

Modular redundancy and opportunistic maintenance optimization for a multi-state weighted k-out-of-n system

Hadi Abdollahzadeh^a, Amir Noroozi^b

^a Department of Industrial Engineering, Malek Ashtar University of Technology
Tehran 15875-1774, Iran

Tel: +989112111745, Email: Hadi.sangrodi@gmail.com

^b Department of Industrial Engineering, Iran University of Science and Technology,
Tehran, Iran

Tel: +989115853008, Email: Amir_noroozie@iust.ac.ir

Abstract

In This paper the redundancy and imperfect opportunistic maintenance optimization of a multi-state weighted k-out-of-n system is formulated. The objective is to determine the k-out-of-n system redundancy level and the maintenance strategy minimizing the life cycle cost subject to an availability constraint. A new condition based opportunistic maintenance approach is developed. Different component health state thresholds are introduced for imperfect maintenance of failed subsystems and working subsystems and preventive dispatching of maintenance teams. In addition, a simulation method is developed to evaluate the performance measures of the system considering different types of subsystems, maintenance activation delays and durations, limited number of maintenance teams, and discrete inspection of the system. Also, a multi-seed tabu search heuristic algorithm is also proposed to solve the formulated problem. An application to the optimal design of a wind farm is provided to illustrate the proposed approach.

Keywords:

Weighted k-out-of-n system, Redundancy, Economic dependence, Periodic inspection, Opportunistic maintenance, Three-phase simulation.

Introduction

Weighted k-out-of-n system structure is widely used in the design of engineering systems. In this kind of system structure, each component/subsystem carries a weight of $w_i > 0$, $i = 1, 2, \dots, n$. The system works if and only if the total weight of working components/subsystems ($W = \sum_{i=1}^n w_i$) is at least a predetermined value (k) [1]. The weight associated with each subsystem refers to the

operational contribution of that subsystem [2]. For example, in a wind power system, the production rate of a wind turbine reflects each turbine weight in the wind farm. Literature indicates that several different researchers have contributed to the optimization of repairable k-out-of-n system performance indices. These studies can be broadly put into three categories according to the types of their decision variables: 1) redundancy optimization, 2) maintenance optimization, and 3) Joint redundancy and maintenance optimization.

Structural redundancy is to consider additional components/subsystems in the system structure so that the system can perform the intended function if one or more subsystems failed [3]. Several different authors [4-6] have considered the k-out-of-n system redundancy optimization problem. In these studies, it is assumed that the system is composed of components which only experience two states. However, in many practical cases, a k-out-of-n system may include multi-state subsystems/component [7]. Furthermore, these researchers focused on unlimited repair capacity. Nevertheless, the infinite capacity is not a practical approach for real cases [8]. On the other hand, Misra [9], Sharma and Misra [10], Fawzi and Hawkes [11], Wang [12, 13] and Gurov et. al [14] have addressed the optimal redundancy design considering limited repair capacity. Also, Khatab et al.[15], Yuan [16], Ruiz-Castro and Li [17] address k-out-of-n system performance assuming that there exists a finite number of repair persons. In these studies, the researchers present their models based on the assumption of perfect maintenance actions. In practice, however, repair efforts usually lead a component to an intermediate condition, worse than a new one but better than just before the failure [18]. In these studies, it is assumed that components are sent for repair immediately upon failure. However, maintenance policies usually rely on economic and performance dependencies among subsystem components [19].

As classified above, the literature indicates that some researchers have contributed to the maintenance optimization of k-out-of-n systems. Pham and Wang [20] proposed a time based opportunistic maintenance policy named as (τ, T) maintenance policy, where τ and T indicate the minimal and perfect repair times, respectively. Krishnamoorthy and Ushakumari [21], Krishnamoorthy et al. [22], Krishnamoorthy and Rekha [23] and Ushakumari and Krishnamoorthy [24], Qi et al. [25] proposed maintenance policies based on the number of failed components, fix time intervals, and combination of time interval and number of failed components. Ding and Tian [26, 27] reported an opportunistic maintenance policy based on the component's age threshold. Abdollahzadeh et al. [28] introduced a bi-objective opportunistic maintenance optimization model considering component reliability thresholds. The maintenance policies introduced in the above-mentioned research works are proposed based on life distribution of system components without consideration of current health state of components. However, the use of current health state of system components in the decision-making process of maintenance activities has a critical role in reducing the waste of resources. Tian et al. [29] developed a condition based maintenance method based on two failure probability thresholds. They assumed that the health states of system components are available when one approaches this policy. In fact, Tian et al. [29] did not discuss that data used in their research are from a continuous monitoring or from inspection in discrete time intervals. In general, continuous monitoring of a system is not always practicable either due to economic or technical constraints. In these cases, data are usually collected intermittently based on system inspection [30] and thus, the inspections must be conducted based on optimized inspection intervals.

Few researchers have considered the third category of studies introduced above. Levitin and Lisnianski [31], De Smidt-Destombes et al. [32], and Moghaddass et al. [8], Nourelfath et al. [33], Liu et al. [34], Chen et al. [35], and Xiao and Peng [36], Atashgar and Abdollahzadeh [37] have provided their researches based on the third approach.

In this paper, a new joint redundancy and maintenance optimization model for a weighted k-out-of-n system is introduced. The paper attempts to minimize the expected system life cycle cost on an availability constraint determining system structure, maintenance policy, system inspection intervals, and maintenance capacity. An opportunistic maintenance approach is developed for k-out-of-n systems considering: 1) multi-state subsystems which compose of several multi-state components, 2) periodic inspection of the components health stats, 3) imperfect preventive repair actions, 4) several types of operational and maintenance costs, 5) maintenance activation delays, 6) limited number of maintenance teams and 7) economic dependence among subsystems and their components. The system performance indices, including availability and life cycle cost, are calculated by a three-phase discrete event simulation method. A tabu search algorithm is also approached to find the optimal solution of the proposed optimization model.

