Available online at www.elixirpublishers.com (Elixir International Journal)

Mechanical Engineering

Elixir Mech. Engg. 40 (2011) 5337-5350

ISSN: 2229-712X

Product design based on total cost of ownership approach

G. Kanagaraj[1] and N. Jawahar[2]

[1]Department of Mechanical Engineering, Thiagarajar, College of Engineering, Madurai, Tamil Nadu, India, 625015

[2]Department of Mechanical Engineering and Dean (R&D) Thiagarajar, College of Engineering, Madurai, Tamil Nadu, India, 625015.

ARTICLE INFO

Article history:
Received: 15 July 2011;
Received in revised form:
20 October 2011;
Accepted: 29 October 2011;

Keywords
Product configuration,
Objective criterion,
Constraints,
Numerical llustration.

ABSTRACT

In an ever increasingly competitive environment, product design forms the foundation for enduring sales responsiveness and enhanced customer satisfaction. Superior product performance, quality and reduced cost of ownership are the results of effective, efficient engineering and design. The impact of product design on total cost of ownership and the profitability of a manufacturing company are significant. This paper addresses design for total cost of ownership approach for allocation of redundancy that can minimize the total cost of the product during the product life time. A genetic algorithm based heuristic is developed to provide an optimal or a near optimal solution. This DTCO approach permits engineers to study product designs with respect to cost, reliability and performance during the conceptual design phase and enables the integrated method to identify design changes that improve performance and reduce total cost of ownership.

Introduction

The expectation of customers, often at the time of purchase of any product involves compromise between several aspects, in general are: initial purchase cost, service aspects, maintainability, availability of spares and resale value. One extreme case is a product with low reliability that is cheap which usually associated with high operating cost (repair/maintenance + service). Another extreme case is a product with high reliability that is normally costly. To improve customer satisfaction, manufacturing companies have started to address the issue of making products, which can be maintained at the least cost and with a minimum expenditure of support resources without adversely affecting the product performance and safety characteristics. As a customer or owner of a particular product he has to invest initial purchase cost for procurement and subsequently spend cost due to maintenance /service or replacement of failure components during the life time of the product. He also faced downtime cost due to non-availability of the product during the service period or replacement period. The first cost for procurement is not the last cost. During the life time of the product the costs involved for maintenance/service or replacement of failures components in the product is also important. The overall cost spends by the customer starting from procurement and usage period till the life of the product addressed here as Total Cost of Ownership (TCO), which is very much important. In the commercial sector, it is not inconceivable that a design-optimized product from the manufacturer's perspective may not be optimum from the customer's cost of ownership viewpoint. The initial investment made by the customer makes in terms of purchase cost will be important. But so too will those costs (Maintenance, service, replacement and downtime costs) which are incurred on a recurring basis over the useful life of the product. In some products these recurring costs will appear and may actually be greater than the initial investment. Customers realize that the initial investment of a product represents only one part of its TCO. Therefore it behooves the customer to understand as well as possible what the total cost of ownership will be over the product lifetime. This paper describes the performance characteristics of a product analyzed based on two different objective criterions. Minimization of initial purchase cost as the major objective not considering other costs such as maintenance/service, replacement cost and downtime cost, the design approach is said to be "Design based on Initial Purchase Cost approach" (DIPC). Minimization of total cost of ownership as the major objective by considering other costs involved during the life time such as maintenance/service, replacement cost and downtime cost the design approach is said to be "Design based on Total Cost of Ownership approach" (DTCO). Nowadays procurement professionals have always felt that the lowest total cost of ownership is the ultimate purchasing strategy. TCO mainly depends upon the product reliability. Lower the product reliability, lesser is the initial purchase cost and higher is the replacement and downtime cost, and vice versa. The relationship between the elements of TCO and reliability is presented in Figure.1. Cost versus reliability curve exhibits the following features:

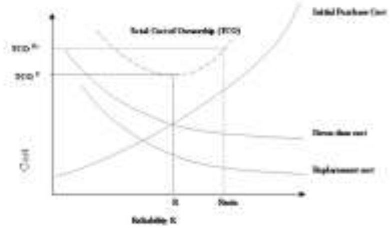


Fig. 1 Relationship between reliability and cost elements of ownership cost

- Initial purchase cost is a monotonic increasing function of reliability
- Downtime cost is a monotonically decreasing function of reliability
- Replacement cost is a monotonically decreasing function of reliability

Tele:
E-mail addresses: gkmech@tce.edu

In order to design and develop a highly reliable product requires high investment in procurement and this will be reflected in considerable measure in the TCO. The production facilities must be sufficiently sophisticated to enable manufacture of precision components, with the result that the production cost also would increase with the requirement of greater reliability. The optimal reliability of the product 'R' corresponds to minimum TCO (ie. TCO^R). However the user requirement on minimum product reliability (R_m) may be higher than 'R'. In such situations the minimum TCO thus becomes 'TCO^{Rm}' and the optimal reliability would be 'R_m'. The level of reliability of the product in general, would fall in the order of 0.96 and above. This requires that elements of the product (components or subsystem) have the higher reliabilities in the order of 0.99 and more depending upon number of elements in the product. Achieving these high reliabilities in an element requires substantial manufacturing cost. Manufacturing cost increases exponentially with increase in product reliability. When total cost of two or more components of lower manufacturing cost with inferior reliability is less than the cost of a single component of higher reliability, then parallel system with more number of inferior components is considered cost effective. In addition to this many systems cannot achieve their intended reliability without using redundancy. If the state of art is such that either it is not possible to produce highly reliable elements or the cost of producing such elements is very high, to improve the product reliability by the technique of introducing redundancies. This redundancy concept is mostly adopted for reliability improvement. Higher redundancy leads to higher investment cost and more space. Lower redundancy results in poor product reliability leads to high service and downtime costs. Hence an optimal level of redundancy is required for minimum TCO. On these concerns this paper presents a problem of product optimization on redundancy consideration.

Total Cost of ownership is an important factor when purchasing any product which includes not only initial purchase cost but also other costs involved during the useful life period. Customers consider only the lowest initial purchase cost which is a small portion of ownership cost will not provide the minimum cost spend during the life period of a product. Hence decisions based on cost of ownership lead to minimum cost spend by a customer during the life period. For the above significant importance of total cost ownership this paper addresses to determine the number of redundant identical components in each element of a product that minimizes the total cost of ownership during the product life time and also meets certain minimum requirement on product reliability under limited space availability constraints.

The rest of the paper is organized as follows: The related literature is reviewed in Section 2. Section 3 describes the mathematical formulation of TCO. In Section.4 a numerical example is given for the justification of DTCO model and need for implementation of the proposed methodology along with sensitivity analysis. Our conclusions are drawn in Section. 5

Literature Review

In today's competitive challenging business environment low cost of ownership is a key requirement. Cost of ownership originated at Intel Corporation during an examination of the total cost of acquiring, maintaining and operating purchased equipment. This concept was introduced to SEMATECH by Dean Toombs, an Intel assignee . These enable analysis of the cost sensitivities of many variables. Often the actual purchase price is not the greatest cost of tool. Productivity, reliability, maintenance, yield and consumable spare parts can have larger costs over tool life.

Cho et al. (2000) proposed a systematic approach for the development of the reliability based seismic safety and cost effective performance criteria for design and upgrading of bridges based on the minimum expected life cycle cost.

The cost functions which consist of the initial cost, maintenance cost, direct damage cost, and indirect losses are formulated as functions of structural damage probability. The system (here considered structure) failure probability is considered for the development of target reliability for seismic design and upgrading of bridges. A rational life cycle cost model for bridges is established and expressed in terms of global damage indices and damage probabilities.

Total ownership cost is being adopted by government and industry to manage business processes. Hitt and Battelle, (1998) this paper describes the use of total ownership cost and the decision support tools in government.

Total ownership cost applied to government (federal, state, local), industries (automobile, aerospace, airlines, electronics, information, etc.) and individual businesses. Also describes that reliability is the major parameter to drive the TOC.

