Journal of Applied Science and Agriculture, 9(4) April 2014, Pages: 1359-1365

AENSI Journals

Journal of Applied Science and Agriculture

ISSN 1816-9112

Journal home page: www.aensiweb.com/jasa/index.html

# Stochastic Optimization Techniques to the Redundancy Allocation Problem for Series Systems

[1]Farshad Kiyoumarsi and [2]Narges Asgharian

[1,2]*Islamic Azad University-Shahrekord Branch, Engineering Department, Faculty of Computer, Shahrekord, Iran.*

A R T I C L E  I N F O

*Article history:*
*Received 20 January 2014*
*Received in revised form 16*
*15 April 2014*
*Accepted 25 April November 2014*
*Available online 5 May 2014*

*Key words:*
*Stochastic Optimization techniques,*
*Redundancy allocation, Genetic algorithm.*

A B S T R A C T

**Background:** Various reliability (Barlow, R.E., Proschan, F., 1975) optimization approaches have been suggested in the past three decades (Tillman, F.A.,Hwang, C.L. and Kuo, W, 1977). Stochastic programming models for general redundancy-optimization problems have been studied by Zhao *et al* (Zhao, R. and Liu, B., 2003). Stochastic programming models pose as reformulations or extensions of reliability optimization problems with chance parameters **Objective:** This paper aims to maximize system reliability for the given chance constraints. To achieve this end, a method is explained to determine optimal solutions to an *n*-stage series system with *m* chance constraints of the redundancy allocation problem. This method transforms the constrained optimization problem into an unconstrained one by penalizing the objective function corresponding to the infeasible solution and then explain the model by deriving the required algorithm to obtain an integer solution along with two numerical example. To convert the above problem to an unconstrained maximization problem, a large negative value (say, −M) is blindly assigned to the objective function for the infeasible solution. **Results:** This paper explains the use of GA to analyze this problem. The GA was examined on two problems and compared with the corresponding results from DP and B&B. GA, as powerful and broadly applicable stochastic search and optimization technique, is the most widely known type of evolutionary computational methods. GA maintains a population of chromosomes. Each chromosome represents a potential solution to the problem at hand and is evaluated to give some measure of its fitness. A new population is formed by selecting the more fitable chromosomes from the parent and offspring populations. After several generations, the algorithm converges to the best chromosome, which represents an optimal or sub-optimal solutions to the problem. One of the distinguishing features of GA is to work with a population of candidate solutions. **Conclusion:** since GA's concept and theory are simple, and GA's ability to search optimal solution is excellent; Therefore, GA approach can be applied to many engineering optimization problems as well as decision making problems in various fields.



**To Cite This Article:** Farshad Kiyoumarsi and Narges Asgharian., Stochastic Optimization Techniques to the Redundancy Allocation Problem for Series Systems. ***J. Appl. Sci. & Agric.***, *9(4): 1359-1365, 2014*

## INTRODUCTION

Chern (Chern,M.S,1992) has proved that even the simplest redundancy allocation problems, a series system with one constraint or a series system with identical components two constraints, are NP-hard. Due to its difficulty, various approaches, such as heuristics and enumerations, have been considered for solving redundancy allocation problems (Kuo, W, Prasad, V.R, Tillman, F.A, Hwang, C.L , 2001).

Various reliability (Barlow, R.E., Proschan, F., 1975) optimization approaches have been suggested in the past three decades (Tillman, F.A., Hwang, C.L. and Kuo, W, 1977). Stochastic programming models for general redundancy-optimization problems have been studied by Zhao *et al* (Zhao, R. and Liu, B., 2003). Stochastic programming models pose as reformulations or extensions of reliability optimization problems with chance parameters. This paper deals with the chance constraints reliability stochastic optimization problem. The purpose is to maximize system reliability for the given chance constraints. A method is explained to determine optimal solutions to an *n*-stage series system with *m* chance constraints of the redundancy allocation problem. Various cases of stochastic with known distributions, such as uniform, normal when the resource variables are random, have been investigated. Once the real number solution is obtained using the technique of chance constraints, the

**Corresponding Author:** Farshad Kiyoumarsi, Islamic Azad University-Shahrekord Branch, Engineering Department, Faculty of Computer, Shahrekord, Iran.
Phone: +98 381 3361078; E-mail: kumarci_farshad@yahoo.com

**Journal of Applied Science and Agriculture, 9(4) April 2014, Pages: 1359-1365**

branch-and-bound (B&B) method, dynamic programming (DP) method, genetic algorithm (GA) method is used to obtain the integer solution. We explain this approach for two 4-stage series system with two chance constraints.

