A Hybrid Metaheuristic for the Series-parallel Redundancy Allocation Problem in Electronic Systems of the Ship

Joo-Young Son[1] · Jae-Hwan Kim†

(Received April 25, 2011; Revised May 18, 2011; Accepted May 18, 2011)

Abstract : The redundancy allocation problem (RAP) is a famous NP-complete problem that has been studied in the system reliability area of ships and airplanes. Recently meta-heuristic techniques have been applied in this topic, for example, genetic algorithms, simulated annealing and tabu search. In particular, tabu search (TS) has emerged as an efficient algorithmic approach for the series-parallel RAP. However, the quality of solutions found by TS depends on the initial solution. As a robust and efficient methodology for the series-parallel RAP, the hybrid metaheuristic (TSA) that is a interactive procedure between the TS and SA (simulated annealing) is developed in this paper. In the proposed algorithm, SA is used to find the diversified promising solutions so that TS can re-intensify search for the solutions obtained by the SA. We test the proposed TSA by the existing problems and compare it with the SA and TS algorithm. Computational results show that the TSA algorithm finds the global optimal solutions for all cases and outperforms the existing TS and SA in cases of 42 and 56 subsystems.

Key words : RAP, Tabu Search, Simulated Annealing, Hybrid Metaheuristic

Acronym

RAP	redundancy allocation problem
TS	tabu search
SA	simulated annealing
GA	genetic algorithm
ACO	ant colony optimization
TSA	hybrid metaheuristic combined with TS and SA

1. Introduction

The reliability of a system can be increased by properly allocating redundancies to subsystems under various resource and technological constraints. A well-known Redundancy Allocation Problem is to determine the optimal number of redundant components in order to maximize the system reliability subject to multiple resource restrictions or system-level constraints for cost, weight, power, etc. Recently, the RAP has become useful for system designs such as electronic systems of the ship, semiconductor integrated circuits, nanotechnology, and etc. The RAP is typically classified into the five categories such as series, parallel, series-parallel, parallel-series, and complex systems. In this paper, we restrict ourselves to the series-parallel system.

Solutions for the series-parallel RAP have been suggested by several authors. Fyffe et al. [1] originally set up the problem and suggested a solution algorithm utilizing a dynamic programming approach. Nakagawa and Miyazaki [2] developed 33 variations of Fyffe's problem, where the weight

1 Division of IT Engineering Korea Maritime University E-mail:mmlab@hhu.ac.kr
† Corresponding author(Data Information, Korea Maritime University, E-mail:jhkim@hhu.co.kr)
Some preliminary results of this work were reported in the Proceedings of AIWARM 2008.

constraint varied its value from 159 to 191. Coit and Liu [3] proposed zero-one integer programming for this problem. They constrained the solution space so that only the identical component type can be allowed for each subsystem.

On the contrary, Coit and Smith [4] extended Fyffe's problem in such a way that the parallel system could be more flexible. They allowed a mixing of component types within a subsystem and employed a GA to obtain solutions. Kulturel-Konak et al. [5] developed a TS and showed that it may produce better solutions in most cases for the test problems they set up than the previous methods. Also, Kulturel-Konak et al. [6] improved their TS [5]. Liang and Smith [7] proposed an ACO algorithm for improving the GA of Coit and Smith [4]. The ACO, however, failed to improve most of solutions obtained by TS for this problem.

The quality of solutions found by TS depends on the initial solution. To overcome this problem and provide a robust and efficient methodology for the RAP, the hybrid metaheuristic combining SA and TS strategy is developed in this paper. The proposed TSA is tested on the benchmark problems and compared with the SA and TS algorithm. Computational results show that the TSA algorithm finds the global optimal solutions for all cases and outperforms the existing TS and SA in cases of 42 and 56 subsystems.

The rest of this paper is organized as follows; in section 2, we present the typical optimization problem of a series-parallel system. In section 3, we propose the hybrid metaheuristic TSA that is an interactive procedure between the TS and SA. In Section 4, computational results by a large set of benchmark problems are given. Finally, conclusions are discussed in section5.

