A Hybrid TS-DE Algorithm for Reliability Redundancy Optimization Problem

Yubao Liu
College of Computer Science and Technology, Jilin University, Changchun, China
College of Computer Science and Technology, Changchun University, Changchun, China
Email: liuyb@ccu.edu.cn, lyb_mailbox@126.com

Guihe Qin*
College of Computer Science and Technology, Jilin University, Changchun, China
Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun, China
Email: qingh@jlu.edu.cn

***Abstract*—In this paper, a hybrid TS-DE algorithm based on Tabu search and differential evolution algorithm is proposed to solve the reliability redundancy optimization problem. A differential evolution algorithm is embedded in Tabu search algorithm. TS is applied for searching solutions space, and DE is used for generating neighborhood solutions. The advantages of both algorithms are considered simultaneously. And an adaptive hybrid TS-DE approach is developed to solve three benchmark reliability redundancy allocation problems. By comparing with other algorithms reported in previous literatures, experimental results show that the proposed method is effective and efficient for solving the reliability redundancy optimization problem.**

***Index Terms*—nonlinear programming, Tabu search, differential evolution, reliability optimization, redundancy allocation**

I. Introduction

The reliability optimization problem is very important in industry and has attracted attention in academic field and engineering fields. In general, two major ways have been used to improve system reliability. The first way is by increasing the reliability of components, and the second way is by using redundant components in the subsystems. In the first way, sometimes it cannot meet our requirements even though the currently highest reliable components are used. The second way is by choosing the components reliability combination and redundancy levels to arrive the highest system reliability. But the cost, weight, volume are all increased. So it is necessary that a trade-off is achieved between these two options for constrained reliability optimization. Such reliability allocation and redundancy allocation problem is called as RRAP (reliability-redundancy allocation problem) [1, 2, 3].

Manuscript received September 16, 2013; revised February 18, 2014; accepted March, 2014..

*Corresponding author.
E-mail address: qingh@jlu.edu.cn

A reliability redundancy allocation problem of maximizing the system reliability subject to multiple nonlinear constraints [4, 7, 8, 9] belongs to mixed-integer programming problems. It can be formulated as following model uniformly:

$$\text{Max } R_s = f(r, n)$$
$$\text{s.t. } g_j(r, n) \le b_j,\ j = 1.. m;\ n_j \in \text{positive integer},\ 0 \le r_j \le 1 \quad (1)$$

Where r_i is the reliability of subsystem i, and n_i is the number of components of subsystem i. The f (.) is the objective function for the system reliability; the g_j (.) is the jth constraint function and b_j is the jth upper limitation of the system; the m is the number of subsystems. The goal is to determine the number of redundant components and the components' reliability in each subsystem so as to maximize the overall system reliability.

RRAP has been proven to be NP-hard problem. There are many different optimization technologies have been presented to resolve it. The approaches called heuristics and meta-heuristics have been widely researched and applied [5, 6, 12].They offer feasible solution within reasonable computational time. Hsieh [5] used a linear programming approach to solve the RRP-MCC with nonlinear constraints. Coit and Smith [13] presented a genetic algorithm (GA) to solve the Reliability-Redundancy problem. Hsieh et al. [14] used genetic algorithm to solve reliability design problems of series systems, series-parallel systems and complex (bridge) systems. You and Chen [15] proposed a greedy genetic algorithm for series–parallel redundant reliability problems. Ta_Cheng Chen[16] used an immune algorithm-based approach to solve the RRP-MCC problem of series system, series–parallel system, and complex(bridge) systems and overspeed protection system. Hsieh and You [17] presented an immune based two-phase approach to solve the reliability-redundancy allocation problem. First, an immune algorithm (IA) is used to get preliminary solutions. Second, the quality of solutions was improved by a procedure to obtain the last solutions. The result showed that the solutions are

doi:10.4304/jcp.9.9.2050-2057

superior to those best solutions of other approaches in the literature. Liang and Chen[18] proposed a variable neighborhood search (VNS) with an adaptive penalty function. This method improved the performance and the solution quality were as good as others. Zavala et al.[21] proposed a particle swarm optimization (PSO) approach named PESDRO to solve a bi-objective redundant reliability problem; And the reliability redundant problems of series system, parallel system and K-out-of-N system are resolved. Zou et al. [19,20] used global harmony search algorithm to solve RRAP. Leandro dos Santos Coelho [22] presents a PSO approach based on Gaussian distribution and chaotic sequence (PSO-GC) to solve the reliability–redundancy allocation problems of complex (bridge) system and overspeed protection system. The PSO-GC has got better solutions than the classical PSO. Harish Garg and S.P. Sharma [23] used PSO to solve multi-objective reliability redundancy allocation problem of a series system. Agarwal and Sharma [24] applied ant colony optimization (ACO) algorithm with an adaptive penalty function to redundancy allocation problem. Nabil Nahas et al. [25] coupled ant colony optimization algorithm with degraded ceiling local search method for redundancy allocation of series–parallel systems. Mohamed Ouzineb [26] presented Tabu search (TS) approach to solve the redundancy allocation problem for multi-state series–parallel systems. Afonso et al. [27] used imperialist competitive algorithm (ICA) to resolve RRAP.

