Applied and Computational Mathematics
2015; 4(2-1): 1-6
Published online February 27, 2015 (http://www.sciencepublishinggroup.com/j/acm)
doi: 10.11648/j.acm.s.2015040201.11
ISSN: 2328-5605 (Print); ISSN: 2328-5613 (Online)

Comparing Two Meta-Heuristic Approaches for Solving Complex System Reliability Optimization

Hadi Akbarzade Khorshidi[1, *], Sanaz Nikfalazar[2]

[1]School of Applied Science and Engineering, Faculty of Science, Monash University, Melbourne, Australia
[2]Faculty of Management, Department of Public Administration, University of Tehran, Tehran, Iran

Email address:
Hadi.khorshidi@monash.edu (H. A. Khorshidi), sanaz_nikfal@alumni.ut.ac.ir (S. Nikfalazar)

To cite this article:
Hadi Akbarzade Khorshidi, Sanaz Nikfalazar. Comparing Two Meta-Heuristic Approaches for Solving Complex System Reliability Optimization. *Applied and Computational Mathematics*. Special Issue: Quality, Reliability, Safety, and Risk Modeling and Optimization. Vol. 4, No. 2-1, 2015, pp. 1-6. doi: 10.11648/j.acm.s.2015040201.11

Abstract: Using meta-heuristic approaches to solve reliability and redundancy allocation problems (RRAP) has become attractive for researchers in recent years. In this paper, an optimization model is presented to maximize system reliability and minimize system cost simultaneously for multi-state weighted *k*-out-of-*n* systems. The model tends to optimize system design and maintenance activities over functioning periods that provides a dynamic modeling. A recently developed meta-heuristic approach imperialist competitive algorithm (ICA) and genetic algorithm (GA) are used to solve the model. The computational results have been compared to find out which approach is more appropriate for solving complex system reliability optimization models. It is shown that GA can find the better solution while ICA is a faster approach. In addition, an investigation is done on different parameters of the ICA.

Keywords: Reliability-Redundancy Allocation Problem (RRAP), Imperialist Competitive Algorithm (ICA), Genetic Algorithm (GA), System Reliability Optimization (SRO), Multi-State Weighted *k*-out-of-*n* Systems

1. Introduction

Reliability optimization models have been developed to find optimal decisions for engineering systems. A reliability-redundancy allocation problem (RRAP) is formulated to determine the best selection for components in system design and improvement actions simultaneously. Different approaches such as exact, heuristic and meta-heuristic have been constructed to solve system reliability optimization (SRO) models [1]. Chern [2] showed that a redundancy allocation problem (RAP) with simple linear constraints for a series system is an NP-hard problem. Therefore, meta-heuristic approaches have been used widely in SRO area. Genetic algorithm (GA) is a meta-heuristic approach that has been mostly applied to solve reliability optimization models [3-5]. Coelho [6] used particle swarm optimization (PSO) to solve a mixed-integer model for RRAP. An ant colony algorithm is employed to find the optimal design of multi-state series-parallel systems which is modelled by universal generating function (UGF) in [7]. A modified imperialist competitive algorithm (ICA) is developed to discover more optimum solutions for series, series-parallel, bridge, and over-speed protection systems which are formulated as RRAP [8]. Sakalli [9] solved a reliability-based nonlinear multi-objective model by simulated annealing (SA).

In [10], the capability of GA and SA is compared to solve a mean time to failure (MTTF) maximization model of series-parallel systems. The model is to find the optimal decisions for a RRAP. The comparison has been done over finding best solutions and computational time. It is found that GA reaches to better solution, however SA is faster. Also, SA-parallel, parallel vibrating damp optimization (VDO), and GA are compared for solving series-parallel RAP model [11]. In this study, two meta-heuristic approaches are used to solve a complex RRAP model for multi-state k-out-of-n systems. This model not only tries to find the optimal number and mixture of components in the system, but also is to apply economic maintenance actions during functioning periods. The objective function is maximizing the difference between incomes generated from a reliable system and system cost. Therefore, the model can maximize system reliability and minimize system cost simultaneously.

The remainder of the paper is organized as follows. Section 2 describes the problem and model formulations. Section 3 introduces the two meta-heuristic approaches briefly. Section 4 shows the solving results. Finally, concluding remarks are brought in section 5.

