

# Local Search Optimization for a Multi-state Series-parallel System

Ming-Ang YIN<sup>1,a</sup>, Zhi-Li SUN<sup>1</sup>, Jian WANG<sup>1</sup> and Yu GUO<sup>1</sup>

<sup>1</sup>*School of Mechanical Engineering and Automation, Northeastern University, Shenyang 110004, China*

**Abstract:** Generally, the redundancy allocation problem is characterized as a kind of NP-hard combinatorial optimization problem, which is difficult to solve with the classical method because it has exponentially increasing complexity with the augmented size. In this paper, we propose the efficient approach with two phases that is comprised of particle swarm optimization (PSO), which can efficiently represent the region search, and a local search, to solve the adjacent domains of best particle, which is great useful to improve optimization efficiency, and the parameter of local search is also studied.

Yun proposes a hybrid genetic algorithm with adaptive local search scheme[1]. Otsuki presents a local search algorithm which utilizes pruning and deepening techniques in the variable depth search framework[2]. Dengiz presents a genetic algorithm with specialized encoding, initialization and local search genetic operators to optimize communication network topologies[3]. Caniou chooses as the basis for these experiments the Adaptive Search method, an efficient sequential Local Search method for constraint satisfaction problems[4]. Gupta develops and compares different local search heuristics for the two-stage flow shop problem with makespan minimization as the primary criterion and the minimization of either the total flow time, total weighted flow time, or total weighted tardiness as the secondary criterion[5]. Cura presents a local search approach to the uncapacitated warehouse location problem that is quite simple and robust and is efficient in some cases[6].

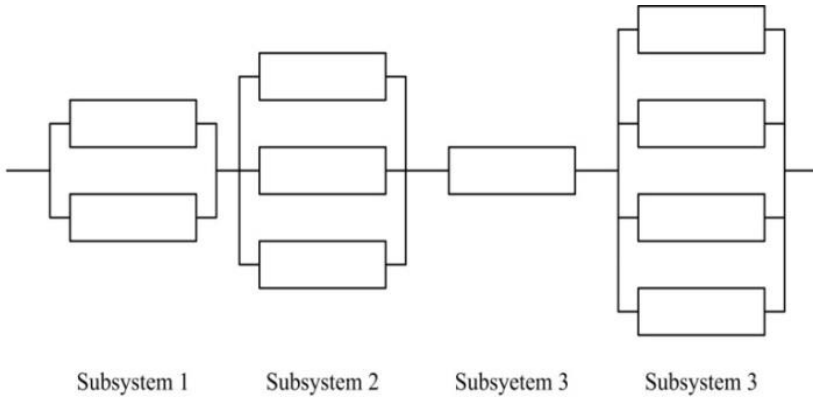
Based on installation and maintenance cost are seen as the objective function, while availability of the system is seen as the constraint function, This paper mainly focuses on PSO combined with LS algorithm is applied to optimize such a problem, and a novel local search method is put forward.

## 1 Description

In a typical multi-state series-parallel system,  $N$  subsystems are put together in series, each of them is consisted of  $M_i$  ( $i \in \{1, 2, \dots, N\}$ ) same devices in parallel. Configuration of one production line is shown in figure 1. With regard to the  $i$ th subsystem, there are  $V_i$  different versions available in market. To make things simplified, the  $i$ th subsystem in which devices with type of  $j$  (where  $j=1, 2, \dots, V_i$ ) is applied as an example below.  $CIN_{ij}$  represents installation charge and there are  $K_{ij}$  states in total for the devices from brand new state to failure. State Transition Probability Matrix of this device is  $\mathbf{P}_{ij} \in RK_{ij} \times K_{ij}$ .  $X_{ijl}(t)$  represents the state of  $l$ th (where  $l=1, 2, \dots, M_i$ ) device at  $t$  moment.  $X_{ijl}(t)=1$

<sup>a</sup> Corresponding author : yinma@me.neu.edu.cn

represents that the  $l$ th device is brand new at  $t$  moment and  $X_{ijl}(t)=K_{ij}$  represents that this device is in failure. Productivity of the devices will be  $\gamma_{ij}(w)$  if  $X_{ijl}(t)=w$ , while production efficiency of invalid devices will be zero, that is  $\gamma_{ij}(K_{ij})=0$ . Production efficiency of subsystem is the sum of that of all its components, while production efficiency of the whole system is the minimum of that of all its subsystems.