Elegbede and Chu (2003) in their paper considered the allocation of reliability and redundancy to parallel-series systems, while minimizing the cost of the system. They have proven that the necessary condition for optimal reliability allocation of parallel-series systems is that the reliability of the redundant components of a given subsystem is identical and the components in each stage of a parallel-series system must have identical reliability.

Coit and Smith (1996) described a redundancy allocation problem for a series parallel system, in which there is a specified number of subsystems and, for each subsystem, there are multiple component choices, which can be selected and used in parallel. For those systems designed using component types, with known cost, reliability, and weight, the system design and component selection become a combinatorial optimization problem. The consumer electronics is one such example where new system designs are composed largely of standard component types (microcircuits, resistors) with known characteristics.

The problem is then to select the optimal combination of parts and redundancy levels to meet reliability and weight constraints. DTCO technique has been used in education, especially for calculating TCO of computers and networking for school districts. TCO is as much management, use of staff time, and best workflow practices as it is about system reliability and quality. The above literature emphasizes and reveals that there is no such thing as a "generic" TCO model and calculation of TCO for a given product.

Design based on Total cost of ownership analysis is an important point to be considered in designing a product; There exists no generalized data model with combining different factors such as manufacturability, serviceability and reliability; At this juncture the paper considers importance of total cost of ownership and the effect of reliability on TCO, a mathematical model developed for considering the costs involved during the life time of the product such as Initial purchase cost, Replacement cost and downtime cost under specified product reliability.

Problem description

Product configuration

Design and development of new products involves the selection of number of components and a system level design configuration to satisfy detailed functional and performance requirements. For this problem, the overall product configuration is partitioned into a specific number of elements *'m'* arranged in series. For each element *'j'* (*j=1,2…. m*) there are *'x_j'* numbers of functionally equivalent redundant components with known characteristics such as reliability *'r_j'*, cost *'c_j'* and space *'s_j'* are used. The components are added until the product constraints are violated. However the number of components in element *'j'* is limited to *'X_j'* due to design constraints. For each element, at least a minimum of one component or maximum of *'Xj'* components may be chosen and arranged in parallel. The overall product reliability always depends on elements reliability. Hence increasing element reliability can efficiently increase the product reliability. It may then become advisable, or even necessary to place additional components (redundant) in parallel in each element to maximize the product reliability. More number of redundant components increases the product cost and space occupancy. Hence optimal redundancy level is required due to cost and space constraints. Figure.2 depicts a typical layout of series product configuration. Failure of any element '*j*' leads to failure of the product. An element '*j*' would be considered functional if at least any one of its 'x_j' components is operational.

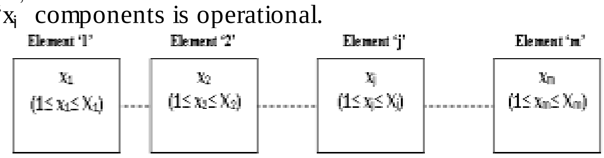


Figure.2 Typical layout of series product configuration

Assumptions

The component's characteristics such as cost, reliability and space occupancy are known and deterministic.

Failures of individual components are independent.

All redundancy is active redundancy without repair.

Failed components do not damage the product and are not repaired.

Money value remains constant

Maintenance and Service cost over the period of life of a product is constant

Product is designed for a specific life period, after the life period it is considered as scrap and sold at scrap value

Objective criterion

Product development considers many objectives such as maximum performance, minimum price, minimum service, more life, and easy availability and so on. Determination of an optimal or near optimal product design is very important to economically produce new products which meet and exceed customer's expectations for reliability, performance, etc. This paper considers design of product based on two different approaches. Design based on Initial Purchase cost (DIPC) and Design based Total Cost of Ownership (DTCO). Design based on total cost of ownership always more benefit to the manufacturer as well as customer. Hence this paper mainly focus on minimization of Total cost of ownership (TCO), which accommodates the three major aspects of price, service and availability, as the objective criterion. TCO relates the cost of acquiring a product and using it over the specified period of life. The various cost elements of TCO are: Initial Purchase Cost 'IPC'; Maintenance and Service Cost 'MSC'; Replacement Cost 'RC' and Down Time Cost 'DTC'. They are explained below:

Initial Purchase Cost:

The Initial Purchase Cost of a product is purely depends on the number of components in each element and number of elements in the product. It includes all the costs of buying a product and bringing it into operation. Thus, in general, the initial purchase cost of the product can be calculated by the following equation.

$$\text{Initial purchase cost} = \sum_{j=1}^{m}(c_j x_j) \quad 1\le x_j \le X_j \,(1)$$

Maintenance and Service Cost:

Maintenance is defined as all actions required to retaining a product in, or restoring it to, a specified condition. The maintenance cost is the annual cost of maintaining a product in working condition.

This includes diagnosis, repair and inspection. Product requires maintenance/service during the life period. Nature of maintenance and level of maintenance depends upon kind of product. Electronic products such as micro circuits, resistors, and motherboard are relatively less maintenance/service cost compared to mechanical components. Fault detection is also difficult and requires proper costly tool, test equipment and leads to longer downtime. If it is no longer economical to repair the failed unit. Instead the failed element is discarded and replaced with a new one. Normally customers choose annual maintenance/service contract for all electronic products. The maintenance/service cost incurred during the life period will product the same effect in IPC and TCO .

Hence for TCO calculation the maintenance/service cost over period of life is not considered here.

Replacement cost

Failure of all redundant components in an element leads to product failure. All the components in a failed element are replaced with new components. The cost of replacing all the components in a particular failed element is called replacement cost. For individual element replacement additional cost incurred (β) during replacement. Hence total cost of replacement of all failed elements in a product can be expressed as follows:

$$\text{Replacementcost RC} = \sum_{j=1}^{m}\frac{\ln}{\left(1-(1-r_j)^{x_j}\right)}\beta\, c_j\, x_j \qquad 1.5 \le \beta \le 1 \;(2)$$

Where β is a cost factor to be included for individual replacement of element.

Down time cost

The product down time is the total time for which the product is not available due to failure of any elements, replacement time, periodic inspection and maintenance/service time. Product downtime is a real cost to the owner that the product is not available for productive work.

$$\text{Down time cost} = \left(1-\prod_{j=1}^{m}\left(1-(1-r_j)^{x_j}\right)\right)N\gamma \qquad (3)$$

To sum of all these costs addressed here as TCO of the product, this is to be minimum for any product apart from the type of brand or manufacturer.

The objective thus becomes minimization of TCO = IPC + RC + DTC

Constraints

Operational points specify a certain minimum level of reliability (R_{min}) for the product. Hence the overall product reliability becomes one of its constraints.

$$\prod_{j=1}^{m}\left(1-(1-r_j)^{x_j}\right)\geq R_{\min} \qquad (4)$$

More number of redundant components will occupy more space. Any element *'j'* can accommodate maximum of *'Xj'* components. Maximum space availability of the product is limited to SA. Hence space limitation for the product is also considered as another constraint.

$$\sum_{j=1}^{m} x_j s_j \leq S \qquad (5)$$

Besides, there is limitation on maximum number of redundant components (X_j) admissible in the element 'j' and product feasibility requirement constraint 'x_j' (i.e. at least a minimum of one component must be functional for the product to be functional.)

$$1 \leq x_j \leq X_j \qquad (6)$$

Problem statement

Development of new product involves the selection of number of components and configuration to satisfy detailed functional and performance specifications. In this problem the overall product is partitioned into a specific number of elements 'm' arranged in series. Each element can accommodate 'X_j' number of redundant identical components arranged in parallel. The component characteristics such as reliability 'r_j', cost 'c_j' and space 's_j' occupied are known. The design optimization problem is to determine the optimal number of parallel components $(x_j)_{opt}$ for all 'm' elements in the product to minimize the total cost of ownership for the specific life period subject to constraints on minimum level product reliability and maximum space available. The integer variable (x_j) denote the number of parallel components connected in each element will decide the overall product reliability, space occupancy and total cost of ownership hence it is considered as a decision variable.

