Optimal Scheduling of Preventive Maintenance for Safety Instrumented Systems Based on Mixed-Integer Programming

Anas Abdelkarim and Ping Zhang(✉)

Institute of Automatic Control, Faculty of Electrical and Computer Engineering, Technische Universität Kaiserslautern, 67663 Kaiserslautern, Germany
abelkari@rhrk.uni-kl.de, pzhang@eit.uni-kl.de

Abstract. Preventive maintenance is essential to guarantee the reliability of the safety instrumented systems in the process industry. The safety integrity level of the safety instrumented systems is evaluated based on the probability of failure on demand, which is significantly influenced by preventive maintenance. In this paper, we give an approach to optimize the schedule of the preventive maintenance for the safety instrumented systems, which considers not only the time instants but also the sequence of the maintenance schedule. The basic idea is to discretize the continuous-time Markov model from the viewpoint of maintenance and then reformulate the problem as a mixed-integer programming problem. Examples are given to illustrate the proposed approach.

Keywords: Safety instrumented systems · Probability of failure on demand · Preventive maintenance · Markov model · Mixed-integer programming

1 Introduction

Safety instrumented systems (SIS) consist of sensors, logic solver and final elements which are added to an industrial process to preclude the risk or extenuate consequences of an accident [1–3]. The international norm IEC 61511 provides guidelines for the design, operation and maintenance of the SIS [3,4]. The safety integrity level is an important index that describes the reliability of the SIS and is determined by the probability of failure on demand (PFD) [4,5]. Several approaches can be applied to model the reliability of SIS, for instance, reliability block diagrams, fault tree analysis, Petri nets and Markov models [6].

Preventive maintenance (PM) is one of the main measures to enhance the reliability of SIS. In many SIS, two kinds of maintenance tests are available, namely, main tests and partial tests [5]. The scheduling of PM tests requires to specify the time instants and the sequence of the tests over a specified period. Different PM plans impact the PFD and thus the reliability of SIS.

M. Zeller and K. Höfig (Eds.): IMBSA 2020, LNCS 12297, pp. 83–96, 2020.
https://doi.org/10.1007/978-3-030-58920-2_6

Machleidt has modelled the SIS as a stochastic and deterministic automaton. Then, an optimization problem is formulated to minimize PFD by finding optimal time instants for a predefined sequence of PM tests and solved using a heuristic nonlinear optimization method (Nelder-Mead method) [7]. However, the method may converge to non-stationary points [8]. Martynova and Zhang [9] have established for the first time the connection between the optimal scheduling problem of SIS with the optimal control problem of switched systems with state jumps. They apply the gradient method proposed in [10] to solve the optimization problem, which converges to a minimal value. Yet the approach in [9] can only find an optimal switching time instants for a predefined sequence. In this paper, we aim to find not only the optimal time instants but also the optimal sequence of the preventive maintenance.

Switched systems are a class of hybrid systems that have been much investigated in the control community in recent years [11–13]. A detailed survey on optimal control of switched systems are presented in [14]. For switched systems with state jumps, several approaches are available to find the optimal switching time instants for a predefined switching sequence. In [11], the switching time instants are determined implicitly by numerical calculation to compute a region on state spate such that the switching instants should occur only if the state belongs to this region. Gradient method [10] and Ant Colony Optimization algorithm method [15] can find the explicit optimal switching time instants.

On the other hand, there are few practical algorithms to find both optimal switching time instants and sequence. The master-slave procedure proposed by [16] iterates between two stages. In the first stage (i.e., slave procedure), the optimal switching time instants are determined under the assumption of a known switching sequence. In the second stage (i.e., master procedure), an optimal switching sequence is determined under the assumption of known switching time instants. The solution obtained by this approach depends strongly on the initial values and may sometimes achieve only a suboptimal solution.

The switching table procedure [16,17] is another approach that is based on the construction of switching tables which specify when the switching shall occur and what the next subsystem shall be. The method is not applicable for our optimization problem, because the tables are basically partitioned regions of the state space ($\mathbb{R}^n$) for the subsystems. But in our case, there is only one subsystem, as between two neighbouring state jumps the system state evolves according to the same state equation. Therefore, the partition of the state space can not be carried out making the method not applicable.

The approach presented in [18] has been extended by Kirches et al. to consider states jumps [19], where their framework involves finding both the optimal switching time instants and the sequence for non-linear, non-autonomous switched dynamic systems. The basic idea is to combine all subsystems into one equation by introducing binary variables in order to produce a mixed-integer optimal control problem. Then the differential non-nonlinear dynamics are approximated using adaptive collocation method (see section five in [19] for more details). The resulted optimization problem is then solved by a nonlinear optimization solver, e.g., IPOPT.

