CONDITION-BASED MAINTENANCE OPTIMISATION USING GENETIC ALGORITHM AND CONTINUOUS EVENT SIMULATION TECHNIQUE

AUTHOR

MANSOOR AHMED SIDDIQUI

REGISTRATION NUMBER

00000148450

SUPERVISOR

DR SHAHID IKRAMULLAH BUTT

DEPARTMENT OF DESIGN & MANUFACTURING ENGINEERING

SCHOOL OF MECHANICAL & MANUFACTURING ENGINEERING

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

FEBRUARY, 2017

Condition-Based Maintenance Optimisation Using Genetic Algorithm and Continuous Event Simulation Technique

Author

Mansoor Ahmed Siddiqui

Registration Number

00000148450

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Design & Manufacturing Engineering

Supervisor

Dr Shahid Ikramullah Butt

Thesis Supervisor's Signature: ____

Department of Design & Manufacturing Engineering

School of Mechanical & Manufacturing Engineering

National University of Sciences and Technology

Islamabad

February, 2017

Declaration

I certify that this research work titled "***Condition-Based Maintenance Optimisation Using Genetic Algorithm and Continuous Event Simulation Technique***" is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources it has been properly acknowledged / referred.

Signature of Student

Mansoor Ahmed Siddiqui

00000148450

Language Correctness Certificate

This thesis has been read by an English expert and is free of typing, syntax, semantic, grammatical and spelling mistakes. Thesis is also according to the format given by the university.

Signature of Student

Mansoor Ahmed Siddiqui

00000148450

Signature of Supervisor

Dr Shahid Ikramullah Butt

Copyright Statement

Acknowledgements

I am thankful to my Creator Allah Subhanahu Wa Ta'ala to have guided me throughout this work at every step and for every new thought which He has setup in my mind to improve it. Indeed, I could have done nothing without His priceless help and guidance. Whosoever helped me throughout the course of my thesis, whether my parents or any other individual was His will, so indeed none be worthy of praise but He.

I am profusely thankful to my beloved parents who raised me when I was not capable of walking and continued to support me throughout in every endeavour of my life.

I would also like to express special thanks to my supervisor Dr Shahid Ikramullah Butt for his help throughout my thesis and also for Computer Integrated Manufacturing, Quality & Reliability Management and Supply Chain Management courses which he has taught me. I can safely say that I haven't learned any other engineering subject in such depth than the ones which he has taught.

I would also like to pay special thanks to Dr Aamer Ahmed Baqai for his tremendous support and also for teaching me the subject of Optimisation of Engineering Systems without which I couldn’t have managed to take on this work. Each time I got stuck in something, he came up with the solution. Without his help, I wouldn’t have been able to complete my thesis. I appreciate his patience and guidance throughout the whole thesis.

I would also like to thank Dr Liaqat Ali and Dr Mushtaq Khan for being on my thesis guidance and evaluation committee.

Finally, I would like to express my gratitude to all the individuals who have rendered valuable assistance to my study.

Dedicated to my exceptional parents, amazing wife and adored children whose tremendous support and cooperation led me to this wonderful accomplishment

Abstract

Effective maintenance strategies are of utmost significance for system engineering due to their direct linkage with financial aspects and safety of the plants' operation. At a point where the state of a system, for instance, level of its deterioration, can be constantly observed, a strategy based on Condition-Based Maintenance (CBM) may be affected; wherein, upkeep of the system is done progressively on the premise of monitored state of the system.

In this thesis, a multi-component framework is considered that is continuously kept under observation. In order to decide an optimal deterioration stage for the said system, Genetic Algorithm (GA) technique has been utilized that figures out when its preventive maintenance should be carried out. The system is configured into a multi-objective problem that is aimed at optimising the two desired objectives, namely profitability and accessibility. For the sake of reality, a prognostic model portraying the advancements of deteriorating system has been employed that will be based on utilization of continuous event simulation techniques. In this regard, Monte Carlo (MC) simulation has been shortlisted as it can take into account a wide range of probable options that can help in reducing uncertainty. The inherent benefits proffered by the said simulation technique are fully utilized to display various elements of a deteriorating system working under stressed environment. The proposed synergic model (GA & MC) is considered to be more effective due to the employment of 'drop-by-drop approach' that permits successful drive of the related search process with regards to the best optimal solutions.

Key Words: *Condition-Based Maintenance (CBM), Optimisation, Genetic Algorithm (GA), Monte Carlo (MC) Simulation*

Table of Contents

List of Figures

List of Tables

CHAPTER 1: INTRODUCTION

The research work in this dissertation consists of six main chapters. First two chapters are related to introduction and literature review about the topic; whereas, chapter three contains the methodology and assumptions made for the designed problem. Proposed framework is explained in details in chapter four. In chapter five, model development based on the proposed framework is explained. Moreover, validation of the said model through a case study is also presented in the same chapter. Finally, discussion and interpretation of results is carried out in chapter six. Objective of this research is to study the effects of using Genetic Algorithms and continuous event simulation techniques in a synergic approach for optimising the Condition-Based Maintenance (CBM) of an engineering system.

1.1. Background

Optimisation of maintenance activity is considered to be an interesting subject for scientists and researchers, as can be observed by the abundantly available related conference papers and journals, and also to the industry, for its notable fallbacks to the activities associated with safety and financial aspects.

Distinctive ways and approaches are available to deal with the issue of maintenance. Generally, maintenance is carried out on a restorative premise or planned occasionally based on experience of the professionals. When state can be observed, constantly for the functional systems through testing or investigations for the reserved safety frameworks, a CBM policy can be enforced, which helps in deciding when the maintenance of system should be carried out. Benefits of CBM strategy exist in the likelihood of maintaining the framework just when required; thus, sparing assets as well as the system's accessibility at the very basic level. This strategy indicates remarkable possibilities in frameworks like atomic power plants, costal establishments and aviation setups, which work under severe conditions which can harm their coherence and usefulness while being consistently observed due to the safety repercussions.

During design stage, the need of building up a productive CBM policy is twofold. Firstly, one need to build up an appropriate perceptive model for the system, depicting its future development of deterioration level in the light of observed factors; secondly, one must have the capacity to assess diverse maintenance procedures in an optimisation plan aimed at optimising two main objectives, namely profitability and accessibility.

1.2. Problem Formulation

Let's review a framework comprising of $'N_n'$ nodes (macro-components) in series, each accomplishing its specific task (**Figure *1***). Every $'i_n th'$ node consists of $'N_p(i_n)'$ parallel branches, wherein each branch of the node contains a series of $'N_s(i_n, i_p)'$ components, $i_n = 1, 2, \ldots \ldots N_n$; $i_p = 1, 2, \ldots \ldots N_p(i_n)$. If $'N_t'$ types of components are present in the system, then total components in the system can be given by following expression:-

For Node '1'	**For Node '2'**	**For Node '3'**
$N_p(1) = 4$	$N_p(2) = 3$	$N_p(3) = 2$
$N_s(1, i_p) = 2, i_p = 1,2,3,4$	$N_s(2, i_p) = 3, i_p = 1,2,3$	$N_s(3, i_p) = 1, i_p = 1,2$

$$N_c = \sum_{i_n=1}^{N_n} \sum_{i_p=1}^{N_p(i_n)} N_s(i_n, i_p) \tag{1}$$

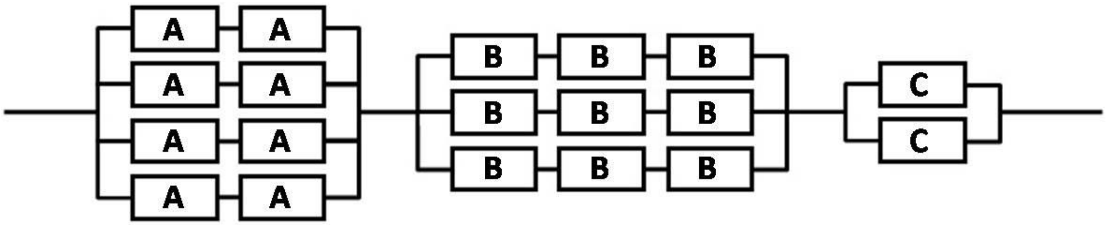

Figure 1. A framework of $N_n = 3$ nodes, comprising $N_t = 3$ component types (A, B and C).

It is assumed that same maintenance policy will be adopted for the components of similar type and the performance of components will deteriorate with the passage of time. Other assumptions made for the system are as follows:-

- The level of degradation for each component is continuously updated;
- Each component has its probability of failure as a function of its deterioration level;
- Each component is subjected to CBM, where the deterioration falls beyond a given level.

The issue in this regard is to determine an optimal deterioration level beyond which the maintenance is necessitated. The issue is configured as an optimisation search based on multiple objective, with the aim to maximize the profitability, $'G'$, and the mean accessibility, $'\bar{A}'$, of the system during a given time $'T_M'$. The mean accessibility, $'\bar{A}'$, of the system with regards to the deteriorating components will be discussed further in Section 3.1. As far as the profitability, $'G'$, is concerned, it can be expressed by following equation:-

$$G = P_t - (M + C) \tag{2}$$

Wherein;

$P_t = S \int_0^{T_M} A\,(t)\,dt$ is the overall profit of plant; $'A\,(t)'$ is the accessibility of plant at time $'t'$ and $'S'$ are the service charges for service of the plant paid by customer per unit time.

$M = \sum_{i=1}^{N_c} M_i \int_0^{T_M} P_M^i\,(t)\,dt$ are the aggregate maintenance charges for all elements of the system, $i = 1, \dots N_c$; $'M_i'$ are the maintenance charges for an element $'i'$ and $'P_M^i\,(t)'$ is the probability of $'i^{th}'$ element being maintained at a time $'t'$.

$C = C_U \int_0^{T_M} ND\,(t)\,dt$ is the compensation paid to the customer due to non-delivery '$ND\,(t)$' of consented services when the system is non-functional; $'C_U'$ is the monetary penalty imposed and $ND\,(t) = 1 - A\,(t)$, $'A\,(t)'$ being the accessibility of plant at time $'t'$.

In order to evaluate system's performance, different mathematical as well as analytical methods exist. For better comprehension and drawing necessary conclusion with regards to their suitability and effectiveness, few of these will be considered in the following sections.

1.3. Objective

This research framework is focused on the issue of finding an optimal condition-based maintenance strategy based on threshold levels of components deterioration beyond which maintenance is necessitated. In this research, the problems associated with stress-dependent deterioration processes of the system components will be considered. The proposed model is expected to prove successful in contributing towards computational resources for searching the optimal solutions.

1.4. Methodology

To carry out this research, relevant information from different sources has been collected and analyzed. The information obtained in this regard belongs to various journals / conference papers, internet databases and other academic literatures. Apart from these, a case study based on real-time data has also been made part of the research. In addition, thorough discussions with my supervisors led to successful completion of my research work.

1.5. Delimitations

This thesis is focused on the basic concepts for optimising condition-based maintenance using Genetic Algorithms and Monte Carlo simulation techniques. Other types of maintenance strategies will not be covered in detail.

CHAPTER 2: LITERATURE REVIEW

Numerous researchers and scientists have concentrated on the issue of developing thorough models for degrading systems. In this regard, Markov's chain and Markov's model have been mostly used for accomplishing the logical outcomes [1]. However, in the abovementioned cases, models are generally based on simple hypotheses. Most of the models available in literature presume that the level of system's deterioration can be ascertained by means of repeated examination only [2]; Vitaly Kopnov [6] considered a situation where the system is consistently observed. Lately, Yeh [3] considered both the cases. An additional hypothesis is to believe that repairs / replacement of the spare parts always re-establish the system to a 'perfect' condition, which, may not be exceptionally practical. As of late, Vitaly Kopnov [6] has considered an issue of limited recuperation. Whenever more practical issues related to dynamic conduct of the system should be taken into account, simple investigative models cannot realistically depict the system, which ultimately forced the researchers to adopt the simulation techniques like the Monte Carlo simulation [5].

As far as optimisation of maintenance system is concerned, the issue fundamentally sums up to decide the level of the observed deteriorated condition of the system after which maintenance should be carried out. Traditional optimisation techniques, e.g. non-linear programming, dynamic programming, gradient methods, mixed integer, and integer programming, regularly involve the utilization of improved models, giving clear assertions regarding optimisation of cost functions, and, perhaps, their derivatives. As already emphasized, the unpredictable performance of the contemporary manufacturing plants and other industrial setups can barely be managed using simpler optimisation models as it is likely that the objective functions and their dependent variables are fixed into complex and more complicated computer codes. Thus, a serious impediment is existing related to the above mentioned optimisation methods in today's dynamic environment.

The researchers are confronted with the daunting challenge of accomplishing several objectives at the same time while trying to optimise a design feature of any engineered framework, such as higher revenues, high dependability, low costs, and low risk of accidents, which may offset each other. Moreover, a few strict requirements pertaining to weight / volume of the system are also required to be fulfilled.

In order to optimise a multi-objective system, the optimisation search methods focused on an individual objective function, being a factored merger of desired objectives while inflicting necessary constraints on the system at the same time, are generally used [10]. This methodology certainly leads to unpredictable definition of the levels of constraints and weighted factors that have been used to optimise the system.

A more useful method is to consider every single objective independently and to identify a set of solutions that is relatively better. Every element of this set is considered to be better or equivalent to the other elements of the set, barring few odds, as far as desired objectives are concerned. Therefore, a set of solutions obtained as a result of multi-objective methodology, described above, gives a range of 'satisfactory' solutions between which a trade-off can be reached.

In this research, a methodology has been proposed that couples the Genetic Algorithms (GAs) [14], with a continuous event simulation technique so as to display a more sensible prognostic model portraying the deteriorating system, and thereby seeking an optimal deterioration level after which preventive maintenance must be carried out while optimising various objectives at the same time. In this regard, Monte Carlo (MC) simulation technique has been shortlisted as it takes into account a wide range of probable options that can help in reducing uncertainty [11]. Basically, we consider 'mean accessibility' and 'total profit', resulted from the operating system over a specified time, as our 'main objectives'. A more common potential solution consists of encoded chromosomes of Genetic Algorithms (GA) population, comprising of a set of variables, one of each type, so as to decide the optimal deterioration level after which preventive maintenance is necessitated. All the potential solutions obtained as a result of GA search will be evaluated using proposed model of MC simulation in order to accomplish the objectives of profitability and accessibility. This coupled approach has already been used by few researchers for both single [12] as well as multi-objective [33] issues. Although different algorithms for Multi-Objective Genetic Algorithm (MOGA) and MC simulation are available, however, for the sake of simplicity and clarity, in-built algorithms offered by MATLAB have been utilized for carrying out the requisite calculations.

2.1. Maintenance Terms and Definitions

Few important terms, that are used in this dissertation, and their definitions have been presented in this section. These terms are directly or indirectly linked with the subject of maintenance engineering [37]:

- *Maintenance:* It includes all actions required either to keep a component or framework operational or to restore it to a specific condition.
- *Maintenance engineering:* A set of maintenance activities that helps in developing the criteria, concepts and technical requirements, to be used during conception as well as procurement phases, for maintaining the components and frameworks during

their operational phase and ensure an efficient maintenance support to the organization.

