Opportunities selective renewal policy for systems subject to propagated failures with global effect and failure isolation phenomena

Ghofrane Maaroufi a, b, Anis Chelbi a, *, Nidhal Rezg b

a Centre de Recherche en Productique (CEREP), University of Tunis, Higher School of Sciences and Techniques of Tunis (ESSTT), 5 Avenue Taha Hussein, Montfleury, 1008 Tunis, Tunisia
b Industrial Engineering and Production Laboratory of Metz (LGIPM), University of Lorraine, l’Île du Saulcy, 57045 Metz Cedex 01, France

A R T I C L E   I N F O

Article info
Received 8 July 2012
Received in revised form 18 December 2012
Accepted 31 December 2012
Available online 9 January 2013

Keywords:
Selective maintenance
Stochastic dependence
Failure isolation
Propagated failures

A B S T R A C T

This paper considers a selective maintenance policy for multi-component systems for which a minimum level of reliability is required for each mission. Such systems need to be maintained between consecutive missions. The proposed strategy aims at selecting the components to be maintained (renewed) after the completion of each mission such that a required reliability level is warranted up to the next stop with the minimum cost, taking into account the time period allotted for maintenance between missions and the possibility to extend it while paying a penalty cost. This strategy is applied to binary-state systems subject to propagated failures with global effect, and failure isolation phenomena. A set of rules to reduce the solutions space for such complex systems is developed. A numerical example is presented to illustrate the modeling approach and the use of the reduction rules. Finally, the Monte-Carlo simulation is used in combination with the selective maintenance optimization model to deal with a number of successive missions.

© 2013 Elsevier Ltd. All rights reserved.

1. Introduction

Maintenance strategies for systems made of one component, or being possibly assimilated to single-component systems, have been extensively treated in literature for several decades. However, in the last few years, an increasing interest has been focused on the development and optimization of maintenance policies for multi-component systems. This is related to the fact that in real life, industrial as well as transport systems and many other types of equipment are generally made of many components, which may have one or more type of dependence (economic, stochastic or structural). Economic dependence means that jointly maintaining some components may be cheaper than maintaining them separately. Stochastic dependence implies that failure or degradation of one component can affect the state of one or more other components of the system. Structural dependence is the fact that maintaining one component may impose the maintenance or at least the disassembly of one or more other components. Cho and Parlar [1], Van Der Duyn Schouten [2], Dekker et al. [3] and, more recently, Nicolai and Dekker [4] provide overviews of optimal maintenance policies of multi-component systems with and without dependency.

In the present work, we focus on multi-component systems for which a high reliability level is required for each mission of known duration to be accomplished. Such systems (like manufacturing equipment, aircrafts, ships, computer systems, military weapons, etc.) must be maintained between consecutive missions. The problem consists in selecting the components to be maintained after the completion of each mission, such that a required reliability level is warranted up to the next stop with the minimum cost and taking into account the limitations on maintenance time and resources before the start of the next mission. This problem has been tackled by Cassady et al. [5] in the case of series–parallel systems and for more general structures displaying redundancy with stochastically independent components. They developed a method to decide which failed components should be repaired before the next mission and which components should be left in a failed condition. They also optimized these selective maintenance decisions in situations where the objective is to maximize the system's reliability under budget and time constraints, and also in the case where maintenance time is minimized under the constraints of cost and reliability. This work has been extended by the same authors [6], considering not only renewal of failed components but also the possibility to perform minimal repairs on failed components and preventive replacement of functioning ones. They addressed the case of system reliability maximization under budget and time constraints, considering time dependent failure rates for all components whose failures are stochastically independent and whose lifetimes follow a Weibull
distribution. They also used simulation in combination with their analytical model to be able to deal with a succession of missions. Rajagopalan and Cassady [7] considered the problem of finding the number of failed components which should be replaced in order to maximize the system reliability for next mission under a maintenance time constraint. The considered systems have a series–parallel structure with constant failure rate components and stochastically independent failures. The authors improved the original total enumeration method proposed by Rice et al. [8]. The need of speeding up the solving procedure becomes of first importance when large-size problems (systems with great number of components) are addressed. For such cases, Galante and Passannanti [9] developed an exact algorithm for solving the same selective maintenance problem, allowing a drastic reduction of the solution space for series–parallel systems. Certa et al. [10] developed an extension of the work of Galante and Passannanti [9]. They proposed a more general approach considering multi-objective (cost and time) maintenance optimization based on the Pareto optimal frontier. Their approach allows obtaining the best solutions about the components to be preventively maintained and among which one can choose the most suitable ones.

More recently, Liu and Huang [11] developed an optimal selective maintenance policy for multi-state systems considering imperfect maintenance (age reduction) as one possible action to be performed on components between missions. They applied their approach to the case of a power station coal transportation system. Imperfect preventive maintenance during maintenance breaks has also been considered by Khatab et al. [12] in the context of a succession of missions of equal or different durations for binary-state series–parallel systems.

As it can be noticed through the above mentioned papers and others in the literature, it is always supposed that the maintenance actions are performed one after the other, making the time constraint expressed in terms of the sum of the replacement durations of each component. In this paper, we consider situations, often encountered in practice where the maintenance actions on different components start at the same time and they are carried out simultaneously. Moreover, most, if not all, of the works on selective maintenance consider series–parallel systems or more general system structures involving redundancy with components failures being local and stochastically independent. In this work, we model the selective maintenance concept considering, on one hand, economic dependency and on the other hand, we apply it to complex systems, with functional dependence, subject to global failure propagation and isolation effects. Propagated failures are common cause-failures originated from a component of a system causing the failure of the entire system (global effect) or the failure of some of its sub-systems (selective effect). propagated failures with global effect can be caused by imperfect fault coverage despite the presence of adequate redundancy and fault tolerant mechanism (see Amari et al. [13] and Levitin and Amari [14]). They can also simply be due to a destructive effect of failures of some components of the system.