The state jumps in [19] are introduced as jumps occur due to switching from subsystem dynamics to another one. However, in our optimization problem, the state jumps occur due to external events on the system that undergoes only one dynamics, which means we can not directly formulate the problem as a mixed-integer optimization problem as in [19]. Therefore, we propose to integrate the state jumps directly into the system difference equations as explained in Sect. 3. Then, we can treat the problem as a mixed-integer problem as in [18] and [19]. In addition, because the system dynamics are linear, the discretization is done by solving the differential equations instead of using adaptive collocation method.

The paper is organized as follows. In Sect. 2, the Markov model of SIS is introduced and the optimization problem is formulated. In Sect. 3, the optimization problem is reformulated as a mixed-integer optimal problem and then solved. Numerical examples in Sect. 4 are provided to illustrate the proposed approach.

2 Preliminary

In this section, we briefly review the multi-phase continuous Markov model of SIS, the definition of PFD and formulate the optimization problem.

2.1 Modelling of the SIS and PM Tests

For the sake of clarity, we consider SIS with only one channel of the 1oo1-architecture. As shown by Machleidt [7], the failures of SIS can be divided into safe failures (SF) and dangerous failures. Depending on the detectability, the dangerous failures can be further classified into non-detectable failures (DN), failures detectable by a diagnosis system (DD), failures detectable by both main test or partial test (DUAB) and failures detectable only by main test (DUA). The state transition diagram of the SIS is shown in Fig. 1. The OK state is the state that the SIS is entirely functional and capable to give the safety function. The SF state represents the safe failures, which may cause an unnecessary shutdown of the plant and economic loss but have no influence on the PFD. The DR state is the state in which a dangerous failure is detected and the functionality of the SIS is not yet restored. The SIS is prevented from providing the safety function, when it is in the DR, DUA, DUAB or DN state.

The SIS can be modelled by [7,9]

$$\begin{aligned}\frac{dp(t)}{dt} &= Qp(t),\\ &= \begin{bmatrix} q_{11} & \mu_{cms} & \mu_{md} & 0\,0\,0\\ \lambda_s & -\mu_{cms} & 0 & 0\,0\,0\\ \lambda_{dd} & 0 & -\mu_{md} & 0\,0\,0\\ \lambda_{duab} & 0 & 0 & 0\,0\,0\\ \lambda_{dua} & 0 & 0 & 0\,0\,0\\ \lambda_{dn} & 0 & 0 & 0\,0\,0 \end{bmatrix} \begin{bmatrix} p_1(t)\\ p_2(t)\\ p_3(t)\\ p_4(t)\\ p_5(t)\\ p_6(t) \end{bmatrix},\\ q_{11} &= -\lambda_s - \lambda_{dd} - \lambda_{duab} - \lambda_{dua} - \lambda_{dn}, \end{aligned} \tag{1}$$

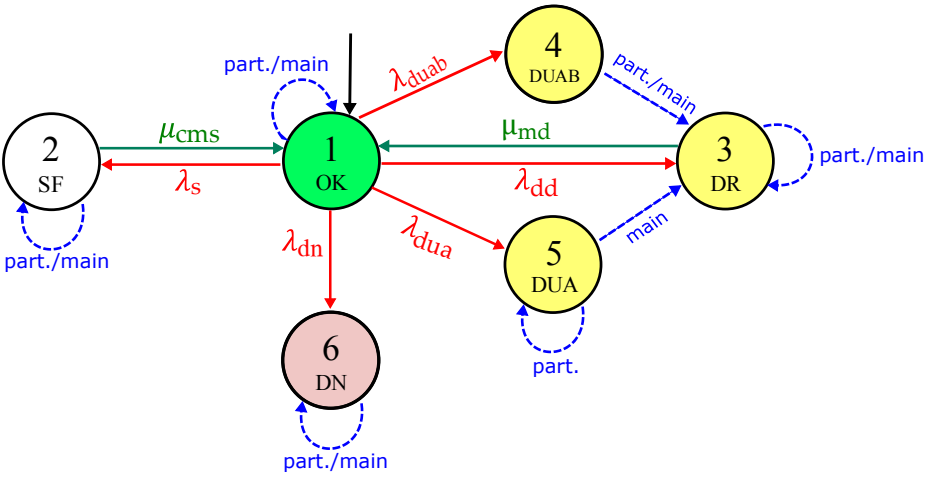

Fig. 1. State transition diagram of a 1oo1 SIS [9]

where $p_1, \cdots, p_6$ denote, respectively, the probability that the SIS is in the OK state, SF, DD, DUAB, DUA and DN, Q is constant matrix, λ_s is the failure rate of safe failures, λ_{dd}, λ_{duab}, λ_{dua}, λ_{dn} are, respectively, DD failure rate, DUAB failure rate, DUA failure rate and DN failure rate, μ_{cms} is the repair rate when a safe failure is recognized and μ_{md} is the repair rate when a DD failure is recognized.

