Int. J. Nonlinear Anal. Appl. 13 (2022) 2, 3139–3144
ISSN: 2008-6822 (electronic)
http://dx.doi.org/10.22075/ijnaa.2022.27935.3760

Using genetic algorithm to optimize a system with repairable components and multi-vacations for repairmen

Mohammadreza Shahriari

Faculty of Industrial Management, South Tehran Branch, Islamic Azad University, Tehran, Iran

(Communicated by Madjid Eshaghi Gordji)

Abstract

In this paper, we present a redundancy allocation problem (RAP) with series-parallel sub-systems and repairable components. The repairmen will go on multiple vacations. In repairable systems, a fundamental aspect to be considered is to predict the reliability of the systems under study. Set a reliability model for repairable systems, however, is still a challenging problem when considering the dependency This paper aims to evaluate the number of components and repairmen in each sub-system. Because this RAP belongs to Np. Hard problems, also, a Genetic algorithm to solve the presented model.

Keywords: Redundancy allocation problem, Multiple vacation repairmen, Reparable components, Genetic algorithm
2020 MSC: 68W50

1 Introduction

In this paper we work on a RAP with series sub-system and repairable components. Also, the repairmen will go to multiple vacations. This model aims to maximize system reliability by determining the number and components in addition to determine the number of the repairmen in each sub-system.

Fyffe [4] in 1968 presented RAP for the first time and Chern [1] proved that RAP belongs to Np. Hard problem, so we used Genetic algorithm for solving the presented model. Table 1 contains some research conducted on RAP.

Table 1: Classification of the Studies on RAP

Author(s)	Decision/Subject	Study Objectives		Year	Ref.
		Single	multiple		
Fyffe et al.	RAP and a Computational Algorithm (Dynamic Programming)	•		1968	[4]
Nakagawa and Miyazaki	Surrogate Constraints Algorithm for Reliability Optimization Problems with Two Constraints	•		1981	[9]

This paper has four parts. The first part is introduction. In the second part, we present the mathematical model. Third part deals with the solving method and the fourth part presents a numerical example. The fifth part is conclusion and further studies.

Email address: shahriari.mr@gmail.com (Mohammadreza Shahriari)

Received: January 2022 *Accepted:* June 2022

Coit and Smith	Optimization approaches to the redundancy allocation to the redundancy allocation problem for series – parallel systems	•		1995	[3]
Liang and Smith	An Ant Colony Optimization Algorithm for RAP	•		2004	[6]
Yun and Kim	Multi-Level Redundancy Optimization in Series Systems		•	2004	[15]
Liang and Wu	A Variable Neighborhood Descent Algorithm (VNA) for RAP	•		2005	[7]
Coit and Konak	Multiple Weighted Objectives Heuristic for the RAP		•	2006	[2]
Tavakkoli-Moghaddam, Safari and Sassani	Reliability Optimization of Series-Parallel Systems with a Choice of Redundancy Strategies Using a Genetic Algorithm	•		2008	[13]
Zaretalab et al.	Redundancy allocation problem with multi-state component systems and reliable supplier selection	•		2020	[16]
Sun et al.	On the optimal redundancy allocation for multi-state series–parallel systems under epistemic uncertainty	•		2019	[12]
Salmasnia et al.	A redundancy allocation problem by using utility function method and ant colony optimization: trade off between availability and total cost.		•	2019	[11]
Ouyang et al.	An improved particle swarm optimization algorithm for reliability-redundancy allocation problem with mixed redundancy strategy and heterogeneous components	•		2019	[10]
Mahdavi-Nasab et al.	Water cycle algorithm for solving the reliability-redundancy allocation problem with a choice of redundancy strategies	•		2020	[8]

2 Mathematical model

2.1 Nomenclatures

λ : Failure rate of components,

μ : Repair time of components,

ν : Idle time rate of repairmen,

R_i : Number of repairmen in sub-system i,

s : Number of sub-systems,

c_{i1} : Price of the component in sub-system i,

c_{i2} : Price of hiring repairmen in sub-system i,

C : Maximum system acceptable cost,

w_i : Weight of the component in sub-system i,

W : Maximum system acceptable weight,

K_i : Minimum components in sub-system i,

$n_{i,\max}$: Maximum components in sub-system i

2.2 System with R repairmen with multiple vacations

Consider a system with n identical repairable components and R repairmen. The repairmen have multiple vacations. The state-space diagram of this model is presented in Figure 1.

