ELSEVIER

Reliability Engineering and System Safety 67 (2000) 193–203

RELIABILITY ENGINEERING & SYSTEM SAFETY

www.elsevier.com/locate/ress

Optimization of imperfect preventive maintenance for multi-state systems

G. Levitin*, A. Lisnianski

Reliability Department, Planning, Development and Technology Division, Bait Amir, Israel Electric Corporation Ltd., P.O. Box 10, Haifa, 31000, Israel

Received 21 August 1999; accepted 2 October 1999

Abstract

The paper generalizes a preventive maintenance optimization problem to multi-state systems, which have a range of performance levels. Multi-state system reliability is defined as the ability to satisfy given demand. The reliability of system elements is characterized by their hazard functions. The possible preventive maintenance actions are characterized by their ability to affect the effective age of equipment. An algorithm is developed which obtains the sequence of maintenance actions providing system functioning with the desired level of reliability during its lifetime by minimum maintenance cost.

To evaluate multi-state system reliability, a universal generating function technique is applied. A genetic algorithm (GA) is used as an optimization technique. Basic GA procedures adapted to the given problem are presented. Examples of the determination of optimal preventive maintenance plans are demonstrated.

Keywords: Imperfect preventive maintenance; Universal generating function; Genetic algorithm

1. Introduction

The evolution of system reliability depends on its structure as well as on the evolution of the reliability of its elements. The latter is a function of element age on a system's operating life. Element aging is strongly affected by maintenance activities performed on the system. Although in some special cases surveillance or maintenance can produce an increment in the effective age of the equipment [1], in this paper we consider the maintenance actions that are characterized by their ability to reduce this age.

Preventive maintenance consists of actions, which improve the condition of system elements before they fail. PM actions such as the replacement of an element by a new one, cleaning, adjustment, etc. either return the element to its initial condition (the element becomes "as good as new") or reduce the age of the element. In some cases the PM activity (surveillance) does not affect the state of the element but ensures that the element is in operating condition. In this case the element remains "as bad as old". All actions that do not reduce to zero element age can be considered to be imperfect PM. When an element of the system fails, corrective maintenance in the form of minimal repair is performed which returns the element to operating condition without affecting its failure rate.

Optimizing the policy of preliminarily planned PM actions with minimal repair at failure for systems with increasing element failure rates is the subject of much research [2–9]. All of these works consider binary-state systems reliability.

When applied to multi-state systems, reliability is considered to be a measure of the ability of a system to meet demand (required performance level). For example, in power engineering, the ability of a system to provide an adequate supply of electrical energy [10,11] is used for evaluating its availability. In this case, the outage effect will be essentially different for units with different nominal capacity and will also depend on consumer demand. Therefore, the performance rates (productivity) of system elements should be taken into account as well as the level of demand when the entire system's reliability is estimated.

The general definition of MSS reliability according to Ref. [12] is:

$$R_{\mathrm{MSS}}(t, W) = \Pr\{G_{\mathrm{MSS}}(t) \geq W\}, \tag{1}$$

where $G_{\mathrm{MSS}}(t)$ is output performance of the MSS at time t and W is required MSS output performance (demand).

For MSS which have a finite number of states there can be K different levels of output performance at each time t: $G(t) \in \mathbf{G} = \{G_k, 1 \leq k \leq K\}$ and system OPD can be defined by two finite vectors $\mathbf{G}$ and $\mathbf{q} = \{q_k(t)\} = \Pr\{G(t) = G_k\}$ $(1 \leq k \leq K)$. Therefore MSS reliability is the probability that a system remains in those states in

* Corresponding author. Tel.: +972-4-649766; fax: +972-4-864-9780.
E-mail address: gregory@rd3.hai.iec.co.il (G. Levitin).

0951-8320/00/$ - see front matter
PII: S0951-8320(99)00067-8

Nomenclature

T	MSS lifetime
τ_j(t)	effective age of the MSS element j at chronological time t
$\tau_j^+(t)$	effective age of the MSS element j immediately after performing a PM action at chronological time t
t_{ji}	chronological time of performing ith PM on element j
ϵ	age reduction coefficient
$r_j(t)$	reliability of jth element
$h_j(t)$	hazard function of jth element
$H_j(t)$	accumulated hazard function for jth element
n_j	number of PM actions performed on MSS element j during lifetime T
N	total number of PM actions performed on MSS during T
J	total number of elements composing MSS
G_j	nominal performance rate of the jth MSS element
G_{MSS}	output performance rate of the entire MSS
W	system demand
$R_{MSS}(t,W)$	probability that $G_{MSS} \geq W$ at time t (MSS reliability)
R^*	desired MSS reliability
v_i	number (type) of ith PM action to be performed
$\mathbf{V}$	vector of PM actions (PM plan)
c_j	cost of minimal repair of element j
C_{Mj}	cost of minimal repairs of element j in interval [0,t]
$C_P(v)$	cost of PM action number v
$C_P(\mathbf{V})$	total cost of PM actions
$C_M(\mathbf{V})$	total cost of minimal repairs
$C_{tot}(\mathbf{V})$	total MSS maintenance cost
θ	time interval between possible PM actions
K	number of MSS states corresponding to different output performance levels
N_S	number of randomly constructed solutions in initial population of GA
N_{rep}	number of reproduction procedures during genetic cycle of GA
N_c	number of cycles in GA

Acronyms[1]

CM	corrective maintenance
GA	genetic algorithm
MSS	multi-state system
PM	preventive maintenance
OPD	output performance distribution
UMGF	universal moment generating function

[1] The singular and plural forms of acronyms are always spelled the same.

which $G_k \geq W$ during (0,t):

$$R_{MSS}(t,W) = \sum_{G_k \geq W} q_k(t). \tag{2}$$

A method for evaluating the reliability of series–parallel MSS consisting of elements with different performance rates was suggested in [13]. This method, based on universal generating functions, proved to be convenient for numeric implementation and effective at solving problems of MSS redundancy and maintenance optimization [14,16], as well as importance analysis [17]. The method can also be used for evaluating the influence of PM actions applied to specific elements on entire MSS reliability. Unlike fault-tree analysis, the universal generating function method provides for the possibility of treating systems with similar topologies but with different nature of elements interaction in a similar way.