Moreover, in practice, many systems experience what is called failure isolation. Xing and Levitin [15] define this phenomenon as follows “the failure of one component (referred to as a trigger component) can cause other components (referred to as dependent components) within the same system to become isolated from the system, which on one hand, makes the isolated dependent components unusable; and on the other hand, prevents the propagation of the failures originated from those dependent components”. I/O controllers of peripheral devices in a computer system are part of many real-world systems with isolation effect. Indeed, when the I/O controller fails, the connected peripheral devices become unusable and at the same time the computer becomes insensitive to any failure originated from those

2. The mathematical formulation of the problem

Consider a multi-component system required to perform a series of successive missions and whose components can be maintained during scheduled downtime periods between successive missions. The problem consists of selecting the components to be maintained (renewed) after the completion of any mission \( k \) and before the start of the next mission \( k+1 \), such that a required reliability level is warranted up to the next stop after \( D_{k+1} \) time units, with the minimum cost and without exceeding the time \( \Delta_k \) scheduled for maintenance between missions \( k \) and \( k+1 \). This should be done considering that it is possible to pay a penalty cost for extending this maintenance period between missions to a certain extent.

The following assumptions are made:

- At the end of a given mission period, each component (as well as the system) is either functioning or failed.
- All maintenance actions consist in components renewals. They could be preventive renewals of working components or renewals of failed ones.
- The replacement actions of the selected components start at the same time (at the beginning of the maintenance period
between missions with sufficient available maintenance resources). Hence, the required duration to complete all replacements is equal to the largest required period per selected component.

The following notations are used:

- \( f_d \) Probability density function associated with time to local failure of component \( i, i=(1,2,\ldots,n) \), \( n \) being the number of components of the system.
- \( R_d \) The reliability function associated with time to local failure of component \( i \).
- \( f_p \) Probability density function associated with time to propagated failure of component \( i \).
- \( R_p \) The reliability function associated with time to propagated failure of component \( i \).
- \( F_c \) The unreliability function associated with time to failure of component \( i \).
- \( t_i \) The time required to replace the component \( i \) between missions.
- \( D_k \) The duration of mission \( k \).
- \( \Delta_k \) The downtime period between missions \( k \) and \( k+1 \) allotted to perform maintenance actions.
- \( \beta_k \) The extension coefficient of maintenance periods \( \Delta_k(\beta_k \geq 0) \), i.e., a maintenance period \( \Delta_k \) can be extended by a maximum of \( (\Delta_k, \beta_k) \).
- \( C_p \) The penalty cost per time unit due to the extension of a maintenance period \( \Delta_i \).
- \( C_r \) The replacement cost of component \( i \).
- \( C_w \) The maintenance labor cost per time unit.
- \( C_f \) A fixed cost incurred for dismantling and reassembling the system in case at least one component is to be replaced. This cost is incurred only one time in case more than one component is replaced.
- \( C(k) \) The total maintenance cost incurred to maintain the system between mission \( k \) and mission \( k+1 \).
- \( E_i(k) \) The age of component \( i \) at the end of mission \( k \).
- \( A_i(k+1) \) The age of component \( i \) at the beginning of mission \( k+1 \).
- \( M(k) \) The maintenance decision vector made of \( n \) elements, each one is either equal to \( 1 \) (replace the corresponding element) or equal to \( 0 \) (do not replace the corresponding element). \( M(k) = (m_1(k), m_2(k), \ldots, m_n(k)) \).

The following binary variables are considered:

- \( m_i(k) \): The replacement decision of component \( i \) at the end of mission \( k \).

\[
m_i(k) = \begin{cases} 1 & \text{if component } i \text{ is replaced} \\ 0 & \text{otherwise} \end{cases}
\]

- \( Y_i(k) \): The component \( i \) state at the end of mission \( k \).

\[
Y_i(k) = \begin{cases} 1 & \text{if component } i \text{ is functioning} \\ 0 & \text{otherwise} \end{cases}
\]

\( X_i(k) \) shows the effects of performing or not performing a replacement of a component on its state and age.

\( \Delta_i(k) \) The total maintenance cost incurred to maintain the system between mission \( k \) and mission \( k+1 \). Table 1 shows the effects of performing or not performing a replacement of a component on its state and age. 'a' designates the age of the component at the end of mission \( k \).

\( \Phi(k) = \begin{cases} 1 & \text{if } \sum_{i=1}^{n} m_i(k) \geq 1 \\ 0 & \text{otherwise} \end{cases} \)

\( \Phi(k) = 1 \) corresponds to the situation of replacing at least one component. In this case, the setup cost is incurred only one time (i.e., economic dependency).

At the end of mission \( k \), the decision maker should consider both possible states (working or failed) and the age of each component. Table 1 shows the effects of performing or not performing a replacement of a component on its state and age.

\[
X_i(k+1) = \begin{cases} 1 & \text{if component } i \text{ is functioning} \\ 0 & \text{otherwise} \end{cases}
\]

\( X_i(k+1) \) is functioning at the beginning of mission \( k \) if component \( i \) is in failed state at the end of mission \( k \).

\( X_i(k+1) = \begin{cases} 1 & \text{if component } i \text{ is functioning} \\ 0 & \text{otherwise} \end{cases} \)

\( X_i(k+1) \) is functioning at the beginning of mission \( k+1 \) if component \( i \) is in failed state at the end of mission \( k \).

\[
z(k) = \begin{cases} 1 & \text{if } \Delta_k \text{ is lower than the largest required replacement period per component.} \\ 0 & \text{otherwise.} \end{cases}
\]

\[
C_f + (C_r + C_W t_i)
\]

Eq. (3) expresses the reliability constraint, \( F_i \) being the probability of failure of the system not to be exceeded for mission \( (k+1) \). Eq. (4) expresses the fact that for a given maintenance option (vector \( M(k) \)), the longest maintenance period among those of the components to be maintained should be smaller than the maintenance scheduled period \( \Delta_k \) plus its maximum extension period.

