CHAPTER IV

EVALUATION OF SERIES SYSTEMS THROUGH DYNAMIC PROGRAMMING TECHNIQUE
4.1 INTRODUCTION

Designing reliability into a system is often required when the system is to perform an important mission for industry/military or space flight and where malfunction (failure) of a constituent part causes heavy losses in terms of money/time.

In order that a multistage system having many subsystems or stages in series have a failure free operation, the subsystems’ reliability should be very high. While dealing with evaluation studies of series systems, one often finds following ways to increases system reliability.

(a) To increase the reliability of the constituent subsystems (components) i.e., to develop highly reliable components. This in turn increases cost of the system. Further, as the number of series components increases, the overall system reliability decreases. In fact, after a certain stage, even a marginal increase in a component’s reliability may result in tremendous cost.

(b) To allocate certain components in redundancy in an optimal manner at various stages of a series system so that the overall system reliability is increased.

(c) To suggest some maintenance policies for an existing system with a view to increase system reliability.

A little concentration on the above discussion reveals that (a) and (b) are useful at the design stage, whereas (c) plays an important role in the evaluation/maintenance problems of the system. More elaborately, the methods developed to meet the requirements of (a) and (b) lose their importance,
once a system is designed and put into operation. Thus, at the latter stage viz., operation stage, one must look for methods that may help in better maintenance of the system. Besides, problems at the maintenance stage are still more important because, sometimes state of the art does not allow a further increase in components' reliability inspite of one's willingness to bear extraordinary amount of money.

There has been a large number of papers in the above directions. [Barlow and Proschan (1965), Bellman and Dreyfus (1958, 1962), Kettle (1962), Proschan and Bray (1965), Fyffe et al. (1968), Hadley (1964), Messigner and Shooman (1970), Jensen (1970), Woodhouse (1972) and Misra (1971a).

Misra (1971a) gave a dynamic programming formulation to determine optimal number of redundant components at each stage in a series system. Lagrangian Multiplier Technique was used to solve the problem. Initially, the problem was formulated under one constraint. However, redundancy allocation under two constraints has also been dealt with.

Bellman and Dreyfus (1962) have discussed the reliability R of a system containing several subsystems, each of which has a redundancy. They maximized the reliability subject to a cost constraint. Kettle (1962) has discussed an approach of least cost allocation with a view to minimize sum of subsystems unreliabilities instead of maximizing system reliability. This was extended by Proschan and Bray (1965) and Barlow and Proschan (1965) to the case of two constraints.

In formulating and solving redundancy allocation problems through dynamic programming with a view to optimize system reliability, number of
constraints causes difficulty because of increased dimensionality. This difficulty has been successfully overcome by employing Lagrangian Multiplier Technique by Bellman and Dreyfus (1958), Fyffe et al. (1968), Hadley (1964), Messigner and Shooman (1970). Kettle (1962) and Woodhouse (1972) have used the concept of dominating sequences for the purpose.

However, (c) has also attracted attention of some of the workers. Kumar and Lal (1980) studied a 2-unit cold standby redundant system and determined an optimal maintenance policy which maximizes expected profit of the system assuming that there is no discounting. Later on Kumar and Kapoor (1981a) extended the procedure to cover the case of discounted cost structure.

It may be worthwhile to point out that in most of the papers cited above, the methods of analysis are based on dynamic programming formulation. In fact, dynamic programming is a very powerful technique and its spectrum is quite wide in that it enables solution procedures for a fairly large reliability situations. Howard's (1971) policy iteration method is also a version of dynamic programming.

In the present chapter we discuss the following problem for series systems, viz., Optimum cost allocation under a constraint on the cost in the subsequent section.
4.2 OPTIMUM COST ALLOCATION IN A SERIES SYSTEM UNDER A CONSTRAINT ON THE COST

System Model

There is an $n$-component series system. The problem is to allocate cost to each component with a view to maximize system reliability subject to total cost. It is assumed that there exist a number of components with varying costs and reliability which are capable of performing the same system task. The reliability of each component is assumed to be monotonically increasing function of its cost.

Notation

\[ R_n(C_n): \text{ reliability of component } n \]
\[ C_n : \text{ cost of component } n \]
\[ R : \text{ system reliability} \]
\[ N : \text{ number of components in the system.} \]

System Reliability

Then, the system reliability $R$ is given as

\[ R(C_1, C_2, ..., C_N) = \prod_{n=1}^{N} R_n(C_n) \]

Now, we obtain a suitable expression for $R_n(C_n)$. Beriphol (1961) has expressed the cost of a component as a function of its reliability:

\[ C_n = \frac{K_{1n}}{1-R_n(C_n)} \exp \left\{ -K_{2n} \{1-R_n(C_n)\} \right\} \quad (4.2.1) \]
where \( K_{1n} \) and \( K_{2n} \) are constants.