Problem description

Let N_{max} denotes the maximum number of subsystems allowed to install in a weighted k-out-of-n: G system. k types of subsystems are available and each subsystem contains N ($j = 1, 2, \dots, N$) main components. In this case, it is assumed that the components of a subsystem are connected in series and each subsystem type is categorized based on initial cost, component failures/repair characteristics, and the performance rate. For each subsystem type i ($i = 1, 2, \dots, k$), there are x_i subsystems connected in parallel (x_i is a decision variable). Therefore, for a given vector $\mathbf{X} = \{x_1, x_2, \dots, x_k\}$ the total initial cost (C_{int}) of the system is calculated as:

$$C_{int} = \sum_{i=1}^k x_i * C_i^{int} \quad (1)$$

where C_i^{int} is the initial costs of i^{th} subsystem type.

The components in a subsystem are assumed to deteriorate over time, and the degradation process of each component follows a multi-state model. In this research the health states of the j^{th} component of each subsystem is represented by a finite set of discrete states $\mathbf{D}_j = \{1, 2, \dots, m_j\}$. State 1 denotes the best health state of the component, and state m_j reflects the failure state of a component. In another word, \mathbf{D}_j represents the deterioration conditions of the j^{th} component. Each state d ($d = 1, 2, \dots, m_j$) is defined by a level of efficiency ($g_{j,d}$), where $g_{j,1} = 1$ to the last addresses the best efficiency rate ($g_{j,1} \geq g_{j,2} \geq \dots \geq g_{j,m_j-1}$) and g_{j,m_j-1} denotes the last acceptable one. The efficiency rate of the failed state is zero (i.e., $g_{j,m_j} = 0$). The efficiency rate of a component ($G_j(t)$) at any time $t \geq 0$ is a random variable that belongs to the set $\mathbf{G}_j = \{g_{j,1}, g_{j,2}, \dots, g_{j,m_j}\}$. Indeed, the subsystem is also multi-state. In this case, it is assumed that the production rate of a subsystem ($GSB(t)$) is a function of the efficiency level of its components and subsystem nominal production rate (NPR). Therefore:

$$GSB(t) = NPR * \underset{j \in \{1, 2, \dots, N\}}{arcmin} (G_j(t)) \quad (2)$$

The total production rate of the system ($W(t)$) at any time $t \geq 0$ equals to the summation of the production rate of all the working subsystems. Given a required demand (W^0), the k-out-of-n system availability is defined as $Pr(W(t) \geq W^0)$. The availability of the system is a function of load demand (W^0), system structure, repair capacity, and maintenance strategy. Assuming the operation period is denoted by T , the availability can be written as follow:

$$A = Pr(W(t) > W^0) = 1 - \frac{TTOL}{T} \quad (3)$$

where $TTOL$ is the total time that the output capacity of the system is lower than required demand.

In addition, in this paper, it is assumed that:

- 1) All the components are stochastically independent and the sojourn time of a component in each health state can follow any distributions function.
- 2) The deteriorating conditions of a component in a subsystem can only be detected when the subsystem is

under inspection. However, if a component deteriorates to the failure state, the failure state is detected immediately, since the failure of a component causes the failure of the subsystem.

- 3) In the case that a component fails, the component must be replaced with a new one.
- 4) A repair action can restore the component state from the state d to any of its previous degraded state r ($d > r$).
- 5) There is a fixed cost for sending maintenance facilities, as well as an access cost to each subsystem.
- 6) Parallel maintenance of the different components of a subsystem is not allowed.
- 7) There is a limit on the number of repair facility or teams that can work simultaneously in the system.
- 8) The time it takes to prepare a maintenance facility and the duration of maintenance activities are also considered in the model and can follow any distribution.

Inspired by Zhou et al. [38], in this paper, three types of maintenance thresholds (TH_j^{PM} , TH_j^{OM} and TH_j^{IM}) are considered. In the proposed policy, the repair facilities are sent to the system either when a failure occurs in the system or the inspection result indicates that at least the state of one of the system components is more than preventive maintenance threshold TH_j^{PM} . After sending a maintenance team, 1) all the failed components are replaced and 2) Preventive repair which restore the component to the previous state TH_j^{IM} is performed on all components whose state are above the preventive maintenance threshold TH_j^{PM} or the opportunistic maintenance threshold TH_j^{OM} .

As discussed above, in many practical situations it is necessary that the system is inspected at discrete time intervals. Therefore, it is assume that the system is inspected every $\Delta T = \rho * \Delta INS$ time interval, where ΔINS is the minimum time between two consecutive inspections and ρ is a decision coefficient ($\rho \in \{1, 2, \dots, N_{INS}\}$). We define N_{INS} as the maximum inspection interval constraint. The proposed maintenance strategy can be described by a vector of $\theta = [\theta_1, \theta_2, \dots, \theta_N, \Delta T, M_c]$, where M_c is the number of maintenance facilities of the system and the sub-vectors $\theta_j = [TH_j^{PM}, TH_j^{OM}, TH_j^{IM}]$ denote the maintenance strategy of j^{th} component type. The thresholds in θ_j have the following relationships:

$$m_j \geq TH_j^{PM} \geq TH_j^{IM} \geq 1 \quad (4)$$

$$m_j \geq TH_j^{OM} \geq TH_j^{IM} \geq 1 \quad (5)$$

Consequently, the objective is to define the optimal system structure, the maintenance policy, the maintenance capacity and inspection intervals for a weighted k-out-of-n system, so that the multi-state system life cycle costs are minimized subject to an availability constraint. The expected life cycle cost of the system is given by:

$$TC = C_{int} + C_M \quad (6)$$

where C_{int} is the total initial investment cost (as describe in Equation (1)) and C_M is the total expected maintenance cost of a k-out-of-n system.

Performance estimation method

This research makes use of a three phase simulation to evaluate the two quantities $TC(X, \theta)$ and $A(\theta, X, W^0)$. Three classes of entities including subsystem components, maintenance teams, and inspection are considered for the studied system. The proposed simulation models the operations in which these entities engaged as a sequence of seven significant events in time. Table 1 addresses these seven events.

