

GAME THEORY

2: THE SEQUENCING

1. Hello Charles. >> Hi Michael. >> How are you today? >> I'm doing just fine. How are you doing? >> Alright. I'm a little out of practice with this lecturing thing. So, I hope this goes well. >> I'm absolutely sure it will. >> So, today's lesson is continuing what you were talking about in terms of Game Theory. But, I'm going to be focusing in on what happens when you are worried about making decisions, with more than one player in a sequence. Which we started to get into at the end of your discussion but I'm going to go, more into it. >> Okay. I really like the logo by the way. >> Thanks very much. Yeah, it's a, it's a specially game theory logo. >> I like it very much, I like it very much. I will point out, however, that all sequels should be called the quickening. >> Yeah, I was going to go with the quickening. Or judgement day, but I didn't, didn't think that made any sense. >> Hm, that's a fair point.

Iterated Prisoner's Dilemma

	C	D
C	-1, -1	-9, 0
D	0, -9	6, -6

What happens
if number
of rounds left
is unknown?

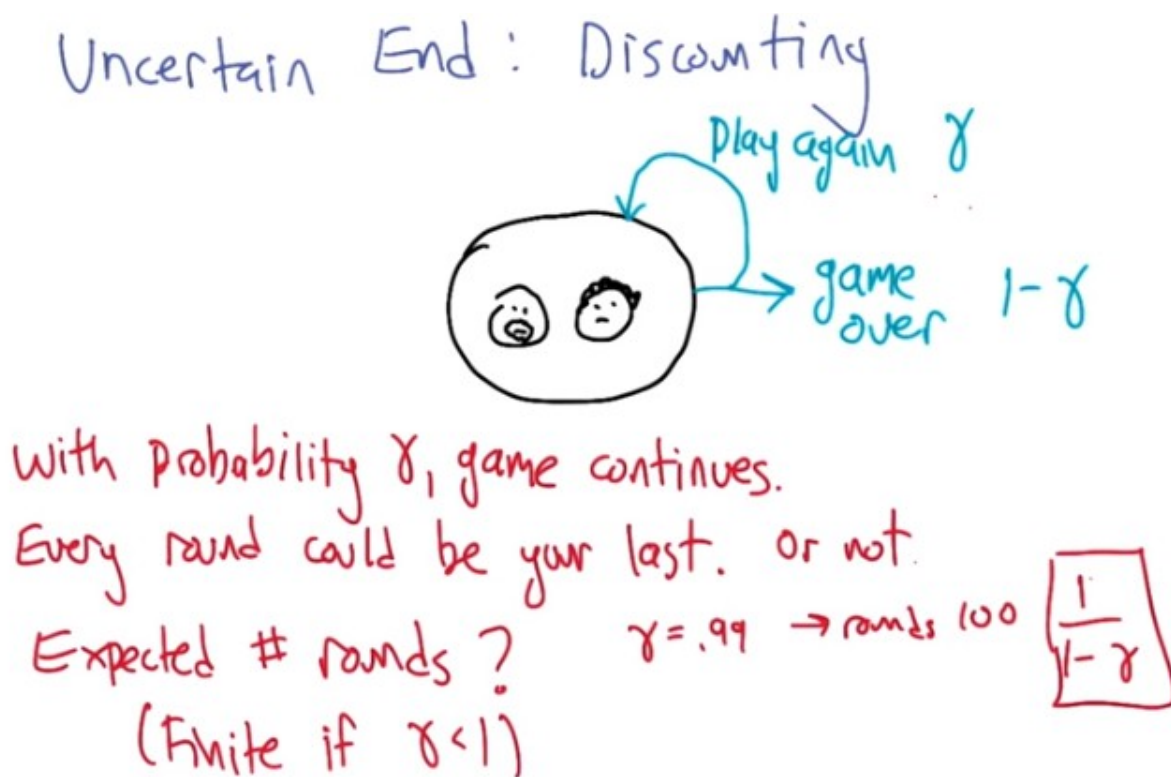
three rounds two rounds one round

...



2. So let me take you back into what we were talking about last time. We were talking about the iterated prisoners dilemma, and here's the prisoners dilemma payoffs that you wrote down for us. You remember this? >> I vaguely remember this. >> And remember it was about these two criminals, Smooth and Curly. And they were, deciding whether to cooperate or defect against each other when,

after they've been arrested. >> Right, and they defect, they always defect. They always defect, right. So in particular, we say well what happens if it's, **if they have multiple rounds** in which to interact, and so here they are, here are the two of them, and if they've got one round to live, we did an analysis and **we indicated that there's really nothing they can do other than defect against each other.** >> Mm-hm. >> It's irrational to do anything else. >> Yep. >> So we said, all right, well, what happens if we allow there to be more than one round? So now we've got two rounds. And what we realized was that if you got two rounds to go, then these players essentially, face a one-round game because, after this round (two rounds 下面那個), what they're going to do in the last round (one round 下面那個) has already been determined. So it's almost as if that round doesn't really matter. There's nothing we can do now that's going to change what they're going to do in that last round, so it's sort of like there's just one round and we're going to defect again. >> Right, so life is terrible and everyone is out to get everyone else. >> Exactly, and not only is it for two rounds, but this same argument continues as you go three rounds or more. >> Oh, it's like a proof by induction. >> It's kind of like a proof by induction, yeah, well it's proof by ellipsis. >> Mm, that's my favorite kind of induction. Proof by ellipsis. >> [LAUGH] So the question then becomes, what happens if the, number of rounds left is unknown, right? So what we've realized is that if you know how many rounds are left, the whole thing comes unraveled and they're just going to defect forever. But we raised the issue of, **what happens if the number of rounds left is unknown?** >> Hm. >> And it seems like it shouldn't really make any difference because if it's say some finite number we just don't know what it is, then it seems like it should still reduce to this same setup that we have. So I was looking into this and it turns out that is it's not the case it actually does make a difference and it's, it's really interesting how it goes and how it connects back with other things we've talked about. >> Woo, tell me more.



3. So here's how I started to think about it. So let's say how can we represent the idea that we have an **uncertain ending**. We'll **one way would be if we had some kind of generic probability distributions over**

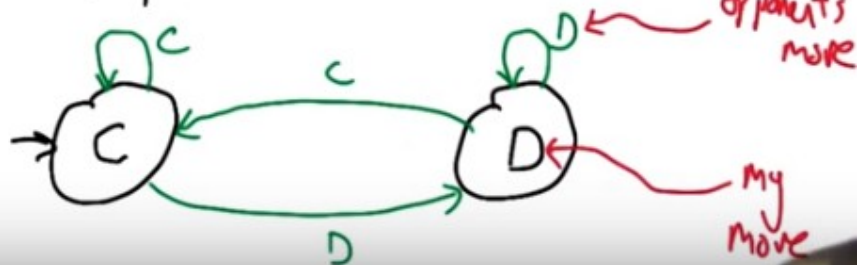
the number of rounds that the games going to be played. But this seems like an, the simplest idea that I could think of. So here, here we have our two criminals, and what they're going to do is their going to play around of prisoner's dilemma. But at the end of that round, they're going to flip a coin. And with probability $1-\gamma$, that will be the last round, it's all over. But with probability γ , they're actually going to play again. And, and they do this after each round. And so each round is basically statistically independent of the other rounds. >> I see. >> So the set up here is, with probability gamma, the game continues. Now, notice that I chose gamma here. This was a, it's representing a probability here, but in the past, we've used this to represent a discount factor. >> Mm-hm. >> But that actually is the same thing, right? In, in, in the normal discounting, we say that the value that you get one step from now, is discounted downward by gamma. And that's exactly what you'd take if you worked out the expected value of a game where you continued making steps with probability gamma. And with probability one minus gamma, it ends if you get zero from then until the rest of time. So every round here could be your last, or not, right? It could be that you actually get to continue playing. Does that make some sense? >> That makes perfect sense. >> Awesome. All right, so, so yeah, this is, this is exactly that kind of situation where, well, here, let me, let me ask you a question. What's the expected number of rounds of this game? >> Well, I'll bet it's finite if gamma's less than one Yes, I even wrote that down. >> Yeah, I'm smart. Or at least I can read. >> Sure, but what's the, but, specifically we could actually write it as a, as a, function of gamma. >> Let's see. If gamma were something like, 99% then I would expect it to be about a 100, right? >> I think that's right. >> Yeah. >> So is that, is that your answer? [LAUGH] My function of gamma is if gamma is .99 the answer is 100. >> Yeah, something like that. It's not a total function but it's a function. >> Well, it's a sample. I mean, you do machine learning. Why don't you tell me what the function would be given that sample. >> Well, we can make it a quiz or I could just tell you. >> Why don't you just tell me. >> Alright. So $1/(1-\gamma)$ is the answer (後面沒證, 我也懶得證). It works for your example. >> Um-hm. >> 1 minus .99 is 100th and we're talking 1 over that so, you get a 100. And, yea we could go through the argument as to why that's that's what it is. But this one over one minus gamma is what shows up all the time. If gamma is zero, then we're talking about one over one. The game lasts one round. That's exactly what we'd expect. >> Mm-hm As gamma gets closer and closer to one this pro, this quantities getting closer and closer to infinity. So, >> Right. >> in fact if you know, it becomes unbounded as gamma hits one. So yeah. So this is the expected of rounds, and so that means like yeah. So as you said if gamma is 0.99, it's a 100 rounds. And we already, reasoned that at a 100 rounds the whole thing falls apart. Right, huh, and I noticed the one over one minus gamma, of course, is just like the way we did discount factors, when we started doing MVP's in the first place. >> Exactly, yeah, that, that kind of links them together. >> Hm, that's actually kind of neat.

Tit for Tat

Famous IPD strategy

⇒ Cooperate on first round

⇒ copy opponent's previous move thereafter



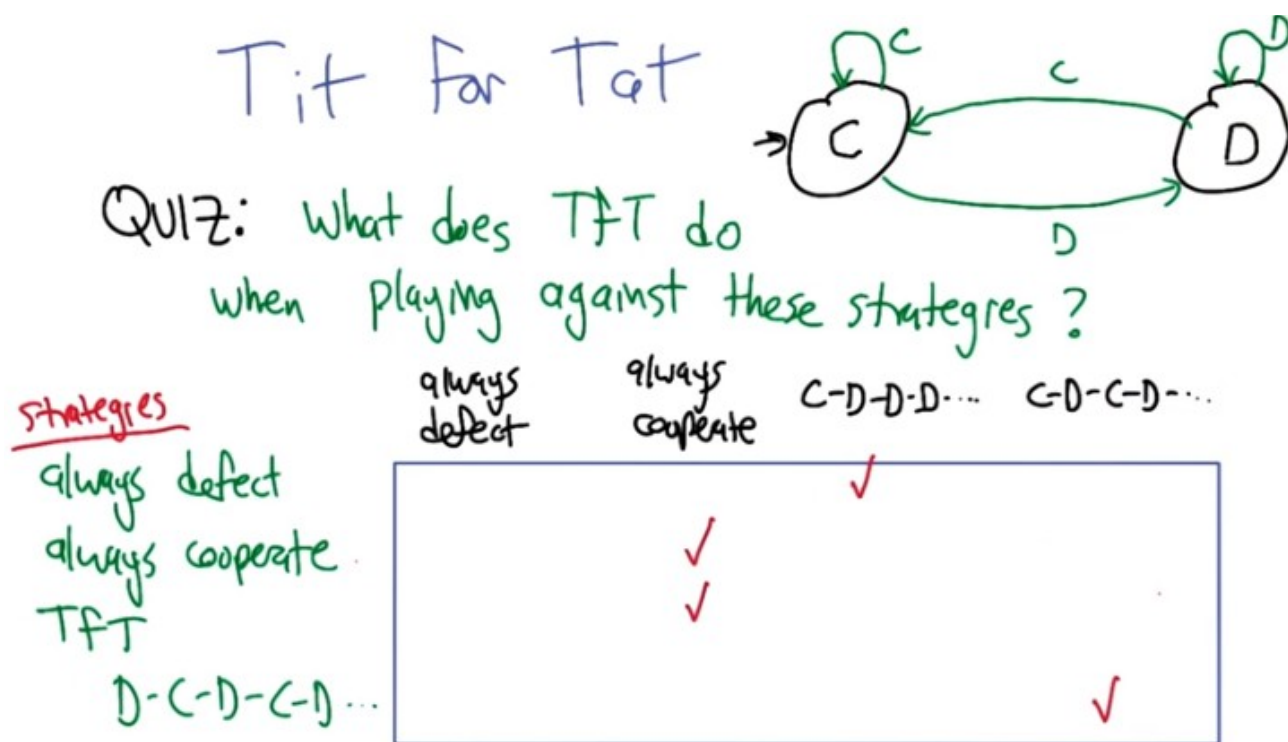
IPD: iterated prisoner's dilemma

注意是 copy 對方, 而不是 do the opposite of 對方

藍色的字表示對方做的, 黑色的字表示根据 Tft 來的 my move.

4. So if we're going to be talking about strategies in this game that has an uncertain ending, we can't just write down sequences of actions anymore. We can't just say cooperate, defect, defect, defect, defect. Or even some kind of tree of possibilities. Because those are going to be finite representations. We need some other representation that allows us to play for an unbounded number of rounds. >> Mm-hm. >> And I'm going to start off by presenting an example of such a strategy, one that's, that's very famous for the iterated prisoner's dilemma, and it's called tit for tat. And the structure of tit for tat goes like this, on the first round of the game, a player playing this strategy will cooperate, and then in all future rounds, the player is going to copy the opponents previous move. Does that make sense? >> It does. So basically, we start, I start out, acting as if, you're going to cooperate with me. And the moment you don't cooperate with me, I will start to defect, and we'll be in the, the old style prisoners' dilemma. Right? >> Well no, not. What this says is that it actually copies the opponent's previous move. So if, if an opponent goes cooperate, defect, cooperate, defect, cooperate, defect, defect, defect, defect, cooperate, cooperate, cooperate. You're going to see something very similar coming out of the tit for tat agent. >> I see, I see, I see. >> In fact, we can represent the strategy as a little finite state machine, like this. >> Yeah, I like that, okay. >> And you, so you can see exactly how it kind of proceeds. It starts off cooperating. And then in each round it waits to see what the opponent does. That's the green letters here. And then it follows the corresponding arrow, to determine whether it is going to cooperate or defect in the, in the current round. >> Sure, that makes sense. >> So in this picture, the black letters here represent my move. And the green letters represent my observation of the opponent's move. Or at least if I'm being tit for tat.

5. >> What happens if we follow Tit for Tat? >> That's a great question. So, let's make it a little more concrete. So, here's a set of strategies that an opponent might adopt. And now the question is, what does it look like Tit for Tat is doing in response to each of these? And I was hoping that you'd, you know, check the corresponding boxes. So basically, for each row, say okay, if you're playing against, if Tit for Tat is playing against always cooperate, what does Tit for Tat do? Does it always defect, does it always cooperate, does it cooperate and then defect, defect, defect, defect, does it alternate between cooperate and defect? And this should, just to give you some practice in, in interpreting the behavior of Tit for Tat. >> Okay, that works for me. I think I could do this. >> Go.



上圖 matrix 中, 藍色的字(即框右邊的)表示對方做的, 黑色的字(即框上面的)表示根据 TfT 來的 my move.