For each component \( i \) with age \( A_i(k+1) \) and state \( X_i(k+1) \) just before initiating mission \( (k+1) \), the reliability function related to local failures is expressed as follows:

\[
\hat{R}_i(D_k+1) = X_i(k+1) \times \frac{R_i(D_k+1+A_i(k+1))}{R_i(A_i(k+1))}
\]

### Table 1

<table>
<thead>
<tr>
<th>State and age of component ( i )</th>
<th>State and age of component ( i )</th>
</tr>
</thead>
<tbody>
<tr>
<td>State and age at the end of mission ( k )</td>
<td>( Y_i(k)=0 )</td>
</tr>
<tr>
<td>( E_i(k)=a )</td>
<td>( E_i(k)=a )</td>
</tr>
<tr>
<td>State and age in case replacement is performed between mission ( k ) and mission ( k+1 )</td>
<td>( X_i(k+1)=1 )</td>
</tr>
<tr>
<td>( A_i(k+1)=0 )</td>
<td>( A_i(k+1)=0 )</td>
</tr>
<tr>
<td>State and age in case replacement is not performed between mission ( k ) and mission ( k+1 )</td>
<td>( X_i(k+1)=0 )</td>
</tr>
<tr>
<td>( A_i(k+1)=a )</td>
<td>( A_i(k+1)=a )</td>
</tr>
</tbody>
</table>
For the components which may also cause propagated failures, the corresponding reliability function is given by:

$$\hat{R}_{ip}(D_{k+1}) = X_i(k+1) \times \frac{R_{ip}(D_{k+1}) + A_i(k+1)}{R_{ip}(A_i(k+1))}$$

(6)

The age $A_i(k+1)$ is obtained as follows:

$$A_i(k+1) = E_i(k) - E_i(k) \times m_i(k)$$

(7)

and the state $X_i(k+1)$ is given by:

$$X_i(k+1) = X_i(k) + m_i(k) \times (1 - Y_i(k))$$

(8)

Hence, given the state and age of each component observed at the end of mission $k$, for each possible maintenance decision vector $M(k)$ among the theoretical number of $2^n$, the decision maker can evaluate the cost $C$ and the state $H_{k+1}$ for each possible maintenance decision vector $M(k)$.

The special gate in Fig. 1, which is called a functional dependence gate (FDEP) (Dugan and Doyle [17]), is used to model the functional dependence. In this example, when the trigger component $A$ fails, the dependent components $B$ and $C$ become unusable. In such a situation, component $A$ and the corresponding FDEP gate are removed, the dependent components $B$ and $C$ are replaced with ‘1’, which means ‘TRUE’, and a Boolean reduction can be applied to the obtained fault tree.

For any given situation before the start of next mission $k+1$ regarding the state $X_i(k+1)$ and the age $A_i(k+1)$ of each component ($i=A,B,C,D$), we apply the reliability assessment method presented in Section 3 in combination with Eqs. (5) and (6) to express the probability of failure of the considered system over the mission’s duration $D_{k+1}$ as follows:

$$F_{i}(D_{k+1}) = \frac{\hat{R}_{Al}(D_{k+1})}{1}$$

$$\times \left[ 1 - \frac{\hat{R}_{ip}(D_{k+1}) \times \hat{R}_{cp}(D_{k+1})}{\hat{R}_{ip}(A_{i}(k+1))} \right] + \hat{R}_{cp}(D_{k+1}) \times \left( 1 - \frac{\hat{R}_{Al}(D_{k+1})}{\hat{R}_{Al}(A_{i}(k+1))} \right)
$$

$$\times \left( \hat{R}_{ip}(D_{k+1}) - 1 - \hat{R}_{ip}(D_{k+1}) \right) \times \left( 1 - \hat{R}_{ip}(D_{k+1}) \right) \right) \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \right] \r...
to compute this unreliability function. Details about the development of this expression are provided in the appendix. We developed a computer program in FORTRAN 90 to compute this unreliability function.

We suppose the time to local and propagated failures follow Weibull distributions with scale parameter \( \lambda \) and shape parameter \( \alpha \) as shown below. When \( \alpha = 1 \) it reduces to an exponential distribution.

\[
\lambda_{AI} = 0.01 \quad \text{and} \quad \alpha_{AI} = 1; \quad \lambda_{B1} = 0.012 \quad \text{and} \quad \alpha_{B1} = 3; \\
\lambda_{B2} = 0.02 \quad \text{and} \quad \alpha_{B2} = 3; \quad \lambda_{CP} = 0.001 \quad \text{and} \quad \alpha_{CP} = 1.
\]

The following arbitrarily chosen input data are considered Tables 2.

Let us suppose that at the end of a mission \( k \), the state and the age of each component are given by the arbitrarily chosen values represented in Table 3.

It is interesting to notice that the above states of the system’s components correspond to a functioning system in case the failure of component C has been a local failure. In case it was a propagated failure, the whole system should be in a failed state.

Let’s suppose that the decision maker must select the components to be replaced such that the probability of failure of the system during next mission \( k+1 \) should not exceed the maximum allowed threshold \( F^*_k = 0.17 \) (i.e., reliability requirement of 83%) at the minimum cost and taking also into account the maximum allowed period for maintenance.

Using the mathematical model combined with Eq. (10) each time the reliability constraint is checked for all feasible maintenance option vectors \( M(k) \) (\( 2^4 = 16 \) vectors), we obtain the optimal solution \( M(k)^* = (m_A^L = 0, m_B^L = 0, m_C^L = 1, m_D^L = 1) \) which corresponds to a preventive replacement of component D and a renewal of component C. This solution yields the minimum cost \( C(k)^* = 3130 \) with system failure probability for next mission \( F_{k+1}(k+1) = 0.082 \) (lower than 17%). As shown in Table 4, the other components remain in the same state found at the end of mission \( k \) with the same age.

