Maintenance Optimization Using Fuzzy Logic Controlled Genetic Algorithm

IMANE MAATOUKA, NAZIR CHEBBOb, IMAN JARKASSb, ERIC CHATELETC
aEDST, Lebanese University, LEBANON
bIUT-Saida, Lebanese University, LEBANON
ICD, UMR 6281 CNRS, LM2S, University of Technology of Troyes, FRANCE
imane.maatouk@yahoo.com (Imane Maatouk), nchebbo@ul.edu.lb (Nazir Chebbo),
ijarkass@ul.edu.lb(Iman Jarkass), eric.chatelet@utt.fr (Eric Châtelet)

Abstract: - In the field of reliability management, using preventive maintenance is one of the major operations. Optimizing the maintenance policy is required so as to provide a low-cost and efficient function of the system. This paper deals with the problem of maintenance optimization in a multi-states series-parallel system. The objective is to optimize for each system component the maintenance policy minimizing a cost function of the system under the constraint of required availability and for a specified period.

Since inspections in the beginning of the life of a component, when the component is very reliable are not efficient, the maintenance policy should identify the dates of first inspection of each system component.

We propose using an evolutionary algorithm called fuzzy logic controlled genetic algorithm (FLC-GA) to solve this optimization problem. Two controllers are used to adaptively adjust the crossover and the mutation probabilities based on the fitness function and on the degree of population diversity.

Simulation results are presented for the proposed method which is compared to a genetic algorithm with fixed crossover and mutation probabilities. The experimental results show the advantages and the efficiency of FLC-GA.

Key-Words: - Fuzzy logic, genetic algorithm, multi-states system, optimization, preventive maintenance.

1. Introduction

Many researchers have studied systems maintenance optimization during their lifetime [1][2][3][4][5][6][7]. These studies aim to enhance the system functionality by increasing its expected performance.

The main interest in the industrial world is to find a balance between the system safety and availability, and the system cost. The reliability allocation involves the design of reliable systems with minimal costs by adopting an applicable and effective methodology. In this family of allocation models, it can be distinguished the approaches based on the reliability maximization under cost constraints, the cost minimization under the constraint of reliability, or the bi-objective ones.

The Preventive maintenance (PM) consists of actions that improve the condition of system elements before they fail. The PM actions either return the element to its initial condition, then the element becomes ‘as good as new’, or reduce the age of the element.

The Optimization of the policy of preliminary planned PM actions is the subject of much research activities. This aim of this paper is to optimize, for each component of a multi-states series-parallel system, the periodic maintenance policy minimizing the cost function, with respect to the system availability constraint \( A_0 \) and for a given mission time \( T_M \). The objective in most researches of periodic PM optimization is to determine the period of maintenance of each system component, but they do not consider the specificity of the first time inspection [5][6]. In the beginning of its life, component is very reliable and its maintenance is not efficient. As a result, the first time inspection should be determined.

In this paper, the first time inspection is determined based on Birnbaum important factor. Component becomes “as good a new” after each inspection. The PM policy is optimized using the genetic algorithm (GA) based on a fuzzy decision making.

Applications of genetic algorithm for optimization problems are widely known [8]. Fuzzy logic may be used to dynamically control the parameters of the genetic algorithm [9][10][11][12].

In this paper, we propose to use an evolutionary algorithm called fuzzy logic controlled genetic algorithm (FLC-GA) to solve this optimization problem. Also GA (without the guide of fuzzy logic) [7] is adopted so as to compare its performance with that of FLC-GA.
The universal generating function is used to assess the availability of the studied system. The solution comprises both the availability and the cost evaluation. The universal generating technique is used to study the reliability of multi-states system [13][14]. A similar optimization problem applied on series-parallel multi-state system has been studied in reference [1] taking into account imperfect component PM actions. This model does not consider the first time of inspection, and the age of component is reduced by a factor after each imperfect maintenance action. The GA was used to solve the optimization problem.

The authors in reference [2] have developed a method to determine an optimal periodic maintenance policy in a series-parallel system but not for multi-states system. The Monte Carlo simulation was used to assess the system availability.