- *Maintenance concept:* Definition of an overall strategy that specifies the maintenance procedure to be adopted for a particular component or system being considered.
- *Maintenance plan:* An outline of technical and managerial procedures required to be adopted for maintenance of a component or system; usually comprises of tools, facilities, resources, and schedules.
- *Corrective maintenance:* Unplanned maintenance / repair actions executed to restore the operational state of a component or system; carried out due to failures or deficiencies perceived by the users or maintenance professionals.
- *Preventive maintenance:* All maintenance / repair activities executed based on periodic, planned, or a specified program, in order to maintain a component or system in its working condition through inspection and refurbishment. These activities are the preventive steps taken to anticipate or reduce the failure probability in the near future, instead of rectifying the same after occurrence.
- *Predictive maintenance:* Use of different methods and gadgets for diagnosing the condition of a component or system accurately during its operation.
- *Condition-based maintenance (CBM):* A maintenance policy that observes the actual state of a component or system in order to decide the required maintenance activity. It works on the principle that maintenance should be performed only when indicated by deteriorating performance of the system.
- *Quality:* The level to which a component, process, or function conforms to the customer and user needs.
- *Reliability:* Probability of a component to perform its defined task adequately for the specified period of time under certain operating conditions.
- *Maintainability:* Probability of restoring a failed component back to the working condition satisfactorily.
- *Mean time to failure (MTTF):* The duration for which a component or system is expected to remain operational; used to estimate reliability of components and systems.
- *Mean time to repair (MTTR):* An efficiency of a component or system that depends upon its maintainability; given by mean repair time of a component. For exponentially distributed cases, MTTR is inversely proportional to the repair rate.

- *Maintenance person:* A person who carries out preventive maintenance and also reacts to a customer's request to perform corrective maintenance on a component or system; also known as service person, custom engineer, field engineer, technician, repair person, mechanic, etc.
- *Active repair time:* The portion of downtime when maintenance persons are actually performing the repair job.
- *Inspection:* The quantifying observation of a component's performance or state.
- *Overhaul:* A detailed inspection and refurbishment of a component or system to a satisfactory level of operation.

2.2. Condition Monitoring Techniques for CBM

This section is concerned with the technologies that are applicable to the equipment related to power plants with speedy results and financial returns. It is imperative to recognize which condition monitoring technology is more useful and profitable in attaining the desired objectives. Every technique is restricted to particular type of machines and is helpful in ascertaining specific type of issues. Moreover, every technique also provides distinct short as well as long term financial benefits.

Short term financial benefits consist of recognizing and troubleshooting the issues such as unbalance, misalignment, worn couplings or gears, deteriorating bearings, oil contamination or deterioration, lack of lubrication, electrical shorting, loose electrical connections, or poor insulation. These paybacks are mostly quantifiable as they encompass particular machines and their repair costs. These short term benefits can be considerable, and are frequently used as an evidence to validate the original investment in software, hardware, training and personnel.

The most substantial financial benefits, on the other hand, are obtained from long term variations or improvements in operating or maintenance procedures. These fundamental variations in operating and maintenance procedures provide a prospect of elimination of few issues or problems completely, instead of providing advance caution of their happening.

It is also significant to consider that there are practical restrictions associated with each of the condition monitoring techniques; and despite of the state-of-the-art technology, human involvement is imperative for success.

Visual inspection, touching and listening are the most commonly used condition-based maintenance techniques in today's industry. In many cases, human element assists in identification of a problem that was not detected using other analytical techniques. This can contain oil leaks, visibly broken or worn parts, hot bearings or chattering gears etc.

These observations are not restricted to unmonitored components only; in addition, sensory data may also be very significant as an extension to the prognostic analysis. It is important for both maintenance as well as operation personnel to be well trained, as it will help in provision of an expert handling of plant equipment.

Diverse technologies should be adopted as a part of all-inclusive condition-based maintenance program. As majority of the plant equipment consists of mechanical machines or systems, thus vibration monitoring is amongst the mostly used condition-based maintenance programs. However, the aforementioned technique cannot give the complete information required for an effective condition-based maintenance program. This technique is restricted to monitoring of mechanical state of the system only, while none of the other crucial parameters as required for maintenance of efficiency and reliability of the machinery can be ascertained. Hence, an all-inclusive condition-based maintenance program should comprise other diagnostic and monitoring techniques as well. These techniques are [36]:

- Acoustic analysis.
- Vibration monitoring.
- Motor operated valve testing.
- Motor analysis technique.
- Tribology.
- Thermography.
- Visual inspections.
- Process parameter monitoring.
- Other nondestructive testing techniques.

2.3. Data Collection / Analysis

The motive behind a condition-based maintenance program is to transmit data regarding state of the system under observation. The report of condition-based maintenance should incorporate the data that can help the reader to comprehend the effects of condition monitoring easily. Following information should be included in the report [36]:

- A report regarding status of the equipment, encompassing its operating as well as component’s accessibility.
- A list of priority works, incorporating work completed, work in progress and pending works.
- Definitions of status, i.e. critical, marginal and satisfactory.

- An overview of operating status of every item, classifying as inoperable, fully operational, critical or marginal etc.
- Status reports of individual component, e.g. worse or marginal.
- Information about plants' operation, maintenance, and engineering.
- Should continuously display the effect of condition-based maintenance on the system to the management.

The period of reporting is decided based on the plant's requirements, however, it should be at least prepared on annual basis. It is pertinent to highlight that no raw data gathered from an analytical system should form part of the report.

Following features should be made part of the periodic report of condition-based maintenance:

- **Management summary** – Consists of a synopsis, highlighting the actions taken during the period under consideration. Photographs of actual state of the system should be used to show the achievements, whenever possible.
- **Information sharing** – Provides information to the maintenance professional by explaining different phases of the program and giving assistance to other departments as well by sharing important information / examples.
- **Equipment performance** – Specifies a list of components indicated by condition-based maintenance to be abnormal or which has been placed under observation. Normally, state of the equipment can be identified by using a windows format along with other allied documentation. Moreover, those components that have been removed from the alert list are also highlighted in this section.
- **Operating experience and continuous improvement** – Deliberations on latest technologies as well as training acquired by condition-based maintenance professionals are covered under this heading. It can also be used to record any external or internal examples related to operating experience linked with condition-based maintenance program.
- **Cost-benefit** – Provides cost benefits related to condition-based maintenance procedures. It also covers different expenditures that were evaded during operation being unnecessary such as labor hours of maintenance, equipment replacement, or purchase of alternate power.

2.4. Optimisation Techniques

2.4.1 Conventional Techniques

Conventional optimisation techniques are categorized under two different groups: (1) Direct search methods, and (2) Gradient based methods.

In direct search methods, the search strategy is guided by the objective function and its constraints only. These methods are relatively slow and require large number of iterations for convergence, as derivatives are not used for evaluation in these techniques.

On the other hand, first / second order derivatives are used in gradient-based methods to direct the search process. These methods rapidly converge to the optimal solutions, only when objective functions and their constraints are continuously differentiable, else they even fail to achieve the approximately optimal solutions. Thus, gradient based methods are not effective for the issues having discontinuous or non-differentiable objective functions and constraint equations [38].

Common problems related with conventional methods include:

- The process of convergence is dependent on the selected initial solution.
- Mostly, the algorithms got trapped into a local optimal solution.
- Problems in a discrete search space are not dealt with these algorithms effectively.
- These algorithms are not suited for the problems associated with parallel machines.

2.4.2 Evolutionary Algorithms

Evolutionary Algorithms (EAs) belong to a higher category of meta-heuristic procedures. These are based on theory of ‘survival-of-the-fittest’ presented by Darwin. EA process is generally started by resetting a population of feasible solutions to an issue. Fresh solutions are then generated by arbitrarily changing the original population of solutions. The solutions are gauged based on how effectively they accomplish the desired mission. In the end, a criteria for selection is affected in order to eliminate those solutions that do not meet the specified conditions. The procedure is repeated using the specified solutions till the time it meets the particular criteria. Benefits of EAs lie in their flexibility to modification and capability to rapidly produce best solutions. Following aspects make EAs distinguished amongst other optimisation techniques [39]:

- It takes into account a population of solutions, instead of a single solution. The population being considered is capable of moving over the crests as well as through the valleys. Therefore, it can determine a global or close to global optimal solution.

As the computation for every solution in the population is autonomous of the others, therefore, EAs are considered to possess integral capability for parallel computation.

- In order to establish the search direction, EAs use information regarding objective functions and their fitness values, instead of depending on their derivatives or other supporting information. Therefore, EAs are capable of dealing with the functions that are non-continuous, non-smooth, and non-differentiable; i.e. the practical optimisation issues.
- Stochastic transition procedures are generally used by EAs instead of deterministic ones in order to evaluate the generations, giving it a look of probabilistic optimisation algorithm, capable of searching even more complex and ambiguous areas so as to discover the global optima. Therefore, EAs are considered to be more robust and flexible as compared to other conventional optimisation techniques.

CHAPTER 3: MARKOV'S MODEL, MONTE CARLO SIMULATION AND GENETIC ALGORITHMS

3.1. Condition-Based Monitoring Using Markov's Model and Monte Carlo Simulation

3.1.1 Markov's Model

Markov's process is a numerical method that transforms a system from one state to another, within a limited number of feasible states. Markov was of the view that future state of the system is not dependent on its past states when its present state is clearly established.

In order to understand Markov's Model correctly, we need to first get clarity about the Markov's Chain. If $Y_{x+1} = m$ shows a stochastic / probabilistic procedure that is supposed to remain at a state $'m'$ for the time $'x + 1'$, then:-

$$P(Y_{x+1} = m | Y_x = n, Y_{x-1} = n_x, \dots, Y_2 = n_2, Y_1 = n_1, Y_0 = n_0)$$

$$P(Y_{x+1} = m | Y_x = n) = P_{nm} \quad (3)$$

for the states $m, n, n_x, \dots, n_0$ and all $x \geq 0$. This kind of probabilistic procedure is termed as Markov's Chain. The value $'P_{nm}'$ indicates the probability of stochastic procedure to transform the system from state $'n'$ to $'m'$. As per Ross's theory, all the probabilities are greater than or equal to $'0'$ and that the stochastic procedure must transform to the other states, thus:-

$$P_{nm} \geq 0 \; and \; \sum_{m=0}^{\infty} P_{nm} = 1, n = 0,1,2, \dots. \quad (4)$$

Representing the $'x + y'$ probabilities of system transition by $P\left(X_{x+y} = m \middle| Y_0 = n\right) = {P_{nm}}^{x+y}$ and computing the same by Chapman-Kolmogorov's equation, we get:-

$$P_{nm}^{x+y} = \sum_{k=0}^{\infty} P_{nk}^{x} P_{km}^{y} \quad (5)$$

for all $x, y \geq 0$. In equation (5), the term $'P_{nk}^{x} P_{km}^{y}'$ shows probability of system's transformation from state $'n'$ to $'m'$ during $'x + y'$ transitions following a path that will take the system to state $'k'$ after $'x^{th}'$ transition.

In general, Markov's process can be termed as an adjunct to the Markov's chain. If $E = \{0,1,2,3, \dots\}$ denotes the state space and $(Y(t)|t \geq 0)$ represents a Markov's process, then as per Markov's chain definition, $(Y(t)|t \geq 0)$ is called a *'continuous-time Markov's chain'*, wherein, states $n_x, n, m \in E$ and $0 \leq z_1 < z_2 < \cdots < z_x < t^* < t^* + t$. Mathematically, the same can be expressed as under:-

$$P_{nm}(t^*, t^* + t) = P(Y(t^* + t) = m | Y(t^*) = n, Y(z_x) = n_x, Y(z_{x-1}) = n_{x-1}, Y(z_{x-2}) = n_{x-2} \dots Y(z_1) = n_1) \quad (6)$$

$$= P(Y(t^* + t) = m | Y(t^*) = n)$$

Where;

$'P_{nm}(t^*, t^* + t)'$ is the transition probability which indicates that if a system were at state $'n'$ at time $'t^*'$, then it will move to the next state $'m'$ at time $'t^* + t'$.

This Markov's process is considered to have Markovian property which affirms that the future state of a system will be independent of its past state $'Y(z_k) = n_k'$, provided present state $'Y(t^*) = n'$ of the system is known. Therefore, equation (6) can be elucidated as follows:-

$$P_{nm}(t) = P(Y(t^* + t) = m | Y(t^*) = n) \quad (7)$$

The abovementioned Markov's process is known as '*continuous-time homogeneous Markov's process*', where $'P_{nm}(t)'$ is the probability of transition of Markov's process that depends the length of time interval $'t'$ rather than the actual time $'t^*'$.

In case, equation (6) cannot be elucidated as the equation (7), it shows that the transition probability depends on the actual time interval $'(t^*, t^* + t)'$ as well as the starting time $'t'$. As per the definitions of stationary and homogeneous transition probabilities [34], these are always modified with respect to time, which is termed as '*continuous-time inhomogeneous Markov process*'.

Chapman-Kolmogorov equation of '*continuous Markov process*' is, therefore, the solution of transition probabilities within a particular time interval. The said equation is as under:-

$$P_{nm}(t^* + t) = P(Y(t^* + t) = m | Y(0) = n) = \sum_{k \in E} P_{nk}(t^*) P_{km}(t) \quad (8)$$

for $n, m, k \in E$. Equation (8) illustrates that the system will move to the condition $'m'$ during a time period $'t^* + t'$ starting from condition $'n'$ on a path that would take the system towards condition $'k'$ at the time $'t^*'$. A feasible interpretation of the time progression of a component is depicted in **Figure *2***.

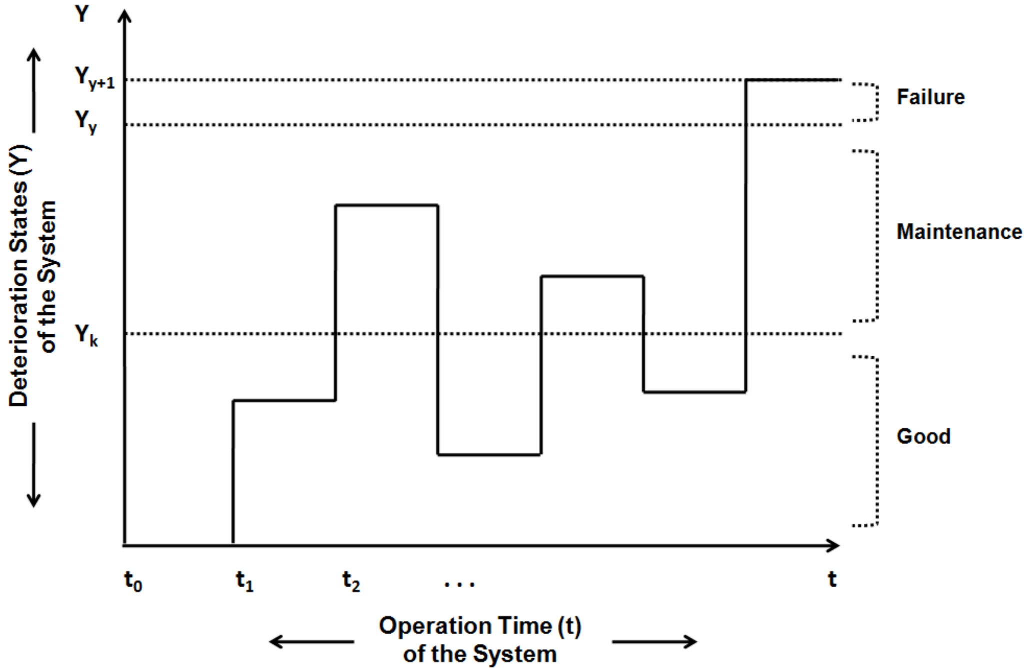

Figure 2. Deterioration process of an element in a given time.

Finally, the system's accessibility $'A(t)'$ at a given time $'t'$ along with its probability of being under repair $'P_M(t)'$ can be expressed using the following equations:-

$$A(t) = \sum_{n=0}^{k} \sum_{m=1}^{k+1} P_{nm} \quad (9)$$

$$P_M(t) = \sum_{n=k+1}^{y} \sum_{m=k+2}^{y+1} P_{nm} \quad (10)$$

Using the abovementioned Markov's Model, objective function as well as the mean system accessibility $'A(t)'$ over a given time $'t'$ can be computed. However, a systematic approach to evaluate the system accessibility is possible only if simplified hypotheses are made. Therefore, a more pragmatic approach towards modeling of the process is essentially required. In order to deal with such situations, Monte Carlo (MC) simulation provides a better bailout solution.

