Journal of Computer Science

Original Research Paper

Optimal Components Assignment Problem for Stochastic-Flow Networks

[1]Abdellah Aissou, [1]Abdelhamid Daamouche and [2]Moatamad Refaat Hassan

[1]*IGEE, Université de M'hamed Bougara Boumerdes, Algeria*
[2]*Computer Science Branch, Mathematics Department, Faculty of Science, Aswan University, Egypt*

Article history
Received: 24-07-2018
Revised: 13-12-2018
Accepted: 7-01-2019

Corresponding Author:
Abdellah Aissou
IGEE, Université de M'hamed Bougara Boumerdes, Algeria
Email: aaissou1969@gmail.com

Abstract: Components assignment problem to maximize the network reliability in the case of each component has both an assignment cost and lead-time is never discussed. Therefore, this paper focuses on solving this problem under the above mentioned constraints. The presented problem is called an Optimal Components Assignment Problem (OCAP) and it is formulated based on three constraints namely total assignment cost, total lead-time and system reliability. Also, an approach based on a Random Weighted Genetic Algorithm (RWGA) is presented to solve the OCAP. The results revealed that an optimal components assignment leads to the maximum reliability, minimum assignment cost and minimum total lead-time using the proposed approach.

Keywords: Stochastic-Flow Networks, Components Assignment Problem, System Reliability, Multi-Objective Genetic Algorithm

Introduction

Assignment Problem (AP) is one of the most important issues in manufacturing and service systems; it has also received significant attention among researchers. AP deals with the allocation of the various resources to the various activities on one to one basis. The generalized assignment problem is defined as the problem of finding the minimum cost assignment of n jobs to m agents such that each job is assigned to exactly one agent, subject to an agent's capacity Chu and Beasley (1997). They used a Genetic Algorithm (GA) based heuristic for solving such problems. As a special case of a generalized assignment problem, Wang (2002) has used a GA to solve the teacher assignments problem. The obtained results showed that the proposed technique can save a significant time spent on teacher assignments problem. Furthermore, Harper *et al*. (2005) used GA to solve the project assignment problem. Also, Lin and Yeh (2015) used GA to find the optimal two-class allocation of a computer network subject to a budget.

The CAP for a Stochastic-Flow Network (SFN) found by Lin and Yeh (2010), they proposed an efficient approach based on a GA to solve the resource assignment problem by finding the optimal resource assignment, which leads to maximal system reliability. The Component Assignment Problem (CAP) aims to find the optimal arrangement of n available components to m positions in a system to maximize the system reliability Lin and Yeh (2011a). In addition, Lin and Yeh (2011b) have solved the above problem as a multi-objective CAP. They proposed a two-stage approach to solving the multi-objective CAP subject to reliability and assignment cost for SFN. Furthermore, Chen (2014) addressed the optimal double resource assignment for the robust design problem, a minimum capacity assignment for each link and node is searched to keep the network working even both links and nodes are subject to failures. Lin and Yeh (2013) presented a new algorithm based on GA to find the optimal two class allocation subject to a budget leading to maximal system reliability of a computer network.

Hassan (2015) discussed each component possessing a lead-time and proposed a GA to search the set of components that maximize the system reliability such that the total lead-time cannot exceed a specified amount.

The system reliability evaluation in the case of solving CAP under cost constraint is based on flow assignment for the assigned components given the demand Lin and Yeh (2013; Lin *et al*., 1995; Lin, 2001; 2002).

In the case of considering each link has a lead-time and capacity, the transmission time of network paths can be evaluated and determined the quickest one (de Queiros *et al*., 1997; Chen and Hung, 1993; Chen and Chin, 1990). The system reliability of an SFN network is evaluated according to a given demand under the time constraints (Sedeno-Noda and Gonzalez-Barrera, 2014;

Lin, 2003; 2009a; 2009b; 2010; 2011a; 2011b; El Khadiri and Yeh, 2016).

GA is a heuristic search method used in artificial intelligence and computing, it uses techniques inspired from evolutionary biology such as selection, mutation, inheritance and recombination to solve a problem. Classical GA was implemented to solve reliability optimization design problems and the idea behind reliability optimization is to find the best way to increase the system reliability (Coit David and Smith, 1994; Berna *et al*., 1995; 1997a; 1997b; Fulya *et al*., 1998; Berna and Abbas, 2001). Some other researchers extended the idea of GA to evaluate the system reliability of a stochastic-flow network such as Younes and Hassan (2011) and Yeh *et al*. (2018). Also, GA is used to solve multiple objective multi-state reliability optimization design problems Taboada *et al*. (2008). Coello and Christiansen (2000) used GA as a tool to solve multi-objective optimization problems in structures and Deb *et al*. (2002) developed a new algorithm based on a GA called non-dominated sorting GA II which leads to better convergence near the true Pareto-optimal front. Azaron *et al*. (2009) have used the GA in solving a discrete reliability optimization problem. In addition, Lin (2016) proposed a new approach incorporating GA with fuzzy control and neural network to evaluate network reliability with the number of network nodes increases exponentially.

The CAP subject to reliability and the total cost studied by Lin and Yeh (2010; 2011a; Yeh *et al*., 2018). The purpose was to maximize the reliability and minimize the total assignment cost. While, in Hassan (2015), the aim was to maximize the reliability and reduce the total lead-time of the assigned components.