Table 1- List of seven important events in the proposed discrete event simulation

Event	Name	description
B_1	End-of-health state	A component degraded to the next healthy state.
B_2	End-of-repair	Maintenance activity of a component is complete.
B_3	End-of-dispatch	A maintenance team arrived and allowed to establish temporarily in the system.
B_4	End-of-inspection interval	The time between two consecutive inspections ends. Causing the whole system to be inspected.
C_1	Begin-repair	Maintenance activity of a component begins.
C_2	Begin-running	A subsystem restarts the production.
C_3	Begin-dispatch	A maintenance facility is sent to the system.

The simulation process is explained in detail as follows:

Step 1: initialize the simulation:

- I. Determine all parameters that are used for the simulation process. The parameters include maximum simulation time (T_{max}), the system configuration ($X = \{x_1, x_2, \dots, x_k\}$) and maintenance strategy vector ($\theta = [\theta_1, \theta_2, \dots, \theta_N, \Delta T, M_c]$).
- II. Determine all the related costs including maintenance activity costs ($CPM_{s,r}^{i,j}$), inspection cost (C_{INS}), fixed cost of sending maintenance facilities (C_{fix}), and the access cost to a subsystem (C_{Access}).
- III. Define the sojourn time distributions of a component in each state, the preparation time distribution of a maintenance team and the distribution of repair activity durations.
- IV. Specify the efficiency rate ($G_j = \{g_1, g_2, \dots, g_{m_j}\}$) of each component, the nominal rate of subsystems (NPR_i), and the total required demand of the system (W^0).
- V. At the beginning of the simulation, the simulation clock (*clock*), loss of load moment (*LOM*), total time of loss (*TTOL*) and the total cost of maintenance (*TCM*) are set to be 0, and will be updated during the simulation process.
- VI. The overall multi-state k-out-of-n system state (*TSS*) is set to be working, which shows the capability of system to meet minimum required demand.

The simulated data of each entity is recorded in a simulation table. Table 2 shows a sample table for a system included two subsystems. In Table 2, "time cell" indicates the occurrence time of a particular event provided by "next activity" column. For example, the first row addresses that after 450 (in an assumed time scale), B1 will happen. It is assumed that all the components are in the initial health state at the beginning of the simulation. Then, *CHS* which



denotes the health state of a component is set to be 1. *OCC* addresses the overall four condition of a component (W: working, F: fail, UR: under repair, and SB: standby). The remaining sojourn time of a component in an individual health state is shown by *CST*. A subsystem may experience four states based on its components overall state as working, fail, under repair and standby. These states are

addressed in the column named *OSS*. In Table 2, *GSB* denotes the production rate of a subsystem. The four states of a maintenance group including idle, dispatch, ready and repair are recorded by *OGS* in the seventh row and the list and sequence of maintenance activity of a maintenance team is shown in the last column. The last row of Table 2 shows the simulated data of the inspection interval.

Table 2- Weighted K-out-of-n system simulation information table

Entity	Name	Time Cell	Availability	Next Activity	Component			Subsystem	
					CHS	OCC	CST	OSS	GSB
1	Component 1 of subsystem 1	450	false	B1	1	W	450	W	600
2	Component 2 of subsystem 1	530	false	B1	1	W	530		
3	Component 3 of subsystem 1	420	false	B1	1	W	420		
4	Component 1 of subsystem 2	760	false	B1	1	W	760	W	600
5	Component 2 of subsystem 2	230	false	B1	1	W	230		
6	Component 3 of subsystem 2	410	false	B1	1	W	410		
Entity	Name	Time Cell	Availability	Next Activity	OGS	List of maintenance activity			
7	Maintenance team 1	--	true	--	Idle	--			
Entity	Name	Time Cell	Availability	Next Activity					
8	Inspection interval	270	false	B4					

Status variable: $clock=0$; $LOM=0$; $TTOL=0$; $TCM = 0$; $TSS=w$; $DueNowlist=[]$; $SML=[]$.

Step 2: Time scan and simulation clock update (phase A). The purpose of this phase is the determination of the next event and updating the total simulated time (*clock*). Considering the simulation table described above, the next event is related to any entity record with the minimum time cell and which has an availability field set to "False". The simulation clock is held constant until the next A phase. Since there may be several non-available entities have this new clock time as their time cell, a list of these entities named as Due-now-list must be formed. For an example, in Table 2 addresses the next event is due at time 230. Thus, the *clock* value is updated to 230. The only entity due to engage in a B at this stage is entity 5, the second component of subsystem 2. Thus, the Due-now-list contains only entity 5.

Step 3: Execute Bs due now (phase B). In this step, The Due-now-list is systematically searched, and the following actions are done for each entity:

- Removes the entity from the Due-now-list.
- Change its availability field to "True".
- Executes the B that is shown in the next activity field.

As shown in Table 1, in the proposed problem there are four events named B. End-of-health state (B_1) represents the changes that will occur in the state of different entities of the system in the instance of components degradation to their next health state. If the component reaches the unacceptable state (m_j), this must be treated as a failure. In addition, the related subsystem state, the overall system state and the loss of load moment must be checked and updated. The changes occur at the end of maintenance activities are approached by End-of-repair (B_2). In this event, the component release to "Standby" state. The maintenance team may move to "Ready" if there is another activity that should be done, otherwise it moves to the "Idle" state. End-of-dispatch (B_3) includes the actions relate to the end of dispatching a maintenance team. In which a maintenance team arrives and allowed to temporary establish in the system. Execution of the End-of-inspection interval (B_4) in the next activity column of the inspection interval indicates that the time between two

consecutive inspections ends. Therefore, it causes the whole system to be inspected and to schedule End-of-inspection after a specified time

Step 4: Attempt all Cs (phase C). In this phase, all the Cs are attempted one after the other. This is done by checking the conditions in the test head of each C. In the studied system; there are three Cs. Begin-Repair (C_1) requires a component waiting for repair and a maintenance team to be "Ready". The effect of this C is to engage the maintenance team and the related component. Therefore, the End-of-repair must be scheduled for some known time. If all the components of a subsystem are in "standby" state, Begin-Running (C_2) can be executed. The related subsystem state is updated to "working". Therefore, the executive must schedule End-of-healthy state for all the subsystem components. Begin-Dispatch (C_3) requires a call for maintenance and a maintenance team to be "Idle". The effect of this C is to engage the maintenance teams and to schedule End-of-dispatch after some known time.