The PM tests can be divided into two types based on the depth of the test: main tests and partial tests. For instance, a complete functional test is the main test. In comparison, a partial stroke test is a partial test. The PM test causes a jump in the state vector in (1) immediately after applying the test. The jump can be described by [9]

$$p(t_i^+) = Mp(t_i^-), \qquad i = 1, 2, \cdots, n, \tag{2}$$

where t_i^- and t_i^+ represent, respectively, the time instants immediately before and after i-th maintenance test, n is the total number of the PM tests over a finite time period and M is a matrix that describes the effect of PM test,

$$M = \begin{cases} M_A, \text{ in case of main test} \\ M_B, \text{ in case of partial test} \end{cases} \tag{3}$$

where

$$M_A = \begin{bmatrix} 1&0&0&0&0&0 \\ 0&1&0&0&0&0 \\ 0&0&1&1&1&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&0&1 \end{bmatrix}, \quad M_B = \begin{bmatrix} 1&0&0&0&0&0 \\ 0&1&0&0&0&0 \\ 0&0&1&1&0&0 \\ 0&0&0&0&0&0 \\ 0&0&0&0&1&0 \\ 0&0&0&0&0&1 \end{bmatrix}. \tag{4}$$

2.2 The Probability of Failure on Demand (PFD)

The PFD value refers to the probability that the SIS has a dangerous failure and can not respond further in case that an accident happens. It is calculated by

$$PFD = p_3(t) + p_4(t) + p_5(t) + p_6(t). \tag{5}$$

The average value of the PFD over the life cycle can be calculated by

$$PFD_{avg} = \frac{1}{t_f - t_0} \int_{t_0}^{t_f} PFD(t)dt, \tag{6}$$

where t_0 and t_f are, respectively, the initial time and the end time of the life cycle.

2.3 The Optimization Problem Formulation

According to the international norm IEC 61511, the safety integrity level of the SIS is evaluated by PFD_{avg} in (6). The smaller PFD_{avg} is, the more reliable is the SIS. The main purpose is to schedule PM tests over the life cycle to minimize the value of PFD_{avg} by selecting both the optimal sequence of the tests (i.e, the order of main and partial tests) and the optimal time instants to apply the corresponding PM tests.

Let us assume there are n_f number of main tests and n_p number of partial tests to be applied over the life cycle. The optimization problem of minimization of PFD_{avg} can be formulated as

$$\min_{t_{Aj}, t_{Bz}} \quad PFD_{avg} \tag{7a}$$

$$\text{s.t.} \quad \frac{dp(t)}{dt} = Qp(t), \qquad t \in [t_0, t_f] \tag{7b}$$

$$p(t_{Aj}^+) = M_A p(t_{Aj}^-), \quad t_{Aj} \in [t_0, t_f],\ j = 1, \cdots, n_f \tag{7c}$$

$$p(t_{Bz}^+) = M_B p(t_{Bz}^-), \quad t_{Bz} \in [t_0, t_f],\ z = 1, \cdots, n_p \tag{7d}$$

where t_{Aj} and t_{Bz} are the time instants that the full and the partial tests occur. Note that the switching sequence is not explicitly formulated as decision variables in (7a)–(7d), rather embedded in the time instants. For example, if we consider one main test and one partial test to be applied and if the solution of the optimization problem gives t_{A1} value higher than t_{B1}, we can conclude that the order is {partial test, main test}.

3 The Solution

In this section, we describe the basic idea of the proposed approach.

3.1 Discretization of the Optimization Problem

In the industrial practice, the PM tests are usually planned in hours or days. Therefore, we assume that the sampling time (h) is 1 h. In terms of the cost function (7a), it is a definite integral and it, therefore, can be approximated using a numerical method such as rectangle, trapezoidal or Simpson's method. For purposes of clarification, we have considered the rectangular method for discretization. As a result, the cost function can be described as

$$\begin{aligned} J &= PFD_{avg} \\ &\approx \frac{1}{N}\sum_{k=0}^{N-1}(p_3(k)+p_4(k)+p_5(k)+p_6(k)), \end{aligned} \tag{8}$$

where $N = \lceil t_f/h \rceil$ represents the total number of the samples in the horizon with $\lceil\ \rceil$ being the ceiling function. Observe that a smaller sampling time h leads to a larger N.
On the other hand, the differential equation in (7b) is linear; hence, it can be discretized without any approximation as