3.1.2 Monte Carlo (MC) Simulation

In the field of reliability sciences, simulation can be termed as a good substitute to the analytical techniques. Prediction of component's accessibility within a given timeframe by MC simulation is the implementation of the said concept. It is pertinent to highlight that the simulation results acquired by MC simulation conform to the ones achieved with the Markov's Process that is considered to be a more traditional technique.

MC simulation can be described in many ways; however, one thing that is common in all the definitions is that it carries out random sampling to reach a solution. The algorithms of MC simulation depend on '*pseudo-random numbers*' to generate a feasible outcome of the process. All realizations are expected to have different probabilities, and by repeating the process using different '*pseudo-random numbers*' as input, accurate data regarding modeled processes is obtained. Based on the acquired data, a statistical analysis can then be performed so as to answer various questions with regards to the process.

After several MC simulation runs, we achieve numerous independent interpretations of the above mentioned pseudo-random variables whose group averages provide estimates of the system accessibility $'A(t)'$ as well as the probability of component $'i'$ being under maintenance $'P_M(t)'$, where $i = 1, \dots, N_c$. Using the said data, an estimate of the desired objective functions, i.e. mean accessibility $'\bar{A}'$ and net profit $'G'$ of the system can be estimated accordingly.

In order to model the dynamics of maintenance, the process strictly depends on the quantity of available maintenance professionals. If none of the professionals is available, due to their commitment on some other components, the faulty equipment has to be kept on hold before its repair can be commenced, i.e. in MC simulation, a stochastic transition of a component is allowed towards an operational state only when the quantity of components being repaired simultaneously is lesser than the quantity of unoccupied maintenance professionals.

3.2. Optimising the Solution through Genetic Algorithm (GA)

Genetic Algorithm (GA) was first validated as an optimisation technique by Holland in 1975 [15]. It is a search method evolved to imitate the processes of natural evolution. GA is different from other optimisation methods owing to its global searching capability accomplished by a population of solutions as opposed to a single solution. Each of the proposed solution is depicted by a vector $'Y'$, comprising variables that are independent of one another, further coded into chromosomes, consisting of different genes, each representing an element of the said vector. Generally, binary coding technique is utilized for the said purpose.

At the start of search process in GA, the generation of randomized initial population of $'X_{ga}'$ number of chromosomes is carried out, that are the feasible solutions. Next, the evaluation of the said chromosomes in terms of their fitness is carried out. This generated initial population is further evolved during successive iterations. Evaluation of objective function is carried out each time as a fresh solution $'Y'$ is suggested as a result of optimisation process. Consequently, individuals' ranking in the present population is updated, keeping in view the values of their individual fitness. The same is then utilized during selection process, giving the best individuals a greater chance of selection as parents. Moreover, ranking of the individuals is also utilized during substitution process in order to decide that whether the parents or the off-springs should carry forward to the next population or otherwise. A genetic algorithm is termed as a '*steady-state genetic algorithm*' based on above mentioned processes [35].

Multi-objective optimisation issue occurs when we have to take into account various objective-functions $f_n(Y)$, $n = 1, 2, ...$, corresponding to each point $'Y'$ in the complete search space, followed by establishment of a point $'Y^*'$ that generates the best possible trade-off between different objective functions. Let's take into account $'N'$ number of distinct objective functions $f_n(Y), n = 1, ..., N$ where $'Y'$ exhibits a vector of variables that are independent of each other, classifying a proposed generalized solution. Solution $'Z'$ is said to be 'dominated' by solution $'Y'$, if $'Y'$ displays improved performance for all the objective functions [10], i.e. if

$$f_n(Y) > f_n(Z) \qquad for \;\; n = 1, 2, ..., N \tag{11}$$

A solution is said to be '*non-dominated*' if none of the cost functions can have improved values without reducing the values of some other objectives. In order to deal with various objective functions simultaneously while following the genetic approach, single-fitness process being used for single-objective GA problems should be generalized by allocating $'N'$ fitnesses to the solutions $'Y'$.

Once the population of chromosomes $[X]$ is created as desired, ranking of the same is carried out as per Pareto domination criteria by going through the N-dimensional search space of different fitness $f_n(Y)$, $n = 1, 2, ..., N$. Ranking of the chromosomes for $N = 2$ is shown in **Figure *3*** below.

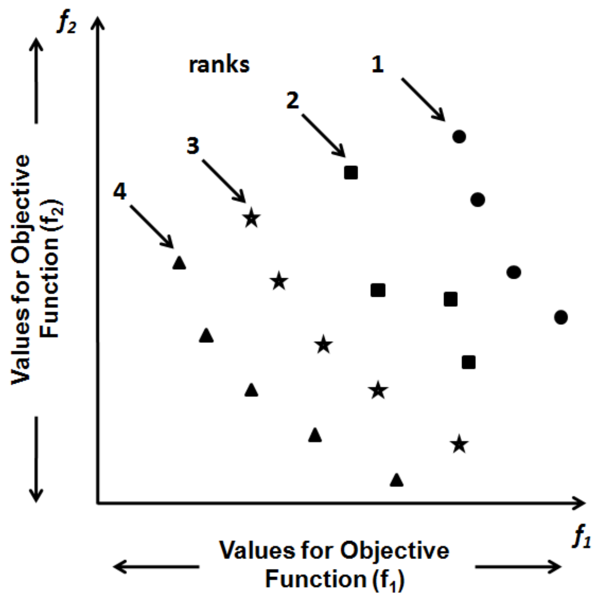

Figure 3. Population ranking for maximization of functions f_1 and f_2.

‘*Non-dominated*’ solutions, in the present population, are determined. The same are then allocated the rank ‘1’, being the best ones. After that, these are separated from the other chromosomes. Then again, a fresh set of ‘*non-dominated*’ solutions is selected, which are allocated the rank ‘2’. The said procedure continues till the time each individual in the population is ranked.

The selection and substitution process of Multiple Objective Genetic Algorithms (MOGA) depends on the above mentioned ranking. Each chromosome that belongs to the same rank should be regarded as equal to the others of the same group, i.e. its probability of selection as a parent and surviving the substitution is equal to any other member of the same group.

While carrying out an optimisation search, the information about vectors updated during previous histories, depicting the Pareto efficiency, are noted and upgraded, which consist of ‘*non-dominated*’ chromosomes and their consequent $'N'$ fitness. On completion of each iteration, evaluation of ‘*non-dominated*’ solutions in the present population is carried out with regards to those already recorded, taking their fitnesses into account. In this regard, following rules are applied:-

- If the existing members in the archive are dominated by a new individual, the same are deleted and the new one is made part of the archive;
- If any member of the archive dominates the new individual, it is not added; and,
- If the new solution is neither dominated nor it dominates any other member of the archive, then:-
  - The new individual is added in the archive, in case the same is not full.
  - In case the archive already has the desired number of solutions, the new individual is exchanged with the one having maximum similarity, present within the record.

Record of ‘*non-dominated*’ solutions is also manipulated by initiating a selection process of elitist parents that is considered to be more effective. Either, every individual in the archive is chosen as a parent during different iterations at least once; or, in case of a larger archive, the quantity of individuals is earmarked by an already defined ratio of the population $'X_{ga}'$, generally 25% i.e. $X_{ga}/4$. This elitism process assures a better implementation of genetic algorithm, resulting into ‘*non-dominated*’ individuals, thus ensuring an effective development of population towards Pareto efficiency, while maintaining the diversity in genetics simultaneously. On completion of the search process, the outcome of optimisation consists of an archive that provides the desired Pareto efficiency region.

CHAPTER 4: PROPOSED FRAMEWORK – MC EMBEDDED IN GA

4.1. Reasons for using Synergic / Coupled Approach

The quest for the optimum maintenance levels necessitates a choice to be made amongst a large number of feasible substitutes. Obviously, running a complete MC simulation for each alternate solution using precise data for the sake of completing only a rudimentary search process is quite unfeasible. Alternatively, when the search process for optimised solution is led by GA, a MC simulation is required to be performed for each individual of the population being considered during consecutive iterations; which is also not practicable. A feasible solution in this regard arises from the fact that in GA approach, better chromosomes appear more number of times during consecutive iterations whereas the others are immediately removed. This leads us to consider a coupled approach, encompassing MC as well as GA and thus ensuring better optimised solution to the problem in much lesser time.

4.2. Proposed Framework

Based on the above mentioned concept, MC simulation is conducted for each of the suggested chromosomes for a specified number of times; e.g. 200 iterations, thus giving a crude approximation of the objective functions. At the time of GA evolution, an updation of record of the best chromosome solutions, acquired during preceding MC iterations, along with their corresponding approximations of the objective functions is carried out.

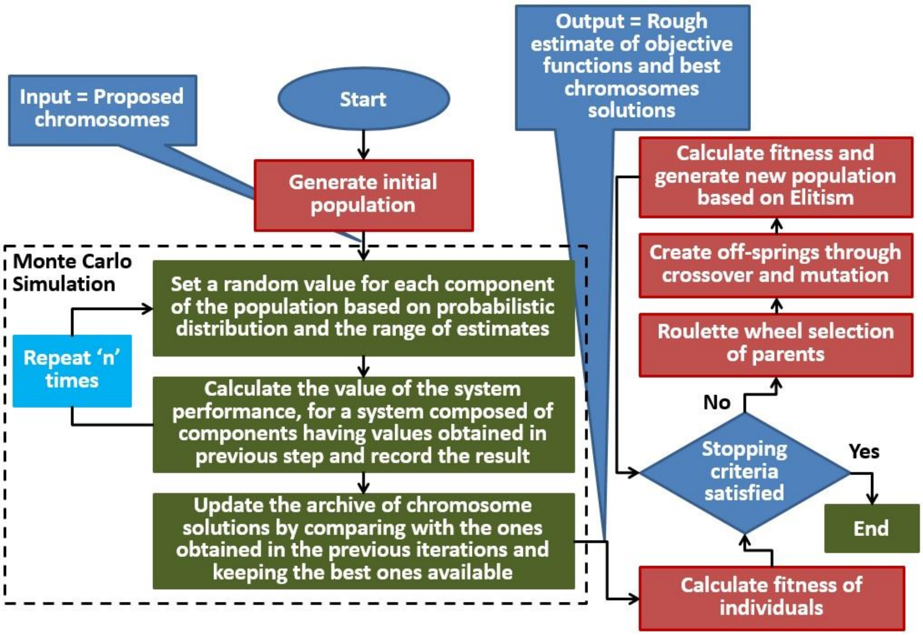

Figure 4. Proposed framework – MC embedded in GA.

When a chromosome is suggested for the second time, the freshly calculated objective function approximation can be stored with those already available in the archive. This repetition can result in the 'best' chromosomes through natural selection procedure while accumulating the outcomes of few last iterations over and over again, thus acquiring arithmetically important results at the end [12]. The said approach is known as 'drop-by-drop approach' due to its resemblance with the process of filling a container with liquid. This technique abstains from working on the 'bad solutions' that are imitated for a lesser repetitions. The proposed framework is illustrated in **Figure *4*** above.

4.3. Pseudo Code for Proposed Framework

Pseudo code for the proposed framework, embedding MC simulation in GA, is illustrated as under. The same will be utilized for further development of MATLAB code for the suggested approach. Interaction of MC simulation and GA is quite evident in the pseudo code. This interaction is expected to provide better results for CBM optimisation problem in lesser time as compared to other combinatorial algorithms.

Technique **GA** (n, iterations) {

Iteration $n = 0$;

Generate initial population $'P(n)'$;

Technique **MC** (N, iterations) {

Initialize $N = 0$;

While (not done)

{

Set random value $'X'$ for each component of population $'P(n)'$;

$X(P(n)) = rand\ (0, 1)$;

Calculate corresponding values of fitness function estimates $'F^*'$;

Compute $F^*(X)$;

If $(F^* \geq F)$, then; % $'F'$ is the previous values of fitness function estimates in the archive;

Compute value of chromosome $'C'$ corresponding to random value $'X'$;

Compute $C\ (X)$;

Accept $'C'$;

Add $'C'$ in population $'P(n)'$ till the archive contains desired number of $'X_{ga}'$ chromosomes;

Else if archive is full, replace $'C'$;

$C += C\ (X)$

Else,

Reject $'C'$;

$N = N + 1;$

}

}

Evaluate new population $'P(n)'$;

While (not done)

{

Carry out Roulette Wheel selection of parents;

$Parents(n) = Select\ parents(P(n))$;

Procreate off-springs through crossover and mutation of parents;

$Off - springs(n) = Procreate(Parents(n))$;

Calculate fitness and generate new population based on the concept of elitism;

$Evaluate\ (Off - springs\ (n))$;

$P\ (n + 1) = Select\ survivors(P(n), Off - springs(n))$;

$n = n + 1;$

}

}

CHAPTER 5: MODEL DEVELOPMENT BASED ON PROPOSED FRAMEWORK

5.1. Variables for Research

5.1.1 Input Arguments

Input arguments specified for this research are as under:-

- **Redundancy** - System redundancy, i.e. number of units;
- **Duration** - Simulation duration (simulation ending criterion);
- **RandFunc** - Anonymous function that will generate the random durations corresponding to state transitions instants;
- **OptAval** will inject the following arguments in this anonymous function:-
  - the number of the simulated unit;
  - its current time; and,
  - its current state.

5.1.2 Output Arguments

Output arguments specified for this research are as under:-

- **Time** - Time vector, concatenating all the time realizations of units;
- **Number** - Number of operational units' vector with respect to the time vector;
- **States** - Matrix concatenating the states of all units with respect to the time vector;
- **Probabilities** - Estimated probabilities of each of the number of operational units;
- **MTTF** - Estimated mean time to failure (mix of all units data); and,
- **MTTR** - Estimated mean time to recovery (mix of all units' data).

5.2. Concept of Redundancy in System Design

Input and output variables set for this research have already been explained in preceding chapters. However, concept of redundancy in this context requires some elaboration. 'Redundancy provides the alternate means in a system for achieving a given mission in such a way that all the techniques are expected to exhaust before resulting into a system failure'. Mean life and reliability of a system can be enhanced by adopting various supplementary techniques at different levels. Approaches available for introducing concept of redundancy in a system include [40]:

- 'Unit or system redundancy' provides an equivalent route for the complete system.
- 'Component redundancy' provides an additional route for each individual component of the system.
- The third technique advocates identification of weak components and strengthening of their reliability.
- The last technique is a combination of aforementioned techniques. It is known as 'mixed redundancy' and is used based on system configuration and reliability requirements.

There are different factors that affect the application of a specific technique. For instance, the size, weight, operating characteristics, and initial cost of items or the systems. Generally, for electronics and electrical equipment, operating characteristics of major systems exhibit nonconformance to the use of redundancy concept at the item level. Therefore, these systems should be given due consideration.

Based on the use, redundancy can be categorized into two main types; i.e. it can be 'active', wherein, all redundant components are operated at the same time while performing the identical job or it can be 'standby', wherein, an equivalent component is brought into action as and when a main component falls.