Mathematical Model

$$\text{Minimize TCO} = \sum_{j=1}^{m}(c_j x_j) + \sum_{j=1}^{m}\frac{\ln}{\left(1-(1-r_j)^{x_j}\right)}\beta\, c_j\, x_j + \left(1-\prod_{j=1}^{m}\left(1-(1-r_j)^{x_j}\right)\right)N\gamma \qquad (7)$$

Subject to constrains:

Constraint 1 represents the minimum product reliability constraint.

$$\prod_{j=1}^{m}\left(1-(1-r_j)^{x_j}\right)\geq R_{\min} \qquad (8)$$

Constraint 2 represents the maximum space availability constraint.

$$\sum_{j=1}^{m} x_j s_j \leq S \qquad (9)$$

Constraint 3 defines the decision variable.

$$1 \leq x_j \leq X_j \qquad (10)$$

Proposed methodology

The formulation of the above problem is a non-linear constrained integer-programming problem. Two algorithms are proposed and discussed in this paper. The structure and characteristic features of them are delineated in the following sub sections.

Exhaustive Search Algorithm

Coding Schema

The exhaustive search algorithm 'ESA' is the complete enumerative search procedure that searches for an optimal feasible solution with the sequential evolution of all possible solutions. The exhaustive search algorithm is developed to enumerate the values of IPC, TCO, R and S for all possible configurations of the product. Figure 3 shows the coding schema of the ESA and is self-explanatory.

```
//* Input_ mod ( )
{
Input: m, SA, Rmin, λ, γ, β, N
for j = 1 to m
{
input: rj ,cj ,sj, Xj
}
}

//* Estimation_ mod ( )
{
for x1 = 1 to X1
{
for x2 = 1 to X2
{
for x3 = 1 to X3
.
.                              .
for xm =1 to Xm
{
find :IPC,TCO,R , S;
}
}
}
.                              .
.

}
}
}

//* Output_ mod ( )
{
print: x1 , x2 , x3 .... xm ,IPC,TCO, R &S.
}
```

Figure. 3 Coding schema of ESA

Numerical Illustration

The mathematical formulation of the problem is illustrated with the numerical values given below.

$m = 5$; $R_{min} = 0.95$; $\gamma = 10000/\text{year}$; $\beta = 1$; $N = 5$ years;

The data associated with the components are given in Table I. Cost, space and reliability of the components and maximum number of components available for each element are given. The value of the constraints at each element is shown in the lower portion of the table. Ten additional example problems are also considered to illustrate the performance of proposed methodology. The dataset for these examples (called Problem 1, 2, 10) are listed in Table. II. The decision variable at each element is, of course, the number of redundancies to be introduced.

All possible solutions and the corresponding the values of initial purchase cost, total cost of ownership, reliability and space occupancy of the product evaluated by the exhaustive search algorithm are presented in Table. III. When a solution satisfies all the constraints it is called a feasible solution; otherwise, the solution is said to be infeasible. The number of possible solutions both feasible and infeasible solutions are always depends on the maximum number of redundant components available for each element. ESA is iterative in nature. The total number of iteration is 24192 in the progression from an initial iteration corresponding to the configuration of (11111) yielding lowest reliability of 0.189 to final iteration corresponding to the configuration of (87967) yielding the highest reliability of 0.999. In this case among 24192 possible solutions 1040 solutions are feasible that means satisfying both minimum product reliability constraint and maximum space available constraints. Among all the possible solutions when there is no redundant component in all the elements the product will have very low reliability value of 0.189 solution corresponding to iteration no.1 (series product reliability) and will have the lowest purchase cost of Rs.200 with very high total cost of ownership of Rs.40815. Product having maximum number of redundant components in all the elements will have very high reliability of 0.999 solution corresponding to iteration no 24192 (series parallel product reliability) and also have the highest purchase cost of Rs.1560 and total cost of ownership of Rs.1588 . Before taking the decision to select the optimal solution, consider all the solutions not necessarily feasible solutions (i.e., product constraints may be violated). It is interesting to note that there are four optimal solutions found among 24192 possible solutions. Two optimal solutions in the feasible region and another two optimal solutions in the infeasible region.

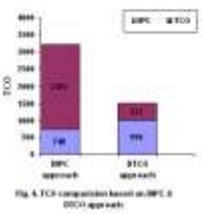
Among all the possible solutions, if the customer considers only the short term benefit of minimum cost of purchase or cheaper product (Design for Initial Purchase Cost approach DIPC) not considering the other factor such as product reliability, space occupancy and total cost of ownership the solution corresponding to iteration no.1 is the best choice among 24192 possible solutions.

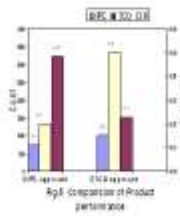
If the customer consider only the long term benefit of minimum total cost of ownership even though product is more expensive initially (Design for Total Cost of Ownership approach DTCO) not considering factors like product reliability, space occupancy and total cost of ownership, solution corresponding to iteration no. 15061 is the best choice among 24192 possible solutions.

Consider only the feasible solution region (not violating the constraints) it is also important to note that there are two optimal solutions were found based on DIPC &DTCO approaches. If the objective is to minimize the initial purchase cost based on DIPC approach. The optimal is solution corresponding to iteration no.8419

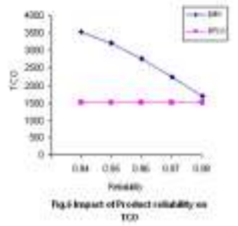
If the objective is to minimize the total cost of ownership based on DTCO approach. The optimal is solution corresponding to iteration no.14140 By comparing both optimal solutions TCO figure shows that over five years, the cheaper product to buy is actually more expensive to own, over five years it will cost Rs.3212 with the lowest initial cost of Rs.740. The more expensive product to buy will cost Rs.990 and to own over five years it will cost Rs.1517. However as can clearly be seen in Figure.4 based on DTCO approach, the product is costly to buy initially but it offers cheaper ownership cost than DIPC approach. Fig.5 shows the performance comparisons of product based on DIPC &DTCO approach. Design based on total cost of ownership provides product with high reliability and least ownership cost.

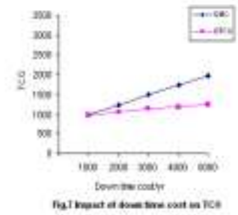
It clearly shows that based on DIPC approach the total cost of ownership over the life period of product is very high compared to DTCO approach. This is always true when increasing the number of elements in the product or variation of parameters like downtime cost/yr, life of the product and reliability of the product. Hence the most important observation is that the recommended design of a product based on Minimum TCO as the objective always gives the best configuration product to the customer. TCO is a new tool that reveals that hidden costs –all of the costs –associated with buying, owning and operating a product over a useful life period. This would seem to imply that the product with lower reliability is less advisable choices and in general, this is true.

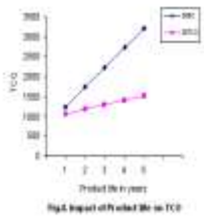




The effect of various parameters like product reliability, downtime cost, product life and number of elements on TCO is tested for both the approaches by varying one parameter and all other parameters being kept as same value using the same dataset given in Table.1 and the results are listed in Table IV to VII.







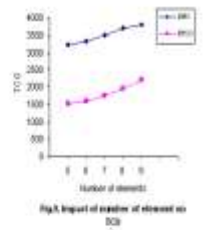


Fig.6 shows the trend of TCO variation by increasing the product reliability from 0.94 to 0.98. DTCO approach always provides the lowest TCO for all values of reliability. DIPC approach provides the maximum TCO when the product is being designed with low reliability value. Further increasing the value of product reliability the TCO is comedown. Product is designed with maximum reliability both the approaches provide same TCO cost. One of the most important cost drivers in TCO equation is number of redundant components (x_j) and its reliability (r_j). DTCO approach always provides the best solution because it searches the configuration corresponding to maximum reliability possible.

