Water Allocation - I

Introduction

As discussed in previous lecture notes, in dynamic programming, a problem is handled as a sequential process or a multistage decision making process. In this lecture, we will explain how a water allocation problem can be represented as sequential process and can be solved using backward recursion method of dynamic programming.

Water allocation problem

Consider a canal supplying water to three fields in which three different crops are being cultivated as shown in figure 1. The maximum capacity of the canal is given as Q units of water. The three fields can be denoted as $i=1,2,3$ and the amount of water allocated to each field as x_i.

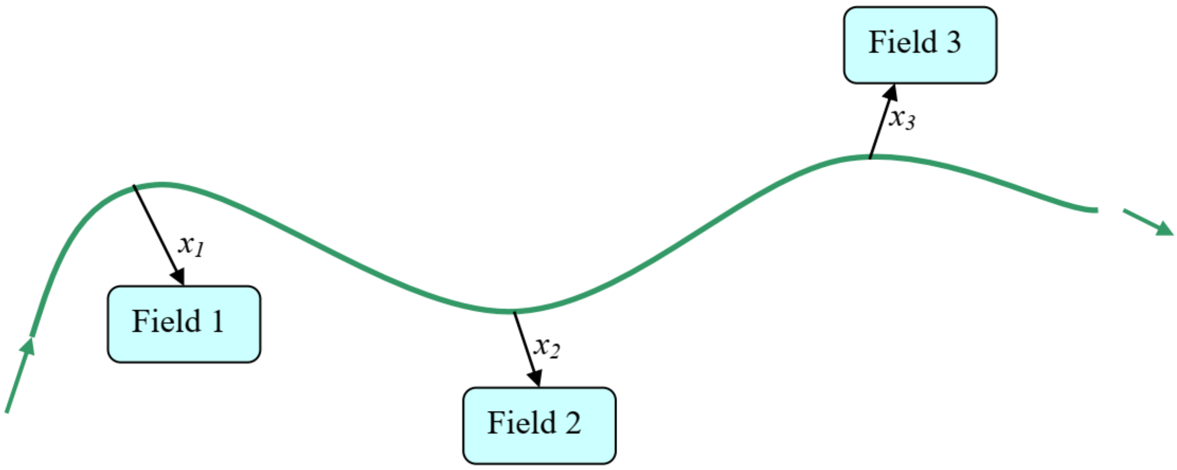

Fig. 1 Water allocation

The net benefits from producing the crops in each field are given by the functions below.

$$NB_1(x_1) = 5x_1 - 0.5x_1^2$$
$$NB_2(x_2) = 8x_2 - 1.5x_2^2$$
$$NB_3(x_3) = 7x_3 - x_3^2$$

The problem is to determine the optimal allocations x_i to each field that maximizes the total net benefits from all the three crops. This type of problem is readily solvable using dynamic programming.

The first step in the dynamic programming is to structure this problem as a sequential allocation process or a multistage decision making procedure. The allocation to each crop is considered as a decision stage in a sequence of decisions. If the amount of water allocated from the total available water Q, to crop i is x_i, then the net benefit from this allocation is $NB_i(x_i)$. Let the state variable S_i define the amount of water available to the remaining $(3-i)$ crops. The state transformation equation can be written as $S_{i+1} = S_i - x_i$ defines the state in the next stage. Figure 2 below shows the allocation problem as a sequential process.

