Hierarchical Control

Laboratory

Exercise 5

Dynamic programming

The aim of this exercise is to apply the dynamic programming method to solve the shortest path finding problem.

1 Problem

In the fig. 1, a simple graph is presented. This graph is a model for a problem of finding the shortest path. Each edge in this graph is assigned a weight which corresponds to distance between the nodes of the edge. The problem is to find the shortest path from the node A to the node D.

In this problem, several stages can be found. All the nodes which names begin with the same letter, belong to the same stage. The stages are "natural": each path starts with the node A, and then goes to one of the B nodes, then to one of the C nodes, and finally to the D node. So, the path from the A node to the D node consists of four different stages.

The graph can be described with matrices. Each matrix corresponds to a different stage. In the matrix for i-th stage, each row corresponds to a different node in the i-th stage, and each column corresponds to a different node in the $(i+1)$-th stage. Elements of these matices are equal to the weights of the edges. The matrix for the first stage is presented below:

$$\underline{A}_1 = \begin{bmatrix} 3 & 4 & 5 \end{bmatrix}$$

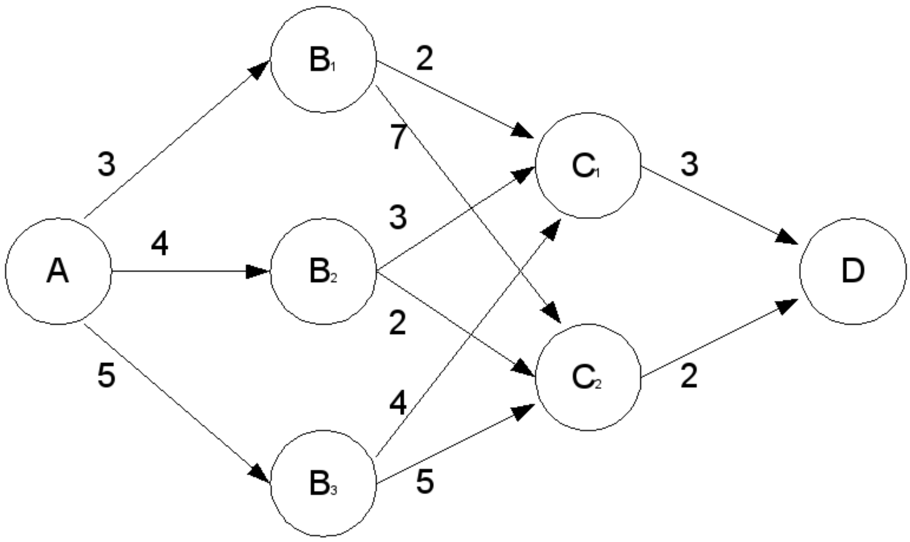

Figure 1: Graph.

The matrix has only one row, because this is only one node (A) in the first stage. The matrix has three columns, because there are three nodes in the next (second) stage. The distance between the first node in the first stage and the first node in the second stage is 3, the distance between the first (and the only one) node in the first stage and the second node in the second stage is 4.

The matrix for the second stage has three rows and two columns:

$$\underline{A}_2 = \begin{bmatrix} 2 & 7 \\ 3 & 2 \\ 4 & 5 \end{bmatrix}$$

The matrix for the third stage has two rows and just one column:

$$\underline{A}_3 = \begin{bmatrix} 3 \\ 2 \end{bmatrix}$$

Although the problem has four stages, no matrix is needed for the fourth stage.

As we can see, the problem can be described with three matrices.

A solution of such a problem can be described as a vector. The vector consists of as many elements, as many matrices are needed to describe the problem. The i-th element of the vector is the number of column to choose from the i-th matrix, and the number of row to choose from the $(i+1)$-th matrix. So, the path A, B_2, C_1, D can be denoted as the vector $[2\,1\,1]^T$. The length of this path can be calculated as the sum $A_1(1,2)+A_2(2,1)+A_3(1,1)$, where $A_i(j,k)$ denotes the element in the j-th row and the k-th column in the i-th matrix. Of course, in the first matrix $A_1(j,k)$ j is always 1, and in the last matrix $A_n(j,k)$ k is always 1, because we deal only with graphs of one starting and one target node.