Fig.7 shows the trend of TCO variation by increasing the product downtime cost/yr from 1000 to 5000. DTCO approach always provides the lowest TCO for all values of increasing downtime cost. Total cost of ownership is always increasing function with the downtime cost in both the approaches. Cost of downtime is crucial for certain product even breakdown by an hour leads to heavy loss, in that case design of product based on DIPC approach will leads to enormous amount when compared to DTCO approach. For very small values of downtime cost/yr, both the approaches provide the same TCO. Hence design based on DTCO approach is more suitable for the product having high downtime cost/yr.

Fig.8. shows the trend of TCO variation by increasing the product life from 1year to 5years. DTCO approach always provides the lowest TCO for all values of increasing product life. Life of the product is not significant for certain type of products. Electronic product cost is come down over the year due to technology advancement. Due course of time the same product with advanced features is also available for same cost purchased earlier. Customer also more willing to buy latest technology product. In that case life of the product is not a critical factor design based on both the approaches will provide the same results.

Fig.9 shows the trend of TCO variation by increasing the number of elements (size of the problem) in the product from 5 to 9. Total cost of ownership is always increasing function with increasing number of elements in the product in both the approaches. DTCO approach always provides the lowest TCO for all increasing values of *'m'*. We have solved a variety of problems in the various values of component reliability, cost, space, downtime cost/yr and number of elements and have concluded the following.

Based on the above detailed discussions, an interesting observation made when examining the total cost of ownership design based on DTCO approach will provide always lower cost of ownership compared to DIPC approach.

The ES process would be tedious when considering large number of elements and redundant components. Computational time increases with the increase in number of elements/subassemblies in the product. Besides needlessly it evaluates many infeasible solutions. To overcome the above said difficulties, search heuristics such as Genetic algorithm, Simulated Annealing algorithm can be used to get the optimal solution with less computational effort.

Many approaches have been considered to deal with these kinds of problems. Fyffe et al. (1968) used the dynamic approach with a Lagrangian multiplier to deal the problem with two constraints. Nakagawa and Miyazaki (1981) presented the use of surrogate constraints to solve s series-parallel system with 14 subsystems and with two constraints on cost and weight. The dynamic programming approach can be used to find an optimal solution. However, it is nearly impossible to obtain optimal solutions within a reasonable computational time when system size becomes large. Geometric programming is usually used to solve nonlinear problems. However solutions derived from this approach are usually non-integer. Thus, for the solutions to be feasible, the real solutions should be rounded off. This leads to the problem of how to properly round off real solutions to integer solutions. Indeed, the rounded solutions cannot be guaranteed as exact optimal solutions. From the above, it can be observed the exact solutions to these kinds of problems cannot be derived easily since the computational complexity. This, in turn, has prompted recent researchers to develop heuristic or meta-heuristic methods to derive approximate solutions of acceptable quality in reasonable computational time. Genetic algorithm is a well-kwon heuristic algorithm used to solve combinatorial optimization problems. Coit and Smith (1996) provided a penalty guided genetic algorithm to deal with the series-parallel redundant allocation problem with multiple component choices. They tested the problems given by Nakagawa and Miyazaki and produced a solution with a higher reliability than those reported by Nakagawa and Miyazaki for 27 of the 33 problems. In addition to genetic algorithm, other heuristic or meta-heuristic approaches such as Simulated Annealing method and the Tabu search method have also been used to handle system reliability problems. Readers interested in these works using these approaches can refer to the excellent review paper by Kuo and Prasad(y). In light of the aforementioned approaches, this paper proposes a heuristic approach Genetic Algorithm to solve the series-parallel redundant components reliability problem.

Genetic Algorithm

Genetic Algorithm (GA) is a search algorithm, which relies on analogies to natural processes, based on the principles of evolution and hereditary. Such systems maintain a population of potential solutions that have some selection process based on fitness of individuals, and some recombination operators. GA's are a random evolutionary search algorithm that mimics the principle of natural genetics. GA maintains a whole family of solutions in parallel. The various solution of this family can be seen as samples of the search space. They complete and cooperate through a number of iterations in order to gain improvements. The selection of individuals to represent next generation, the method of reproduction and the method introducing genetic material are carried out randomly. The genetic algorithm is based on the recognition that evolution, with its principle of mutation and selection, represents an efficient process for solving optimization problems. GA applies a local search operator "crossover" and a global search operator "mutation" in an evolutionary framework. These algorithms combine the advantages of both worlds, the efficiency of local search and the robust of evolution. Local search offers further improvements of solutions resulting from heuristics. The advantages of GA compared with classical optimization technique including shorter computing times without determinant of high quality or goal attainment in the solution found. On the above considerations a GA based heuristic is suggested. The proposed GA is explained in this section.

Structure of GA

The proposed GA, coded in C++ language, has various modules and figure 10 illustrates the structure of it.

Main ( )
{
Input_ mod ( );
Initialization_ mod ( );
Termination_mod ( );
Evaluation_mod ( );
Sorting_mod ( );
New_ pop_ gen_ mod ( )
END:
Output_mod ( );
}
Input_ mod ( )
*// * relevant data to the problem are given as input*//*
{
Input: m,, SA, R_{min}*,* λ*,* γ*,* β*, N;*
For(j = 1 to m)
{

```
input: r_j; c_j ,s_j,  X_j
            }
}
Initialization_ mod ( )
// *This module generates initial population of chromosomes,
each representing a set of x_j values (feasible or infeasible)* //
{
set: Gen_no = 0;
set: Pop_size= 5*m;
set: TCOopt = M (Where M is a largest number);
set: p_cross =0.6;
set: p_mut =0.06;
for( c = 1 to pop_size)
{
for( j = 1 to m )
            {
            generate chromosomes c(j) = ran(X_j);
            }
}
}

Termination_mod ( )
If (Gen_no =20 * m)
            {
            Go to output ( )
            }

Evaluation_mod ( )
for c= 1 to pop_size
{
decode  chromosome for x_j for all j;

find: R(c);, S (c) ; TCO(c);
}

Sorting _mod( )
{
for c = 1 to pop_size
{
If (R(c) < Rmin (OR)  S(c))  > SA)
                        {
                        TCO (c))= m*TCO;
                        }
set: fit(c)=TCO(c);
}
if (TCO (c) <TCO_opt)
                        {
x_jopt for j
TCO_opt = TCO;
R_opt = R(c);
S_opt = S(c);
}
}

New_ pop_ gen_ mod ( )
{
Sel_mod ( );
Cross_mod( );
Mut_mod ( );
}

Sel_mod ( )
{
find: new_fit(c) = e^(-kfit(c)) = e^(-0.0001fit(c))// k is set based on trail;
p(c) = new_fit(c) /Σ new_fit(c)
c p(c)= Σ p(c);
}
for c = 1 to pop_size
{
generate: r1; (where ' r1' is a random number between 0 -1)
set: c' = c that corresponds to the chromosome 'c' that satisfies
the  condition (c p(c) ≥ r > c p(c-1);
}
}

Cross_mod( )
{
for c = 1 to Pop_size
            {
generate: r2; (where 'r2' is a random number between 0 -1)
if r <= p_cross
            {
set: c'' = c'
}
            }
repeat till all c'' are exhausted
{
select: two c'' at a time
do: cross over //*single point cross over with a random number
between 1 to m * //
set: c''' = c''
}

}

Mut_mod( )
{
for c= 1 to Pop_size
            {
            for j = 1 to m
                        {
            generate: r3; (where ' r3' is a random number between
0 -1)
            if (r < P_mut)
                        {
generate 'r4' // a random number between 1 to X_j
                        set c(j) = r4;
                        }
}
            }
}

Output_ Mod ( )
{
printf: x_jopt for j; TCO_opt = TCO; R_opt = R(c); S_opt = S(c);
}
```

Fig.10. Coding structure of Genetic Algorithm

Numerical Illustration

Date Input Module:

The GA is illustrated with the data given in section 3.1.2, which is used to illustrate ESA. In addition to the problem data, the parameters of [Goldberg DE (2002)] are set as given in Table .VIII.

