Game Theory

The purpose of game theory is to study the (optimal) behaviours of agents when they have multiple choices (strategies), depending on the context, on the amount of information they have and other parameters.

Dilemmas

Let's look at a very simple example, the prisoner's dilemma. Two criminals are caught, and proposed a deal : they can testify against their partner for a reduced sentence. If prisoner A testifies against B who doesn't testify, A doesn't go to jail and B goes for 5 years. If they both stay silent they both go for 1 year, and if they both testify they both go for 4 years. We can sum it up in a tidy little matrix (where the numbers are the years in jail, which we try to minimize) :

	testify	stay silent
testify	4 – 4	0 – 5
stay silent	5 – 0	1 – 1

One can observe that no matter what the other does, one should always testify to reduce the sentence. This means that both should testify, and in the intuitive sense the only equilibrium is the 4-4, which is far from optimal. This is called a Nash Equilibrium. A precise definition of such an equilibrium is a combination of strategies such that no player has interest in unilaterally changing their strategy. Such an equilibrium is not necessary bad, as we can see in the following variant :

	testify	stay silent
testify	4 – 4	1 – 4
stay silent	4 – 1	1 – 1

There the only equilibrium is the 1-1. Here we can see that from any strategy we can go to the optimal equilibrium while improving the solution at each step. However, this is not necessarily the case. Suppose two people want to watch a movie in the cinema. If they meet at the same movie they both win, if not they lose, but they have different preferences for the movie. We can modelise this as the following game (where we try to maximize the gain) :

	Horror	Comedy
Horror	3 – 1	0 – 0
Comedy	0 – 0	1 – 3

Here we have two equilibria, and we can't switch from one to the other. Lets look at the following variant :

	Horror	Comedy
Horror	1 – 1	0 – 0
Comedy	0 – 0	5 – 5

Here if both players start at Horror, they will have no inclination to go towards Comedy because it temporarily lowers their gains. We can repeat a game many times and see if the behaviors change. Interestingly, the longer you play the (usual) prisoner's dilemma, the more you should cooperate. However, one can prove that – if both know how many turns there are – at the last turn both should testify because it has no consequence on future cooperation. By induction, we can prove that this is true for the penultimate turn, until we show that the optimal play is to testify at each turn. This is true as long as both players know how long the game will last (but can also be true if that's not the case).

Remark. A while ago, some researchers decided to let people compete to see who would code the best program to play the repeated version of the game (called iterated prisoner's dilemma). It turns out that there is a very simple strategy that is extremely efficient : copy whatever the other did on their last move. This means that if the other cooperates you also cooperate, and everyone wins, but if the other testifies you punish him at the next turn. The only way to improve this is to add something called a forgiving method, which is a very small chance to stay silent even if the other testified at the previous turn.

So far all the games we have seen had one or two equilibria, but this is not necessarily the case. Consider Rock-Paper-Scissors :

	Rock	Paper	Scissors
Rock	0 – 0	-1 – 1	1 – -1
Paper	1 – -1	0 – 0	-1 – 1
Scissors	-1 – 1	1 – -1	0 – 0

Here we have no Nash equilibrium. However, so far we've only consider what is called pure strategies, where one chooses a possibility and sticks to it. We could also consider mixed strategies, where we assign a probability to each action, and do it with that probability. Then there is a nice result by Nash, stating that any game with a finite number of players, and for each player a finite number of pure strategies has a mixed equilibrium (or multiple equilibria). Here we are only talking about games with perfect information (hence not card games like poker).

Nims

Let's consider another type of game, quite popular in many cultures, called a nim (and very close to the mancara family of games, like oware or awari). We have a set of n pebbles, and at a player's turn that player removes between 1 and k pebbles. This is called an impartial game because the moves available only depends on the state of the game and not whose player's turn it is (as opposed to most board games where one player has a defined set of pawns). There are two different ways to play it :

- Under the "normal" convention, any player who can't make a move loses (so you lose if there are no pebbles at the beginning of your turn).
- Under the "misère" convention, the player who plays the last move loses.

For the simple nim, there are winning strategies under both conventions. Let's consider the normal convention first. If I manage to remove the last pebble I win. Hence if there are fewer than k pebbles at my turn I win. The only way to force that is for the opponent to have exactly $k+1$ pebbles at the start of his turn (hence leaving me with at most k pebbles). But to achieve that reliably there needs to be between $k+2$ and $2k+1$ pebbles at the start of my previous turn. The strategy is then to make sure that at the start of your opponent's turn the number of pebbles is a multiple of $k+1$. Then if they remove i pebbles you can remove $k+1-i$, and get back to a multiple of $k+1$. Both players can try to implement this strategy, but in this case the winner is :

- The one who starts if the number of pebbles is not initially a multiple of $k+1$
- The second one in the other case.

Under the misere convention the goal is similar, except that we want to be left with between 2 and $k+1$ pebbles instead of 1 and k (but the strategy is the same, we just try to get to a multiple of $k+1$, plus one).

We can create a more complex kind of game called a nimber, where instead of one stack of pebbles we look at multiple stacks. The reason to study nims then becomes obvious, due to the following theorem :

Theorem. *(Sprague-Grundy) Every impartial game under the normal convention is equivalent to a nimber.*

The proof is a bit hard, hence not detailed here, and gives a winning strategy for any such game (either for the first or second player). A winning strategy here is an algorithm that tells a player which action to do at each turn and automatically wins. It is worth noting that the theorem still sometimes holds under the misere convetion, but not systematically.