1.1. System description
We adopt a series-parallel multi-states system with non-identical binary independent components. The system maintenance cost is function of the inspection cost of each component. In our study, the cost of inspection for each component is the same for all the mission period. All components are immediately and perfectly repairable after failure. The failure times of each component for a fixed load occur according to an exponential distribution (constant rate).

Once the maintenance period of system component is fixed, one can evaluate the component performance distribution and its maintenance cost. Hence, the system availability, the system performance, and its total maintenance cost are evaluated. The objective function (the system cost) is known. The optimization procedure seeks for the components maintenance periods that maximize the objective function under system availability constraint. A numerical example is considered and the process is applied for different required availability values.

Section 2 describes the PM model for general series-parallel system. Section 3 formulates the optimization problem. Section 4 provides the details of genetic algorithm and fuzzy logic. In sections 5 and 6, simulations results and conclusions are addressed.

2. Preventive Maintenance Model for General Series-Parallel System

2.1. Maintenance model for basic components
We assume that the PM actions improve the reliability of basic component to as good as new, thus the component’s age is restored to zero. The problem to find the optimal vector $T_0$ is closely connected with another problem, i.e. to find the optimal first inspection time vector $T_0$: because, it makes no sense to carry out inspections in the beginning of the life of a system, when both the system and its basic components are very reliable [2]. Thus the optimal vector $T_0$ must be found for each of the basic components. The optimal vector $T_0$ must be constructed so that it takes into account both cost and reliability point of views.

2.2. General series–parallel structure
In this paper, we adopt a series-parallel system structure that is shown in Fig. 1. $K$ is the number of series sub-systems, and $E_{ik}$ is the number of parallel components in the $k^{th}$ series subsystem.

2.3. Cost model
The cost of the presented preventive maintenance policy of a series-parallel system can be calculated as a function of the inspections costs of the system components. One can write:

$$C_{PM} = \sum_{k=1}^{K} \sum_{i=1}^{E_{ik}} \sum_{j=1}^{\eta_{iik}} c_i(e(i,k))$$  \hspace{1cm} (1)

$\eta_{iik}$ is the number of total inspections of the $i^{th}$ component in the $k^{th}$ series sub-system (parallel block) during the mission time. $c_i(e(i,k))$ is the cost of the $k^{th}$ inspection of the $i^{th}$ component in the $k^{th}$ series sub-system. $K$ is the number of series sub-systems, and $E_{ik}$ is the number of parallel components in the $k^{th}$ series sub-system.
system. \( N = \sum_{k=1}^{K} \sum_{i=1}^{E_k} e(i, k) \) is the number of system components. We assume that the inspection cost of the component is constant during the mission time. Hence one can write:
\[
C_{PM} (e(i, k)) = \sum_{i=1}^{\eta_e(k)} c_i (e(i, k)) = \eta_e(k) \times c(e(i, k))
\] (2)
\( c(e(i, k)) \) is the cost inspection of the \( i \)th component in the \( k \)th series sub-system with :
\[
\eta_e(k) = 1 + \left[ \frac{T_M(e(i,k))-T_0(e(i,k))}{T_P(e(i,k))} \right]
\] (3)
\( T_M, T_0, \) and \( T_p \) are respectively the mission time, the time of the first inspection and the maintenance period of the component (the brackets notation \([x]\) is the integer value of \( x \)).

### 3. Problem formulation

Each component \( j \), is characterized by its failure rate \( \lambda_j(t) \), and PM cost of one inspection: \( c(e(i, k)) \) (\( i \) is the number of the \( j \)th component in the series subsystem \( k \)). Maintenance actions or inspections are carried out periodically for \( j \)th basic component with the period of \( T_p(j) \). The inspections are perfect, which means that the component is renewed-model as good as new is assumed. The inspection of the \( j \)th component begins at the time \( T_0(j) \), The time in which a component is not available due to PM activity is negligible, if compared to the time elapsed between consecutive activities. Components are supposed to be binary (failed or operating) but the entire system is a multi-states system.