Step 5: check termination condition. In this paper, we consider the maximum simulated time (T_{max}) as a termination condition of the simulation. If the simulation clock does not exceeds the maximum simulated time, repeat step 2, 3 and 4.

Step 6: Performance evaluation. The system availability and the expected system life cycle cost can be calculated as:

$$A(\theta, X, W^0) = 1 - \frac{TTOL}{clock} \quad (7)$$

$$TC(\theta, X) = C_{int} + C_M = \sum_{i=1}^k x_i * C_i^{int} + TCM \quad (8)$$

The model formulation

The general mathematical formulation of the proposed problem will take the following form:

$$Maximum TC(X, \theta) \quad (9)$$

S.T.

$$A(X, \theta, W^0) \geq A_0 \quad (10)$$

$$M_c \leq L_{max} \quad (11)$$



$$\sum_{i=1}^k x_i \leq N_{max} \quad (12)$$

$$m_j \geq TH_j^{PM} \geq TH_j^{IM} \geq 1 \quad \forall j = 1, 2, \dots, J \quad (13)$$

$$m_j \geq TH_j^{OM} \geq TH_j^{IM} \geq 1 \quad \forall j = 1, 2, \dots, J \quad (14)$$

$$\Delta T = \rho * \Delta INS \quad (15)$$

$$\rho \leq N_{INS} \quad (16)$$

$$x_i, \rho, TH_j^{PM}, TH_j^{OM}, TH_j^{IM} \in N \quad \forall i = 1, 2, \dots, k, \quad \forall j = 1, 2, \dots, N \quad (17)$$

The first constraint, Equation (10), requires that the availability of the system should not less than a pre-specified threshold (A_0). Equation (11) ensures the maximum number of maintenance team (L_{max}). Equation (12) specifies that maximum of N_{max} subsystems can be installed in the system. The upper bound on the maximum number of subsystems (N_{max}) can be determined based on physical constraints such as weight, volume, etc. The logical relationships of the each subsystem type maintenance strategy are represented in Equation (13) and (14). Equations (15) and (16) are related to the inspection interval. The last constraint, Equation (17), confirms that the model variables belong to the natural numbers.

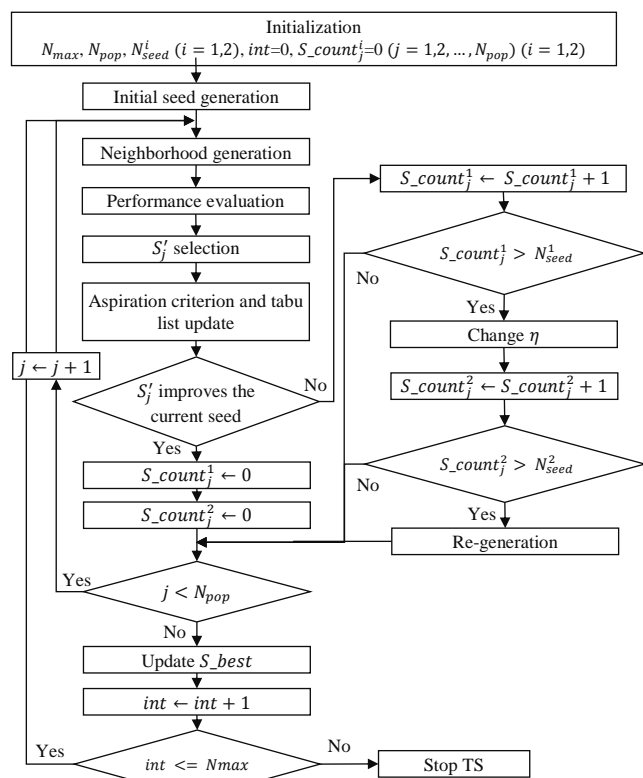


Figure 1- Tabu search flow chart

Tabu search algorithm

The optimization problem formulated in previous section is a constrained non-linear integer programming model. Due to computational time constraint, complete enumeration of all the solution space is not realistic. In this paper, a tabu search optimization algorithm has been developed based on the work of Costa et al. [39]. Tabu

search, which was first introduced by Glover, is a meta heuristic search method that employs neighborhood search methods in the hope of finding an improved solution [40]. The applied tabu search optimization algorithm, unlike regular single-path tabu search, benefits from the search ability of N_{pop} paths. Figure 1 shows the flow chart of the proposed tabu search algorithm. A neighborhood generation procedure is used to search the feasible space. Furthermore, a procedure named as re-generation has been developed to enhance the search mechanism. A detailed description of the different procedures of proposed tabu search algorithm is reported on the following.

Initial seed generation

In this paper, a seed solution is shown in Figure 2. The first K elements of each seed show the redundancy level of each subsystem type. The next $3 * N$ elements of the seed indicate the maintenance strategy thresholds. The last two elements show the inspection interval coefficient and number of maintenance teams, respectively. In addition, each seed solution is randomly produced in accordance with the Equations (11) to (17).

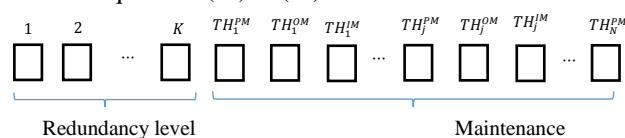


Figure 2- Structure of a seed in the tabu search algorithm

Move representation and neighborhood generation

In this paper, a neighborhood structure is made by two distinct moves. Each move is specified by the notation $[\eta, v]$, where η is the amount added/subtracted to the v^{th} element of the seed solution. If S is a seed solution and $V(S)_1$ is the value of the first element of the solution then the first neighborhood S'_1 is equal to S with the exception of the first element that its value will be $V(S'_1)_1 = V(S)_1 - \eta$. Similarly, for the second neighborhood S'_2 we will have $V(S'_2)_2 = V(S)_2 - \eta$, and so on. The positive value of η makes the second sub-structures. In the same way, the first neighborhood is equal to S except for $V(S')_1 = V(S)_1 + \eta$. It should be noted that infeasible moves with respect to constraint (11) to (17) must be eliminated. Table 3 displays all the feasible or infeasible neighborhood for the seed solution $S = (3, 4, 2, 1, 5, 2, 2)$ for redundancy allocation problem of a k-out-of-n system with one type of subsystem which has one 4 health state element. Underline elements signify the difference between each neighborhood and the seed solution.