Initialization Module

The main issue in designing a genetic algorithm is that of finding the proper encoding. There are four commonly used encoding: binary encoding, permutation encoding, direct value encoding and tree coding. Binary encoding is the most common and simplest one. In binary encoding every chromosome is string of bits, 0 or 1. In direct value encoding, every chromosome is a string of some values. Traditionally, solution encodings have been a binary string (book ref). For combinatorial optimization, an encoding using integer values can be more efficient (J. Antonisse). In this paper these two coding were adopted to find the optimal and the performances of GA is tested with binary encoding and direct value encoding methods.

Method: 1 GA with binary coding

In this method of GA design, using the lower & upper bounds of the decision variables are converted into binary strings which are used in the chromosome representation of solutions. A chromosome is a collection of xj number of redundant components in parallel for all m elements in the product.

Chromosome 1: 0110 1101 0101 1110 0101

Chromosome 2: 1001 1000 0110 1001 1011

Each bit in the chromosome is called a gene. A set of four genes represents the number of redundant component in an element.

For example:

A set of first four genes in chromosome 1 (0110) represents the 4 number of redundant component may be used in the first element. All set of genes can be decoded as follows:

X1= { (0 x 23)+ (1 x 22)+ (1x 21)+ (0 x 20)} 7/15 +1=3.8□4

Method: 2 GA with direct value encoding

In this method of GA design, direct value encoding is used to represent a chromosome. Each chromosome consists of string of genes, where each gene represents number of redundant components in an element.

Chromosome 1: 4 7 4 6 3

Chromosome 2: 6 8 4 3 2

For example solution chromosome 4 7 4 6 3 represents a solution which has element 1 has 4 number of redundant components, element 2 has 7 number of redundant components etc.,The length or size of chromosome (chr_length) is equal to the number of elements in the product. The initial population with 25 chromosomes (5*m) are generated randomly.

Evaluation Module:

The fitness parameter of fit(c) , modified fitness parameter new_fit(c), and propability of survival of the chromosome p(c) are calculated using equation 7 for the entire population. The TCO of the product [fit(c)] is calculated using the objective function equation. The second evaluation is to convert the fitness parameter to a new fitness value [new_fit(c)] suitable for the minimization of objective function and scaling them high so that a very few extremely superior individuals would be selected as parents too many times. The fitness value is scaled suing the negative exponential function. After many trials, the value of the constant k is assumed as 0.0001.

Selection module:

The next population of the same size is selected with survival probability [p(c)] in a Roulette wheel procedure. A random number 'r' between 0 and 1 is spun and a chromosome c is selected which satisfied the condition given below.The selection process is repeated as many times as equal to population size.

c p(c) =□ p(c)

c p(c) ≤ r1> cp(c-1)

4.3.5 Crossover module:

This involves two steps, viz., i) Selection of chromosome for crossover and ii) crossover operation. Considering the probability of crossover is 0.6 [p_cross]. A random number between 0 and 1 is generated for each chromosome and the chromosome that get a random number less than p-cross are selected for crossover. The crossover operation is carried out using a Partially Mapped Crossover (PMX) operator with two cutting points. The number of chromosome is selected for crossover and they become parent. The selected chromosomes (parent) c' and c' undergo crossover operation and give offspring. The random cut points generated are 4 and 6 and the offspring chromosomes after crossover are c" and c".

Mutation module:

The crossed chromosomes set of new population are mutated randomly with probability of mutation 0.06 [p_mut]. A random number between 0 and 1 is generated for each gene. If the random number of a gene is less than the probability of mutation then the gene type value of that gene is changed by another random number generated between 0 and maximum number of component X(g). The elements selected for mutations are shown in bold letters and the mutated values are shown as bold italics in Table IX, X.

Termination module:

The entire process of evaluation, selection crossover and mutation modules are repeated for the fixed number of generations [iter_no] which depends on the size of the problem. The number of iterations is considered as the termination criterion and fixed as m*20. The best solution is obtained from two GA methods are listed in Table. XI.

Output module:

The chromosome that has the minimum fitness value (local best) is stored separately in each iteration. The best among the local bests stored is sorted out and becomes the global best. The obtained chromosome is called optimal chromosome corresponding to the string/sequence provide the optimal redundant components for 'm' elements in the product. The optimal values of TCOopt , Ropt and Sopt corresponding to the optimal chromosome are calculated and listed in Table. XI. Table IX & X illustrates the procedural steps of initial and new population generation for method-I & method-II. . The best solution in each generation is stored (i.e. generation best). The best among the best solutions of 100 generations (i.e. 20 * m) is the global best.

Performance comparison

The performance of GA is tested with two methods in different size of 10 sample problems. The optimal solutions to the sample problems are derived from Exhaustive search algorithm. The optimal value of TCO obtained from Exhaustive search algorithm and GA methods are given in Table XI. The GA using binary coding (Method-1) has produced the optimal solution in six cases for 100 iterations. Increasing the iteration to 1000 the solution suddenly converges to optimal quickly. The GA using value coding (Method-II) has produced the near optimal solution always. By choosing proper mutation probability it produces optimal solution. However the percentage of deviation from optimal is always less than 5%. The results reveal that GA is capable of providing either optimal or near-optimal solutions in most of the cases. Selection of GA parameter and tuning will improve the efficiency of GA. It consumes lesser computational time than ES algorithm. When large size of products and more number of elements are used, this GA will be efficient than ES.

Conclusion

This paper emphasizes a new concept of product design based on total cost of ownership perspective. DTCO approach permits engineers to study product designs with respect to cost, reliability and performance during the conceptual design phase and enables the integrated method to identify design changes that improve performance and reduce cost. The mathematical formulation of the DTCO approach was developed by considering major costs such as PC, RC and DTC. Besides, it accounts for the practical constraints of product reliability and space limitation. The present work is an attempt to justify the concept and methodology of a reliability based design strategy with a basic of DTCO approach. The design approach is based on building a product with more than one low reliability components for the situation where the cost of manufacturing a component increases exponentially with slight increase in reliability. The DTCO approach presented in this paper will be useful for the design of new products, especially electronic products.

Acknowledgement:

The authors thank the Management, Principal, and Head of Mechanical Engineering of Thiagarajar College of Engineering, Madurai for their support and encouragement provided to carry out this work.

Reference

Brown R.E, Gupta S, Christie R.D, Venkata S.S, Fletcher R, (1997), "*Automated Primary Distribution System Design: Reliability and Cost Optimization*", IEEE Transactions On Power Delivery, Vol.12, No.2, PP-1017-1022.

Chern MS. On the computational complexity of reliability redundancy allocation in a series system, Operations research Letter 1992; 11:309-15.

Cho. H.N, Ang A.H-S, (2000) "*Determination Of Cost-Effective Optimal Target Reliability For Seismic Design And Upgrading Of Long Span PC Bridges"*,ASCE Speciality Conference on Probabilistic Mechanics and Structural Reliability, ,PP-1-6.

Coit.D.W, Alice E. Smith,(1996) "*Reliability Optimization of Series-Parallel Systems Using a Genetic Algorithm*", IEEE Transactions On Reliability, Vol.45, No.2, June, PP-254-259.

Elegbede.C, Chu.C,(2003) "*Reliability Allocation Through Cost Minimization*",IEEE Transactions On Reliability, Vol. 52,No.1, March, PP- 106-111.

Fane.S, Willetts.J, Abeysuriya K, Mitchell C, Etnier C, Johnstone S, (2004) "*Evaluating Reliability and Life- Cycle Cost for Decentralized Wastewater Within the Context of Asset Management*", Paper presented at 1st International Conference On Onsite Wastewater Treatment and Recycling/ 6th Specialist Conference on Small water Systems , Fremantle, Australia, February 11-13, , PP- 1-8.

Fyfffe DE, Hines WW, Lee NK. System reliability allocation and a computational algorithm. IEEE Transactions on Reliability 1968; 17:49-64..

Goldberg DE (2002) Genetic algorithms in search, optimization and machine learing. Pearson Education, Singapore.

Govil A.K, "*Reliability, Availability, Maintainability for Engineers, consultants and Managers*", Parashar printers, Delhi, 1983.

Heieh YC,chen TC, ?Bricker DL. Genetic algorithm for reliability design problems. Microelectronics Reliability 1998; 38: 1599-605.