The objective of this study is to optimize for each system component, the maintenance policy minimizing the cost function \( C_{PM} \) and respecting the availability constraint \( A(t) \geq A_0, \forall t, 0 < t \leq T_M \). Thus we have to find optimal vectors \( T_p = [T_p(1), T_p(2), ..., T_p(N)] \) and \( T_0 = [T_0(1), T_0(2), ..., T_0(N)] \) minimizing \( C_{PM} \) under given availability constraint \( A_0 \). In order to find the first inspection time \( T_0(j) \), of the component \( j \), we use the time dependent ratio-criterion of efficiency that is defined as:
\[
R_j(t) = \frac{c_j(t)}{IFB_j(t)}, j = 1, 2, ..., N
\] (4)
\( N \) is the number of system components. \( IFB_j(t) \) is the Birnbaum’s measure of importance of \( j \)th component at time \( t \) (e.g. definition in reference [15]). Details of calculating the first inspection vector can be founded in reference [7].

We suppose that the failure distribution of the component \( j \) follows the exponential law, hence its availability can be written as:
\[
A_j(t) = \exp \left( -\frac{t}{\lambda(j)} \right) = \exp \left( -\frac{t}{MTTF(j)} \right)
\] (5)

Each component \( j \) is subject to ideal preventive maintenance. The maintenance period is \( T_p(j) \). Hence the asymptotic component availability is:
\[
A_j = \exp \left( -\frac{T_p(j)}{MTTF(j)} \right)
\] (6)
The system availability can be deduced from the components availabilities using the universal generating function [7].

The problem can be formulated with the following equation:
\[
\begin{align*}
C_{PM} & \rightarrow Min \\
A(t) & \geq A_0 \\
0 & \leq T_M
\end{align*}
\] (7)
The determination of the \( T_0 \) vector is based on the Birnbaum importance factor of system components, evaluated in the context of multi-states. The component maintenance cost is function of its inspection cost, the maintenance period and the date of the first inspection. Having the above assumptions, and for each maintenance policy, the performance distribution of each component can be deduced and thus the performance distribution of the entire MSS is obtained using the universal generating function. Then the system availability, and the corresponding maintenance cost can be obtained.

### 4. Fuzzy Logic Controlled Genetic Algorithm

#### 4.1. Genetic algorithm

Several universal optimization techniques have been designed through the years. References [16][17][3] present a study about optimization techniques applied in the field of system reliability. The authors have distinguished between two classes of optimization: direct methods or exact resolutions, and indirect methods. The direct methods guarantee the optimality in a finite time, as Dynamic programming, Implicit enumeration, Separation and evaluation (branch and bound), Lexicographic research, etc. The indirect methods are used when there is no sufficient information; the global optimum is not always guaranteed, for example, Heuristics, Meta-heuristics as Genetic algorithms, Tabu search, Simulated annealing, Ant colony, etc.

Our problem is characterized by a large space of solutions. As the objective function (the quality of the solution) is the only available information, the resolution should be done through meta-heuristic methods. The
genetic algorithm is one of the most powerful meta-heuristic methods. It has been adopted to solve many reliability optimization problems [7][4][5][6][1][2][18] etc. It is inspired from the genetic biology. It is based on the principle of evolutionary search.

The solutions are represented by chromosomes in the form of strings. Any maintenance policy can be represented as an $N$-length integer string $x$ in which any element $x_j (0 \leq x_j \leq T_M)$ determines the maintenance period for system component $j$.

The adopted genetic algorithm can be described by the following steps:

1. Built an initial population of $N_s$ solutions generated randomly.
2. Evaluate each chromosome in the population.
3. Obtain new solutions by using crossover, and mutations with probability $p_m$ and $p_c$ respectively. The crossover facilitates the inheritance of some basic proprieties from the parents by the offspring, and mutation maintains a diversity of solutions. This procedure avoids convergence to a local optimum, and facilitates jumps in the solution space [19].
4. Decode the string and evaluate the solution corresponding to the obtained maintenance policy (determine the system availability and the preventive maintenance cost).
5. The objective value is used to compare different solutions. The solutions are ordered from the best to the worst. The best solutions join the population, and the other ones are discarded.
6. Repeat $N_r$ times steps 2 to 5.