In this work, we deal with the stochastic integer programming problem for *n*-stage series system with *m* chance constraints. Firs we change these problems to constrained equivalent ones and then use optimization method.

*Methodology:*

*Stochastic optimization: n-stage series system with m chance constraints:*

The chance constrained optimization problem for a *n*-stage series system with m chance constraints can be formulated as

$$\text{Max } R_s(x) = \prod_{j=1}^{n} [1-(1-r_j)^{x_j}] \quad (1)$$

s.t

$\Pr(g_i(x) \leq c_i) \geq 1-\alpha_i$, i=1,2,..,m

$x_j \geq 1$, j=1,2,…,n.

where resource vector $c_i$ is random; $R_S$ is the reliability of the system; $r_j$ is reliability of components *j*; $x_j$ is the number of components used at stage *j*; $g_i(x)$ is the chance constraint *i*; $c_i$ is the amount of resource *i* available (random), and $\alpha_i$ is the level of significance.

Definition 1: A random variable X is said to have a uniform distribution if its probability density function is given by

$$f(X) = \begin{cases} \frac{1}{b-a} & a < x < b \\ 0 & \text{otherwise} \end{cases}$$

When a random variable X is uniformly distributed, we shall express it as X ~ U (a, b).

Definition 2: A random variable X is said to have a normal distribution with parameters $\mu$ (mean) and $\sigma 2$(variance) if its probability density function is given by

$$f(X; \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2\right);$$

$$-\infty < x, \mu < \infty, \sigma > 0$$

When a random variable X is normally distributed with mean $\mu$ and standard deviation $\sigma^2$, we shall express it as X~ N($\mu$, $\sigma^2$).

*Case 1. $c_i$ is uniformly distributed:*

Let $c_i: U(l_i, u_i)$, the constraint in system (1) is equivalent to $g_i(x) \leq \tau_i$, where $\beta_i = 1 - \alpha_i$, $\int_{\tau_i}^{u_i} [\frac{dx}{u_i - l_i}) = \beta_i$, i.e. $\tau_i = \alpha_i u_i + \beta_i l_i$. Hence, the deterministic equivalent of system (1) is

$$\text{Max } R_s(x) = \prod_{j=1}^{n} [1-(1-r_j)^{x_j}]$$

s.t $g_i(x) \leq \tau_i$, i=1,2,..,m

$x_j \geq 1$, j=1,2,…,n.

*Case 2. $c_i$ is normally distributed:*

Let $c_i: N(\mu_{c_i}, \sigma^2_{c_i})$ where $\mu_{c_i}$ and $\sigma^2_{c_i}$ are mean and variance of the normal random variable $c_i$ Using the *i*th chance constraint of the system (1), restate the chance constraint as

$Pr(c_i \geq g_i(x)) \geq 1 - \alpha_i$, $i = 1, 2, ..., m$, so can be

written as $g_i(x) \leq \mu_{c_i} + e_i \sigma_{c_i}$ where $e_i$ is the value of the standard normal differ for which $\varphi(e_i) = \alpha_i$.

$$\varphi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} exp(-\frac{t}{2})\, dt$$

Hence, the deterministic equivalent of system (1) is

$$\text{Max } R_s(x) = \prod_{j=1}^{n} [1-(1-r_j)^{x_j}]$$

s.t

$g_i(x) \leq \mu_{c_i} + e_i \sigma_{c_i}$, i=1,2,..,m

**Journal of Applied Science and Agriculture, 9(4) April 2014, Pages: 1359-1365**

$x_j \geq 1$, j=1,2,…,n.