Recently some hybrid meta-heuristic methods have been proposed to solve the reliability redundant allocation problems. Nima Safaei et al. [28] presented an Annealing-based PSO (APSO) method. Even though APSO didn't obtain the better solution than other well-known meta-heuristic method, it applied Metropolis-Hastings strategy and affected the performance of the basic PSO. Wang and Li [29] presented a coevolutionary differential evolution with harmony search algorithm (CDEHS) to solve the reliability redundancy optimization problem. The method divided the problem into two parts: the continuous part and the integer part. The continuous part evolved by differential evolution algorithm, and the integer part evolved by harmony search approach. Thus two populations evolve simultaneously and cooperatively to get the solutions. Shi-Ming Chen et al. [37] proposed SAABC algorithm coupled simulated annealing algorithm(SA) with artificial bee colony (ABC) algorithm. The SAABC outperformed ABC and GABC in terms of convergence speed and accuracy.

Although the above methods can get the near optimal solution in a limited computational time, generally there are some problems such as slow convergence speed and low precision of the and so on. Moreover some methods can not consider the problem on the balance of convergence speed and accuracy.

In this paper, a hybrid Tabu search and differential evolution algorithm is proposed. DE algorithm has quick convergence speed, but it will lose the information of no selected individual. On the contrary, TS method has strong memory function, and it can get higher accuracy solution. Whereas its convergence speed is slow. This method considers the advantages of both algorithms simultaneously. It is used to solve three problems on reliability redundancy optimization. The experimental results show that the proposed TS-DE method has higher precision, faster convergence speed, and is more effective for reliability redundancy optimization problem.

The paper is organized as follows. Section 2 provides the general procedure of the original Tabu search (TS) and differential evolution algorithm (DE). In Section 3, a hybrid TS-DE algorithm based on TS and DE is proposed. The simulation results and comparisons are provided in Section 4. Finally, the conclusion of the paper is summarized and the future work is directed in Section 5.

II. The Hybrid Algorithm based on TS and DE

A. *The Tabu Search Algorithm (TS)*

Tabu search is a metaheuristic algorithm that has become the methods of choice for solving many complex applied optimization problems. It combines local search with a Tabu list in order to avoid searching the same solutions repeatedly. It has been proved to be very effective in many optimization problems.

Tabu search is an iterative procedure where moving from a current solution to a new solution in a neighborhood at each iteration. The steps are described as follows:

Step 1. The initial solution is generated randomly. The parameters are initialized. The Tabu list is set null.

Step 2. The stop criterion is checked. If it is satisfied, the search is terminated and the current best solution is accepted. Otherwise, continue to the following steps.

Step 3.Generating all (or some) neighborhood solution, and choosing some candidate solutions.

Step 4.The aspiration criterion is checked. If one candidate solution satisfies aspiration criterion, it is accepted to be new current best solution, then return to step 2. Otherwise continue to the following steps.

Step 5. Choosing the best status from the candidate solutions to be current best solution and updating Tabu list.

Step 6. Go to step 2.

B. *Differential Evolution Algorithm (DE)*

Differential evolution algorithm was first presented by Price and Storn in 1995. It is a kind of evolutionary algorithm using real number code. Compared with the traditional genetic algorithm, differential evolution algorithm generates new species by mutation and crossover operations on the current population, and then adopts the competition strategy of one-to-one to update the population. So DE is simple, effective and efficient method for solving optimization problems[30,31,32]. At present, a variety of the DE algorithms have been proposed. Among them the DE/rand/1/bin has been widely used[32]. Here, we choose DE/rand/1/bin with mutation, crossover and selection operations. The procedure is described as follows:

Step 1: Initializing parameters.

The parameters are: F, CR and M. Where F is scale factor, CR is crossover rate; M is the size of population.

Step 2: Randomly generating initial population of individuals.

Step 3: Evaluate all individuals of the population.

Step 4: Mutation.

The mutation operator is:

$$V_i^{k+1} = x_{i3}^k + F\times (x_{i1}^k - x_{i2}^k) \tag{2}$$

v_i^{k+1} is the trial vector. The x_{i1}^k, x_{i2}^k, x_{i3}^k are three different individuals randomly selected from kth generation population, i_1,i_2,i_3 is random number ranged from 1 to M, and mutation factor F is a scale coefficient.

Step 5: Crossover.

The crossover operator is:

$$u_{i,j}^{k+1} = \begin{cases} v_{i,j}^{k+1} & \text{if rand} < CR \text{ or } j = r_d \\ x_{i,j}^k & \text{otherwise} \end{cases} \tag{3}$$

u^{k+1} is the offspring vector. Where, r_d is a random integer between 1 and D (D is the number of variables). CR is a real number between 0 and 1.

Step 6: Selection.

The selection operation is performed as follows:

$$x_i^{k+1} = \begin{cases} u_i^{k+1} & \text{if } f(u_i^{k+1}) < f(x_i^k) \\ x_i^k & \text{otherwise} \end{cases} \tag{4}$$

If the fitness of the offspring is better than that of the parent, the offspring u_i^{k+1} is selected to replace the parent x_i^k.

Step 7: stopping criterion.

If the stopping criterion is met, the process is end. Otherwise, go back to Steps 4.

