Available online at http://ijim.srbiau.ac.ir/
Int. J. Industrial Mathematics (ISSN 2008-5621)
Vol. 11, No. 3, 2019 Article ID IJIM-0801, 12 pages
Research Article

Effects of Technical and Organizational Activities on Redundancy Allocation Problem with Selecting Redundancy Strategies Using Memetic Algorithm

M. Sharifi *†, MR. Shahriari ‡, A. Zaretalab §

Received Date: 2016-12-19 Revised Date: 2018-02-12 Accepted Date: 2019-02-07

Abstract

Redundancy allocation problem is one of the most important problems in reliability area. This problem involves the suitable redundancy levels under certain strategies to maximize system reliability under some constraints. However, it has undergone many changes to come closer to the real situations. Selecting the redundancy strategy and using different system configuration are some of these changes. In this paper, we studied the effects of technical and organizational activities on this problem and showed the difference between the system reliability with and without using these activities. Also, we worked on a system containing s subsystems connected serially. Each subsystem contains n_i , $i = 1, 2, \ldots, s$ parallel components that can be selected from m_i , $i = 1, 2, \ldots, s$ different component types and all subsystem components must be identical. Because redundancy allocation problem belongs to Np. Hard problems, we used a new metaheuristic algorithm called memetic competition algorithm for solving the presented problem and compared the results of this algorithm with other solving methods.

Keywords : Reliability; Redundancy allocation problem; Memetic algorithm; Technical and Organizational activities.

1 Introduction

In this paper, we investigated a system containing s subsystems which are serially connected. Each subsystem contains n_i , $i = 1, 2, \ldots, s$ parallel components that can be selected from m_i , $i = 1, 2, \ldots, s$ different component types and all subsystem components must be identical.

In 1968, Fyffe et al. [5] presented the mathematical model of RAP (Redundancy Allocation Problem) with active strategy. The presented model aimed to maximize system reliability under cost and weight constraints and solved their model using dynamic programing. Nakagawa and Miyazaki [10] presented a nonlinear programing of reliability optimization with solving method. In fact they made some changes on Fyffe problem and solved the model with surrogate constraints algorithm. They showed that this model was more effective than dynamic programing. Coit [3]

*Corresponding author. Mani.Sharifi@yahoo.com, Tel: +98(912)2825776.
†Department of Industrial & Mechanical Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran.
‡Department of Management & Accounting, South Tehran Branch, Islamic Azad University, Tehran, Iran.
§Department of Industrial Engineering, Amir Kabir University of Technology, Tehran, Iran.

presented a solving method based on linear programing for an RAP with cold-standby strategy in 2001 and then Ramirez-Marquez and Coit [11] presented a heuristic algorithm for RAP. Tian et al. [15] presented a model based on physical programming and genetic algorithms and Ramirez-Marquez et al. [12] used max-min approach for a series-parallel RAP. In all presented studies, two issues are obvious. First in all mentioned works, the redundancy strategies of subsystems are predefined. Tillman et al. [16] reviewed 144 papers in this area and only 14 papers worked on cold-standby strategy. Considering redundancy strategy as system variable is one way to draw RAP nearer to real situations that Coit [4] carried out. The second issue is that redundancy allocation is considered as the only way to increase system reliability. Tian et al. [14] presented a new approach for optimizing system reliability. Their approach was based on redundancy optimization and effective on performance rates of system components. They also presented a common reliability-redundancy optimization. In this paper, we considered four bases for reliability optimization problem; the number of redundant components in each subsystem, type of redundant components in each subsystem, redundancy strategy of each subsystem, and technical and organizational activities. The last base affects failure rate of a single component or all components of a subsystem. Chern [1] proved that RAP belongs to Np. Hard problems and after that many researchers used metaheuristic algorithms for solving this problem. Coit and Smith [2] used genetic algorithm for solving a series-parallel RAP. Gen and Kim [6] used a GA-based hybrid algorithm. Kulturel-Konak et al. [7] used Tabu search algorithm. Liang and Smith [8] used ant colony algorithm and Liang and Chen [9] used variable neighborhood search algorithm for solving RAP. In this paper, we used MA (Memetic Algorithm) for solving an RAP. The paper is divided into five sections. The second part defines the problem. Third section deals with solving method and in fourth part, we present a numerical example. The last section is conclusion and further studies.