*Different Stochastic Optimization Approaches:*

The most used global optimization method in redundancy allocation problems is dynamic programming. Dynamic programming solutions to the redundancy allocation problem are presented in Bellman and Dreyfus (Bellman, R.E., Dreyfus, E.,1962), Fyffe, Hines and Lee (Fyffe, D. E., Hines, W. W., Lee, N. K, 1968) and Nakagawa and Miyazaki (Nakagawa, Y, Miyazaki, S, 1981). The Fyffe, Hines and Lee formulation uses a Lagrangian multiplier ($\lambda$) within the objective function to reduce the number of problem constraints to one, and alternatively, the Nakagawa and Miyazaki formulation uses a surrogate constraint to combine the constraints into one. The implementation of dynamic programming, however, is limited by the number of constraints and the system structures it can be applied to. For a system which has more than two constraints, the computational complexity of dynamic programming increases exponentially. Dynamic programming is still not applicable to nonseparable systems such as reliability optimization problems with complex structures.

*Branch-and-Bound Technique:*

Due to their flexibility and their optimality properties, branch-and-bound methods have been extensively used in mathematical programming (Ha,C, Kuo,W, 2006). The general branch-and-bound method for maximizing integer nonlinear programming (INLP) problems is based on the following procedures (Sun, X, Li Duan,D,2002):

(1) to find a feasible solution and set the current optimum as the value of the initial feasible solution;

(2) if there is no unsolved sub-problem, terminate the procedure;

(3) branch into sub-problem where a decision a variable is fixed or bounded;

(4) apply relaxation for each sub-problem and solve the problem, then, update the upper bound as the maximum value of the solutions;

(5) fathom this tree node if the upper bound is less than the current optimum;

(6) if the upper bound is feasible, update the current optimal solution and go to the next tree node; otherwise branch into the sub-problems again.

*Genetic Algorithm Implementation:*

A GA is a stochastic optimization technique patterned after natural selection in biological evolution as initially described by Holland (Holland, J., 1975). Genetic algorithms produce a complete population of answering points. Each point is tested separately and to establish new populations, including modified points, existing points merits could be tested (Dehini,R, Ferdi,B,and Bekkouche,B,2012, Farshad Kyoomarsi,2008). The equivalent deterministic constraint optimization problem for a series system is a non-linear constrained integer programming problem. During the last few decades, several methods were proposed for handling constraints by GA (Deb, K., 2000, Sakawa,M,2002, F Kiyoumarsi,2010). Each of these methods has some advantages and disadvantages. Penalty function methods are the most popular methods used in GAs for constrained optimization problems. This method transforms the constrained optimization problem into an unconstrained one by penalizing the objective function corresponding to the infeasible solution and then explain the model by deriving the required algorithm to obtain an integer solution along with two numerical example. To convert the above problem to an unconstrained maximization problem, a large negative value (say, −M) which is blindly assigned to the objective function for the infeasible solution. In this case the reduced problem is as follows:

$$\text{Max} \quad R_s(x) = R(x) + \theta(x)$$

where

$$\theta(x) = \begin{cases} 0 & \text{if } x \in S \\ -R(x) + (-M) & \text{if } x \notin S \end{cases}$$

This is a non-linear unconstrained integer programming problem and S be the set of feasible solutions.

Solution procedure:

For the purpose of solving the nonlinear maximization problem, we have developed an advanced GA for integer variables with fitness function, tournament selection, uniform crossover, uniform mutation and elitism. The different steps of this algorithm are described as follows:

Algorithm

Step1: Initialize the parameters of Genetic Algorithm.

Step2: t = 0 (t represents the number of current generation).
Step3: Initialize P (t) (P (t) represents the population at t -th generation).
Step4: Evaluate P (t).
Step5: Find the best result from P (t).
Step6: t = t + 1.
Step7: If (t > maximum generation number) go to step-14.
Step8: Select P (t) from P (t−1) by tournament selection process
Step9: Alter P (t) by crossover, mutation and elitism process.
Step10: Evaluate P (t).
Step11: Find the best result from P (t).
Step12: Compare the best results of P (t) and P (t − 1) and accept the better one.
Step13: Go to Step-6.
Step14: Print the result.
Step15: stop.

For implementing the above GA in solving the reliability optimization problems, the following components are to be considered.