III. The Hybrid TS-DE Algorithm based on TS and DE

In DE algorithm, if an individual is not selected the information of that individual is lost, but TS algorithm has memory ability. In DE algorithm the new candidate solution is generated randomly from the population. This method can increase the ability of getting the global optima, but it cannot ensure a better solution, TS can promote the search for an optimal solution. But TS may waste resources on poor individuals without a selection operator. In order to increase the convergence speed and to produce better quality solution, a differential evolution algorithm is embedded in Tabu search algorithm. TS is applied for searching solutions space, and DE is used for generating neighborhood solutions. In this way the TS can help getting a better solution of each generation in DE algorithm, and accept the best current solution by using aspiration criterion. These consider the advantages of both algorithms simultaneously.

In the original DE algorithm, scale factor F and crossover rate CR are set to fixed values for all solutions. In order to improve the performance of DE algorithm, we have introduced adaptive parameters to get better feasible solutions. So an adaptive adjustment strategy of modifying scale factor F and crossover rate CR is developed. The formulas are described as follows:

$$F=F_0+\eta\times\sin(K/MAXCOUNT\times 2\pi) \tag{5}$$

$$CR=CR_0\times\sin(K/MAXCOUNT\times \pi/2) \tag{6}$$

Where k is the current number of iterations, MAXCOUNT is the total counts of iterations. And η is the coefficient between 0 and 1. F_0 and is the initial value of F, CR_0 is the initial value of CR. And we assumed that the values of F and CR are in the range of [0.0, 2.0] and [0.0, 1.0] respectively.

The hybrid approach is described as follows:

Step 1. Initializing parameters. The Tabu list is set null.

Step 2. Initializing a random population x, Generate current best solution xbest from x.

Step 3.The stop criterion is checked. If it is satisfied, the search is terminated and the current best solution is accepted. Otherwise, continue to the following steps.

Step 4.Generating neighborhood solution by DE algorithm, and choosing the best one from candidate solutions.

Step 5.The aspiration criterion is checked. If one candidate solution satisfies aspiration criterion, it is accepted to be new current best solution, then return to step 2. Otherwise continue to the following steps.

Step 6. Choose the best status from the candidate solutions to be current best solution and updating Tabu list. Go to step 3.

The main procedure of Generating neighborhood by DE algorithm is shown in Fig. 1:

For i = 1 to M
    Randomly generate three integers i_1, i_2 and i_3 in [1, M],
    and $i_1 \neq i_2 \neq i_3 \neq i$.
    $v_i^{k+1}= x_{i3}^k+ (F_0+\sin(K/MAXCOUNT\times 2\pi))\times (x_{i1}^k- x_{i2}^k)$
    Randomly generate an integer r_d in the range [1, N]
    For j = 1 to N
        If rand < $CR_0\times\sin(K/MAXCOUNT*\pi/2)$ or j = r_d
            $u_{i,j}^{k+1} = v_{i,j}^{k+1}$
        Elseif
            $u_{i,j}^{k+1} = x_{i,j}^k$
        End If
    EndFor
    If $f(u_i^{k+1}) < f(x_i^k)$
        $x_i^{k+1}= u_i^{k+1}$
    Elseif
        $x_i^{k+1}= x_i^k$
    End If
EndFor
Choosing the best solution from this generation candidate solutions

Figure 1. Pseudo code of Generating neighborhood by DE algorithm

IV. Simulations and Comparisons

In this section, we implement the simulations based on three benchmark problems to test the performances of the proposed TSDE for reliability-redundancy optimization problems. And we compared the TSDE with some other typical algorithms from the literatures.

A penalty function method is used to handle constrains, it is described as follows:

$$\min \ F(x) = -f(x) + \lambda \sum_{j=1}^{p} \max \{0, g_j(x)\}^2 \tag{7}$$

Where F (x) represents penalty function, f (x) represents objective function. $g_j(x)$, (j = 1, 2, p) represents the jth constraint, and λ is a large positive constant which imposes penalty on unfeasible solutions, and it is named as penalty coefficient.

A. *Problem1: Series-parallel System*

The Series-parallel system [10][18][34] is shown as Figure 2:

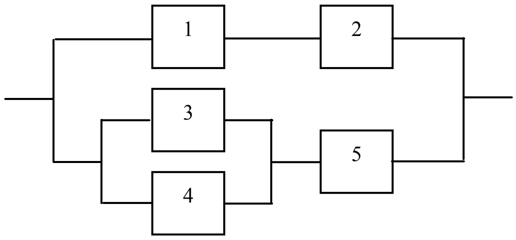

Figure 2. Series-parallel system

This problem is formulated as follows:

$$\begin{aligned}
&\text{Max } f(r,n) = 1-(1-R_1R_2)(1-(1-(1-R_3)(1-R_4))R_5) \\
&\text{s.t.} \quad g_1(r,n) = \sum_{i=1}^{m} w_i v_i^2 n_i^2 \le V \\
&\qquad g_2(r,n) = \sum_{i=1}^{m} \alpha_i(-1000/\ln r_i)^{\beta_i}(n_i + \exp(n_i/4)) \le C \qquad (8) \\
&\qquad g_3(r,n) = \sum_{i=1}^{m} w_i n_i \exp(n_i/4)) \le W
\end{aligned}$$