2 Problem definition

2.1 Nomenclature

The parameters that we used in this paper are:

s : Number of subsystems,

n_i : Number of components in subsystems i,

$n_{Max,i}$: Upper bound of allocated component in subsystem i,

m_i : Number of available components in subsystem i,

z_i : Selected component index in subsystem i, $z_i \in (1, 2, \ldots, m_i)$,

Z : Set of z_i, $Z = (z_1, z_2, \ldots, z_s)$,

t : Operation time,

$R(t, z, n)$: System reliability at the time t for design vector z and n,

$\tilde{R}(t, z, n)$: An estimate for $R(t, z, n)$:,

$r_{ij}(t)$: Reliability of j^{th} component type in subsystem i at the time t,

λ_{ij} : Shape parameter of Gama p.d.f,

k_{ij} : Scale parameter of Gama p.d.f,

C : Maximum acceptable cost of the system,

W : Maximum acceptable weight of the system,

c_{ij} : Cost of j^{th} component in subsystem i,

w_{ij} : Weight of j^{th} component in subsystem i,

$\rho_i(t)$: Continuous detect switch in subsystem i at the time t,

ρ_i : Failure detect switch in subsystem i,

tkh_{ij} : Binary variables related to technical activities of ${}_j$sty component in subsystem i,

tk_i : Binary variables related to organizational activities of components in subsystem i,

ckh_{ij} : Cost of technical activities of $j$sty component in subsystem i,

ck_i : Cost of organizational activities of components in subsystem i,

$$\sigma_i(t,j) = \begin{cases} 1 & ; \ Perfect\ switch \\ \rho_i(t) & ; \ Switch\ reliability\ at \\ & \quad time\ t \\ \rho_i^{j-1} & ; \ Switch\ active\ when \\ & \quad a\ defect\ causes \end{cases}$$

2.2 Basic mathematical model

The mathematical model presented by Coit [4] is as follows:

$$\begin{array}{lll} \max & : & R(t,z,n) \\ s.t & : & \sum_{i=1}^{s} c_{iz_i} n_i \le C, \\ & & \sum_{i=1}^{s} w_{iz_i} n_i \le W \\ & & n_i \in (1,\,2,\,\ldots,n_{\max,i}) \\ & & z_i \in (1,\,2,\,\ldots,m_i) \end{array} \tag{2.1}$$

The system reliability in objective function of equation 2.1 depends on switch type and calculated using equations 2.2 (for continuous detecting switch SP1) and (for failure reacted switch SP2).

$$\begin{aligned} R(t,z,n) &= \prod_{i\in A}\left\{1-(1-r_{i,z_i}(t))^{n_i}\right\} \\ &\times \prod_{i=N} r_{i,z_i}(t) \times \prod_{i\in S}\left\{r_{i,z_i}(t)\right. \\ &\left. + \sum_{j=1}^{n_i-1}\int_0^t \rho_i(u)\, f_{i,z_i}^{(j)}(u)\, r_{i,z_i}(t-u)\,du\right\} \end{aligned} \tag{2.2}$$

$$\begin{aligned} R(t,z,n) &= \prod_{i\in A}\{1-(1-r_{i,z_i}(t))^{n_i}\} \\ &\times \prod_{i=N} r_{i,z_i}(t) \times \prod_{i\in S}\left\{r_{i,z_i}(t)\right. \\ &\left. + \sum_{j=1}^{n_i-1}\rho_i^j\int_0^t f_{i,z_i}^{(j)}(u)\, r_{i,z_i}(t-u)\,du\right\} \end{aligned} \tag{2.3}$$

In equations 2.2 and 2.3, $f_{i,z_i}^j(t)$ is the sum of j time failure of z_i th component in subsystem$i$. Parameter A defines the set of subsystems with active redundancy strategy, parameter S refers to the set of subsystems with cold-standby redundancy strategy and N refers to the set of a subsystem with only one component (No redundancy strategy). Coit approximated the system reliability as follows:

$$\begin{aligned} R(t,z,n) &= \prod_{i\in A}\{1-(1-r_{i,z_i}(t))^{n_i}\} \\ &\times \prod_{i=N} r_{i,z_i}(t) \times \prod_{i\in S}\left\{r_{i,z_i}(t)+\sigma_i(t,n_i)\right. \\ &\left.\times \exp(-\lambda_{i,z_i}t)\sum_{l=k_{i,z_i}}^{k_{i,z_i}n_i-1}\frac{(\lambda_{i,z_i}t)^l}{l\,!}\right\} \end{aligned} \tag{2.4}$$

$$r_{i,z_i}(t) = \exp(-\lambda_{i,z_i}t)\sum_{l=0}^{k_{i,z_i}-1}\frac{(\lambda_{i,z_i}t)^l}{l\,!} \tag{2.5}$$