In this paper we present an algorithm which determines a minimal cost plan of PM actions during MSS lifetime, which provides the required level of system reliability. The algorithm answers the questions of when, where (to which element) and what kind of available PM actions should be applied to keep the system on the required level of output performance with desired reliability during a specified time.

To solve the problem, a genetic algorithm is used. The solution encoding technique is adapted to represent replacement policies. A solution quality index comprises both reliability and cost estimations.

An illustrative example is presented in which the optimal PM plan is found for a series–parallel system.

2. Imperfect PM model

Imperfect PM is modeled using the age reduction concept [18]. According to this concept the PM action reduces the effective age of the element that it has immediately before it enters maintenance. The used proportional age setback model [19] assumes that the effective age τ_j of element j which undergoes PM actions at chronological times

$$(t_{j1},\ldots,t_{jn}) \tag{3}$$

is

$$\tau_j(t) = \tau_j^+(t_{ji}) + (t - t_{ji}) \qquad \text{for } t_{ji} < t < t_{ji+1}\ (0 \leq i \leq n), \tag{4}$$

$$\tau_j^+(t_{ji}) = \epsilon_i \tau_j(t_{ji}) = \epsilon_i[\tau_j^+(t_{ji-1}) + (t_{ji} - t_{ji-1})],$$

where $\tau_j^+(t_{ji})$ is the age of the element immediately after the ith PM, ϵ_i is the age reduction coefficient associated with the ith PM action which ranges in the interval [0,1], and $\tau_j(0) = t_{j0} = 0$ by definition.

The two extreme effects of PM on the state of the element correspond to the cases when $\epsilon = 1$, or $\epsilon = 0$. In the first case the model simply reduces to "as bad as old" which

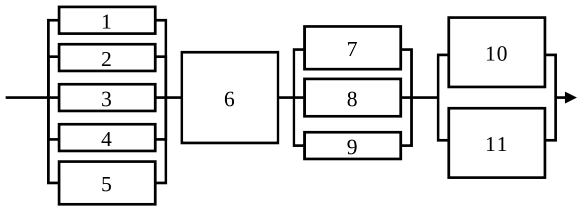

Fig. 1. Example of series–parallel MSS structure.

assumes that PM (surveillance) does not affect the age of the element. In the second case the model reduces to "as good as new", which means that the element's age is restored to zero (replacement). All the PM actions with $0 < \epsilon < 1$ lead to a partial improvement in the state of the element.

The induced hazard function of the MSS element j can be expressed as

$$h_j^*(t) = h_j(\tau_j(t)) + h_{j0},$$

where $h_j(t)$ is the hazard function of the element defined for the case when it does not undergo PM actions and h_{j0} is the term corresponding to the initial age of the element, which can differ from zero at the beginning of MSS lifetime [19].

The reliability of element j in the interval between PM actions i and $i+1$ is

$$r_j(t) = \exp\left[-\int_{\tau_j^+(t_{ji})}^{\tau_j(t)} h_j^*(x)\,\mathrm{d}x \right] = \exp[H_j(\tau_j^+(t_{ji})) - H_j(\tau_j(t))], \quad t_{ji} \le t \le t_{ji+1}, \tag{5}$$

where $H_j(\tau)$ is the accumulated hazard function for element j. One can see that immediately after PM action i when $t = t_{ji}$, the reliability of element $r_j(t_{ji}) = 1$.

Minimal repairs are performed if MSS elements fail between PM actions. The cost associated with minimal repairs depends on the failure rates of the elements. According to Ref. [20], the expected minimal repair cost of element j in interval $[0,t]$ is

$$C_{Mj} = c_j \int_0^t h_j(x)\,\mathrm{d}x.$$

For element j, which undergoes PM actions at times $t_{j1},\ldots,t_{jn_j}$, the total minimal repair cost is

$$C_{Mj} = c_j \sum_{i=0}^{n_j} \int_{\tau_j^+(t_{ji})}^{\tau_j(t_{ji+1})} h_j(x)\,\mathrm{d}x = c_j \sum_{i=0}^{n_j} [H(\tau_j(t_{ji+1})) - H(\tau_j^+(t_{ji}))], \tag{6}$$

where $t_{j0} = 0$ and $t_{jn_j+1} = T$ by definition.

3. MSS model and problem formulation

A system consisting of components connected in series is considered. Each component contains different elements connected in parallel (see, for example, Fig. 1). Each element j is characterized by its nominal performance rate G_j, hazard (failure rate) function $h_j(t)$ and associated minimal repair cost c_j. All MSS elements are elements with total failures, i.e. they have two states: functioning with nominal performance and failure. The states of MSS elements are mutually independent. The MSS is supposed to meet the demand W^*.

PM and CM actions can be realized on each MSS element. PM actions modify element reliability while CM actions (minimal repairs) do not affect it. The effectiveness of each PM action is defined by the age reduction coefficient ϵ ranging from 0 ("as good as new") to 1 ("as bad as old"). The time in which the element is not available due to a CM or PM activity is negligible if compared to the time elapsed between consecutive activities. A list of possible PM actions (PM list) is available for given MSS. In this list each PM action v is associated with the cost of its implementation $C_P(v)$, the number of element affected $e(v)$ and the age reduction coefficient $\epsilon(v)$.

System lifetime T is divided into Y intervals, each with duration θ_y $(1 \le y \le Y)$. PM actions can be performed at the end of each interval. They are performed if MSS reliability $R(t,W^*)$ becomes lower than the desired level R^*. It should be mentioned that each θ_y must not necessarily be equal and can be chosen for practical reasons. An increase in the number of intervals Y during the lifetime increases solution precision. On the other hand, the number of intervals can be limited for organizational or technical reasons.

The sequence of PM actions performed to maintain MSS reliability can be defined by a vector **V** of numbers of these actions as they appear on the PM list. Each time PM is necessary to improve system reliability, the action to be performed is defined by the next number from this vector. Note that the chosen PM action v_i may be insufficient to increase MSS reliability to the desired level. In this case the next action v_{i+1} should be performed at the same time, and so on. One can see that the total number N of PM actions cannot be predefined and depends on the composition of vector **V**.

