Redundancy-Allocation in Neel Metal Products Limited

Sarita Devi* and Deepika Garg

G. D. Goenka University, Gurgaon - 122103, Haryana, India; Sarita.fame@gmail.com, deepikanit@yahoo.com

Abstract

Objective: In manufacturing setup, limited budget is allocated for each system. In order to increase reliability of the system, redundancy is allocated within given cost constraints. Objective of this paper is to come out with the best optimal solution and increase their liability of a system under cost constraint in a manufacturing plant. Methods: In this paper Heuristic Algorithm (HA) and Constrained Optimization Genetic Algorithm (COGA) are used to optimize constrained Redundancy Allocation Problem (RAP) in a manufacturing plant. These methods are used to allocate the best redundancy strategy for each subsystem with a view to increase the reliability of the system under cost constraints. Best optimal solution is reached by comparing results of CPU time taken by these two methods. Findings: Generally RAP is a NP hard problem and a non-linear integer programming problem, which is difficult to solve. Both methods are applied and comparison between reliability of both the methods is made on the basis of which result obtained by COGA is 0.8632 which is found better against HA which is 0.8380. Application: Results applied in the manufacturing plant which resulted in the increase of reliability by using best redundancy strategy.

Keywords: COGA, HA, Optimization, Reliability, Redundancy

1. Introduction

Reliability engineering aims to achieve the required and desired system reliability level. In the field of engineering, redundancy allocation problem is an important and complex mathematical programming problem. It is very recent and interesting topic for researchers and engineers1–5. It has attended more attention from the research community. RAP includes selection of elements and level of redundancy to increase the reliability of the system. Redundancy is allocated to increase the reliability of the system with certain constraints imposed such as cost, weight, power requirements etc. Reliability of a system can be enhanced by allocating proper redundancies in the subsystems of the system6. In the system reliability field, it has been noticed that a wide variety of single objective optimization techniques such as integer programming, Meta-heuristic algorithms, nonlinear programming, mixed integer and dynamic programming7–14 are applied to solve variety of problems. Heuristic methods are approximate methods for the optimum allocation of redundancies15, but some problem emerges with these methods while solving RAP so other algorithms such as Genetic Algorithms (GAs) are proposed for reduction in these difficulties.

GA is one of the broadly used technique of optimization and is used to get the minimum or maximum of a function rooted on the doctrine of genetics and natural selection. The group of all the solutions of an optimization problem constitutes the search space. The problem consists in finding out the best fit solution from all the possible solutions. When the search space becomes large, we need a specific method like GA to find the optimal solution16,17 explained that GAs uses fitness function for evaluation rather than derivatives. GA is inspired by the principles of population genetics and natural selection. It simulates a group or population of individuals considered as likely solutions to the available problem which

*Author for correspondence
mate, generate offspring, convert and develop into better and superior individuals\textsuperscript{17–19}. In various areas together with reliability optimization problems, more attention has been attracted to GAs over other heuristic methods. For instance, Painton and Campbell\textsuperscript{20–26} have reported efficient and effective solutions by using GAs in integer reliability problems. Authors in\textsuperscript{27–30} did repetition of selection, crossover and mutation until satisfaction of termination condition.

In this paper, RAP is solved by using first using HA and then COGA to find out the most optimal solution.

2. System Description

The Neel Metal Products Limited consists of various units viz. Overhead Crane, Roller, Blanking Machine, Stacker Machine, Press Machine, Molding and Packing. These subsystems are arranged in series. Initially Steel roll is rolled using roller connected to the blanking machine. Blanking machine cuts the blanks and then the blanks are stacked with the help of stacker. Then blanks are pressed for desired shape using press shop and molded as per desired design. Thereafter, final product gets ready for packing.

2.1 Assumptions

Coherence: Property of a system or component, as defined by\textsuperscript{4}.

Path Set: A set of subsystem such that, if all the subsystems in the system operate, the system is guaranteed to operate.

Minimal path set: A path set such that, if any subsystem is removed from the set, remaining subsystem no longer form a path.

2.2 Notations

| $y_i$ | i	extsuperscript{th} Subsystem |
| $R_i(y_i)$, $Q_i(y_i)$ | Reliability, unreliability of subsystem-$y_i$ |
| $R(y)$ | System reliability |
| $n_i$ | number of i	extsuperscript{th} subsystems |

$\Delta R_i$ Difference in reliability of i	extsuperscript{th} subsystem by adding one more redundant component

$h_i(y_i)$ $i$th resource consumed by i	extsuperscript{th} subsystem

$P_1$ minimal path set of the system

$n = 7$ number of subsystems

$l$ number of constraints

$f(.)$ a function that yields the system reliability, based on unique subsystems, and which depends on the configuration of the subsystems

2.3 Problem Formulation

In case of manufacturing plant, problem is to maximize reliability with cost constraint as defined in Table 1. Considering cost constraints $C = 30490000$ (Here number of constraints is one).