Note that in order to find out if there is at least one feasible solution satisfying the unreliability constraint (Eq. 3), one has first to check if the vector \( M(k) = (1, 1, 1, 1) \) corresponding to a renewal of all components allows the satisfaction of the constraint. In case it does not, there is no need to test any other maintenance option (see Rule #2 for solution space reduction in the next section). The operations manager should find a way to shorten the next mission duration such that the reliability requirement can be met.

### Table 2

<table>
<thead>
<tr>
<th>Input parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintenance labor cost per time unit ( C_w )</td>
<td>400 $</td>
</tr>
<tr>
<td>Replacement cost for each component ( i ), ( C_i )</td>
<td>1200 $</td>
</tr>
<tr>
<td>Penalty cost per time unit, ( C_p )</td>
<td>200 ($/time unit)</td>
</tr>
<tr>
<td>The setup cost, ( C_t )</td>
<td>50 $</td>
</tr>
</tbody>
</table>

### Table 3

| Input parameters regarding the state and age of each component at the end of mission \( k \). |
|-----------------|-----------------|-----------------|-----------------|-----------------|
| \( Y(k) \) | \( Y_A(k) = 1 \) | \( Y_B(k) = 1 \) | \( Y_C(k) = 0 \) | \( Y_D(k) = 1 \) |
| \( E(k) \) | \( E_A(k) = 20 \) | \( E_B(k) = 16 \) | \( E_C(k) = 11 \) | \( E_D(k) = 16 \) |

### Table 4

Components state and age before starting mission \( k+1 \).

<table>
<thead>
<tr>
<th>( X(k+1) )</th>
<th>( X_A(k+1) = 1 )</th>
<th>( X_B(k+1) = 1 )</th>
<th>( X_C(k+1) = 1 )</th>
<th>( X_D(k+1) = 1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( A(k+1) )</td>
<td>( A_A(k+1) = 20 )</td>
<td>( A_B(k+1) = 16 )</td>
<td>( A_C(k+1) = 11 )</td>
<td>( A_D(k+1) = 16 )</td>
</tr>
</tbody>
</table>

### 5. Reduction of the solutions space

In practice, systems such as the ones considered in this paper may have a large number of components and not only four as considered in the numerical example above. The computation time for solving the problem and finding the optimal maintenance decision vector \( M(k)^* \) increases exponentially with the number of components \( n \). Indeed, theoretically \( 2^n \) possible vectors should be checked. We propose the following set of rules to reduce the solution space and consequently speed-up the optimization procedure.

For a given maintenance decision vector \( M(k) = (m_1, m_2, ..., m_n) \), let \( M_j(k) \) be the set of components \( i \) for which \( m_i = j \) (components to be replaced) and \( M_0(k) \) the set of components \( i \) for which \( m_i = 0 \) (components not to be replaced). Obviously: \( j \neq 1 \) and \( \dim(M_j(k)) + \dim(M_0(k)) = \dim(M(k)) \).

Given the theoretical solution space of \( 2^n \) maintenance decision vectors, the five following rules should be applied in order to potentially reduce this space. Illustrations related to the system of Fig. 1 considered in the numerical example of Section 4 will be given for each rule.

**Rule #1.** In case any component \( i \) with an exponentially distributed time to failure is found functioning at the end of mission \( k \), \( Y_i(k) = 1 \), all vectors implying the renewal of this component should not be checked. In fact, replacing such a constant failure rate component yields unnecessary costs.

**Rule #2.** Start checking the sequence of theoretical vectors one by one. Once a checked vector \( M(k) \) is found satisfying the maintenance duration constraint (4) and not satisfying the unreliability constraint (3), all other remaining non checked vectors for which at least one component moves from \( M_1(k) \) to \( M_0(k) \) have to be discarded. In fact, a component transition from \( M_1(k) \) to \( M_0(k) \) implies certainly a greater system unreliability. Consequently, according to this rule, the vector \( M(k) = (1, 1, 1, 1) \) must always be the first vector to be checked, because in case it does not satisfy the unreliability constraint there exists no solution to the problem and, as mentioned previously, the next mission duration should be shortened, such that the reliability requirement can be met when replacing all the components.

**Illustration of Rule #2:** For a given sequence of decision vectors, suppose that the first checked maintenance vector found not satisfying constraint (3) is \( M(k) = (1, 1, 0, 0) \) (i.e., \( M_0(k) = (CD) \) and \( M_1(k) = (AB) \)).

For a mission duration \( D(k+1) = 23 \) time units and for the following component states at the end of mission \( k \): \( Y_A(k) = 1 \), \( Y_B(k) = 1 \), \( Y_C(k) = 1 \), and \( Y_D(k) = 1 \), we obtain for this checked vector: \( F_k(k+1) = 0.217 > F^*_k = 0.17 \) and max \( (t_j) = t_4 = 0.5 < 2 \).
Hence, we are observing the conditions of Rule #2. We can simply say Table 5 shows all the vectors to be discarded in case they have not been checked yet.

**Rule #3.** In case an isolation phenomenon occurs during mission \( k \) (i.e., the trigger component, \( s \), has failed: \( Y_s(k)=0 \)) and none of the dependent components has failed before that, all the dependent components are then unusable. In such a situation, check the maintenance vector \( M(k)=(0,0,0,0) \) which corresponds to doing no maintenance on any component, if it does not satisfy the unreliability constraint, then all vectors implying the replacement of any dependent component, \( d \), while not replacing the failed trigger component should be discarded \((0,0,1,0,0,0), (0,1,0,0,0,0), (0,0,0,1,0,0), (0,1,1,0,0,0), \ldots\). This is justified by the fact that replacing a dependent component cannot increase the system reliability while the trigger component is in failed state.

**Illustration of Rule #3:** Suppose the following components states at the end of mission \( k \): \( Y_A(k)=0; Y_B(k)=1; Y_C(k)=1; Y_D(k)=0 \). Suppose also that vector \( M(k)=(0,0,0,0) \) does not meet the unreliability constraint (3). Recalling that as shown in Fig. 1, the trigger component is component \( A \), one can see that these components states do correspond to a failed trigger component and protected (unusable) dependent components \( B \) and \( C \).