While the CAP subject to system reliability, total assignment cost and total lead-time is never discussed in the previous literature. Therefore, this paper examines a multi-objective CAP problem for the case where each component has three attributes: An assignment cost, lead-time and system reliability. Such a problem is called OCAP. The main goal is to search the optimal components that maximize system reliability and minimize both the total assignment cost and the total lead-time. Furthermore, a multi-objective GA based on a Random Weighted GA (RWGA) is proposed to solve the presented problem.

The rest of this paper is organized as follows: Section 2 deals with the needed notations and section 3 presents the problem formulation. Next, Section 4 discusses the reliability evaluation under a time constraint. Section 5 explains the proposed multi-objective GA-based on RWGA. To demonstrate the usability of the proposed approach, several examples included in Section 6. Finally, Section 7 draws conclusions and future work.

Notations

n	Set of nodes.
m	$\{a_e \mid 1 \le e \le m\}$: Set of arcs.
MPs	Minimal paths.
np	Number of minimal paths.
mp_j	Minimal path no. j; $I = 1, 2, \ldots, np$.
cn	The number of available components.
cp_k	The components number k, $k = 1,2,\ldots,cn$.
$l(cp_k)$	Lead-time of components cp_k.
L_j	The lead-time of mp_j.
$c(cp_k)$	The cost of the component cp_k
$R_{d,T}$	The system reliability to the demand d under time limit T.
X	Capacity vector defined as $X = (x_1, x_2 \cdots, x_e)$
ps	Population size.
$Maxgen$	Maximum number of generations.
gn	Generation number.
p_m	GA mutation rate.
p_c	GA crossover rate.

Problem Formulation

Let $CP = \{cp_k \mid 1 \le k \le cn\}$ be a set of available components and $B = (b_1, b_2 \cdots, b_m)$ be the components assignment in which cp_k is assigned to the arc a_e if $b_e = k$. The total lead-time and the total cost associated with a specified components assignment B are, respectively, $S_l(B) = \sum_{e=1}^{m} l(b_e)$ and $C(B) = \sum_{e=1}^{m} c(b_e)$. Therefore, the mathematical programming formulation of the OCAP is given by:

$$Maximize\ R_{d,T}(B) \tag{1}$$

$$Minimize\ S_l(B) \tag{2}$$

$$Minimize\ C(B) \tag{3}$$

Subject to:

$$b_e = k, k \in \{1, 2 \cdots, cn\}\ for\ e = 1, 2, \cdots, m. \tag{4}$$

$$b_e \ne b_v \quad for\ e \ne v \tag{5}$$

$$L_j \le T, j = 1, 2, \cdots, np. \tag{6}$$

where, $L_j = \sum_{e=1}^{m} l(b_e)|_{b_e \in mp_j}$ be the total lead-time of a MP_j. Constraints (4) and (5) assert that each link should be given one component and each component can be assigned to at most one link. All feasible component assignments are generated using constraints (4) and (5). Constraint (6) states that the lead-time of the path MP_j (L_j) is less than the time limit (T) Hassan (2015).

DOI: 10.3844/jcssp.2019.108.117

Since, the multi-objective components assignments problem in the case of a maximal and minimal objective transformed into either a multi-objective minimization problem or a multi-objective maximization problem (Konak *et al*., 2006; Murata and Ishibuchi, 1995; Murata *et al*., 1996). Therefore, the original problem formulation can be modified to be of the minimal type:

$$Minimize\ S_1 = 1 - R_{d,T}(B) \quad (7)$$

$$Minimize\ S_2 = S_l(B) \quad (8)$$

$$Minimize\ S_3 = C(B) \quad (9)$$

Subject to:

$$b_e = k, k \in \{1, 2 \cdots, cn\}\ for\ e = 1, 2, \cdots, m. \quad (10)$$

$$b_e \neq b_v\ for\ e \neq v \quad (11)$$

$$L_j \leq T, j = 1, 2, \cdots, np. \quad (12)$$

Reliability Evaluation

Each component cp_k has a maximum capacity M^k and capacity values of cp_k vary from 0-M^k. Also, the lead-time of component cp_k is $l(cp_k)$ and the system reliability of the candidate chromosome evaluated as follows:

Step 1. Check if the candidate chromosome satisfies constraint (12).
Step 2. Use the procedure described by Lin, (2003), to generate the capacity vector X^j corresponding to the path mp_j.
Step 3. Calculate the network reliability of the chromosome:

$$R_{d,T} = \Pr\left\{\bigcup_{i=1}^{q}\left\{X \middle| X \geq X^i\right\}\right\}$$

using the inclusion-exclusion rule given by Xue (1985).

Proposed Approach Based on RWGA

In this section, we develop an approach to solve the multi-objective optimization problem based on RWGA, which is used to determine the highest ranking solution to the problem. The initial inputs include data related to the components such as the lead-time, assignment cost, probability, the parameters of RWGA and the network topology.

Crossover, Mutation and Selection Operations

We use the modified uniform crossover and mutation presented in Hassan (2015) to generate new offspring. The crossover operation is described as follows: Given two parents, the new offspring is filled randomly by selecting genes from them. This crossover process showed in Fig. 1. We note that swap mutation is used to avoid duplicated genes in a chromosome. The mutation process showed in Fig. 2.