Where m is the number of subsystems, n_i is the number of components of subsystem i, $R_i(n_i)$ is the reliability of subsystem i, $f(r,n)$ is the reliability of the system; The w_i is the weight of each component in subsystem i, v_i is the volume of each component in subsystem i; The r_i is the reliability of each component in subsystem i; The item $\alpha_i(-1000/\ln r_i)^{\beta i}$ is the cost of each component in subsystem i, the parameters α_i and β_i is the constant value(usually assume that have been given),1000 is the task time of the components(it is commonly expressed in T_m); The V is the upper limit of total volume of the system, C is the upper limit of total cost of the system, W is the upper limit of total weight of the system. The parameters for this problem are listed in Table Ⅰ:

TABLE I.

THE PARAMETERS OF SERIES-PARALLEL SYSTEM.

Subsystem i	$10^5\alpha_i$	β_i	$w_iv_i^2$	w_i	V	C	W
1	2.500	1.5	2	3.5	180	175	100
2	1.450	1.5	4	4.0			
3	0.541	1.5	5	4.0			
4	0.541	1.5	8	3.5			
5	2.100	1.5	4	4.5			

B. *Problem 2: Complex (bridge) System*

The complex (bridge) system [33,35] is shown as Figure 3:

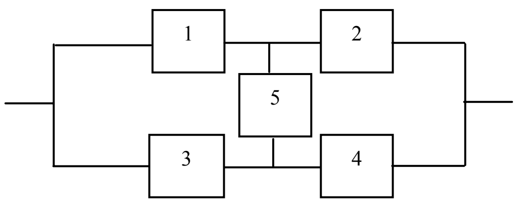

Figure 3. Complex (bridge) system

This problem is formulated as follows:

$$\begin{aligned}
\text{Max} \quad f(r,n) = {} & R_1R_2 + R_3R_4 + R_1R_4R_5 + R_2R_3R_5 \\
& - R_1R_2R_3R_4 - R_1R_2R_3R_5 - R_1R_2R_4R_5 \qquad (9) \\
& - R_1R_3R_4R_5 - R_2R_3R_4R_5 + 2R_1R_2R_3R_4R_5
\end{aligned}$$

The constraints are the same as series system. The parameters for this problem are listed in Table Ⅱ:

TABLE II.

THE PARAMETERS OF COMPLEX (BRIDGE) SYSTEM.

Subsystem i	$10^5\alpha_i$	β_i	$w_iv_i^2$	w_i	V	C	W
1	2.33	1.5	1	7	110	175	200
2	1.450	1.5	2	8			
3	0.541	1.5	3	8			
4	8.050	1.5	4	6			
5	1.950	1.5	2	9			

C. *Problem 3: Overspeed Protection System*

The problem is used to overspeed protection of a gas turbine. When the overspeed occurs, the system will be cut off. The overspeed protection system [36] is shown as Figure 4:

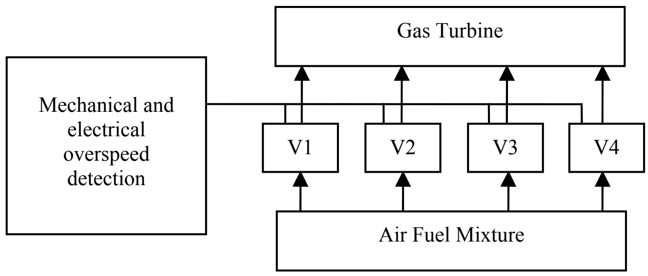

Figure 4. The overspeed protection system of a gas turbine

The control system can be viewed as an N-stage (N=4) mixed series-parallel systems. The model is formulated as follows:

$$\begin{aligned}
&\text{Max} \qquad f(r,n) = \prod_{i=1}^{m} [1-(1-r_i)^{n_i}] \\
&\text{s.t.} \qquad h_1(r,n) = \sum_{i=1}^{m} v_i n_i^2 \le V \\
&\qquad h_2(r,n) = \sum_{i=1}^{m} C(r_i)\cdot[n_i + \exp(n_i/4)] \le C \qquad (10) \\
&\qquad h_3(r,n) = \sum_{i=1}^{m} w_i n_i \exp(n_i/4) \le W \\
&\qquad 1.0 \le n_i \le 10, n_i \in Z^+ \\
&\qquad 0.5 \le r_i \le 1-10^{-6}, r_i \in R^+
\end{aligned}$$

Here $C(r_i) = \alpha_i(-T/\ln r_i)^{\beta_i}$, T is the task time of the components, the parameters α_i and β_i is the same as series system.

The parameters for this problem are listed in Table III

TABLE III.

THE PARAMETERS OF OVERSPEED PROTECTION SYSTEM.

Subsystem i	10^5 α_i	β_i	v_i	w_i	V	C	W	T
1	1	1.5	1	6	250	400	500	1000
2	2.3	1.5	2	6				
3	0.3	1.5	3	8				
4	2.3	1.5	2	7				

To analyze the performance of the TSDE, the TS and DE are developed as well for comparison. For the TS algorithm, the maximum number of iterations is set to 1500, and the length of Tabu list is set to 24. For the DE and TSDE algorithms, set F_0=0.1, CR_0=1.0, η=1, population size M=40, and the maximum number of iterations is set to 1500. Every algorithm runs 50 times independly for each problem, and the statistical results are listed in Table IV,Table V, and Table VI, including the best results(Best), the worst results(Worst), the mean results (Mean)and standard deviation(SD).