For a given vector **V** the total number n_j and chronological times of PM actions (3) are determined for each system element j $(1 \le j \le J)$. If $n_j = 0$, which corresponds to the case in which $e(v_i) \ne j$ for all $v_i \in \mathbf{V}$, the minimal repair cost C_{Mj} given by Eq. (6) is defined by $t_{j0} = 0$ and $t_{jn_j+1} = t_{j1} = T$.

The vector **V** defines both the cost of PM actions as

$$C_P(\mathbf{V}) = \sum_{i=1}^{N} C_P(v_i)$$

and the cost of minimal repairs as

$$C_M(\mathbf{V}) = \sum_{j=1}^{J} c_j \sum_{i=0}^{n_j} [H(\tau_j(t_{ji+1})) - H(\tau_j^+(t_{ji}))].$$

The problem is therefore formulated as follows: find the optimal sequence of PM actions chosen from the list of available actions which minimizes the total maintenance cost while providing the desired MSS reliability

$$\mathbf{V} = \arg\{C_{\text{tot}}(\mathbf{V}) = C_P(\mathbf{V}) + C_M(\mathbf{V}) \to \min|R(\mathbf{V}, t, W^*) \geq R^*, 0 \leq t \leq T\}. \tag{7}$$

4. MSS reliability estimation based on a universal moment generating function

The procedure used in this paper for MSS OPD and reliability evaluation is based on the universal z-transform (or universal moment generating function) technique. A detailed description of UMGF applied to MSS reliability estimation is presented in [14]. A brief introduction to the technique is given here:

The UMGF (u-transform) of a discrete random variable X is defined as a polynomial

$$U(z) = \sum_{k=1}^{K} q_j z^{x_k}, \tag{8}$$

where the variable X has K possible values and q_k is the probability that X is equal to x_k. To obtain the probability that X is not less than some arbitrary value w, the coefficients of polynomial $u(z)$ should be summed for every term with $x_k \geq w$. This can be done using the operator δ as follows:

$$\Pr(X \geq w) = \delta(U(z), w) = \sum_{x_k \geq w} q_j(t). \tag{9}$$

In our case, the polynomial $U(z)$ can define performance distributions, i.e. it represents all the possible states of the system (or element) by relating the probabilities of each state q_k to the performance G_k of the system in that state. Note that the system OPD defined by the vectors $\{G_k, 1 \leq k \leq K\}$ and $\{q_k(t), 1 \leq k \leq K\}$ can now be represented as

$$U(t,z) = \sum_{k=1}^{K} q_k(t) Z^{G_k}. \tag{10}$$

Consider single elements with total failures. Since each element j has nominal performance G_j and reliability $r_j(t)$,

$$\Pr(X = G_j) = r_j(t),$$

$$\Pr(X = 0) = 1 - r_j(t).$$

The u-function of such an element has only two terms and can be defined at time t as

$$U_j(z) = (1 - r_j(t))z^0 + r_j(t)z^{G_j}, \tag{11}$$

which corresponds to the following vectors of element performance distribution: $\{0, G_j\}$ and $\{(1 - r_j(t)), r_j(t)\}$.

To obtain the u-function of a subsystem (component) containing a number of elements, composition operators are introduced. These operators determine the polynomial $U(z)$ for a group of elements connected in parallel and in series, respectively, using simple algebraic operations on the individual u-functions of elements. (In some cases composition operators can be developed for structures with more complex topologies, such as bridges [21].) All the composition operators take the form

$$\Omega_\omega(U_1(z), U_2(z)) = \Omega_\omega\left[\sum_{i=1}^{I} \alpha_i z^{a_i}, \sum_{j=1}^{J} \beta_j z^{b_j}\right] = \sum_{i=1}^{I}\sum_{j=1}^{J} \alpha_i \beta_j z^{\omega(a_i,b_j)} \tag{12}$$

and satisfy the following conditions:

$$\Omega_\omega\{U_1(z), \ldots, U_k(z), U_{k+1}(z), \ldots, U_n(z)\} = \Omega_\omega\{U_1(z), \ldots, U_{k+1}(z), U_k(z), \ldots, U_n(z)\}, \tag{13}$$

$$\Omega_\omega\{U_1(z), \ldots, U_k(z), U_{k+1}(z), \ldots, U_n(z)\} = \Omega_\omega\{\Omega_\omega\{U_1(z), \ldots, U_k(z)\}, \Omega_\omega\{U_{k+1}(z), \ldots, U_n(z)\}\}$$

for arbitrary k.

The function $\omega(\cdot)$ in composition operators expresses the entire performance rate of a subsystem consisting of two elements connected in parallel or in series in terms of the individual performance rates of the elements. The definition of the function $\omega(\cdot)$ strictly depends on the physical nature of system performance measure and on the nature of the interaction of system elements. In Ref. [14] two types of MSS are considered. For the sake of simplicity we consider here only those MSS in which performance measure is defined as productivity or capacity (continuous materials or energy transmission systems, manufacturing systems, power generation systems). To apply the suggested method to other types of MSS one has only to choose the corresponding functions $\omega(\cdot)$ [14,16].

In MSS of considered type the total capacity of elements connected in parallel is equal to the sum of the capacities of its elements. Therefore for a pair of elements connected in parallel

$$\omega(a, b) = \pi(a, b) = a + b. \tag{14}$$

The u-function $\tilde{U}_m$ of MSS component m containing J_m elements with their individual u-functions $U_j(z)$ defined in

Eq. (11) can be obtained as a product of polynomials:

$$\tilde{U}_m(z) = \Omega_\pi(U_1(z), \dots, U_{Jm}(z)) = \prod_{j=1}^{J_m} U_j(z)$$

$$= \prod_{j=1}^{J_m} [1 - r_j(t) + r_j(t)z^{G_j}].$$

When the components (subsystems) are connected in series, the element with the least capacity becomes the bottleneck of the system. Therefore for a pair of elements connected in series

$$\omega(a,b) = \sigma(a,b) = \min(a,b). \quad (15)$$

For MSS consisting of M components connected in series the u-function of the entire system is defined as

$$\Omega_\sigma(\tilde{U}_1, \dots, \tilde{U}_M).$$

Consequently applying composition operators one can obtain the u-function of the entire MSS in the form (10). After obtaining this u-function one actually has two vectors **G** and **q** of OPD. (The simple numerical example of using the universal z-transform for MSS OPD evaluation can be found in Ref. [17].) Using the operator δ for arbitrary W one can evaluate MSS reliability (1).