Determining the initial value of η for an individual seed solution and changing its value during the search process can directly influence the convergence to a local optimum. In this paper, the initial value of η is set to be one. As reported in Figure 1, the process of changing the value of η is enabled whether there is no improvement in an

individual seed for N_{seed}^1 consecutive iterations. In this procedure, as long as the value of η is less than N_{ss} the value of η is increased by one. Otherwise, the value of η is set to be one.

Table 3- Example of neighborhood for seed $S = (3,4,2,1,5,2,2)$

η	neighbor	v						
		1	2	3	4	5	6	7
+1	1	4	4	2	1	5	2	2
	eliminated	3	5	2	1	5	2	2
	2	3	4	3	1	5	2	2
	3	3	4	2	2	5	2	2
	4	3	4	2	1	6	2	2
	5	3	4	2	1	5	3	2
	6	3	4	2	1	5	2	3
-1	7	2	4	2	1	5	2	2
	8	3	3	2	1	5	2	2
	9	3	4	1	1	5	2	2
	Eliminated	3	4	2	0	5	2	2
	10	3	4	2	1	4	2	2
	11	3	4	2	1	5	1	2
	12	3	4	2	1	5	2	1

Regeneration procedure

To improve the exploration ability of proposed tabu search and to avoid permanent convergence to local optima, this procedure randomly reallocates the current seed solution to another region in solution space based on long-term historical information of all seed solutions. The basic idea is to divide the solution space into distinct regions. Each region is assigned a fitness equal to the result of dividing any number (FN) (i.e. $FN = 10$) by the number of visited solutions from that region. This aims to decrease the fitness of those regions that contain more visited solutions. Then, we apply roulette-wheel selection using these fitness values to select the region from which we will randomly generate a new solution. As shown in Figure 1, the regeneration procedure is enabled whether there is not any improvement after N_{seed}^2 consecutive changes in η .

Constraints handling mechanism

The main idea of this paper to handle infeasible solutions, which violates the availability constraint, is to apply a tournament selection operator. In this procedure, the following criteria are always enforced when two solutions are compared:

1. If the two solutions being compared are both feasible, then select based on the minimum value of the objective function.
2. If one of the two individuals being compared is feasible and the other one is infeasible then select the feasible individual.
3. If both individuals are infeasible, then select the individual with the highest availability.

Further characteristic of tabu search

Tabu moves allow the tabu search to escape from local optima. Nevertheless, good solutions could be rejected because of same tabu moves. Aspiration criterion is what disregards a tabu move in certain situations. In this paper,

aspiration criterion consists of allowing a tabu move, whether it can improve the current best solution.

The total number of iterations N_{max} is used as termination condition of the proposed tabu search. The size of each seed tabu list is considered to be $Tlist$. In addition, each seed solution is evaluated based on the described performance evaluation method.

Application to the optimal design of a wind farm

In this section, to illustrate the proposed approach an application to the optimal design of a 6 megawatt (MW) wind farm is provided. the wind farm case studied by Abdollahzadeh et al. [41] is considered to illustrate the proposed approach of this paper. In studied case, it is assumed that there are four wind turbine suppliers and each supplier produces 600 kilowatt (KW) rated power wind turbines (i.e. $NPR = 600 KW$). Each turbine type involves four main components (i.e. rotor, main bearing, gearbox and generator). The number of the health state of rotor, main bearing, gearbox and generator are equal to 6, 8, 6, and 5, respectively. The sojourn time of a component in each state is distributed by Weibull. Furthermore, the maintenance activities durations, the required time for preparation of equipment and spare parts, and the required time for dispatching of maintenance teams follow lognormal distribution. The access cost of a wind turbine and the fixed cost of dispatching maintenance facilities are 7000\$ and 50000\$, respectively. The inspection cost is equal to 10000 \$ and the minimum and maximum inspection time intervals between two consecutive inspections are 30 and 360 days, respectively.

Sensitivity analysis and simulation validation

This section discusses the influence of the different assumption and parameters over the weighted k-out-of-n system performance. Three maintenance strategies are considered for the system structure consists of three turbine of type 1, four turbines of type 2, two turbines of type 3 and four turbines of type 4 as follows:

- Strategy 1: where only corrective replacement is performed ($\theta = \{[6,6,1], [8,8,1], [6,6,1], [5,5,1], 3,1\}$).
- Strategy 2: where inspections is conducted every 90 days without opportunistic maintenance ($\theta = \{[3,6,1], [5,8,1], [4,6,1], [3,5,1], 3,1\}$).
- Strategy 3: where inspections is conducted every 90 days with opportunistic maintenance ($\theta = \{[3,2,1], [5,3,1], [4,2,1], [3,2,1], 3,1\}$).

Figure 3 and Figure 4 show the estimated wind farm performance according to different simulation time $T_{max} = \{5,10,15, \dots, 60\}$. For each value of T_{max} , the simulation is executed 100 times. It is illustrated that as the simulation time increases, the system performance under three considered maintenance strategy gradually stabilizes



to a certain value. This follows the Monte Carlo simulation sampling principle.

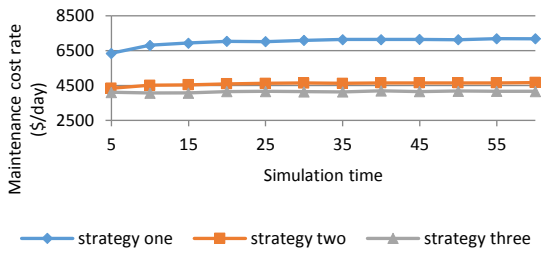


Figure 3- The system expected availability with different simulation time

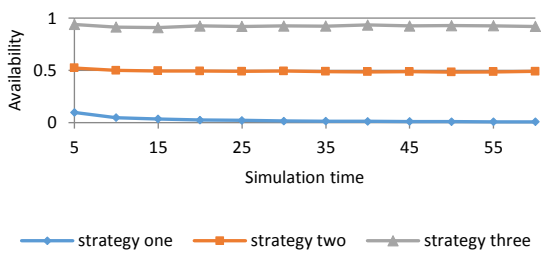


Figure 4- The system expected maintenance cost with different simulation time

In addition, comparing these three strategies indicates that utilization of opportunistic maintenance strategy has an important role in both terms of total expected cost and expected availability. The maximum simulation time set to be $T_{max} = 25$ (year) to control the amount and speed of calculation. Sensitivity analysis of major parameters including the number of maintenance teams, dispatch and access cost, inspection cost, maintenance activity cost and scale parameter of Weibull sojourn times is carried out considering the above three maintenance strategy. For each case, only one parameter is changed for studying and keeping other parameter fixed.