### 4.2. Fuzzy logic

In order to improve the search performance of evolutionary algorithm, it was proposed to use fuzzy logic for adjusting the crossover rate and the mutation rate in ten consecutive generations as used in [10][11][12]. The average fitness of the population and the population diversity were taken as inputs to fuzzy controllers for controlling crossover and mutation rates because appropriate crossover and mutation rates can improve the search performance reflected by the average fitness values of the population and control the population diversity to prevent premature convergence from happening in the population [10]. The output of the controller is the changes of the two probabilities $p_c$ and $p_m$. Lau et al. [10] have introduced a fuzzy logic controller which aimed to set proper parameters value of the genetic algorithm, in order to solve a transportation problem. Li et al. [11] have used a similar FLC to solve the parallel machine scheduling problem.

In this work, to solve the preventive maintenance optimization problem, we have used a similar FLC than [10]. This FLC is adopted in GA to change its crossover and mutation probabilities in order to improve the search ability of the algorithm. Hence, our modified algorithm is called FLC-GA.

The implementation of FLC-GA is shown as follows. Let $f_a(t-9)$, $f_a(t)$, $f_a(t+1)$ and $f_a(t+10)$ be the average fitness values of the solutions of the population obtained at generations $(t-9), t, (t+1)$, and $(t+10)$, respectively, $d(t)$ and $d(t+10)$ be the degree of population diversity at generations $t$ and $(t+10)$, respectively, $p_c(t)$ and $p_m(t)$ be the crossover rate and mutation rate adopted at generation $t$, $\Delta p_c(t+1)$ and $\Delta p_m(t+1)$ be the change of crossover rate and mutation rate at generation $(t+1)$, respectively.

When the generation $t$ is reached, the values of $f_a(t-9)$, $f_a(t)$ and $d(t)$ are inputted to the two fuzzy controllers. $d$ is the average of bit difference of all pairs of chromosomes in the population. This can be calculated by using the following formula:

$$d = \frac{1}{N_s(N_s-1)} \sum_{i=1}^{N_s} \sum_{j=i+1}^{N_s} \sum_{k=1}^{N_s} \delta(g_{ik}, g_{jk})$$

(8)

$N_s$ is the total number of chromosomes in the population, $N$ is the chromosome length, $g_{ik}$ is the value of $k^{th}$ gene of $i^{th}$ chromosome, and $\delta(g_{ik}, g_{jk}) = 1$ if $g_{ik} = g_{jk}$, 0 otherwise.

Then, the two fuzzy controllers will determine the values of $\Delta p_c(t+1)$ and $\Delta p_m(t+1)$ as the outputs.

The fuzzy logic, which was suggested by Zadeh [20], is a powerful useful tool for performing reasoning in decision-making problems involving uncertainty and vagueness. The FLC is composed by three steps: fuzzification, decision making, and defuzzification. The details are explained below.

#### 4.2.1. Fuzzification

Before performing fuzzification, it is necessary to choose, membership functions for associating system input and output values with fuzzy input and output membership values. There is no general rule to select a membership function, and it is based on the expert knowledge and experience. Triangular membership
functions, which are commonly used, have been adopted in this study. The meaning of each linguistic term in the selected membership functions is shown in Table 1. The membership functions of the two system inputs, \( f_a(t) - f_a(t - 9) \) and \( d(t) \) for the two fuzzy controllers are shown in Fig. 2. The former is the average value of the objective function of the solutions in the population, and the latter is the sum of the hamming distance between each solution. Also the two system outputs \( \Delta p_c(t + 1) \) and \( \Delta p_m(t + 1) \) are depicted in Fig. 3.

4.2.2. Decision making
After obtaining input membership values, output membership values can be inferred from input ones through a number of IF-THEN rules. These rules are based on the expert knowledge and experience. In our study, the rule tables of [10] are adopted. They are presented in Table 2 and Table 3.