Example 2.1 *For clarifying this system, Fyffe et al. [15] example has been solved. The presented system contains 14 subsystems and for each subsystem 3 or 4 different component types are allocated.*

The parameter of components and k-Erlang distribution can be found in Fyffe et al. study [15]. The object is to maximize system reliability in 100 hours working time. The upper limits of system cost and weight are $C = 130$ and $W = 170$. The system has a continuous detecting switch and the switch reliability in 100 hours working time is considered as$P = 0.99$. The maximum number of components in each subsystem is 6 and two active and cold-standby redundancy strategies are available for each sub-system. The optimal solution presented for only Active strategy that presented in Fyffe et al. (1968) is $R = 0.9700$, only cold-standby strategy that presented in Coit [10] is $R = 0.9863$, considering redundancy strategy as the system variable [3] is $R = 0.9875$ and the results that presented by Tavakoli-Moghaddam et al. [13], using a hybrid GA $R = 0.9705$.

2.3 Technical and organizational activities

These activities affect components performance rate and decrease their failure rate. These activities are presented as follows.

2.3.1 Technical activities

This activity effects components of the subsystem individually and its cost depends on the number of affected components. Parameter tkh_{ij} is a binary variable. If the value of this parameter is equal to 1, the technical activity will be done on component type j in subsystem i.

2.3.2 Organizational activities

This activity effects all components of the subsystem and its cost is fixed and does not depend on the number of affected components. Parameter tk_i is a binary variable. If the value of this parameter is equal to 1, the organizational activity will be done on all components in subsystem i.

We add these activities to the basic model to draw the RAP nearer to real situations.

2.4 Assumptions

- System consists of s serially connected subsystems and the components in each subsystem are parallel,
- Only one type of components can be allocated to each subsystem,
- Redundancy strategy of each subsystem is a system variable and can be active or cold-standby,
- All components are binary, non-repairable with k-Erlang distribution,
- All system parameters are fixed and predefined,
- Components failures are independent,
- Technical and organizational activities effect components failure rates.

2.5 Mathematical model

The presented mathematical model is as follows:

$$\begin{aligned}
\max \quad & R\,(t, z, n) \\
s.t \quad : \; & \sum_{i=1}^{s} \Big\{ n_i(c_i z_i \\
& +ckh_{iz_i}.tkh_{iz_i}) + ck_i.tk_i \Big\} \leq C, \\
& \sum_{i=1}^{s} w_{iz_i}.n_i \leq W \\
& \lambda_{iz_i} = \lambda_{iz_i}\left(1 - \alpha_{iz_i}.tkh_{iz_i}\right), \\
& \lambda_{iz_i} = \lambda_{iz_i}\left(1 - \beta_i.tk_i\right) \\
& 0 \leq n_i \leq n_{\max,i},\ z_i \geq 0,\ \lambda_{ij} \geq 0, \\
& tkh_{ij} = 0,\ 1,\ tk_i = 0, \\
& 1,\ i = 1,\ 2,\ \ldots, s,\ j = 1,\ 2,\ \ldots, m_i
\end{aligned} \tag{2.6}$$

In Equation 2.6, parameters α_{i,z_i} and β_i are the technical and organizational activities influencing on components failure rates. These parameters are constant and their values are $\alpha_{i,z_i} = 0.3$ and $\beta_i = 0.1$ for $i = 1, 2, \ldots, s$.

Example 2.2 *In this example we add technical and organizational activities to example 2.1. The cost of technical and activities are presented in Table 1. The other problem parameters are similar to problem 2.1 parameters.*

Memetic algorithm

One of the most popular methods in RAP is GA. Memetic Algorithm (MA) is an algorithm combining GA with a local search, i.e., GA + Local search=MA. The pseudo code of MA is shown in Figure 1.