Determine Fitness

Let $R_{d,T}(i)$, $C(B)$ and $S_l(i)$ be the corresponding values for the solution i, $i = 1, 2, \ldots, ps$:

Step 1. Find the normalized values of $R_{d,T}$, $C(B)$ and S_l as follows:

Step 1.1. Normalized value for $R_{d,T}(i)$:

$$NR_{d,T}(i) = \frac{R_{d,T}(i)}{Max\left(R_{d,T}(1), R_{d,T}(2), \cdots, R_{d,T}(PS)\right)}$$

Step 1.2. Normalized value for $S_l(i)$:

$$NS_l(i) = \frac{Min\left(S_l(1), S_l(2), \cdots, S_l(PS)\right)}{S_l(i)}.$$

Step 1.3. Normalized value for $C(i)$:

$$NC(i) = \frac{Min\left(C(1), C(2), \cdots, C(Ps)\right)}{C(I)}$$

Step 2. Calculate the Fitness value for each solution as follows:

Step 2.1. Generate a random number u_k in [0,1] for each objective k, k = 1, 2 and 3.
Step 2.2. Calculate the random weight of each objective k as $w_k = \frac{u_k}{\sum_{i=1}^{3} u_i}$.
Step 2.3. Calculate the fitness of the solution as

$$f(i) = w_1 * NR_{d,T}(i) + w_2 * NS_l(i) + w_3 NC(i)$$

Step 3. Calculate the selection probability of each solution:

$$P(i) = \frac{\left(f(i) - f^{\min}\right)}{\sum_{j \in ps}\left(f(i) - f^{min}\right)}$$

where, $f^{\min} = \min\{f(i), i \in ps\}$.

DOI: 10.3844/jcssp.2019.108.117

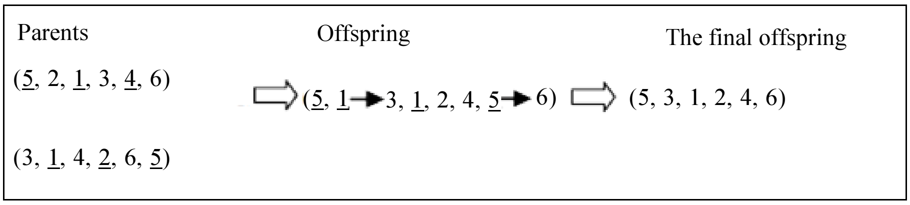

Fig. 1: Modified uniform crossover

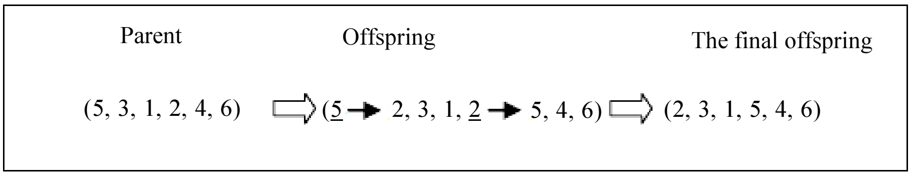

Fig. 2: Mutation process

The Entire Algorithm Based on RWGA

Step 1. Set the population size (*ps*), the crossover rate (*pc*), the mutation rate (*pm*) and the number of generations (*gn*).

Step 2. Generate the initial population including successful individuals B_1, B_1,··· B_{ps}.

Step 3. For each individual, evaluate the network unreliability S_1 = 1-$R_{d,T}(B)$, total lead time S_2 = $S_l(B)$ and the total assignment cost S_3 = $C(B)$

Step 4. Determine the set of dominated solutions E and the number of non-dominated solutions NE.

Step 5. Calculate the Fitness value and the selection probability for each individual B in the current population as presented in section 5.2.

Step 6. Select parents using the selection probabilities calculated in Step 5. Apply the GA operations (described in section 5.1) to generate new populations. Apply crossover on the selected parent pairs to create N offspring. Mutate offspring with a predefined mutation rate. Copy all offspring to Pt +1 and update E if necessary.

Step 7. Randomly remove NE solutions from the new population and add the same number of solutions from E to NE.

Step 8. If the stopping condition is not reached, set *gn* = *gn* +1 and go to Step 5. Otherwise, return to E.

After obtaining a Pareto set, change the network unreliability to be the network reliability for each Pareto solution.

Illustrative Examples

In this section, we demonstrate the effectiveness of our proposed approach using examples taken from literature (six nodes network and Taiwan Academic Network (TANET)). The genetic parameters used in the proposed multi objective GA are *ps* = 10, Maxgen = 100, P_c = 0.95 and P_m = 0.05. The algorithm was iterated 10 times. Also, N_E equals to 3, i.e., for each objective the algorithm search the best solution and stores it as a member of *E*.

Six-Node Network Example

The network has six nodes and 10 links (Fig. 3), as studied by Hassan (2015). The *mps* are as follows:

$$\begin{aligned}
&mp_1 = \{a_1, a_4, a_9\}, mp_2 = \{a_1, a_4, a_7, a_8\},\\
&mp_3 = \{a_1, a_5, a_8\}, mp_4 = \{a_1, a_5, a_7, a_9\},\\
&mp_5 = \{a_1, a_3, a_6, a_8\}, mp_6 = \{a_1, a_3, a_6, a_8\},\\
&mp_7 = \{a_2, a_3, a_8\}, mp_8 = \{a_2, a_6, a_7, a_9\},\\
&mp_9 = \{a_2, a_3, a_4, a_9\}, mp_{10} = \{a_2, a_3, a_4, a_7, a_8\},\\
&mp_{11} = \{a_2, a_3, a_5, a_8\} \textit{ and}\\
&mp_{12} = \{a_2, a_3, a_5, a_7, a_9\}.
\end{aligned}$$

Table 1 lists the 20 components and associated information. Table 2 lists the candidate solutions found by the proposed approach for this network given *d* = 6 and *T* = 7. While, Table 3 concisely lists the best candidate solutions for given different values for *d*, *T*. i.e., Maximum value for R_d, *C*(*B*) and minimum values for $S_l(B)$ and *C*(*B*).