6. >> Alright, so let's start off, maybe you can tell me what happens when Tit for Tat plays against always cooperate. >> So what happens if you always cooperate? Well, let's see, I start out cooperating. >> Mm-hm. >> And since the other person is cooperating I will continue to cooperate because that's what they did the last time. >> Mm-hm. >> So, I will always cooperate. >> That's right. Good. Alright, so what about if we play against always defect. >> Well if we always defect, the first time I'm going to cooperate because that's what you said Tit for Tat is. >> Hm. >> But from that point on, I will do what my opponent does, which is defect. So I will cooperate and then defect, defect, defect, defect, defect, defect, ellipses. >> Yeah, so I put this always defect in there, but actually it can't ever be the right answer right [LAUGH] because Tit for Tat always starts off cooperating. So the, the other three seem like they might be possible, always defect is not possible. Good. Alright. So, what if Tit for Tat plays against another Tit for Tat? >> Well, I started out cooperating my opponent cooperated. And since I'm going to do what that person does I will cooperate, but since that person's doing what I did, they will also cooperate. And so we will both cooperate forever, so we will always cooperate. >>

Nice. So isn't that kind of interesting? So Tit for Tat even though, if, when it's playing against itself, is a very cooperative fellow. >> Hm, I like that. >> But if Tit for Tat is playing against something that defects, it becomes a little bit more vengeful. >> Yes. >> Alright, so what if it plays against something that is a little unsure of itself? So it starts off defecting, then cooperating, then defecting, it's sort of almost an anti kind of thing, right? So it's, this is one that starts off with a defect. >> Right, so it always, so I, first thing I do is cooperate. And then after that, effectively I do what the opponent does one step before. So I basically take what you, I'm pointing to the screen, you can't see me. I take the the D-C-D-C-D, and I just put a C in front of it, because that's what I'm going to do. So, I will do, C-D-C-D-C-D-C-D, ellipsis. So that's your last choice. >> Isn't the C-D-C-D-C-D, ellipsis in Atlanta? >> It is, actually. It is the home of all such analysis of diseases in the United States. >> The Center for Disease Control? >> Yes. >> Is that what it's called? >> Yes, that's exactly what it is. Come to Atlanta and work for us. >> [LAUGH] Alright, so good. So, that, that's the pattern of, of responses that Tit for Tat makes against this set of strategies. >> Oh, so we answered my question.

Facing TFT

What's the best response to TFT?

always D $0 + \frac{-6\gamma}{1-\gamma}$ best for low γ

always C $\frac{-1}{1-\gamma}$ best for high γ

For what γ are they equally good?

	C	D
C	-1, -1	-9, 0
D	0, -9	6, -6

$\frac{-6\gamma}{1-\gamma} = \frac{-1}{1-\gamma}$
 $\gamma = \frac{1}{6}$

$\frac{1}{6}$

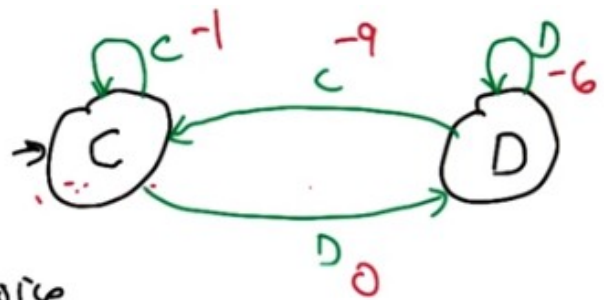
7. Alright so now that we have a sense of what tit for tat does against various strategies, lets try to think about what we should do against tit for tat, so what do we do if we're facing tit for tat. So I'm just going to break it down to two possibilities it turns out there's actually more, but these these two are pretty instructive. So lets pretend that we have to choose between always defect as a way of playing against tit for tat, or we have to be always cooperate playing against tit for tat. So what I've written down here (即紅字) is what the total discounted reward (即要坐幾個月的牢) is going to be or the total reward in this case, as a function of gamma. So, lets start with what happens if you play always cooperating against tit for tat. Well, you already told me that it. Such a thing will result in tit for tat always cooperating. >> Mm Hm. >> And that means we're going to play in this box (即右上角的(-1, -1)). The cooperate-cooperate box. And that means on every single round, we're going to to get a -1. Which means over an infinite run we're going to get an average of one, sorry $-1/(1-\gamma)$. >> Mm Hm. That makes sense. >> Okay? You agree with that? >> I do. Just minus one repeated over and over again. Now always defect as you recall you told me that, that will result, well for the always defect agent against tit for tat, the first thing that is going to happen is it's going to defect while the tit for tat cooperates right? So we're going to get zero for playing that strategy on the first round. Zero doesn't sound very good but look at

the alternatives. They're all negative so zero's pretty good. >> Mm-hm. >> So it does this sort of you know good thing in the first step then after that tit for tat responds by always defecting in response. Right? And that means we're going to be stuck in this box (即右上角的(-6, -6)), the defect defect box, where you get -6 for the rest of ever. >> Yes. >> So that means $-6/(1-\gamma)$. But that starts one step from now so we multiply it by another γ (Avadoles). >> Okay. >> Alright. So these are two different expression that represents what our pay off would be for adopting two different strategies. And in fact, if gamma is really high, very close to 1 (則 always C 比 always D 好, 因為 always D 負得更多), then this is a really good answer, right? Because it sort of grows, it's like, minus over one minus gamma. So, for high gamma, we're talking about something that's minus one times a really big number. >> Mm-hm. >> Whereas this first one is not so good for high gamma because what's going to happen is it's going to get, end up getting minus six on every step. So it's going to do worse overall. But if, if we're talking about the low gamma. Then, let's say, you know, zero for example (則 always D 比 always C 好, 因為 always D 是 0, 而 always C 是 -1). A gamma of zero will, will always defect. We'll get zero plus zero. But always cooperate, we'll get negative one over one. And zero is better than minus one. So for really small gamma, like if the games unlikely to last many rounds, you should defect. But is the game is going to last a long time, then you should cooperate. >> I believe that. >> Cool! Alright, so then my question to you is: What's the value of gamma for which these two different strategies to play against tit for tat are equally good. >> I think I know the answer. >> Woah! That was fast. Let's give everybody else a chance to think about it. >> Okay. >> Go.

(本段圖跟上段圖是一樣的)

8. >> Alright Charles, what, you said you had the answer how do we figure it out? >> It's 1/6th. >> I guess that's what, which, we don't, I don't know if it will accept that, whatever 1/6th is expressed as a decimal, but yes 1/6th. How did you get that? >> I saw the number six and figured it had to be 1/6th because you said it was low, but here is what actually I did, well, I was thinking about it is, you said, well when are they equally good? Well, if you always defect, you get minus 6 gamma over 1 minus gamma. And if you always cooperate, you get minus 1 over 1 minus gamma, so they're equally good when those two values are the same. >> Good, alright, and the denominators are the same, as long as gamma's not one, that's fine. >> Right. >> If we divide by the negative 6, we get gamma equals 1/6th. >> Exactly. >> Excellent. So, so that's interesting, right? So, that's, it's saying that for gamma values that are less than 1/6, we should be doi, we should just defect because there's no. Well the games not going to last long enough for us to form any kind of coalition. But for things higher than 1/6. A half. 3 quarters. 0.999. It's going to be better to cooperate than to defect against tit for tat. >> Or 6 plus epsilon. >> Indeed. >> Yeah. I like that. That's actually very cool.

Best Response To A Finite-state Strategy



states labeled with opponent's choice

edges labeled with our choice

edges annotated with our payoff for that choice

our choice impacts our payoff and
future decisions of the opponent.

① the matrix was all
we needed. once!

② MDP! - anything
- value iteration

上圖跟前面的圖是反過來的，黑的表示 oppent's choice, 藍的表示 our choice. 紅色數字表示我們作了相應決定後，要坐幾個月牢。

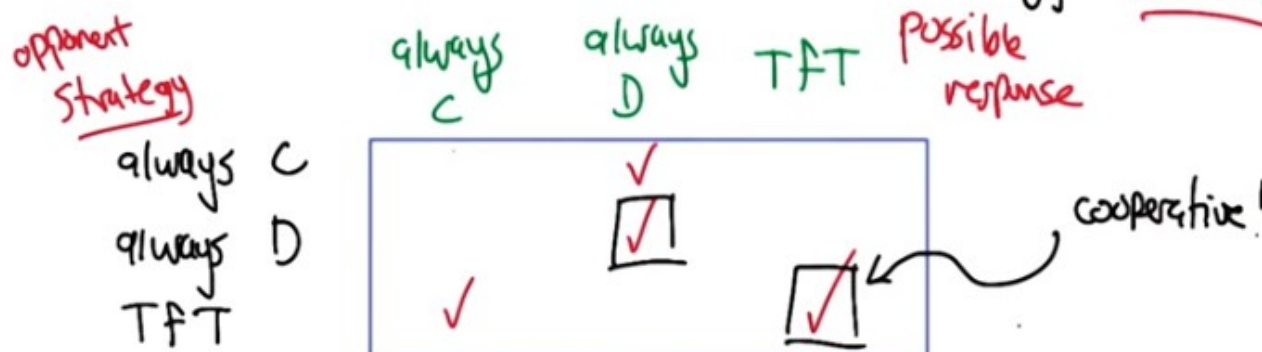
9. Now, we kind of cheated here. Because I told you there's just these, those two strategies. But there's actually a bunch of other strategies you can play against tit for tat. And it's worth thinking through, how do you compute a best response to some finite-state strategy? So tit for tat is a finite-state strategy in that it has these two [LAUGH], these two states. And the strategies expressed in terms of transitions between those two states. But in general, if we have some kind of finite-state strategy, like tit for tat, how do we figure out how to maximize our own reward in the face of playing against that strategy? So in this picture here that I drew, the states are labeled with the opponent's choice, the finite state strategies choice, okay? >> Mm-hm. >> The edges, that's in black, the edges and labeled in green here, are labeled with our choice. So, for example, if we're in this state of the, sorry, if our opponent is in this state. >> Mm-hm. >> We have a choice. We can either cooperate or defect. On this round. >> Mm-hm. >> So, the green arrows tell us how that will impact the state of the opponent. And then these red numbers, I just added the information about well I know that if the opponant is about to cooperate and I choose to cooperate. I can just look up in the pay off matrix that that's a -1 for me. Right? Agreed? >> Agreed. >> So I just annotated all these edges, all these choices with these extra numbers. So, one of the things that's cool about this is unlike just the payoff matrix representation that we had before, our choice, it impacts the payoff, which is the same as that, but it also impacts the future decisions of the opponent. And that gives us this structure here and also says that maybe this is a slightly harder thing to figure out because of the fact that we can't just maximize our, the number. We actually have to think about where that's going to lead us in the future as well. >> So two things then. One, I was always fond of saying that the matrix was all that you needed. But that really only made sense when you were just playing once. >> Yes. That's right. >> Right? And, two, I look at this and it's a finite state machine but you know what else it looks like to me? It looks like an MDP. >> Excellent. It is indeed an MDP. Now, it's a, in this case, my opponent's finite state strategy is deterministic, so it's a deterministic MDP, but it is. It's a discounted MDP. Gamma's playing the role of the discount factor.

The entries (紅色數字) from the payoff matrix are playing the roles of rewards. Our action is playing the choice of our action, and the opponent's internal state structure is playing the role our states. So it is, it's an MDP, and so how do we figure out what an optimal strategy is against a finite state strategy? >> We solve the MDP. >> Yeah, exactly. So any method for solving an MDP can then be used to actually compute the strategy, so what is the strategy going to look like? It is going to be a mapping from states of the opponent to action choices for us. >> Right, but that's fine because a state does not have to be your state, it's just what matters. What matters in this case is what they opponent is going to do. >> Right. So now what are the strategies that can be meaningful against tit for tat? >> So if we cooperate then we're going to stay in this state and it's always going to be the right thing to do to cooperate. So always cooperate is one. If we always, if we defect, now we have a choice again so we could defect from this state which would cause us to defect forever, so always defect is another one. But what's the other thing that could happen? >> Well, we could tit for tat ourselves. >> Well, sort of. I mean, so, we could defect, so we could defect against cooperate, but cooperate against defect. Which would actually cause us to do D-C, D-C, D-C. So those are the only, oh I see. No, you're right. I'm not sure how to say it. But the policy is, defect when you're in this state, and cooperate when you're in this state. But the effect of that is to go back and forth against tit for tat. >> Right. >> Basically, take this loop here. And those are the only policies that matter. And in this case, we worked out that. Always cooperate is good against tit for tat if it has a high discount factor and always defect is better if you have a low discount factor. But, we can get that for real by solving the MDP. >> Right and that makes sense and the reason that those are only 3, let me see if I get this right, the reason those are the only 3 that makes sense because if you think of this as an MDP then it has no history, so when you are in C there are only 2 choices and when you are in D there are really only two choices so if you look at the way you've drawn it. You either stay were you are in c or d, or you take the loop. And those are really the only three options because the rest of them would require you remember what you did, you know, a time step or two ago, and there's no way to do that in an MDP, at least not as you've written it. >> Well, there's a way to do it, it just would never be better than doing it this way. So an MDP always has a mark off deterministic optimal policy. >> Right. >> So we only need to consider those. >> I think that was the same thing that I was trying to say, but with different words. >> [LAUGH] Ok.

10. Alright so, now that we have a handle on what it means to compute a best response against some finite state strategy. Let's actually take a look at these. So, so this is a quiz. Imagine that we have a gamma that is large, so greater than a six. >> Mm-hm. >> What is the best response to each of these strategies? So this will be a quiz, but let me just make it clear what the, the goal of the quiz is. So, if you're, if you are playing against, always cooperate. Which of these is the best response to it, which of these is going to actually have maximum reward? If we're playing against always defect which is going to have maximum reward? And if we're playing against tit for tat which are going to have maximum reward? Any questions about that, does that make sense? >> I'm just making certain I have the rows and the columns right. So my opponent is playing on the rows. >> Yep. >> And so I'm choosing among the columns for each one of those rows. Yes? >> Yeah. So I labelled it best possible response for the columns. And these are the opponent's strategies on the rows. >> Okay. >> Okay, go.

Best Responses in IPD

What's the best response to each strategy? ($\gamma > 1/6$)



Mutual best response. Pair of strategies where each best response to other. Nash!

上圖 matrix 中，黑色的字(即框右邊的)表示對方做的，藍色的字(即框上面的)表示我們的 best response.

11. >> Alright. So is this, is it clear what these answers are? >> Let's find out. So, let's see. we, we already worked out the math on this. So we know that, for gamma, greater than 1/6th, cooperating is better than defecting, in general. So if I'm going against someone who's always going to cooperate, then I should always cooperate. >> Incorrect. >> What? >> Yes, so, so, we didn't actually work that out. What we worked out was what to do if you were playing against tit for tat. If you were playing against someone who was always cooperating, and is completely oblivious to us. >> No, no, no, no, you're right. You should always defect because you're always going to win. Yes, yes, yes. >> You're always going to win. You're going to get zero on every time step. >> Right, right. >> Yeah. >> No, that makes sense. That's beautiful. >> Alright, what about always defect? >> Well, if you're always going to defect, you might as well defect. >> Indeed. >> because we're just in the regular old prisoner's dilemma world. [LAUGH] >> good, alright. So now we, now we have this, this other strange beast here. So our opponent is playing tit for tat. So we could always defect. >> Right. >> But we would do, we'd get a higher score if we can convince tit for tat to cooperate with us. >> For a gamma greater than 1/6. >> That's right. >> So that you should always cooperate. >> That is true, however. >> Mm-hm. >> What if we played tit for tat against tit for tat. >> You'll end up in the same place (因為此時我 tit for tat 其實就是 I always cooperate). >> Yeah, so that's just as good. >> Mm-hm. >> And that's, that's kind of interesting. If you think about mutual best responses. >> Yes. >> So that's a strategy that, a pair of strategies (兩個人都是 TftT) where each is a best response to the other, there's a, we have another name for that. Do you remember? >> No. >> [LAUGH] You taught, you told us what it was. >> Yeah, but I probably used different words. >> It's, it's a Nash equilibrium. >> Oh. >> This, that's what a Nash equilibrium is. A pair of strategies where each is a best response to the other. Each, each there's no way that either would prefer to switch to something else to get higher reward (就算我換成 always C, 也沒有更好, 故 there is no reason for me to switch, 這正是 Nash equilibrium 的定義). >> And

that makes perfect sense. You're in an equilibrium. And that is a Nash equilibrium. Okay. >> So we can use this little table here to actually identify Nash equilibrium. So, what would a strategy be? So, so if one player plays always (對方) cooperate, then the best response to that is always (我) defect. >> Mm-hm. >> But the best response to always (我) defect, is always (對方) defect (即我 defect 時, 對方會想從 cooperate 換成 defect). So always cooperate is not part of a Nash equilibrium. >> Right. >> But what about always defect versus always defect? >> No, it is. >> So that's a Nash. Right? >> Yep. >> Because they're both doing the thing that is the best response to the other. >> Right. >> Alright, this box here, always cooperate against tit for tat. That's not okay (not Nash equilibrium), because if a player does always cooperate, it's always better to switch to always defect. >> Yep. >> But, check this out. If you are playing tit for tat, and the other player's playing tit for tat, there's no reason to switch because it's actually a best response, it's the optimal thing to do. And that works from both players' perspectives. >> And that makes sense. So like you said check this out and you've been using check marks that's very good. [LAUGH]. >> So we're in this situation where we have two Nash equilibria (兩個方框中的). >> Indeed, and one of these Nash equilibria is cooperative (前面我已經說了, 兩人都 TtT 其實就是兩人都 cooperative). Which is the thing that we were sad about, or at least that I was feeling really sad about, in the last lesson. The idea that, man, there's just, it's clear that they should just try to get along (cooperate). You explained that you can modify the reward structure, and then they would get along better. But here it turns out, well, no, another thing you can do is just open it up to the possibility of playing multiple rounds, as long as you don't know how many rounds, it becomes possible to have a strategy that is best off cooperating, and is in fact a Nash Equilibrium. >> Isn't that equivalent to changing the reward structure? >> It's definitely related to changing the reward structure. It's a particular way of changing it. >> A very particular way. >> But it's. Yeah, because it's not true any more that we can do this in the one shot case. You have to be in the, in the repeated game, setting. >> Right, so you change your rewards structure to be a sum of rewards. But it's actually an expected sum of rewards, and you don't know where it is you're going to stop. So I guess you're changed, you're changed the game, you changed the, the rewards. But in a, sort of very subtle way. >> Yeah, the whole game is different, really. >> Yeah man, you changed the game. [LAUGH]. >> Sometimes you gotta change the game. Don't hate the game. >> No, you are supposed to hate the game. Don't hate the player, hate the game.