5.3. Algorithm for the Proposed Framework

Based on the variables finalized and the framework proposed in Chapter 4, an algorithm has been developed in MATLAB to validate this research. Detailed discussion and interpretation of results will be carried out in Chapter 6 of dissertation. MATLAB code developed for the purpose is as follows. In this code, effort has been made to incorporate the help module as well for easy comprehension:-

```
%%    Optimising System Accessibility using GA and MC Simulation
%     Author              : Mansoor Ahmed Siddiqui
%     Version             : v1.0
%     Date                : 10 Dec 2016
%     Calling Sequence    :
      [Time, Number, States, Probabilities, MTTF, MTTR] =
      OptAval(Redundancy, Duration, RandFunc)
```

```
%       Input Arguments    :
%       - Redundancy       : System redundancy, i.e. number of units
%       - Duration         : Simulation duration (simulation ending
                             criterion)
%       - RandFunc         : Anonymous function generating the random
                             durations corresponding to state transitions
                             instants
%       OptAval injects the following arguments in this anonymous
        function:
%             - the number of the simulated unit
%             - its current time
%             - its current state
%
%       Output Arguments:
%       - Time             : time vector, concatenating all the time
                             realizations of units
%       - Number           : number of operational units' vector wrt TIME
                             vector
%       - States           : matrix concatenating the states of all units
                             wrt TIME vector
%       - Probabilities    : estimated probabilities of each of the number
                             of operational units
%       - MTTF             : estimated mean time to failure (mix of all
                             units data)
%       - MTTR             : estimated mean time to recovery (mix of all
                             units data)
%
%       Example:
%
%       %%    Optimising System Accessibility using GA and MC Simulation
%       %     Redundancy         : 4 units
%       %     Time to failure    : MTTF = 5 days, exponential distribution
%       %     Time to recovery   : MTTR = 12h,    exponential distribution
%       %
%       %     This example could be analyzed more directly through the
              study of its Markov chain's stationary distributions
%
%       %     Redundancy
%             Redundancy = 4;
%
%       %     MTTF
%             MTTF = 5*24;
```

```
%
%     %     MTTR
%           MTTR = 12;
%
%     %     Simulation duration
%           Duration = 300*MTTF;
%
%     %     Mean time wrt previous state
%           MT = @(State) eq(State,0)*MTTR + eq(State,1)*MTTF;
%
%     %     Random duration exponential generator (parameter: MTTF or
            MTTR wrt previous state)
%           RandFunc = @(Unit, Duration, State)-log(1-
            rand(1,1))*MT(State);
%
%     %     Optimising System Accessibility using GA and MC Simulation
            including the Redundant Units
%           [Time, Number, States, Probabilities, MTTF, MTTR] =
            OptAval(Redundancy, Duration, RandFunc);
%
%     %     Report in MATLAB console
%           Time_format = @(t)sprintf('%ud %s', floor(t/24),
            datestr(t/24,'HH:MM:SS'));
%           fprintf(1,'\n%s\n',repmat('-',1,80));
%           fprintf(1,'\nOptimised system accessibility using GA & MC
            simulation : system including 3 redundant units.\n\n');
%           fprintf(1,'- Temporal data :\n');
%           fprintf(1,'  . Simulation duration:
            %s\n',Time_format(Time(end)));
%           fprintf(1,'  . Estimated unit MTTF:
            %s\n',Time_format(MTTF));
%           fprintf(1,'  . Estimated unit MTTR:
            %s\n\n',Time_format(MTTR));
%           fprintf(1,'- Minimum number of operational units :\n\n');
%           fprintf(1,'  Number | Probability [%%]\n');
%           fprintf(1,' --------|-----------------\n');
%           for p = 1:numel(Probabilities)
%                 Prob = 100*Probabilities(p);
%                 if Prob < 10, Character = ' ';
%                 else Character = '';
%                 end
%                 fprintf(1,' %2u  |   %s%.3f\n',p-1,Character,Prob);
```

```
%           end
%           fprintf(1,'\n');
%

function [Time, Number, States, Probabilities, MTTF, MTTR] = OptAval(R,
T, RandFunc)

%     Preallocations
N = 1e6;

%     Event instants and states for each of the unit
InstantsUnit = cell(R,1);
StatesUnit   = cell(R,1);
TTF          = cell(R,1);
TTR          = cell(R,1);

%     Generation of random state transitions
for   Unit = 1:R %#ok<FXUP>

      %     Initialization
      InstantsUnit{Unit} = zeros(R,N);
      StatesUnit{Unit}   = ones(R,N);
      TTF{Unit}          = zeros(R,N);
      TTR{Unit}          = zeros(R,N);

      %     Current step and instant
      Step = 1;
      t    = 0;

      while t < T

            % Current instant
            t0 = InstantsUnit{Unit}(Step);

            % Current state
            State = StatesUnit{Unit}(Step);

            % Current random time to next event
            dt = RandFunc(Unit,t,State);
            % Transition
            switch State

                  % Failure: operational -> inoperational
                  case 1, s = 0;
```

```
                % Recovery: inoperational -> operational
                case 0, s = 1;

            end

            % Step counter incrementation
            Step = Step+1;

            % Time incrementation
            t = t0+dt;

            % Current instant and state
            InstantsUnit{Unit}(Step) = t;
            StatesUnit{Unit}(Step)   = s;
            Step2 = floor(Step/2);

            % TTF and TTR
            switch s
                case 0, TTF{Unit}(Step2) = dt;
                case 1, TTR{Unit}(Step2) = dt;
            end

        end

        %   Useful data
        InstantsUnit{Unit} = InstantsUnit{Unit}(1:Step);
        StatesUnit{Unit}   = StatesUnit{Unit}(1:Step);
        Step2              = floor(Step/2);
        TTF{Unit}          = TTF{Unit}(1:Step2);
        TTR{Unit}          = TTR{Unit}(1:Step2);

    end

%-----------------------------------------------------------------------
%   Concatenation of unitary simulation

%   Time vector
Time =
cat(2,cell2mat(arrayfun(@(Unit)InstantsUnit{Unit},1:R,'UniformOutput',fa
lse)));
Time = sort(Time);
Time = Time(Time<=T);

%   Time vector length
N = numel(Time);
```

```
%	StatesUnit matrix
States = ones(R,N);

%	Extraction of states for a commmon time vector
for	n = 2:N

	%	Memorization
	States(:,n) = States(:,n-1);

	%	Update
	for	Unit = 1:R %#ok<FXUP>
		nt = find(InstantsUnit{Unit} == Time(n),1,'first');
		if	nt
			States(Unit,n) = StatesUnit{Unit}(nt);
		end
	end

end

%	Instantaneous number of operational units
Number = sum(States,1);

%	Estimated probabilities
Durations = [diff(Time) 0];
T = Time(end);
Probabilities = arrayfun(@(p)sum(Durations(Number == p))/T,0:R);

%	TTF and TTR
TTF0 =
cat(2,cell2mat(arrayfun(@(Unit)TTF{Unit},1:R,'UniformOutput',false)));
TTR0 =
cat(2,cell2mat(arrayfun(@(Unit)TTR{Unit},1:R,'UniformOutput',false)));
[TTFf, Fttf] = ECDF(TTF0);
[TTRf, Fttr] = ECDF(TTR0);
MTTF = mean(TTF0);
MTTR = mean(TTR0);

%----------------------------------------------------------------------
%	Graphic representation
Representation()

	%	Graphic representation
	function Representation()
```

```
%       Figure
Figure = figure('Color','w');

%       Full screen
drawnow;
warning('off','all');
jFrame = get(Figure,'JavaFrame');
jFrame.setMaximized(true);
warning('on','all');
pause(0.1);

%       Title
axes('Position',[0 0.95 1 0.05],'Visible', 'off');
text(0.5,0.5,'Optimising System Accessibility using GA and
MC      Simulation','HorizontalAlignment','Center','Fontname',
        'Times','Fontsize',14,'Fontweight','Demi','FontAngle',
        'Normal');

%       States per unit
Axes(1) = subplot(3,2,[1 3]); hold('on');
Delta = 2*((1:R)-1);
Ytick = zeros(3*R,1);
Yticklabel = cell(3*R,1);
Ytick2 = zeros(2*R,1);
Yticklabel2 = cell(2*R,1);
for     Unit = 1:R %#ok<FXUP>
        if      Unit == 1
                [Axes2,Line1,~] =
                plotyy(Time,States(Unit,:)+Delta(Unit),nan,nan,'
                stairs','stairs');
                hold('on');
                set(Line1,'Color','b');
                set(Axes2,'Ycolor','k');
        else
                stairs(Axes2(1),Time,
                States(Unit,:)+Delta(Unit),'b-');
        end
        Ytick(3*(Unit-1)+1)  = 2*(Unit-1);
        Yticklabel{3*(Unit-1)+1} = 'KO';
        Ytick(3*(Unit-1)+2)  = 2*(Unit-1)+1/2;
        Yticklabel{3*(Unit-1)+2} = sprintf('Unit #%u  ',Unit);
        Ytick(3*(Unit-1)+3)  = 2*(Unit-1)+1;
```

```
        Yticklabel{3*(Unit-1)+3} = 'OK';
        p = sum(TTR{Unit})/T;
        Ytick2(2*(Unit-1)+1) = 2*(Unit-1);
        Yticklabel2{2*(Unit-1)+1} = sprintf('%.2f%%',100*p);
        Ytick2(2*(Unit-1)+2) = 2*(Unit-1)+1;
        Yticklabel2{2*(Unit-1)+2} = sprintf('%.2f%%',100*(1-
        p));
end
xlabel('Time [h]');
set(Axes(1),'Ytick',Ytick,'Yticklabel',Yticklabel,'ylim',[-1
2*R],'XMinorGrid','on');
box('on');
set(Axes(1),'Xgrid','on');
set(Axes2(2),'Ytick',Ytick2,'Yticklabel',Yticklabel2);
linkaxes(Axes2);

%      Number of operational units
Axes(2) = subplot(3,2,5); hold('on');
[Axes2,Line1,~] =
plotyy(Time,Number,nan,nan,@stairs,@stairs);
plot([Time(1) Time(end)],[0 0],'r--');
xlabel('Time [h]');
ylabel('Number of operational units');

%      Axes style
Labels1 =
@(Ytick)arrayfun(@(d)sprintf('%u',d),Ytick,'UniformOutput',f
alse);
Labels2 =
@(Ytick)arrayfun(@(d)sprintf('%2.2f%%',d),Ytick,'UniformOutp
ut',false);
set(Axes2(1),'YColor','K');
set(Line1,'Color','g');
set(Axes2(2),'YColor','k');
set(Axes(2),'Ytick',0:R,'Yticklabel',Labels1(0:R),'ylim',[-1
2*R],'XMinorGrid','on');
set(Axes2(2),'Ytick',0:R,'Yticklabel',Labels2(Probabilities*
100));
ylim([-1 R+1]);
box('on');
grid('on');
```

```
%     Axes link
linkaxes(Axes,'x');
linkaxes(Axes2,'y');

%     TTF and TTR CDF
Axes(3) = subplot(3,2,[2 4]);
[Axes2,Line1,Line2] =
plotyy(Fttf,TTFf,Fttr,TTRf,'stairs','stairs');
set(Axes2(1),'Ycolor','r');
set(Axes2(2),'Ycolor','b');
set(Line1,'Color','r');
set(Line2,'Color','b');
xlabel('Cumulative density function');
ylabel(Axes2(1),'Time to failure [h]');
ylabel(Axes2(2),'Time to recovery  [h]');
grid('on');
if     MTTF < 24,   FormatMTTF = 'HH:MM:SS';
else                FormatMTTF = 'DD HH:MM:SS';
end
if     MTTR < 24,   FormatMTTR = 'HH:MM:SS';
else                FormatMTTR = 'DD HH:MM:SS';
end
legend({sprintf('MTTF = %s',
      strrep(datestr(MTTF/24,FormatMTTF),' ','d ')),...
      sprintf('MTTR = %s',
      strrep(datestr(MTTR/24,FormatMTTR),' ','d '))},...
      'Location','NorthWest');
set(Axes(3),'YMinorGrid','on','YMinorTick','on');

%     Estimated probability
Axes(4) = subplot(3,2,6);
hold('on');
Colors = colormap('hsv');
for    Unit = 0:R %#ok<FXUP>
      Index = floor(26/64*Unit*(64-1)/R+1);
      Color = Colors(Index,:);
      barh(Unit,Probabilities(Unit+1),'Facecolor',Color);
end
xlabel('Estimated probability');
ylabel('Number of operational units');
grid('on');
box('on');
```

```
            set(Axes(4),'Xscale','log');

            %      Ordinates link
            linkaxes(Axes([2 4]),'y');

      end

end

%      Empirical cumulative distribution function
function [X,F] = ECDF(X)

x = sort(X);
N = numel(X);
F = zeros(N,1);
for    n = 1:N
      F(n)  = sum(x(n)>=X)/N;
end
X = x;

end
```

CHAPTER 6: DISCUSSION & INTERPRETATION OF RESULTS

6.1. Description of the Framework

The framework taken into account consists of $N_n = 3$ nodes. All these nodes are in series. Every single node is made up of $N_p(i_n) = 3, i_n = 1, 2, 3,$ branches of single components in parallel ($N_s(i_n, i_p) = 1;\ i_n, i_p = 1, 2, 3$). Components that belong to the similar node are supposed to be of the same type (thus, $N_t = N_n$) and therefore will be subjected to the identical maintenance strategy. Pictorial description of the abovementioned framework is shown in **Figure *5***.

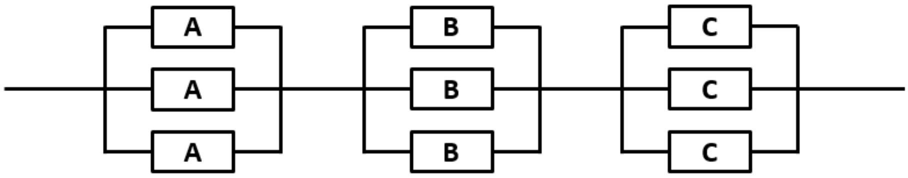

Figure 5. Pictorial description of framework considered in section 6.1.

Data that has been used for optimisation of the said framework is tabulated in **Table 1** below.

Serial	Data	Value
1.	Overall profit of the plant, P_t	500
2.	Monetary penalty for downtime, C_U	100
3.	Operation time, T_M	1000

Table 1. Arbitrary data of the described framework.

6.2. Case Study

In this case study, multi-objective genetic algorithm has been used for optimising the system mean accessibility, $'\bar{A}'$, and the profitability, $'G'$, over a given time $'T_M'$. During this process, following assumptions are made with regards to the failure probabilities of the components in the framework that are resulting into different deterioration stages:-

- The probabilities are not dependent on the load shared by the components; and,
- The number of maintenance professionals is enough to maintain the faulty components of the framework simultaneously.

These conditions facilitate the application of analytical modeling as well as the validation of suggested coupled approach (GA + MC) on the described framework which is essential for their comparison. The design variables for optimisation, i.e. the thresholds, $'k_{th}^{l}'$, after which the preventive maintenance should be carried out for all component types i.e. $l = 1, 2, 3,$ are assumed to be selected within the array $(1, 64)$. Parameters and rules set for the multi-objective genetic algorithm technique are mentioned in **Table 2**.

Serial	Data	Value
1.	No. of chromosomes (population size)	200
2.	No. of generations (stopping criteria)	200
3.	Selection procedure	Roulette wheel
4.	Replacement strategy	Children-parents
5.	Probability of mutation	1%
6.	Probability of crossover	100%
7.	Generations considered without elitist selection	100
8.	Parents' fraction picked with elitist selection	25%

Table 2. Parameters set for GA.

6.3. Results and Their Interpretation

According to the assumptions, the search space, required to be optimised, consisted of a large number of alternatives, i.e. 64^3. The same were considered and the objective functions were assessed by utilizing the Markov's model explained in Section 3.1.1 above. The results obtained after this analytical modelling of the framework are shown in **Figure *6*** (for the purpose of clarity, only corresponding objective functions' values with the multiples of 6 are shown). The processing time required for the said simulation on a Dell Inspiron N5110 Core-i5 was approximately 3 hours.

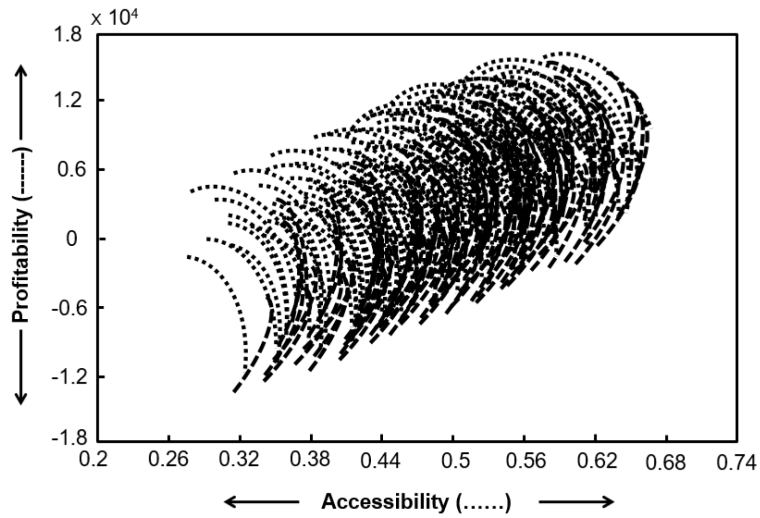

Figure 6. Graphical representation of objective functions' values evaluated by Markov's model.