4.2.3. Defuzzification
Defuzzification aims at generating a deviation value of the crossover and mutation probabilities \( \Delta p_c(t + 1) \) and \( \Delta p_m(t + 1) \) respectively, by the membership functions shown in Fig. 3.
There exist many defuzzification methods and each one has both advantages, and disadvantages. The center of gravity (COG) defuzzification method is adopted here, as it is the best known defuzzification method [11]. This method computes the gravity center of the area under the membership function.

5. Simulation results

Our optimization problem is applied on the system adopted by [18]. It is a power plant multi-stage coal feeding system with nine conveyors. Each conveyor is characterized by a constant failure rate $\lambda_0$ and by a fixed cost of preventive maintenance action $c(e(1, k))$, $i$ is the number of the component in the series subsystem $K$. The mean time of failure of the components follows the exponential law. The mission time is $T_M = 50$ years. The structure of the system is presented in Fig. 4.

The parameters $\lambda_0$ and $c(e(1, k))$ for each component are given in Table 4.

The first inspection should not be too late, because it will be followed by a periodic preventive maintenance. We decide to choose this time between 1 and 20 years.

![Fig.4: Block Diagram of the Coal Feeding System](image)

Table 4–Parameters of the Conveyors in the Coal Feeding System

<table>
<thead>
<tr>
<th>Number of element</th>
<th>$\lambda_0 (y^{-1})$</th>
<th>$c(e(1, k))$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0692</td>
<td>6.92</td>
</tr>
<tr>
<td>2</td>
<td>0.1005</td>
<td>8.04</td>
</tr>
<tr>
<td>3</td>
<td>0.1229</td>
<td>9.83</td>
</tr>
<tr>
<td>4</td>
<td>0.0383</td>
<td>7.66</td>
</tr>
<tr>
<td>5</td>
<td>0.0383</td>
<td>7.66</td>
</tr>
<tr>
<td>6</td>
<td>0.1203</td>
<td>9.63</td>
</tr>
<tr>
<td>7</td>
<td>0.1203</td>
<td>9.63</td>
</tr>
<tr>
<td>8</td>
<td>0.0929</td>
<td>11.15</td>
</tr>
<tr>
<td>9</td>
<td>0.0929</td>
<td>11.15</td>
</tr>
</tbody>
</table>

We discretized the time and we generate twenty times of maintenance $t_m(1), i = 1, \ldots, 20$. Then, the Birnbaum importance factor $IFB_j(t_m(1))$ for each component is saved. Then we get the optimal first inspection vector $T_0 = [12, 13, 19, 9, 15, 15, 14, 14]$. After getting the optimal vector $T_0$, the optimal vector $T_p$ is determined. The corresponding preventive maintenance cost and system availability are deduced.

5.1. Using GA

The optimization problem is treated under the availability constraint $A_0 = 0.9$ in the first time by using GA. Many tests have been produced for the simulation with different values of parameters ($N_5, N_6, \rho_c, \rho_m$; population size, generation number, probability of crossover, and probability of mutation respectively). The best solution were obtained for $N_5 = 100, N_6 = 2000, \rho_c = 0.8$, and $\rho_m = 0.05$. The quasi optimal solution is obtained for $\rho_c = 0.8, \rho_m = 0.05$. The corresponding optimal vector of maintenance period is:


The quasi optimal cost is $C_{PM} = 561.64$.

Many runs of GAs with different required system availability have been done:

$A_0 = 0.9, A_0 = 0.8, A_0 = 0.7, \text{ and } A_0 = 0.6$. The effect of this factor on the optimal preventive maintenance cost is studied. Results are summarized in the Table 5.