2. Problem Formulation

Notation

$R(\mathbb{X})$ system reliability depending on $\mathbb{X}$

$\mathbb{X}$ $(x_{11}, x_{12}, .., x_{1m_i}, ..., x_{sm_1}, .., x_{sm_s})$

i index for a subsystem

j index for a component in a subsystem

x_{ij} quantity of the jth available component in the subsystem i

m_i number of available components for subsystem i

s number of subsystems

n_i total number of components used in subsystem i

C cost limit

W allowed weight

c_{ij} cost for the jth available component for the subsystem i

w_{ij} weight for the jth available component for the subsystem i

q_{ij} unreliability for the jth available component for the subsystem i

$n_{\max}$ maximum number of components that can be used in parallel

k_i minimum number of components in parallel required for subsystem i to function

The series-parallel RAP is a well-known reliability design problem. Its configuration is sketched in a series system of s independent k-out-of-n :G subsystems (see Figure 1). Subsystem i functions properly if at least k_i of its n_i components are operational. If k_i = 1 for all subsystems, then it is a series-parallel system. We set up the following case for the series-parallel RAP as formulated by Coit and Smith [4].

(P) Maximize

$$R(\mathbb{X}) = \prod_{i=1}^{s}(1 - \prod_{j=1}^{m_i} q_{ij}^{x_{ij}})$$

subject to

$$\sum_{i=1}^{s}\sum_{j=1}^{m_i} c_{ij}\, x_{ij} \le C$$

$$\sum_{i=1}^{s}\sum_{j=1}^{m_i} w_{ij}\, x_{ij} \le W$$

$$k_i \le \sum_{j=1}^{m_i} x_{ij} \le n_{\max}, \quad \forall\ i = 1, 2, \ldots, s$$

$$x_{ij} \in \{0, 1, 2, \ldots, n_{\max}\}$$

Chern [8] showed that (P) is NP-hard and various heuristic methods have been developed (Kuo et al.[9]). Recently, Kulturel-Konak et al. ([5], [6]) developed TS algorithms for (P) and showed that these algorithms produce better solutions in most of the test problems than the previous methods (Coit and Smith[4], Liang and Smith[7]).

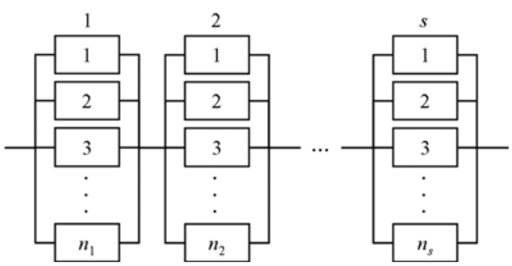

Figure 1: Series-parallel System Configuration

3. Hybrid metaheuristic : TSA

In this section, we propose a hybrid metaheuristic based on SA and tabu search, in which SA is used to find the promising elite solutions and TS intensifies search around these solutions. The TSA is a straightforward implementation of TS intensification strategy. A strong diversification strategy using SA procedure to find promising solutions is equipped with the TS and directs the intensified search to explore other regions of the solution space. The TSA algorithm is also related to the strategy that designs the more efficient forms of finite convergent TS based on recency memory. The general frame work of TSA algorithm for the series-parallel RAP is outlined as follows:

Step 1. (Initialization) Generate the random initial solution x_0 and calculate the system reliability $R(x_0)$. Set the current solution $x_c = x_0$ and best solution $x_b = x_0$. Initialize the TS memory.

Step 2. (TS phase)

a. Set iter=iter+1 and generate neighbors of the current solution x_c by the neighborhood structure.

b. Select the best neighbor which is not on tabu or satisfies the aspiration, store it as the new current solution x_c. Update the tabu list.

c. If $R(x_c) > R(x_b)$, then set $x_b = x_c$ and iter=0.

d. If iter <= Stopiter, then go to Step 2.a. Otherwise, go to Step 3.a .