5. Optimization technique

Eq. (7) formulates a complicated combinatorial optimization problem. An exhaustive examination of all possible solutions is not realistic, considering reasonable time limitations. As in most combinatorial optimization problems, the quality of a given solution is the only information available during the search for the optimal solution. Therefore, a heuristic search algorithm is needed which uses only estimates of solution quality and which does not require derivative information to determine the next direction of the search.

The recently developed family of genetic algorithms (GAs) is based on the simple principle of evolutionary search in solution space. GAs have been proven to be effective optimization tools for a large number of applications. Successful applications of GAs to maintenance optimization problems are reported in [7–9,22–24].

It is recognized that, under general assumptions [25], the GAs theoretically achieve the global optimum of a given function. Nevertheless theoretical convergence can be useless from the practical point of view when the good solutions should be obtained in a limited time. Despite the fact that GA convergence reliability and convergence velocity are contradictory, for most practical, moderately sized combinatorial problems, the proper choice of GA parameters allows optimal solutions to be obtained in a short time.

5.1. Genetic algorithm

Basic notions of GA are originally inspired by biological genetics. GA operate with "chromosomal" representation of solutions, where crossover, mutation and selection procedures are applied. Unlike various constructive optimization algorithms that use sophisticated methods to obtain a good singular solution, the GA deals with a set of solutions (population) and tends to manipulate each solution in the simplest manner. "Chromosomal" representation requires the solution to be coded as a finite length string.

A brief introduction to genetic algorithms is presented in Ref. [26]. More detailed information on GA can be found in Goldberg's comprehensive book [27], and recent developments in GA theory and practice can be found in book [25]. The basic structure of the version of GA, referred to as GENITOR [28], is as follows:

First, the initial population of N_s randomly constructed solutions (strings) is generated. Within this population, new solutions are obtained during the genetic cycle by using crossover and mutation operators. The crossover produces a new solution (offspring) from a randomly selected pair of parent solutions, facilitating the inheritance of some basic properties from the parents by the offspring. Mutation results in slight changes to the offspring's structure and maintains a diversity of solutions. This procedure avoids premature convergence to a local optimum and facilitates jumps in the solution space.

Each new solution is decoded and its objective function (fitness) values are estimated. These values, which are a measure of quality, are used to compare different solutions.

The comparison is accomplished by a selection procedure that determines which solution is better: the newly obtained solution or the worst solution in the population. The best solution joins the population, while the other is discarded. If the population contains equivalent solutions following selection, redundancies are eliminated and the population size decreases as a result.

After new solutions are produced N_{rep} times, new randomly constructed solutions are generated to replenish the shrunken population, and a new genetic cycle begins.

The GA is terminated after N_c genetic cycles. The final population contains the best solution achieved. It also contains different near-optimal solutions, which may be of interest in the decision-making process.

5.2. Solution representation

To apply the genetic algorithm to a specific problem, a solution representation and decoding procedure must be defined. The natural representation of a vector of PM action numbers V corresponding to problem formulation (7) is by finite length integer string S containing numbers generated in the range $(1,P)$, where P is the total number of possible types of PM actions (the length of the PM list). Since the number of PM actions performed can vary from solution to

Table 1
Parameters of system elements

No. of element	Productivity	λ	γ	h_0	Minimal repair cost
1	0.4	0.050	1.8	0.0001	0.9
2	0.4	0.050	1.8	0.0001	0.9
3	0.4	0.050	1.8	0.0	0.9
4	0.4	0.070	1.2	0.0003	0.8
5	0.6	0.010	1.5	0.0	0.5
6	1.3	0.010	1.8	0.00007	2.4
7	0.6	0.020	1.8	0.0	1.3
8	0.5	0.008	2.0	0.0001	0.4
9	0.4	0.020	2.1	0.0	0.7
10	1.0	0.034	1.6	0.0	1.2
11	1.0	0.008	1.9	0.0004	1.9

solution, a redundant number of positions F should be provided in each string. In this case after decoding the solution represented by string $S = \{s_1, \ldots, s_F\}$, only the first N numbers will define the PM plan. The elements of the string from s_{N+1} to s_F do not affect the solution but can affect its offspring by participating in crossover and mutation procedures.

5.3. Crossover and mutation procedures

The crossover operator for given parent strings **P1**, **P2** and the offspring string **O** is defined as follows: first **P1** is copied to **O**, then all numbers of elements belonging to the fragment between k and m positions of the string **P2** (where k and m are random values, $1 \le k < m \le F$) are copied to the corresponding positions of **O**. The following example illustrates the crossover procedure for $F = 6$:

$$\mathbf{P1} = \boldsymbol{s_1 s_2 s_3 s_4 s_5 s_6},$$

$$\mathbf{P2} = s_1 s_2 s_3 s_4 s_5 s_6,$$

$$\mathbf{O} = \boldsymbol{s_1} s_2 s_3 s_4 \boldsymbol{s_5 s_6}.$$

The mutation procedure swaps elements initially located in two randomly chosen positions.