The Tanet Example

The Taiwan Academic Network (TANET) with one source and one sink as shown in Fig. 4 has 6 *MPs* found by Chen and Lin (2012). The 6 paths are as follows:

$mp_1 = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, a_{11}, a_{12}, a_{13}\}$,

$mp_2 = \{a_1, a_2, a_{21}, a_{15}, a_{16}, a_{17}, a_{19}, a_{20}\}$,

$mp_3 = \{a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_{18}, a_{19}, a_{20}\}$,

$mp_4 = \{a_{14}, a_{15}, a_{16}, a_{17}, a_{19}, a_{20}\}$,

$mp_5 = \{a_{22}, a_{23}, a_{24}, a_{25}, a_{26}.a_{27}, a_{28}\}$ *and*

$mp_6 = \{a_{22}, a_{23}, a_{24}, a_{25}, a_{26}, a_{27}, a_{29}, a_{30}\}$.

Table 4 lists 80 available components, with capacities and probabilities taken from Chen and Lin, (2012); the corresponding lead-time for each component is randomly assigned in this article. Table 5 shows the results of applying the proposed approach to the TANET example.

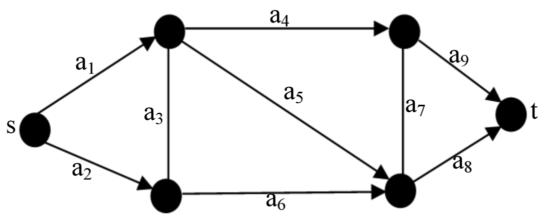

Fig. 3: The six-nodes network example

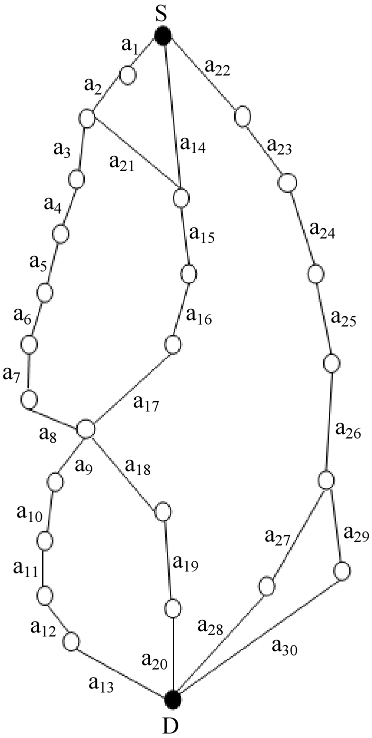

Fig. 4: TANET example

Table 1: Arc capacity, probability, lead-time and cost for the 20 available components

	Capacity								
cp_k	0	1	2	3	4	5	6	$l(cp_k)$	$c(cp_k)$
1	0.01	0.00	0.01	0.00	0.01	0.00	0.97	2	10
2	0.05	0.05	0.05	0.15	0.20	0.50	0.00	3	60
3	0.07	0.08	0.00	0.85	0.00	0.00	0.00	2	10
4	0.70	0.00	0.00	0.00	0.00	0.30	0.00	2	20
5	0.01	0.00	0.00	0.05	0.00	0.00	0.94	1	50
6	0.01	0.00	0.00	0.01	0.00	0.00	0.98	3	60
7	0.50	0.50	0.00	0.00	0.00	0.00	0.00	3	20
8	0.25	0.25	0.50	0.00	0.00	0.00	0.00	1	50
9	0.15	0.25	0.10	0.10	0.10	0.10	0.20	2	80
10	0.00	0.05	0.05	0.90	0.00	0.00	0.00	2	100
11	0.01	0.99	0.00	0.00	0.00	0.00	0.00	1	70
12	0.02	0.00	0.05	0.00	0.05	0.00	0.88	1	60
13	0.07	0.00	0.28	0.00	0.00	0.65	0.00	3	10
14	0.05	0.05	0.90	0.00	0.00	0.00	0.00	2	20
15	0.60	0.40	0.00	0.00	0.00	0.00	0.00	2	50
16	0.15	0.00	0.00	0.00	0.85	0.00	0.00	1	60
17	0.10	0.10	0.10	0.70	0.00	0.00	0.00	1	20
18	0.70	0.00	0.00	0.00	0.00	0.30	0.00	3	50
19	0.07	0.18	0.75	0.00	0.00	0.00	0.00	2	80
20	0.40	0.40	0.20	0.00	0.00	0.00	0.00	3	100

Table 2: The candidate solutions to the network of Fig. 3

$R_{6,7}(B)$	$S_l(B)$	$C(B)$	Assigned components									
0.727650	15	510	15	5	11	9	1	12	17	16	8	18
0.824670	15	500	15	5	11	9	17	12	8	16	1	18
0.782595	15	510	5	12	17	19	11	16	8	6	15	1
0.816849	15	510	5	15	17	19	11	16	6	8	12	1
0.738374	16	530	9	5	11	15	1	19	17	16	6	8
0.973036	15	510	5	11	17	12	8	19	15	16	1	6
0.947129	15	510	5	11	17	6	8	19	12	16	1	15
0.966886	15	510	5	15	17	19	1	16	8	6	12	11
0.922435	15	510	5	15	11	19	17	16	8	6	12	1