In GA, after producing initial population, the roulette wheel is used for selecting parents with crossover operator. Also the mutation operator on initial population makes some offspring and then the new generation replaces the old one and this process repeats until algorithm meets the stop condition. Memetic local search attempts to achieve the solutions which have the weight and cost close to the boundaries of its constraints. These solutions have the better objective functions than the other solutions. In local search at first the subsystems have been ranked based on their reliabilities. The first rank allocates to the subsystem with the lowest reliability and the

Table 1: Cost of technical and organizational activities.

i	ckh_{i1}	ckh_{i2}	ckh_{i3}	ckh_{i4}	ck_i
1	0.3333	0. 3333	0.6667	0.6667	2.5000
2	0.6667	0.3333	0. 3333	–	2.7500
3	0.6667	1.0000	0.3333	1.3333	2.5000
4	1.000	1.3333	1.6667	–	2.0000
5	0.6667	0.6667	1.0000	–	2.0000
6	1.0000	1.0000	0.6667	0.6667	2.5000
7	1.3333	1.3333	1.6667	–	3.0000
8	1.0000	1.6667	2.0000	–	2.5000
9	0.6667	1.0000	1.3333	1.0000	2.7500
10	1.3333	1.3333	1.6667	–	2.5000
11	1.0000	1.3333	1.6667	–	2.7500
12	0.6667	1.0000	1.3333	1.6667	3.0000
13	0.6667	1.0000	0.6667	–	2.0000
14	1.3333	1.3333	1.6667	2.0000	2.5000

s^{th} rank to the subsystem with the highest reliability. Each subsystem randomly selects an improved policy based on its rank. There is two different improved policies. The flowcharts of these two policies are presented in figures 2 and 3. In this flowchart, n is the number of subsystem components.

Using these two types of improved policies, the obtained solutions are equal or better that the old ones. The ranges of MA parameters are presented in Table 2 and the results in Table 3. Also the results of RSM are shown in figures 4, 5 and the optimal parameters are presented in Figure 6. In this tables and figures, *npop* refers to the population size, P_C is the crossover probability and P_M is the mutation probability.

Improved policy type 1

In this type of improved policy, the algorithm attempts to improve the subsystem reliability first by redundancy allocation and then by technical and organizational activities.

Improved policy type 2

In this type of improved policy, the algorithm attempts to improve the subsystem reliability first by technical and organizational activities and then by redundancy allocation.

3 Response surface methodology

RSM[1] is a set of statistical methods for improving and optimizing processes. In this method, the response of a process is optimized by altering some input parameters. We used this method to optimize the algorithm parameters. The ICA algorithm has 5 parameters, so for parameter tuning of algorithm we need $2^4 = 16$ corner points, $2 \times 4 = 8$ axial points, and 5 central points. Using RSM, 28 combinations of parameters are considered and the stop condition of algorithm is 200 iterations. The parameter boundaries and optimal parameters are presented in Tables 2 and 3. Based on RSM, the optimal parameters are found as $npop = 100,\ p_c = 0.5,\ n_m = 0.3$.

[1] Response surface methodology

```
Parameter Setting (number of iteration, Pop Size,
Selection Strategy, Crossover Size, Mutation Size)

Best solution = [ ]

For I = 1 to number of Pop Size do
     Population (I) =Randomly

     Fitness Population (I) =evaluate (Population (I))

     Individual of Population (I) = local-search
    (Population (I))

End

For it = 1 to number of iteration do

For I = 1 to number of Crossover Size do

    Parents=Roulette wheel Selection (Population)
    Childs of Crossover = Crossover (Parents)

    Fitness Childs of Crossover=evaluate (Childs of
Crossover)

    Individuals of Crossover = local-search (Childs of
Crossover)

End

For I = 1 to number of Mutation Size do

    x = Roulette wheel Selection (Population)
    Childs of Mutation = Mutation (x)

     Fitness Childs of Mutation =evaluate (Childs of
    Mutation)

    Individuals of Mutation = local-search (Childs of
Mutation)

End

Population = merge (Individuals of Population,
Individuals of Crossover, Individuals of Mutation)

Update (Best solution)

End
```