Repeated Games and the Folk Theorem

General idea: In repeated games, the possibility of retaliation opens the door for cooperation.

What's a "Folk Theorem"? oral tradition.

In mathematics: Results known, at least to experts in the field, and considered to have established status, but not published in complete form.

retaliation: 報復

黑字是該 folk theorem 的 idea(可以理解為該 folk theorem 的內容, 但下段會詳細地講它的內容). 紅字講的是 folk theorem 是一些甚麼樣的 theorem.

12. I'm really proud of us for figuring out a way to make the prisoner's dilemma a little more friendly. It turns out, though, that this, this idea, this sort of core idea, is very general and kind of cool. So, it leads us to a topic that we could call, repeated games, and the folk theorem. >> Mm-hm. >> So the general idea here is that when you're in the repeated game setting, this possibility of retaliation, the possible of, you know, not being cooperative actually opens the door for cooperation. Because now it becomes, it can become better to just get along and cooperate than to get retaliated against. >> Right. And that makes sense as long as the retaliation is plausible. >> Well, we're not talking about plausibility of the retaliation. We're just talking about, right, because the tit for tat, it's, it's not analyzed in that way. It, it doesn't say whether or not it's actually plausible. It just says, well, if you have this strategy and you play against this strategy, there's no incentive to switch. >> Right. >> But we'll, yeah, we'll get into this plausibility thing in a little bit. >> Oh, I, I stumbled across a word. Okay, go. >> [LAUGH] But I want to point out one thing first, which actually kind of irritates me. Kind of along the same lines as like, regression is the wrong word and reinforcement is the wrong word. Folk theorem is the wrong word. So, what is a folk theorem. >> It's two words. >> In general mathematics, sorry say again? >> A folk theorem is two words. >> That's fair, yes. So, I think it's actually two different terms of art. So in mathematics folk theorems are results that are known, at least to experts in the field, and they're considered to have established status but they're not really published in their complete form. There isn't some kind of original publication saying, oh look, I, I found this thing. It's more, something that like, kind of everybody knows, and so you don't really give anybody credit for it. So it's like a theorem that is in the general mm, cloud of understanding. It's sort of among the population, among the group. >> Wait! Does, does anybody every prove these folk theorems? >> You can, yeah! All folk theorems are provable. But it's not like you say you know this is Charles Isabel's theorem. >> Mm. >> It's like. No, it's just a folk theorem. It's like. Charles, Charles can prove it. We can all prove it. But we, we're not really sure whoever proved it first. It was just one of those things that everybody knows. >> Oh, so it's like an oral tradition. >> Yeah, yeah. I think that's a good way to think about it. >> Okay. I'm just imagining mathematicians sitting around a fire in the winter, cuddled up against one another sharing theorems that have been proved since the beginning of time to one another. >> Exactly. Right that's the image that I have as well. This, just it's folk theorems. It's this sort of, you know, I learned it from my grandmother and now I'm telling you. >> Hm. I like that. I like to think of mathematicians that way. >> [LAUGH] As grandmothers. >> Yes. >>

Folk Theorem

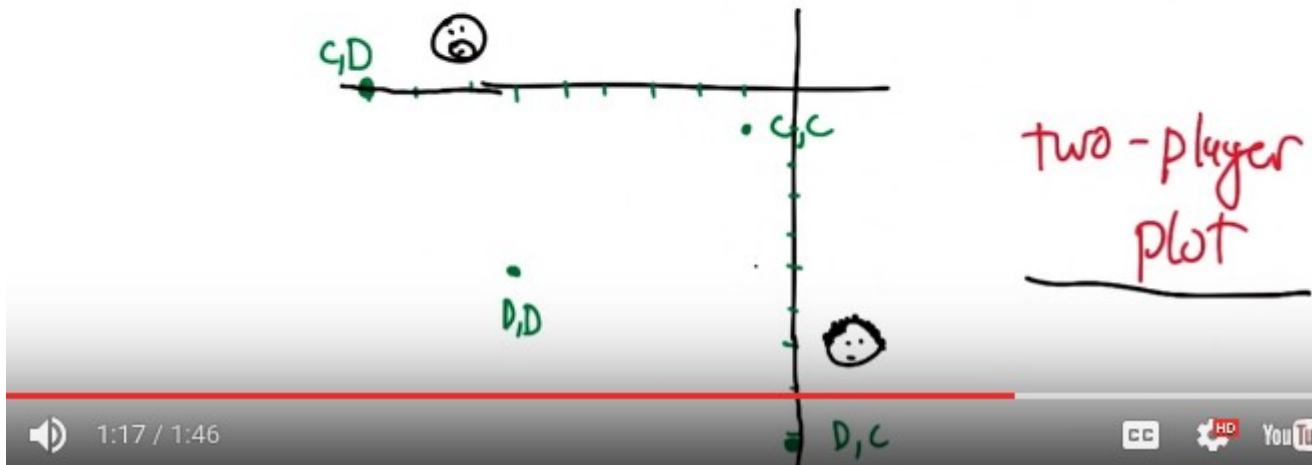
In game theory, though, Folk Theorem refers to a particular result:

Describes the set of payoffs that can result from Nash strategies in repeated games.

So that's what a folk theorem is in mathematics. However, in game theory it means something different. It is referring to a folk theorem, but it's also referring to a particular folk theorem. So the folk theorem in game theory refers to a particular result that describes the set of payoffs that can result from Nash strategies in repeated games. So it's a funny thing, it's a funny way that they use the word. It's, it's a, it's a folk theorem but it's also the folk theorem. >> Hm, well, well, maybe it was actually, proven the first time by some guy named Folk. >> That's a good idea, we should probably, we should push that forward like call him Folke or something like that. Folke's theorem >> Yeah. >> But but no. [LAUGH] >> Oh well, I tried. >> Yeah that's that's really great idea that we're going to just move on from. >> [LAUGH] >> But what I, what I'd like to do next is kind of build up to this notion of the Folk Theorem and it's going to require a couple basic concepts that are not too different from stuff we've already talked about, but we're going to kind of make them concrete and then we're going to show how they all come together to provide us with this idea of a Folk Theorem. >> Okay.

Repeated Games and the Folk Theorem

General idea: In repeated games, the possibility of retaliation opens the door for cooperation.



13. >> The thing that I find most useful in trying to understand the folk theorem and, and what it says and how it works is a thing that I call a two-player plot. I don't know if other people have other names for it, but this, this concept is out and, and often discussed. I just don't know what it's called [LAUGH]. But here's the idea of it, it's a really simple idea. >> It's like a folk plot. >> It is a, it is a folk, a folk plot, right, I don't know who invented this plot. So here's what we're going to do, remember this prisoner's dilemma game. We've got two players, there's Smoove and Curly. >> Mm-hm. >> And, what we're going to do, is we're, there's a bunch of joint, actions that they can take. So Smoove cooperates and Curly defects, or they both defect, or they both cooperate, or one defects defect cooperate in the other direction. So what we're going to do is for each of those joint outcomes, each of those joint action choices. I'm going to plot a dot, put a dot on a two-dimensional plot, where this is the Smoove axis, and this is the Curly axis, okay. >> Okay. >> So cooperate-cooperate, remember from the prisoner, prisoner dilemma payoffs, is minus one, minus one. So I put a dot at minus one, minus one. Defect, defect. Is it minus six, minus six? Cooperate defect is it minus nine zero? And defect cooperate, is it zero minus nine? . So do those four points make sense to you? Do you understand like the idea? >> They do. >> It may not be so obvious why we do, do it this way, because as you have told us, the matrix is all you need. >> Mm-hm. >> But this is really just representing the matrix in another form. But it actually is sort of losing some information. Because these dots don't tell us what the relationship is in terms of if, if one player keeps the same action and the other player changes to a different action. The matrix captures that but this plot doesn't anymore, it's kind of washed out. >> I see.

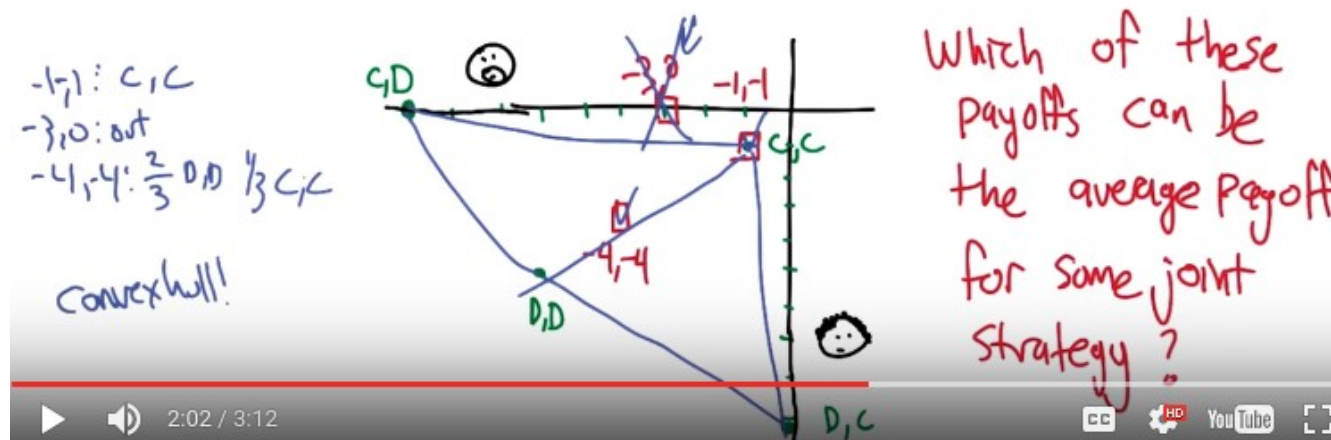
(本段的圖跟下段的圖是一樣的)

14. All right. Now so just to kind of reinforce this idea and also to let us think about it in a slightly different direction, I want you to consider solving the following quiz. Here's four payoffs that I've put

little boxes around. This one at minus 1, minus 1, this one at minus 3, 0, and this one at minus 4, minus 4. And the question is, which of these payoffs can be the average payoff for some joint strategy? And this is in the repeated game setting, [CROSSTALK] okay. So what we're imagining is these two players can coordinate in any way they want. We're not talking about trying to maximize reward or, or, being Nash or anything like that. They're just going to execute some, some pattern of, of strategies over infinite run. We're going to average the payoffs that the two players get and we'll say, you know is it possible for them to adopt a strategy so that the average per time step gets minus 3 for Smooth and 0 for Curly? Yes or no? So check that box if that's possible. Can it be that they both get minus one on average? If so, check this box. And can it be that they both get minus 4 on average? Check this box. Does that, does that make sense? >> It does make sense. >> All right, so I'll give you a chance to think it through and answer it. >> Okay.

Repeated Games and the Folk Theorem

General idea: In repeated games, the possibility of retaliation opens the door for cooperation.



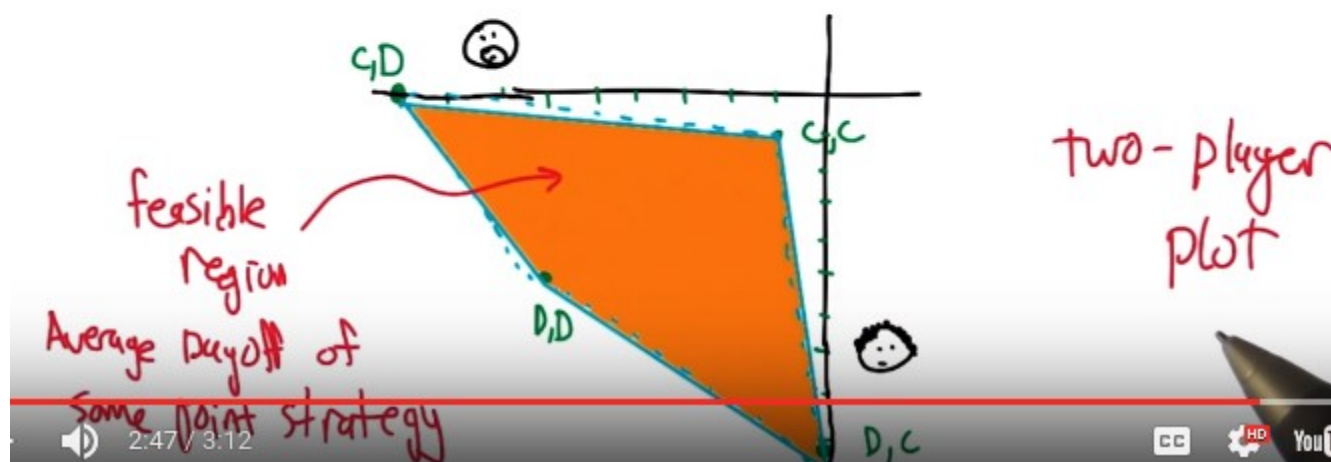
上圖左下角應為-4,-4: 3/5 D, D, 2/5 C, C. Instructor Notes 中說了。

15. >> Alright, so are any of these easy to answer right off the bat? >> Well, one of them is really easy to answer right off the bat, and that's $(-1, -1)$. >> Good. >> Because that's one of the points, so. [LAUGH] >> Yeah. So, the joint strategy to get the minus one minus one would be for both players to cooperate, right? Again, they're just, you know, they're just willing to do that just for the sake of showing they can make that value. >> Right, and I think I know the answers for the other ones as well >> Alright, hit me. >> So here's how I'm going to, here's how I'm going to, going to talk you through my reasoning so that if any point I'm wrong, you can gently steer me away from embarrassing myself. Basically I was looking at these points and I thought hm, they form a kind of a convex hull. >> Aha! >> And I thought well, surely that's just an accident of the numbers and then I thought, oh wait, of course we're talking about averages here. So that means that all sort of possibilities have to be inside the convex hull of the outer points. So if I drew a line between those four points, I would end up with all possible, achievable averages. >> What, achievable averages, how? How would you achieve things inside the convex hull? >> I would appropriately average them. So in particular, the first thing I'd

notice that $(-3, 0)$, is outside that. >> Mm. >> So, it can't be, it can't be something you can achieve. >> Right, there's no way to make this minus three, zero, certainly not by choosing any of these points, but also no combi, no convex combination, no probabilistic combination of them, is going to do that either. >> Right. >> So, this one is just right out. >> Right, so that leaves $(-4, -4)$, and it seems pretty. That's inside the convex hull, so there is some combination of them that would work. >> That's right. Any, any sense of what it would be? >> 2 3rds, I think 2 3rds D,D 1 3rd C,C. And I get that by noticing that D,D is minus six, minus six, and C,C is minus one minus one, and four minus four is two thirds of the way between there. >> Boom. Cool. Alright, so you're good with that? >> I'm good with that if you're good with that. It's your quiz. >> Yeah. I'm excited. >> Oh good. >> So those, it's this, this, the minus one minus one and the minus four minus four are, as as you pointed out there is a more general result here. Having to do with the convex hull. >> Right.