Step 3. (SA phase)

a. Set $x_0 = x_b$ and iter=0. Initialize the temperature T.

b. Set iter=iter+1 and generate the neighborhood solution x_h.

c. If $e^{(R(x_b) - R(x_h))} / T$ > random[0, 1], then set $x_c = x_h$. Otherwise, go to Step 3.b.

d. If $R(x_c) > R(x_b)$ then set $x_b = x_c$.

e. If iter <= Stopiter, then go to Step 3.b. If a temperature criterion is satisfied then go to Step 4.

Otherwise, update the temperature T, iter=0, and go to Step 3.b.

Step 4. (Stop)

If the termination criterion is satisfied then stop with the best solution found so far.

Otherwise, set $x_c = x_b$ and initialize the TS memory, and go to Step 2.a.

Table 1: Component data for the test problems.

Subsystem		Component choices											
		1			2			3			4		
i	k_i	r_{i1}	c_{i1}	w_{i1}	r_{i2}	c_{i2}	w_{i2}	r_{i3}	c_{i3}	w_{i3}	r_{i4}	c_{i4}	W_{i4}
1	1	0.90	1	3	0.93	1	4	0.91	2	2	0.95	2	5
2	2	0.95	2	8	0.94	1	10	0.93	1	9	—	—	—
3	1	0.85	2	7	0.90	3	5	0.87	1	6	0.92	4	4
4	2	0.83	3	5	0.87	4	6	0.85	5	4	—	—	—
5	1	0.94	2	4	0.93	2	3	0.95	3	5	—	—	—
6	2	0.99	3	5	0.98	3	4	0.97	2	5	0.96	2	4
7	1	0.91	4	7	0.92	4	8	0.94	5	9	—	—	—
8	2	0.81	3	4	0.90	5	7	0.91	6	6	—	—	—
9	3	0.97	2	8	0.99	3	9	0.96	4	7	0.91	3	8
10	3	0.83	4	6	0.85	4	5	0.90	5	6	—	—	—
11	3	0.94	3	5	0.95	4	6	0.96	5	6	—	—	—
12	1	0.79	2	4	0.82	3	5	0.85	4	6	0.90	5	7
13	2	0.98	2	5	0.99	3	5	0.97	2	6	—	—	—
14	3	0.90	4	6	0.92	4	7	0.95	5	6	0.99	6	9

The above TSA algorithm is devised by an interactive procedure between the TS and SA to exploit their advantages, namely, the intensification strategy of TS and diversification strategy of SA. If the interactive procedure has failed to improve the best solution x_b to the predetermined number of iteration (the termination criterion of Step 4), then all procedure of TSA is terminated with the best solution found so far. In our work, the termination criterion was defined as 10 iterations without updating in the best solution, and the strategy for generating neighborhood solutions of Kulturel-Konak *et al*. [6] and Cerny [10] was adopted in TS phase (Step 2.a) and SA phase (Step 3.b), respectively. Also the same penalty function proposed by Kulturel-Konak *et al*. [9] was exploited to allow search in the promising infeasible region.

4. Computational experience

In this section, we describe some numerical experiments that we conducted with the problem originally proposed by Fyffe et al. [1] to evaluate the performances of metaheuristics. The system of the problem had 14 subsystems in series and each had three or four component alternatives. Nakagawa nd Miyazaki [2] devised 33 variations of the original problem. They fixed the cost constraint C at 130 and the weight constraint varied from 159 to 191.

We applied the TS, ACO, SA and the presented TSA algorithm to these problems and compared the experimental results with the previous studies. These algorithms were coded in Microsoft Visual C++ 6.0. All of the numerical experiments were executed on an IBM PC compatible with a Pentium IV 3.0 GHz. Table 1 shows the component data of the problem. The size of the search space was larger than 4.3×1037 on considering the component mixing. Until now, because of the complexity of the problems, metaheuristics were usually employed to solve them. Some of the best solutions to these problems were proposed without referring to their global optimal solutions. Kim and Kim [11] proved that the solutions suggested by Kultruel-Konak et al. [6] for the above 33 cases were all global solutions. Table 2 summarizes the experimental results of the TSA, TS and SA algorithms including previous results of GA, and ACO. We noticed that

Table 2: The experimental results for the 33 test problems.