Table 3: The best candidate solutions for different values for d,T.

d,T	$R_{d,T}(B)$	$S_l(B)$	$C(B)$	Assigned components
6,7	0.973036	15	510	5 11 17 12 8 19 15 16 1 6
6, 8	0.987345	14	520	5 16 11 19 14 1 8 12 17 10
6, 9	0.985979	19	540	1 19 18 17 2 10 12 5 8 6

Table 4: Component information

	Capacity						
cp_k	0	1	2	3	4	$l(cp_k)$	$c(cp_k)$
1	0.0004	0.0392	0.9604	0	0	1	100
2	0.000512	0.017664	0.203136	0.778688	0	1	50
3	0.000343	0.013671	0.181629	0.804357	0	1	65
4	0.015	0.985	0	0	0	2	80
5	0.0016	0.0768	0.9216	0	0	2	70
6	0.005929	0	0.142142	0	0.851929	1	135
7	0.003	0	0.997	0	0	2	60
8	0.007225	0	0.15555	0	0.837225	1	35
9	0.005929	0	0.142142	0	0.851929	1	35
10	0.003	0.997	0	0	0	2	80
11	0.034	0.966	0	0	0	2	55
12	0.0036	0.1128	0.8836	0	0	3	40
13	0.000001	0.000297	0.029403	0.970299	0	2	110
14	0.000784	0.054432	0.944784	0	0	1	65
15	0.000225	0.02955	0.970225	0	0	1	70
16	0.095	0.905	0	0	0	3	15
17	0.005776	0.140448	0.853776	0	0	3	35
18	0.000625	0.04875	0.950625	0	0	2	75
19	0.000729	0.022113	0.223587	0.753571	0	1	40
20	0.001	0.027	0.243	0.729	0	2	35
21	0.000512	0.017664	0.203136	0.778688	0	1	45
22	0.004225	0.12155	0.874225	0	0	3	30
23	0.005929	0	0.142142	0	0.851929	1	85
24	0.003	0	0.997	0	0	2	70
25	0.000216	0.010152	0.159048	0.830584	0	3	55
26	0.034	0.966	0	0	0	2	30
27	0.000512	0.017664	0.203136	0.778688	0	2	55
28	0.000343	0.013671	0.181629	0.804357	0	1	60
29	0.001	0.027	0.243	0.729	0	3	35
30	0.0009	0.0582	0.9409	0	0	2	85
31	0.002809	0.100382	0.896809	0	0	1	60
32	0.000166375	0.008575875	0.147349125	0.843908625	0	2	70
33	0.000125	0.007125	0.135375	0.857375	0	2	80
34	0.0001	0.0198	0.9801	0	0	1	140
35	0.025	0.975	0	0	0	3	10
36	0.024	0.976	0	0	0	3	60
37	0.000125	0.007125	0.135375	0.857375	0	2	75
38	0.000110592	0.006580224	0.130507776	0.862801408	0	1	85
39	0.0001	0	0.0198	0	0.9801	1	100
40	0.001849	0	0.082302	0	0.915849	3	60
41	0.001024	0.061952	0.937024	0	0	2	60
42	0.000676	0.050648	0.948676	0	0	2	65
43	0.007921	0.162158	0.829921	0	0	4	35
44	0.000512	0.017664	0.203136	0.778688	0	2	25
45	0.001	0.027	0.243	0.729	0	5	20
46	0.097	0	0.903	0	0	4	40
47	0.000001	0.000297	0.029403	0.970299	0	3	135
48	0.022	0.978	0	0	0	2	70
49	0.000256	0	0.031488	0	0.968256	1	145
50	0.001225	0	0.06755	0	0.931225	1	70
51	0.025	0.975	0	0	0	3	70
52	0.000274625	0.011851125	0.170473875	0.817400375	0	2	65
53	0.000529	0	0.044942	0	0.954529	3	120

Table 4: Continue

54	0.000144	0	0.023712	0	0.976144	1	110
55	0.000216	0.010152	0.159048	0.830584	0	2	70
56	0.000117649	0.006850053	0.132946947	0.860085351	0	1	60
57	0.046	0	0.954	0	0	2	50
58	0.083	0	0.917	0	0	3	40
59	0.000015625	0.001828125	0.071296875	0.926859375	0	3	105
60	0.000274625	0.011851125	0.170473875	0.817400375	0	2	60
61	0.001369	0.071262	0.927369	0	0	2	85
62	0.000001	0.000297	0.029403	0.970299	0	2	125
63	0.000512	0.017664	0.203136	0.778688	0	3	50
64	0.006084	0.143832	0.850084	0	0	2	40
65	0.004096	0.119808	0.876096	0	0	5	45
66	0.003481	0.111038	0.885481	0	0	4	50
67	0.035	0.965	0	0	0	2	60
68	0.022	0	0.978	0	0	3	70
69	0.000166375	0.008575875	0.147349125	0.843908625	0	3	85
70	0.000042875	0.003546375	0.097778625	0.898632125	0	3	95
71	0.000024389	0.002449833	0.082027167	0.915498611	0	2	100
72	0.000324	0	0.035352	0	0.964324	1	95
73	0.000000343	0.000145971	0.020707029	0.979146657	0	2	145
74	0.004356	0.123288	0.872356	0	0	3	30
75	0.055	0.945	0	0	0	2	15
76	0.001936	0.084128	0.913936	0	0	5	55
77	0.000035937	0.003159189	0.092573811	0.904231063	0	4	85
78	0.000484	0	0.043032	0	0.956484	2	115
79	0.000121	0	0.021758	0	0.978121	1	100
80	0.001	0.999	0	0	0	2	100