**Figure 1.** Schematic diagram of typical series-parallel system

## 2 Objective function

Formula (1) model of redundancy allocation problem (RAP) whose goal is to get the least acquisition and maintenance cost and meanwhile availability  $A_s$  of the system during the whole running time is higher than required value  $A_0$ .

$$\begin{cases} \text{Goal: } \min C_S = C_{IN} + C_M \\ \text{Constraint: } A_s \geq A_0 \end{cases}$$

Maintenance scheme will be confirmed according to detection results after detection, and furthermore, according to the second hypothesis that maintenance is instantaneous, so maintenance of system occurs at detection point, mean detection and maintenance cost of subsystem in unit time can be calculated through renewal theory[7].

$$C_{M,ij}(\theta_{ij}) = \sum_{i=1}^N (C_{INS,ij} + E(C_{ST,ij}(\theta_{ij})) + E(C_{CM,ij}(\theta_{ij})) + E(C_{PM,ij}(\theta_{ij}))) / E(T_{SR,ij}(\theta_{ij})) \tag{2}$$

Acquisition expenses  $C_{IN}$  of system can be expressed as

$$C_{IN} = \sum_{i=1}^N M_{ij} \times C_{IN,ij}$$

The system efficiency threshold value is  $W_R$ , so the system availability can be expressed as

$$A = \sum_{\{v|W_v \geq W_R\}} A_v(\theta)$$

One important problem in swarm optimization analysis is how to search in feasible region, penalty function can be used to solve this problem effectively. So, objective function is defined as the sum of objective function and penalty function. Calculation method of penalty function is put forward below.

$$C_P = \begin{cases} 0 & \text{,If } A_S \geq A_0 \\ C_\gamma \tan^{1.5} [-0.5\pi (1 + A_S / A_0)], & \text{else} \end{cases}$$

Wherein  $C_\gamma$  is the penalty coefficient, objective function can be changed to

$$C_S = C_{IN} + C_M + C_P$$

### 3 Coding of RAP problem

According to the description of above model, the whole series-parallel system can be coded  $[X, Y]$ ,  $X$  is the position vector and  $Y$  is the velocity vector. With regard to position vector  $X=[X_1, X_2, \dots, X_N]$ , wherein,  $X_i$  represents the feature of the subsystem,  $M_i$  the number of devices of the  $i$ th subsystem,  $j_i$  the type selected in the subsystem and the serial number that the detection and maintenance strategy selected. Too much devices will add the acquisition expenses of the system and it's impossible to achieve the goal of low cost, so there is need to set an upper limit  $XB$  to restrain the number of devices in each subsystem.

### 4 Local search strategy

To compensate for the effectiveness that discretization works on particle's position and velocity vector, local search method is applied on individual optimal solution  $XPB$  and global optimal solution  $XGB$ . To reduce the calculation cost, local search is used on particle from  $XPB$  that take up very little proportion  $\eta$  in every  $n$  iteration, while for  $XGB$ , if there's a better one, local search will be applied on it.

Main idea of local search is that a solution with lower cost and higher availability. Research object of local search is characteristic of the subsystem which includes number and model of devices in the subsystem and maintenance strategy. Number and version of devices in the subsystem should be enumerated firstly, maintenance strategy of the subsystem will be worked out by restricting maintenance cost, of which the upper limit is the minimum cost of the subsystem before this local search, value of  $A_i$  will be ranked from high to low and the first  $N_s$  particles will be selected. Characteristics of other subsystems stay unchangeable when the subsystem undergoes local search. The particle with lowest cost is the optimal one to be output after all subsystems have been searched.

### 5 The detail of LS algorithm steps

Table 1 shows a part of one step LS, where  $N_L=3$ , other condition is not changed. The initial particle is [2 5 34 4 5 49 2 4 19 4 4 19]. Every sub step is local-searches one subsystem. Under the 1 version 1 device, finding the maintenance strategy which cost is less than the original one. Because of cost or the availability, it can be hardly to find the 4-verion-six-devices situation. The enumeration of version, number and maintenance strategy is calculated before optimization, so the ls needs just call objective function for a few times. At one ls, objective function is called  $N \times M_i \times V_i \times N_L = 4 \times 6 \times 4 \times 3 = 288$  times at most.

**Table 1.** Details of Local Search

information	particle
Original particle	2 5 34 4 5 49 2 4 19 4 4 19
LS maintenance strategy for subsystem 1 , with 1 version 1 device	1 1 1 4 5 49 2 4 19 4 4 19 1 1 2 4 5 49 2 4 19 4 4 19 1 1 3 4 5 49 2 4 19 4 4 19

LS maintenance strategy for 1 2 2 4 5 49 2 4 19 4 4 19  
 subsystem 1 , with 1 version 2 1 2 3 4 5 49 2 4 19 4 4 19  
 devices 1 2 6 4 5 49 2 4 19 4 4 19

LS maintenance strategy for 4 3 40 4 5 49 2 4 19 4 4 19  
 subsystem 1 , with 4 version 3 4 3 46 4 5 49 2 4 19 4 4 19  
 devices 4 3 17 4 5 49 2 4 19 4 4 19