$$p(k+1) = Q_d p(k), \tag{9}$$

where $p(k) = p(kh)$, $Q_d = e^{Qh}$, h is the sampling time and k only holds for time steps that does not belongs to switching periods. By assuming that the PM tests can take place only as multiples of the sampling time, when a PM test is applied at a time k_i, the state vector $p(k_i)$ will jump to $Mp(k_i)$. By substituting the updated state vector $Mp(k_i)$ in (9), the states at next time step, $p(k_i+1)$, after doing the PM test is described by

$$p(k_i+1) = Q_d(Mp(k_i)), \qquad i = 1, 2, \cdots, n, \tag{10}$$

where M is either M_A or M_B depending on the type of PM test as shown in (3) and $n = n_f + n_p$.

Based on (9) and (10), there are three modes during the entire dynamic, namely, in normal mode without maintenance described by (9), main test mode or partial test modes described by (10). By introducing binary variables, it is possible to combine all modes in one difference equation. As a result, the dynamics (7b)–(7d) can be equivalently described by

$$\begin{aligned} p(k+1) = \gamma(k)(Q_d p(k)) + \alpha(k)(Q_d(M_A p(k)) + \\ \beta(k)(Q_d(M_\mathrm{B} p(k)) \end{aligned} \tag{11}$$

where $k = 0, 1, \cdots, N-1$ and $\alpha(k)$, $\beta(k)$ and $\gamma(k)$ are binary variables that take value either 0 or 1. The binary variables are indicators of the active mode at time instant k. Therefore, at any time k, only one binary variable has the value of one and the other two binary variables are zero.

Based on (8)–(11), the optimization problem (7a)–(7d) can be reformulated as

$$\min_{\alpha(k),\beta(k),\gamma(k)} \frac{1}{N}\sum_{k=0}^{N-1}(p_3(k)+p_4(k)+p_5(k)+p_6(k)) \tag{12a}$$

$$\text{s.t.} \quad p(k+1)=\gamma(k)Q_d p(k)+ \tag{12b}$$
$$\alpha(k)Q_d M_A p(k)+\beta(k)Q_d M_B p(k),$$
$$\sum_{k=0}^{N-1}\alpha(k)\le n_p, \qquad \alpha(k)\in\{0,1\} \tag{12c}$$
$$\sum_{k=0}^{N-1}\beta(k)\le n_f, \qquad \beta(k)\in\{0,1\} \tag{12d}$$
$$\alpha(k)+\beta(k)+\gamma(k)=1, \qquad k=0,1,\cdots,N-1 \tag{12e}$$

where α, β and γ are binary sequences. The number of the partial tests and main tests are, respectively, specified in the relaxed constraints (12c)–(12d). The constraint in (12e) guarantees that only one mode is active at any time k. This optimization problem is a mixed-integer programming due to the fact that the decision variables involve both binary and continuous variables.