Equation (4.2.1) cannot be solved for \( R_n(C_n) \).

Therefore, we devise a function given below in (4.2.2) which leads to approximately the same set of values for reliability and cost as is given by (4.2.1) for \( 0.9 < R_n(C_n) < 1 \).

\[
R_n(C_n) = a_n + b_n \log C_n \tag{4.2.2}
\]

where \( 0 < a_n < 1 \), \( b_n > 0 \) and \( \log \) is to the base e.

Now the actual problem becomes

Maximize \( R(C_1, C_2, \ldots, C_N) \)

subject to

\[
\sum_{n=1}^{N} C_n = C, \quad C_n > 0
\]

i.e., Maximize \( \left[ \prod_{n=1}^{N} (a_n + b_n \log C_n) \right] \)

subject to

\[
\sum_{n=1}^{N} C_n = C, \quad C_n > 0 \tag{4.2.3}
\]

Dynamic Programming Formulation

With a view to maximize system reliability in \( N \) stages, define stage \( j \) (\( j = 1, 2, \ldots, N \)) to consist of \( j \) components counted from the last i.e., \( N \) to
(N-j+1).

Define

\[ F(j, C_{N-j+1}, C) = \text{Reliability of stage } j \text{ when cost } C_{N-j+1} \text{ is allocated to component } N-j+1, \ C-C_{N-j+1} \text{ is optimally allocated to the remaining } j-1 \text{ components contained in stage } j, \ j = 1, 2, \ldots, N. \]

\[ F^* (j, C) = \text{Maximum of } F(j, C_{N-j+1}, C) \text{ with respect to } \]

\[ C_{N-j+1}, \ 0 < C_{N-j+1} < C, \ j = 1, 2, \ldots, N. \]  \hfill (4.2.4)

Evidently, \[ F^* (N, C) = \text{Max } [R(C_1, C_2, \ldots, C_N)] \]

with respect to \( C_1, C_2, \ldots, C_N \)

subject to

\[ C_1 + C_2 + \ldots + C_N = C \]

Bellman's Principle of optimality results in the following functional equations:

\[ F^* (j, C) = \text{Max } [R_{N-j+1} (C_{N-j+1}) F^* (j-1, C-C_{N-j+1})] \]

with respect to \( C_{N-j+1}, \ 0 < C_{N-j+1} < C, \ j = 2, \ldots, N \)

\[ F^* (1, C) = \text{Max } [R_N (C_N)] \]

with respect to \( C_N, \ 0 < C_N < C \)  \hfill (4.2.5)

The optimum values of \( C_1, C_2, \ldots, C_N \) can be obtained by solving above functional equations.

If standard dynamic programming procedures are followed to solve (4.2.5) the solution is quite voluminous and cumbersome. Therefore, it is
desirable to achieve approximate solutions. We describe below a simple approach to obtain approximate but explicit expression for optimal costs through an illustration.

Illustration

Let

\[ N = 3, \]

From (4.2.5)

\[ F^*(1, C) = \text{Max} \{ R_3(C_3) \} \text{ with respect to } C_3, \, 0 < C_3 < C \]

\[ = \text{Max} \{ a_3 + b_3 \log C_3 \} \text{ with respect to } C_3, \, 0 < C_3 < C \]

\[ = a_3 + b_3 \log C \]  \hspace{1cm} (4.2.6a)

Evidently,

\[ F(2, C_2, C) = R_2(C_2) F^* (1, C-C_2) \]

\[ = (a_2 + b_2 \log C_2) [ a_3 + b_3 \log (C-C_2) ] \] \hspace{1cm} (4.2.6b)

Let

\[ K = C_2/C \] \quad (0 < K < 1) \]

\[ g_i = a_i + b_i \log C, \quad i = 1, 2, 3 \]

Then

\[ F(2, CK, C) = g_2g_3 + b_2b_3 \log (1-K) + b_2g_3 \log K + b_2b_3 \log K \cdot \log (1-K) \]

Neglecting \( b_2b_3 \log K \log (1-K) \) as \( b_2 \) and \( b_3 \) are small quantities.

\[ F(2, CK, C) = g_2g_3 + b_2g_3 \log (1-K) + b_2g_3 \log K = \phi(K) \]

The maximum value of \( K \) denoted by \( K^* \) is obtained by differentiating
\( \varphi(K) \) with respect to \( K \) and equating it to zero i.e., \( \frac{d \varphi(K)}{dK} = 0 \) which gives

\[
K^* = \frac{b_2g_3}{b_3g_2 + b_2g_3}
\]

To decide that \( K^* \) provides maximum, we further examine the sign of

\[
\frac{d^2 \varphi(K)}{dK^2} \bigg|_{K=K^*} \text{ i.e.,}
\]

\[
\frac{d^2 \varphi(K)}{dK^2} = -\left[ \frac{b_2g_3}{K^2} + \frac{b_3g_2}{(1-K)^2} \right]
\]