Figure 1: Pseudo-code of MA

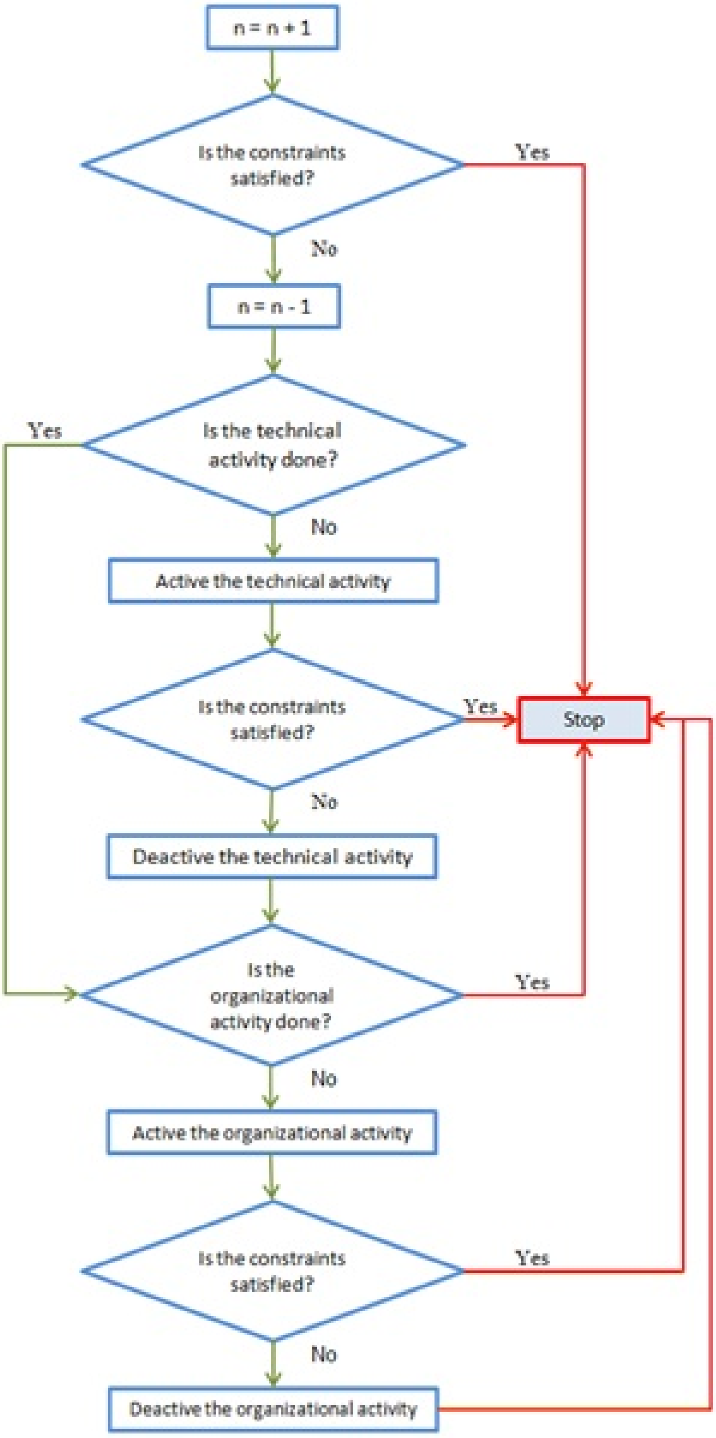

Figure 2: Improved policy type 1

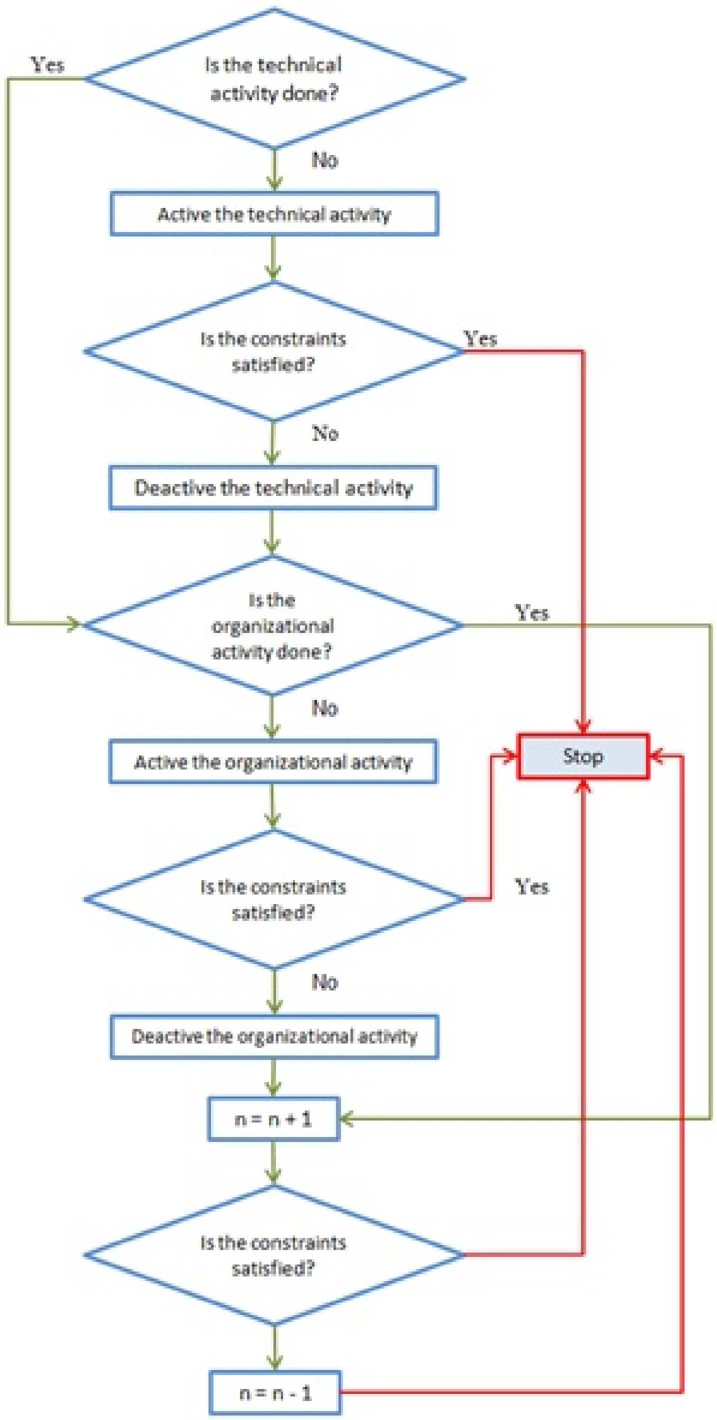

Figure 3: Improved policy type 2

Term	Coef	SE Coef	T	P
Constant	0.937528	0.042039	22.301	0.000
npop	-0.000045	0.000342	-0.131	0.899
pc	0.144293	0.147731	0.977	0.357
PM	0.105181	0.068014	1.546	0.161
npop*npop	0.000001	0.000002	0.446	0.667
pc*pc	-0.085952	0.121159	-0.709	0.498
PM*PM	0.014048	0.121159	0.116	0.911
npop*pc	-0.000050	0.000282	-0.177	0.864
npop*PM	-0.000080	0.000282	-0.284	0.784
pc*PM	-0.162500	0.070513	-2.305	0.050

S = 0.00199441 PRESS = 0.000259747
R-Sq = 56.24%

Figure 4: Memetic estimated regression coefficients for $R(t)$

Analysis of Variance for R(t)

Source	DF	Seq SS	Adj SS	Adj MS	F	P
Regression	9	0.000041	0.000041	0.000005	1.14	0.431
Linear	3	0.000017	0.000017	0.000006	1.43	0.303
npop	1	0.000009	0.000000	0.000000	0.02	0.899
pc	1	0.000002	0.000004	0.000004	0.95	0.357
PM	1	0.000005	0.000010	0.000010	2.39	0.161
Square	3	0.000002	0.000002	0.000001	0.18	0.905
npop*npop	1	0.000000	0.000001	0.000001	0.20	0.667
pc*pc	1	0.000002	0.000002	0.000002	0.50	0.498
PM*PM	1	0.000000	0.000000	0.000000	0.01	0.911
Interaction	3	0.000022	0.000022	0.000007	1.81	0.224
npop*pc	1	0.000000	0.000000	0.000000	0.03	0.864
npop*PM	1	0.000000	0.000000	0.000000	0.08	0.784
pc*PM	1	0.000021	0.000021	0.000021	5.31	0.050
Residual Error	8	0.000032	0.000032	0.000004		
Lack-of-Fit	5	0.000019	0.000019	0.000004	0.86	0.589
Pure Error	3	0.000013	0.000013	0.000004		
Total	17	0.000073				

Figure 5: Memetic analyses of variance for $R(t)$

Table 2: Memetic parameters range.