## 6 Example

An auto parts factory plans to put a vehicle air condition valve plug production line into production which includes four main processes. Units in every process are uniform. There are four versions of units in every process. Working performance of units from each factory varies from each other. Performance degradation process of all units all follow discrete time Markov process. Discrete interval is 1 week. State transition probability matrix of units produced from these four factories in the first process can be expressed as

$$P_{11} = \begin{bmatrix} 0.6 & 0.2 & 0.15 & 0.04 & 0.01 \\ 0 & 0.6 & 0.3 & 0.08 & 0.02 \\ 0 & 0 & 0.75 & 0.2 & 0.05 \\ 0 & 0 & 0 & 0.75 & 0.25 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad P_{12} = \begin{bmatrix} 0.6 & 0.2 & 0.15 & 0.03 & 0.015 & 0.005 \\ 0 & 0.6 & 0.3 & 0.06 & 0.025 & 0.015 \\ 0 & 0 & 0.75 & 0.2 & 0.03 & 0.02 \\ 0 & 0 & 0 & 0.75 & 0.15 & 0.1 \\ 0 & 0 & 0 & 0 & 0.7 & 0.3 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_{13} = \begin{bmatrix} 0.7 & 0.15 & 0.1 & 0.04 & 0.01 \\ 0 & 0.65 & 0.25 & 0.08 & 0.02 \\ 0 & 0 & 0.6 & 0.25 & 0.15 \\ 0 & 0 & 0 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$P_{14} = \begin{bmatrix} 0.7 & 0.15 & 0.1 & 0.03 & 0.015 & 0.005 \\ 0 & 0.7 & 0.2 & 0.06 & 0.025 & 0.015 \\ 0 & 0 & 0.75 & 0.2 & 0.03 & 0.02 \\ 0 & 0 & 0 & 0.75 & 0.15 & 0.1 \\ 0 & 0 & 0 & 0 & 0.8 & 0.2 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

To reduce redundant description, state transition probability matrix of units in other processes are the same with that of the first process such as  $P_{21}=P_{11}$ , productivity and installation cost of each unit is listed in Table 2. Number of inspection threshold of units  $N_{INS,ij}$  in each process is 2, then maintenance strategy includes  $L_{PM,ij}$ ,  $L_{OM,ij}$  and two inspection thresholds  $L_{INS,d,ij}$ , where  $d=1,2$

**Table 2.** Productivity, installation and maintenance cost of different versions in each subsystem

<i>i</i>	<i>j</i>	$\gamma_{ij}(1)$	$\gamma_{ij}(2)$	$\gamma_{ij}(3)$	$\gamma_{ij}(4)$	$\gamma_{ij}(5)$	$\gamma_{ij}(6)$	$C_{IN,ij}$	$C_{PR,ij}$	$C_{CR,ij}$	$C_{H,ij}$	$C_{INS,ij}$
1	1	0.2	0.15	0.1	0.05	0	-	1.56	0.686	0.312	0.156	0.052
	2	0.4	0.32	0.24	0.16	0.08	0	1.86	1.085	0.62	0.155	0.031
	3	0.24	0.2	0.16	0.08	0	-	2.16	0.756	0.36	0.216	0.036
	4	0.8	0.7	0.6	0.4	0.2	0	3.9	1.869	1.068	0.134	0.13
2	1	0.08	0.06	0.04	0.02	0	-	1.548	0.539	0.31	0.155	0.052
	2	0.25	0.2	0.15	0.1	0.05	0	2.748	1.603	0.916	0.229	0.046
	3	0.15	0.125	0.1	0.05	0	-	2.901	1.015	0.484	0.29	0.048
	4	0.6	0.525	0.45	0.3	0.15	0	4.101	4.55	1.64	0.205	0.21
3	1	0.24	0.18	0.12	0.06	0	-	0.642	0.364	0.128	0.064	0.021
	2	0.45	0.36	0.27	0.18	0.09	0	1.152	0.672	0.384	0.096	0.019
	3	0.54	0.45	0.36	0.18	0	-	1.602	0.56	0.267	0.16	0.027
	4	1.6	1.4	1.2	0.8	0.4	0	5.4	1.288	0.737	0.092	0.092
4	1	0.1	0.075	0.05	0.025	0	-	2.049	0.714	0.41	0.205	0.068
	2	0.125	0.1	0.075	0.005	0.025	0	1.935	1.127	0.645	0.161	0.032
	3	0.09	0.075	0.06	0.03	0	-	2.091	0.735	0.349	0.209	0.035
	4	0.56	0.49	0.42	0.28	0.14	0	3.57	3.92	1.428	0.179	0.18

Proportion and productivity requirements of busy season, normal season and slack season in one year are listed in Table 3.