TABLE IV.
RESULTS OF THE SERIES PARALLEL SYSTEM USING THREE ALGORITHMS

Algorithm	Best	Worst	Mean	SD
TS	0.9999725346	0.9928314610	0.9991606706	1.3553e-03
DE	0.9998984359	0.9835803872	0.9975356371	2.8176e-03
TSDE	0.9999766491	0.9999647634	0.9999762814	1.8994e-06

TABLE V.
RESULTS OF THE COMPLEX (BRIDGE) SYSTEM USING THREE ALGORITHMS

Algorithm	Best	Worst	Mean	SD
TS	0.9998281844	0.9668848787	0.9949616616	7.3952e-03
DE	0.9995739226	0.9617663790	0.9912332520	9.6013e-03
TSDE	0.9998896376	0.9998893505	0.9998894366	1.3290e-07

TABLE VI.
RESULTS OF THE OVERSPEED PROTECTION SYSTEM USING THREE ALGORITHMS

Algorithm	Best	Worst	Mean	SD
TS	0.9999192216	0.9136394661	0.9841950104	2.2715e-02
DE	0.9988103068	0.9523992418	0.9876744210	1.1077e-02
TSDE	0.9999546747	0.9999461512	0.9999545042	1.2504e-06

TABLE VII.
BEST RESULTS COMPARISON ON SERIES PARALLEL SYSTEM

Parameter	Hikita et al. [34]	Hsieh et al. [14]	Chen [16]	This paper
n_1-n_5	(3,3,1,2,3)	(2,2,2,2,4)	(2,2,2,2,4)	(2,2,2,2,4)
r_1	0.838193	0.785452	0.812485	0.819659
r_2	0.855065	0.842998	0.843155	0.844981
r_3	0.878859	0.885333	0.897385	0.895507
r_4	0.911402	0.917958	0.894516	0.895506
r_5	0.850355	0.870318	0.870590	0.868448
f(r,n)	0.99996875	0.99997418	0.99997658	**0.9999766491**
MPI (%)	25.2771	9.5627	0.2950	-
Slack(g1)	53	40	40	40
Slack(g2)	0.000000	1.194440	0.002627	0.000000
Slack(g3)	7.110849	1.609289	1.609829	1.609289

Note: (1) the bold values denote the best values of those obtained by all the algorithms.
(2)Slack is the unused resources.

TABLE VIII.
BEST RESULTS COMPARISON ON COMPLEX (BRIDGE) SYSTEM

Parameter	Hikita. et al. [34]	Hsieh et al. [14]	Chen [16]	Coelho[22]	This paper
n_1-n_5	(3,3,2,3,2)	(3,3,3,3,1)	(3,3,3,3,1)	(3,3,2,4,1)	(3,3,2,4,1)
r_1	0.814483	0.814090	0.812485	0.826678	0.828086
r_2	0.821383	0.864614	0.867661	0.857172	0.857805
r_3	0.896151	0.890291	0.861221	0.914629	0.914241
r_4	0.713091	0.701190	0.713852	0.648918	0.648146
r_5	0.814091	0.734731	0.756699	0.715290	0.704162
f(r,n)	0.9997894	0.99987916	0.99988921	0.99988957	**0.9998896376**
MPI (%)	47.5962	8.6706	0.3860	0.0612	-
Slack(g1)	18	18	18	5	5
Slack(g2)	1.854075	0.376347	0.001494	0.000339	0.000000
Slack(g3)	4.264770	4.264770	4.264770	1.560466	1.560466

Note: (1) the bold values denote the best values of those obtained by all the algorithms.
(2) Slack is the unused resources.

TABLE IX.
BEST RESULTS COMPARISON ON OVERSPEED PROTECTION SYSTEM

Parameter	Yokota et al. [35]	Dhingra[36]	Chen[16]	Coelho [22]	This paper
n_1-n_4	(3,6,3,5)	(6,6,3,5)	(5,5,5,5)	(5,6,4,5)	(5,6,4,5)
r_1	0.965993	0.81604	0.903800	0.902231	0.901615
r_2	0.760592	0.80309	0.874992	0.856325	0.849921
r_3	0.972646	0.98364	0.919898	0.9481450	0.948141
r_4	0.804660	0.80373	0.890609	0.883156	0.888223
f(r,n)	0.999468	0.99961	0.999942	0.999953	**0.9999546747**
MPI (%)	91.4802	88.3781	21.8529	3.5632	-
Slack(g1)	92	65	50	55	55
Slack(g2)	70.733576	0.064	0.002152	0.975465	0.000000
Slack(g3)	127.583189	4.348	28.803701	24.801882	24.801882

Note: (1) the bold values denote the best values of those obtained by all the algorithms.
(2)Slack is the unused resources.