2. Model Description

The problem that is formulated to solve is for k-out-of-n systems. In binary view, a k-out-of-n:G system works if and only if at least k components work. However, a multi-state weighted k-out-of-n system is in state j or above if the total weight of all components is equal to or greater than a threshold value k_j [12]. In this study, income generated during functioning periods is used as the weight of each component in each reliability state. The UGF approach is used to compute the system reliability which is equivalent with system income. Therefore, a maximization objective function can work as a bi-criteria optimization model as presented in [13]. It is an optimization model which is constructed based on an evaluation approach proposed in [14].

2.1. Notations

X: Variable vector
$NPV(X)$: Net present value of system;
$E(X)$: System income;
$C(X)$: System cost;
x_i: Integer variable for the number of component i;
y_i: Boolean variable to show the selection of the component i;
$ms_k(t)$: Boolean variable for applying system-level maintenance plans k;
$mc_{ik}(t)$: Boolean variable for applying component-level maintenance plans k;
H: The number of component types in the system;
Ks: The number of system-level maintenance plans;
Kc: The number of component-level maintenance plans;
T: The total number of working periods;
M: The number of reliability levels;
$R_t^s(k_j, n)$: The availability of the system with n components in state j;
k_j: System value in state j;
r: Interest rate;
g_t: Income generated at time t;
$u_t^i(z)$: The u-function of component i at time t;
$U_t(z)$: The u-function of the system at time t;
$P_j^i(t)$: The probability that component i is in state j at time t;
$p_{mj}^i(t)$: The probability that component i transits from state m to state j at time t;
as_{mjk}^i: The system-level effect of plan k on the transition probability of the component i from state m to state j
ac_{mjk}^i : The component-level effect of plan k on the transition probability of the component i from state m to state j
C_{com}: Components' cost;
$nmax_i$: The maximum available number of component i;
C_{mtc}: The cost of maintenance activities;
C_{mp}: Maintenance plan cost;
C_{rm}: Routine maintenance cost;
Crm_i: Routine maintenance cost for component i;
$Cmcf_{ik}$: Fixed cost for maintenance plan k on component i;
$Cmcv_{ik}$: Variable cost for maintenance plan k on component i;
Cms_k: The cost of system-level plan k;

2.2. Mathematical Model

$$\text{Maximize } NPV(X) = E(X) - C(X) \quad (1)$$

$x_i \geq 0, integer \ \ (i = 1, 2, \dots, H)$
$y_i = 0 \ or \ 1, \quad (i = 1, 2, \dots, H)$
$ms_k(t) = 0 \ or \ 1, (k = 1,2, \dots, Ks; \ t = 0,1, 2, \dots, T-1)$
$mc_{ik}(t) = 0 \ or \ 1, \ \ (i = 1,2, \dots, H; \ k = 1,2, \dots, Kc; \ t = 0,1, 2, \dots, T-1)$

System income:

$$E(X) = \frac{\sum_{j=0}^{M}\sum_{t=0}^{T-1} R_t^s(k_j, n)\cdot k_j}{(1+r)^t} \quad (2)$$

$$R_t^s(k_j, n) = \sum P_t\,, \quad \text{for} \quad k_j \leq g_t < k_{j+1} \quad (3)$$

$$U_t(z) = \otimes_+ \left(u_t^1(z)^{x_1}, u_t^2(z)^{x_2}, \dots, u_t^H(z)^{x_H}\right) \quad (4)$$

$$P_j^i(t) = \sum_{m=0}^{M} P_m^i(t-1). p_{mj}^i(t-1)\,, \forall i; t = 1, \dots, T-1 \quad (5)$$

$$p_{mj}^i(t) = p_{mj}^i(t-1) + \sum_{k=1}^{Ks}\left(\left(p_{mj}^i(t-1). as_{mjk}^i - p_{mj}^i(t-1)\right)\cdot ms_k(t)\right) + \sum_{k=1}^{Kc}\left(\left(p_{mj}^i(t-1). ac_{mjk}^i - p_{mj}^i(t-1)\right)\cdot mc_{ik}(t)\right), \quad \forall i, t = 1, \dots, T-1 \quad (6)$$

$$p_{mj}^i(t) = \frac{p_{mj}^i(t)}{\sum_{j=0}^{M} p_{mj}^i(t)}, \ \forall i, t = 0, \dots, T-1 \quad (7)$$