Table 2
Parameters of PM actions

No. of PM action	No. of element	Age reduction ϵ	PM cost C
1	1	1.00	2.2
2	1	0.56	2.9
3	1	0.00	4.1
4	2	1.00	2.2
5	2	0.56	2.9
6	2	0.00	4.1
7	3	1.00	2.2
8	3	0.56	2.9
9	3	0.00	4.1
10	4	0.76	3.7
11	4	0.00	5.5
12	5	1.00	7.3
13	5	0.60	9.0
14	5	0.00	14.2
15	6	0.56	15.3
16	6	0.00	19.0
17	7	0.75	4.3
18	7	0.00	6.5
19	8	0.80	5.0
20	8	0.00	6.2
21	9	1.00	3.0
22	9	0.65	3.8
23	9	0.00	5.4
24	10	1.00	8.5
25	10	0.70	10.5
26	10	0.00	14.0
27	11	1.00	8.5
28	11	0.56	12.0
29	11	0.00	14.0

5.4. Solution decoding procedure

The following procedure determines the fitness value for an arbitrary solution defined by integer string $S = \{s_1, \ldots, s_F\}$

1. Define for all the MSS elements ($1 \le j \le J$) effective ages $\tau_j = -\theta_1$, $H_j(\tau_j^+) = 0$. Assign 0 to chronological time t, the interval number y, the number of PM actions performed m, and total maintenance cost C_{tot}.
2. Increment the interval number y by 1. Increment chronological time t and ages τ_j of all the system elements ($1 \le j \le J$) by θ_y.
3. Calculate $H_j(\tau_j)$ for each MSS element ($1 \le j \le J$).
4. Calculate reliability $r_j(\tau_j) = \exp[H_j(\tau_j^+) - H_j(\tau_j)]$ for all system elements ($1 \le j \le J$).
5. Using the UMGF approach described in Section 4, define MSS OPD and calculate system reliability $R(t,W^*)$ for given demand W^*.

Table 3
The best PM plan obtained for $R(t,0.8) > 0.9$

t	No. of PM action	No. of element effected	$R_p(t,0.8)$
14.250	6	2	0.949
17.875	8	3	0.923
19.500	15	6	0.948
21.750	21	9	0.932
23.000	2	1	0.947

Table 4
The best PM plan obtained for $R(t,1.0) > 0.9$

t	No. of PM action	No. of element effected	$R_p(t,1.0)$
10.625	18	7	0.956
13.625	3	1	0.939
16.000	15	6	0.934
17.625	6	2	0.925
19.000	8	3	0.930
20.500	10	4	0.913
21.250	18	7	0.956
24.375	7	3	0.915

6. If $R(t, W^*) < R^*$ increment m by 1 and define the PM action to be performed at time t as $v = s_m$.
   Add the cost of PM $C_P(v)$ to C_{tot}.
   Determine the cost of minimal repairs for element $e(v)$ in the interval between previous and current PM as $c_{e(v)}[H_{e(v)}(\tau_{e(v)}) - H_{e(v)}(\tau^+_{e(v)})]$ and add this value to C_{tot}.
   Modify the age $\tau_{e(v)}$ of element $e(v)$ by multiplying it by the age reduction coefficient $\epsilon(v)$.
   Calculate the new value of $H_{e(v)}(\tau^+_{e(v)})$ for modified age $\tau_{e(v)}$.
   Recalculate the reliability of element $e(v)$ and return to step 5.
7. If $R(t,W^*) \geq R^*$ and $t < T$, return to step 2.
8. If $R(t,W^*) \geq R^*$ and $t \geq T$, evaluate the costs of minimal repairs during the last interval for all the elements ($1 \leq j \leq J$) as $c_j[H_j(\tau_j) - H_j(\tau^+_j)]$ and add these costs to C_{tot}.

Finally, the procedure determines solution fitness as $C^* - C_{tot}$ where C^* is a sufficiently large value in order to let the GA look for solutions with maximal fitness.

6. Illustrative example

Consider a series–parallel MSS consisting of four components connected in series (Fig. 1). The system contains 11 elements with different performance rates and reliability functions. The reliability of each element is defined by a Weibull intensity function

$$h(t) = \lambda^\gamma \gamma[\tau(t)]^{\gamma-1} + h_0,$$

which is widely adopted to fit repairable equipment. The accumulated hazard function takes the form

$$H(t) = [\lambda \tau(t)]^\gamma + h_0 \tau(t).$$

The element's nominal performance rate (productivity) G, intensity function scale parameter λ, shape parameter γ and hazard constant h_0 are presented in Table 1. This table also contains the cost of minimal repair for each element (in thousand US\$). One can see that elements with increasing failure rates ($\gamma > 1$) are considered.

A set of possible PM actions is defined for the given MSS. Each action is characterized by its cost, the number of the element effected and the age reduction coefficient ϵ. PM activity can include replacements ($\epsilon = 0$), surveillance ($\epsilon = 1$) and imperfect PM actions with a partial improvement effect ($0 < \epsilon < 1$).

MSS lifetime is 25 years. The times for possible PM actions are evenly spaced intervals of $\theta = 1.5$ month (0.125 year). The problem is to generate a PM plan which provides the system work during its lifetime with a performance rate not less than W^* and probability not less than R^*.

The length of the integer string representing solutions was chosen to be $F = 25$, and GA parameters $N_s = 100$, $N_{rep} = 2000$, and $N_c = 50$ were chosen.

To see the influence of parameters W^* and R^* on the optimal solution, four optimal PM plans were obtained for different compositions of these parameters: $W^* = 0.8$, $W^* = 1.0$ and $R^* = 0.90$, $R^* = 0.95$. The plans obtained are presented in Tables 3–6. Each table contains a description of the PM actions performed and includes the time from the beginning of MSS life in which the action should be performed, the number of the action (as appears in Table 2),

Table 5
The best PM plan obtained for $R(t, 0.8) > 0.95$

t	No. of PM action	No. of element effected	$R_p(t,0.8)$
11.750	8	3	0.969
13.500	18	7	0.955
14.000	6	2	0.963
15.875	3	1	0.953
16.500	16	6	0.988
21.625	9	3	0.962
23.125	21	9	0.963
24.500	1	1	0.955

Table 6
The best PM plan obtained for $R(t, 1.0) > 0.95$

t	No. of PM action	No. of element effected	$R_p(t,0.8)$
7.750	18	7	0.982
10.750	3	1	0.963
11.875	10	4	0.959
12.625	18	7	0.964
14.000	16	6	0.978
16.125	6	2	0.969
17.625	9	3	0.965
18.875	3	1	0.955
19.375	27	11	0.963
20.375	18	7	0.983
23.125	10	4	0.965
24.250	17	7	0.958
24.750	4	2	0.956