Hitt, Battelle, (1998) "Total ownership cost use in management" Digital Avionics systems Conference, Proceedings, 17th DASC. The AIAA/IEEE/SAE 31oct-7 Nov 1998, vol:1 on pages: A32-5

Ignacio J, Rosado R, Jose L, Agustin B,(2001) "*Reliability and Costs Optimization for Distribution Networks Expansion Using an Evolutionary Algorithm*", IEEE Transactions ON Power Systems, Vol.16, NO.1, February,PP-111-118.

Kuo W, Prasad VR. "An annotated overview of system-reliability optimization." IEEE Transactions on Reliability 2000; 49(2): 176-87.

Megala N, Jawahar N, (2006) "Genetic algorithm and Hope field neural network for a dynamic lot sizing problem" International Journal Advanced Manufacturing Technology 27; 1178-1191.

Michalewize Z (1996) Genetic algorithm + data structures = evolution programming. Springer, Berlin Heidelberg New York.

Nakagawa Y. and Miyazaki S.(1981), 'An experimental comparison of the heuristic methods for solving reliability optimization problems', IEEE Transactions on Reliability, Vol. 30, pp 181–184.

Rajendra Prasad .V and Kuo.W ,(2000) *"Reliability Optimization of Coherent Systems",* IEEE Transactions On Reliability, Vol.49, No.3, , PP No.323-328.

Tillman F.A, Hwang CL. Kuo W. Optimization techniques for systems reliability with redundancy – A review. IEEE Transactions on reliability 1977; R-26:148-55.

Table I. Component related data

j	c_j	s_j	r_j	X_j
1	50	50	0.75	8
2	20	100	0.70	7
3	40	150	0.60	9
4	30	100	0.80	6
5	60	200	0.75	8
Constraints				
		3000	0.95	

Table II. Component related data for additional example problems

Problem No.1					Problem No.2					Problem No.3					Problem No.4				
j	c_j	s_j	r_j	X_j	j	c_j	s_j	r_j	X_j	j	c_j	s_j	r_j	X_j	j	c_j	s_j	r_j	X_j
1	50	50	0.75	8	1	50	50	0.75	8	1	50	50	0.75	8	1	50	50	0.75	8
2	20	100	0.70	7	2	20	100	0.70	7	2	20	100	0.70	7	2	20	100	0.70	7
3	40	150	0.60	9	3	40	150	0.60	9	3	40	150	0.60	9	3	40	150	0.60	9
4	30	100	0.80	6	4	30	100	0.80	6	4	30	100	0.80	6	4	30	100	0.80	6
5	60	200	0.75	8	5	60	200	0.75	8	5	60	200	0.75	8	5	60	200	0.75	8
6	40	250	0.90	10	6	40	250	0.90	10	6	40	250	0.90	10	6	40	250	0.90	10
					7	60	100	0.85	8	7	60	100	0.85	8	7	60	100	0.85	8
										8	50	100	0.90	4	8	50	100	0.90	4
															9	60	100	0.85	8
Constraints																			
		4000	0.95				5000	0.95				6000	0.95				7000	0.95	

Problem No.5					Problem No.6					Problem No.7				
j	c_j	s_j	r_j	X_j	j	c_j	s_j	r_j	X_j	j	c_j	s_j	r_j	X_j
1	50	50	0.75	8	1	50	50	0.75	8	1	50	50	0.75	8
2	20	100	0.70	7	2	20	100	0.70	7	2	20	100	0.70	7
3	40	150	0.60	9	3	40	150	0.60	9	3	40	150	0.60	9
4	30	100	0.80	6	4	30	100	0.80	6	4	30	100	0.80	6
5	60	200	0.75	8	5	60	200	0.75	8	5	60	200	0.75	8
6	40	250	0.90	10	6	40	250	0.90	10	6	40	250	0.90	10
7	60	100	0.85	8	7	60	100	0.85	8	7	60	100	0.85	8
8	50	100	0.90	4	8	50	100	0.90	4	8	50	100	0.90	4
9	60	100	0.85	8	9	60	100	0.85	8	9	60	100	0.85	8
10	200	75	0.98	7	10	200	75	0.98	7	10	200	75	0.98	7
					11	100	50	0.96	6	11	100	50	0.96	6
										12	50	20	0.97	4
Constraints														
	7000		0.95				8000	0.95				9000	0.95	

Problem No.8					Problem No.9					Problem No.10				
j	c_j	s_j	r_j	X_j	j	c_j	s_j	r_j	X_j	j	c_j	s_j	r_j	X_j
1	50	50	0.75	8	1	50	50	0.75	8	1	50	50	0.75	8
2	20	100	0.70	7	2	20	100	0.70	7	2	20	100	0.70	7
3	40	150	0.60	9	3	40	150	0.60	9	3	40	150	0.60	9
4	30	100	0.80	6	4	30	100	0.80	6	4	30	100	0.80	6
5	60	200	0.75	8	5	60	200	0.75	8	5	60	200	0.75	8
6	40	250	0.90	10	6	40	250	0.90	10	6	40	250	0.90	10
7	60	100	0.85	8	7	60	100	0.85	8	7	60	100	0.85	8
8	50	100	0.90	4	8	50	100	0.90	4	8	50	100	0.90	4
9	60	100	0.85	8	9	60	100	0.85	8	9	60	100	0.85	8
10	200	75	0.98	7	10	200	75	0.98	7	10	200	75	0.98	7
11	100	50	0.96	6	11	100	50	0.96	6	11	100	50	0.96	6
12	50	20	0.97	4	12	50	20	0.97	4	12	50	20	0.97	4
13	60	50	0.98	5	13	60	50	0.98	5	13	60	50	0.98	5
					14	70	60	0.97	6	14	70	60	0.97	6
										15	75	55	0.96	5
Constraints														
	7000		0.95				8000	0.95				9000	0.95	

The results obtained by exhaustive search algorithm for the dataset given in Table. III

Table. III. Program output for Exhaustive search Algorithm

Iter_No	x_1	x_2	x_3	x_4	x_5	IPC	TCO	Reliability	Space	Remarks
1	1	1	1	1	1	200	40815	0.189	600	Infeasible-Optimal
2	1	1	1	1	2	260	38503	0.236	800	Infeasible
3	1	1	1	1	3	320	37968	0.248	1000	Infeasible
4	1	1	1	1	4	380	37878	0.251	1200	Infeasible
:	:	:	:	:	:	:	:	:	:	:
	2	1	8	1	1	530	30893	0.393	1700	Infeasible
	2	1	8	1	2	590	26024	0.491	1900	Infeasible
:	:	:	:	:	:	:	:	:	:	:
	3	1	3	1	1	380	31072	0.386	1000	Infeasible
	3	1	3	1	2	440	26285	0.483	1200	Infeasible
:	:	:	:	:	:	:	:	:	:	:
7189	3	3	6	5	5	900	3273	0.952	2850	Feasible
7195	3	3	6	6	4	870	3371	0.957	2750	Feasible
:	:	:	:	:	:	:	:	:	:	:
	3	3	7	1	1	580	21921	0.573	1800	Infeasible
:	:	:	:	:	:	:	:	:	:	:
	3	6	5	3	2	680	5421	0.905	2200	Infeasible
8419	*3*	*6*	*5*	*3*	*3*	*740*	*3212*	*0.950*	*2400*	*Optimal*
8420	3	6	5	3	4	800	2704	0.962	2600	Feasible
8421	3	6	5	3	5	860	2622	0.964	2800	Feasible
:	:	:	:	:	:	:	:	:	:	:
	4	1	1	5	1	470	34832	0.313	1150	Infeasible
:	:	:	:	:	:	:	:	:	:	:
10149	4	3	5	3	5	850	3321	0.950	2550	Feasible
:	:	:	:	:	:	:	:	:	:	:
11964	4	7	7	2	4	920	3387	0.950	2950	Feasible
:	:	:	:	:	:	:	:	:	:	:
	5	1	2	3	8	920	21806	0.582	2550	Infeasible
:	:	:	:	:	:	:	:	:	:	:
14139	5	5	7	4	3	990	2041	0.977	2800	Feasible
14140	*5*	*5*	*7*	*4*	*4*	*990*	*1517*	*0.989*	*3000*	*Optimal*
14141	5	5	7	4	5	1050	1431	0.992	3200	Infeasible
:	:	:	:	:	:	:	:	:	:	:
15061	5	7	8	5	5	1160	1317	0.996	3650	Infeasible-optimal
:	:	:	:	:	:	:	:	:	:	:
24190	8	7	9	6	6	1440	1480	0.999	4250	Infeasible
24191	8	7	9	6	7	1500	1531	0.999	4450	Infeasible
24192	8	7	9	6	8	1560	1588	0.999	4650	Infeasible