No.	*W*	CPLEX (The optimal solution)	Metaheuristic methods				
			GA	ACO	TS	SA	TSA
1	191	**0.986811**	0.98675	**0.9868**	**0.986811**	**0.986811**	**0.986811**
2	190	**0.986416**	0.98603	0.9859	**0.986416**	**0.986416**	**0.986416**
3	189	**0.985922**	0.98556	0.9858	**0.985922**	**0.985922**	**0.985922**
4	188	**0.985378**	0.98503	0.9853	**0.985378**	**0.985378**	**0.985378**
5	187	**0.984688**	0.98429	**0.9847**	**0.984688**	**0.984688**	**0.984688**
6	186	**0.984176**	0.98362	0.9838	**0.984176**	**0.984176**	**0.984176**
7	185	**0.983505**	0.98311	**0.9835**	**0.983505**	**0.983505**	**0.983505**
8	184	**0.982994**	0.98239	**0.9830**	**0.982994**	**0.982994**	**0.982994**
9	183	**0.982256**	0.98190	**0.9822**	**0.982256**	**0.982256**	**0.982256**
10	182	**0.981518**	0.98102	**0.9815**	**0.981518**	**0.981518**	**0.981518**
11	181	**0.981027**	0.98006	0.9807	**0.981027**	**0.981027**	**0.981027**
12	180	**0.980290**	0.97942	**0.9803**	**0.980290**	**0.980290**	**0.980290**
13	179	**0.979505**	0.97906	**0.9795**	**0.979505**	**0.979505**	**0.979505**
14	178	**0.978400**	0.97810	**0.9784**	**0.978400**	**0.978400**	**0.978400**
15	177	**0.977596**	0.97715	**0.9776**	**0.977596**	**0.977596**	**0.977596**
16	176	**0.976690**	0.97642	0.9765	**0.976690**	**0.976690**	**0.976690**
17	175	**0.975708**	0.97552	**0.9757**	**0.975708**	**0.975708**	**0.975708**
18	174	**0.974926**	0.97435	**0.9749**	**0.974926**	**0.974926**	**0.974926**
19	173	**0.973827**	0.97362	**0.9738**	**0.973827**	**0.973827**	**0.973827**
20	172	**0.973027**	0.97266	**0.9730**	**0.973027**	**0.973027**	**0.973027**
21	171	**0.971930**	0.97186	**0.9719**	**0.971930**	**0.971930**	**0.971930**
22	170	**0.970760**	**0.97076**	**0.9708**	**0.970760**	**0.970760**	**0.970760**
23	169	**0.969291**	0.96922	**0.9693**	**0.969291**	**0.969291**	**0.969291**
24	168	**0.968125**	**0.96813**	**0.9681**	**0.968125**	**0.968125**	**0.968125**
25	167	**0.966335**	**0.96634**	**0.9663**	**0.966335**	**0.966335**	**0.966335**
26	166	**0.965042**	**0.96504**	**0.9650**	**0.965042**	**0.965042**	**0.965042**
27	165	**0.963712**	**0.96371**	**0.9637**	**0.963712**	**0.963712**	**0.963712**
28	164	**0.962422**	**0.96242**	**0.9624**	**0.962422**	**0.962422**	**0.962422**
29	163	**0.960642**	**0.96064**	**0.9606**	**0.960642**	**0.960642**	**0.960642**
30	162	**0.959188**	0.95912	**0.9592**	**0.959188**	**0.959188**	**0.959188**
31	161	**0.958035**	**0.95803**	**0.9580**	**0.958035**	**0.958035**	**0.958035**
32	160	**0.955714**	0.95567	**0.9557**	**0.955714**	**0.955714**	**0.955714**
33	159	**0.954565**	0.95432	**0.9546**	**0.954565**	**0.954565**	**0.954565**

* The optimal solutions are delineated with bold characters.
* GA (Coit and Smith, 1996), ACO (Liang and Smith, 2004), TS (Kulturel-Konak et al., 2004)

the TSA, TS and SA algorithms obtained global optimal solutions for all 33 problems and outperformed GA, and ACO in most cases.