- (1) Number of maintenance team (M_c): Figure 5 to Figure 7 show the increasing trend of the system availability and the expected maintenance cost rate under strategy two and three. Under first strategy, the system performance did not change obviously. Since, under corrective maintenance policy, only failure replacement is performed when a component fails in the wind farm and therefore using one maintenance team will be sufficient.
- (2) Dispatch and access costs of a maintenance team (C_{fix} and C_{Access}): The sensitivity analysis verifies that as the dispatch and access costs increase, the maintenance cost rate also increases (shown in Figure 8). In addition, Figure 8 indicates that maintenance cost rate under corrective maintenance strategy is more sensitive to the dispatch and access cost than under other two strategies.
- (3) Maintenance activity cost ($CPM_{s,r}^{i,j}$): The sensitivity analysis indicates that as the maintenance costs increase, the maintenance cost rate also increases (shown in Figure 9). However, maintenance cost rate

is more sensitive to the maintenance activity costs than the dispatch and access cost.

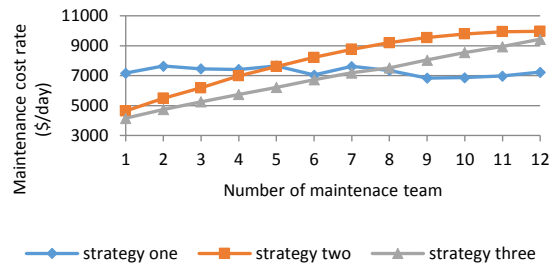


Figure 5- Expected maintenance cost rate with different number of maintenance teams

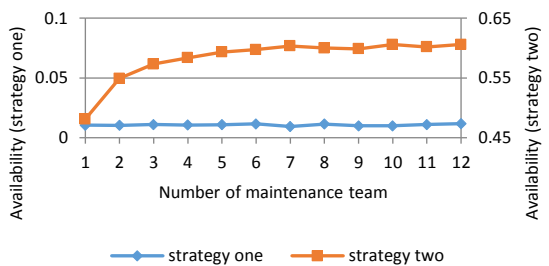


Figure 6- Expected availability with different number of maintenance teams (strategy one and two)

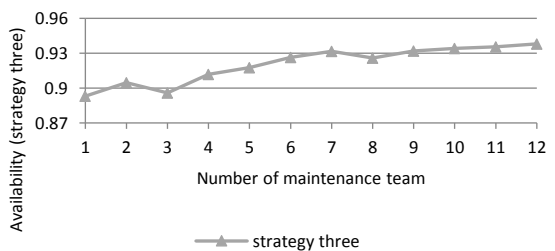


Figure 7- Expected availability with different number of maintenance teams (strategy three)

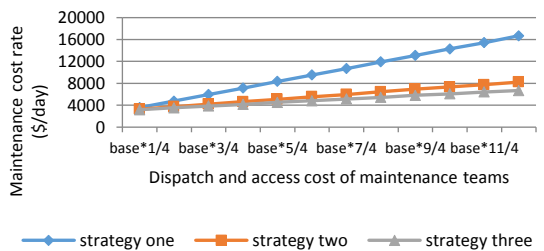


Figure 8- Expected maintenance cost rate with different dispatching and access cost of a maintenance team

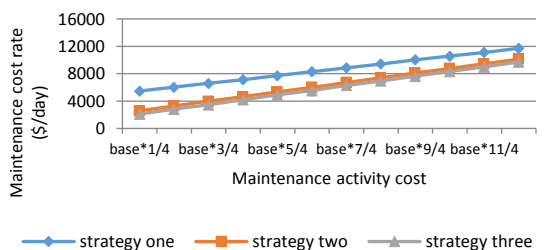


Figure 9- Expected maintenance cost rate with different maintenance activity cost



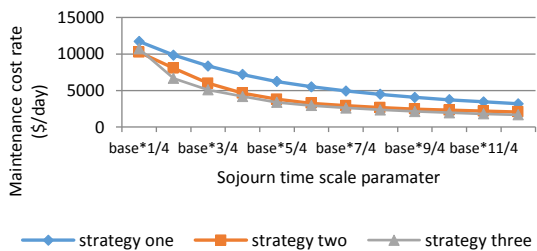


Figure 10- Expected maintenance cost rate with different failure distribution scale parameter

Optimization results

Consider a case that an engineer decides to design a WF with a 6 MW capacity. In this case, it is assumed that due to the land size constraint, one is allowed to install up to 25 turbines, i.e. $N_{max} = 25$. In this paper, a tabu search algorithm is used to optimize the decision variables of the proposed optimization model which corresponds to system structure, maintenance strategy, maintenance capacity and inspection interval. This algorithm uses the average of thirty runs of the proposed simulation method as the evaluation criteria for the feasible solution. The maximum simulation time is set to be 25 years. Table 4 shows the tabu search algorithm parameters tuned based on a sensitivity analysis.

Table 4- Attributes of tabu search algorithm used for processing

Attributes	value
Number of seed solutions (N_{pop})	5
Maximum iteration (N_{max})	80
neighborhood factor (N_{seed}^1)	3
Re-generation factor (N_{seed}^2)	2
Maximum move factor (N_{ss})	3
Tabu list size ($Tlist$)	7

The optimization results obtained for desired availability level of 95% (i.e. $A_0 = 0.95$) are shown in Table 5. The table presents the system availability (A), the system initial cost (C_{int}), the system life cycle maintenance cost (C_M), the structure of the system, the maintenance policy, and the number of maintenance teams. The best founded solution involves the installation of 13 wind turbine of type four which must be inspected every 180 days. This solution suggests using two maintenance team.