Table 7
Costs of the best PM plans obtained

W / R^*	0.8	1.0
0.9	34.824	51.301
0.95	63.669	82.625

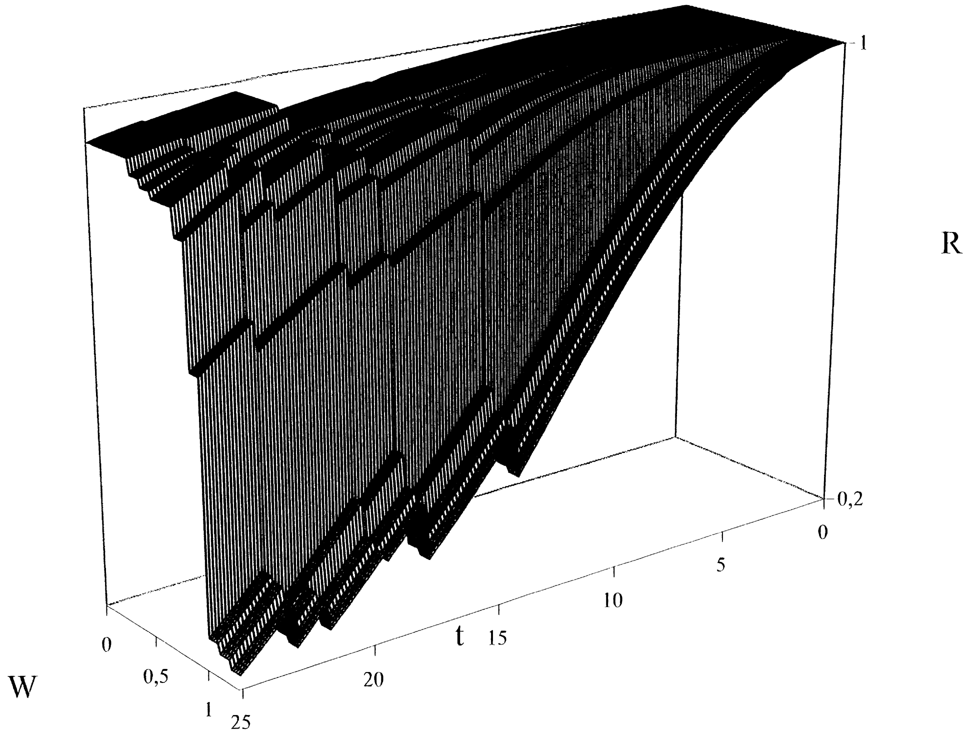

Fig. 2. MSS reliability function $R(t, W)$ for PM plan obtained for $R(t, 0.8) > 0.90$.

the number of the element affected by the action and entire MSS reliability immediately after performing the action. The total cost of the obtained plans is presented in Table 7. The MSS reliability functions $R(t,W)$ are presented in Figs. 2–5. One can see that the obtained PM plans provide high probabilities only for performance rates not exceeding desired values of W^*. The probabilities $\Pr(G_{\mathrm{MSS}} > W^*)$ decrease drastically with time. The probabilities of different states differ much more in OPD where $W^* = 0.8$ because the states corresponding to performance rates between $G_{\mathrm{MSS}} = 0.8$ and $G_{\mathrm{MSS}} = 1.3$ can have any probability, while for $W^* = 1.0$ only the

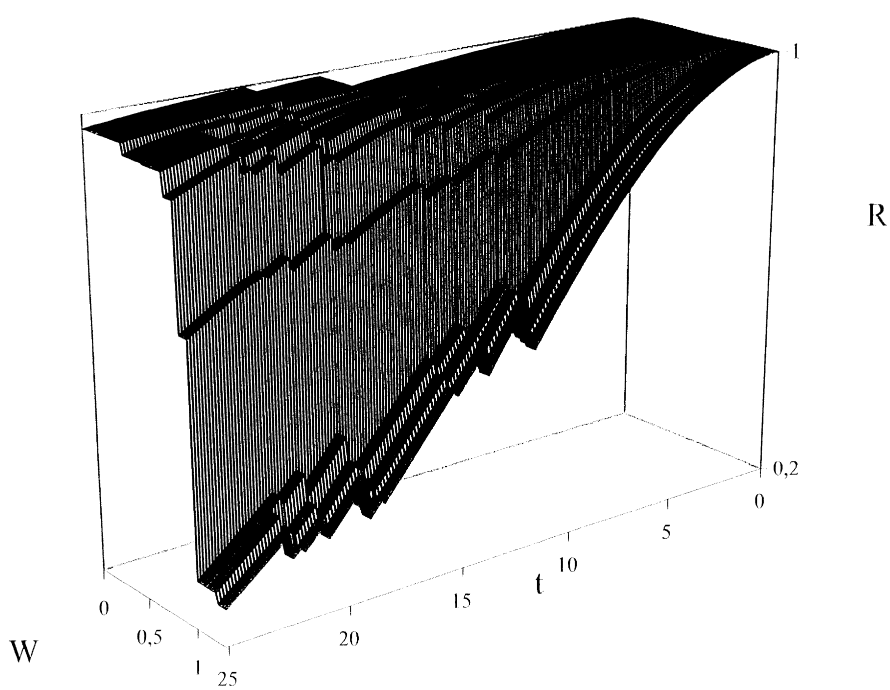

Fig. 3. MSS reliability function $R(t, W)$ for PM plan obtained for $R(t, 1.0) > 0.90$.