The results obtained by adopting the coupled approach (GA + MC) are depicted in **Figure *7*** Processing time elapsed for the procedure on the same machine was approximately 1.5 min. The objective functions' values corresponding to the non-dominated solutions are illustrated in this figure.

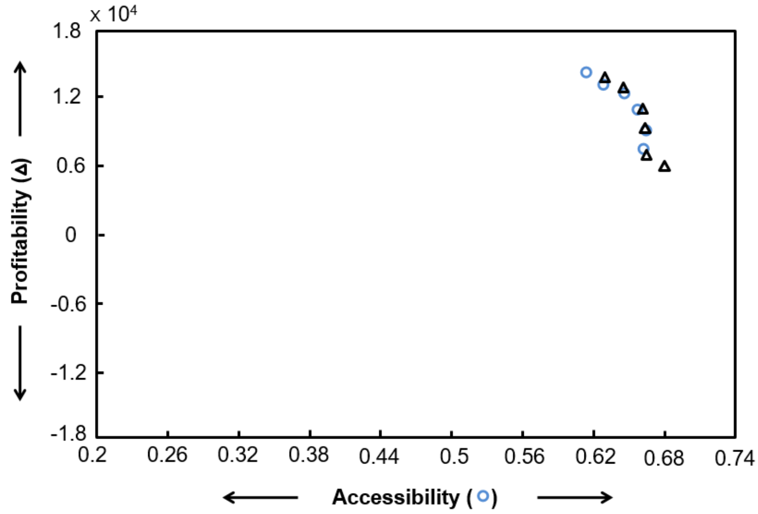

Figure 7. Graphical representation of objective functions' values evaluated by coupled approach.

It can be clearly seen that the optimal solutions lie on the boundary of Pareto domination front. The same can be observed by making a comparison of **Figure *6*** and **Figure *7*** above. The difference of processing times taken by the analytical modelling as compared to the suggested approach highlights the effectiveness of the later for taking the search towards the area of interest within the search space more rapidly. Objective functions' values for the non-dominated solutions achieved as a result of adopting the suggested coupled approach are enumerated in **Table 3** below.

Serial	**Accessibility, $\bar{A}$**	**Profitability, *G***
1.	0.6206	1.4 x 10^4
2.	0.6385	1.3 x 10^4
3.	0.6426	1.2 x 10^4
4.	0.6545	1.0 x 10^4
5.	0.6613	0.7 x 10^4
6.	0.6705	0.6 x 10^4

Table 3. Numerical values of objective functions for non-dominated solutions.

6.4. Graphical Solution to the Problem

Using the developed algorithm, graphical solution to the problem is also obtained. In this regard, effort has been made to present different dimensions of the solution. These dimensions range from combined accessibility to individual accessibility of the components. In addition, optimal values for mean times to failure and repair, and estimated probability of the operational components of the framework have also been illustrated through graphs. **Figure *8*** illustrates the combined percentage of accessibility based on operational components of the framework over a specified period of time. Using CBM strategy, all the components remained operational for 90.92% of the total time, whereas the framework was under maintenance for 9.08% of the time.

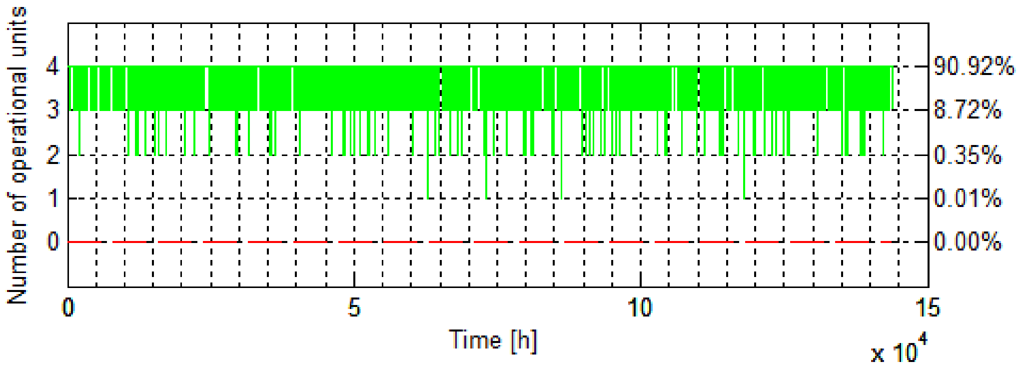

Figure 8. Combined accessibility of operational components over specified period of time.

Accessibility of individual components of the framework over the given time duration is shown in **Figure *9*** below. Individual percentage of accessibility for the components range from 97.54% to 97.84% and the individual downtime for the component range from 2.16% to 2.46%. It is pertinent to mention that these values are obtained for the framework being monitored under the conditions of condition-based maintenance using the coupled approach i.e. GA + MC.

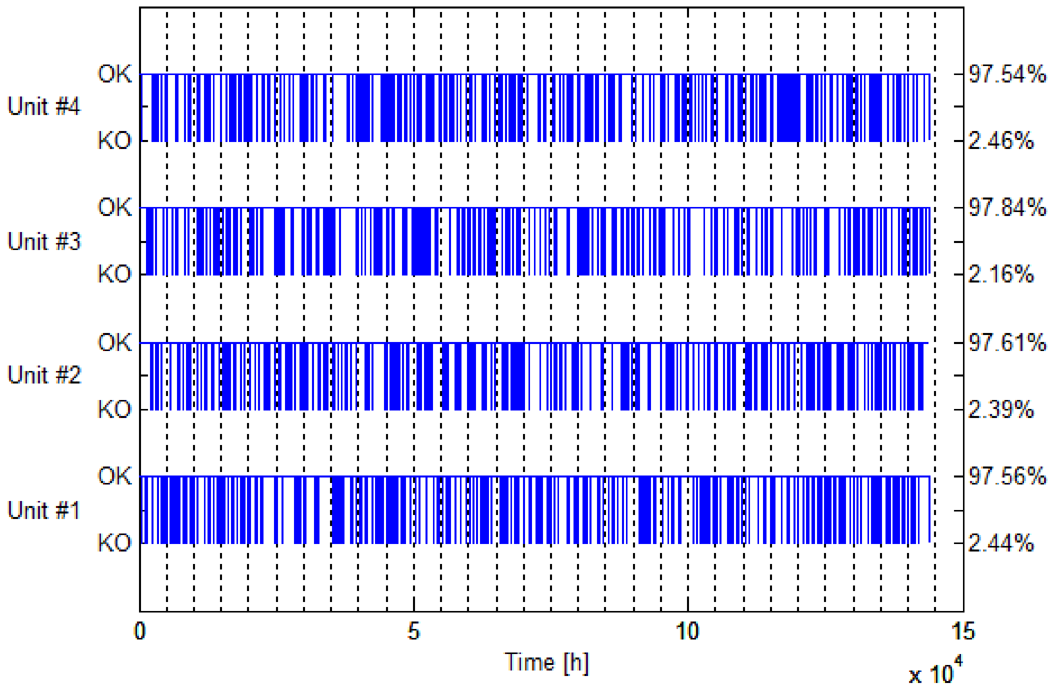

Figure 9. Accessibility of individual components over specified period of time.

Figure *10* depicts the optimal values for mean time to failure and mean time to repair, i.e. MTTF and MTTR, for the framework under CBM strategy. In this case, these values appears to be 478.6152 *hours* and 11.6775 *hours* respectively.

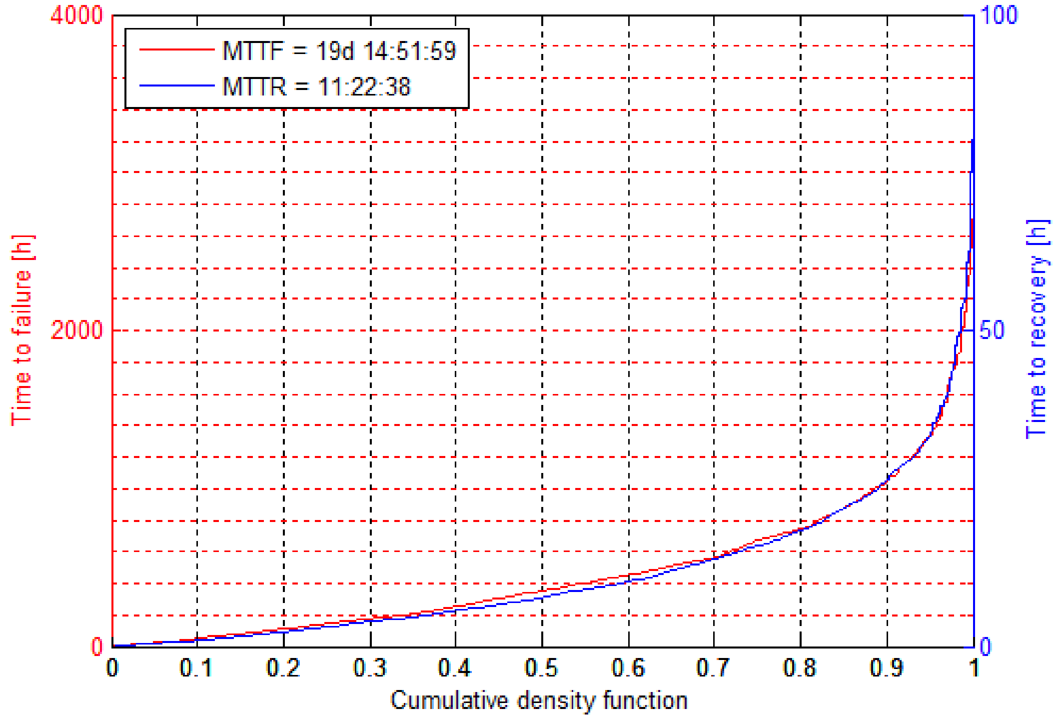

Figure 10. Optimal values of MTTF and MTTR for the framework under CBM strategy.

Estimated probability of operational components of the framework is illustrated in ***Figure 11*** below. This horizontal bar graph is actually the inverse of graph for combined accessibility as shown in ***Figure 8*** above. In this case, the estimated probability for all the components of the framework to be operational is $0.9301 \cong 10^0$.

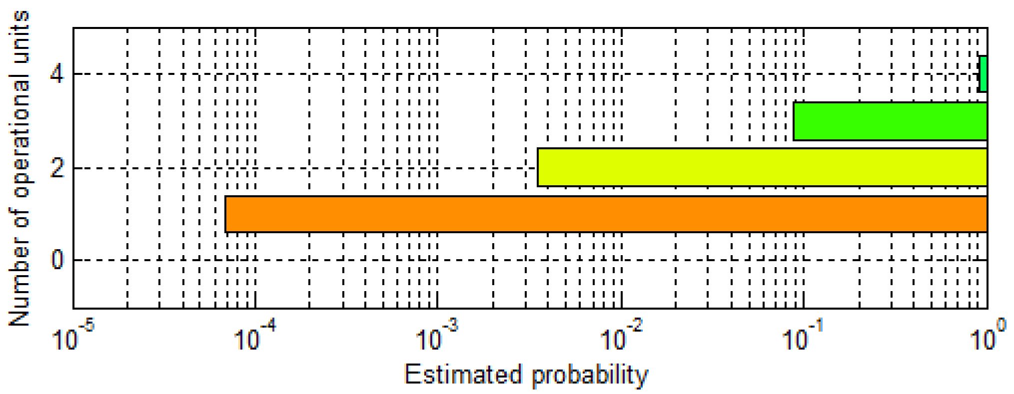

Figure 11. Estimated probability of operational components of the framework.

Computed results obtained by running the simulation on MATLAB are attached as **APPENDIX I** to this dissertation.

CHAPTER 7: THESIS SUMMARY, CONTRIBUTION AND CONCLUSION

This chapter summarizes the thesis, discusses its findings and contributions, points out limitations of the current work, and also outlines directions for future research. Optimisation of condition-based maintenance has been demonstrated in this dissertation with the help of a proposed model based on Genetic Algorithms and Monte Carlo simulation. However, still many extensions of this research deserve further consideration.

The chapter is divided into four sections. Section 7.1 is a summary of the thesis. Section 7.2 presents a discussion of the contribution and limitations of the current work. Section 7.3 discusses the future work, and finally Section 7.4 brings the thesis to a conclusion.

7.1. Summary of the Thesis

This thesis has introduced a model for optimising the condition-based maintenance of a deteriorating system working under stressed environment. The proposed synergic model (based on GA & MC simulation has proved to be more effective due to the employment of 'drop-by-drop approach' that permits the drive of related search process with regards to the best optimal solutions. This model has not only proved to be successful in contributing towards computational resources for searching the optimal solutions but also show remarkable time saving as compared to the contemporary analytical modelling procedures.

Chapter 1 covered the background of the problem, followed by problem formulation, main objective, methodology adopted and the delimitations. The need of building up a productive CBM policy during design stage of an engineering system is effectively highlighted in this chapter. In addition, it also defines two main objectives, i.e. profitability and accessibility, which are required to be optimised while planning the maintenance policy of an organization.

Chapter 2 reviewed the literature on the subject. In this chapter, some important terms and definitions regarding maintenance are also covered. In addition, condition monitoring techniques for CBM, data collection / analysis tools and conventional / evolutionary optimisation techniques are also explained. This chapter helps in building up the basic concept of the work and acts as a foundation for the proposed model.

Chapter 3 discussed the analytical modelling procedure on the subject. In this chapter, Markov's model for condition-based maintenance has been explained in detail and relevant equations depicting the system have been drawn for comparison of the same with the proposed model at later stages. In addition, prediction of system's accessibility within a given timeframe with the help of MC simulation has also been explained in the same chapter. It is pertinent to highlight that the simulation results acquired by MC simulation conform to the ones achieved with the Markov's Process that is considered to be a more traditional technique. Towards the end of this chapter, efforts have been made to optimise the solution using GA. In this regard, the role of non-dominated solutions has also been discussed in detail for in-depth comprehension of the subject.

In Chapter 4, the proposed framework has been described. Moreover, reasons for using the synergic approach and the pseudo code for the proposed framework has also been given in the same chapter. The proposed framework helped us to reach the optimised solution in a time efficient way based on non-dominated solutions only.

Chapter 5 described the detailed procedure of model development based on proposed framework in MATLAB. It starts with defining the parameters, deciding the required outputs and ends at writing the algorithm based on the defined variables and the proposed framework. In the algorithm, detailed help module is also included for easy comprehension.

In Chapter 6, validation of the developed model through a case study has been done followed by discussion and interpretation of results. In this chapter, initially a simple problem is formulated which is then solved under defined conditions for GA and MC simulation. The results so obtained are discussed in detail. These results include numerical as well as graphical solution to the problem.

7.2. Contribution of the Thesis

In this thesis, a framework for optimisation of CBM policy has been proposed. For validation of the said framework, a model has been developed in MATLAB and the same has been used to optimise the parameters of an engineering system. The obtained results have been compared with the ones acquired through analytical modelling of the system. These results show remarkable time saving with the proposed algorithm. Keeping in view the achieved results, the proposed model has proved to be successful in contributing towards computational resources for searching the optimal solutions.

7.3. Future Work

While this thesis has demonstrated the effectiveness of the proposed algorithm, many opportunities for extending the scope of this thesis still remain open. In the next phase, efforts may be made to consider the problems associated with stress-dependent deterioration processes of the components of an engineering system while taking into account the limited quantity of maintenance professionals present on site.