Optimal solution 1 infeasible region –Minimum IPC (Based on DIPC approach)

Iter_No	x_1	x_2	x_3	x_4	x_5	IPC	TCO	Reliability	Space	Remarks
1	1	1	1	1	1	200	40815	0.189	600	Infeasible-optimal

Optimal solution 2 infeasible regions – Minimum TCO (Based on DTCO approach)

Iter_No	x_1	x_2	x_3	x_4	x_5	IPC	TCO	Reliability	Space	Remarks
15061	5	7	8	5	5	1160	1317	0.996	3650	Infeasible-optimal

Optimal solution 3 feasible region –Minimum IPC (Based on DIPC approach)

Iter_No	x_1	x_2	x_3	x_4	x_5	IPC	TCO	Reliability	Space	Remarks
8419	*3*	*6*	*5*	*3*	*3*	*740*	*3212*	*0.950*	*2400*	*Optimal*

Optimal solution 4 feasible regions – Minimum TCO (Based on DTCO approach)

Iter_No	x_1	x_2	x_3	x_4	x_5	IPC	TCO	Reliability	Space	Remarks
14140	*5*	*5*	*7*	*4*	*4*	*990*	*1517*	*0.989*	*3000*	*Optimal*

Table. IV. Impact on TCO by varying parameter – Product reliability

Reliability	DIPC approach					DTCO approach				
	sequence	IPC	TCO	R	S	sequence	IPC	TCO	R	S
0.94	3-4-5-3-3	700	3523	0.9436	2200	5-5-7-4-4	990	1517	0.989	3000
0.95	3-6-5-3-3	740	3212	0.9507	2400	5-5-7-4-4	990	1517	0.989	3000
0.96	4-5-5-3-3	770	2757	0.9603	2350	5-5-7-4-4	990	1517	0.989	3000
0.97	4-5-5-3-4	830	2243	0.9718	2550	5-5-7-4-4	990	1517	0.989	3000
0.98	4-5-6-4-4	900	1695	0.9841	2800	5-5-7-4-4	990	1517	0.989	3000

Table. V. Impact on TCO by varying parameter – Downtime cost/yr

Downtime cost/yr	DIPC approach					DTCO approach				
	sequence	IPC	TCO	R	S	sequence	IPC	TCO	R	S
1000	3-6-5-3-3	740	994	0.950	2400	4-5-5-4-3	800	973	0.966	2450
2000	3-6-5-3-3	740	1241	0.950	2400	4-5-6-4-4	900	1061	0.984	2800
3000	3-6-5-3-3	740	1487	0.950	2400	4-6-6-4-4	920	1135	0.985	2900
4000	3-6-5-3-3	740	1733	0.950	2400	5-6-6-4-4	970	1197	0.988	2950
5000	3-6-5-3-3	740	1980	0.950	2400	5-6-6-4-4	970	1253	0.988	2950

Table. VI. Impact on TCO by varying parameter – Product Life

Product Life in year	DIPC approach					DTCO approach				
	sequence	IPC	TCO	R	S	sequence	IPC	TCO	R	S
1	3-6-5-3-3	740	1241	0.950	2400	4-5-6-4-4	900	1061	0.984	2800
2	3-6-5-3-3	740	1733	0.950	2400	5-6-6-4-4	970	1197	0.988	2950
3	3-6-5-3-3	740	2226	0.950	2400	5-5-7-4-4	990	1307	0.989	3000
4	3-6-5-3-3	740	2719	0.950	2400	5-5-7-4-4	990	1412	0.989	3000
5	3-6-5-3-3	740	3212	0.950	2400	5-5-7-4-4	990	1517	0.989	3000

Table. VII. Impact on TCO by varying parameter – Number of elements (size of the problem)

m	DIPC approach					DTCO approach					Number of solutions
	sequence	IPC	TCO	R	S	sequence	IPC	TCO	R	S	
6	455332	850	3318	0.950	2850	566553	1180	1585	0.991	4000	241920
7	3554333	1050	3506	0.950	3450	5685544	1540	1753	0.995	4950	1935360
8	35543333	1200	3704	0.950	3750	57855443	1710	1948	0.995	5350	7741440
9	455333333	1400	3807	0.952	4000	578554434	1950	2213	0.994	5750	61931520

Table. VIII. GA parameter for Method I & Method II

GA parameter	Method -I	Method -II
Chromosome length	chr_len = m	chr_len = 4*m
Chromosome encoding	Binary coding	Value coding
Population size	pop_size = 5* m	pop_size = 5*m
Selection operator	Roulette wheel	Roulette wheel
Crossover operator	Single point crossover with randomization	Single point crossover with randomization
Probability of crossover	p_cross = 0.6	p_cross = 0.6
Mutation operator	Random Exchange	Random Exchange
Probability of mutation	p_mut = 0.06	p_mut = 0.06
Constant 'k'	k = 0.005	k = 0.005
Number of generations	it_no = 20 *m	it_no = 20 *m

Table IX Procedural steps of new population generation - Method -I

Initialization _Modules ( )	c	1	2	3	4….24	25
	j	1234567891011121314151617181920	1234567891011121314151617181920	1234567891011121314151617181920	---	1234567891011121314151617181920
	x_j	0 1 1 0 1 1 0 1 0 1 0 1 1 1 1 1 0 0 1 0 1	1 0 0 1 1 0 0 0 0 1 1 0 1 0 0 1 1 0 1 1	0 1 1 1 1 0 1 1 1 1 0 1 1 0 0 1 0 1 0 0		1 1 1 1 1 0 1 1 1 0 0 0 1 1 0 1 0 1 0 0
	Decode	4 7 4 6 3	5 4 4 4 6	4 6 8 4 3		8 6 5 6 3
Evaluation_mod ( )	R(c)	0.955158	0.963783	0.977606		0.973508
	S(c)	2700	2850	3000		2950.
	TCO(c)	2213.06	2061.83	1615.92		1879.77
	fit(c)	2213.06	2061.83	1615.92		1879.77
Selection	New_fit(c)	0.000				
	p(c)	0.007	0.014	0.133		0.035
	cp(c)	0.007	0.021	0.154		1.000
	R1	0.539	0.077	0.182		0.08
	Old (c) / New (c')	(15) 1'	(3)2'	(5)3'		(16) 25'
New_pop_gen_mod( )	c'	0 1 1 0 1 1 0 1 1 0 1 0 0 1 0 1 0 1 0 1	0 1 1 1 1 0 1 1 1 1 0 1 1 0 0 1 0 1 0 0	0 1 1 0 1 1 1 0 0 1 1 1 0 1 1 0 1 0 0 1		1 0 1 1 1 0 0 1 1 0 0 1 0 1 0 1 0 1 1 1
	R2	0.07	0.68	0.14		0.75
	Selected Chrom c'	Selected	Not Selected	Selected		Not Selected
Crossover	Cut points	7	--	7		--
	after cross over c''	0 1 1 0 1 1 0 0 0 1 1 1 0 1 1 0 1 0 0 1	--	0 1 1 0 1 1 1 1 1 0 1 0 0 1 0 1 0 1 0 1		--
	R3	.07 .05 .03 .08 .07 .03 .09 .06 .05 .04	.09 .05 .03 .02 .08 .07 .03 .07 .09 .03			
Mutation	after mutation c'''	0 0 0 0 1 0 0 1 1 0	1 0 1 0 1 0 0 0 0 0			