The system reliability of this model has been driven by Yuan and Zhen-Dong [14] as $A(t, n, R)$.

2.3 Redundancy allocation problem

Consider a system with s sub-system that is connected serially together. RAP aims to determine optimal number of components in each sub-system to maximize system reliability. Because the components are repairable, we want to calculate the optimal number of repairmen in each sub-system. The RAP mathematical mode is as follows:

$$\max A(t) = \prod_{i=1}^{s} A(t, n_i, R_i) \tag{2.1}$$

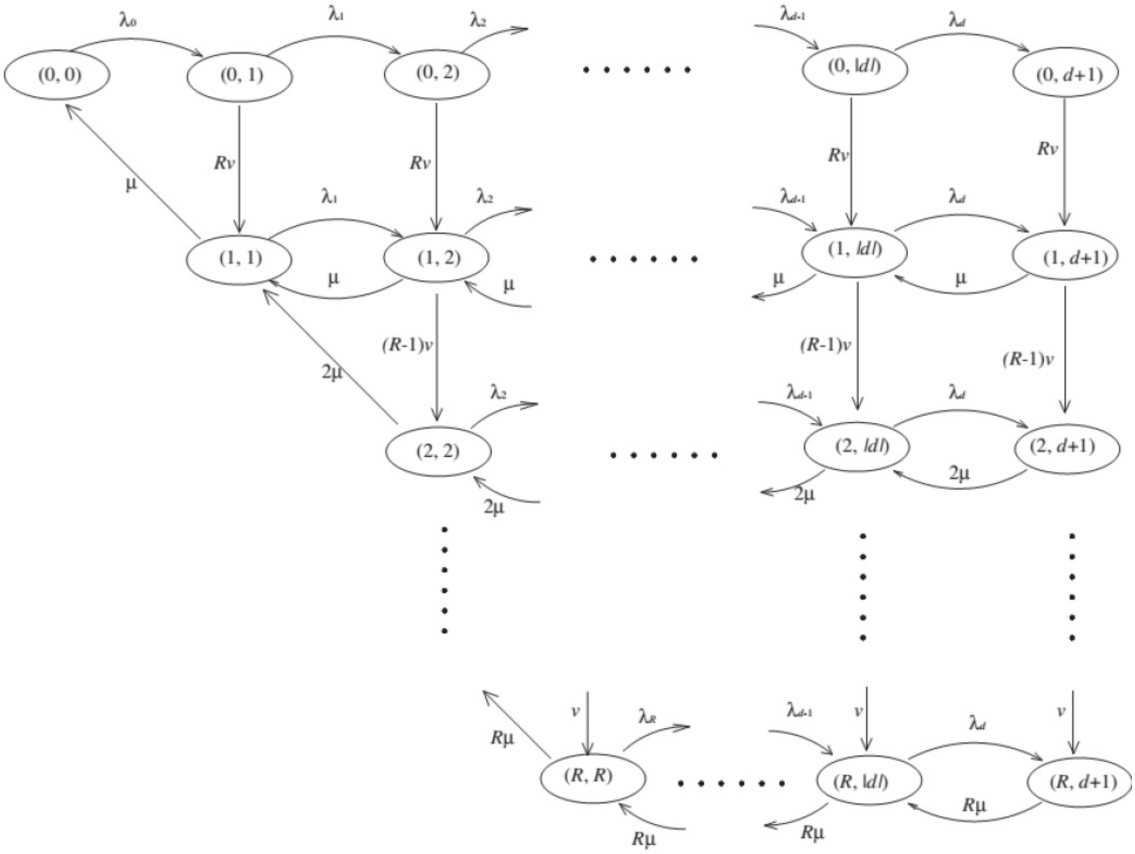

Figure 1: State-space diagram of the system

such that

$$\sum_{i=1}^{S}\{c_{i1}n_i + c_{i2}R_i\} \leq C \tag{2.2}$$

$$\sum_{i=1}^{S} w_i n_i \leq W \tag{2.3}$$

$$k_i \leq n_i \leq n_{i\,\max} \tag{2.4}$$

$$0 \leq R_i \leq n_i \tag{2.5}$$

$$n_i, R_i \in W. \tag{2.6}$$

Equation (2.1) is the objective function that maximizes system reliability. Equation (2.2) is the cost constraint that contains the cost of purchasing components and hiring repairmen. Equation (2.3) is the system weight constraint and Equation (2.4) is the lower and upper values of the components in each sub-system. Equation (2.5) is the minimum and maximum repairmen in each sub-system and Equation (2.6) denotes that the objective functions are integers.

3 Solving method

3.1 Genetic algorithm

Genetic algorithm (GA) was presented by Holland [5] in 1975 for the first time. An overview of this algorithm is as follows:

- Producing the initial population,
- Calculation of the chromosome's fitness function,

- Producing the new population by three operators:
  - Crossover,
  - Mutation,
  - Elitism,
- Substituting the new offspring in new population,
- Using new population for the next algorithm iterations,

Chromosome structure: The chromosome of presented model is a $2 \times s$ dimensional matrix that s is the number of sub-systems. The number of the components in each sub-system appears in the first row and the number of repairmen in each sub-system appears in the second row. Figure 2 is the problem chromosome structure.

n_1	n_2	...	n_s
R_1	R_2	...	R_s