Hence, according to this Rule #3, the following vectors should be discarded: \((0,1,0,0), (0,0,1,0), (0,1,1,0), (0,0,1,0), (0,0,0,1,1), (0,1,1,1)\).

**Rule #4.** While checking the sequence of theoretical vectors one by one, once a checked vector \( M(k) \) with at least one component \( i \) to be replaced is found not satisfying the maintenance duration constraint (4) because of that component replacement duration, all the remaining non checked vectors implying the replacement of component \( i \) should be discarded.

**Illustration of Rule #4:** Let the times required to replace each component be: \( t_A=0.5, t_B=2.2, t_C=0.4, \) and \( t_D=0.9 \) with the maximum allotted time for maintenance (including the possible extension) being equal to 2 time units. For a given sequence of decision vectors, suppose that the first checked maintenance vector found not satisfying constraint (4) is \( M(k)=(1,0,0,0,0) \). Since \( t_B=2.2>2 \), according to Rule #4, the vectors shown in Table 6 below may be discarded.

**Rule #5.** Consider the first vector that is found to satisfy both constraints (Eqs. (3) and (4)). All other remaining non checked vectors, for which at least one component moves from \( M_d(k) \) to \( M_t(k) \), have to be discarded. Indeed, moving a component from \( M_d(k) \) to \( M_t(k) \) implies automatically an additional cost related to the replacement of that component.

**Illustration of Rule #5:** Suppose the first maintenance vector found to satisfy both constraints (Eqs. (3) and (4)) is \( M(k)=(1,0,0,0,0) \) (i.e., \( M_d(k) \)=(B,C,D) and \( M_t(k) \)=(A)).

For a mission duration of \( D(k+1)=10 \) time units and for the following component states at the end of mission \( k \), for example: \( Y_A(k)=1, Y_B(k)=1, Y_C(k)=1, \) and \( Y_D(k)=1 \) with \( t_A=0.5, t_B=1.2, t_C=0.4, \) and \( t_D=0.9 \), we obtain \( F_A(k+1)=0.0531 < F_B(k)=0.17 \) and max \( t_i \) \( i \in M_t(k) \)=\( t_B=2.2 < 2 \).

Table 7 above shows all the vectors to be discarded because they yield additional costs.

It should be noticed that, except Rule #1 that should be systematically applied first, the rules to be used and the overall performance of their use in terms of solution space reduction, depend on the order in which the maintenance vectors will be considered. Once Rule #1 has been applied, as stated in Rule #2, one should always start by checking the maintenance vector \( M(k)=(1,1,1,1) \) first if it has not been discarded through the previous use of Rule #1. The next step is to check if we are in a configuration similar to the one related to Rule #3 (trigger components in failed state at the end of mission \( k \)).

We present below two examples of the use of a combination of some of the proposed rules for two different sequences of maintenance vectors corresponding to the same system of Fig. 1.

### 5.1. Illustrative examples

Consider the system of Fig. 1, suppose the following input data:

- \( D(k+1)=23 \) time units.
- Components states at the end of mission \( k \): \( Y_A(k)=1, Y_B(k)=0, Y_C(k)=0, \) and \( Y_D(k)=1 \).
- The required times for maintaining each component: $t_A=2.2$, $t_B=1.2$, $t_C=0.4$, and $t_D=0.9$ (in time units).
- $F_0^*=0.17$.
- The maintenance period including the possible extension: 2 time units.

Suppose that the 16 maintenance vectors are to be checked a priori in the following order: $(1,1,1,1), (1,1,1,0), (1,1,0,1), (1,1,0,0), (1,0,1,1), (1,0,1,0), (1,0,0,1), (1,0,0,0), (0,1,1,1), (0,1,1,0), (0,1,0,1), (0,1,0,0), (0,0,1,1), (0,0,1,0), (0,0,0,1), and (0,0,0,0)$.

The application of our model considering the different solution space reduction rules yields the following results:

First of all, the vectors implying the preventive replacement of component A are discarded according to Rule #1 since this component’s time to local failure distribution is exponential and it is in functioning state ($Y_A(0)=1$). The remaining maintenance vectors to be checked are: $(0,1,1,1), (0,1,1,0), (0,1,0,1), (0,1,0,0), (0,0,1,1), (0,0,1,0), (0,0,0,1),$ and $(0,0,0,0)$.

The first checked vector is $M(k)=(0,1,1,1)$. It is found satisfying both constraints: the unreliability constraint (3) and the maintenance duration constraint (4). Rule #5 is therefore applied as shown in Table 8 below which displays the whole checking sequence with the vectors that have been eliminated.

Thus, one can see that for the considered order in which the sixteen possible maintenance vectors were to be checked a priori, only four have been checked thanks to the use of Rule #1, Rule #2 and Rule #5. This corresponds to a 75% reduction of the solution space.

Now, suppose the order of the maintenance vectors to be checked a priori is different and comes as follows: $(1,1,1,1), (1,1,1,0), (1,1,0,1), (1,1,0,0), (1,0,1,1), (1,0,1,0), (1,0,0,1), (1,0,0,0), (0,1,1,1), (0,1,1,0), (0,1,0,1), (0,1,0,0), (0,0,1,1), (0,0,1,0), (0,0,0,1),$ and $(0,0,0,0)$. The sequence in bold is the part of the sequence different from the one considered above.

As in the previous example, the same vectors discarded according to Rule #1 must also be eliminated, regardless of the order of the maintenance vectors to be checked. Hence, the remaining vectors to be checked are the ones in bold.

The obtained results are shown in Table 9.

Hence, as shown in Table 9, only six vectors have been checked among the potential 16. This corresponds to 62.5% reduction of the solution space, which is different from the ratio of 75% obtained in the first example considering a different checking order of the maintenance vectors.