7.4. Conclusion

The recognition of effective strategies, for repair / maintenance of plants / engineering systems, is of significant importance from financial as well as safety point of view. Particularly, the benefits of carrying out condition-based maintenance have become more critical nowadays. Actually, the related phenomenon behind deterioration processes is very complex; especially when higher-risk frameworks, for instance atomic power plants, offshore establishments, etc., are concerned. In such cases, comprehensive simulation of the plant performance is very much essential, which can only be carried out analytically under simplified hypothesis.

This research framework is focused on the issue of finding an optimal condition-based maintenance strategy based on threshold levels of components deterioration beyond which maintenance is necessitated.

In order to deal with the issue, the problem is configured as a multiple objective search, to be dealt by a synergic approach, encompassing both GA as well as MC framework. The optimised level of deterioration for each element is explored while keeping in view the mean accessibility and net profit of the system. A population of chromosomes is considered by GA, each encoding an optimum level of deterioration for every specified element. For any particular chromosome, MC simulation approximates the two desired objective functions.

Through validation of the proposed framework, it has been proved that the obtained results are compatible with the ones acquired through analytical modelling of the system. These results not only help in achieving the remarkable time saving with the proposed algorithm but also the proposed model has proved to be successful in contributing towards computational resources for searching the optimal solutions.