System cost:

$$C(X) = C_{com} + C_{mtc} \quad (8)$$

$$C_{com} = \sum_{i=1}^{H}(Cf_i \cdot y_i + Cv_i \cdot x_i) \quad (9)$$

$$x_i \leq nmax_i \cdot y_i \quad \text{for} \quad \forall i \in \{1, \dots, H\} \quad (10)$$

$$C_{mtc} = C_{rm} + C_{mp} \quad (11)$$

$$C_{rm} = \sum_{i=1}^{H} Crm_i . x_i \quad (12)$$

$$C_{mp} = \sum_{t=0}^{T-1}\sum_{k=1}^{Kc}\sum_{i=1}^{H} mc_{ik}(t)\cdot \frac{Cmcf_{ik} + Cmcv_{ik}\cdot x_i}{(1+r)^t} + \sum_{t=0}^{T-1}\sum_{k=1}^{Ks} ms_k(t)\cdot \frac{Cms_k}{(1+r)^t} \quad (13)$$

3. Meta-Heuristic Approaches

In this section, a brief introduction is brought for GA and ICA that have been applied on the complex reliability optimization model.

3.1. Genetic Algorithm

The GA is an effective and commonly used meta-heuristic approach to solve optimization problems. It also used frequently in SRO area to find the optimal solution in system design and maintenance planning. GA is an evolutionary based method which starts with a population of feasible solutions. Each solution is known as a chromosome which consists of genes as variables. The chromosomes in GA are evaluated by a fitness function which is the objective function in this study. Initial population will be converted to new population in the next generation via using crossover and mutation operators. These operators help to generate new solutions that could be evaluated by the fitness function in order to find the better feasible solutions. In crossover operator, two or more chromosomes as parents create new chromosomes as children for the new population. However, mutation operator changes a single chromosome to make a new chromosome. Therefore, new chromosomes are ranked and selected for the new population. The process of producing new generations continues until reaching the stopping criteria. Finally, the chromosome with the best fitness value in the last generation is considered as the optimum solution of the model.

3.2. Imperialist Competitive Algorithm

This algorithm was firstly introduced by Atashpaz-Gargari in 2007 [15]. It is an evolutionary based algorithm like GA and SA. However, they are based on natural phenomena, while ICA is inspired by socio-political evolution of humans.

In this approach, like all evolutionary algorithms, an initial population is formed so that each solution is a country. These countries are divided into two groups of colonies and imperialists. Imperialist countries try to dominate more countries and assimilate the colonies to their selves. This struggling process continues until just one imperialist country exists. Therefore, the last imperialist is considered as the optimal solution.

In ICA, first of all, an initial population including countries is constructed. Some characteristics which are important to measure the power of each country and prioritization are allocated to each country as Eq. 14.

$$Country = \{P_1, P_2, \dots, P_{N_{var}}\} \tag{14}$$

where P_i is the *i*th characteristic for each country that works as a variable, and N_{var} is the total number of variables. The cost function (power) of each country can be defined based on these variables as Eq. 15.

$$cost = f(country) = f(p_1, p_2, \dots, p_{N_{var}}) \tag{15}$$

The higher the cost function, the more the country's power. All countries are ranked by the value of the cost function, and the imperialists are selected. The normalized cost function is calculated based on the countries' cost functions as Eq. 16.

$$C_n = c_n - max_i\, c_i \tag{16}$$

where c_n and C_n are the cost function and normalized cost function of *n*th empire respectively. Number of allocated colonies to each imperialist is obtained by a probability value which is computed as Eq. 17. Therefore, colonies are allocated to each imperialist randomly.

$$P_n = \left| \frac{C_n}{\sum_{i=1}^{N_{imp}} C_i} \right| \tag{17}$$

where N_{imp} is the number of imperialists, and P_n is the allocation probability of *n*th empire.

Each imperialist tends to make its colonies similar to itself. Hence, they are moved toward the imperialist by a vector with size of x' and angle of θ. These two parameters follow the uniform distribution, and are called assimilation coefficient (AC) and assimilation angle coefficient (AAC) respectively. This assimilation process provides an opportunity to generate new solutions for the problem.