**Table 3.** Productivity requirements of the system

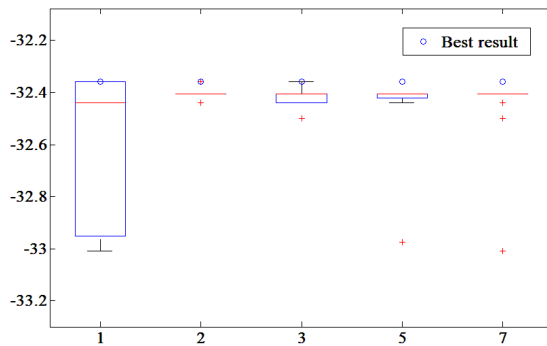
$W_{op}$	0.40	0.30	0.20
$T_p(\%)$	20	30	50

Some other parameters need in optimal algorithm are listed in Table 4.

**Table 4.** Parameters used in optimization algorithm

$X_B$	$t_{max}$	$N_P$	$C_\gamma$	$\eta$	$n$	$N_S$
6	50	500	2000	2%	10	5

Figure 2 shows that 20 optimization results, for  $W_R=15$ ,  $A_0=0.999$  and other parameters without changed. For comparison, the times of optimization result and the dispersion are listed. The best particle is [4 2 3 4 3 10 4 2 27 4 3 3], the system cost is 32.358, and system availability is 0.9993.



**Figure 2.** 20 times Optimization results with different  $N_L$

In figure 2, the blue cycle is the best result. The upper quartile is the best result, as  $N_L=1$ . Meanwhile for other condition, the best result only can be the upper boundary or singularity, even cannot be

optimized. The LS with  $N_L=1$  is generally superior to the other  $N_L$  since there are more trials out of 20 runs achieving the lowest cost, but the variation of the 20 solution is larger. It is an efficiency method to get the best solution to run more times optimization.

There is a LS strategy proposed in Literature [8], and the details are listed in figure 3. Five LS strategies are listed in figure 3. For improving the efficiency of algorithm, combination of five LS strategies is used.

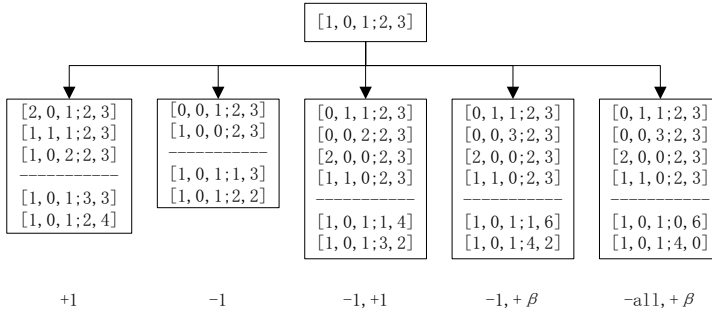


Figure 3. LS from reference [8]

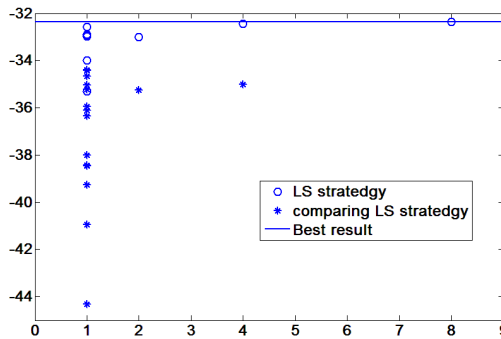


Figure 4. Comparison of two LS methods

Figure 4 shows comparison of two different ls strategies. The ls strategy with little cost more availability shows apparent solution improvement. It also lists times of achieving the best solution. In 20 runs, there 8 runs optimize the best result -32.358, and 4 runs optimize the second-best result, so it is show that the ls is efficiency and stability.

## Acknowledgement

This study was supported by the Fundamental Research Funds for the Central Universities of China (Grant No. N140303010).

## References

1. Y. S. Yun. Comput Ind Eng, **51**, 128-141 (2006)
2. T. Otsuki, K. Aihara. J Heuristics, 1-19 (2014)
3. B. Dengiz, F. Altiparmak. Ieee T Evolut Comput, **1**, 650-657 (1998)
4. Y. Caniou, P. Codognet, F. Richoux, et al. Constraints, **20**, 30-56 (2014)

5. J. N. D. Gupta, K. Hennig, F. Werner. *Comput Oper Res*, **29**, 123-149 (2002)
6. T. Cura. *Comput Ind Eng*, **59**, 1000-1009(2010)
7. Y. Zhou, Z. Zhang, T. R. Lin, et al. *Reliab Eng Syst Safe*, **111**, 248-259(2013)
8. Y. Wang, L. Li. *Ieee T Syst Man Cy A*, **42**, 464-474(2012)