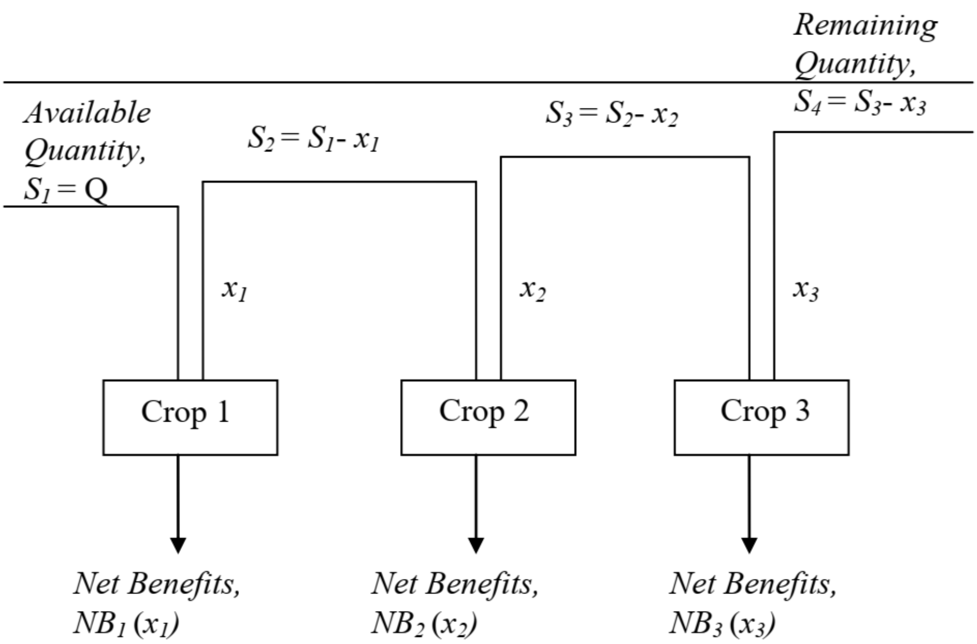

Fig. 2 Sequential process of allocation

The objective function for this allocation problem is defined to maximize the net benefits, i.e., $\max \sum_{i=1}^{3} NB_i(x_i)$. The constraints can be written as

$$x_1 + x_2 + x_3 \le Q$$
$$0 \le x_i \le Q \qquad for\ \ i = 1,2,3$$

Let $f_1(Q)$ be the maximum net benefits that can be obtained from allocating water to crops 1,2 and 3. Thus,

$$f_1(Q) = \max_{\substack{x_1+x_2+x_3 \le Q \\ x_1, x_2, x_3 \ge 0}} \left[\sum_{i=1}^{3} NB_i(x_i) \right]$$

Transforming this into three problems each having only one decision variable,

$$f_1(Q) = \max_{\substack{x_1 \\ 0 \le x_1 \le Q}} \left[NB_1(x_1) + \max_{\substack{x_2 \\ 0 \le x_2 \le Q - x_1 = S_2}} \left\{ NB_2(x_2) + \max_{\substack{x_3 \\ 0 \le x_3 \le S_2 - x_2 = S_3}} NB_3(x_3) \right\} \right]$$

<u>Backward recursive equations</u>

Considering the last term of this equation, let $f_3(S_3)$ be the maximum net benefits from crop 3. The state variable for this stage is S_3 which can vary from 0 to Q. Therefore,

$$f_3(S_3) = \max_{\substack{x_3 \\ 0 \le x_3 \le S_3}} NB_3(x_3)$$

Since $S_3 = S_2 - x_2$, $f_3(S_3) = f_3(S_2 - x_2)$. Thus $f_1(Q)$ can be rewritten as

$$f_1(Q) = \max_{\substack{x_1 \\ 0 \le x_1 \le Q}} \left[NB_1(x_1) + \max_{\substack{x_2 \\ 0 \le x_2 \le Q - x_1 = S_2}} \left\{ NB_2(x_2) + f_3(S_2 - x_2) \right\} \right]$$

Now, let $f_2(S_2)$ be the maximum benefits derived from crops 2 and 3 for a given quantity S_2 which can vary between 0 and Q. Therefore $f_2(S_2)$ can be written as,

$$f_2(S_2) = \max_{\substack{x_2 \\ 0 \le x_2 \le Q - x_1 = S_2}} \left\{ NB_2(x_2) + f_3(S_2 - x_2) \right\}$$

Again, since $S_2 = Q - x_1$, $f_1(Q)$ which is the maximum total net benefit from the allocation to the crops 1, 2 and 3, can be rewritten as

$$f_1(Q) = \max_{\substack{x_1 \\ 0 \le x_1 \le Q}} \left[NB_1(x_1) + f_2(Q - x_1) \right]$$

Now, once the value of $f_3(S_3)$ is calculated, the value of $f_2(S_2)$ can be determined, from which $f_1(Q)$ can be determined.