According to the movements of the colonies, it is possible that a colony becomes more powerful than the imperialist. In this situation, the colony and the imperialist swap their positions. This process can be called as a revolution.

After that, the imperialist competition should be formulized. To reach this goal, a total cost function of each empire is developed as Eq. 18. The cost of each colony is considered in total cost of the empire.

$$TC_n = c_n + \varepsilon . Mean\{c_{n_i}\} \tag{18}$$

where $\mathrm{TC_n}$ is the total cost of the empire, c_{n_i} is the cost of *i*th colony of the *n*th empire, and ε is a positive small constant.

The normalized total cost of each imperialist (NTC_n) is calculated by Eq. 19. According to this value, the weakest empire is identified. Then, the weakest colony of the empire is separated, and joint to another empire.

$$NTC_n = TC_n - max_i\, TC_i \tag{19}$$

The destination empire is selected randomly based on the equations 20-23.

$$P_{P_n} = \left| \frac{NTC_n}{\sum_{i=1}^{N_{imp}} NTC_i} \right| \tag{20}$$

$$P = \left[P_{P_1}, P_{P_2}, \dots, P_{P_{N_{imp}}}\right] \tag{21}$$

$$R = \left[r_1, r_2, \dots, r_{N_{imp}}\right] \tag{22}$$

$$D = P - R = \left[D_1, D_2, \dots, D_{N_{imp}}\right] \tag{23}$$

where P_{P_n} is the probability of selecting *n*th empire as the destination, P is a vector which consists of the probabilities, R is the same size vector with random numbers between 0 and 1 which ensures that the selection is not based on the power of the empires solely. The vector D is the difference between these two vectors. The highest value of D_i determine the host empire of the separated colony. When an empire has no colony, it will be eliminated. In ICA, revolution and imperialist competition processes converge the algorithm [8, 15, 16].

4. Comparison between GA and ICA

To compare the capability of the meta-heuristic approaches for solving the proposed complex system reliability model, an example is developed. In the example, three component types work together, and there are three reliability levels as failure, deteriorated, and perfect working. The components have different income generating rates in different states. The number of the periods in which system works is four. Also, Table **1**-10. There are three states for the whole system as well. The equivalent generated income for failure, deteriorated, and perfect states are 0, 4, and 10 respectively. three system-level and three component-level maintenance plans can be applied on the system. Therefore, the model has to decide on the number of each component type, and implementing of maintenance plans. Hence, the number of variables in this example is 51. As can be seen, it is a complex problem to solve.

The information about components, costs, income rates, and maintenance plans are presented in

Also, cost of system-level maintenance plans (Cms_k) are 2.5, 3, and 3.5.

Table 1. *Components data*

Comp	Cf_i	Cv_i	Crm_i	$Cmcf_{i1}$	$Cmcf_{i2}$	$Cmcf_{i3}$	$Cmcv_{i1}$	$Cmcv_{i2}$	$Cmcv_{i3}$	g_0^i	g_1^i	g_2^i
1	1.5	3	1.5	1	0.9	1	0.5	0.45	0.5	0	2	3
2	2.5	5.6	2	1.2	0.7	1.1	0.6	0.35	0.55	0	3	4
3	2.5	4.5	2.5	1.3	0.8	0.8	0.65	0.4	0.4	0	3	5

Table 2. *Transition probabilities for component 1*

State	State	0	1	2
0		0.5	0.35	0.15
1		0.2	0.7	0.1
2		0.05	0.15	0.8

Table 3. *Transition probabilities for component 2*

State	State	0	1	2
0		0.45	0.3	0.25
1		0.15	0.65	0.2
2		0.03	0.07	0.9

Table 4. *Transition probabilities for component 3*

State	State	0	1	2
0		0.5	0.4	0.1
1		0.25	0.7	0.05
2		0.05	0.25	0.7

Table 5. *Effect factor of system-level plans for component 1*

State	State	0			1			2		
	Plan	1	2	3	1	2	3	1	2	3
0		1	1	1	1.5	2	1	2.7	1	1
1		0.4	0.5	1	1	1	1	2.5	2	1
2		0.5	1	1	0.7	0.5	1	1	1	1