Repeated Games and the Folk Theorem

General idea: In repeated games, the possibility of retaliation opens the door for cooperation.



>> Alright, so through the magic of computer graphics, I have a slightly better depiction of this particular region now. As you pointed out this, this convex hull of the points is really important and what it represents is the, we can call it the feasible region, these are average payoffs of some joint strategy they, they may have to collude to do this. And they may not be particularly happy to do that. [LAUGH]. But the fact of the matter is, they can achieve, by working together, payoffs anywhere inside this region. >> Huh, so that's what my student Liam always meant when he talked about feasible regions. >> Maybe so, I mean it could be that he meant any number of other things like places he's willing to live when he goes to get a job. >> No, it was all game theory stuff, I just never knew what he was talking about, but now I do Michael, thanks to you. >> Sure, hey I'm, I'm happy to help. So this is a really useful kind of geometric way of picturing something that would otherwise be a little bit harder to see, I think in the matrix form. >> Sure.

Minmax Profile \rightarrow minmax: pure
 \rightarrow security level: mixed

Pair of payoffs, one for each player, that represent the payoffs that can be achieved by a player defending itself from a malicious adversary. Zero sum game!

(mixed)

trying to lower my score.

Minmax profile of BS?

$x = \frac{2}{3}$ $1-x = \frac{1}{3}$

Bach, Stravinsky

Battle of Sexes

Backstreet, Sting

$x = 2(1-x)$ $x = 2 - 2x$ $3x = 2$ $x = \frac{2}{3}$

	B	S
B	1, 2	0, 0
S	0, 0	2, 1

$\frac{2}{3}$ $\frac{2}{3}$

上圖的 4*4 matrix 中, 數字(如 1,2)前面的表示 Curly 的, 後一個表示 Smooth 的. 答案(1*2 matrix 中)也是這樣表示的.

16. >> All right, the next concept that we're going to need to understand the folk theorem is the notion of a minmax profile. So a minmax profile is going to be a pair of payoffs, one for each player. And the value for player represents the payoffs that that player can achieve by defending itself from a malicious adversary(敵手). So what do you suppose a malicious adversary would mean in a game theory context? >> Someone who's desperately trying to hurt you. >> And what does hurt mean? >> Gives you the lowest score. >> Yeah, and what does that remind you of from your lesson? >> My grad students. >> You think they're malicious? >> It would explain a few things. >> Yeah, I don't think they're malicious. >> They're sweet. >> [LAUGH] Yeah, I know a lot of them and they're, they're wonderful people. >> Well, what it reminds me of is, they are wonderful people. It reminds me of zero-sum games. >> Exactly. So you can imagine thinking about the game that we're playing, now, no longer as being I get my payoff and you get yours, but I get my payoff and you get the negative of my payoff. So you don't, you don't really care about yourself anymore. All you care about is hurting me. And that's, that's the idea of a malicious adversary. >> I have some ex-girlfriends like that. >> I'm so. Oh. [LAUGH]. It is. People do get into this mode sometimes. And that, that's actually going to be important in understanding the folk there. >> Hmm. >> So what I'd like to do is figure out what the min-max profile is for this game. So this is a very famous Game theory, game example. Sometimes goes by the name battle of the sexes. >> That what the b and the s stand for? >> Sometimes it stands for Bach and Stravinsky. >> Blech. >> Those are like composers, I think. >> You mean like the Backstreet Boys and Sting. >> Ahh. Alright, that works for me. So, so, let me explain this story. It turns out that Smooth and Curly actually got away, they didn't make any kind of deal, they actually just figured out a way of escaping from the jail. So they're, they're back out on the streets again, and they decided that they'd like to celebrate their freedom by going out to see a concert. And they both decided that in advance, but what they didn't know was which concerts were available. Once they escaped out into the world, they couldn't communicate with each other, they discovered that there's in fact two concerts in the city that night. The Backstreet Boys are playing, and Sting is playing. >> Okay. >>

Now as it turns out, each of them is now going to have to choose, whether to go to the concert with Backstreet Boys or Sting, and they're choosing independently. Now, if they end up going to different concert's they're both going to be unhappy and get zero's. >> I see. >> If they end up at the same concert, then they're going to be happier, but in fact, as it turns out, Smooth really likes the Backstreet Boys and would prefer that they both end up at the Backstreet Boy concert. But Curly really likes Sting and would prefer that they end up at the Sting Concert. >> That's not realistic. >> Which part? >> The fact that I prefer the Backstreet Boys to Sting. >> What, what do you mean you? I mean this is Smooth. He's a criminal. >> Mm, that's a fine point that you make there. There is no connection between these characters and ourselves. >> Real life characters, living, dead or fictional, or mathematical, or instructional. >> If so, otherwise purely coincidental. >> Yeah. I could switch these around if you'd prefer. >> No, no. I'll go with your fantasy. >> [LAUGH] All right. I, yeah, I think that payoff matrix may look something like we both have twos in, in the s, the same place. >> Mhm. >> But anyway, but let's say for the purposes of this example, there's a little bit of a disagreement. Okay, so now what we need to figure out is what the minmax profile is for this game. >> Okay. >> Alright, so that's going to be a pair of numbers. >> Mm-hm. >> One number corresponds to the payoff for Curly and one number corresponds to the payoff for Smooth. And it should be, the payoff for Curly should be the payoff that Curly can guarantee himself even if Smooth is trying to get him to have a low score. >> Okay. >> And vice versa. Smooth's score is going to be the score that it can guarantee, he can guarantee himself even if Curly is trying to minimize Smooth's score. >> All right. >> So let's do this as a quiz. So, I want you to find the min-max profile for this game, this Bach, Stravinsky game or Backstreet, Sting game, and put the number for Curly in the first box and Smooth in the second box. >> Okay. >> Go.

Minmax Profile → minmax: pure
✓ security level: mixed

Pair of Payoffs, one for each player, that represent Zero sum game!
the payoffs that can be achieved by a player
defending itself from a (mixed) malicious adversary. ← trying to lower my score.

Minmax profile of BS? B S

	X $\frac{2}{3}$ ⊗ 1-X $\frac{1}{3}$	
B	1, 2	0, 0
S	0, 0	2, 1

Bach, Stravinsky
Battle of Sexes
Backstreet, Sting

2(1-x) x = 2(1-x) x = 2 - 2x
8x = 2 x = $\frac{2}{3}$

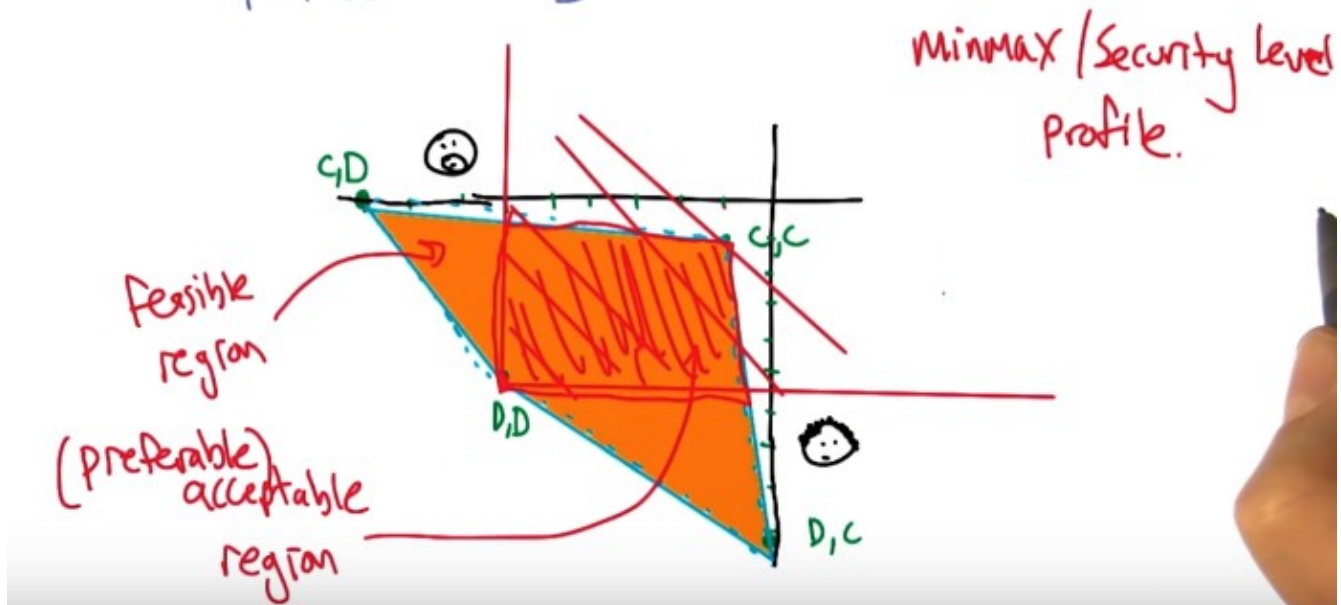
2/3 2/3

(這裡將上圖 copy 過來, 好方便看)

17. Alright, what'd you got? >> I'm going to say it's (1, 1). >> Alright, how would you figure that out? >> Well first, I would ask you if that's correct. [LAUGH] >> [LAUGH] We'll see when you try to. >> [LAUGH] >> Figure it out. >> Okay, so, the idea here is I'm trying to figure out what Curly would get,

if Smoove was out to get, make certain that Curly got the worst possible value. So, Curly gets to choose among rows. And Smoove gets to choose among columns. So, If I (Smoove) chose the B column, then Curly could get a one. You're given choice Curly would go for the first row instead of the second row, because Curly would get a one. If I took the S column, as Smoove, Smoove took the S column, then Curly would choose the second row, and would get two. And one is lower than two. So, Smoove would choose the first column and Curly would have to get the first row and would end up with a one. That make sense? >> Yes, but here's what you haven't figured out. >> Yes? >> What happens if Smoove, no because, because Smoove says, I'm always going to choose the Backstreet Boys. >> Mm-hm. >> Then you're right, Curly should also choose that and at least coordinate. >> Mm-hm. >> But what if Smoove is random say half and half random between the two options. >> And I don't know what's going to happen before then? >> And you don't know what's going to happen, sorry? >> I know that, I know it's half Curly knows it's half and half but doesn't know which one he's choosing any given time? >> Exactly. >> I see. Then, I would end up with, one half and one. >> Then for choosing B. >> Uh-huh. >> Expect the score would be a half, and for choosing S, they'd expect the wor, expect the score would be one. >> Yep. >> Oh, I see. And, and then Curly would choose S and get the one. >> Right. >> And that's still consistent with, with that? >> Mm-hm. >> Can we work out what the what the worst case is? >> What do you mean? >> Well I feel like, we should solve it like a [INAUDIBLE] game, like we did in your lesson. >> Oh. >> Say that Smoove is actually choosing a probability either X or $1 - X$. >> So I thought you were asking for like a pure decision, I wasn't even thinking about mixed decisions. >> Uh-huh, yes, it could be a malicious and randomized adversary. >> [SOUND] So we are. >> [LAUGH] You said yuck [LAUGH]. >> We, we are really talking about my ex-girlfriends. Okay so you just do the math. Do the math. >> Alright, so if we, if Smoove chooses Backstreet Boys with probability X and Sting with probability of one minus X . >> Mm-hm. >> Then, Curly for choosing Backstreet Boys will get X on average and for choosing Sting, we'll get two times one minus six on average. And the useful point is going to be to discover when these $(x \text{ 和 } 2(1-x))$ are equal to each other. >> Yep. >> So in fact, Smoove, by being malicious and stochastic, can actually force things down to 2 3rds. >> Hm. And things being symmetric as they are, Curly can do the same. >> Okay. >> So, basically, Curly can behave in such a way, that even against a malicious adversary, it could, he could guarantee himself to a score of $2/3$. >> Yeah, a malicious possibly mixed adversary. >> That's right. >> Okay. But one, one would be right if we were sticking with pure strategies, but why would we do that? >> That's right, but for the purpose, if that's right, and you can do a version of the folk theorem. In fact, there's lots of different flavors of the folk theorem. The one that we're going to focus on is going to allow for these mixed strategies. >> Mm-hm. >> But. In fact in general you could say you know, no I kind of like the mixed strategies, let's just stick with that. >> No, I like the mixed strategies too, I just wasn't thinking about them. >> So, I want to point out that in fact the solution that you gave $(1, 1)$, I think that actually does correspond to what is usually called Minmax, which is the pure strategy. >> Yeah! >> So the minmax is in fact $(1, 1)$. The other concept is really important too though. And I think it's sometimes called the security level profile. So instead of the min max level profile the security level profile. And that allows for the possibility of mixed strategies. So that gets you down to the $(2/3, 2/3)$. I think you know, it turns out that there's folk theorems that can be defined with with either of those concepts (即 minmax profile 和 security level profile). I prefer this one. But I do like this name better [LAUGH]. So I, I apologize if that made things confusing. >> I'm not confused now. I think in the end Michael, the important thing is we were both right. >> Well, the example that I'm going to next, these two concepts line up. So let's let's do that, and then we don't have to care. [LAUGH] >> I, I'm all for that. Let's do that.

Repeated Games and the Folk Theorem



18. So, here we are, back at the prisoner's dilemma, again. You may recall this picture. >> Vaguely. >> Let's add to this, the minmax, or security level profile. So, for prisoner's dilemma, what is the minmax? >> Isn't it d comma d ? >> It is indeed. (D, D) . Right so this is value that you can guarantee yourself against a malicious adversary. Malicious adversary is just going to defect on you, and the best thing you can do in that case is defect yourself. >> Yep. >> Agreed, agreed? >> Agreed. >> Alright, so now let's take a look at the intersection of the following two regions. There's this nice yellow region that we've already got, and then we've got a new region that's defined by this minmax point. This minmax profile. So the region that is above and to the right of this, of this minmax point. >> Mm-hm. >> So, the, that's the region. This, this region, alright we already said that this yellow region is called the feasible region. >> Mm-hm, or orange or whatever color it is. >> So, I'm thinking we can call this other region (紅線組成的 region) the acceptable region. And it, and, the, what I mean by that is if you think about it, payoffs in this region are, Smooth, getting more than what smooth can guarantee itself in an adversarial situation. And Curly getting more than Curly could guarantee himself in an adversarial situation. So, these are all, like, you know, better than it could be. >> So, why not call it the preferable region? >> The preferable regions, preferable to not being in this region. >> Mm-hm >> The intersection of these two Is the feasible preferable acceptable region? >> [LAUGH] Exactly. It's kind of, you know, special, from the perspective that it is both feasible and preferable. And now we are ready to state the Folk Theorem.

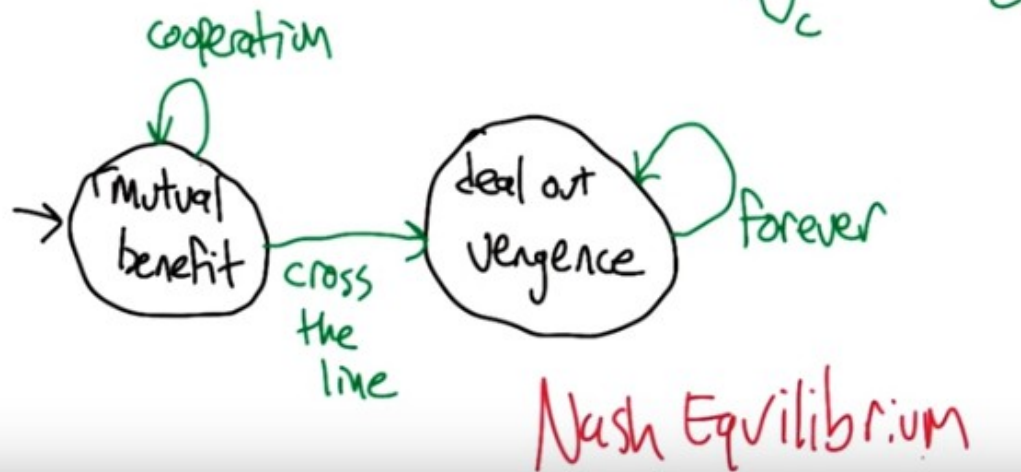
Folk Theorem

Any feasible payoff profile that strictly dominates the minmax/security level profile can be realized as a Nash equilibrium payoff profile, with sufficiently large discount factor.