Forward recursive equations

The problem explained above can also be solved using a forward proceeding manner. Let the function $f_i(S_i)$ be the total net benefit from crops *1* to *i* for a given input of S_i which is allocated to those crops. Considering the first stage alone,

$$f_1(S_1) = \max_{\substack{x_1 \\ x_1 \le S_1}} NB_1(x_1)$$

Since, the value of S_1 is not known (except that S_1 should not exceed *Q*), the equation above has to be solved for a range of values from *0* to *Q*. Now, considering the first two crops together, with S_2 units of water available to these crops, $f_2(S_2)$ can be written as,

$$f_2(S_2) = \max_{\substack{x_2 \\ x_2 \le S_2}} \left[NB_2(x_2) + f_1(S_2 - x_2) \right]$$

This equation also should be solved for a range of values for S_2 from *0* to *Q*. Finally, considering the whole system i.e., crops 1, 2 and 3, $f_3(S_3)$ can be expressed as,

$$f_3(S_3) = \max_{\substack{x_3 \\ x_3 \le S_3 = Q}} \left[NB_3(x_3) + f_2(S_3 - x_3) \right]$$

Here, if it is given that the whole *Q* units of water should be allocated, then the value of S_3 can be taken as equal to *Q*. Otherwise, $f_3(S_3)$ should be solved for a range of values from *0* to *Q*.

The basic equations for the water allocation problem using both the approaches are discussed. A numerical problem and its solution will be described

Water Allocation - II

Introduction

In the previous topic , recursive equations for a basic water allocation problem were developed for both backward recursion and forward recursion. This lecture will further explain the water allocation problem by a numerical example.

Numerical problem and solution

Consider the example previously discussed with the maximum capacity of the canal as 4 units. The net benefits from producing the crops for each field are given by the functions below.

$$NB_1(x_1) = 5x_1 - 0.5{x_1}^2$$
$$NB_2(x_2) = 8x_2 - 1.5{x_2}^2$$
$$NB_3(x_3) = 7x_3 - {x_3}^2$$

The possible net benefits from each crop are calculated according to the functions given and are given in Table 1.

Table 1

x_i	$NB_1(x_1)$	$NB_2(x_2)$	$NB_3(x_3)$
0	0.0	0.0	0.0
1	4.5	6.5	6.0
2	8.0	10.0	10.0
3	10.5	10.5	12.0
4	12.0	8.0	12.0

The problem can be represented as a set of nodes and links as shown in the figure 1. The nodes represent the state variables and the links represent the decision variables.