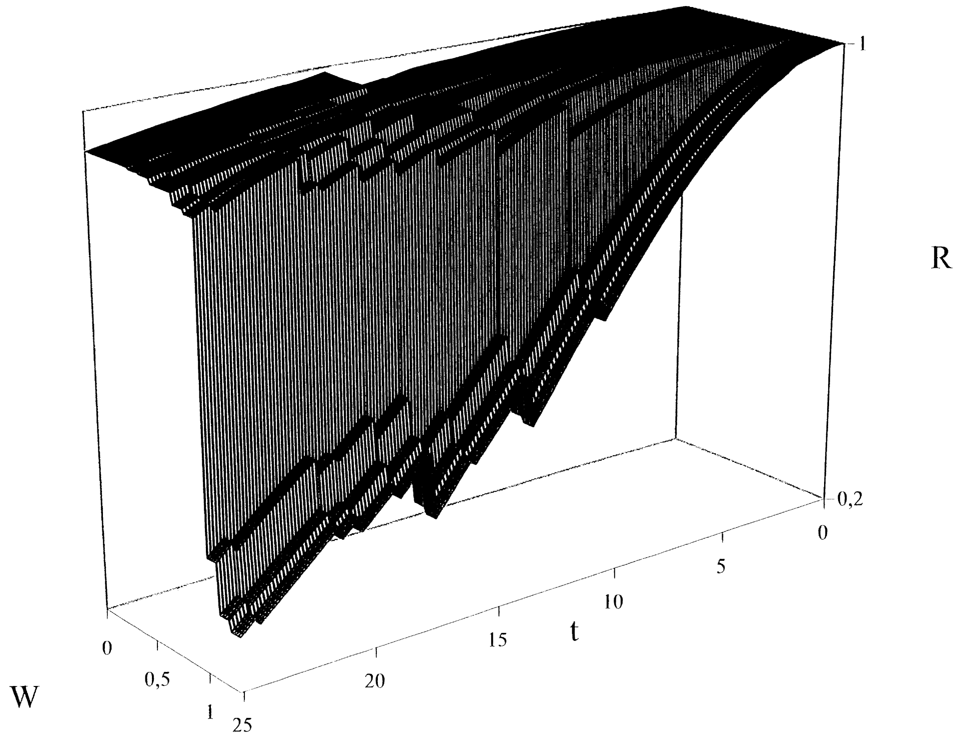

Fig. 4. MSS reliability function $R(t, W)$ for PM plan obtained for $R(t, 0.8) > 0.95$.

states corresponding to a narrower range ($1.0 < G_{MSS} < 1.3$) can have any probability. One can also see that the influence of PM actions (altitude of peaks) on the probability of different states is different. This depends on the performance rate of the element and its allocation in the MSS.

Since we are concerned with MSS reliability corresponding to fixed W^*, the section of $R(t,W)$ function with the plain

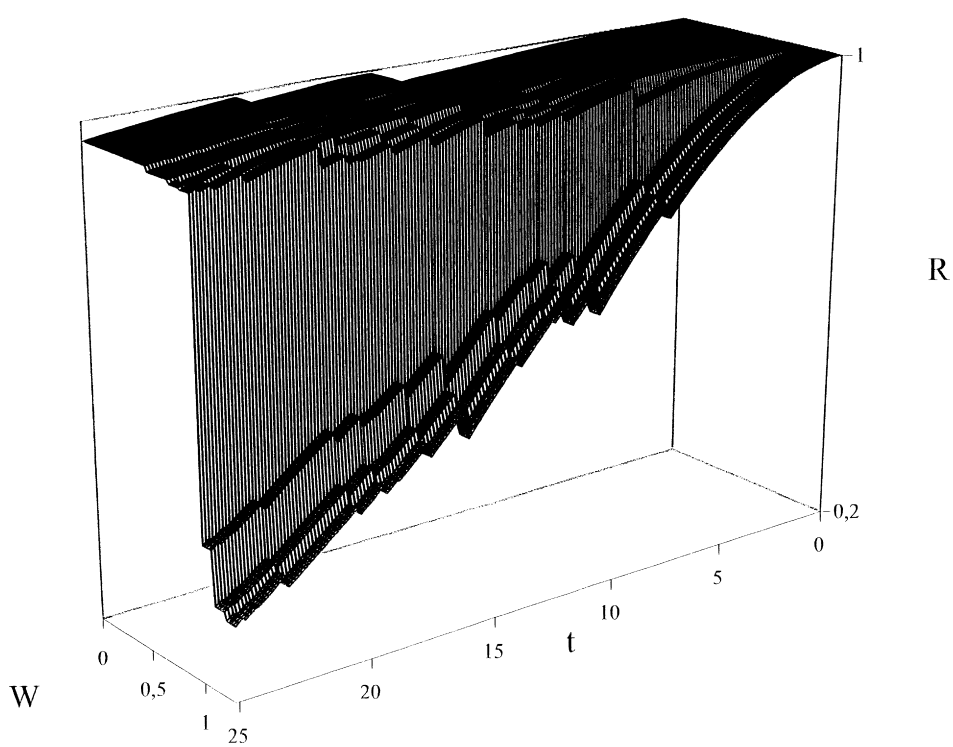

Fig. 5. MSS reliability function $R(t, W)$ for PM plan obtained for $R(t, 1.0) > 0.95$.