Proof: If it strictly dominates the minmax profile, can use it as a threat. Better off doing what you are told!

19. So here's the Folk Theorem. Any feasible payoff profile that strictly dominates the minmax or security level profile can be realized as a Nash equilibrium payoff profile, with a sufficiently large discount factor. >> Prove it. >> What we're going to do is we're going to construct a way of behaving where both players are going to play so that they achieve this feasible profile. And the reason to make that a Nash equilibrium, what we need to do is make it so that it's a best response. And the way that we are going to make it a best response is we're going to say do what you're told. Follow, follow your instructions to achieve that feasible payoff. And if you don't, then the other player is going to attack you, is going to adopt a strategy that forces you down to your minmax or security level, and that's your threat. So the best response to that threat is to just go along and, and and do what you're told to achieve that feasible payoff. The only way that that's going to be stable though is if the thing that you're asked to do, the feasible payoff, is better than the minmax, right? Because that has to be a threat. You can't threaten somebody and say, you know, do this or I'm going to give you candy. It's gotta be do this or I'm going to do, give you something that's less pleasant than what I've asked you to do. >> Okay, that actually makes sense. >> Yeah, so this is, this is a really cool idea. >> I like it. Hey, could you try saying that again, but with a Southern accent, just the Folk Theorem part? >> Any feasible payoff profile that strictly dominates the minmax/security level profile can be realized as a Nash equilibrium payoff profile with sufficiently large discount factor. >> I like that because now it's a folksey theorem. [LAUGH]

Grim Trigger

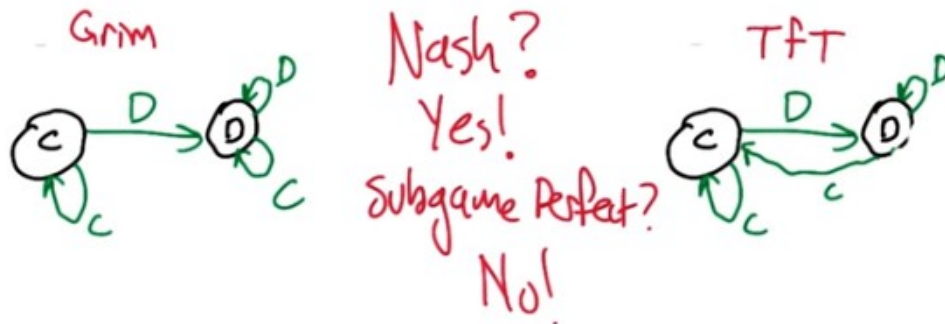


Nash Equilibrium

20. So another way to think about the proof of the folk theorem, is you could prove it with a little strategy that is referred to as grim trigger. >> Mm. I like that. I like the way it sounds. >> So here's, here's the basic structure of grim trigger. It says that what we're going to do, is we're going to start off, taking some kind of action or pattern of actions, it's a mutual benefit. And as long as it's cooperation continues this mutual beneficial, behavior will continue. But however, if you ever cross the line, and fail to cooperate with me. Then I will deal out vengeance against you forever. >> Hm, so once again, we're talking about my ex girlfriends. I don't know why you're obsessed with this. >> [LAUGH] Maybe it's just, maybe it's just trying to help you understand. So you know kind of what this situation is. Anyways here's, here (右上角) is what that looks like in the context of prisoners dilemma. Alright so cooperation is the mutually beneficial action. >> Yes. >> And as long as you continue to cooperate with me, that's this C arrow here, then I will continue to cooperate with you, but if you ever defect on me, I swear I will spend the rest of my life making you pay. So no matter what you do at this point, defect or cooperate, I will just continue to defect on you. Pain will rain from the sky. >> Okay, well this makes sense. So, the idea here is that if you know that I'm going to do this, then hopefully it makes sense for us to continue to mutually benefit. >> Right. So the whole purpose of this is to create a Nash Equilibrium System kind of situation. Right? Where if I'm playing this strategy. And you're playing this strategy, then neither of us has any incentive to cross the line, and so we're just going to continue to cooperate. Crossing the line is going to decrease your reward, so there's no benefit to doing it, so you won't do it. So it, it's nice because it gets us a Nash Equilibrium. >> Hm. >> But there's a problem with it. >> Of course. >> And you pointed it out before, so let's dive in and make sure that we understand it and see if we can fix it. >> Okay.

Implausible Threats

Subgame perfect : Always best response independent
of history.

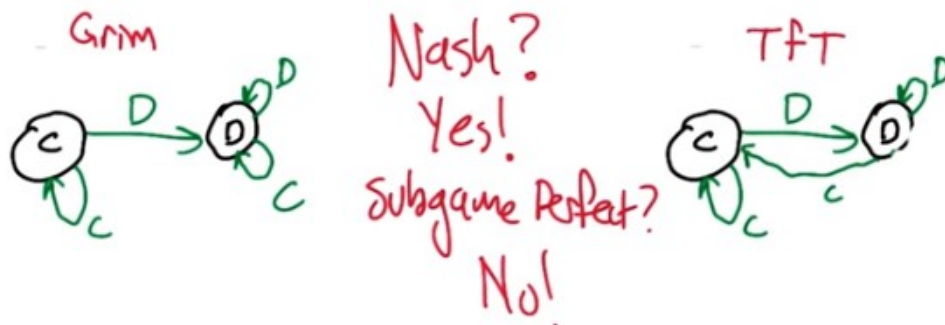


21. The problem is that in some sense, the threat is implausible (難以置信的). And it, and it, in a very kind of real sense. So what's happened is that if you do fake out on me, if you do cross that line, the idea that I will then spend the rest of my days punishing you, forgoing(放棄) reward myself, right? Not taking the best response against you, seems kind of crazy. Do you agree? >> Yeah, again, my ex-girlfriend. Yeah, I totally get this. >> No, but no, I'm saying that nobody would do that. >> Right, so it, it would be like being in a elevator with a stick of dynamite and seeing that someone has a hundred dollars and saying, give me your hundred dollars or I'll blow us both up. That's not really a reasonable threat because the alternative to me not giving you a hundred dollars is, you die, which seems probably not worth it. >> That's right, so you could think about the possibility of, okay, I'm going to not give you the \$100, you say that you're going to blow me up, but you will hurt yourself more than you hurt me, so it won't be a best response. Not blowing me up and just not getting the \$100 and leaving the elevator is better for you than blowing me up. >> Right. >> So that is an implausible threat. So the way we formalize this idea, in the game theoretic context, is to say that we're interested. A plausible threat corresponds to something that's called a subgame perfect equilibrium. >> Okay. >> So subgame perfect means that each player is always taking a best response, independent of the history. All right, so let's actually look at a concrete example here, let's imagine playing Grim Trigger, against Tit for Tat (意思是一個犯人用 Grim, 另一個犯人用 Tft). So my first question to you is, are these two strategies in Nash equilibrium with each other? >> Yeah I guess so. >> And why is that? >> Because, the fact, if I'm playing Tit, if one player is playing Tit for Tat then the Grim Trigger thing doesn't matter anyway because both of you are going to cooperate forever and it doesn't make any sense to deviate. >> Right, so any strategy that I could choose that's different than Grim Trigger is going to on average do no better, possibly worse. >> Right. >> So I might as well stick with Grim Trigger and Tit for Tat has the same kind of feeling about it, that, it's cooperating with Grim. And any, it can't really do anything better so it might as well do that. >> Right. >> So the next question to ask is. Are these two strategies in a subgame perfect equilibrium with each other? And the way that you actually test that, is you say, well, they are **not** subgame perfect if there's some history of actions that we could feed to these machines, so that, so that, you know, here's, here's what Grim is doing. It's some sequence of cooperates and defects, and here's what Tit for Tat is doing. It's some sequence of cooperates and defects. And once we've

reached some particular point. Is it the case that one or the other of these machines is not taking a best response that it could actually change its behavior away from what the machine says and do better than what the machine says. If that is the case then it's not subgame perfect. But if it's the case of all histories, they're always taking a best response, then it is subgame perfect. So, so do you see a sequence of moves that these two players can take where one or the other of them is not going to be doing a best response?

Implausible Threats

Subgame perfect : Always best response independent of history.



(將上圖 copy 過來了, 好方便看)

>> It's, can take, right? As opposed to, will take. >> I don't understand. >> Yeah, I'm not sure I do either. That's why I asked the question. It's not a, you know made up history, it's like an actual set of moves that are consistent with Grim and Tit for Tat. >> No no, no no, so it is, it is not necessarily. So we know that if we actually play these against each other the only history that we're going to see is >> Cooperate forever. >> Right, Grim is going to do cooperate cooperate cooperate cooperate..., Tit for Tat is going to do cooperate cooperate cooperate cooperate..., And so they are, and everything's fine. The question is, can we actually go in and alter, the history, so that one or the other in the machines could take a better action than the one that the machine tells it to take. >> Yeah if Tit for Tat, ever does defect. >> Alright, so let's take a look at that. So, let's say, on the first move Grim cooperates and Tit for Tat defects. Okay, so let's say that, that's the moment in time. What will the machines do at this point? >> Well, at this point the and next time step, Tit for Tat will cooperate and Grim will defect. >> Good and then thereafter. >> Grim will always defect. >> And then Tit for Tat will always defect. >> Right. >> So the pay off that Grim gets at this point (Grim D, Tft C) is going to be, well initially high but then very very low. >> Mm-hm. >> On the other hand could Grim have changed its behavior to do better than this? >> Yeah. Just by doing just, by choosing to cooperate. >> By choosing to cooperate, so it sort of ignore the fact that, that Tit for Tat did the defect, and instead (Grim) do a cooperate here (即(Grim D, Tft C)處, 改為(Grim C, Tft C)), then Grim would do better. So the idea is that Grim is making a threat, but when it comes time to actually follow through on that threat, it's actually doing something that is worse for itself. Then what it would do otherwise. Do, do you see that? >> I do. >> So is it subgame perfect? No. And the proof of that is exactly, exactly what you said, Take,

take a look at this history. Here's a history where Grim would not be willing to actually follow through on its threat. >> Right. >> So it's an implausible threat, and that's bad. So maybe we've now just undone(毀滅, 廢除) all the awesomeness that we had done. >> No. >> Well maybe. I mean the awesomeness was hey look we can actually get machines that are in nash equilibrium and they're cooperative in, in prisoner's dilemma so they're actually kind of doing the right thing. And, turns out well they are, but they're depending on this notion of implausible threats to do it. >> Mm, how should I feel about that? >> Well, let's see if we can fix it. >> Okay.

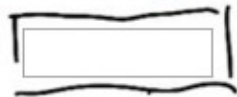
Is TFT vs. TFT subgame Perfect?

 Yes

 No

Here is a sequence that proves it:

TFT #1



TFT #2



22. All right. So let's make sure that we get this concept. So let's evaluate, tit-for-tat versus tit-for-tat, spy versus spy, and ask whether or not they are subgame perfect, or in a subgame perfect equilibrium with each other. And your choices are yes and no. >> Mm hm. >> But if you say no, I'd like you to give me a sequence to show that it is not subgame perfect. In other words, that if they were to take this sequence of actions and this one was to take this sequence of actions, it would leave the machines in a position where they would not be willing to follow through on their threat, that it would be better for them to do something else in the long run, assuming that they're still playing against the other tit-for-tat machine. >> From that point on. >> Yup. >> Just imagine that someone could go in and change one thing. Would you still want to follow tit-for-tat the next time step or not? >> Yeah, I don't know if that has to be just one thing, but yeah. That's right. Change, change the sequence leading up to this point and then say, okay, do you still want to do what tit-for-tat is telling you to do, or would you rather do something else? >> Right. Okay. I think I got it. >> Cool. All right. Go.

Is TFT vs. TFT subgame Perfect?

□ Yes

□ No

Here is a sequence that proves it:

TfT #1

D
C

TfT #2

C
↑D

D C D C ...
C D C D ...
0 -9 0 -9 ...
-4.5

23. Okay. What's your answer? >> My answer is no. >> No, I'm sorry, that's wrong. No, no, I think that, well, if it's, if it's right, then we need to provide a sequence that proves it. >> Okay. >> So what are you thinking? >> Well, I was thinking actually something very similar to what we just saw. >> Okay. >> Where so tit for tat, what they're going to do is they're going to do cooperate, cooperate, cooperate, cooperate, right? That's what they normally want to do. >> Exactly. >> So what would happen if at one point, one of them defected? >> Okay, just for simplicity let's make it the very first point. >> Mm-hm. >> So, tit for tat number one defects, and tit for tat number two also defects. >> At the very first time? No, it cooperates, because it's done at the same time. >> Well, I mean so we're, we can feed it anything we want. So we could tell tit for tat two to cooperate. So it's sort of like we've taken over its brain for a brief amount of time. >> Right. So I'm not yet convinced it's going to matter for this, but the thing is that from that point on, tit for tat two is going to want to, for the next step tit for tat two is going to want to defect (即 TfT #2 是 CDCD...). >> That's right. >> And, tit for tat one would want to cooperate (即 TfT #1 是 DCDC...). >> Uh-huh. >> And that's sort of. Sucks. >> [LAUGH] >> For tit-for-tat two, right? >> So, I don't know, so maybe we should try to think through. What is the expected reward, for TfT #2 (實際上寫的 0 -9 0 -9 是 TfT #1 的), to actually do this defect at this time? >> Or wait, no. So, so, sorry to, to stick with the tit-for-tat machine at this time? >> Well, what's going to happen at that point is, it's going to keep alternating. >> That's exactly right. So it's going to get the, the rewards corresponding to D versus C, C versus D, D versus C, C versus D. >>

Over and over again. >> Yeah. So let's thi, let's think about it in the average reward case. So, in the D versus C, if it does D when the other machine does C, then it gets zero. >> Mm-hm. >> If it does C when the other one does D it gets minus nine. >> Mm-hm. >> And then this alternates. So if we look at the average award, which is basically what you get when the discount factor's very, very high. >> Mm-hm. >> It's scoring -4.5. >> Right.

Is TFT vs. TFT subgame Perfect?

☐ Yes
☒ No

Here is a sequence that proves it:

TfT #1

D

TfT #2

C

D C D C ...
C D C D ...
0 -9 0 -9 ...

-4.5 vs -1

>> Is there any way, that it could behave against tit for tat, starting from this point that would do better than -4.5. >> Just go ahead and cooperate. >> Just cooperate forever? >> Well cooperate the next time and then keep doing tit for tat from that point on. It'll work out to be cooperate forever (即 TfT # 2 是 CCCCC...). >> On average, that's right, what will get is a -1. So not being tit for tat at that point (因為若是 TfT, 就得跟著 TfT #1 做 D) but instead, instead turning always to cooperate would actually get it better. So the idea that is should 'defect at this point' is an implausible threat. >> Exactly. >> So this is not sub-game perfect. So yes, you nailed it. >> Yeah! >> Does that make sense? >> It did make sense. >> Good, alright. So that leaves open the question of, is there a way, to be sub game perfect in Prisoner's dilemma?

Is TFT vs. TFT subgame Perfect?

