Optimal Maintenance for a System with Weibull Distribution of Failure Using Search Technique Approach

5.1. Introduction

Consider a system composed of \( m \) independent subsystems (subsystem1, subsystem2, ..., subsystem \( m \)) connected in series. Let each subsystems \( i \) contain a set of \( n_i \) independent and identical components connected in parallel. In addition, each component, each subsystem and the whole system can be in only one of the two states: proper functioning state or failed state.

Assume that the system is required to perform a sequence of identical missions. Let \( r_i \) be the reliability of a component in \( i^{th} \) subsystem and let \( b_i \) denote the number of components in the subsystem \( i \) functioning at start of the mission. Since a subsystem consists of a parallel arrangement of its components, the reliability of the system can be written as

\[
R = 1 - \prod_{i=1}^{m} (1 - r_i)^{b_i}
\]  

(5.1)

At the completion of a particular mission, each component in the system is either functioning or failed. The failed components (if there are any) are repaired to a functioning condition if there is a possibility that the repaired component works as a new one otherwise it is replaced by new one before next mission starts. Ideally all the failed components are repaired prior to the beginning of the next mission.
However, it may not be possible to repair all the failed components. Let $t_i$ denote the amount of time required to repair a component of subsystem $i$, and let $k_i$ denote the number of failed components in subsystem $i$ at the end of the mission. The total time required to repair all the failed components in the system prior to the next mission is given by

$$ T = \sum_{i=1}^{m} t_i k_i \quad (5.2) $$

Suppose that the total amount of time allotted to perform the repair of failed components between consecutive missions is $T_0$ time units. If $T_0 < T$, then all the failed components may not be repaired prior to the beginning of the next mission. In such cases, a method is needed to decide how many failed components should be repaired in a subsystem prior to the next mission and the rest be left in a failed condition. This process is referred to as selective maintenance.

Rice et al. (1998) define the following mathematical programming model for this selective maintenance problem. Let $d_i$ denote the number of components in subsystem $i$ to be repaired prior to the next mission. Let $n_i$ denote the total number of components in the $i^{th}$ subsystem. We define

$$ b_i = (n_i - k_i) + d_i \quad (5.3) $$

The mathematical programming model is given as

$$ P: \quad \text{Max } R = \prod_{i=1}^{m} \left\{ 1 - (1 - r_i)^{b_i} \right\} $$
Subject to

\[ \sum_{i=1}^{m} t_i d_i \leq T_0 \]

\[ 0 \leq d_i \leq k_i \]

and \[ d_i \text{ integers, } i=1,2,\ldots,m \]

Cassady, Pohl et al. (2001) extend this problem by permitting subsystems to be comprised of non identical components in any structure, adding a second resource constraint (representing maintenance cost), and creating two additional selective maintenance formulations that minimize resource consumption as the objective function and include mission reliability as a constraint. Cassady, Murdock et al. (2001) extend problem \( P \) in two ways. First, the life distributions of system components are specified to be Weibull distributions. Second, the decision maker is given multiple maintenance options: minimal repair on failed components, replacement of failed components and replacement of functioning components (preventive maintenance). M. Jain and G.C. Sharma (2004) work with Search Technique and find the optimum number of redundancy components. In this chapter we use Search Technique for finding the optimum number of repairable components in the series-parallel system. The work in this chapter is published in *International Review of pure & Applied Mathematics*, see khan, M. Faisal et al. (2011).

### 5.2. The definition of the problem

For the given system, the reliability can be obtained as
Optimal Maintenance Using Search Technique

\[ R = \prod_{i=1}^{m} \left\{ 1 - (1 - r_i)^{n_i - k_i + d_i} \right\} \quad (5.4) \]

and repair time constraint is obtained as

\[ T = \sum_{i=1}^{m} t_i \left[ d_i + \exp(\theta_i d_i) \right] \leq T_0 \quad (5.5) \]

where \( t_i \) is the time of repairing a component in subsystem \( i \), \( \theta_i \) is a constant parameter for the \( i^{th} \) subsystem, and \( \exp(\theta_i d_i) \) is the additional time amount due to the interconnection between parallel components.