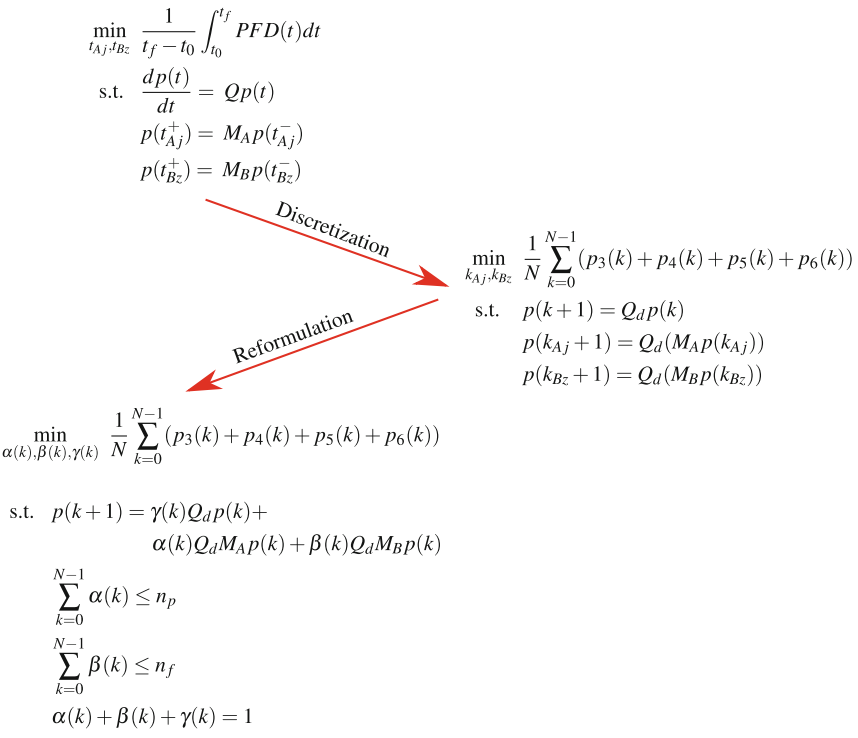

Fig. 2. Illustration of the proposed approach

Solving the optimization problem (12a)–(12e) will give an optimal solution $\alpha^*(k)$, $\beta^*(k)$ and $\gamma^*(k)$, $k = 0, 1, \cdots, N-1$. The schedule of the PM tests can be extracted from $\alpha^*(k)$ and $\beta^*(k)$. To sum up, the proposed approach involves two steps: discretization of the continuous model and then using the binary variables to reformulate the problem as a mixed-integer programming problem as illustrated in Fig. 2.

3.2 Numerical Solution to the Optimization Problem

After the reformulation of optimization problem (12a)–(12e), a solver can be applied to get an optimal solution. Due to the variety of mixed-integer solvers based on different algorithms, higher-level algebraic modelling languages (i.e. AMPL, GAMS, MPS and YALMIP) are used as interface environments between the optimization problem and the solvers, which provide flexibility in using different solvers with the same code. The AMPL is chosen as environment to describe our optimization problem due to its features, for instance, its syntax is similar to the mathematical notation of optimization problems and it is compatible with most solvers supported by the free NEOS Server [20–22]. Furthermore, the AMPL software can detect the proper solver for the defined problem. It has chosen MINOS (Modular In-core Nonlinear Optimization System) solver for our optimization problem (12a)–(12e). For the final implementation, we have used the solvers available on the NEOS server that accept the AMPL as input interface to solve the problem. Several non-linear mixed-integer solvers on the NEOS server such as BARON, scip, filterMPEC, FilMINT, Couenne, Bonmin, MINLP and filter are able to solve the optimization problem for relatively small horizon (e.g., with 14 days as sampling time over the life cycle of 10 years). In comparison, the MINOS solver is able to solve the problem for a larger horizon with more PM tests to be planned. The MINOS solver uses the simplex method for linear programming and the reduced-gradient method combined with a quasi-Newton method for non-linear optimization problem [23].

4 Numerical Examples

In this section, numerical examples are provided to illustrate the proposed approach.

4.1 Description of SIS

We consider the SIS that has been used in [7] and [9]. The SIS is described by (1)–(2) with parameters given by

$$\begin{aligned} dc &= \frac{\lambda_{dd}}{\lambda_d} \\ \lambda_d &= \lambda_{dd} + \lambda_{duab} + \lambda_{dua} + \lambda_{dn} \\ \lambda_{dd} &= dc \cdot \lambda_d \\ \lambda_{duab} &= tcb \cdot (1 - dc) \cdot \lambda_d \\ \lambda_{dua} &= (tca - tcb) \cdot (1 - dc) \cdot \lambda_d \\ \lambda_{dn} &= (1 - tca) \cdot (1 - dc) \cdot \lambda_d \end{aligned} \tag{13}$$

where dc, tca, tcb denote, respectively, the diagnostic coverage factor, test coverage of main test and test coverage of partial test. The concrete values in (13) are shown in Table 1, where h^{-1} refers to per hour.

Table 1. Parameters of the SIS

Parameter	Value	Parameter	Value
$\lambda_s\ (h^{-1})$	3×10^{-7}	$\lambda_d\ (h^{-1})$	3×10^{-8}
$\mu_{cms}\ (h^{-1})$	5×10^{-3}	$\mu_{md}\ (h^{-1})$	1×10^{-3}
dc	0.40	tca	0.80
tcb	0.48		

Therefore, the matrix Q in (1) is

$$\mathbf{Q} = \begin{bmatrix} -3.30 \times 10^{-7} & 5 \times 10^{-3} & 1 \times 10^{-3} & 0 & 0 & 0 \\ 3.00 \times 10^{-7} & -5 \times 10^{-3} & 0 & 0 & 0 & 0 \\ 1.20 \times 10^{-8} & 0 & -1 \times 10^{-3} & 0 & 0 & 0 \\ 8.64 \times 10^{-9} & 0 & 0 & 0 & 0 & 0 \\ 5.76 \times 10^{-9} & 0 & 0 & 0 & 0 & 0 \\ 3.60 \times 10^{-9} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{14}$$