	Lower value	Upper value
npop	50	100
P_c	0.5	0.7
p_m	0.1	0.3

Table 3: Optimal values of Memetic algorithm.

npop	Pc	Pm	R(t)
50	0.5	0.1	0.9872
100	0.5	0.1	0.9920
50	0.7	0.1	0.9942
100	0.7	0.1	0.9947
50	0.5	0.3	0.9946
100	0.5	0.3	0.9948
50	0.7	0.3	0.9913
100	0.7	0.3	0.9948
50	0.6	0.2	0.9942
100	0.6	0.2	0.9949
75	0.5	0.2	0.9939
75	0.7	0.2	0.9924
75	0.6	0.1	0.9942
75	0.6	0.3	0.9941
75	0.6	0.2	0.9933
75	0.6	0.2	0.9940
75	0.6	0.2	0.9936
75	0.6	0.2	0.9895

Table 4: Memetic parameters range.

				tkh				
i	Z_i	n_i	R S	i_1	i_2	i_3	i_4	tk_1
1	3	2	C	0	0	1	0	0
2	1	2	C	1	0	0	0	0
3	3	2	C	0	0	1	0	1
4	2	2	C	0	1	0	0	0
5	2	3	A	0	0	0	0	0
6	3	2	C	0	0	1	0	0
7	2	2	C	0	1	0	0	0
8	3	2	C	0	0	1	0	0
9	2	2	C	0	0	0	0	0
10	2	2	C	0	1	0	0	1
11	2	2	C	0	0	0	0	1
12	3	2	C	0	0	1	0	1
13	2	2	A	0	0	0	0	1
14	1	2	C	1	0	0	0	0
$R(t) = 0.9917$				$w = 165$			$c = 129.25$	

4 Problem solving

Using optimal parameters of MA, we solved example 2.2 and the results are presented in Table

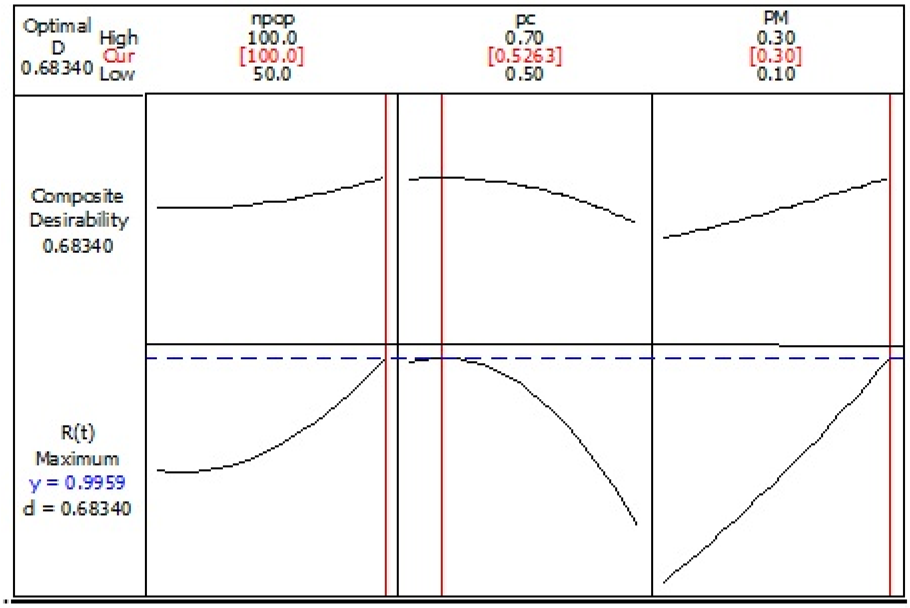

Figure 6: Memetic optimal parameters

4. The optimal solution with technical and organizational activities is $R(100) = 0.9917$ that is better than the optimal solution without using these activities.

5 Conclusion and further studies

In this paper, we showed that adding technical and organizational activities increases the reliability of the system besides other technics like redundancy allocation. In other word, in a series-parallel system not only redundancy allocation can increase the system reliability, improving the components performance rate is a helpful way, too. In example 2.2, after adding technical and organizational activities, the optimal solution increases from $R(100) = 0.9875$ to$R(100) = 0.9917$. For further studies, other metaheuristic algorithms can be used to get better solutions. Also these activities can be added to other models to evaluate their effects.