☐ Yes
☒ No

Here is a sequence that proves it:

TFT #1
D D D D D

TFT #2
D D D D D

(-1, -3)

D C D C ...
C D C D ...
0 -9 0 -9 ...
-4.5 4.5

>> Can I ask you a question? >> Sure. >> Before you answer that. So, I had sort of convinced myself that it didn't matter whether tit for tat number two started out with C or started out with D. I'm trying to decide whether that's actually true. >> 'Kay, that's a good question. So what will happen, at this from this point on if we now continue. We, you know, we took over the brains for tit for tat, and we forced them to play defect against defect. >> Mm-hm. >> And now we release that, and we let them do whatever it is that they're going to do. And what is th, what are they going to do? Is [CROSSTALK] >> They would defect forever [CROSSTALK] >> Defect forever. >> Yeah. >> And is there anything that tit-for-tat two machine, could do to get a better score than that? >> Cooperate. >> Yeah, so it could. Cooperating with tit for tat will bring it back into mutual cooperation. >> Hm. >> It will actually get a better score. >> Yeah. >> So, in one case, it would average to -1 (cooperate) and the other one, it would average to -3 (defect) (這是另一個 evidence, 證明這不是 subgame perfect) and minus one is better, so, you're right. Good point. >> Okay. >> So what matters is that we get we get them defecting. >> Right. Okay, so that makes sense. So, I, I was right that it didn't matter, although you do get slightly different answers, or slightly different averages. >> That's right. But in both cases, there's a way of getting a higher average. >> Right. Okay, cool, that's what I thought. I thought it was something like that. So now, let's go back to what you wanted to do. So, are we going to be able to figure out how to do Prisoner's dilemma in a way that is sub game perfect? >> Well, how about I propose a machine, and we'll see what it does? >> Okay.

Pavlov

Men on a
basketball
court



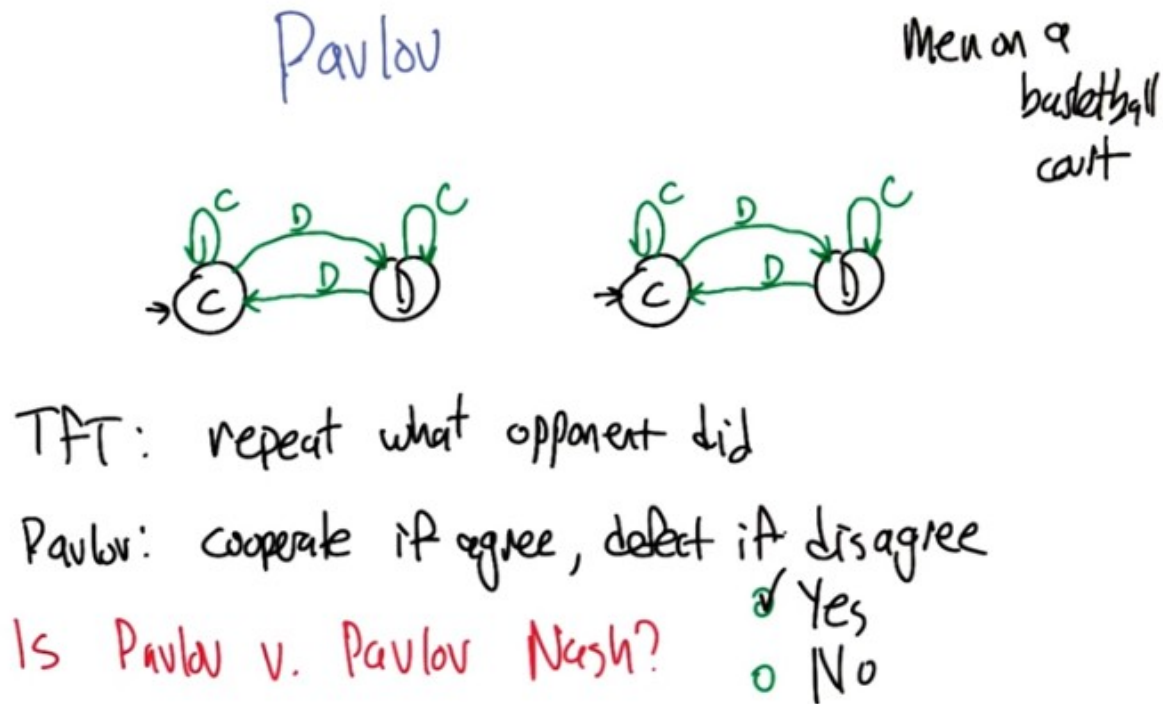
TFT: repeat what opponent did

Pavlov: cooperate if agree, defect if disagree

上圖左右兩都是 Pavlov. Pavlov 俱體如何變, 見 27 段小明和小紅的例子就明白了.

24. So here's a machine that is sometimes referred to as Pavlov named after the Russian psychologists who was studying gastric juices and then figured out how animals learn. So I don't know why it's called that in this particular case but, here's what the machine looks like. It says start off cooperating and as long as the opponent keeps cooperating, then cooperate. So, so far it sounds a lot like tit for tat. >> Yeah. >> If you defect then then move to the defect state. So, okay still looks like tit for tat. And again, this defect state has two arrows coming out of it. But their reversed from what they were in tit for tat. Here it says, if you cooperate with me, I will continue to defect [LAUGH] against you. But if you defect against me, then I'm willing to cooperate with you. So, that's a little strange, right? >> That is a little strange, yes. >> So, you know, the way to get me to stop defecting against you is to defect against me. And then, I I become cooperative again. >> So, in other words, take advantage of you until you sort of, pull a trigger on me. >> Seems like it. Yeah. >> It's a funny thing. So so again, so tit for tat is like repeat what the opponent did. **Pavlov is basically cooperate as long as we're agreeing. If we both defect, I'll cooperate. If we both cooperate, I'll cooperate. But if we disagree, if I dis, if I defect and you cooperate or you cooperate and I defect, then I will defect against you on the next round.** That sort of makes sense. >> Really? >> Yeah, right? I mean, it says: if you're cooperating, I'm going to cooperate. If, we're both going to start out cooperating. And then if you ever defect on me, then you have basically attacked me, and so I'm going to defect on you. Unless you start cooperating again, in which case I think that you're, you're being reasonable, because of what I did, and so now we're going to start cooperating again. >> Except that's tit for tat [LAUGH]. >> No, you're right. You're right, I, I take it back. It doesn't make any sense, too many. >> Yeah, it's weird. It's a little bit weird. >> too many V's. Too many V's. >> Too many V's? >> Mm-hm. >> In Pavlov. >> Yes. >> Two V's. Yeah, but you can think of them being like arrowheads. >> Oh well then that makes much more sense. >> Yeah, exactly. >> Okay. >> Yeah, so it's this weird thing where I'm going to continue to defect against you until you realize I'm hurting you, you punch me once. [SOUND] And now okay, good, we're even again. We good? Yeah we're good. >> Oh this is like men on a basketball court. >> Yeah, sure but, now here's the question. Is this Nash? So, we'll make that a quiz.

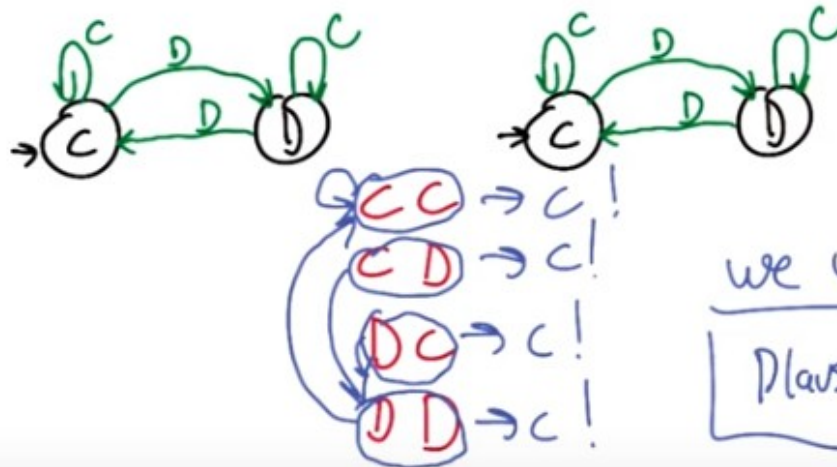
25. So, here's a quiz. And this is, I chose it this way so that I can maximize the number of v's in one sentence. Is Pavlov v. Pavlov, Nash? >> [LAUGH] I think I know the answer. >> Alright. Let's, let's give everybody else a chance to think about it. >> Okay. Go.



26. >> Alright, what'd you get? >> I say yes. >> And the answer, that comes from what? >> Well, we both start off and cooperate. And so, you are always going to cooperate and it doesn't make any sense for anyone to ever move away from cooperation. >> Indeed. >> Hence, you are at equilibrium, in particular, you are at Nash equilibrium. >> That is correct.

27. So, that's very good. So, you, were able to realize that Pavlov is in Nash equilibrium. But we can go further than that. Or farther, than that. >> Further. >> We can go further than that, and show that in fact Pavlov is sub-game perfect. So, unlike tit for tat. It's actually subgame perfect. So let's let's take a look at how we could convince ourselves of that. So I think the important thing to see is that by feeding these two Pavlov machines different sequences we can get into any of these four different combinations of states: they're both in the cooperation state (CC), they're both in the defect state (DD), one's cooperate one's defect (CD), one's defect and one's cooperate (DC). >> Mh-hm. >> so is it the case, that no matter which state that those are in, that the average reward is going to be mutual cooperation (即 CC, 後面馬上證明). So let's check, if they are both cooperating, and we continue with these Pavlov machines then they will mutually cooperate, so yes. >> Mh-hm. Alright if they're in defect defect then what's going to happen? >> Well then they both agree, so then they cooperate. >> So they're going to move to cooperate and then they'll stay there forever, so then that'll be mutual cooperation. Awesome. What if ones in cooperation and ones in defect?

Pavlov is Subgame Perfect!



Average reward: mutual cooperation

CD 情況是如何變成 CC 的:

	第一步	第二步	第三步
小明	C	D	C
小紅	D	D	C

注意, 第一步中 兩個人是同時變成 CD 的, 第二步中 兩個人是同時變成 DD 的..., 第二步中小明如何變? 答曰看第一步中小紅是甚麼. 第二步中 小紅如何變? 答曰看第一步中小明是甚麼.

>> Then they disagree. >> Right. And they move to the other state. >> More specifically what? >> Oh, I don't know [LAUGH], I can't remember. I'm trying to keep track of who, who's who. So if I cooperate. and you defect, then let's see the guy who cooperates moves to defect. And the guy who defects moves to defect. Because you, and now you agree, and so you're going to cooperate. >> Boom. So, when we're in the cooperate defect state, then on the first move. Let's see, you just, yeah, the right-hand Pavlov just defected, so that causes this transition. And this guy just cooperated, which causes this transition, so that we've gone to defect defect. >> Right. Which means that we're going to average cooperation, because [CROSSTALK] >> Mm-hm. >> That's where we're going to get stuck in the long run. >> Right. >> And the same thing works through here. Boom. >> That's actually very cool, and kind of counter-intuitive. >> Yeah. And truly neat. So sort of **no matter what weird sequence we've been fed we manage to resynchronize and then return to a mutually cooperative state.** >> So I have a question for you. >> Go for it. So presumably this is really cool like mathmatically because now we should all do, we should all be Pavlovians. Like we're all Kinseans. AND then we just kind of move forward from there. D people do this? >> I don't know the answer to that question. >> I mean other than men on a basketball court. >> Sure, you can always return to that. Though I'm not sure I'm aware of any analyses of men on a basketball court and whether or not You know, people have analyzed

that. >> Hmm. >> But how about this. If I find out, I will post something on the instructor's comments. >> Okay, that sounds reasonable. So Pavlov is subgame perfect. That's awesome. So remind me again why I care that something is subgame perfect? Because it means that, so let's say that you actually, so I'm being this left Pavlov and you defect and me and you're like yeah I'm going to defect on you because I just want to take advantage of you, and you're, you're going to forgive me and I will have gotten this extra bonus for that. And what it turns out is that No, if we do Pavlov versus Pavlov, we're going to fall into mutual cooperation no matter what. So, so, so this defection that I do, this, this threat, this punishment that I deal out to you, you can earn it back, and we can go back into a cooperative state. >> Right. >> So, it's worth it to me to punish you, because I know that it's not going to cost me anything in the long run, and it stabilizes your behavior in the short run. Sure. It makes perfect sense. >> So it becomes a plausible threat. >> I like it.

Handwritten notes on a whiteboard:

- Computational Folk theorem
- 2 player bimatrix game → average reward repeated
- Can build Pavlov-like machines for any game.
- Construct subgame perfect Nash equilibrium for any game in polynomial time.
- Pavlov if possible
- zero-sumlike (Solve an LP)
- at most one player improves
- Peter Stone
- Me ☺
- MIMC

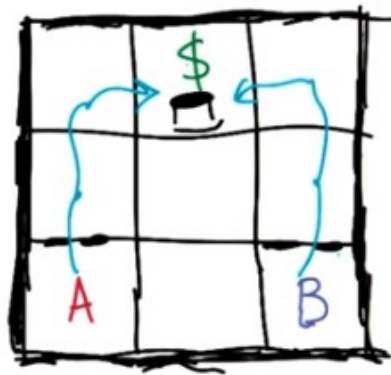
LP: linear program

28. So this Pavlov idea actually is more general than just the prisoner's dilemma or iterated prisoner's dilemma. And in fact, led to a result that I like to call the computational folk theorem. The idea of the computational folk theorem says that you give me any two player, bimatrix game. >> What's a bimatrix game? >> Just that there's two players. [LAUGH] >> Okay. [LAUGH] >> So it seems kind of redundant, doesn't it? >> It does. >> What makes it bimatrix as opposed to two player zero sum game which you can write down with a single matrix, this is like, each, each player has its own reward matrix. >> I see. >> But you're right, I should have, I could've just said bimatrix game and left out the two player. And it's an average reward repeated game. So we're going to play. Rip, Round after round after round. And we're going to look at the average reward. Or, you can also think of it as discounted with an extremely high discount factor. >> Okay. >> So you give me one of those games. And what I can do is, I can build a Pavlov-like machine for, the, for any of these games. And use that to construct a subgame-perfect Nash equilibrium, for any of these games, in polynomial time. >> Wow. >> And, so 後面便是講如何 construct Nash equilibrium for any game: the way that this works is if it is possible for us to have some kind of mutually beneficial relationship, then I can build a Pavlov-like machine

quickly. If not, the game is actually zero sum like, right? because in a zero sum game we can't mutually benefit, so we can't do anything like Pavlov we're just going to beat each other up. So we can actually solve, linear program in polynomial time, and work out what the strategies would be if we're playing a zero sum like game. And so either that works, and produces a Nash equilibrium, and we can test that. Or it doesn't work, but at most one player can improve its behavior. And by taking that best response against what the other player does in a zero-sum like sense, then that will be a Nash equilibrium. So there's three possible forms of the Nash equilibrium. But, we can tick through these, figure out which one is right and derive the actual strategies in polynomial time. >> Wow, that's pretty impressive, who came up with this idea? >> So this is a result due to Peter Stone and somebody, Oh yeah me. >> Oh, well that's very impressive, so you managed to find a way to sneak in some of your own work into this class? >> Here, let's do some more of that. >> Okay, I'm a big fan. And I think that's fair because I did that way back when on mimic. >> Mimic. So, yeah, so what the last topic that, this is, that's all I really wanted to say about the Folk theorem and repeated games. What I'd like to do now is move to stochastic games, which is a generalization of repeating games. And, talk a little bit about how this relates Back to things like queue learning and MDPs. >> Oh, okay. That sounds cool, almost sounds like you're wrapping up. >> It is, that. And that will be the end of, end of the new material. >> Wow. Well that means we're coming towards the end of the entire course. >> I know. We're going to all cry with, disappointment. And I think. And, and I just say this, you know, as a, as an idle suggestion, that the students should demand that we teach more classes. >> I concur. So let's get there so that they can demand. >> [LAUGH]

Stochastic Games and Multiagent RL

Nash?
pair of policies
s.t. neither would
prefer to switch.
A: $\frac{2}{3}$
B: $\frac{2}{3}$



- N, S, E, W, X
- First to reach goal gets \$100.
- Both arrive, both win
- Semi wall (50% go through)
- coin flip if collide

MDP: RL :: stochastic game: multiagent RL

29. So what I would like to tell you about is a generalization of both MDPs and repeated games, that is, that goes by the name of Stochastic games, also sometimes Markov games. >> Mm. >> I like the name Markov game better, but I used Stochastic game because that's what people call it and sometimes it's good to use words that other people use. And what what Stochastic games give us is a formal model for multiagent reinforcement learning. In fact, I like to think of this in terms of an analogy. Which is something like MDP is to RL as stochastic game is to multiagent RL. It's a formal model. That lets us express the sorts of problems that take place in this formalized problem setting. >> Hm. That sounds

very promising. >> Cool. Alright so let me let me give you a, I'll start off by explaining it in terms of an example and then I'll give a more formal definition because you know, I can't not. So so this is a little game played between A and B. Oh, I should have it between smooth and curly, but At the traditionally it's played between A and B. >> Mm, and sometimes it's good to use the words that other people use. >> [LAUGH] I've heard that. I wouldn't say it quite that way. So this is a three by three grid each of the players can go north, south, east and west, and can stay put if that's helpful. And the, the transitions are deterministic, except for through these, these walls here which are called semi-walls. >> Mm-hm. >> So these thick lines represent walls that you can't go through, the thin, wall, lines just represent cell boundaries, but this kind of dashed line here is a semi-wall, and that means If you try to go through that, say by going north from, if A goes north from this position, then 50% probability A will actually go to the next state, and 50% probability A will stay where A is. So, the goal is to get to the dollar sign. And if you get to the dollar sign you get a hundred dollars.