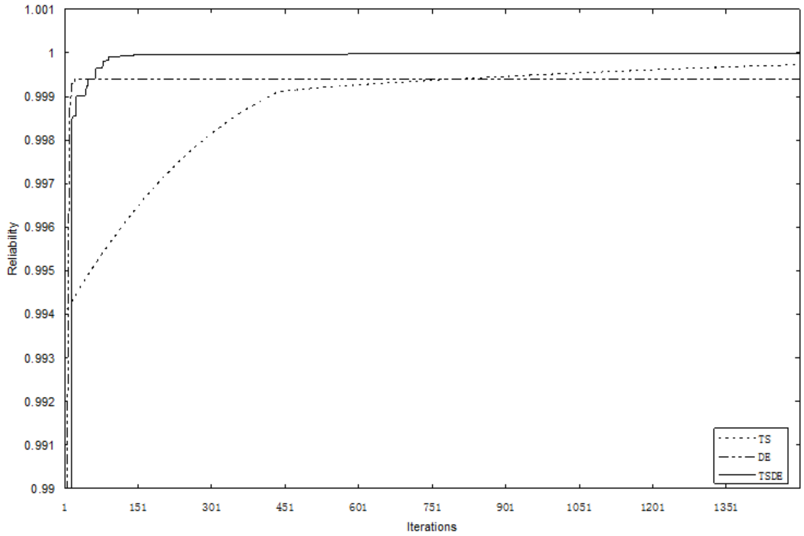

Figure 5. The result obtained by three algorithms for series-parallel system

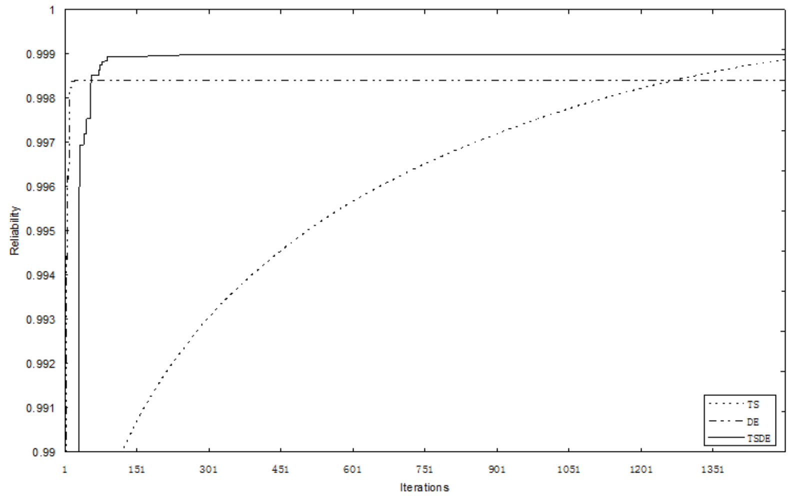

Figure 6. The result obtained by three algorithms for complex (bridge) system

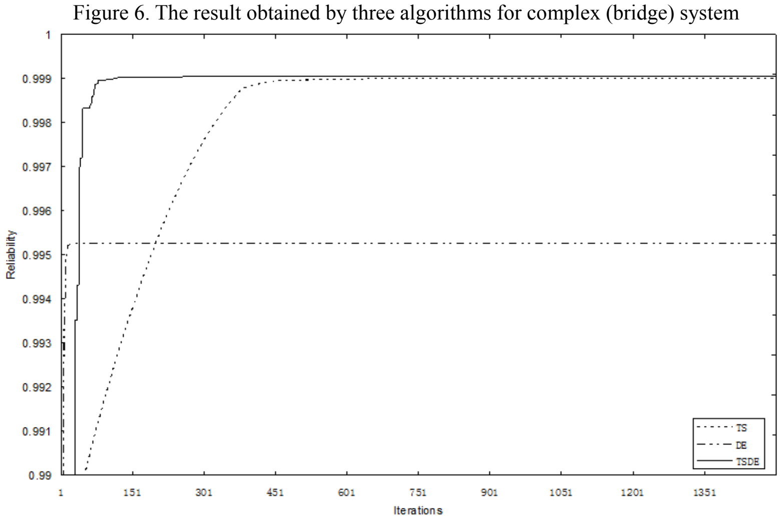

Figure 7. The result obtained by three algorithms for overspeed protection system

Table VII, Table VIII and Table IX compare the best result obtained in this paper with those of other methods in the literature.

It can be clearly seen from Table IV, Table V and Table VI that the best, worst, mean results obtained by TSDE are superior to those obtained by TS and DE for three benchmark problems. The best values obtained by TSDE are 0.9999766491, 0.9998896376 and 0.9999546747 respectively. And the standard deviations (SD) are 1.8994e-06, 1.3290e-07 and 1.2504e-06 respectively. These results have shown that the TSDE has strong ability of get the best result and stability than TS and DE (As Figure 5, Figure 6 and Figure 7). It is worth mentioning that the adaptive parameters F and CR make TSDE have better capacity of solution space exploration.

Table VII, Table VIII and Table IX compare the best results obtained by TSDE for three reliability optimization problems with those reported in the literature. It can be seen that the proposed algorithm can get a better solution than any other methods presented in literature. MPI (maximum possible improvement) is used to measure the amount improvement of the solutions obtained by the presented method to the best solutions found by other best known methods, and it is described as: MPI (%) = $(f - f_{other})/(1 - f_{other})$, where f represents the best value obtained by the proposed algorithm, and f_{other} represents the best value obtained by one of the other methods in literature. It should be emphasized that even very small improvements in reliability are critical and beneficial in high reliability applications.