Thus,

\[
\frac{d^2 \varphi(K)}{dK^2} \bigg|_{K=K^*} < 0
\]

So \( K^* \) maximizes \( \varphi(K) \) and hence

\[
F^*(2, C) = \varphi(K^*) = F(2, CK^*, C)
\]

On similar lines for third stage

\[
F(3, C_i, C) = R_1(C_i) F^*(2, C-C_i)
\]

\[
= [a_1 + b_1 \log C_i] [a_2 + b_2 \log (C-C_i) + b_2 \log K^*] .
\]

\[
[a_3 + b_3 \log (C-C_i) + b_3 \log (1-K^*)]
\]

Let

\[
L = C_i/C \quad (0 < L < 1)
\]

Then
\[ F(3, \text{CL, C}) = \left[ a_1 + b_1 \log C + b_1 \log L \right] \left[ a_2 + b_2 \log C + b_2 \log (1-L) + b_2 \log K' \right] \left[ a_3 + b_3 \log C + b_3 \log (1-L) + b_3 \log (1-K') \right] \]
\[ = [h_1 + b_1 \log L] [h_2 + b_2 \log (1-L)] [h_3 + b_3 \log (1-L)] \]

where

\[ h_1 = g_1 \]
\[ h_2 = g_2 + b_2 \log K' \]
\[ h_3 = g_3 + b_3 \log (1-K'). \]

So, on neglecting the terms containing \( b_1 b_2, b_2 b_3, b_1 b_3 \) and \( b_1 b_2 b_3 \) as \( b_1, b_2, b_3 \) are small quantities.

\[ F(3, \text{CL, C}) = h_1 h_2 h_3 + h_1 h_2 b_3 \log (1-L) + h_1 b_2 h_3 \log (1-L) \]
\[ + b_1 h_2 h_3 \log L \]
\[ = \theta(L) \text{ (say)}. \]

The maximum value of \( L \) denoted by \( L' \) is obtained by differentiating \( \theta(L) \) with respect to \( L \) and equating it to zero i.e.,

\[ \frac{d\theta(L)}{dL} = 0 \]

which gives

\[ L' = \frac{b_1 h_2 h_3}{b_1 h_2 h_3 + b_2 h_1 h_3 + b_3 h_1 h_2} \]

To decide \( L' \) provides maximum, we further examine the sign of
\[
\frac{d^2 \theta(L)}{dL^2} \bigg|_{L=L^*} \text{ i.e.,}
\]
\[
\frac{d^2 \theta(L)}{dL^2} = - \left[ \frac{b_1 h_2 h_3}{L^2} + \frac{b_2 h_4 h_3 + b_3 h_1 h_2}{(1-L)^2} \right]
\]

Thus \( \frac{d^2 \theta(L)}{dL^2} \bigg|_{L=L^*} < 0 \)

So, \( L^* \) maximizes \( \theta(L) \).

Optimum values of \( C_1, C_2, C_3 \) denoted by \( C_1^*, C_2^*, C_3^* \), are obtained as follows

\[
C_1^* = CL^*, \quad C_2^* = CK^* (1-L^*), \quad C_3^* = C(1-K^*) \quad (1-L^*).
\]

**Illustration**

We now illustrate the results numerically. Let us take \( C=15,00,000 \).

Further, assume the following data taken from Beriphol (1961)

<table>
<thead>
<tr>
<th>( R_1 )</th>
<th>( C_1 )</th>
<th>( R_2 )</th>
<th>( C_2 )</th>
<th>( R_3 )</th>
<th>( C_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>.999</td>
<td>1820</td>
<td>.999</td>
<td>1030</td>
<td>.999</td>
<td>6310</td>
</tr>
<tr>
<td>.995</td>
<td>100</td>
<td>.995</td>
<td>100</td>
<td>.995</td>
<td>200</td>
</tr>
</tbody>
</table>

Using the above set of values of \( R_n \) and \( C_n \) (\( n=1, 2, 3 \)), the values of \( a_n \) and \( b_n \) (\( n=1, 2, 3 \)) are obtained from (4.2.2) as

\[
a_1 = .97910, \quad a_2 = .97530, \quad a_3 = .98086
\]

\[
b_1 = .00138, \quad b_2 = .00171, \quad b_3 = .00116
\]

Consequently,

\[
K^* = .5937, \quad L^* = .3246
\]
and

\[ C_1^* = 4,86,900; \quad C_2^* = 6,03,503.67; \quad C_3^* = 4,09,596.33 \]

Evidently in this case the maximum attainable reliability of the system due to optimum allocation of cost to each component for a given total cost of 15 lakhs is .99111.