FACILITY LOCATION PROBLEM

The general facility location problem is: given a set of facility locations and a set of customers who are served from the facilities then:

- which facilities should be used
- which customers should be served from which facilities so as to minimise the total cost of serving all the customers.

Other factors often encountered here:

- customers have an associated demand with capacities(limits) on the total customer demand that can be served from a facility
- customers being served by more than one facility

The problems of facility size and facility location are very closely linked and should be considered simultaneously.

Let us consider *A Facility Location Problem:*

The Ansal Plaza in New-Delhi is interested in expanding its market efforts to other parts of the city. A pilot survey of the possible expansion sites has identified three location alternatives within a 50km radius of the current corporate marketing area. Table 1 summarizes potential revenues and authorised construction costs for four different building sizes that are being considered at each construction site.

Costs and potential revenues are expressed in crores of rupees. If Ansal Plaza can afford to invest only 21 crores in the total expansion project, what locations should be selected and how large a store should be constructed at each new location?

Table 1

BUILDING SIZES, REVENUES AND COSTS

	Size									
	B_1		B_2		B_3		B_4		None	
Site	R_1	C_1	R_2	C_2	R_3	C_3	R_4	C_4	R_5	C_5
I	0.50	1	0.65	2	0.80	3	1.40	5	0	0
II	0.62	2	0.78	5	0.96	6	1.80	8	0	0
III	0.71	4	1.20	7	1.60	9	2.00	11	0	0

The problem can be solved by using Dynamic Programming. The parameters to be specified are related to the definition of *stages, state variables, decision variables, the return function* and *the transition function.*

- *Stages:* The stages for this problem are identified by considering what primarily influences and controls the decision logic. The definition of stages is the element in dynamic programming that transforms the problem from a multivariable optimization problem to an equivalent series of single variable optimization problems. If a site is not chosen for any construction, then the decision of building size is not relevant. Hence the elements that control the entire decision process are the building sites. This defines a three-stage dynamic programming problem.

- *State Variables:* The state variable is that element in a dynamic programming problem that directly controls the decision variable selection. For this problem the state variable would be the amount of money remaining to be spent at each site as the decision process unfolds. Initially the amount is 21 crores.

- *Decision Variables:* In this problem the decision variable is what size building to construct. And this decision variable would automatically decide where to build.

- *Return Function:* The return function should reflect the intrinsic worth of making a decision at any given stage. In this case, the return function at each stage is the profit obtained by construction a certain size of store.

- *Transition Function:* The transition function reflects the change in available capital as the decision proceeds from stage to stage. In this problem the transition function relates the amount of money originally available at stage n to the amount of money available at stage $(n-1)$ after a decision d_n has been made at stage n.This is mathematically expressed as follows:

$$S_{n-1} = S_n - C_n.$$

where S_{n-1} =amount of capital available for construction at stage $n-1$.

S_n =amount of capital originally available for construction at stage n.

C_n = cost of building(directly related to d_n).

The Recursive Formula. The Dynamic programming recursive formula is as follows:

$$f_n^*(S_n) = \max_{4 \geqslant d_n \geqslant 0} \{f_n(S_n, d_n) + f_{n-1}^*(S_{n-1})\} \qquad n = 1, 2, 3, 4$$

where

$$S_{n-1} = S_n - C_n$$

$$f_n(S_n, d_n) = r_n(S_n, d_n) = R_n$$

The decision to not build at site n is represented by $d_n = 0$. The return function represents the profit realized by building a store of size d_n which is determined by the amount of capital S_n, available at the stage where a decision is made; $R_n = r_n(S_n, d_n)$. The entire problem can be carried out through decision tables.