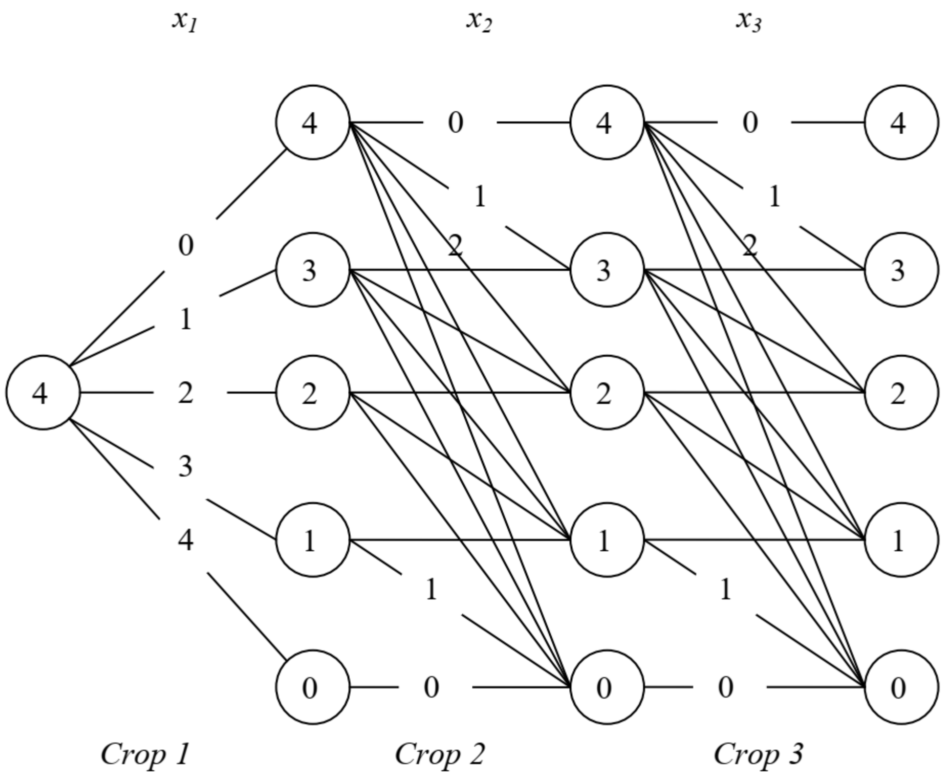

Fig. 1 Representation as nodes and links

The values inside the nodes show the value of possible state variables at each stage. Number of nodes for any stage corresponds to the number of discrete states possible for each stage. The values over the links show the different values taken by decision variables corresponding to the value taken by state variables. It may be noted that link values for all links are not shown in the above figure.

Solution using Backward Recursion:

Starting from the last stage, the suboptimization function for the 3[rd] crop is given as,

$$f_3(S_3) = \max_{\substack{x_3 \\ 0 \le x_3 \le S_3}} NB_3(x_3) \text{ with the range of } S_3 \text{ from 0 to 4.}$$

The calculations for this stage are shown in the table 2.

Table 2

State S_3	$NB_3(x_3)$ x_3: 0	1	2	3	4	$f_3(S_3)$	x_3^*
0	0					0	0
1	0	6				6	1
2	0	6	10			10	2
3	0	6	10	12		12	3
4	0	6	10	12	12	12	3,4

Next, by considering last two stages together, the suboptimization function is

$$f_2(S_2) = \max_{\substack{x_2 \\ x_2 \le S_2}} \left[NB_2(x_2) + f_1(S_2 - x_2)\right].$$

This is solved for a range of S_2 values from 0 to 4. The value of $f_3(S_2 - x_2)$ is noted from the previous table. The calculations are shown in Table 3.

Table 3

State S_2	x_2	$NB_2(x_2)$	$(S_2 - x_2)$	$f_3(S_2 - x_2)$	$f_2(S_2) = NB_2(x_2) + f_3(S_2 - x_2)$	$f_2^*(S_2)$	x_2^*
0	0	0	0	0	0	0	0
1	0	0	1	6	6	6.5	1
	1	6.5	0	0	6.5		
2	0	0	2	10	10	12.5	1
	1	6.5	1	6	12.5		
	2	10	0	0	10		
3	0	0	3	12	12	16.5	1
	1	6.5	2	10	16.5		
	2	10	1	6	16		
	3	10.5	0	0	10.5		

Table contd. on next page

4	0	0	4	12	12	20	2
	1	6.5	3	12	18.5		
	2	10	2	10	20		
	3	10.5	1	6	16.5		
	4	8	0	0	8		

Finally, by considering all the three stages together, the sub-optimization function is

$f_1(Q) = \max_{\substack{x_1 \\ 0 \le x_1 \le Q}} \left[NB_1(x_1) + f_2(Q - x_1) \right]$. The value of $S_1 = Q = 4$. The calculations are shown in the table 4.

Table 4

State $S_1 = Q$	x_1	$NB_1(x_1)$	$(Q - x_1)$	$f_2(Q - x_1)$	$f_1(S_1) = NB_1(x_1) + f_2(Q - x_1)$	$f_1^*(S_1)$	x_1^*
4	0	0	4	20	20	21	1
	1	4.5	3	16.5	21		
	2	8	2	12.5	20.5		
	3	10.5	1	6.5	17		
	4	12	0	0	12		

Now, backtracking through each table to find the optimal values of decision variables, the optimal allocation for crop 1, x_1^* = 1 for a S_1 value of 4. This will give the value of S_2 as $S_2 = S_1 - x_1 = 3$. From Table 3, the optimal allocation for crop 2, x_2 for $S_2 = 3$ is 1. Again, $S_3 = S_2 - x_2 = 2$. Thus, x_3* from Table 2 is 2. The maximum total net benefit from all the crops is 21. The optimal solution is given below.