Figure 2: Problem chromosome structure

Initial population: The initial population creates randomly,

Fitness function: The fitness function is calculated as follows:

$$\text{Fitness Function} = \frac{\prod_{i=1}^{s} A(t, n_i, R_i)}{1 + \max\{0, \sum_{i=1}^{s}\{c_{i1}n_i + c_{i2}R_i\} - C\} + \max\{0, \sum_{i=1}^{s}(w_i n_i) - W\}} \tag{3.1}$$

Parent selection: We used roulette wheel for selecting parents for crossover and mutation operators.

Crossover operator: For crossover, we selected two parents and changed the seven genomes of the chromosomes with each other randomly.

Mutation operator: For mutation operator, we selected a parent and changed two genomes of parent with each other's.

3.2 Parameter tuning

For algorithm parameter tuning, we used Response Surface Methodology (RSM). The upper bound and lower bound of the parameters and the optimal values of these parameters are presented in Table 2.

Table 2: Lower bound, upper bound and optimal values of algorithm parameters.

	L	**U**	**Optimal**
$nPop$	100	300	300
P_c	0.70	0.90	0.83
P_m	0.10	0.30	0.10

4 Numerical example

We used the numerical example that was presented by Fyffe [4]. The values of the parameters are presented in Table 3.

The other parameters values are $\mu_i = 0.015,\ v_i = 1,\ W = 250,\ c_{i2} = 2 \times c_{i1}$. The optimal chromosome of model is presented in Figure 3.

Table 3: Component data for example

Sub-system	λ_i	k_i	c_{i1}	w_i	**Sub-system**	λ_i	k_i	c_{i1}	w_i
1	0.00532	2	1	3	8	0.0105	3	3	4
2	0.00818	3	2	8	9	0.00268	2	2	8
3	0.01330	3	2	7	10	0.01410	3	4	6
4	0.00741	2	3	5	11	0.00394	2	3	5
5	0.00619	1	2	4	12	0.00236	1	2	4
6	0.00436	3	3	5	13	0.00215	2	2	5
7	0.01050	3	4	7	14	0.01100	3	4	6

Sub-System														
	1	2	3	4	5	6	7	8	9	10	11	12	13	14
k	2	3	3	2	1	3	3	3	2	3	2	1	2	3
c1	1	2	2	3	2	3	4	3	2	4	3	2	2	4
w	3	8	7	5	4	5	7	4	8	6	5	4	5	6
c2	2	4	4	6	4	6	8	6	4	8	6	4	4	8
n	3	4	3	3	2	4	3	3	4	3	3	2	3	3
R	2	3	2	2	1	3	2	2	2	1	1	1	2	2
Cost=	7	20	14	21	8	30	28	21	16	20	15	8	14	28
Weight=	9	32	21	15	8	20	21	12	32	18	15	8	15	18
Total Cost=250 , Total Weight=244														

Figure 3: optimal chromosome.

