Backward induction imperfect information

Popularized by films such as Beautiful Mind, game theory is mathematical modeling of strategic interaction between rational (and irrational) agents. Besides what we call games in common language such as chess, poker, football, etc., it includes simulations of conflicts between nations, political campaigns, competition among firms, and trade behavior in markets like the NYSE. How could you start modeling keyword auctions and peer-to-peer file sharing networks without considering the incentives of the people who use them? The course will provide the basics: presenting games and strategies, a broad shape (which computer scientists call game trees), Bayesian games (simulations of things like auctions), repeat and stooch games and more. We will include various examples including classic games and multiple apps. You can find the full curriculum and course description here: jacksonm/GTOC-Syllabus.html There is also an extended follow-up course to this, for people already familiar with game theory: You can find the introductory video here: jacksonm/Intro_Networks.mp4View Learning Theory, Reverse Induction, Bayesian Game, solving problemsCalbanakytaiska (simplified)Englisheeston germanyJapaneseCasakhLituarian MongoliaNepariaRoman Russian SkerbiancaSwedishTamilTeluguThaiTurkishIn this video I am going to tell you about how to determine formally, imperfect information great form and how to rape about stretegui in this case. So, in perfect informational broad form, we have a player action on every selection node in the game. And the consequence of that definition is that we see that the players know what kind of knot they are all the time. And that means they know the whole history of all the moves that took place earlier in the game. It's smart for some games, such as chess, where you really get to see what your opponent does in every other move. It's not smart for games like Battleship, where your game is, the one your opponent can do something and you don't get to see what it is. And it matters what happens to you later in the game. This may matter to your payout. It may matter whether the game stops or goes on. So in order to model this richer situation where players are unable to observe everything their opponents do, we are going to add something new to the game's representation. So we will call it imperfect information of a great form. And the way it's going to work, we're going to embrace the old definition we had before, but we're going to say that the players consider some nodes of choice to be equivalent to each other. So, there are some selection nodes that the player cannot repent of. And it will mean they are unable to fully figure out the history of where they are in the tree because they are not going to which of the multiple selection nodes they are in when they have to make a choice. And we're going to do that by taking a set of selection knots for a given player, and putting them into equivalence classes. So, what that means is kind of schematic if it's some other node of choice in the game that all belong to the same player. We can say it's one equivalence class, it's a different equivalence class, and it's third-grade equivalence. And whatever it was, the player wouldn't know which of those two knots of choice I was, he was when asked to make a choice. but he would have known that he was on one of these two, not one of those two, because they are in different equivalence classes. So let's say it's a little more formal. So, to formally define an imperfect informational great form of the game. We begin with a perfect information wide-form game that we have already learned to define earlier. And then we'll add this ingredient of equivalence classes. So we'll add this Element I, which is a set of equivalents, a set of equivalent classes, one for each player. So for Player 1 we have this set of equivalence classes or, say, for Player I, we have a set of equivalence classes, 1 to K sub I. So, these are all different equivalence classes. And each of these classes is some number of different selection nodes, one or more nodes of choice. And these will be different nodes of choice that this player cannot tell. So, if each of these equivalence classes contains only one selection node. We are back with a perfect information case, and if any of these equivalence classes contains more than one thing, then we have something new. We have a game where the player doesn't quite know what happens all the time. Now, for this definition to work, we need to add a couple of limitations to make sense of it. So we want to say that if two different choice nodes are part of the same equivalence class. Then, above all, they must belong to the same player. They have to be seen with the same player because if they are not, you will be able to tell them apart because they, the other player will act. So a player, sort of when he figures out, he'll know which, the player he was, he wou, he really wouldn't be confused about all the knots in the equivalence class. And the second limitation we have is that the two selection nodes should have the same set of available actions because the player can't tell them apart, he needs to know what to do. And these are the only limitations. So let's look at the example of the game here, and see what equivalence classes are. So in this game, Player 1 has 2 different equivalence classes. It's Grade 1 equivalence, and it's another class of equivalence. So in other words, we're going to use the dotted line here to connect together a selection of nodes that belong to the same equivalence class. And what we want to say about it is that Player 1 moves and if it goes right then the game will just end and they're just going to get everyone to pay off 1. If he goes left then they're going to get to make, Player 2 is going to get to make a choice. and