APPENDIX I

COMPUTED RESULTS OBTAINED BY RUNNING SIMULATION ON MATLAB

Redundancy	=	3
MTTF	=	480
MTTR	=	12
Duration	=	144000
Time	=	1.0e+005 *

Colm 1 to 10

0	0	0	0.0051	0.0052	0.0063	0.0063	0.0074	0.0075	0.0080

Colm 11 to 20

0.0080	0.0093	0.0094	0.0097	0.0101	0.0102	0.0103	0.0131	0.0134	0.0145

Colm 21 to 30

0.0146	0.0178	0.0179	0.0225	0.0225	0.0226	0.0228	0.0259	0.0259	0.0261

Colm 31 to 40

0.0262	0.0264	0.0265	0.0273	0.0273	0.0274	0.0274	0.0283	0.0283	0.0286

Colm 41 to 50

0.0288	0.0318	0.0319	0.0321	0.0322	0.0333	0.0333	0.0334	0.0338	0.0351

Colm 51 to 60

0.0352	0.0378	0.0379	0.0379	0.0379	0.0395	0.0396	0.0413	0.0413	0.0417

Colm 61 to 70

0.0421	0.0431	0.0433	0.0442	0.0442	0.0505	0.0506	0.0507	0.0510	0.0511

Colm 71 to 80

0.0512	0.0520	0.0523	0.0570	0.0571	0.0586	0.0586	0.0587	0.0587	0.0604

Colm 81 to 90

0.0605	0.0632	0.0632	0.0633	0.0634	0.0643	0.0643	0.0643	0.0643	0.0644

Colm 91 to 100

0.0645	0.0656	0.0656	0.0680	0.0681	0.0689	0.0689	0.0723	0.0728	0.0728

Colm 101 to 110

0.0728	0.0734	0.0735	0.0737	0.0738	0.0759	0.0760	0.0765	0.0767	0.0782

Colm 111 to 120

0.0782	0.0798	0.0798	0.0813	0.0813	0.0837	0.0838	0.0851	0.0852	0.0863

Colm 121 to 130

0.0863 0.0865 0.0867 0.0870 0.0870 0.0916 0.0918 0.0928 0.0928 0.0953

Colm 131 to 140

0.0954 0.0959 0.0959 0.1005 0.1006 0.1014 0.1016 0.1037 0.1038 0.1040

Colm 141 to 150

0.1045 0.1087 0.1089 0.1092 0.1093 0.1112 0.1113 0.1132 0.1133 0.1139

Colm 151 to 160

0.1139 0.1161 0.1162 0.1203 0.1204 0.1238 0.1238 0.1245 0.1247 0.1258

Colm 161 to 170

0.1258 0.1344 0.1344 0.1349 0.1351 0.1389 0.1389 0.1409 0.1410 0.1419

Colm 171 to 180

0.1419 0.1434 0.1434 0.1437 0.1437 0.1496 0.1499 0.1501 0.1501 0.1506

Colm 181 to 190

0.1506 0.1535 0.1537 0.1576 0.1576 0.1584 0.1584 0.1585 0.1586 0.1586

Colm 191 to 200

0.1587 0.1598 0.1598 0.1626 0.1626 0.1636 0.1636 0.1645 0.1646 0.1648

Colm 201 to 210

0.1649 0.1666 0.1669 0.1687 0.1687 0.1700 0.1700 0.1717 0.1719 0.1765

Colm 211 to 220

0.1767 0.1778 0.1779 0.1798 0.1798 0.1813 0.1813 0.1833 0.1833 0.1839

Colm 221 to 230

0.1839 0.1900 0.1901 0.1912 0.1912 0.1934 0.1935 0.1945 0.1946 0.1961

Colm 231 to 240

0.1961 0.1966 0.1969 0.1995 0.1996 0.2024 0.2025 0.2028 0.2029 0.2042

Colm 241 to 250

0.2042 0.2059 0.2059 0.2060 0.2063 0.2066 0.2066 0.2073 0.2075 0.2083

Colm 251 to 260

0.2084 0.2086 0.2088 0.2134 0.2135 0.2143 0.2149 0.2157 0.2158 0.2181

Colm 261 to 270

0.2182 0.2187 0.2188 0.2192 0.2193 0.2194 0.2195 0.2233 0.2234 0.2239

Colm 271 to 280

0.2239 0.2252 0.2253 0.2269 0.2270 0.2286 0.2287 0.2326 0.2326 0.2331

Colm 281 to 290

0.2331 0.2340 0.2342 0.2345 0.2345 0.2381 0.2381 0.2423 0.2424 0.2481

Colm 291 to 300

0.2486 0.2516 0.2517 0.2543 0.2546 0.2596 0.2597 0.2602 0.2602 0.2615

Colm 301 to 310

0.2616 0.2651 0.2652 0.2653 0.2653 0.2680 0.2681 0.2727 0.2728 0.2743

Colm 311 to 320

0.2743 0.2750 0.2752 0.2754 0.2754 0.2792 0.2793 0.2802 0.2802 0.2827

Colm 321 to 330

0.2828 0.2842 0.2842 0.2844 0.2844 0.2877 0.2877 0.2888 0.2888 0.2896

Colm 331 to 340

0.2899 0.2905 0.2906 0.2909 0.2910 0.2910 0.2910 0.2916 0.2917 0.2923

Colm 341 to 350

0.2926 0.2952 0.2953 0.2955 0.2957 0.2961 0.2961 0.2974 0.2974 0.3041

Colm 351 to 360

0.3041 0.3048 0.3048 0.3130 0.3132 0.3143 0.3144 0.3145 0.3147 0.3168

Colm 361 to 370

0.3169 0.3177 0.3178 0.3195 0.3197 0.3213 0.3215 0.3226 0.3226 0.3245

Colm 371 to 380

0.3246 0.3246 0.3248 0.3277 0.3277 0.3282 0.3284 0.3292 0.3295 0.3300

Colm 381 to 390

0.3301 0.3304 0.3304 0.3313 0.3314 0.3344 0.3345 0.3375 0.3377 0.3400

Colm 391 to 400

0.3401 0.3418 0.3418 0.3425 0.3425 0.3438 0.3439 0.3456 0.3456 0.3457

Colm 401 to 410

0.3457 0.3467 0.3468 0.3469 0.3469 0.3530 0.3531 0.3552 0.3553 0.3553

Colm 411 to 420

0.3554 0.3581 0.3581 0.3591 0.3593 0.3610 0.3612 0.3643 0.3647 0.3681

Colm 421 to 430

0.3682 0.3704 0.3704 0.3719 0.3721 0.3724 0.3725 0.3748 0.3749 0.3790

Colm 431 to 440

0.3790 0.3819 0.3819 0.3824 0.3824 0.3877 0.3877 0.3888 0.3890 0.3906

Colm 441 to 450

0.3909 0.3926 0.3926 0.3944 0.3946 0.3960 0.3966 0.4023 0.4023 0.4028

Colm 451 to 460

0.4030 0.4031 0.4031 0.4037 0.4039 0.4061 0.4062 0.4075 0.4078 0.4084

Colm 461 to 470

0.4084	0.4085	0.4085	0.4128	0.4128	0.4131	0.4134	0.4155	0.4156	0.4174

Colm 471 to 480

0.4176	0.4176	0.4177	0.4201	0.4201	0.4214	0.4214	0.4218	0.4220	0.4241

Colm 481 to 490

0.4241	0.4262	0.4263	0.4280	0.4282	0.4282	0.4283	0.4309	0.4310	0.4324

Colm 491 to 500

0.4324	0.4357	0.4358	0.4386	0.4387	0.4423	0.4423	0.4437	0.4439	0.4467

Colm 501 to 510

0.4467	0.4482	0.4483	0.4493	0.4495	0.4505	0.4506	0.4508	0.4508	0.4523

Colm 511 to 520

0.4523	0.4541	0.4541	0.4584	0.4584	0.4594	0.4597	0.4600	0.4600	0.4623

Colm 521 to 530

0.4626	0.4650	0.4651	0.4653	0.4654	0.4669	0.4670	0.4691	0.4692	0.4694

Colm 531 to 540

0.4694	0.4721	0.4721	0.4753	0.4754	0.4767	0.4768	0.4779	0.4779	0.4789

Colm 541 to 550

0.4791	0.4797	0.4799	0.4820	0.4821	0.4834	0.4835	0.4837	0.4837	0.4842

Colm 551 to 560

0.4844	0.4846	0.4848	0.4873	0.4875	0.4878	0.4879	0.4884	0.4884	0.4884

Colm 561 to 570

0.4885	0.4895	0.4898	0.4906	0.4908	0.4913	0.4914	0.4946	0.4948	0.4967

Colm 571 to 580

0.4970	0.4991	0.4991	0.5046	0.5046	0.5071	0.5072	0.5076	0.5076	0.5079

Colm 581 to 590

0.5080	0.5093	0.5094	0.5111	0.5113	0.5149	0.5149	0.5177	0.5180	0.5201

Colm 591 to 600

0.5203	0.5210	0.5211	0.5233	0.5233	0.5296	0.5296	0.5296	0.5298	0.5312

Colm 601 to 610

0.5313	0.5323	0.5324	0.5344	0.5346	0.5349	0.5349	0.5388	0.5389	0.5396

Colm 611 to 620

0.5396	0.5421	0.5423	0.5431	0.5432	0.5447	0.5449	0.5458	0.5458	0.5458

Colm 621 to 630

0.5459	0.5460	0.5462	0.5473	0.5476	0.5479	0.5479	0.5481	0.5483	0.5492

Colm 631 to 640

0.5493 0.5495 0.5496 0.5528 0.5529 0.5540 0.5543 0.5559 0.5562 0.5569

Colm 641 to 650

0.5575 0.5625 0.5626 0.5631 0.5631 0.5635 0.5636 0.5646 0.5646 0.5658

Colm 651 to 660

0.5660 0.5665 0.5666 0.5666 0.5666 0.5685 0.5686 0.5736 0.5737 0.5805

Colm 661 to 670

0.5805 0.5824 0.5825 0.5852 0.5852 0.5877 0.5882 0.5912 0.5917 0.5950

Colm 671 to 680

0.5956 0.5993 0.5993 0.6006 0.6007 0.6013 0.6014 0.6040 0.6040 0.6043

Colm 681 to 690

0.6044 0.6046 0.6047 0.6047 0.6048 0.6067 0.6067 0.6087 0.6089 0.6113

Colm 691 to 700

0.6118 0.6129 0.6129 0.6130 0.6131 0.6137 0.6138 0.6152 0.6153 0.6155

Colm 701 to 710

0.6157 0.6160 0.6161 0.6168 0.6168 0.6210 0.6212 0.6225 0.6228 0.6232

Colm 711 to 720

0.6233 0.6235 0.6237 0.6243 0.6246 0.6252 0.6252 0.6266 0.6266 0.6269

Colm 721 to 730

0.6269 0.6287 0.6290 0.6363 0.6363 0.6366 0.6367 0.6372 0.6372 0.6373

Colm 731 to 740

0.6375 0.6398 0.6398 0.6436 0.6437 0.6446 0.6448 0.6459 0.6459 0.6476

Colm 741 to 750

0.6479 0.6486 0.6490 0.6509 0.6509 0.6532 0.6533 0.6543 0.6545 0.6553

Colm 751 to 760

0.6553 0.6612 0.6619 0.6633 0.6636 0.6643 0.6643 0.6643 0.6644 0.6644

Colm 761 to 770

0.6646 0.6655 0.6657 0.6662 0.6664 0.6677 0.6677 0.6685 0.6685 0.6694

Colm 771 to 780

0.6694 0.6712 0.6713 0.6714 0.6715 0.6726 0.6727 0.6727 0.6729 0.6741

Colm 781 to 790

0.6741 0.6757 0.6759 0.6771 0.6771 0.6787 0.6787 0.6790 0.6790 0.6821

Colm 791 to 800

0.6822 0.6823 0.6824 0.6825 0.6828 0.6870 0.6871 0.6885 0.6885 0.6887

Colm 801 to 810

0.6889	0.6891	0.6892	0.6906	0.6913	0.6921	0.6922	0.6922	0.6923	0.6928

Colm 811 to 820

0.6929	0.6960	0.6960	0.6966	0.6967	0.6971	0.6973	0.6997	0.6997	0.7011

Colm 821 to 830

0.7013	0.7071	0.7073	0.7080	0.7082	0.7099	0.7101	0.7122	0.7122	0.7134

Colm 831 to 840

0.7135	0.7162	0.7163	0.7188	0.7189	0.7252	0.7252	0.7269	0.7270	0.7282

Colm 841 to 850

0.7282	0.7283	0.7285	0.7301	0.7302	0.7317	0.7318	0.7333	0.7334	0.7348

Colm 851 to 860

0.7348	0.7358	0.7359	0.7377	0.7377	0.7382	0.7383	0.7413	0.7417	0.7435

Colm 861 to 870

0.7436	0.7454	0.7459	0.7481	0.7481	0.7499	0.7500	0.7506	0.7506	0.7512

Colm 871 to 880

0.7512	0.7516	0.7517	0.7563	0.7563	0.7570	0.7573	0.7594	0.7595	0.7616

Colm 881 to 890

0.7616	0.7658	0.7659	0.7681	0.7681	0.7685	0.7685	0.7686	0.7686	0.7689

Colm 891 to 900

0.7690	0.7698	0.7700	0.7704	0.7705	0.7708	0.7709	0.7711	0.7712	0.7713

Colm 901 to 910

0.7714	0.7719	0.7720	0.7721	0.7723	0.7728	0.7730	0.7747	0.7748	0.7805

Colm 911 to 920

0.7806	0.7821	0.7821	0.7836	0.7836	0.7836	0.7837	0.7862	0.7862	0.7862

Colm 921 to 930

0.7864	0.7865	0.7866	0.7870	0.7871	0.7948	0.7952	0.7964	0.7965	0.7967

Colm 931 to 940

0.7967	0.7971	0.7971	0.7980	0.7981	0.7997	0.7998	0.8003	0.8004	0.8010

Colm 941 to 950

0.8012	0.8033	0.8034	0.8060	0.8061	0.8061	0.8062	0.8064	0.8066	0.8074

Colm 951 to 960

0.8076	0.8076	0.8076	0.8079	0.8081	0.8084	0.8086	0.8087	0.8090	0.8151

Colm 961 to 970

0.8152	0.8185	0.8185	0.8221	0.8222	0.8257	0.8258	0.8266	0.8267	0.8269

Colm 971 to 980

0.8269	0.8272	0.8272	0.8279	0.8282	0.8298	0.8298	0.8317	0.8318	0.8321

Colm 981 to 990

0.8323	0.8331	0.8333	0.8337	0.8339	0.8340	0.8342	0.8365	0.8366	0.8400

Colm 991 to 1000

0.8400	0.8433	0.8433	0.8434	0.8436	0.8469	0.8469	0.8472	0.8474	0.8480

Colm 1001 to 1010

0.8481	0.8486	0.8486	0.8488	0.8488	0.8520	0.8520	0.8528	0.8528	0.8537

Colm 1011 to 1020

0.8537	0.8559	0.8561	0.8604	0.8605	0.8610	0.8610	0.8616	0.8617	0.8627

Colm 1021 to 1030

0.8628	0.8631	0.8632	0.8632	0.8635	0.8645	0.8646	0.8647	0.8650	0.8655

Colm 1031 to 1040

0.8656	0.8661	0.8661	0.8666	0.8666	0.8678	0.8678	0.8685	0.8688	0.8688

Colm 1041 to 1050

0.8689	0.8697	0.8697	0.8706	0.8706	0.8712	0.8713	0.8735	0.8739	0.8739

Colm 1051 to 1060

0.8741	0.8760	0.8761	0.8769	0.8769	0.8770	0.8771	0.8809	0.8809	0.8843

Colm 1061 to 1070

0.8843	0.8847	0.8847	0.8880	0.8881	0.8886	0.8887	0.8890	0.8891	0.8894

Colm 1071 to 1080

0.8895	0.8896	0.8896	0.8935	0.8935	0.8936	0.8936	0.8938	0.8939	0.8943

Colm 1081 to 1090

0.8943	0.8946	0.8950	0.9021	0.9023	0.9032	0.9032	0.9045	0.9047	0.9061

Colm 1091 to 1100

0.9061	0.9072	0.9072	0.9102	0.9102	0.9110	0.9110	0.9110	0.9111	0.9162

Colm 1101 to 1110

0.9164	0.9171	0.9172	0.9174	0.9176	0.9184	0.9184	0.9207	0.9210	0.9221

Colm 1111 to 1120

0.9225	0.9276	0.9277	0.9277	0.9279	0.9279	0.9280	0.9340	0.9340	0.9348

Colm 1121 to 1130

0.9348	0.9348	0.9349	0.9378	0.9379	0.9392	0.9392	0.9400	0.9401	0.9420

Colm 1131 to 1140

0.9420	0.9426	0.9426	0.9427	0.9430	0.9433	0.9436	0.9442	0.9444	0.9447

Colm 1141 to 1150

0.9448 0.9460 0.9460 0.9479 0.9479 0.9483 0.9484 0.9493 0.9493 0.9507

Colm 1151 to 1160

0.9507 0.9516 0.9519 0.9535 0.9535 0.9554 0.9555 0.9580 0.9582 0.9601

Colm 1161 to 1170

0.9603 0.9604 0.9604 0.9609 0.9610 0.9617 0.9617 0.9652 0.9652 0.9670

Colm 1171 to 1180

0.9671 0.9679 0.9679 0.9686 0.9688 0.9713 0.9714 0.9714 0.9716 0.9724

Colm 1181 to 1190

0.9726 0.9743 0.9744 0.9811 0.9811 0.9843 0.9844 0.9855 0.9857 0.9867

Colm 1191 to 1200

0.9870 0.9873 0.9877 0.9885 0.9888 0.9888 0.9889 0.9890 0.9890 0.9894

Colm 1201 to 1210

0.9896 0.9920 0.9921 0.9932 0.9933 0.9941 0.9945 0.9972 0.9973 0.9986

Colm 1211 to 1220

0.9986 0.9990 0.9991 1.0003 1.0004 1.0007 1.0007 1.0030 1.0035 1.0081

Colm 1221 to 1230

1.0082 1.0087 1.0087 1.0091 1.0094 1.0110 1.0111 1.0136 1.0139 1.0150

Colm 1231 to 1240

1.0153 1.0165 1.0166 1.0168 1.0168 1.0173 1.0173 1.0183 1.0184 1.0193

Colm 1241 to 1250

1.0194 1.0207 1.0208 1.0248 1.0250 1.0317 1.0318 1.0318 1.0323 1.0335

Colm 1251 to 1260

1.0335 1.0337 1.0342 1.0343 1.0343 1.0354 1.0354 1.0358 1.0360 1.0440

Colm 1261 to 1270

1.0442 1.0453 1.0455 1.0458 1.0461 1.0484 1.0484 1.0491 1.0491 1.0501

Colm 1271 to 1280

1.0505 1.0538 1.0539 1.0570 1.0572 1.0581 1.0586 1.0607 1.0607 1.0614

Colm 1281 to 1290

1.0614 1.0640 1.0640 1.0650 1.0651 1.0654 1.0655 1.0655 1.0656 1.0660

Colm 1291 to 1300

1.0661 1.0675 1.0678 1.0690 1.0690 1.0691 1.0691 1.0693 1.0694 1.0707

Colm 1301 to 1310

1.0708 1.0710 1.0710 1.0720 1.0722 1.0741 1.0743 1.0744 1.0744 1.0775

Colm 1311 to 1320

1.0780 1.0822 1.0822 1.0826 1.0829 1.0829 1.0830 1.0856 1.0858 1.0868

Colm 1321 to 1330

1.0870 1.0886 1.0886 1.0894 1.0894 1.0916 1.0917 1.0948 1.0949 1.0955

Colm 1331 to 1340

1.0962 1.0963 1.0963 1.0964 1.0968 1.0974 1.0974 1.0977 1.0978 1.0979

Colm 1341 to 1350

1.0981 1.0987 1.0989 1.0993 1.0993 1.0995 1.0997 1.1000 1.1000 1.1007

Colm 1351 to 1360

1.1007 1.1020 1.1020 1.1021 1.1024 1.1046 1.1047 1.1065 1.1066 1.1076

Colm 1361 to 1370

1.1076 1.1077 1.1078 1.1090 1.1092 1.1116 1.1117 1.1128 1.1132 1.1154

Colm 1371 to 1380

1.1156 1.1157 1.1157 1.1180 1.1181 1.1199 1.1199 1.1216 1.1216 1.1289

Colm 1381 to 1390

1.1289 1.1296 1.1296 1.1301 1.1303 1.1322 1.1322 1.1325 1.1325 1.1350

Colm 1391 to 1400

1.1351 1.1372 1.1372 1.1372 1.1373 1.1373 1.1375 1.1389 1.1389 1.1394

Colm 1401 to 1410

1.1398 1.1416 1.1416 1.1431 1.1431 1.1433 1.1433 1.1482 1.1484 1.1494

Colm 1411 to 1420

1.1495 1.1500 1.1501 1.1511 1.1511 1.1514 1.1515 1.1531 1.1533 1.1621

Colm 1421 to 1430

1.1621 1.1657 1.1657 1.1688 1.1689 1.1707 1.1708 1.1715 1.1716 1.1716

Colm 1431 to 1440

1.1717 1.1730 1.1730 1.1746 1.1746 1.1750 1.1750 1.1809 1.1813 1.1813

Colm 1441 to 1450

1.1813 1.1814 1.1814 1.1845 1.1846 1.1847 1.1850 1.1862 1.1862 1.1888

Colm 1451 to 1460

1.1889 1.1890 1.1891 1.1895 1.1896 1.1901 1.1902 1.1904 1.1904 1.1911

Colm 1461 to 1470

1.1911 1.1920 1.1922 1.1934 1.1939 1.1968 1.1968 1.1993 1.1993 1.2004

Colm 1471 to 1480

1.2004 1.2033 1.2034 1.2057 1.2058 1.2077 1.2077 1.2088 1.2088 1.2091

Colm 1481 to 1490

1.2091	1.2109	1.2110	1.2110	1.2110	1.2144	1.2145	1.2172	1.2174	1.2210

Colm 1491 to 1500

1.2211	1.2241	1.2242	1.2245	1.2246	1.2251	1.2254	1.2273	1.2274	1.2306

Colm 1501 to 1510

1.2307	1.2322	1.2323	1.2329	1.2332	1.2343	1.2343	1.2360	1.2362	1.2371

Colm 1511 to 1520

1.2372	1.2390	1.2390	1.2451	1.2453	1.2492	1.2493	1.2498	1.2499	1.2517

Colm 1521 to 1530

1.2518	1.2528	1.2529	1.2565	1.2566	1.2587	1.2587	1.2602	1.2602	1.2606

Colm 1531 to 1540

1.2606	1.2669	1.2669	1.2671	1.2672	1.2691	1.2692	1.2718	1.2719	1.2726

Colm 1541 to 1550

1.2728	1.2742	1.2744	1.2794	1.2795	1.2812	1.2816	1.2829	1.2830	1.2865

Colm 1551 to 1560

1.2866	1.2894	1.2895	1.2929	1.2929	1.2939	1.2940	1.2957	1.2958	1.2959

Colm 1561 to 1570

1.2961	1.2967	1.2969	1.2977	1.2978	1.2985	1.2985	1.2992	1.2994	1.3020

Colm 1571 to 1580

1.3021	1.3023	1.3023	1.3060	1.3060	1.3060	1.3060	1.3068	1.3069	1.3100

Colm 1581 to 1590

1.3102	1.3111	1.3112	1.3121	1.3121	1.3144	1.3144	1.3158	1.3158	1.3164

Colm 1591 to 1600

1.3165	1.3170	1.3171	1.3173	1.3173	1.3201	1.3203	1.3219	1.3221	1.3256

Colm 1601 to 1610

1.3256	1.3260	1.3263	1.3288	1.3288	1.3290	1.3291	1.3302	1.3304	1.3327

Colm 1611 to 1620

1.3328	1.3332	1.3333	1.3333	1.3334	1.3346	1.3347	1.3373	1.3373	1.3373

Colm 1621 to 1630

1.3374	1.3375	1.3376	1.3389	1.3392	1.3394	1.3394	1.3399	1.3400	1.3430

Colm 1631 to 1640

1.3431	1.3450	1.3452	1.3458	1.3458	1.3469	1.3470	1.3476	1.3477	1.3504

Colm 1641 to 1650

1.3505	1.3505	1.3511	1.3516	1.3516	1.3522	1.3523	1.3564	1.3564	1.3565

Colm 1651 to 1660

1.3568	1.3573	1.3577	1.3603	1.3603	1.3608	1.3610	1.3617	1.3617	1.3620

Colm 1661 to 1670

1.3621	1.3625	1.3626	1.3638	1.3639	1.3642	1.3642	1.3675	1.3675	1.3676

Colm 1671 to 1680

1.3677	1.3689	1.3690	1.3703	1.3704	1.3736	1.3737	1.3750	1.3750	1.3766

Colm 1681 to 1690

1.3768	1.3782	1.3783	1.3785	1.3785	1.3787	1.3787	1.3798	1.3801	1.3810

Colm 1691 to 1700

1.3813	1.3836	1.3836	1.3841	1.3842	1.3858	1.3858	1.3865	1.3866	1.3879

Colm 1701 to 1710

1.3879	1.3899	1.3899	1.3902	1.3902	1.3905	1.3906	1.3944	1.3944	1.3946

Colm 1711 to 1720

1.3947	1.3965	1.3965	1.3994	1.3995	1.4000	1.4000	1.4064	1.4064	1.4074

Colm 1721 to 1730

1.4075	1.4094	1.4095	1.4095	1.4096	1.4121	1.4121	1.4135	1.4139	1.4153

Colm 1731 to 1740

1.4153	1.4170	1.4170	1.4179	1.4180	1.4213	1.4214	1.4226	1.4227	1.4227

Colm 1741 to 1750

1.4231	1.4243	1.4243	1.4244	1.4245	1.4261	1.4264	1.4305	1.4306	1.4331

Colm 1751 to 1760

1.4331	1.4331	1.4332	1.4343	1.4348	1.4357	1.4359	1.4365	1.4365	1.4381

Colm 1761 to 1765

1.4382	1.4389	1.4389	1.4398	1.4400

Number =

Colm 1 to 22

3	3	3	2	3	2	3	2	3	2	3	2	3	2	1	2	3	2	3	2	3	2

Colm 23 to 44

3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2

Colm 45 to 66

3	2	3	2	3	2	3	2	1	2	3	2	3	2	3	2	3	2	3	2	3	2

Colm 67 to 88

3	2	3	2	3	2	3	2	3	2	3	2	3	2	3	2	1	2	3	2	1	2

Colm 89 to 110

1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 111 to 132

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 133 to 154

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 155 to 176

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 177 to 198

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 199 to 220

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 221 to 242

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 243 to 264

1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 265 to 286

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 287 to 308

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 309 to 330

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 331 to 352

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 353 to 374

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 375 to 396

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 397 to 418

3 2 3 2 3 2 1 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2

Colm 419 to 440

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 441 to 462

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 463 to 484

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 485 to 506

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 507 to 528

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 529 to 550

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2

Colm 551 to 572

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 573 to 594

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 595 to 616

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 617 to 638

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 639 to 660

3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2

Colm 661 to 682

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2

Colm 683 to 704

1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 705 to 726

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 727 to 748

3 2 1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 749 to 759

3 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2

Colm 771 to 792

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2

Colm 793 to 814

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 815 to 836

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 837 to 858

3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 859 to 880

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 881 to 902

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 903 to 924

3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2 1 2 3 2 3 2

Colm 925 to 946

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 947 to 968

3 2 3 2 1 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2

Colm 969 to 990

1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2

Colm 991 to 1012

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1013 to 1034

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1035 to 1056

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1057 to 1078

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1079 to 1100

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2

Colm 1101 to 1122

3 2 3 2 3 2 3 2 3 2 3 2 1 2 1 2 3 2 3 2 3 2

Colm 1123 to 1144

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1145 to 1166

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1167 to 1188

3 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2

Colm 1189 to 1210

3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1211 to 1232

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1233 to 1254

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2

Colm 1255 to 1276

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1277 to 1298

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1299 to 1320

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1321 to 1342

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2

Colm 1343 to 1364

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1365 to 1386

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1387 to 1408

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2

Colm 1409 to 1430

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2

Colm 1431 to 1452

3 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2

Colm 1453 to 1474

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1475 to 1496

3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1497 to 1518

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1519 to 1540

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1541 to 1562

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2

Colm 1563 to 1584

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1585 to 1606

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1607 to 1628

3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1629 to 1650

3 2 3 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2

Colm 1651 to 1672

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1673 to 1694

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1695 to 1716

3 2 3 2 3 2 3 2 3 2 1 2 3 2 3 2 3 2 3 2 3 2

Colm 1717 to 1738

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1739 to 1760

3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2 3 2

Colm 1761 to 1765

3 2 3 2 3

States =

Colm 1 to 22

1 1 1 0 1 1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 0

1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 1 1 1 1

1 1 1 1 1 0 1 1 1 1 1 0 1 0 0 1 1 1 1 0 1 1

Colm 23 to 44

1 0 1 1 1 0 1 1 1 0 1 0 1 0 1 1 1 0 1 1 1 0

1 1 1 0 1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1

Colm 45 to 66

1 0 1 0 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 1 1 0

1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 1 1 1 1 0 1 1

1 1 1 1 1 0 1 1 0 0 1 1 1 1 1 0 1 0 1 1 1 1

Colm 67 to 88

1 0 1 0 1 1 1 1 1 1 1 0 1 0 1 0 0 1 1 1 0 0

1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0	0	1	1	1	1
1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	0	0	1

Colm 89 to 110

0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
1	1	1	1	1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	1	1	0
0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Colm 111 to 132

1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1
1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1
1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	0

Colm 133 to 154

1	1	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1
1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1

Colm 155 to 176

1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1
1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	0

Colm 177 to 198

1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1
1	1	1	0	1	0	1	1	1	0	1	0	1	1	1	0	1	0	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Colm 199 to 220

1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	0
1	0	1	0	1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1

Colm 221 to 242

1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	0	1	0	1	1
1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	1

