be calculated efficiently using linear programming. Q star values can be used to determine optimal behavior. General sum games are more difficult to solve, and it is unknown if they can be solved in polynomial time.

3.3.35 General-Sum Games

In the lecture, the speaker discusses general-sum games, which are not limited to being zero-sum. They explain that the minimax algorithm cannot be used in this context because it assumes the other player is trying to minimize one's reward. Instead, the speaker suggests using the Nash equilibrium concept to compute the value of the game and propagate it back. This approach can also be applied in the Q-learning setting, resulting in the Nash-Q algorithm. However, value iteration does not work well with Nash-Q, as it may not converge to a solution. The reasoning is challenging because there is no unique solution to Q star, and different Nash equilibria can have different values. Additionally, computing a Nash equilibrium is a computationally intensive task, as hard as any problem in the class known as PPAD, which is believed to be as hard as NP. Finally, even if Q values are efficiently computed, they are not sufficient to determine the policy. In general-sum games, the leverage provided by Q-type algorithms is lost.

3.3.36 Lots of Ideas

Several ideas have been proposed for addressing general sum games. One approach is to consider stochastic games as repeated games, allowing for the development of folk theorem-like ideas. Efficient algorithms exist for dealing with this. Another idea is using cheap talk, a form of communication that is not binding but allows for a limited form of coordination between players. This can be used to compute a correlated equilibrium, which is an efficient approximation of the solution to stochastic games. The notion of a cognitive hierarchy is another idea, where players assume that others have limited computational resources and respond accordingly. This is a good model for how people actually play games in the laboratory. Side payments can also be used to encourage cooperation and balance the aspects of zero-sum and mutual benefit in games. These ideas bring us closer to solving the general sum games problem, although it is not fully understood yet.

3.3.37 What Have We Learned

In this lecture, the speaker reflects on the key concepts learned in the context of game theory and reinforcement learning. They highlight the following points: - The Iterated Prisoners Dilemma is an important

concept that can be connected to reinforcement learning through discounting. - Repeated games allow for cooperation and the emergence of new Nash equilibria, as demonstrated by the idea of the Folk Theorem. - Plausible threats can be used to enforce cooperation in subgame perfect equilibria. - The speaker mentions Min-max Q and the Computational Folk theorem in the context of repeated games. - The discussion then moves on to stochastic games, which generalize Markov Decision Processes (MDPs) and repeated games. - Nash Q is mentioned as a concept that offers hope despite its limitations. - The speaker concludes that research requires perseverance and a willingness to learn, even when faced with impossibility results. The lecture ends on a positive note, with the speakers expressing gratitude and anticipation for wrapping up the class in person.