It can be seen from Table Ⅶ, that the best results reported by Hikita et al. [34], Hsieh, et al. [14] and Chen[16] were 0.99996875, 0.99997418 and 0.99997658 for the series–parallel system respectively. The result obtained by TSDE is better than the above three best solution, and the corresponding improvements made by the presented method are 25.2771%, 9.5627% and 0.2950% respectively.

It can be seen from Table Ⅷ, that the best results reported by Hikita et al. [34], Hsieh et al. [14], Chen [16] and Coelho [22] were 0.9997894, 0.99987916, 0.99988921 and 0.99988957 for the complex (bridge) system respectively. The result obtained by TSDE is better than the above four best solution, and the corresponding improvements made by the presented method are 47.5962%, 8.6706%, 0.3860% and 0.0612% respectively.

It can be seen from Table Ⅸ, that the best results reported by Yokota, et al. [35], Dhingra[36], Chen [16] and Coelho[22] were 0.999468, 0.99961, 0.999942 and0.999953 for the overspeed protection system respectively. The result is better than the above four best solution, and the corresponding improvements made by the presented method are 91.4802%, 88.3781%, 21.8529% and 3.5632% respectively.

In short, the proposed TSDE is an effective algorithm, and it outperforms the other methods in literature for reliability optimization problems.

V. Conclusion

In this paper, we proposed a hybrid TS-DE algorithm to solve the reliability redundancy optimization problems. The proposed approach benefits from advantages of both Tabu search algorithm and differential evolution algorithm. The adaptive parameters F and CR in DE make the algorithm have higher exploration capability of solution space. Simulation experiments based on three benchmark problems and compared with some algorithms in the literature. The results showed that the TS-DE algorithm was effective, efficient and performed better than the other methods in the literature. The future work is to apply it to solve other more complex mixed-integer programming problems.

Acknowledgment

The authors wish to thank Department of education of Jilin Province, China. This work was supported by the science research project of Department of education of Jilin Province, China (No. 2013461).

References

[1] K.B. Misra, and J. Sharma, “A new geometric programming formulation for a reliability system, ” International Journal of Control, vol. 18, pp. 497–503, 1973.

[2] M.S. Chern, “On the computational complexity of reliability redundancy allocation in a series system,” Operations Research Letters, vol. 11, pp. 309–315, June 1992.

[3] W. Kuo, and V.R. Prasad, “An annotated overview of system-reliability optimization,” IEEE Transaction on Reliability, vol. 49 (2), pp. 176–187, June 2000.

[4] Peifeng Wu et al., “An improved particle swarm optimization algorithm for reliability problems,” ISA Transactions, vol. 50, pp. 71-81, Jan 2011.

[5] Y.-C. Hsieh, “A linear approximation for redundant reliability problems with multiple component choices,” Computers and Industrial Engineering, vol. 44, pp. 91–103, 2003.

[6] Y. Nakagawa, and S. Miyazaki, “Surrogate constraints algorithm for reliability optimization problems with two constraints,” IEEE Transaction on Reliability, vol. R 30, pp. 175–180, 1981.

[7] K.S. Park, “Fuzzy apportionment of system reliability,” IEEE Transactions on Reliability, vol. R 36, pp. 129–132, 1987.

[8] Z. Xu, W. Kuo, and H.H. Lin, “Optimization limits in improving system reliability,” IEEE Transactions on Reliability, vol. 39, pp. 51–60, 1990.

[9] V.R. Prasad, and W. Kuo, “Reliability optimization of coherent system,” IEEE Trans. Reliab, vol. 49, pp. 323–330, 2000.

[10] Lu Xiong, Shen-guang Gong, “Reliability Allocation of Underwater Experiment System Based on Particle Swarm Optimization,” Journal of Networks, vol. 8, pp. 1292–1299, June 2013.

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

[12] Kuo W, Hwang CL, and Tillman FA, “A note on heuristic methods in optimal system reliability,” IEEE Transactions on Reliability, vol. R-27(5), pp. 320–324, 1978.

[13] D.W. Coit, and A.E. Smith, “Reliability optimization of series–parallel systems using a genetic algorithm, ” IEEE Transactions on Reliability, vol. 45, pp. 254–260, 1996.

[14] Yi-Chih Hsieh, Ta-Cheng Chen, and Dennis L. Bricker, “Genetic algorithms for reliability design problems,” Microelectronics Reliability, vol. 38, pp. 1599-1605, 1998.

[15] P.S. You, and T.C.Chen, “An efficient heuristic for series–parallel redundant reliability problems,” Computers and Operations Research, vol. 32(8), pp. 2117–2127, 2005.