- GA Parameters
- Chromosome representation
- Initialization of population
- Evaluation of fitness function
- Selection process
- Genetic operators (crossover, mutation and elitism)

In the applications of GA, there are different types of chromosome, viz. Among these representations, real coding representation is very popular as this type of chromosome representation looks like a vector. As our proposed problem is non-linear containing n discrete variables, an integer decimal number representation is used here. An integer row matrix Vj = ($V_{j1}$ , $V_{j2}$ , . . . , $V_{jn}$ ) is used as a chromosome where the components $V_{j1}$ , $V_{j2}$ , . . . , $V_{jn}$ represent the decision variables, $x_1$ , $x_2$ , . . . , $x_n$ of the problem, respectively. In this present work, for each component of the chromosome, a random value is selected from the discrete set of values within the bounds. In order to check the quality of each potential solution from the population of potential solutions obtained by chromosome representation, the fitness value for each chromosome needs to be calculated. The first operator of GA is the selection operator. Here, we have used the well known tournament selection process of size two with replacement. In this process, it selects the better chromosome/individual from randomly selected two chromosomes/individuals based on the following assumptions:

(1) When both the chromosomes/individuals are feasible then the one with better fitness value is selected.

(2) When one chromosome/individual is feasible and another is infeasible then the feasible one is selected.

(3) When both the chromosomes/individuals are infeasible with unequal constraint violation, then the chromosome with less constraint violation is selected.

(4) When both the chromosomes/individuals are infeasible with equal constraint violation, then any one chromosome/individual is selected.

After selection of chromosomes, the crossover operation is applied. Here the crossover operation is done in the following manner:

Step-1: Find the integral value of (p_cros* p_size) and store it in N .

Step-2: Select the chromosomes Vk and Vi randomly among the population for crossover. Step-3: The components $V_{kj}$ 0 and $V_{ij}$ 0 (j = 1, 2, . . . , n) of two offspring will be created by

$V_{kj}^{\square} = V_{kj} + g$ & $V_{ij}^{\square} = V_{ij} - g$ if $V_{ij} > V_{kj}$
or $V_{kj}^{\square} = V_{kj} - g$ & $V_{ij}^{\square} = V_{ij} + g$
where g is a random integer number between 0 and |Vkj − Vij |.

Step-4: Repeat Step-2 and Step-3 for N /2 times.

After crossover of chromosomes, the uniform mutation operation is applied. If the element (gene) Vik of chromosome Vi is selected for mutation and domain of Vik is (lik , uik ), then the reduced value of Vik is given by

$V_{ik}^{\square} = V_{ik} + \Delta(u_{ik} - v_{ik})$ if a random digit is 0
$V_{ik}^{\square} = V_{ik} - \Delta(v_{ik} - l_{ik})$ if a random digit is 1
where $k \in \{1, 2, . . . , n\}$ and $\Delta$ (y) returns a value in the range (0, y).

Journal of Applied Science and Agriculture, 9(4) April 2014, Pages: 1359-1365

In order to preserve the best solution created in previous generations, a process called elitism process is considered.

The termination criterion is a rule for which the algorithm is going to stop. For this purpose any one of the following three is considered the termination criterion.

1. The best individual does not improve over the specified generations.
2. The number of generations reaches maximum number of generations.
3. The total improvement of last certain number of best solutions is less than a pre-assigned small positive number.

In this work we have used the first one as the termination criterion.

## RESULTS AND DISCUSSION

To illustrate our proposed GA based on penalty technique for solving the reliability stochastic optimization problem, we have considered two numerical examples. Each example has been formulated using Case 1. In this computation, the following values of GA parameters are used: p_size = 90, p_cross = 0.70, p_mute = 0.10, max_gen = 90.

*Example 1:*

A four-stage system with chance constraints is formulated as a pure stochastic integer programming problem using the data given in Table 1.