5 Conclusion and Further Studies

In this paper, we worked on an RAP with series sub-system and repairable component. Also we considered that the repairmen would go to multiple vacations. This model was presented by Yuan and Zhen-Dong [14]. We maximized system reliability and calculated the optimal values of components besides calculating the optimal number of the repairmen in each sub-system.

For further studies, we propose to consider that each repairman can work on the failed components of all sub-systems. Also using different available components type for each sub-system may more draw the problem to real world conditions.

References

[1] M.S. Chern, *On the computational complexity of reliability redundancy allocation in a series system*, Operat. Res. Lett. **11** (1992), 309–315.

[2] D.W. Coit and A. Konak, *Multiple weighted objectives heuristic for the redundancy allocation problem*, IEEE Trans. Reliab. **55** (2006), no. 3, 551–558.

[3] D.W. Coit and A. Smith, *Optimization approaches to the redundancy allocation to the redundancy allocation problem for series – parallel systems*, Proc. Fourth Ind. Engin. Res. Conf., 1995, pp. 342–349.

[4] D.E. Fyffe, W.W. Hines and N.K. Lee, *System reliability allocation and a compactional algorithm*, IEEE Trans. Reliab. **17** (1968), 64–69.

[5] J. Holland, *Adaptation in natural and artificial systems*, University of Michigan Press, 1975.

[6] Y.C. Liang and A. Smith, *An ant colony optimization algorithm for the redundancy allocation problem (RAP)*, IEEE Trans. Reliab. **53** (2004), no. 3, 417–423.

[7] Y.C. Liang and C.C. Wu, *A variable neighborhood descent algorithm for the redundancy allocation problem*, Ind. Eng. Manag. Syst. **4** (2005), no. 1, 109–116.

[8] N. Mahdavi-Nasab, M. Abouei Ardakan and M. Mohammadi, *Water cycle algorithm for solving the reliability-redundancy allocation problem with a choice of redundancy strategies*, Commun. Statist. Theory Meth. **49** (2020), no. 11, 2728–2748.

[9] Y. Nakagawa and S. Miyazaki, *Surrogate constraints algorithm for reliability optimization problems with two constraints*, IEEE Trans. Reliab. **30** (1981), 175–180.

[10] Z. Ouyang, Y. Liu, S.J. Ruan and T. Jiang, *An improved particle swarm optimization algorithm for reliability-redundancy allocation problem with mixed redundancy strategy and heterogeneous components*, Reliab. Engin. Syst. Safety **181** (2019), 62–74.

[11] A. Salmasnia, S. Noori and H. Mokhtari, *A redundancy allocation problem by using utility function method and ant colony optimization: tradeoff between availability and total cost*, Int. J. Syst. Assurance Eng. Manag. **10** (2019), no. 3, 416–428.

[12] M.X. Sun, Y.F. Li and E. Zio, *On the optimal redundancy allocation for multi-state series–parallel systems under epistemic uncertainty*, Reliab. Engin. Syst. Safety **192** (2019), 106019.

[13] R. Tavakkoli-Moghaddam, J. Safari and F. Sassani, *Reliability optimization of series-parallel systems with a choice of redundancy strategies using a genetic algorithm*, Reliab. Engin.Syst. Safety **93** (2008), 550–556.

[14] L. Yuan and Z.D. Cui, *Reliability analysis for the consecutive-k-out-of-n: F system with repairmen taking multiple vacations*, Appl. Math. Model. **37** (2013), no. 7, 4685–4697.

[15] W.Y. Yun and J.W. Kim, *Multi-level redundancy optimization in series systems*, Comput. Indust. Eng. **46** (2004), 337–346.

[16] A. Zaretalab, V. Hajipour and M. Tavana, *Redundancy allocation problem with multi-state component systems and reliable supplier selection*, Reliab. Engin. Syst. Safety **193** (2020), 106629.