Under sampling time $h = 1\,\text{h}$, the matrix Q_d in (9) is

$$\mathbf{Q}_d = \begin{bmatrix} 1 & 5.000 \times 10^{-3} & 9.995 \times 10^{-4} & 0 & 0 & 0 \\ 2.993 \times 10^{-7} & 9.950 \times 10^{-1} & 1.497 \times 10^{-10} & 0 & 0 & 0 \\ 1.199 \times 10^{-8} & 2.994 \times 10^{-11} & 9.990 \times 10^{-1} & 0 & 0 & 0 \\ 8.640 \times 10^{-9} & 2.156 \times 10^{-11} & 4.319 \times 10^{-12} & 1 & 0 & 0 \\ 5.760 \times 10^{-9} & 1.438 \times 10^{-11} & 2.879 \times 10^{-12} & 0 & 1 & 0 \\ 3.600 \times 10^{-9} & 8.985 \times 10^{-12} & 1.799 \times 10^{-12} & 0 & 0 & 1 \end{bmatrix}. \tag{15}$$

At the very beginning (i.e., $t = t_0$), the SIS is in the failure free state (i.e., OK state). Hence,

$$p(0) = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0]^T.$$

4.2 PM Schedule for 1 Main Test and 2 Partial Tests

In this example, we considered scheduling of 3 PM tests for the SIS described in Sect. 4.1, over the life cycle of 10 years, which includes 1 main test and 2 partial tests. Hence, the parameters of the optimization problem in (12a)–(12e) are $n_f = 1$, $n_p = 2$ and $N = 87,600$. The mixed-integer optimization problem is solved using the MINOS on the free NEOS server and the results are shown in the Table 2.

By checking the values of α^* and β^* over the horizon, we find the main test presented by α^* occurs at $k = 43,301$. The partial tests resented by β^* occur at $k = 21,652$ and 64,951. Map k values to continuous time by

$$t = t_0 + \frac{k}{N}(t_f - t_0). \tag{16}$$

After mapping k to years, the main test occurs at 4.943 year and the partial tests occur at 2.472 year and 7.414 year. Hence, optimal switching time instants and sequence, and the corresponding PFD_{avg} value are

$$\begin{aligned} \sigma^* &= \{\sigma_1^*, \sigma_2^*, \sigma_3^*\} \\ &= \{(\text{partial, } 2.472), (\text{main, } 4.943), (\text{partial, } 7.414)\} \\ PFD_{avg} &= 3.996 \times 10^{-4}. \end{aligned} \tag{17}$$

Table 2. The optimal solution $\alpha^*(k)$, $\beta^*(k)$ and $\gamma^*(k)$ to the optimization problem (12a)–(12e) obtained by the MINOS solver

k	$\alpha^*(k)$	$\beta^*(k)$	$\gamma^*(k)$
0	0	0	1
⋮	⋮	⋮	⋮
21651	0	0	1
21652	0	1	0
21653	0	0	1
⋮	⋮	⋮	⋮
43300	0	0	1
43301	1	0	0
43302	0	0	1
⋮	⋮	⋮	⋮
64950	0	0	1
64951	0	1	0
64952	0	0	1
⋮	⋮	⋮	⋮
87599	0	0	1

In terms of computational cost, it takes on average about 3.5 min. Recall that, if the switching sequence is pre-specified, the method given in [9] can find optimal switching time instants. As there are 1 main test and 2 partial tests, there are only 3 possible switching sequences, i.e., B-B-A, B-A-B and A-B-B, where A denotes main test and B denotes partial test. Combining the enumeration approach with the method in [9], it is also possible to get an optimal schedule including an optimal switching sequence and optimal switching time instants. For the purpose of validation, we compare the optimal solution (17) obtained by the proposed approach with the results obtained by applying [9] to all possible switching sequences.

Table 3 shows the optimal switching time instants for these three switching sequences obtained with the method of [9]. As can be seen from Table 3, the switching sequence B-A-B gives the minimal performance index (i.e., achieves the minimal value of PFD_{avg}). Thus, the switching sequence B-A-B is the optimal switching sequence. The optimal switching time instants under the switching sequence B-A-B (see the row highlighted in the table) are

$$t_1 = 2.471, \qquad t_2 = 4.943, \qquad t_3 = 7.414. \tag{18}$$

By comparing (17) with (18), we can see that the optimal time instants in (17) and (18) are same for the first 3 digits precision. That means, the optimal solution (17) got by the approach proposed in Sect. 3 can find an optimal solution of the scheduling problem. Note that the proposed approach needs to solve only one optimization problem, while the method of [9] needs to solve an optimization problem for each possible sequence to get the optimal solution.

Table 3. Optimal switching time instants for different switching sequences obtained with the method of [9].