player 2 is going to go either a or b. And then, Player 1 is going to get to move a second time. But Player 1 is not going to get to watch the move that Player 2 took. So he'll have to take the same action, whether he's on that selection node or that selection node. And indeed, you see, we marked it the same way. So, if he says he wants to go to the left, he'll have to go left of both of those notes. And just to complete the answer to my question where the equivalence classes are for player two, well player two is only one node of choice really as a table here. There shouldn't be two player two has only one node of choice and so it only has one class of equivalence. So how should we identify clean strategies for every player in this game? What makes sense as a definition of clean strategies? Well, intuitively remember before what we said is that we have a cross product of all the different sets of actions for each player. We don't want it here because we don't want it to be possible for Player 1 to do something different in this selection mode and then in that selection mode. So instead of what we're going to use as a definition is that pure strategies for the player I'm a cross product of action sets in every different equivalence class that he has. So, the pure strategy of Player 1 here will be the choice here, and self-selection here. So player 1 can take action L, l or he can take action R, l or he can take action L, r or R, r. So player 1 here has 4 different clean strategies, not 8 as he would have if we hadn't had that equivalence class here. Thus, in an imperfect normal form of information, we have a more powerful representation than in a perfect information business. And, in one way we can see that we can represent any normal form of game in this representation that you may recall, we couldn't do with perfect information games. So here I show you how to represent a TCP backoff game or in other words, a prisoner dilemma game, in perfect informational broad form. So how does it work? Well, first of all, Player 1 gets to decide whether to cooperate or defect. And after that, Player 2 gets to decide whether to cooperate or defect. Now, of course, in the prisoner dilemma, you don't get to see what the other person, determined to do when you do your own actions. So, it might sound like a problem that Player 2 gets to move second. But, Player 2 is unable to say what action Player 1 took. And so while our representation of the game says it goes second. It doesn't really make a difference that it goes second. because you did not report what Player 1 did. And then, as soon as they we will know our actions, we with some payments. So if they go C, D, then they end up with that payoff here. And these are the same payoffs we have in the matrix game. Note that we could, there are things working the same way if we put a Player 2. At the root node and player 1 down here below because time doesn't really play a role in this game. So what I told you on the previous slide was how to start with a normal form game and make a great play out of it. I can also still do what we talked about with perfect information of great form, which is to start with a great form of play and make a normal form of the game out of it. And how it works exactly as it once did. So, I take all the clean strategies for one player and I make them into the ranks. I take all the clean strategies for player two and I make them into speakers and then, it gives me my matrix. And for each matrix cell, I say well if Player 1 took this clean strategy. And Player 2 took this clean strategy, what payback would lead? And I put that in a matrix cell. And I fill the whole matrix that way. And once I have it, a matrix like this. Then, then I kind of finished. Mine, my definition of mixed strategies is exactly what it was before. It's just mixed strategies in an induced normal way. determining the best response in our balance for imperfect information of big form games again is simply the kind of leverage-induced normal shape. And so all those concepts that you already understand from regular form games are transferred directly to imperfect information games. And so, for example, we know with Nash's theorem that our equilibrium always exists for every imperfect information great form of the game because I can make the ultimate normal form of the game out of it. Now, it will be the case that this transformation can make the game exponentially bigger, as it might before, even with the perfect information case. But, for example, for the existence of an equilibrium that does not matter. Now, finally, you may wonder what will happen if I take both of these transformations that I have shown you and I will apply them together. So, for example, I could start with imperfect information of a great form of the game, make it into a normal form of play and then make it back into imperfect information of a great form of play again. So you might wonder, do I end up with the same game? And the answer, no, I won't. Because I might have a tree game that's pretty deep. it could have all kinds of different equivalence classes. This, he can have all kinds of different consistent moves by different players. And when I do it in a normal form of play. I'll take that flat table. And then when I take a flat table, and turn that into a great form of the game. It will be a great form of the game that has only 2 levels. With all this material in the 1st grade of great equivalence. So, it's going to be a great form game that doesn't look like I started with. importantly, it will have the same strategy spaces, the same sets of clean strategies for both agents. And it will have the same set of our balance. So while these games may look different than in terms of how they explicitly talk about time, they will be strategically equivalent games. And that's for this video. Video.