$A_0 = 0.9$


$C_{PM} = 561.64$

$A_0 = 0.8$


$C_{PM} = 555.94$

$A_0 = 0.7$


$C_{PM} = 453.41$

$A_0 = 0.6$


$C_{PM} = 376.14$

5.2. Using FLC-GA

The overall process is repeated by using FLC-GA. In order to produce fair performance comparisons of GA and FLC-GA, in the parameters setting of the GA, we used the same values of population size, generation number, probability of crossover, and probability of
The solutions are presented in Table 6. The results show that FLC-GA outperforms the GA.

\[ A_0 = 0.9 \]
\[ T_P = \{8.0509 \ 31.8892 \ 2.9161 \ 22.9457 \ 5.4204 \ 19.3450 \ 2.7015 \ 22.6367 \ 3.4946\} \]
\[ C_{PM} = 544.47 \]

Table 5–Preventive maintenance cost for different values of \( A_0 \) using GA

<table>
<thead>
<tr>
<th>( A_0 )</th>
<th>0.9</th>
<th>0.8</th>
<th>0.7</th>
<th>0.6</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_{PM} )</td>
<td>561.64</td>
<td>555.94</td>
<td>453.41</td>
<td>376.14</td>
</tr>
</tbody>
</table>

Table 6–Preventive maintenance cost for different values of \( A_0 \) using FLC-GA

<table>
<thead>
<tr>
<th>( A_0 )</th>
<th>0.9</th>
<th>0.8</th>
<th>0.7</th>
<th>0.6</th>
</tr>
</thead>
<tbody>
<tr>
<td>( C_{PM} )</td>
<td>544.47</td>
<td>491.32</td>
<td>410.87</td>
<td>359.09</td>
</tr>
</tbody>
</table>

\[ A_0 = 0.8 \]
\[ T_P = \{4.0298 \ 4.5011 \ 41.4511 \ 5.9314 \ 5.4165 \ 7.3014 \ 46.2941 \ 5.3601 \ 4.9597\} \]
\[ C_{PM} = 491.32 \]

\[ A_0 = 0.7 \]
\[ T_P = \{4.8120 \ 4.8499 \ 38.2748 \ 5.4048 \ 21.6269 \ 7.4072 \ 5.3523 \ 42.2494 \ 5.5202\} \]
\[ C_{PM} = 410.87 \]

\[ A_0 = 0.6 \]
\[ T_P = \{3.5465 \ 19.0846 \ 19.6715 \ 14.1135 \ 12.3103 \ 3.5233 \ 25.0515 \ 10.2191 \ 12.5569\} \]
\[ C_{PM} = 359.09 \]

It can be seen that, the cost of the optimal maintenance policy decreases quickly when the required availability decreases in both cases (using GA or using FLC-GA). This is due to the fact that the number of preventive maintenance actions of components required to answer the availability constraints decreases (components can be less available). Hence, the cost of preventive maintenance of system components, and thus the entire system maintenance cost decreases. Increasing the system availability when working with high required availability need high maintenance cost. It can be seen that the optimal maintenance cost obtained by using FLC-GA is less than that obtained by using GA, and for all different values of required availability. Thus FLC-GA outperforms GA in all of these cases. The reason is that fuzzy logic can adjust the crossover rate and the mutation rate to suitable values of various evolution states of the population. The adjustable rates of crossover and mutation can balance the degree of convergence and diversity of the population no matter what the current state of the population is. This can provide a good convergence and prevent trend to local optimization. The additional time of calculation in case of FLC-GA is negligible.

6. Acknowledgements
The authors are grateful to the Lebanese CNRS for their financial support.

7. Conclusions and perspectives
This paper shows the efficiency of an optimization method to minimize the PM cost of series–parallel systems based on the time dependent Birnbaum importance factor and using universal generating function, GA, and FLC-GA. The effect of the required availability on the preventive maintenance cost is also studied. Varying required performance can allow directing the optimization more or less in the way of component loading. The performance of GA has been compared with that of FLC-GA. The results showed that the FLC-GA dominated GA in all cases of required performance. The presented method can be extended to more complex systems, viz. no exponential failure rates, complex structures different than series–parallel ones, dependent, etc.

Introduction of Markov process in the described approach is possible and it allows the study of optimization problem for a finite horizon time (at a given point of time), or even to integrate more type of dependencies between system elements. The process can increase significantly the needed computing time for solving the problem, but hybridization of the genetic algorithm with other optimization methods like local optimization can reduce this time. Also, more investigation to study the mathematical properties of the objective function and a complexity analysis should help to improve the solving procedure and compare the performance of the different optimization methods.

References


