Working backwards

by Kwek Keng Huat

In many situations, a solution of a problem can be obtained by working backwards from the final configuration of the problem towards the beginning. The idea of working backwards is quite commonly used in solving dynamic programming problems, such as network, inventory and resource allocation problems. We shall illustrate how working backwards can make a seemingly difficult problem almost trivial to solve via the following well-known puzzles:

Example 1

Suppose that there are 40 match sticks on a table. I begin by picking up 1, 2, 3 or 4 sticks. Then it's my opponent's turn to pick up also 1, 2, 3 or 4 sticks; and after that it's my turn again to pick up 1 to 4 sticks. We continue in this manner until the last stick(s) is picked up. The player who picks up the last stick is the loser. How can I (the first player) be sure of winning?

Solution

It is clear that if at the end I can force my opponent's turn with 1 match stick left, I will win. Working one step backwards, if I can force my opponent's turn with 6 sticks left, I can be sure of winning. The reason is no matter how many (1 to 4) sticks he picks up when 6 sticks are left for him, when it comes to my turn, I will be able to leave the last stick to him. Similarly, working backwards, if I can force my opponents's turn to occur with 11, 16, 21, 26, 31 or 36 match sticks left on the table, I will win. Therefore if I pick up 40 – 36 = 4 matches at my first turn and at each successive turn, leave him with 31, 26, 21, 16, 11 or 6 sticks, I will be sure of winning.

If the one who picks up the last match stick is the winner, how should I (as the first player) modify the winning strategy?

Example 2

Given a 7-ounce cup and a 3-ounce cup, how do we return from a well with 5 ounces of water?

Solution

We note that if there is 1 ounce of water in the 3-ounce cup and we fill up the 7-ounce cup, we can then empty 2 ounces of water from the 7-ounce cup into the 3-ounce cup and we are done. To get 1 ounce of water in the 3-ounce cup we can simply fill up the 7-ounce cup. Then empty the 7-ounce cup twice into the 3-ounce cup and pour the remaining ounce from the 7-ounce cup into the 3-ounce cup. The above solution of the problem is summarised in the table below starting from state (9) to the final state (1).

state	Amount of water (in oz.) in 7-ounce cup	Amount of water (in oz.) in 3-ounce cup
(9)	5	0
(8)	5	3
(7)	7	1
(6)	0	1
(5)	1	0
(4)	1	3
(3)	4	0
(2)	4	3
(1)	7	0

Example 3

Given 20 identical coins and one counterfeit coin with a lighter weight. If the counterfeit looks exactly like a genuine one, how many weighings, using a perfectly balanced two-pan scale, do you need at the most in identifying the counterfeit?

Solution

With one weighing we can find the counterfeit coin from a lot of three as follows: put any two coins on the scale, one in each pan. If one of them is the counterfeit, we can identify it directly from the scale. If not, the scale will be balanced and certainly the third coin is the counterfeit. Now we can ask the question: What is the number of coins (including the counterfeit) in a lot so that in at most two weighings we can identify the counterfeit?

Working backwards, the second (final) weighing will be the one we described above with the lighter coin in a lot of three. Using the same principle, we can identify such a lot of three coins in one weighing if we work with a lot of 7 coins (by putting 3 coins in each pan, and of course if we were lucky we may identify the counterfeit in one weighing). Thus we can solve the given problem in 2 weighings at the most if there are just 7 coins. Now with 21 coins, we can easily decide in one weighing which lot of 7 coins contains the counterfeit — simply divide the 21 coins in 3 lots of 7 coins and apply the same principle which we had used earlier for a lot of 3 coins. We can therefore solve the problem in 3 weighings at the most.

If we did not know whether the counterfeit is heavier or lighter, how many weighings do you need at the most to identify it? Happy weighing! M²

Reference

Winston, W. L., *Operations Research : Applications and Algorithms,* 3rd ed. Duxbury Press, 1993.

Dr Kwek Keng Huat is a senior lecturer at the Department of Mathematics, National University of Singapore. His areas of interest are applications of dynamical systems and optimisation.

a winning strategy