Table 5. Parameters of optimal solutions

A	C_{int} ($e^6\$$)	C_M ($e^6\$$)	System structure				Maintenance policy				No. teams	
			Type 1	Type 2	Type 3	Type 4	θ_1	θ_2	θ_3	θ_4		ΔT (day)
0.97	5.2	12.35	0	0	0	13	[2,1,1]	[7,2,2]	[2,1,1,1]	[2,1,1]	180	2

Figure 11 depicts the fitness convergence for the case studied. The near optimal solution is approximately achieved after 50 iterations of the proposed tabu search algorithm. As it is clearly shown in Figure 11 and Figure 12, the first ten iterations of the proposed algorithm pertain to the feasibility of the availability constraint.

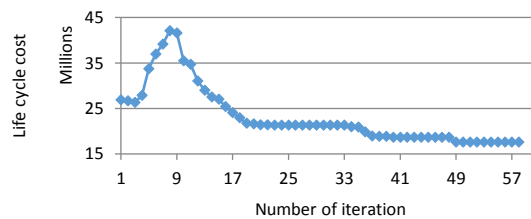


Figure 11. Life cycle cost convergence of the case study

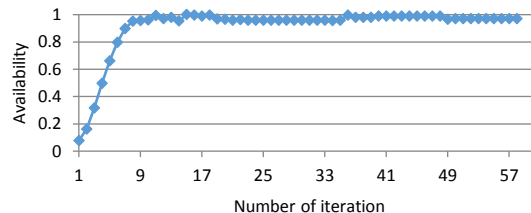


Figure 12. System availability versus the optimization iterations

Conclusion

The performance measures of a weighted k-out-of-n system are directly influenced by the system redundancy level. Additionally, the strategy to manage the maintenance efforts affects the performance of a weighted k-out-of-n system. The objective of this paper is to determine jointly the system configuration, the proposed opportunistic maintenance thresholds, inspection interval and the maintenance capacity that minimized system life cycle cost subject to availability constraint. The main contribution and conclusion of the present study are summarized as follows:

- This study for the first time formulated a joint redundancy and maintenance optimization model for a weighted k-out-of-n system structure which made up of several subsystems compose of several multi-state components.
- A new imperfect condition-based maintenance strategy was introduced for a weighted k-out-of-n system considering inspection interval and economic dependence between subsystems and the subsystem components.
- Three phase discrete event simulation method was developed to evaluate performance measures of a weighted k-out-of-n system considering limited maintenance capacity.
- A novel tabu search optimization algorithm which exploits the search ability of N_{pop} seeds was used to search for the minimum cost solutions.
- A numerical application to the optimal design of a 6 MW wind farm is provided to illustrate the proposed approach.
- Sensitivity analysis was conducted to discuss the influence of the different parameters of the simulation model over the system performance.

Further research will continue to study the stochastic dependence between different components of the system. The stochastic dependency arises when the state of the



component influences the lifetime distribution of the other components or when components are subject to common-cause failures. The proposed mathematical model can also be extended to consider imperfect inspection efforts and a closer analysis of the redundancy optimization considering demand randomness and the cost of unsupplied demand.