It is interesting to mention that existing algorithms, for instance the ones proposed by Galante and Passannanti [9] and Certa and al. [10], basically applied to series–parallel systems, could possibly be considered to solve the same problem for systems with competing failures subject to failure isolation and propagation effects, but with some necessary modifications. Indeed, these algorithms integrate, in a certain way, ideas close to the ones supporting Rules #2, #4 and #5 presented above. However, they do not include rules like Rule #3 which considers the competing failure isolation effects.

### 6. Simulation for sequential missions

So far, we considered that the age and the state of each component at the end of a given mission are supposed to be known. The developed mathematical programming model allows selecting the components to be replaced before the start of next mission. In case one must deal with a new system intended to perform a set of successive missions with known durations, we propose a combination of the mathematical model with a Monte-Carlo simulation model which assesses the system performance (components age and state) at the end of each mission.

As an illustrative example of this combined approach, let us consider once again the system of Fig. 1 with the same input data used in Section 4 except the scale parameter of the Weibull distribution associated with times to local failure of component A which is now taken as $\lambda_{AI}=0.04$. A sequence of five successive missions of durations $D(k)=(25, 25, 15, 25, 10)$ is considered with a maintenance period (including the possible extension) of 2 time units between missions and a maximum allowed failure probability $F_e=0.12$ for each mission.

Thus, starting with new components, at the end of each simulated mission yielding a set of components age and state, the mathematical programming model along with the solution space reduction rules are used to select the components to be

---

### Table 8

Example of the combined use of some of the solution space reduction rules.

<table>
<thead>
<tr>
<th>Checked maintenance decision vectors</th>
<th>Applied rule</th>
<th>Discarded vectors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rule #1</td>
<td>(1,1,1,1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1,1,1,0)</td>
</tr>
<tr>
<td>(0,1,1,1)</td>
<td>Rule #5</td>
<td>(1,1,1,1) already discarded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1,1,0,1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1,0,1,0)</td>
</tr>
<tr>
<td>(0,1,1,0)</td>
<td>Rule #5</td>
<td>(1,1,1,1) already discarded</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1,0,1,1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1,0,0,0)</td>
</tr>
<tr>
<td>(0,1,0,1)</td>
<td>Rule #2</td>
<td>(1,1,1,1) already checked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1,0,0,1)</td>
</tr>
<tr>
<td>(0,0,1,1)</td>
<td>Rule #2</td>
<td>(1,1,1,1) already checked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0,0,0,1)</td>
</tr>
<tr>
<td>(0,1,1,1)</td>
<td>Rule #2</td>
<td>(1,1,1,1) already checked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0,0,1,0)</td>
</tr>
<tr>
<td>(0,0,1,0)</td>
<td>Rule #2</td>
<td>(1,1,1,1) already checked</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0,0,0,0)</td>
</tr>
</tbody>
</table>

---

### Table 9

Example of the combined use of some of the solution space reduction rules.

<table>
<thead>
<tr>
<th>Checked maintenance decision vectors</th>
<th>Applied rule</th>
<th>Discarded vectors</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Rule #1</td>
<td>(1,1,1,1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(1,1,1,0)</td>
</tr>
<tr>
<td>(0,1,1,1)</td>
<td>Rule #2</td>
<td>(1,1,1,1) (Violation of constraint (3))</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0,0,1,1)</td>
</tr>
<tr>
<td>(0,0,1,0)</td>
<td>Rule #2</td>
<td>(1,1,1,1) (Violation of constraint (3))</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0,0,0,1)</td>
</tr>
<tr>
<td>(0,1,1,1)</td>
<td>Rule #5</td>
<td>(1,1,1,1) (Violation of constraint (3))</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0,1,0,0)</td>
</tr>
<tr>
<td>(0,0,1,1)</td>
<td>Rule #5</td>
<td>(1,1,1,1) (Violation of constraint (3))</td>
</tr>
<tr>
<td></td>
<td></td>
<td>(0,0,0,0)</td>
</tr>
</tbody>
</table>

---
replaced during the allotted inter-mission time. Fig. 2 shows the obtained results with specifically the optimal maintenance vector and the corresponding cost as well as the system unreliability for each maintenance period between missions.

It should be noted that 200,000 replications were used for the Monte-Carlo simulation to generate the components age and state at the end of each mission.

As shown in Fig. 2, a replacement of both components A and D should be performed during the first maintenance period. After the end of the second mission, only a replacement of component D must be made. Then, a renewal of the whole system must be performed between the third and the fourth missions. Finally, no replacements have to be made between the last two missions.

7. Conclusion

This paper proposed a formulation of the problem of selective maintenance for multi-component systems to help the decision maker in selecting the components to be maintained between consecutive missions such that a required reliability level is guaranteed up to the next stop with the minimum cost and taking into account limitations on the available time for maintenance. In this respect related to the time constraint, we considered situations where all the component replacements start at the same time. Also, the proposed approach takes into account economic dependence on one hand, and on the other hand, it has been applied to the case of complex multi-component systems with stochastic dependence subject to failure isolation and propagation effects. It should be noted that the mathematical model can also be applied to any kind of system structure with any kind of components failure time distribution, provided one can assess the system’s reliability function with new or aged components.

For the type of systems considered in this work, the computational time for solving the problem and finding the optimal maintenance decision vector M(k)* increases exponentially with the number of components n. In this regard, we proposed a series of rules to potentially reduce the solution space and consequently accelerate the optimization procedure.

Finally, a combination of Monte-Carlo simulation and the mathematical programming model has been used to deal with a number of successive missions starting with a new system.

This work is currently being extended in some respects including the possibility of performing imperfect preventive maintenance actions between missions. We also look at refining the use of the solution space reduction rules in order to find indications on the order according to which the possible maintenance vectors should be considered so as to get the best reduction performance. Another possible extension of this work consists in considering more complex systems with multiple functional dependence (displaying more than one trigger component) like the systems recently studied by Wang et al. [18].