Let \( C_0 \) be the total repair cost allotted for the system. The repair cost constraints is

\[ C = \sum_{i=1}^{m} c_i \left[ d_i + \exp(\beta_i d_i) \right] \leq C_0 \quad (5.6) \]

where \( c_i \) is the cost of repairing a component in subsystem \( i \), \( \beta_i \) is a constant parameter for the \( i^{th} \) subsystem, and \( \exp(\beta_i d_i) \) is the additional cost amount due to the interconnection between parallel components. Note that we have considered the following assumptions: for our model:

1. Assumption 1: the system consists of subsystems, each of which can work properly if at least one of its components is operational.
2. Assumption 2: all components states are independent.
3. Assumption 3: reliability, cost, time and weight/size of each component in a subsystem are same.
4. The L.H.S in each of the two constraints (5.5) and (5.6) is an increasing function of \( d_i \)

The formal description of reliability allocation problem is as follows:
We have to find the optimum number of repairable components $d_i; (0 \leq d_i \leq a_i)$, $i=1,2,\ldots,m$, which maximize

$$R = \prod_{i=1}^{m} 1 - (1 - r_i)^{n_i - k_i + d_i}$$

Under the constraints

$$\sum_{i=1}^{m} t_i [d_i + \exp(\theta_i d_i)] \leq T_0$$

$$\sum_{i=1}^{m} c_i [d_i + \exp(\beta_i d_i)] \leq C_0$$

and $0 \leq d_i \leq k_i, d_i \in I, i=1,2,\ldots,m$.

5.3. A Search Technique approach to the optimum allocation of repairable components in system reliability

We know that reliability of a component is the probability of its survival for a complete mission. So taking $p_i = r_i$, $q_i = (1 - r_i)$ and $x_i = n_i - k_i + d_i$.

The equation (5.1) can be written as

$$R = \prod_{i=1}^{m} (1 - q_i^{x_i})$$

Taking the log on both sides

$$\log R = \sum_{i=1}^{m} \log(1 - q_i^{x_i})$$

Differentiating w.r.to $d_i$, we get

$$\frac{1}{R} \frac{dR}{dd_i} = -\frac{q_i^{x_i} \log q_i}{(1 - q_i^{x_i})}$$
Or \[
\frac{dR}{dd_i} = -q_i^n \log q_i \prod_{i=1}^{m} (1-q_i^{n_i})
\]

On neglecting the higher terms of \((1-q_i^{n_i})\) we get

\[
\frac{dR}{dd_i} = -q_i^{n_i-k_i+d_i} \log q_i, \ i = 1, \ldots, m
\] (5.7)

Now let us suppose that we repair a fixed number of failed components of

subsystem number one then the optimal value of \(d_i\) is fixed as \(d_i^* \approx \frac{k_i}{2}\). As shown

in M. Jain et al. (2004). The other values of \(d_i, i = 2, \ldots, m\) are compute from

\[
n_i - k_i + d_i = \log \left[ \frac{q_i^{n_i-k_i+d_i} \log q_i}{\log q_i} \right], \ i = 2, \ldots, m
\] (5.8)

Algorithm:

**Step1.** The number of repairable components for first subsystem \(d_i\) is chosen by

guess. The best policy is to choose \(d_i^* \approx \frac{k_i}{2}\).

**Step2.** The number of repairable components \(d_i\) are calculated for \(i = 2, \ldots, m\) by

using (5.8), and go to step 3.

**Step3.** We compute the LHS of the constraints (5.5) and (5.6).

(i) If \((T-T_0) < 0\) and/or \((C-C_0) < 0\) then go to step 4.

(ii) If \((T-T_0) > 0\) and/or \((C-C_0) > 0\) then go to step 5.

(iii) If \((T-T_0) = 0\) and \((C-C_0) = 0\) then go to step 6.

**Step4.** Take \(d_i = d_i + 1\) and go to step 1.
Step 5. Take $d_i = d_i - 1$ and go to step 1.

Step 6. The optimal solution is arrived at and we compute system reliability, repair cost, repair time for the system.

5.4. Numerical illustration

Consider a system consisting of $m = 3$ subsystems. Let the total available time $T_0$ between two missions for the repair of the components be 135 minutes and the total repair cost be Rs. 500. The remaining parameters for the system are provided in table 5.1.

<table>
<thead>
<tr>
<th>Subsystems</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n_i$</td>
<td>7</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>$r_i$</td>
<td>0.6</td>
<td>0.7</td>
<td>0.9</td>
</tr>
<tr>
<td>$a_i$</td>
<td>5</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>$c_i$</td>
<td>25</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>$t_i$</td>
<td>5</td>
<td>10</td>
<td>15</td>
</tr>
</tbody>
</table>

Table 5.1: The values of the various constraints for the system

Now let us find the repairable components for each subsystem in order to maximize the system reliability and the satisfying the cost and time constraints for repairable components by using search technique.