Table X. Procedural steps of new population generation - Method –II

<table>
<tr><td colspan="2" rowspan="3">Initialisation_Modules ( )</td><td>c</td><td colspan="5">1</td><td colspan="5">2</td><td colspan="5">3</td><td>4..9</td><td colspan="5">10</td><td colspan="5">11</td><td>12..18</td><td colspan="5">24</td><td colspan="5">25</td></tr>
<tr><td>j</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>---</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td></td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td></tr>
<tr><td>x_j</td><td>7</td><td>6</td><td>7</td><td>1</td><td>4</td><td>3</td><td>1</td><td>7</td><td>3</td><td>7</td><td>1</td><td>6</td><td>2</td><td>3</td><td>1</td><td></td><td>3</td><td>4</td><td>8</td><td>4</td><td>2</td><td>5</td><td>3</td><td>7</td><td>3</td><td>4</td><td></td><td>6</td><td>6</td><td>6</td><td>3</td><td>7</td><td>5</td><td>1</td><td>2</td><td>2</td><td>5</td></tr>
<tr><td colspan="2" rowspan="4">Evaluation_mod ( )</td><td>R(c)</td><td colspan="5">0.7949</td><td colspan="5">0.6823</td><td colspan="5">0.468</td><td></td><td colspan="5">0.9892</td><td colspan="5">0.9916</td><td></td><td colspan="5">0.9892</td><td colspan="5">0.9916</td></tr>
<tr><td>S(c)</td><td colspan="5">2900</td><td colspan="5">3000</td><td colspan="5">1450</td><td></td><td colspan="5">3200</td><td colspan="5">3200</td><td></td><td colspan="5">3200</td><td colspan="5">3200</td></tr>
<tr><td>TCO(c)</td><td colspan="5">1128</td><td colspan="5">1685</td><td colspan="5">2702</td><td></td><td colspan="5">1539</td><td colspan="5">1538</td><td></td><td colspan="5">1539</td><td colspan="5">1538</td></tr>
<tr><td>fit(c)</td><td colspan="5">56405</td><td colspan="5">84256</td><td colspan="5">135137</td><td></td><td colspan="5">7699</td><td colspan="5">7691</td><td></td><td colspan="5">7699</td><td colspan="5">7691</td></tr>
<tr><td rowspan="12">New_pop_gen_mod( )</td><td rowspan="8">Selection</td><td>New_fit(c)</td><td colspan="5">0.000000</td><td colspan="5">0.000000</td><td colspan="5">0.000000</td><td></td><td colspan="5">0.002457</td><td colspan="5">0.001828</td><td></td><td colspan="5">0.000003</td><td colspan="5">0.000000</td></tr>
<tr><td>p(c)</td><td colspan="5">0.037</td><td colspan="5">0.037</td><td colspan="5">0.085</td><td></td><td colspan="5">0.037</td><td colspan="5">0.038</td><td></td><td colspan="5">0.037</td><td colspan="5">0.038</td></tr>
<tr><td>cp(c)</td><td colspan="5">0.000</td><td colspan="5">0.074</td><td colspan="5">0.159</td><td></td><td colspan="5">0.430</td><td colspan="5">0.468</td><td></td><td colspan="5">0.0961</td><td colspan="5">1.000000</td></tr>
<tr><td>R1</td><td colspan="5">0.64</td><td colspan="5">0.96</td><td colspan="5">0.49</td><td></td><td colspan="5">0.08</td><td colspan="5">0.40</td><td></td><td colspan="5">0.42</td><td colspan="5">0.21</td></tr>
<tr><td>Old (c) / New (c’)</td><td colspan="5">(11) 1′</td><td colspan="5">(10) 2′</td><td colspan="5">(7) 3′</td><td></td><td colspan="5">(11)10’</td><td colspan="5">(17)11’</td><td></td><td colspan="5">(17)24′</td><td colspan="5">(10)20′</td></tr>
<tr><td></td><td>5</td><td>3</td><td>7</td><td>3</td><td>4</td><td>3</td><td>4</td><td>8</td><td>4</td><td>2</td><td>1</td><td>6</td><td>2</td><td>3</td><td>1</td><td></td><td>5</td><td>3</td><td>7</td><td>3</td><td>4</td><td>6</td><td>3</td><td>7</td><td>3</td><td>2</td><td></td><td>6</td><td>3</td><td>7</td><td>3</td><td>2</td><td>5</td><td>3</td><td>7</td><td>3</td><td>4</td></tr>
<tr><td>R2</td><td colspan="5">0.07</td><td colspan="5">0.68</td><td colspan="5">0.14</td><td></td><td colspan="5">0.25</td><td colspan="5">0.34</td><td></td><td colspan="5">0.78</td><td colspan="5">0.92</td></tr>
<tr><td>Selected Chrom c’</td><td colspan="5">Selected</td><td colspan="5">Not Selected</td><td colspan="5">Selected</td><td></td><td colspan="5">Selected</td><td colspan="5">Selected</td><td></td><td colspan="5">Not Selected</td><td colspan="5">Not Selected</td></tr>
<tr><td rowspan="3">Crossover</td><td>Cut points</td><td colspan="5">2</td><td colspan="5">--</td><td colspan="5">2</td><td></td><td colspan="5">4</td><td colspan="5">4</td><td></td><td colspan="5">--</td><td colspan="5">--</td></tr>
<tr><td>Chrom after cross over c’’</td><td>7</td><td>6</td><td>2</td><td>3</td><td>1</td><td>3</td><td>1</td><td>7</td><td>3</td><td>7</td><td>1</td><td>6</td><td>7</td><td>3</td><td>4</td><td></td><td>5</td><td>3</td><td>7</td><td>3</td><td>2</td><td>6</td><td>3</td><td>7</td><td>3</td><td>4</td><td></td><td>6</td><td>3</td><td>7</td><td>3</td><td>2</td><td>5</td><td>3</td><td>7</td><td>3</td><td>4</td></tr>
<tr><td>R3</td><td>.07</td><td>.04</td><td>.03</td><td>.08</td><td>.09</td><td>.06</td><td>.05</td><td>.07</td><td>.06</td><td>.01</td><td>.02</td><td>.07</td><td>.01</td><td>.03</td><td>.09</td><td></td><td>.04</td><td>.02</td><td>.09</td><td>.07</td><td>.05</td><td>.04</td><td>.05</td><td>.08</td><td>.04</td><td>.09</td><td></td><td>.04</td><td>.01</td><td>.05</td><td>.09</td><td>.01</td><td>.05</td><td>.08</td><td>.05</td><td>.01</td><td>.09</td></tr>
<tr><td>Mutation</td><td>Chrom after mutation c’’’</td><td>7</td><td>5</td><td>3</td><td>3</td><td>1</td><td>4</td><td>3</td><td>7</td><td>5</td><td>1</td><td>2</td><td>6</td><td>6</td><td>3</td><td>4</td><td></td><td>2</td><td>1</td><td>7</td><td>3</td><td>3</td><td>2</td><td>1</td><td>7</td><td>5</td><td>4</td><td></td><td>3</td><td>4</td><td>2</td><td>3</td><td>4</td><td>4</td><td>3</td><td>6</td><td>1</td><td>4</td></tr>
</table>

Table. XI. Performance comparison of GA

Problem No	***ESA Optimal TCO***	***GA Method-I***		***% of Deviation from optimal***	***GA Method-II***		***% of Deviation from optimal***
		100 iterations	*1000 iterations*		*100 iterations*	*1000 iterations*	
1.	1585	1585	1585	*0*	1700		*0.06*
2.	1753	1753	1753	*0*	1791		*0.02*
3.	1948	1948	1948	*0*	2096		*0.07*
4.	2213	2213	2213	*0*	2467		*0.10*
5.	2674	2674	2674	*0*	2731		*0.02*
6.	2954	2968	2954	*0*	3174		*0.06*
7.	3099	3109	3099	*0*	3629		*0.14*
8.	3239	3264	3243	*0.02*	4175		*0.22*
9.	3423	3444	3423	*0*	4602		*0.25*
10.	*3652*	3652	3652	*0*	4635		*0.21*