Appendix A. Details on the development of Eq. (10)

We apply the approach of Xing and Levitin [15] summarized in Section 3 as follows:

Step1—Determination of the probabilities of occurrence of events E1, E2 and E3

E1: ‘The isolating/trigger element does not fail at all’

Pr(E1) = Pr (A) = 1 − \int_0^1 f_A(\tau_1)d\tau_1 \tag{A.1}

E2: ‘At least one dependent element fails globally before the failure of the isolating element’

Two possible scenarios in this case: propagated failure of component B followed by failure of the isolating component A or propagated failure of component C followed by the failure of component A. This can be expressed as follows:

Pr(E2) = Pr(Bp → A) or (Cp → A) \tag{A.2}

Applying the “inclusion–exclusion” method (Dugan and Doyle [17]), Xing and Levitin [15] show that:

Pr(E2) = Pr(Bp → A) + Pr(Cp → A) − {Pr(Bp → Cp → A) + Pr(Cp → Bp → A)} \tag{A.3}

Hence, Pr(E2) can be obtained by calculating the probability of sequential events as follows:

Pr(Bp → A) = \int_0^1 \int_{\tau_2}^1 f_A(\tau_1)d\tau_1 f_{Bp}(\tau_2)d\tau_2 \tag{A.4}
Using Eqs. (A.1), (A.10), (A.7), and (A.12), it can be expressed as follows:

\[ Pr(Cp \rightarrow A) = \int_0^t \left[ \int_{t_1}^{t_2} f_{A}(t_1) d\tau_1 \right] f_{Cp}(t_2) d\tau_2 \]  
(A.5)

\[ Pr(Bp \rightarrow Cp \rightarrow A) + Pr(Cp \rightarrow Bp \rightarrow A) = \int_0^t \int_{t_1}^{t_2} f_{Bp}(t_1) f_{cp}(t_2) f_{A}(t_3) d\tau_2 d\tau_1 + \int_0^t \int_{t_1}^{t_2} f_{cp}(t_1) f_{Bp}(t_2) f_{A}(t_3) d\tau_2 d\tau_1 \]  
(A.6)

As a result,

\[ Pr(E2) = \int_0^t \left[ \int_{t_1}^{t_2} f_{Bp}(t_1) f_{cp}(t_2) d\tau_2 \right] f_{Cp}(t_2) d\tau_2 + \int_0^t \left[ \int_{t_1}^{t_2} f_{cp}(t_1) f_{Bp}(t_2) d\tau_2 \right] f_{Cp}(t_2) d\tau_2 \]

\[- \int_0^t \int_{t_1}^{t_2} f_{Bp}(t_1) f_{cp}(t_2) f_{A}(t_3) d\tau_2 d\tau_1 \]

\[- \int_0^t \int_{t_1}^{t_2} f_{cp}(t_1) f_{Bp}(t_2) f_{A}(t_3) d\tau_2 d\tau_1 \]  
(A.7)

E3: ‘the trigger element fails before any global failure originated in the dependent components’

\[ Pr(E3) = 1 - Pr(E1) - Pr(E2) \]  
(A.8)

**Step 2—Expression of all the conditional probabilities of system failure given the occurrences of E1, E2 and E3**

E1 corresponds to the case where there is no isolation effect. Consequently, the system will fail with at least one propagated failure or without propagated failure.

Hence,

\[ Pr(\text{system fails} | E1) = 1 - P_u(t) + Q(t) \times P_a(t) \]  
(A.9)

where \( P_u(t) = Pr(\text{no propagated failure}) \) and \( Q(t) = Pr(\text{system fails} | \text{no propagated failure}) \).

Xing and Levitin (2010) [15] computed \( P_u(t) \) and \( Q(t) \) and they obtained:

\[ Pr(\text{system fails} | E1) = 1 - \left( 1 - \int_0^t f_{Bp}(t_2) d\tau_2 \right) \left( 1 - \int_0^t f_{Cp}(t_3) d\tau_3 \right) \]

\[ + \left( 1 - \int_0^t f_{Cp}(t_3) d\tau_3 \right) \left( \int_0^t f_{D}(t_4) d\tau_4 \right) \left( \int_0^t f_{Bp}(t_2) d\tau_2 \right) \]

\[ + \left( \int_0^t f_{D}(t_4) d\tau_4 \right) \left( 1 - \int_0^t f_{Bp}(t_2) d\tau_2 \right) - \int_0^t f_{B}(t_2) d\tau_2 \]  
(A.10)

As for the conditional probability of system failure given the occurrence E2, since we have a global propagated failure which precedes any isolation, the whole system will surely fail.

Hence,

\[ Pr(\text{system fails} | E2) = 1 \]  
(A.11)

Finally, when event E3 occurs, isolation effect is there and therefore components B and C are isolated and failure will depend only on component D. Indeed, the trigger component A and the corresponding FDEP gate are removed, the dependent components B and C are replaced with ‘1’ meaning ‘TRUE’, and a Boolean reduction is applied to the fault tree of Fig. 1.

Hence,

\[ Pr(\text{system fails} | E3) = Pr(D) = \int_0^t f_{D}(t_4) d\tau_4 \]  
(A.12)

**Step 3—Computation of the system unreliability given by:**

\[ F_s(t) = Pr(\text{system fails} | E1) \times Pr(E1) + Pr(\text{system fails} | E2) \times Pr(E2) + Pr(\text{system fails} | E3) \times Pr(E3) \]

\[ = Pr(\text{system fails} | E1) \times Pr(E1) + Pr(E2) + Pr(D) \times (1 - Pr(E1) - Pr(E2)) \]  
(A.13)