References

- [1] W. Li and M. J. Zuo, "Reliability evaluation of multi-state weighted k-out-of-n systems," *Reliability Engineering & System Safety*, vol. 93, pp. 160-167, 2008.
- [2] J. Huang, M. J. Zuo, and Y. Wu, "Generalized multi-state k-out-of-n: G systems," *Reliability, IEEE Transactions on*, vol. 49, pp. 105-111, 2000.
- [3] A. Zaretalab, V. Hajipour, M. Sharifi, and M. R. Shahriari, "A knowledge-based archive multi-objective simulated annealing algorithm to optimize series-parallel system with choice of redundancy strategies," *Computers & Industrial Engineering*, vol. 80, pp. 33-44, 2015.
- [4] S. Kulturel-Konak, A. E. Smith, and D. W. Coit, "Efficiently solving the redundancy allocation problem using tabu search," *IIE transactions*, vol. 35, pp. 515-526, 2003.
- [5] S. V. Amari and H. Pham, "A novel approach for optimal cost-effective design of complex repairable systems," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 37, pp. 406-415, 2007.
- [6] W. Xie, H. Liao, and T. Jin, "Maximizing system availability through joint decision on component redundancy and spares inventory," *European Journal of Operational Research*, vol. 237, pp. 164-176, 2014.
- [7] Y. Wang and L. Li, "A PSO algorithm for constrained redundancy allocation in multi-state systems with bridge topology," *Computers & Industrial Engineering*, vol. 68, pp. 13-22, 2014.
- [8] R. Moghaddass, M. J. Zuo, and M. Pandey, "Optimal design and maintenance of a repairable multi-state system with standby components," *Journal of Statistical Planning and Inference*, vol. 142, pp. 2409-2420, 2012.
- [9] K. B. Misra, "Reliability design of a maintained system," *Microelectronics Reliability*, vol. 13, pp. 495-500, 1974.
- [10] U. Sharma and K. B. Misra, "Optimal availability design of a maintained system," *Reliability Engineering & System Safety*, vol. 20, pp. 147-159, 1988.
- [11] B. Fawzi and A. G. Hawkes, "Availability of an R-out-of-N system with spares and repairs," *Journal of applied probability*, pp. 397-408, 1991.
- [12] K.-H. Wang, "Cost analysis of the M/M/R machine-repair problem with mixed standby spares," *Microelectronics Reliability*, vol. 33, pp. 1293-1301, 1993.
- [13] K.-H. Wang, "An approach to cost analysis of the machine repair problem with two types of spares and service rates," *Microelectronics Reliability*, vol. 35, pp. 1433-1436, 1995.
- [14] S. V. Gurov, L. V. Utkin, and I. B. Shubinsky, "Optimal reliability allocation of redundant units and repair facilities by arbitrary failure and repair distributions," *Microelectronics Reliability*, vol. 35, pp. 1451-1460, 1995.
- [15] A. Khatab, N. Nahas, and M. Nourelfath, "Availability of K-out-of-N: G systems with non-identical components subject to repair priorities," *Reliability Engineering & System Safety*, vol. 94, pp. 142-151, 2009.
- [16] L. Yuan, "Reliability analysis for a k-out-of-n: G system with redundant dependency and repairmen having multiple vacations," *Applied Mathematics and Computation*, vol. 218, pp. 11959-11969, 2012.
- [17] J. E. Ruiz-Castro and Q.-L. Li, "Algorithm for a general discrete k-out-of-n: G system subject to several types of failure with an indefinite number of repairpersons," *European Journal of Operational Research*, vol. 211, pp. 97-111, 2011.
- [18] C.-H. Huang and C.-H. Wang, "Optimization of preventive maintenance for a multi-state degraded system by monitoring component performance," *Journal of Intelligent Manufacturing*, pp. 1-20, 2014.
- [19] R. P. Nicolai and R. Dekker, *Optimal maintenance of multi-component systems: a review*: Springer, 2008.
- [20] H. Pham and H. Wang, "Optimal (τ, T) opportunistic maintenance of ak-out-of-n: G system with imperfect PM and partial failure," *Naval Research Logistics (NRL)*, vol. 47, pp. 223-239, 2000.
- [21] A. Krishnamoorthy and P. Ushakumari, "k-out-of-n: G system with repair: the D-policy," *Computers & Operations Research*, vol. 28, pp. 973-981, 2001.
- [22] A. Krishnamoorthy, P. Ushakumari, and B. Lakshmy, "K-out-of-n-system with repair: The N-policy," *Asia-Pacific Journal of Operational Research*, vol. 19, p. 47, 2002.
- [23] A. Krishnamoorthy and A. Rekha, "k-out-of-n-system with repair: T-policy," *Korean Journal of Computational & Applied Mathematics*, vol. 8, pp. 199-212, 2001.
- [24] P. Ushakumari and A. Krishnamoorthy, "k-out-of-n system with repair: the max (N, T) policy," *Performance Evaluation*, vol. 57, pp. 221-234, 2004.
- [25] X. Qi, Z. Zhang, D. Zuo, and X. Yang, "Optimal Maintenance Policy for High Reliability Load-Sharing Computer Systems with k-out-of-n: G Redundant Structure," *Appl. Math*, vol. 8, pp. 341-347, 2014.
- [26] F. Ding and Z. Tian, "Opportunistic maintenance optimization for wind turbine systems considering imperfect maintenance actions," *International Journal of Reliability, Quality and Safety Engineering*, vol. 18, pp. 463-481, 2011.
- [27] F. Ding and Z. Tian, "Opportunistic maintenance for wind farms considering multi-level imperfect maintenance thresholds," *Renewable Energy*, vol. 45, pp. 175-182, 2012.
- [28] H. Abdollahzadeh, K. Atashgar, and M. Abbasi, "Multi-objective opportunistic maintenance optimization



of a wind farm considering limited number of maintenance groups," *Renewable Energy*, vol. 88, pp. 247-261, 2016.

[29] Z. Tian, T. Jin, B. Wu, and F. Ding, "Condition based maintenance optimization for wind power generation systems under continuous monitoring," *Renewable Energy*, vol. 36, pp. 1502-1509, 2011.

[30] Ü. Gürler and A. Kaya, "A maintenance policy for a system with multi-state components: an approximate solution," *Reliability Engineering & System Safety*, vol. 76, pp. 117-127, 2002.

[31] G. Levitin and A. Lisnianski, "Joint redundancy and maintenance optimization for multistate series-parallel systems," *Reliability Engineering & System Safety*, vol. 64, pp. 33-42, 1999.

[32] K. S. de Smidt-Destombes, M. C. van der Heijden, and A. van Harten, "Joint optimisation of spare part inventory, maintenance frequency and repair capacity for k-out-of-N systems," *International Journal of Production Economics*, vol. 118, pp. 260-268, 2009.

[33] M. Nourelfath, E. Châtelet, and N. Nahas, "Joint redundancy and imperfect preventive maintenance optimization for series-parallel multi-state degraded systems," *Reliability Engineering & System Safety*, vol. 103, pp. 51-60, 2012.

[34] Y. Liu, H.-Z. Huang, Z. Wang, Y. Li, and Y. Yang, "A joint redundancy and imperfect maintenance strategy optimization for multi-state systems," *Reliability, IEEE Transactions on*, vol. 62, pp. 368-378, 2013.

[35] L. Chen, Z.-S. Ye, and M. Xie, "Joint maintenance and spare component provisioning policy for k-out-of-n systems," *Asia-Pacific Journal of Operational Research*, vol. 30, p. 1350023, 2013.

[36] H. Xiao and R. Peng, "Optimal allocation and maintenance of multi-state elements in series-parallel systems with common bus performance sharing," *Computers & Industrial Engineering*, vol. 72, pp. 143-151, 2014.

[37] k. Atashgar and H. Abdollahzadeh, "Reliability optimization of wind farms considering redundancy and opportunistic maintenance strategy," *Energy Conversion and Management*, vol. 112, pp. 445-458, 2016.

[38] Y. Zhou, Z. Zhang, T. R. Lin, and L. Ma, "Maintenance optimisation of a multi-state series-parallel system considering economic dependence and state-dependent inspection intervals," *Reliability Engineering & System Safety*, vol. 111, pp. 248-259, 2013.

[39] A. Costa, A. Alfieri, A. Matta, and S. Fichera, "A parallel tabu search for solving the primal buffer allocation problem in serial production systems," *Computers & Operations Research*, vol. 64, pp. 97-112, 2015.

[40] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM Computing Surveys (CSUR)*, vol. 35, pp. 268-308, 2003.

[41] H. Abdollahzadeh, L. Varela, K. Atashgar, and G. D. Putnik, "Condition Based Maintenance Optimization for Multi-State Wind Power Generation Systems under Periodic Inspection," *FME Transactions*, vol. 43, pp. 319-327, 2015.