Table 5: The results of applying the proposed approach to the network given in Fig. 4

d,T	$R_{d,T}(B)$	$S_l(B)$	$C(B)$	Assigned components																	
4,16	0.9999745	66	1735	29	4	5	69	13	32	56	50	47	21	68	28	24	38	76			
				34	12	58	78	43	1	62	72	19	52	79	17	33	71	8	31		
6,16	0.999986	61	1435	17	56	39	47	67	62	5	69	16	55	36	48	8	79	49			
				6	44	50	31	64	43	7	51	73	802	52	14	60	23	32	15	9	
8,18	0.999172	61	1825	41	11	39	12	2	49	42	19	31	53	14	78	38	9	72	21		
				43	30	73	54	70	6	40	37	20	23	28	5	46	77	18	79		
9,18	0.985317	61	1825	41	11	39	54	2	49	12	19	31	53	14	78	38	9	72			
				21	70	42	6	40	37	20	23	28	5	46	77	18	79	26	43	30	73

Discussion and Comparison

This study is presented and solved the optimal components assignment in the case of coexistence of three conflicting objectives: System reliability, Cost and Total lead-time. Previous studies focused on studying two objectives such as Lin and Yeh, (2013); solved the CAP to maximize the system reliability under cost constraint. While in Hassan (2015) the CAP is studied subject to reliability and lead-time constraints. So, to assert the efficiency of our algorithm we will apply it to the CAP with two objectives; reliability and lead time. Accordingly, the optimization problem is formulated in the following minimization form:

$$Minimize\ \ S_1 = 1 - R_{d,T}\left(B\right) \tag{13}$$

$$Minimize\ \ S_2 = S_l\left(B\right) \tag{14}$$

Subject to:

$$b_e = k, k \in \{1, 2, \cdots, cn\}\ \ for\ e = 1, 2, \cdots, m \tag{15}$$

$$b_e \neq b_v\ \ for\ e \neq v \tag{16}$$

$$L_j \leq T, j = 1, 2, \cdots, np \tag{17}$$

Table 7 and 8 show the results of applying the proposed algorithm in comparison with Hassan (2015). The components information for network given in Fig. 3 and 4 are shown in Table 1 and 4 respectively. While the components information for the network of Fig. 5 is given in Table 6, Hassan (2015).

From the results shown in Table 7 it is observed that the proposed approach leads to a very good values except in the case where d = 4 and T = 9 where the minimum of a total lead-time is slightly greater than the one obtained by Hassan (2015), in addition the proposed approach gave good results in the case d = 4 and T = 8, whereas Hassan's method could not get any result.

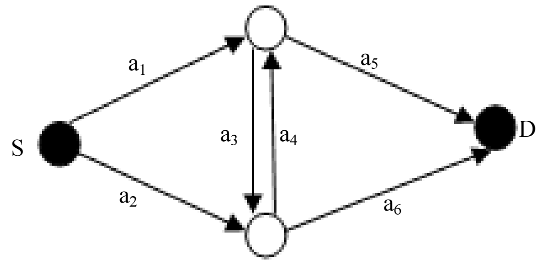

Fig. 5: Computer network with 4 nodes and 6 arcs

Table 6: Components capacities, probabilities and lead-time for the network of Fig. 5

	Capacity					
cp_k	0	1	2	3	4	$l(cp_k)$
1	0.05	0.05	0.10	0.80	0.00	2
2	0.05	0.05	0.10	0.80	0.00	1
3	0.05	0.10	0.85	0.00	0.00	3
4	0.10	0.90	0.00	0.00	0.00	3
5	0.10	0.90	0.00	0.00	0.00	1
6	0.05	0.05	0.10	0.20	0.60	2
7	0.05	0.10	0.10	0.10	0.55	2
8	0.05	0.05	0.10	0.80	0.00	1

Table 7: Comparison results for the Four-node network example

Hassan's approach					Proposed approach		
d,T	$*S^0_l$	Assigned components	$R_{d,T}$	S_l	Assigned components	$R_{d,T}$	S_l
4,6	9	2 6 7 5 8 1	0.983875	9	8 6 7 5 2 1	0.983875	9
4,7		2 7 5 1 6 8	0.993725	9	2 6 3 5 7 8	0.992316	9
4,8		-	-	-	7 6 1 5 8 2	0.994246	9
4,9		6 2 7 5 1 8	0.994668	9	1 2 7 6 3 8	0.994358	11

Table 8: Comparison results for the *Six-node network example*

Hassan's approach					Proposed approach		
d,T	S^0_l	Assigned components	$R_{d,T}$	S_l	Assigned components	$R_{d,T}$	S_l
6,7	14	16 5 11 15 10 19 17 8 12	0.981486	12	5 17 11 14 10 15 8 12 16	0.989789	12
6,8		4 15 5 1 19 3 16 11 12	0.989187	13	5 8 11 1 16 10 17 3 12	0.990176	12
6,9	15	8 5 12 2 10 19 11 14 13	0.989316	15	1 14 16 11 6 12 17 5 8	0.992350	13
8,9		1 14 17 11 19 12 3 5 8	0.979129	13	1 12 11 10 7 6 16 17 5	0.993952	15

Comparing the results obtained by the proposed approach to those found by the algorithm of Hassan (2015), it is observed that the values of system reliability found by the proposed approach are better than those obtained by Hassan (2015) and also the values of minimum lead-time are better than those obtained in Hassan (2015). Therefore, the proposed approach obtains better optimal solutions.