$f^* = 21$

$x_1^* = 1$

$x_2^* = 1$

$x_3^* = 2$

<u>Solution using Forward Recursion:</u>

While starting to solve from the first stage and proceeding towards the final stage, the suboptimization function for the first stage is given as,

$$f_1(S_1) = \max_{\substack{x_1 \\ x_1 \le S_1}} NB_1(x_1) \ .$$

The range of values for S_1 is from 0 to 4.

<u>Table 5</u>

State S_1	x_1	$NB_1(x_1)$	$f_2^*(S_2)$	x_1^*
0	0	0	0	0
1	0	0	4.5	1
	1	4.5		
2	0	0	8	2
	1	4.5		
	2	8		
3	0	0	10.5	3
	1	4.5		
	2	8		
	3	10.5		
4	0	0	12	4
	1	4.5		
	2	8		
	3	10.5		
	4	12		

Now, considering the first two crops together, $f_2(S_2)$ can be written as,

$f_2(S_2) = \max\limits_{\substack{x_2 \\ x_2 \le S_2}} \left[NB_2(x_2) + f_1(S_2 - x_2)\right]$ with S_2 ranging from *0* to *4*. The calculations for this stage are shown in Table 6.

Table 6

State S_2	x_2	$NB_2(x_2)$	$(S_2 - x_2)$	$f_1(S_2 - x_2)$	$f_2(S_2) = NB_2(x_2) + f_1(S_2 - x_2)$	$f_2^*(S_2)$	x_2^*
0	0	0	0	0	0	0	0
1	0	0	1	4.5	4.5	6.5	1
	1	6.5	0	0	6.5		
2	0	0	2	8	8	11	1
	1	6.5	1	4.5	11		
	2	10	0	0	10		
3	0	0	3	10.5	10.5	14.5	1,2
	1	6.5	2	8	14.5		
	2	10	1	4.5	14.5		
	3	10.5	0	0	10.5		
4	0	0	4	12	12	18	2
	1	6.5	3	10.5	17		
	2	10	2	8	18		
	3	10.5	1	4.5	15		
	4	8	0	0	8		

Now, considering the whole system i.e., crops 1, 2 and 3, $f_3(S_3)$ can be expressed as,

$f_3(S_3) = \max\limits_{\substack{x_3 \\ x_3 \le S_3 = Q}} \left[NB_3(x_3) + f_2(S_3 - x_3)\right]$ with the value of $S_3 = 4$.

The calculations are shown in Table 7.

Table 7

State S_3	x_3	$NB_3(x_3)$	$S_3 - x_3$	$f_2(S_3 - x_3)$	$f_3(S_3) = NB_3(x_3) + f_2(S_3 - x_3)$	$f_3^*(S_3)$	x_3^*
4	0	0	4	18	18	21	2
	1	6	3	14.5	20.5		
	2	10	2	11	21		
	3	12	1	6.5	18.5		
	4	12	0	0	12		

In order to find the optimal solution, a backtracking is done. From Table 7, the optimal value of x_3* is given as 2 for the S_3 value of 4. Therefore, $S_2 = S_3 - x_3 = 2$. Now, from Table 6, the value of $x_2^* = 1$. Then, $S_1 = S_2 - x_2 = 1$ and from Table 5, for $S_1 = 1$, the value of $x_1^* = 1$. Thus, the optimal values determined are shown below.

$f^* = 21$
$x_1^* = 1$
$x_2^* = 1$
$x_3^* = 2$

These optimal values are same as those we got by solving using backward recursion method.

BIBLIOGRAPHY / FURTHER READING:

1. Bellman, R., *Dynamic Programming*, Princeton University Press, Princeton, N.J, 1957.
2. Hillier F.S. and G.J. Lieberman, *Operations Research*, CBS Publishers & Distributors, New Delhi, 1987.
3. Loucks, D.P., J.R. Stedinger, and D.A. Haith, *Water Resources Systems Planning and Analysis*, Prentice-Hall, N.J., 1981.
4. Rao S.S., *Engineering Optimization – Theory and Practice*, Fourth Edition, John Wiley and Sons, 2009.
5. Taha H.A., *Operations Research – An Introduction*, 8th edition, Pearson Education India, 2008.
6. Vedula S., and P.P. Mujumdar, *Water Resources Systems: Modelling Techniques and Analysis*, Tata McGraw Hill, New Delhi, 2005.