Problem is to maximize

$$R_s(y) = f(R_1(y_1), \ldots, R_7(y_7)) = \prod_{i=0}^{7} R_i(y_i)$$

$$\sum_{i=1}^{7} h_i(y_i) \cdot n_i \leq 30490000$$

2.4 Methodology

Methodology of HA and COGA is illustrated below:

HA. A practical method that not guaranteed to give exact optimal solution but near about optimal solution and sufficient for the immediate goal.

2.4.1 Algorithm

1. Assign $y_i = 1$, for $i = 1, 2, \ldots, 7$
2. Now find the subsystem which is most unreliable. Add one redundant component to that subsystem.
3. Check the constraints:
   - If any constraint is contravened, go to Step 4.
   - If no constraint has been contravened, go to Step 2.
   - If any constraint is exactly satisfied then stop.
   - The current $y_i$'s are the optimum values for the system.

| Table 1. Reliability cost of each subsystem |
|---|
| **Subsystem** | $y_1$ | $y_2$ | $y_3$ | $y_4$ | $y_5$ | $y_6$ | $y_7$ |
| Reliability of subsystem | 0.99 | 0.9762 | 0.9188 | 0.8155 | 0.8655 | 0.9287 | 0.9453 |
| Cost of subsystem | 1280000 | 960000 | 2500000 | 1050000 | 1050000 | 950000 | 250000 |
Table 2. Steps in HA

| S.No. | Number of components in each subsystem | Consumed Resources | Subsystem selection factor |
|-------|---------------------------------------|--------------------|----------------------------|
|       | \( n_1, n_2, n_3, n_4, n_5, n_6 \) |                    | \( \sum_{i=1}^{7} K_i(y_i) \cdot n_i \) | \( S_1, S_2, S_3, S_4, S_5, S_6 \) |
| 1     | 1 1 1 1 1 1                             | 17490000           | 0.99 0.9762 0.9188 0.8155 0.8655 0.9287 0.9453 |
| 2     | 1 1 1 2 1 1                             | 18540000           | 0.99 0.9762 0.9188 0.9659 0.8688 0.9287 0.9453 |
| 3     | 1 1 1 2 2 1                             | 29040000           | 0.99 0.9762 0.9188 0.9659 0.9827 0.9287 0.9453 |
| 4     | 1 1 2 2 2 1                             | 31540000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 5     | 1 1 1 2 2 2                             | 29990000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 6     | 1 1 1 2 2 2                             | 30240000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 7     | 1 1 1 3 2 2                             | 31290000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 8     | 1 2 1 2 2 2                             | 31200000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 9     | 1 1 1 2 3 2                             | 40740000⁷           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 10    | 1 1 1 2 2 3                             | 31190000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 11    | 1 1 1 2 2 3                             | 30490000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 12    | 2 1 1 2 2 3                             | 31770000⁷           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |
| 13    | 1 1 1 2 2 3                             | 30490000           | 0.99 0.9762 0.9659 0.9827 0.9287 0.9453 |

ALGORITHM STOPS HERE

1. Now we will remove the redundant component added in Step 2. The resulting number is the optimum allocation for that subsystem. Remove this subsystem from further consideration.

2. If all the subsystems have been \( y_i \)'s are the optimum values for the system; otherwise go to Step 2.

2.4.2 Implementation

The steps which follows the algorithm is shown in Table 2.

2.4.3 Results

Table 2 depicts the steps of the algorithm which is evaluated by HA. The optimal solution obtained by HA is shown in Table 3.

2.5 COGA

The new approach in GAs community is to solve numerical optimization problems with constraints based on GAs.

Introducing the concept of constraints can be beneficial and improve the behavior of the methods by limiting the space to be searched.

Table 3. HA result

| \( n_1 \) | \( n_2 \) | \( n_3 \) | \( n_4 \) | \( n_5 \) | \( n_6 \) |
|---------|---------|---------|---------|---------|---------|
| 1       | 1       | 1       | 2       | 2       | 2       |

The units of the \( n_1, n_2, \ldots, n_6 \) are used as 1, 1, …, 3 respectively

2.5.1 Algorithm

- Set the parameters (Population size, maximum number of generations, crossover rate, and mutation rate) of GAs, bounds of variables.
- Generate initial population of \( n \) chromosomes (\( p[i] \) represents the population at \( i^{th} \) generation).
- Set \( i=0 \) [represent the initial generation]
2.5.2 Results of COGA

The result of the same RAP is also evaluated by COGA. This algorithm is programmed using MATLAB.

The optimal solution obtained by COGA is shown in Table 4.

| $n_1$ | $n_2$ | $n_3$ | $n_4$ | $n_5$ | $n_6$ | $n_7$ |
|-------|-------|-------|-------|-------|-------|-------|
| 2     | 2     | 3     | 4     | 1     | 3     | 3     |

The units of $n_1, n_2, \ldots, n_7$ are used 2, 2, \ldots, 3 respectively.

3. Conclusion

Reliability of the system obtained by HA’s optimal solution = 0.8380; whereas reliability of the system obtained by COGA’s optimal solution = 0.8632. The CPU time taken by HA and COGAs are 50.66 seconds and 66.511 seconds.