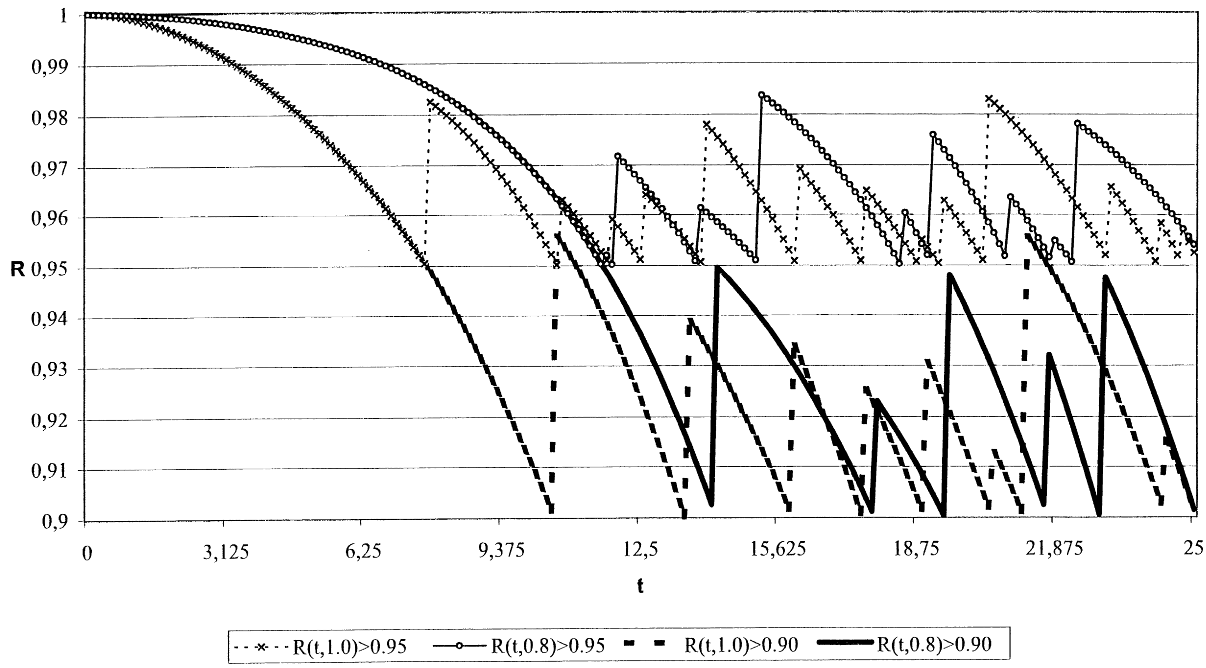

Fig. 6. MSS reliability function $R(t, W^*)$ for fixed W^* corresponding to obtained PM plans.

$W = W^*$ can illustrate the effectiveness of the PM plan. The functions $R(t,1.0)$ and $R(t,0.8)$ obtained for the optimal plans are presented in Fig. 6. One can see that the timing of PM actions is defined by the behavior of the entire MSS reliability function. Indeed the plan includes the PM actions when the reliability curve approaches the level R^*.

Running time on a Pentium II PC for the problems considered was about 3 min. Numerous tests on a set of different randomly generated problems with number of elements up to 20 and number of PM actions up to 50 show that the best times of convergence to the optimal solutions are obtained when GA parameters vary in the following ranges: $70 \le N_s \le 150$, $2000 \le N_{rep} \le 3000$. The running times for each problem did not exceed 5 min and the number of genetic cycles N_c till obtaining the optimal solution did not exceed 50.

References

[1] Kim I, Martorell S, Vesely W, Samanta P. Risk analysis of surveillance requirements including their adverse effects. Reliability Engineering & System Safety 1994;45:225–34.
[2] Ushakov I. Handbook of reliability engineering, New York: Wiley, 1994.
[3] Berg M, Epstein B. Comparison of age, block and failure replacement policies. IEEE Transactions on Reliability 1978;R-27:25–9.
[4] Deshpande J, Singh H. Optimal replacement of improving and deteriorating repairable systems. IEEE Transactions on Reliability 1992;R-44(3):427–32.
[5] Chun Y, Lee C. Optimal replacement policy for warranted system with imperfect PM operations. Microelectronics and Reliability 1992;32(6):839–43.
[6] Dagpunar S, Jack N. Preventive maintenance strategy for equipment under warranty. Microelectronics and Reliability 1994;34(6):1089–93.
[7] Monga A, Zuo M. Optimal system design considering maintenance and warranty. Computers and Operational Research 1998;25(9):691–705.
[8] Munoz A, Martorell S, Serdarell V. Genetic algorithms in optimizing surveillance and maintenance of components. Reliability Engineering & System Safety 1997;57(2):107–20.
[9] Savic D, Walters G, Knezevic J. Optimal opportunistic maintenance policy using genetic algorithms. 1: formulation. Journal of Quality in Maintenance Engineering 1995;2:34–49.
[10] Billinton R, Allan R. Reliability of power systems, London: Pitman, 1994.
[11] Lisnianski A, Levitin G, Ben-Haim H, Elmakis D. Power system structure optimization subject to reliability constraints. Electric Power Systems Research 1996;40:145–52.
[12] Xue J, Yang K. Dynamic reliability analysis of coherent multi-state systems. IEEE Transactions on Reliability 1995;44:683–8.
[13] Ushakov I. Universal generating function. Soviet Journal of Computing System Science 1986;24(5):118–29.
[14] Levitin G, Lisnianski A, Beh-Haim H, Elmakis D. Redundancy optimization for series-parallel multi-state systems. IEEE Transactions on Reliability 1998;47(2):165–72.
[15] Levitin G, Lisnianski A. Joint redundancy and maintenance optimization for multi-state series–parallel systems. Reliability Engineering and System Safety 1999;64:33–42.
[16] Levitin G, Lisnianski A. Optimal multistage modernization of power system subject to reliability and capacity requirements. Electric Power Systems Research 1999;50:183–90.
[17] Levitin G, Lisnianski A. Importance and sensitivity analysis of multi-state systems using the universal generating function method. Reliability Engineering and System Safety 1999;65(3):271–82.
[18] Nakagawa T. Sequential imperfect preventive maintenance policies. IEEE Transactions on Reliability 1988;37(3):295–8.
[19] Martorell S, Sanchez A, Serdarell V. Age-dependent reliability model considering effects of maintenance and working conditions. Reliability Engineering & System Safety 1999;64(1):19–31.

[20] Boland P. Periodic replacement when minimal repair costs vary with time. Naval Research Logistics Quarterly 1982;29(4):541–6.
[21] Levitin G, Lisnianski A. Structure optimization of power system with bridge topology. Electric Power Systems Research 1998;45:201–8.
[22] Huang S. Generator maintenance scheduling: a fuzzy system approach with genetic enhancement. Electric Power Systems Research 1997;41:233–9.
[23] Kim H, Nara K, Gen M. method for maintenance scheduling using GA combined with SA. Computers in Industrial Engineering 1994;27(1/4):477–80.
[24] Ramachandran V, Kannan J, Sathiyanarayanan K, Sivakumar V. Optimal replacement strategies—genetics algorithms approach. Microelectronics and Reliability 1997;37(4):665–7.
[25] Bäck T. Evolutionary algorithms in theory and practice. Evolution strategies. Evolutionary programming. Genetic algorithms, Oxford: Oxford University Press, 1996.
[26] Austin S. An introduction to genetic algorithms. AI Expert 1990(5):49–53.
[27] Goldberg D. Genetic algorithms in search, optimization and machine learning, Reading, MA: Addison-Wesley, 1989.
[28] Whitley D, Kauth J. GENITOR: a different genetic algorithm. Technical Report CS-88-101, Colorado State University, Fort Collins, CO, 1988.