Conclusion

This paper explored how to determine the optimal assignment component for an SFN with maximal system reliability under a minimum of both assignment cost and total lead-time. A multi-objective components assignments problem subject to system reliability, assignment cost and total lead-time is presented and formulated as a multi-objective minimization problem. Furthermore, a multi-objective GA-based on RWGA approach is proposed to solve the presented problem. Using the proposed approach, the optimal solution is obtained i.e., the system reliability is maximized and both the total lead-time and the assignment cost are minimized. Future work may consider the OCAP for an SFN with multiple sources subject to multiple constraints.

Many real-life systems, such as computer network systems, manufacturing systems, electrical power systems and logistics systems are modelled as SFN to evaluate the reliability taking into account one or more constraints. However, this study presented a solution approach to maximize the reliability subject to assignment cost and total lead-time constraints and helping the decision makers to determine the optimal solution.

Acknowledgement

The authors have no support or funding to report. The authors thank the anonymous referees for their careful readings and provisions of helpful suggestions to improve the presentation.

Author's Contributions

All authors are equally contributed in this work and the article.

Ethics

Authors confirm that this manuscript has not been published elsewhere and that no ethical issues are involved.

References

Azaron, A., C. Perkgoz, H. Katagiri, K. Kato and M. Sakawa, 2009. Multi-objective reliability optimization for dissimilar-unit cold-stand by systems using a genetic algorithm. Comput. Operat. Res., 36: 1562-1571. DOI:10.1016/j.cor.2008.02.017

Berna, D. and C. Abbas, 2001. A simulated annealing algorithm for design of computer communication networks. Proceedings of the World Multiconference on Cybernetics and Informatics, (SCI' 01).

Berna, D., A. Fulya and A.E. Smith, 1995. A genetic algorithm approach to optimal topological design of all terminal networks. Intell. Eng. Syst. Through Artificial Neural Netw., 5: 405-410.

Berna, D., A. Fulya and A.E. Smith, 1997a. Genetic algorithms design of networks considering all-terminal reliability. Proceedings of the 6th Industrial Engineering Research Conference Proceedings, May 17-18, Miami, Beach, Florida, USA, pp: 30-35.

Berna, D., A. Fulya and A.E. Smith, 1997b. Local search genetic algorithm for optimization of highly reliable communications networks. IEEE Trans. Evolut. Comput., 1: 179-188. DOI: 10.1109/4235.661548

Chen, G.H. and Y.C. Hung, 1993. On the quickest path problem. Inform. Process. Lett., 46: 125-128. DOI: 10.1016/0020-0190(93)90057-G

Chen, S.G. and Y.K. Lin, 2012. Search for all minimal paths in a general large flow network. IEEE Trans. Reliability, 61: 949-956. DOI: 10.1109/TR.2012.2220897

Chen, S.G., 2014. Optimal double-resource assignment for the robust design problem in multistate computer networks. Applied Math. Model., 38: 263-277. DOI: 10.1016/j.apm.2013.06.020

Chen, Y.L. and Y.H. Chin, 1990. The quickest path problem. Comput. Operat. Res., 17: 153-161. DOI: 10.1016/0305-0548(90)90039-A

Chu, P.C. and J.E. Beasley, 1997. Genetic algorithm for generalized assignment problem. Comput. Operat. Res., 24: 17-23. DOI: 10.1016/S0305-0548(96)00032-9

Coello, C.A. and A.D. Christiansen, 2000. Multi objective optimization of trusses using genetic algorithms. Comput. Struct., 75: 647-660. DOI: 10.1016/S0045-7949(99)00110-8

Coit David, W. and A.E. Smith, 1994. Use of a genetic algorithm to optimize a combinatorial reliability design problem. Proceeding of the 3rd IIE Research Conference, (RC' 94), pp: 467-472.

de Queiros, E., V. Martins and J.L.E. dos Santos, 1997. An algorithm for the quickest path problem. Operat. Res. Lett., 20: 195-198. DOI: 10.1016/S0167-6377(97)00008-4

Deb, K., A. Pratap, S. Agarwal and T. Meyarivan, 2002. A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans. Evolut. Comput., 6: 182-198. DOI: 10.1109/4235.996017

El Khadiri, M. and W.C. Yeh, 2016. An efficient alternative to the exact evaluation of the quickest path flow network reliability problem. Comput. Operat. Res., 76: 22-32. DOI: 10.1016/j.cor.2016.06.010

Fulya, A., D. Berna and A.E. Smith, 1998. Reliability optimization of computer communication networks using genetic algorithms. Proceedings of the IEEE International Conference on Systems, Man and Cybernetics-Intelligent Systems For Humans In A Cyberworld, Oct. 11-14, Hyatt Regency La Jolla, San Diego, California, USA, pp: 4676-4681. DOI: 10.1109/ICSMC.1998.727590

Harper, P.R., V. de Senna, I.T. Vieira and A.K. Shahani, 2005. A genetic algorithm for the project assignment problem. Comput. Operat. Res., 32: 1255-1265. DOI: 10.1016/j.cor.2003.11.003