Table 6. *Effect factor of system-level plans for component 2*

State	State	0			1			2		
	Plan	1	2	3	1	2	3	1	2	3
0		1	1	1	1.3	2	1	1.25	1	1
1		0.85	0.5	1	1	1	1	1.2	2	1
2		0.93	1	1	0.97	0.5	1	1	1	1

Table 7. *Effect factor of system-level plans for component 3*

State	State	0			1			2		
	Plan	1	2	3	1	2	3	1	2	3
0		1	1	1	1.7	2	2	2.8	2	2
1		0.5	0.5	0.5	1	1	1	3.7	2	1
2		0.3	1	0.5	0.5	0.5	0.5	1	1	1

Table 8. *Effect factor of component-level plans for component 1*

State	State	0			1			2		
	Plan	1	2	3	1	2	3	1	2	3
0		1	1	1	1.1	1.5	1	1.2	1.5	1
1		0.9	0.5	1	1	1	1	1.25	1.5	1
2		0.8	0.5	1	0.95	0.5	1	1	1	1

Table 9. *Effect factor of component-level plans for component 2*

State	State	0			1			2		
	Plan	1	2	3	1	2	3	1	2	3
0		1	1	1	1.05	1	1	1.1	1	1
1		0.95	1	1	1	1	1	1.07	1	1
2		0.95	1	1	0.97	1	1	1	1	1

Table 10. *Effect factor of component-level plans for component 3*

State	State	0			1			2		
	Plan	1	2	3	1	2	3	1	2	3
0		1	1	1	1.2	1	1.4	1.22	1	1.4
1		0.9	1	0.7	1	1	1	1.4	1	1.4
2		0.8	1	0.7	0.85	1	0.7	1	1	1

The example is modelled by Matlab, and is solved by GA and ICA. For GA, population size is set as 100, and the process of producing new generations will stop after 200 generations. In order to have a fair comparison, the ICA solving process will stop on the 200th generation, and the population consists of 100 countries. Also, seven imperialists are adjusted in the initial population. To have a statistical view, the example has been solved 30 times. The results of the investigation are shown in Table 11.

Table 11. *Comparison results*

Approach	Mean-NPV	Max-NPV	Min-NPV	STD-NPV	Mean-Time
GA	7.0387	7.7216	5.2703	0.6491	3111.6076
ICA	6.1108	7.5063	4.6359	0.5709	1611.8238

As can be seen, GA reaches to better solutions than ICA over 200 generations. However, the standard deviation (STD) of ICA's solutions is less than GA. It shows that ICA process to find optimal solution has less diversity. Furthermore, the speed of ICA is more than GA.

In addition to the above comparison, the model has been solved by ICA with different parameters. It provides an opportunity to make a comparison when ICA's parameters are changed. Firstly, the stopping criterion is increased from 200 to 500. In the second case, the number of countries in each population is changed to 200, and the initial number of emperors becomes 8 over 200 generations. In all previous cases, AC is adjusted by 2. In the third case, all parameters are like case 2 but AC is varied to 3. The comparison results for ICA are presented in Table 12.

Table 12. *Comparison results for ICA*

ICA	Mean-NPV	Max-NPV	Min-NPV	STD-NPV	Mean-Time
100,7,200, AC=2	6.1108	7.5063	4.6359	0.5709	1611.8238
100,7,500, AC=2	6.5382	7.5353	5.0033	0.5818	5551.4707
200,8,200, AC=2	7.0580	7.5063	6.0938	0.4617	3154.0953
200,8,200, AC=3	7.2132	7.7216	6.5367	0.3578	3375.5644

The quality of the solutions has been improved by increasing the number of the generations, while the spent time is increased dramatically. Also, increasing the number of countries and imperialists makes an improvement in solutions as well. The best results are for the last case so that the highest value for mean, maximum and minimum and the lowest value for standard deviation are obtained. Although the speed of the process is reduced, it is not too much.

5. Conclusion

In this paper, an optimization model for multi-state weighted k-out-of-n system reliability is solved by two meta-heuristic approaches. The model is a RRAP that can find optimal decisions for system design and maintenance activities. Different scenarios are examined in terms of reliability improvement and cost reduction as a bi-criteria optimization model. Two meta-heuristic approaches that are used to solve the model are GA and ICA. A brief introduction is provided for both approaches in the context.