Max $R_s(x) = \prod_{j=1}^{n}[1-(1-r_j)^{x_j})$

s.t

$\Pr(\sum_{j=1}^{4} a_{ij}x_j \leq ci) \geq 1-\alpha i$ , $i = 1, 2$

**Table 1:** Data for Example 1.

| Stage, j | 1 | 2 | 3 | 4 | Available resource | | | |
|---|---|---|---|---|---|---|---|---|
| rj | 0.75 | 0.80 | 0.75 | 0.85 | | $l_i$ | $u_i$ | $\alpha_i$ |
| a1j | 1.5 | 3.3 | 3.2 | 4.4 | $C_1$ | 50 | 60 | 0.1 |
| a2j | 4.0 | 5.0 | 7.0 | 9.0 | $C_2$ | 110 | 140 | 0.15 |

*Example 2:*

A four stage system with chance constraints is formulated as a pure stochastic integer programming problem using the data given in Table 2.

Max $R_s(x) = \prod_{j=1}^{n}[1-(1-r_j)^{x_j})$

s.t

$\Pr(\sum_{j=1}^{4} a_{ij}x_j \leq ci) \geq 1-\alpha i$ , $i = 1, 2$

**Table 2:** Data for Example 2.

| Stage, j | 1 | 2 | 3 | 4 | Available resource | | | |
|---|---|---|---|---|---|---|---|---|
| rj | 0.76 | 0.81 | 0.78 | 0.86 | | $l_i$ | $u_i$ | $\alpha_i$ |
| a1j | 1.5 | 3.3 | 3.2 | 4.4 | $C_1$ | 50 | 60 | 0.1 |
| a2j | 4.0 | 5.0 | 7.0 | 9.0 | $C_2$ | 110 | 140 | 0.15 |

**Table 3:** Best numerical results for examples.

| Example | xj 's | Best reliability | CPU(GA) | CPU(DP) | CPU(B&B) |
|---|---|---|---|---|---|
| 1 | (5, 4, 5, 3) | 0.993088 | 0.001 | 0.047 | 0.038 |
| 2 | (5, 4, 5, 3) | 0.994650 | 0.001 | 0.060 | 0.042 |

To study the performance of our developed GA, sensitivity analyses have been done graphically for second example on the system reliability with respect to GA parameters separately keeping the other parameters at their original values. These are shown in Figs. 1-4.

Journal of Applied Science and Agriculture, 9(4) April 2014, Pages: 1359-1365

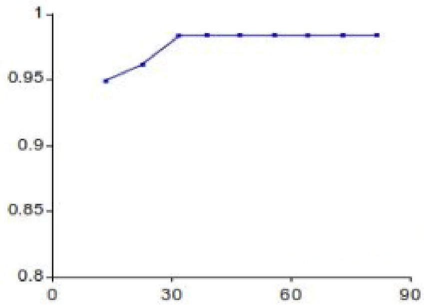


**Fig. 1:** System reliability vs. p_size..

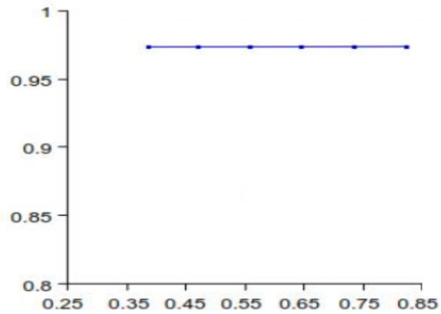


**Fig. 2:** System reliability vs. p_cross..

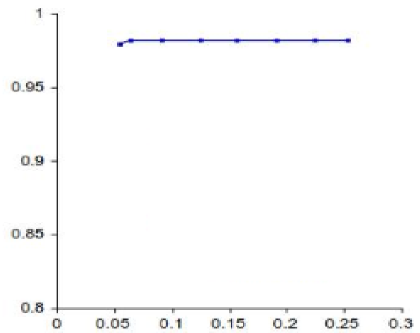


**Fig. 3:** System reliability vs. p_mute.

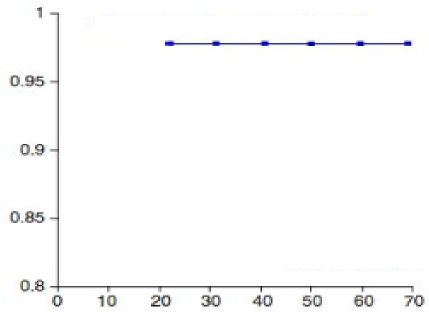


**Fig. 4:** System reliability vs. max_gen.

*Conclusion:*

The redundancy allocation problem has been solved using DP and B&B models. There are distinct differences between these various optimization approaches, and different classes of the problem are suited to one particular model. This paper explains the use of GA to analyze this problem. The GA was examined on two problems and compared with the corresponding results from DP and B&B.