Nash?
pair of policies
s.t. neither would
prefer to switch.
A: $\frac{2}{3}$
B: $\frac{2}{3}$

- N, S, E, W, X
- First to reach goal gets \$100.
- Both arrive, both win
- semi wall (50% go through)
- coin flip if collide

MDP: RL :: stochastic game: multiagent RL

(將上圖 copy 過來了, 為了方便看)

So if we ignore A for a second, what should B do to minimize the number of steps necessary to get the reward. >> Go left, and then go up and go up. Oh, I'm sorry. Go west, and then go north and then go north. >> Yeah, and what should A do ignoring B? >> Go east and then go north and then north. >> Yeah. Unfortunately these guys live in the world together, and what happens is, they can't occupy the same square (即最下層中間的). And as soon as somebody reaches the dollar sign the game ends and the other player, if the other player hasn't reached the dollar sign, gets nothing. >> I see. >> So now there's a little bit of contention. >> So what happens if A and B both try to go, to the same square at the same time? >> Let's say that we flip a coin and one of them gets to go first and then the other one will bounce off of the first one. >> But that's not a problem when it comes to reaching the money. >> But it's not a problem, yes, right, so the money is kind of like a money pit. >> [LAUGH] I don't think that's what a money pit is, but okay. >> And so they can dive in and they both get the money, because they're in the money pit. >> I like it. >> So what do you do if you're A? How do you play this game? Oh! Let's think of another thing. Is, can you think of what it might mean to have a Nash Equilibrium in a

game like this? >> Oh, that's an interesting question. It would mean, well, it would mean, well, what do you mean, what would it mean? It would mean that, neither one of them would want to deviate. >> It would mean a pair of strategies for the two players. Now the strategies are now multi-step things that say, they're like policies, right? >> So... >> Yeah. >> Like it's a pair of policies, such that neither would prefer to switch. So can you think of a pair of policies that would have that property. >> Well, no I'm not sure. I was trying to think about that. I was thinking that kind of, if I were a nice guy what I would want to do is I would want us both to try to go through the, the semi walls, and if we both go through the semi-walls we just go up again and then we, we hit the dollar sign at the same time. And that's very nice. >> So okay, good. So that, that seems like a cooperative kind of strategy, right? Where they're both you know, 50% oh I'm sorry, 25% of the time both will get through, both will go to the goal together. Hooray. But... >> 25% of the time neither one will get through and then we're in the same place we were before, so that's okay. >> That's right. >> The problem is the other 50% where one of them gets through and the other one doesn't. >> Right, so what do, what you do if you make it through and the other one doesn't? >> What do I do, if I get through, and the other one doesn't? Well if I am only going to do this the one time then I just keep going and get the dollar, and the other person loses. >> Yeah, alright, so what this works out to be, is that A is going to get to the goal 2/3 of the time, and B is going to get to the goal 2/3 of the time (why?). >> Mm-hm.

Stochastic Games and Multiagent RL

Nash?
pair of policies
s.t. neither would
prefer to switch.

A: $\frac{1}{2}$
B: 1

- N, S, E, W, X
- First to reach goal gets \$100.
- Both arrive, both win
- Semi wall (50% go through)
- coin flip if collide

MDP: RL :: stochastic game: multiagent RL

>> So, alright, so if that's the case, if I say, okay, A, that's what you should do, B, that's what you should do. Then is there a way that either A or B can switch strategies and do better? >> Well, if B, for example, decides to go west and then go up, what happens? >> Yes, that's a good question. B will now make it to the goal a 100% of the time, and A will only make it to the goal 50% of the time. So B has an incentive to switch to that, to this strategy if we tell them to both go through the semi-wall. >> Right. >> So that (上上圖中的) wasn't a Nash Equilibrium. B would want to switch this new policy. >> Mm-hm. >> Is this (上圖中的) a Nash Equilibrium? >> No. Wait, is it? No. Because, why doesn't A just choose to go west east? >> Well, would, would A do better on average by switching to this strategy? >> Well let's see. no, actually. Oh, no, no, no, you said half the time they go through. >> Yeah. >> So half

the time you flip a coin (跟 B 相碰, 本句應該這樣說: 若往右走, 則跟 B 相碰後, you flip a coin, after you flip the coin, half of time you will reach the money). So half the time I don't make it. >> Right. >> But half the time I do. >> Right. >> So, actually, it looks the same. >> It looks the same. That's right. >> And B would go from 1 to 1/2. >> Yeah, that's true. [LAUGH] So, it, A doesn't have an incentive to do it, but B is hoping very much that A doesn't do that. 這是 Nash equilibrium >> Right. >> So so, yeah. So that, so there's one Nash Equilibrium where B takes the center. Another one where A takes the center. I guess if, if they do, if we do this coin flip thing, it, it works out this way. If it's the case that if they both if we change the rules here. So that if they collide, neither of them gets to go. Then go, both trying to go to the center is not a Nash equilibrium anymore, because you can do better by actually going up the semi-wall. >> Right. And so if we, if, if collision means nobody goes through, then, suddenly, you'd want to do the other thing. >> Exactly. >> Or one of you goes through the semi-wall and one goes the direct way. >> Right. So we can see that there's a bunch of different Nash equilibrium here, sorry, Nash equilibria here. And that it's not so obvious how you'd find them, but it is at least clear that they exist and they have a different form than what we had before, because they're not policies instead of these otherwise simplified just you know, choose this row of the matrix. >> Mm-hm. >> Cool. Alright. So let's think about how we might learn in these kinds of environments. >> Oh, okay, I like that already.

Stochastic Games (Shapley)

S : states s

A_i : Actions for player i . a, b $a \in A_1$, $b \in A_2$

T : Transitions $T(s, (a, b), s')$

R_i : Rewards for player i $R_1(s, (a, b))$, $R_2(s, (a, b))$

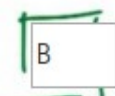
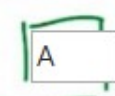
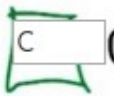
γ : Discount γ

30. So, stochastic games, originally due to Shapley, have a bunch of different quantities. State, actions, transitions, rewards, and discount factors. And here's how we're going to do it. We're going to, we're going to say that s , little s , is, is one of the states. And actions could be like little a , but actually, since we're going to focus mostly on two player games for the moment, I'm going to write actions as a and b . Where a is an action for player one and b is an action for player two. Sound okay? >> Sure. >> Alright. So next we have the transition function. So, the transition function says: If you're in some state, s , and there's some joint action that's taken, like all the players choose their action simultaneously, (a, b) , then what's the probability of reaching some next state s' ? And we can write rewards the same way. So there's reward for player one, given that we're in state s and there's a joint action (a, b) . And there's the reward for the other player, the second player. And a discount factor is you know, like a discount factor. >> Totally makes sense. So oh its the same discount factor for everyone. >> Yes!

Good a good point. One need not define things that way, but in fact that is the way it's always defined. >> Hey, not to go on a tangent here, but sometimes I see MDP's and things like stochastic games defined with a discount Factor being a part of the definition, and sometimes not. Like it's just a part of the prob, definition of the problem or sometimes its a parameter of an algorithm. Which do you prefer? Why do you, why haven't, why have the discount factor actually listed as part of the definition of the game? >> I have no justification other than it's nice to have listed the things that might be important as oppose to you know, working through algorithms for while and then saying, oh yeah there's this other number that kind of matters too. >> Okay that's fair. I was just curious. So one of the things that I actually find really interesting is that this model was laid out by Shapley. >> Mm-hm. >> One of the, like a former Nobel prize winner. I guess once you're a Nobel prize winner, you're a Nobel prize winner forever. >> Yeah, [INAUDIBLE]. >> So, Shapley, the Nobel prize winner, and as we're going to see in a moment, this model is actually a generalization of MDPs, but Shapley published it before Bellman published about MDPs. >> Oh. This is pre-Bellman. So MDPs, to some extent, can be thought of as a narrowing of the definition of a stochastic game. >> Huh. >> So, all right. Let's do a little quiz. And see that we really understand the relationship between this model and other things that we've talked about. >> Okay.

Models & Stochastic Games

$\langle S, A_i, T, R_i, \gamma \rangle$

-  ① $R_1 = -R_2$
-  ② $T(s, (a,b), s') = T(s, (a,b'), s') \forall b'$
 $R_2(s, (a,b)) = 0, R_1(s, (a,b)) = R_1(s, (a,b')) \forall b'$
-  ③ $|S| = 1$

Ⓐ MDP, Ⓑ zero sum stochastic game
 Ⓒ repeated game

31. >> Alright so, Stochastic Games are more general than other models that we've talked about. And so, just to make that case here's a way of making the Stochastic Game settings more constrained. And, by making them more constrained, actually turning them into other models that we've talked about or could talk about. So, I wrote down three different ways of constraining the Stochastic Game model. One says that we're going to make the reward function for one player the opposite of the reward function for the other. The next one says that the transition function has the property that for any state and joint action and next state. If that's going to be equal to the transition probability for state joint action, next state, where we've changed potentially the choice of action for player two. >> Mm-hm. >> So basically, player two doesn't matter to the transitions or the rewards for player one, and the rewards

for player two are always zero. So that's, that's again, you can specify this as a Stochastic Game and then in the third case we are saying that the number of states in the environment is exactly one. >> So I claim that by doing these restrictions. We get out the mark-off decision process model, a zero sum statistic game model, and the repeated game model that we've been talking about in the context of, like, the folk theorem. So, what I'd like you to do is write the letters, A, B, and C in the correct boxes. >> Okay. >> Go.

32. Alright, talk me through it. >> Okay, so I'm going to say that I think I know the answers for this one. And let's start with the first one. So R_1 equals minus R_2 , which you'll notice they're equal and opposite. And in fact if you add them up, that is you sum them you end up with zero. So I'm going to say that's a zero sum stochastic game. >> Nice. >> For two, basically you're saying that for all intents and purposes, there's only one agent. Which just makes it a regular Markov decision process. >> Yeah. So isn't that interesting? That just by the other player irrelevant, then that's what an MDP is. It's like a game where the other players are irrelevant. >> Yeah, which, both of my children are like that. But okay, I think that's pretty cool. And in fact, I'd be right in saying that R_2 doesn't have to equal to zero. As long as it just equals to some constant. >> Yeah, that's, I mean, constant. Actually, depending on how you think about it, it could be, we could just ignore the whole R_2 thing and just say that. As far as the first player is concerned, since the second player really has no impact on anything. It doesn't matter. But the reason I put that in is I got kind of scared that like. I feel like if I lived my life and knew that my actions effected the state and my rewards, but they were also effecting the rewards of somebody who didn't matter. Like I feel like that would actually still have an influence on me. Sure, but then the way you get around that is you would say, well, your R_1 is actually equal to your R_2 . >> Oh. >> [CROSSTALK] It would somehow [UNKNOWN]. >> So, so if I had gone like that, wouldn't that be the case then that we're saying? Oh yeah, I see. That the second player is irrelevant, but the reward, but the first player may be relevant to both. >> Right. >> Yeah, okay, yeah I like that a little bit better. Yeah, I mean, once again, it all boils down to changing the rewards. Okay, and so given A and B, I know the answer to three must be C, unless you're tricking me, and it could be A or B again. And which I suppose you could have done, you didn't say they were mutually exclusive. So let me actually argue why it would be C? Well there is only one state and since you're in a stochastic game and you're going to be continually doing this. It means that you're basically doing the same game over and over and over again, so it's a repeated game. >> Yeah, yeah, yeah so in particular the actions impact the rewards, but they're not going to impact the transitions because you're always just going to back to where you were. The discount factor plays the role of, of decided when the game's going to end, stochastically. And, so yeah, it's exactly a repeated game, this is the one I feel most comfortable about because this really does recover that same model we've been talking about. >> I like it. >> Cool! Now, given that we actually are now in a model that generalizes MDPs, it would be nice if we can generalize some of the things that we did with MDPs, like Q learning and value iteration to this, this more general setting. So, that's what we're going to try next. >> Cool.

Zero-sum Stochastic Games

$$Q_i^*(s, (a, b)) = R_i(s, (a, b)) + \gamma \sum_{s'} T(s, (a, b), s') \max_{a', b'} Q_i^*(s', (a', b'))$$

→ assume joint actions benefit me! optimistic delusion

33. Now what makes stochastic games more interesting, perhaps, than repeated games, is the idea that the actions that the players take impact not just the rewards, but also future states. Right? And, so this is the same issue that comes up when we're talking about markov decision processes and the way we dealt with it in that setting was by defining a value function. So, it seems pretty reasonable to try to go after that same idea again. So what I've got here is actually the Bellman Equation And let's look at this together and let's see if we can fix it because it's not quite right. >> Okay. >> For dealing with the idea of zero sum in a stochastic game. Okay so you remember the Belmen equation? We've got Q_i^* . >> Mm-hmm. >> So there was no I before, but Q^* is the state. Is to find over state actions, so here we're going to define it over action, joint actions >> Mm-hm. >> For the two players, action pairs. The immediate reward to player i for take, for that joint action in that state plus the discounted expected value of the next state. So we need to factor in the transition probabilities So the transition of actually going to some next state s' is s , sorry t of s , AB as prime. Right? So now, we're imagining what happens when we land in s' . So what I've written here says, well, we're going to basically look at the Q values in the state that we landed in, and kind of summarize them, summarize that value back up so that we can use it to define the, the value of the state that we left. You with me? >> I am with you. >> Alright so if we put this in if, if we say the way we're going to summarize the value for the new state that we land in. Is we think of it as actually a matrix game. That there's payoffs for each action choices of A' and B' . And over all of those, we need to kind of summarize well which of those actions in this table of values that we get for s' . Which of those values are we going to propagate forward and call the value of that state? So what we did in regular MDPs is that we said we'll take a max over all the actions or in this case all the joint actions. >> Uhun. So what do you think that translates to? >> Well you wrote down max but that doesn't make sense, that doesn't, that can't be right. >> Well it translates, it means something. It just doesn't mean what we mean it to mean. >> That's true, that's fair. >> So what does it mean and then, and then, how can we fix it? So let's start off with what does it mean. >> It means that you kind of always assume that the joint actions that are going to be taken will benefit you the most. So, everyone is trying to make you happy so, this makes you optimistic? >> yeah, sort of optimistic to the point of, of >> Delusion? >> Yes, very good. Right, it just basically says that whenever we're in a state, the whole world is going to choose their actions to benefit me, and this is not what we get a say a zero sum stacastic game. But a zero sum stacastic game, we should be you know. Like fighting it out at this point. >> So that would work out if everybody's rewards were the same, or everybody's rewards were the sum of everyone's rewards, or something like that.