By using the data in table 5.1 we obtain the following non-linear integer programming:

$$Max \ Z = \left[ 1 - (1 - 0.6)^{(2+d_1)} \right] \times \left[ 1 - (1 - 0.5)^{(2+d_2)} \right] \times \left[ 1 - (1 - 0.9)^{(4+d_1)} \right]$$

Subject to

$$25[d_1 + \exp(0.25d_1)] + 30[d_2 + \exp(0.25d_2)] + 40[d_3 + \exp(0.25d_3)] \leq 500$$
$$5[k_1 + \exp(0.25k_1)] + 10[k_2 + \exp(0.25k_2)] + 15[k_3 + \exp(0.25k_3)] \leq 135$$

$$0 \leq d_1 \leq 5, \quad 0 \leq d_2 \leq 5, \quad 0 \leq d_3 \leq 3$$

and integers.

**First Iteration**

Step 1: The number of repairable components for the first subsystem is to be chosen at the initial stage as $d_1 = 2$.

Step 2: Using the equation (5.8) $d_2 = 1.2748 \cong 1$

and $d_3 = -2 < 0 \cong 0$.

Step 3: The value of the Objective function is:

$$Z = \left[1 - (1 - 0.6)^{(2+d_1)}\right] \times \left[1 - (1 - 0.7)^{(2+d_2)}\right] \times \left[1 - (1 - 0.9)^{(4+d_3)}\right] = 0.9444$$

The LHS of Cost Constraints is:

$$25[d_1 + \exp(0.25d_1)] + 30[d_2 + \exp(0.25d_2)] + 40[d_3 + (0.25d_3)] = 199.7388$$

The LHS of Time Constraints is:

$$5[d_1 + \exp(0.25d_1)] + 10[d_2 + \exp(0.25d_2)] + 15[d_3 + (0.25d_3)] = 56.083$$

So case (i) arises and we go to step 4.

Step 4: The number of repairable components for the first subsystem is taken $d_1 = 2 + 1 = 3$ and go to step 2.

**Iteration 2:**

Step 2: $d_2 = 2.0755 \cong 2$ and $d_3 = -1.5910 \cong -2 < 0 \cong 0$.

Step 3: Objective function value is 0.9816

LHS of Cost Constraints is 277.3866
LHS of Time Constraints is 77.07221
Again Case (i) arises and we go to step 4.
Step4: The number of repairable components for the first subsystem is to be chosen now as \( d_1 = 3+1 = 4 \).

**Iteration 3:**

Step2: using equation (5.8) \( d_2 = 2.8830 \leq 3 \) and \( d_3 = -1.761 \leq -2 \leq 0 \geq 0 \).

Step3: Now objective function value is 0.9933
The LHS of Cost Constraints is 361.4670
The LHS of Time Constraints is 99.7614.
Again Case (i) arises and we go to step 4.
Step4. The number of repairable components for the first subsystem is now chosen as \( d_1 = 5 \).

**Iteration 4:**

Step2. Using equation (5.8) \( d_2 = 3.6254 \leq 4 \) and \( d_3 = -0.7782 < 0 \geq 0 \).

Step3. The Objective function value is 0.9973.
The LHS of Cost Constraints is 453.8070
The LHS of Time Constraints 124.6345
Again Case (i) arises and we got step 4.
Step 4. The number of repairable components for first subsystem is to be chosen as \( d_1 = 6 \).
Iteration 5:

Step 2. Using equation (5.8) \( d_2 = 4.4 \pm 4 \) and \( d_3 = -0.3618 < 0 \pm 0 \).

Step 3. The Objective function value is 0.9985.

The LHS of Cost Constraints is 503.59

The LHS of Time Constraints 137.838

In this iteration i.e. for \( d_1 = 6, \ d_2 = 4 \) and \( d_3 = 0 \), the constraints are not satisfied at this allocation of repairable components. Hence the allocation of repairable components are \( d_1 = 5, \ d_2 = 4 \) and \( d_3 = 0 \).

The above nonlinear integer programming problem represented by, is solved by using “Search Technique Method”. Since \( d_1, d_2, d_3 \geq 0 \), are repairable components respectively, it must be an integer. The optimal Solution so obtained after fourth iterations are as follows.

\[ R(t) = 0.9973, \ d_1 = 5, \ d_2 = 4, \ d_3 = 0. \]