GA, as powerful and broadly applicable stochastic search and optimization technique, is the most widely

**Journal of Applied Science and Agriculture, 9(4) April 2014, Pages: 1359-1365**

known type of evolutionary computational methods. GA maintains a population of chromosomes. Each chromosome represents a potential solution to the problem at hand and is evaluated to give some measure of its fitness. A new population is formed by selecting the more fitable chromosomes from the parent and offspring populations. After several generations, the algorithm converges to the best chromosome, which represents an optimal or sub-optimal solutions to the problem. One of the distinguishing features of GA is to work with a population of candidate solutions. since GA's concept and theory are simple, and GA's ability to search optimal solution is excellent; Therefore, GA approach can be applied to many engineering optimization problems as well as decision making problems in various fields.

## REFERENCES

Barlow, R.E., F. Proschan, 1975. Statistical Theory of Reliability and Life Testing: Probability Models, Rinehart and Winston, New York.

Bellman, R.E., E. Dreyfus, 1962. Applied Dynamic Programming, Princeton University Press.

Chern, M.S., 1992. On the computational complexity of reliability redundancy allocation in a series system, Operations Research Letter, (11): 309–315.

Deb, K., 2000. An efficient constraint handling method for genetic algorithms, Computer Methods in Applied Mechanics and Engineering, 186: 311-338.

Dehini, R., B. Ferdi and B. Bekkouche, 2012. Fuzzy Logic Controller Optimization Based on GA for Harmonic Mitigation. Journal of Artificial Intelligence, 5: 26-36.

Farshad Kyoomarsi, Hamid Khosravi, Esfandiar Eslami, Pooya Khosravyan Dehkordy, Asghar Tajoddin 2008/5/14. "Optimizing text summarization based on fuzzy logic". Computer and Information Science, 2008. ICIS 08. Seventh IEEE/ACIS International Conference on 347-352 IEEE

Kiyoumarsi, F., E. Hamid Khosravi, Seslami, M. Davoudi, 2010/10. "Extraction-based text sumarization using fuzzy". Iranian Journal of fuzzy systems.

Fyffe, D.E., W.W. Hines, N.K. Lee, 1968. "System reliability allocation and a computational algorithm", IEEE Transactions on Reliability, R-17: 64-69.

Goldberg, D.E., 1989. Genetic Algorithms: Search, Optimization and Machine Learning, Reading, MA, Addison Wesley.

Ha, C., W. Kuo, 2006. Reliability redundancy allocation: an improved realization for nonconvex nonlinear programming problems, European Journal of Operational Research, 171: 124-138.

Holland, J., 1975. Adaptation in Natural and Artificial Systems; University of Michigan Press.

Kuo, W., H-H. Lin, Z. Xu and W. Zhang, 1987. Reliability optimization with the Lagrange-multiplier and branch-and-bound technique. IEEE Trans. Reliability, R36(5): 624-630.

Kuo, W., V.R. Prasad, F.A. Tillman, C.L. Hwang, 2001.Optimal Reliability Design: Fundamentals and Applications, Cambridge University Press, ambridge.

Nakagawa, Y., S. Miyazaki, 1981 "Surrogate constraints algorithm for reliability optimization problems with two constraints", IEEE Transactions on Reliability, R-30: 175-180.

Sakawa, M., 2002. Genetic Algorithms and Fuzzy Multiobjective Optimization, Kluwer Academic Publishers.

Sun, X., D. Li Duan, 2002. Optimal condition and branch and bound algorithm for constrained redundancy optimization in series system, Optimization and Engineering, 3: 53-65.

Tillman, F.A., C.L. Hwang and W. Kuo, 1977. Optimization techniques for system reliability with redundancy – A review. IEEE Trans. Reliability, R26: 148-155.

Zhao, R. and B. Liu, 2003. Stochastic programming models for general redundancy – optimization problems. IEEE Trans. Reliability, R52(2): 181-191.