Zero-sum Stochastic Games

$$Q_i^*(s, (a, b)) = R_i(s, (a, b)) + \gamma \sum_{s'} T(s, (a, b), s') \min_{a', b'} Q_i^*(s', (a', b'))$$

$$\langle s, (a, b), (r_1, r_2), s' \rangle: Q_i(s, (a, b)) \leftarrow r_i + \gamma \min_{a', b'} Q_i(s', (a', b')) \quad \text{minimax-Q}$$

>> That's right. If it was some kind of team based game. >> Mm. >> Or if everybody was, you know, going to sacrifice their own happiness for the benefit of Q_i , or i I mean. >> Hm. So it's not reasonable to assume that. In fact, what was it that we were assuming when we had a zero sum game that was just a single stage? Right? Just a single game and then we were done. >> Oh, that people were doing minimax. >> Right. >> And maximin. >> So what if we changed the equation to look like that? So, what I mean by this is when we evaluate the value of a state, we actually solve the zero sum game in the Q values and take that value and use it in the context of this equation. >> That seems closer to right. >> Yeah. I mean, it's not an unreasonable thing to do. It's just to say, I'm going to summarize the future by imagining that we're going to play that game that represents, you know, all the future. >> Sure. >> And I'm going to act in such a way to try to maximize that assuming you're trying to minimize it, which makes perfect sense if it's a zero-sum game. >> Right. I was, yeah, and we're still, we're still acting as if there are only two people here. >> Yeah, yeah, that's right. It turns out that when you're talking about zero-sum it really implies that there's only two players. Because if you have a zero-sum three player game, it really is just a general sum game. You can imagine that the third player is just an extra factor that's just messing with the numbers to make things sum up to zero. So, yeah, so zero sum really does kind of focus on this two player setting. >> That makes sense. >> So we got this modified Bellman equation and we can even translate it into a form that's like Q learning. So the analog of the Bellman equation and the Q learning update in this setting would be that we. If we're in some state, there's some joint action that's taken, there's some pair of rewards that comes and some next state that's visited that the Q value for that state, joint action pair, is going to be updated to be closer to, the reward for player i plus the discounted expected value, or sorry, the discounted Summarized value or value of the new states as prime, and we'll again, we'll use mini-max to summarize what the values are in that new state. >> I like it. >> And that equation is sometimes referred to as mini-max Q , because it's like the Q learning update but just with the mini-max operator instead of a max. >> That makes sense.

Zero-sum Stochastic Games

$$Q_i^*(s_i, (a_{-i})) = R_i(s_i, (a_{-i})) + \gamma \sum_{s'} T(s_i, (a_{-i}), s') \min_{a_{-i}'} \max_{a_i'} Q_i^*(s', (a_{-i}'))$$

$$\langle s_i, (a_{-i}), (r_i, f_i), s' \rangle: Q_i(s_i, (a_{-i})) \leftarrow r_i + \gamma \min_{a_{-i}'} \max_{a_i'} Q_i(s', (a_{-i}'))$$

- Value Iteration works
- minimax-Q converges
- Unique solution to Q^*
- Policies can be computed independently
- update efficient



2:34 / 2:35

Q functions sufficient to specify policy



YouTube



34. So, if we set things up this way, we actually get some wonderful properties coming out. So here are some things we know about this set up for zero-sum stochastic games. Value iteration works, so we can actually solve this system of equations (第一行的式子, 即 $Q = R + \dots$) by using the value iteration trick, which is to say, we initialize these Q values to whatever and then we just iterate this as an assignment, right, we just say, you know, equals. So value iteration works. This minimax Q algorithm (第二行的式子, 即 $\langle \dots \rangle Q \dots$) converges under the same kinds of conditions that Q learning converges, so we get this nice, you know, Q learning analogue in this multi-agent setting. The Q^* that's defined by these equations (第一行的式子) is unique, so we iterate it and we find it and it's just, there's just that one answer. The policies for the two players can be computed independently, that is to say, if two different players are running minimax Q on their own and not really coordinating with each other except for by playing the game, that the policies that they get out will actually converge to minimax optimal policies. So it really does solve the zero sum game, which is maybe not so surprising because, you know, they are trying to kill each other after all. [LAUGH] >> Yeah. >> So, the idea that they'd have to collaborate to do that efficiently would be weird. >> [LAUGH] I never thought about it like that, but, yeah, that would be weird. >> The, this update that I've written here (第二行的式子) can be computed efficiently, which is to say in polynomial time. Because this minimax can be computed using linear programming. >> Yes of course. >> And, finally, if we actually iterate this Q equation (第二行的式子) and, and it's converging to Q star, knowing Q star is enough to figure out how to behave optimally. So we can convert these Q values into an actual behavior, again, by using the, the solution in the linear program. >> So it's just like MDPs of value iteration with Q -learning? >> Exactly. It's like we've gone to, to a second agent and it really hasn't impacted things negatively at all. This is, this is all the, pretty much all the things that we want, come out. There, there are some things that don't come out. For example, in the case of an MDP, we can solve these, this system of linear equations (第一行的式子) in polynomial time. Not just by value iteration, but we can actually set up it as a single linear program and solve it and be done in linear time or, sorry, not linear time, polynomial time. This is not known to be true in the zero-sum stochastic game case, it's not known whether it can be solved in polynomial time. >> Hm. >> So there, it is a little harder as a problem, but it's, you know, not harder, not deeply harder and not harder

in a way that matters in a machine learning setting. >> Cool. >> So this is really great. So let's, let's try to take this same approach and see if we can deal with general sum games. >> Okay.

General ~~Zero-sum~~ Stochastic Games

$$Q_i^*(s, (a,b)) = R_i(s, (a,b)) + \gamma \sum_{s'} T(s, (a,b), s') \min_{a', b'} \max_{a'', b''} Q_i^*(s', (a'', b''))$$

$\langle s, (a,b), (r_1, r_2), s' \rangle: Q_i(s, (a,b)) \leftarrow r_i + \gamma \min_{a', b'} \max_{a'', b''} Q_i(s', (a'', b''))$

- value iteration ~~works~~ doesn't work
- ~~minimax~~ ^{Nash} ~~Q converges~~ doesn't converge
- No unique solution to Q^*
- Policies can ^{not} be computed independently
- update ^{not} efficient $P = PPA$

incompatible

insufficient

functions ^{not} sufficient to specify policy

35. >> So okay, so let's think about General sum games, so not zero sum any more. But we're not, you know, restricted, it could be any kind of relationship between the two players. And so the first thing we need to do is realize well, well we can't really do minimax here any more. Right, because that doesn't make sense. >> Right. That only works with zero-sum games. >> Well it's only, yeah. That's, well, it sort of assumes that the other player's trying to minimize my reward and that's not the concept of the Nash equilibrium. We'd like to do something analogous and find a Nash equilibrium in this general sum setting. So what, what operator do you think we would need in this context here? >> Nash equilibrium? >> Yeah, so that would be a very reasonable thing to do, is instead of computing minimax, we actually compute of the two matrix game, right, using Q_1 and Q_2 , compute the Nash equilibrium of that and propagate that value back. It's a well defined notion, right, that we can summarize the value of these two pay off matrices with with a pair of numbers which are the values of the Nash equilibrium. >> Mm-hm. >> Alright, so so good. So we can do the same thing in the Q learning setting. Substitute in a Nash equilibrium. And we can call that algorithm Nash- Q , which is, appears in the literature. >> Nice. >> Oh minimax Q by the way is something that I wrote about. Nash- Q is a different algorithm. >> So it's not as cool, is what you're saying. >> Well, let's let's see how it goes. So this is now an algorithm, you can actually, well, this set of equations it's not exactly clear what it means, but we can think about turning that into value iteration, right? By turning this into an assignment statement. >> Mm-hm. >> So, what happens? Well, value iteration doesn't work. >> No. >> So, yeah, so if you repeat this over and over again, things, weird things can happen, it doesn't really converge, it doesn't really solve this system of equations necessarily. >> Hm. >> And unfortunately the, the reasoning here is even harder in the case of Nash- Q because in the case of Nash- Q , it's really trying to solve this system of equations using something like value iteration, but with extra stochasticity. And so it also suffers the same problem. It doesn't necessarily converge. There's not really a unique solution to Q^* because you can have different Nash equilibria that have different values.

>> Right. >> So there isn't really much hope of converging to the answer because there isn't the answer. The policies can not be computed independently, right, so Nash equilibrium is really defined as a joint behavior, and so we can't just have two different players computing Q values. Even if we could compute the Q values. It wouldn't necessarily tell us what to do with the policies (應該是討論的 independently 那一項), because if you take two different policies that are both half of a Nash equilibrium, two halves of a Nash equilibrium do not necessarily make a whole Nash equilibrium. >> Right. >> because they could be incompatible. So, you know, so far so good, right? >> Yeah, I can't wait to see what happens next. >> The update is not efficient unless P equals PPAD, which is to say, computing a Nash equilibrium is not a polynomial time operation as far as we know. It is as hard as any problem in a class that's known as PPAD. And this is actually a relatively recent result, in the last five, ten years. And this class is believed to be as hard as NP. So, possibly harder. So it doesn't really give us any leverage to, computational leverage to kind of break it down in this way. So that's unfortunate. And finally, the last little hope of (即上圖最後一行), well, maybe we can define this kind of learning scenario using Q functions the same way we've been doing, Q functions are not sufficient to specify the policy. That is to say, even if I could do all these other things (即上圖前面的幾行), efficiently compute a solution of, you know, build the Q values, make them so that they're compatible with each other. And now I just tell you, here's your Q function. Now decide how to behave, you can't. It's, there's not enough information. >> You're depressing me, Michael. >> Yes, so this is kind of sad. We go to the general sum case, which in some sense is the only case that matters' because zero sum never really happens. And what we discover is that we lose all, seemingly lose all of the leverage that we have in the context of Q type algorithms. >> Mm, mm, mm. >> And that's where we'll stop. >> Oh. So we're going to end on a high note. >> No, maybe we should say something before we depart. >> Let's do that. Come up with something positive to say. >> Okay.

Lots of Ideas

- repeated stochastic games (Folk theorem)
- cheap talk → correlated equilibria
- cognitive hierarchy → best responses
- side payments (coo values)

36. So even though things are kind of grim, with regard to solving the general sum games. There are lots of ideas that have proven themselves to be pretty useful for addressing this class of games. It is not the case that any one of them has emerged as the dominant view, but, but these are all really cool ideas. So here's one. You can think about stochastic games as themselves being repeated. So, repeated stochastic games. We're going to play a stochastic game and when it's over, we're going to play it again. And that allows us to build folk theorem-like ideas at the level of stochastic games. >> Oh, that's cool. >> And so there are some efficient algorithms for dealing with that. So that's one idea. Another one is to make use of a little bit of communication side-channel (即 cheap talk 那個) to be able to say, hey, other player. Here's this thing that I'm thinking about. And it's cheap talk in a sense that, it's nothing that's being said is binding in any way but it gives the two players the ability to co, to

coordinate a little bit. And you can actually ultimately compute a correlated equilibrium, which is a, a version of a Nash equilibrium that you know, requires just a, a little bit of coordination, but can be much more efficient to compute. And you can actually get a near optimal approximations of the solution to stochastic games using that idea. >> Yeah, that's cool. Didn't, didn't I do some work in this space? >> You did. That's where I got the idea from. >> Oh okay. >> There's some, some work by Amy Greenwald looking at how correlated equilibria play into stochastic games and then your, your student Liam and you developed a, a really cool algorithm that actually probably approximates, the solutions. >> Nice. >> Another idea that I've heard a lot about lately, that I really like, is the notion of a [cognitive hierarchy](#). The idea that what you're going to do is instead of trying to solve for an equilibrium, you think about each player as assuming that the other players have somewhat more limited computational resources than they do. And then taking a best response to what they believe the other players are going to do. This turns out to be a really good model of how people actually play when, when you ask them to do games like this in the laboratory. >> Huh. >> Yeah, the good news about this idea is that, because they're best responses, they can be more easily computed. That, that it's more like, cue learning in MDPs again because you're assuming that the other player is, is fixed. >> Okay. I'll buy that. >> And, the last idea I want to throw out is the notion of actually using [side payments](#) so that the players, as they're playing together, cannot only take joint actions, but they can say, hey, I'll give, I'm going to get a lot, but if we take this action, I'm going to get a lot of reward. I'm going to give some of that reward back to you, and that will maybe encourage you to take the action that I need you to take so that we'll both do better. And so there's this lovely theory by a father and son duo that they call coco values. Coco sounds awesome but it stands for Cooperative competitive values [CROSSTALK] and so it actually balances the zero sum aspect of games with the mutual benefit aspect of games. So it's, it's, it's a really elegant idea. >> [So basically, the problem isn't solved but there are a lot of cool ideas that are getting us close to solving it.](#) >> That's right. Yeah. So [even though the one player and the zero sum cases are pretty well understood at this point, the general sum case is not as well understood. But there's a lot of really creative ways that people are trying to address it.](#) >> So, that is good news.

What Have We Learned?

- Iterated PD
- connect IPD & RL (discounting) repeated games
- folk theorem (threats)
- subgame perfection, plausible threats
- computational folk theorem max-acceptable
- stochastic games, generalize MDPs, repeated games
- zero sum stochastic games. minimax Q works.
- general sum games. Nash Q doesn't. (End hopefully).

37. Okay Charles, what have we learned? And I mean specifically in the context of this game theory two lesson. >> That's a that's a good question. We learned about Iterated Prisoners Dilemma. Which turns out to be cool, and it solves the problem, and [we learned about how we can connect iterated prison's dilemma to reinforcement learning](#). >> What do you mean? >> [Through the discount](#). >> Yeah, so I think of that as being the idea of repeated games. >> Right. Let's see, what else have we learned? So we learned about iterated prison's dilemma, which allowed us to get past this really scary thing with repeated games, connected it with reinforcement learning. The discounting. And then we learned other things like for example, I don't remember. What, what did we learn? >> Well so the, the connection between iterated prisoner's dilemma and repeated games was the idea that we can actually encourage cooperation. And in fact, there's a whole bunch of new Nash equilibria that appear when you work in repeated games. That was the concept of the Folk Theorem. >> Right. The Folk Theorem. So, the Folk Theorem is really cool. And this whole notion of repeated games really seems like a clever way of getting out of what appear to be limitations in game theory. >> Right. Yeah. And in particular by using things like threats. >> Right. But only plausible threats. >> Right, so that was the next thing we talked about. The idea that an equilibrium could be subgame perfect or not, and if it wasn't then the threats could be implausible. But in the subgame perfect setting, they're more plausible. >> Right let's see and then we learned about Min-max Q. >> Well there was one last thing we did on the repeated games, which was the Computational Folk theorem. >> Yes you're right. So basically what we learned is that Michael Littman does cool stuff in game theory. >> Or at least he does stuff that he's willing to talk about in a MOOC. >> Yes, so that's, that's there's actually a technical term for that right? MOOC acceptable research? >> Oh, I didn't know that. >> Mm-hm. >> So all these things are by virtue of the fact that they showed up in this class look acceptable. >> Exactly. >> Alright, you're right, but then we switch to stochastic games. >> Mm-hm. >> And they generalize MDP's and repeated games. >> Mm-hm. >> Anything else? >> Well, that particularly got us to min-max Q and then eventually to Nash Q. But despite the fact that Nash Q doesn't work, we ended up in a place of hope. >> [LAUGH] We end with some hopefulness. >> Yeah, and you know, I think that that's actually a lesson for the entire course. That at the end of the day, sometimes it doesn't always work, but there is always hope. >> [LAUGH] We don't give up and that's, that's, that's how research works. Even when we have

impossibility results for things like clustering, or multi-agent multi-agent learning and decision making, we still keep struggling forward. >> And keep learning, and isn't that what's really important. I think so. >> Its important for us, and its important for machines. >> Yes, that is beautiful. I feel like we've made it to a good place, Michael. Perhaps we should stop. >> [LAUGH] Well it has been, it has been delightful getting to talk to everyone, and it has been very fun getting to talk with you, Charles. And thanks to everybody for making this happen. >> I agree. And we have one more chance to talk with one another as we wrap up the class. And I look forward to that. So, I will see you then, Michael. >> Awesome. Do we get to see each other in person for that? >> We get to see each other in person for that. That will be fun. >> Yay! >> Okay, well, bye, Michael. I'll see you next time. >> Bye. See yeah. >> Bye, bye.