A chocolate problem

Let's look at a chocolate bar, made from tiny rectangles, and composed of $m \times n$ pieces. Each piece is then designated by a number (x, y). This game – called CHOMP – is played under the misere convention (or it wouldn't be interesting because of the previous theorem), so the one to eat the last piece of chocolate loses. Each turn a player chooses a piece not yet eaten and eats it, as well as all the pieces which have bigger x and y. Hence when we eat the piece (x, y) we really eat all the pieces (x', y'), with $x' \geq x; y' \geq y$. This game is interesting for two reasons :

- It is generally not solved : the only cases where we know the winning strategy are the ones with $m \leq 3$, $n \leq 3$ or $m = n$.
- We still know who has a winning strategy.

For the cases with $m \leq 3$ or $n \leq 3$, see this recent paper https://www.emis.de/journals/INTEGERS/papers/fg7/fg7.pdf. For the square, the first player eats the piece $2, 2$. We are left with one row and one column. The second player chooses a piece (x, y), and the first player only has to take (y, x) (by symmetry). Which means that the first player can always play, and will never take $(1, 1)$.

Remark. This type of symmetrical copying is quite frequent. For example, consider a round table and a collection of identical coins. The players alternate placing a coin on the table until they can't do that anymore. The first player has a winning strategy : play in the center and then copy the other player's previous move applying a central symmetry first. If the second player can put a coin down somewhere, by symmetry the opposite place will be empty.

Let's now prove that there is a winning strategy for the first player (by what is called a strategy stealing argument). Let's suppose that the second player has a winning strategy. Then the first player eats the piece (m, n). The second player eats any other (x, y). Then no matter what the first player does, the second player can force a win. But what if the first player were to play (x, y) as a first move instead of (m, n), and then follow that same strategy ? Then she would win and this shows that the second player has no winning strategy (and as the game has to end at some point and there can be no tie, it also proves that the first player has a winning strategy). This is because playing (m, n) and skipping your first turn are nearly equivalent. Indeed, if the first player is allowed to skip her turn (exactly once), then the first player always has a a winning strategy.

Remark. One can also notice that at no point in the game (if the first player follows the winning strategy) will we have a game state that looks like an (m', n') rectangle when it's the second player's turn, except if $m' = n' = 1$. This is because there is a winning strategy for any such position, and the second player would become first player (and reciprocally).

Finding the poisoned one

The final problem is not really related to game theory but has links. Suppose you have $n = 1000$ batches of chocolate. However, a mad teacher poisoned one of the batches. You can "rent" some students (for extra credit) to try to see which ones are poisoned. You also have a big party the following day where you plan to distribute the chocolate but would like to identify which batch is poisoned first. The poison takes between 12 and 24 hours to have an impact. If you had a lot of time you could just ask a student to eat from one batch of chocolate, wait 12 hours, make them eat from another batch, and so on until you see that they are poisoned and remove the one chocolate they ate between 12 and 24 hours earlier. This however takes a long time, and you only have one day. How to design a protocol that uses as few students as possible to quickly find the chocolate. The protocol should look like :

- You take k students.
- You make each student eat from a specific and personalized subset of batches at the same time.
- You observe which one is still alive after 24 hours.
- You find out which batch is poisoned using only the information of who's left.

There are naturally trivial protocols : assign one student per batch and see who dies. This can be improved slightly, by choosing one batch we assign to no-one. Then if no-one dies that batch is the poisoned one (we go from $k = n$ to $k = n - 1$). Our goal is to minimize k and this can be done.

The first "smart" method is the following : let's arrange the chocolate batches in a square (or a rectangle). Let's assign one student to each row, and one to each column (they eat from each batch on their row or column). Then by finding which combination of two students died we can always find where the poisoned batch is. For $n = 1000$ we don't have a perfect square so we have two possibilities : getting a square of side 32, where the last row will be partially unused, or getting a rectangle of size 20×50. In the second case we need 70 students, and in the first 64. This is much better than the 999 students used previously, but still not optimal.

Remark. If you try to maximize $m \times n$ for a constant $m + n$ (or equivalently minimize $m + n$ for constant $m \times n$), the best way is always to have m and n extremely close.

We can improve the previous bound. Let's now consider putting the students in a cube. The cube has side 10, and each batch is designated by 3 students. Hence we use 30 students total. This can be generalized, and we can get $k\sqrt[k]{n}$. This is good, however is still not optimal. To get to the best bound, we can assign to each batch a number between 1 and n. Then the binary forms are all written with $\log_2(n)$ bits, all equal to 0 or 1. If we have one student eat from each batch that has a 1 in first position, we can know whether the number corresponding to the poisoned batch has a 1 in first position. By doing this for all $\log_2(n)$ positions, we can find the number of the batch using $\log_2(n)$ students. Here this is 10. To show the efficiency of this, if $n = 10^9$, the best method using $k\sqrt[k]{n}$ gives 57 students, as opposed to 30 for the logarithmic method.

This problem might seem artificial but was actually used during the US-Vietnam war. Many US soldiers had syphillis and the army needed to quickly find a way to find who was infected with as few tests as possible (because tests were expensive). As soon as we have multiple poisoned batches the techniques become much harder (generally using something called disjunct matrices). Finding an optimal protocol (or proving that we have the best possible) is still an open problem (our upper and lower bounds differ by a factor $\log(d)$ where d is the number of poisoned batches). To know more, this problem is known as non-adaptive group testing.