References

[1] M. S. Chern, On the computational complexity of reliability redundancy allocation in a series system, *Operations research letters* 11 (1992) 309-315.

[2] D. W. Coit, A. E. Smith, Reliability optimization of series-parallel systems using a genetic algorithm, *IEEE Transactions on Reliability* 45 (1996) 254-260.

[3] D. W. Coit, Cold-standby redundancy optimization for nonrepairable systems, *Iie Transactions* 33 (2001) 471-478.

[4] D. W. Coit, Maximization of system reliability with a choice of redundancy strategies, *Iie Transactions* 35 (2003) 535-543.

[5] D. E. Fyffe, W. W. Hines and N. K. Lee, System reliability allocation and a computational algorithm, *IEEE Transactions on Reliability* 17 (1968) 64-69.

[6] M. Gen, J. R. Kim, GA-based reliability design: state-of-the-art survey, *Computers & Industrial Engineering* 37 (1999) 151-155.

[7] S. Kulturel-Konak, A. E. Smith, D. W. Coit, Efficiently solving the redundancy allocation problem using tabu search, *Iie Transactions* 35 (2003) 515-526.

[8] Y. C. Liang, A. E. Smith, An ant colony optimization algorithm for the redundancy allocation problem (RAP), *IEEE Transactions on Reliability* 53 (2004) 417-423.

[9] Y.-C. Liang, Y. C. Chen, Redundancy allocation of series-parallel systems using a variable neighborhood search algorithm, *Reliability Engineering & System Safety* 92 (2007) 323-331.

[10] Y. Nakagawa, S. Miyazaki, Surrogate constraints algorithm for reliability optimization problems with two constraints, *IEEE Transactions on Reliability* 30 (1981) 175-180.

[11] J.E. Ramirez-Marquez, D. W. Coit, A heuristic for solving the redundancy allocation problem for multi-state series-parallel systems, *Reliability Engineering & System Safety* 83 (2004) 341-349.

[12] J.E. Ramirez-Marquez, D. W. Coit, A. Konak, Redundancy allocation for series-parallel systems using a max-min approach, *Iie Transactions* 36 (2004) 891-898.

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

[14] Z. Tian, G. Levitin, M. J. Zuo, A joint reliability redundancy optimization approach for multi-state series parallel systems, *Reliability Engineering & System Safety* 94 (2009) 1568-1576.

[15] Z. Tian, M. J. Zuo, H. Huang, Reliability-redundancy allocation for multi-state series-parallel systems, *IEEE Transactions on Reliability* 57 (2008) 303-310.

[16] F. A. Tillman, C.-L. Hwang, W. Kuo, Determining component reliability and redundancy for optimum system reliability, *IEEE Transactions on Reliability* 26 (1977) 162-165.

Mani Sharifi was born in 1973 in Tehran, Iran. Mani is an associate professor at Qazvin Islamic Azad University (QIAU). Mani holds a B.Sc. degree from Qazvin Islamic Azad University in 1996, M.Sc. degree from south Tehran branch Islamic Azad University in 1998, and Ph.D. degree from Tehran Research and Science Islamic Azad University in industrial engineering filed in 2006. He is a member of the industrial engineering department at QIAU and he was senior advisor to the president of Iran National Steel Industrial Group (INSIG) and vice-president and member of board of directors of Esteghlal Khouzestan F.C.

Mohammadreza Shahriari received his Ph.D. in Operational Research from the Islamic Azad University in 2009. He worked out more than 2 years to find out he how can present and solve a Time -Cost Trade off problem based on time value of money, considering to each crashing. He was also six years in charge of chancellor of Islamic Azad University in Dubai (UAE) and more than one year at the same position in Islamic Azad University in Oxford (UK). He has had another executive academic mandate, like, establishing an Irani University in Germany, Italy, Russia and Oman. After almost running the abovementioned mandate, he again backed to his academic and teaching activities in the university.

Arash Zaretalab is a Ph.D. candidate in the department of Industrial Engineering at Amirkabir University of technology (Tehran polytechnic) in Iran. He received his B.Sc. and M.Sc. degrees in Industrial Engineering from Islamic Azad University, Qazvin Branch (QIAU). His research interests are in Reliability Engineering, Combinatorial Optimization, Machining process optimization, computational intelligence and Data Mining.