Colm 243 to 264

1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0
0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0	1	1
0	0	1	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1

Colm 265 to 286

1	1	1	1	1	0	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	0	1	0	1	1	1	0
1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1

Colm 287 to 308

1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	0	1	0	1	0	1	0	1	1	1	0
1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1

Colm 309 to 330

1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1	0	1	0	1	0
1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1

Colm 331 to 352

1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1
1	1	1	1	1	0	1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	1

Colm 353 to 374

1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	0	1	0	1	0	1	0	1	0	1	0	1	1	1	0	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1

Colm 375 to 396

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1
1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	1	1	0

Colm 397 to 418

1	1	1	1	1	0	0	1	1	0	1	1	1	1	1	0	1	0	1	0	1	1
1	0	1	0	1	1	0	0	1	1	1	1	0	0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	0

Colm 419 to 440

1	1	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	0
1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	1
1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1

Colm 441 to 462

1	1	1	0	1	1	1	0	1	0	1	0	1	0	1	0	1	1	1	0	1	1

1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Colm 463 to 484

1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0
1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1

Colm 485 to 506

1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	0	1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	0	1	1

Colm 507 to 528

1	1	1	0	1	0	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	0
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1

Colm 529 to 550

1	1	1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	1	0	0	1	0
1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	0	0	1	1	1

Colm 551 to 572

1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1
1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	1
1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0

Colm 573 to 594

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	0	1	0	1	0	1	1	1	0	1	1	1	0	1	1	1	1
1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	0	1	1

Colm 595 to 616

1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1

Colm 617 to 638

1	0	1	0	1	0	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0

Colm 639 to 660

1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	1	1	1	0	1	0
1	1	1	0	1	0	1	0	1	0	1	1	1	0	0	1	1	0	1	1	1	1

Colm 661 to 682

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	0	0	1	1
1	1	1	0	1	0	1	0	1	0	1	0	1	1	1	0	1	1	0	1	1	1

Colm 683 to 704

0	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	0	1	0
1	1	1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	1
0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1

Colm 705 to 726

1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0
1	0	1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1

Colm 727 to 748

1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	1	0	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	0
1	0	0	1	1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	1

Colm 749 to 770

1	1	1	0	1	1	1	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	0	1	1	1	1	0	0	1	1	1	1	1	1	1	0	1	1
1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	0	1	0	1	1	1	0

Colm 771 to 792

1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	0	0	1
1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	0	0
1	0	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1

Colm 793 to 814

1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1
1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	0	1	1	1	1	1	0

Colm 815 to 836

1	0	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	1

1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	1	1	1	0
1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Colm 837 to 858

1	1	1	1	0	0	1	0	1	1	1	1	1	0	1	0	1	0	1	0	1	1
1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	0	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1

Colm 859 to 880

1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1
1	1	1	1	1	0	1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	0

Colm 881 to 902

1	0	1	1	1	1	1	0	1	1	1	0	1	0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	0
1	1	1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1

Colm 903 to 924

1	1	1	1	1	0	1	0	1	1	1	1	0	1	1	1	0	0	1	0	1	0
1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
1	0	1	0	1	1	1	1	1	1	1	0	0	0	1	0	0	1	1	1	1	1

Colm 925 to 946

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1
1	1	1	0	1	1	1	0	1	0	1	1	1	0	1	0	1	0	1	1	1	0

Colm 947 to 968

1	1	1	0	0	0	1	0	1	1	0	1	1	1	1	0	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	1
1	1	1	1	0	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0

Colm 969 to 990

0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	0
1	1	1	0	1	1	1	1	1	0	1	1	1	0	1	1	0	0	1	0	1	1
0	0	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1

Colm 991 to 1012

1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	0
1	0	1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Colm 1013 to 1034

1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
1	0	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	0
1	1	1	0	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1

Colm 1035 to 1056

1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1	0
1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	1	1	1
1	1	1	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Colm 1057 to 1078

1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1
1	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	0

Colm 1079 to 1100

1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	0	0	0	1	0
1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1

Colm 1101 to 1122

1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	0	1	1	0	1	0	1	1	0	1	0	1	1
1	0	1	1	1	1	1	0	1	1	1	0	0	0	0	0	1	1	1	1	1	0

Colm 1123 to 1144

1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	0	1	0
1	0	1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1

Colm 1145 to 1166

1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1
1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0	1	0

Colm 1167 to 1188

1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	0	1	1	0	1	1	1	0	1	0	1	0

Colm 1189 to 1210

1	1	1	1	1	0	0	1	1	0	1	0	1	0	1	1	1	0	1	1	1	1

1 0 1 1 1 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 1 0

Colm 1211 to 1232

1 0 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 0 1 1

1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 0 1 0 1 0 1 1 1 1 1 0 1 0 1 1 1 0

Colm 1233 to 1254

1 1 1 1 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 0 0 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0

1 0 1 0 1 0 1 0 1 1 1 0 1 1 1 0 1 0 1 1 1 1

Colm 1255 to 1276

1 0 1 1 1 0 1 0 1 1 1 1 1 1 1 0 1 0 1 1 1 0

1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 0 1 0 1 0 1 1 1 1 1 0 1 1

Colm 1277 to 1298

1 0 1 1 1 1 1 0 1 1 1 0 1 0 1 1 1 0 1 1 1 0

1 1 1 0 1 0 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 1

1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1

Colm 1299 to 1320

1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 0 1 1

1 0 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1 1 1 1 0

1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 1

Colm 1321 to 1342

1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 0 1 1 1 1 1 1

1 0 1 1 1 0 1 0 1 1 1 0 1 1 1 1 1 1 0 1 1 1

1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 0 0 1 0

Colm 1343 to 1364

1 1 1 1 1 0 1 0 1 0 1 1 1 1 1 0 1 1 1 1 1 1

1 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 0

1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 1 1

Colm 1365 to 1386

1 1 1 0 1 1 1 1 1 1 1 0 1 1 1 1 1 0 1 1 1 1

1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1

1 0 1 1 1 1 1 0 1 0 1 1 1 0 1 1 1 1 1 0 1 0

Colm 1387 to 1408

1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0
1	0	1	0	1	0	1	0	1	1	1	0	1	1	1	1	1	1	0	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	0	0	1	1

Colm 1409 to 1430

1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
1	0	1	1	1	1	1	0	1	0	1	1	1	0	1	0	1	0	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	0	0	0

Colm 1431 to 1452

1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	0	1	1	0	0	1	1	1	1	1	1	1	0	1	1
1	1	1	0	1	0	1	1	1	0	0	1	1	1	1	0	1	0	1	1	1	0

Colm 1453 to 1474

1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0
1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1
1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	1

Colm 1475 to 1496

1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1
1	1	1	0	1	1	1	1	0	1	1	0	1	1	1	1	1	0	1	1	1	0
1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1

Colm 1497 to 1518

1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0
1	1	1	1	1	0	1	0	1	0	1	0	1	0	1	1	1	0	1	1	1	1

Colm 1519 to 1540

1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	0	1	1	1	0
1	1	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	0	1	1

Colm 1541 to 1562

1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	0	1	0	1	1	1	0	1	1	1	1	0	1	1	0
1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0	1	1	1	1	1	1

Colm 1563 to 1584

1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1

1	0	1	1	1	0	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1
1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	0

Colm 1585 to 1606

1	0	1	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	0
1	1	1	1	1	0	1	1	1	1	1	1	1	0	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	0	1	1	1	1	1	1

Colm 1607 to 1628

1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	0
1	0	1	1	1	1	0	1	1	0	1	1	1	1	1	0	1	0	1	1	1	1
1	1	1	0	1	0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1

Colm 1629 to 1650

1	0	1	1	1	1	1	0	1	1	1	0	0	1	1	1	1	0	1	0	1	1
1	1	1	0	1	1	1	1	1	0	1	1	0	0	1	0	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Colm 1651 to 1672

1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1	1	0
1	1	1	1	1	0	1	0	1	0	1	1	1	0	1	1	1	1	1	0	1	1

Colm 1673 to 1694

1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	0	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0

Colm 1695 to 1716

1	1	1	0	1	1	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	1
1	0	1	1	1	0	1	1	1	1	0	1	1	0	1	1	1	1	1	0	1	0
1	1	1	1	1	1	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	1

Colm 1717 to 1738

1	1	1	1	1	0	1	0	1	0	1	1	1	1	1	0	1	1	1	0	1	1
1	1	1	1	1	1	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	0
1	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1

Colm 1739 to 1760

1	1	1	0	1	1	1	1	1	0	1	0	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	0	1	1
1	0	1	1	1	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1	1	0

Colm 1761 to 1765

1	1	1	1	1
1	0	1	1	1
1	1	1	0	1

Probabilities = 0 0.0017 0.0682 0.9301

MTTF = 478.6152

MTTR = 11.6775

REFERENCES

[1] Samanta, P. K., Vesely, W. E., Hsu, F., & Subudly, M. (1991). Degradation modeling with application to ageing and maintenance effectiveness evaluations. *Division of Systems Research, Office of Nuclear Regulatory Research, US Nuclear Regulatory Commission*.

[2] Hontelez, J. A., Burger, H. H., & Wijnmalen, D. J. (1996). Optimum condition-based maintenance policies for deteriorating systems with partial information. *Reliability Engineering & System Safety*, *51*(3), 267-274.

[3] Lam, C. T., & Yeh, R. H. (1996). Optimal maintenance-policies for deteriorating systems under various maintenance strategies. *Microelectronics Reliability, 3*(36), 446.

[4] Yeh, R. H. (1997). State-age-dependent maintenance policies for deteriorating systems with Erlang sojourn time distributions. *Reliability Engineering & System Safety*, *58*(1), 55-60.

[5] Grall, A., Bérenguer, C., & Chu, C. (1998, June). Optimal dynamic inspection/replacement planning in condition-based maintenance for deteriorating systems. In *Proceedings of the European Conference on Safety and Reliability (ESREL'98), June* (pp. 16-19).

[6] Kopnov, V. A. (1999). Optimal degradation processes control by two-level policies. *Reliability Engineering & System Safety*, *66*(1), 1-11.

[7] Bérenguer, C., Grall, A., & Castanier, B. (2000). Simulation and evaluation of condition-based maintenance policies for multi-component continuous-state deteriorating systems. In *Proceedings of the foresight and precaution conference* (pp. 275-82).

[8] Barata, J., Guedes Soares, C., Marseguerra, M., & Zio, E. (2001). Monte Carlo simulation of deteriorating systems. *Proceedings ESREL 2001*, 879-886.

[9] Barata, J., Soares, C. G., Marseguerra, M., & Zio, E. (2002). Simulation modeling of repairable multi-component deteriorating systems for 'on condition' maintenance optimisation. *Reliability Engineering & System Safety*, *76*(3), 255-264.

[10] Sawaragi, Y., Nakayama, H., & Tanino, T. (Eds.). (1985). *Theory of multi-objective optimisation* (Vol. 176). Elsevier.

[11] Borgonovo, E., Marseguerra, M., & Zio, E. (2000). A Monte Carlo methodological approach to plant availability modeling with maintenance, aging and obsolescence. *Reliability Engineering & System Safety*, *67*(1), 61-73.

[12] Cantoni, M., Marseguerra, M., & Zio, E. (2000). Genetic algorithms and Monte Carlo simulation for optimal plant design. *Reliability Engineering & System Safety*, *68*(1), 29-38.

[13] Martorell, S., Carlos, S., Sanchez, A., & Serradell, V. (2000). Constrained optimisation of test intervals using a steady-state genetic algorithm. *Reliability Engineering & System Safety*, *67*(3), 215-232.

[14] Holland, J. H. (1975). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. U Michigan Press.

[15] Holland, J. H., & Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimisation and Machine Learning.

[16] Beasley, D., Martin, R. R., & Bull, D. R. (1993). An overview of genetic algorithms: Part 1. Fundamentals. *University computing*, *15*, 58-69.

[17] Beasley, D., Bull, D. R., & Martin, R. R. (1993). An overview of genetic algorithms: Part 2, research topics. *University computing*, *15*(4), 170-181.

[18] Fonseca, C. M., & Fleming, P. J. (1993, June). Genetic Algorithms for Multi-objective Optimisation: Formulation, Discussion and Generalization. In *ICGA* (Vol. 93, pp. 416-423).

[19] Fogel, D. B. (1994). An introduction to simulated evolutionary optimisation. *IEEE transactions on neural networks*, *5*(1), 3-14.

[20] Coit, D. W., & Smith, A. E. (1994). Use of a genetic algorithm to optimise a combinatorial reliability design problem. In *Proceedings of the Third IIE Research Conference* (pp. 467-472).

[21] Srinvas, N., & Deb, K. (1994). Multi-objective function optimisation using non-dominated sorting genetic algorithms. *Evolutionary Computation*, *2*(3), 221-248.

[22] Fonseca, C. M., & Fleming, P. J. (1995). An overview of evolutionary algorithms in multi-objective optimisation. *Evolutionary computation*, *3*(1), 1-16.

[23] Herrera, F., & Lozano, M. (1996). Adaptation of genetic algorithm parameters based on fuzzy logic controllers. *Genetic Algorithms and Soft Computing*, *8*, 95-125.

[24] Parks, G. T. (1997). Multi-objective pressurized water reactors reload core design using genetic algorithm search. *Nuclear Science and Engineering*, *124*, 178-187.

[25] Joyce, P. A., Withers, T. A., & Hickling, P. J. (1998, December). Application of genetic algorithms to optimum offshore plant design. In *Safety and Reliability* (Vol. 18, No. 3, pp. 6-12). Taylor & Francis.

[26] Zitzler, E., & Thiele, L. (1999). Multi-objective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE transactions on Evolutionary Computation*, *3*(4), 257-271.

[27] Zitzler, E. (1999). Evolutionary algorithms for multi-objective optimisation: Methods and applications.

[28] Toshinsky, V. G., Sekimoto, H., & Toshinsky, G. I. (2000). A method to improve multi-objective genetic algorithm optimisation of a self-fuel-providing LMFBR by niche induction among non-dominated solutions. *Annals of Nuclear Energy*, *27*(5), 397-410.

[29] Busacca, P. G., Marseguerra, M., & Zio, E. (2001). Multi-objective optimisation by genetic algorithms: application to safety systems. *Reliability Engineering & System Safety*, *72*(1), 59-74.

[30] Marseguerra, M., & Zio, E. (1993). Nonlinear Monte Carlo reliability analysis with biasing towards top event. *Reliability Engineering & System Safety*, *40*(1), 31-42.

[31] Marseguerra, M., & Zio, E. (2000). Optimising maintenance and repair policies via a combination of genetic algorithms and Monte Carlo simulation. *Reliability Engineering & System Safety*, *68*(1), 69-83.

[32] Dubi, A. (2000). *Monte Carlo applications in systems engineering*. Wiley.

[33] Marseguerra, M., Zio, E., & Podofillini, L. (2005). Multi-objective spare part allocation by means of genetic algorithms and Monte Carlo simulation. *Reliability engineering & system safety*, *87*(3), 325-335.

[34] Feng, Z. (2013). Markov process applied to degradation modelling: Different modelling alternatives and their properties.

[35] Gacôgne, L. (2006). Methods to apply operators in a steady-state evolutionary algorithm. In *IPMU Conference, Paris*.

[36] Cheng, H. (2007). Implementation strategies and tools for condition based maintenance at nuclear power plants. *International Atomic Energy Agency, Vienna*.

[37] Dhillon, B. S. (2002). *Engineering maintenance: A modern approach*. Boca Raton, FL: CRC Press.

[38] Rao, S. S. (2009). *Engineering optimisation: Theory and practice*. Hoboken, NJ: John Wiley & Sons.

[39] Rajnarayanan, P. N. (2015). *Application and comparison of Meta heuristic techniques to optimal reactive power dispatch*.

[40] Reddy, S. K. (2014). *Reliability design of active redundant complex system by using stochastic dynamic programming and Monte - Carlo simulation techniques*.