Switching sequence	t_1 year	t_2 year	t_3 year	PFD_{avg}
B-A-B	2.471	4.943	7.414	3.996×10^{-4}
A-B-B	3.707	5.767	7.826	4.150×10^{-4}
B-B-A	2.060	4.119	6.179	4.149×10^{-4}

4.3 PM Schedule for 3 Main Tests and 5 Partial Tests

To investigate the ability of the proposed approach to scheduling more PM tests, we consider scheduling for the SIS described in Sect. 4.1 with 8 tests, namely 3 main tests and 5 partial tests over 3 different lengths of life cycles: 5, 10 and 15 years. The results are shown in Table 4. The sampling time is 1 h and time units in the table are in years.

Table 4 shows an optimal switching sequence and an optimal time instants found by the approach proposed in Sect. 3. The MINOS solver is able to find an

Table 4. Optimal PM schedule for the SIS with 3 main tests and 5 partial tests over different length of life cycle (A: main test, B: partial test)

Optimal PM schedule	Life cycle 5 years	Life cycle 10 years	Life cycle 15 years
σ_1^*	(B, 0.612)	(B, 0.800)	(B, 1.864)
σ_2^*	(A, 1.223)	(A, 2.185)	(A, 3.723)
σ_3^*	(B,1.834)	(B, 2.935)	(B, 5.820)
σ_4^*	(A, 2.445)	(B, 3.939)	(B, 7.444)
σ_5^*	(B, 2.814)	(A, 4.944)	(A, 8.649)
σ_6^*	(B, 3.234)	(B, 6.276)	(B, 9.889)
σ_7^*	(A, 3.829)	(B, 7.607)	(A, 11.128)
σ_8^*	(B, 4.359)	(A, 8.747)	(B, 13.007)
PFD_{avg}	1.556×10^{-4}	2.886×10^{-4}	4.251×10^{-4}

optimal solution for all cases. Note that if the enumeration approach combined with the approach in [9] is considered, the number of the possible sequences is 56. However, the method in [9] becomes cumbersome to use because the first and second derivatives of the cost function must be calculated for every sequence. Moreover, the number of possible sequences increases exponentially with the increase in the number of main tests and partial tests.

The resulting $PFD(t)$ of the 1oo1 SIS over 15 years life cycle is shown in Fig. 3. The PFD_{avg} got by the proposed approach is 4.251×10^{-4}.

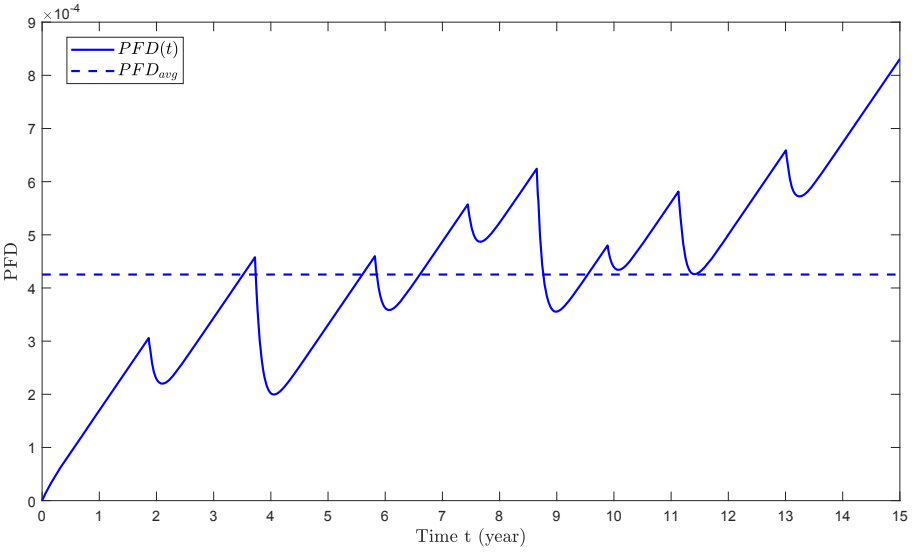

Fig. 3. $PFD(t)$ (solid line) and PFD_{avg} (dashed line) of the 1oo1 SIS with 3 main tests and 5 partial tests over 15 years life cycle

5 Conclusions

In this paper, we give an approach to minimize the average PFD for safety instrumented system (SIS) by optimizing the schedule of main tests and partial tests of the preventive maintenance. The proposed approach is able to optimize both the switching sequence and time instants. The basic idea is to present the state jumps as sub-dynamics described by difference equations and then to combine all sub-dynamics in one difference equation using binary variables. After that, the optimization problem is reformulated as a mixed-integer optimization problem. AMPL can be used as a describing language to input the reformulated optimization problem to the MINOS solver available on the free NEOS server. It is an efficient approach in terms of computational cost. This approach can be extended easily to consider more than two types of maintenance tests. For the SIS with complex architectures (e.g., 2oo3), the matrices Q and M as well as the state vector $p(t)$ in the SIS model (1)–(2) need to be adapted. The next work is to extend this approach to minimize the costs of preventive maintenance or the number of PM tests while satisfying the requirements on the average PFD.