An example is programming by Matlab software. After that, the example is solved by both approaches. The results show that GA finds the better solution for the complex system reliability model. However, ICA is faster than GA. In addition, an investigation is done on different parameters of the ICA. The investigation confirms that ICA can become a more efficient approach by changing its parameters. An increase in number of countries, emperors, and generations can lead to better optimal solutions.

For further research, the model can be solved by other meta-heuristic approaches like SA, PSO, cuckoo search, ant colony, etc. Also, other parameters of ICA would be examined to find the best combination in solving complex optimization models. In addition, the optimization model can be applied on a practical case.

References

[1] Khorshidi, H. A., Gunawan, I. & Ibrahim, M. Y., Investigation on system reliability optimization based on classification of criteria.in *Proc. IEEE International Conference on Industrial Technology (ICIT)*, 2013.

[2] Chern, M. S., On the computational complexity of reliability redundancy allocation in a series system, *Operations Research Letters* 1992; 11 (5), 309-315.

[3] Ebrahimipur, V., Qurayshi, S. F., Shabani, A. & Maleki-Shoja, B., Reliability optimization of multi-state weighted k-out-of-n systems by fuzzy mathematical programming and genetic algorithm, *International Journal of System Assurance Engineering and Management* 2011; 2 (4), 312-318.

[4] Hamadani, A. Z. & Khorshidi, H. A., System reliability optimization using time value of money, *International Journal of Advanced Manufacturing Technology* 2013; 66 (1-4), 97-106.

[5] Konak, A., Coit, D. W. & Smith, A. E., Multi-objective optimization using genetic algorithms: A tutorial, *Reliability Engineering and System Safety* 2006; 91 (9), 992-1007.

[6] Coelho, L. D. S., An efficient particle swarm approach for mixed-integer programming in reliability-redundancy optimization applications, *Reliability Engineering and System Safety* 2009; 94 (4), 830-837.

[7] Di, P., Xu, Y. F., Li, F. & Chen, T. 2014. Reliability optimization for multi-state series-parallel system design using ant colony algorithm. *Applied Mechanics and Materials.*

[8] Afonso, L. D., Mariani, V. C. & Dos Santos Coelho, L., Modified imperialist competitive algorithm based on attraction and repulsion concepts for reliability-redundancy optimization, *Expert Systems with Applications* 2013; 40 (9), 3794-3802.

[9] Sakalli, U. S., A simulated annealing approach for reliability-based chance-constrained programming, *Applied Stochastic Models in Business and Industry* 2014; 30 (4), 497-508.

[10] Najafi, A. A., Karimi, H., Chambari, A. & Azimi, F., Two metaheuristics for solving the reliability redundancy allocation problem to maximize mean time to failure of a series-parallel system, *Scientia Iranica* 2013; 20 (3), 832-838.

[11] Sharifi, M., Mousa Khani, M. & Zaretalab, A., Comparing Parallel Simulated Annealing, Parallel Vibrating Damp Optimization and Genetic Algorithm for Joint Redundancy-Availability Problems in a Series-Parallel System with Multi-State Components, *Journal of Optimization in Industrial Engineering* 2014; 7 (14), 13-26.

[12] Li, W. & Zuo, M. J., Reliability evaluation of multi-state weighted k-out-of-n systems, *Reliability Engineering and System Safety* 2008; 93 (1), 161-168.

[13] Khorshidi, H. A., Gunawan, I. & Ibrahim, M. Y., Multi-objective optimization model on reliability-redundancy allocation problem in multi-state weighted k-out-of-n system, *IEEE Transactions on Industrial Informatics* 2015; (in press)

[14] Khorshidi, H. A., Gunawan, I. & Ibrahim, M. Y., On Reliability Evaluation of Multistate Weighted k-Out-of-n System Using Present Value, *Engineering Economist* 2015; 60 (1), 22-39.

[15] Atashpaz-Gargari, E. & Lucas, C., Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition, *2007 IEEE Congress on Evolutionary Computation, CEC 2007*. 2007, 4661-4667.

[16] Azad, H. R. L., Boushehri, N. S. & Mollaverdi, N., Investigating the application of opposition concept to colonial competitive algorithm, *International Journal of Bio-Inspired Computation* 2012; 4 (5), 319-329.