Two methods, i.e., HA and COGA have been proposed in this paper to solve the RAP. The result shows that Heuristic methods are faster but gives only nearly optimal solution in comparison to the result obtained by COGA which is better, as we can use more units of each subsystem in COGA.

4. References

1. Ghare PM, Taylor RE. Optimal redundancy for reliability in series system. Operation Research. 1969; 17(5):838–47. Crossref
2. Tillman FA, Hwang CL, Kuo W. Optimization of system reliability. New York: Marcel Dekker; 1980. PMCid:PMC1272065.
3. Chern MS. On the computational complexity of reliability redundancy allocation in series system. Operation Research Letter. 1992; 11(5):309–15. Crossref
4. Ardaken MA, Hamadani AZ. Reliability redundancy allocation problem with cold-standby redundancy strategy. Simulation Modelling Practice and Theory. 2014; 42(3):107–18. Crossref
5. Coelho LS. Reliability-redundancy optimization by means of a chaotic differential evolution approach. Chaos, Solitons and Fractals. 2009; 41(2):594–602. Crossref
6. Kim JH, Yum BJ. A heuristic method for solving redundancy optimization problems in complex systems. IEEE Transactions on Reliability. 1993; 42(4):572–8. Crossref
7. Kuo W, Prasad VR. An annotated overview of system-reliability optimization. IEEE Transactions on Reliability. 2000; 49(2):176–87. Crossref
8. Kuo W, Wan R. Recent advances in optimal reliability allocation. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans. 2007; 37:143–56. Crossref
9. Deeter DL, Smith AE. Economic design of reliable network. IEEE Transactions on Reliability. 1998; 30:1161–74. Crossref
10. Ravi V, Murty BSN, Reddy PJJ. Nonequilibrium simulated annealing algorithm applied to reliability optimization of complex systems. IEEE Transactions on Reliability. 1997; 46:233–9. Crossref
11. Ramirez MJE, Coit DW, Konak A. Redundancy allocation for series-parallel systems using a max-min approach. IIE Transactions. 2004; 36(2):891–8. Crossref
12. Liang YC, Smith AE. An ant colony optimization algorithm for the redundancy allocation problem. IEEE Transactions on Reliability. 2004; 53(3):417–23. Crossref
13. Onishi J, Kimura S, James RJW, Nakagawa Y. Solving the redundancy allocation problem with a mix of components.
using the improved surrogate constraint method. IEEE Transactions on Reliability. 2007; 56(1):94–101. Crossref
14. Billionnet A. Redundancy allocation for series-parallel systems using integer linear programming. IEEE Transactions on Reliability. 2008; 57(3):507–16. Crossref
15. Garg D, Kumar K, Pahuja GL. Redundancy-allocation in pharmaceutical plant. International Journal of Engineering Science and Technology. 2010; 2(5):1088–97.
16. Holland JH. Adaptation in natural and artificial systems. Cambridge: MIT Press; 1975.
17. Sivanandam SN, Deepa SN. Introduction to genetic algorithms. India: Springer-Verlag; 2008.
18. Kinnear KE. A perspective on the work in this book. Advances in Genetic Programming. Cambridge: MIT Press; 1994. PMCID:PMC1238770.
19. Goldberg DE. Genetic algorithms in search, optimization, and machine learning. Reading: Addison-Wesley; 1989.
20. Gen M, Yokota T, Taguchi T. Optimal design of system reliability using genetic algorithm. Proc of 10th Fuzzy Sys Symp; 1994. p. 181–4.
21. Coit DW, Smith AE. Reliability optimization of series-parallel systems using a genetic algorithm. IEEE Transaction on Reliability. 1996; 45(2):254–60. Crossref
22. Coit DW, Smith AE. Use of a genetic algorithm to optimize a combinatorial reliability problem. Proceedings of 3rd IIE Residual Conference; 1995. p. 342–9.
23. Coit DW, Smith AE. Penalty guided genetic search for reliability design optimization. Computers and Industrial Engineering. 1996; 30(4):895–904. Crossref
24. Ida K, Gen M, Yokata I. System reliability optimization with several failure modes by genetic algorithm. Proceedings of 16th International Conference on Camp and Indus Engng; 1994. p. 348–52.
25. Painton L, Campbell J. Identification of components to optimize improvements in system reliability. Proc of the SRA PSAM-II Conf on System-Based Methods for the Design and Operation of Technological Systems and Processes; 1994. p. 10–20.
26. Painton L, Campbell J. Genetic algorithms in optimization of system reliability. IEEE Transaction on Reliability. 1995; 44:172–8. Crossref
27. Holland JH. Genetic algorithm and the optimal allocation of trials. SIAM Journal on Computing. 1973; 2(2):88–105. Crossref
28. Jong KE. De. An analysis of the behavior of a class of genetic adaptive systems [PhD thesis] Univ of Michigan; 1975.
29. Davis L. Genetic algorithms and simulated annealing. Los Altos, CA: Morgan Kaufman Publishers Inc; 1987.
30. Liepins GE, Hilliard MR. Genetic algorithms: Foundations and applications. Annals of Operations Research. 1989; 21:31–58. Crossref