[16] Ta-Cheng Chen, “IAs based approach for reliability redundancy allocation problems,” Applied Mathematics

and Computation, vol. 182(2), pp. 1556-1567, 2006.
[17] Y.-C. Hsieh, and P.-S. You, “An effective immune based two-phase approach for the optimal reliability–redundancy allocation problem,” Applied Mathematics and Computation, vol. 218, pp. 1297–1307, 2011.
[18] Y.C. Liang, and Y.C. Chen, “Redundancy allocation of series–parallel systems using a variable neighborhood search algorithm,” Reliability Engineering and System Safety, vol. 92, pp. 323–331, 2007.
[19] Dexuan Zou, Liqun Gao, Jianhua Wu, Steven Li, and Yang Li, “A novel global harmony search algorithm for reliability problems,” Computers & Industrial Engineering, vol. 58, pp. 307–316, 2010.
[20] Dexuan Zou, Liqun Gao, Steven Li, and Jianhua Wu, “An effective global harmony search algorithm for reliability problems,” Expert Systems with Applications, vol. 38, pp. 4642–4648, 2011.
[21] A.E.M. Zavala, E.R.V. Diharce, and A.H. Aguirre, “Particle evolutionary swarm for design reliability optimization,” Lecture Notes in Computer Science 3410, Presented at the Third International Conference on Evolutionary Multi-Criterion Optimization, Guanajuato, Mexico, pp. 856–869, March 9–11, 2005.
[22] Leandro dos Santos Coelho, “An efficient particle swarm approach for mixed-integer programming in reliability–redundancy optimization applications,” Reliability Engineering and System Safety, vol. 94, pp. 830–837, 2009.
[23] Harish Garg, and S.P. Sharma, “Multi-objective reliability-redundancy allocation problem using particle swarm optimization,” Computers & Industrial Engineering, vol. 64, pp. 247–255, 2013.
[24] Manju Agarwal, and Vikas K. Sharma, “Ant colony approach to constrained redundancy optimization in binary systems,” Applied Mathematical Modelling, vol. 34, pp. 992–1003, 2010.
[25] Nabil Nahas, Mustapha Nourelfath, and Daoud Ait-Kadi, “Coupling ant colony and the degraded ceiling algorithm for the redundancy allocation problem of series–parallel systems,” Reliability Engineering and System Safety, vol. 92, pp. 211–222, 2007.
[26] Mohamed Ouzineb, Mustapha Nourelfath, and Michel Gendreau, “Tabu search for the redundancy allocation problem of homogenous series–parallel multi-state systems,” Reliability Engineering and System Safety, vol. 93, pp. 1257–1272, 2008.
[27] Leonardo Dallegrave Afonso, Viviana Cocco Mariani, and Leandro dos Santos Coelho, “Modified imperialist competitive algorithm based on attraction and repulsion concepts for reliability-redundancy optimization,” Expert Systems with Applications, vol. 40, pp. 3794–3802, 2013.
[28] Nima Safaei, Reza Tavakkoli-Moghaddamb, and Corey Kiassat, “Annealing-based particle swarm optimization to solve the redundant reliability problem with multiple component choices,” Applied Soft Computing, vol. 12, pp. 3462–3471, 2012.
[29] Ling Wang, and Ling-po Li, “A coevolutionary differential evolution with harmony search for reliability-redundancy optimization,” Expert Systems with Applications, vol. 39, pp. 5271–5278, 2012.
[30] R. Storn, and K. Price, Differential evolution: a simple and efficient adaptive scheme for global optimization over continuous spaces, Technical Report,TR-95-012, International Computer Science Institute, Berkeley, USA, 1995.
[31] Storn R., and Price K., “Differential evolution-A simple and efficient heuristic for global optimization over continuous spaces,” Journal of Global Optimization, vol. 11, pp. 341–359, 1997.
[32] Price K.V., Storn R.M., and Lampinen J.A., Differential evolution: A practical approach to global optimization (natural computing series) (1st ed.). New York: Springer, 2005.
[33] K. Gopal, K. Aggarwal, and J.S. Gupta, “An improved algorithm for reliability optimization,” IEEE Trans. Reliab, vol. 27, pp. 325–328, 1978.
[34] M. Hikita, Y. Nakagawa, and H. Harihisa, “Reliability optimization of systems by a surrogate constraints algorithm,” IEEE Trans. Reliab, vol. 41, pp. 473–480, 1992.
[35] T. Yokota, M. Gen, and Y.X. Li, “Genetic algorithm for non-linear mixed integer programming problems and its applications,” Comput. Ind. Eng, vol. 30, pp. 905–917, 1996.
[36] Anoop K. Dhingra, “Optimal Apportionment of Reliability & Redundancy in Series Systems Under Multiple Objectives,” IEEE Transactions on Reliability, vol. 41(4), pp. 576–582, 1992.
[37] Shi-Ming Chen, Ali Sarosh, Yun-Feng Dong, “Simulated annealing based artificial bee colony algorithm for global numerical optimization,” Applied Mathematics and Computation, vol. 219, pp. 3575–3589, 2012.

Yubao Liu received M. Sc. from Changchun University of Science and Technology in 2004. He is a doctoral student in Jilin University now and his research interests are mainly focused on system reliability, stochastic optimization, and artificial intelligence.

Guihe Qin received M. Sc. and Sc. D. degrees in Communication and Electronic Engineering from Jilin University, in 1988 and 1997, respectively. He is a professor in College of Computer Science and Technology, Jilin University. Currently, His research interests are embedded system, intelligent control, Real-time systems, and automotive electronics.