References

1. Torres-Echeverria, A.C.: Modelling and optimization of safety instrumented systems based on dependability and cost measures. Ph.D. thesis, University of Sheffield (2009)
2. Gruhn, P., Cheddie, H.: Safety shutdown systems: design, analysis, and justification. Instrument Society of America (ISA), Research Triangle Park, NC (1998)
3. Catelani, M., Ciani, L., Luongo, V.: A simplified procedure for the analysis of safety instrumented systems in the process industry application. Microelectron. Reliab. **51**(9–11), 1503–1507 (2011)
4. Mechri, W., Simon, C., BenOthman, K.: Switching Markov chains for a holistic modeling of SIS unavailability. Reliab. Eng. Syst. Saf. **133**, 212–222 (2015)
5. Innal, F., Lundteigen, M.A., Liu, Y., Barros, A.: PFDavg generalized formulas for SIS subject to partial and full periodic tests based on multi-phase Markov models. Reliab. Eng. Syst. Saf. **150**, 160–170 (2016)
6. Liu, Y., Rausand, M.: Reliability assessment of safety instrumented systems subject to different demand modes. J. Loss Prevent. Process Ind. **24**(1), 49–56 (2011)
7. Machleidt, K.: Preventive maintenance of safety-related systems-modeling, analysis, and optimization. Ph.D. thesis, University of Kaiserslautern, Department of Electrical and Computer Engineering (2016)
8. McKinnon, K.I.: Convergence of the Nelder-Mead simplex method to a nonstationary point. SIAM J. Optimiz. **9**(1), 148–158 (1998)
9. Martynova, D., Zhang, P.: Optimization of maintenance schedule for safety instrumented systems. IFAC-PapersOnLine **50**(1), 12484–12489 (2017)
10. Xu, X., Antsaklis, P.J.: Optimal control of hybrid autonomous systems with state jumps. In: Proceedings of the 2003 American Control Conference, Denver, USA, pp. 5191–5196 (2003)
11. Giua, A., Seatzu, C., Van Der Mee, C.: Optimal control of autonomous linear systems switched with a pre-assigned finite sequence. In: Proceedings of the 2001 IEEE International Symposium on Intelligent Control, Mexico City, Mexico, pp. 144–149 (2001)

12. Analysis and design of hybrid systems (chap). In: The Control Systems Handbook: Control System Advanced Methods, p. 31. CRC Press (2018)
13. De Marchi, A.: On the mixed-integer linear-quadratic optimal control with switching cost. IEEE Control Syst. Lett. **3**(4), 990–995 (2019)
14. Zhu, F., Antsaklis, P.J.: Optimal control of hybrid switched systems: a brief survey. Discrete Event Dyn. Syst. **25**(3), 345–364 (2014). https://doi.org/10.1007/s10626-014-0187-5
15. Majdoub, N., Sakly, A., Sakly, M.: ACO-based optimization of switching instants for autonomous switched systems with state jumps. IFAC Proc. Vol. **43**(8), 449–454 (2010)
16. Seatzu, C., Corona, D., Giua, A., Bemporad, A.: Optimal control of continuous-time switched affine systems. IEEE Trans. Autom. Control **51**(5), 726–741 (2006)
17. Bemporad, A., Giua, A., Seatzu, C.: Synthesis of state-feedback optimal controllers for continuous-time switched linear systems. In: Proceedings of the 41st IEEE Conference on Decision and Control, Las Vegas, USA, vol. 3, pp. 3182–3187 (2002)
18. Bock, H.G., Kirches, C., Meyer, A., Potschka, A.: Numerical solution of optimal control problems with explicit and implicit switches. Optimiz. Methods Softw. **33**(3), 450–474 (2018)
19. Kirches, C., Kostina, E., Meyer, A., Schlöder, M.: Numerical solution of optimal control problems with switches, switching costs and jumps. Optimiz. Online J. **6888**, 1–30 (2018)
20. Fourer, R., Gay, D.M., Kernighan, B.W.: AMPL A Modeling Language for Mathematical Programming. Thomson (2002)
21. Kirches, C., Leyffer, S.: TACO a toolkit for AMPL control optimization. Math. Program. Comput. **5**(3), 227–265 (2013)
22. NEOS Solver Statistics. https://neos-server.org/neos/report.html. Accessed 20 May 2020
23. Murtagh, B.A., Saunders, M.A.: MINOS 5.51 user's guide. Technical report, Stanford University, Systems Optimization Lab (2003)