2 Full-inspection method

The problem can be solved using full-inspection method. The algorithm is quite simple: check all the possible solutions (paths), and choose the best one.

In our case, in the first stage there are 3 different nodes to choose from, in the second stage there are 2 different nodes to choose from, and in the third stage this is onle one, final node. Therefore there are $3 \cdot 2 = 6$ different solutions of this problem. Let's analyse them all.

Path	Length
A, B_1, C_1, D	$3+2+3=8$
A, B_1, C_2, D	$3+7+2=12$
A, B_2, C_1, D	$4+3+3=10$
A, B_2, C_2, D	$4+2+2=8$
A, B_3, C_1, D	$5+4+3=12$
A, B_3, C_2, D	$5+5+2=12$

Table 1: Full-inspection approach.

As we can see, there are two different shortest paths: A, B_1, C_1, D and A, B_2, C_2, D, both of the length 8. The analysis was simple, and so is the method, but in case of graphs with more nodes the computational cost will be much higher.

3 Dynamic programming method

The same problem can be solved using the dynamic programming method. In this case, we have to decompose the problem into many simpler problems, and solve them in similar way.

Let's start with the last node D. The optimal path from A to D meets the condition $L_D = \min\{L_{C_1} + 3, L_{C_2} + 2\}$, where L_D is the length of the optimal path from the node A to the node D. The problem of finding the shortest path from the A node to the D node can be solved by solving two optimization problems: finding the shortest path from the A node to the C_1 node and finding the shortest path from the A node to the C_2 node.

The optimal path from A to C_1 meets the condition $L_{C_1} = \min\{L_{B_1} + 2, L_{B_2} + 3, L_{B_3} + 4\}$. The optimal path from A to C_2 meets the condition $L_{C_2} = \min\{L_{B_1} + 7, L_{B_2} + 2, L_{B_3} + 5\}$.

Now, we can solve the problems of finding the optimal paths from the A node to the nodes B_1, B_2 and B_3. It's very easy, since all these destination nodes are directly connected to the starting node. Therefore $L_{B_1} = 3$ (and the optimal path from the A node to the B_1 node is simply $P_{B_1} = (A,\ B_1)$), $L_{B_2} = 4$ and $L_{B_3} = 5$.

After solving the problems of finding the shortest paths from the A node to the nodes B_1, B_2 and B_3, the problems of finding the optimal path from the A node to the nodes C_1 and C_2 can be solved.

Let's start with the node C_1. The computations are simple: $L_{C_1} = \min\{\mathbf{3} + \mathbf{2}, 4 + 3, 5 + 4\} = 5$. So $P_{C_1} = (A,\ B_{\mathbf{1}},\ C_1)$. For the node C_2 it's also simple: $L_{C_2} = \min\{3 + 7, \mathbf{4} + \mathbf{2}, 5 + 5\} = 6$, and $P_{C_2} = (A,\ B_{\mathbf{2}},\ C_2)$.

Now, the problem can be finally solved. Since we know values of L_{C_1} and L_{C_2}, we can compute the value of L_D: $L_D = \min\{5 + 3, 6 + 2\} = 8$. In this case, both choices are optimal (but the optimal paths are different). The optimal path is either $P_D = (A,\ B_1,\ C_1,\ D)$ or $P_D = (A,\ B_2,\ C_2,\ D)$.

Note, that using the dynamic programming method, no calculations were performed for calculating paths which just can't be optimal. Thus, less calculations are required in order to obtain an optimal solution.

4 The exercise program

During the exercise, each section will be given a unique set of matrices and an identification number (used to identificate the set of matrices). The set of matrices describe the graph.

1. Find the shortest path from the starting node to the target node in the graph, using the full-inspection method.
2. Find the shortest path from the starting node to the target node in the graph, using the dynamic programming method.
3. Compare the computation times of those methods.