To additionally compare the performances between the TSA, TS and SA algorithms for three larger problems, we replicated the data in Table 1 of the original problem by h-fold where h ranges from 2, 3 to 4, that is, the number of subsystems of each problem became 14 × h (h = 2, 3, 4). For instance, when h = 2, the number of subsystems was 28 (= 14 × 2) in which each of the original subsystems had its own twin with an identical component specification. For a large set of the new benchmark problems, we can also obtain global solutions by adopting the zero-one integer programming formulation proposed by Kim and Kim [11]. The computational results of the TSA, TS, SA algorithms and global solutions using CPLEX 9.1 (bold-face) are given in Table 3 when the original problem has the value of the weight constraint (W = 191).

Table 3: The results for the larger test problems

n	CPLEX	TS	SA	TSA
14	**0.986811**	**0.986811**	**0.986811**	**0.986811**
28	**0.974072**	**0.974072**	**0.974072**	**0.974072**
42	**0.961237**	0.961129	0.961124	**0.961237**
56	**0.948816**	0.948578	0.948663	**0.948816**

As the results in Table 3 indicate, the TS and SA algorithms found the optimal solution up to 28 subsystems, but global optimal solutions could not be obtained in the cases of systems with 42 and 56 subsystems. We notice that the TSA algorithm finds the global optimal solutions for all cases, and substantially outperforms the existing TS and SA in cases of 42 and 56 subsystems.

5. Conclusions

The metaheuristic TS [9] has emerged as an effective algorithmic approach for the series-parallel RAP. However, the quality of solutions found by TS depends on the initial solution. To overcome this problem and provide a robust and efficient methodology for the series-parallel RAP, the hybrid TSA (an interactive procedure between the TS and SA) has been proposed in this paper. The main principle of our method is that SA is used to find the diversified elite solutions so that TS can re-intensify search for the promising solutions obtained by the SA. The proposed method was tested on the benchmark problems and compared with the SA and TS algorithm. The computational results showed that the hybrid TSA substantially outperformed the TS and SA in the cases of 42 and 56 subsystems.

References

[1] D. E. Fyffe, W. W. Hines and N. K. Lee, “System reliability allocation and a computational algorithm”, IEEE Transactions on Reliability, vol. 17, pp. 64–69, 1968.

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

[3] D. E. Coit and J. Liu, “System reliability optimization with *k*-out-of-*n* subsystems”, International Journal of Reliability, Quality and Safety Engineering, vol. 7, pp. 129–142, 2000.

[4] D. E. 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.

[5] S. Kulturel-Konak, A. E. Smith and D. W. Coit, “Efficiently solving the redundancy allocation problem using tabu search”, IIE Transactions, vol. 35, pp. 515–526, 2003.

[6] S. Kulturel-Konak, B. A. Norman, D. W. Coit and A. E. Smith, “Exploiting tabu search memory in constrained problems”, INFORMS Journal on Computing, vol. 16, pp. 241–254, 2004.

[7] Y. C. Liang and A. E. Smith, “An ant colony optimization algorithm for the reliability allocation problem”, IEEE Transactions on Reliability, vol. 53, pp. 417–423, 2004.

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

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

[10] V. Cerny, “Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm”, Journal of Optimization Theory and Applications vol. 45, pp. 41-51, 1985.

[11] J. H. Kim and J. S. Kim,"Globally solving the redundancy allocation problem for the case of series-parallel systems", Proceeding of the 2nd Asian International Workshop on Advanced Reliability Modeling, pp. 150-157, 2006.

Author Profile

Joo-Young Son

He is Professor of Division of IT Engineering at Korea Maritime University. He received his BS in Computer Science & Statistics from SNU, and his ME and Ph.D in Computer Engineering from SNU. He was Senior Researcher at LG Electronics, Inc. His recent research interests are Protocols for High Speed Maritime Data Networks, e-Navigation, MANET, VANET, WMN, WiMAX MMR.

Jae-Hwan Kim

He is Professor and Chair of Data Information at Korea Maritime University. He received his BS in Industrial Engineering from Korea University, and his MS & PhD in Industrial Engineering from Korea Advanced Institute of Science and Technology (KAIST). His research centers on reliability optimization, application and development of metaheuristic algorithms.