Using Eqs. (A.1), (A.10), (A.7), and (A.12), it can be expressed as follows:

\[ F_s(t) = \left[ 1 - \int_0^t f_{A}(t_1) d\tau_1 \right] \left[ 1 - \left( \int_0^t f_{Bp}(t_2) d\tau_2 \right) \left( \int_0^t f_{Cp}(t_3) d\tau_3 \right) \right] \]

\[ \times \left[ \left( \int_0^t f_{Bp}(t_2) d\tau_2 \right) \left( \int_0^t f_{Cp}(t_3) d\tau_3 \right) \right] \left( \int_0^t f_{D}(t_4) d\tau_4 \right) \]

\[ + \left( \int_0^t f_{Bp}(t_2) d\tau_2 \right) \left( \int_0^t f_{Cp}(t_3) d\tau_3 \right) \left( \int_0^t f_{D}(t_4) d\tau_4 \right) \]

\[ + \left( \int_0^t f_{Cp}(t_3) d\tau_3 \right) \left( \int_0^t f_{D}(t_4) d\tau_4 \right) \left( \int_0^t f_{Bp}(t_2) d\tau_2 \right) \]

\[ \times \left( \int_0^t f_{D}(t_4) d\tau_4 \right) \left( \int_0^t f_{Bp}(t_2) d\tau_2 \right) \left( \int_0^t f_{Cp}(t_3) d\tau_3 \right) \]

\[ - \int_0^t \int_{t_1}^{t_2} f_{Bp}(t_1) f_{Cp}(t_2) f_{A}(t_3) d\tau_2 d\tau_1 \]

\[ - \int_0^t \int_{t_1}^{t_2} f_{Bp}(t_1) f_{Cp}(t_2) f_{A}(t_3) d\tau_2 d\tau_1 \]  
(A.14)
Using the following well-known relationship: \( f(t) = -dR(t)/dt \), we obtain:

\[
F_s(t) = R_0(t) \left[ 1 - (1 - R_{bg}(t) \times R_{cp}(t) + (R_{ct}(t) \times (1 - R_{bg}(t)) \times (1 - R_{ct}(t)))) + (1 - R_{bg}(t) \times (1 - R_{ct}(t))) \right] \\
+ \int_0^t \int_0^{t_1} \int_0^{t_2} \left( \frac{dR_{bg}(t_1)}{dt_1} \times \frac{dR_{cp}(t_2)}{dt_2} \right) \times \left( \frac{dR_{ct}(t_3)}{dt_3} \right) dt_3 dt_2 dt_1 + ((1 - R_0(t)) \\
\times \left[ 1 - R_0(t) \left( \int_0^t \int_0^{t_1} \int_0^{t_2} \left( \frac{dR_{bg}(t_1)}{dt_1} \times \frac{dR_{cp}(t_2)}{dt_2} \right) \times \left( \frac{dR_{ct}(t_3)}{dt_3} \right) dt_3 dt_2 dt_1 + ((1 - R_0(t)) \right) \right] \\
\right]
\]

(A.15)

The expression above supposes that at instant \( t = 0 \) all components are new (in working state and with age equal zero). In the model developed in this work (particularly Eq. (3)), the period \( t \) corresponds to the next mission duration \( D(k+1) \) given the state \( X(k+1) \) and the age \( A_i(k+1) \) of each component \( i \) at the end of the maintenance period (i.e., just before the start of the next mission \( k+1 \)).

We recall Eqs. (5) and (6) below expressing the reliability function associated with time to local failure and time to propagated failure of component \( i \) having a given age \( A_i(k+1) \) and being in a given state \( X(k+1) \):

\[
\hat{R}_b(D_{k+1}) = X(k+1) \times \frac{R_0(D_{k+1} + A_i(k+1))}{R_0(A_i(k+1))} \quad (5')
\]

\[
\hat{R}_p(D_{k+1}) = X(k+1) \times \frac{R_0(D_{k+1} + A_i(k+1))}{R_0(A_i(k+1))} \quad (6')
\]

\( A_i(k+1) \) and \( X(k+1) \) are given by Eqs. (7) and (8). Hence, Eq. (10') is obtained by replacing in Eq. (A.15) the reliability terms for each component by the ones of Eqs. (5') and (6'), which take into account the age and the state of each component:

\[
F_s(D_{k+1}) = \hat{R}_d(D_{k+1}) \left[ 1 - \left( \hat{R}_b(D_{k+1}) \times \hat{R}_p(D_{k+1}) \right) + \left( \hat{R}_c(D_{k+1}) \right) \times \left( 1 - \hat{R}_b(D_{k+1}) \right) \right] \\
+ \left[ 1 - \hat{R}_0(D_{k+1}) \right] \left( \hat{R}_b(D_{k+1}) - \left( 1 - \hat{R}_0(D_{k+1}) \right) \right] \left( 1 - \hat{R}_c(D_{k+1}) \right) \right] \\
+ \left[ 1 - \hat{R}_0(D_{k+1}) \right] \left( \hat{R}_b(D_{k+1}) - \left( 1 - \hat{R}_0(D_{k+1}) \right) \right] \left( 1 - \hat{R}_c(D_{k+1}) \right) \right] \\
+ \left[ 1 - \hat{R}_0(D_{k+1}) \right] \left( \hat{R}_b(D_{k+1}) - \left( 1 - \hat{R}_0(D_{k+1}) \right) \right] \left( 1 - \hat{R}_c(D_{k+1}) \right) \right] \\
\times \int_0^{D_{k+1}} \int_0^{D_{k+1}} \int_0^{D_{k+1}} \left( \hat{R}_0(t_1 + A_i(k+1)) \hat{R}_b(t_2 + A_i(k+1)) \hat{R}_c(t_3 + A_i(k+1)) \right) dt_3 dt_2 dt_1 + \left[ 1 - \hat{R}_0(D_{k+1}) \right] \\
\times \int_0^{D_{k+1}} \int_0^{D_{k+1}} \int_0^{D_{k+1}} \left( \hat{R}_0(t_1 + A_i(k+1)) \hat{R}_b(t_2 + A_i(k+1)) \hat{R}_c(t_3 + A_i(k+1)) \right) dt_3 dt_2 dt_1 \right]
\]

(10')

References