Hassan, M.R., 2015. Solving a component assignment problem for a stochastic-flow network under lead-time constraint. Ind. J. Sci. Technol., 8: 1-5. DOI: 10.17485/ijst/2015/v8i35/70455

Konak, A., W.C. David and A.E. Smith, 2006. Multi-objective optimization using genetic algorithms: A tutorial. Reliability Eng. Syst. Safety, 91: 992-1007. DOI: 10.1016/j.ress.2005.11.018

Lin, J.S., C.C. Jane and J. Yuan, 1995. On reliability evaluation of a capacitated-flow network in terms of minimal path sets. Networks, 25: 131-138. DOI: 10.1002/net.3230250306

Lin, Y.K. and C.T. Yeh, 2011a. Maximizing network reliability for stochastic transportation networks under a budget constraint by using a genetic algorithm. Int. J. Innovative Comput. Inform. Control, 7: 7033-7050.

Lin, Y.K. and C.T. Yeh, 2011b. Multistate components assignment problem with optimal network reliability subject to assignment budget. Applied Math. Comput., 217: 10074-10086. DOI: 10.1016/j.amc.2011.05.001

Lin, Y.K. and C.T. Yeh, 2015. System reliability maximization for a computer network by finding the optimal two-class allocation subject to budget. Applied Soft Comput., 36: 578-588. DOI: 10.1016/j.asoc.2015.07.037

Lin, Y.K., 2001. A simple algorithm for reliability evaluation of a stochastic-flow network with node Failure. Comput. Operat. Res., 28: 1277-1285. DOI: 10.1016/S0305-0548(00)00039-3

Lin, Y.K., 2002. Two-commodity reliability evaluation for a stochastic-flow network with node failure. Comput. Operat. Res., 29: 1927-1939. DOI: 10.1016/S0305-0548(01)00069-7

Lin, Y.K., 2003. Extend the quickest path problem to the system reliability evaluation for a stochastic-flow network. Comput. Operat. Res., 30: 567-575. DOI: 10.1016/S0305-0548(02)00025-4

Lin, Y.K., 2009a. Time version of the shortest path problem in a stochastic-flow network. J. Comput. Applied Math., 228: 150-157. DOI: 10.1016/j.cam.2008.09.003

Lin, Y.K., 2009b. Routing policy of stochastic-flow networks under time threshold and budget constraint. Expert Syst. Applic., 36: 6076-6081. DOI: 10.1016/j.eswa.2008.07.080

Lin, Y.K., 2010. Reliability of k separate minimal paths under both time and budget constraints. IEEE Trans. Reliability, 59: 183-190. DOI: 10.1109/TR.2009.2035794

Lin, Y.K., 2011a. Network reliability of a time-based multistate network under spare routing with ρ minimal paths. IEEE Trans. Reliability, 60: 61-69. DOI: 10.1109/TR.2010.2103594

Lin, Y.K., 2011b. Stochastic flow networks via multiple paths under time threshold and budget constraint. Comput. Math. Applic., 62: 2629-2638. DOI: 10.1016/j.camwa.2011.08.002

Lin, Y.K. and C.T. Yeh, 2010. Evaluation of optimal network reliability under components assignments subject to a transmission Budget. Trans. Reliability, 59: 539-550. DOI: 10.1109/TR.2010.2055920

Lin, Y.K. and C.T. Yeh, 2013. A two-stage approach for a multi-objective component assignment problem for a stochastic-flow network. Eng. Optimiz., 45: 265-285. DOI: 10.1080/0305215X.2012.669381

Liu, L., 2016. Optimization design of computer network reliability based on genetic algorithms. Chem. Eng. Trans., 51: 775-780. DOI: 10.3303/CET1651130.

Murata, T. and H. Ishibuchi, 1995. MOGA: Multi-objective genetic algorithms. Proceedings of the IEEE International Conference on Evolutionary Computation, Nov. 29-Dec. 1, IEEE Xplore Press, Perth, WA, Australia: pp: 289-289. DOI: 10.1109/ICEC.1995.489161

Murata, T., H. Ishibuchi and H. Tanaka, 1996. Multi-objective genetic algorithm and its applications to flow shop scheduling. Comput. Indust. Eng., 30: 957-968. DOI: 10.1016/0360-8352(96)00045-9

Sedeno-Noda, A. and J.D. Gonzalez-Barrera, 2014. Fast and fine quickest path algorithm. Eur. J. Operat. Resources, 238: 596-606. DOI: 10.1016/j.ejor.2014.04.028

Taboada, H.A., J.F. Espiritu and D.W. Coit, 2008. MOMS-GA: A Multi-Objective Multi-State Genetic Algorithm for system reliability optimization design problems. IEEE Trans. Reliability, 57: 182-191. DOI: 10.1109/TR.2008.916874

Wang, Y.Z., 2002. An application of genetic algorithm methods for teacher assignment problems. Expert Syst. Applic., 22: 295-302. DOI: 10.1016/S0957-4174(02)00017-9

Xue, J., 1985. On multistate system analysis. IEEE Trans. Reliability, 34: 329-337. DOI: 10.1109/TR.1985.5222178

Yeh, C.T., Y.K. Lin and J.Y. Yang, 2018. Network reliability maximization for stochastic-flow network subject to correlated failures using genetic algorithm and tabu search. Eng. Optimiz., 50: 1212-1231.

Younes, A. and M.R. Hassan, 2011. A genetic algorithm for reliability evaluation of a stochastic-flow network with node failure. Int. J. Comput. Sci. Security, 4: 528-528.